

Computer Systems Technology

U.S. DEPARTMENT OF
COMMERCE
Technology Administration
National Institute of
Standards and
Technology

Stable Implementation Agreements for Open Systems Interconnection Protocols Version 6 Edition 1 December 1992

Based on the Proceedings of the OSE Implementors' Workshop (OIW)

NIST

NAT'L INST. OF STAND. & TECH. R.L.C.



A11104 122846

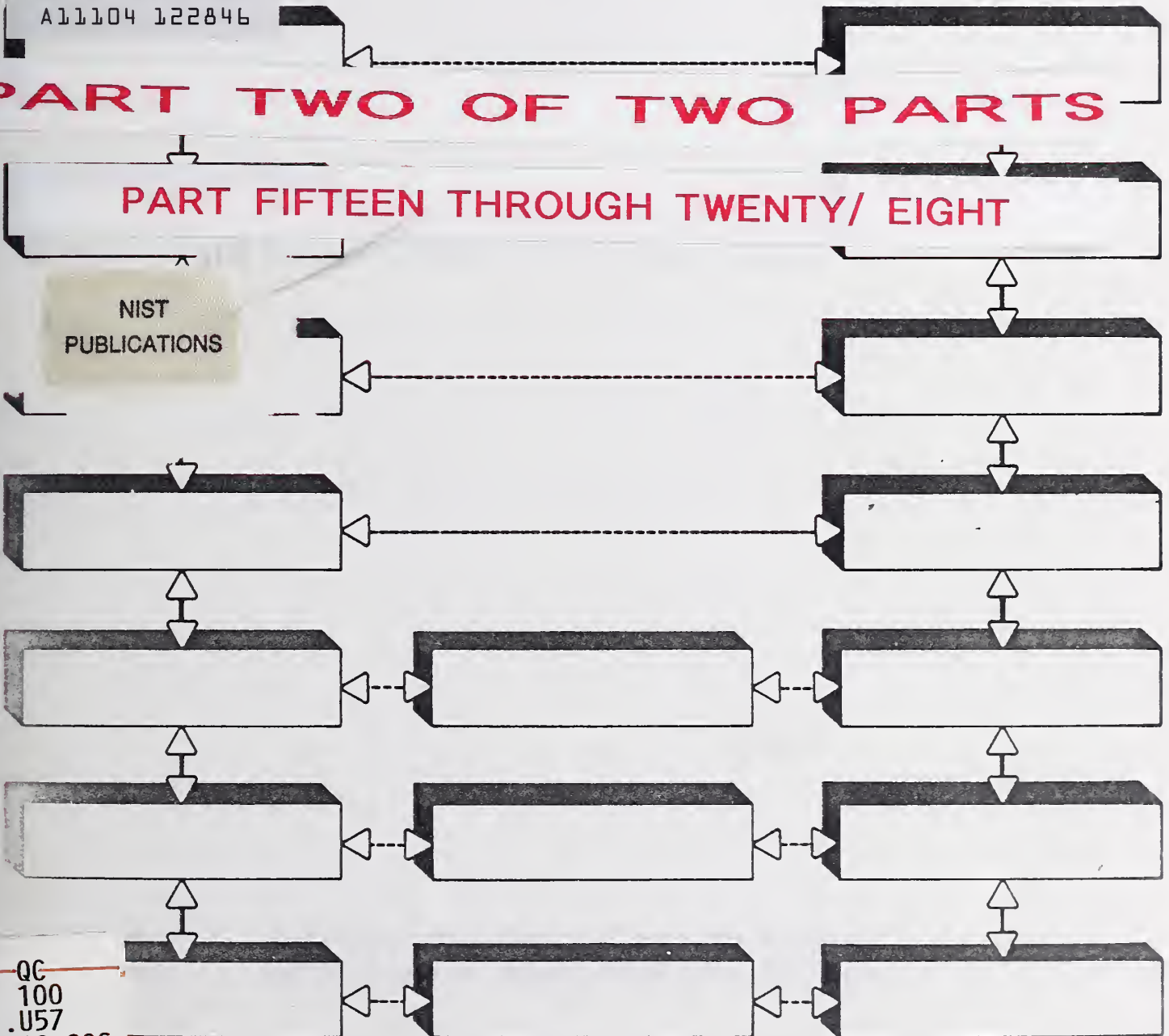
Workshop Chairman/Technical Editor
Tim Boland, NIST

Workshop Editor
Brenda Gray

PART TWO OF TWO PARTS

PART FIFTEEN THROUGH TWENTY/ EIGHT

NIST
PUBLICATIONS



QC
100
U57
500-206
Pt. 2
1993
C. 2

Stable Implementation Agreements for Open Systems Interconnection Protocols: Part 15 - Transaction Processing

Output from the December 1992 Open Systems
Environment Implementors' Workshop (OIW)

SIG Chair: Jeff Hildebrand, Boeing Computer Services
SIG Editor: Jeff Hildebrand, Boeing Computer Services

Table of Contents

Part 1

1. SCOPE	1
2. NORMATIVE REFERENCES	1
3. DEFINITIONS AND ABBREVIATIONS	3
4. NOTATION	3
5. TAXONOMY STRUCTURE	5
5.1 GUIDELINES FOR SPLITTING UP THE PROFILES	5
5.2 TRANSACTION PROCESSING PROFILES TREE	5
6. CONFORMANCE	6
1. SCOPE	2
2. NORMATIVE REFERENCES	2
3. DEFINITIONS AND ABBREVIATIONS	2
4. NOTATION	2
5. SUPPORT OF OSI TP PROTOCOL	2
6. CONFORMANCE	2
ANNEX A: SUPPORT OF THE OSI TP PROTOCOL (NORMATIVE)	3
A.1. IDENTIFICATION	3
A.2. CLAIMED CONFORMANCE TO STANDARDS	3
A.2.1. ISO/IEC 10026-3	3
A.2.1.1. VERSION NUMBER(S)	3
A.2.1.2. GLOBAL CONFORMANCE CLAIM	3
A.2.2. ISO/IEC 10026 AMENDMENTS	3
A.2.3. ISO/IEC 10026 TECHNICAL CORRIGENDA	3
A.2.4. CONFORMANCE CLASS(ES) SUPPORTED	4
A.3. FUNCTIONAL UNITS, LIMITS AND PROTOCOL MECHANISMS	5
A.3.1. SUPPORT OF FUNCTIONAL UNITS	5

A.3.2. PROTOCOL MECHANISMS IMPLEMENTED	5
A.3.2.1. CONCATENATION/SEPARATION.....	5
A.3.2.2. ASSOCIATION ESTABLISHMENT	6
A.3.2.3. SUPPORT FOR MANDATORY AND OPTIONAL BIDDING	7
A.3.2.4. CONTENTION	7
A.3.2.5. BID MECHANISM	8
A.3.2.6. DIALOGUE ESTABLISHMENT	8
A.3.2.7. TRANSACTION BRANCH ESTABLISHMENT.....	9
A.3.2.8. ROLES IN A TRANSACTION TREE SUPPORTED.....	10
A.3.2.9. SUPPORT FOR RECOVERY.....	10
A.4. TP PROTOCOL - GENERAL	11
A.5. TP PROTOCOL - SUPPORT OF THE DIALOGUE FUNCTIONAL UNIT.....	11
A.5.1. DIALOGUE FU APDUS.....	11
A.5.2. TP-BEGIN-DIALOGUE-RI APDU	12
A.5.2.1. DETAIL OF THE DIALOGUE FIELD OF TP-BEGIN-DIALOGUE-RI APDU	12
A.5.2.1.1. DETAIL OF TPSU-TITLE FIELDS FOR THE DIALOGUE FIELD OF TP-BEGIN-DIALOGUE-RI APDU	13
A.5.3. TP-BEGIN-DIALOGUE-RC APDU.....	13
A.5.3.1. DETAIL OF THE DIALOGUE FIELD OF TP-BEGIN-DIALOGUE-RC APDU.....	13
A.5.4. TP-END-DIALOGUE-RI APDU.....	14
A.5.5. TP-ABORT-RI APDU	14
A.5.5.1. DETAIL OF THE USER FIELD OF TP-ABORT-RI APDU.....	14
A.5.5.2. DETAIL OF THE PROVIDER FIELD OF TP-ABORT-RI APDU	14
A.5.5.3. TP-BID-RI APDU.....	15
A.5.6. TP-BID-RC APDU	15
A.5.7. TP-INITIALIZE-RI APDU	15
A.5.8. TP-INITIALIZE-RC APDU	15

A.6. SUPPORT OF THE SHARED CONTROL FUNCTIONAL UNIT	16
A.6.1. SHARED CONTROL FUNCTIONAL UNIT APDUS	16
A.7. SUPPORT OF THE POLARIZED CONTROL FUNCTIONAL UNIT	17
A.7.1. POLARIZED CONTROL FUNCTIONAL UNIT APDUS	17
A.8. SUPPORT OF THE HANDSHAKE FUNCTIONAL UNIT	18
A.8.1. HANDSHAKE FUNCTIONAL UNIT APDUS	18
A.8.2. TP-HANDSHAKE-RI APDU	18
A.8.3. TP-HANDSHAKE-AND-GRANT-CONTROL-RI APDU	19
A.9. TP PROTOCOL - SUPPORT OF THE COMMIT FUNCTIONAL UNIT	19
A.9.1. COMMIT FUNCTIONAL UNIT APDUS	19
A.9.1.1. TP-PREPARE-RI APDU	20
A.9.2. TP-DEFER-RI APDU	20
A.9.3. TP-HEURISTIC-REPORT-RI APDU	20
A.9.4. TP-TOKEN-GIVE-RI APDU	21
A.10. TP PROTOCOL - SUPPORT OF THE RECOVERY FUNCTIONAL UNIT	22
A.10.1. RECOVERY FUNCTIONAL UNIT APDUS	22
A.10.2. TP-BEGIN-DIALOGUE-RI APDU	23
A.10.2.1. DETAIL OF THE CHANNEL FIELD OF TP-BEGIN-DIALOGUE-RI APDU	23
A.10.3. TP-BEGIN-DIALOGUE-RC APDU	24
A.10.3.1. DETAIL OF THE CHANNEL FIELD OF TP-BEGIN-DIALOGUE-RC APDU	24
A.10.4. TP-END-DIALOGUE-RI APDU	24
A.10.5. TP-BID-RI APDU	24
A.10.6. TP-RECOVER-RI APDU	25
A.10.7. TP-TOKEN-GIVE-RI APDU	25

Part 3

1. SCOPE	2
----------------	---

2. NORMATIVE REFERENCES	2
3. DEFINITIONS AND ABBREVIATIONS	2
4. NOTATION	2
5. SUPPORT OF CCR APDUS	2
6. CONFORMANCE	2
A.1 DATE OF STATEMENT	3
A.2 IMPLEMENTATION DETAILS	3
A.3 ISO/IEC 9805-1	3
A.4 AMENDMENTS IMPLEMENTED	3
A.5 TECHNICAL CORRIGENDA IMPLEMENTED	3
A.6 GLOBAL STATEMENT OF CONFORMANCE	3
A.6.1 MANDATORY FEATURES IMPLEMENTED	3
A.7 INITIATOR/RESPONDER CAPABILITIES	4
A.7.1 ATOMIC-ACTION-BRANCH ESTABLISHMENT	4
A.7.2 SUPPORT FOR THE CONCATENATION MECHANISM	4
A.7.3 OTHER IMPLEMENTATION CAPABILITIES	4
A.8 CCR PROTOCOL - GENERAL	5
A.9 CCR PROTOCOL	5
A.9.1 CCR PDUS	5
A.9.2 C-BEGIN-RI	5
9.2.1. ATOMIC-ACTION IDENTIFIER	6
A.9.3 C-BEGIN-RC	6
A.9.4 C-PREPARE-RI	6
A.9.5 C-READY-RI	7
A.9.6 C-COMMIT-RI	7
A.9.7 C-COMMIT-RC	7
A.9.8 C-ROLLBACK-RI	7

A.9.9 C-ROLLBACK-RC	8
A.9.10 C-RECOVER-RI	8
A.9.10.1. ATOMIC-ACTION IDENTIFIER	9
A.9.10.2. BRANCH IDENTIFIER	10
A.9.10.3. RECOVERY STATE	11
A.9.11 C-RECOVER-RC	11
A.9.11.1. ATOMIC-ACTION IDENTIFIER	12
A.9.11.2. BRANCH IDENTIFIER	13
A.9.11.3. RECOVERY STATE	13

Part 4

1. SCOPE	1
2. NORMATIVE REFERENCES	1
3. DEFINITIONS AND ABBREVIATIONS	1
4. NOTATION	1
5. SUPPORT OF SESSION SPDUS	1
6. SUPPORT OF PRESENTATION PPDUS	1
7. SUPPORT OF ACSE APDUS	1
8. CONFORMANCE	1
ANNEX A: SESSION PROTOCOL PDUS (NORMATIVE)	3
A.1 SUPPORTED FUNCTIONS	3
A.2 ISO 8327 PROTOCOL VERSIONS IMPLEMENTED	3
A.3 PROTOCOL MECHANISMS	4
A.4 INITIATOR/RESPONDER CAPABILITIES	5
A.5 SESSION PROCEDURES USAGE BY PROFILE	5
A.6 SUPPORTED PARAMETERS OF SESSION PDUS	11
A.6.1 CONNECT (CN) SPDU	11

A.6.2 OVERFLOW ACCEPT (OA) SPDU	12
A.6.3 CONNECT DATA OVERFLOW (CDO) SPDU	12
A.6.4 ACCEPT (AC) SPDU	13
A.6.5 REFUSE (RF) SPDU.....	14
A.6.6 FINISH (FN) SPDU	14
A.6.7 DISCONNECT (DN) SPDU	15
A.6.8 NOT FINISHED (NF) SPDU	15
A.6.9 ABORT (AB) SPDU.....	15
A.6.10 ABORT ACCEPT (AA) SPDU.....	15
A.6.11 DATA TRANSFER (DT) SPDU.....	15
A.6.12 EXPEDITED DATA (EX) SPDU.....	15
A.6.13 TYPED DATA (TD) SPDU.....	16
A.6.14 CAPABILITY DATA (CD) SPDU	16
A.6.15 CAPABILITY DATA ACK (CDA) SPDU	16
A.6.16 GIVE TOKENS (GT) SPDU.....	17
A.6.17 PLEASE TOKENS (PT) SPDU	17
A.6.18 MINOR SYNC POINT (MIP) SPDU	18
A.6.19 MINOR SYNC POINT ACK (MIA) SPDU	18
A.6.20 MAJOR SYNC POINT (MAP) SPDU	18
A.6.21 MAJOR SYNC POINT ACK (MAA) SPDU	18
A.6.22 RESYNCHRONIZE (RS) SPDU	18
A.6.23 RESYNCHRONIZE ACK (RA) SPDU	19
A.6.24 PREPARE (PR) SPDU.....	20
A.6.25 EXCEPTION REPORT (ER) SPDU.....	20
A.6.26 EXCEPTION DATA (ED) SPDU	20
A.6.27 GIVE TOKENS CONFIRM (GTC) SPDU.....	20
A.6.28 GIVE TOKENS CONFIRM ACK (GTA) SPDU	20

A.6.29	ACTIVITY START (AS) SPDU	20
A.6.30	ACTIVITY RESUME (AR) SPDU	20
A.6.31	ACTIVITY INTERRUPT (AI) SPDU	20
A.6.32	ACTIVITY INTERRUPT ACK (AIA) SPDU	20
A.6.33	ACTIVITY DISCARD (AD) SPDU	21
A.6.34	ACTIVITY DISCARD ACK (ADA) SPDU	21
A.6.35	ACTIVITY END (AE) SPDU	21
A.6.36	ACTIVITY END ACK (AEA) SPDU	21
	ANNEX B: PRESENTATION PROTOCOL PDUS (NORMATIVE)	22
B.1.	PRESENTATION SERVICE PARAMETERS	22
B.1.2.	CONNECTION INITIATOR OR RESPONDER CAPABILITIES	22
B.1.3.	PROTOCOL MECHANISMS	22
B.1.4.	FUNCTIONAL UNITS	23
B.1.5.	PRESENTATION PDU USAGE BY PROFILE	23
B.1.6.	PRESENTATION CONTEXT MANAGEMENT PDU USAGE BY PROFILE	24
B.1.7.	OTHER PRESENTATION PDU USAGE BY PROFILE	24
B.1.8.	PRESENTATION CONTEXT RESTORATION FUNCTIONAL UNIT	24
B.2.	SUPPORTED PARAMETERS	25
B.2.1.	CONNECT PRESENTATION (CP) PPDU	25
B.2.2.	CONNECT PRESENTATION ACCEPT (CPA) PPDU	25
B.2.3.	CONNECT PRESENTATION REJECT (CPR) PPDU	26
B.2.4.	ABNORMAL RELEASE USER (ARU) PPDU	26
B.2.5.	ABNORMAL RELEASE PROVIDER (ARP) PPDU	27
B.2.6.	ALTER CONTEXT (AC) PPDU	27
B.2.7.	ALTER CONTEXT ACKNOWLEDGE (ACA) PPDU	27
B.2.8.	PRESENTATION DATA (TD) PPDU	27
B.2.9.	PRESENTATION TYPED DATA (TTD) PPDU	27

B.2.10. EXPEDITED DATA (TE) PPDU	28
B.2.11. CAPABILITY DATA (TC) PPDU	28
B.2.12. CAPABILITY DATA ACKNOWLEDGE (TCC) PPDU	28
B.2.13. RESYNCHRONIZE (RS) PPDU	29
B.2.14. RESYNCHRONIZE ACKNOWLEDGE (RSA) PPDU	29
B.2.15. SESSION SERVICE PRIMITIVES NOT CARRYING PRESENTATION PCI.....	30
B.3. SUPPORT OF SYNTAXES.....	31
B.3.1. TRANSFER SYNTAXES SUPPORTED	31
B.3.2. ABSTRACT SYNTAXES SUPPORTED	31
ANNEX C: ACSE PROTOCOL PDUS (NORMATIVE)	1
C.1. SUPPORTED FUNCTIONS.....	1
C.2. INITIATOR/RESPONDER CAPABILITIES	1
C.3. FUNCTIONAL UNITS	2
C.4. ACSE PDU USAGE BY PROFILE	2
C.5. A-ASSOCIATE-REQUEST (AARQ)	3
C.6. A-ASSOCIATE-RESPONSE (AARE)	4
C.7 A-RELEASE-REQUEST (RLRQ)	4
C.8. A-RELEASE-RESPONSE (RLRE)	5
C.9. A-ABORT (ABRT).....	5
C.10. AE TITLE NAME FORMS	5
C.11. AUTHENTICATION VALUE FORM.....	6

Part 5

1. SCOPE	1
2. NORMATIVE REFERENCES	1
3. DEFINITIONS AND ABBREVIATIONS.....	1
4. OVERVIEW.....	1

5. USE OF FUNCTIONAL UNITS.....	1
6. SCENARIO	2
7. USAGE OF UNDERLYING STANDARDS	2
8. DETAILED DESCRIPTION.....	2
9. CONFORMANCE.....	2

Part 6

1. SCOPE	1
2. NORMATIVE REFERENCES.....	1
3. DEFINITIONS AND ABBREVIATIONS.....	1
4. OVERVIEW.....	1
5. USE OF FUNCTIONAL UNITS.....	1
6. SCENARIO	2
7. USAGE OF UNDERLYING STANDARDS	2
8. DETAILED DESCRIPTION.....	2
9. CONFORMANCE.....	2

Part 7

1. SCOPE	1
2. NORMATIVE REFERENCES.....	1
3. DEFINITIONS AND ABBREVIATIONS.....	1
4. OVERVIEW.....	1
5. USE OF FUNCTIONAL UNITS.....	1
6. SCENARIO	2
7. USAGE OF UNDERLYING STANDARDS	2
8. DETAILED DESCRIPTION.....	2
9. CONFORMANCE.....	2

Part 8

1. SCOPE1

2. NORMATIVE REFERENCES1

3. DEFINITIONS AND ABBREVIATIONS1

4. OVERVIEW1

5. USE OF FUNCTIONAL UNITS1

6. SCENARIO2

7. USAGE OF UNDERLYING STANDARDS2

8. DETAILED DESCRIPTION2

9. CONFORMANCE2

Part 9

1. SCOPE1

2. NORMATIVE REFERENCES1

3. DEFINITIONS AND ABBREVIATIONS1

4. OVERVIEW1

5. USE OF FUNCTIONAL UNITS1

6. SCENARIO2

7. USAGE OF UNDERLYING STANDARDS2

8. DETAILED DESCRIPTION2

9. CONFORMANCE2

Part 10

1. SCOPE1

2. NORMATIVE REFERENCES1

3. DEFINITIONS AND ABBREVIATIONS1

4. OVERVIEW	1
5. USE OF FUNCTIONAL UNITS	1
6. SCENARIO	2
7. USAGE OF UNDERLYING STANDARDS	2
8. DETAILED DESCRIPTION	2
9. CONFORMANCE	2

Part 12 OIW Application Contexts

15.12.1 UDT WITH CCR	1
15.12.2 UDT WITHOUT CCR	6

List of Tables

Table 1 - Conformance classes supported	4
Table 2 - Support of Functional Units.....	5
Table 3 - Support for concatenation/separation.....	5
Table 4 - Association establishmentT	6
Table 5 - Support for mandatory and optional bidding	7
Table 6 - Support for contention management	7
Table 7 - Support for the bid mechanism	8
Table 8 - TP dialogue establishment.....	8
Table 9 - Transaction branch establishment.....	9
Table 10 - Roles in a transaction tree supported	10
Table 12 - TP APDUs for the DIALOGUE FU	11
Table 13 - TP-BEGIN-DIALOGUE-RI for Dialogue.....	12
Table 14 - Detail of TPSU-TITLE fields for the dialogue field of TP-BEGIN-DIALOGUE-RI APDU...13	13
Table 15 - TP-BEGIN-DIALOGUE-RC for Dialogue	13
Table 16 - TP-END-DIALOGUE-RI for Dialogue	14
Table 17 - TP-ABORT-RI, for user.....	14
Table 18 - TP-ABORT-RI, for provider.....	14
Table 19 - TP-BID-RI.....	15
Table 20 - TP-BID-RC	15
Table 21 - TP-INITIALIZE-RI	15
Table 22 - TP-INITIALIZE-RC.....	15
Table 23 - TP APDUs for the SHARED Control FU.....	16
Table 24 - TP APDUs for the Polarized Control FU.....	17
Table 25 - TP APDUs for the Handshake FU	18
Table 26 - TP-HANDSHAKE-RI.....	18
Table 27 - TP-HANDSHAKE-AND-GRANT-CONTROL-RI.....	19

Table 28 - TP APDUs for Commit FU	19
Table 29 - TP-PREPARE-RI	20
Table 30 - TP-DEFER-RI	20
Table 31 - TP-HEURISTIC-REPORT-RI	20
Table 32 - TP-TOKEN-GIVE-RI	21
Table 33 - TP APDUs for Recovery FU	22
Table 34 - TP-BEGIN-DIALOGUE-RI (Recovery FU)	23
Table 35 - TP-BEGIN-DIALOGUE-RC (Recovery FU)	24
Table 36 - TP-END-DIALOGUE-RI (Recovery FU)	24
Table 37 - TP-BID-RI (Recovery FU)	24
Table 38 - TP-RECOVER-RI APDU	25
Table 39 - TP-TOKEN-GIVE-RI	25
Table 1 - AMENDMENTS IMPLEMENTED	3
Table 2 - ATOMIC-ACTION-BRANCH ESTABLISHMENT BY PROFILE	4
Table 3 - SUPPORT FOR THE CONCATENATION MECHANISM BY PROFILE	4
Table 4 - CCR PDU USAGE BY PROFILE	5
Table 5 - C-BEGIN-RI	5
Table 6 - ATOMIC-ACTION IDENTIFIER DETAIL	6
Table 7 - C-BEGIN-RC	6
Table 8 - C-PREPARE-RI	6
Table 9 - C-BEGIN-RC	7
Table 10 - C-COMMIT-RI	7
Table 11 - C-COMMIT-RC	7
Table 12 - C-ROLLBACK-RI	7
Table 13 - C-ROLLBACK-RC	8
Table 14 - C-RECOVER-RI	8
Table 15 - ATOMIC-ACTION IDENTIFIER DETAIL	9

Table 16 - BRANCH IDENTIFIER DETAIL	10
Table 17 - RECOVERY-STATE.....	11
Table 18 - C-RECOVER-RC	11
Table 19 - ATOMIC-ACTION IDENTIFIER DETAIL	12
Table 20 - BRANCH IDENTIFIER DETAIL	13
Table 21 - RECOVERY-STATE.....	13
Table 1 - FUNCTIONAL UNITS	3
Table 2 - ISO 8327 Protocol Versions Implemented.....	3
Table 3 - PROTOCOL MECHANISMS	4
Table 4 - INITIATOR/RESPONDER CAPABILITIES.....	5
Table 5 - KERNEL FUNCTIONAL UNIT PDU USAGE BY PROFILE	5
Table 6 - NEGOTIATED RELEASE FUNCTIONAL UNIT PROCEDURE USAGE BY PROFILE	6
Table 7 - HALF DUPLEX FUNCTIONAL UNIT PROCEDURE USAGE BY PROFILE	7
Table 8 - EXPEDITED DATA FUNCTIONAL UNIT PROCEDURE USAGE BY PROFILE.....	7
Table 9 - TYPED DATA FUNCTIONAL UNIT PROCEDURES USAGE BY PROFILE.....	8
Table 10 - CAPABILITY DATA FUNCTIONAL UNIT PROCEDURES USAGE BY PROFILE	8
Table 11 - MINOR SYNCHRONIZE FUNCTIONAL UNIT PROCEDURE USAGE BY PROFILE	8
Table 12 - SYMMETRIC SYNCHRONIZE FUNCTIONAL UNIT PROCEDURES USAGE BY PROFILE.....	9
Table 13 - MAJOR SYNCHRONIZE FUNCTIONAL UNIT PROCEDURES USAGE BY PROFILE	9
Table 14 - RESYNCHRONIZE FUNCTIONAL UNIT PROCEDURES USAGE BY PROFILE.....	9
Table 15 - EXCEPTIONS DATA FUNCTIONAL UNIT PROCEDURES USAGE BY PROFILE	10
Table 16 - ACTIVITY MANAGEMENT FUNCTIONAL UNIT PROCEDURES USAGE BY PROFILE.....	10
Table 17 - CONNECT (CN)	11
Table 18 - ACCEPT(AC).....	13
Table 19- REFUSE (RF).....	14
Table 20 - FINISH (FN)	14

Table 21 - DISCONNECT (DN)	15
Table 22 - ABORT (AB).....	15
Table 23 - DATA TRANSFER (DT).....	15
Table 24 - TYPED DATA (TD).....	16
Table 25 - GIVE TOKENS (GT).....	17
Table 26 - PLEASE TOKENS (PT)	17
Table 27 - MINOR SYNC POINT (MIP)	18
Table 28 - MINOR SYNC POINT ACK (MIA)	18
Table 29 - RESYNCHRONIZE (RS)	19
Table 30 - RESYNCHRONIZE ACK (RA)	19
Table 31 - PREPARE (PR).....	20
Table 1 -CONNECTION INITIATOR OR/ RESPONDER CAPABILITIES	22
Table 2 -PROTOCOL MECHANISMS	22
Table 3 - FUNCTIONAL UNITS.....	23
Table 4 KERNEL PDU USAGE BY PROFILE.....	23
Table 5 PRESENTATION CONTEXT MANAGEMENT PDU USAGE BY PROFILE.....	24
Table 6 OTHER PRESENTATION PDU USAGE BY PROFILE	24
Table 7 - CONNECT PRESENTATION (CP)	25
Table 8 - CONNECT PRESENTATION ACCEPT (CPA).....	25
Table 9 - CONNECT PRESENTATION REJECT (CPR)	26
Table 10 - ABNORMAL RELEASE USER (ARU).....	26
Table 11 - ABNORMAL RELEASE PROVIDER (ARP)	27
Table 12- PRESENTATION DATA (TD)	27
Table 13 - PRESENTATION TYPED DATA (TTD)	27
Table 14 - RESYNCHRONIZE (RS)	29
Table 15 - RESYNCHRONIZE ACKNOWLEDGE (RSA).....	29
Table 16 SESSION SERVICE PRIMITIVES NOT CARRYING PRESENTATION PCI	30

Table 17 - TRANSFER SYNTAXES SUPPORTED.....	31
Table 18 - ABSTRACT SYNTAXES.....	31
Table 1 - SUPPORTED FUNCTIONS.....	1
Table 2 -INITIATOR/RESPONDER CAPABILITIES.....	1
Table 3 -FUNCTIONAL UNITS.....	2
Table 4 ACSE PDU USAGE BY PROFILE	2
Table 5 - A-ASSOCIATE-REQUEST (AARQ).....	3
Table 6 - A-ASSOCIATE-RESPONSE (AARE).....	4
Table 7 - A-RELEASE-REQUEST (RLRQ)	4
Table 8 - A-RELEASE-RESPONSE (RLRE)	5
Table 9 - A-ABORT (ABRT).....	5
Table 10 - AE TITLE NAME FORMS.....	5
Table 11- AUTHENTICATION VALUE FORM	6

THIS PAGE IS INTENTIONALLY LEFT BLANK

TITLE: Information Technology - Open Systems Interconnection - International Standardized Profile
12061-1: OSI TP

Part 1 : INTRODUCTION

SOURCE: Joint AOW / EWOS / OIW on Transaction Processing

DATE: December 18, 1992

STATUS: This document has been harmonized among the three workshops (AOW/EWOS/OIW) and has been submitted to SGFS for progression to ISP.

CONTENTS

Introduction

1. SCOPE
2. NORMATIVE REFERENCES
3. DEFINITIONS AND ABBREVIATIONS
4. NOTATION
5. TAXONOMY STRUCTURE
 - 5.1 Guidelines for splitting up the profiles
 - 5.2 Support of OSI TP Functional Units
 - 5.3 Transaction Processing Profiles tree
6. CONFORMANCE

INTRODUCTION

The aim of Open Systems Interconnection is to allow, with a minimum of technical agreement outside the interconnection standards, the interconnection of computer systems:

- a. from different manufacturers,
- b. under different management,
- c. of different levels of complexity,
- d. of different technologies.

Transaction Processing is concerned with identifiable information which can be related as transactions, which may involve two or more Open Systems. In the framework of Open Systems Interconnection (OSI) a transaction is defined as "a set of related operations characterized by four properties: atomicity, consistency, isolation and durability."

The definition highlights that a distributed transaction is more than a simple exchange of messages, but that the exchanges form a protected indivisible set.

This multi-part document contains the complete specification of the six profiles identified in M-IT-02 and TR 10000.

Part 1 contains the taxonomy for the OSI TP profiles.

Part 2 contains the specification of the support of OSI TP APDUs for each of the profiles specified in Parts 5 to 10.

Part 3 contains the specification of the support of the CCR APDUs for each of the profiles specified in Part 5 to 10.

Part 4 contains the specification of the support of ACSE, Presentation and Session APDUs for each of the profiles specified in Part 5 to 10.

Parts 5 to 10 specify the six profiles which are defined, based on the OSI TP standard. These six parts make reference to Parts 2 to 4.

THIS PAGE IS INTENTIONALLY LEFT BLANK

Information Technology - Open Systems Interconnection - International Standardized Profiles
12061-1: OSI Distributed Transaction Processing.

Part 1: INTRODUCTION

1. SCOPE

Part I of this document introduces the overall structure of the specification of the OSI TP profiles. This includes:

- a) the identification of the Transaction Processing profiles defined in this document, together with the Transaction Processing Profiles Tree;
- b) the identification of the various Parts which constitute this document;
- c) the list of the references to other standards relevant to the definition of the OSI TP profiles;
- d) the definitions and abbreviations used through the various parts of this document.

2. NORMATIVE REFERENCES

The following documents contain provisions which, through reference in this text, constitute provisions of this part of ISO/IEC ISP 12061. At the time of publication, the editions indicated were valid. All documents are subject to revision, and parties to agreements based on this part of ISO/IEC ISP 12061 are warned against automatically applying any more recent editions of the documents listed below, since the nature of references made by ISPs to such documents, is that they may be specific to a particular edition. Members of IEC and ISO maintain registers of currently valid International Standards and ISPs, and CCITT maintains published editions of its current Recommendations.

The following ISO standards contain provisions for the definition of Transaction Processing profiles and are referenced in this document:

ISO/IEC 8327 ¹	Information Processing Systems - Open Systems Interconnection - Basic connection oriented session protocol specification
ISO/IEC DIS 8327-2	Information Processing Systems - Open Systems Interconnection - Basic connection oriented session PICS Proforma.
ISO/IEC 8327 AM3	Information Processing Systems- Open Systems Interconnection - Session. Additional resynchronisation functionality.
ISO/IEC 8650	Information Processing Systems - Open Systems Interconnection

¹ Second edition to be published

	Protocol Specification for the Association Control Service Element.
ISO/IEC DIS 8650-2	Information Processing Systems - Open Systems Interconnection - PICS Proforma for the Association Control Service Element.
ISO/IEC 8823:1988	Information Processing Systems - Open Systems Interconnection - Connection Oriented Presentation Protocol Specification.
ISO/IEC DIS 8823-2	Information Processing Systems - Open Systems Interconnection - Connection Oriented Presentation PICS Proforma.
ISO/IEC 8823 AM5	Information Technology - Open Systems Interconnection - Presentation. Additional resynchronisation functionality.
ISO/IEC 8825:1990	Information Processing Systems - Open Systems Interconnection - Specification of Basic Encoding Rules for Abstract Syntax Notation One (ASN.1).
ISO/IEC 9805-1:1990	Information Technology - Open Systems Interconnection - Protocol Specification for the Commitment, Concurrency and Recovery service element.
ISO/IEC DIS 9805-2	Information Technology - Open Systems Interconnection - CCR PICS Proforma.
ISO/IEC 9805 AM2	Information Technology - Open Systems Interconnection - CCR. Session mapping changes.
ISO/IEC 10026-1:1992	Information Technology - Open Systems Interconnection - Distributed Transaction Processing: Model.
ISO/IEC 10026-2:1992	Information Technology - Open Systems Interconnection - Distributed Transaction Processing: Service Definition.
ISO/IEC 10026-3:1992	Information Technology - Open Systems Interconnection - Distributed Transaction Processing: Protocol Specification.
ISO/IEC DIS 10026-4	Information Technology - Open Systems Interconnection - Distributed Transaction Processing: PICS Proforma
ISO/IEC 11188-1 ²	Information Technology - International Standardized Profile- Common Upper Layer Requirements

² Currently a regional workshop document

3. DEFINITIONS AND ABBREVIATIONS

The definitions and abbreviations in this document not listed in this section are found in the TP model document (ISO/IEC 10026-1).

- AS - Conformance class for Application Supported Transaction
- CP - Conformance class for Chained Provider Supported Transaction Branches
- UP - Conformance class for Unchained Provider Supported Transaction Branches
- FU - Functional Unit
- ISP - International Standardized Profile
- PDU - Protocol Data Unit
- PPDU - Presentation Protocol Data Unit
- SPDU - Session Protocol Data Unit

4. NOTATION

- The following notations are used in the tables contained in Parts 2 to 4 of this ISP:
- 1) The Item Number uniquely identifies each capability, parameter or field within the tables.
 - 2) The Parameter column provides the name of each PDU parameter.
 - 3) The Base Standard Status column indicates static requirements as defined by the base standard PICS Proforma.

The notation used in this column is as defined:

- in the OSI TP and CCR PICS proforma documents for the OSI TP and CCR related tables;
- in the "Common Upper Layer Requirement" document for ACSE, Presentation and Session related tables.

However, as the ISP is not intended to duplicate the information contained in the documents listed above, the notation has been simplified as follows:

- "C" is used instead of any "Cxxx", where xxx is an integer.
- "O.n" is used instead of any "O.xxx", where xxx is an integer.

The exact definition of conditions and options will be found in the referenced documents.

Note that the "M" and "O" notations have not been changed.

Note that the "D" (default) notation may be used in this column, whilst it is not used in the PROFILE Status column (included within the M notation).

- 4) The Profile ID column, if present, defines how this parameter is used by a specific profile.
When this column is empty, the feature applies to all profiles to which the PDU applies.
- 5) The Profile Status column indicates the requirements for the feature.

These requirements are valid only within the scope of a specific profile, i.e., they apply to an instance of an implementation operating within the limits specified by that profile. For instance, the notation NA used for some feature does not preclude an implementation from supporting more than one profile.

M = Mandatory. The feature shall be supported, i.e. its syntax and procedures shall be implemented as specified in the base standard or restricted by this ISP, by all implementations claiming conformance to this profile. It is not necessary that a mandatory parameter appears in all instances of

communication when either a default value has been specified, or this parameter is not used at the service level.

- C = Conditional. Any feature so marked must be implemented under the conditions specified in the profile (e.g. Mandatory, Not Applicable, etc., for a certain instance of communication). The requirements for the feature then follow the rules of M, NA, etc.
- O = Optional. This is an optional feature in the base standard. It is left to the implementor as to whether this feature is implemented. Optional features for a sender need not be implemented. Optional features for a receiver must be recognized and may be processed in a manner consistent with the base standard, or these profiles. If implemented, a feature will be subject to conformance testing.
- NA = Not Applicable. This feature is not defined in the context where it is mentioned because it is either logically or physically impossible for the feature to occur. The occurrence of this feature is a protocol error. It will be handled as specified in the base standard or this ISP.
- I = Out of Scope. This is an optional feature of a base standard. However, this feature is not used by the profile nor by a referencing specification. If such a feature is received it is processed according to local procedures. Local procedures are procedures that are not defined by any base standards or profiles. It will not generate a protocol error.

There are presently some instances where features marked 'M' in the base standard are marked 'I' in this profile. This has been done only because the base standard PICS are not yet at full international standard status and it is believed that the markings of the feature will change during progression to full international standard status. It is intended to remove all such inconsistencies with the base standard before publication of this ISP.

- * = U-ASE defined. This is an optional feature of a base standard. This feature may or may not be used by a referencing specification. How it is used is specified by the referencing specification. When a default value is given in this profile, the referencing specification may choose not to specify any value in which case, the default value applies. A default value is specified in parentheses following the *, e.g., *(I).

- O.N= The notation O.N, where N is an integer, is used in some Status Columns. This notation indicates a reference to a unique group of capabilities. A note (as indicated in the Notes Column) will explain the exact requirement using one of the following forms:

O.N = Support of exactly one of these items is required.

O.N = Support of at least one of these items is required.

It is always necessary to consult the corresponding note to determine which situation applies.

When status for sending and receiving values differ, they will be separated by a slash, with the sender on the left and receiver on the right. If they are the same there will only be one status in the cell. The integer suffix to a status refers to a condition that will be found either immediately following that table, or following an earlier table in the same part of this ISP.

- 6) The T/L/V Allowed column specifies the range of types, length, or values this parameter can assume or contain. This column can have multiple definitions based on which profile is being described. When multiple definitions are possible this column will be defined in conjunction with the Profile ID column. The notation {} denote no bits in the parameter's value.
- 7) The Notes column points to notes following the table.

5. TAXONOMY STRUCTURE

5.1. GUIDELINES FOR SPLITTING UP THE PROFILES

This subclause specifies which functional units combine to form each profile. Refer to the appropriate part of this ISP for the specification of how a specific profile uses a PDU and its parameters. Profiles are identified by a coding method which consists of two levels, but which can easily be expanded as future needs warrant. The first level indicates the conformance class. The second level indicates whether polarized or shared control is used. The levels are defined as:

Level one:

- 1.- Application Supported transactions.
- 2.- Provider supported unchained transactions.
- 3.- Provider supported chained transactions.

Level two:

- 1.- Polarized control.
- 2.- Shared control.

5.2 TRANSACTION PROCESSING PROFILES TREE

The figure hereafter gives the Transaction Processing Profiles tree:

Transaction Processing	ATP
Application Supported Transactions	ATP1
Polarized Control	ATP11
Shared Control	ATP12
Provider Supported Unchained Transactions	ATP2
Polarized Control	ATP21
Shared Control	ATP22
Provider Supported Chained Transactions	ATP3
Polarized Control	ATP31
Shared Control	ATP32

The first level of this decomposition (ATPx) corresponds to the definition of the three conformance classes defined in the OSI TP standard. The second level (ATPxy) corresponds to the selection between Polarized Control and Shared Control for each of the conformance classes. The conformance classes and the functional units that compose them are summarized in the following list:

- 1. ATP-11 Application Supported Transactions - Polarized Control:
DIALOGUE + HANDSHAKE + POLARIZED CONTROL**
- 2. ATP-21 Provider Supported Unchained Transactions - Polarized Control:
DIALOGUE + HANDSHAKE + POLARIZED CONTROL + COMMIT + RECOVERY +
UNCHAINED TRANSACTION**
- 3. ATP-31 Provider Supported Chained Transactions - Polarized Control:
DIALOGUE + HANDSHAKE + POLARIZED CONTROL + COMMIT + RECOVERY +
CHAINED TRANSACTION**
- 4. ATP-12 Application Supported Transactions - Shared Control:
DIALOGUE + HANDSHAKE (Optional) + SHARED CONTROL**
- 5. ATP-22 Provider Supported Unchained Transaction - Shared Control:
DIALOGUE + HANDSHAKE (Optional) + SHARED CONTROL + COMMIT + RECOVERY +
UNCHAINED TRANSACTION**
- 6. ATP-32 Provider Supported Chained Transactions - Shared Control:
DIALOGUE + HANDSHAKE (Optional) + SHARED CONTROL + COMMIT + RECOVERY +
CHAINED TRANSACTION**

Since the Profile ID is not carried as a protocol parameter, implementations may determine the profile governing a particular instance of communication by the TP Functional Units selected for that dialogue.

6. CONFORMANCE

This part of ISO/IEC ISP 12061 states requirements upon implementors to achieve inter networking. A claim of conformance to one of parts five to ten of this ISP is a claim that all requirements in the relevant base standards are satisfied, and that all requirements in the relevant parts are satisfied.

Annexes to parts two, three and four state the relationship between these requirements and those of the base standard.

There is no conformance requirement from this ISP on features marked "" in the annexes of parts two and four.

Each of parts five to ten contain specific conformance requirements for that part.

TITLE: WORKING DOCUMENT FOR

**Information Technology - Open Systems Interconnection - International Standardized Profile
12061-2: OSI TP**

Part 2: Support of OSI TP APDUs.

SOURCE: Joint AOW / EWOS / OIW on Transaction Processing

DATE: December 18, 1992

**STATUS: This document has been harmonized among the three workshops (AOW/EWOS/OIW)
and has been submitted to SGFS for progression to ISP.**

CONTENTS

INTRODUCTION

1. SCOPE
2. NORMATIVE REFERENCES
3. DEFINITIONS and ABBREVIATIONS
4. NOTATION
5. SUPPORT OF OSI TP PROTOCOL
6. CONFORMANCE

ANNEX

- A. SUPPORT OF THE OSI TP PROTOCOL (NORMATIVE).

INTRODUCTION

The aim of Open Systems Interconnection is to allow, with a minimum of technical agreement outside the interconnection standards, the interconnection of computer systems

- a. from different manufacturers,
- b. under different management,
- c. of different levels of complexity,
- d. of different technologies.

Transaction Processing is concerned with identifiable information which can be related as transactions, which may involve two or more Open Systems. In the framework of Open Systems Interconnection (OSI) a transaction is defined as "a set of related operations characterized by four properties: atomicity, consistency, isolation and durability."

The definition highlights that a distributed transaction is more than a simple exchange of messages, but that the exchanges form a protected indivisible set.

This multi-part document contains the complete specification of the six profiles identified in M-IT-02 and TR 10000.

Part 1 contains the taxonomy for the OSI TP profiles.

Part 2 contains the specification of the support of OSI TP APDUs for each of the profiles specified in Parts 5 to 10.

Part 3 contains the specification of the support of the CCR APDUs for each of the profiles specified in Part 5 to 10.

Part 4 contains the specification of the support of ACSE, Presentation and Session APDUs for each of the profiles specified in Part 5 to 10.

Parts 5 to 10 specify the six profiles which are defined, based on the OSI TP standard. These six parts make reference to Parts 2 to 4.

This page is intentionally left blank

**Information Technology - Open Systems Interconnection -
International Standardized Profiles 12061-2: OSI Distributed
Transaction Processing.**

Part 2: SUPPORT OF OSI TP APDUs

1. SCOPE

This part of this ISP specifies the status for the support of the OSI TP protocol for the profiles identified in Part 1 of this ISP.

2. NORMATIVE REFERENCES

The references listed in Part 1 of this ISP apply.

3. DEFINITIONS AND ABBREVIATIONS

The definitions and abbreviations contained in Part 1 of this ISP apply to this Part.

4. NOTATION

The notation introduced in Part 1 of this ISP applies to this Part.

5. SUPPORT OF OSI TP PROTOCOL

The support of the OSI TP protocol is as described in Annex A (normative).

The structure of Annex A is based on the structure of Annex A of ISO/IEC 10026-4, in particular in the numbering of the clauses.

When a clause of Annex A of ISO/IEC 10026-4 is not relevant to the profiles, this is stated.

6. Conformance

To conform to the OSI TP Protocol used in any of the profiles in this ISP, an implementation shall implement, according to the specifications given in ISO/IEC 10026-3:

1. All the mandatory features identified in Annex A.
2. All the selected optional features, as identified in the completed TP PICS.

ANNEX A: SUPPORT OF THE OSI TP PROTOCOL (Normative)

A.1. IDENTIFICATION

No restriction is applied to clause A.1 of ISO/IEC 10026-4 by this part of ISO/IEC ISP 12061.

A.2. CLAIMED CONFORMANCE TO STANDARDS

A.2.1. ISO/IEC 10026-3

A.2.1.1. VERSION NUMBER(S)

Answer shall be "NONE".

A.2.1.2. GLOBAL CONFORMANCE CLAIM

Answer shall be "YES".

A.2.2. ISO/IEC 10026 AMENDMENTS

Both answers shall be "NONE".

A.2.3. ISO/IEC 10026 TECHNICAL CORRIGENDA

Both answers shall be "NONE".

Note: At the time of the approval of the final text of this ISP, no Technical Corrigenda was approved for ISO/IEC 10026. When this will become false, the present ISP will be corrected accordingly.

A.2.4. CONFORMANCE CLASS(ES) SUPPORTED

Table 1 - CONFORMANCE CLASSES SUPPORTED

ITEM #	CONFORMANCE CLASSES	ISO/IEC 10026-4	PROFILES						
			11	21	31	12	22	32	NOTES
1	Application Transaction Branches	O	M	NA	NA	M	NA	NA	
2	Chained Provider Supported Transaction Branches	O	NA	NA	M	NA	NA	M	
3	Unchained Provider Supported Transaction Branches	O	NA	M	NA	NA	M	NA	

NOTES

Conformance to more than one profile may be claimed for an implementation. For example, an implementation that conforms to profile 21 will often be capable of conforming to the corresponding Profile 11.

A.3. FUNCTIONAL UNITS, LIMITS AND PROTOCOL MECHANISMS

A.3.1. SUPPORT OF FUNCTIONAL UNITS

Table 2 - SUPPORT OF FUNCTIONAL UNITS

ITEM #	FUNCTIONAL UNITS	ISO/IEC 10026-4			PROFILES						NOTES
		AS	CP	UP	11	21	31	12	22	32	
1	Dialogue	M	M	M	M	M	M	M	M	M	
2	Shared Control	O.n	O.n	O.n	NA	NA	NA	M	M	M	
3	Polarized Control	O.n	O.n	O.n	M	M	M	NA	NA	NA	
4	Handshake	O	O	O	M	M	M	O	O	O	
5	Commit	NA	M	M	NA	M	M	NA	M	M	
6	Chained Transactions	NA	M	NA	NA	NA	M	NA	NA	M	
7	Unchained Transactions	NA	NA	M	NA	M	NA	NA	M	NA	
8	Recovery	NA	M	M	NA	M	M	NA	M	M	

A.3.2. PROTOCOL MECHANISMS IMPLEMENTED

A.3.2.1. CONCATENATION/SEPARATION

Table 3 - SUPPORT FOR CONCATENATION/SEPARATION

ITEM #	ROLE	ISO/IEC 10026-4	PROFILES						NOTES
			11	21	31	12	22	32	
1	Concatenation	O	O	O	O	O	O	O	
2	Separation	M	M	M	M	M	M	M	

A.3.2.2. ASSOCIATION ESTABLISHMENT

Table 4 - ASSOCIATION ESTABLISHMENT

ITEM #	ROLE	ISO/IEC 10026-4	PROFILES						
			11	21	31	12	22	32	NOTES
1	Initiator	C	O	M	M	O	M	M	1,2,4
2	Acceptor	C	O	M	M	O	M	M	1,3,4
3	Rejector	O	M	M	M	M	M	M	5,6

NOTES

- 1 When the commitment is supported, then the implementation must be able to support both the association initiator and acceptor role in order to be able to perform recovery adequately.
- 2 The initiator role here implies being capable of issuing an A-ASSOCIATE request and being capable of receiving an A-ASSOCIATE confirm.
- 3 The Acceptor role here implies being capable of receiving an A-ASSOCIATE indication and being capable of issuing an A-ASSOCIATE response with a positive answer.
- 4 For Profiles 11 and 12, at a minimum, the association establishment initiator role or the association establishment acceptor role shall be supported.
- 5 The Rejector role here implies being capable of receiving an A-ASSOCIATE indication and being capable of issuing an A-ASSOCIATE response with a negative answer.
- 6 Although it is mandatory to be able to reject an association, note that in some particular environments it could occur that the reject is always performed by some lower protocol machine (e.g. ACSE).

A.3.2.3. SUPPORT FOR MANDATORY AND OPTIONAL BIDDING

Table 5 - SUPPORT FOR MANDATORY AND OPTIONAL BIDDING

ITEM #	ROLE	ISO/IEC 10026-4	PROFILES						
			11	21	31	12	22	32	NOTES
1	Initiator with Bid mandatory	C	C100	M	M	C100	M	M	
2	Initiator with Bid optional	C	C101	O	O	C101	O	O	
3	Responder with Bid mandatory	C	C102	M	M	C102	M	M	
4	Responder with Bid optional	C	C103	M	M	C103	M	M	

100 If the Initiator of an association role is supported(A.3.2.2/1) then O else NA.

101 If the initiator of the association role is supported (A.3.2.2/1) then M else NA.

102 If the Acceptor of an Association role is supported(A.3.2.2/2) then O else NA.

103 If the Acceptor of an Association role is supported(A.3.2.2/2) then M else NA.

A.3.2.4. CONTENTION

Table 6 - SUPPORT FOR CONTENTION MANAGEMENT

ITEM #	ROLE	ISO/IEC 10026-4	PROFILES						
			11	21	31	12	22	32	NOTES
1	Contention Winner	O.n	O	M	M	O	M	M	1,2
2	Contention Loser	O.n	O	M	M	O	M	M	1,2

NOTES

- When the commitment is supported, in order to enable channel establishment (initiator or acceptor) to be accepted, it is required to be able to support both the contention winner and contention loser roles.
- For profiles 11 and 12, at least one of the contention winner and contention loser roles shall be supported.

A.3.2.5. BID MECHANISM

Table 7 - SUPPORT FOR THE BID MECHANISM

ITEM #	ROLE	ISO/IEC 10026-4	PROFILES						
			11	21	31	12	22	32	NOTES
1	Initiator	C	C105	M	M	C105	M	M	1,3
2	Responder	C	C106	M	M	C106	M	M	2,3

105 If the Contention Loser role is supported (A.3.2.4/2) then
 If Associations with Bid Mandatory are supported (A.3.2.3/1 or A.3.2.3/3)
 then M else O
 else NA

106 If the Contention Winner role is supported (A.3.2.4/1) then M else NA

NOTES

1. The initiator role here implies being capable of sending a TP-BID-RI APDU and being capable of receiving a TP-BID-RC APDU.
2. The responder role here implies being capable of receiving a TP-BID-RI APDU and being capable of sending a TP-BID-RC APDU with a positive answer.
3. When the commitment is supported, in order to enable channel establishment (initiator and acceptor) to be accepted, it is required to be able to support both the bid initiator and bid responder roles.

A.3.2.6. DIALOGUE ESTABLISHMENT

Table 8 - TP DIALOGUE ESTABLISHMENT

ITEM #	ROLE	ISO/IEC 10026-4	PROFILES						
			11	21	31	12	22	32	NOTES
1	Initiator	O.n	O	O	O	O	O	O	1,3
2	Acceptor	O.n	O	O	O	O	O	O	2,3
3	Rejector	M	M	M	M	M	M	M	

NOTES

1. The initiator role here implies being capable of sending a TP-BEGIN-DIALOGUE-RI APDU and being capable of receiving a TP-BEGIN-DIALOGUE-RC APDU.

2. The Acceptor role here implies being capable of receiving a TP-BEGIN-DIALOGUE-RI APDU and being capable of sending a TP-BEGIN-DIALOGUE-RC APDU with a positive answer.
3. For each of the profiles, at least one of the Acceptor or initiator roles shall be implemented.

A.3.2.7. TRANSACTION BRANCH ESTABLISHMENT

Table 9 - TRANSACTION BRANCH ESTABLISHMENT

ITEM #	ROLE	ISO/IEC 10026-4	PROFILES						
			11	21	31	12	22	32	NOTES
1	Initiator	C	NA	C107	C107	NA	C107	C107	
2	Acceptor	C	NA	C108	C108	NA	C108	C108	

107 If the implementation is capable of initiating a dialogue (A.3.2.6/1) then M else NA.

108 If the implementation is capable of accepting a dialogue (A.3.2.6/2) then M else NA.

A.3.2.8. ROLES IN A TRANSACTION TREE SUPPORTED

Table 10 - ROLES IN A TRANSACTION TREE SUPPORTED

ITEM #	ROLE	ISO/IEC 10026-4	PROFILES						
			11	21	31	12	22	32	NOTES
1	Root Node	C	NA	O	O	NA	O	O	1
2	Intermediate Node	C	NA	O	O	NA	O	O	1
3	Leaf Node	C	NA	C109	C109	NA	C109	C109	1

109 If capable of acting as an intermediate node then M else O.

NOTES

1. An implementation must be capable of acting as either a root, an intermediate or a leaf node.

A.3.2.9. SUPPORT FOR RECOVERY

This clause does not apply to profiles 11 and 12.

Table 11 - SUPPORT FOR RECOVERY

ITEM #	ROLE	ISO/IEC 10026-4	PROFILES						
			11	21	31	12	22	32	NOTES
1	One-Way recovery	M	NA	M	M	NA	M	M	
2	Two-Way Recovery	O	NA	O	O	NA	O	O	

A.4. TP PROTOCOL - GENERAL

The clause A.4 in ISO/IEC 10026-4 is not relevant.

A.5. TP PROTOCOL - SUPPORT OF THE DIALOGUE FUNCTIONAL UNIT

A.5.1. DIALOGUE FU APDUS

Table 12 - TP APDUS FOR THE DIALOGUE FU

ITEM #	PROTOCOL DATA UNIT	ISO/IEC 10026-4	PROFILES						
			11	21	31	12	22	32	NOTES
1	TP-BEGIN-DIALOGUE-RI	C	C107/ M	C107/ M	C107/ M	C107/ M	C107/ M	C107/ M	1
2	TP-BEGIN-DIALOGUE-RC	C	M/ C107	M/ C107	M/ C107	M/ C107	M/ C107	M/ C107	2
3	TP-END-DIALOGUE-RI	C	M	M	NA	M	M	NA	
4	TP-END-DIALOGUE-RC	C	M	M	NA	M	M	NA	
5	TP-U-ERROR-RI	M	M	M	M	M	M	M	
6	TP-ABORT-RI	M	M	M	M	M	M	M	
7	TP-BID-RI	C	C111/ C110	M	M	C111/ C110	M	M	
8	TP-BID-RC	C	C110/ C111	M	M	C110/ C111	M	M	
9	TP-INITIALIZE-RI	C	C112/ M	M	M	C112/ M	M	M	3
10	TP-INITIALIZE-RC	C	M/ C112	M	M	M/ C112	M	M	4

110 If the implementation is capable of receiving a Bid (A.3.2.5/2) then M else NA

111 If the implementation is capable of initiating a Bid (A.3.2.5/1) then M else NA

112 If the implementation is capable of initiating an association (A.3.2.2/1) then M else NA

NOTES

1. It is mandatory for every implementation to receive and recognize the TP-BEGIN-DIALOGUE-RI APDU (ref ISO/IEC 10026-3 13.1.2.1.c and 13.1.2.1.f).

2. It is mandatory for every implementation to be capable of rejecting a TP-BEGIN-DIALOGUE-RI (ref ISO/IEC 10026-3 13.1.2.1f)
3. It is mandatory for every implementation to receive and recognize the TP-INITIALIZE-RI APDU (ref ISO/IEC 10026-3 13.1.2.1.a).
4. It is mandatory for every implementation to be capable of rejecting a TP-INITIALIZE-RI APDU

A.5.2. TP-BEGIN-DIALOGUE-RI APDU

A.5.2.1. DETAIL OF THE DIALOGUE FIELD OF TP-BEGIN-DIALOGUE-RI APDU

Table 13 - TP-BEGIN-DIALOGUE-RI for Dialogue

ITEM #	ISO/IEC 10026-4		PROFILE			
	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	Initiating-TPSU-Title	O/M		M	See Table 14	
2	Recipient-TPSU-Title	O/M		M	See Table 14	
3	Functional-Units	D	11	M	{0,4}	
			12	M	{1} OR {1,4}	
			21	M	{0,3,4}	
			22	M	{1,3} or {1,3,4}	
			31	M	{0,2,4}	
			32	M	{1,2} or {1,2,4}	
4	Begin-Transaction	C	11,12,31,32	NA		
			21,22	M		
5	Confirmation	D		M		
6	Correlator	M		M	0..2**31-1	
7	Last-Partner-Identifier	O/M		M	0..2**31-1	1
8	User-Data	O/M		M	0..10K octets+	2

NOTES

1. The Last-Partner-Identifier is marked M because an implementation shall be able to support more than one dialogue or channel on an association.
2. The receiver shall be capable of receiving at least 10K octets of user-data.

A.5.2.1.1. DETAIL OF TPSU-TITLE FIELDS FOR THE DIALOGUE FIELD OF TP-BEGIN-DIALOGUE-RI APDU

Table 14 - Detail of TPSU-TITLE fields for the dialogue field of TP-BEGIN-DIALOGUE-RI APDU

ITEM #	ISO/IEC 10026-4		PROFILE			
	TYPE	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	T61String	O.n/M		C113/M	1..64 octets	1
2	PrintableString	O.n/M		C113/M	1..64 octets	1
3	INTEGER	O.n/M		C113/M	0..2**31-1	1

113 If the TPSU-TITLE is used to carry a RECIPIENT-TPSU-TITLE value then M else O

NOTES

- At least one of the three types shall be supported for the INITIATING-TPSU-TITLE.

A.5.3. TP-BEGIN-DIALOGUE-RC APDU

A.5.3.1. DETAIL OF THE DIALOGUE FIELD OF TP-BEGIN-DIALOGUE-RC APDU

Table 15 - TP-BEGIN-DIALOGUE-RC for Dialogue

ITEM #	ISO/IEC 10026-4		PROFILE			
	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	Functional-Units	O/M		M		
2	Result	D		M		
3	Diagnostic	M		M		
4	Correlator	M		M	0..2**31-1	
5	User-Data	O/M		M	0..10K octets+	1

NOTES

- The receiver shall be capable of receiving at least 10K octets of user-data

A.5.4. TP-END-DIALOGUE-RI APDU

This APDU does not apply to profiles 31 and 32.

Table 16 - TP-END-DIALOGUE-RI for Dialogue

ITEM #	ISO/IEC 10026-4		PROFILE			
	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	Confirmation	D	11,12,21,22	M		

A.5.5. TP-ABORT-RI APDU

A.5.5.1. DETAIL OF THE USER FIELD OF TP-ABORT-RI APDU

Table 17 - TP-ABORT-RI, for user

ITEM #	ISO/IEC 10026-4		PROFILE			
	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	User-Data	O/M		M	0..10K octets +	1

NOTE

1. The receiver shall be capable of receiving at least 10K octets of user-data

A.5.5.2. DETAIL OF THE PROVIDER FIELD OF TP-ABORT-RI APDU

Table 18 - TP-ABORT-RI, for provider

ITEM #	ISO/IEC 10026-4		PROFILE			
	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	Provider-diagnostic	M		M		

A.5.5.3. TP-BID-RI APDU

Table 19 - TP-BID-RI

ITEM #	ISO/IEC 10026-4		PROFILE			
	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	CCR-Token-Requested	D	11,12	M	False	
			21,22,31,32	M		
2	Last-Partner-Identifier	O/M		M	0..2**31-1	

A.5.6. TP-BID-RC APDU

Table 20 - TP-BID-RC

ITEM #	ISO/IEC 10026-4		PROFILE			
	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	Result	D		M		

A.5.7. TP-INITIALIZE-RI APDU

Table 21 - TP-INITIALIZE-RI

ITEM #	ISO/IEC 10026-4		PROFILE			
	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	Protocol-Version	D		M	{version 1}	
2	Contention-Winner-Assignment	D		M		
3	Bid-Mandatory	D		M		
4	Recovery-Context-Handle	C	11,12	I		
			21,22,31,32	O/M	1..64 octets	1

NOTES

- 1 It is optional to send a RECOVERY-CONTEXT-HANDLE (RCH) as the sender may have no use for it. It is mandatory to receive an RCH and be able to send it on the TP-RECOVER-RI APDU.

A.5.8. TP-INITIALIZE-RC APDU

Table 22 - TP-INITIALIZE-RC

ITEM #	ISO/IEC 10026-4		PROFILE			
	PARAMETER	STATUS	PROFILE ID	STATUS	T/LV ALLOWED	NOTES
1	Protocol-Version	D		M	{version 1}	
2	Recovery-Context-Handle	O/M	11,12	I		
			21,22,31,32	O/M	1..64 octets	1
3	Diagnostic	O/M		M		

NOTES

- 1 It is optional to send a Recovery-Context-Handle (RCH) as the sender may have no use for it. It is mandatory to receive an RCH and be able to send it on the TP-RECOVER-RI APDU.

A.6. SUPPORT OF THE SHARED CONTROL FUNCTIONAL UNIT

A.6.1. SHARED CONTROL FUNCTIONAL UNIT APDUS

This clause does not apply to profiles 11, 21 and 31.

Table 23 - TP APDUs for the SHARED Control FU

ITEM #	PROTOCOL DATA UNITS	ISO/IEC 10026-4	PROFILES						
			11	21	31	12	22	32	NOTES
1	TP-U-ERROR-RC	M	NA	NA	NA	M	M	M	

A.7. SUPPORT OF THE POLARIZED CONTROL FUNCTIONAL UNIT

This clause does not apply to profiles 12, 22 and 32..

A.7.1. POLARIZED CONTROL FUNCTIONAL UNIT APDUS

Table 24 - TP APDUs for the Polarized Control FU

ITEM #	PROTOCOL DATA UNITS	ISO/IEC 10026-4	PROFILES						
			11	21	31	12	22	32	NOTES
1	TP-GRANT-CONTROL-RI	M	M	M	M	NA	NA	NA	
2	TP-REQUEST-CONTROL-RI	M	M	M	M	NA	NA	NA	

A.8. SUPPORT OF THE HANDSHAKE FUNCTIONAL UNIT

This clause applies to all profiles.

A.8.1. HANDSHAKE FUNCTIONAL UNIT APDUS

Table 25 - TP APDUs for the Handshake FU

ITEM #	PROTOCOL DATA UNITS	ISO/IEC 10026-4	PROFILES						
			11	21	31	12	22	32	NOTES
1	TP-HANDSHAKE-RI	M	M	M	M	C114	C114	C114	
2	TP-HANDSHAKE-RC	M	M	M	M	C114	C114	C114	
3	TP-HANDSHAKE-AND-GRANT-CONTROL-RI	C	M	M	M	NA	NA	NA	
4	TP-HANDSHAKE-AND-GRANT-CONTROL-RC	C	M	M	M	NA	NA	NA	

114 If the Handshake FU is implemented (A.3.1/4) then M else NA

A.8.2. TP-HANDSHAKE-RI APDU

This APDU is not applicable for profiles 12, 22 and 32 when the handshake FU is not implemented.

Table 26 - TP-HANDSHAKE-RI

ITEM #	ISO/IEC 10026-4		PROFILE			
	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	Confirmation-Urgency	C	11,21,31	NA		
			12,22,32	M		

A.8.3. TP-HANDSHAKE-AND-GRANT-CONTROL-RI APDU

This APDU does not apply to profiles 12, 22 and 32.

Table 27 - TP-HANDSHAKE-AND-GRANT-CONTROL-RI

ITEM #	ISO/IEC 10026-4		PROFILE			
	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	Confirmation-Urgency	D	11,21,31	M		

A.9. TP PROTOCOL - SUPPORT OF THE COMMIT FUNCTIONAL UNIT

This clause applies only to profiles 21, 22, 31 and 32.

A.9.1. COMMIT FUNCTIONAL UNIT APDUS

Table 28 - TP APDUs for Commit FU

ITEM #	PROTOCOL DATA UNITS	ISO/IEC 10026-4	PROFILES						NOTES
			11	21	31	12	22	32	
1	TP-PREPARE-RI	C	NA	C115 /C116	C115 /C116	NA	C115 /C116	C115 /C116	
2	TP-DEFER-RI	C	NA	C115 /C116	C115 /C116	NA	C115 /C116	C115 /C116	
3	TP-HEURISTIC-REPORT-RI	C	NA	C116 /C115	C116 /C115	NA	C116 /C115	C116 /C115	
4	TP-TOKEN-GIVE-RI	M	NA	M	M	NA	M	M	

115 If the implementation is capable of initiating a dialogue (A.3.2.6/1) then M else NA

116 If the implementation is capable of accepting a dialogue (A.3.2.6/2) then M else NA

A.9.1.1. TP-PREPARE-RI APDU

This APDU does not apply to profiles 11 and 12.

Table 29 - TP-PREPARE-RI

ITEM #	ISO/IEC 10026-4		PROFILE			
	PARAMETER	STATUS	PROFILE ID	STATUS	T/LV ALLOWED	NOTES
1	Data-Permitted	C	21,31	M		
			22,32	NA		

A.9.2. TP-DEFER-RI APDU

This APDU does not apply to profiles 11 and 12.

Table 30 - TP-DEFER-RI

ITEM #	ISO/IEC 10026-4		PROFILE			
	PARAMETER	STATUS	PROFILE ID	STATUS	T/LV ALLOWED	NOTES
1	Type	D	22,32	M	End-Dialogue	
			21,31	M		

A.9.3. TP-HEURISTIC-REPORT-RI APDU

This APDU does not apply to profiles 11 and 12.

Table 31 - TP-HEURISTIC-REPORT-RI

ITEM #	ISO/IEC 10026-4		PROFILE			
	PARAMETER	STATUS	PROFILE ID	STATUS	T/LV ALLOWED	NOTES
1	Heuristic-Report	D		M		

A.9.4. TP-TOKEN-GIVE-RI APDU

This APDU does not apply to profiles 11 and 12.

Table 32 - TP-TOKEN-GIVE-RI

ITEM #	ISO/IEC 10026-4		PROFILE			
	PARAMETER	STATUS	PROFILE ID	STATUS	T/LV ALLOWED	NOTES
1	Reason	D		M	Regular, Keep	
2	Correlator	M		M		

A.10. TP PROTOCOL - SUPPORT OF THE RECOVERY FUNCTIONAL UNIT

This clause applies only to profiles 21, 22, 31 and 32.

A.10.1. RECOVERY FUNCTIONAL UNIT APDUS

Table 33 - TP APDUs for Recovery FU

ITEM #	PROTOCOL DATA UNITS	ISO/IEC 10026-4	ATP11	ATP21	ATP31	ATP12	ATP22	ATP32	NOTES
1	TP-BEGIN-DIALOGUE-RI	M	NA	M	M	NA	M	M	1
2	TP-BEGIN-DIALOGUE-RC	M	NA	M	M	NA	M	M	1
3	TP-BID-RI	C	NA	M	M	NA	M	M	1
4	TP-BID-RC	C	NA	M	M	NA	M	M	1,3
5	TP-RECOVER-RI	M/C	NA	M /C117	M /C117	NA	M /C117	M /C117	
6	TP-END-DIALOGUE-RI	M	NA	M	M	NA	M	M	
7	TP-TOKEN-PLEASE-RI	C	NA	C118	C118	NA	C118	C118	3
8	TP-INITIALIZE-RI	M	NA	M	M	NA	M	M	3
9	TP-INITIALIZE-RC	M	NA	M	M	NA	M	M	3
10	TP-TOKEN-GIVE-RI		NA	O	O	NA	O	O	2

117 If the recovery-context-handle field (A.5.7/4, A.5.8/2) is supported in the TP-INITIALIZE-RC and TP-INITIALIZE-RI APDUs then M else NA

118 If Two-Way recovery (A.3.2.9) is supported then M, else NA

NOTES

- 1 When the commitment is supported, in order to enable channel establishment (initiator or acceptor) to be accepted, it is required to be able to support both the bid initiator and the bid responder roles.
- 2 This PDU is not in the Recovery FU in ISO/IEC 10026-4. However it has been added to this ISP as required for support of Two-Way recovery.
- 3 This APDU is specified in clause A.5.

A.10.2. TP-BEGIN-DIALOGUE-RI APDU

A.10.2.1. DETAIL OF THE CHANNEL FIELD OF TP-BEGIN-DIALOGUE-RI APDU

This table does not apply to profiles 11 and 12.

Table 34 - TP-BEGIN-DIALOGUE-RI (Recovery FU)

ITEM #	ISO/IEC 10026-4		PROFILE			
	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	Functional-Units	D		M	{5}	
2	Correlator	M		M	0..2**31-1	
3	Channel-Utilization	D		M	one-way-recovery	
				O	two-way-recovery	
4	Last-Partner-Identifier	O/M		M		

A.10.3. TP-BEGIN-DIALOGUE-RC APDU

A.10.3.1. DETAIL OF THE CHANNEL FIELD OF TP-BEGIN-DIALOGUE-RC APDU

This table does not apply to profiles 11 and 12.

Table 35 - TP-BEGIN-DIALOGUE-RC (Recovery FU)

ITEM #	ISO/IEC 10026-4		PROFILE			
	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	Result	D		M		
2	Diagnostic	M		M		
3	Correlator	M		M	0..2**31-1	

A.10.4. TP-END-DIALOGUE-RI APDU

This table does not apply to profiles 11 and 12.

Table 36 - TP-END-DIALOGUE-RI (Recovery FU)

ITEM #	ISO/IEC 10026-4		PROFILE			
	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	Confirmation	D		M	False	

A.10.5. TP-BID-RI APDU

This table does not apply to profiles 11 and 12.

Table 37 - TP-BID-RI (Recovery FU)

ITEM #	ISO/IEC 10026-4		PROFILE			
	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	CCR-Tokens-Requested	M		M		
2	Last-Partner-Identifier	O/M		M	0..2**31-1	

A.10.6. TP-RECOVER-RI APDU

This APDU does not apply to profiles 11 and 12.

Table 38 - TP-RECOVER-RI APDU

ITEM #	ISO/IEC 10026-4		PROFILE			
	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	Recovery-Context-Handle	M		M	1..64 octets	

A.10.7. TP-TOKEN-GIVE-RI APDU

Table 39 - TP-TOKEN-GIVE-RI

ITEM #	ISO/IEC 10026-4		PROFILE			
	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	Reason	D		M	Two-Way-recovery	
2	Correlator	C		M		

NOTES

This table is not in ISO/IEC 10026-4. However it has been added to this ISP as required for support of Two-Way recovery.

TITLE: WORKING DOCUMENT FOR

Information Technology - Open Systems Interconnection - International Standardized Profile
12061-3: OSI TP

Part 3: Support of the CCR Protocol.

SOURCE: Joint AOW / EWOS / OIW on Transaction Processing .

DATE: December 18, 1992

STATUS: This document has been harmonized among the three workshops (AOW/EWOS/OIW) and has been submitted to SGFS for progression to ISP.

CONTENTS

INTRODUCTION

1. SCOPE
2. NORMATIVE REFERENCES
3. DEFINITIONS and ABBREVIATIONS
4. NOTATION
5. SUPPORT OF OSI CCR PROTOCOL
6. CONFORMANCE

ANNEX

- A. SUPPORT OF THE CCR PROTOCOL (NORMATIVE).

Introduction

The aim of Open Systems Interconnection is to allow, with a minimum of technical agreement outside the interconnection standards, the interconnection of computer systems

- a. from different manufacturers,
- b. under different management,
- c. of different levels of complexity,
- d. of different technologies.

Transaction Processing is concerned with identifiable information which can be related as transactions, which may involve two or more Open Systems. In the framework of Open Systems Interconnection (OSI) a transaction is defined as "a set of related operations characterized by four properties: atomicity, consistency, isolation and durability."

The definition highlights that a distributed transaction is more than a simple exchange of messages, but that the exchanges form a protected indivisible set.

This multi-part document contains the complete specification of the six profiles identified in M-IT-02 and TR 10000.

Part 1 contains the taxonomy for the OSI TP profiles.

Part 2 contains the specification of the support of OSI TP APDUs for each of the profiles specified in Parts 5 to 10.

Part 3 contains the specification of the support of the CCR APDUs for each of the profiles specified in Part 5 to 10.

Part 4 contains the specification of the support of ACSE, Presentation and Session APDUs for each of the profiles specified in Part 5 to 10.

Parts 5 to 10 specify the six profiles which are defined, based on the OSI TP standard. These six parts make reference to Parts 2 to 4.

This page is intentionally left blank

**Information Technology - Open Systems Interconnection -
International Standardized Profile 12061-3: OSI Distributed
Transaction Processing.**

Part 3: Support of CCR APDUs

1. SCOPE

This part of this ISP specifies the status for the support of the CCR protocol for the profiles identified in Part 1 of this ISP.

2. NORMATIVE REFERENCES

The references listed in Part 1 of this ISP applies.

3. DEFINITIONS AND ABBREVIATIONS

The Definitions and Abbreviations listed in Part 1 of this ISP applies.

4. NOTATION

The notation described in PART 1 of this ISP Applies.

5. SUPPORT OF CCR APDUs

Annex A specifies the support of CCR protocol.

It applies to profiles 21, 22, 31 and 32. It does not apply to profiles 11 and 12.

6. CONFORMANCE

To conform to the OSI CCR protocol used in any of the profiles defined in this ISP, an implementation shall implement, according to the specifications given in ISO/IEC 9805:

- All mandatory features identified in Annex A.
- All selected optional features, as identified in the completed CCR PICS.

ANNEX A: CCR PDU Supports (Normative)

Temporary Editor's Note: This current Annex A is based on a version of the CCR PICS Proforma which is based on Version 1 of the CCR protocol. It is expected that the CCR PICS Proforma will be aligned to Version 2. This part of the OSI TP ISP will be updated accordingly during the DISP ballot period. Would the CCR PICS Proforma not be aligned on time, the necessary material will be inserted in the OSI TP ISP.

A.1 DATE OF STATEMENT

No restrictions applied to clause A.1 of ISO/IEC 9805-2 by this ISP.

A.2 IMPLEMENTATION DETAILS

No restrictions applied to clause A.2 of ISO/IEC 9805-2 by this ISP.

A.3 ISO/IEC 9805-1

The answer shall be "Version 2".

A.4 AMENDMENTS IMPLEMENTED

Table 1 - AMENDMENTS IMPLEMENTED

ITEM #	Amendment	Profiles						
		11	21	31	12	22	32	Notes
1	ISO/IEC 9805-1 Amendment 1	NA	NA	NA	NA	NA	NA	
2	ISO/IEC 9805-1 Amendment 2	NA	M	M	NA	M	M	

A.5 TECHNICAL CORRIGENDA IMPLEMENTED

The answer shall be "None".

At the time of approval of the final text of this ISP no technical corrigenda was approved for ISO/IEC 9805. When this condition changes, the present ISP will be amended.

A.6 GLOBAL STATEMENT OF CONFORMANCE

A.6.1 MANDATORY FEATURES IMPLEMENTED

The answer shall be "Yes".

A.7 INITIATOR/RESPONDER CAPABILITIES

A.7.1 ATOMIC-ACTION-BRANCH ESTABLISHMENT

Table 2 - ATOMIC-ACTION-BRANCH ESTABLISHMENT BY PROFILE

ITEM #	Roles	ISO/IEC 9805-2	Profiles						
			11	21	31	12	22	32	Notes
1	SUPERIOR	O	NA	C101	C101	NA	C101	C101	1
2	SUBORDINATE	O	NA	C102	C102	NA	C102	C102	1
3	MASTER	O	NA	C103	C103	NA	C103	C103	

101 If capable of acting as a root node or intermediate node then M else I.

102 If capable of acting as a leaf node or intermediate node then M else I.

103 If capable of acting as a root node then M else I.

NOTES

- At least one of the superior or subordinate roles must be implemented.

A.7.2 SUPPORT FOR THE CONCATENATION MECHANISM

Table 3 - SUPPORT FOR THE CONCATENATION MECHANISM

ITEM #	Roles	ISO/IEC 9805-2	Profiles						
			11	21	31	12	22	32	Notes
1	SENDER	O	NA	O	M	NA	O	M	
2	RECEIVER	M	NA	M	M	NA	M	M	

Temporary note The detail of the markings for row 1, sender role may require further study. A draft technical corrigendum to ISO/IEC 9805:1990 adds procedures for the combined use of the C-COMMIT and C-BEGIN services, and the C-ROLLBACK and C-BEGIN services. Support for concatenation for these combined services may not be optional. An issue is that TP makes no use of the C-ROLLBACK and C-BEGIN services. This issue may apply to ISO/IEC DIS 9805-2 (CCR PICS Proforma).

A.7.3 OTHER IMPLEMENTATION CAPABILITIES

No restriction is applied to clause A.7.3 of ISO/IEC 9805-2 by this ISP.

A.8 CCR PROTOCOL - GENERAL

This subclause details TP's requirements of the CCR protocol. The protocol tables described below, except for the CCR PDU Usage by Profile, **do not apply** to TP profiles 11 and 12.

A.9 CCR PROTOCOL

A.9.1 CCR PDUs

This table specifies the support level of each PDU with respect to each profile.

Table 4 - CCR PDU USAGE BY PROFILE

ITEM #	Protocol Data Units	ISO/IEC 9805-2	Profiles						
			11	21	31	12	22	32	Notes
1	C-BEGIN-RI	C	NA	C104 /C105	C104 /C105	NA	C104 /C105	C104 /C105	
2	C-BEGIN-RC	O/C	NA	C105 /C104	C105 /C104	NA	C105 /C104	C105 /C104	
3	C-PREPARE-RI	O/C	NA	C104 /C105	C104 /C105	NA	C104 /C105	C104 /C105	
4	C-READY-RI	C	NA	C105 /C104	C105 /C104	NA	C105 /C104	C105 /C104	
5	C-COMMIT-RI	C	NA	C104 /C105	C104 /C105	NA	C104 /C105	C104 /C105	
6	C-COMMIT-RC	C	NA	C105 /C104	C105 /C104	NA	C105 /C104	C105 /C104	
7	C-ROLLBACK-RI	M	NA	M	M	NA	M	M	
8	C-ROLLBACK-RC	M	NA	M	M	NA	M	M	
9	C-RECOVER-RI	M	NA	M	M	NA	M	M	
10	C-RECOVER-RC	M	NA	M	M	NA	M	M	

104. If capable of acting in the role of superior then M, else NA.

105. If capable of acting in the role of subordinate then M, else NA.

A.9.2 C-BEGIN-RI

Table 5 - C-BEGIN-RI

BASE STANDARD ISO/IEC 9805-2			PROFILE			
ITEM#	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	Atomic Action Identifier	M		M	See Table 6	
2	Branchd-Suffix - Octet String	O/M		O/M	1 .. 64 octets	1
	Branch-Suffix - Integer	O/M		O/M	0..2**31-1	1
3	User Data	O/M		NA		

NOTES

1. At least one of these forms must be supported.

9.2.1. ATOMIC-ACTION IDENTIFIER

This clause provides detail about the Atomic-Action Identifier field of the C-BEGIN APDU.

Table 6 - ATOMIC-ACTION IDENTIFIER DETAIL

		Atomic-Action Identifier of the C-BEGIN APDU				
		PROFILE				
ITEM#	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	Master's Name AE-Title-Form 1 (Directory name)	C/M		O/M	1 .. 1024 octets	1,2
	Master's Name AE-Title-Form 2 (Object Id)	C/M		M	1 .. 64 octets	2
2	Atomic Action Id.- Suffix - Octet String	C/M		C106/M	1 .. 64 octets	2
	Atomic Action Id.- Suffix - Integer	C/M		C106/M	0..2**31-1	

106 If only the Master role is supported then at least one form shall be supported, otherwise both forms shall be supported.

NOTES

1. It is optional to be able to generate a Master's name AE-TITLE-FORM-1 (RDN), but it is mandatory to be able to propagate it when received from a superior by an intermediate node
2. The maximum length of the Atomic-Action Identifier shall be 1024 octets for Form 1 (Directory Name) and 64 octets for Form 2 (object ID). This length includes both Master's Name and suffix.

A.9.3 C-BEGIN-RC

Table 7 - C-BEGIN-RC

		ISO/IEC 9805-2		PROFILE		
ITEM#	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	User-data	O/M		NA		

A.9.4 C-PREPARE-RI

Table 8 - C-PREPARE-RI

	ISO/IEC 9805-2		PROFILE			
ITEM#	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	User-data	O/M		M		

A.9.5 C-READY-RI

Table 9 - C-READY-RI

	ISO/IEC 9805-2		PROFILE			
ITEM#	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	User-data	O/M		NA		

A.9.6 C-COMMIT-RI

Table 10 - C-COMMIT-RI

	ISO/IEC 9805-2		PROFILE			
ITEM#	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	User-Data	O/M		M		

A.9.7 C-COMMIT-RC

Table 11 - C-COMMIT-RC

	BASE STANDARD ISO/IEC 9805-2		PROFILE			
ITEM#	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	User-data	O/M		M		

A.9.8 C-ROLLBACK-RI

Table 12 - C-ROLLBACK-RI

	ISO/IEC 9805-2		PROFILE			
ITEM#	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
	User-data	O/M		M		

A.9.9 C-ROLLBACK-RC

Table 13 - C-ROLLBACK-RC

	ISO/IEC 9805-2		PROFILE			
ITEM#	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	User-data	O/M		M		

A.9.10 C-RECOVER-RI

Table 14 - C-RECOVER-RI

	BASE STANDARD ISO/IEC 9805-2		PROFILE			
ITEM#	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	Atomic Action Identifier	M		M	See Table 15	
2	Branch Identifier	M		M	See Table 16	
3	Recovery State	M		M	See Table 17	
4	User Data	O/M		M		

A.9.10.1. ATOMIC-ACTION IDENTIFIER

This clause provides detail about the Atomic-Action Identifier field of the C-RECOVER RI APDU.

Table 15 - ATOMIC-ACTION IDENTIFIER DETAIL

ITEM#	Atomic-Action Identifier of the C-RECOVER-RI APDU		PROFILE			
	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	Master's Name AE-Title-Form 1 (Directory name)	C		O/M	1 .. 1024 octets	1
	Master's Name AE-Title-Form 2 (Object Id)	C		M	1 .. 64 octets	1
2	Atomic Action Id.- Suffix - Octet String	C		M	1 .. 64 octets	1
	Atomic Action Id.- Suffix - Integer	C		M	0..2**31-1	

NOTES

1. The maximum length of the Atomic-Action Identifier shall be 1024 octets for Form 1 (Directory Name) and 64 octets for Form 2 (object ID). This length includes both Master's Name and suffix.

A.9.10.2. BRANCH IDENTIFIER

This clause provides detail about the Branch Identifier field of the C-RECOVER RI APDU.

Table 16 - BRANCH IDENTIFIER DETAIL

		Branch Identifier of the C-RECOVER-RI APDU		PROFILE		
ITEM#	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	Superior's-Name AE-Title-Form 1 (Directory name)	C		O/M	1 .. 1024 octets	1
	Superior's-Name AE-Title-Form 2 (Object Id)	C		M	1 .. 64 octets	1
2	Branch-- Suffix - Octet String	C		M	1 .. 64 octets	1
	Branch-- Suffix - Integer	C		M	0 .. 2**31-1	

NOTES

1. The maximum length of the Branch Identifier shall be 1024 octets for Form 1 (Directory Name) and 64 octets for Form 2 (object ID). This length includes both Superior's Name and suffix.

A.9.10.3. RECOVERY STATE

This clause provides detail about the RECOVERY-STATE field of the C-RECOVER RI APDU.

Table 17 - RECOVERY-STATE

Recovery-State of the C-RECOVER-RI APDU			PROFILE			
ITEM#	STATE	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	Commit	C		C101/C102		
2	Ready	C		C102/C101		

A.9.11 C-RECOVER-RC

Table 18 - C-RECOVER-RC

BASE STANDARD ISO/IEC 9805-2			PROFILE			
ITEM#	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	Atomic Action Identifier	M		M	See Table 19	
2	Branch Identifier	M		M	See Table 20	
3	Recovery State	M		M	See Table 21	
4	User Data	O/M		M		

A.9.11.1. ATOMIC-ACTION IDENTIFIER

This clause provides detail about the Atomic-Action Identifier field of the C-RECOVER RC APDU.

Table 19 - ATOMIC-ACTION IDENTIFIER DETAIL

		Atomic-Action Identifier of the C-RECOVER-RC APDU					PROFILE			
ITEM#	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES				
1	Master's Name AE-Title-Form 1 (Directory name)	C		O/M	1 .. 1024 octets	1				
	Master's Name AE-Title-Form 2 (Object Id)	C		M	1 .. 64 octets	1				
2	Atomic Action Id.- Suffix - Octet String	C		M	1 .. 64 octets	1				
	Atomic Action Id.- Suffix - Integer	C		M	0 .. 2**31-1					

NOTES

1. The maximum length of the Atomic Action Identifier shall be 1024 octets for Form 1 (Directory Name) and 64 octets for Form 2 (object ID). This length includes both Master's Name and suffix.

A.9.11.2. BRANCH IDENTIFIER

This clause provides detail about the Branch Identifier field of the C-RECOVER RC APDU.

Table 20 - BRANCH IDENTIFIER DETAIL

		Branch Identifier of the C-RECOVER-RC APDU		PROFILE		
ITEM#	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	Superior's-Name AE-Title-Form 1 (Directory name)	C		O/M	1 .. 1024 octets	1
	Superior's-Name AE-Title-Form 2 (Object Id)	C		M	1 .. 64 octets	1
2	Branch-- Suffix - Octet String	C		M	1 .. 64 octets	1
	Branch-- Suffix - Integer	C		M	0 .. 2**31-1	

NOTES

1. The maximum length of the Branch Identifier shall be 1024 octets for Form 1 (Directory Name) and 64 octets for Form 2 (object ID). This length includes both Superior's Name and suffix.

A.9.11.3. RECOVERY STATE

This clause provides detail about the RECOVERY-STATE field of the C-RECOVER RC APDU.

Table 21 - RECOVERY-STATE

		Recovery-State of the C-RECOVER-RC APDU		PROFILE		
ITEM#	STATE	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	Done	C		C102 /C101		
2	Unknown	C		C101 /C102		
3	Retry-later	M		M		

TITLE: WORKING DOCUMENT FOR

Information Technology - Open Systems Interconnection - International Standardized Profile
12061-4: OSI TP

Part 4: Support of Session, Presentation and ACSE Protocols.

SOURCE: Joint AOW / EWOS / OIW on Transaction Processing

DATE: December 18, 1992

STATUS: This document has been harmonized among the three workshops (AOW/EWOS/OIW) and has been submitted to SGFS for progression to ISP.

CONTENTS

INTRODUCTION

1. SCOPE
2. NORMATIVE REFERENCES
3. DEFINITIONS and ABBREVIATIONS
4. NOTATION
5. SUPPORT OF OSI SESSION PROTOCOL
6. SUPPORT OF OSI PRESENTATION PROTOCOL
7. SUPPORT OF OSI ACSE PROTOCOL
8. CONFORMANCE

ANNEX

- A. SUPPORT OF THE SESSION PROTOCOL (NORMATIVE)
- B. SUPPORT OF THE PRESENTATION PROTOCOL (NORMATIVE)
- C. SUPPORT OF THE ACSE PROTOCOL (NORMATIVE).

INTRODUCTION

The aim of Open Systems Interconnection is to allow, with a minimum of technical agreement outside the interconnection standards, the interconnection of computer systems

- a. from different manufacturers,
- b. under different management,
- c. of different levels of complexity,
- d. of different technologies.

Transaction Processing is concerned with identifiable information which can be related as transactions, which may involve two or more Open Systems. In the framework of Open Systems Interconnection (OSI) a transaction is defined as "a set of related operations characterized by four properties: atomicity, consistency, isolation and durability."

The definition highlights that a distributed transaction is more than a simple exchange of messages, but that the exchanges form a protected indivisible set.

This multi-part document contains the complete specification of the six profiles identified in M-IT-02 and TR 10000.

Part 1 contains the taxonomy for the OSI TP profiles.

Part 2 contains the specification of the support of OSI TP APDUs for each of the profiles specified in Parts 5 to 10.

Part 3 contains the specification of the support of the CCR APDUs for each of the profiles specified in Part 5 to 10.

Part 4 contains the specification of the support of Session , Presentation and ACSEAPDUs for each of the profiles specified in Part 5 to 10.

Parts 5 to 10 specify the six profiles which are defined, based on the OSI TP standard. These six parts make reference to Parts 2 to 4.

This page is intentionally left blank

Information Technology - Open Systems Interconnection - International Standardized Profile 12061-4: OSI Distributed Transaction Processing.

Part 4: Support of SESSION, PRESENTATION AND ACSE PDUs

1. SCOPE

This part of this ISP specifies the status for the support of the Session, Presentation and ACSE protocols for the profiles identified in Part 1 of this ISP.

2. NORMATIVE REFERENCES

The references listed in Part 1 of this ISP.

3. DEFINITIONS AND ABBREVIATIONS

The definitions and abbreviations contained in Part 1 of this ISP apply to this part.

4. NOTATION

The notation described in PART 1 of this ISP Applies.

5. SUPPORT OF SESSION SPDUs

Annex A specifies the support of Session protocol.

6. SUPPORT OF PRESENTATION PPDUs

Annex B specifies the support of Presentation protocol.

7. SUPPORT OF ACSE APDUs

Annex C specifies the support of ACSE protocol.

8. CONFORMANCE

To conform to the OSI ACSE protocol used in any of the profiles defined in this ISP, an implementation shall implement, according to the specifications given in ISO/IEC 8650:

- All mandatory features identified in Annex C.

- All selected optional features, as identified in the completed ACSE PICS.
- All restrictions as specified in the Common Upper Layer Requirements
ISP - ISO 11188-1.

To conform to the OSI Presentation protocol used in any of the profiles defined in this ISP, an implementation shall implement, according to the specifications given in ISO/IEC 8823:

- All mandatory features identified in Annex B.
- All selected optional features, as identified in the completed Presentation PICS.
- All restrictions as specified in the Common Upper Layer Requirements
ISP - ISO 11188-1.

To conform to the OSI Session protocol used in any of the profiles defined in this ISP, an implementation shall implement, according to the specifications given in ISO/IEC 8327:

- All mandatory features identified in Annex A.
- All selected optional features, as identified in the completed Session PICS.
- All restrictions as specified in the Common Upper Layer Requirements
ISP - ISO 11188-1.

ANNEX A: Session PROTOCOL PDUs (Normative)

This subclause details TP's requirements on the Session protocol. The reader should consult the Upper Layer agreements for a detailed discussion of these services. This ISP only specifies PDU parameters necessary for this ISP.

A.1 SUPPORTED FUNCTIONS

Table 1 - SUPPORTED FUNCTIONS

ITEM#	BASE STANDARD ISO/IEC 8327-2		PROFILE		
	CAPABILITY	STATUS	PROFILE ID	STATUS	NOTES
1	Kernel	M		M	
2	Negotiated Release	O		*(I)	
3	Half Duplex	O.n		NA	3
4	Duplex	O.n		M	
5	Expedited Data	O		*(I)	
6	Typed Data	O	11,12	*(I)	
			21,22,31,32	M	
7	Capability Data Exchange	C	11,12	*(I)	
			21,22,31,32	NA	3
8	Minor Synchronize	O	11,12	*(I)	
			21,22,31,32	M	
9	Symmetric Synchronize	O	11,12	*(I)	
			21,22,31,32	NA	3
10	Major Synchronize	O		*(I)	
11	Resynchronize	O	11,12	*(I)	
			21,22,31,32	M	
12	Exceptions	C		NA	1,3
13	Activity Management	O	11,12	*(I)	
			21,22,31,32	NA	2,3
14	Data Separation	C	11,12	*(I)	
			21,22,31,32	M	

NOTES

- Exceptions FU cannot be negotiated because Half Duplex is not allowed.
- Activity Management FU cannot be negotiated for these profiles because the Data Separation FU does not allow the Activity Management FU to also be selected.
- Successfully accepting these functional units is a protocol error. If any of the following Functional Units is proposed on a CN SPDU the Functional Unit shall not be accepted and the corresponding bit shall be set to zero on the Accept SPDU.

A.2 ISO 8327 Protocol Versions Implemented

Table 2 -ISO 8327 PROTOCOL VERSIONS IMPLEMENTED

ITEM#	BASE STANDARD ISO/IEC 8327 -2		PROFILE		
	CAPABILITY	STATUS	PROFILE ID	STATUS	NOTES
1	Version 1	O		I	
2	Version 2	O		M	

A.3 PROTOCOL MECHANISMS

Table 3- PROTOCOL MECHANISMS

ITEM#	ISO/IEC 8327-2		PROFILE		
	CAPABILITY	STATUS	PROFILE ID	STATUS	NOTES
1	Use of Transport Expedited Data	O		O	
2	Reuse of Transport Connection	O		* (I)	
3	Basic Concatenation	M		M	
4	Extended Concatenation	O		I	
5	Segmentation	O		* (I)	
6	Segmentation of Unlimited User Data	O		* (I)	

A.4 INITIATOR/RESPONDER CAPABILITIES

Table 4 - INITIATOR/RESPONDER CAPABILITIES

ITEM#	ISO/IEC 8327-2		PROFILE		
	CAPABILITY	STATUS	PROFILE ID	STATUS	NOTES
1	Initiator	O		C101	
2	Responder	O		M	

101. If capable of initiating an Association then M, else I.

A.5 SESSION PROCEDURES USAGE BY PROFILE

This table specifies the supported level of each Session PDU with respect to each profile.

Table 5 - KERNEL FUNCTIONAL UNIT PROCEDURES USAGE BY PROFILE

ITEM #	Protocol Data Units	ISO/IEC 8327-2	Profiles						
			11	21	31	12	22	32	Notes
1	Connect (CN)	C/C	C103 /M	C103 /M	C103 /M	C103 /M	C103 /M	C103 /M	
2	Overflow Accept(OA)	C	I	I	I	I	I	I	
3	Connect Data Overflow(CDO)	C	I	I	I	I	I	I	
4	Accept(AC)	C/C	C104 /C103	C104 /C103	C104 /C103	C104 /C103	C104 /C103	C104 /C103	
5	Refuse(RF)	C/C	M /C103	M /C103	M /C103	M /C103	M /C103	M /C103	
6	Finish(FN)	O/C	O/M	O/M	O/M	O/M	O/M	O/M	

Table 5 - continued

ITEM #	Protocol Data Units	ISO/IEC 8327-2	Profiles						
			11	21	31	12	22	32	Notes
7	Disconnect(DN)	O	M /C105	M /C105	M /C105	M /C105	M /C105	M /C105	
8	Abort(AB)	M	M	M	M	M	M	M	
9	Abort Accept(AA)	O	C106	C106	C106	C106	C106	C106	
10	Data Transfer(DT)	O/C	M	M	M	M	M	M	
11	Prepare(PR)	C/C	C107	C107	C107	C107	C107	C107	

103. If capable of initiating an Association then M, else I.

104. If capable of responding to a AARQ APDU then M, else I.

105. If capable of initiating a FINISH then M, else NA.

106. If reusing T-Connection then M, else I.

107. If transport expedited available then M, else NA.

Table 6 - NEGOTIATED RELEASE FUNCTIONAL UNIT PROCEDURES USAGE BY PROFILE

ITEM #	Protocol Data Units	ISO/IEC 8327-2	Profiles						
			11	21	31	12	22	32	Notes
1	Not Finished(NF)	O/M	*	*	*	*	*	*	
2	Give Tokens(GT)	O	*	*	*	*	*	*	1
3	Please Tokens(PT)	O/M	*	*	*	*	*	*	1

NOTES

- These PDUs are marked * in this table because the Negotiated Release FU is marked * in Table 1. These PDUs may be used else where in different ways.

Table 7 - HALF DUPLEX FUNCTIONAL UNIT PROCEDURES USAGE BY PROFILE

ITEM #	Protocol Data Units	ISO/IEC 8327-2	Profiles						
			11	21	31	12	22	32	Notes
1	Give Tokens(GT)	O	NA	NA	NA	NA	NA	NA	1
2	Please Tokens(PT)	O/M	NA	NA	NA	NA	NA	NA	1

NOTES

- These PDUs are marked NA in this table because the Half Duplex FU is marked NA in Table 1. These PDUs may be used elsewhere in different ways.

DUPLEX FUNCTIONAL UNIT PROCEDURES

No additional SPDUs (this clause is present for completeness).

Table 8 - EXPEDITED DATA FUNCTIONAL UNIT PROCEDURES USAGE BY PROFILE

ITEM #	Protocol Data Units	ISO/IEC 8327-2	Profiles						
			11	21	31	12	22	32	Notes
1	Expedited Data(EX)	O/M	*	*	*	*	*	*	

Table 9 - TYPED DATA FUNCTIONAL UNIT PROCEDURES USAGE BY PROFILE

ITEM #	Protocol Data Units	ISO/IEC 8327-2	Profiles						
			11	21	31	12	22	32	Notes
1	Typed Data(TD)	O/M	*	M	M	*	M	M	

Table 10 - CAPABILITY DATA FUNCTIONAL UNIT PROCEDURES USAGE BY PROFILE

ITEM #	Protocol Data Units	ISO/IEC 8327-2	Profiles						
			11	21	31	12	22	32	Notes
1	Capability Data(CD))	O/M	*	NA	NA	*	NA	NA	1
2	Capability Data Ack(CDA)	M/C	*	NA	NA	*	NA	NA	1

NOTES

1. Because the Capability Data FU shall never be selected on a Session Connection for Profiles 21, 22, 31, and 32, the Session Protocol Machine will generate a protocol error when a CD or CDA SPDU is received in these profiles.

Table 11 -MINOR SYNCHRONIZE FUNCTIONAL UNIT PROCEDURES USAGE BY PROFILE

ITEM #	Protocol Data Units	ISO/IEC 8327-2	Profiles						
			11	21	31	12	22	32	Notes
1	Minor Sync Point(MIP)	O	*	M	M	*	M	M	
2	Minor Sync Point Ack(MIA)	O/C	*	M	M	*	M	M	
3	Give Tokens(GT)	O	*	M	M	*	M	M	
4	Please Tokens(PT)	O/M	*	M	M	*	M	M	

Table 12 -SYMMETRIC SYNCHRONIZE FUNCTIONAL UNIT PROCEDURES USAGE BY PROFILE

ITEM #	Protocol Data Units	ISO/IEC 8327-2	Profiles						
			11	21	31	12	22	32	Notes
1	Minor Sync Point(MIP)	O	*	NA	NA	*	NA	NA	1
2	Minor Sync Point Ack(MIA)	O/C	*	NA	NA	*	NA	NA	1

NOTES

1. Because the Symmetric Synchronize FU shall never be selected on a Session Connection for Profiles 21, 22, 31, and 32, the MIP and MIA SPDUs have been marked NA. They may be used differently elsewhere.

Table 13 -MAJOR SYNCHRONIZE FUNCTIONAL UNIT PROCEDURES USAGE BY PROFILE

ITEM #	Protocol Data Units	ISO/IEC 8327-2	Profiles						
			11	21	31	12	22	32	Notes
1	Major Sync Point(MAP)	O/M	*	*	*	*	*	*	
2	Major Sync Point Ack(MAA)	M/C	*	*	*	*	*	*	
3	Give Tokens(GT)	O	*	*	*	*	*	*	1
4	Please Tokens(PT)	O/M	*	*	*	*	*	*	1
5	Prepare(PR)	C/C	*	*	*	*	*	*	1

NOTES

1. Because the Major Synchronize FU has been marked * in Table 1, these PDUs have been marked *. They may be used differently elsewhere.

Table 14 -RESYNCHRONIZE FUNCTIONAL UNIT PROCEDURES USAGE BY PROFILE

ITEM #	Protocol Data Units	ISO/IEC 8327-2	Profiles						
			11	21	31	12	22	32	Notes
1	RESYNCHRONIZE(RS)	O/M	*	M	M	*	M	M	1
2	RESYNCHRONIZE Ack(RA)	M/C	*	M	M	*	M	M	1
3	Prepare(PR)	C/C	*	C107	C107	*	C107	C107	1

NOTES

1. Because the ReSynchronize FU has been marked in Table 1 with * for Profiles 11 and 12 these PDUs have been marked with * in this table. They may be used differently elsewhere.

Table 15 -EXCEPTIONS FUNCTIONAL UNIT PROCEDURES USAGE BY PROFILE

ITEM #	Protocol Data Units	ISO/IEC 8327-2	Profiles						
			11	21	31	12	22	32	Notes
1	Exception Report(ER)	O/M	NA	NA	NA	NA	NA	NA	1
2	Exception Data(ED)	O/M	NA	NA	NA	NA	NA	NA	1

NOTES

1. Because the ExceptionFU shall never be selected for a Session Connection, the Session Protocol Machine will generate a protocol error when an ER or ED SPDU is received.

Table 16 -ACTIVITY MANAGEMENT FUNCTIONAL UNIT PROCEDURES USAGE BY PROFILE

ITEM #	Protocol Data Units	ISO/IEC 8327-2	Profiles						
			11	21	31	12	22	32	Notes
1	Activity Start(AS)	O/M	*	NA	NA	*	NA	NA	2
2	Activity Resume(AR)	O/M	*	NA	NA	*	NA	NA	2
3	Activity Interrupt(AI)	O/M	*	NA	NA	*	NA	NA	2
4	Activity Interrupt Ack(AIA)	M/C	*	NA	NA	*	NA	NA	2
5	Activity Discard(AD)	O/M	*	NA	NA	*	NA	NA	2
6	Activity Discard Ack(ADA)	M/C	*	NA	NA	*	NA	NA	2
7	Activity End(AE)	O/M	*	NA	NA	*	NA	NA	2
8	Activity End Ack(AEA)	M/C	*	NA	NA	*	NA	NA	2
9	Prepare(PR)	C/C	*	NA	NA	*	NA	NA	2,3
10	Give Tokens(GT)	O	*	NA	NA	*	NA	NA	2,3
11	Please Tokens(PT)	O/M	*	NA	NA	*	NA	NA	2,3
12	Give Tokens Confirm(GTC)	O/M	*	NA	NA	*	NA	NA	1,2
13	Give Tokens Confirm Ack(GTA)	O/M	*	NA	NA	*	NA	NA	1,2

NOTES

1. The Give tokens confirm and Ack are a result of the control give service and require the activity management FU.
2. Because the Activity Management FU shall never be selected on a Session connection for profiles 21, 22, 31 and 32, the Session Protocol Machine will generate a protocol error when the SPDU is received in these Profiles.
3. Because the Activity Management FU has been marked in Table 1 with * or NA the PDUs have been marked with * or NA in this table. They may be used differently elsewhere.

A.6 SUPPORTED PARAMETERS OF SESSION PDUs.

A.6.1 CONNECT (CN) SPDU

Table 17 - CONNECT (CN) SPDU

ISO/IEC 8327-2			PROFILE			
ITEM#	PARAMETER	STATUS	PROFILE ID	STATUS	T/LV ALLOWED	NOTES
PGI Connection Identifier						
1	PGI default (absent)	O/M		I		
2	PGI default (empty)	O/M		I		
3	Calling SS-User Reference	O/M		I		
4	Common Reference	O/M		I		
5	Additional Reference Info	O/M		I		
PGI Connect/Accept Item						
6	PGI default (absent)	O/M		I		
7	PGI default (empty)	O/M		I		
8	PGI default (not empty)	O/M		M		
9	Protocol Options	C/M		I/M		
10	TSDU-Maximum-size	O/M		I/M		
11	Version Number	O/M		M	Version 2	
12	Initial Serial Number	O/M	11,12	*		
			21,22,31,32	M		
13	Token Setting Item	O/M	11,12	*		
			21,22,31,32	M		
14	Second Initial Serial Number	C/C	11,12	C108		
			21,22,31,32	I		1

Table 17 - continued

ITEM#	ISO/IEC 8327-2		PROFILE			
	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
	Single Items					
15	Session User Requirements	O/M		M		
16	Calling Session Selector	O/M		M		
17	Called Session Selector	O/M		M		
18	PGI "User Data"	O/M		M		
19	Data Overflow	C/C		I		
20	PGI "Extended User Data"	C/C		M		

108. If Symmetric-Sync supported then O else I.

NOTES

1. Because the Symmetric Synchronize FU shall never be selected for a Session connection for Profiles 21, 22, 31 and 32, the Session Protocol machine will ignore the Second Initial Serial Number parameter if it is present on a CN SPDU in Profiles 21, 22, 31, and 32.

A.6.2 OVERFLOW ACCEPT (OA) SPDU

This SPDU is not used by TP. Its use by a U-ASE is not specified by this ISP.

A.6.3 CONNECT DATA OVERFLOW (CDO) SPDU

This SPDU is not used by TP. Its use by a U-ASE is not specified by this ISP.

A.6.4 ACCEPT (AC) SPDU

Table 18 - ACCEPT(AC) SPDU

	ISO/IEC 8327-2		PROFILE			
ITEM#	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
	PGI Connection Identifier					
1	PGI default (absent)	O/M		I		
2	PGI default (empty)	O/M		I		
3	Calling SS-User Reference	O/M		I		
4	Common Reference	O/M		I		
5	Additional Reference Info	O/M		I		
	PGI Connect/Accept Item					
6	PGI default (absent)	O/M		NA		1
7	PGI default (empty)	O/M		NA		1
8	PGI default (not empty)	O/M		M		
9	Protocol Options	C/M		I		
10	TSDU-Maximum-size	O/M		I/M		
11	Version Number	C/M		M	Version 2	
12	Initial Serial Number	O/M	11,12	*		
			21,22,31,32	M		
13	Token Setting Item	O/M	11,12	*		
			21,22,31,32	M		
14	Second Initial Serial Number	C/C	11,12	C108		
			21,22,31,32	I		2
	Single Items					
15	Token Item	O/M		O/M		
16	Session User Requirements	O/M		M		
17	Calling Session Selector	O/M		M		
18	Called Session Selector	O/M		M		
19	PGI "User Data"	O/M		M		
20	Enclosure Item	C		*		

NOTES

1. Because Session Version 2 shall be selected, the Session Protocol Machine will generate a protocol error when the PGI Connect/Accept item is absent or empty.
2. Because the Symmetric Synchronize FU shall never be selected for a Session connection for Profiles 21, 22, 31 and 32, the Session Protocol Machine will ignored the Second Initial Serial Number parameter if it is present on an AC SPDU in Profiles 21, 22, 31, and 32.

A.6.5 REFUSE (RF) SPDU

Table 19 - REFUSE (RF) SPDU

ITEM#	ISO/IEC 8327-2		PROFILE			
	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	PGI default (empty)	O/M		I		
2	PGI default (not empty)	O/M		I		
3	Called SS-User Reference	O/M		I		
4	Common Reference	O/M		I		
5	Additional Reference Info	O/M		I		
	Single Items					
6	Transport Disconnect	O/M		O/M		
9	Session User Requirements	O/M		M		
10	Version Number	O/M		M		
11	Reason Code	O/M		M		
12	Enclosure Item	C		*		

A.6.6 FINISH (FN) SPDU

Table 20 - FINISH (FN) SPDU

ITEM#	ISO/IEC 8327-2		PROFILE			
	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	Transport Disconnect	O/M		O/M		
2	PGI "User Data"	O/M		M		
3	Enclosure Item	C/C		*		

A.6.7 DISCONNECT (DN) SPDU

Table 21 - DISCONNECT (DN) SPDU

ITEM#	ISO/IEC 8327-2		PROFILE			
	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	PGI "User Data"	O/M		M		
2	Enclosure Item	C/C		*		

A.6.8 NOT FINISHED (NF) SPDU

This SPDU is not used by TP. Its use by a U-ASE is not specified by this ISP.

A.6.9 ABORT (AB) SPDU

Table 22 - ABORT (AB) SPDU

ITEM#	ISO/IEC 8327-2		PROFILE			
	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	Transport Disconnect	M		M		
2	Reflect Parameter Values	O/M		I		
3	PGI "User Data"	O/M		M		
4	Enclosure Item	C/C		*		

A.6.10 ABORT ACCEPT (AA) SPDU

This SPDU is not used by TP. Its use by a U-ASE is not specified by this ISP.

A.6.11 DATA TRANSFER (DT) SPDU

Table 23 - DATA TRANSFER (DT) SPDU

ITEM#	ISO/IEC 8327-2		PROFILE			
	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	User Information Field	O/M		M		
2	Enclosure Item	C/C		*		

A.6.12 EXPEDITED DATA (EX) SPDU

This SPDU is not used by TP. Its use by a U-ASE is not specified by this ISP.

A.6.13 TYPED DATA (TD) SPDU

Table 24 - TYPED DATA (TD) SPDU

ITEM#	ISO/IEC 8327-2		PROFILE			
	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	Enclosure Item	C/C		*		
2	User Information Field	O/M		M		

A.6.14 CAPABILITY DATA (CD) SPDU

This SPDU is not used by TP. Its use by a U-ASE in profiles 11 and 12 is not specified by this ISP; however, its use by a U-ASE in profiles 21,22,31 and 32 is prohibited

A.6.15 CAPABILITY DATA ACK (CDA) SPDU

This SPDU is not used by TP. Its use by a U-ASE in profiles 11 and 12 is not specified by this ISP; however, its use by a U-ASE in profiles 21,22,31 and 32 is prohibited

A.6.16 GIVE TOKENS (GT) SPDU

This PDU is not used by Profiles 11 and 12.

Table 25 - GIVE TOKENS (GT) SPDU

ISO/IEC 8327-2			PROFILE			
ITEM#	PARAMETER	STATUS	PROFILE ID	STATUS	T/LV ALLOWED	NOTES
1	Token Item	O/M		M	Minor Sync	
				*	Other Tokens	
2	PGI "User Data"	C/C		M		
3	Enclosure Item	C/C		*		

A.6.17 PLEASE TOKENS (PT) SPDU

This PDU is not used by Profiles 11 and 12.

Table 26 - PLEASE TOKENS (PT) SPDU

ISO/IEC 8327-2			PROFILE			
ITEM#	PARAMETER	STATUS	PROFILE ID	STATUS	T/LV ALLOWED	NOTES
1	Token Item	O/M		M	Minor Sync	
				*	Other Tokens	
2	PGI "User Data"	O/M		M		
3	Enclosure Item	C/C		*		

A.6.18 MINOR SYNC POINT (MIP) SPDU

This PDU is not used by Profiles 11 and 12.

Table 27 - MINOR SYNC POINT (MIP) SPDU

ITEM#	ISO/IEC 8327-2		PROFILE			
	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	Sync Type Item	O/M		M		
2	Serial Number	M		M		
3	PGI "User Data"	O/M		M		
4	Enclosure Item	C		M		

A.6.19 MINOR SYNC POINT ACK (MIA) SPDU

This PDU is not used by Profiles 11 and 12.

Table 28 - MINOR SYNC POINT ACK (MIA) SPDU

ITEM#	ISO/IEC 8327-2		PROFILE			
	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	Serial Number	M		M		
2	PGI "User Data"	O/M		M		
3	Enclosure Item	C/C				

A.6.20 MAJOR SYNC POINT (MAP) SPDU

This SPDU is not used by TP. Its use by a U-ASE is not specified by this ISP.

A.6.21 MAJOR SYNC POINT ACK (MAA) SPDU

This SPDU is not used by TP. Its use by a U-ASE is not specified by this ISP.

A.6.22 RESYNCHRONIZE (RS) SPDU

This PDU is not used by TP for profiles 11 and 12.

Table 29 - RESYNCHRONIZE (RS) SPDU

ISO/IEC 8327-2			PROFILE			
ITEM#	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	Enclosure Item	C		*		
2	Token Setting Item	O/M		M	Minor Sync	
				*	Other Tokens	
3	Resync Type	O/M		M		
4	Serial Number	O/M		M		
5	Second Resync Type	C		I		
6	Second Serial Number	C		I		
7	PGI "User Data"	O/M		M		

A.6.23 RESYNCHRONIZE ACK (RA) SPDU

Table 30 RESYNCHRONIZE ACK (RA) SPDU

This PDU is not used by TP for profiles 11 and 12.

ISO/IEC 8327-2			PROFILE			
ITEM#	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	Enclosure Item	C		*		
2	Token Setting Item	O/M	32	M	Minor Sync	
				*	Other Tokens	
3	Resync Type	O/M		I		
4	Serial Number	O/M		M		
5	Second Resync Type	C/C		I		
6	Second Initial Serial Number	C/C		I		
7	PGI "User Data"	O/M		M		

A.6.24 PREPARE (PR) SPDU

Table 31 - PREPARE (PR) SPDU

This PDU is not used by TP for profiles 11 and 12.

ISO/IEC 8327-2			PROFILE			
ITEM#	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	Prepare Type	M		M		
2	Resync Type	C		I		
3	Second Resync Type	C		I		

A.6.25 EXCEPTION REPORT (ER) SPDU

This SPDU is not used by TP. Its use by a U-ASE is prohibited

A.6.26 EXCEPTION DATA (ED) SPDU

This SPDU is not used by TP. Its use by a U-ASE is prohibited

A.6.27 GIVE TOKENS CONFIRM (GTC) SPDU

This SPDU is not used by TP. Its use by a U-ASE in profiles 11 and 12 is not specified by this ISP; however, its use by a U-ASE in profiles 21,22,31 and 32 is prohibited

A.6.28 GIVE TOKENS CONFIRM ACK (GTA) SPDU

This SPDU is not used by TP. Its use by a U-ASE in profiles 11 and 12 is not specified by this ISP; however, its use by a U-ASE in profiles 21,22,31 and 32 is prohibited

A.6.29 ACTIVITY START (AS) SPDU

This SPDU is not used by TP. Its use by a U-ASE in profiles 11 and 12 is not specified by this ISP; however, its use by a U-ASE in profiles 21,22,31 and 32 is prohibited

A.6.30 ACTIVITY RESUME (AR) SPDU

This SPDU is not used by TP. Its use by a U-ASE in profiles 11 and 12 is not specified by this ISP; however, its use by a U-ASE in profiles 21,22,31 and 32 is prohibited

A.6.31 ACTIVITY INTERRUPT (AI) SPDU

This SPDU is not used by TP. Its use by a U-ASE in profiles 11 and 12 is not specified by this ISP; however, its use by a U-ASE in profiles 21,22,31 and 32 is prohibited

A.6.32 ACTIVITY INTERRUPT ACK (AIA) SPDU

This SPDU is not used by TP. Its use by a U-ASE in profiles 11 and 12 is not specified by this ISP; however, its use by a U-ASE in profiles 21,22,31 and 32 is prohibited

A.6.33 ACTIVITY DISCARD (AD) SPDU

This SPDU is not used by TP. Its use by a U-ASE in profiles 11 and 12 is not specified by this ISP; however, its use by a U-ASE in profiles 21,22,31 and 32 is prohibited

A.6.34 ACTIVITY DISCARD ACK (ADA) SPDU

This SPDU is not used by TP. Its use by a U-ASE in profiles 11 and 12 is not specified by this ISP; however, its use by a U-ASE in profiles 21,22,31 and 32 is prohibited

A.6.35 ACTIVITY END (AE) SPDU

This SPDU is not used by TP. Its use by a U-ASE in profiles 11 and 12 is not specified by this ISP; however, its use by a U-ASE in profiles 21,22,31 and 32 is prohibited

A.6.36 ACTIVITY END ACK (AEA) SPDU

This SPDU is not used by TP. Its use by a U-ASE in profiles 11 and 12 is not specified by this ISP; however, its use by a U-ASE in profiles 21,22,31 and 32 is prohibited

ANNEX B: Presentation PROTOCOL PDUs (Normative)

B.1. PRESENTATION SERVICE PARAMETERS

This subclause details TP's requirements on the presentation protocol. The reader should consult the Upper Layer agreements for a detailed discussion of these services. This ISP only specifies PDU parameters necessary for this ISP.

B.1.2. CONNECTION INITIATOR or RESPONDER CAPABILITIES

Table 1 - CONNECTION INITIATOR OR RESPONDER CAPABILITIES

ITEM#	ISO/IEC 8823-2		PROFILE			
	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	Initiator	O		C101		
2	Responder	O		M		

101. If capable of initiating an association then M, else I.

B.1.3. PROTOCOL MECHANISMS

Table 2 -PROTOCOL MECHANISMS

ITEM#	ISO/IEC 8823-2		PROFILE			
	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	X.410 (1984)	O		NA		1
2	Normal	O		M		

NOTES

- Implementations that support X.410 mode shall respond to a CP PDU proposing X.410-1984 mode with an S-U-ABRT; where the user data contains an AbortInformation data element (defined in X.410), which contains an AbortReason (type INTEGER) of value 4 (protocol error). Implementations that do not support X.410-1984 mode shall respond with either a Presentation-Provider abort or a Presentation-Provider Reject.

B.1.4. FUNCTIONAL UNITS

Table 3 - FUNCTIONAL UNITS

ITEM#	ISO/IEC 8823-2		PROFILE			
	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	Kernel	M		M		
2	Presentation Context Management	O		*		
3	Presentation Context Restoration	C		*		

B.1.5. PRESENTATION PDU USAGE BY PROFILE

Table 4 KERNEL PDU USAGE BY PROFILE

ITEM #	PROTOCOL DATA UNITS	ISO/IEC 8823-2	PROFILES						
			11	21	31	12	22	32	NOTES
1	CONNECT PRESENTATION (CP)	C/C	C101 /M	C101 /M	C101 /M	C101 /M	C101 /M	C101 /M	
2	CONNECT PRESENTATION ACCEPT (CPA)	C/C	C102 /C101	C102 /C101	C102 /C101	C102 /C101	C102 /C101	C102 /C101	
3	CONNECT PRESENTATION REJECT (CPR)	C/C	M /C101	M /C101	M /C101	M /C101	M /C101	M /C101	
4	ABNORMAL RELEASE PROVIDER (ARP)	M	M	M	M	M	M	M	
5	ABNORMAL RELEASE USER (ARU)	M	M	M	M	M	M	M	
6	PRESENTATION DATA (TD)	M	M	M	M	M	M	M	

102. If capable of accepting an AARQ APDU then M, else NA.

B.1.6. PRESENTATION CONTEXT MANAGEMENT PDU USAGE BY PROFILE

Table 5 PRESENTATION CONTEXT MANAGEMENT PDU USAGE BY PROFILE

ITEM #	PROTOCOL DATA UNITS	ISO/IEC 8823-2	PROFILES						
			11	21	31	12	22	32	NOTES
1	ALTER CONTEXT (AC)	O/M	*	*	*	*	*	*	
2	ALTER CONTEXT ACKNOWLEDGE (ACA)	M	*	*	*	*	*	*	

B.1.7. OTHER PRESENTATION PDU USAGE BY PROFILE

Table 6 OTHER PRESENTATION PDU USAGE BY PROFILE

ITEM #	PROTOCOL DATA UNITS	ISO/IEC 8823-2	PROFILES						
			11	21	31	12	22	32	NOTES
1	PRESENTATION TYPED DATA (TTD)	M	*	M	M	*	M	M	
2	EXPEDITED DATA (TE)	O/M	*	*	*	*	*	*	
3	CAPABILITY DATA (TC)	O/M	*	NA	NA	*	NA	NA	
4	CAPABILITY DATA ACKNOWLEDGE (TCC)	O/M	*	NA	NA	*	NA	NA	
5	RESYNCHRONIZE (RS)	O/M	*	M	M	*	M	M	
6	RESYNCHRONIZE ACKNOWLEDGE (RSA)	O/M	*	M	M	*	M	M	

B.1.8. PRESENTATION CONTEXT RESTORATION FUNCTIONAL UNIT

No additional PPDUs.

B.2. SUPPORTED PARAMETERS

B.2.1. CONNECT PRESENTATION (CP) PPDU

Table 7 - CONNECT PRESENTATION (CP) PPDU

ITEM#	ISO/IEC 8823-2		PROFILE			
	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	Calling Presentation Selector	O/M		M		
2	Called Presentation Selector	O/M		M		
3	Mode Selector	M		M	Normal	
4	Presentation Context Definition List	O/M		M		
5	Default Context Name	O/M		C103		
6	Protocol Version	O/M		M		
7	Presentation Requirements	O/M		*		1
8	User Session Requirements	O/M		C104		1
9	User Data	O/M		M		

103. If the Expedited Data FU is supported then M, else I.

104. If the Context Management FU is supported then M, else I.

NOTES

- TP does not use this parameter, however a U-ASE is not prohibited from using it, subject to the restrictions imposed by TP on the use of Session Functional Units.

B.2.2. CONNECT PRESENTATION ACCEPT (CPA) PPDU

Table 8 - CONNECT PRESENTATION ACCEPT (CPA) PPDU

ITEM#	ISO/IEC 8823-2		PROFILE			
	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	Responding Presentation Selector	O/M		M		
2	Mode Selector	M		M	Normal	
3	Presentation Context Definition Result List	O/M		M		
4	Protocol Version	O/M		M		
5	Presentation Requirements	O/M		*		1
6	User Session Requirements	O/M		C104		1
7	User Data	O/M		M		

NOTES

- TP does not use this parameter, however a U-ASE is not prohibited from using it, subject to restrictions imposed by TP on the use of Session Functional Units.

B.2.3. CONNECT PRESENTATION REJECT (CPR) PPDU

Table 9 - CONNECT PRESENTATION REJECT (CPR) PPDU

ITEM#	ISO/IEC 8823-2		PROFILE			
	PARAMETER	STATUS	PROFILE ID	STATUS	T/LV ALLOWED	NOTES
1	Responding Presentation Selector	O/M		M		
2	Presentation Context Definition Result List	O/M		M		
3	Protocol Version	O/M		M		
4	Default Context Result	O/M		C105		
5	Provider Reason	O/M		M		1
6	User Data	O/M		M		

105. If capable of proposing Default Context then M, else I.

NOTES

- For enhanced interoperability it is recommended that appropriate provider reason values be sent with all CPR PPDUs.

B.2.4. ABNORMAL RELEASE USER (ARU) PPDU

Table 10 - ABNORMAL RELEASE USER (ARU) PPDU

ITEM#	ISO/IEC 8823-2		PROFILE			
	PARAMETER	STATUS	PROFILE ID	STATUS	T/LV ALLOWED	NOTES
1	Presentation Context Identifier List	O/M		M		
2	User Data	O/M		M		

B.2.5. ABNORMAL RELEASE PROVIDER (ARP) PPDU

Table 11 - ABNORMAL RELEASE PROVIDER (ARP) PPDU

ITEM#	ISO/IEC 8823-2		PROFILE			
	PARAMETER	STATUS	PROFILE ID	STATUS	T/LV ALLOWED	NOTES
1	Abort Reason	O/M		M		1
2	Event Identifier	O/M		M		2

NOTES

- For enhanced interoperability it is recommended that appropriate provider reason values be sent with all ARP PPDUs.

2. For enhanced interoperability it is recommended that appropriate event identifier values be sent with all ARP PPDU.

B.2.6. ALTER CONTEXT (AC) PPDU

This PPDU is not used by TP. Its use by a U-ASE is not specified by this ISP.

B.2.7. ALTER CONTEXT ACKNOWLEDGE (ACA) PPDU

This PPDU is not used by TP. Its use by a U-ASE is not specified by this ISP.

B.2.8. PRESENTATION DATA (TD) PPDU

Table 12 - PRESENTATION DATA (TD) PPDU

ISO/IEC 8823-2			PROFILE			
ITEM#	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	User-data	O/M		M		

B.2.9. PRESENTATION TYPED DATA (TTD) PPDU

Table 13 - PRESENTATION TYPED DATA (TTD) PPDU

This PPDU is not applicable to profiles 11 and 12.

BASE STANDARD ISO 8823			PROFILE			
ITEM#	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	User-data	O/M		M		

B.2.10. EXPEDITED DATA (TE) PPDU

This PPDU is not used by TP. Its use by a U-ASE is not specified by this ISP.

B.2.11. CAPABILITY DATA (TC) PPDU

This PPDU is not used by TP. Its use by a U-ASE is not specified by this ISP. It can be used by Profiles 11 and 12 However its use by a U-ASE in profiles 21,22,31 and 32 is prohibited.

B.2.12. CAPABILITY DATA ACKNOWLEDGE (TCC) PPDU

This PPDU is not used by TP. Its use by a U-ASE is not specified by this ISP. It can be used by Profiles 11 and 12 However its use by a U-ASE in profiles 21,22,31 and 32 is prohibited.

B.2.13. RESYNCHRONIZE (RS) PPDU

Table 14- RESYNCHRONIZE (RS) PPDU

This PPDU is not applicable to profiles 11 and 12.

BASE STANDARD ISO 8823			PROFILE			
ITEM#	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	Presentation Context Identifier List	C/C		C104		
2	User-data	O/M		M		

B.2.14. RESYNCHRONIZE ACKNOWLEDGE (RSA) PPDU

Table 15 - RESYNCHRONIZE ACKNOWLEDGE (RSA) PPDU

This PPDU is not applicable to profiles 11 and 12.

BASE STANDARD ISO 8823			PROFILE			
ITEM#	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTE
1	Presentation Context Identifier List	O/M		C104		
2	User-data	O/M		M		

B.2.15. SESSION SERVICE PRIMITIVES NOT CARRYING PRESENTATION PCI

Table 16 SESSION SERVICE PRIMITIVES NOT CARRYING PRESENTATION PCI

ITEM #	PRIMITIVES	ISO/IEC 8823-2	PROFILES						
			11	21	31	12	22	32	NOTES
1	S-REL req/ind	C	O/M	O/M	O/M	O/M	O/M	O/M	
2	S-REL rsp/cnf	C	*(M) /C106	*(M) /C106	*(M) /C106	*(M) /C106	*(M) /C106	*(M) /C106	
3	S-PT req/ind	C	*	M	M	*	M	M	
4	S-SYNm req/ind	C	*	M	M	*	M	M	
5	S-SYNm rsp/cnf	C	*	M	M	*	M	M	
6	S-SYNM req/ind	C	*	*	*	*	*	*	
7	S-SYNM rsp/cnf	O/M	*	*	*	*	*	*	
8	S-UER req/ind	C	NA	NA	NA	NA	NA	NA	
9	S-ACTS req/ind	C	*	NA	NA	*	NA	NA	
10	S-ACTR req/ind	C	*	NA	NA	*	NA	NA	
11	S-ACTE req/ind	C	*	NA	NA	*	NA	NA	
12	S-ACTE rsp/cnf	C	*	NA	NA	*	NA	NA	

106. If capable of initiating an S-REL req M then else I.

B.3. SUPPORT OF SYNTAXES

B.3.1. TRANSFER SYNTAXES SUPPORTED

Table 17 - TRANSFER SYNTAXES SUPPORTED

ITEM#	TYPE	DETAIL	SUPPORT	
			BASE	P
1	Object Identifier	joint-iso-ccitt asn1(1) basic-encoding(1)	M	M

B.3.2. ABSTRACT SYNTAXES SUPPORTED

Table 18 - ABSTRACT SYNTAXES

ITEM#	TYPE	DETAIL	PROFILE ID	SUPPORT		NOTES
				BASE	P	
1	Object Identifier	joint-iso-ccitt association-control(2) abstract-syntax(1) apdus(0) version1(1)		O	M	
2	Object Identifier	joint-iso-ccitt ccr(7) abstract-syntax(1) apdus(0) version2(2)	11,12	O	I	1
			21,22,31,32	O	M	
3	Object Identifier	joint-iso-ccitt tp(10) abstract-syntax(1) apdus(0) version1(1)		O	M	

NOTES

1. This ISP specifies that a referencing specification shall not use CCR when operating with profile 11 or profile 12 (in particular, refer to Parts 5 and 6 clause 7). However, when the abstract syntaxes are negotiated at the Presentation level, it is not possible to identify whether a protocol error shall be detected or not (this can be detected at the OSI TP level).

ANNEX C: ACSE PROTOCOL PDUs (Normative)

This subclause details TP's use of ACSE services and parameters. The reader should consult the upper layer agreements for a detailed discussion of these services. This ISP only specifies PDU parameters necessary for this ISP.

C.1. SUPPORTED FUNCTIONS**Table 1 - SUPPORTED FUNCTIONS**

ISO/IEC 8650-2			PROFILE			
ITEM#	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	Normal Mode	O		M		
2	X.410 - 1984 mode	O		NA		1
3	Rules for Extensibility	M		M		
4	Supports Operation of Session Vers. 2	O		M		

NOTES

1. Implementations that support X.410 mode shall respond to a CP PPDU proposing X.410-1984 mode with an S-U-ABRT; where the user data contains an AbortInformation data element (defined in X.410), which contains an AbortReason (type INTEGER) of value 4 (protocol error). Implementations that do not support X.410-1984 mode shall respond with either a Presentation-Provider abort or a Presentation-Provider Reject.

C.2. INITIATOR/RESPONDER CAPABILITIES**Table 2 -INITIATOR/RESPONDER CAPABILITIES**

ISO/IEC 8650-2			PROFILE			
ITEM#	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	Association Initiator	O	11,12	O		
			21,22,31,32	M		
2	Association Responder	O	11,12	M		1
			21,22,31,32	M		

NOTES

1. An implementation shall be capable of rejecting an AARQ APDU, acceptance of an AARQ APDU is optional.

C.3. FUNCTIONAL UNITS

Table 3 -FUNCTIONAL UNITS

ITEM#	ISO/IEC 8650-2		PROFILE			
	PARAMETER	STATUS	PROFILE ID	STATUS	T/LV ALLOWED	NOTES
1	Kernel	M		M		
2	Authentication	O		*		

C.4. ACSE PDU USAGE BY PROFILE

Table 4 ACSE PDU USAGE BY PROFILE

ITEM #	Protocol Data Units	ISO/IEC 8650-2	Profiles						
			11	21	31	12	22	32	Notes
1	A-ASSOCIATE-REQUEST (AARQ)	C/C	C101/M	M	M	C101/M	M	M	1
2	A-ASSOCIATE-RESPONSE (AARE)	C/C	M/C101	M	M	M/C101	M	M	1
3	A-RELEASE-REQUEST (RLRQ)	O/M	O/M	O/M	O/M	O/M	O/M	O/M	
4	A-RELEASE-RESPONSE (RLRE)	M/C	M/C102	M/C102	M/C102	M/C102	M/C102	M/C102	
5	A-ABORT (ABRT)	C/C	M	M	M	M	M	M	

101. If capable of initiating an Association then M, else I.

102. If capable of initiating A-RELEASE then M, else I.

NOTES

1. All implementations, including initiator only ones, shall have the capability to receive an AARQ APDU and rejecting it with an AARE APDU.

C.5. A-ASSOCIATE-REQUEST (AARQ)

Table 5 - A-ASSOCIATE-REQUEST (AARQ)

ISO/IEC 8650-2			PROFILE			
ITEM#	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	Protocol Version	C/M		M		
2	Application Context Name	M		M		
3	Calling AP Title	O/M	11,12	O/M	See Table 10	1
			21,31,22,32	M	See Table 10	1
4	Calling AE Qualifier	O/M	11,12	O/M	See Table 10See Table	1
			21,31,22,32	M	10	1
5	Calling AP Invocation Identifier	O/M	11,12	O/M		1
			21,31,22,32	O/M		1, 2
6	Calling AE Invocation Identifier	O/M	11,12	O/M		1
			21,31,22,32	O/M		1,2
7	Called AP Title	O/M		O/M	See Table 10	
8	Called AE Qualifier	O/M		O/M	See Table 10	
9	Called AP Invocation Identifier	O/M	11,12	O/M		1,3
			21,31,22,32	O/M		1,3
10	Called AE Invocation Identifier	O/M	11,12	O/MO/M		1,3
			21,31,22,32			1,3
11	Implementation Information	C/M		I		
12	Requester ACSE Requirements	C/M		*		
13	Mechanism Name	C/M		*		
14	Calling Authentication Value	C/M		*		
15	User Information	O/M		M		

NOTES

- For implementations using association pools, these parameters are recommended in order to re-use associations.
- If this parameter is received, then it is recommended to be logged and used for the reestablishment of the association during transaction recovery.
- If this parameter is sent, then it is recommended to be logged and used for the reestablishment of the association during transaction recovery.

C.6. A-ASSOCIATE-RESPONSE (AARE)**Table 6 - A-ASSOCIATE-RESPONSE (AARE)**

ITEM#	ISO/IEC 8650-2		PROFILE			
	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	Protocol Version	C/M		M		
2	Application Context Name	M		M		
3	Responding AP Title	O/M	11,12	O/M	See Table 10	1
			21,31,22,32	M	See Table 10	1
4	Responding AE Qualifier	O/M	11,12	O/M	See Table 10	1
			21,31,22,32	M	See Table 10	1
5	Responding AP Invocation Identifier	O/M	11,12	O/M		
			21,31,22,32	O/M		2
6	Responding AE Invocation Identifier	O/M	11,12	O/M		
			21,31,22,32	O/M		2
7	Result	M		M		
8	Result Source - Diagnostic	M		M	1 - 10	
		C		*	11 - 14	
9	Implementation Information	O/M		I		
10	Requester ACSE Requirements	C/M		*		
11	Mechanism Name	C/M		*		
12	Calling Authentication Value	C/M		*		
13	User Information	O/M		M		

NOTES

1. This parameter shall be present on an AARE APDU if the called parameter was present in the AARQ APDU and the responding parameter is different from the called parameter.
2. If this parameter is received, then it is recommended to be logged and used for re-establishment of the association during transaction recovery.

C.7 A-RELEASE-REQUEST (RLRQ)**Table 7 - A-RELEASE-REQUEST (RLRQ)**

ITEM#	ISO/IEC 8650-2		PROFILE			
	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	Reason	O/M		I		
2	User information	O/M		I		

C.8. A-RELEASE-RESPONSE (RLRE)**Table 8 - A-RELEASE-RESPONSE (RLRE)**

ISO/IEC 8650-2			PROFILE			
ITEM#	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	Reason	O/M		I		
2	User information	O/M		I		

C.9. A-ABORT (ABRT)**Table 9 - A-ABORT (ABRT)**

ISO/IEC 8650-2			PROFILE			
ITEM#	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	Abort Source	M		M		
2	Abort Diagnostic	C		*		
3	User Information	O/M		M		

C.10. AE TITLE NAME FORMS**Table 10 - AE TITLE NAME FORMS**

ISO/IEC 8650-2			PROFILE			
ITEM#	SYNTAX FORM	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	Form 1 (Directory Name)	O/M		O/M	1 .. 1024 octets	1
2	Form 2 (Object ID)	O/M		M	1 .. 64 octets	1

NOTES

1. The limits specified apply to the AE-Title which is the combination of the AP-Title and AE-Qualifier. They are specified in line with the limits given in Part 3 on the Atomic-Action Identifier and the Branch Identifier.

C.11. AUTHENTICATION VALUE FORM

Table 11 - AUTHENTICATION VALUE FORM

ITEM#	ISO/IEC 8650-2		PROFILE			
	PARAMETER	STATUS	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1	Graphic String	C/C		*		
2	Bit String	C/C		*		
3	External	C/C		*		
4	Any Defined By	C/C		*		

TITLE: WORKING DOCUMENT FOR

Information Technology - Open Systems Interconnection - International Standardized Profile 12061-5:
OSI TP

Part 5: Application Supported Transaction - Polarized Control (ATP11)

SOURCE: Joint AOW / EWOS / OIW on Transaction Processing

DATE: December 18, 1992

STATUS: This document has been harmonized among the three workshops (AOW/EWOS/OIW) and has been submitted to SGFS for progression to ISP.

CONTENTS

INTRODUCTION

1. SCOPE
2. NORMATIVE REFERENCES
3. DEFINITIONS AND ABBREVIATIONS
4. OVERVIEW
5. USE OF FUNCTIONAL UNITS
6. SCENARIO
7. USAGE OF UNDERLYING STANDARDS
8. DETAILED DESCRIPTION

Introduction

The aim of Open Systems Interconnection is to allow, with a minimum of technical agreement outside the interconnection standards, the interconnection of computer systems

- a. from different manufacturers,
- b. under different management,
- c. of different levels of complexity,
- d. of different technologies.

Transaction Processing is concerned with identifiable information which can be related as transactions, which may involve two or more Open Systems. In the framework of Open Systems Interconnection (OSI) a transaction is defined as "a set of related operations characterized by four properties: atomicity, consistency, isolation and durability."

The definition highlights that a distributed transaction is more than a simple exchange of messages, but that the exchanges form a protected indivisible set.

This multi-part document contains the complete specification of the six profiles identified in M-IT-02 and TR 10000.

Part 1 contains the taxonomy for the OSI TP profiles.

Part 2 contains the specification of the support of OSI TP APDUs for each of the profiles specified in Parts 5 to 10.

Part 3 contains the specification of the support of the CCR APDUs for each of the profiles specified in Part 5 to 10.

Part 4 contains the specification of the support of ACSE, Presentation and Session APDUs for each of the profiles specified in Part 5 to 10.

Parts 5 to 10 specify the six profiles which are defined, based on the OSI TP standard. These six parts make reference to Parts 2 to 4.

THIS PAGE IS INTENTIONALLY LEFT BLANK

**Information Technology - Open Systems Interconnection - International
Standardized Profiles 12061-5: OSI Distributed Transaction Processing.**

**Part 5: APPLICATION SUPPORTED TRANSACTIONS - POLARIZED CONTROL
(ATP11)**

1. SCOPE

This Part of this ISP defines the OSI TP profile used for Application Supported Transaction while the application is using the Polarized Control paradigm for communications.

2. NORMATIVE REFERENCES

The references listed in Part 1 of this ISP apply to this Part.

3. DEFINITIONS AND ABBREVIATIONS

The definitions and abbreviations listed in Part 1 of this ISP apply to this Part.

4. OVERVIEW

Profile ATP11 is applicable to end systems concerned with operating in the Open Systems Interconnection (OSI) environment. This profile specifies a combination of OSI standards, which collectively provide support for Application Supported Distributed Transactions, where the applications take responsibility for ensuring that transaction semantics are maintained, and for restoring consistency after any failure. The dialogue between the applications is subject to strict turn control. The handshake facility is available.

5. USE OF FUNCTIONAL UNITS

An implementation of this profile supports the OSI TP functional units identified in the table hereafter:

DIALOGUE	mandatory
POLARIZED CONTROL	mandatory
HANDSHAKE	mandatory

It conforms to the Application Transactions conformance class defined in ISO 10026-3.

- ISO/IEC 8327-2 (Session)

TITLE: WORKING DOCUMENT FOR

**Information Technology - Open Systems Interconnection - International Standardized Profile 12061-6:
OSI TP**

Part 6: Application Supported Transactions- Shared Control (ATP12)

SOURCE: Joint AOW / EWOS / OIW on Transaction Processing

DATE: December 18, 1992

**STATUS: This document has been harmonized among the three workshops (AOW/EWOS/OIW) and
has been submitted to SGFS for progression to ISP.**

CONTENTS

INTRODUCTION

- 1. SCOPE**
- 2. NORMATIVE REFERENCES**
- 3. DEFINITIONS AND ABBREVIATIONS**
- 4. OVERVIEW**
- 5. USE OF FUNCTIONAL UNITS**
- 6. SCENARIO**
- 7. USAGE OF UNDERLYING STANDARDS**
- 8. DETAILED DESCRIPTION**

INTRODUCTION

The aim of Open Systems Interconnection is to allow, with a minimum of technical agreement outside the interconnection standards, the interconnection of computer systems

- a. from different manufacturers,
- b. under different management,
- c. of different levels of complexity,
- d. of different technologies.

Transaction Processing is concerned with identifiable information which can be related as transactions, which may involve two or more Open Systems. In the framework of Open Systems Interconnection (OSI) a transaction is defined as "a set of related operations characterized by four properties: atomicity, consistency, isolation and durability."

The definition highlights that a distributed transaction is more than a simple exchange of messages, but that the exchanges form a protected indivisible set.

This multi-part document contains the complete specification of the six profiles identified in M-IT-02 and TR 10000.

Part 1 contains the taxonomy for the OSI TP profiles.

Part 2 contains the specification of the support of OSI TP APDUs for each of the profiles specified in Parts 5 to 10.

Part 3 contains the specification of the support of the CCR APDUs for each of the profiles specified in Part 5 to 10.

Part 4 contains the specification of the support of ACSE, Presentation and Session APDUs for each of the profiles specified in Part 5 to 10.

Parts 5 to 10 specify the six profiles which are defined, based on the OSI TP standard. These six parts make reference to Parts 2 to 4.

THIS PAGE IS INTENTIONALLY LEFT BLANK

Information Technology - Open Systems Interconnection - International
Standardized Profiles 12061-6: OSI Distributed Transaction Processing.

Part 6: APPLICATION SUPPORTED TRANSACTIONS - SHARED CONTROL
(ATP12)

1. SCOPE

This Part of this ISP defines the OSI TP profile used for Application Supported Transaction while the application is using the Shared Control paradigm for communications.

2. NORMATIVE REFERENCES

The references listed in Part 1 of this ISP apply to this Part.

3. DEFINITIONS AND ABBREVIATIONS

The definitions and abbreviations listed in Part 1 of this ISP apply to this Part.

4. OVERVIEW

Profile ATP12 is applicable to end systems concerned with operating in the Open Systems Interconnection (OSI) environment. This profile specifies a combination of OSI standards, which collectively provide support for Application Supported Distributed Transactions, where the applications take responsibility for ensuring that transaction semantics are maintained, and for restoring consistency after any failure. The dialogue between the applications is not subject to turn control. The support of the handshake facility is optional.

5. USE OF FUNCTIONAL UNITS

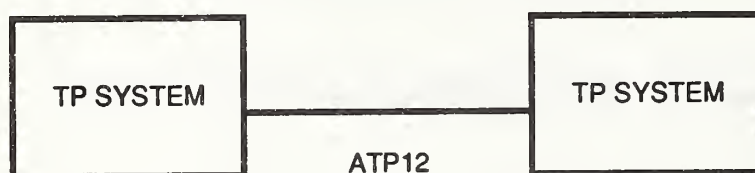
An implementation of this profile supports the OSI TP functional units as shown hereafter:

DIALOGUE	mandatory
SHARED CONTROL	mandatory
HANDSHAKE	optional

It conforms to the Application Transactions conformance class defined in ISO 10026-3.

6. SCENARIO

The applicability of the ATP12 profile is illustrated by the figure hereafter:



7. USAGE OF UNDERLYING STANDARDS

This profile specifies the required functions from the supporting protocol stacks shown below. The use of ISO/IEC 9804/9805 (CCR) by a referencing specification is forbidden and is a protocol error.

Application Layer	ISO 10026-3:1992 (OSI TP) ISO 8650 ¹ (ACSE)
Presentation Layer	ISO 8825:1990 (BER ASN.1) ISO 8823:1988(Presentation)
Session Layer	ISO 8327 ⁴

8. DETAILED DESCRIPTION

The support of the OSI TP, ACSE, Presentation and Session PDUs for ATP12 is as described in Parts 1,2 and 4 of this standard.

9. CONFORMANCE

Conformance requirements specified in ISO/IEC ISP 12061-1, ISO/IEC ISP 12061-2, ISO/IEC ISP 12061-4 apply to this part.

For each implementation claiming conformance to this part of ISO/IEC 12061, the following PICS Proformas shall be completed and made available :

- ISO/IEC 10026-4 (OSI TP)

⁴To be published

- **ISO/IEC 8650-2 (ACSE)**
- **ISO/IEC 8823-2 (Presentation)**
- **ISO/IEC 8327-2 (Session)**

TITLE: WORKING DOCUMENT FOR

**Information Technology - Open Systems Interconnection - International Standardized Profile 12061-7:
OSI TP**

Part 7: Provider Supported Unchained Transactions - Polarized Control (ATP21)

SOURCE: Joint AOW / EWOS / OIW on Transaction Processing

DATE: December 18, 1992

**STATUS: This document has been harmonized among the three workshops (AOW/EWOS/OIW) and
has been submitted to SGFS for progression to ISP.**

CONTENTS

INTRODUCTION

- 1. SCOPE**
- 2. NORMATIVE REFERENCES**
- 3. DEFINITIONS AND ABBREVIATIONS**
- 4. OVERVIEW**
- 5. USE OF FUNCTIONAL UNITS**
- 6. SCENARIO**
- 7. USAGE OF UNDERLYING STANDARDS**
- 8. DETAILED DESCRIPTION**

INTRODUCTION

The aim of Open Systems Interconnection is to allow, with a minimum of technical agreement outside the interconnection standards, the interconnection of computer systems

- a. from different manufacturers,
- b. under different management,
- c. of different levels of complexity,
- d. of different technologies.

Transaction Processing is concerned with identifiable information which can be related as transactions, which may involve two or more Open Systems. In the framework of Open Systems Interconnection (OSI) a transaction is defined as "a set of related operations characterized by four properties: atomicity, consistency, isolation and durability."

The definition highlights that a distributed transaction is more than a simple exchange of messages, but that the exchanges form a protected indivisible set.

This multi-part document contains the complete specification of the six profiles identified in M-IT-02 and TR 10000.

Part 1 contains the taxonomy for the OSI TP profiles.

Part 2 contains the specification of the support of OSI TP APDUs for each of the profiles specified in Parts 5 to 10.

Part 3 contains the specification of the support of the CCR APDUs for each of the profiles specified in Part 5 to 10.

Part 4 contains the specification of the support of ACSE, Presentation and Session APDUs for each of the profiles specified in Part 5 to 10.

Parts 5 to 10 specify the six profiles which are defined, based on the OSI TP standard. These six parts make reference to Parts 2 to 4.

THIS PAGE IS INTENTIONALLY LEFT BLANK

Information Technology - Open Systems Interconnection - International
Standardized Profiles 12061-7: OSI Distributed TRANSACTION PROCESSING.

Part 7: PROVIDER SUPPORTED UNCHAINED TRANSACTIONS - POLARIZED
CONTROL (ATP21)

1. SCOPE

This Part of this ISP defines the OSI TP profile used for unchained sequences of Provider Supported Transaction branches while the application is using the Polarized Control paradigm for communications.

2. NORMATIVE REFERENCES

The references listed in Part 1 of this ISP apply to this Part.

3. DEFINITIONS AND ABBREVIATIONS

The definitions and abbreviations listed in Part 1 of this ISP apply to this Part.

4. OVERVIEW

Profile ATP21 is applicable to end systems concerned with operating in the Open Systems Interconnection (OSI) environment. This profile specifies a combination of OSI standards, which collectively provide support for Provider Supported Distributed Transactions, where the provider of the OSI TP service takes responsibility for ensuring transaction ACID properties and for restoring consistency after any failure. Two applications operate on a dialogue in an Unchained sequence of Provider Supported Transaction Branches. The dialogue between the applications is subject to strict turn control. The handshake facility is available.

5. USE OF FUNCTIONAL UNITS

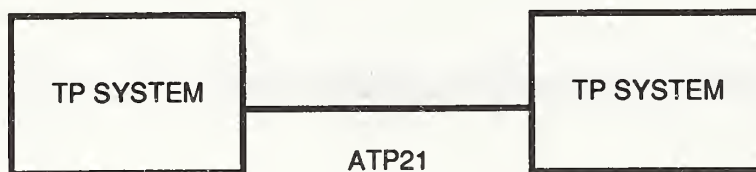
An implementation of this profile supports the OSI TP functional units as shown hereafter:

DIALOGUE	mandatory
POLARIZED CONTROL	mandatory
HANDSHAKE	mandatory
COMMIT	mandatory
UNCHAINED TRANSACTIONS	mandatory
RECOVERY	mandatory

It conforms to the Unchained Provider Supported Transaction Branches conformance class defined in ISO 10026-3.

6. SCENARIO

The applicability of the ATP21 profile is illustrated by the figure hereafter:



7. USAGE OF UNDERLYING STANDARDS

This profile specifies the required functions from the supporting protocol stacks shown below.

Application Layer	ISO 10026-3:1992 (OSI TP) ISO 8650 ¹ (ACSE) ISO 9805:1990(CCR) ISO 9805 AM2
Presentation Layer	ISO 8825:1990 (BER ASN.1) ISO 8823:1988(Presentation) ISO 8823 AM5
Session Layer	ISO 8327 ⁵ ISO 8327 AM3

8. DETAILED DESCRIPTION

The support of the OSI TP, CCR, ACSE, Presentation and Session PDUs for ATP21 is as described in Parts 1 - 4 of this standard.

9. CONFORMANCE

Conformance requirements specified in ISO/IEC ISP 12061-1, ISO/IEC ISP 12061-2, ISO/IEC ISP 12061-3, ISO/IEC ISP 12061-4 apply to this part.

For each implementation claiming conformance to this part of ISO/IEC 12061, the following PICS Proformas shall be completed and made available :

- ISO/IEC 10026-4 (OSI TP)
- ISO/IEC 9805-2 (CCR)

⁵To be published

- **ISO/IEC 8650-2 (ACSE)**
- **ISO/IEC 8823-2 (Presentation)**
- **ISO/IEC 8327-2 (Session)**

TITLE: WORKING DOCUMENT FOR

**Information Technology - Open Systems Interconnection - International Standardized Profile 12061-8:
OSI TP**

Part 8: Provider Supported Unchained Transactions - Shared Control (ATP22)

SOURCE: Joint AOW / EWOS / OIW on Transaction Processing

DATE: December 18, 1992

**STATUS: This document has been harmonized among the three workshops (AOW/EWOS/OIW) and
has been submitted to SGFS for progression to ISP.**

CONTENTS

INTRODUCTION

1. SCOPE
2. NORMATIVE REFERENCES
3. DEFINITIONS AND ABBREVIATIONS
4. OVERVIEW
5. USE OF FUNCTIONAL UNITS
6. SCENARIO
7. USAGE OF UNDERLYING STANDARDS
8. DETAILED DESCRIPTION

INTRODUCTION

The aim of Open Systems Interconnection is to allow, with a minimum of technical agreement outside the interconnection standards, the interconnection of computer systems

- a. from different manufacturers,
- b. under different management,
- c. of different levels of complexity,
- d. of different technologies.

Transaction Processing is concerned with identifiable information which can be related as transactions, which may involve two or more Open Systems. In the framework of Open Systems Interconnection (OSI) a transaction is defined as "a set of related operations characterized by four properties: atomicity, consistency, isolation and durability."

The definition highlights that a distributed transaction is more than a simple exchange of messages, but that the exchanges form a protected indivisible set.

This multi-part document contains the complete specification of the six profiles identified in M-IT-02 and TR 10000.

Part 1 contains the taxonomy for the OSI TP profiles.

Part 2 contains the specification of the support of OSI TP APDUs for each of the profiles specified in Parts 5 to 10.

Part 3 contains the specification of the support of the CCR APDUs for each of the profiles specified in Part 5 to 10.

Part 4 contains the specification of the support of ACSE, Presentation and Session APDUs for each of the profiles specified in Part 5 to 10.

Parts 5 to 10 specify the six profiles which are defined, based on the OSI TP standard. These six parts make reference to Parts 2 to 4.

THIS PAGE IS INTENTIONALLY LEFT BLANK

Information Technology - Open Systems Interconnection - International Standardized Profiles 12061-8: OSI Distributed Transaction Processing.

Part 8: PROVIDER SUPPORTED UNCHAINED TRANSACTIONS - SHARED CONTROL (ATP22)

1. SCOPE

This Part of this ISP defines the OSI TP profile used for Unchained sequences of Provider Supported Transaction branches while the application is using the Shared Control paradigm for communications.

2. NORMATIVE REFERENCES

The references listed in Part 1 of this ISP apply to this Part.

3. DEFINITIONS AND ABBREVIATIONS

The definitions and abbreviations listed in Part 1 of this ISP apply to this Part.

4. OVERVIEW

Profile ATP22 is applicable to end systems concerned with operating in the Open Systems Interconnection (OSI) environment. This profile specifies a combination of OSI standards, which collectively provide support for Provider Supported Distributed Transactions, where the provider of the OSI TP service takes responsibility for ensuring transaction ACID properties and for restoring consistency after any failure. Two applications operate on a dialogue in an Unchained sequence of Provider Supported Transaction Branches. The dialogue between the applications is not subject to turn control. The support of the handshake facility is optional.

5. USE OF FUNCTIONAL UNITS

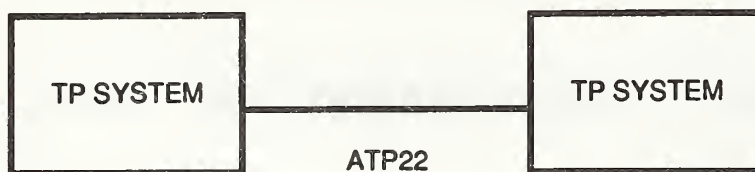
An implementation of this profile supports the OSI TP functional units as shown hereafter:

DIALOGUE	mandatory
SHARED CONTROL	mandatory
HANDSHAKE	optional
COMMIT	mandatory
UNCHAINED TRANSACTION	mandatory
RECOVERY	mandatory

It conforms to the Unchained Provider Supported Transactions Branches conformance class defined in ISO 10026-3.

6. SCENARIO

The applicability of the ATP22 profile is illustrated by the figure hereafter:



7. USAGE OF UNDERLYING STANDARDS

This profile specifies the required functions from the supporting protocol stacks shown below.

Application Layer	ISO 10026-3:1992 (OSI TP) ISO 8650 ¹ (ACSE) ISO 9805:1990(CCR) ISO 9805 AM2
Presentation Layer	ISO 8825:1990 (BER ASN.1) ISO 8823:1988(Presentation) ISO 8823 AM5
Session Layer	ISO 8327 ⁶ ISO 8327 AM3

8. DETAILED DESCRIPTION

The support of the OSI TP, CCR, ACSE, Presentation and Session PDUs for ATP22 is as described in Parts 1 - 4 of this standard.

9. CONFORMANCE

Conformance requirements specified in ISO/IEC ISP 12061-1, ISO/IEC ISP 12061-2, ISO/IEC ISO/IEC ISP 12061-3, ISP 12061-4 apply to this part.

For each implementation claiming conformance to this part of ISO/IEC 12061, the following PICS Proformas shall be completed and made available :

- ISO/IEC 10026-4 (OSI TP)
- ISO/IEC 9805-2 (CCR)
- ISO/IEC 8650-2 (ACSE)

⁶To be published

- **ISO/IEC 8823-2 (Presentation)**
- **ISO/IEC 8327-2 (Session)**

TITLE: WORKING DOCUMENT FOR

**Information Technology - Open Systems Interconnection - International Standardized Profile 12061-9:
OSI TP**

Part 9: Provider Supported Chained Transactions - Polarized Control (ATP31)

SOURCE: Joint AOW / EWOS / OIW on Transaction Processing

DATE: December 18, 1992

**STATUS: This document has been harmonized among the three workshops (AOW/EWOS/OIW) and
has been submitted to SGFS for progression to ISP.**

CONTENTS

INTRODUCTION

- 1. SCOPE**
- 2. NORMATIVE REFERENCES**
- 3. DEFINITIONS AND ABBREVIATIONS**
- 4. OVERVIEW**
- 5. USE OF FUNCTIONAL UNITS**
- 6. SCENARIO**
- 7. USAGE OF UNDERLYING STANDARDS**
- 8. DETAILED DESCRIPTION**

INTRODUCTION

The aim of Open Systems Interconnection is to allow, with a minimum of technical agreement outside the interconnection standards, the interconnection of computer systems

- a. from different manufacturers,
- b. under different management,
- c. of different levels of complexity,
- d. of different technologies.

Transaction Processing is concerned with identifiable information which can be related as transactions, which may involve two or more Open Systems. In the framework of Open Systems Interconnection (OSI) a transaction is defined as "a set of related operations characterized by four properties: atomicity, consistency, isolation and durability."

The definition highlights that a distributed transaction is more than a simple exchange of messages, but that the exchanges form a protected indivisible set.

This multi-part document contains the complete specification of the six profiles identified in M-IT-02 and TR 10000.

Part 1 contains the taxonomy for the OSI TP profiles.

Part 2 contains the specification of the support of OSI TP APDUs for each of the profiles specified in Parts 5 to 10.

Part 3 contains the specification of the support of the CCR APDUs for each of the profiles specified in Part 5 to 10.

Part 4 contains the specification of the support of ACSE, Presentation and Session APDUs for each of the profiles specified in Part 5 to 10.

Parts 5 to 10 specify the six profiles which are defined, based on the OSI TP standard. These six parts make reference to Parts 2 to 4.

THIS PAGE IS INTENTIONALLY LEFT BLANK

Information Technology - Open Systems Interconnection - International Standardized Profiles 12061-9: OSI Distributed Transaction Processing.

Part 9: PROVIDER SUPPORTED CHAINED TRANSACTIONS - POLARIZED CONTROL (ATP31)

1. SCOPE

This Part of this ISP defines the OSI TP profile used for chained sequences of Provider Supported Transaction branches while the application is using the Polarized Control paradigm for communications.

2. NORMATIVE REFERENCES

The references listed in Part 1 of this ISP apply to this Part.

3. DEFINITIONS AND ABBREVIATIONS

The definitions listed in Part 1 of this ISP apply to this Part.

4. OVERVIEW

Profile ATP31 is applicable to end systems concerned with operating in the Open Systems Interconnection (OSI) environment. This profile specifies a combination of OSI standards, which collectively provide support for Provider Supported Distributed Transactions, where the provider of the OSI TP service takes responsibility for ensuring transaction ACID properties and for restoring consistency after any failure. Two applications operate on a dialogue in an Chained Sequence of Provider Supported Transaction Branches. The dialogue between the applications is subject to strict turn control. The handshake facility is available.

5. USE OF FUNCTIONAL UNITS

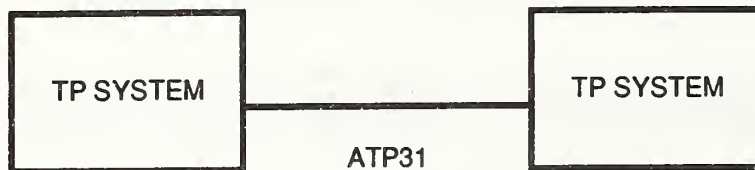
An implementation of this profile supports the OSI TP functional units as shown hereafter:

DIALOGUE	mandatory
POLARIZED CONTROL	mandatory
HANDSHAKE	mandatory
COMMIT	mandatory
CHAINED TRANSACTION	mandatory
RECOVERY	mandatory

It conforms to the Chained Provider Supported Transaction Branches conformance class defined in ISO 10026-3.

6. SCENARIO

The applicability of the ATP31 profile is illustrated by the figure hereafter:



7. USAGE OF UNDERLYING STANDARDS

This profile specifies the required functions from the supporting protocol stacks shown below.

Application Layer	ISO 10026-3:1992 (OSI TP) ISO 8650 ¹ (ACSE) ISO 9805:1990(CCR) ISO 9805 AM2
Presentation Layer	ISO 8825:1990 (BER ASN.1) ISO 8823:1988(Presentation) ISO 8823 AM5
Session Layer	ISO 8327 ⁷ ISO 8327 AM3

8. DETAILED DESCRIPTION

The support of the OSI TP, CCR, ACSE, Presentation and Session PDUs for ATP31 is as described in Parts 1 -4 of this standard.

9. CONFORMANCE

Conformance requirements specified in ISO/IEC ISP 12061-1, ISO/IEC ISP 12061-2, ISO/IEC ISO/IEC ISP 12061-3, ISP 12061-4 apply to this part.

For each implementation claiming conformance to this part of ISO/IEC 12061, the following PICS Proformas shall be completed and made available :

- ISO/IEC 10026-4 (OSI TP)
- ISO/IEC 9805-2 (CCR)
- ISO/IEC 8650-2 (ACSE)

⁷To be published

- **ISO/IEC 8823-2 (Presentation)**
- **ISO/IEC 8327-2 (Session)**

TITLE: WORKING DOCUMENT FOR

**Information Technology - Open Systems Interconnection - International Standardized Profile 12061-10:
OSI TP**

Part 10: Provider Supported Chained Transactions - Shared Control (ATP32)

DATE: December 18, 1992

**STATUS: This document has been harmonized among the three workshops (AOW/EWOS/OIW) and
has been submitted to SGFS for progression to ISP.**

CONTENTS

INTRODUCTION

- 1. SCOPE**
- 2. NORMATIVE REFERENCES**
- 3. DEFINITIONS AND ABBREVIATIONS**
- 4. OVERVIEW**
- 5. USE OF FUNCTIONAL UNITS**
- 6. SCENARIO**
- 7. USAGE OF UNDERLYING STANDARDS**
- 8. DETAILED DESCRIPTION**

INTRODUCTION

The aim of Open Systems Interconnection is to allow, with a minimum of technical agreement outside the interconnection standards, the interconnection of computer systems

- a. from different manufacturers,
- b. under different management,
- c. of different levels of complexity,
- d. of different technologies.

Transaction Processing is concerned with identifiable information which can be related as transactions, which may involve two or more Open Systems. In the framework of Open Systems Interconnection (OSI) a transaction is defined as "a set of related operations characterized by four properties: atomicity, consistency, isolation and durability."

The definition highlights that a distributed transaction is more than a simple exchange of messages, but that the exchanges form a protected indivisible set.

This multi-part document contains the complete specification of the six profiles identified in M-IT-02 and TR 10000.

Part 1 contains the taxonomy for the OSI TP profiles.

Part 2 contains the specification of the support of OSI TP APDUs for each of the profiles specified in Parts 5 to 10.

Part 3 contains the specification of the support of the CCR APDUs for each of the profiles specified in Part 5 to 10.

Part 4 contains the specification of the support of ACSE, Presentation and Session APDUs for each of the profiles specified in Part 5 to 10.

Parts 5 to 10 specify the six profiles which are defined, based on the OSI TP standard. These six parts make reference to Parts 2 to 4.

THIS PAGE IS INTENTIONALLY LEFT BLANK

**Information Technology - Open Systems Interconnection - International
Standardized Profiles 12061-10: OSI Distributed Transaction Processing.**

**Part 10: PROVIDER SUPPORTED CHAINED TRANSACTIONS - SHARED
CONTROL (ATP32)**

1. SCOPE

This Part of this ISP defines the OSI TP profile used for chained sequences of Provider Supported Transaction branches while the application is using the Shared Control paradigm for communications.

2. NORMATIVE REFERENCES

The references listed in Part 1 of this ISP apply to this Part.

3. DEFINITIONS AND ABBREVIATIONS

The definitions and abbreviations listed in Part 1 of this ISP apply to this Part.

4. OVERVIEW

Profile ATP32 is applicable to end systems concerned with operating in the Open Systems Interconnection (OSI) environment. This profile specifies a combination of OSI standards, which collectively provide support for Provider Supported Distributed Transactions, where the provider of the OSI TP service takes responsibility for ensuring transaction ACID properties and for restoring consistency after any failure. Two applications operate on a dialogue in an Chained sequence of Provider Supported Transaction Branches. The dialogue between the applications is not subject to turn control. The support of the handshake facility is optional.

5. USE OF FUNCTIONAL UNITS

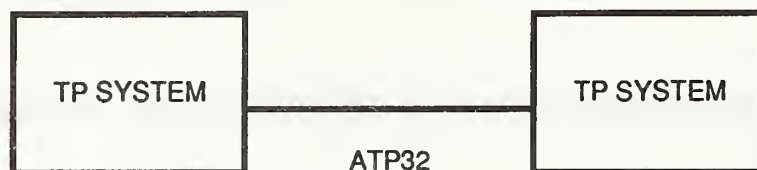
An implementation of this profile supports the OSI TP functional units as shown hereafter:

DIALOGUE	mandatory
SHARED CONTROL	mandatory
HANDSHAKE	optional
COMMIT	mandatory
CHAINED TRANSACTION	mandatory
RECOVERY	mandatory

It conforms to the Chained Provider Supported Transaction Branches conformance class defined in ISO 10026-3.

6. SCENARIO

The applicability of the ATP32 profile is illustrated by the figure hereafter:



7. USAGE OF UNDERLYING STANDARDS

This profile specifies the required functions from the supporting protocol stacks shown below.

Application Layer	ISO 10026-3:1992 (OSI TP) ISO 8650 ¹ (ACSE) ISO 9805:1990(CCR) ISO 9805 AM2
Presentation Layer	ISO 8825:1990 (BER ASN.1) ISO 8823:1988(Presentation) ISO 8823 AM5
Session Layer	ISO 8327 ⁸ ISO 8327 AM3

8. DETAILED DESCRIPTION

The support of the OSI TP, CCR, ACSE, Presentation and Session PDUs for ATP32 is as described in Parts 1 - 4 of this standard.

9. CONFORMANCE

Conformance requirements specified in ISO/IEC ISP 12061-1, ISO/IEC ISP 12061-2, ISO/IEC ISO/IEC ISP 12061-3, ISP 12061-4 apply to this part.

For each implementation claiming conformance to this part of ISO/IEC 12061, the following PICS Proformas shall be completed and made available :

- ISO/IEC 10026-4 (OSI TP)
- ISO/IEC 9805-2 (CCR)

⁸To be published

- ISO/IEC 8650-2 (ACSE)
- ISO/IEC 8823-2 (Presentation)
- ISO/IEC 8327-2 (Session)

15.12. OIW Application Contexts

15.12.1. Application context name

Title: UDT with Commit Profiles

Object identifier:{iso(1) identified-organization(3) oiw(14) tpsig(15)
application-context(5) udt-with-CCR(1) version1(0)}

15.12.1.1. Purpose and Scope

The purpose of this application context is to provide the TPSUs participating in a TP dialogue with a mechanism for exchanging unstructured data by mapping it onto an APDU consisting of a simple octet string. A bilateral agreement will be required between two cooperating TPSUs using this context since the syntax and semantics of the application protocol are not defined here.

This application context supports both application and provider supported transactions executed serially over the same association.

The use of the base standards identified here is restricted by the specification of the OSI TP profiles.

15.12.1.2. Referenced Standards

This application context definition references in whole or in part the following specifications:

ISO/IEC 10026-3:1992, Information Technology - Open Systems Interconnection - Distributed Transaction Processing - Part 3: Protocol specification

ISO/IEC 10026-5:1992, Information Technology - Open Systems Interconnection - Distributed Transaction Processing Protocol - Part 5: Application Context Proforma and Guidelines When using OSI TP

ISO/IEC 10026-6:1992, Information Technology - Open Systems Interconnection - Distributed Transaction Processing Protocol - Part 6: Unstructured Data Transfer

ISO/IEC 8650 Second edition, Information Technology - Open Systems Interconnection - Association Control Service Element

ISO/IEC 9805-1:1990, Information Technology - Open Systems Interconnection - Protocol Specification for Commitment, Concurrency and Recovery Service Elements

ISO/IEC 9805-1 AM2:1992, Information Technology - Open Systems Interconnection - Protocol Specification for Commitment, Concurrency and Recovery Service Elements, Amendment 2.

ISO/IEC PDISP 12061, Information Technology - Open Systems Interconnection - Distributed Transaction Processing ISP

15.12.1.2.1. Referenced application contexts

None

15.12.1.3. Components ASEs and ASOs

The following ASEs are contained in this application context:

ACSE

TP-ASE

CCR

UDT

15.12.1.3.1. Association Control Service Element (ACSE)

15.12.1.3.1.1. References

ISO/IEC 12061 part 4

ISO/IEC 8650 Second Edition

15.12.1.3.1.2. Version Number

Version 1 of the ACSE protocol is used.

15.12.1.3.1.3. Brief description

ACSE is used to establish and terminate associations. The ACSE functions are not exercised directly by UDT or through the TP service, but are exercised by association management facilities within the TP service provider.

15.12.1.3.2. Distributed Transaction Processing ASE (TP-ASE)

15.12.1.3.2.1. References

ISO/IEC 12061-2 and 12061-3

ISO/IEC 10026-3

15.12.1.3.2.2. Version Number

Version 1 of the OSI TP protocol is used.

15.12.1.3.2.3. Brief description

OSI TP provides communications mechanisms for the support of processing transactions across two or more separate systems.

15.12.1.3.3. Commitment, Concurrency and Recovery (CCR)

15.12.1.3.3.1. References

ISO/IEC 12061-3,

ISO/IEC 9805-1 and

ISO/IEC 9805-1 AM2

15.12.1.3.3.2. Version Number

Version 2 of the CCR protocol is used.

15.12.1.3.3.3. Brief description

CCR is used in support of the commitment, rollback and recovery functions. Only, TP makes use of the CCR ASE services.

15.12.1.3.4. Unstructured Data Transfer ASE (UDT)**15.12.1.3.4.1. References**

ISO/IEC 10026-6.

15.12.1.3.4.2. Version Number

Version 1 of the UDT protocol is used.

15.12.1.3.4.3. Brief description

This ASE transfers unstructured (i.e. structure unknown to the TPPM) data between cooperating TPSUIs.

15.12.1.3.4.4. Use of other ASEs or ASOs

None

15.12.1.4. Persistent application context rules

None

15.12.1.5. Control function rules**15.12.1.5.1. SACF rules****15.12.1.5.1.1. Objective/summary**

There are no SACF rules beyond those specified in the base standard, ISO/IEC 10026-3.

15.12.1.5.1.2. Temporal ordering rules

There are no SACF temporal ordering rules beyond those specified in the base standard, ISO/IEC 10026-3 for the TP-DATA generic service.

15.12.1.5.1.3. Concatenation rules

There are no SACF concatenation rules beyond those specified in the base standard, ISO/IEC 10026-3.

15.12.1.5.1.4. Mapping rules

The UDT-TRANSFER-RI APDU may be mapped onto P-DATA, the User-Data parameter of the TP-BEGIN-DIALOGUE service, or the User-Data parameter of TP-U-ABORT service. There are no state transitions beyond those specified in the base standard, ISO/IEC 10026-3 for the TP-DATA generic service.

15.12.1.5.1.5. References to base rules

ISO/IEC 10026-3

15.12.1.5.1.6. Other rules

None

15.12.1.5.2. MACF rules**15.12.1.5.2.1. Objective/summary**

There are no MACF rules beyond those specified in the base standard, ISO/IEC 10026-3.

15.12.1.5.2.2. Temporal ordering rules

There are no MACF sequencing rules beyond those specified in the base standard, ISO/IEC 10026-3.

15.12.1.5.2.3. Concatenation rules

There are no MACF concatenation rules beyond those specified in the base standard, ISO/IEC 10026-3.

15.12.1.5.2.4. Mapping rules

There are no MACF mapping rules.

15.12.1.5.2.5. References to base rules

ISO/IEC 10026-3

15.12.1.5.2.5.1. Other rules

None

15.12.1.5.3. Optional features

None

15.12.1.5.4. Error handling

For this application context, if the rules and constraints of the application context are violated and the commit functional unit has been selected, the transaction should be rolled back. If the commit functional unit is not selected, the TP-U-ERROR service may be used or the dialogue may be aborted depending on the severity of the error.

15.12.1.5.5. Conformance

Conformance to this application context consists of conformance to ISO/IEC PDISP 12061 and ISO/IEC 10026-6.

15.12.1.5.6. Collision handling

No collision handling rules beyond those of ISO/IEC 10026-3 are required.

15.12.2. Application context name

Title: Application Supported Transactions using UDT

Object identifier:{iso(1) identified-organization(3) oiw(14) tpsig(15)
application-context(5) udt-without-CCR(2) version1(0)}

15.12.2.1. Purpose and Scope

The purpose of this application context is to provide the TPSUs participating in a TP dialogue with a mechanism for exchanging unstructured data by mapping it onto an APDU consisting of a simple octet string. A bilateral agreement will be required between two cooperating TPSUs using this context since the syntax and semantics of the application protocol are not defined here.

This application context supports application supported transactions executed serially over the same association.

The use of the base standards identified here is restricted by the specification of the OSI TP profiles.

15.12.2.2. Referenced Standards

This application context definition references in whole or in part the following specifications:

ISO/IEC 10026-3:1992, Information Technology - Open Systems Interconnection - Distributed Transaction Processing - Part 3: Protocol specification

ISO/IEC 10026-5:1992, Information Technology - Open Systems Interconnection - Distributed Transaction Processing Protocol - Part 5: Application Context Proforma and Guidelines When using OSI TP

ISO/IEC 10026-6:1992, Information Technology - Open Systems Interconnection - Distributed Transaction Processing Protocol - Part 6: Unstructured Data Transfer

ISO/IEC 8650 Second edition, Information Technology - Open Systems Interconnection - Association Control Service Element

ISO/IEC PDISP 12061, Information Technology - Open Systems Interconnection - Distributed Transaction Processing ISP

15.12.2.3. Referenced application contexts

None

15.12.2.4. Components ASEs and ASOs

The following ASEs are contained in this application context:

ACSE

TP-ASE

UDT

15.12.2.4.1. Association Control Service Element (ACSE)**15.12.2.4.1.1. References**

ISO/IEC 12061 part 4
ISO/IEC 8650 Second Edition

15.12.2.4.1.2. Version Number

Version 1 of the ACSE protocol is used.

15.12.2.4.1.3. Brief description

ACSE is used to establish and terminate associations. The ACSE functions are not exercised directly by UDT or through the TP service, but are exercised by association management facilities within the TP service provider.

15.12.2.4.2. Distributed Transaction Processing ASE (TP-ASE)**15.12.2.4.2.1. References**

ISO/IEC 12061-2 and 12061-3
ISO/IEC 10026-3

15.12.2.4.2.2. Version Number

Version 1 of the OSI TP protocol is used.

15.12.2.4.2.3. Brief description

OSI TP provides communications mechanisms for the support of processing transactions across two or more separate systems.

15.12.2.4.3. Unstructured Data Transfer ASE (UDT)**15.12.2.4.3.1. References**

ISO/IEC 10026-6.

15.12.2.4.3.2. Version Number

Version 1 of the UDT protocol is used.

15.12.2.4.3.3. Brief description

This ASE transfers unstructured (i.e. structure unknown to the TPPM) data between cooperating TPSUIs.

15.12.2.4.4. Use of other ASEs or ASOs

None

15.12.2.5. Persistent application context rules

None

15.12.2.6. Control function rules**15.12.2.6.1. SACF rules****15.12.2.6.1.1. Objective/summary**

There are no SACF rules beyond those specified in the base standard, ISO/IEC 10026-3.

15.12.2.6.1.2. Temporal ordering rules

There are no SACF temporal ordering rules beyond those specified in the base standard, ISO/IEC 10026-3 for the TP-DATA generic service.

15.12.2.6.1.3. Concatenation rules

There are no SACF concatenation rules beyond those specified in the base standard, ISO/IEC 10026-3.

15.12.2.6.1.4. Mapping rules

The UDT-TRANSFER-RI APDU may be mapped onto P-DATA, the User-Data parameter of the TP-BEGIN-DIALOGUE service, or the User-Data parameter of TP-U-ABORT service. There are no state transitions beyond those specified in the base standard, ISO/IEC 10026-3 for the TP-DATA generic service.

15.12.2.6.1.5. References to base rules

ISO/IEC 10026-3

15.12.2.6.1.6. Other rules

None

15.12.2.6.2. MACF rules**15.12.2.6.2.1. Objective/summary**

There are no MACF rules beyond those specified in the base standard, ISO/IEC 10026-3.

15.12.2.6.2.2. Temporal ordering rules

There are no MACF sequencing rules beyond those specified in the base standard, ISO/IEC 10026-3.

15.12.2.6.2.3. Concatenation rules

There are no MACF concatenation rules beyond those specified in the base standard, ISO/IEC 10026-3.

15.12.2.6.2.4. Mapping rules

There are no MACF mapping rules.

15.12.2.6.2.5. References to base rules

ISO/IEC 10026-3

15.12.2.6.2.6. Other rules

None

15.12.2.6.3. Optional features

None

15.12.2.6.4. Error handling

For this application context, if the rules and constraints of the application context are violated the TP-U-ERROR service may be used or the dialogue may be aborted depending on the severity of the error.

15.12.2.6.5. Conformance

Conformance to this application context consists of conformance to ISO/IEC PDISP 12061 and ISO/IEC 10026-6.

15.12.2.6.6. Collision handling

No collision handling rules beyond those of ISO/IEC 10026-3 are required.

Stable Implementation Agreements for Open Systems Interconnection Protocols: Part 16 - Open Document Architecture Level 3 DAP

**Output from the December 1992 Open Systems
Environment Implementors' Workshop (OIW)**

**SIG Chair: Jim Wing, IBM
SIG Editor: Jim Wing, IBM**

**PART 16 - Office Document Architecture
Level 3 DAP**

December 1992 (Stable)

Foreword

This part of the Stable Implementation Agreements was prepared by the Office Document Architecture Special Interest Group (ODASIG) of the Open Systems Environment Implementors' Workshop (OIW).

Text in this part has been approved by the Plenary of the above-mentioned Workshop.

Future changes and additions to this version of these Implementor Agreements will be published as change pages. Deleted and replaced text will be shown as struckout. New and replacement text will be shown as shaded.

Table of Contents

Part 16 - Office Document Architecture Level 3 DAP	1
---	----------

Part 16 - Office Document Architecture Level 3 DAP

NOTE - Text for the International Standardized Profile 11182-1 (FOD36) follows this page.

This Agreement provides a Document Application Profile which has been developed through the joint efforts of ODA expert groups within the:

OSE Implementors' Workshop (OIW);

Asia-Oceania Workshop (AOW);

European Workshop for Open Systems (EWOS);

CCITT Study Group VIII.

This effort was conducted through meetings of the Profile Alignment Group for ODA (PAGODA) for the purpose of facilitating the interworking of applications which interchange documents based on [ISO 8613 | T.410 series of CCITT Recommendations]. This Agreement specifies one of the profiles resulting from that work:

ISO/IEC ISP 11182-1 : 1992, Information technology - Standardized Profile FOD36 - Office Document Format : Extended document structure - Character, raster graphics and geometric graphics content architectures - Document Application Profile.

This Agreement accepts the entirety of the definitions and provisions of this ISP as the agreement for the OIW Stable Agreements and the standards upon which the specified ISP is based as referenced by the text of the ISP. For this reason the text of the ISP is not reproduced here, but referenced to avoid any doubt as to the official text being agreed upon.

ISO/IEC ISP 11182-1

**Information technology -
International Standardized Profile FOD36 -
Office Document Format : Extended document structure -
Character, raster graphics and geometric graphics
content architectures -**

Part 1 :
Document Application Profile

Date : 1992-10-01
Source : OIW ODA SIG
Reference : OIW/ODA 92/1205

Contents

Foreword	v
Introduction	v
1 Scope	1
2 Normative references	2
3 Definitions	4
4 Relationship with other profiles	4
5 Conformance	4
5.1 Data stream conformance	5
5.2 Implementation conformance	5
6 Characteristics supported by this document application profile	7
6.1 Overview	7
6.2 Logical characteristics	10
6.3 Layout characteristics	28
6.4 Document layout characteristics	61
6.5 Content layout and imaging characteristics	72
6.6 Miscellaneous features	87
6.7 Document management features	99
7 Specification of constituent constraints	105
7.1 Introduction	105
7.2 Document profile constituent constraints	119
7.3 Logical constituent constraints	131
7.4 Layout constituent constraints	167
7.5 Layout style constituent constraints	213
7.6 Presentation style constituent constraints	221
7.7 Content portion constituent constraints	224
8 Interchange format	226
8.1 Interchange format class A	226
8.2 Interchange format SDIF	227
Annexes	
A Amendments and corrigenda	229
B Recommendations	231
C Bibliography	238

List of Figures

Figure 1 — Body layout types A, B and C with header and footer above and below the body area	33
Figure 2 — Header and footer frame layouts corresponding to Figure 1	33
Figure 3 — Body layout types A, B and C with header and footer to the right and left of the body area	34
Figure 4 — Header and footer frame layouts corresponding to Figure 3	34
Figure 5 — Example of body area layout for body layout type A	37
Figure 6 — Example of body area layout for body layout type B	38
Figure 7 — Example of body area layout for body layout type C	39
Figure 8 — Example of the layout of a snaking columns frame	41
Figure 9 — Another example of the layout of a snaking columns frame	41
Figure 10 — Example of the layout of a synchronized column	43
Figure 11 — Example of a CompositeFloat frame	44
Figure 12 — Example of a CompositeFixtureVariable frame	46
Figure 13 — Example of the layout of a form	48
Figure 14 — Example of a <i>simple</i> table	52
Figure 15 — Example of a <i>complex</i> table	53
Figure 16 — Frames used to represent table labels	55
Figure 17 — Frames used to represent table rows	56
Figure 18 — Specification of margins	71
Figure 19 — Code extension features (basic case)	76
Figure 20 — Code extension features (all possible cases)	77
Figure 21 — DocumentLogicalRoot, 1st level	105
Figure 22 — DocumentLogicalRoot, 2nd level	106
Figure 23 — Title, Paragraph and ListTerm	106
Figure 24 — Phrase, Caption and Description	107
Figure 25 — Footnote	107
Figure 26 — Figure	108
Figure 27 — Form	108
Figure 28 — Table	109
Figure 29 — Reference	109
Figure 30 — NumberedList	110
Figure 31 — UnNumberedList	110
Figure 32 — DefinitionList	111
Figure 33 — CommonContent	111
Figure 34 — DocumentLayoutRoot	112
Figure 35 — Page, RectoPage and VersoPage	112
Figure 36 — CompositeHeader, CompositeFooter and CompositeCommon	113
Figure 37 — FixedCompositeBody	113
Figure 38 — VariableCompositeBody	114
Figure 39 — SnakingColumns	114
Figure 40 — SynchronizedColumns	115
Figure 41 — CompositeFloat	115
Figure 42 — CompositeColumnVariable and CompositeColumnFixed	116
Figure 43 — CompositeFixtureVariable and CompositeFixtureFixed	116
Figure 44 — FormArea and EntryGroup	117
Figure 45 — TableArea	117

List of Tables

Table 1 — Nominal page sizes	31
Table 2 — Layout of Figure	66
Table 3 — Layout of Tables	67
Table 4 — Layout of Forms	69
Table 5 — List of number string identifiers for logical constituents	90
Table 6 — List of number string identifiers for layout constituents	91

Foreword

Development of this document application profile has been done in liaison with several organizations. These include ODA expert groups within the:

- Asia-Oceania Workshop (AOW);
- CCITT Study Group VIII;
- European Workshop for Open Systems (EWOS);
- OSE Implementors Workshop (OIW).

The liaison between these organizations has occurred within the meetings of the Profile Alignment Group for ODA (PAGODA). These meetings have focused on the development of a single set of Internationally aligned ODA document application profiles.

The profile defined in this ISP is a part of the ODA profile taxonomy defined in TR 10000-2, 4.4.4.3 and 5.4.1. This profile is specific to the profile identifier FOD36.

At present, this ISP consists of one part:

Part 1: Document application profile.

Further parts may be added to this ISP.

This part contains three annexes:

- Annex A (normative): Amendments and corrigenda;
- Annex B (informative): Recommendations;
- Annex C (informative): Bibliography.

Introduction

The purpose of this International Standardized Profile (ISP) | CCITT Recommendation is to facilitate the interworking of applications interchanging documents based on [ISO 8613 | T.410 series of CCITT Recommendation], ODA. This ISP named FOD36 | CCITT Recommendation named PM-36 is suitable for interchanging documents in formatted form, processable form or formatted processable form and has been defined in accordance with [ISO 8613-1 | CCITT Recommendation T.411]. The format of this profile is in accordance with the standardized proforma and notation defined in Addendum 1 - Annex F of [ISO 8613-1 | CCITT Recommendation T.411].

The specification of FOD36 is identical to the specification of PM-36, except that PM-36 only defines the use of the ODIF interchange format.

Information technology -
International Standardized Profile FOD36 -
Office Document Format : Extended document structure -
Character, raster graphics and geometric graphics
content architectures -

DOCUMENT APPLICATION PROFILE PM-36 FOR THE INTERCHANGE OF CHARACTER, RASTER GRAPHICS, AND GEOMETRIC GRAPHICS CONTENT DOCUMENTS IN PROCESSABLE, FORMATTED AND FORMATTED PROCESSABLE FORMS

Document Application Profile

1 Scope

This profile specifies interchange formats for the transfer of structured documents between equipment designed for word or document processing. Such documents may contain character, raster graphics and geometric graphics content.

The documents that can be interchanged using this profile range from simple documents to highly structured technical reports, articles and typeset documents such as brochures. This profile provides a comprehensive level of features for the transfer of documents between these systems.

This profile allows documents to be interchanged in the following forms:

- a) formatted form;
- b) processable form;
- c) formatted processable form.

The architecture levels defined for these three forms have matching functionalities so that the interchange formats of a document are convertible from a processable form to any other form.

This profile is independent of the processes carried out in an end system to create, edit or reproduce documents. It is also independent of the means to transfer documents which, for example, may be by means of communication links or storage media.

(
NOTE 1 - The following texts apply to the ISP only.

A document structured in accordance with this profile is represented for interchange by one of two DAPs. One DAP uses the Office Document Interchange Format (ODIF), the other DAP uses the Office Document Language (ODL), both as defined in ISO 8613-5.

When this document refers to *this profile*, it is referring to either of the document application profiles defined by this specification.

)

2 Normative references

The following documents contain provisions which, through reference in this text, constitute provisions of this Specification. At the time of publication, the editions indicated were valid. All documents are subject to revision, and parties to agreements based on this Specification are warned against automatically applying any more recent editions of the documents listed below, since the nature of references made by profiles to such documents is that they may be specific to a particular edition. Members of IEC and ISO maintain registers of currently valid International Standards and ISPs. The CCITT secretariat maintains a list of the currently valid CCITT recommendations.

ISO 8613-1 : 1989, *Information Processing - Text and Office Systems; Office Document Architecture (ODA) and Interchange Format - Part 1: Introduction and General Principles*;

ISO 8613-2 : 1989, *Information Processing - Text and Office Systems; Office Document Architecture (ODA) and Interchange Format - Part 2: Document Structures*;

ISO 8613-4 : 1989, *Information Processing - Text and Office Systems; Office Document Architecture (ODA) and Interchange Format - Part 4: Document Profile*;

ISO 8613-5 : 1989, *Information Processing - Text and Office Systems; Office Document Architecture (ODA) and Interchange Format - Part 5: Office Document Interchange Format*;

ISO 8613-6 : 1989, *Information Processing - Text and Office Systems; Office Document Architecture (ODA) and Interchange Format - Part 6: Character Content Architectures*;

ISO 8613-7 : 1989, *Information Processing - Text and Office Systems; Office Document Architecture (ODA) and Interchange Format - Part 7: Raster Graphics Content Architectures*;

ISO 8613-8 : 1989, *Information Processing - Text and Office Systems; Office Document Architecture (ODA) and Interchange Format - Part 8: Geometric Graphics Content Architectures*;

ISO 8613-1 Addendum 1 : *Information Processing - Text and Office Systems; Office Document Architecture (ODA) and Interchange Format Part 1: Introduction and General Principles Addendum 1 - Annex F : A Document Application Profile Proforma and Notation*;

CCITT Recommendation T.400 : 1988, *Introduction to Document Architecture, Transfer and Manipulation*;

CCITT Recommendation T.411 : 1988, *Open Document Architecture (ODA) and Interchange Format: Introduction and General Principles*;

CCITT Recommendation T.412 : 1988, *Open Document Architecture (ODA) and Interchange Format: Document Structures*;

CCITT Recommendation T.414 : 1988, *Open Document Architecture (ODA) and Interchange Format: Document Profile*;

CCITT Recommendation T.415 : 1988, *Open Document Architecture (ODA) and Interchange Format: Open Document Interchange Format*;

CCITT Recommendation T.416 : 1988, *Open Document Architecture (ODA) and Interchange Format: Character Content Architecture*;

CCITT Recommendation T.417 : 1988, *Open Document Architecture (ODA) and Interchange Format: Raster Graphics Content Architecture*;

CCITT Recommendation T.418 : 1988, *Open Document Architecture (ODA) and Interchange Format: Geometric Graphics Content Architecture*;

ISO/IEC 646 : 1991, *Information technology - ISO 7-bit coded character set for information interchange*;

ISO 8859-1 : 1987, *Information Processing - 8-bit Single-byte coded graphic character sets - Part 1: Latin alphabet No. 1*;

ISO 6937-2 : 1983, *Information Processing - Coded character sets for text communication - Part 2: Latin alphabetic and non-alphabetic graphic characters*;

ISO 6937-2 ADDENDUM 1 : 1989, *Information Processing - Coded character sets for text communication - PART 2: Latin alphabetic and non-alphabetic graphic characters ADDENDUM 1*;

ISO 2022 : 1986, *Information Processing - ISO 7-bit and 8-bit coded character sets - Code extension techniques*;

ISO 2375 : 1985, *Data Processing - Procedure for registration of escape sequences*;

ISO 7350 : 1984, *Text communication - Registration of graphic character subrepertoires*;

ISO 8824 : 1987, *Information Processing Systems - Open Systems Interconnection - Abstract Syntax Notation One (ASN.1)*;

ISO 8825 : 1987, *Information Processing Systems - Open Systems Interconnection - Basic encoding rules for abstract syntax notation one (ASN.1)*;

ISO 8879 : 1986, *Information processing - Text and office systems - Standard Generalized Markup Language (SGML)*;

ISO 9069 : 1988, *Information processing - SGML support facilities - SGML Document Interchange Format (SDIF)*;

CCITT Recommendation T.502 | ISO/IEC ISP 10610-1 : 1992, *Information technology - Standardized Profile PM-11 | FOD11 - Open Document Format : Simple document structure - Character content architecture only - Document Application Profile*;

CCITT Recommendation T.505 | ISO/IEC ISP 11181-1 : 1992, *Information technology - Standardized Profile PM-26 | FOD26 - Open Document Format : Enhanced document structure - Character, raster graphics and geometric graphics content architectures - Document Application Profile*;

CCITT Recommendation T.516 : (to be published) *Implementation requirement for document application profile PM-36*;

ISO/IEC TR 10000-1 : 1990, *Information Technology - Framework and Taxonomy of International Standardized Profiles - Part 1: Framework*;

ISO/IEC TR 10000-2 : 1990, *Information Technology - Framework and Taxonomy of International Standardized Profiles - Part 2: Taxonomy*;

3 Definitions

For the purposes of this Specification, the following definitions apply.

The definitions given in [ISO 8613-1 | CCITT Recommendation T.411] are applicable.

Constituent names

Each constituent that may be included in a document that conforms to this profile has been given a unique name which serves to identify that constituent throughout this profile.

The convention is that full names are used (i.e., no abbreviations are used), two or more words in a name are concatenated and each word begins with a capital. Examples of constituent names used in this profile are BodyText, Footnote, RectoPage and ColumnFixed.

In clause 6, each constituent provided by this profile is underlined once at the point in the text at which the purpose of that constituent is defined. This also serves to identify all the constituents provided by this profile.

The same constituent names are also used in the technical specification in clause 7 so that there is a one-to-one correspondence between the use of these names in clauses 6 and 7.

Although the constituent names relate to the purpose of the constituents, the semantics of constituents shall not be implied from the actual names that are used. Also, these names do not appear in an interchanged document but a mechanism for identifying constituents in an interchange document is provided (see 6.6.5). Thus in an application using this profile, the constituents may be known to the user by different names.

4 Relationship with other profiles

This profile belongs to a series of hierarchically related profiles which include PM-11 | FOD11 and PM-26 | FOD26.

The features supported by this profile are a superset of the features supported by the profiles PM-11 | FOD11 and PM-26 | FOD26 and thus all data streams that are conformant to PM-11 | FOD11 and PM-26 | FOD26 are also conformant to this profile apart from the DAP identifier.

5 Conformance

In order to conform to this profile, a data stream representing a document must meet the requirements specified in 5.1.

(

NOTE 2 - The following text applies to the ISP only.

This part of the ISP does not define implementation support requirements. These requirements are defined in other parts that make use of this ISP.

)

(

NOTE 3 - The following text applies to the CCITT Recommendations only.

This Recommendation does not define implementation or service requirements. These requirements are defined in other Recommendations that make use of this profile.

5.1 Data stream conformance

The following requirements apply to the encoding of data streams which conform to this profile:

- a) (NOTE 4 - The following text applies to the ISP only.

the data stream shall be encoded either in accordance with the ASN.1 encoding rules defined in [ISO 8825 | CCITT Recommendation X.209] or the SGML encoding rules defined in ISO 8879;

)

- (NOTE 5 - The following text applies to CCITT Recommendation only.

the data stream shall be encoded in accordance with the ASN.1 encoding rules defined in [CCITT Recommendation X.209 | ISO 8825];

)

- b) the data stream shall be structured in accordance with the interchange formats defined in clause 8;
- c) the document, as represented by the data stream after resolution of any external references, shall be structured in accordance with one of the documents architecture classes as defined in 6.1 and shall contain all mandatory constituents specified for that class; other constituents may be included, provided that they are permitted for that class as specified in clause 7;
- d) each constituent shall contain all those attributes specified as required for that constituent in this profile; other attributes may be specified provided that they are permitted for that constituent;
- e) the attribute values specified shall be within the range of permissible values specified in this profile;
- f) the encoded document shall be constructed in accordance with the abstract document architecture defined in [ISO 8613-2 | CCITT Recommendation T.412];
- g) the document shall be structured in accordance with the characteristics and constraints specified in clause 6.

5.2 Implementation conformance

This subclause states the requirements for implementations claiming conformance to this profile.

(

NOTE 6 - The following text applies to the ISP only.

A conforming receiving implementation shall be capable of receiving either any data streams conforming to this profile structured in accordance with ODIF, or any data streams conforming to this profile structured in accordance with ODL or both of these. Receiving usually, but not always, involves recognizing and further processing the data stream elements. The explicit meaning of "receiving" is determined by another part of this ISP.

A receiving system which claims conformance to this DAP shall be capable of handling data streams which are conformant to DAPs that are subordinate to this DAP within the taxonomy described in clause 4, i.e. FOD11 and FOD26.

)
(

NOTE 7 - The following text applies to the ISP only.

The implementation requirements associated with this profile are defined in Recommendation T.516.

)

6 Characteristics supported by this document application profile

This clause describes the characteristics of documents which may be represented by data streams conforming to this profile. This clause also describes how these characteristics are represented in terms of constituent constraints.

6.1 Overview

6.1.1 General

This profile supports the interchange of documents in the following forms:

- processable form, which facilitates the revision of a document by a recipient;
- formatted form, which facilitates the reproduction of a document as intended by the originator;
- formatted processable form, which facilitates the reproduction of a document as intended by the originator or facilitates the revision of a document by a recipient;

In addition this profile supports the interchange of:

- generic-document;
- document profile.

The constituents that make up these five classes of data stream are defined in 6.1.2 to 6.1.6. Constituents defined as 'required' shall occur in any data stream that conforms to this profile. Constituents listed as 'optional' may or may not be present in the data stream depending on the requirements of the particular data stream.

NOTE 8 - The constituents that make up a complete data stream that is conformant to this profile includes all those referenced and contained in resource and external documents, if any (see 6.6.1 and 6.6.2).

6.1.2 Formatted form documents

Required constituents:

- a document profile;
- layout object descriptions representing a specific layout structure.

Optional constituents:

- layout object class descriptions representing a 'factor' generic layout structure;
- presentation styles;
- content portion descriptions.

6.1.3 Processable form documents

Required constituents:

- a document profile;
- logical object class descriptions representing a 'complete' or 'partial' generic logical structure;
- logical object descriptions representing a specific logical structure.

Optional constituents:

- layout object class descriptions representing a 'complete' generic layout structure;
- layout styles;
- presentation styles;
- content portion descriptions.

In the case of processable form documents, when the generic layout structure is not present, additional restrictions are placed on the layout directives that may be included in layout styles. These restrictions are defined in 6.4.3 of this profile.

6.1.4 Formatted processable form documents

Required constituents:

- a document profile;
- logical object class descriptions representing a 'complete' or 'partial' generic logical structure;
- logical object descriptions representing a specific logical structure;
- layout object class descriptions representing a 'complete' generic layout structure;
- layout object descriptions representing a specific layout structure.

Optional constituents:

- layout styles;
- presentation styles;
- content portion descriptions.

6.1.5 Generic documents

A generic-document consists of one of the following sets of constituents:

a)

- a document profile;
- logical object class descriptions which represent a 'complete' or 'partial' generic logical structure;
- layout styles whose presence is optional;
- presentation styles whose presence is optional;
- generic content portions whose presence is optional.

b)

- a document profile;
- layout object class descriptions which represent a 'complete' generic layout structure or 'factor set';
- presentation styles whose presence is optional;
- generic content portions whose presence is optional.

c)

- a document profile;
- logical object class descriptions which represent a 'complete' or 'partial' generic logical structure;
- layout object class descriptions which represent a 'complete' generic layout structure;
- layout styles whose presence is optional;
- presentation styles whose presence is optional;
- generic content portions whose presence is optional.

6.1.6 Document profile

This form of document contains a document profile only.

6.2 Logical characteristics

6.2.1 Introduction

This subclause defines the logical constituent constraints provided by this profile to represent the characteristics of documents.

Different constituent constraints may be used to represent and distinguish parts of a document that have different logical characteristics. This subclause describes the general characteristics and typical uses of the constituent constraints that are provided.

The descriptions of the logical characteristics represented by each of the constituent constraints is provided for guidance only. It is the responsibility of the user to determine how a document is to be represented using the constituents provided. Adherence to these guidelines can enhance the mutual understanding of a document by an originator and a recipient.

6.2.2 Overview of the logical structure

From the logical point of view, the document consists of two parts, namely a 'body' part and a 'common' part.

The 'body' part represents the main content of a document and is intended to be reproduced in the body area of the pages that make up the document. The 'body' part must be included in all documents that are interchanged in accordance with this profile.

The 'common' part represents common content that is to be placed in reserved header and footer areas on each page of a document. Header and footer content are independently optional and so may be included in an interchanged document only if required.

6.2.3 Body part of the logical structure

6.2.3.1 DocumentLogicalRoot

DocumentLogicalRoot is a constituent constraint representing the top level in the document logical structure. Its immediate subordinates consist of an arbitrary ordered sequence of one or more constituent constraints of the types `Passage` and `NumberedSegment`.

The automatic numbering schemes that apply to constituent constraints of the types `NumberedSegment`, `Figure`, `Table`, `NumberedList` and `Footnote` may be initialized on the `DocumentLogicalRoot`.

6.2.3.2 Passage

Passage is a constituent constraint that represents a subdivision of a document that corresponds to a logical grouping of subordinate parts of a document. This grouping may be regarded as a logical entity for reading, or it may represent parts of a document that have common layout and presentation characteristics.

Passages are typically used to represent:

- the contents to be placed on the title page of a report;
- the front matter, consisting of the table of contents or foreword;
- the main matter of the document;
- the back matter, consisting of appendices, glossary or index;
- an illustration with associated text which is inserted as a distinct entity within a section of a document.

The automatic numbering schemes that apply to subordinate constituent constraints of the types `NumberedSegment`, `Figure`, `Table`, `NumberedList` and `Footnote` may be initialized on a `Passage`.

The immediate subordinates of a `Passage` consist of an optional constituent constraint of the type `Title` which is followed by an arbitrary ordered sequence of one or more of the following constituent constraints:

- `Paragraph`;
- `BodyGeometric`;
- `BodyRaster`;
- `BodyText`;
- `Figure`;
- `Table`;
- `NumberedList`;
- `UnNumberedList`;
- `DefinitionList`;
- `NumberedSegment`;
- `Passage`.

A `Passage` shall have at least one of the above constituent constraints as a subordinate.

Therefore, a `Passage` may itself contain one or more subordinates of the type `Passage`, and thus `Passages` may be nested to any number of levels. This allows logical entities within a document to be described in terms of their component logical entities. Also, a `NumberedSegment` may contain one or more subordinate `Passages` which allows different logical entities within a `NumberedSegment` to be distinguished.

A document may contain several different class definitions of the type `Passage`, each of which defines the common characteristics of sets of `Passages` within the document, such as their allowed subordinates or layout properties. For example, a class of `Passages` may be defined which always begins on a new page set.

6.2.3.3 NumberedSegment

NumberedSegment is a constituent constraint that represents a logical entity within a document that is distinguished by an identifier. This logical entity may be a subdivision of a document or a higher level Passage or NumberedSegment. The entity may also be distinguished by having some common layout characteristics.

The automatic numbering schemes that apply to the subordinate constituent constraints NumberedSegment, Figure, Table, NumberedList and Footnote may be initialized on any logical object or logical object class, typically on Passage or a NumberedSegment.

The immediate subordinates of a NumberedSegment consist of the constituent constraint Number whose presence is mandatory and serves to carry the identifier of the NumberedSegment. This is followed optionally by a constituent constraint Title which in turn is followed by an optional arbitrary ordered sequence of one or more of the following constituent constraints:

- Paragraph;
- BodyGeometric;
- BodyRaster;
- BodyText;
- Figure;
- Table;
- NumberedList;
- UnNumberedList;
- DefinitionList;
- Passage;
- NumberedSegment.

The above indicates that a document may contain any number of nested levels of the constituent constraint NumberedSegment.

A NumberedSegment is typically used to represent entities such as chapters, sections, nested sub-sections and appendices which contain an identifier that serves to distinguish that entity for human comprehension.

A document may contain any number of different class definitions of NumberedSegment which define the common characteristics of sets of NumberedSegments, such as their allowed subordinates and layout properties.

Class definitions of NumberedSegments may be recursively defined. In this case, only one class of NumberedSegment may be specified, and the <simple-expr> construction in the macro USENUMBERSTRINGS in the bindings attribute of this class shall use the optional ORD construction only. If recursive class definitions are used for NumberedSegment, the following constraints will also apply. For all levels which reference recursively defined classes:

- numbering format will be the same;
- no initial value other than 1 or re-initialization of the numbering is possible;
- it is not possible to continue the numbering across Passages.

6.2.3.4 Number

Number is a constituent constraint that represents the identifier of a NumberedSegment, Figure or NumberedList to which it is subordinate. This identifier allows the superior constituent constraint to which it belongs to be distinguished within the document for machine processing or human comprehension.

A Number is a basic logical constituent constraint which contains a content generator which, when evaluated, produces the identifier referred to above. This evaluation takes place during the layout process.

The identifiers are structured and consist of a sequence of one or more numerals that allow NumberedSegments at the same or different levels in a document structure to be uniquely distinguished. The numerals may be represented by Arabic or Roman numerals or by their alphabetic equivalent in lower or upper case characters (the number 1 is represented by 'A', etc.). Each numeral in an identifier is distinguished by means of 'separator' characters such as spaces and full stops (the character 'period'); a typical example is '6.2.3.4'.

NOTE 9 - The separator may be an empty string.

Further details of the structure and generation of the identifiers are given in 6.6.7.

6.2.3.5 Title

Title is a constituent constraint that is used to represent the title, heading or name of the Passage or NumberedSegment to which it is an immediate subordinate. This constituent constraint may consist of character, raster graphics and geometric graphics content.

The immediate subordinates of this constituent constraint consist of an arbitrary ordered sequence of one or more of the following constituent constraints:

- BodyText;
- BodyRaster;
- BodyGeometric;
- Phrase;
- Reference;
- Footnote.

The character content associated with a Title may be concatenated to form a continuous stream of character content which may contain single or multiple references to footnotes or other parts of the document, and may be laid out as single unit. Entities within this content which have particular logical significance or presentation features may be distinguished using the constituent constraint Phrase.

Content from any subordinate basic text objects within a paragraph may be run-on one from another (that is, to continue on the same line) by use of Concatentation (see 6.4.2.5). Alternatively, content from subordinates of a paragraph may be separated one from another to give white space between them, using Separation (see 6.4.2.2). This may be used to give an effect similar to that achieved with empty lines of text. Use of empty text lines to achieve white space between areas of text or other content may lead to unintended blank areas adjacent to the leading edge of layout objects, whereas the use of Separation avoids this.

6.2.3.6 Paragraph

Paragraph is a constituent constraint that is a subdivision of a Passage or NumberedSegment. It is typically used to represent the grouping of parts of a document that deals with a single theme or topic. These parts may consist of character, raster graphics and geometric graphics content.

The immediate subordinates of a Paragraph consist of an arbitrary ordered sequence of one or more of the following constituent constraints:

- BodyText;
- BodyRaster;
- BodyGeometric;
- Phrase;
- Reference;
- Footnote.

The character content associated with a Paragraph may be concatenated to form a continuous stream of character content which may contain single or multiple references to footnotes or other parts of the document, and may be laid out as single unit. Entities within this content which have particular logical significance or presentation features may be distinguished using the constituent constraint Phrase.

Content from subordinates of a paragraph may be separated one from another to give white space between them using Separation (see 6.4.2.2). This may be used to give an effect similar to that achieved with empty lines of text. Use of empty text lines to achieve white space between areas of text or other content may lead to unintended blank areas adjacent to the leading edge of layout objects (e.g., at page breaks) whereas the use of Separation avoids this.

6.2.3.7 Phrase

Phrase is a constituent constraint that is used to group together an amount of character content that represents a single logical entity that needs to be distinguished for some purpose. That is, the content represented by a Phrase may have a particular logical significance, or it may have certain layout or presentation requirements. The character content may contain embedded footnotes and references to other parts of the document content. A typical example is a quotation that is to be reproduced in italics.

A Phrase may be used as subdivision of constituent constraints of the types Paragraph, Title, Caption, Description, Artwork and ListItem. Also, a Phrase may be subordinate to another Phrase and, therefore, constituent constraints of the type Phrase may be nested.

The immediate subordinates of this constituent constraint consist of an arbitrary ordered sequence of one or more of the following constituent constraints:

- BodyText;
- Phrase;
- Reference;
- Footnote.

The character content associated with a Phrase may be concatenated to form a continuous stream of character content which may contain single or multiple references to footnotes or other parts of the document, and may be laid out as single unit. Alternatively, the character content may contain hard line terminators, which will cause parts of the content to be separated when laid out.

6.2.3.8 BodyText, BodyRaster and BodyGeometric

BodyText, BodyRaster and BodyGeometric are constituent constraints which represent the lowest level of logical subdivision of a document. These constituent constraints act as carriers for the document content and may be specified as subordinates of constituent constraints:

- Passage;
- NumberedSegment;
- Paragraph;
- Title;
- ListTerm;
- Artwork;
- Phrase;
- Reference;
- UnNumberedList.

In addition, BodyText may be specified as a subordinate to Phrase, Caption and Description. These constituent constraints allow the layout and presentation requirements of different parts of the content of a document to be specified.

These are basic logical constituent constraints that directly refer to content portions that contain character, raster graphics and geometric graphics content respectively. BodyText shall refer to one or more content portions which may contain either processable, formatted or formatted processable character content. BodyRaster and BodyGeometric shall only refer to a single content portion containing formatted processable raster graphics content or formatted processable geometric graphics content respectively.

These constituent constraints in the generic logical structure may refer to generic content. This provides the means of defining common content within the body part of a document.

Where the superior constituent constraint referenced is subordinate to a FootnoteBody, it is required to specify one of the layout category names for this constituent constraint, 'Footnote' or 'Footnote-<n>'.

This along with a Permitted categories attribute of the same name on the footnote frame will ensure that a logical object from this constituent constraint is laid out in a FootnoteArea frame when generic layout structure is specified within the document.

6.2.3.9 Constituents representing footnotes

This subclause defines the constituents that are used to represent footnotes.

6.2.3.9.1 Footnote

Footnote is a constituent constraint that is used to represent footnotes within a document. This constituent constraint may be specified as a subordinate to:

- Paragraph;
- Title;
- ListTerm;
- Phrase;
- Caption;
- Description.

A footnote is an amount of content that is logically associated with a particular part of the document body, but which is intended to be read and laid out separately from its associated part of the document. Typically, a footnote consists of a footnote identifier, which is embedded within the document body, and the footnote itself, which is laid out elsewhere.

A Footnote is a composite logical constituent constraint whose immediate subordinates consist of the constituent constraint FootnoteReference, which represents the footnote identifier, followed by the constituent constraint FootnoteBody, which represents the footnote itself. Both of these subordinates are mandatory.

6.2.3.9.2 FootnoteReference

FootnoteReference is a constituent constraint that is used to represent a footnote reference within the body of a document.

FootnoteReference is a basic logical constituent that contains a content generator which, when evaluated, produces a character string which constitutes the footnote reference referred to above.

The generated character string consists of a label with optional prefix and suffix character strings. The label is used to uniquely identify a particular footnote, and may consist of a number which is represented in the form of Arabic or Roman numerals or by an alphabetic equivalent. The number may be automatically generated so that its value is incremented for each successive footnote. Alternatively, the label may consist of a user defined character string.

In a sequence of footnotes, automatic numbers and user defined labels may be freely mixed (for example, giving the sequence 1,2,*,3,4). If the label consists of a user-defined character string, the automatically generated number sequence is not incremented.

An example of a footnote reference is '(2)' where '(' and ')' are user defined prefix and suffix strings respectively and '2' is the automatically generated label. Another example is 'note⁵' where '5' is the label and 'note' is a prefix string which also contains the control function PLU to enable the label to be represented in the form of a superscript. In this case, a suffix string containing the control function PLD would be required to cause the superscripting to be cancelled before the following text.

The format of the content generator referred to above is described in 6.6.7.7.

6.2.3.9.3 FootnoteBody

FootnoteBody is a constituent constraint which represents the content of a footnote. The content consists of a stream of character content which may contain embedded references to other parts of the document.

FootnoteBody is a composite logical constituent constraint whose subordinates consist of the constituent constraint FootnoteNumber, which is mandatory and represents the footnote identifier, followed by a sequence of one or more constituent constraints of the type FootnoteText and Reference which represent the footnote content. The identifier referred to above is identical to the corresponding footnote identifier which is embedded in the content of the document body and represented by the constituent constraint, FootnoteReference.

The constituent constraints subordinate to FootnoteBody are intended to be laid out separately from the other parts of the document content. When a generic layout structure is specified for the document, these constituent constraints are constrained to be laid out in a FootnoteArea frame (see 6.3.5.20).

6.2.3.9.4 FootnoteNumber

FootnoteNumber is a constituent constraint that represents the footnote identifier within the footnote body.

This identifier is identical to the content associated with the constituent constraint FootnoteReference, but is intended to be laid out so that it immediately precedes the content of the footnote body.

FootnoteNumber is a basic logical constituent constraint that contains a content generator which when evaluated produces the identifier referenced above. The format of this content generator is the same as the content generator that may be specified for the constituent constraint FootnoteReference.

It is required to specify one of the layout category names for this constituent constraint, 'Footnote' or 'Footnote-<n>'. This along with a permitted categories attribute of the same name on the footnote frame will ensure that a logical object from this constituent constraint is laid out in a FootnoteArea frame when generic layout structure is specified within the document.

6.2.3.9.5 FootnoteText

FootnoteText is a constituent constraint that is used to represent the footnote content. It is the lowest logical subdivision of a FootnoteBody.

FootnoteText is a basic logical constituent constraint that references one or more content portions each containing processable, formatted or formatted processable character content.

It is required to specify one of the layout category names for this constituent constraint, 'Footnote' or 'Footnote-<n>'. This along with a permitted categories attribute of the same name on the footnote frame will ensure that a logical object from this constituent constraint is laid out in a FootnoteArea frame when generic layout structure is specified within the document.

6.2.3.10 Constituents that provide for a general referencing mechanism

This subclause defines the constituent constraints that are provided to support a general referencing mechanism in a document.

6.2.3.10.1 Reference

A Reference is a constituent constraint which represents a reference consisting of character content that is derived either fully or partially from another part of the document. This constituent constraint provides a general cross-referencing mechanism in a document.

This constituent constraint may be specified as a subordinate to constituent constraints of the types:

- Paragraph;
- Title;
- ListTerm;
- Phrase;
- Caption;
- Description;
- FootnoteBody.

It is a composite constituent constraint whose immediate subordinates may consist of an ordered sequence of constituent constraints of the types BodyText, ReferencedContent and BodyText.

The general format of the content associated with a constituent constraint of the type Reference is:

[<prefix-string>]<reference-string>[<suffix-string>]

The prefix and suffix strings are optional and, if required, are represented by constituent constraints of the type BodyText. The reference string is represented by the constituent constraint, ReferencedContent.

A reference string may, for example, contain references to identifiers such as a number that distinguishes a chapter or section, a table, a figure, a footnote, an item in a numbered list or a page number. Each reference may contain multiple, concatenated references to different parts of a document; a typical example is the reference 'see table 3 in chapter 2 on page 4' where the values '3', '2' and '4' are derived automatically from the appropriate table, chapter and page in the document.

6.2.3.10.2 ReferencedContent

ReferencedContent is a constituent constraint that represents a character string that contains a single reference to content in other parts of the document (see 6.2.3.10.1).

It is a basic logical constituent constraint that is an immediate subordinate to the constituent constraint Reference. It contains a content generator which, when evaluated, produces the character string containing the referenced content.

A sequence of two or more constituent constraints of this type may be used to represent a composite reference string such as 'see table 2 in section 3.1 beginning on page 6', where the strings '2', '3.1' and '6' are automatically generated by referring to number strings attached to particular parts of the document.

The format of this content generator and its evaluation is described in 6.6.7.9.

Where the superior constituent constraint referenced is subordinate to a FootnoteBody, it is required to specify one of the layout category names for this constituent constraint, 'Footnote' or 'footnote-<n>'. This along with a permitted categories attribute of the same name on the footnote frame will ensure that a logical object from this constituent constraint is laid out in a FootnoteArea frame when generic layout structure is specified within the document.

6.2.3.11 Constituents representing illustrations

6.2.3.11.1 Introduction

This profile supports the representation of illustrations or figures consisting of artwork or simple forms.

Artwork typically consists of a diagram or figure which is an image composed of a single content type or which is formed by overlaying two or more separate images consisting of character, raster graphics or geometric graphics content.

A form typically consists of a collection of logical entities, each of which has a certain logical significance. Each logical entity may be further subdivided into subordinate logical entities. A typical example is an order form consisting of a reference number, information about the originator, including the name, address and telephone number, a list of items required and their expected delivery dates.

The entities which make up a form are intended to be laid out in a designated area which is subdivided into areas that are specially reserved for each particular type of entity. This designated area is specified by the frame FormArea when the generic layout structure is present in the document.

Optionally, an illustration may also contain an identifier which may be used to distinguish the illustration from other parts of the document, a caption which may be used to identify the purpose of the illustration, and an amount of associated descriptive text.

Further information concerning the layout of illustrations consisting of artwork or forms is given in 6.4.1.3.7.

The constituent constraints used to represent illustrations are defined below.

6.2.3.11.2 Figure

Figure is a constituent constraint that is typically used to represent an illustration. Such an illustration may consist of artwork or a form as described in 6.2.3.11.1.

A Figure is a composite logical constituent constraint which may be specified as a subordinate to a Passage or NumberedSegment.

The subordinates of a Figure may consist of a sequence, in any order, of the following constituent constraints:

- either Artwork or Form;
- Number;
- Caption;
- Description.

A constituent constraint of the type Artwork or Form must always be present as a subordinate to the constituent constraint Figure. Constituent constraints of the types Number, Caption and Description are independently optional. The constituent constraints may occur in any order, except that a Number and a Caption shall be in this order.

6.2.3.11.3 Artwork

Artwork is a constituent constraint that is typically used to represent a graphical image within an illustration or figure. This may be a simple image that is represented by character, raster graphics or geometric graphics content, or it may be a composite image consisting of a combination of these content types.

This constituent constraint shall only occur as a subordinate to a constituent constraint of the type Figure.

The immediate subordinates of Artwork consist of an arbitrary ordered sequence of one or more of the following constituent constraints:

- Phrase;
- BodyRaster;
- BodyGeometric.

The constituent constraint Phrase is used to enable the image to contain character content which contains references to footnotes or other parts of the document.

6.2.3.11.4 Form

Form is a constituent constraint that is used to represent a simple form within an illustration or figure. It is a composite logical constituent constraint which shall only occur as a subordinate to a constituent constraint of the type Figure.

A Form consists of an arbitrary order sequence of basic entries and composite entries. Basic entries, which may consist of character, raster graphics or geometric graphics content, are represented by subordinate constituent constraints of the type `EntryElement`. Composite entries are represented by subordinate constituent constraints of the type `EntryGroup`.

Constituent constraints of the type `EntryGroup` may be further subdivided into a set of one or more basic or composite entries. Thus a composite entry may be nested to any number of levels such that each level may consist of a set of subordinate basic and composite entries.

This profile does not define a mechanism for defining the semantics associated with each basic or composite entry. This can be achieved by individual applications by making use of the parameter "external data" in the attribute "application comments" (see 6.6.5).

6.2.3.11.5 Caption

Caption is a constituent constraint that typically represents a title or header that is associated with an illustration or figure. This constituent constraint represents character content that may contain embedded references to footnotes and to other content within the document.

A Caption is a composite logical constituent constraint which shall only occur as a subordinate to a constituent constraint of the type `Figure`.

The immediate subordinates of a Caption consist of an arbitrary ordered sequence of one or more of the following constituent constraints:

- `BodyText`;
- `Reference`;
- `Phrase`;
- `Footnote`.

The above constituent constraints may be concatenated to form a continuous stream of character content which is to be laid out as a single unit.

6.2.3.11.6 Description

Description is a constituent constraint that typically represents some general supplementary information that forms part of an illustration that contains a figure or form.

This constituent constraint is a composite logical constituent constraint which shall only occur as a subordinate to a constituent constraint of the type `Figure`.

The immediate subordinates of a Description consist of an arbitrary ordered sequence of one or more of the following constituent constraints:

- `BodyText`;
- `Reference`;
- `Phrase`;

The above constituent constraints may be concatenated to form a continuous stream of character content which is to be laid out as a single unit.

6.2.3.11.7 EntryGroup

EntryGroup is a constituent constraint that represents a composite logical entry within a form.

It is a composite logical constituent constraint which shall only occur as a subordinate to a constituent constraint of the types Form or EntryGroup.

The immediate subordinates of a EntryGroup consist of an arbitrary ordered sequence of one or more constituent constraints of the types EntryElement and EntryGroup. Therefore, the subordinates to EntryGroup may be nested to any number of levels.

NOTE 10 - Constituent constraint EntryElement is defined in 6.2.3.12.6.

6.2.3.12 Constituents used to represent tables

6.2.3.12.1 Introduction

For the purpose of this profile, a table is a logical entity that consists of an ordered sequence of elements, called 'cells', that are arranged into a two dimensional array of rows and columns.

Each row may consist of a 'simple' row which contains a sequence of one or more 'cells'. Alternatively, a row may consist of a 'composite' row which contains a single cell followed by a sequence of one or more subrows, each of which contains a sequence of 'cells'.

The subclauses below define the logical constituents used to represent tables. Figure 28 illustrates the structural relationships between the constituents used to represent a table. Subclause 6.4.1.3.8 describes how tables are intended to be laid out.

6.2.3.12.2 Table

Table is a logical constituent constraint that represents a table as a whole. This constituent constraint may be specified as a subordinate to constituent constraints Passage and NumberedSegment.

The immediate subordinates of this constituent constraint consist of a sequence of constituent constraints Row. Each row may or may not have the same characteristics with regard to its sub-structure.

6.2.3.12.3 Row

Row is a constituent constraint that is a subordinate of the constituent constraint Table and represents a row of entries in a table. This may consist of a sequence of entries, or it may consist of a single entry followed by a sequence of subrows of entries.

In order to represent these two cases, the immediate subordinates of a Row may consist of either:

- a sequence of constituent constraints EntryElement, or;
- a single constituent constraint EntryElement, followed by a single constituent constraint of the type TableComponent.

6.2.3.12.4 TableComponent

TableComponent is a constituent constraint that is a subordinate of the constituent constraint, Row, and represents a sequence of one or more subrows of entries within a row of a table.

The immediate subordinates of this constituent constraint consist of a sequence of one or more constituent constraints of the type RowComponent. Each RowComponent shall have the same characteristics with regard to its sub-structure.

6.2.3.12.5 RowComponent

RowComponent is a constituent constraint that is a subordinate of the constituent constraint TableComponent and represents a sub-row of entries in a row of entries in a table.

The immediate subordinates of this constituent constraint consist of a sequence of one or more components of the type EntryElement. Each EntryElement may or may not have the same characteristics with regard to its sub-structure.

6.2.3.12.6 EntryElement

EntryElement is a constituent constraint that represents a single entry in a form or table. It is a sub-division of constituent constraints of the types Table and Form. In the case of Table, it is specified as a subordinate to a Row or a RowComponent and represents a single entry in a table. In the case of Form, it is specified as a subordinate of a Form itself or of an EntryGroup.

Each entry in a table or form may consist of character, raster graphics or geometric graphics content and, hence, EntryElement has a single immediate subordinate constituent constraint of the type EntryText, EntryRaster or EntryGeometric.

6.2.3.12.7 EntryText, EntryRaster and EntryGeometric

EntryText, EntryRaster and EntryGeometric are constituent constraints which represent content that is to be entered into tables and forms. These constituent constraints may be specified as subordinates of the constituent constraint EntryElement and allow the layout and presentation requirements for the content allocated to tables and forms to be specified.

These are basic logical constituent constraints that directly refer to content portions that contain character, raster graphics and geometric graphics content respectively. EntryText shall refer to one or more content portions which may contain either processable, formatted or formatted processable character content. EntryRaster and EntryGeometric shall only refer to a single content portion containing formatted processable raster graphics content or formatted processable geometric graphics content respectively.

Constituent constraints of these types in the generic logical structure may refer to generic content. This provides the means of defining common content within tables and forms.

6.2.3.13 Constituents representing lists

6.2.3.13.1 Introduction

This profile supports the representation of three types of lists, as follows:

- numbered lists consisting of ordered lists of items, each of which is preceded by an identifier such as an alphabetic character or numeral;
- unnumbered lists consisting of unordered lists of items, each of which may optionally be preceded by a separator such as a hyphen, bullet or small circle;
- definition lists consisting of lists of ordered pairs of items such as a term and its corresponding definition.

Each type of list may be nested without restriction, and one particular type of list may be composed of lists of other types. For example, an item in a numbered list can consist of a subordinate numbered list, unnumbered list or definition list.

The constituent constraints that may be used to represent these list types are defined below.

6.2.3.13.2 NumberedList

A NumberedList is a constituent constraint that is used to represent a list of items, each of which is preceded by an identifier that serves to distinguish that item.

This constituent constraint may be specified as an immediate subordinate to a Passage, NumberedSegment, or ListItem.

The immediate subordinates of a NumberedList consist of the constituent constraint Number which contains a content generator that generates the identifier corresponding to each item in the list, followed by the constituent constraint, ListItem. This pair of constituent constraints may be repeated without limitation.

Further information concerning the numbering of items in a list is contained in 6.6.7.5.

6.2.3.13.3 UnNumberedList

UnNumberedList is a constituent constraint that is used to represent a list of items, each of which may be preceded by an optional separator consisting of character, raster graphics or geometric graphics content.

This constituent constraint may be specified as an immediate subordinate to a Passage, NumberedSegment, or ListItem.

The immediate subordinates of an UnNumberedList consist of a separator, which is represented by a constituent constraint of the type BodyText, BodyRaster or BodyGeometric, followed by the constituent constraint ListItem. This pair of constituent constraints may be repeated without limitation.

6.2.3.13.4 DefinitionList

DefinitionList is a constituent constraint that represents a sequence of ordered pairs of items.

This constituent constraint may be specified as an immediate subordinate to a Passage, NumberedSegment, or ListItem.

The immediate subordinates of this item consist of the constituent constraint ListTerm, followed by the constituent constraint ListItem. This pair of constituent constraints may be repeated without limitation.

6.2.3.13.5 ListItem

ListItem is a constituent constraint that represents an item within a NumberedList, UnNumberedList or DefinitionList. That is, this constituent constraint represents the second element of each pair of elements within a numbered, unnumbered or definition list.

The immediate subordinates of this constituent constraint may consist of a sequence of one or more constituent constraints Phrase, or one of the constituent constraints NumberedList, UnNumberedList or DefinitionList.

Thus this constituent constraint represents an amount of character content that may contain embedded references to footnote and other parts of the document, or it represents a subordinate list of items.

6.2.3.13.6 ListTerm

ListTerm is a constituent constraint that represents a term element within a DefinitionList. A term element is the first item of each pair of items that constitutes a definition list.

The immediate subordinates of this constituent constraint consist of an arbitrary ordered sequence of one or more of the following constituent constraints:

- BodyText;
- BodyRaster;
- BodyGeometric;
- Phrase;
- Reference;
- Footnote.

Thus this constituent constraint represents an amount of character, raster graphics and geometric graphics content. Character content may contain embedded phrases, references and footnotes.

6.2.4 Common content part of the logical structure

6.2.4.1 CommonContent

CommonContent is a constituent constraint that represents common content that is laid out in the body, header, or footer area of the pages of a document. Common content consists of any combination of character, raster graphics and geometric graphics content.

Any number of constituent constraints CommonContent may be contained in a document. CommonContent is a composite logical object class whose immediate subordinates consist of an arbitrary ordered sequence of one or more of the following constituent constraints:

- CommonText;
- CommonRaster;
- CommonGeometric;
- CommonReference;
- CurrentInstance;
- TableNumber;
- PageNumber;
- CommonNumber.

When the generic layout structure is present, constituent constraints of the type CommonContent and their associated subordinate constituent constraints are constrained to be laid out in a specified frame within a body, header or footer area using the 'logical source' mechanism (see 6.3.6).

6.2.4.2 CommonText

CommonText is a constituent constraint that represents the common character content that is to be laid out in a specified area within a page.

CommonText is a constituent constraint for a basic logical object class that references one or more content portions each containing either processable, formatted and formatted processable character content.

6.2.4.3 PageNumber

PageNumber is a constituent constraint that represents the common character content that is to be laid out in a specified area in a page. This constituent constraint is specifically used when it is required to represent an automatically generated page number.

PageNumber is a basic logical object class that contains a content generator. This content generator contains a reference to a page number which is automatically evaluated when the document is laid out. For example, this provides the means of representing the page numbers that are displayed on the consecutive pages of a document.

Each page number consists of a single number which may be represented in the form of Arabic or Roman numerals or in its alphabetic equivalent. Page numbering schemes may start at 0 or any value greater than 0. The page number that is generated may have a prefix or suffix character string.

The format of the content generators is defined in 6.6.7.8.

6.2.4.4 CommonRaster

CommonRaster is a constituent constraint that represents the common raster graphics content that is to be laid out in a specified area within a page. For example, this constituent constraint may be used to represent a logo which is to be laid out on each page of a document.

Common raster is a constituent constraint for a basic logical object class which references a single content portion containing formatted processable raster graphics content.

6.2.4.5 CommonGeometric

CommonGeometric is a constituent constraint that represents the common geometric graphics content that is to be laid out in a specified area within a page. For example, this constituent constraint may be used to represent a graphical icon which is to be laid out on each page of a document.

CommonGeometric is a constituent constraint for a basic logical object class which references a single content portion containing formatted processable geometric graphics content.

6.2.4.6 CommonReference

CommonReference is a constituent constraint that represents the common character content that is to be laid out in a specified area in a page and which represents a character string that contains references to other parts of the document. Such a reference may consist of a reference to a number that relates to a segment, table, figure, footnote or page number.

CommonReference is a constituent constraint for a basic logical object class that contains a content generator which, when evaluated, produces a character string containing the references. The format of this reference string is defined in 6.6.7.10.

6.2.4.7 CurrentInstance

CurrentInstance is a constituent constraint that represents the common character content that is to be laid out in a specified area in a page. This constituent constraint is specifically used when it is required to refer to a character string that is attached to any logical constituent constraint in the document or any of the layout constituent constraints DocumentLayoutRoot, PageSet, RectoPage, VersoPage or Page. The number string may represent, for example, the title of a sub-section or table that is contained elsewhere in the document.

CurrentInstance is a constituent constraint for a basic logical object class that contains a content generator which, when evaluated, produces a copy of the character string associated with a specified constituent. The format of this reference is defined in 6.6.7.11.

6.2.4.8 TableNumber

TableNumber is a constituent constraint that represents the common character content that is to be laid out in a specified area in a page. This constituent constraint is specifically used when it is required to represent a table number which is to be placed within the header area of a table.

TableNumber is a constituent constraint for a basic logical object class that contains a content generator which, when evaluated, generates the required table number. The format of the content generator is defined in 6.6.7.6.

6.2.4.9 CommonNumber

CommonNumber is a constituent constraint that represents the common character content that is to be laid out in a specified area in a page. This constituent constraint is specifically used when it is required to refer to character content consisting of a number string which can, for example, represent the number of a sub-section or a current page.

CommonNumber is a constituent constraint for a basic logical object class that contains a content generator which, when evaluated, generates the reference to the required number. The format of this reference is defined in 6.6.7.12.

6.3 Layout characteristics

This subclause defines the layout constituent constraints provided by this profile to represent the characteristics of documents.

Different constituent constraints may be used to represent and distinguish parts of a document that have different layout characteristics. This subclause describes the general characteristics and typical uses of the constituent constraints that are provided.

The descriptions of the layout characteristics represented by each of the constituent constraints is provided for guidance only. It is the responsibility of the user to determine how a document is to be represented using the constituents provided. Adherence to these guidelines can enhance the mutual understanding of a document by an originator and a recipient.

6.3.1 Overview of the layout characteristics

The document structure allows the document content to be laid out and presented in one or more page sets. Each page set may be used for different parts of the document, for example, the title page, foreword, table of contents, document body and appendices.

Each page set consists of a series of pages. In general, each page may be sub-divided into three areas: the body area, which is used to layout the document body; and the header and footer areas, which may be used to layout the common content.

NOTE 11 - In the case of FOD36, common content may also be laid out in the body area, as well as the header/footer area.

Three body layout types are supported by this profile. Each body layout type specifies how the body is positioned within each page, and how the content may be presented within the body. These are referred to as body layouts A, B, and C, and are defined in 6.3.4.5.

It is intended that all applications which use this profile shall support body layout A, whereas support for the other two body layouts may be specified as optional.

Body layout A is used when the character content is to be laid out horizontally (from left to right or from right to left) and from top to bottom within the body area. This layout is typically used for contents written in Latin based, Hebrew, Arabic, and Japanese (most cases) languages.

Body layout B is used when the character content is to be laid out vertically (bottom to top or top to bottom) and from left to right within the body area. This layout is typically used for contents written in Latin based, Hebrew, Arabic, and Japanese (most cases) languages in which it is required to layout the content in landscape orientation within the body area of the page.

Body layout C is used when the character content is to be laid out vertically and from right to left within the body area. This layout may be typically used for contents written in languages which use ideograms, such as Japanese and Chinese characters.

The body, header and footer areas may be further sub-divided into areas to support a wide range of different layout requirements. These features are described further in 6.3.5 and 6.3.6.

6.3.2 DocumentLayoutRoot

DocumentLayoutRoot is a constituent constraint that represents the top level in the document layout structure. Its immediate subordinates consist of a sequence of one or more constituent constraints of the type, PageSet. The numbering schemes for pages may be initialized on this constituent constraint.

6.3.3 PageSet

PageSet is a constituent constraint that represents a grouping of pages within a document. A PageSet is typically used to represent a part of a document that has different layout requirements from other parts of a document. Also, a PageSet may correspond to a part of a document that has a certain logical significance, for example, a PageSet might represent the front matter in a document or an individual chapter.

Only one level of PageSet is allowed in a document. However, a document may contain any number of class definitions of the type PageSet which may be used, for example, to provide a choice of alternative layouts for different parts of a document or to specify the exact layout requirements for each successive part of a document.

The immediate subordinates of a PageSet consist of a combination of constituent constraints of the types Page, RectoPage and VersoPage, as described in 6.3.4.1.

6.3.4 Page characteristics

6.3.4.1 Page constituents

Three constituent constraints are provided to represent the pages within a document, namely Page, RectoPage and VersoPage.

The pages that make up a page set consist of an arbitrary sequence of one or more of the constituent constraints Page, RectoPage and VersoPage.

The only difference in the characteristics of these constituent constraints concerns the values that may be specified for the parameter "side of sheet" in the attribute "medium type". In the case of Page, the value of this parameter may be specified as 'recto', 'verso' or 'unspecified'. In the case of RectoPage, the value of this parameter may be specified as 'recto' or 'unspecified'; in the case of VersoPage, the value of this parameter may be specified as 'verso' or 'unspecified'. The values, 'recto' and 'verso', of the "side of sheet" parameter of the "medium type" attribute are non-basic.

The following restrictions also apply to the pages within a page set:

- a) all the pages shall have the same dimensions, but may differ in orientation (see 6.3.4.2);
- b) all the pages are to be laid out on the same size of presentation medium (see 6.3.4.3);
- c) all the pages instantiated from a given page class shall have the same layout characteristics (see note). That is, for a given page class, there is not a choice of layout characteristics. However, the layout characteristics of pages in a page set may or may not be the same.

NOTE 12 - The layout characteristics of pages are specified in 6.3.4.5. Pages having the same layout characteristics are pages for which the body area, header area (if present) and footer area (if present) have the same dimensions and position within the page (see 6.3.4.3). However, pages having the same layout characteristics do not necessarily have the same position on the presentation medium (see 6.3.4.4).

6.3.4.2 Page dimensions

The dimensions of the pages may be specified as any value (in SMUs) that is equivalent to or less than ISO A0 or ANSI E paper sizes. The dimensions may be specified in portrait or landscape orientation. Japanese page sizes B4 and B5 are also supported, but the dimension of these pages lie within the range of dimensions given above.

Dimensions equivalent to or less than the common assured reproduction area of ISO A4 and ANSI-A in portrait or landscape orientation are basic values. Larger page sizes are non-basic and their use shall be indicated in the document profile.

Any default page dimensions may be specified in the document profile subject to the maximum dimensions defined above.

NOTE 13 - The size termed "North American Letter (NAL)" in [CCITT Recommendation T.410 series | ISO 8613] (e.g., in [CCITT Recommendation T.412 | ISO 8613-2] clause 7) is called "ANSI-A" in this specification to be consistent with the other reference to ANSI standard paper sizes.

6.3.4.3 Nominal page sizes

The nominal page sizes that may be specified are listed in table 1. These may be specified in portrait or landscape orientation. All values of nominal page size are non-basic and hence all values used in a document shall be indicated in the document profile.

Any value of nominal page size defined in table 1, subject to the restrictions specified above, may be specified as the default value in the document profile.

Table 1 also includes the recommended assured reproduction area (ARA). Information loss may occur when a document is reproduced if the dimensions of constituent constraints of the type page exceed the ARA for the specified nominal page size.

Table 1 — Nominal page sizes

Page type	Size in inches or millimeters	Size in BMUs	ARA in BMUs
ISO A5	148mm x 210mm	7015 x 9920	not defined
ISO A4	210mm x 297mm	9920 x 14030	9240 x 13200
ISO A3	297mm x 420mm	14030 x 19840	13200 x 18480
ISO A2	420mm x 594mm	19840 x 28060	18898 x 27118
ISO A4	594mm x 841mm	28060 x 39680	26173 x 37843
ISO A4	841mm x 1189mm	39680 x 56120	37843 x 54283
ANSI legal	8.5in. x 14in.	10200 x 16800	9240 x 15480
ANSI D	8.5in. x 14in.	10200 x 13200	9240 x 12400
ANSI E	11in. x 17in.	13200 x 20400	12744 x 19656
ANSI C	17in. x 22in.	20400 x 26400	19500 x 25800
ANSI D	22in. x 34in.	26400 x 40800	25800 x 39600
ANSI E	34in. x 44in.	40800 x 52800	39600 x 52200
ANSI F	28in. x 40in.	33600 x 48000	32400 x 47400
Japan-legal	257mm x 364mm	12141 x 17196	11200 x 15300
Japan-letter	182mm x 257mm	8598 x 12141	7600 x 10200

6.3.4.4 Page offset

The page offset is the distance of the position of the left and top edges of the page relative to the left and top edges respectively of the presentation medium on which each page is reproduced. Any value of page offset may be specified provided that no part of the page area lies outside the area of the nominal page. Also, page offsets specified for the initial, recto and verso pages within a given page set may differ. The default page offset may be specified in the document profile.

6.3.4.5 Page layout characteristics

Each page in a document may be subdivided into three rectangular areas, as follows:

- a body area, which is reserved for content that belongs to the body part of the document (see 6.3.5);
- a header area, which is reserved for common header content (see 6.3.6);
- a footer area, which is reserved for common footer content (see 6.3.6).

The body area is mandatory and shall occur on every page in a document. The header and footer areas are both optional.

Also these three areas shall be entirely contained within the page area and must not overlap.

Three types of layout of body area are defined:

- a) Body layout type A. In this case, the layout path in the body area is specified as 270 degrees.
- b) Body layout type B. In this case, the layout path in the body area is specified as 0 degrees.
- c) Body layout type C. In this case, the layout path in the body area is specified as 180 degrees.

In addition, four types of layout of header/footer area are defined:

- a) H/F layout A1. In this case, the layout path in the header and footer area is 270 degrees. If the header or footer area is composite, the layout paths in the lowest frames are 270 degrees;
- b) H/F layout A2. In this case, the layout path in the header and footer area is 0 degrees. If the header or footer area is composite, the layout paths in the lowest frames are 270 degrees;
- c) H/F layout B1. In this case, the layout path in the header and footer area is 180 degrees. If the header or footer area is composite, the layout paths in the lowest frames are 180 degrees;
- d) H/F layout B2. In this case, the layout path in the header and footer area is 270 degrees. If the header or footer area is composite, the layout paths in the lowest frames are 180 degrees.

Page layout type is determined by a combination of the body layout type and the H/F layout type. Any combination is permitted. However, it is intended that all applications which use this profile shall support the combinations of body layout A and H/F layout A1 and A2, whereas support for other combinations may be specified as optional.

NOTE 14 - The combinations of body layout A and H/F layout A1, B and A1, C and A1, and C and B1 are equivalent to page layouts A, B, C and D in FOD26 respectively.

The header and footer of H/F layout A1 or A2 are laid out above and below the body area. Figure 1 illustrates this case, and figure2 illustrates H/F layout type A1 and A2 corresponding to this case.

the header and footer of H/F layout B1 or B2 are laid out to the right and left of the body area. Figure illustrates this case, and figure 4 illustrates H/F layout type B1 and B2 corresponding to this case.

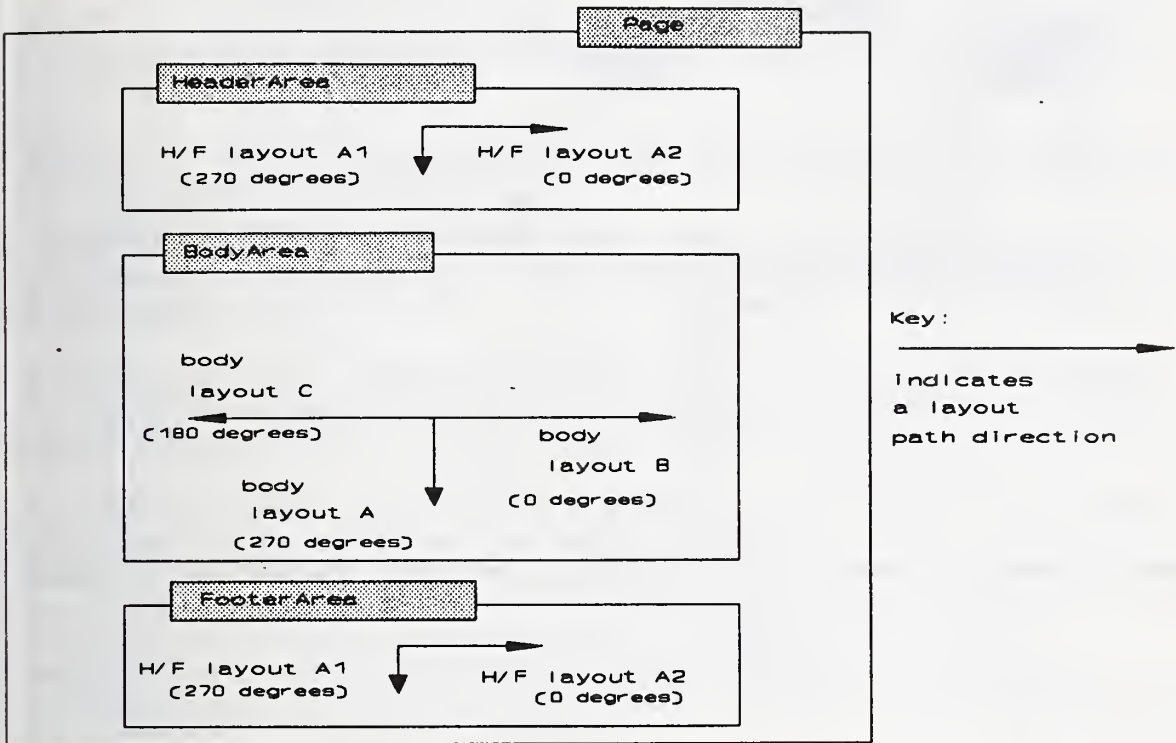


Figure 1 — Body layout types A, B and C with header and footer above and below the body area

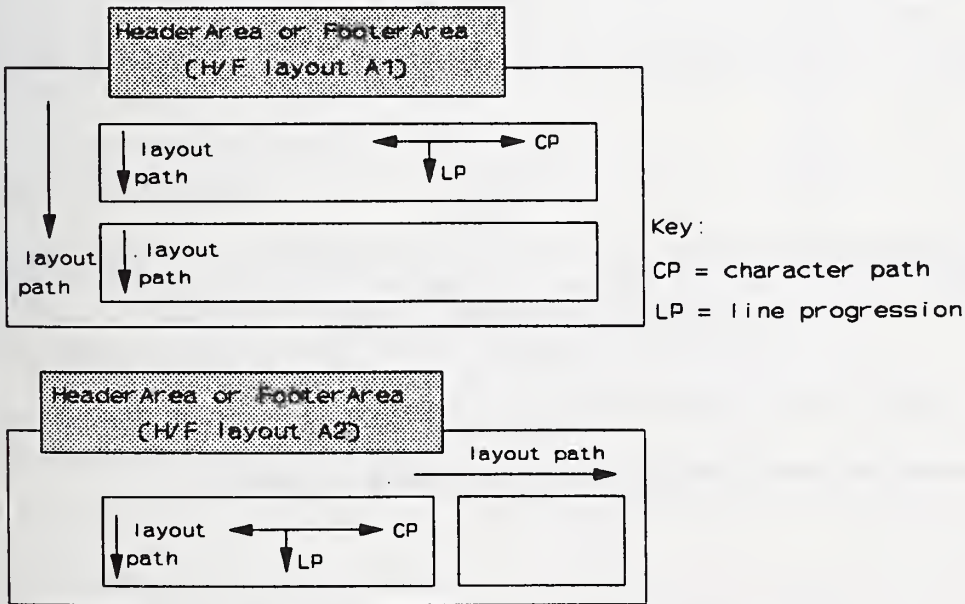


Figure 2 — Header and footer frame layouts corresponding to Figure 1

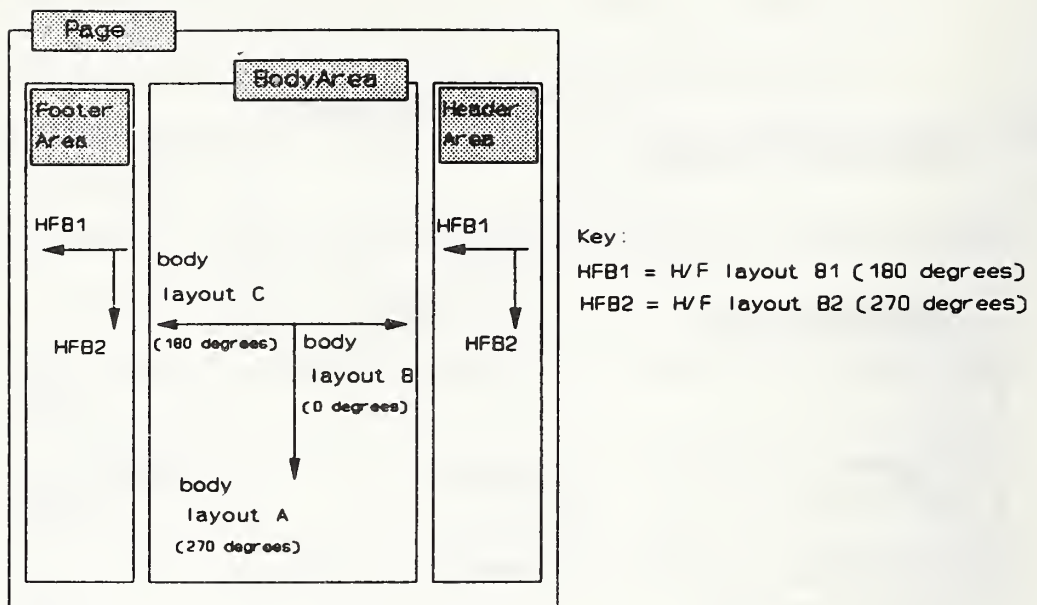


Figure 3 — Body layout types A, B and C with header and footer to the right and left of the body area

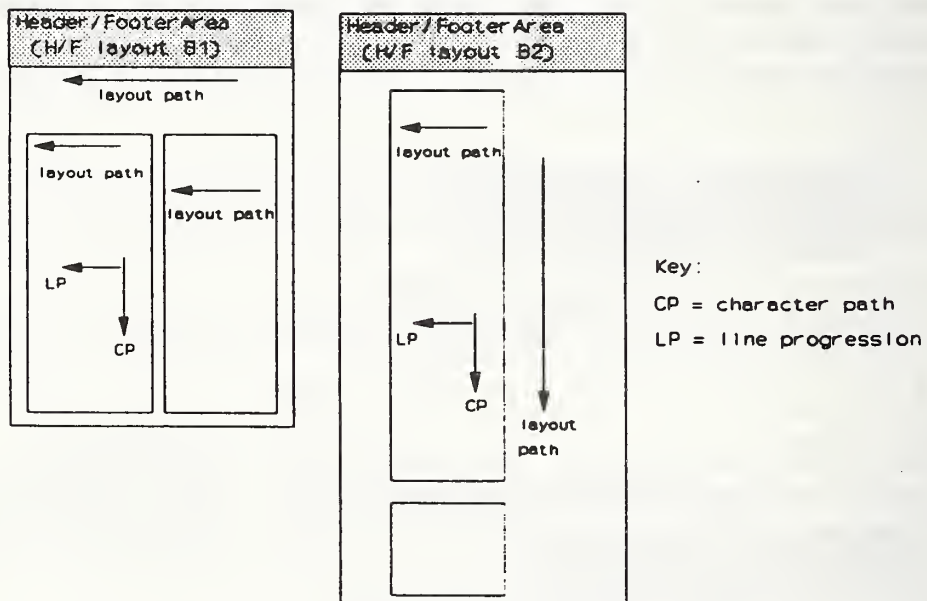


Figure 4 — Header and footer frame layouts corresponding to Figure 3

6.3.5 Body area characteristics

6.3.5.1 General characteristics

The body area is the area within a page where the main matter of the document, that is the 'body' part of the document, is laid out. The layout path specified in the body area determines the body layout type being used.

The body area may consist of a single frame into which the content is directly laid out. In this case, the body area is represented by a BasicBody frame.

Alternatively, the body area may be subdivided into different rectangular areas to provide for different layout requirements. In this case, the body frame is represented by a VariableCompositeBody or FixedCompositeBody frame.

The subordinate areas within a VariableCompositeBody frame are represented by variably positioned frames. Thus the subordinate areas are not pre-determined and are automatically adjusted during the layout process to accommodate the content that is allocated to them.

When a FixedCompositeBody frame is used, the subordinate areas are represented by fixed positioned frames, and hence the body area layout can be precisely specified.

However, in order to provide both areas whose layout is fixed and areas whose layout is variable within a single body area, it is possible to specify one or more VariableCompositeBody frames as subordinates to a FixedCompositeBody frame. In this case, the layout paths for each of the VariableCompositeBody frames may or may not be the same. This allows, for example, text belonging to different languages to be laid out on the same page.

6.3.5.2 BasicBody

BasicBody is a constituent constraint which defines a lowest level frame which represents a body area into which content is directly laid out.

The position and dimensions of this frame are fixed. The layout path specified depends upon the body layout type being used (see 6.3.4.5).

6.3.5.3 VariableCompositeBody

VariableCompositeBody is a constituent constraint that defines a composite frame which represents either the entire body area or a part of it, and which contains one or more subordinate variably positioned frames. A VariableCompositeBody frame has a fixed position and fixed dimensions. The layout path specified for this frame depends upon the layout type used (see 6.3.1).

The immediate subordinates of this frame consist of an arbitrary ordered sequence of one or more frames of the following constituent constraints:

- BasicFloat;
- SnakingColumns;
- SynchronizedColumns;

- CompositeFloat;
- CompositeFixtureVariable;
- TableArea;
- FootnoteArea;
- ArrangedContentVariable;
- SourcedContentVariable.

The subordinate frames are all variably positioned and have variable dimensions except for ArrangedContentVariable frames. Thus the relative positions of these frames in the body area may vary and depend upon the positions of other frames (if any) that are placed in the same VariableCompositeBody frame.

The layout path for VariableCompositeBody frames may be specified as 270, 0 or 180 degrees. This determines the body layout type used in the case where VariableCompositeBody represents the entire body area (see 6.3.4.5).

All immediate subordinate frames are laid out along the layout path specified (in 'normal' positioning fill order). FootnoteArea frames are laid out in the same direction as the body area layout path, but reverse fill order is used.

Also frames are constrained to have the same layout path as the VariableCompositeBody frame to which they are subordinate. However, exceptions to this rule are frames of the types CompositeFixtureVariable, CompositeFloat, SnakingColumns and SourcedContentVariable (see appropriate subclause below).

Figures 5, 6 and 7 provide illustrations of the layout of frames within a VariableCompositeBody frame for the various body layout types.

A choice of subordinate frames of the types listed above may be specified for a VariableCompositeBody frame. Different frame types may be selected using various layout directives (see 6.4) and, therefore, the layout characteristics of the body areas within a page set may change from page to page within a page set.

6.3.5.4 FixedCompositeBody

FixedCompositeBody is a constituent constraint that defines a composite frame which represents the body area and which contains one or more subordinate frames that are fixed in position. The position and dimensions of this frame are fixed.

The immediate subordinates of frames of this type consist of an arbitrary ordered sequence of one or more frames of the following constituent constraints:

- BasicFixture;
- ColumnFixed;
- CompositeCommon;
- CompositeFixtureFixed;

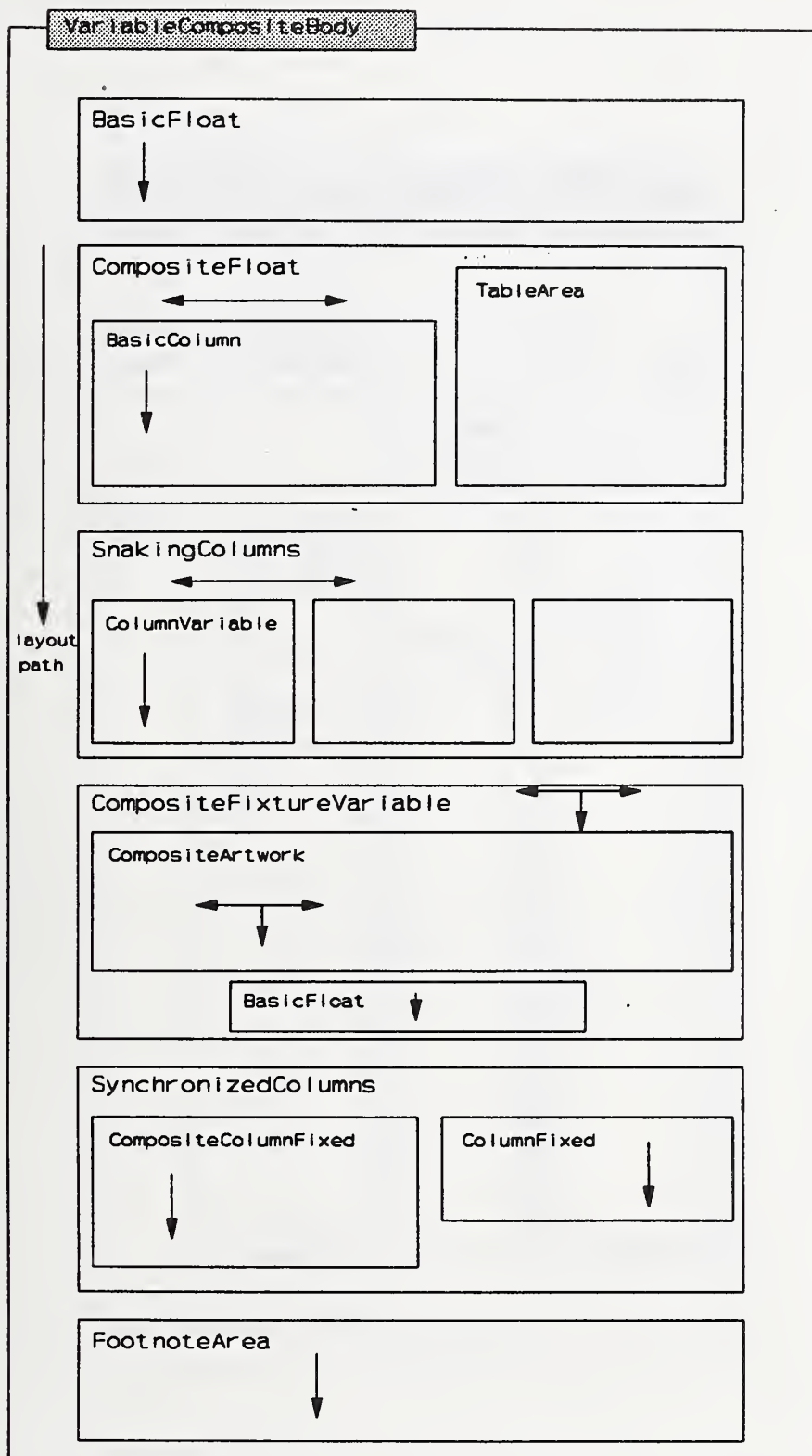


Figure 5 — Example of body area layout for body layout type A

— ArrangedContentFixed;

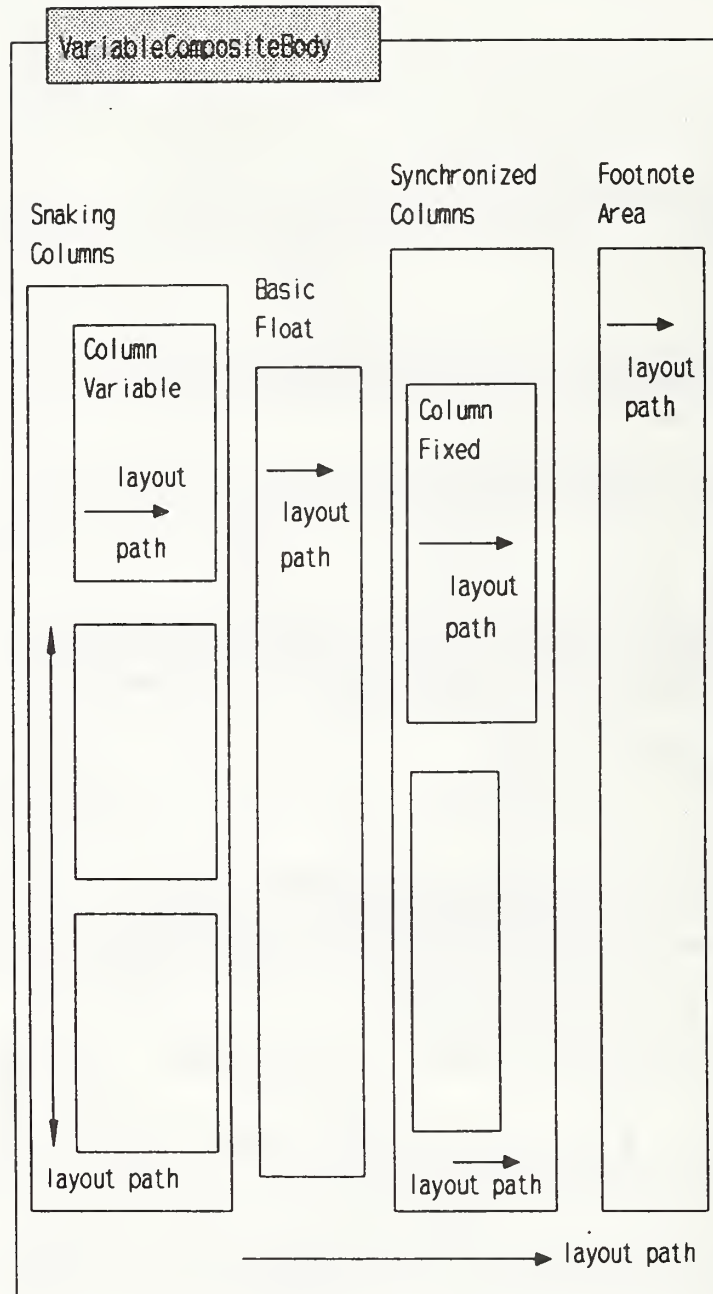


Figure 6 — Example of body area layout for body layout type B

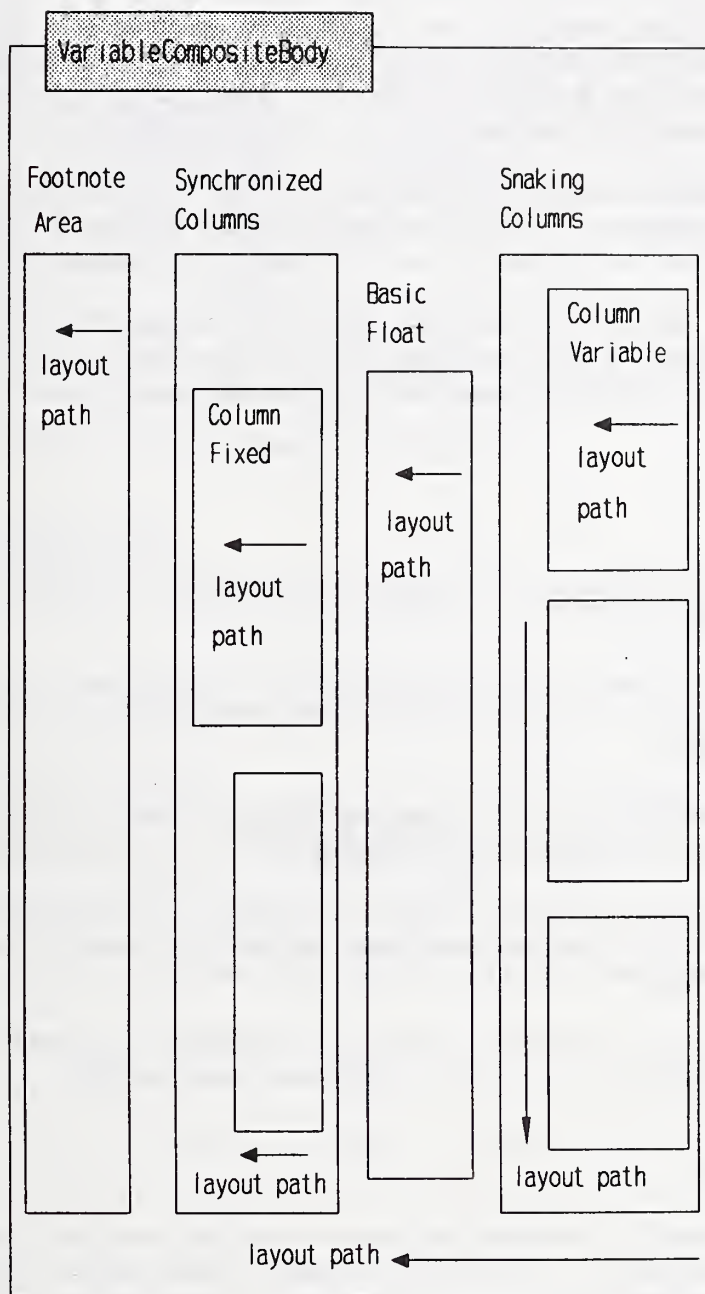


Figure 7 — Example of body area layout for body layout type C

- SourcedContentFixed;
- VariableCompositeBody.

The subordinate frames may overlap without restriction.

The layout path for FixedCompositeBody frames may be specified as 270, 0 or 180 degrees. This determines the body layout type used (see 6.3.4.5). The layout path specified does not affect the positioning of frames, but it may affect their dimensions since some of the frames listed above have variable dimensions in a particular direction.

Also frames are constrained to have the same layout path as the FixedCompositeBody frame to which they are subordinate. However, exceptions to this rule are frames of the types VariableCompositeBody, CompositeFixtureFixed and SourcedContentFixed (see appropriate subclause below).

A choice of subordinate frames of the types listed above may be specified for a FixedCompositeBody frame. Different frame types may be selected using various layout directives (see 6.4), and, therefore, the layout characteristics of the body areas within a page set may change from page to page within a page set.

6.3.5.5 BasicFloat

BasicFloat is a constituent constraint that defines a lowest level frame that is used to represent a single column area within a body area.

This is a variably positioned frame which may be specified as a subordinate to frames of the types VariableCompositeBody, CompositeColumnVariable, CompositeColumnFixed, CompositeFixtureVariable and CompositeFixtureFixed.

The dimension of this frame in the direction orthogonal to the layout path of the body area is fixed or defaults to the maximum value allowed within the body area.

The dimension of this frame in the direction parallel to the layout path of the body area is specified as 'Rule-B'. This dimension is therefore automatically adjusted during the layout process to be the minimum required to contain all the content allocated to the frame.

The layout path specified for this frame is the same as that specified for its superior frame. Content shall only be laid out in this frame in the direction of the layout path specified.

6.3.5.6 SnakingColumns

SnakingColumns is a constituent constraint that defines a composite frame that represents a snaking columns area within a body area. A snaking columns area is typically used for the layout of one or more columns of content in which the content is allowed to flow freely from one column to the next. Examples are shown in figure 8 and figure 9.

This is a variably positioned frame which may only be specified as a subordinate to a VariableCompositeBody frame.

Its immediate subordinates consist of an arbitrary ordered sequence of one or more frames of the following constituent constraints:

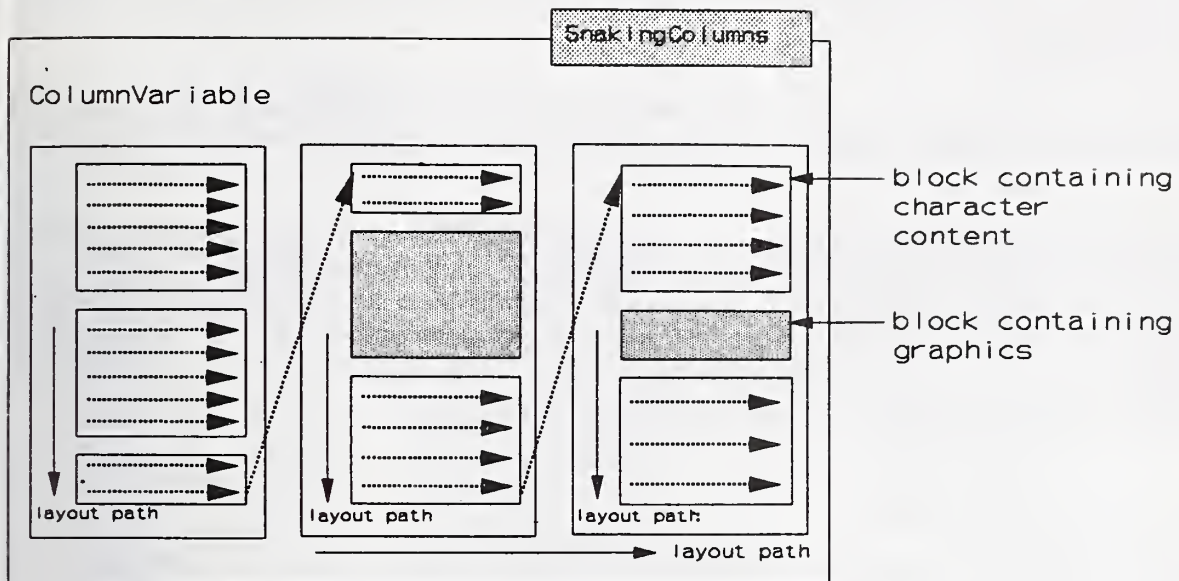


Figure 8 — Example of the layout of a snaking columns frame

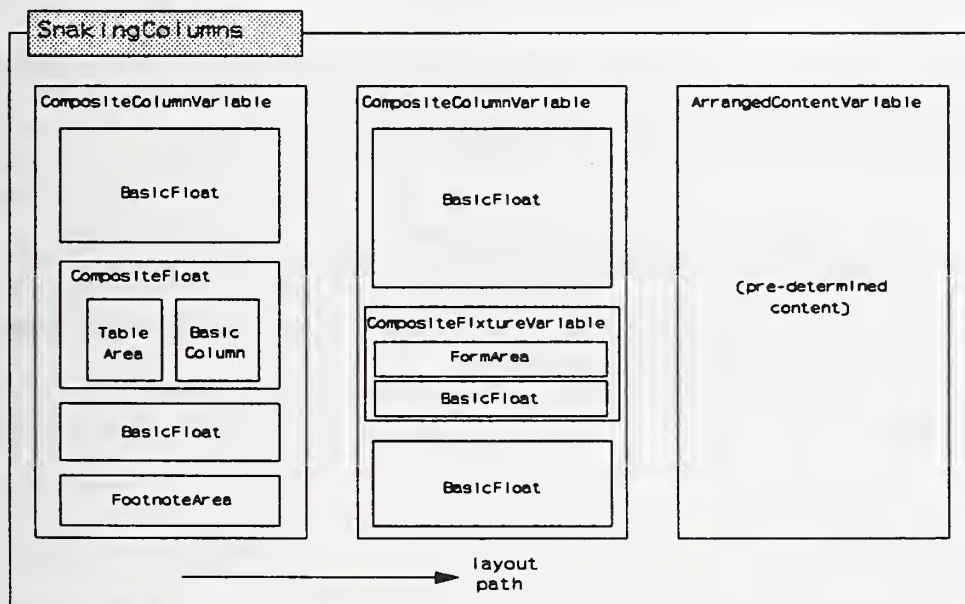


Figure 9 — Another example of the layout of a snaking columns frame

- ColumnVariable;
- CompositeColumnVariable;
- ArrangedContentVariable;
- SourcedContentVariable.

The dimension of the **SnakingColumns** frame in the direction orthogonal to the layout path of the body area is fixed or defaults to the maximum value allowed within the body area.

The dimension of this frame in the direction parallel to the layout path of the body area is specified as 'Rule-B'. This dimension is therefore automatically adjusted to accommodate the subordinate frames which are laid out in it.

The layout path for a SnakingColumns frame may be specified as 0 or 180 degrees in the case of body layout A, 90 or 270 degrees in the case of body layout B, and 270 degrees in the case of body layout C.

The attribute "balance" may be specified for a SnakingColumns frame to indicate that two or more of the subordinate ColumnVariable frames are to be approximately equal in length in the vertical dimension in the case of body layout A and approximately equal in length in the horizontal dimension in the cases of body layouts B and C. Note that "approximately equal" in the context of the "balance" attribute means that the leading edges of the layout objects being balanced are aligned as closely as possible to a line orthogonal to the layout path for the objects.

6.3.5.7 SynchronizedColumns

SynchronizedColumns is a constituent constraint that defines a composite frame that represents a synchronized columns area within a body area. A synchronized columns area is typically used to represent one or more columns of content such that the content laid out in each column belongs to different layout streams. Thus content laid out in one column is not allowed to flow into the next column.

This type of column layout is typically used when it is required to layout separate amounts of content in parallel with one another such that they are aligned. Examples are the synchronized layout of content belonging to different languages and the layout of a figure in parallel with some text. An example is shown in figure 10.

With regard to positioning and dimensioning, SynchronizedColumns frames have the same characteristics as SnakingColumns frames.

The immediate subordinates of a SynchronizedColumns frame consist of an arbitrary ordered sequence of one or more of the following constituent constraints:

- ColumnFixed;
- CompositeColumnFixed;
- ArrangedContentFixed;
- SourcedContentFixed.

The layout path for a SynchronizedColumns frame is 270 degrees for body layout A, 0 degrees for body layout B and 180 degrees for body layout C.

6.3.5.8 CompositeFloat

CompositeFloat is a constituent constraint that defines a composite frame that specifies an area which is used for the side by side layout of complex objects such as figures, forms or tables and simple columns of text.

This is a variably positioned frame which may be specified as a subordinate to a VariableCompositeBody, CompositeColumnVariable or CompositeColumnFixed frame.

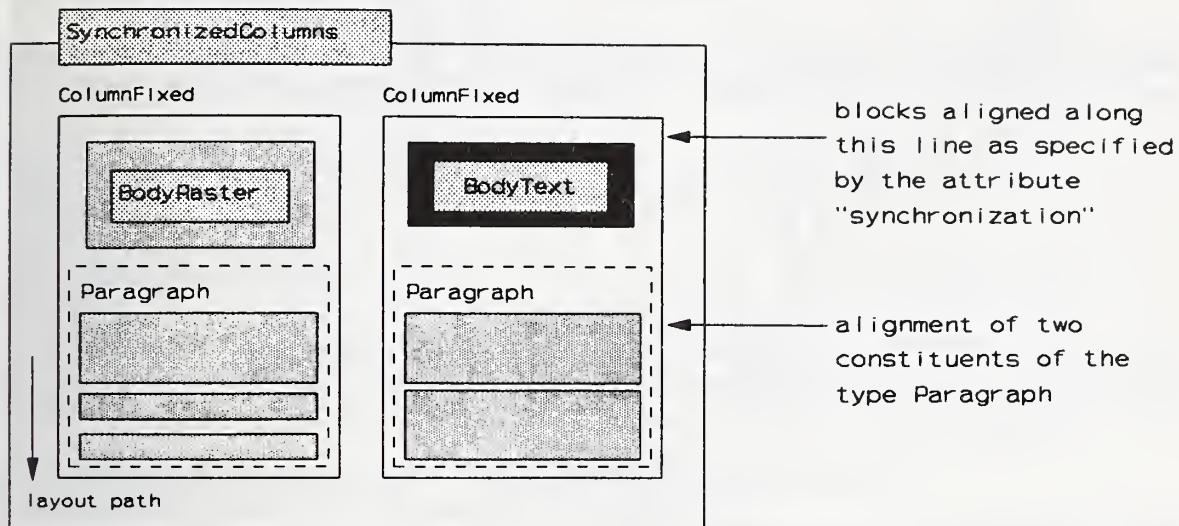


Figure 10 — Example of the layout of a synchronized column

The dimension of a CompositeFloat frame in the direction orthogonal to the layout path of its superior frame is fixed or defaults to the maximum value allowed.

The dimension of the frame in the direction parallel to the layout path is specified by 'Rule-A'. Thus the dimension in this direction is determined by the dimension in the same direction of the first frame that is laid out in the CompositeFloat frame.

The immediate subordinates of this frame consist of an arbitrary ordered sequence of one or more of the following constituent constraints:

- BasicColumn;
- CompositeFixtureVariable;
- TableArea;
- ArrangedContentVariable;
- SourcedContentVariable.

The layout path for a CompositeFloat may be specified as 0 or 180 degrees in the case of body layout A and as 90 or 270 degrees in the case of body layouts B and C.

A typical example of the use of this constituent is a form or an illustration with character content flowing to its side as shown in figure 11.

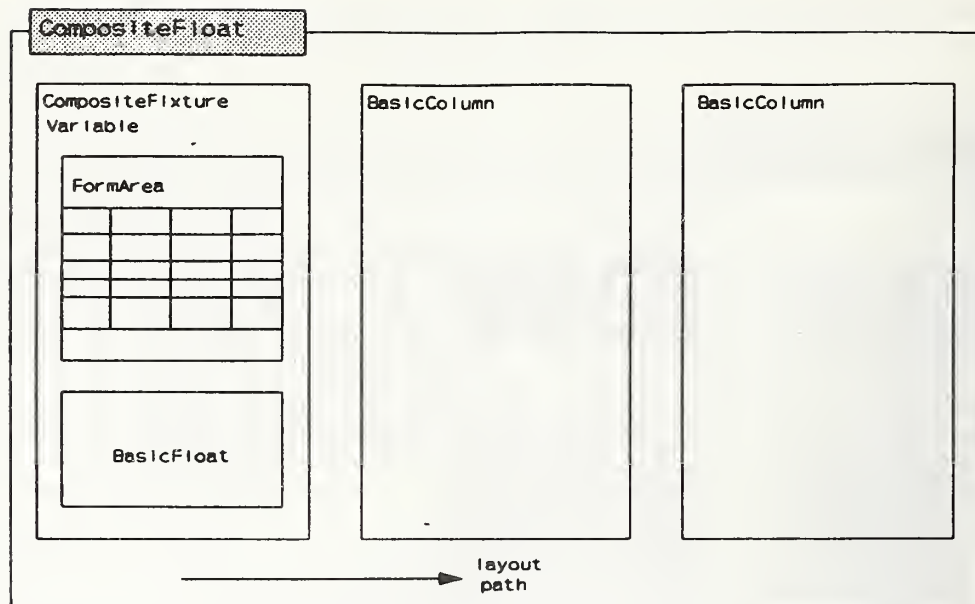


Figure 11 — Example of a CompositeFloat frame

6.3.5.9 CompositeFixtureVariable

CompositeFixtureVariable is a constituent constraint that defines a composite frame used to specify an area which is used to layout an illustration, such as a figure or form, with which is associated some descriptive text and a caption. Hence, the prime purpose of this frame is to layout logical constituent constraints of the type Figure.

This is a variably positioned frame which may be specified as a subordinate to a VariableCompositeBody, CompositeColumnVariable or CompositeColumnFixed frame.

The dimension of a CompositeFixtureVariable frame in the direction orthogonal to the layout path of the superior frame is fixed, specified as 'Rule-B' or defaults to the maximum allowed. In the direction parallel to the layout path of the superior frame, the dimension may be fixed or specified as 'Rule-B'.

The immediate subordinates of this frame consist of an arbitrary ordered sequence of one or more of the following constituent constraints:

- BasicFloat;
- CompositeArtwork;
- FormArea;
- FootnoteArea.

NOTE 15 - The layout path of a CompositeFixtureVariable frame may be set to 0, 180 or 270 degrees in the case of body layout A, to 0, 90 or 270 degrees in the case of body layout B and 180 or 270 degrees in the case of body layout C.

All subordinate frames are laid out in 'normal' positioning fill order with the exception of FootnoteArea frames which are laid out in 'reverse' fill order.

Examples of the layout of CompositeFixtureVariable frames are shown in figure 12.

6.3.5.10 CompositeFixtureFixed

CompositeFixtureFixed is a constituent constraint that defines a composite frame which has the same characteristics as that of a CompositeFixtureVariable frame, except for the positioning and dimensions.

This frame may be specified as a subordinate to a FixedCompositeBody frame.

6.3.5.11 CompositeArtwork

CompositeArtwork is a constituent constraint that defines a composite frame that represents an area that contains an illustration such as a figure or diagram. It is used for laying out logical constituents of the type Artwork which are subordinate to logical constituent constraints of the type Figure.

This is a variably positioned frame that may be specified as a subordinate to a CompositeFixtureVariable or CompositeFixtureFixed frame.

The dimensions of this frame in the directions orthogonal and parallel to the layout path of the superior frame may be independently either fixed or specified as 'Rule-B'.

The immediate subordinates of a CompositeArtwork frame consist of a sequence of one or more lowest level frames of the type BasicFixture which contain character, raster graphics or geometric graphics content. BasicFixture frames may overlap to allow composite images to be formed.

The layout path of a CompositeArtwork frame is set to be the same as that of its superior frame (see note).

NOTE 16 - Subject to this restriction, this means that the layout path of a CompositeArtwork frame may be set to 0, 180 or 270 degrees in the case of body layout A, to 0, 90 or 270 degrees in the case of body layout B and 180 or 270 degrees in the case of body layout C.

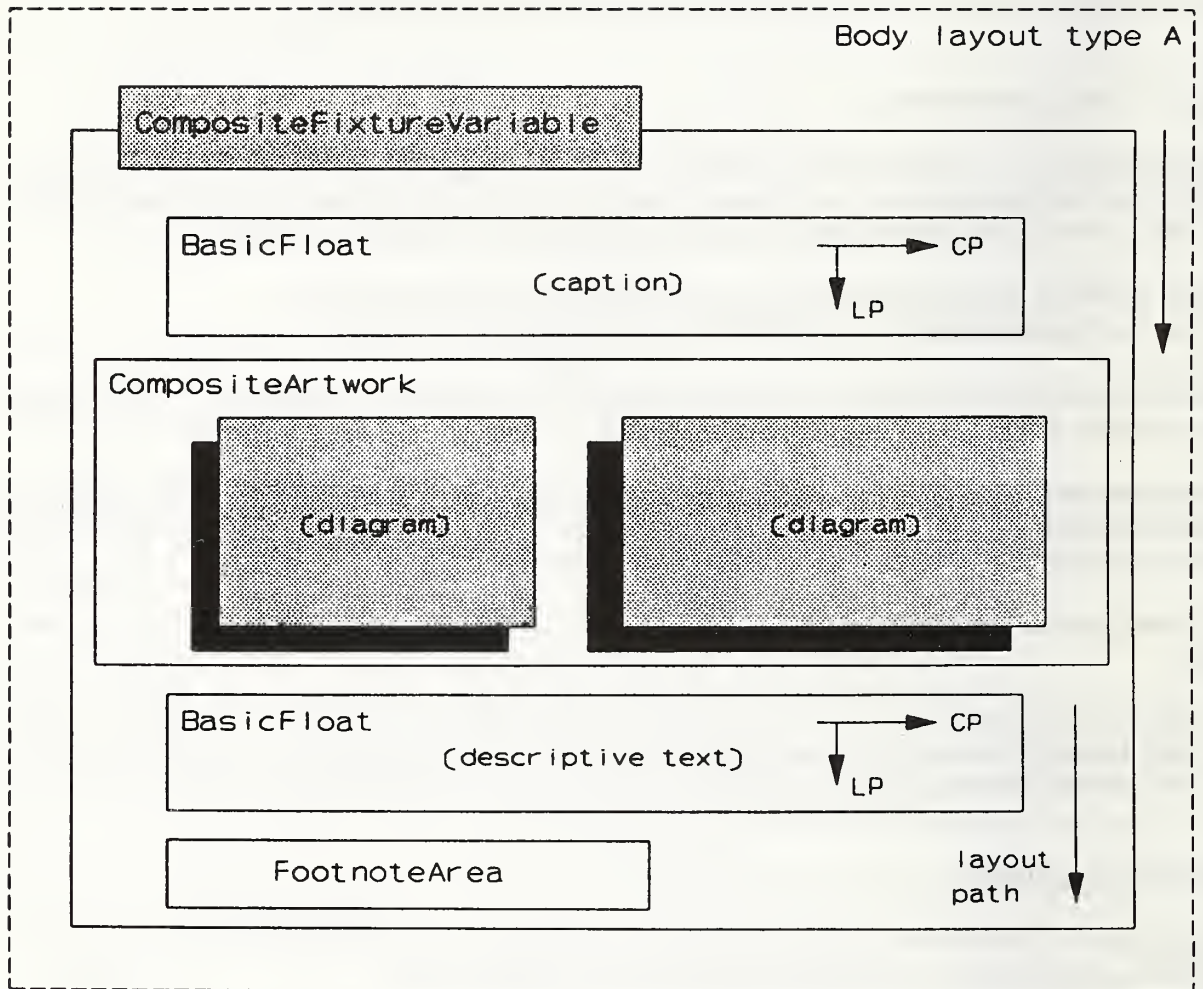
6.3.5.12 BasicFixture

BasicFixture is a constituent constraint that defines a lowest level frame which specifies an area for laying out content within FixedCompositeBody and CompositeArtwork.

This frame has a fixed position within its superior frame. The dimensions of this frame in the directions orthogonal and parallel to the layout path of the superior frame may be independently either fixed or specified as 'Rule-B'.

If the superior frame of a BasicFixture frame is FixedCompositeBody, the layout path of the BasicFixture frame is set to be the same as that of its superior frame.

If the superior frame of a BasicFixture frame is CompositeArtwork, the layout path of the BasicFixture frame is restricted as follows:



Key :

CP = character path

LP = line progression

Figure 12 — Example of a CompositeFixtureVariable frame

- if the layout path of the superior frame is set to 270 degrees, the layout path of BasicFixture may be set to 270 or 180 degrees;
- if the layout path of the superior frame is set to 0 degrees, the layout path of BasicFixture may be set to 0 or 270 degrees;
- if the layout path of the superior frame is set to 180 degrees, the layout path of BasicFixture may be set to 180 or 270 degrees.

6.3.5.13 FormArea

FormArea is a constituent constraint that defines a composite frame which represents a fixed dimensioned area reserved for laying out an illustration consisting of a form.

This area is divided into one or more 'simple' and 'composite' areas. A simple area is one in which content is directly laid out. A 'composite' area is an area which is further divided into one or more simple and composite areas.

This is a variably positioned frame that may be specified as a subordinate to a CompositeFixtureVariable or CompositeFixtureFixed frame.

The dimensions of this frame are fixed in both directions.

The immediate subordinates of a FormArea frame consists of an arbitrary ordered sequence of one or more fixed positioned frames of the following constituent constraints:

- FormEntryArea;
- EntryGroupArea;
- ArrangedContentFixed.

Frames of the type FormEntryArea and ArrangedContentFixed represent 'simple' areas in a form as described above. EntryGroupArea frames represent 'composite' areas.

The layout path of a FormArea frame defaults to 270 degrees and does not affect the layout of subordinate frames.

An example of the layout of a form is shown in Figure 13.

6.3.5.14 EntryGroupArea

EntryGroupArea is a constituent constraint that represents a composite area within a FormArea frame. The positions and dimensions of this frame are fixed in both directions. The layout path of this frame defaults to 270 degrees.

The immediate subordinates of this frame consist of an arbitrary ordered sequence of one or more fixed positioned frames of the type FormEntryArea.

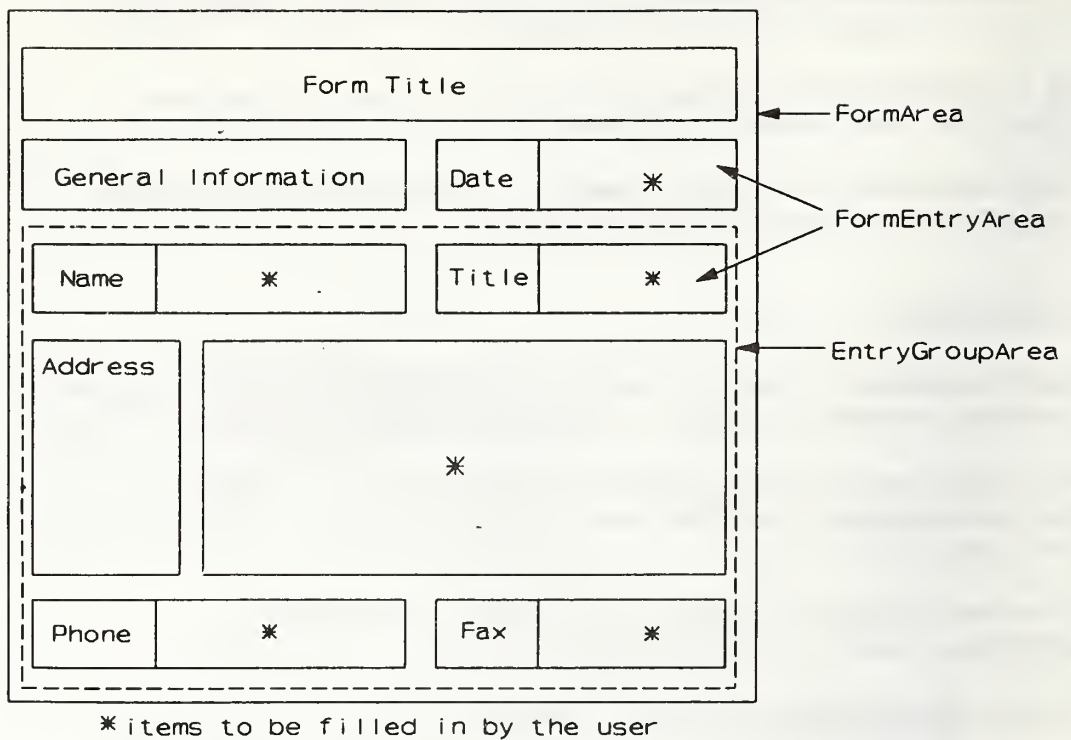


Figure 13 — Example of the layout of a form

6.3.5.15 FormEntryArea

FormEntryArea is a constituent constraint that defines a lowest level frame that specifies an area for laying out content within a FormArea frame. This frame may contain character, raster graphics or geometric graphics content.

The position and dimensions of this frame are fixed. Its layout path is 270 degrees regardless of the page layout type.

6.3.5.16 BasicColumn

BasicColumn is a constituent constraint that defines a lowest level frame that specifies an area for laying out content in the form of a column within a CompositeFloat frame. This frame has a variable position within its superior frame.

The dimension of this frame in the direction orthogonal to the layout path of its superior frame may be fixed or specified as 'Rule-B'. Its dimension in the direction parallel to the layout path of the superior frame may be fixed, specified as 'Rule-B' or defaulted to its maximum size. Thus the dimensions of this frame may be allowed to be automatically adjusted so that it contains all the content allocated to it.

The layout path specified for this frame is specified as 270 degrees in the case of body layout A, 0 degrees in the case of body layout B and 180 degrees in the case of body layout C.

6.3.5.17 ColumnVariable

ColumnVariable is a constituent constraint that defines a lowest level frame that is used to represent a column of content within a SnakingColumns frame. This is a frame which is variably positioned.

The dimension of this frame in the direction parallel to the layout path of the superior SnakingColumns frame (that is, the column width) is fixed. The dimensions of different instances of ColumnVariable frames within a given SnakingColumns frame may differ to allow columns of different widths to be specified.

The dimension in the direction orthogonal to the layout path of the superior frame (that is, the column length) may be specified as 'Rule-B' or 'maximum-size'.

The layout path for ColumnVariable frames is 270 degrees in the case of body layout A, 0 degrees in body layout B and 180 degrees in body layout C.

All ColumnVariable frames subordinate to the same SnakingColumns frame shall have the same category name; different names may be used for ColumnVariable frames laid out in different SnakingColumns frames.

6.3.5.18 CompositeColumnVariable

CompositeColumnVariable is a constituent constraint that defines a composite frame which specifies an area representing a column within a SnakingColumns frame. The column is subdivided into different areas to allow different layout requirements to be achieved. For example, this frame can be used to represent a column containing a table and a complex diagram which is embedded in a stream of character content. This is a frame which is variably positioned.

The dimension of this frame in the direction parallel to the layout path of the superior SnakingColumns frame (that is, the column width) is fixed. The dimensions of different instances of CompositeColumnVariable frames within a given SnakingColumns frame may differ to allow columns of different widths to be specified.

The dimension in the direction orthogonal to the layout path of the superior frame (that is, the column length) may be specified as 'Rule-B' or 'maximum-size'.

The immediate subordinates of this frame consist of an arbitrary ordered sequence of frames of the following constituent constraints;

- BasicFloat;
- CompositeFloat;
- CompositeFixtureVariable;
- TableArea;
- FootnoteArea;
- ArrangedContentVariable;
- SourcedContentVariable.

The layout path for CompositeColumnVariable frames is 270 degrees in the case of body layout A, 0 degrees in body layout B and 180 degrees in body layout C.

All subordinate frames are laid out in 'normal' positioning fill order except FootnoteArea which is laid out in 'reverse' fill order.

6.3.5.19 ColumnFixed

ColumnFixed is a constituent constraint that defines a lowest level frame that is used to represent a column of content within a FixedCompositeBody or SynchronizedColumns frame. This is a frame which has a fixed position.

The dimension of this frame in the direction orthogonal to the layout path of the superior frame (that is, the column width) may be fixed or specified as 'maximum size' (see note). This dimension may differ for different instances of ColumnFixed frames within a given SynchronizedColumns frame to allow columns of different widths to be specified. However, the widths shall be specified such that the columns do not overlap.

The dimension of this frame in the direction parallel to the layout path of the superior frame (that is, the column length) may be specified as 'Rule-B' or 'maximum-size' in the cases of body layouts A and B. In the case of body layout C, this dimension shall only be specified as 'maximum-size'.

The ColumnFixed frames subordinate to a given SynchronizedColumns frame must have different category names.

The layout path for ColumnFixed frames shall be equal to the layout path of the superior SynchronizedColumns frame.

The content laid out in different ColumnFixed frames within the same SynchronizedColumns frame may be specified as 'synchronized' by using the attribute "synchronization".

NOTE 17 - The value 'maximum size' shall only be specified for the last ColumnFixed frame laid out in a SynchronizedColumns frame to prevent overlapping of the frames. That is, for a page coordinate system with its reference point in the upper left corner, only the right most ColumnFixed frame shall specify 'maximum size' without the risk of overlapping frames.

6.3.5.20 CompositeColumnFixed

CompositeColumnFixed is a constituent constraint that defines a composite frame which specifies an area representing a column within a SynchronizedColumns frame. The column is subdivided into different areas to allow different layout requirements to be achieved.

The characteristics of this frame are the same as those of CompositeColumnVariable frames, except that this frame has a fixed position, and its dimensions are the same as for ColumnFixed frames.

6.3.5.21 FootnoteArea

FootnoteArea is a constituent constraint that defines a lowest level frame that is used to represent an area reserved for the layout of footnotes. Footnotes may be placed in body areas and also in columns and areas reserved for illustrations within body areas.

This frame may be specified as a subordinate to frames of the constituent constraints:

- VariableCompositeBody;
- CompositeColumnVariable;
- CompositeColumnFixed;
- CompositeFixtureVariable;
- CompositeFixtureFixed.

Frames of this type are variably positioned with a positioning fill order specified as 'reverse'. Therefore, this frame is positioned adjacent to the leading edge of its superior frame.

The dimension of FootnoteArea frames in the direction orthogonal to the layout path of its superior frame is fixed or specified as 'maximum size'. In the direction of the layout path, the dimension is specified by 'Rule-B' which means that this dimension is automatically adjusted to contain all the content that is allocated to it.

The layout path for FootnoteArea frames is the same as that specified for the superior frame.

The content that may be laid out in this frame is limited to the content that is associated with basic logical objects which are directly or indirectly subordinate to the composite logical object FootnoteBody. To achieve this, the "permitted categories" attribute of this frame shall specify the same category name required on the basic logical objects for footnotes (see 6.2.3.9.4 and 6.2.3.9.5).

6.3.5.22 CompositeCommon

CompositeCommon is a constituent constraint that defines a composite frame which specifies an area within the body area that is to contain common content. This area may be subdivided so that different types of common content may be laid out.

This is a frame which has fixed positions and dimensions. It may be specified as a subordinate to a FixedCompositeBody frame.

The subordinates of a CompositeCommon frame may consist of either:

- a) any number and combination of variably positioned frames of the types SourcedContentVariable and ArrangedContentVariable, or;
- b) any number and combination of fixed positioned frames of the types SourcedContentFixed and ArrangedContentFixed.

In case b), the subordinate frames may overlap without restriction.

The layout path of a CompositeCommon frame is 270, 0 and 180 degrees in the cases of body layouts A, B and C respectively.

6.3.5.23 Constituents used for laying out tables

This subclause defines the constituents used to support the layout of tables. An overview of the layout features pertaining to tables is given in 6.3.5.22.1 and the subsequent subclauses define the individual constituents provided.

6.3.5.23.1 Overview

A table consists of three main areas:

- a single optional header area which is placed at the top of the table;
- a single optional table label area which is placed immediately below the header area;
- one or more row areas, which are placed in sequence below the header area and table label area.

The table header area is typically used to contain a title or caption that describes the purpose of the table. It is an area which is subdivided into one or more areas, each of which contains common content derived from the logical structure using the 'logical source' mechanism.

The table label area is typically used to contain the labels which relate to the columns in the table. The row areas form the main body of the table and are used to layout the information that belongs to the table. These areas may be 'simple' or 'composite' as described below.

An example of a simple table is shown in figure 14. In this example, the table label and each row consist of a sequence of areas called 'cells' which are laid out horizontally across the table area. These are examples of 'simple' table label and row areas.

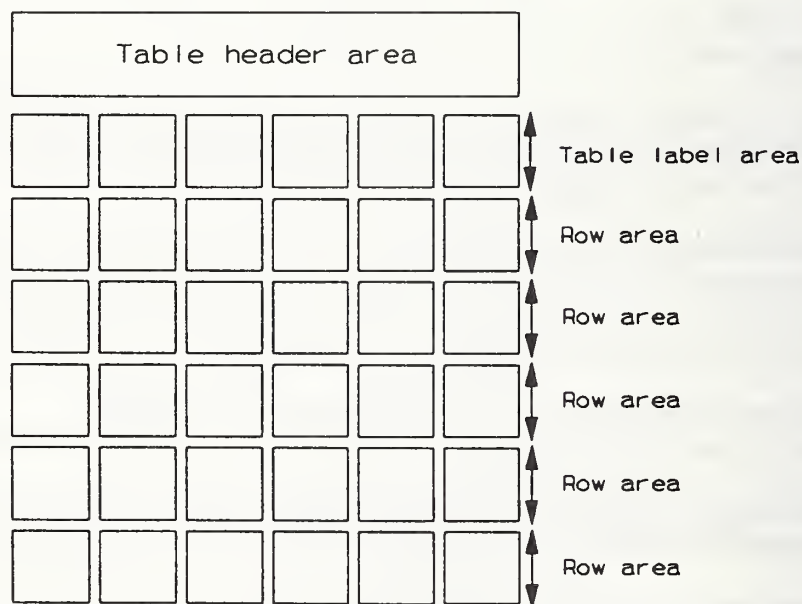


Figure 14 — Example of a *simple* table

An example of a more complex table is shown in figure 15 which illustrates the use of 'composite' table labels and row areas. A composite table label area consists of a sequence of two or more sub-rows, each of which is divided into separate cells. The cells in each row may or may not be the same size to allow a label to refer to a single column or several columns. The sequence of sub-rows may be preceded on the left by a single cell which, for example, can be used to contain a title which refers to the group of sub-rows.

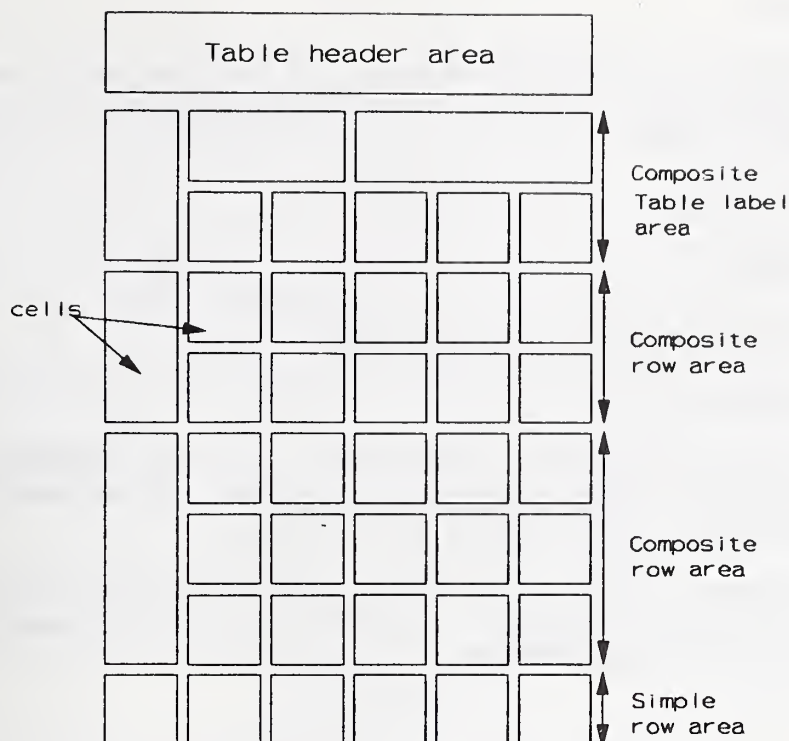


Figure 15 — Example of a *complex* table

Composite row areas have the same general structure as composite table label areas. By mixing together different combination of 'simple' and 'composite' table label and row areas, it is possible to obtain a wide range of different table types.

The frames that are used to represent table label and row areas are shown in figures 16 and 17 respectively. The areas labelled as 'cells' are intended to accommodate content belonging to a single content type. The frames which represent the 'cells' are fixed in position and have a fixed horizontal dimension. However, the vertical dimension of a cell may be specified as variable, so that this dimension is automatically adjusted during the layout process to accommodate all the content allocated to it.

In the case of a composite table label or row, the containing frame (that is, the LabelComponent or SubRow frame respectively) also has a dimension which is variable in the vertical direction. Thus the dimension of this frame may be automatically adjusted during the layout process so that it is large enough to accommodate the largest cell in that row.

Also, the containing frames (that is, the CompositeTableLabel, TableLabel, SubRowGroup and SubRow frames) may all be specified as having a variable vertical dimension, and, therefore, the vertical dimension of each table label and row area may be automatically adjusted to take into account all the information that is to be laid out in these areas.

The frames which specifies the complete table area (that is, the TableArea frame), which is not shown in the figures, also may have a variable vertical dimension.

The mechanism by which content is allocated to the 'cells' in a table is described in 6.4.1.3.8.

6.3.5.23.2 TableArea

TableArea is a constituent constraint that defines a composite frame that is used to specify an area reserved for the layout of a table. This constituent constraint may be specified as a subordinate to the following constituent constraints:

- VariableCompositeBody;
- CompositeFloat;
- CompositeColumn Variable;
- CompositeColumnFixed.

This is a frame that has a variable position. Its dimension in the direction orthogonal to the layout path of the superior frame is fixed. Its dimension in the direction parallel to the layout path is fixed or specified as 'Rule-B'. Its layout path is specified as 270 degrees.

The immediate subordinates of this constituent constraint consist of an optional TableHeader, followed by an optional TableLabel, which is followed by a sequence of one or more constituent constraints of the types RowArea and an optional TableLabel.

6.3.5.23.3 TableHeader

TableHeader is a constituent constraint that specifies a composite frame that specifies an area within a TableArea frame that is typically used to present the header information associated with a table.

This is a frame whose position is variable. Its dimension in the direction orthogonal to the layout path of the superior frame is fixed. Its dimension in the direction parallel to the layout path is fixed or specified as 'Rule-B'.

The immediate subordinates of this constituent constraint consist of a sequence of constituent constraints of the type SourcedContentFixed. Hence, the content laid out in a TableHeader frame is derived from logical constituent constraints of the type CommonContent.

6.3.5.23.4 TableLabel

TableLabel is a constituent constraint that defines a composite frame that specifies an area within a TableArea frame that is used for laying out labelling information relating to the columns of information in the table.

This is a frame whose position is variable. Its dimension in the direction orthogonal to the layout path of the superior frame is fixed. Its dimension in the direction parallel to the layout path is fixed or specified as 'Rule-B'.

The immediate subordinates of this constituent constraint consist of either:

- a) a sequence of one or more constituent constraints of the type TableLabelContent, or;
- b) a sequence of a constituent constraint of the type TableLabelContent, and a constituent constraint of the type CompositeTableLabel.

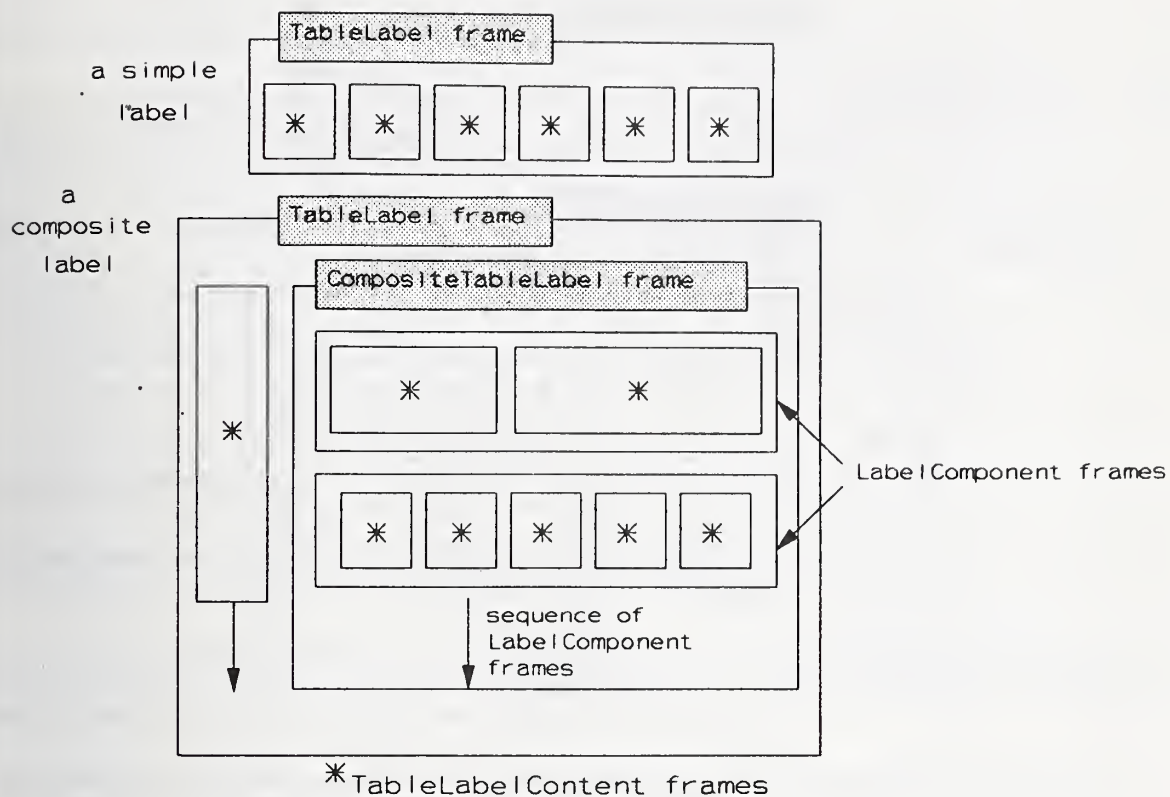


Figure 16 — Frames used to represent table labels

6.3.5.23.5 CompositeTableLabel

CompositeTableLabel is a constituent constraint that defines a composite frame that specifies an area with a TableLabel frame for laying out a composite table label.

This is a frame whose position is fixed. Its dimension in the direction orthogonal to the layout path of its superior frame is fixed. Its dimension in the direction parallel to the layout path is fixed, specified as 'Rule-B' or defaults to the maximum size allowed.

The immediate subordinates of this constituent constraint consist of a sequence of one or more constituent constraints of the type LabelComponent.

6.3.5.23.6 LabelComponent

LabelComponent is a constituent constraint that defines a composite frame that specifies an area within an CompositeTableLabel frame for laying out a row of labels within a table header.

This is a frame whose position is variable. Its dimension in the direction orthogonal to the layout path of the superior frame is fixed. Its dimension in the direction parallel to the layout path is fixed, specified as 'Rule-B' or defaults to the maximum size allowed.

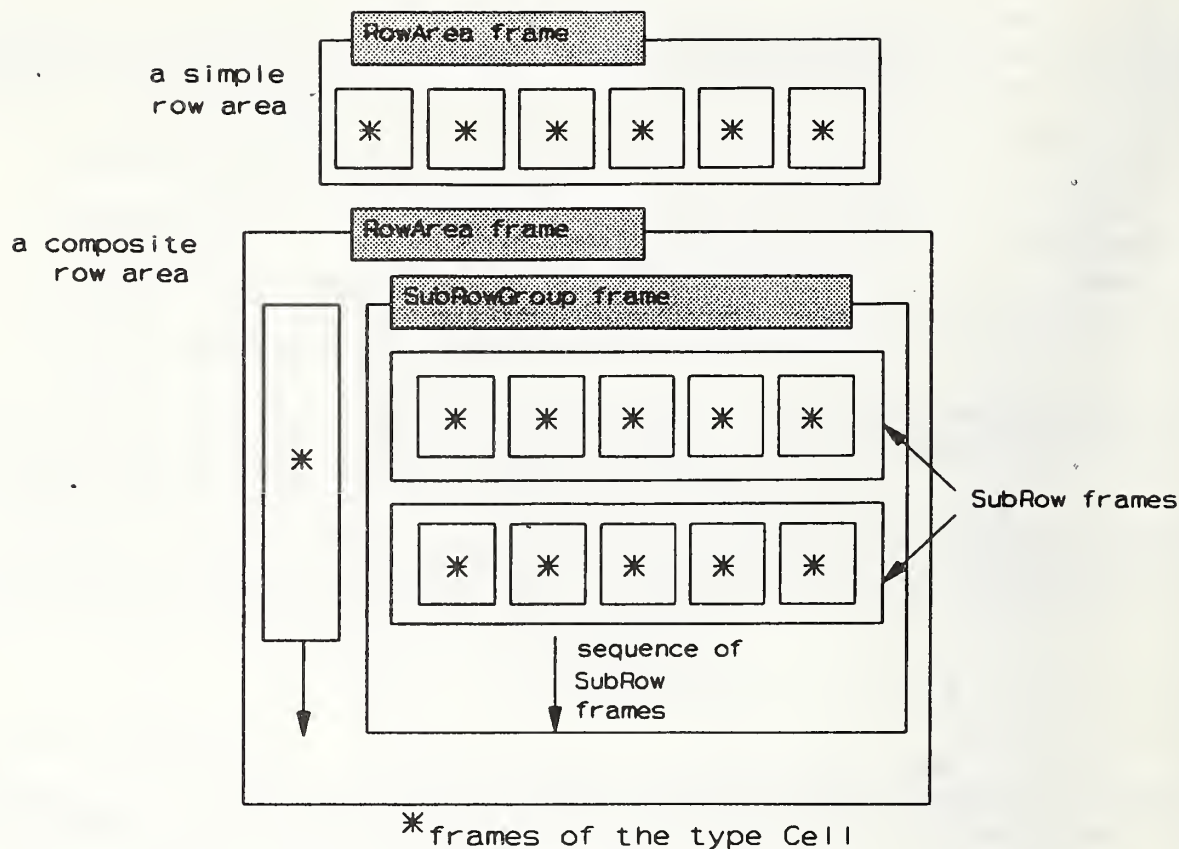


Figure 17 — Frames used to represent table rows

The immediate subordinates of this frame consist of a sequence of constituent constraints of the type `TableLabelContent`.

6.3.5.23.7 `TableLabelContent`

`TableLabelContent` is a constituent constraint that defines a lowest level frame that defines an area within a `TableLabel` or `LabelComponent` frame that is used for laying out header information that relates to one or more columns in a table. Character, raster graphics or geometric graphics content may be allocated to this frame.

This is a frame whose position is fixed. Its dimension in the direction orthogonal to the layout path of the superior frame is fixed. Its dimension in the direction parallel to the layout path is fixed or defaults to the maximum size allowed.

The content of a frame of this type is derived from a logical constituent constraint of the type `CommonContent`, using the logical source mechanism.

6.3.5.23.8 RowArea

RowArea is a constituent constraint that defines a composite frame that specifies an area within a TableArea frame used for laying out a row of entries in a table.

This is a frame whose position is variable. Its dimension in the direction orthogonal to the layout path of the superior frame is fixed. Its dimension in the direction parallel to the layout path is fixed or specified as 'Rule-B'.

The immediate subordinates of this constituent constraint consist of either:

- a) a sequence of one or more constituent constraints of the type Cell, or;
- b) a sequence of a single constituent constraint of the type Cell, and a constituent constraint of the type SubRowGroup.

6.3.5.23.9 SubRowGroup

SubRowGroup is a constituent constraint that defines a composite frame that specifies an area within a RowArea frame for laying out a composite row of entries in a table.

This is a frame whose position is fixed. Its dimension in the direction orthogonal to the layout path of its superior frame is fixed. Its dimension in the direction parallel to the direction of the layout path is fixed, specified as 'Rule-B' or defaults to the maximum size allowed.

The immediate subordinates of this constituent constraint consist of a sequence of one or more constituent constraints of the type SubRow.

6.3.5.23.10 SubRow

SubRow is a constituent constraint that defines a composite frame that specifies an area within a SubRowGroup frame for laying out a sub row of entries within a composite row in a table.

This is a frame whose position is variable. Its dimension in the direction orthogonal to the layout path of the superior frame is fixed. Its dimension in the direction parallel to the layout path is fixed, specified as 'Rule-B' or defaults to the maximum size allowed.

The immediate subordinates of this frame consist of a sequence of constituent constraints of the type Cell.

6.3.5.23.11 Cell

Cell is a constituent constraint that defines a lowest level frame that specifies an area within a RowArea or SubRow frame for laying out an entry in a table.

This is a frame whose position is fixed. Its dimension in the direction orthogonal to the layout path of the superior frame is fixed. Its dimension in the direction parallel to the layout path is fixed, specified as 'Rule-B' or defaults to the maximum size allowed.

The content of frames of this type is derived from logical constituent constraints of the type EntryElement.

6.3.6 Header and footer area characteristics

6.3.6.1 General characteristics

The header and footer areas may consist of either basic areas or composite areas.

A basic header or footer area is an area into which the content is directly laid out. This type of area is represented by a constituent constraint of the types BasicHeader or BasicFooter respectively.

A composite header or footer area is an area which is subdivided into separate sourced content and arranged content areas to provide greater versatility with regard to the layout of the content. This type of area is represented by a constituent constraint of the types CompositeHeader or CompositeFooter respectively.

In the case of basic header or footer areas, the content allocated to these areas is derived from the common part of the logical structure of a document. In the case of composite header or footer areas, the content may again be derived from the common part of the logical structure of a document, but the content may also be derived from common content specified in the generic layout structure.

6.3.6.2 BasicHeader and BasicFooter

BasicHeader and BasicFooter are constituent constraints that define lowest level frames that represent areas within a page that are reserved for common content.

These types of frame have fixed positions and dimensions. The positioning of these frames within a page and the layout paths that may be specified for them depends upon the H/F layout type used (see 6.3.4.5).

The content that is laid out in these frames is derived, using the logical source mechanism, from the content associated with the composite logical object classes of the type CommonContent.

6.3.6.3 CompositeHeader and CompositeFooter

CompositeHeader and CompositeFooter are constituent constraints that define composite frames that represent areas within a page that are reserved for common content.

These types of frame have fixed positions and dimensions. The positioning of these frames within a page and the layout paths that may be specified for them depends upon the H/F layout type used (see 6.3.4.5)

The subordinates of these frames may consist of either:

- a) any number and combination of variably positioned frames of the types SourcedContentVariable and ArrangedContentVariable, or;
- b) any number and combination of fixed positioned frames of the types SourcedContentFixed and ArrangedContentFixed.

In case b), the subordinate frames may overlap without restriction.

6.3.7 Sourced content and arranged content area characteristics

6.3.7.1 SourcedContentVariable

A SourcedContentVariable frame is a constituent constraint that defines a lowest level frame that contains common content derived from the generic logical structure. This frame is variably positioned and is used for the positioning content which is generated during the layout process, such as a character sequence containing a page number, a chapter title, etc.

This frame may be placed in the body area as well as the header or footer area:

- When this frame is in the header or footer area, it is the immediate subordinate of the frame of the constituent constraint type CompositeHeader or CompositeFooter.
- When this frame is in the body area, it is the immediate subordinate of the frame of the constituent constraint type VariableCompositeBody, CompositeColumnVariable, CompositeColumnFixed, CompositeCommon, SnakingColumns or CompositeFloat.

SnakingColumns are used to place common contents in the multi-column format. CompositeFloat is used to place common contents along side a figure or a form.

The attribute "logical source" must be specified for this frame to indicate the particular instance of the constituent constraint CommonContent which contains the content to be laid out.

When this frame is a subordinate of CompositeHeader or CompositeFooter;

a) the layout path of the frame is:

- 270 degrees for H/F layouts A1 and A2;
- 180 degrees for H/F layouts B1 and B2 (see 6.3.4.5 and the comment in 7.4.3.21).

b) the horizontal dimension of the frame is:

- either fixed or 'maximum-size' for H/F layouts A1, A2 and B2;
- either fixed or 'Rule-B' for H/F layout B1.

c) the vertical dimension of the frame is:

- either fixed or 'Rule-B' for H/F layouts A1 and A2;
- either fixed or 'maximum-size' for H/F layout B1;
- only fixed in the case of H/F layout B2.

When this frame is a subordinate of VariableCompositeBody, CompositeColumnVariable, CompositeColumnFixed or CompositeCommon:

- a) the layout path of the frame is 270, 0 or 180 degrees for body layouts A, B or C respectively (see 6.3.4.5 and the comment in 7.4.3.21);
- b) the dimension of the frame in the direction orthogonal to the layout path of the superior frame is either fixed or 'maximum-size';

- c) the dimension of the frame in the direction parallel to the layout path of the superior frame is either fixed or 'Rule-B'.

When this frame is a subordinate of SnakingColumns:

- a) the layout path of the frame is 270, 0 or 180 degrees for body layouts A, B or C respectively (see 6.3.4.5 and the comment in 7.4.3.21);
- b) the dimension of the frame in the direction orthogonal to the layout path of the superior frame is either 'Rule-B' or 'maximum-size', except that only 'maximum-size' is permitted for body layout C;
- c) the dimension of the frame in the direction parallel to the layout path of the superior frame is fixed.

When this frame is a subordinate of CompositeFloat:

- a) the layout path of the frame is 270, 0 or 180 degrees for body layouts A, B or C respectively (see 6.3.4.5 and the comment in 7.4.3.21);
- b) the dimension of the frame in the direction orthogonal to the layout path of the superior frame is either fixed, 'Rule-B' or 'maximum-size';
- c) the dimension of the frame in the direction parallel to the layout path of the superior frame is either fixed or 'Rule-B'.

6.3.7.2 ArrangedContentVariable

An ArrangedContentVariable frame is a constituent constraint that defines a lowest level frame that contains pre-defined common content contained in the generic layout structure. This frame is variably positioned, and its dimensions are fixed.

This frame references one or more blocks of type GenericBlock (see 6.3.8) which contain the content to be laid out in this frame. Thus, this frame is typically used when it is required to layout pre-determined common content.

6.3.7.3 SourcedContentFixed

A SourcedContentFixed frame is a constituent constraint that defines a lowest level frame that contains common content derived from the generic logical structure. This frame has a fixed position and dimensions.

This frame is required to specify the attribute "logical source" which indicates the particular instance of the constituent constraint CommonContent which contains the content to be laid out in this frame.

This frame may be placed in the body area as well as the header or footer area:

- When this frame is in the header or footer area, the frame is the immediate subordinate of the frame of the constituent constraint type CompositeHeader or CompositeFooter;

- When this frame is in the body area, the frame is the immediate subordinate of the frame of the constituent constraint type `FixedCompositeBody`, `CompositeCommon` or `SynchronizedColumns`.

When this frame is a subordinate of `CompositeHeader` or `CompositeFooter`:

- the layout path of this frame is 270 degrees for H/F layouts A1 and A2;
- the layout path of this frame is 180 degrees for H/F layouts B1 and B2 (see 6.3.4.5).

When this frame is a subordinate of `FixedCompositeBody`, `CompositeCommon` or `SynchronizedColumns`:

- the layout path of this frame is 270, 0 or 180 degrees for body layouts A, B or C respectively (see 6.3.4.5).

Thus, as in the case of `SourcedContentVariable` frames, this frame is used for the positioning of content which is generated during the layout process, such as a character sequence containing a page number.

6.3.7.4 `ArrangedContentFixed`

An `ArrangedContentFixed` frame is a constituent constraint that defines a lowest level frame that contains pre-defined common content derived from the generic layout structure. The position and dimensions of this frame are fixed.

This frame references one or blocks of type `GenericBlock` (see 6.3.8) which contain the content to be laid out in this frame. Thus this frame is typically used when it is required to layout common content at pre-determined positions in the header or footer areas.

6.3.8 `GenericBlock` and `SpecificBlock`

Two types of constituent constraints of the type 'block' are defined, namely `GenericBlock` and `SpecificBlock`.

Object classes of the type `GenericBlock` may occur in the generic layout structure referenced by the "generator for subordinates" of object classes of the types `ArrangedContentVariable` and `ArrangedContentFixed`. When the layout process is performed to produce a document in formatted processable form, equivalent blocks may occur in the specific layout structure.

Objects of the type `SpecificBlock` shall only occur in the specific layout structure. They are created during the document layout process and result from the layout of basic logical objects into lowest level frames that constitute the body, header and footer areas.

6.4 Document layout characteristics

Mechanisms for controlling the allocation of logical constituents to various areas in the layout structure are defined in 6.4.1. Mechanisms for controlling the layout of the content within the allocated areas are defined in 6.4.2.

These mechanisms relate to documents for which a generic layout structure is specified. When a generic layout structure is not present, then these mechanisms are restricted as described in 6.4.3.

6.4.1 Flow controls

Various mechanisms are provided to control the allocation of constituent constraints representing the 'body' parts of the logical structure of a document to page sets, pages and body areas. These are described in 6.4.1.1, 6.4.1.2 and 6.4.1.3. The mechanisms for controlling the layout of the 'common' parts of a document are described in 6.4.1.4.

6.4.1.1 Allocation of content to page sets

Two methods of allocating the constituent constraints associated with the 'body' part of the document to page sets are provided.

- a) layout in a nominated page set;
- b) starting a new page set.

The first method provides the ability to specify that a part of a document is to be laid out entirely within a specified page set. This may be specified for constituent constraints of the types Passage, NumberedSegment, Paragraph, Figure, NumberedList, UnNumberedList and DefinitionList using the attribute "layout object class" which specifies the object class identifier of the required class of page set.

The second method provides the ability to specify that the logical objects derived from a particular logical constituent constraint in a document and all subsequent parts of a document are to be laid out starting at the beginning of a new page set. This may be specified for logical object from the following logical constituent constraints:

- Passage;
- NumberedSegment;
- Number;
- Title;
- Paragraph;
- Figure;
- NumberedList;
- UnNumberedList;
- DefinitionList;
- BodyText;
- BodyRaster;
- BodyGeometric.

This is achieved using the attribute "new layout object" which specifies the object class identifier of the required class of page set.

6.4.1.2 Page breaks

This provides the ability to specify that the logical objects derived from a particular logical constituent constraint in a document and all subsequent parts of a document are to be laid out starting at the beginning of a new page. The page specified shall belong to the page set in which the logical objects from the immediate preceding logical constituent constraint is laid out (see note).

This may be specified for logical objects from the following logical constituent constraints:

- Passage;
- NumberedSegment;
- Number;
- Title;
- Paragraph;
- Figure;
- NumberedList;
- UnNumberedList;
- DefinitionList;
- BodyText;
- BodyRaster;
- BodyGeometric.

This is achieved using the attribute "new layout object". This attribute may specify the value 'page' indicating that the logical object is to be laid out starting on the next available page which may be of any class. Alternatively, the attribute may specify an object class identifier indicating the page class that the object is to be laid out in.

NOTE 18 - The specification of a page breaks shall not be used to layout part of a document in a new page set. If a new page set is required, then this shall be explicitly specified as described in 6.4.1.1.

6.4.1.3 Allocation of content to body areas

6.4.1.3.1 Introduction

If the page to which the content is allocated contains a basic body area, then the content is laid out in sequential order in that body area in the form of a single column.

If the page contains a composite body area, i.e., a `VariableCompositeBody` or `FixedCompositeBody` frame, then the content is allocated to subordinate areas in that body area as described below.

The general layout mechanism is described in 6.4.1.3.2. However, particular layout facilities are provided for the layout of logical constituent constraints of the type `Table` (see 6.4.1.3.8) and the type

Figure, which may contain either Artwork or Forms (see 6.4.1.3.7). Also, the layout of Footnotes is described in 6.4.1.3.10.

6.4.1.3.2 General mechanism for laying out content in a composite body area

When laying out content into a composite body area having more than one subordinate frame class (excluding FootnoteArea frame classes), it is necessary to indicate, directly or indirectly, which of the possible areas is to be used.

Basic logical objects other than those which are within a footnote structure may be specified to be laid out in instances of one or more lowest level frame class. This is done by giving each such basic logical component a value of the attribute "layout category" which corresponds to the value of the attribute "permitted categories" that applies to the lowest level frame in which the content is to be laid out.

Note that any basic logical objects in the specific logical structure to which this attribute does not apply will be laid out only in a lowest level frame which has the implicit value of the attribute "permitted categories".

The use of the attribute "layout category" ensures that if there is insufficient area on one page to lay out all of the content allocated to a particular type of area, the laying out of the content will automatically continue in the same type of area in a succeeding page when possible. Thus content is allowed to flow freely from one page to another when the type of layout used at the end of one page is the same as that at the beginning of the succeeding page. When continuation to the same type of area in a succeeding page is not possible because of conflict with other layout directives or because the "generator for subordinates" of the page class does not allow such choice, backtracking may occur or other type of area may be selected.

It is necessary to ensure the correct use of the mechanism for the layout of independent layout streams. In the absence of additional layout directives, content may be placed in available space within an earlier frame of the specified "permitted categories". If this is not intended, it may be prevented by the use of the attribute "new layout object" (or "layout object class" in some cases).

The attribute "new layout object" may be applied to logical components whenever a change in layout is required. The attribute "new layout object" may specify the identifier or the category name corresponding to the frame class that is required.

When layout occurs in a snaking columns area, column breaks may be indicated by using the attribute "new layout object". This attribute may specify the identifier or the category name of the frame corresponding to the column in which the layout is to continue. However, only the use of category name will ensure that a single column break is always obtained, irrespective of the frame class actually used.

When the layout is to occur in a synchronized columns area, category names may be used to control the particular columns that are to be used to lay out the logical entities. Each column within a synchronized columns area shall have a different permitted category and each basic logical object to be laid out in this particular area shall have a category name corresponding to a name allocated to one of the columns. The logical entities allocated to different columns may be aligned using the attribute "Synchronization".

The following subclauses describe the layout mechanism applicable to subordinate areas for each of the frame types listed above.

6.4.1.3.3 Layout into BasicFloat and BasicFixture frames

These are lowest level frames and hence content continues to be directly laid out in these frame types until an occurrence of the attribute "new layout object" causes the layout to proceed starting with an alternative frame class.

6.4.1.3.4 Layout in SnakingColumns frames

A SnakingColumns frame is a composite frame which contains columns represented by lowest level or composite frames.

In the case of lowest frames, all the frames may have the same category name so that content can flow from one frame to the next. That is, a column break will occur naturally when the size of one column reaches the limit imposed by the superior frame and the layout process will continue automatically in the next column.

In the case of composite columns, the subordinate areas are represented by subordinate frames of the types BasicFloat, CompositeFloat, CompositeFixtureVariable, TableArea and FootnoteArea. The frame type into which the constituents are to be laid out is selected using the attribute "new layout object" which indicates the identifier or category name of the required subordinate frame, or is automatically selected according to "layout categories". Logical constituents will continue to be laid out in the selected frame type until a different frame type is selected. Also, the layout process continues automatically from one column to the next, but column breaks can again be forced as described in 6.4.1.3.2.

6.4.1.3.5 Layout in SynchronizedColumns frames

A SynchronizedColumns frame is a composite frame which contains columns represented by subordinate lowest level or composite frames.

In the case of lowest frames, all the frames are required to have different categories. Hence, the layout of logical objects from the constituents into different columns is controlled by the category name specified for each constituent. The attribute "new layout object" may also be used for this purpose, but this is not necessary.

In the case of composite columns, the subordinate areas are represented by subordinate frames of the types BasicFloat, CompositeFloat, CompositeFixtureVariable, TableArea and FootnoteArea.

The selection of a particular composite column or a particular sub-area within a composite column can be achieved using the attribute "new layout object" which specifies the identifier or category name of the particular frame class required.

6.4.1.3.6 Layout in CompositeFloat frames

This is a composite frame which contains two or more subordinate frames that are laid out 'side-by-side'. The appropriate subordinate frame is chosen according to the category names or chosen using the attribute "new layout object" which specifies the appropriate identifier or category name of the required subordinate frame class.

6.4.1.3.7 Layout of Figures

Frames of the types `CompositeFixtureVariable` and `CompositeFixtureFixed` are provided specifically for the layout of logical constituent constraints of the type `Figure`. Similarly, frames of the type `CompositeArtwork` and `FormArea` are provided for the layout of logical constituent constraints of the types `Artwork` and `Form` respectively.

The schematic diagram in table 2 shows how the logical constituent constraint `Figure` and its subordinates are allocated to the frame `CompositeFixtureVariable` or `CompositeFixtureFixed` and their subordinates.

Table 2 — Layout of Figure

Logical constituent constraint	Frame class
Figure>	<code>CompositeFixtureVariable</code> or <code>CompositeFixtureFixed</code>
Artwork>	<code>CompositeArtwork</code>
Phrase>	<code>BasicFixture</code>
BodyRaster>	<code>BasicFixture</code>
BodyGeometric >	<code>BasicFixture</code>
Form>	<code>FormArea</code>
EntryGroup>	<code>EntryGroupArea</code>
EntryElement>	<code>FormEntryArea</code>
Number.....>	<code>BasicFloat</code>
Caption.....>	<code>BasicFloat</code>
Description>	<code>BasicFloat</code>
Footnote>	<code>FootnoteArea</code>

Table 2 indicates a mapping between logical constituent constraints and frames, and their respective subordinates. Also, the diagram indicates that this mapping is hierarchical.

For example, `Figure` is to be laid out into a single instance of the frame `CompositeFixtureVariable` or `CompositeFixtureFixed`. A subordinate constituent constraint of the type `Artwork` is to be laid out in a single instance of a frame of type `CompositeArtwork` within the specified `CompositeFixtureVariable` in a `CompositeFixtureFixed` frame.

Also, each instance of a subordinate `Phrase`, `BodyRaster` or `BodyGeometric` is to be laid out in a single instance of a subordinate frame of the type `BasicFixture`. `BasicFixture` frames may overlap to form a composite image.

Similarly, a `Form` is to be laid out in a single instance of a `FormArea` frame. Frames subordinate to this `FormArea`, that is, `EntryGroupArea` and `FormEntryArea` frames, will each receive single instances of logical constituent constraints of the type `EntryGroup` and `EntryElement` respectively.

This layout mechanism is achieved by specifying the attribute "new layout object" for Figure with a value indicating the identifier of the appropriate frame classes in which that constituent is to be laid out.

The constituent constraints Number, Caption and Description (and their subordinates in the case of Caption and Description) are laid out in frames of the type BasicFloat. This is achieved automatically according to category names or explicitly using the attribute "new layout object" which indicates the identifier or category name of the frame class required. More than one instance of the constituent constraints Caption, Description and Number may be laid out in a particular BasicFloat frame.

Frames of the type FootnoteArea may be generated within a CompositeFixtureVariable or CompositeFixtureFixed to accommodate instances of the logical constituent constraint Footnote which occurs as a subordinate of Phrase, Caption or Description.

6.4.1.3.8 Layout of Tables

Frames of the type TableArea are provided specifically for the layout of logical constituent constraints of the type Table.

Table 3 illustrates the relationships between the logical constituent constraint Table and its subordinates, and the frames used to lay out these constituent constraints.

Table 3 — Layout of Tables

Logical constituent constraint	Frame class
Table>	TableArea
Row>	RowArea
EntryElement>	Cell
TableComponent>	SubRowGroup
RowComponent>	SubRow
EntryElement>	Cell
CommonContent <.....	TableHeader
CommonContent <.....	TableLabel

Table 3 indicates that there is a hierarchical mapping between logical constituent constraints and their corresponding frames. For example, each Row is to be laid out in a separate frame of the type RowArea. Each TableComponent that is subordinate to that Row must be laid out in a specific SubRowGroup that is subordinate to the RowArea indicated.

The layout mechanism is achieved for the logical constituent constraints Table, Row and RowComponent by specifying the attribute "new layout object" with a value indicating the identifier of the required frame class of the type TableArea, RowArea and SubRow respectively.

For the logical constituent constraint EntryElement, the layout mechanism is achieved by one of the following:

- a) If the attribute "generator for subordinates" of the superior (RowArea or SubRow) of the affected EntryElement is constructed using SEquence, the attribute "new layout object" is used to specify a Cell into which the contents is laid out. The attribute value specifies the identifier of the required frame class in the EntryElement.
- b) If the attribute "generator for subordinates" of the superior (RowArea or SubRow) of the affected EntryElement is constructed by REPetition, the value of the attribute "new layout object" indicates a category name to be used for the EntryElement. In this case, the category name shall be specified in the attribute "layout category" for EntryText, EntryRaster or EntryGeometric, and in the attribute "permitted categories" for the Cell into which the contents is laid out.

In the case of TableComponent, the attribute "layout object class" is used to specify that this logical constituent constraint is to be laid out in a SubRowGroup frame.

This mechanism allows a table to be laid out such that it is split over two or more successive frames or pages. A split may occur at the boundary of a RowArea frame, or such that a RowArea frame is split over two successive frames or pages. A split cannot occur within a SubRowGroup frame.

When such a split does occur, then the TableHeader and TableLabel frames are repeated at the top of each frame of page in which the table is continued.

The content allocated to the frames TableHeader and TableLabel are derived from logical constituent constraints of the type CommonContent in the generic logical structure using the 'logical source' mechanism.

6.4.1.3.9 Layout of Forms

Frames of the type FormArea are provided specifically for the layout of logical constituent constraints of the type Form.

Table 4 illustrates the relationships between the logical constituent constraint Form and its subordinates, and the frames used to lay out these constituent constraints.

Table 4 indicates there is a hierarchical mapping between logical constituent constraints and the corresponding frames, and their respective subordinates.

The layout mechanism is achieved by specifying for logical constituent constraints the attribute "layout object class" which indicates the object class identifier of an appropriate frame in accordance with the above diagram. This mechanism does not allow frames of the type FormArea to be split over two or more superior frames.

The content associated with the logical constituent constraint EntryElement is specified by one of the constituent constraints EntryText, EntryRaster or EntryGeometric. The layout of this content is controlled by the layout directives "offset" and "block alignment".

Table 4 — Layout of Forms

Logical constituent constraint	Frame class
Form>	FormArea
EntryGroup>	EntryGroupArea
EntryElement>	FormEntryArea

6.4.1.3.10 Layout of Footnotes

The logical objects derived from basic logical constituent constraints that represent the content belonging to a footnote (i.e., FootnoteReference, FootnoteNumber and FootnoteText) are constrained to be laid out in a footnote area which is represented by a FootnoteArea frame (see 6.3.5.20).

This constraint is specified by means of category names. That is, the logical constituent constraints of the types FootnoteNumber and FootnoteText, and layout constituent constraints of the type FootnoteArea are all required to have the category name 'Footnote' or 'Footnote <n>'.

More than one footnote may be placed in a footnote area. In this case, the content belonging to the footnotes are laid out sequentially in the footnote area in accordance with their reading order.

If the content belonging to a footnote cannot all be accommodated in the footnote area on one page, then the content may freely flow into the next footnote area. Alternatively, it is possible to specify that a footnote is to be laid out entirely within a particular footnote area. This is achieved using the attribute "indivisibility".

6.4.1.4 Allocation of content to header and footer areas

A header or footer area may be basic or composite (see 6.3.6.1). In the case of a basic area, the frame representing that area specifies the attribute "logical source" which indicates the particular instance of the constituent constraint of the type CommonContent that is to be laid out in that area. The basic logical constituent constraints subordinate to CommonContent are then laid out in accordance with their sequential order.

In the case of a composite header or footer area (see 6.3.6.3), the area is divided into one or more separate areas, each of which is represented by a lowest level frame. The content allocated to the separate areas may be derived from one of two sources. That is, the content may be pre-defined and represented by one or more blocks which are directly associated with the lowest level frame. Alternatively, the lowest level frame may specify the attribute "logical source" which, as above, indicates the particular logical object of the type CommonContent that is to be laid out in that frame.

6.4.2 Layout of document content

Various constraints may be specified to control the layout of the content into the body, header and footer areas. These constraints are described below.

6.4.2.1 Margins

The margins are the minimum distances, or offsets, between a part of the document content and the edge of the particular area in which that content is laid out. The margins define the maximum extents of the available area into which the content shall be positioned.

Margins may be specified for any constituent constraint representing a basic logical object; different margin values may be specified for different constituent constraints without restriction.

Four margins may be independently specified for each constituent constraint, namely:

- trailing edge margin;
- leading edge margin;
- right hand edge margin;
- left hand edge margin.

These margins are defined in relationship to the layout path specified for the frame into which the content is to be laid out (see figure 18).

Any combination of the above margins may be specified for a particular constituent constraint. These margins are specified by the attribute "offset". Any value may be specified in units of SMUs. If a particular margin is not specified, then it is assumed to be 0 SMUs.

6.4.2.2 Separation

Leading separation is the minimum distance between one basic logical object and the next one, if any, when they are laid out. Trailing separation is the minimum distance between one basic logical object and the previous one, if any, when they are laid out. Both may be specified for basic logical components of any constituent constraint types. These distances are specified in SMUs by the attribute "separation". If no value is specified, then the minimum distance is assumed to be 0 SMUs.

6.4.2.3 Indivisibility

Indivisibility provides the means to specify whether or not a logical object derived from a basic or composite logical constituent constraint is allowed to be split over more than one page or over more than one area within a page. It may be specified for logical objects from constituent constraints of the types Passage, NumberedSegment, Number, Title, Paragraph, Caption, Phrase, Reference, Description, ListItem, ListTerm, FootnoteText, ReferencedContent, FootnoteBody, Artwork, EntryElement, Row, RowComponent, Footnote, Figure, Table, UnNumberedList, NumberedList, DefinitionList, FootnoteReference and BodyText. The attribute "indivisibility" is used to specify this feature.

6.4.2.4 Same layout object

Same layout object provides the means to specify that the start of the content associated with a logical object and the end of the content associated with the previous logical object are to be laid out within a single layout object. This may be specified for logical objects of the types Passage, NumberedSegment, Title, Caption, Number, Paragraph, Phrase, Footnote, FootnoteBody, Figure, FootnoteReference, ReferencedContent, Reference, Description, Table, NumberedList,

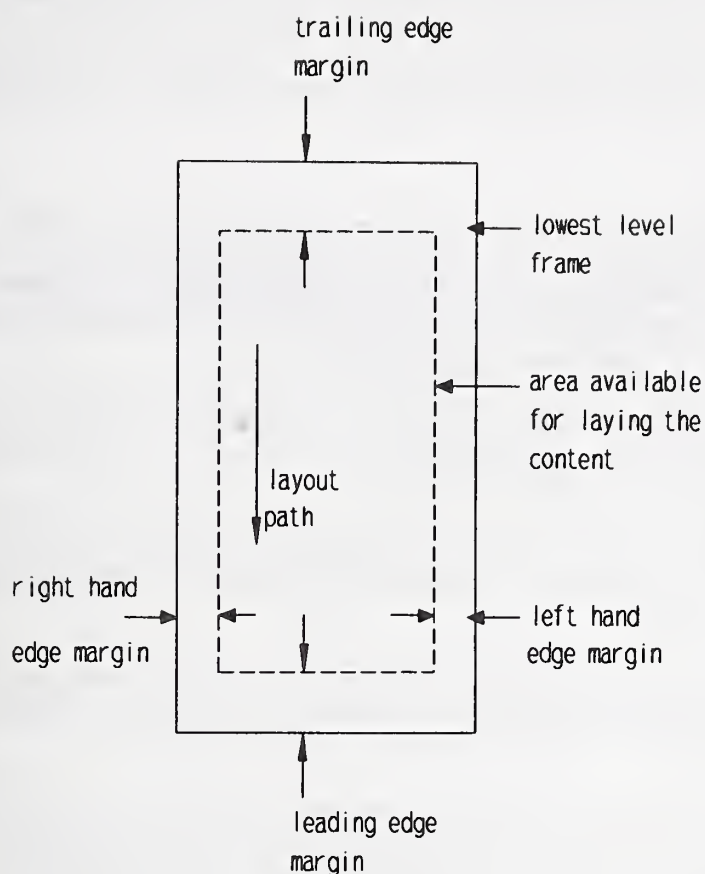


Figure 18 — Specification of margins

UnNumberedList, DefinitionList, ListItem, ListTerm, BodyText, BodyRaster and BodyGeometric. The attribute "same layout object" is used to specify this feature.

6.4.2.5 Concatenation

Concatenation provides the means to specify that the content associated with a logical object derived from a basic logical constituent constraint and the content associated with the previous basic logical object are to be regarded as an unbroken stream of content. This may be specified for logical objects from constituent constraints of the types BodyText, Number, ReferencedContent, FootnoteReference, FootnoteText, TableNumber, CurrentInstance, CommonNumber, CommonReference, CommonText and PageNumber. The attribute "concatenation" is used to specify this feature.

6.4.2.6 Block alignment

Block alignment allows the content associated with a basic logical object to be specified as 'left aligned', 'right aligned' or 'centred' within the area in which that content is laid out. Left aligned means that the content is laid out adjacent to the left hand edge margin. Right aligned means that the content is laid out adjacent to the right hand edge margin, and centred means that the content is laid out midway between the left and right margins.

This feature may only be specified using the attribute "block alignment" for logical objects from constituent constraints of the types BodyText, EntryText, Number, CommonNumber, PageNumber, TableNumber, FootnoteNumber, FootnoteText, FootnoteReference, CommonReference, ReferencedContent, CurrentInstance, and CommonText, and when they contain formatted character content, BodyRaster, and BodyGeometric, EntryRaster, EntryGeometric, CommonRaster and CommonGeometric.

6.4.3 Layout controls applicable in the absence of a generic layout structure

In processable form documents the generic layout structure is optional. If the generic layout structure is omitted, then it is the responsibility of the receiver to define an appropriate layout structure. No limitations are placed on the layout structure that is used.

When a generic layout structure is not specified within a processable form document, then restrictions are placed on the layout control functions described in 6.4.1 and 6.4.2 that may be specified within the document. These restrictions are indicated as follows:

- a) It is not possible to specify that certain logical parts of a document are to be allocated to a given page set or that a part of a document is to be laid out starting in a new page set, as defined in 6.4.1.1;
- b) It is possible to specify page breaks as defined in 6.4.1.2, but it is only possible to indicate that the layout shall begin on a new page. It is not possible to specify a particular page class;
- c) The logical parts of the document that are intended to be laid out in the body area and in the header/footer areas of each page may be distinguished from each other by means of application comments specified for them (see 6.6.5). An exception is that it is not possible to distinguish whether a particular portion of common content is to be placed in a header or a footer area (or both);
- d) It is not possible to indicate the type of layout area to be used to layout each logical constituent in the body part of a document. That is, it is not possible to indicate whether single column or multiple column areas are to be used (see 6.4.1.3). This shall be decided by the receiver;
- e) Footnotes within the body part of a document may be distinguished by use of the attribute "application comments". Footnotes are intended to be read and laid out separately from the other logical constituents of the body part (see 6.4.1.3). However, it is the responsibility of the receiver to decide how footnotes are laid out;
- f) Margins, separation, indivisibility, same layout object, concatenation and block alignment, as defined in 6.4.2, may all be specified. Only one restriction applies. Indivisibility (see 6.4.2.3) may be assumed to specify that a logical constituent constraint is not to split over more than one page, but indivisibility shall not be specified for other types of layout areas, such as single or multiple column areas.

6.5 Content layout and imaging characteristics

A document may contain character, raster graphics and geometric graphics content.

The content architectures that may be specified using the attribute "content architecture class" are formatted character, processable character, formatted processable character, formatted processable

raster graphics and formatted processable geometric graphics. Any of these may be specified as the default in the document profile.

6.5.1 Character content

6.5.1.1 Introduction

This subclause defines the features that are applicable to the character content contained in a document and the presentation attributes and control functions that may be used to specify these features. These features may apply to basic logical and layout components unless otherwise indicated.

The default values for the following features may be specified in the document profile:

- graphic character sets;
- graphic character subrepertoire;
- code extension announcers;
- line spacing;
- character spacing;
- character path;
- line progression;
- character orientation;
- graphic rendition, including the parameter values: default rendition, increased intensity (bold), italicized, underlined, crossed-out, slowly blinking, rapidly blinking, negative image, positive image, primary font, 1st alternative font, 2nd alternative font, 3rd alternative font, 4th alternative font, 5th alternative font, 6th alternative font, 7th alternative font, 8th alternative font, 9th alternative font, doubly underlined, normal intensity, not italicized, decreased intensity, not underlined, not blinking, not crossed-out;
- line layout table;
- indentation;
- alignment;
- first line offset;
- itemization;
- widow size;
- orphan size;
- character fonts;

- kerning offset;
- pairwise kerning;
- proportional line spacing;
- formatting indicator;
- initial offset.

The specification in a document of a non-basic feature by a presentation attribute or control function shall be indicated in the document profile.

6.5.1.2 Character content architecture

Processable and formatted processable form documents may contain processable, formatted or formatted processable character content. Formatted form documents may contain formatted and formatted processable character content.

When using character content, any number of content portions may be associated with a basic component.

The content information in a content portion may be absent. This is to allow the representation and interchange of documents in which parts of the content may be supplied, for example, during subsequent editing.

6.5.1.3 Character repertoire

The basic character repertoire supported by this profile is composed of the 94 characters of ISO-IR6(the IRV of ISO 646 revised 1991) plus the character space.

Any other graphic character set which is registered in accordance with ISO 2375 may be designated and invoked at any point in the document provided its use is indicated in the document profile as a non-basic value using the character presentation feature "graphic character sets". No locking shift functions are specified in this presentation feature.

The code extension techniques allowed for the designation and invocation of character sets to the left hand side and right hand side of the 8-bit code table (GL and GR respectively) are defined in 6.5.1.4.

Using these code extension techniques, the graphic character sets designated and/or invoked at the beginning of a content portion containing character content are specified using the presentation attribute "graphic character sets". The character sets may also be changed at any point within a content portion.

The default graphic character sets which apply to the content portions within a document may be specified in the document profile using the presentation attribute "graphic character sets".

If the character set defined in ISO 6937-2 with or without Addendum 1 is designated and invoked, then the use of any of its subrepertoires registered according to ISO 7350 may be specified using the presentation attribute "graphic character subrepertoire". All subrepertoires are non-basic and their use shall be indicated in the document profile. The subrepertoire shall not be changed within a content portion.

NOTE 19 - The basic character repertoire supported by this profile is not the standard default value specified in

[CCITT Recommendation T.416 | ISO 8613-6]; hence it may be necessary to specify in the document profile of a particular document that this is the default value being used for that document.

NOTE 20 - Revised CCITT Recommendations T.50 and T.51 and new CCITT Recommendation T.52 are under preparation. CCITT Recommendations T.50 and T.51 are intended to be completely compatible with ISO 646 revised 1991 (ISO IR-6) and ISO 6937 (under revision) respectively.

6.5.1.4 Code extension techniques

The code extension techniques specified in ISO 2022 may be used subject to the following restrictions:

- a) G0 set: only ISO-IR6 (the IRV of ISO 646 revised 1991), ISO-IR2 (the primary set of ISO 6937-2), or any other version of ISO 646 (revised 1991) may be designated as the G0 set; these graphic character sets may only be invoked in GL;
- b) G1, G2, G3 sets: no restrictions are placed on the character sets that may be designated for these sets; these graphic character sets may only be invoked in GR;
- c) The locking and single shift functions allowed are as follows:
 - 1) LS0, to invoke the G0 set into GL;
 - 2) LS1R, to invoke the G1 set into GR;
 - 3) LS2R, to invoke the G2 set into GR;
 - 4) LS3R, to invoke the G3 set into GR;
 - 5) SS2, to invoke one character from the G2 set into GL;
 - 6) SS3, to invoke one character from the G3 set into GL;

NOTE 21 - GL and GR refer to the left and right hand parts respectively of the 8-bit code table.

- d) When specifying the presentation attribute "graphic character sets", it is necessary to invoke character sets for both GL and GR. Thus an allowed character set shall be designated into G0 (see item (a) above) and invoked into GL. It is also necessary to invoke a graphic character set into GR which has been designated into the G1, G2 or G3 set;
- e) The empty set shall be designated into G1 and invoked into GR if no other specific graphic character set is invoked into GR;

The code extension techniques allowed are illustrated in figures 19 and 20.

The announcement and encoding of these functions are to be as specified in ISO 2022.

The code extension techniques that are used or may be used in a basic component shall be specified by the presentation attribute "code extension announcers." The default code extension announcers used throughout a document may be specified in the document profile, also using the presentation attribute "code extension announcers".

NOTE 22 - In accordance with [CCITT Recommendation T.416 | ISO 8613-6], there is no restriction concerning

the number of graphic character sets which may be designated and/or invoked in the presentation attribute "graphic character sets" providing the restrictions defined in this subclause are applied. Hence designation to a particular G set overrides a previous designation to that set and invocation to GL or GR overrides the previous invocation to the GL or GR respectively. Thus the sequential order of designation and/or invocation sequences in the attribute "graphic character sets" is significant.

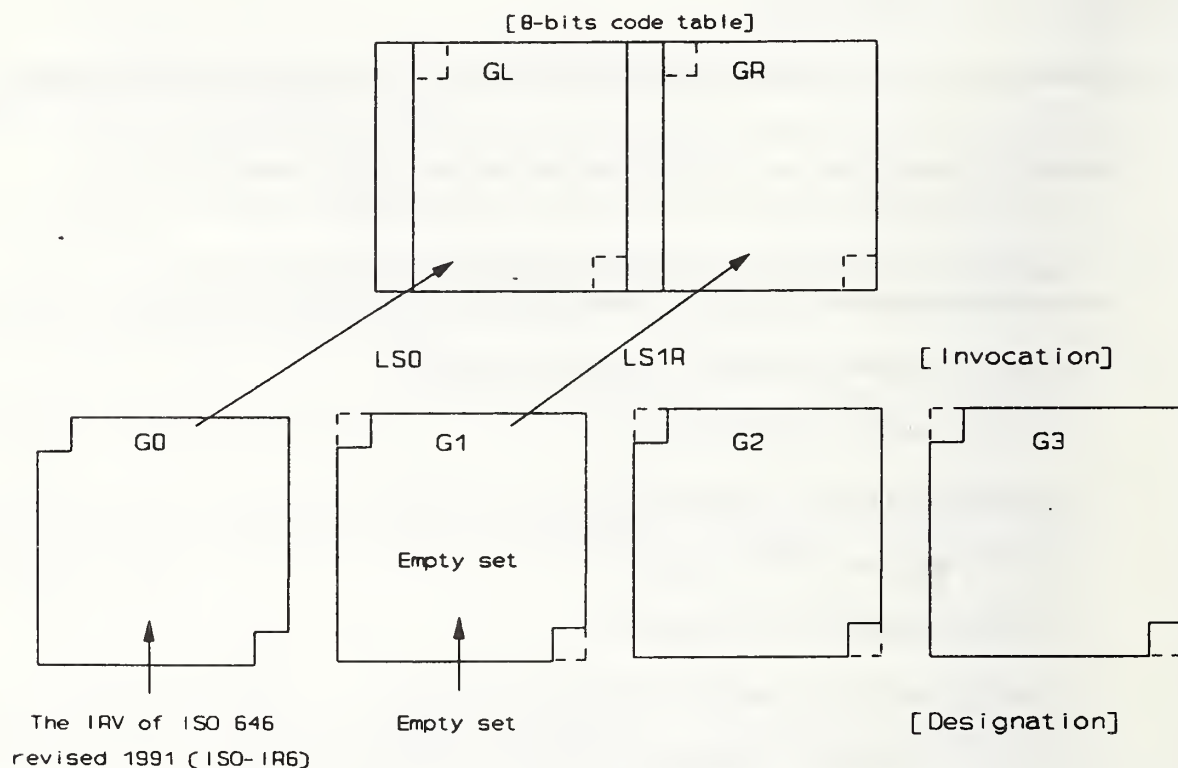


Figure 19 — Code extension features (basic case)

6.5.1.5 Line spacing

Any value of line spacing may be specified. Values of 100, 150, 200, 300, 400 and 600 BMUs are basic; the use of any other value in a document is non-basic and shall be indicated in the document profile.

The line spacing may be specified at the beginning of the content associated with a basic component using the presentation attribute "line spacing". The value may be changed anywhere within the content portion using the control functions select line spacing (SVS) and set line spacing (SLS).

6.5.1.6 Character spacing

Any value of character spacing may be specified as basic values.

The character spacing may be specified at the beginning of the content associated with a basic component using the attribute "character spacing". The value may be changed anywhere within a

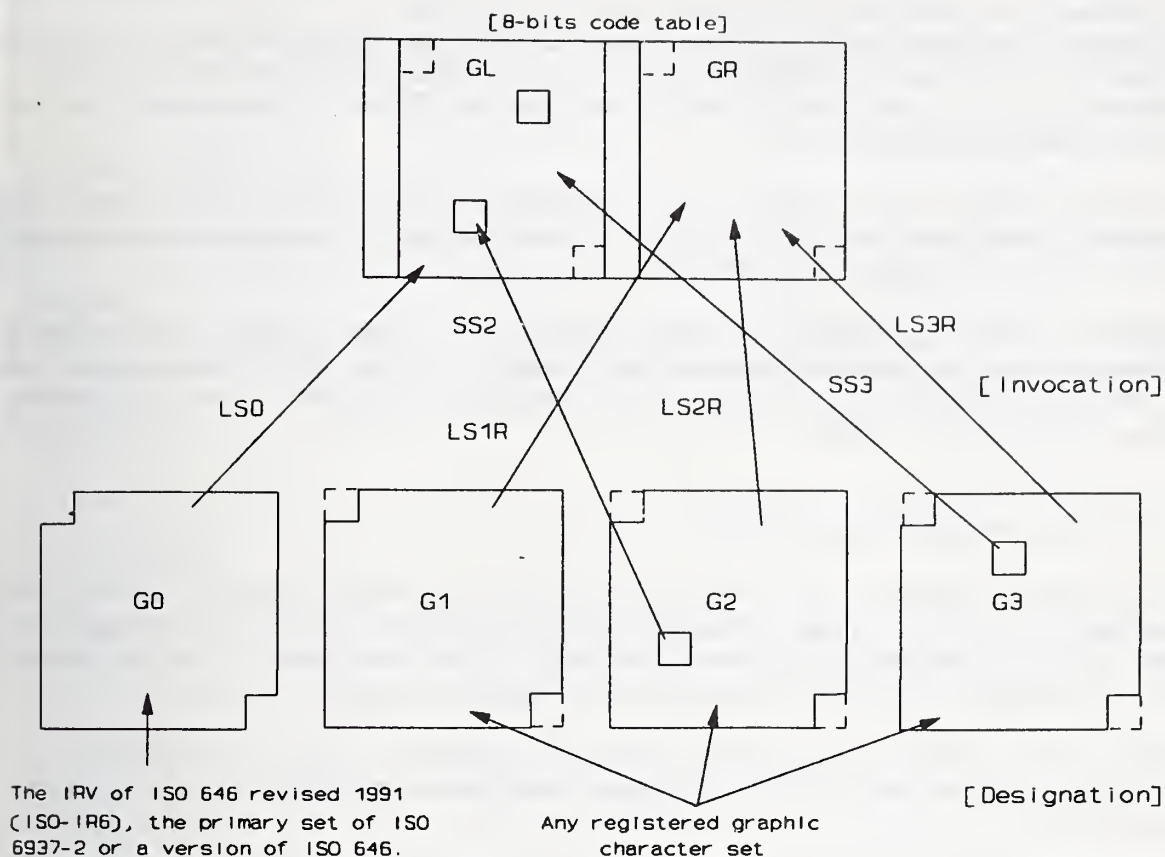


Figure 20 — Code extension features (all possible cases)

content portion using the control functions select character spacing (SHS) or set character spacing (SCS).

NOTE 23 - SHS parameters of 0, 1, 2, 3 and 4 are currently provided. The use of parameters 5 and 6 may be provided in a future edition of this profile for use with Chinese characters.

6.5.1.7 Character path and line progression

Both horizontal and vertical writing directions may be used within a document. In the case of horizontal writing, the characters progress either from left to right or from right to left across the page and the line progression is from the top of the page to the bottom. In the case of vertical writing, the characters progress from the top of the page to the bottom and the line progression is from the right to the left. The use of these writing directions is restricted by the page layout type used.

For body layout A, only horizontal writing may be used in the body area. Thus the character path and line progression is specified either as 0 and 270 degrees respectively or 180 and 90 degrees respectively.

For body layout B, again only horizontal writing may be used in the body area. However, in this case the content in the body area is presented for viewing with the page in landscape orientation and the content in the header and footer areas is presented for viewing with the page in portrait orientation. Thus for body layout B, in the body area the character path and line progression is specified either as 90 and 270 degrees respectively or 270 and 90 degrees respectively.

For body layout C, only vertical writing may be used in the body area. Thus the character path and line progression are specified as 270 and 270 degrees respectively.

With regard to the header and footer areas, if these areas are placed above and below the body area, as shown in figure 1, then only horizontal writing is allowed in these areas. Thus, in this case, the character path and line progression is specified either as 0 and 270 degrees respectively or 180 and 90 degrees respectively (see figure 2).

If the header and footer areas are placed to the left and right of the body area, then only vertical writing is allowed in these areas. Thus, in this case, the character path and line progression are specified as 270 and 270 degrees respectively (see figure 4).

All values of character path and line progression are basic. The values of character path and line progression may be specified at the beginning of the content associated with a basic component using the presentation attributes "character path" and "line progression" respectively. These values cannot be changed within a content portion.

6.5.1.8 Character positioning controls

The active position of a character (as defined in [CCITT Recommendation T.416 | ISO 8613-6]) can be moved forward or backward along the direction of the line progression using the control functions line position backward (VPB) and line position relative (VPR). These control functions may be specified in all forms of character content, and any parameter value may be specified.

The active position of a character can be moved forward or backward along the direction of the character path using the control functions character position backward (HPB) and character position relative (HPR).

The spacing between characters can be increased or decreased using the control functions set additional character spacing (SACS) and set reduced character spacing (SRCS) respectively. Also, the width of the character SPACE can be set using the control function set SPACE width (SSW).

The control functions HPB, HPR, SACS, SRCS and SSW shall only be specified in formatted or formatted processable character content; any parameter value may be specified.

6.5.1.9 Character orientation

The character orientation may be specified as 0, 90, 180 or 270 degrees.

The orientations of 0, 90, and 180 degrees are usually applied depending on whether vertical or horizontal writing is used. When horizontal writing is used, characters may only be orientated at 0 degrees. When vertical writing is used, characters may be orientated at 90 or 180 degrees.

All values of character orientation are basic. The value of the character orientation is specified at the beginning of the content associated with a basic component by the presentation attribute "character orientation". This value cannot be changed within the content.

6.5.1.10 Graphic character composition

A string of two or more characters may be combined to form a single character using the control function graphic character composition (GCC). Parameter values 0, 1 and 2 may be specified.

6.5.1.11 Emphasis

The following modes of emphasising graphic characters may be distinguished:

- normal rendition;
- normal intensity;
- increased intensity (bold);
- italicized;
- not italicized;
- underlined;
- doubly underlined;
- not underlined;
- crossed-out;
- not crossed-out;
- slowly blinking;
- rapidly blinking;
- not blinking;
- negative image;
- positive image.

All the above modes of emphasis are basic. If no default mode is explicitly specified in the document profile, then the default mode is normal rendition.

The mode of emphasis may be specified at the beginning of the content associated with a basic component using the presentation attribute "graphic rendition". The mode may be changed anywhere within the content using the control function set graphic rendition (SGR).

The mode of emphasis remains in effect within the content associated with a basic component until changed into a mutually exclusive mode or by the specification of 'normal rendition'. Mutually exclusive modes are normal/increased/decreased intensity, not/slowly/rapidly blinking, italicized/not italicized, underlined/doubly underlined/not underlined, crossed-out/not crossed-out and positive/negative image. One mode from each mutually exclusive set may be in operation at any point in the document content.

Normal rendition cancels the effect of all modes of emphasis that are currently in operation and specifies that the text shall be displayed in accordance with the default rendition parameters set for the presentation device. Thus, for example, if it is required to ensure that the content is not underlined, then it is necessary to explicitly specify that underlined is not to be used.

6.5.1.12 Tabulation

Tabulation stop positions may be specified at any position along the character path. Each stop is specified by means of the following:

- a) The tabulation position relative to the position of that margin in the direction opposite to the character path;
- b) An optional alignment qualifier that specifies the type of alignment to be used at the designated tabulation position. The type may be specified as one of the following:
 - 1) start aligned;
 - 2) end aligned;
 - 3) centred;
 - 4) aligned around.

These alignment qualifiers are defined in [CCITT Recommendation T.416 | ISO 8613-6]. If the alignment qualifier is not explicitly specified, then it is assumed that start aligned is to be used.

Only one set of tabulation stops may be specified to be applicable to the content associated with a basic component. No limit is placed on the number of tabulation stops that may be specified within a given set.

The set of tabulation stop positions associated with the content of a basic component are specified using the presentation attribute "line layout table". Tabulation stop positions are invoked within the content using the control function selective tabulation (STAB).

The tabulation reference numbers used in the STAB controls and associated Line Layout Table shall be chosen so that in a given Line Layout Table, the reference numbers are unique, sequential in the direction of the character path, and do not include leading zeroes.

6.5.1.13 Indentation

Indentation is the distance between the first character on a line of content and the position of the margin that is in the direction opposite to the character path. Thus the value of the indentation specified determines the line home position (as defined in [CCITT Recommendation T.416 | ISO 8613-6]).

Indentation is an offset of that margin that is in the direction opposite to the character path. When text is formatted, it is intended to be laid out between the indentation position and the margin position in the direction of the character path.

Any value of indentation may be specified for basic logical components using the presentation attribute "indentation". The indentation value shall not be changed within a content portion.

6.5.1.14 Alignment

This feature is concerned with how the first and last characters on each line of character content is to be laid out during the formatting process.

The following values of alignment may be specified as basic:

- start aligned;
- end aligned;
- centred;
- justified.

The semantics of these values are as defined in [CCITT Recommendation T.416 | ISO 8613-6].

The presentation attribute "alignment" is used to specify the alignment that is applicable to the content associated with a basic component. The alignment value cannot be changed within a content portion.

6.5.1.15 First line format

This feature specifies how the first line of the content associated with a basic component is to be laid out and provides for the itemization of paragraphs.

It allows the first character in the content to be positioned at some point along the character path relative to the indentation position (as specified in 6.5.1.13). This point may be in the direction of the character path or in the direction opposite to the direction of the character path relative to the indentation position.

In addition, this feature provides for the specification of an item identifier on the first line. The item identifier is a string of characters that precedes and is separated from the remaining characters that form the first line. The control function carriage return (CR) is used as the separator.

The features provided correspond to examples 10.1 to 10.5 shown in figure 10 of [CCITT Recommendation T.416 | ISO 8613-6].

First line format is specified by the presentation attributes "first line offset", "indentation" and "itemization". Only those values of the attributes which combine to form the examples shown in figure 10 of [CCITT Recommendation T.416 | ISO 8613-6] may be used.

6.5.1.16 Widow and orphan sizes

The widow size specifies the minimum number of lines of content that shall be allocated to a following frame or page when the content associated with a basic logical component is laid out such that it flows over two frames or pages. To accommodate this, it may be necessary to move a number of lines of content from one frame or page to the next frame or page.

The orphan size specifies the minimum number of lines of content that shall be placed in the current frame or page when the content associated with a basic logical component is split over two frames or pages. If this minimum cannot be accommodated, then the whole content shall be placed in the next frame or page.

Any value of widow or orphan size may be specified using the presentation attributes "widow size" and "orphan size" respectively.

Widow and orphan size shall only be specified for character content placed in the body area of pages.

6.5.1.17 Fonts

Any number of fonts may be used within a document. The fonts used in a particular document are specified in the document profile using the attribute "fonts list".

Further information concerning the specification of font references in the document profile is given in Annex B.2.

The fonts that may be used within the content associated with each basic component are specified by the presentation attribute "character fonts". Up to 10 fonts taken from the list specified by the attribute "fonts list" may be specified by the attribute "character fonts".

The font to be used at the start of the content associated with a basic component is specified using the attribute "graphic rendition". The fonts used within the content may be changed using the control function set graphic rendition (SGR).

The document profile may specify, using the attribute "character fonts", a default set of up to 10 fonts that are applicable to the whole document.

6.5.1.18 Reverse character strings

Bi-directional writing is supported by this profile. Hence, a string of characters in a content portion associated with a basic component may be specified to be imaged in the reverse direction of the immediately preceding character string. Such strings may be specified by the control function start reverse string (SRS) as defined in [CCITT Recommendation T.416 | ISO 8613-6].

This control function is provided for cases in which the text belongs to different languages and the character content is written, for example, from left to right or from right to left within the same line of characters, dependent upon the language and/or character set being used.

6.5.1.19 Parallel text

A string of characters in a content portion associated with a basic component may be specified to be imaged in parallel with another character string. Typical example is "ruby" in the Japanese language.

In processable and formatted processable content, parallel text may be specified by the control function parallel text (PTX).

In formatted content, the control functions character position relative (HPR), character position backward (HPB), line position relative (VPR) and line position backward (VPB) may be used to specify parallel text. These control functions may also be present in formatted processable content provided that they are contained within strings delimited by the control functions start of string (SOS) and string terminator (ST).

6.5.1.20 Kerning offset

A kerning offset value for the content associated with a basic component may be specified using the presentation attribute "kerning offset". It is necessary to specify such a value when certain fonts are invoked to ensure that no part of character images are positioned outside the boundary of the available area.

6.5.1.21 Pairwise kerning

Pairwise kerning may be specified to take place during the layout process using the attribute "pairwise kerning". This process depends upon the font used and the modification applied to the positions of characters depends on the kerning information in the font attributes. Pairwise kerning shall only be carried out if a variably spaced font is used; the attribute "pairwise kerning" is ignored if a constant spaced font is used.

6.5.1.22 Proportional line spacing

The use of proportional line spacing may be invoked for the content associated with a basic logical component using the attribute "proportional line spacing". When this invocation occurs, the line spacing between each pair of consecutive lines is determined in an implementation-defined way from the attributes associated with the fonts used within the two lines and may vary from one line to the next. This process is application dependent.

6.5.1.23 Superscripts and subscripts

Superscripts and subscripts may be specified anywhere within the content associated with a basic component by using the control functions partial line up (PLU) and partial line down (PLD). The use of these control functions shall be in accordance with [CCITT Recommendation T.416 | ISO 8613-6].

6.5.1.24 Line breaks

The control functions break permitted here (BPH) and no break here (NBH) may be inserted in processable or formatted processable form character content to indicate where line breaks may occur or may not occur respectively when the content is laid out.

6.5.1.25 Substitution characters

The control function SUB is provided to represent characters produced by a local system that cannot be represented by a character within a character set supported by this profile.

6.5.1.26 Initial point

The initial point which is applicable to basic layout components may be specified by the attribute "initial offset". Any value may be specified.

6.5.1.27 Use of control functions

The following is a list of all the control functions and parameter values (where applicable) that may be specified in character content:

- SHS — select character spacing (allowed parameter values: 0, 1, 2, 3, 4);
- SCS — set character spacing (allowed parameter values: any);
- SVS — select line spacing (allowed parameter values: any);

SLS — set line spacing (allowed parameter values: any);

SGR — set graphic rendition (allowed parameter values: any);

STAB— selective tabulation (allowed parameter values: any);

SRS — start reverse string (allowed parameter values: any);

GCC — graphic character composition (allowed parameter values: any);

IGS — identify graphic subrepertoire (allowed parameter values: any);

VPB — line position backward (allowed parameter values: any);

VPR — line position relative (allowed parameter values: any);

HPB — character position backward (allowed parameter values: any);

HPR — character position relative (allowed parameter values: any);

SACS— set additional character spacing (allowed parameter values: any);

SRCS— set reduced character spacing (allowed parameter values: any);

SSW — set SPACE width (allowed parameter values: any);

PTX — parallel text;

PLD — partial line down;

PLU — partial line up;

BPH — break permitted here;

NBH — no break here;

JFY — no justify;

SUB — substitute character;

SP — space;

CR — carriage return;

LF — line feed;

SOS — start of string;

ST — string terminator;

— code extension control functions (see 6.5.1.4).

The use of all these control functions, with the exception of SP, CR, LF, SOS and ST, are described in various subclauses in 6.5.1.

6.5.1.28 Formatting the content

The attribute "formatting indicator" may be specified for particular basic objects that are conformant with this profile.

The effect is to provide for transmission of formatted (or formatted processable) objects for which the precise placement of individual characters has been fully computed, and all necessary control functions included. The implication is that most operations normally performed by the imaging processing in handling formatted character text will be rendered unnecessary.

To make use of this, the imaging process of the recipient shall operate with a font containing metrics identical to that used by the originator.

6.5.2 Raster graphics content

6.5.2.1 Introduction

This subclause defines the features that are applicable to the raster graphics content contained in a document. These features may apply to basic logical and layout components unless otherwise indicated.

The default values for the following features may be specified in the document profile:

- type of coding;
- compression;
- pel spacing;
- spacing ratio;
- clipping;
- image dimensions;
- pel path;
- line progression.

The specification in a document of a non-basic feature by a presentation or coding attribute or control function shall be indicated in the document profile.

6.5.2.2 Raster graphics content architecture

Only the formatted processable raster graphics content architecture class may be used in documents that conform to this document application profile. This type of content may be used in processable, formatted and formatted processable form documents.

When using raster graphics content, only one content portion may be associated with an object or object class.

The content information in a content portion may be absent. This is to allow the representation and interchange of documents in which parts of the content may be supplied, for example, during subsequent editing.

Also, the scalable or fixed dimension content layout process may be used when laying out and imaging the content depending upon the specification of the presentation attributes "pel spacing" and "imaging dimensions" as described in 6.5.2.6 and 6.5.2.8. Both forms of content layout processes may be used in a single document.

6.5.2.3 Raster graphics encoding methods

The content may be encoded in accordance with the encoding schemes defined in CCITT Recommendations T.4 and T.6. In the case of T.4, either the one-dimensional or two dimensional encoding scheme may be used. Also the 'bit-map encoding' scheme defined in [CCITT Recommendation T.417 | ISO 8613-7] may be used. All these forms of encoding may be used in a single document, and all are basic. 'Uncompressed' mode of encoding may also be used, but as a non-basic feature.

When using the T.4 or T.6 encoding method, the relationship between the order of pels and the order of bits in the octets in the coded data stream shall be such that the first pel in the order of bits is allocated to the least significant bit of an octet. In the case of bit-map encoding, the order of encoding shall be that the first pel is allocated to the most significant bit of an octet.

In a content portion, if content information is specified, it is required that the coding attribute "number of pels per line" is specified; the coding attribute "number of lines" may also be specified. No restriction is placed on the values that may be specified for these coding attributes. Thus this profile places no restriction on the size of the pel arrays that may be used.

The type of encoding method used is specified by the attribute "type of coding". The use of this attribute is non-mandatory. Thus, if this attribute is not specified for a particular content portion and if the content architecture class specified corresponds to the formatted processable raster graphics content architecture class, then the default encoding method is assumed to be T.6.

6.5.2.4 Pel path and line progression

The pel path direction may be specified as 0, 90, 180 or 270 degrees. The line progression direction may be specified as 90 or 270 degrees.

A pel path of 0 degrees and a line progression of 270 degrees are basic values. All other values are non-basic and their use shall be indicated in the document profile.

6.5.2.5 Clipping

A sub-region within a pel array represented by a content portion associated with a basic component may be defined using the presentation attribute "clipping". No restriction is placed on the use of this attribute.

6.5.2.6 Pel spacing

The pel spacing is the distance in SMUs between any two pels on a line when a pel array is imaged. Any value may be explicitly specified provided that the spacing between pels is not less than 1 SMU.

The pel spacing need not be an integer value. Also, the value 'null' may be specified, indicating that the scalable layout process is to be used.

The specification of the value 'null' or spacings of 16, 12, 8, 6, 5, 4, 3, 2, and 1 SMU between adjacent pels are basic. The specification of any other spacing is non-basic and shall be indicated in the document profile.

The pel spacing applicable to content associated with basic logical components is specified by the presentation attribute "pel spacing".

NOTE 24 - The basic pel spacing values listed above are equivalent to resolutions of 75, 100, 150, 200, 240, 300, 400, 600 and 1200 pels per 25.4mm respectively when the SMU is interpreted as 1/1200 inch.

NOTE 25 - The attribute "pel spacing" specifies two integers, the ratio of which determines the pels spacing. No restriction is placed on the values of these integers.

6.5.2.7 Spacing ratio

The spacing ratio is the ratio between the pel spacing and the line spacing when a pel array is imaged. This ratio is used to determine the line spacing from the pel spacing specified.

No restrictions are placed on the value of this ratio providing that the resultant line spacing is not less than 1 SMU. Also, the line spacing need not be an integral number of SMUs. All values are basic.

The default value may be specified in the document profile. If no default value is explicitly specified then the default value is the ratio 1:1, that is, the line spacing is equal to the pel spacing.

The spacing ratio applicable to the content associated with a basic logical component is specified by the presentation attribute "spacing ratio".

6.5.2.8 Image dimensions

The image dimensions are the constraints to be applied to the size of the image produced when laying out a pel array represented by a content portion associated with a basic logical component.

These constraints are specified for basic logical components by the presentation attribute "image dimensions". The value of this attribute is only taken into account if the value of the attribute "pel spacing" is 'null'.

6.5.3 Geometric graphics content

A document may contain graphic images composed of geometric graphic content encoded as CGM metafiles in accordance with ISO 8632. Each CGM figure shall consist of a single picture only. Each CGM figure may specify its minimum dimensions.

6.6 Miscellaneous features

6.6.1 Resource documents

Object classes of the types BodyText, BodyRaster, BodyGeometric, CommonText, CommonRaster, CommonGeometric, EntryText, EntryRaster, EntryGeometric and GenericBlock may refer to corresponding constituent constraints in a resource generic-document.

The constituent constraints in the resource document may refer to content portions and to layout and presentation styles that are contained within the resource document. The constituent constraints listed above are the only ones that are allowed to be referenced from another document via the resource attribute: however, generic-documents used as resource documents may contain any combination of generic constituent constraints which is conformant to this document application profile.

6.6.2 External documents

In the case of processable and formatted processable, the generic logical structure and the generic layout structure, if present, may be contained in an external document. Note that it is not permitted to exchange one generic structure in the interchanged document while referencing the other through the external document.

6.6.3 Border

Borders may be specified for all the frame types defined in 6.3.5 and 6.3.6 using the attribute "border". All the features of borders specified in [CCITT Recommendation T.412 | ISO 8613-2] may be specified as basic values. Borders may also be specified in presentation styles.

6.6.4 Unit scaling

The document profile attribute "unit scaling" may be specified to indicate a scaling factor that shall be applied to all attributes and control functions values that specify absolute or relative positions and dimensions. This attribute specifies two integers, m and n , which indicate that these values are to be interpreted as being equal to m/n MU.

6.6.5 Application comments

Specification of the attribute "application comments" is mandatory for all object classes contained in a document that conforms to this profile. Specification and use of this attribute is optional.

This attribute is structured so that it contains two fields. The first field is mandatory when the attribute is specified and contains a numeric string which uniquely identifies the constituent constraint applicable to the constituent for which the attribute is specified. This facilitates the processing of documents. A list of these identifiers is given in table 5 and 6.

NOTE 26 -

- a) The values of the constituent constraint numeric identifiers are not unique between the logical and layout structures, and therefore in order to identify the constituent constraint applicable to a constituent, it is necessary to know the structure of which the constituent is a part.

- b) For constituent constraints that correspond to each other between the hierarchically related profiles to which this profile belongs, the same constituent constraint numeric identifier is specified.

The second field is optional and may contain any information that is relevant to the application or user. The format of the second field is not defined in this profile and the interpretation of this field depends upon a private agreement between the originator and recipient of the document.

The encoding of the attribute "application comments" is defined in 8.1.3 and 8.2.3.

6.6.6 Alternative representation

The content information in a content portion may be replaced by a string of characters specified in the attribute "Alternative representation". This attribute may be specified in content portions that contain character, raster graphics or geometric graphics content.

The specification and use of this attribute is optional. The string of characters specified shall belong to the character repertoires indicated in the document profile attribute "alternative representation character sets" (see 6.7.4.3). If the latter attribute is not explicitly specified in the document profile, then the default defined in [CCITT Recommendation T.410 series | ISO 8613] is assumed. The control functions space (SP), carriage return (CR) and line feed (LF) may also be used within the character string but no other control function is allowed; hence graphic character sets cannot be changed within the alternative representation.

6.6.7 Automatic numbering and referencing mechanisms

6.6.7.1 Introduction

This profile provides general mechanisms to support the automatic numbering of various types of constituents in a document and for referencing those numbers from other constituents in the document. For example, the numbering of segments (such as chapters, sections and annexes), tables, figures, footnotes, lists of items and pages is supported.

Also character strings may be specified for various constituents and these strings may be referenced from other parts of a document in order to support a general referencing mechanism within a document.

To achieve these features, this profile provides the bindings listed in 6.6.7.2. Subclauses 6.6.7.3 to 6.6.7.11 describe how these bindings are used in the automatic numbering and referencing schemes. These descriptions are not intended to restrict the use of the bindings provided by this profile or the mechanisms used to achieve these numbering and referencing schemes.

6.6.7.2 Bindings

The binding listed below may be specified, unless otherwise stated, on any composite logical constituent, on basic logical constituent constraints of the types BodyText, BodyRaster and BodyGeometric and on layout constituent constraints of the types DocumentLayoutRoot, PageSet, Page, RectoPage and VersoPage.

Table 5 — List of number string identifiers for logical constituents

Logical constituent	Numeric string identifier
DocumentLogicalRoot	0
Passage	1
NumberedSegment	2
Number	3
Title	4
Caption	5
Paragraph	6
Phrase	7
Footnote	8
FootnoteNumber	9
FootnoteReference	10
FootnoteBody	11
FootnoteText	12
Figure	13
BodyText	14
Reference	15
ReferencedContent	16
BodyRaster	17
BodyGeometric	18
CommonContent	19
CommonText	20
CommonRaster	21
CommonGeometric	22
Description	23
Artwork	24
NumberedList	25
UnNumberedList	26
DefinitionList	27
ListItem	28
ListTerm	29
Table	30
Row	31
TableComponent	32
RowComponent	33
Form	34
EntryElement	35
EntryGroup	36
CommonReference	37
CommonNumber	38
CurrentInstance	39
PageNumber	40
EntryText	41
EntryRaster	42
EntryGeometric	43
TableNumber	44

Groups of bindings have names whose general form is '<name>-<n>'. The character string <name> serves to identify a particular group of bindings and <n> is a string of characters that serves to identify a particular binding. The field <n> is a sequence of characters taken from the set of characters '0..9'; this sequence may be of any length, but shall consist of a string representing an integer with no leading zeroes.

Table 6 — List of number string identifiers for layout constituents

Layout constituent	Numeric string identifier
DocumentLayoutRoot	0
PageSet	1
Page	2
RectoPage	3
VersoPage	4
CompositeHeader	5
FixedCompositeBody	6
VariableCompositeBody	7
ColumnFixed	8
ColumnVariable	9
SnakingColumns	10
SynchronizedColumns	11
BasicFloat	12
CompositeFloat	13
BasicColumn	14
FootnoteArea	15
ArrangedContentFixed	16
ArrangedContentVariable	17
SourcedContentFixed	18
SourcedContentVariable	19
CompositeFixtureVariable	20
CompositeFixtureFixed	21
BasicFixture	22
CompositeColumnFixed	23
CompositeColumnVariable	24
CompositeCommon	25
CompositeArtwork	26
BasicHeader	27
BasicBody	28
GenericBlock	29
SpecificBlock	30
FormArea	31
CompositeFooter	32
BasicFooter	33
TableHeader	34
EntryGroupArea	35
TableArea	36
TableLabel	37
CompositeTableLabel	38
LabelComponent	39
RowArea	40
Cell	41
SubRowGroup	42
SubRow	43
TableLabelContent	44
FormEntryArea	45

Binding values may consist of integers or character strings. In the case of integers, any value may be specified. In the case of character strings, the string may consist of any of the 94 characters of the IRV of ISO 646 revised 1991, ISO-IR6, plus the character space. Any other character repertoire may be used provided it is designated and invoked by the appropriate designation and invocation sequences, and indicated in the document profile as a non-basic value. No other control functions may be used.

6.6.7.2.1 Binding 'prefixes-<n>'

This binding specifies a character string that is typically used to specify a prefix string in a character string represented by another binding. Examples are prefixes to segment, footnote and page numbers.

6.6.7.2.2 Binding 'suffixes-<n>'

This binding specifies a character string that is typically used to specify a suffix string in a character string represented by another binding. Examples are suffixes to segment, footnote and page numbers.

6.6.7.2.3 Binding 'numberstring-<n>'

This binding specifies a character string that typically consists of one or more numerals and separators that constitutes, for example, a segment, figure, table, list item or footnote number in a document. An example is the string '3.4.3.6' which might identify a sub-section in a document.

6.6.7.2.4 Binding 'numbers-<n>'

This binding specifies an integer that is associated with a particular constituent. This integer is typically used to generate a numeral or a sequence of numerals, represented by the binding 'numberstring-<n>' that identifies, for example, a particular segment, footnote, table, figure, list item or page within a document.

6.6.7.2.5 Binding 'separator-<n>'

This binding specifies a character string that is typically used to represent the separators between numerals in a string represented by the binding 'numberstring-<n>'. An example is the string '3.4.3.6' where the character '.' forms the separator.

6.6.7.2.6 Binding 'string-<n>'

This binding specifies a character string that is associated with a constituent and is used to support a general referencing mechanism with a document. Typically this binding is used to support the referencing of character content specified in one part of the document from another part of the document. For example, this binding might be used to carry the title of a chapter or figure that is referenced in some other part of the document.

6.6.7.2.7 Binding 'PGnum'

This binding specifies an integer that typically represents a page number. This binding may only be specified on layout constituent constraints of the types DocumentLayoutRoot, PageSet, RectoPage, VersoPage and Page.

6.6.7.2.8 Binding 'fnotenumber'

This binding specifies an integer that is specifically provided to represent the numbers that identify footnotes. This binding may only be specified for the logical constituent constraints DocumentLogicalRoot, Passage and Footnote, and is provided for compatibility with FOD26.

6.6.7.2.9 Binding 'fnotestring'

This binding specifies a character string which is the equivalent of the number represented by the binding 'fnotenumber'. This binding may only be specified for the logical constituent constraint Footnote, and is provided for compatibility with FOD26.

6.6.7.3 Numbering of segments

The constituent Number contains a content generator which, when evaluated during the layout process, produces a character string that serves to identify the NumberedSegment to which the constituent constraint Number belongs.

The format of this character string is as follows:

`<pre-str><num-str><suf-str>`

This format is defined by a string expression which is specified by the macro SEGMENTNUMBER (see 7.3.1). The description below indicates how this string is typically generated.

The fields `<pre-str>` and `<suf-str>` are optional prefix and suffix character strings respectively which may be of any length. These may be predefined in the expression or derived from bindings of the type 'prefix-`<n>`' and 'suffix-`<n>`' respectively that are defined on constituents at higher levels in the document structure.

The field `<num-str>` is the segment identifier which has the following general form:

`<number>[<separator><number>]...`

where [...]... indicates optional repetition. That is, a segment identifier consists of a single numeral or a sequence of two or more numerals, each of which is separated by a character string called a 'separator'. An example is the string '7.3.3.1'.

Segment identifiers are represented by the binding 'numberstring-`<n>`'. This binding may be explicitly specified or generated automatically by a string expression defined by the macro USENUMBERSTRINGS (see 7.3.1) which has the general form:

`<numberstring-x><separator-y><number-z>`

The field `<numberstring-x>` is a reference to the segment identifier (that is, another instance of a binding of the type 'numberstring-`<n>`') that is specified for the immediately superior NumberedSegment. This allows hierarchically structured numbering schemes to be specified. If this NumberedSegment does not exist, then this field is empty and, in this case, non-hierarchical.

The field `<separator-y>` is character string derived from a binding of the type 'separator-`<n>`' specified at some higher level in the document structure. This field may be empty.

The field <number-z> is the number represented in the form of a character string that is applicable to the NumberedSegment whose identifier is being constructed. This number may be represented in the form of an Arabic numeral string, upper or lower case Roman numeral string or by upper or lower case alphabetic characters.

The integer value corresponding to the field <number-z> may be generated by one of two methods. The value may be generated by a ORDINAL function within the expression defined by the macro USENUMBERSTRINGS. Alternatively, it may be derived by a binding of the type 'number-<n>' which is specified for the NumberedSegment whose identifier is being constructed. The binding 'number-<n>' is initialized at some suitable point in the document and is then automatically incremented on each successive NumberedSegment. This is achieved using an expression defined by the macro USENUMBERS (see 7.3.1).

The constructed binding 'numberstring-<n>' generated by the macro USENUMBERSTRINGS may then be referred to by a content generator specified by the constituent Number to generate the identifier of the NumberedSegment as indicated earlier in this subclause.

This binding is also available for constructing the segment identifiers used at lower levels of NumberedSegment. This means that a hierarchical numbering scheme may be specified for the NumberedSegments at different levels of the document structure.

The level number shall be indicated using the smallest possible number of characters, that is, there shall be no leading zeroes.

A document may contain any number of different independent numbering schemes. This is achieved by setting the value of bindings of the type 'number-<n>', 'numberstring-<n>', 'prefix-<n>' and 'suffix-<n>'; and the expressions indicated above at appropriate points in the document structure.

The above mechanism may be used for different purposes; subsequent subclauses illustrate how this mechanism is typically used for the numbering of figures, tables, lists of items and footnotes.

6.6.7.4 Numbering of figures

The mechanism used for numbering figures is the same as that used for numbering segments (see 6.6.7.3). That is, the number of a figure is generated by a content generator specified in the constituent Number which is immediately subordinate to the given Figure. This content generator contains an expression whose value is defined by the macro SEGMENTNUMBER. The number associated with each figure may be represented by a binding of the type 'number-<n>' and a binding 'numberstring-<n>' is used to specify the character string that represents the figure number. The generation of these bindings is as described in 6.6.7.3.

The figures in a document may be consecutively numbered throughout the document irrespective of the part of the document in which the figure are contained. Alternatively, the numbers may be linked to the part of the document to which they relate; for example, the figures in chapter 3 of a document can be specified as 3.1, 3.2, 3.3 and so on.

6.6.7.5 Numbering of lists

A list of items that are individually numbered is represented by the logical constituent NumberedList. Each item is represented by a constituent constraint of the type ListItem and each number belonging to each item is represented by a preceding constituent constraint of the type Number.

Number contains a content generator which, when evaluated, generates the number belonging to the subsequent item. The format of this content generator is the same as that used for numbering segments (see 6.6.7.3) and is defined by the macro SEGMENTNUMBER.

As described in 6.6.7.3, the binding 'numberstring-<n>' is used to represent the number string belonging to each item. In this case, this binding is generated on the constituent constraint Number which is referred to by the content generator contained in the same constituent constraint. (Note that in the case of segment numbers, the binding 'numberstring-<n>' is generated on the superior object.)

6.6.7.6 Numbering of tables

The constituent constraint, TableNumber, is typically used to represent a character string that constitutes the number that relates to a Table. This string is laid out in a TableHeader frame by means of a SourcedContentFixed frame. The latter frame specifies the attribute "logical source" which indicates the instance of the constituent constraint CommonContent that contains the subordinate TableNumber which specifies the required table number.

The character string represented by TableNumber is generated by a content generator which defines a string expression specified by the macro TABLENUMBER (see 7.3.1). The general format of this character string is as follows:

<pre-str><num-str><suf-str>

The fields <pre-str> and <suf-str> are optional prefix and suffix character strings which are pre-defined in the expression or are derived from bindings of the types 'prefix-<n>' and 'suffix-<n>' respectively that are specified for constituents at higher levels in the document structure.

The field <num-str> is a character string that represents the identifier of the table being laid out. It is obtained from the binding 'numberstring-<n>' which is specified for the logical object of the type Table which is being laid out. That is, the field <num-str> is derived from the current instance of the constituent constraint of the type Table.

The general format of the field <num-str> and the mechanisms used to specify and generate this field are described in 6.6.7.3.

Tables, like figures, may be independently numbered throughout a document, or their numbers may be linked to the segments in which they are contained. An example of a table number is the string '1.1.5' where '1.1' is the number of the segment to which the table belongs, '.' is a separator and '5' is the number associated with the particular table.

6.6.7.7 Footnote numbering

The constituent constraints FootnoteReference and FootnoteNumber contain content generators which, when evaluated during the layout process, produce character strings that serve to identify the Footnote to which the constituent constraints FootnoteReference and FootnoteNumber are subordinate.

The format of this character string is as follows:

<pre-str><num-str><suf-str>

This string is defined by a string expression specified by the macro FNNUMBER (see 7.3.1).

The numbering mechanism is the same as that used for numbering segments (see 6.6.7.3). Thus the fields <pre-str> and <suf-str> are optional prefix and suffix character strings respectively which may be of any length. These are derived from bindings of the type 'prefix-<n>' and 'suffix-<n>' respectively that are defined on constituents at a higher level in the document structure.

The field <num-str> may be derived by one of three methods. It may be represented by bindings of the type 'numberstring-<n>' which are derived using an expression specified by the macro USENUMBERSTRINGS as described in 6.6.7.3. Alternatively, it may be derived from a binding of the type 'fnotestring' which is specified on the constituent Footnote to which the constituent constraints FootnoteReference and FootnoteNumber are subordinate. This field is automatically generated using expressions defined by the macros INCFNOTENUMBER and FNOTENUMBERSTRING. The field <num-str> may also be explicitly specified; this case is defined by the macro FNOTESTRINGLITERAL.

The above mechanisms allow footnotes to be numbered consecutively throughout a document, or any number of independent footnote numbering schemes may be used. For example, the footnotes applicable to segments, figures and tables may all be independently numbered.

6.6.7.8 Page numbering

The constituent constraint PageNumber is specifically provided to represent common content that contains a page number and that is to be placed on each successive page of a document. A mechanism is provided which allows the page number to be automatically incremented on each successive page of a document.

The format of the content generator specified by the constituent PageNumber is as follows:

<pre-str><num-str><pre-str>

This format is defined by a string expression specified by the macro PGNUMBER (see 7.3.1).

The fields <pre-str> and <suf-str> are optional prefix and suffix character strings respectively which may be of any length. These may be explicitly specified in the expression, or they may be derived from bindings of the type 'prefix-<n>' and 'suffix-<n>' respectively that are defined on constituents at a higher level in the document structure.

The field <num-str> is the page number. This consists of a single number derived from the binding 'number-<n>' or 'PGnum' which is specified for the current instance of the frame or page in which the page number is to be laid out. A page number may be represented in the form of Arabic numeric strings, an upper or lower case Roman numeric string or an equivalent upper or lower case alphabetic string.

The binding 'number-<n>' is initialized at the document layout root, page set level or a particular page class (using the macro INITIALISEBINDINGS defined in 7.3.1). The binding 'PGnum' is initialized at the document layout root or page set level (using the macro INITIALISEPGNUM defined in 7.3.1). This binding is automatically incremented on each successive page using an expression specified by the macro USEPGNUMBERS (see 7.3.1). By placing initialization on the layout root, rather than on the pageset classes, the pagenumbers may be defined to be continued from one pageset to the next.

The content associated with logical object classes of the type PageNumber is laid out in a frame of one of the following types: BasicHeader, BasicFooter, SourcedContentVariable, SourcedContentFixed (see 6.3.6) using the logical source mechanism. Thus when the appropriate frame is being laid out, the field <num-expr> in the content generator contained in a logical object class of the type PageNumber is evaluated, and this determines the value of the binding 'number-<n>' or PGnum that is associated with the current page being laid out.

Similar numbering is applicable for page sets.

6.6.7.9 Referenced content

ReferencedContent is a constituent constraint that is provided to support a general referencing mechanism within the content of the body part of a document. This constituent constraint, therefore, is used to represent character content that contains a reference to content specified elsewhere in a document. Examples are references to strings that represent the numbers of segments, figures, tables, footnotes and pages. Particular strings specified by 'string-<n>' may be referenced.

This constituent constraint contains a content generator which, when evaluated, produces a character string of the following form:

<pre-str> a sequence of <ref-str><suf-str>

This content generator is defined by a string expression which is specified by the macro REF (see 7.3.1).

The fields <pre-str> and <suf-str> are optional prefix and suffix character strings which may be explicitly defined in the expression or may be derived from bindings of the type 'prefix-<n>' and 'suffix-<n>' respectively that are defined on constituents at a higher level in the document structure.

The field <ref-str> is a character string that is obtained from content specified on a particular constituent in the document by reference to one of the following bindings: 'numberstring-<n>', 'number-<n>', 'string-<n>', 'fnotestring' or 'PGnum'.

The following referencing mechanisms are permitted in this case:

- a particular logical object is referenced for the required binding;
- a particular logical object is referenced and a search for the required binding is made on the logical objects superior to the referenced object.

The referencing mechanism shall take into account the constituents for which the particular bindings are permitted, as defined in 6.6.7.2.

6.6.7.10 Common references

CommonReference is a constituent constraint with in the common part of the logical structure of a document that represents common character content that may be reproduced in more than one place in a document. This constituent constraint is specifically provided to represent content which contains a reference to content specified elsewhere in a document. This content is specified by bindings. Examples are references to character strings that represent the numbers of segments, figures, tables and footnotes in a document.

This constituent constraint contains a content generator, the general format of which is the same as the content generator specified for the constituent constraint ReferencedContent (see 6.6.7.9) and content may be obtained from the following bindings: 'numberstring-<n>', 'numbers-<n>', 'string-<n>', 'fnotestring' or 'PGnum'.

This content generator is defined by an expression which is specified by the macro COMMONREF.

The following referencing mechanisms are permitted in this case:

- a current page or frame, or a logical object which is laid out currently is referenced for the required binding;
- a current page or frame, or a logical object which is laid out currently is referenced and a search for the required binding is made on the constituents superior to a referenced constituent;
- a page or frame into which a particular logical object is laid out is referenced for the required binding;
- a page or frame into which a particular logical object is laid out is referenced and a search for the required binding is made on the layout objects superior to the referenced page or frame.

The referencing mechanism shall take into account the constituents for which the particular bindings are permitted, as defined in 6.6.7.2.

6.6.7.11 Current instance references

The constituent constraint `CurrentInstance`, like `CommonReference`, represents common character content that may be reproduced in more than one place in a document when it is laid out.

This constituent constraint is specifically provided to represent the current instance of a character string that is to be laid out in multiple places in a document. A typical example is the reproduction of a title of chapter on each page of a document in which that chapter is reproduced.

The constituent constraint `CurrentInstance` contains a content generator, the general format of which is the same as the content generator specified for the constituent constraint `ReferencedContent`. This content generator is restricted to referencing content obtained from bindings of the type `'string-<n>'`.

The referencing mechanism allows bindings of the type `'string-<n>'` to be referenced for the current instance of a logical constituent of a specified class or the current instance of a particular frame or page.

The format of this content generator is defined by the macro `CURRENTINSTANCE` (see 7.3.1).

6.6.7.12 Common number references

The constituent constraint `CommonNumber` represents common character content that may be reproduced in more than one place in a document when it is laid out.

This constituent constraint is specifically provided to represent identifiers of other parts of a document, that is, character strings that represent the numbers of segments, figures and tables within a document.

This constituent constraint contains a content generator, the general format of which is the same as the content generator specified for the constituent constraint `ReferencedContent` (see 6.6.7.9). This content generator may reference content specified by the bindings `'numberstring-<n>'` and `'numbers-<n>'`.

The format of this content generator is defined by the macro `COMMONNUMBER` (see 7.3.1).

6.6.8 User readable comments

Information which is to be interpreted as comments relevant to constituents and associated content portions may be specified using the attribute "user readable comments". This information is intended for presentation to humans.

The information consists of a string of characters which shall belong to one of the graphic character sets indicated in the document profile attribute "comments character sets" (see 6.7.4.2). If the latter attribute is not explicitly specified, then the default defined in [CCITT Recommendation T.410 series | ISO 8613] is assumed. The control functions space (SP), carriage return (CR) and line feed (LF) and code extension control functions may also be used within the character string, but no other control functions are allowed.

6.6.9 User visible name

Information which may be used to identify constituents within a document may be specified using the attribute "user visible name". This information is intended for presentation to humans, for example, to assist in the editing of documents.

The information consists of a string of characters which shall belong to one of the graphic character sets indicated in the document profile attribute "comments character sets" (see 6.7.4.2). If the latter attribute is not explicitly specified, then the default defined in [CCITT Recommendation T.410 series | ISO 8613] is assumed. The control functions space (SP), carriage return (CR), line feed (LF) and code extension control functions may also be used within the character string, but no other control functions are allowed.

6.7 Document management features

Information relating to the document as a whole is specified in the document profile which is represented by the constituent DocumentProfile. This constituent shall be specified in every document.

The information in the document profile is classified into the following categories:

- a) document constituent information;
- b) document identification information;
- c) document default information;
- d) non-basic characteristics information;
- e) document management information.

The information in the document profile may be of interest to the user, or may be used for machine processing of the document.

6.7.1 Document constituent information

This information specifies which constituents are used to represent the document, including constituents that are external to the interchanged document. This information is divided into three categories.

6.7.1.1 Presence of document constituents

This information indicates which constituents are included in the document. That is, this information indicates whether or not the document contains a generic logical structure, a specific logical structure, a generic layout structure, a specific layout structure, layout styles and presentation styles (see note). It is mandatory to specify this information in the document profile.

NOTE 27 - If the generic logical or layout structure is external to the document (see 6.7.1.3), then it is still necessary to indicate that these structures are present and form part of the document.

6.7.1.2 Resource document information

This information consists of a reference to a generic-document referred to as a resource document (see 6.6.1). This is specified by the attribute "resource document". If constituents in the document contain references to object classes in a resource document, then it is mandatory to specify this information in the document profile.

6.7.1.3 External document information

This information consists of a reference to an external document which may consist of a generic logical structure or both the generic logical and the generic layout structures (see 6.6.2). If such a reference is required, then it is specified by the attribute "external document class" in the document profile.

6.7.2 Document identification information

This information relates to the identification of the document. This information is divided into six categories.

6.7.2.1 Document application profile information

This information indicates the document application profile to which the document belongs. It is mandatory to specify this information using the attribute "document application profile".

6.7.2.2 Document architecture class information

This information indicates the document architecture class to which the document belongs (see 6.1). It is mandatory to specify this information using the attribute "document architecture class".

6.7.2.3 Content architecture classes information

This information indicates the content architecture classes used in the document (see 6.5.1.2, 6.5.2.2 and 6.5.3). It is mandatory to specify this information using the attribute "content architecture classes".

6.7.2.4 Interchange format class information

This information indicates the interchange format class used to represent the document (see clause 8). It is mandatory to specify this information using the attribute "interchange format class".

6.7.2.5 ODA version information

This information indicates the [CCITT Recommendation | ISO standard] to which the document conforms. It also specifies a calendar date, which indicates that the document conforms to the version of the [CCITT Recommendation | ISO standard] and any addenda that are current on that date. It is mandatory to specify this information using the attribute "ODA version".

6.7.2.6 Document reference

This information serves to identify the document. Typically this information is allocated to the document by the creator of the document. The identifier may consist of an ASN.1 object identifier or a string of characters. It is mandatory to specify this information using the attribute "document reference".

6.7.3 Document default information

This information specifies various default values for attributes used in the document. The default values that are allowed are specified in the various subclauses of clause 6. The specification of this information is only required when it is required to specify a default value which is other than the standard default value specified in [CCITT Recommendation T.410 series | ISO 8613].

Default values for the following groups of attributes may be specified:

- document architecture attributes;
- character content attributes;
- raster graphics attributes;
- geometric graphics attributes.

6.7.4 Non-basic characteristics information

This information specifies the non-basic attribute values specified in the document. It is mandatory to specify a non-basic attribute values in the document profile when such a value is used in the document.

The following types of non-basic attribute values may be specified:

- profile character sets;
- comments character sets;
- alternative representation character sets;
- page dimensions;
- medium type;
- character presentation features;
- raster graphics presentation features;
- raster graphics coding attributes.

Further information concerning document profile, comments and alternative representation character sets is given below.

6.7.4.1 Profile character sets

Some document profile attribute have values consisting of character strings, for example, the document management attributes. The character sets used in these character strings are specified by the document profile attribute "profile character sets".

This attribute "profile character sets" specifies a code extension announcer and designations of character sets, which are subject to the following restrictions:

- a) the code extension announcer shall be 04/03 when specified. This code extension announcer means to use G0 and G1 sets in an 8-bit environment and also the invocation of G0 and G1 sets into GL and GR respectively. Thus, in each attribute to which this attribute applies, invocation shift functions are not necessary because G0 and G1 sets are implicitly invoked by this code extension announcer.
- b) G0 set: only ISO-IR6 (the IRV of ISO 646 revised 1991), ISO-IR2 (the primary set of ISO 6937-2), or any other version of ISO 646 may be designated for this set; these graphic character sets are implicitly invoked in GL.
- c) G1 set: no restrictions are placed on the graphic character sets that may be designated for this set. These graphic character sets are implicitly invoked in GR.
- d) the empty set shall be designated into G1 and invoked into GR if no other specific character set is invoked in GR.

If the attribute "profile character sets" is not specified, then the default defined in [CCITT Recommendation T.410 series | ISO 8613] is assumed.

6.7.4.2 Comments character sets

The character sets assumed to have been designated and optionally invoked at the beginning of the character strings specified by the attributes "user readable comments" (see 6.6.8) and "user visible name" (see 6.6.9) are specified using the document profile attribute "comments character sets".

It also specifies code extension techniques and the graphic character sets which may be used in the attributes "user readable comments" and "user visible name".

If this attribute is specified, the code extension techniques which may be used in the attributes "user readable comments" and "user visible name" shall be announced by appropriate code extension announcers. The use of G0 set and GL shall always be announced. Other code extension announcers are to be specified according to the requirements of a particular document.

Two kinds of code extension techniques are permitted for this attribute. One is to GL and GR without shift functions, and the other is to use various character sets by shift functions. The former is rather restricted, but no shift functions are necessary in the "user readable comments" and "user visible name".

The same restrictions as in 6.7.4.1 is applied in this case. The latter permits various usages of character sets, but invocations shall be specified by shift functions in the "user readable comment" and "user visible name". The same restriction as in 6.5.4 is applied in this case.

All the graphic character sets which may be used in the attribute "user readable comments" and "user visible name" should be designated in the "comments character sets".

There are no restrictions concerning the number of graphic character sets which are designated and/or invoked in the "comments character sets"; hence designation to the same G set overrides the previous G set.

If the attribute "comments character sets" is not specified, the default defined in [CCITT Recommendation T.410 series | ISO 8613] is assumed.

6.7.4.3 Alternative representation character sets

This attribute specifies the graphic character sets designated and invoked at the beginning of the attribute "alternative representation" other than the standard default graphic character sets.

The restriction on profile character sets described in 6.7.4.1 is also applied. If this attribute is not explicitly specified in the document profile, the default defined in [CCITT Recommendation T.410 series | ISO 8613] is assumed.

6.7.5 Fonts list

This information specifies all the fonts (if any) used in the document. It is specified using the attribute "Fonts list" (see Annex B.2).

•

6.7.6 Document management attributes

Document management attributes contain information about the content of the document and its purpose. Information relating to the following may be specified:

- document description (see note);
- dates and times;
- originators;
- other user information;
- external references;
- local file references;
- content attributes;
- security information.

The attributes that may be used to specify this information are defined in [CCITT Recommendation T.414 | ISO 8613-4].

The string of characters used in the document management attributes shall belong to the character set indicated in the document profile attribute "profile character sets" (see 6.7.4.1). If the latter attribute is not explicitly specified in the document profile, then the default character set is the minimum subrepertoire of ISO 6937-2.

The control functions space (SP), carriage return (CR) and line feed (LF) may also be used within the character strings, but no other control functions are allowed. Therefore, the graphic character set cannot be changed in the document management attributes.

NOTE 29 - The document description includes the specification of the document reference (see 6.7.2.6).

7 Specification of constituent constraints

7.1 Introduction

The structure diagrams illustrating the relationships between the constituents in the logical structures are shown in 7.1.1. The macros indicated on these diagrams are defined in 7.3.1. These macros define the permissible values for the attribute "generator for subordinates" that are applicable to the constituents, and define the allowed structures that are supported by this profile.

The structure diagrams illustrating the layout structures are shown in 7.1.2. The macros indicated on these diagrams are defined in 7.4.1.

7.1.1 Diagrams of relationships of logical constituents

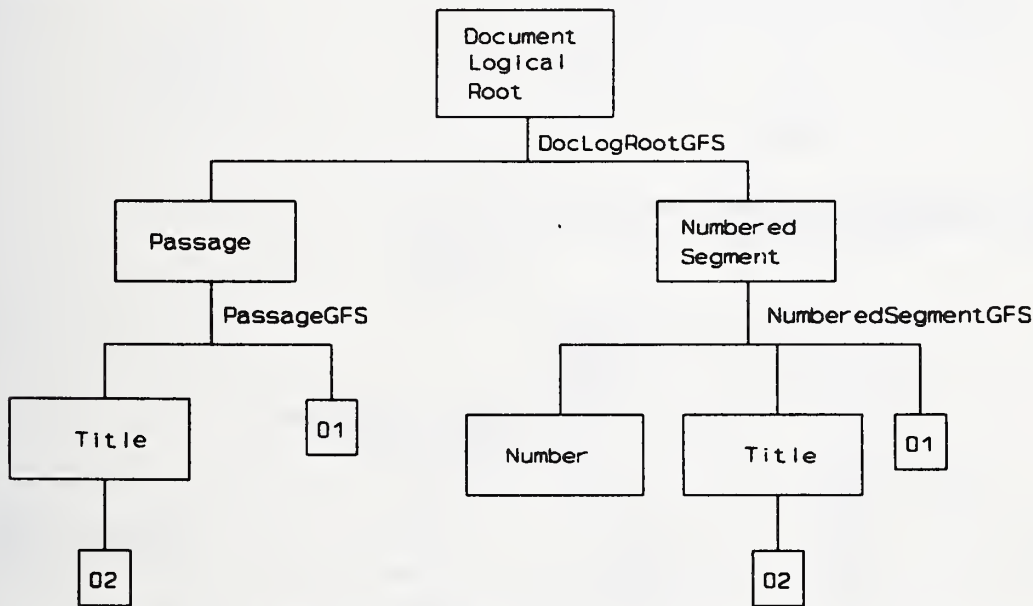


Figure 21 — DocumentLogicalRoot, 1st level

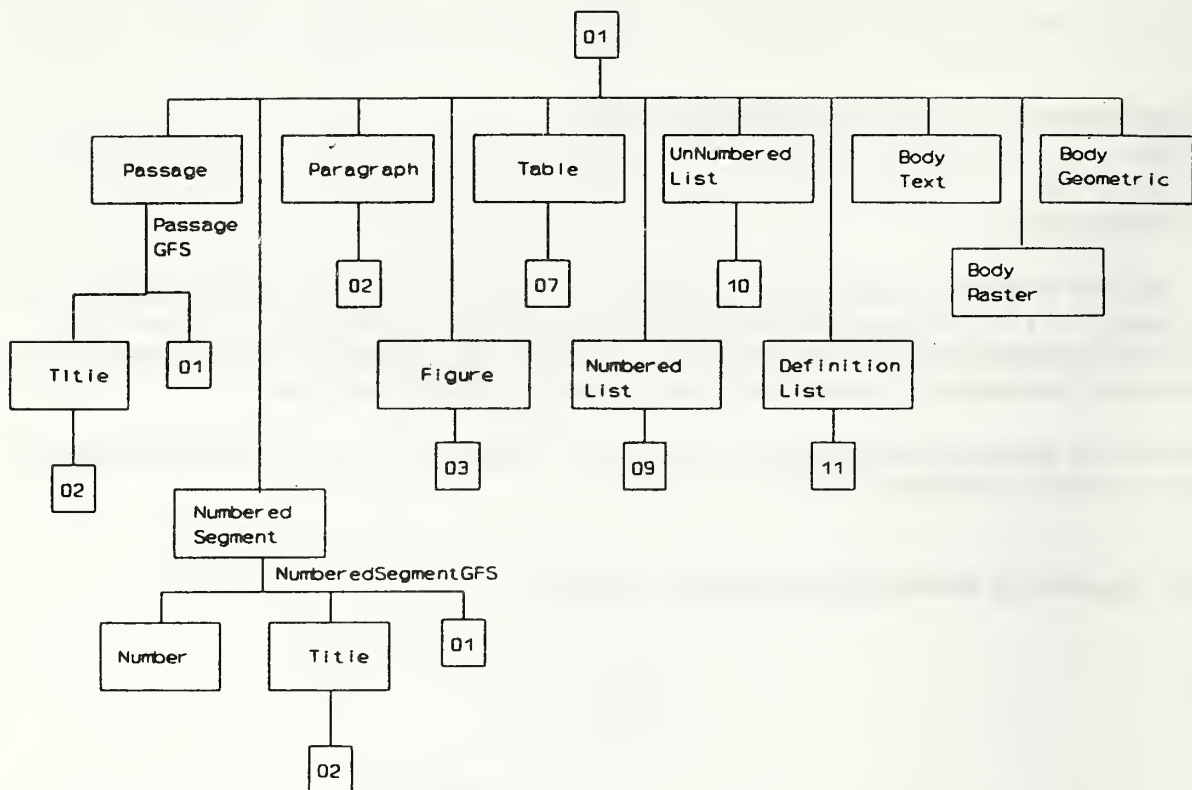


Figure 22 — DocumentLogicalRoot, 2nd level

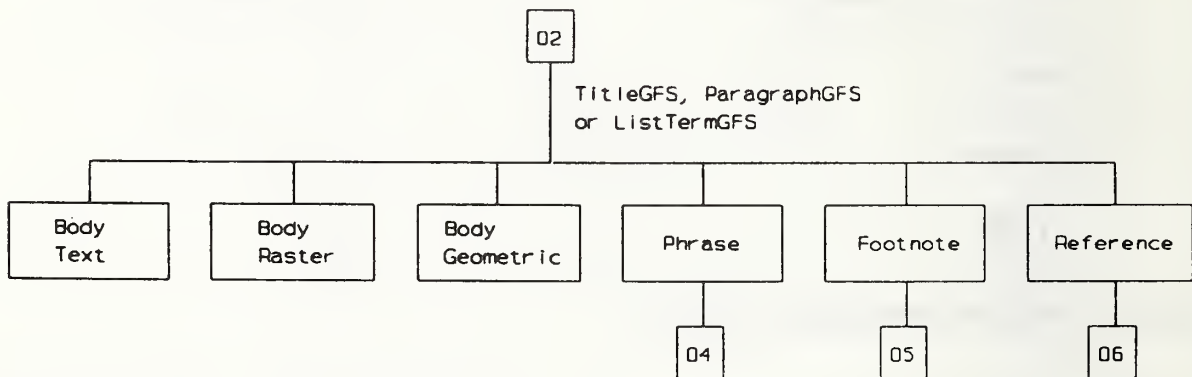


Figure 23 — Title, Paragraph and ListTerm

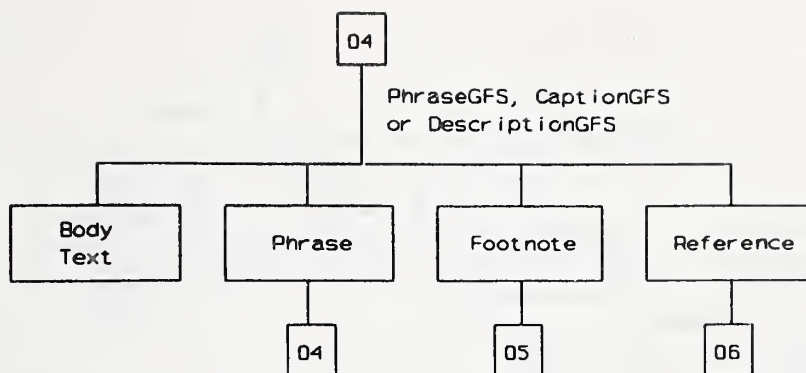


Figure 24 — Phrase, Caption and Description

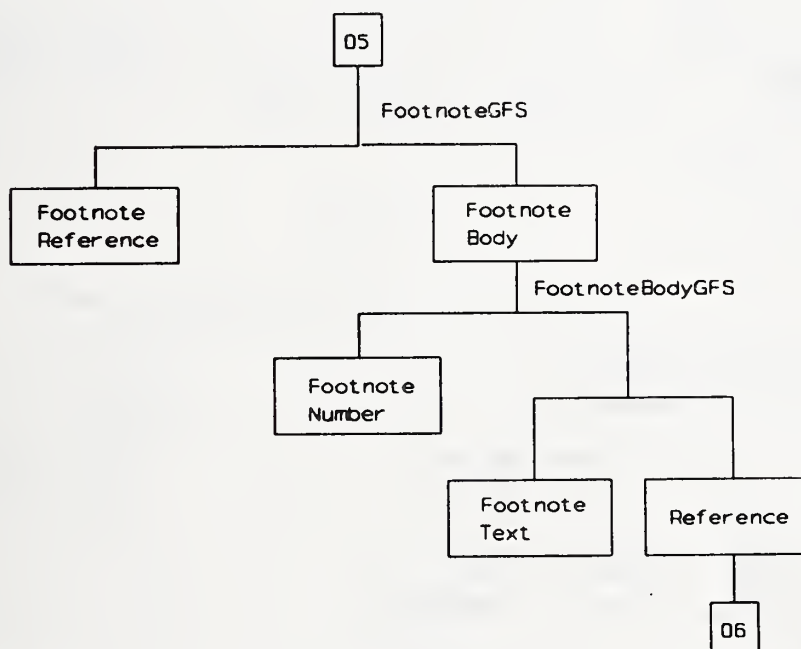


Figure 25 — Footnote

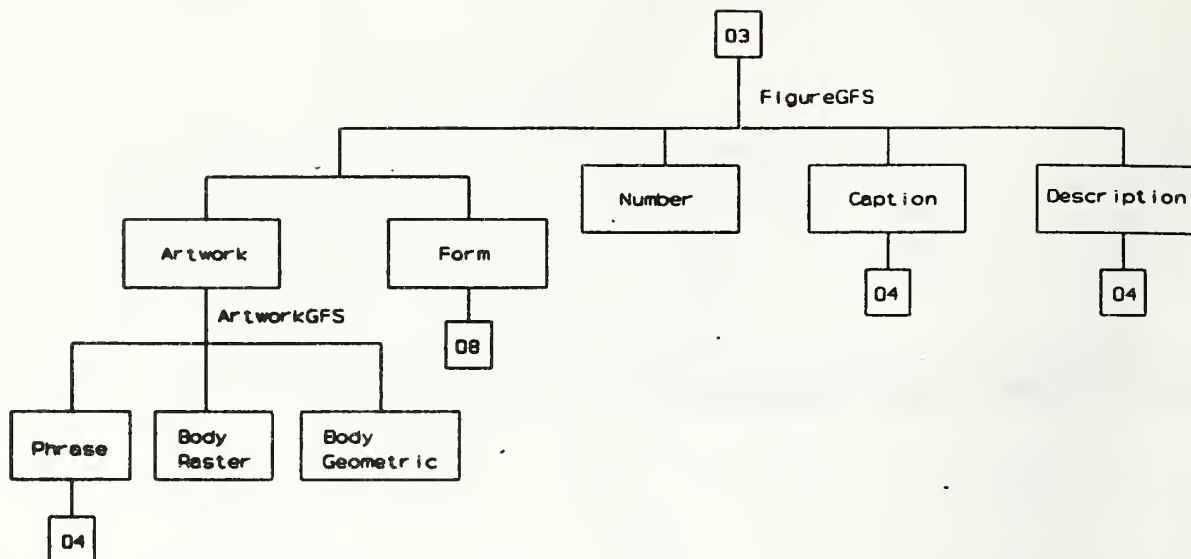


Figure 26 — Figure

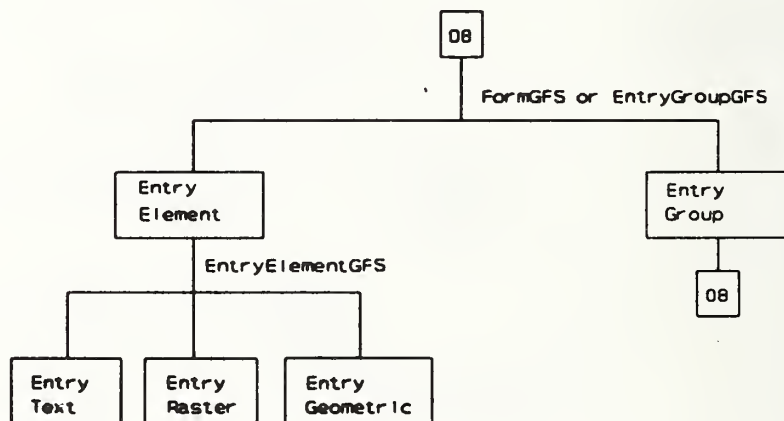


Figure 27 — Form

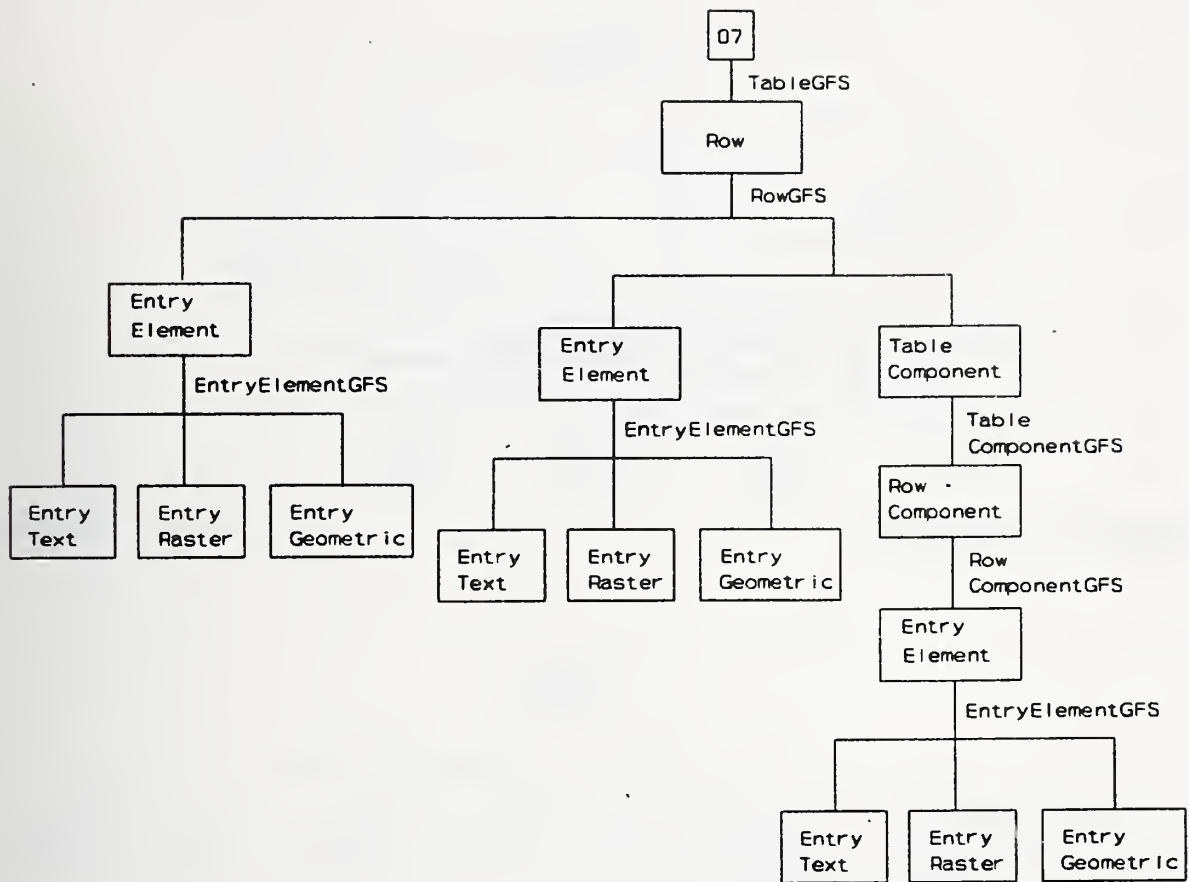


Figure 28 — Table

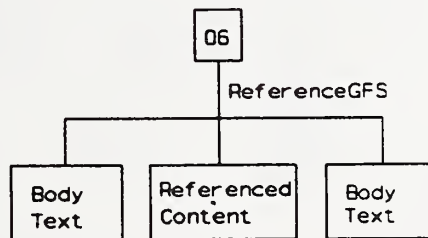


Figure 29 — Reference

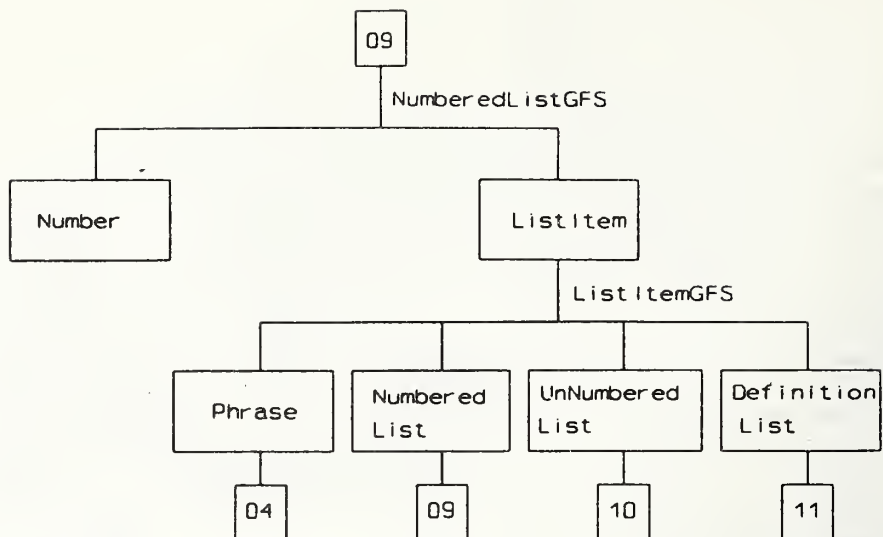


Figure 30 — NumberedList

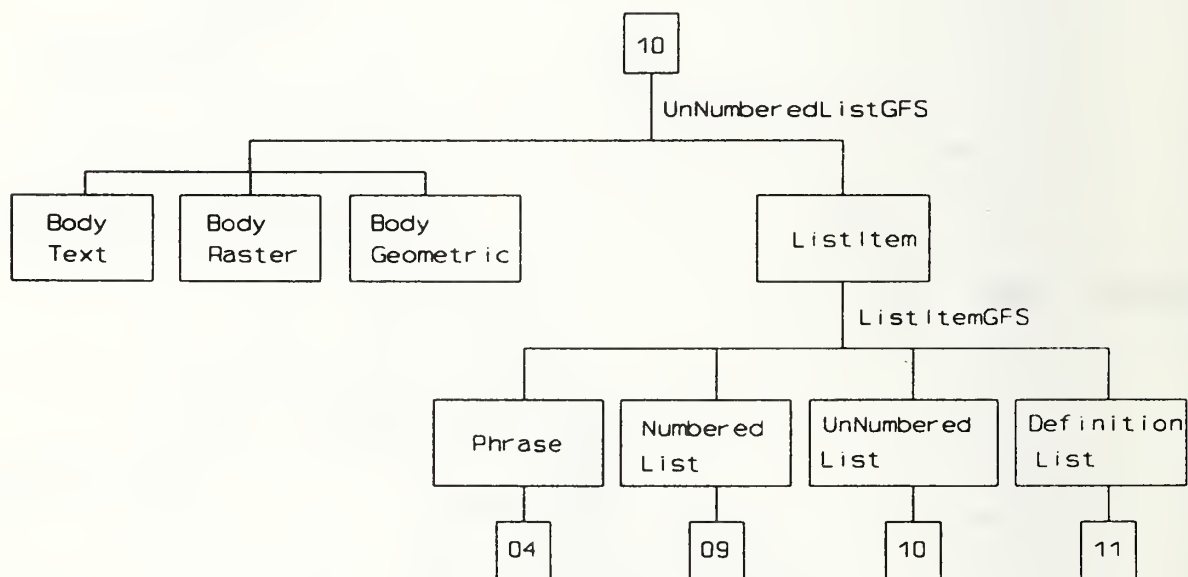


Figure 31 — UnNumberedList

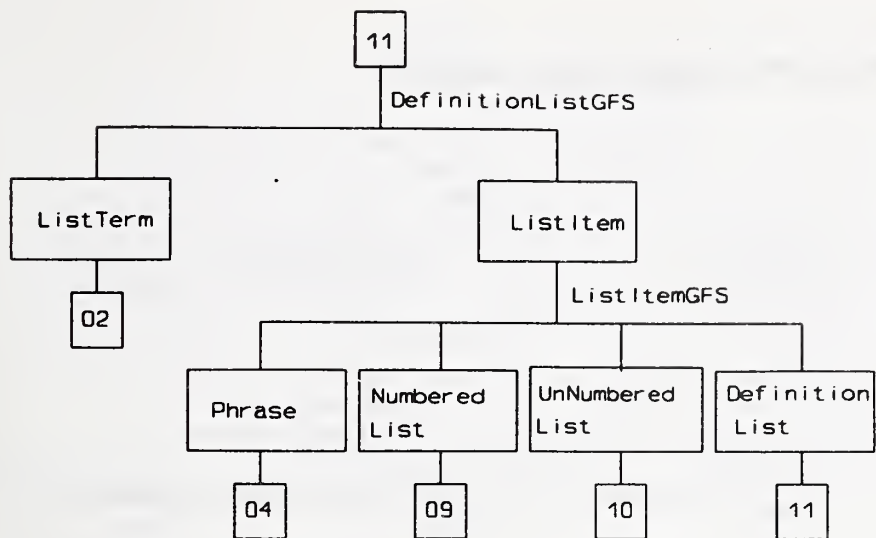


Figure 32 — DefinitionList

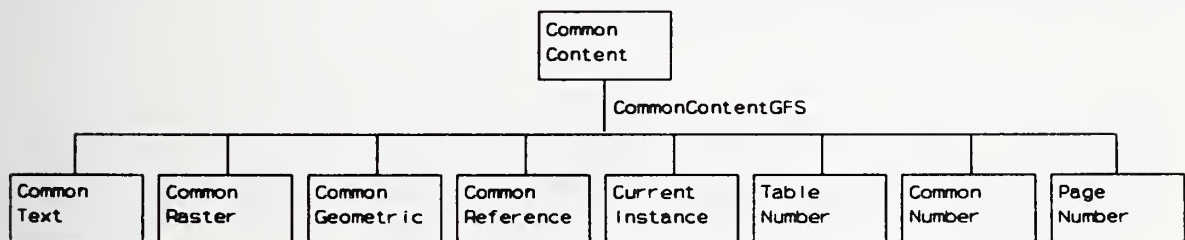


Figure 33 — CommonContent

7.1.2 Diagrams of relationships of layout constituents

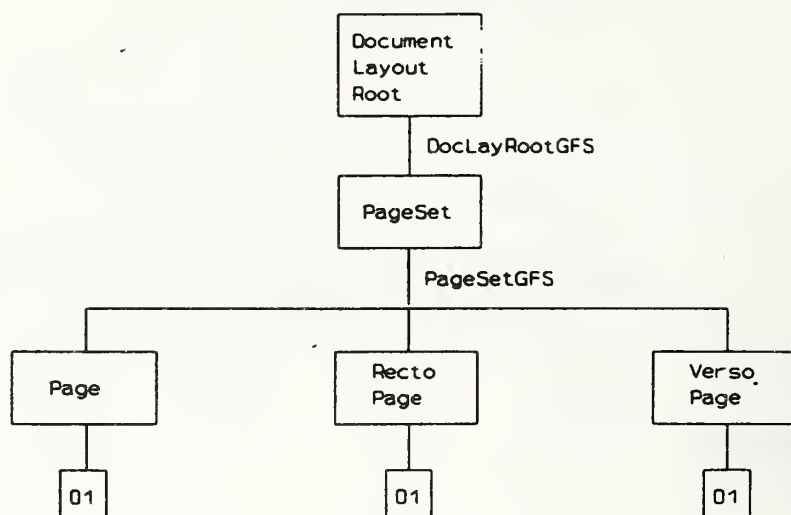


Figure 34 — DocumentLayoutRoot

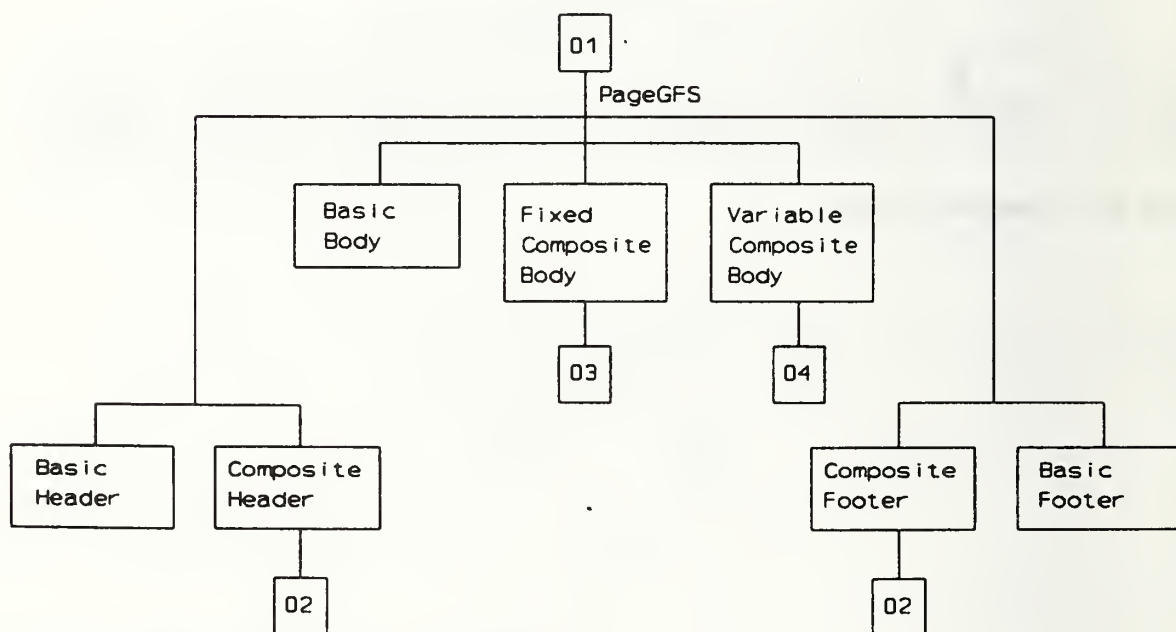


Figure 35 — Page, RectoPage and VersoPage

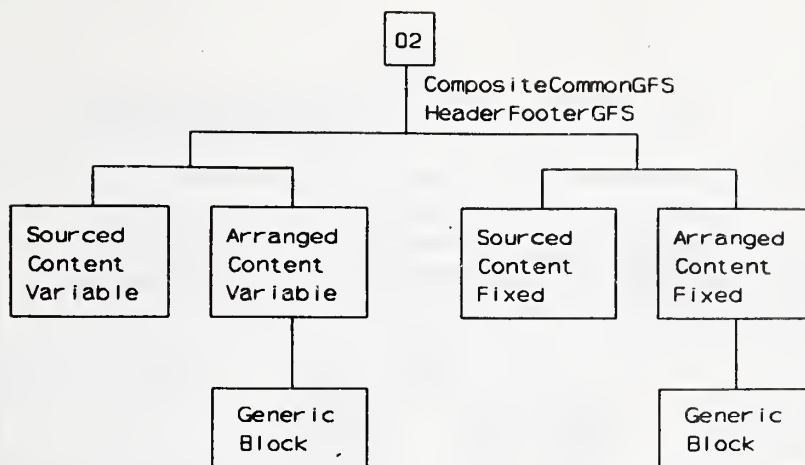


Figure 36 — CompositeHeader, CompositeFooter and CompositeCommon

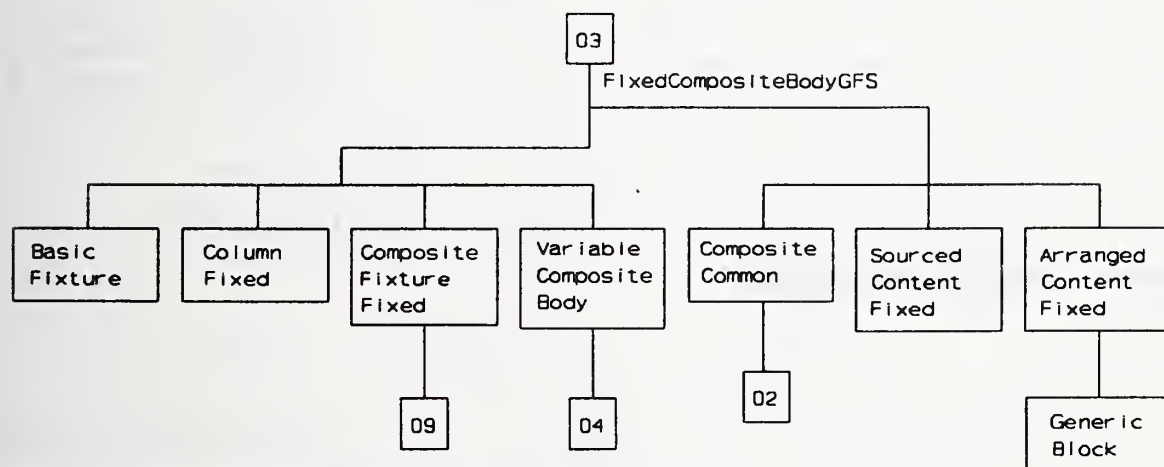


Figure 37 — FixedCompositeBody

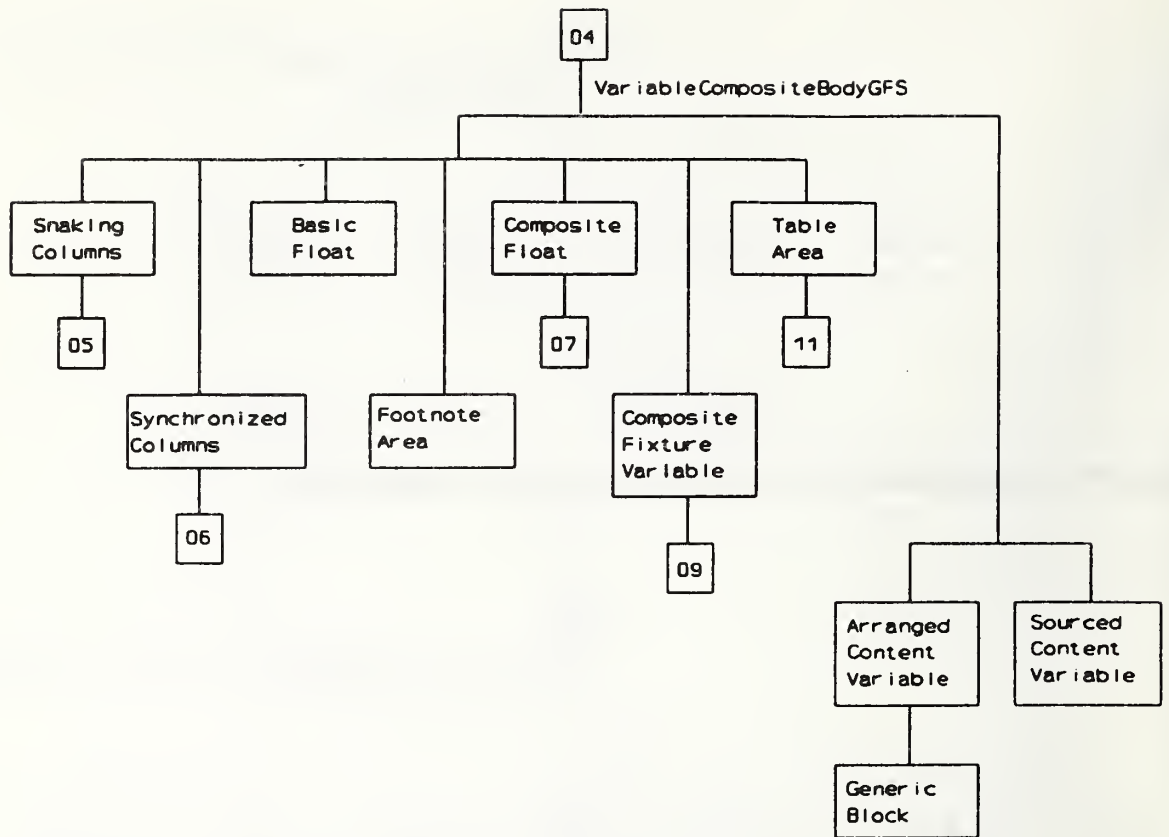


Figure 38 — VariableCompositeBody

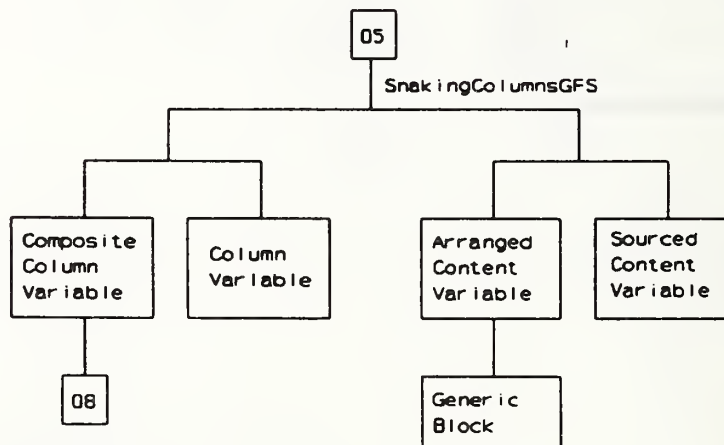


Figure 39 — SnakingColumns

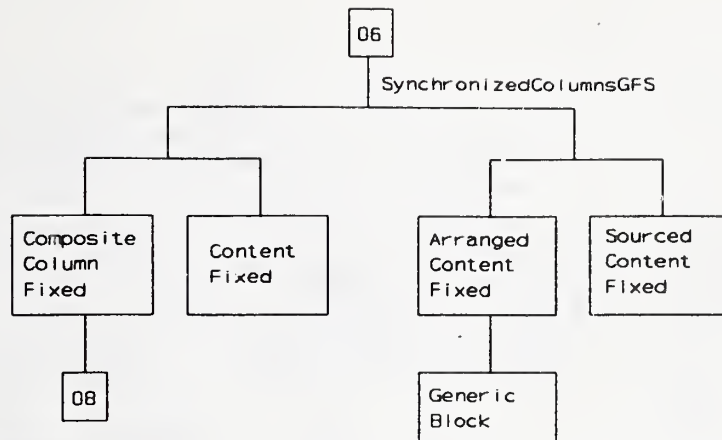


Figure 40 — SynchronizedColumns

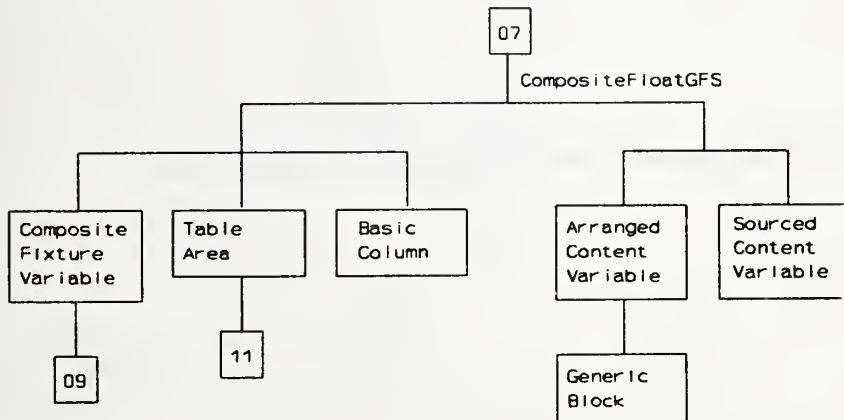


Figure 41 — CompositeFloat

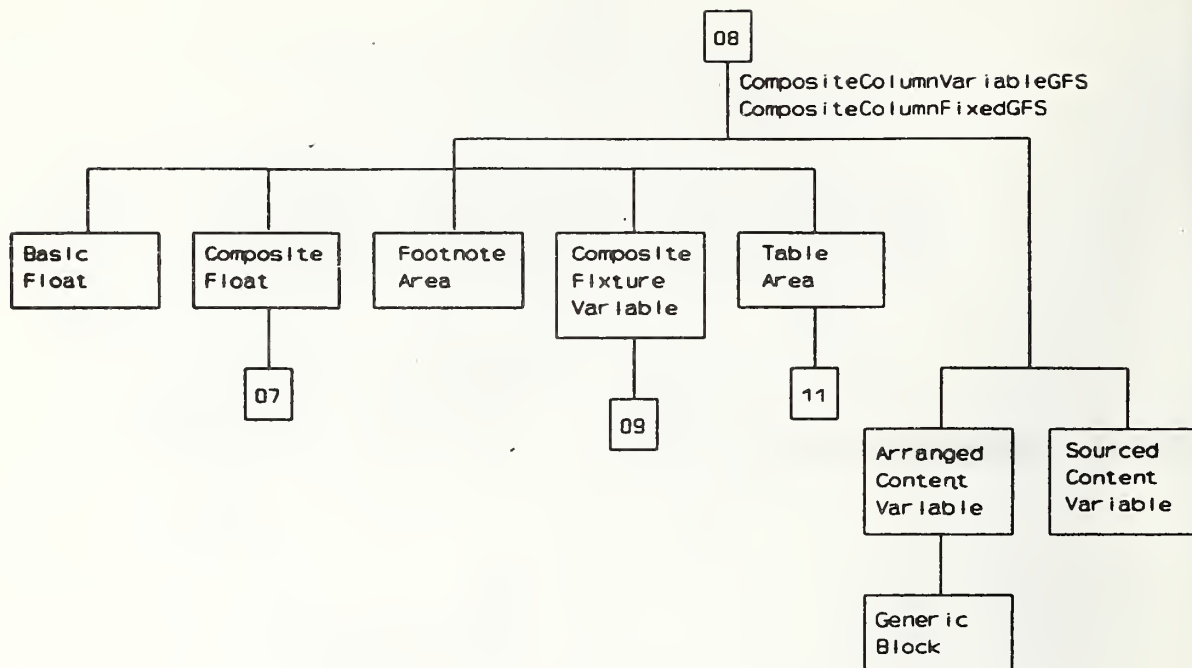


Figure 42 — CompositeColumnVariable and CompositeColumnFixed

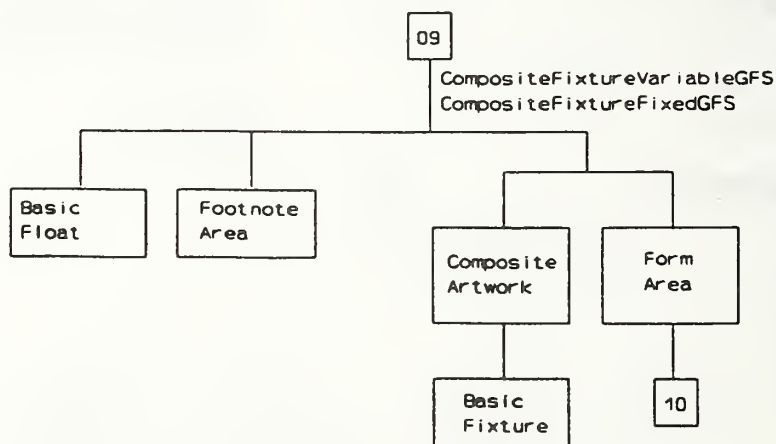


Figure 43 — CompositeFixtureVariable and CompositeFixtureFixed

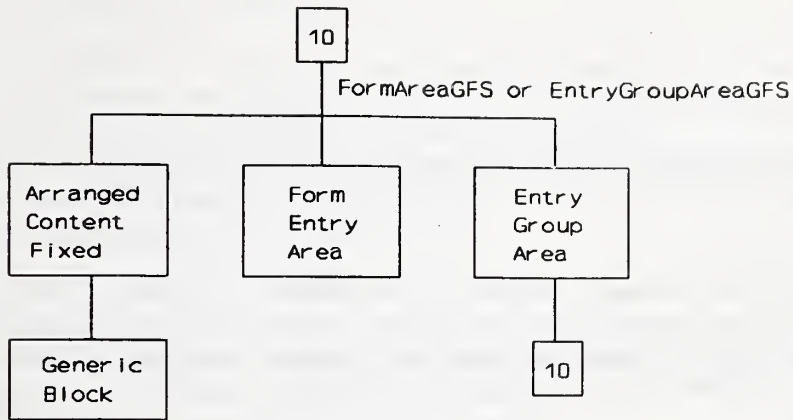


Figure 44 — FormArea and EntryGroup

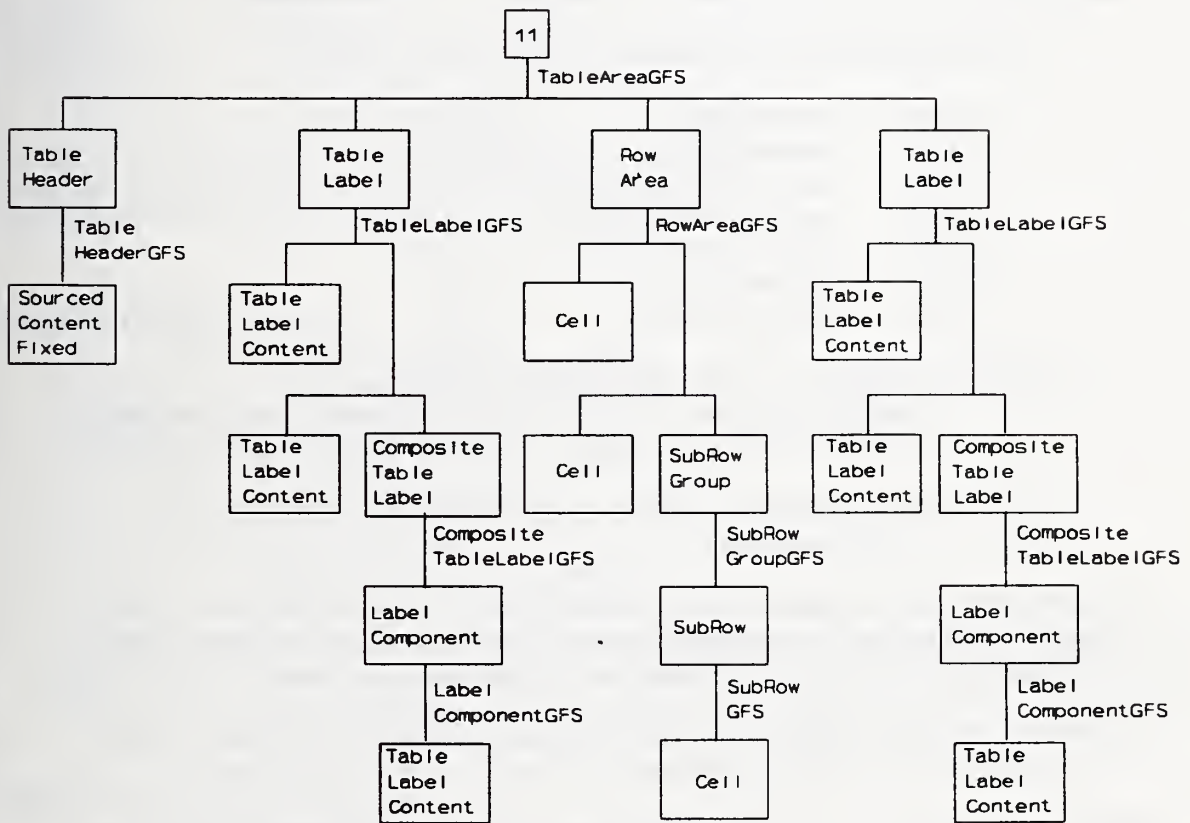


Figure 45 — TableArea

7.1.3 Notation

This clause is written in accordance with the Document Application Profile and Notation (DAPPN) of [CCITT Recommendation T.413 | ISO 8613-1], Annex F. The following clarifications and minor extensions apply:

- a) [Clarification] The value range definition for the attributes "subordinates" and "imaging-order" specify the set of object instances that may occur. The ordering and number (which may be zero) of object instances for the attribute "subordinates" shall be in accordance with the value of the attribute "generator for subordinates" in the respective object class.
- b) [Clarification] The value "ANY_STRING" can include code extension control functions as well as graphic characters.
- c) [Extension] In order to write the specification of the usage of character sets and code extension control functions precisely, the following extensions are applied:

1) Following symbols are introduced to denote shift functions:

Symbol	Shift function	Coded representation
LS0	Locking shift zero	00/15
LS1R	Locking shift one right	ESC 07/14
LS2R	Locking shift two right	ESC 07/13
LS3R	Locking shift three right	ESC 07/12
SS2	Single shift two	08/14
SS3	Single shift three	08/15

2) <escape-sequence> is extended to include shift functions:

<escape-sequence> ::= 'ESC' <octet>... [<invocation-control-function>];
<invocation-control-function> ::= 'LS0'|'LS1R'|'LS2R'|'LS3R'|'SS2'|'SS3';

3) Data type specification for #ESC in content information is extended as:

<escape-sequence>...

- d) [Clarification] When an attribute value is specified by a set of production rules, a non-terminating symbol which occurs first is its start symbol. Note that start symbols other than <object-id-expr>, <string-expr> and <construction-expr> are used.
- e) [Extension] Data type specifications other than those specified in the tables in DAPPN are applied for some attributes within the range that the base standard permits.
- f) [Extension] "|" is used in CASE SUPERIOR expressions in the following format in order to shorten the text:
CASE SUPERIOR ({const1|const2|...|constn}(aaaa)) OF {.....}
where const1, const2, ... , constn are names of constituent constraint, aaaa is a name of attribute.

This expression is syntactically equivalent to the following expression:

CASE SUPERIOR (const1(aaaa)) OF {.....}
CASE SUPERIOR (const2(aaaa)) OF {.....}

....

CASE SUPERIOR (constn(aaaa)) OF {.....}

When CASE SUPERIOR is evaluated, constituents are searched from the immediate superior to the root. Only the first one which satisfies one of the constituent constraints const1, const2, ..., and constn is selected and the attribute aaaa in it is tested.

7.2 Document profile constituent constraints

7.2.1 Macro definitions

```
DEFINE(FC, "ASN.1{2 8 2 6 0}" -- formatted character content --)
DEFINE(PC, "ASN.1{2 8 2 6 1}" -- processable character content --)
DEFINE(FPC, "ASN.1{2 8 2 6 2}" -- formatted processable character content --)
DEFINE(FPR, "ASN.1{2 8 2 7 2}" -- formatted processable raster graphics content --)
DEFINE(FPG, "ASN.1{2 8 2 8 0}" -- formatted processable geometric graphics content --)
```

```
DEFINE(FDA,      "{formatted}")
DEFINE(PDA,      "{processable}")
DEFINE(FPDA,     "{formatted-processable}")
DEFINE(PDA-FPDA, "{processable}'formatted-processable}")
```

```
DEFINE(DAC, "DocumentProfile (Document-architecture-class)")
DEFINE(GLAS, "DocumentProfile (Generic-layout-structure)")
DEFINE(COMPLETE, "{complete-generator-set}")
```

```
DEFINE(BasicPageDimensions, "
    REQ #horizontal-dimension
        {REQ #fixed-dimension{<=9240}},
    REQ #vertical-dimension
        {REQ #fixed-dimension{<=12400}}
| REQ #horizontal-dimension
    {REQ #fixed-dimension{<=12400}},
    REQ #vertical-dimension
        {REQ #fixed-dimension{<=9240}} ")
```

-- Any size equal to or smaller than CARA (Common Assured Reproduction Area) of ISO A4 and ANSI-A. Both Portrait and Landscape may be specified. Note that the above macro is defined for clarification of the specification and is not used in any other part of this DAP specification. --

```
DEFINE(NonBasicPageDimensions, "
    REQ #horizontal-dimension
        {REQ #fixed-dimension{<=39680}},
    REQ #vertical-dimension
        {REQ #fixed-dimension {12401..56120}}
|REQ #horizontal-dimension
    {REQ #fixed-dimension{9241..39680}},
    REQ #vertical-dimension
        {REQ #fixed-dimension {<=56120}}
        -- up to ISO A0 portrait --
|REQ #horizontal-dimension
```

```

    {REQ #fixed-dimension {<=56120}},
REQ #vertical-dimension
    {REQ #fixed-dimension {9241..39680}}
|REQ #horizontal-dimension
    {REQ #fixed-dimension {12401..56120}},
REQ #vertical-dimension
    {REQ #fixed-dimension {<=39680}}
    -- up to ISO A0 landscape --
|REQ #horizontal-dimension
    {REQ #fixed-dimension {<=40800}},
REQ #vertical-dimension
    {REQ #fixed-dimension {12401..52800}}
|REQ #horizontal-dimension
    {REQ #fixed-dimension {9241..40800}},
REQ #vertical-dimension
    {REQ #fixed-dimension {<=52800}}
    -- up to ANSI-E portrait --
|REQ #horizontal-dimension
    {REQ #fixed-dimension {<=52800}},
REQ #vertical-dimension
    {REQ #fixed-dimension {9241..40800}}
|REQ #horizontal-dimension
    {REQ #fixed-dimension {12401..52800}},
REQ #vertical-dimension
    {REQ #fixed-dimension {<=40800}}
    -- up to ANSI-E landscape --
|REQ #horizontal-dimension
    {REQ #fixed-dimension {<=12141}},
REQ #vertical-dimension
    {REQ #fixed-dimension {12401..17196}}
|REQ #horizontal-dimension
    {REQ #fixed-dimension {9241..12141}},
REQ #vertical-dimension
    {REQ #fixed-dimension {<=17196}}
    -- up to JIS B4(Japanese legal) portrait --
|REQ #horizontal-dimension
    {REQ #fixed-dimension {<=17196}},
REQ #vertical-dimension
    {REQ #fixed-dimension {9241..12141}}
|REQ #horizontal-dimension
    {REQ #fixed-dimension {12401..17196}},
REQ #vertical-dimension
    {REQ #fixed-dimension {<=12141}}
    -- up to JIS B4(Japanese legal) landscape --

```

")


```

DEFINE(PermissiblePageDimensions, "
    REQ #horizontal-dimension
        {REQ #fixed-dimension {<=39680}},
    REQ #vertical-dimension
        {REQ #fixed-dimension {<=56120}} -- up to ISO A0 portrait --
|REQ #horizontal-dimension
        {REQ #fixed-dimension {<=56120}},
    REQ #vertical-dimension
        {REQ #fixed-dimension {<=39680}} -- up to ISO A0 landscape --
|REQ #horizontal-dimension
        {REQ #fixed-dimension {<=40800}},
    REQ #vertical-dimension
        {REQ #fixed-dimension {<=52800}} -- up to ANSI-E portrait --
|REQ #horizontal-dimension
        {REQ #fixed-dimension {<=52800}},
    REQ #vertical-dimension
        {REQ #fixed-dimension {<=40800}} -- up to ANSI-E landscape --
")

```

```

DEFINE(NominalPageSizes, "
    REQ #horizontal-dimension {7015},    REQ #vertical-dimension {9920}
        -- ISO A5 portrait --
| REQ #horizontal-dimension {9920},    REQ #vertical-dimension {7015}
        -- ISO A5 landscape--
| REQ #horizontal-dimension {9920},    REQ #vertical-dimension {14030}
        -- ISO A4 portrait --
| REQ #horizontal-dimension {14030},    REQ #vertical-dimension {9920}
        -- ISO A4 landscape--
| REQ #horizontal-dimension {14030},    REQ #vertical-dimension {19840}
        -- ISO A3 portrait --
| REQ #horizontal-dimension {19840},    REQ #vertical-dimension {14030}
        -- ISO A3 landscape --
| REQ #horizontal-dimension {19840},    REQ #vertical-dimension {28060}
        -- ISO A2 portrait --
| REQ #horizontal-dimension {28060},    REQ #vertical-dimension {19840}
        -- ISO A2 landscape --
| REQ #horizontal-dimension {28060},    REQ #vertical-dimension {39680}
        -- ISO A1 portrait --
| REQ #horizontal-dimension {39680},    REQ #vertical-dimension {28060}
        -- ISO A1 landscape --
| REQ #horizontal-dimension {39680},    REQ #vertical-dimension {56120}
        -- ISO A0 portrait --
| REQ #horizontal-dimension {56120},    REQ #vertical-dimension {39680}
        -- ISO A0 landscape --
| REQ #horizontal-dimension {10200},    REQ #vertical-dimension {16800}
        -- ANSI legal portrait --
| REQ #horizontal-dimension {16800},    REQ #vertical-dimension {10200}
        -- ANSI legal landscape --
| REQ #horizontal-dimension {10200},    REQ #vertical-dimension {13200}
        -- ANSI-A portrait --
| REQ #horizontal-dimension {13200},    REQ #vertical-dimension {10200}
        -- ANSI-A landscape --
| REQ #horizontal-dimension {13200},    REQ #vertical-dimension {20400}
        -- ANSI-B portrait --

```

```

| REQ #horizontal-dimension {20400}, REQ #vertical-dimension {13200}
-- ANSI-B landscape --
| REQ #horizontal-dimension {20400}, REQ #vertical-dimension {26400}
-- ANSI-C portrait --
| REQ #horizontal-dimension {26400}, REQ #vertical-dimension {20400}
-- ANSI-C landscape --
| REQ #horizontal-dimension {26400}, REQ #vertical-dimension {40800}
-- ANSI-D portrait --
| REQ #horizontal-dimension {40800}, REQ #vertical-dimension {26400}
-- ANSI-D landscape --
| REQ #horizontal-dimension {40800}, REQ #vertical-dimension {52800}
-- ANSI-E portrait --
| REQ #horizontal-dimension {52800}, REQ #vertical-dimension {40800}
-- ANSI-E landscape --
| REQ #horizontal-dimension {33600}, REQ #vertical-dimension {48000}
-- ANSI-F portrait --
| REQ #horizontal-dimension {48000}, REQ #vertical-dimension {33600}
-- ANSI-F landscape --
| REQ #horizontal-dimension {12141}, REQ #vertical-dimension {17196}
-- JIS B4 (Japanese legal) portrait --
| REQ #horizontal-dimension {17196}, REQ #vertical-dimension {12141}
-- JIS B4 (Japanese legal) landscape --
| REQ #horizontal-dimension {8598}, REQ #vertical-dimension {12141}
-- JIS B5 (Japanese letter) portrait --
| REQ #horizontal-dimension {12141}, REQ #vertical-dimension {8598}
-- JIS B5 (Japanese letter) landscape--

```

")

DEFINE(GRAPHICRENDITIONS, "

```

{'cancel'|'increased-intensity' |'decreased-intensity'
|'italicised' |'underlined' |'slowly-blinking'
|'rapidly-blinking'|'negative-image' |'crossed-out'
|'primary-font'|'first-alternative-font'
|'second-alternative-font' |'third-alternative-font'
|'fourth-alternative-font' |'fifth-alternative-font'
|'sixth-alternative-font'|'seventh-alternative-font'
|'eighth-alternative-font' |'ninth-alternative-font'
|'doubly-underlined' | 'normal-intensity' |'not-italicised'
|'not-underlined'|'steady'|'positive-image'
|'not-crossed-out' }... ")

```

-- Macro defining permissible code extension announcers. Note that all the values are basic. --

```

DEFINE(CDTEXTEN, "ESC 02/00 05/00, -- Use G0 & LS0 --
[ESC 02/00 05/03] , -- Use G1 & LS1R --
[ESC 02/00 05/05] , -- Use G2 & LS2R --
[ESC 02/00 05/07] , -- Use G3 & LS3R --
[ESC 02/00 05/10] , -- Use G2 & SS2 --
[ESC 02/00 05/11] -- Use G3 & SS3 --

```

")

-- Macro defining code extension announcers for DAP defaults --

DEFINE(DAP-DEFAULT-CDEXTEN, "\$CDEXTEN")

-- Macros defining final character for designation --

DEFINE(FCORE, "04/02 -- A final character designating ISO-IR 6 (the IRV of ISO 646 revised 1991, i.e ASCII) --")

DEFINE(F646, " -- A final character designating any version of ISO 646 except ISO-IR 6 --")

DEFINE(F94S, " -- A final character designating any registered 94 single byte graphic character set, optionally preceded by one or more intermediate characters as define in Annex C of ISO 2022 --")

DEFINE(F94M, " -- A final character designating any registered 94 multi byte graphic character set, optionally preceded by one or more intermediate characters as define in Annex C of ISO 2022 --")

DEFINE(F96S, " -- A final character designating any registered 96 single byte graphic character set, optionally preceded by one or more intermediate characters as define in Annex C of ISO 2022 --")

DEFINE(F96M, " -- A final character designating any registered 96 multi byte graphic character set, optionally preceded by one or more intermediate characters as define in Annex C of ISO 2022 --")

DEFINE(FEMPTY, "07/14 -- The empty set --")

-- Macro defining a revision number of a character set --

DEFINE(REV, "-- An octet between 04/00 and 07/14, which represents a revision number as defined in ISO 2022. --")

-- Macros defining designation sequences --

DEFINE(DEG-CORE-G0, "ESC 02/08 \$FCORE")
-- Designate 94 characters of ISO-IR 6 (the IRV of ISO 646 revised 1991) to G0 --

DEFINE(DEG-646-G0, "ESC 02/08 \$F646")
-- Designate any version of ISO 646, except ISO-IR 6, to G0 --

DEFINE(DEG-ANY-G1, "[ESC 02/06 \$REV
{ ESC 02/09 \$F94S
| ESC 02/04 02/09 \$F94M
| ESC 02/13 \$F96S
| ESC 02/04 02/13 \$F96M}])")
-- Designate any character set to G1 --

DEFINE(DEG-ANY-G2, "[ESC 02/06 \$REV
{ ESC 02/10 \$F94S
| ESC 02/04 02/10 \$F94M
| ESC 02/14 \$F96S
| ESC 02/04 02/14 \$F96M}])")

-- Designate any character set to G2 --

```
DEFINE(DEG-ANY-G3,      "[[ESC 02/06 $REV]
                        { ESC 02/11 $F94S
                        | ESC 02/04 02/11 $F94M
                        | ESC 02/15 $F96S
                        | ESC 02/04 02/15 $F96M}}]")
```

-- Designate any character set to G3 --

```
DEFINE(DEG-EMPTY-G1,    "ESC 02/09 $FEMPTY")
```

-- Designate the empty set to G1 --

-- Macro defining permissible graphic character sets. --

```
DEFINE(PERMIT-GRCHAR, "[$DEG-CORE-G0 LS0 |$DEG-646-G0 LS0],
                      {[$DEG-ANY-G1 LS1R
                      |$DEG-ANY-G2 LS2R
                      |$DEG-ANY-G3 LS3R]...
                      |$DEG-EMPTY-G1 LS1R}")
```

-- Macro defining graphic character sets for DAP defaults --

```
DEFINE(DAP-DEFAULT-GRCHAR, "$PERMIT-GRCHAR")
```

-- Macro defining basic graphic character sets. Note that this macro is defined for clarification of the specification and is not used in any other part of this DAP specification. --

```
DEFINE(BASIC-GRCHAR,    "$DEG-CORE-G0 LS0,
                        $DEG-EMPTY-G1 LS1R")
```

-- Macro defining non-basic graphic character sets --

```
DEFINE(NON-BASIC-GRCHAR, "{[$DEG-646-G0
                          |$DEG-ANY-G1
                          |$DEG-ANY-G2
                          |$DEG-ANY-G3]...}")
```

-- Macro defining character sets used in document profile attributes --

```
DEFINE(PROFCHAR, "
    ESC 02/00 04/03                                -- announcement to use G0 and G1, and to invoke into
                                                    GL and GR respectively. (no shift functions are
                                                    necessary)--
```

```
    {$DEG-CORE-G0 | $DEG-646-G0 }    -- designate G0 --
    {$DEG-ANY-G1  | $DEG-EMPTY-G1 }  -- designate G1 --
```

")

Macro defining comments character sets --

EFINE(COMCHAR, "

-- in the case to use both GL and GR without shift functions --

ESC 02/00 04/03

-- announcement to use G0 and G1, and to invoke into
GL and GR respectively. (no shift functions are
necessary)--

{ \$DEG-CORE-G0 | \$DEG-646-G0 } -- designate G0 --

{ \$DEG-ANY-G1 | \$DEG-EMPTY-G1 } -- designate G1 --

| -- in the case to use various character sets (shift functions are necessary) --

{ESC 02/00 05/00, -- announcement to use G0 and LS0 --

[ESC 02/00 05/03], -- announcement to use G1 and LS1R --

[ESC 02/00 05/05], -- announcement to use G2 and LS2R --

[ESC 02/00 05/07], -- announcement to use G3 and LS3R --

[ESC 02/00 05/10], -- announcement to use G2 and SS2 --

[ESC 02/00 05/11] } -- announcement to use G3 and SS3 --

{ \$DEG-CORE-G0 | \$DEG-646-G0 } -- designate G0 --

{{ \$DEG-ANY-G1 -- designate G1 --

| \$DEG-ANY-G2 -- designate G2 --

| \$DEG-ANY-G3}... -- designate G3 --

| \$DEG-EMPTY-G1}

")

-- Macro defining character sets used for alternative representation --

DEFINE(ALTCHAR, "\$PROFCHAR")

7.2.2 Constituent constraints

7.2.2.1 DocumentProfile

{
CASE \$DAC OF {

\$FDA: PERM Generic-layout-structure {'factor-set'},
PERM Specific-layout-structure {'present'},
-- must be present in the case of complete document and not present
in the case of generic document --
PERM Presentation-styles {'present'}

\$PDA: PERM Generic-layout-structure {'complete-generator-set'},
PERM Generic-logical-structure {'complete-generator-set'
| 'partial-generator-set'},
-- must be present if there is no external document class reference --
PERM Specific-logical-structure {'present'},
-- must be present in case of complete document and not present
in the case of generic document --
PERM Presentation-styles {'present'},

```

        PERM Layout-styles                                {'present'}

$FPDA: PERM Generic-layout-structure                    {'complete-generator-set'},
      -- must be present if there is no external document class reference --
      PERM Specific-layout-structure                    {'present'},
      -- must be present in the case of complete document and not present
      in the case of generic document --
      PERM Generic-logical-structure                    {'complete-generator-set'
                                                         |'partial-generator-set'},
      -- must be present if there is no external document class reference --
      PERM Specific-logical-structure                    {'present'},
      -- must be present in the case of complete document and not present
      in the case of generic document --
      PERM Presentation-styles                          {'present'},
      PERM Layout-styles                                {'present'}
    },

    PERM External-document-class                        {ANY_VALUE},

    PERM Resource-document                             {ANY_VALUE},

    PERM Resources      {MUL {REQ #resource-identifier      {ANY_VALUE},
                              REQ #resource-object-class-identifier {ANY_VALUE}}}
    },

    -- document characteristics --

REQ Document-application-profile                        {-- See clause 8 for a definition of the permitted values
                                                         for this attribute --},

PERM Document-application-profile-defaults {

    CASE $DAC OF {
        $FDA : {PERM #content-architecture-class    {$FC|$FPC}}
        $PDA : {PERM #content-architecture-class    {$FC|$PC|$FPC}}
        $FPDA : {PERM #content-architecture-class   {$FC|$FPC}}
    },

    PERM #dimensions      {$PermissiblePageDimensions},

    PERM #medium-type      {PERM #nominal-page-size {$NominalPageSizes},
                           PERM #side-of-sheet    {ANY_VALUE }},

    PERM #transparency     {ANY_VALUE},

    PERM #colour            {ANY_VALUE},

    PERM #layout-path       {ANY_VALUE},

    PERM #block-alignment   {ANY_VALUE},

    PERM #border            {ANY_VALUE},

```

PERM #page-position {ANY_VALUE},

PERM #type-of-coding

{ASN.1 {2 8 3 6 0} -- character encoding --
{ASN.1 {2 8 3 7 0} -- T.6 encoding --
{ASN.1 {2 8 3 7 1} -- T.4 one dimensional encoding --
{ASN.1 {2 8 3 7 2} -- T.4 two dimensional encoding --
{ASN.1 {2 8 3 7 3} -- bitmap encoding --
{ASN.1 {2 8 3 8 0} -- geometric encoding --},

PERM #character-content-defaults {

PERM #alignment {ANY_VALUE},
PERM #character-fonts {ANY_VALUE},
PERM #character-path {ANY_VALUE},
PERM #character-spacing {ANY_VALUE},
PERM #character-orientation {ANY_VALUE},
PERM #code-extension-announcers {\$DAP-DEFAULT-CDEXTEN},
PERM #first-line-offset {ANY_VALUE},
PERM #graphic-character-sets {\$DAP-DEFAULT-GRCHAR},
PERM #graphic-character-subrepertoire {ANY_VALUE},
PERM #graphic-rendition {\$GRAPHICRENDITIONS},
PERM #itemisation {ANY_VALUE},
PERM #kerning-offset {ANY_VALUE},
PERM #line-layout-table {ANY_VALUE},
PERM #line-progression {ANY_VALUE},
PERM #line-spacing {ANY_VALUE},
PERM #pairwise-kerning {ANY_VALUE},
PERM #indentation {ANY_VALUE},
PERM #orphan-size {ANY_VALUE},
PERM #proportional-line-spacing {ANY_VALUE},
PERM #widow-size {ANY_VALUE},
PERM #formatting-indicator {ANY_VALUE},
PERM #initial-offset {ANY_VALUE}

},

PERM #raster-graphics-content-defaults {

PERM #pel-path {ANY_VALUE},
PERM #line-progression {ANY_VALUE},
PERM #clipping {ANY_VALUE},
PERM #image-dimensions {ANY_VALUE},
PERM #pel-spacing {ANY_VALUE},
PERM #spacing-ratio {ANY_VALUE},
PERM #compression {ANY_VALUE}

},

PERM #geometric-graphics-content-defaults {ANY_VALUE}

},

REQ Document-architecture-class {\$FDA| \$PDA |\$FPDA},

REQ Content-architecture-classes {[\$FC],[\$PC],[\$FPC],[\$FPR],[\$FPG]},

REQ Interchange-format-class {-- See clause 8 for the definition of the permitted values for this attribute. --},

REQ Oda-version {REQ #standard-or-recommendation {"ISO 8613"},

-- non basic document characteristics --

PERM Profile-character-sets (\$PROFCHAR).

PERM Comments-character-sets (\$COMCHAR).

PERM Alternative-representation-character-sets (\$ALTCHAR).

PERM Page-dimensions {PMUL{\$NonBasicPageDimensions}}.

PERM Medium-types {PMUL {PERM #nominal-page-size {\$NominalPageSizes},
PERM #side-of-sheet {'recto' | 'verso'}}}.

-- All values of "medium type" are non-basic --

PERM Layout-paths {'0-degrees' | '90-degrees' | '180-degrees'...}.

-- These values need not to be declared, they may be specified here for upwards compatibility from FOD26 | PM-26 --

PERM Borders (ANY_VALUE).

-- Any values need not to be declared, they may be specified here for upwards compatibility from FOD26 | PM-26 --

PERM Coding-attributes {
PERM #raster-graphics-coding-attributes {
PERM #compression {'uncompressed'}
}
}.

PERM Presentation-features {

PERM #character-presentation-features {

PERM #character-orientation {'90-degrees'},

-- This value needs not to be declared, they may be specified here for upwards compatibility from FOD26 | PM-26 --

PMUL {PERM #character-path {'90-degrees' | '180-degrees' | '270-degrees'}}.

-- These values need not to be declared, they may be specified here for upwards compatibility from FOD26 | PM-26 --

PMUL {PERM #character-spacing {<100 | 100 | 160 | 200}}.

-- These values need not to be declared, they may be specified here for upwards compatibility from FOD26 | PM-26 --

PMUL {PERM #graphic-rendition {'crossed-out' | 'not-crossed-out'}}.

-- These values need not to be declared, they may be specified here for upwards compatibility from FOD26 | PM-26 --

PMUL {PERM #line-spacing (ANY_VALUE) EXCEPT{200, 300, 400}}.

-- Values 100, 150 and 600 need not to be declared, they may be specified here for upwards compatibility from FOD26 | PM-26 --

PERM #line-progression {'90-degrees'}.

-- This value needs not to be declared, they may be specified here for upwards compatibility from FOD26 | PM-26 --

PMUL {PERM #graphic-character-sets {\$NON-BASIC-GRCHAR}},
PMUL {PERM #graphic-character-subrepertoire {ANY_VALUE}}

},

PERM #raster-graphics-presentation-features {
PMUL {PERM #pel-path {'90-degrees'|'180-degrees'|'270-degrees'}},
PERM #line-progression {'90-degrees'},
PMUL {PERM #pel-spacing {ANY_VALUE}EXCEPT{16, 12, 8, 6, 5, 4, 3, 2, 1}},
-- Any value of #pel spaces is permitted as basic --
-- Basic values of #length are multiples of #pel spaces as listed --

PMUL {PERM #spacing-ratio
{REQ #line-spacing-value {ANY_VALUE} EXCEPT {1},
REQ #pel-spacing-value {ANY_VALUE} EXCEPT {1}}}

}

},

-- additional document characteristics --

PERM Unit-scaling {REQ #unit-scaling-m {ANY_INTEGER},
REQ #unit-scaling-n {ANY_INTEGER}},

PERM Fonts-list {PMUL {REQ #font-identifier {ANY_VALUE},
REQ #font-reference {ANY_VALUE}}}

},

-- document management attributes --

-- document description --

PERM Title {ANY_STRING},
PERM Subject {ANY_STRING},
PERM Document-type {ANY_STRING},
PERM Abstract {ANY_STRING},
PERM Keywords {ANY_STRING...},
REQ Document-reference {ANY_VALUE},

-- dates and times --

PERM Document-date-and-time {ANY_STRING},
PERM Creation-date-and-time {ANY_STRING},
PERM Local-filing-date-and-time {ANY_VALUE},
PERM Expiry-date-and-time {ANY_STRING},
PERM Start-date-and-time {ANY_STRING},
PERM Purge-date-and-time {ANY_STRING},
PERM Release-date-and-time {ANY_STRING},
PERM Revision-history {ANY_VALUE},

-- originators --

PERM Organizations {ANY_STRING...},
PERM Preparers {ANY_VALUE},
PERM Owners {ANY_VALUE},
PERM Authors {ANY_VALUE},

```

-- other user information --
PERM Copyright {ANY_VALUE},
PERM Status {ANY_STRING},
PERM User-specific-codes {ANY_STRING...},
PERM Distribution-list {ANY_VALUE},
PERM Additional-information {ANY_VALUE},

-- external references --
PERM References-to-other-documents {ANY_VALUE},
PERM Superseded-documents {ANY_VALUE},

-- local file references --
PERM Local-file-references {ANY_VALUE},

-- content attributes --
PERM Document-size {ANY_INTEGER},
PERM Number-of-pages {ANY_INTEGER},
PERM Languages {ANY_STRING...},

-- security information --
PERM Authorization {ANY_VALUE},
PERM Security-classification {ANY_STRING},
PERM Access-rights {ANY_STRING...}
}

```

7.3 Logical constituent constraints

7.3.1 Macro definitions

-- Defines any logical objects in a specific logical structure. --

```

DEFINE(LogicalObjects, "
<logical-objects> ::=
    OBJECT_ID_OF(DocumentLogicalRoot)
    | OBJECT_ID_OF(Passage)
    | OBJECT_ID_OF(NumberedSegment)
    | OBJECT_ID_OF(Number)
    | OBJECT_ID_OF(Title)
    | OBJECT_ID_OF(Caption)
    | OBJECT_ID_OF(Paragraph)
    | OBJECT_ID_OF(Phrase)
    | OBJECT_ID_OF(Footnote)
    | OBJECT_ID_OF(FootnoteNumber)
    | OBJECT_ID_OF(FootnoteReference)
    | OBJECT_ID_OF(FootnoteBody)
    | OBJECT_ID_OF(FootnoteText)
    | OBJECT_ID_OF(Figure)
    | OBJECT_ID_OF(BodyText)
    | OBJECT_ID_OF(Reference)
    | OBJECT_ID_OF(ReferencedContent)
    | OBJECT_ID_OF(BodyRaster)
    | OBJECT_ID_OF(BodyGeometric)
    | OBJECT_ID_OF(Description)
    | OBJECT_ID_OF(Artwork)

```

```

| OBJECT_ID_OF(NumberedList)
| OBJECT_ID_OF(UnNumberedList)
| OBJECT_ID_OF(DefinitionList)
| OBJECT_ID_OF(ListItem)
| OBJECT_ID_OF(ListTerm)
| OBJECT_ID_OF(Table)
| OBJECT_ID_OF(Row)
| OBJECT_ID_OF(TableComponent)
| OBJECT_ID_OF(RowComponent)
| OBJECT_ID_OF(Form)
| OBJECT_ID_OF(EntryElement)
| OBJECT_ID_OF(EntryGroup)
| OBJECT_ID_OF(EntryText)
| OBJECT_ID_OF(EntryRaster)
| OBJECT_ID_OF(EntryGeometric);

```

")

-- Defines any logical object classes other than classes referred by 'logical-source' --

```

DEFINE(LogicalObjectClasses, "
<logical-object-classes> ::=
    OBJECT_CLASS_ID_OF(DocumentLogicalRoot)
    | OBJECT_CLASS_ID_OF(Passage)
    | OBJECT_CLASS_ID_OF(NumberedSegment)
    | OBJECT_CLASS_ID_OF(Number)
    | OBJECT_CLASS_ID_OF>Title)
    | OBJECT_CLASS_ID_OF(Caption)
    | OBJECT_CLASS_ID_OF(Paragraph)
    | OBJECT_CLASS_ID_OF(Phrase)
    | OBJECT_CLASS_ID_OF(Footnote)
    | OBJECT_CLASS_ID_OF(FootnoteNumber)
    | OBJECT_CLASS_ID_OF(FootnoteReference)
    | OBJECT_CLASS_ID_OF(FootnoteBody)
    | OBJECT_CLASS_ID_OF(FootnoteText)
    | OBJECT_CLASS_ID_OF(Figure)
    | OBJECT_CLASS_ID_OF(BodyText)
    | OBJECT_CLASS_ID_OF(Reference)
    | OBJECT_CLASS_ID_OF(ReferencedContent)
    | OBJECT_CLASS_ID_OF(BodyRaster)
    | OBJECT_CLASS_ID_OF(BodyGeometric)
    | OBJECT_CLASS_ID_OF>Description)
    | OBJECT_CLASS_ID_OF(Artwork)
    | OBJECT_CLASS_ID_OF(NumberedList)
    | OBJECT_CLASS_ID_OF(UnNumberedList)
    | OBJECT_CLASS_ID_OF(DefinitionList)
    | OBJECT_CLASS_ID_OF(ListItem)
    | OBJECT_CLASS_ID_OF(ListTerm)
    | OBJECT_CLASS_ID_OF(Table)
    | OBJECT_CLASS_ID_OF(Row)
    | OBJECT_CLASS_ID_OF(TableComponent)
    | OBJECT_CLASS_ID_OF(RowComponent)
    | OBJECT_CLASS_ID_OF(Form)
    | OBJECT_CLASS_ID_OF(EntryElement)
    | OBJECT_CLASS_ID_OF(EntryGroup)
    | OBJECT_CLASS_ID_OF(EntryText)

```

```
| OBJECT_CLASS_ID_OF(EntryRaster)
| OBJECT_CLASS_ID_OF(EntryGeometric);
```

```
")
```

```
DEFINE(N, "<n>::={""0""1""2""3""4""5""6""7""8""9""...;}")
--      any string of characters from the set of characters: '0'...'9' --
```

```
--      Defines the prefix binding. This binding may be used to associate a string literal with an object
or object class. In addition, this binding is used to prefix text to another binding, such as a
segment number, figure number, table number, footnote number or page number. The
instances are differentiated by a suffix number. --
```

```
DEFINE(PREFIX, "
<prefix>      ::=      ""prefix-""<n>;
$N
")
```

```
--      Defines the suffix binding. This binding may be used to associate a string literal with an object
or object class. In addition, this binding is used to suffix text to another binding, such as a
segment number, figure number, table number, footnote number or page number. The
instances are differentiated by a suffix number. --
```

```
DEFINE(SUFFIX, "
<suffix>      ::=      ""suffix-""<n>;
$N
")
```

```
--      Defines the separator binding. This binding is used to provide a separator character for a
hierarchical form of a segment number, figure number, table number, footnote number or page
number. The instances are differentiated by a suffix number. --
```

```
DEFINE(SEPARATOR, "
<separator>   ::=      ""separator-""<n>;
$N
")
```

```
--      Defines the general number binding. This binding may be instanced for use as the numeric
value for use such as in segment number, figure number, table number, list number, footnote
number or page number bindings. The instances are differentiated by a suffix number. --
```

```
DEFINE(NUMBER, "
<number>      ::=      ""number-""<n>;
$N
")
```

```
--      Defines the general number string binding. This binding may be instanced for use as the string
value such as for segment number, figure number, table number, list number, footnote number
or page numbers. The instances are differentiated by a suffix number. --
```



```

DEFINE(NUMBERSTRING, "
<numberstring>      ::=      ""numberstring-""<n>;
$N
")

```

-- Defines the general string binding. The instances are differentiated by a suffix number. --

```

DEFINE(STRING, "
<string>           ::=      ""string-""<n>;
$N
")

```

-- Defines the names for footnote categories. The instances are differentiated by a suffix number. --

```

DEFINE(FOOTNOTECATEGORY, "
<footnotecategory>  ::=      ""Footnote""[""-""<n>];
$N
")

```

```

DEFINE(INITIALISEBINDINGS, "
    REQ #binding-name      {$PREFIX},
    REQ #binding-value     {ANY_STRING}
|REQ #binding-name      {$SUFFIX},
    REQ #binding-value     {ANY_STRING}
|REQ #binding-name      {$SEPARATOR},
    REQ #binding-value     {ANY_STRING}
|REQ #binding-name      {$NUMBER},
    REQ #binding-value     {ANY_INTEGER}
|REQ #binding-name      {$NUMBERSTRING},
    REQ #binding-value     {ANY_STRING}
|REQ #binding-name      {$STRING},
    REQ #binding-value     {ANY_STRING}
")

```

-- Used to make a simple or compound string out of the number bindings. --

```

DEFINE(USENUMBERSTRINGS, "
    REQ #binding-name      {$NUMBERSTRING},
    REQ #binding-value
{<string-expr> ::=      <hierarchic-expr>|<simple-expr> ;

<hierarchic-expr>      ::=      B_REF(SUP(CURR-OBJ))(<numberstring>)
                             B_REF(SUP(CURR-OBJ))(<separator>)
                             <simple-expr>;

<simple-expr> ::=      MK-STR(B_REF(CURR-OBJ))(<number>))
                     | U-ALPHA(B_REF(CURR-OBJ))(<number>))
                     | L-ALPHA(B_REF(CURR-OBJ))(<number>))
                     | U-ROM(B_REF(CURR-OBJ))(<number>))
                     | L-ROM(B_REF(CURR-OBJ))(<number>))
                     | MK-STR(ORD(CURR-OBJ))

```

```

| U-ALPHA(ORD(CURR-OBJ))
| L-ALPHA(ORD(CURR-OBJ))
| U-ROM(ORD(CURR-OBJ))
| L-ROM(ORD(CURR-OBJ))
| ANY_STRING;

```

```

$NUMBERSTRING
$SEPARATOR
$NUMBER}
")

```

-- Used to increment any of the number bindings. --

```

DEFINE(USENUMBERS, "
    REQ #binding-name      {$NUMBER},
    REQ #binding-value
        {<num-expr> ::= INC(B_REF(PREC(CURR-OBJ))(<number>))};
$NUMBER}
")

```

-- Used to initialise/specify or manipulate the bindings. The bindings defined by this macro are permitted to:

- any logical object class,
- any logical object,
- any layout object class except frame classes and block classes.

```

DEFINE(SPECIFYBINDINGS, "
$INITIALISEBINDINGS | $USENUMBERS | $USENUMBERSTRINGS
")

```

-- Used to initialise fnotenumber and fnotestring bindings. --

-- Note that footnote numbering is realized as a particular case of the general number binding mechanism supported by this DAP using the bindings <number> and <numberstring>. "fnotenumber" and "fnotestring" may be used for the compatibility with FOD26 | PM-26. --

```

DEFINE(INITIALISEFNOTE, "
    REQ #binding-name      {"fnotenumber"},
    REQ #binding-value
        {>=0}
")

```

-- Used to increment fnotenumber binding. --

```

DEFINE(INCFNOTENUMBER, "
    REQ #binding-name      {"fnotenumber"},
    REQ #binding-value
        {<num-expr> ::= INC(B_REF(PREC(CURR-OBJ))("fnotenumber"))};
")

```

-- Used to create a fnotestring from a fnotenumber binding. --

```
DEFINE(FNOTENUMBERSTRING, "  
  REQ #binding-name      {""fnotestring""},  
  REQ #binding-value      {<str-expr>::=  
    MK-STR(B_REF(CURR-OBJ)(""fnotenumber""))  
    | U-ALPHA(B_REF(CURR-OBJ)(""fnotenumber""))  
    | L-ALPHA(B_REF(CURR-OBJ)(""fnotenumber""))  
    | U-ROM(B_REF(CURR-OBJ)(""fnotenumber""))  
    | L-ROM(B_REF(CURR-OBJ)(""fnotenumber""))};  
")
```

-- Used to reset the footnote number string to a string literal. This provides a mechanism for setting the footnote number string to something other than a numeric value. --

```
DEFINE(FNOTESTRINGLITERAL, "  
  REQ #binding-name      {""fnotestring""},  
  REQ #binding-value      {ANY_STRING}  
")
```

-- Used to initialise PGnum binding. --

-- Note that a page numbering is realized as a particular case of the general number binding mechanism supported by this DAP using the bindings <number> and <numberstring>. "PGnum" may be used for the compatibility with FOD26 | PM-26. --

```
DEFINE(INITIALISEPGNUMBER, "  
  REQ #binding-name      {""PGnum""},  
  REQ #binding-value      {>= -1}  
")
```

-- Used to increment PGnum binding. --

```
DEFINE(USEPGNUMBERS, "  
  REQ #binding-name      {""PGnum""},  
  REQ #binding-value      {<num-expr>::= INC(B_REF(PREC(CURR-OBJ))(""PGnum""))};  
")
```

-- This string expression is allowed in a content generator for Number to automatically generate text for segment numbers, figure numbers or list numbers. (Note: B_REF(CURR-OBJ) is used for list numbers.) --

```
DEFINE(SEGMENTNUMBER, "  
<string-expr> ::=    [<pre-str>]<num-str>[<suf-str>];  
<num-str>      ::=    B_REF(SUP(CURR-OBJ))(<numberstring>)  
                  | B_REF(CURR-OBJ)(<numberstring>);  
<pre-str>      ::=    B_REF(SUP(CURR-OBJ))(<prefix>) | ANY_STRING;  
<suf-str>      ::=    B_REF(SUP(CURR-OBJ))(<suffix>) | ANY_STRING;  
$NUMBERSTRING  
$PREFIX  
$SUFFIX  
")
```

-- This string expression is allowed in a content generator for TableNumber to automatically generate text for a table number. --

```

DEFINE(TABLENUMBER, "
<string-expr> ::= [<pre-str>]<num-str>[<suf-str>];
<num-str> ::= B_REF
              (CURR-INST(OBJECT_CLASS_ID_OF(Table),(CURR-OBJ)))
              (<numberstring>);
<pre-str> ::= B_REF(SUP
                  (CURR-INST(OBJECT_CLASS_ID_OF(Table),(CURR-OBJ))))
              (<prefix>)
              | ANY_STRING;
<suf-str> ::= B_REF(SUP
                  (CURR-INST(OBJECT_CLASS_ID_OF(Table),(CURR-OBJ))))
              (<suffix>)
              | ANY_STRING;
$NUMBERSTRING
$PREFIX
$SUFFIX
")

```

-- This string expression is allowed in a content generator for PageNumber to automatically generate text for a page number. --

-- Note that a page number may be generated either from <number> or <numberstring> or from PGnum. PGnum is kept for the compatibility with FOD26 | PM-26. --

```

DEFINE(PGNUMBER, "
<string-expr> ::= [<pre-str>]<num-str>[<suf-str>];
<pre-str> ::= B_REF(SUP(<current-layout-object>))(<prefix>) | ANY_STRING;
<suf-str> ::= B_REF(SUP(<current-layout-object>))(<suffix>) | ANY_STRING;

<num-str> ::= MK-STR(<numeric-expr>)
            | U-ALPHA(<numeric-expr>)
            | L-ALPHA(<numeric-expr>)
            | U-ROM(<numeric-expr>)
            | L-ROM(<numeric-expr>)
            | B_REF(SUP(<layout-object-1>))(<numberstring>)
            | B_REF(<layout-object-2>)(<numberstring>);

<numeric-expr> ::= B_REF(SUP(<layout-object-1>))(<number>)
                 | B_REF(SUP(<layout-object-1>))("PGnum")
                 | B_REF(<layout-object-2>)(<number>)
                 | B_REF(<layout-object-2>)("PGnum");

<current-layout-object> ::= <layout-object-1> | <layout-object-2>;
<layout-object-1> ::= CURR-INST(<class-or-type-1>,(CURR-OBJ));
<layout-object-2> ::= CURR-INST(<class-or-type-2>,(CURR-OBJ));
<class-or-type-1> ::= 'frame';
<class-or-type-2> ::= 'page'
                   | OBJECT_CLASS_ID_OF(Page)
                   | OBJECT_CLASS_ID_OF(RectoPage)
                   | OBJECT_CLASS_ID_OF(VersoPage);
$PREFIX

```


\$SUFFIX
\$NUMBER
\$NUMBERSTRING

This string expression is allowed in a content generator for FootnoteNumber and FootnoteReference to automatically generate text for a footnote number. --
Note that a footnote number may be generated either from <numberstring> or from "fnotestring". "fnotestring" is kept for the compatibility with FOD26 | PM-26. --

```

DEFINE(FNNUMBER, "
<string-expr> ::= [<pre-str>]<num-str>[<suf-str>];
<num-str> ::= B_REF(SUP(CURR-OBJ))(<numberstring>)
           | B_REF(SUP(CURR-OBJ))("fnotestring")
           | ANY_STRING;
<pre-str> ::= B_REF(SUP(CURR-OBJ))(<prefix>)
           | ANY_STRING;
<suf-str>  ::= B_REF(SUP(CURR-OBJ))(<suffix>)
           | ANY_STRING;
$NUMBERSTRING
$PREFIX
$SUFFIX
")

```

This string expression is allowed in a content generator for ReferencedContent to automatically generate text for references such as to segment numbers, table numbers, figure numbers, list numbers, footnote numbers and <string> bindings associated with a referring (i.e. a target) logical object, or to page numbers, pageset numbers and <string> bindings associated with a layout object in which the referring logical object is laid out. --

```

DEFINE(REF, "
<string-expr> ::= [<pre-str>]<ref-str>[<suf-str>];
--
These are a prefix and a suffix of ReferencedContent itself, not those of referring text. e.g.
' (See' and ') '. --
<pre-str> ::= B_REF(SUP(CURR-OBJ))(<prefix>) | ANY_STRING;
<suf-str>  ::= B_REF(SUP(CURR-OBJ))(<suffix>) | ANY_STRING;

```

```

<ref-str> ::= { <ref-numberstring>
               | <ref-fnotestring>
               | <ref-pgnum>
               | <ref-number>
               | <ref-string>
               | ANY_STRING }... ;

```

```

<ref-numberstring> ::= [<pre-str-a>] <ref-str-a> [<suf-str-a>];
<pre-str-a> ::= B_REF(SUP(<target-object-1>))(<prefix>)
               | B_REF(<target-object>)(<prefix>) ;
<suf-str-a> ::= B_REF(SUP(<target-object-1>))(<suffix>)
               | B_REF(<target-object>)(<suffix>) ;
<ref-str-a> ::= B_REF(SUP(<target-object-1>))(<numberstring>)
               | B_REF(<target-object>)(<numberstring>);

```

```

<ref-fnotestring> ::= [<pre-str-b>] <ref-str-b> [<suf-str-b>];
<pre-str-b> ::= B_REF(SUP(<target-logical-object-1>))(<prefix>)
               | B_REF(<target-logical-object>)(<prefix>) ;
<suf-str-b> ::= B_REF(SUP(<target-logical-object-1>))(<suffix>)
               | B_REF(<target-logical-object>)(<suffix>) ;
<ref-str-b> ::= B_REF(SUP(<target-logical-object-1>))("fnotestring")
               | B_REF(<target-logical-object>)("fnotestring");

<ref-pgnum> ::= [<pre-str-c>] <ref-str-c> [<suf-str-c>];
<pre-str-c> ::= B_REF(SUP(<target-layout-object>))(<prefix>) ;
<suf-str-c> ::= B_REF(SUP(<target-layout-object>))(<suffix>) ;
<ref-str-c> ::= MK-STR(B_REF(SUP(<layout-object-1>))("PGnum"))
               | U-ALPHA(B_REF(SUP(<layout-object-1>))("PGnum"))
               | L-ALPHA(B_REF(SUP(<layout-object-1>))("PGnum"))
               | U-ROM(B_REF(SUP(<layout-object-1>))("PGnum"))
               | L-ROM(B_REF(SUP(<layout-object-1>))("PGnum"))
               | MK-STR(B_REF(<layout-object-2>))("PGnum"))
               | U-ALPHA(B_REF(<layout-object-2>))("PGnum"))
               | L-ALPHA(B_REF(<layout-object-2>))("PGnum"))
               | U-ROM(B_REF(<layout-object-2>))("PGnum"))
               | L-ROM(B_REF(<layout-object-2>))("PGnum");

<ref-number> ::= [<pre-str-d>] <ref-str-d> [<suf-str-d>];
<pre-str-d> ::= B_REF(SUP(<target-object-1>))(<prefix>)
               | B_REF(<target-object>)(<prefix>) ;
<suf-str-d> ::= B_REF(SUP(<target-object-1>))(<suffix>)
               | B_REF(<target-object>)(<suffix>) ;
<ref-str-d> ::= MK-STR(B_REF(SUP(<target-object-1>))(<number>))
               | U-ALPHA(B_REF(SUP(<target-object-1>))(<number>))
               | L-ALPHA(B_REF(SUP(<target-object-1>))(<number>))
               | U-ROM(B_REF(SUP(<target-object-1>))(<number>))
               | L-ROM(B_REF(SUP(<target-object-1>))(<number>))
               | MK-STR(B_REF(<target-object>)(<number>))
               | U-ALPHA(B_REF(<target-object>)(<number>))
               | L-ALPHA(B_REF(<target-object>)(<number>))
               | U-ROM(B_REF(<target-object>)(<number>))
               | L-ROM(B_REF(<target-object>)(<number>));

<ref-string> ::= [<pre-str-e>] <ref-str-e> [<suf-str-e>];
<pre-str-e> ::= B_REF(SUP(<target-object-1>))(<prefix>)
               | B_REF(<target-object>)(<prefix>) ;
<suf-str-e> ::= B_REF(SUP(<target-object-1>))(<suffix>)
               | B_REF(<target-object>)(<suffix>) ;
<ref-str-e> ::= B_REF(SUP(<target-object-1>))(<string>)
               | B_REF(<target-object>)(<string>);

<target-object> ::= <target-logical-object> | <target-layout-object>;
<target-object-1> ::= <target-logical-object-1> | <target-layout-object>;

<target-logical-object> ::= <logical-objects> | CURR-INST(<class-or-type-logical>, <logical-objects>);
<target-logical-object-x> ::= <logical-objects> | CURR-INST(<class-or-type-logical>, <logical-objects>);
<target-logical-object-1> ::= CURR-INST(<class-or-type-logical>, <logical-objects>);
<class-or-type-logical> ::= <logical-object-classes>
                          | 'composite-logical-object'
                          | 'basic-logical-object';

```

```

<target-layout-object> ::= <layout-object-1> | <layout-object-2>;
<layout-object-1> ::= CURR-INST(<class-or-type-layout-1>,<target-logical-object-x>);
<layout-object-2> ::= CURR-INST(<class-or-type-layout-2>,<target-logical-object-x>);
<class-or-type-layout-1> : <class-or-type-layout-1> ::= 'frame';
<class-or-type-layout-2> ::= 'page'
                        | OBJECT_CLASS_ID_OF(Page)
                        | OBJECT_CLASS_ID_OF(RectoPage)
                        | OBJECT_CLASS_ID_OF(VersoPage);

```

```

$PREFIX
$SUFFIX
$NUMBERSTRING
$NUMBER
$STRING
$LogicalObjects
$LogicalObjectClasses

```

```

")

```

-- This string expression is allowed in a content generator for CommonReference to automatically generate text for references such as to segment numbers, table numbers, figure numbers, list numbers, footnote numbers and <string> bindings associated with a logical object which is laid out in a current layout object, or to page numbers, pageset numbers and <string> bindings associated with a current or a superior layout object. --

```

DEFINE(COMMONREF, "

```

```

<string-expr> ::= [<pre-str>]<ref-str>[<suf-str>];

```

```

<pre-str>      ::= B_REF(SUP(<current-layout-object>))(<prefix>) | ANY_STRING;

```

```

<suf-str>      ::= B_REF(SUP(<current-layout-object>))(<suffix>) | ANY_STRING;

```

```

<ref-str>      ::= { <ref-numberstring>
                    | <ref-fnotestring>
                    | <ref-pgnum>
                    | <ref-number>
                    | <ref-string>
                    | ANY_STRING } ... ;

```

```

<ref-numberstring> ::= [<pre-str-a>] <ref-str-a> [<suf-str-a>];

```

```

<pre-str-a> ::= B_REF(SUP(<current-object>))(<prefix>)
              | B_REF(<current-object>)(<prefix>) ;

```

```

<suf-str-a> ::= B_REF(SUP(<current-object>))(<suffix>)
              | B_REF(<current-object>)(<suffix>) ;

```

```

<ref-str-a> ::= B_REF(SUP(<current-object>))(<numberstring>)
              | B_REF(<current-object>)(<numberstring>);

```

```

<ref-fnotestring> ::= [<pre-str-b>] <ref-str-b> [<suf-str-b>];

```

```

<pre-str-b> ::= B_REF(SUP(<current-logical-object>))(<prefix>)
              | B_REF(<current-logical-object>)(<prefix>) ;

```

```

<suf-str-b> ::= B_REF(SUP(<current-logical-object>))(<suffix>)
              | B_REF(<current-logical-object>)(<suffix>) ;

```

```

<ref-str-b> ::= B_REF(SUP(<current-logical-object>))("fnotestring")
              | B_REF(<current-logical-object>)("fnotestring");

```

```

<ref-pgnum> ::= [<pre-str-c>] <ref-str-c> [<suf-str-c>];

```

```

<pre-str-c> ::= B_REF(SUP(<current-layout-object>))(<prefix>) ;

```



```

<suf-str-c> ::= B_REF(SUP(<current-layout-object>))(<suffix>) ;
<ref-str-c> ::=
    MK-STR(B_REF(SUP(<layout-object-1>))("PGnum"))
    | U-ALPHA(B_REF(SUP(<layout-object-1>))("PGnum"))
    | L-ALPHA(B_REF(SUP(<layout-object-1>))("PGnum"))
    | U-ROM(B_REF(SUP(<layout-object-1>))("PGnum"))
    | L-ROM(B_REF(SUP(<layout-object-1>))("PGnum"))
    | MK-STR(B_REF(SUP(<layout-object-2>))("PGnum"))
    | U-ALPHA(B_REF(SUP(<layout-object-2>))("PGnum"))
    | L-ALPHA(B_REF(SUP(<layout-object-2>))("PGnum"))
    | U-ROM(B_REF(SUP(<layout-object-2>))("PGnum"))
    | L-ROM(B_REF(SUP(<layout-object-2>))("PGnum"));

```

```

<ref-number> ::= [<pre-str-d>] <ref-str-d> [<suf-str-d>];

```

```

<pre-str-d> ::= B_REF(SUP(<current-object>))(<prefix>)

```

```

    | B_REF(<current-object>)(<prefix>) ;

```

```

<suf-str-d> ::= B_REF(SUP(<current-object>))(<suffix>)

```

```

    | B_REF(<current-object>)(<suffix>) ;

```

```

<ref-str-d> ::=
    MK-STR(B_REF(SUP(<current-object>))(<number>))
    | U-ALPHA(B_REF(SUP(<current-object>))(<number>))
    | L-ALPHA(B_REF(SUP(<current-object>))(<number>))
    | U-ROM(B_REF(SUP(<current-object>))(<number>))
    | L-ROM(B_REF(SUP(<current-object>))(<number>))
    | MK-STR(B_REF(<current-object>)(<number>))
    | U-ALPHA(B_REF(<current-object>)(<number>))
    | L-ALPHA(B_REF(<current-object>)(<number>))
    | U-ROM(B_REF(<current-object>)(<number>))
    | L-ROM(B_REF(<current-object>)(<number>));

```

```

<ref-string> ::= [<pre-str-e>] <ref-str-e> [<suf-str-e>];

```

```

<pre-str-e> ::= B_REF(SUP(<current-object>))(<prefix>)

```

```

    | B_REF(<current-object>)(<prefix>) ;

```

```

<suf-str-e> ::= B_REF(SUP(<current-object>))(<suffix>)

```

```

    | B_REF(<current-object>)(<suffix>) ;

```

```

<ref-str-e> ::= B_REF(SUP(<current-object>))(<string>)

```

```

    | B_REF(<current-object>)(<string>);

```

```

<current-object> ::= <current-logical-object> | <current-layout-object>;

```

```

<current-logical-object> ::= CURR-INST(<class-or-type-logical>,(CURR-OBJ));

```

```

<class-or-type-logical> ::= <logical-object-classes>

```

```

    | 'composite-logical-object'

```

```

    | 'basic-logical-object';

```

```

<current-layout-object> ::= <layout-object-1> | <layout-object-2>;

```

```

<layout-object-1> ::= CURR-INST(<class-or-type-layout-1>,(CURR-OBJ));

```

```

<layout-object-2> ::= CURR-INST(<class-or-type-layout-2>,(CURR-OBJ));

```

```

<class-or-type-layout-1> ::= 'frame';

```

```

<class-or-type-layout-2> ::= 'page'

```

```

    | OBJECT_CLASS_ID_OF(Page)

```

```

    | OBJECT_CLASS_ID_OF(RectoPage)

```

```

    | OBJECT_CLASS_ID_OF(VersoPage);

```

```

$PREFIX

```

```

$SUFFIX

```

```

$NUMBERSTRING

```



```

$NUMBER
$STRING
$LogicalObjectClasses
")

```

-- This string expression is allowed in a content generator for CommonNumber to automatically generate text for general references to any kinds of numbers including segment numbers, table numbers, figure numbers, list numbers, footnote numbers, page numbers or pageset numbers, etc. --

```

DEFINE(COMMONNUMBER, "
  <string-expr> ::= <ref-numberstring>
                  | <ref-number>;

  <ref-numberstring> ::= [<pre-str-a>] <ref-str-a> [<suf-str-a>];
  <pre-str-a> ::= B_REF(SUP(<current-object>))(<prefix>)
                  | B_REF(<current-object>)(<prefix>) | ANY_STRING;
  <suf-str-a> ::= B_REF(SUP(<current-object>))(<suffix>)
                  | B_REF(<current-object>)(<suffix>) | ANY_STRING;
  <ref-str-a> ::= B_REF(SUP(<current-object>))(<numberstring>)
                  | B_REF(<current-object>)(<numberstring>);

  <ref-number> ::= [<pre-str-b>] <ref-str-b> [<suf-str-b>];
  <pre-str-b> ::= B_REF(SUP(<current-object>))(<prefix>)
                  | B_REF(<current-object>)(<prefix>) | ANY_STRING;
  <suf-str-b> ::= B_REF(SUP(<current-object>))(<suffix>)
                  | B_REF(<current-object>)(<suffix>) | ANY_STRING;
  <ref-str-b> ::= MK-STR(B_REF(SUP(<current-object>))(<number>))
                  | U-ALPHA(B_REF(SUP(<current-object>))(<number>))
                  | L-ALPHA(B_REF(SUP(<current-object>))(<number>))
                  | U-ROM(B_REF(SUP(<current-object>))(<number>))
                  | L-ROM(B_REF(SUP(<current-object>))(<number>))
                  | MK-STR(B_REF(<current-object>)(<number>))
                  | U-ALPHA(B_REF(<current-object>)(<number>))
                  | L-ALPHA(B_REF(<current-object>)(<number>))
                  | U-ROM(B_REF(<current-object>)(<number>))
                  | L-ROM(B_REF(<current-object>)(<number>));

  <current-object> ::= <current-logical-object> | <current-layout-object>;

  <current-logical-object> ::= CURR-INST(<class-or-type-logical>,(CURR-OBJ));
  <class-or-type-logical> ::= <logical-object-classes>
                           | 'composite-logical-object'
                           | 'basic-logical-object';

  <current-layout-object> ::= CURR-INST(<class-or-type-layout>,(CURR-OBJ));
  <class-or-type-layout> ::= 'frame'
                           | 'page'
                           | OBJECT_CLASS_ID_OF(Page)
                           | OBJECT_CLASS_ID_OF(RectoPage)
                           | OBJECT_CLASS_ID_OF(VersoPage);

  $PREFIX
  $SUFFIX

```

```

$NUMBERSTRING
$NUMBER
$LogicalObjectClasses
")

```

-- This string expression is allowed in a content generator for CurrentInstance to automatically generate text for general references to <string> bindings associated with a current logical or layout object. --

```

DEFINE(CURRENTINSTANCE, "
<string-expr> ::= [<pre-str>]<ref-str>[<suf-str>];
<pre-str>      ::= B_REF(SUP(<current-object>))(<prefix>)
                  | B_REF(<current-object>)(<prefix>) | ANY_STRING;
<suf-str>      ::= B_REF(SUP(<current-object>))(<suffix>)
                  | B_REF(<current-object>)(<suffix>) | ANY_STRING;
<ref-str>      ::= B_REF(SUP(<current-object>))(<string>)
                  | B_REF(<current-object>)(<string>);

<current-object>      ::= <current-logical-object> | <current-layout-object>;

<current-logical-object> ::= CURR-INST(<class-or-type-logical>,(CURR-OBJ));
<class-or-type-logical>  ::= <logical-object-classes>
                          | 'composite-logical-object'
                          | 'basic-logical-object';

<current-layout-object>  ::= CURR-INST(<class-or-type-layout>,(CURR-OBJ));
<class-or-type-layout>  ::= 'frame'
                          | 'page'
                          | OBJECT_CLASS_ID_OF(Page)
                          | OBJECT_CLASS_ID_OF(RectoPage)
                          | OBJECT_CLASS_ID_OF(VersoPage);

$PREFIX
$SUFFIX
$STRING
$LogicalObjectClasses
")

```

-- This string expression is allowed in a content generator for GenericBlock to automatically generate text for general references to bindings associated with a current layout object. --

```

DEFINE(GENERICBLOCKREF, "
<string-expr> ::= [<pre-str>]<ref-str>[<suf-str>];

<pre-str>      ::= B_REF(SUP(CURR-OBJ))(<prefix>) | ANY_STRING;
<suf-str>      ::= B_REF(SUP(CURR-OBJ))(<suffix>) | ANY_STRING;

<ref-str>      ::= MK-STR(B_REF(SUP(CURR-OBJ))(<number>))
                  | U-ALPHA(B_REF(SUP(CURR-OBJ))(<number>))
                  | L-ALPHA(B_REF(SUP(CURR-OBJ))(<number>))
                  | U-ROM(B_REF(SUP(CURR-OBJ))(<number>))
                  | L-ROM(B_REF(SUP(CURR-OBJ))(<number>))
                  | MK-STR(B_REF(SUP(CURR-OBJ))("PGnum"))

```

```

| U-ALPHA(B_REF(SUP(CURR-OBJ))("PGnum"))
| L-ALPHA(B_REF(SUP(CURR-OBJ))("PGnum"))
| U-ROM(B_REF(SUP(CURR-OBJ))("PGnum"))
| L-ROM(B_REF(SUP(CURR-OBJ))("PGnum"))
| B_REF(SUP(CURR-OBJ))(<numberstring>)
| B_REF(SUP(CURR-OBJ))(<string>);

```

```

$PREFIX
$SUFFIX
$NUMBER
$NUMBERSTRING
$STRING
")

```

DEFINE(DocLogRootGFS, "

```

<construction-expr> ::= <construction-term>
                        | <construction-type>;

<construction-term> ::= <construction-factor>
                        | OPT <construction-factor>
                        | REP <construction-factor>
                        | OPT REP <construction-factor>;

<construction-type> ::= SEQ({<construction-term>}...)
                        | CHO({<construction-term>}...);

<construction-factor> ::= OBJECT_CLASS_ID_OF(Passage)
                        | OBJECT_CLASS_ID_OF(NumberedSegment)
                        | <construction-type>;

")

```

DEFINE(CONSTRAINT-1, "

```

<constraint-1> ::= <construction-term>
                  | <construction-type>;

<construction-term> ::= <construction-factor>
                        | OPT <construction-factor>
                        | REP <construction-factor>
                        | OPT REP <construction-factor>;

<construction-type> ::= SEQ({<construction-term>}...)
                        | CHO({<construction-term>}...);

<construction-factor> ::= OBJECT_CLASS_ID_OF(Passage)
                        | OBJECT_CLASS_ID_OF(NumberedSegment)
                        | OBJECT_CLASS_ID_OF(Paragraph)
                        | OBJECT_CLASS_ID_OF(BodyText)
                        | OBJECT_CLASS_ID_OF(BodyRaster)
                        | OBJECT_CLASS_ID_OF(BodyGeometric)
                        | OBJECT_CLASS_ID_OF(Figure)
                        | OBJECT_CLASS_ID_OF(Table)
                        | OBJECT_CLASS_ID_OF(NumberedList)
                        | OBJECT_CLASS_ID_OF(UnNumberedList)
                        | OBJECT_CLASS_ID_OF(DefinitionList)

```

```

        | <construction-type>;
")

DEFINE(CONSTRAINT-2, "
    <constraint-2> ::= OBJECT_CLASS_ID_OF(Title)
                    | OPT OBJECT_CLASS_ID_OF(Title);
")

```

```

DEFINE(PassageGFS, "
    <construction-expr> ::= <constraint-1>
                          | SEQ(<constraint-2><constraint-1>);

    $CONSTRAINT-1
    $CONSTRAINT-2
")

```

```

DEFINE(NumberedSegmentGFS, "
    <construction-expr> ::=
        SEQ(<term-1>[<constraint-2>][<constraint-1>]);

    <term-1> ::= OBJECT_CLASS_ID_OF(Number);

    $CONSTRAINT-1
    $CONSTRAINT-2
")

```

```

DEFINE(CONSTRAINT-3, "
    <construction-expr> ::= <construction-term>
                          | <construction-type>;

    <construction-term> ::= <construction-factor>
                          | OPT <construction-factor>
                          | REP <construction-factor>
                          | OPT REP <construction-factor>;

    <construction-type> ::= SEQ({<construction-term>}...)
                          | CHO({<construction-term>}...);

    <construction-factor> ::= OBJECT_CLASS_ID_OF(BodyText)
                          | OBJECT_CLASS_ID_OF(BodyRaster)
                          | OBJECT_CLASS_ID_OF(BodyGeometric)
                          | OBJECT_CLASS_ID_OF(Footnote)
                          | OBJECT_CLASS_ID_OF(Phrase)
                          | OBJECT_CLASS_ID_OF(Reference)
                          | <construction-type>;
")

```

```

DEFINE(ParagraphGFS, "$CONSTRAINT-3")

```


DEFINE(TitleGFS, "\$CONSTRAINT-3")

DEFINE(FigureGFS, "

```
<construction-expr> ::=
    SEQ([<term-1>][<term-2>][<term-3>]<term-4>)
    | SEQ([<term-1>][<term-2>]<term-4>[<term-3>])
    | SEQ([<term-3>][<term-1>][<term-2>]<term-4>)
    | SEQ(<term-4>[<term-1>][<term-2>][<term-3>])
    | SEQ([<term-3>]<term-4>[<term-1>][<term-2>])
    | SEQ(<term-4>[<term-3>][<term-1>][<term-2>]);

<term-1> ::= OBJECT_CLASS_ID_OF(Number)
           | OPT OBJECT_CLASS_ID_OF(Number);
<term-2> ::= OBJECT_CLASS_ID_OF(Caption)
           | OPT OBJECT_CLASS_ID_OF(Caption);
<term-3> ::= OBJECT_CLASS_ID_OF(Description)
           | OPT OBJECT_CLASS_ID_OF(Description);
<term-4> ::= OBJECT_CLASS_ID_OF(Artwork)
           | OBJECT_CLASS_ID_OF(Form);

")
```

DEFINE(ArtworkGFS, "

```
<construction-expr> ::= <construction-term>
                        | <construction-type>;

<construction-term> ::= <construction-factor>
                        | OPT <construction-factor>
                        | REP <construction-factor>
                        | OPT REP <construction-factor>;

<construction-type> ::= SEQ({<construction-term>}...)
                        | CHO({<construction-term>}...);

<construction-factor> ::= OBJECT_CLASS_ID_OF(Phrase)
                        | OBJECT_CLASS_ID_OF(BodyRaster)
                        | OBJECT_CLASS_ID_OF(BodyGeometric)
                        | <construction-type>;

")
```

DEFINE(CONSTRAINT-5, "

```
<construction-expr> ::= <construction-term>
                        | <construction-type>;

<construction-term> ::= <construction-factor>
                        | OPT <construction-factor>
                        | REP <construction-factor>
                        | OPT REP <construction-factor>;

<construction-type> ::= SEQ({<construction-term>}...)
                        | CHO({<construction-term>}...);

<construction-factor> ::= OBJECT_CLASS_ID_OF(Phrase)
```

```

| OBJECT_CLASS_ID_OF(Footnote)
| OBJECT_CLASS_ID_OF(Reference)
| OBJECT_CLASS_ID_OF(BodyText)
| <construction-type>;

")

DEFINE(PhraseGFS, "$CONSTRAINT-5")

DEFINE(CaptionGFS, "$CONSTRAINT-5")

DEFINE(DescriptionGFS, "$CONSTRAINT-5")

DEFINE(FootnoteGFS, "
<construction-expr> ::=
    SEQ(OBJECT_CLASS_ID_OF(FootnoteReference)
        OBJECT_CLASS_ID_OF(FootnoteBody));

")

DEFINE(FootnoteBodyGFS, "
<construction-expr> ::=    SEQ(OBJECT_CLASS_ID_OF(FootnoteNumber)
                                <term-1>);

<term-1> ::=
    OBJECT_CLASS_ID_OF(FootnoteText)
    | OBJECT_CLASS_ID_OF(Reference)
    | REP OBJECT_CLASS_ID_OF(FootnoteText)
    | REP OBJECT_CLASS_ID_OF(Reference)
    | CHO({OBJECT_CLASS_ID_OF(FootnoteText)
           | OBJECT_CLASS_ID_OF(Reference)})...
    | REP CHO({OBJECT_CLASS_ID_OF(FootnoteText)
               | OBJECT_CLASS_ID_OF(Reference)})...;

")

DEFINE(ReferenceGFS, "
<construction-expr> ::=    OBJECT_CLASS_ID_OF(ReferencedContent)
                            | SEQ([<term>]
                                OBJECT_CLASS_ID_OF(ReferencedContent)
                                [<term>]);

<term> ::=
    OBJECT_CLASS_ID_OF(BodyText)
    | OPT OBJECT_CLASS_ID_OF(BodyText)
    | CHO( {OBJECT_CLASS_ID_OF(BodyText)}... );

")

DEFINE(CommonContentGFS, "
<construction-expr> ::=    <construction-factor>
                            | SEQ(<construction-factor>...);

<construction-factor> ::=    OBJECT_CLASS_ID_OF(CommonText)
                            | OBJECT_CLASS_ID_OF(PageNumber)

```

```

| OBJECT_CLASS_ID_OF(CommonRaster)
| OBJECT_CLASS_ID_OF(CommonGeometric)
| OBJECT_CLASS_ID_OF(CommonReference)
| OBJECT_CLASS_ID_OF(CommonNumber)
| OBJECT_CLASS_ID_OF(CurrentInstance)
| OBJECT_CLASS_ID_OF(TableNumber);
")

DEFINE(TableGFS,"
<construction-expr> ::=      REP CHO(OBJECT_CLASS_ID_OF(Row)...)
                             | REP OBJECT_CLASS_ID_OF(Row)
                             | SEQ(OBJECT_CLASS_ID_OF(Row)...);
")

DEFINE(RowGFS,"
<construction-expr> ::=      <simple-table> | <complex-table>;
<simple-table>                ::=      REP OBJECT_CLASS_ID_OF(EntryElement)
                             | REP CHO(OBJECT_CLASS_ID_OF(EntryElement)...)
                             | SEQ(OBJECT_CLASS_ID_OF(EntryElement)...);
<complex-table>              ::=      SEQ(OBJECT_CLASS_ID_OF(EntryElement)
                             OBJECT_CLASS_ID_OF(TableComponent));
")

DEFINE(TableComponentGFS,"
<construction-expr> ::=      REP OBJECT_CLASS_ID_OF(RowComponent);
")

DEFINE(RowComponentGFS,"
<construction-expr> ::=      REP OBJECT_CLASS_ID_OF(EntryElement)
                             | REP CHO(OBJECT_CLASS_ID_OF(EntryElement)...)
                             | SEQ(OBJECT_CLASS_ID_OF(EntryElement)...);
")

DEFINE(FormGFS,"
<construction-expr> ::=      AGG(<factor>...);
<factor>                    ::=      OBJECT_CLASS_ID_OF(EntryElement)
                             | OBJECT_CLASS_ID_OF(EntryGroup);
")

DEFINE(EntryGroupGFS,"$FormGFS")

DEFINE(EntryElementGFS,"
<construction-expr> ::=      OBJECT_CLASS_ID_OF(EntryText)
                             | OBJECT_CLASS_ID_OF(EntryRaster)
                             | OBJECT_CLASS_ID_OF(EntryGeometric);
")

```

```

DEFINE(NumberedListGFS,"
<construction-expr> ::=      SEQ(OBJECT_CLASS_ID_OF(Number)
                                OBJECT_CLASS_ID_OF(ListItem))
                                | REP SEQ(OBJECT_CLASS_ID_OF(Number)
                                           OBJECT_CLASS_ID_OF(ListItem));
")

DEFINE(UnNumberedListGFS,"
<construction-expr> ::=      OBJECT_CLASS_ID_OF(ListItem)
                                | REP OBJECT_CLASS_ID_OF(ListItem)
                                | SEQ(<separator-obj> OBJECT_CLASS_ID_OF(ListItem))
                                | REP SEQ(<separator-obj> OBJECT_CLASS_ID_OF(ListItem));

<separator-obj>      ::= OBJECT_CLASS_ID_OF(BodyText)
                                | OBJECT_CLASS_ID_OF(BodyRaster)
                                | OBJECT_CLASS_ID_OF(BodyGeometric);
")

DEFINE(DefinitionListGFS,"
<construction-expr>  ::= SEQ(OBJECT_CLASS_ID_OF(ListTerm)
                                OBJECT_CLASS_ID_OF(ListItem))
                                | REP SEQ(OBJECT_CLASS_ID_OF(ListTerm)
                                           OBJECT_CLASS_ID_OF(ListItem));
")

DEFINE(ListItemGFS,"
<construction-expr>  ::= <term> | CHO(<term>...);

<term>               ::= REP OBJECT_CLASS_ID_OF(Phrase)
                                | OBJECT_CLASS_ID_OF(NumberedList)
                                | OBJECT_CLASS_ID_OF(UnNumberedList)
                                | OBJECT_CLASS_ID_OF(DefinitionList);
")

DEFINE(ListTermGFS,"$CONSTRAINT-3")

```

7.3.2 Factor constraints

7.3.2.1 FACTOR ANY-LOGICAL

```

{
GENERIC:
    REQ  Object-type           {VIRTUAL},
    REQ  Object-class-identifier {ANY_VALUE}
SPECIFIC:
    PERM Object-type           {VIRTUAL},
    REQ  Object-identifier     {ANY_VALUE},
    REQ  Object-class          {VIRTUAL}
SPECIFIC_AND_GENERIC:

```


PERM	Protection	{ANY_VALUE},
PERM	User-readable-comments	{ANY_STRING},
PERM	User-visible-name	{ANY_STRING}

}

7.3.2.2 FACTOR COMP-LOGICAL

```

:ANY-LOGICAL {
GENERIC:
  REQ  Object-type          {'composite-logical-object'}
SPECIFIC:
  REQ  Subordinates         {VIRTUAL},
  PERM Object-type          {'composite-logical-object'}
SPECIFIC_AND_GENERIC:
  PERM Bindings              {PMUL {$SPECIFYBINDINGS}},
  PERM Default-value-lists   {REQ #basic-logical-attributes
                                {PERM #presentation-style {ANY_VALUE},
                                PERM #layout-style      {ANY_VALUE}}}}

```

7.3.2.3 FACTOR BASIC-LOGICAL

```

:ANY-LOGICAL {
GENERIC:
  REQ  Object-type          {'basic-logical-object'},
  PERM Resource              {ANY_VALUE}
SPECIFIC:
  PERM Object-type          {'basic-logical-object'}
SPECIFIC_AND_GENERIC:
  PERM Bindings              {PMUL {$SPECIFYBINDINGS}}

```

7.3.2.4 FACTOR ANY-COMMON

```

{
GENERIC:
  REQ  Object-type          {VIRTUAL},
  REQ  Object-class-identifier {ANY_VALUE},
  PERM Bindings              {PMUL {$SPECIFYBINDINGS}},
  PERM Protection            {ANY_VALUE},
  PERM User-readable-comments {ANY_STRING},
  PERM User-visible-name     {ANY_STRING}
}

```

7.3.3 Constituent constraints

7.3.3.1 DocumentLogicalRoot

```
:ANY-LOGICAL {  
GENERIC:  
  REQ  Object-type          {'document-logical-root'},  
  REQ  Generator-for-subordinates {$DocLogRootGFS},  
  REQ  Application-comments    {REQ #constraint-name {"0"},  
                                PERM #external-data {ANY_VALUE}}  
SPECIFIC:  
  PERM Object-type          {'document-logical-root'},  
  REQ  Object-class         {OBJECT_CLASS_ID_OF(DocumentLogicalRoot)},  
  REQ  Subordinates         {SUB_ID_OF(NumberedSegment)+,  
                              SUB_ID_OF(Passage)+},  
  PERM Application-comments {REQ #constraint-name {"0"},  
                              PERM #external-data {ANY_VALUE}}  
SPECIFIC_AND_GENERIC:  
  PERM Bindings              {PMUL {$SPECIFYBINDINGS},  
                              PERM {$INITIALISEFNOTE}},  
  PERM Default-value-lists   {REQ #basic-logical-attributes  
                              {PERM #presentation-style {ANY_VALUE},  
                              PERM #layout-style {ANY_VALUE}}}  
}
```

7.3.3.2 Passage

```
:COMP-LOGICAL {  
GENERIC:  
  REQ  Generator-for-subordinates {$PassageGFS},  
  REQ  Application-comments    {REQ #constraint-name {"1"},  
                                PERM #external-data {ANY_VALUE}}  
SPECIFIC:  
  REQ  Object-class         {OBJECT_CLASS_ID_OF(Passage)},  
  REQ  Subordinates         {SUB_ID_OF(Title),  
                              SUB_ID_OF(Passage)+,  
                              SUB_ID_OF(NumberedSegment)+,  
                              SUB_ID_OF(BodyText)+,  
                              SUB_ID_OF(BodyRaster)+,  
                              SUB_ID_OF(BodyGeometric)+,  
                              SUB_ID_OF(Figure)+,  
                              SUB_ID_OF(Paragraph)+,  
                              SUB_ID_OF(Table)+,  
                              SUB_ID_OF(NumberedList)+,  
                              SUB_ID_OF(DefinitionList)+,  
                              SUB_ID_OF(UnNumberedList)+},  
  PERM Application-comments {REQ #constraint-name {"1"},  
                              PERM #external-data {ANY_VALUE}}  
SPECIFIC_AND_GENERIC:  
  PERM Bindings              {PMUL {$SPECIFYBINDINGS},  
                              PERM {$INITIALISEFNOTE}},
```

```

    PERM  Layout-style          {STYLE_ID_OF(LStyle1)}
}

```

7.3.3.3 NumberedSegment

```

:COMP-LOGICAL {
GENERIC:
    REQ  Generator-for-subordinates  {$NumberedSegmentGFS},
    REQ  Application-comments        {REQ #constraint-name {"2"},
                                     PERM #external-data {ANY_VALUE}}

SPECIFIC:
    REQ  Object-class                {OBJECT_CLASS_ID_OF(NumberedSegment)},
    REQ  Subordinates                {SUB_ID_OF(Number),
                                     SUB_ID_OF(Title),
                                     SUB_ID_OF(Passage)+,
                                     SUB_ID_OF(NumberedSegment)+,
                                     SUB_ID_OF(BodyText)+,
                                     SUB_ID_OF(BodyRaster)+,
                                     SUB_ID_OF(BodyGeometric)+,
                                     SUB_ID_OF(Paragraph)+,
                                     SUB_ID_OF(Figure)+,
                                     SUB_ID_OF(Table)+,
                                     SUB_ID_OF(NumberedList)+,
                                     SUB_ID_OF(DefinitionList)+,
                                     SUB_ID_OF(UnNumberedList)+},
    PERM  Application-comments        {REQ #constraint-name {"2"},
                                     PERM #external-data {ANY_VALUE}}

SPECIFIC_AND_GENERIC:
    PERM  Layout-style                {STYLE_ID_OF(LStyle1)}
}

```

7.3.3.4 Number

```

:BASIC-LOGICAL {
GENERIC:
    REQ  Content-generator           {$SEGMENTNUMBER},
    REQ  Application-comments         {REQ #constraint-name {"3"},
                                     PERM #external-data {ANY_VALUE}}

SPECIFIC:
    REQ  Object-class                {OBJECT_CLASS_ID_OF(Number)},
    PERM  Content-generator           {$SEGMENTNUMBER},
    PERM  Application-comments        {REQ #constraint-name {"3"},
                                     PERM #external-data {ANY_VALUE}}

SPECIFIC_AND_GENERIC:
    PERM  Layout-style                {STYLE_ID_OF(LStyle2)},
    PERM  Presentation-style          {STYLE_ID_OF(PStyle1)},
    PERM  Content-architecture-class  {$FC|$PC|$FPC}
}

```

7.3.3.5 Title

```
:COMP-LOGICAL {
GENERIC:
    REQ    Generator-for-subordinates {$TitleGFS},
    REQ    Application-comments      {REQ #constraint-name {"4"},
                                     PERM #external-data {ANY_VALUE}}

SPECIFIC:
    REQ    Object-class              {OBJECT_CLASS_ID_OF(Title)},
    REQ    Subordinates              {SUB_ID_OF(BodyText)+,
                                     SUB_ID_OF(BodyRaster)+,
                                     SUB_ID_OF(BodyGeometric)+,
                                     SUB_ID_OF(Phrase)+,
                                     SUB_ID_OF(Footnote)+,
                                     SUB_ID_OF(Reference)+,
    PERM    Application-comments      {REQ #constraint-name {"4"},
                                     PERM #external-data {ANY_VALUE}}

SPECIFIC_AND_GENERIC:
    PERM    Layout-style              {STYLE_ID_OF(LStyle1)}
}
```

7.3.3.6 Caption

```
:COMP-LOGICAL {
GENERIC:
    REQ    Generator-for-subordinates {$CaptionGFS},
    REQ    Application-comments      {REQ #constraint-name {"5"},
                                     PERM #external-data {ANY_VALUE}}

SPECIFIC:
    REQ    Object-class              {OBJECT_CLASS_ID_OF(Caption)},
    REQ    Subordinates              {SUB_ID_OF(BodyText)+,
                                     SUB_ID_OF(Phrase)+,
                                     SUB_ID_OF(Footnote)+,
                                     SUB_ID_OF(Reference)+,
    PERM    Application-comments      {REQ #constraint-name {"5"},
                                     PERM #external-data {ANY_VALUE}}

SPECIFIC_AND_GENERIC:
    PERM    Layout-style              {STYLE_ID_OF(LStyle1)}
}
```

7.3.3.7 Paragraph

```
:COMP-LOGICAL {
GENERIC:
    REQ    Generator-for-subordinates {$ParagraphGFS},
    REQ    Application-comments      {REQ #constraint-name {"6"},
                                     PERM #external-data {ANY_VALUE}}

SPECIFIC:
    REQ    Object-class              {OBJECT_CLASS_ID_OF(Paragraph)},
    REQ    Subordinates              {SUB_ID_OF(BodyText)+,
                                     SUB_ID_OF(Footnote)+,
                                     SUB_ID_OF(BodyRaster)+,
                                     SUB_ID_OF(BodyGeometric)+,
}
```



```

        SUB_ID_OF(Phrase)+,
        SUB_ID_OF(Reference)+},
    PERM Application-comments {REQ #constraint-name {"6"},
        PERM #external-data {ANY_VALUE}}
SPECIFIC_AND_GENERIC:
    PERM Layout-style {STYLE_ID_OF(LStyle1)}
}

```

7.3.3.8 Phrase

```

:COMP-LOGICAL {
GENERIC:
    REQ Generator-for-subordinates {$PhraseGFS},
    REQ Application-comments {REQ #constraint-name {"7"},
        PERM #external-data {ANY_VALUE}}
SPECIFIC:
    REQ Object-class {OBJECT_CLASS_ID_OF(Phrase)},
    REQ Subordinates {SUB_ID_OF(BodyText)+,
        SUB_ID_OF(Footnote)+,
        SUB_ID_OF(Phrase)+,
        SUB_ID_OF(Reference)+},
    PERM Application-comments {REQ #constraint-name {"7"},
        PERM #external-data {ANY_VALUE}}
SPECIFIC_AND_GENERIC:
    PERM Layout-style {STYLE_ID_OF(LStyle1)}
}

```

7.3.3.9 Footnote

```

:COMP-LOGICAL {
GENERIC:
    REQ Generator-for-subordinates {$FootnoteGFS},
    PERM Bindings {PMUL {$SPECIFYBINDINGS},
        PERM{REQ{$INCFNOTENUMBER,$FNOTENUMBERSTRING}
            | $FNOTESTRINGLITERAL} },
    REQ Application-comments {REQ #constraint-name {"8"},
        PERM #external-data {ANY_VALUE}}
SPECIFIC:
    REQ Object-class {OBJECT_CLASS_ID_OF(Footnote)},
    REQ Subordinates {SUB_ID_OF(FootnoteReference),
        SUB_ID_OF(FootnoteBody)},
    PERM Bindings {PMUL {$SPECIFYBINDINGS},
        PERM {$FNOTESTRINGLITERAL}},
    PERM Application-comments {REQ #constraint-name {"8"},
        PERM #external-data {ANY_VALUE}}
SPECIFIC_AND_GENERIC:
    PERM Layout-style {STYLE_ID_OF(LStyle1)}
}

```

7.3.3.10 FootnoteNumber

:BASIC-LOGICAL {

GENERIC:

REQ	Content-generator	{ \$FNNUMBER },
REQ	Application-comments	{ REQ #constraint-name {"9"},
		PERM #external-data {ANY_VALUE}}

SPECIFIC:

REQ	Object-class	{ OBJECT_CLASS_ID_OF(FootnoteNumber) },
PERM	Content-generator	{ \$FNNUMBER },
PERM	Application-comments	{ REQ #constraint-name {"9"},
		PERM #external-data {ANY_VALUE}}

SPECIFIC_AND_GENERIC:

PERM	Layout-style	{ STYLE_ID_OF(LStyle9) },
PERM	Presentation-style	{ STYLE_ID_OF(PStyle1) },
PERM	Content-architecture-class	{ \$FC \$PC \$FPC }

}

7.3.3.11 FootnoteReference

:BASIC-LOGICAL {

GENERIC:

REQ	Content-generator	{ \$FNNUMBER },
REQ	Application-comments	{ REQ #constraint-name {"10"},
		PERM #external-data {ANY_VALUE}}

SPECIFIC:

REQ	Object-class	{ OBJECT_CLASS_ID_OF(FootnoteReference) },
PERM	Content-generator	{ \$FNNUMBER },
PERM	Application-comments	{ REQ #constraint-name {"10"},
		PERM #external-data {ANY_VALUE}}

SPECIFIC_AND_GENERIC:

PERM	Layout-style	{ STYLE_ID_OF(LStyle10) },
PERM	Presentation-style	{ STYLE_ID_OF(PStyle1) },
PERM	Content-architecture-class	{ \$FC \$PC \$FPC }

}

7.3.3.12 FootnoteBody

:COMP-LOGICAL {

GENERIC:

REQ	Generator-for-subordinates	{ \$FootnoteBodyGFS },
REQ	Application-comments	{ REQ #constraint-name {"11"},
		PERM #external-data {ANY_VALUE}}

SPECIFIC:

REQ	Object-class	{ OBJECT_CLASS_ID_OF(FootnoteBody) },
REQ	Subordinates	{ SUB_ID_OF(FootnoteNumber),
		SUB_ID_OF(FootnoteText)+,
		SUB_ID_OF(Reference)+ },
PERM	Application-comments	{ REQ #constraint-name {"11"},
		PERM #external-data {ANY_VALUE}}

SPECIFIC_AND_GENERIC:

PERM	Layout-style	{ STYLE_ID_OF(LStyle11) }
------	--------------	---------------------------

}

7.3.3.13 FootnoteText

```
:BASIC-LOGICAL {  
GENERIC:  
    REQ    Application-comments    {REQ #constraint-name {"12"},  
                                     PERM #external-data {ANY_VALUE}}  
SPECIFIC:  
    REQ    Object-class            {OBJECT_CLASS_ID_OF(FootnoteText)},  
    PERM    Application-comments    {REQ #constraint-name {"12"},  
                                     PERM #external-data {ANY_VALUE}}  
SPECIFIC_AND_GENERIC:  
    PERM    Layout-style            {STYLE_ID_OF(LStyle6)},  
    PERM    Presentation-style      {STYLE_ID_OF(PStyle1)},  
    PERM    Content-architecture-class {$FC|$PC|$FPC},  
  
    PERM    Content-portions        {CONTENT_ID_OF(Character-content-portion)+}  
}
```

7.3.3.14 Figure

```
:COMP-LOGICAL {  
GENERIC:  
    REQ    Generator-for-subordinates {$FigureGFS},  
    REQ    Application-comments    {REQ #constraint-name {"13"},  
                                     PERM #external-data {ANY_VALUE}}  
SPECIFIC:  
    REQ    Object-class            {OBJECT_CLASS_ID_OF(Figure)},  
    REQ    Subordinates            {SUB_ID_OF(Number),  
                                    SUB_ID_OF(Caption),  
                                    SUB_ID_OF(Description),  
                                    SUB_ID_OF(Artwork),  
                                    SUB_ID_OF(Form)},  
  
    PERM    Application-comments    {REQ #constraint-name {"13"},  
                                     PERM #external-data {ANY_VALUE}}  
SPECIFIC_AND_GENERIC:  
    PERM    Layout-style            {STYLE_ID_OF(LStyle1)}  
}
```

7.3.3.15 BodyText

```
:BASIC-LOGICAL {  
GENERIC:  
    REQ    Application-comments    {REQ #constraint-name {"14"},  
                                     PERM #external-data {ANY_VALUE}}  
SPECIFIC:  
    REQ    Object-class            {OBJECT_CLASS_ID_OF(BodyText)},  
    PERM    Application-comments    {REQ #constraint-name {"14"},  
                                     PERM #external-data {ANY_VALUE}}  
SPECIFIC_AND_GENERIC:  
    PERM    Layout-style            {STYLE_ID_OF(LStyle2)},  
    PERM    Presentation-style      {STYLE_ID_OF(PStyle1)},  
    PERM    Content-architecture-class {$FC|$PC|$FPC},
```

PERM Content-portions {CONTENT_ID_OF(Character-content-portion)+}
 -- The attribute "content portions" must be specified either in the specific or generic part,
 otherwise the attribute "resource" must be specified. --

}

7.3.3.16 Reference

:COMP-LOGICAL {

GENERIC:

REQ Generator-for-subordinates {\$ReferenceGFS},
 REQ Application-comments {REQ #constraint-name {"15"},
 PERM #external-data {ANY_VALUE}}

SPECIFIC:

REQ Object-class {OBJECT_CLASS_ID_OF(Reference)},
 REQ Subordinates {SUB_ID_OF(BodyText)+,
 SUB_ID_OF(ReferencedContent)},
 PERM Application-comments {REQ #constraint-name {"15"},
 PERM #external-data {ANY_VALUE}}

SPECIFIC_AND_GENERIC:

PERM Layout-style {STYLE_ID_OF(LStyle1)}

}

7.3.3.17 ReferencedContent

:BASIC-LOGICAL {

GENERIC:

REQ Application-comments {REQ #constraint-name {"16"},
 PERM #external-data {ANY_VALUE}}

SPECIFIC:

REQ Object-class {OBJECT_CLASS_ID_OF(ReferencedContent)},
 PERM Content-generator {\$REF},
 PERM Content-portions {CONTENT_ID_OF(Character-content-portion)+,
 -- Either Content-generator or Content-portions is specified --
 PERM Application-comments {REQ #constraint-name {"16"},
 PERM #external-data {ANY_VALUE}}

SPECIFIC_AND_GENERIC:

PERM Layout-style {STYLE_ID_OF(LStyle10)},
 PERM Presentation-style {STYLE_ID_OF(PStyle1)},
 PERM Content-architecture-class {\$FC|\$PC|\$FPC}

}

7.3.3.18 BodyRaster

:BASIC-LOGICAL {

GENERIC:

REQ Content-architecture-class {\$FPR},
 REQ Application-comments {REQ #constraint-name {"17"},
 PERM #external-data {ANY_VALUE}}

SPECIFIC:

REQ Object-class {OBJECT_CLASS_ID_OF(BodyRaster)},
 PERM Content-architecture-class {\$FPR},
 PERM Application-comments {REQ #constraint-name {"17"},


```

                                PERM #external-data {ANY_VALUE}}
SPECIFIC_AND_GENERIC:
  PERM  Layout-style             {STYLE_ID_OF(LStyle5)},
  PERM  Presentation-style       {STYLE_ID_OF(PStyle3)},
  PERM  Content-portions        {CONTENT_ID_OF(Raster-graphics-content-portion)}
  -- The attribute "content portions" must be specified either in the specific or generic part,
  otherwise the attribute "resource" must be specified. --
}

```

7.3.3.19 BodyGeometric

```

:BASIC-LOGICAL {
GENERIC:
  REQ  Content-architecture-class {$FPG},
  REQ  Application-comments       {REQ #constraint-name {"18"},
                                PERM #external-data {ANY_VALUE}}

SPECIFIC:
  REQ  Object-class              {OBJECT_CLASS_ID_OF(BodyGeometric)},
  PERM Content-architecture-class {$FPG},
  PERM Application-comments       {REQ #constraint-name {"18"},
                                PERM #external-data {ANY_VALUE}}

SPECIFIC_AND_GENERIC:
  PERM Layout-style              {STYLE_ID_OF(LStyle5)},
  PERM Presentation-style        {STYLE_ID_OF(PStyle2)},
  PERM Content-portions          {CONTENT_ID_OF(Geometric-graphics-content-portion)}
  -- The attribute "content portions" must be specified either in the specific or generic part,
  otherwise the attribute "resource" must be specified. --
}

```

7.3.3.20 CommonContent

```

:ANY-COMMON {
GENERIC:
  REQ  Object-type               {'composite-logical-object'},
  REQ  Generator-for-subordinates {$CommonContentGFS},
  REQ  Application-comments       {REQ #constraint-name {"19"},
                                PERM #external-data {ANY_VALUE}},
  PERM Default-value-lists       {REQ #basic-logical-attributes
                                {PERM #presentation-style {ANY_VALUE},
                                PERM #layout-style {ANY_VALUE}}}
}

```

7.3.3.21 CommonText

```

:ANY-COMMON {
GENERIC:
  REQ  Object-type               {'basic-logical-object'},
  PERM Content-portions          {CONTENT_ID_OF(Character-content-portion)+},
  PERM Resource                  {ANY_VALUE},
  PERM Layout-style              {STYLE_ID_OF(LStyle3)},
  PERM Presentation-style        {STYLE_ID_OF(PStyle4)},
  PERM Content-architecture-class {$FC|$PC|$FPC},
  REQ  Application-comments       {REQ #constraint-name {"20"},
                                PERM #external-data {ANY_VALUE}}
}

```

-- Either the attribute "content portions" or "resource" must be specified in the above constituent constraint. --

}

7.3.3.22 CommonReference

:ANY-COMMON {

GENERIC:

REQ	Object-type	{'basic-logical-object'}.
PERM	Content-generator	{\$COMMONREF}.
PERM	Layout-style	{STYLE_ID_OF(LStyle3)}.
PERM	Presentation-style	{STYLE_ID_OF(PStyle4)}.
PERM	Content-architecture-class	{\$FC \$PC \$FPC}.
REQ	Application-comments	{REQ #constraint-name {"37"}, PERM #external-data {ANY_VALUE}}

}

7.3.3.23 CommonNumber

:ANY-COMMON {

GENERIC:

REQ	Object-type	{'basic-logical-object'}.
PERM	Content-generator	{\$COMMONNUMBER}.
PERM	Layout-style	{STYLE_ID_OF(LStyle3)}.
PERM	Presentation-style	{STYLE_ID_OF(PStyle4)}.
PERM	Content-architecture-class	{\$FC \$PC \$FPC}.
REQ	Application-comments	{REQ #constraint-name {"38"}, PERM #external-data {ANY_VALUE}}

}

7.3.3.24 CurrentInstance

:ANY-COMMON {

GENERIC:

REQ	Object-type	{'basic-logical-object'}.
PERM	Content-generator	{\$CURRENTINSTANCE}.
PERM	Layout-style	{STYLE_ID_OF(LStyle3)}.
PERM	Presentation-style	{STYLE_ID_OF(PStyle4)}.
PERM	Content-architecture-class	{\$FC \$PC \$FPC}.
REQ	Application-comments	{REQ #constraint-name {"39"}, PERM #external-data {ANY_VALUE}}

}

7.3.3.25 CommonRaster

:ANY-COMMON {

GENERIC:

REQ	Object-type	{'basic-logical-object'}.
PERM	Content-portions	{CONTENT_ID_OF(Raster-graphics-content-portion)}.
PERM	Resource	{ANY_VALUE}.
PERM	Layout-style	{STYLE_ID_OF(LStyle8)}.

```

PERM Presentation-style      {STYLE_ID_OF(PStyle3)},
REQ Content-architecture-class {$FPR},
REQ Application-comments     {REQ #constraint-name {"21"},
                             PERM #external-data {ANY_VALUE}}
-- Either the attribute "content portions" or "resource" must be specified in the above constituent
constraint. --

```

7.3.3.26 CommonGeometric

```

:ANY-COMMON {
GENERIC:
REQ Object-type              {'basic-logical-object'},
PERM Content-portions        {CONTENT_ID_OF(Geometric-graphics-content-portion)},
PERM Resource                 {ANY_VALUE},
PERM Layout-style             {STYLE_ID_OF(LStyle8)},
PERM Presentation-style       {STYLE_ID_OF(PStyle2)},
REQ Content-architecture-class {$FPG},
REQ Application-comments      {REQ #constraint-name {"22"},
                              PERM #external-data {ANY_VALUE}}
-- Either the attribute "content portions" or "resource" must be specified in the above constituent
constraint. --

```

7.3.3.27 PageNumber

```

:ANY-COMMON {
GENERIC:
REQ Object-type              {'basic-logical-object'},
PERM Content-generator        {$PGNUMBER},
PERM Layout-style             {STYLE_ID_OF(LStyle3)},
PERM Presentation-style       {STYLE_ID_OF(PStyle4)},
PERM Content-architecture-class {$FC|$PC|$FPC},
REQ Application-comments      {REQ #constraint-name {"40"},
                              PERM #external-data {ANY_VALUE}}

```

7.3.3.28 TableNumber

```

:ANY-COMMON {
GENERIC:
REQ Object-type              {'basic-logical-object'},
PERM Content-generator        {$TABLENUMBER},
PERM Layout-style             {STYLE_ID_OF(LStyle3)},
PERM Presentation-style       {STYLE_ID_OF(PStyle4)},
PERM Content-architecture-class {$FC|$PC|$FPC},
REQ Application-comments      {REQ #constraint-name {"44"},
                              PERM #external-data {ANY_VALUE}}

```

7.3.3.29 Description

```
:COMP-LOGICAL {  
  GENERIC:  
    REQ Generator-for-subordinates {$DescriptionGFS},  
    REQ Application-comments {REQ #constraint-name {"23"},  
                                PERM #external-data {ANY_VALUE}}  
  SPECIFIC:  
    REQ Object-class {OBJECT_CLASS_ID_OF(Description)},  
    REQ Subordinates {SUB_ID_OF(BodyText)+,  
                      SUB_ID_OF(Footnote)+,  
                      SUB_ID_OF(Phrase)+,  
                      SUB_ID_OF(Reference)+},  
    PERM Application-comments {REQ #constraint-name {"23"},  
                                PERM #external-data {ANY_VALUE}}  
  SPECIFIC_AND_GENERIC:  
    PERM Layout-style {STYLE_ID_OF(LStyle1)}  
}
```

7.3.3.30 Artwork

```
:COMP-LOGICAL {  
  GENERIC:  
    REQ Generator-for-subordinates {$ArtworkGFS},  
    REQ Application-comments {REQ #constraint-name {"24"},  
                                PERM #external-data {ANY_VALUE}}  
  SPECIFIC:  
    REQ Object-class {OBJECT_CLASS_ID_OF(Artwork)},  
    REQ Subordinates {SUB_ID_OF(BodyRaster)+,  
                      SUB_ID_OF(BodyGeometric)+,  
                      SUB_ID_OF(Phrase)+},  
    PERM Application-comments {REQ #constraint-name {"24"},  
                                PERM #external-data {ANY_VALUE}}  
  SPECIFIC_AND_GENERIC:  
    PERM Layout-style {STYLE_ID_OF(LStyle12)}  
}
```

7.3.3.31 NumberedList

```
:COMP-LOGICAL {  
  GENERIC:  
    REQ Generator-for-subordinates {$NumberedListGFS},  
    REQ Application-comments {REQ #constraint-name {"25"},  
                                PERM #external-data {ANY_VALUE}}  
  SPECIFIC:  
    REQ Object-class {OBJECT_CLASS_ID_OF(NumberedList)},  
    REQ Subordinates {SUB_ID_OF(Number)+,  
                      SUB_ID_OF(ListItem)+ },  
    PERM Application-comments {REQ #constraint-name {"25"},  
                                PERM #external-data {ANY_VALUE}}  
  SPECIFIC_AND_GENERIC:  
    PERM Layout-style {STYLE_ID_OF(LStyle1)}  
}
```


7.3.3.32 UnNumberedList

```
:COMP-LOGICAL {  
  GENERIC:  
    REQ  Generator-for-subordinates  {$UnNumberedListGFS},  
    REQ  Application-comments        {REQ #constraint-name {"26"},  
                                     PERM #external-data {ANY_VALUE}}  
  
  SPECIFIC:  
    REQ  Object-class                {OBJECT_CLASS_ID_OF(UnNumberedList)},  
    REQ  Subordinates                {SUB_ID_OF(BodyText)+,  
                                     SUB_ID_OF(BodyRaster)+,  
                                     SUB_ID_OF(BodyGeometric)+,  
                                     SUB_ID_OF(ListItem)+ },  
    PERM Application-comments        {REQ #constraint-name {"26"},  
                                     PERM #external-data {ANY_VALUE}}  
  
  SPECIFIC_AND_GENERIC:  
    PERM Layout-style                {STYLE_ID_OF(LStyle1)}  
}
```

7.3.3.33 DefinitionList

```
:COMP-LOGICAL {  
  GENERIC:  
    REQ  Generator-for-subordinates  {$DefinitionListGFS},  
    REQ  Application-comments        {REQ #constraint-name {"27"},  
                                     PERM #external-data {ANY_VALUE}}  
  
  SPECIFIC:  
    REQ  Object-class                {OBJECT_CLASS_ID_OF(DefinitionList)},  
    REQ  Subordinates                {SUB_ID_OF(ListTerm)+,  
                                     SUB_ID_OF(ListItem)+ },  
    PERM Application-comments        {REQ #constraint-name {"27"},  
                                     PERM #external-data {ANY_VALUE}}  
  
  SPECIFIC_AND_GENERIC:  
    PERM Layout-style                {STYLE_ID_OF(LStyle1)}  
}
```

7.3.3.34 ListItem

```
:COMP-LOGICAL {  
  GENERIC:  
    REQ  Generator-for-subordinates  {$ListItemGFS},  
    REQ  Application-comments        {REQ #constraint-name {"28"},  
                                     PERM #external-data {ANY_VALUE}}  
  
  SPECIFIC:  
    REQ  Object-class                {OBJECT_CLASS_ID_OF(ListItem)},  
    REQ  Subordinates                {SUB_ID_OF(Phrase)+,  
                                     SUB_ID_OF(NumberedList)+,  
                                     SUB_ID_OF(UnNumberedList)+,  
                                     SUB_ID_OF(DefinitionList)+ },  
    PERM Application-comments        {REQ #constraint-name {"28"},  
                                     PERM #external-data {ANY_VALUE}}  
  
  SPECIFIC_AND_GENERIC:
```

```

    PERM Layout-style {STYLE_ID_OF(LStyle1)}
}

```

7.3.3.35 ListTerm

```

:COMP-LOGICAL {
GENERIC:
    REQ Generator-for-subordinates {$ListTermGFS},
    REQ Application-comments {REQ #constraint-name {"29"},
        PERM #external-data {ANY_VALUE}}

SPECIFIC:
    REQ Object-class {OBJECT_CLASS_ID_OF(ListTerm)},
    REQ Subordinates {SUB_ID_OF(BodyText)+,
        SUB_ID_OF(BodyRaster)+,
        SUB_ID_OF(BodyGeometric)+,
        SUB_ID_OF(Reference)+,
        SUB_ID_OF(Phrase)+,
        SUB_ID_OF(Footnote)+ },
    PERM Application-comments {REQ #constraint-name {"29"},
        PERM #external-data {ANY_VALUE}}

SPECIFIC_AND_GENERIC:
    PERM Layout-style {STYLE_ID_OF(LStyle1)}
}

```

7.3.3.36 Table

```

:COMP-LOGICAL {
GENERIC:
    REQ Generator-for-subordinates {$TableGFS},
    REQ Application-comments {REQ #constraint-name {"30"},
        PERM #external-data {ANY_VALUE}},
    REQ Layout-style {STYLE_ID_OF(LStyleT4)}

SPECIFIC:
    REQ Object-class {OBJECT_CLASS_ID_OF(Table)},
    REQ Subordinates {SUB_ID_OF(Row)+},
    PERM Application-comments {REQ #constraint-name {"30"},
        PERM #external-data {ANY_VALUE}},
    PERM Layout-style {STYLE_ID_OF(LStyleT8)}
}

```

7.3.3.37 Row

```

:COMP-LOGICAL {
GENERIC:
    REQ Generator-for-subordinates {$RowGFS},
    REQ Application-comments {REQ #constraint-name {"31"},
        PERM #external-data {ANY_VALUE}},
    REQ Layout-style {STYLE_ID_OF(LStyleT5)}

SPECIFIC:
    REQ Object-class {OBJECT_CLASS_ID_OF(Row)},
    REQ Subordinates {SUB_ID_OF(EntryElement)+,
        SUB_ID_OF(TableComponent)},
}

```

```

    PERM Application-comments {REQ #constraint-name {"31"},
                                PERM #external-data {ANY_VALUE}}
}

```

7.3.3.38 TableComponent

```

:COMP-LOGICAL {
GENERIC:
    REQ Generator-for-subordinates {$TableComponentGFS},
    REQ Application-comments {REQ #constraint-name {"32"},
                                PERM #external-data {ANY_VALUE}},
    REQ Layout-style {STYLE_ID_OF(LStyleT6)}
SPECIFIC:
    REQ Object-class {OBJECT_CLASS_ID_OF(TableComponent)},
    REQ Subordinates {SUB_ID_OF(RowComponent)+},
    PERM Application-comments {REQ #constraint-name {"32"},
                                PERM #external-data {ANY_VALUE}}
}

```

7.3.3.39 RowComponent

```

:COMP-LOGICAL {
GENERIC:
    REQ Generator-for-subordinates {$RowComponentGFS},
    REQ Application-comments {REQ #constraint-name {"33"},
                                PERM #external-data {ANY_VALUE}},
    REQ Layout-style {STYLE_ID_OF(LStyleT7)}
SPECIFIC:
    REQ Object-class {OBJECT_CLASS_ID_OF(RowComponent)},
    REQ Subordinates {SUB_ID_OF(EntryElement)+},
    PERM Application-comments {REQ #constraint-name {"33"},
                                PERM #external-data {ANY_VALUE}}
}

```

7.3.3.40 Form

```

:COMP-LOGICAL {
GENERIC:
    REQ Generator-for-subordinates {$FormGFS},
    REQ Application-comments {REQ #constraint-name {"34"},
                                PERM #external-data {ANY_VALUE}},
    REQ Layout-style {STYLE_ID_OF(LStyleT1)}
SPECIFIC:
    REQ Object-class {OBJECT_CLASS_ID_OF(Form)},
    REQ Subordinates {SUB_ID_OF(EntryElement)+,
                                SUB_ID_OF(EntryGroup)+},
    PERM Application-comments {REQ #constraint-name {"34"},
                                PERM #external-data {ANY_VALUE}}
}

```

7.3.3.41 EntryElement

```
:COMP-LOGICAL {  
GENERIC:  
  REQ Generator-for-subordinates {$EntryElementGFS},  
  REQ Application-comments      {REQ #constraint-name {"35"},  
                                PERM #external-data {ANY_VALUE}},  
  REQ Layout-style              {STYLE_ID_OF(LStyleT2))}  
SPECIFIC:  
  REQ Object-class              {OBJECT_CLASS_ID_OF(EntryElement)),  
  REQ Subordinates              {SUB_ID_OF(EntryText),  
                                SUB_ID_OF(EntryRaster),  
                                SUB_ID_OF(EntryGeometric)),  
  PERM Application-comments     {REQ #constraint-name {"35"},  
                                PERM #external-data {ANY_VALUE}}  
}
```

7.3.3.42 EntryGroup

```
:COMP-LOGICAL {  
GENERIC:  
  REQ Generator-for-subordinates {$EntryGroupGFS},  
  REQ Application-comments      {REQ #constraint-name {"36"},  
                                PERM #external-data {ANY_VALUE}}  
SPECIFIC:  
  REQ Object-class              {OBJECT_CLASS_ID_OF(EntryGroup)),  
  REQ Subordinates              {SUB_ID_OF(EntryElement)+,  
                                SUB_ID_OF(EntryGroup)+},  
  PERM Application-comments     {REQ #constraint-name {"36"},  
                                PERM #external-data {ANY_VALUE}}  
SPECIFIC_AND_GENERIC:  
  PERM Layout-style             {STYLE_ID_OF(LStyleT3))}  
}
```

7.3.3.43 EntryText

```
:BASIC-LOGICAL {  
GENERIC:  
  REQ Application-comments      {REQ #constraint-name {"41"},  
                                PERM #external-data {ANY_VALUE}}  
SPECIFIC:  
  REQ Object-class              {OBJECT_CLASS_ID_OF(EntryText)),  
  PERM Application-comments     {REQ #constraint-name {"41"},  
                                PERM #external-data {ANY_VALUE}}  
SPECIFIC_AND_GENERIC:  
  PERM Layout-style             {STYLE_ID_OF(LStyleT9)),  
  PERM Presentation-style       {STYLE_ID_OF(PStyle1)),  
  PERM Content-architecture-class {$FC|$PC|$FPC},  
  PERM Content-portions         {CONTENT_ID_OF(Character-content-portion)+}  
  -- The attribute "content portions" must be specified either in the specific or generic part,  
  otherwise the attribute "resource" must be specified. --  
}
```


7.3.3.44 EntryRaster

:BASIC-LOGICAL {

GENERIC:

REQ Content-architecture-class {\$FPR},
REQ Application-comments {REQ #constraint-name {"42"},
PERM #external-data {ANY_VALUE}}

SPECIFIC:

REQ Object-class {OBJECT_CLASS_ID_OF(EntryRaster)},
PERM Content-architecture-class {\$FPR},
PERM Application-comments {REQ #constraint-name {"42"},
PERM #external-data {ANY_VALUE}}

SPECIFIC_AND_GENERIC:

PERM Layout-style {STYLE_ID_OF(LStyleT9)},
PERM Presentation-style {STYLE_ID_OF(PStyle3)},
PERM Content-portions {CONTENT_ID_OF(Raster-graphics-content-portion)}

-- The attribute "content portions" must be specified either in the specific or generic part,
otherwise the attribute "resource" must be specified. --

}

7.3.3.45 EntryGeometric

:BASIC-LOGICAL {

GENERIC:

REQ Content-architecture-class {\$FPG},
REQ Application-comments {REQ #constraint-name {"43"},
PERM #external-data {ANY_VALUE}}

SPECIFIC:

REQ Object-class {OBJECT_CLASS_ID_OF(EntryGeometric)},
PERM Content-architecture-class {\$FPG},
PERM Application-comments {REQ #constraint-name {"43"},
PERM #external-data {ANY_VALUE}}

SPECIFIC_AND_GENERIC:

PERM Layout-style {STYLE_ID_OF(LStyleT9)},
PERM Presentation-style {STYLE_ID_OF(PStyle2)},
PERM Content-portions {CONTENT_ID_OF(Geometric-graphics-content-portion)}

-- The attribute "content portions" must be specified either in the specific or generic part,
otherwise the attribute "resource" must be specified. --

}

7.4 Layout constituent constraints

7.4.1 Macro definitions

DEFINE(DocLayRootGFS, "

<construction-expr> ::= <construction-term>|<construction-type>;

<construction-term> ::= <construction-factor>
| OPT <construction-factor>
| REP <construction-factor>
| OPT REP <construction-factor>;

<construction-type> ::= SEQ(<construction-term>...)

```

        | CHO(<construction-term>...);

<construction-factor> ::= OBJECT_CLASS_ID_OF(PageSet)
        | <construction-type>;

")

DEFINE(PageSetGFS, "
<construction-expr> ::= <construction-term>|<construction-type>;

<construction-term> ::= <construction-factor>
        | OPT <construction-factor>
        | REP <construction-factor>
        | OPT REP <construction-factor>;

<construction-type> ::= SEQ(<construction-term>...)
        | CHO(<construction-term>...);

<construction-factor> ::= OBJECT_CLASS_ID_OF(Page)
        | OBJECT_CLASS_ID_OF(RectoPage)
        | OBJECT_CLASS_ID_OF(VersoPage)
        | <construction-type>;

")

DEFINE(PageGFS, "
<construction-expr> ::= SEQ([<headerarea>]<bodyarea>[<footerarea>])
        | SEQ(<bodyarea>[<headerarea>][<footerarea>])
        | SEQ([<headerarea>][<footerarea>]<bodyarea>)
        | <bodyarea>;

<headerarea> ::= OBJECT_CLASS_ID_OF(BasicHeader)
        | OBJECT_CLASS_ID_OF(CompositeHeader);

<bodyarea> ::= OBJECT_CLASS_ID_OF(VariableCompositeBody)
        | OBJECT_CLASS_ID_OF(FixedCompositeBody)
        | OBJECT_CLASS_ID_OF(BasicBody);

<footerarea> ::= OBJECT_CLASS_ID_OF(BasicFooter)
        | OBJECT_CLASS_ID_OF(CompositeFooter);

")

DEFINE(CompositeCommonGFS, "
<construction-expr> ::= <fixed-common-content-frames>
        | <variable-common-content-frames>;

<fixed-common-content-frames>
        ::= SEQ((OBJECT_CLASS_ID_OF(SourcedContentFixed)
        | OBJECT_CLASS_ID_OF(ArrangedContentFixed))...);

<variable-common-content-frames>
        ::= SEQ((OBJECT_CLASS_ID_OF(SourcedContentVariable)
        | OBJECT_CLASS_ID_OF(ArrangedContentVariable))...);

")

```

DEFINE(HeaderFooterGFS, "\$CompositeCommonGFS")

DEFINE(FixedCompositeBodyGFS, "

<construction-expr> ::= SEQ(<construction-term>...);

<construction-term> ::= <construction-factor1>
| OPT <construction-factor1>
| CHO({<construction-factor1>}...)
| <construction-factor2>;

<construction-factor1> ::= OBJECT_CLASS_ID_OF(BasicFixture)
| OBJECT_CLASS_ID_OF(ColumnFixed)
| OBJECT_CLASS_ID_OF(CompositeFixtureFixed)
| OBJECT_CLASS_ID_OF(VariableCompositeBody);

<construction-factor2> ::= OBJECT_CLASS_ID_OF(CompositeCommon)
| OBJECT_CLASS_ID_OF(SourcedContentFixed)
| OBJECT_CLASS_ID_OF(ArrangedContentFixed);

")

DEFINE(VariableCompositeBodyGFS, "

<construction-expr> ::= <construction-term>|<construction-type>
| SEQ(<construction-term>, <construction-footnote>)
| SEQ(<construction-type>, <construction-footnote>);

<construction-term> ::= <construction-factor1>
| OPT <construction-factor1>
| REP <construction-factor1>
| OPT REP <construction-factor1>;

<construction-type> ::= SEQ({<construction-term> | <construction-factor2>}...)
| CHO({<construction-term>}...);

<construction-factor1> ::= OBJECT_CLASS_ID_OF(BasicFloat)
| OBJECT_CLASS_ID_OF(SnakingColumns)
| OBJECT_CLASS_ID_OF(SynchronizedColumns)
| OBJECT_CLASS_ID_OF(CompositeFloat)
| OBJECT_CLASS_ID_OF(CompositeFixtureVariable)
| OBJECT_CLASS_ID_OF(TableArea)
| OBJECT_CLASS_ID_OF(FooterArea)
| <construction-type>;

<construction-footnote> ::= OBJECT_CLASS_ID_OF(FooterArea)
| OPT OBJECT_CLASS_ID_OF(FooterArea);

<construction-factor2> ::= OBJECT_CLASS_ID_OF(ArrangedContentVariable)
| OBJECT_CLASS_ID_OF(SourcedContentVariable);

")

DEFINE(SnakingColumnsGFS, "

<construction-expr> ::= REP <construction-factor1>

```

| <construction-term>
| SEQ(<construction-type>...);
<construction-term> ::= SEQ(<construction-type><following-term>);
<following-term> ::= OPT<construction-factor1>
| <construction-factor2>
| OPT<construction-term> ;

<construction-type> ::= <construction-factor1>|<construction-factor2>;
<construction-factor1> ::= OBJECT_CLASS_ID_OF(ColumnVariable)
| OBJECT_CLASS_ID_OF(CompositeColumnVariable);
<construction-factor2> ::= OBJECT_CLASS_ID_OF(ArrangedContentVariable)
| OBJECT_CLASS_ID_OF(SourcedContentVariable);
")

```

```

DEFINE(SynchronizedColumnsGFS, "
<construction-expr> ::= SEQ({<construction-type>}...)|<construction-term>;

<construction-term> ::= SEQ(<construction-type><following-term>);
<following-term> ::= OPT<construction-factor1>
| <construction-factor2>
| OPT<construction-term> ;

<construction-type> ::= <construction-factor1>|<construction-factor2>;
<construction-factor1> ::= OBJECT_CLASS_ID_OF(ColumnFixed)
| OBJECT_CLASS_ID_OF(CompositeColumnFixed);
<construction-factor2> ::= OBJECT_CLASS_ID_OF(ArrangedContentFixed)
| OBJECT_CLASS_ID_OF(SourcedContentFixed);
")

```

```

DEFINE(CompositeFloatGFS, "
<construction-expr> ::= SEQ(<construction-term1> [<construction-term2>]...);

<construction-term1> ::= <construction-factor1>|<construction-factor2>;
<construction-term2> ::= <construction-term1>
| OPT<construction-factor1>;
<construction-factor1> ::= OBJECT_CLASS_ID_OF(BasicColumn)
| OBJECT_CLASS_ID_OF(CompositeFixtureVariable)
| OBJECT_CLASS_ID_OF(TableArea);
<construction-factor2> ::= OBJECT_CLASS_ID_OF(ArrangedContentVariable)
| OBJECT_CLASS_ID_OF(SourcedContentVariable);
")

```

```

DEFINE(CompositeColumnGFS, "
<construction-expr> ::= <construction-term>
| <construction-type>
| SEQ(<construction-term> <construction-footnote>)
| SEQ(<construction-type> <construction-footnote>);

<construction-term> ::= <construction-factor1>
| OPT <construction-factor1>
| REP <construction-factor1>
| OPT REP <construction-factor1>;

```



```
<construction-type> ::= SEQ({<construction-term> | <construction-factor2>}...)
                        | CHO({<construction-term>}...);
```

```
<construction-factor1> ::= OBJECT_CLASS_ID_OF(BasicFloat)
                        | OBJECT_CLASS_ID_OF(TableArea)
                        | OBJECT_CLASS_ID_OF(CompositeFloat)
                        | OBJECT_CLASS_ID_OF(CompositeFixtureVariable)
                        | OBJECT_CLASS_ID_OF(FootnoteArea)
                        | <construction-type>;
```

```
<construction-footnote> ::= OBJECT_CLASS_ID_OF(FootnoteArea)
                        | OPT OBJECT_CLASS_ID_OF(FootnoteArea);
```

```
<construction-factor2> ::= OBJECT_CLASS_ID_OF(ArrangedContentVariable)
                        | OBJECT_CLASS_ID_OF(SourcedContentVariable);
```

")

```
DEFINE(CompositeColumnVariableGFS, "$CompositeColumnGFS")
```

```
DEFINE(CompositeColumnFixedGFS, "$CompositeColumnGFS")
```

```
DEFINE(CompositeFixtureGFS, "
```

```
<construction-expr> ::= <construction-factor>
                        | REP CHO(<construction-factor>...);
```

```
<construction-factor> ::= OBJECT_CLASS_ID_OF(BasicFloat)
                        | OBJECT_CLASS_ID_OF(FootnoteArea)
                        | OBJECT_CLASS_ID_OF(CompositeArtwork)
                        | OBJECT_CLASS_ID_OF(FormArea);
```

")

```
DEFINE(CompositeFixtureFixedGFS, "$CompositeFixtureGFS")
```

```
DEFINE(CompositeFixtureVariableGFS, "$CompositeFixtureGFS")
```

```
DEFINE(CompositeArtworkGFS, "
```

```
<construction-expr> ::= REP OBJECT_CLASS_ID_OF(BasicFixture);
```

")

```
DEFINE(TableAreaGFS, "
```

```
<construction-expr> ::= <row-area>
                        | SEQ([<table-header>] [<table-label>] <row-area> [<table-label>]);
```

```
<table-header> ::= OBJECT_CLASS_ID_OF(TableHeader);
```

```
<table-label> ::= OBJECT_CLASS_ID_OF(TableLabel);
```

```
<row-area> ::= REP OBJECT_CLASS_ID_OF(RowArea)
            | REP CHO(OBJECT_CLASS_ID_OF(RowArea)...);
```

")

```
DEFINE(RowAreaGFS,"
```

```

<construction-expr> ::= SEQ(OBJECT_CLASS_ID_OF(Cell)...)
                      | SEQ(OBJECT_CLASS_ID_OF(Cell)
                          OBJECT_CLASS_ID_OF(SubRowGroup));
")

DEFINE(SubRowGroupGFS,"
<construction-expr> ::= REP OBJECT_CLASS_ID_OF(SubRow);
")

DEFINE(SubRowGFS,"
<construction-expr> ::= SEQ(OBJECT_CLASS_ID_OF(Cell)...);
")

DEFINE(TableHeaderGFS,"
<construction-expr> ::= SEQ(OBJECT_CLASS_ID_OF(SourcedContentFixed)...);
")

DEFINE(TableLabelGFS,"
<construction-expr> ::= SEQ(OBJECT_CLASS_ID_OF(TableLabelContent)...)
                      | SEQ(OBJECT_CLASS_ID_OF(TableLabelContent)
                          OBJECT_CLASS_ID_OF(CompositeTableLabel));
")

DEFINE(CompositeTableLabelGFS,"
<construction-expr> ::= SEQ(OBJECT_CLASS_ID_OF(LabelComponent)...);
")

DEFINE(LabelComponentGFS,"
<construction-expr> ::= SEQ(OBJECT_CLASS_ID_OF(TableLabelContent)...);
")

DEFINE(FormAreaGFS,"
<construction-expr> ::= AGG(<factor>...);
<factor>              ::= OBJECT_CLASS_ID_OF(ArrangedContentFixed)
                      | OBJECT_CLASS_ID_OF(FormEntryArea)
                      | OBJECT_CLASS_ID_OF(EntryGroupArea);
")

DEFINE(EntryGroupAreaGFS,"$FormAreaGFS")

```

7.4.2 Factor constraints

7.4.2.1 FACTOR ANY-LAYOUT

```

{
GENERIC:

```

```

REQ Object-type {VIRTUAL},
REQ Object-class-identifier {ANY_VALUE}
SPECIFIC:
  PERM Object-type {VIRTUAL},
  REQ Object-identifier {ANY_VALUE},
  CASE $DAC OF {
    $FDA: PERM Object-class {VIRTUAL}
    $FPDA: REQ Object-class {VIRTUAL}
  },
  REQ Subordinates {VIRTUAL}
SPECIFIC_AND_GENERIC:
  PERM User-readable-comments {ANY_STRING},
  PERM User-visible-name {ANY_STRING}
}

```

7.4.2.2 FACTOR ANY-PAGE

```

:ANY-LAYOUT {
GENERIC:
  REQ Object-type {'page'},

  CASE $DAC OF {
    $PDA-FPDA:
      REQ Generator-for-subordinates {$PageGFS},
      PERM Bindings {PMUL {$SPECIFYBINDINGS},
        PERM {$INITIALISEPGNUMBER,$USEPGNUMBERS}}
  }
SPECIFIC:
  PERM Object-type {'page'},
  REQ Subordinates {SUB_ID_OF(BasicHeader),
    SUB_ID_OF(CompositeHeader),
    SUB_ID_OF(BasicBody),
    SUB_ID_OF(FixedCompositeBody),
    SUB_ID_OF(VariableCompositeBody),
    SUB_ID_OF(BasicFooter),
    SUB_ID_OF(CompositeFooter)}
SPECIFIC_AND_GENERIC:
  PERM Dimensions {$PermissiblePageDimensions},
  PERM Transparency {ANY_VALUE},
  PERM Colour {ANY_VALUE},
  PERM Page-position {ANY_VALUE}
}

```

7.4.2.3 FACTOR ANY-FRAME-FIXED

```

:ANY-LAYOUT {
GENERIC:
  REQ Object-type {'frame'}
SPECIFIC:
  PERM Object-type {'frame'}
SPECIFIC_AND_GENERIC:
  PERM Position {REQ #fixed-position
    {REQ #horizontal-position {ANY_VALUE},

```

```

        REQ #vertical-position      {ANY_VALUE}}},
    PERM  Dimensions                {REQ #horizontal-dimension
        {REQ #fixed-dimension      {ANY_VALUE}},
        REQ #vertical-dimension    {ANY_VALUE}}},
    PERM  Transparency              {ANY_VALUE},
    PERM  Colour                    {ANY_VALUE},
    PERM  Border                     {ANY_VALUE}
}

```

7.4.2.4 FACTOR ANY-FRAME-VARIABLE

```

:ANY-LAYOUT {
GENERIC:
    REQ  Object-type                {'frame'}
SPECIFIC:
    PERM Object-type                {'frame'},
    CASE $DAC OF {
        $FPDA:
            REQ Position             {REQ #fixed-position
                {REQ #horizontal-position {ANY_VALUE},
                REQ #vertical-position   {ANY_VALUE}}},
            REQ Dimensions           {REQ #horizontal-dimension
                {REQ #fixed-dimension {ANY_VALUE}},
                REQ #vertical-dimension {REQ #fixed-dimension {ANY_VALUE}}}
        }
    SPECIFIC_AND_GENERIC:
        CASE $DAC OF {
            $FDA:
                PERM Position         {REQ #fixed-position
                    {REQ #horizontal-position {ANY_VALUE},
                    REQ #vertical-position   {ANY_VALUE}}},
                PERM Dimensions       {REQ #horizontal-dimension
                    {REQ #fixed-dimension {ANY_VALUE}},
                    REQ #vertical-dimension {REQ #fixed-dimension {ANY_VALUE}}}
            },
            PERM Transparency         {ANY_VALUE},
            PERM Colour                {ANY_VALUE},
            PERM Border                {ANY_VALUE}
        }
    }
}

```


7.4.3 Constituent constraints

7.4.3.1 DocumentLayoutRoot

```
:ANY-LAYOUT {
GENERIC:
  REQ Object-type                {'document-layout-root'},
  CASE $DAC OF {
    $PDA-FPDA: .
      REQ Generator-for-subordinates {$DocLayRootGFS},
      PERM Bindings                 {PMUL {$SPECIFYBINDINGS},
                                     PERM {$INITIALISEPGNUMBER}}
    },
  REQ Application-comments        {REQ #constraint-name {"0"},
                                    PERM #external-data {ANY_VALUE}}
SPECIFIC:
  PERM Object-type                {'document-layout-root'},
  CASE $DAC OF {
    $FDA:
      PERM Object-class            {OBJECT_CLASS_ID_OF
                                     (DocumentLayoutRoot))
    $FPDA:
      REQ Object-class             {OBJECT_CLASS_ID_OF
                                     (DocumentLayoutRoot))
    },
  REQ Subordinates                {SUB_ID_OF(PageSet)+},
  PERM Application-comments        {REQ #constraint-name {"0"},
                                    PERM #external-data {ANY_VALUE}}
}
```

7.4.3.2 PageSet

```
:ANY-LAYOUT {
GENERIC:
  REQ Object-type                {'page-set'},

  CASE $DAC OF {
    $PDA-FPDA:
      REQ Generator-for-subordinates {$PageSetGFS},
      PERM Bindings                 {PMUL {$SPECIFYBINDINGS},
                                     PERM {$INITIALISEPGNUMBER}}
    },
  REQ Application-comments        {REQ #constraint-name {"1"},
                                    PERM #external-data {ANY_VALUE}}
SPECIFIC:
  PERM Object-type                {'page-set'},
  CASE $DAC OF {
    $FDA:
      PERM Object-class            {OBJECT_CLASS_ID_OF(PageSet)}
    $FPDA:
      REQ Object-class             {OBJECT_CLASS_ID_OF(PageSet)}
  }
}
```

```

    },
    REQ Subordinates {SUB_ID_OF(Page)+,
                      SUB_ID_OF(RectoPage)+,
                      SUB_ID_OF(VersoPage)+},
    PERM Application-comments {REQ #constraint-name {"1"},
                              PERM #external-data {ANY_VALUE}}
}

```

7.4.3.3 Page

```

:ANY-PAGE {
GENERIC:
    REQ Application-comments {REQ #constraint-name {"2"},
                              PERM #external-data {ANY_VALUE}}

SPECIFIC:
    CASE $DAC OF {
        $FDA:
            PERM Object-class {OBJECT_CLASS_ID_OF(Page)}
        $FPDA:
            REQ Object-class {OBJECT_CLASS_ID_OF(Page)}
    },
    PERM Application-comments {REQ #constraint-name {"2"},
                              PERM #external-data {ANY_VALUE}}

SPECIFIC_AND_GENERIC:
    PERM Medium-type {PERM #nominal-page-size {$NominalPageSizes},
                      PERM #side-of-sheet {ANY_VALUE}}
}

```

7.4.3.4 RectoPage

```

:ANY-PAGE {
GENERIC:
    REQ Application-comments {REQ #constraint-name {"3"},
                              PERM #external-data {ANY_VALUE}},
    REQ Medium-type {PERM #nominal-page-size {$NominalPageSizes},
                     REQ #side-of-sheet {'recto'|'unspecified'}}

SPECIFIC:
    CASE $DAC OF {
        $FDA:
            PERM Object-class {OBJECT_CLASS_ID_OF(RectoPage)}
        $FPDA:
            REQ Object-class {OBJECT_CLASS_ID_OF(RectoPage)}
    },
    PERM Application-comments {REQ #constraint-name {"3"},
                              PERM #external-data {ANY_VALUE}},
    PERM Medium-type {PERM #nominal-page-size {$NominalPageSizes},
                      PERM #side-of-sheet {'recto'|'unspecified'}}
}

```

4.3.5 VersoPage

ANY-PAGE {

GENERIC:

REQ	Application-comments	{REQ #constraint-name {"4"}, PERM #external-data {ANY_VALUE}},
REQ	Medium-type	{PERM #nominal-page-size {\$NominalPageSizes}, REQ #side-of-sheet {'verso' 'unspecified'}}

SPECIFIC:

CASE \$DAC OF {

\$FDA:

PERM Object-class {OBJECT_CLASS_ID_OF(VersoPage)}

\$FPDA:

REQ Object-class {OBJECT_CLASS_ID_OF(VersoPage)}

},

PERM	Application-comments	{REQ #constraint-name {"4"}, PERM #external-data {ANY_VALUE}},
------	----------------------	---

PERM	Medium-type	{PERM #nominal-page-size {\$NominalPageSizes}, PERM #side-of-sheet {'verso' 'unspecified'}}
------	-------------	--

7.4.3.6 CompositeHeader

ANY-FRAME-FIXED {

GENERIC:

CASE \$DAC OF {

\$PDA-FPDA:

REQ Generator-for-subordinates {\$HeaderFooterGFS}},

PERM	Layout-path	{'270-degrees' -- H/F layouts A1,B2 -- '180-degrees' -- H/F layout B1 -- '0-degrees' -- H/F layout A2 --},
------	-------------	--

REQ	Application-comments	{REQ #constraint-name {"5"}, PERM #external-data {ANY_VALUE}}
-----	----------------------	--

SPECIFIC:

CASE \$DAC OF {

\$FDA:

PERM Object-class {OBJECT_CLASS_ID_OF(CompositeHeader)}

\$FPDA:

REQ Object-class {OBJECT_CLASS_ID_OF(CompositeHeader)}

},

REQ	Subordinates	{SUB_ID_OF(SourcedContentFixed)+, SUB_ID_OF(ArrangedContentFixed)+, SUB_ID_OF(SourcedContentVariable)+, SUB_ID_OF(ArrangedContentVariable)+},
-----	--------------	--

PERM	Imaging-order	{SUB_ID_OF(SourcedContentFixed)+, SUB_ID_OF(ArrangedContentFixed)+},
------	---------------	---

PERM	Application-comments	{REQ #constraint-name {"5"}, PERM #external-data {ANY_VALUE}}
------	----------------------	--

7.4.3.7 CompositeFooter

```
:ANY-FRAME-FIXED {
GENERIC:
  CASE $DAC OF {
    $PDA-FPDA:
      REQ Generator-for-subordinates    {$HeaderFooterGFS}},
    PERM Layout-path                    {'270-degrees' -- H/F layouts A1,B2 --
                                         {'180-degrees' -- H/F layout B1 --
                                         {'0-degrees' -- H/F layout A2 --},
      REQ Application-comments          {REQ #constraint-name {"32"},
                                         PERM #external-data {ANY_VALUE}}
  },
SPECIFIC:
  CASE $DAC OF {
    $FDA:
      PERM Object-class                 {OBJECT_CLASS_ID_OF(CompositeFooter)}
    $FPDA:
      REQ Object-class                 {OBJECT_CLASS_ID_OF(CompositeFooter)}
  },
  REQ Subordinates                     {SUB_ID_OF(SourcedContentFixed)+,
                                         SUB_ID_OF(ArrangedContentFixed)+,
                                         SUB_ID_OF(SourcedContentVariable)+,
                                         SUB_ID_OF(ArrangedContentVariable)+},
  PERM Imaging-order                   {SUB_ID_OF(SourcedContentFixed)+,
                                         SUB_ID_OF(ArrangedContentFixed)+},
  PERM Application-comments            {REQ #constraint-name {"32"},
                                         PERM #external-data {ANY_VALUE}}
}
```

7.4.3.8 FixedCompositeBody

```
:ANY-FRAME-FIXED {
GENERIC:
  CASE $DAC OF {
    $PDA-FPDA:
      REQ Generator-for-subordinates    {$FixedCompositeBodyGFS},
      PERM Layout-path                  {'270-degrees' -- body layout A --
                                         {'0-degrees' -- body layout B --
                                         {'180-degrees' -- body layout C --}
  },
  REQ Application-comments              {REQ #constraint-name {"6"},
                                         PERM #external-data {ANY_VALUE}}
SPECIFIC:
  CASE $DAC OF {
    $FDA:
      PERM Object-class                 {OBJECT_CLASS_ID_OF
                                         (FixedCompositeBody)}
    $FPDA:
      REQ Object-class                 {OBJECT_CLASS_ID_OF
                                         (FixedCompositeBody)}
  },
  REQ Subordinates                     {SUB_ID_OF(CompositeCommon)+,
                                         SUB_ID_OF(BasicFixture)+,
                                         SUB_ID_OF(ColumnFixed)+,
```


PERM	Imaging-order	SUB_ID_OF(CompositeFixtureFixed)+, SUB_ID_OF(VariableCompositeBody)+, SUB_ID_OF(SourcedContentFixed)+, SUB_ID_OF(ArrangedContentFixed)+}, {SUB_ID_OF(CompositeCommon)+, SUB_ID_OF(BasicFixture)+, SUB_ID_OF(ColumnFixed)+, SUB_ID_OF(CompositeFixtureFixed)+, SUB_ID_OF(VariableCompositeBody)+, SUB_ID_OF(SourcedContentFixed)+, SUB_ID_OF(ArrangedContentFixed)+}, {REQ #constraint-name {"6"}, PERM #external-data {ANY_VALUE}}
PERM	Application-comments	

4.3.9 VariableCompositeBody

ANY-FRAME-FIXED {

GENERIC:

CASE \$DAC OF {
\$PDA-FPDA:

REQ	Generator-for-subordinates	{ \$VariableCompositeBodyGFS},
PERM	Layout-path	{ '270-degrees' -- body layout A -- '0-degrees' -- body layout B -- '180-degrees' -- body layout C --}

},

REQ	Application-comments	{REQ #constraint-name {"7"}, PERM #external-data {ANY_VALUE}}
-----	----------------------	--

SPECIFIC:

CASE \$DAC OF {
\$FDA:

PERM	Object-class	{OBJECT_CLASS_ID_OF (VariableCompositeBody)}
------	--------------	---

\$FPDA:

REQ	Object-class	{OBJECT_CLASS_ID_OF (VariableCompositeBody)}
-----	--------------	---

},

REQ	Subordinates	{SUB_ID_OF(SnakingColumns)+, SUB_ID_OF(SynchronizedColumns)+, SUB_ID_OF(BasicFloat)+, SUB_ID_OF(FootnoteArea)+, SUB_ID_OF(CompositeFloat)+, SUB_ID_OF(CompositeFixtureVariable)+, SUB_ID_OF(TableArea)+, SUB_ID_OF(ArrangedContentVariable)+, SUB_ID_OF(SourcedContentVariable)+}, {REQ #constraint-name {"7"}, PERM #external-data {ANY_VALUE}}
PERM	Application-comments	

7.4.3.10 ColumnFixed

:ANY-FRAME-VARIABLE {

GENERIC:

CASE \$DAC OF {

\$PDA-FPDA:

REQ Position

{REQ #fixed-position

{REQ #horizontal-position {ANY_VALUE},

REQ #vertical-position {ANY_VALUE}}},

CASE SUPERIOR ({VariableCompositeBody

| FixedCompositeBody} (Layout-path)) OF {

{'270-degrees': -- body layout A --

REQ Dimensions

{REQ #horizontal-dimension

{REQ #fixed-dimension {ANY_VALUE}

|REQ #maximum-size {'applies'}}},

REQ #vertical-dimension

{REQ #rule-b {ANY_VALUE}

|REQ #maximum-size {'applies'}}},

PERM Layout-path

{'270-degrees'}

{'0-degrees': -- body layout B --

REQ Dimensions

{REQ #horizontal-dimension

{REQ #rule-b {ANY_VALUE}

|REQ #maximum-size {'applies'}}},

REQ #vertical-dimension

{REQ #fixed-dimension {ANY_VALUE}

|REQ #maximum-size {'applies'}}},

REQ Layout-path

{'0-degrees'}

{'180-degrees': -- body layout C --

REQ Dimensions

{REQ #horizontal-dimension

{REQ #maximum-size {'applies'}}},

REQ #vertical-dimension

{REQ #fixed-dimension {ANY_VALUE}

|REQ #maximum-size {'applies'}}},

REQ Layout-path

{'180-degrees'}

} },

REQ Application-comments

{REQ #constraint-name {"8"},

PERM #external-data {ANY_VALUE}}

SPECIFIC:

CASE \$DAC OF {

\$FDA:

PERM Object-class

{OBJECT_CLASS_ID_OF(ColumnFixed)}

\$FPDA:

REQ Object-class

{OBJECT_CLASS_ID_OF(ColumnFixed)}

},

REQ Subordinates

{SUB_ID_OF(SpecificBlock)+},

PERM Imaging-order

{SUB_ID_OF(SpecificBlock)+},

PERM Application-comments

{REQ #constraint-name {"8"},

PERM #external-data {ANY_VALUE}}

SPECIFIC_AND_GENERIC:

PERM Permitted-categories

{ANY_STRING...}

}

4.3.11 ColumnVariable

ANY-FRAME-VARIABLE {

GENERIC:

CASE \$DAC OF {

\$PDA-FPDA:

REQ Position {REQ #variable-position {
 PERM #offset {ANY_VALUE},
 PERM #separation {ANY_VALUE},
 PERM #alignment {ANY_VALUE},
 PERM #fill-order {'normal-order'}}},

CASE SUPERIOR (VariableCompositeBody(Layout-path)) OF {

{'270-degrees'}: -- body layout A --

REQ Dimensions {REQ #horizontal-dimension
 {REQ #fixed-dimension {ANY_VALUE}},
 REQ #vertical-dimension
 {REQ #rule-b {ANY_VALUE}
 {REQ #maximum-size {'applies'}}},
 PERM Layout-path {'270-degrees'}

{'0-degrees'}: -- body layout B --

REQ Dimensions {REQ #horizontal-dimension
 {REQ #rule-b {ANY_VALUE}
 {REQ #maximum-size {'applies'}}},
 REQ #vertical-dimension
 {REQ #fixed-dimension {ANY_VALUE}}},
 REQ Layout-path {'0-degrees'}

{'180-degrees'}: -- body layout C --

REQ Dimensions {REQ #horizontal-dimension
 {REQ #rule-b {ANY_VALUE}
 {REQ #maximum-size {'applies'}}},
 REQ #vertical-dimension
 {REQ #fixed-dimension {ANY_VALUE}}},
 REQ Layout-path {'180-degrees'}

} }.

REQ Application-comments {REQ #constraint-name {"9"},
 PERM #external-data {ANY_VALUE}}

SPECIFIC:

CASE \$DAC OF {

\$FDA:

PERM Object-class {OBJECT_CLASS_ID_OF(ColumnVariable)}

\$FPDA:

REQ Object-class {OBJECT_CLASS_ID_OF(ColumnVariable)}

},

REQ Subordinates {SUB_ID_OF(SpecificBlock)+},

PERM Imaging-order {SUB_ID_OF(SpecificBlock)+},

PERM Application-comments {REQ #constraint-name {"9"},
 PERM #external-data {ANY_VALUE}}

SPECIFIC_AND_GENERIC:

PERM Permitted-categories {ANY_STRING...}

}

7.4.3.12 SnakingColumns

:ANY-FRAME-VARIABLE {

GENERIC:

CASE \$DAC OF {

\$PDA-FPDA:

```

    REQ Generator-for-subordinates    {$SnakingColumnsGFS},
    REQ Position                      {REQ #variable-position {
                                     PERM #offset {ANY_VALUE},
                                     PERM #separation {ANY_VALUE},
                                     PERM #alignment {ANY_VALUE},
                                     PERM #fill-order {'normal-order'}}},
    PERM Balance                      {ANY_VALUE},

```

CASE SUPERIOR (VariableCompositeBody(Layout-path)) OF {

{'270-degrees'}: -- body layout A --

```

    REQ Dimensions                    {REQ #horizontal-dimension
                                     {REQ #fixed-dimension {ANY_VALUE}
                                     |REQ #maximum-size {'applies'}},
                                     REQ #vertical-dimension
                                     {REQ #rule-b {ANY_VALUE}}},
    REQ Layout-path                   {'0-degrees'|'180-degrees'}

```

{'0-degrees'}: -- body layout B --

```

    REQ Dimensions                    {REQ #horizontal-dimension
                                     {REQ #rule-b {ANY_VALUE}},
                                     REQ #vertical-dimension
                                     {REQ #fixed-dimension {ANY_VALUE}
                                     |REQ #maximum-size {'applies'}}},
    PERM Layout-path                  {'90-degrees'|'270-degrees'}

```

{'180-degrees'}: -- body layout C --

```

    REQ Dimensions                    {REQ #horizontal-dimension
                                     {REQ #rule-b {ANY_VALUE}},
                                     REQ #vertical-dimension
                                     {REQ #fixed-dimension {ANY_VALUE}
                                     |REQ #maximum-size {'applies'}}},
    PERM Layout-path                  {'270-degrees'}

```

} },

```

REQ Application-comments              {REQ #constraint-name {"10"},
                                       PERM #external-data {ANY_VALUE}}

```

SPECIFIC:

CASE \$DAC OF {

\$FDA:

```

    PERM Object-class                 {OBJECT_CLASS_ID_OF(Snakingcolumns))

```

\$FPDA:

```

    REQ Object-class                  {OBJECT_CLASS_ID_OF(Snakingcolumns))

```

},

```

REQ Subordinates                      {SUB_ID_OF(ColumnVariable)+,
                                       SUB_ID_OF(CompositeColumnVariable)+,
                                       SUB_ID_OF(ArrangedContentVariable)+,
                                       SUB_ID_OF(SourcedContentVariable)+,
PERM Application-comments              {REQ #constraint-name {"10"},

```


PERM #external-data {ANY_VALUE}}

7.4.3.13 SynchronizedColumns

ANY-FRAME-VARIABLE {

GENERIC:

CASE \$DAC OF {

\$PDA-FPDA:

REQ Generator-for-subordinates {\$SynchronizedColumnsGFS},

REQ Position {REQ #variable-position {
PERM #offset {ANY_VALUE},
PERM #separation {ANY_VALUE},
PERM #alignment {ANY_VALUE},
PERM #fill-order {'normal-order'}}},

CASE SUPERIOR (VariableCompositeBody(Layout-path)) OF {

{'270-degrees'}: -- body layout A --

REQ Dimensions {REQ #horizontal-dimension
{REQ #fixed-dimension {ANY_VALUE}
|REQ #maximum-size {'applies'}},
REQ #vertical-dimension
{REQ #rule-b {ANY_VALUE}}},
PERM Layout-path {'270-degrees'}

{'0-degrees'}: -- body layout B --

REQ Dimensions {REQ #horizontal-dimension
{REQ #rule-b {ANY_VALUE}},
REQ #vertical-dimension
{REQ #fixed-dimension {ANY_VALUE}
|REQ #maximum-size {'applies'}}},
REQ Layout-path {'0-degrees'}

{'180-degrees'}: -- body layout C --

REQ Dimensions {REQ #horizontal-dimension
{REQ #rule-b {ANY_VALUE}},
REQ #vertical-dimension
{REQ #fixed-dimension {ANY_VALUE}
|REQ #maximum-size {'applies'}}},
REQ Layout-path {'180-degrees'}

} },

REQ Application-comments {REQ #constraint-name {"11"},
PERM #external-data {ANY_VALUE}}

SPECIFIC:

CASE \$DAC OF {

\$FDA:

PERM Object-class {OBJECT_CLASS_ID_OF(SynchronizedColumns)}

\$FPDA:

REQ Object-class {OBJECT_CLASS_ID_OF(SynchronizedColumns)}

},

REQ Subordinates {SUB_ID_OF(ColumnFixed)+,
SUB_ID_OF(CompositeColumnFixed)+,
SUB_ID_OF(ArrangedContentFixed)+,

```

        SUB_ID_OF(SourcedContentFixed)+},
    PERM Application-comments {REQ #constraint-name {"11"},
    PERM #external-data {ANY_VALUE}}
}

```

7.4.3.14 BasicFloat

```

:ANY-FRAME-VARIABLE {
GENERIC:

```

```

    CASE $DAC OF {
        $PDA-FPDA:

```

```

            REQ Position {REQ #variable-position {
                PERM #offset {ANY_VALUE},
                PERM #separation {ANY_VALUE},
                PERM #alignment {ANY_VALUE},
                PERM #fill-order {'normal-order'}}},

```

```

        CASE SUPERIOR ({VariableCompositeBody
            | CompositeColumnVariable | CompositeColumnFixed
            | CompositeFixtureVariable | CompositeFixtureFixed} (Layout-path)) OF {

```

```

            {'270-degrees'}: -- body layout A --

```

```

                REQ Dimensions {REQ #horizontal-dimension
                    {REQ #fixed-dimension {ANY_VALUE}
                    |REQ #maximum-size {'applies'}},
                    REQ #vertical-dimension
                    {REQ #rule-b {ANY_VALUE}}},
                PERM Layout-path {'270-degrees'}

```

```

            {'0-degrees'}: -- body layout B --

```

```

                REQ Dimensions {REQ #horizontal-dimension
                    {REQ #rule-b {ANY_VALUE}},
                    REQ #vertical-dimension
                    {REQ #fixed-dimension {ANY_VALUE}
                    |REQ #maximum-size {'applies'}}},
                REQ Layout-path {'0-degrees'}

```

```

            {'180-degrees'}: -- body layout C --

```

```

                REQ Dimensions {REQ #horizontal-dimension
                    {REQ #rule-b {ANY_VALUE}},
                    REQ #vertical-dimension
                    {REQ #fixed-dimension {ANY_VALUE}
                    |REQ #maximum-size {'applies'}}},
                REQ Layout-path {'180-degrees'}

```

```

            } },

```

```

    REQ Application-comments {REQ #constraint-name {"12"},
    PERM #external-data {ANY_VALUE}}

```

```

SPECIFIC:

```

```

    CASE $DAC OF {
        $FDA:

```

```

            PERM Object-class {OBJECT_CLASS_ID_OF(BasicFloat)}

```

```

        $FPDA:

```

```

            REQ Object-class {OBJECT_CLASS_ID_OF(BasicFloat)}

```

```

    },

```

```

REQ Subordinates {SUB_ID_OF(SpecificBlock)+},
PERM Imaging-order {SUB_ID_OF(SpecificBlock)+},
PERM Application-comments {REQ #constraint-name {"12"},
                           PERM #external-data {ANY_VALUE}}
SPECIFIC_AND_GENERIC:
PERM Permitted-categories {ANY_STRING...}

```

4.3.15 CompositeFloat

```

ANY-FRAME-VARIABLE {
GENERIC:
CASE $DAC OF {
  $PDA-FPDA:
    REQ Generator-for-subordinates {$CompositeFloatGFS},
    REQ Position {REQ #variable-position {
                  PERM #offset {ANY_VALUE},
                  PERM #separation {ANY_VALUE},
                  PERM #alignment {ANY_VALUE},
                  PERM #fill-order {'normal-order'}}},

    CASE SUPERIOR (VariableCompositeBody(Layout-path)) OF {

      {'270-degrees': -- body layout A --
        REQ Dimensions {REQ #horizontal-dimension
                        {REQ #fixed-dimension {ANY_VALUE}
                        {REQ #maximum-size {'applies'}},
                        REQ #vertical-dimension
                        {REQ #rule-a {ANY_VALUE}}},
        PERM Layout-path {'0-degrees'|'180-degrees'}

      {'0-degrees': -- body layout B --
        REQ Dimensions {REQ #horizontal-dimension
                        {REQ #rule-a {ANY_VALUE}},
                        REQ #vertical-dimension
                        {REQ #fixed-dimension {ANY_VALUE}
                        {REQ #maximum-size {'applies'}}},
        REQ Layout-path {'90-degrees'|'270-degrees'}

      {'180-degrees': -- body layout C --
        REQ Dimensions {REQ #horizontal-dimension
                        {REQ #rule-a {ANY_VALUE}},
                        REQ #vertical-dimension
                        {REQ #fixed-dimension {ANY_VALUE}
                        {REQ #maximum-size {'applies'}}},
        REQ Layout-path {'90-degrees'|'270-degrees'}

    } },
REQ Application-comments {REQ #constraint-name {"13"},
                           PERM #external-data {ANY_VALUE}}
SPECIFIC:
CASE $DAC OF {
  $FDA:
    PERM Object-class {OBJECT_CLASS_ID_OF(CompositeFloat)}
  $FPDA:

```

```

        REQ Object-class {OBJECT_CLASS_ID_OF(CompositeFloat))
    },
    REQ Subordinates {SUB_ID_OF(CompositeFixtureVariable)+,
        SUB_ID_OF(TableArea)+,
        SUB_ID_OF(BasicColumn)+,
        SUB_ID_OF(ArrangedContentVariable)+,
        SUB_ID_OF(SourcedContentVariable)+,
    PERM Application-comments {REQ #constraint-name {"13"},
        PERM #external-data {ANY_VALUE}}
}

```

7.4.3.16 BasicColumn

:ANY-FRAME-VARIABLE {

GENERIC:

CASE \$DAC OF {

\$PDA-FPDA:

```

    REQ Position {REQ #variable-position {
        PERM #offset {ANY_VALUE},
        PERM #separation {ANY_VALUE},
        PERM #alignment {ANY_VALUE},
        PERM #fill-order {'normal-order'}}},

```

CASE SUPERIOR (VariableCompositeBody(Layout-path)) OF {

{'270-degrees'}: -- body layout A --

```

    REQ Dimensions {REQ #horizontal-dimension
        {REQ #fixed-dimension {ANY_VALUE}
        |REQ #rule-b {ANY_VALUE}},
    REQ #vertical-dimension
        {REQ #fixed-dimension {ANY_VALUE}
        |REQ #rule-b {ANY_VALUE}
        |REQ #maximum-size{'applies'}}},
    PERM Layout-path {'270-degrees'}

```

{'0-degrees'}: -- body layout B --

```

    REQ Dimensions {REQ #horizontal-dimension
        {REQ #fixed-dimension {ANY_VALUE}
        |REQ #rule-b {ANY_VALUE}
        |REQ #maximum-size{'applies'}},
    REQ #vertical-dimension
        {REQ #fixed-dimension {ANY_VALUE}
        |REQ #rule-b {ANY_VALUE}}},
    REQ Layout-path {'0-degrees'}

```

{'180-degrees'}: -- body layout C --

```

    REQ Dimensions {REQ #horizontal-dimension
        {REQ #fixed-dimension {ANY_VALUE}
        |REQ #rule-b {ANY_VALUE}
        |REQ #maximum-size{'applies'}},
    REQ #vertical-dimension
        {REQ #fixed-dimension {ANY_VALUE}
        |REQ #rule-b {ANY_VALUE}}},
    REQ Layout-path {'180-degrees'}

```



```

    } },
    REQ Application-comments      {REQ #constraint-name {"14"},
                                  PERM #external-data {ANY_VALUE}}

SPECIFIC:
CASE $DAC OF {
    $FDA:
        PERM Object-class      {OBJECT_CLASS_ID_OF(BasicColumn)}
    $FPDA:
        REQ Object-class       {OBJECT_CLASS_ID_OF(BasicColumn)}
    },
    REQ Subordinates           {SUB_ID_OF(SpecificBlock)+},
    PERM Application-comments  {REQ #constraint-name {"14"},
                                PERM #external-data {ANY_VALUE}}

SPECIFIC_AND_GENERIC:
    PERM Permitted-categories  {ANY_STRING...}

```

7.4.3.17 FootnoteArea

```

:ANY-FRAME-VARIABLE {
GENERIC:
CASE $DAC OF {
    $PDA-FPDA:
        REQ Position          {REQ #variable-position {
                                PERM #offset {ANY_VALUE},
                                PERM #separation {ANY_VALUE},
                                PERM #alignment {ANY_VALUE},
                                PERM #fill-order {'reverse-order'}}},

        CASE SUPERIOR ({VariableCompositeBody
                        | CompositeColumnVariable | CompositeColumnFixed
                        | CompositeFixtureVariable | CompositeFixtureFixed} (Layout-path)) OF {

            {'270-degrees': -- body layout A --
                REQ Dimensions      {REQ #horizontal-dimension
                                    {REQ #fixed-dimension {ANY_VALUE}
                                    |REQ #maximum-size {'applies'}},
                                    REQ #vertical-dimension
                                    {REQ #rule-b {ANY_VALUE}}},
                PERM Layout-path    {'270-degrees'}

            {'0-degrees': -- body layout B --
                REQ Dimensions      {REQ #horizontal-dimension
                                    {REQ #rule-b {ANY_VALUE}},
                                    REQ #vertical-dimension
                                    {REQ #fixed-dimension {ANY_VALUE}
                                    |REQ #maximum-size {'applies'}}},
                REQ Layout-path     {'0-degrees'}

            {'180-degrees': -- body layout C --
                REQ Dimensions      {REQ #horizontal-dimension
                                    {REQ #rule-b {ANY_VALUE}},
                                    REQ #vertical-dimension
                                    {REQ #fixed-dimension {ANY_VALUE}

```

```

                                |REQ #maximum-size {'applies'}),
REQ Layout-path                {'180-degrees'}
} }.
REQ Permitted-categories        {$FOOTNOTECATEGORY},
    -- For example,
        For CompositeBody        "Footnote-1"
        For SnakingColumns       "Footnote-2"
        For SynchronizedColumns  "Footnote-3", "Footnote-4",
                                "Footnote-5"
        For CompositeFixture     "Footnote-6"
    --
REQ Application-comments        {REQ #constraint-name {"15"},
                                PERM #external-data {ANY_VALUE}}
SPECIFIC:
CASE $DAC OF {
    $FDA:
        PERM Object-class        {OBJECT_CLASS_ID_OF(FootnoteArea)}
    $FPDA:
        REQ Object-class         {OBJECT_CLASS_ID_OF(FootnoteArea)}
},
REQ Subordinates                {SUB_ID_OF(SpecificBlock)+},
PERM Permitted-categories        {$FOOTNOTECATEGORY},
PERM Application-comments        {REQ #constraint-name {"15"},
                                PERM #external-data {ANY_VALUE}}

```

7.4.3.18 ArrangedContentFixed

```

:ANY-FRAME-FIXED {
GENERIC:
CASE $DAC OF {
    $PDA-FPDA:
        REQ Generator-for-subordinates {<construction-expr>::=
                                SEQ(OBJECT_CLASS_ID_OF(GenericBlock)...)},
REQ Application-comments        {REQ #constraint-name {"16"},
                                PERM #external-data {ANY_VALUE}}
SPECIFIC:
CASE $DAC OF {
    $FDA:
        PERM Object-class        {OBJECT_CLASS_ID_OF
                                (ArrangedContentFixed)}
    $FPDA:
        REQ Object-class         {OBJECT_CLASS_ID_OF
                                (ArrangedContentFixed)}
},
REQ Subordinates                {SUB_ID_OF(GenericBlock)+},
PERM Imaging-order              {SUB_ID_OF(GenericBlock)+},
PERM Application-comments        {REQ #constraint-name {"16"},
                                PERM #external-data {ANY_VALUE}}
}

```

7.4.3.19 ArrangedContentVariable

```

:ANY-FRAME-VARIABLE {
GENERIC:
  CASE $DAC OF {
    $PDA-FPDA:
      REQ  Generator-for-subordinates {<construction-expr>::=
                                         SEQ(OBJECT_CLASS_ID_OF(GenericBlock)...)},
      REQ  Position                    {REQ #variable-position {
                                         PERM #offset {ANY_VALUE},
                                         PERM #separation {ANY_VALUE},
                                         PERM #alignment {ANY_VALUE},
                                         PERM #fill-order {'normal-order'}}},
      REQ  Dimensions                  {REQ #horizontal-dimension
                                         {REQ #fixed-dimension {ANY_VALUE}},
                                         REQ #vertical-dimension
                                         {REQ #fixed-dimension {ANY_VALUE}}}
    },
    REQ  Application-comments          {REQ #constraint-name {"17"},
                                         PERM #external-data {ANY_VALUE}}
SPECIFIC:
  CASE $DAC OF {
    $FDA:
      PERM Object-class                {OBJECT_CLASS_ID_OF
                                         (ArrangedContentVariable)}
    $FPDA:
      REQ  Object-class                {OBJECT_CLASS_ID_OF
                                         (ArrangedContentVariable)}
    },
    REQ  Subordinates                  {SUB_ID_OF(GenericBlock)+},
    PERM  Imaging-order                 {SUB_ID_OF(GenericBlock)+},
    PERM  Application-comments          {REQ #constraint-name {"17"},
                                         PERM #external-data {ANY_VALUE}}
  }

```

7.4.3.20 SourcedContentFixed

```

:ANY-FRAME-VARIABLE {
GENERIC:
  CASE $DAC OF {
    $PDA-FPDA:
      REQ  Logical-source               {OBJECT_CLASS_ID_OF(CommonContent)},
      REQ  Position                     {REQ #fixed-position
                                         {REQ #horizontal-position {ANY_VALUE},
                                         REQ #vertical-position {ANY_VALUE}}},
      REQ  Dimensions                   {REQ #horizontal-dimension
                                         {REQ #fixed-dimension {ANY_VALUE}},
                                         REQ #vertical-dimension
                                         {REQ #fixed-dimension {ANY_VALUE}}},
    CASE SUPERIOR ((CompositeHeader | CompositeFooter | TableHeader
                    | FixedCompositeBody | CompositeCommon
                    | SynchronizedColumns) (Layout-path)) OF {

```

{'270-degrees'}: -- H/F layout A1 or B2 when the immediate superior is CompositeHeader, CompositeFooter or TableHeader, or --
 -- body layout A when the immediate superior is FixedCompositeBody, CompositeCommon or SynchronizedColumns --
 PERM Layout-path {'270-degrees' -- H/F layout A1 or body layout A --
 {'180-degrees'} -- H/F layout B2 --

{'180-degrees'}: -- H/F layout B1 when the immediate superior is CompositeHeader or CompositeFooter, or --
 -- body layout C when the immediate superior is FixedCompositeBody, CompositeCommon or SynchronizedColumns --
 REQ Layout-path {'180-degrees'} -- H/F layout B1 or body layout C --

{' 0-degrees'}: -- H/F layout A2 when the immediate superior is CompositeHeader or CompositeFooter, or --
 -- body layout B when the immediate superior is FixedCompositeBody, CompositeCommon or SynchronizedColumns --
 PERM Layout-path {'270-degrees' -- H/F layout A2 --
 {'0-degrees'} -- body layout B --

} },
 REQ Application-comments {REQ #constraint-name {"18"},
 PERM #external-data {ANY_VALUE}}

SPECIFIC:

CASE \$DAC OF {
 \$FDA: PERM Object-class {OBJECT_CLASS_ID_OF
 (SourcedContentFixed))

\$FPDA: REQ Object-class {OBJECT_CLASS_ID_OF
 (SourcedContentFixed))

},
 REQ Subordinates {SUB_ID_OF(SpecificBlock)+},
 PERM Application-comments {REQ #constraint-name {"18"},
 PERM #external-data {ANY_VALUE}}

}

7.4.3.21 SourcedContentVariable

:ANY-FRAME-VARIABLE {

GENERIC:

CASE \$DAC OF {
 \$PDA-FPDA:
 REQ Logical-source {OBJECT_CLASS_ID_OF(CommonContent)),
 REQ Position {REQ #variable-position {
 PERM #offset {ANY_VALUE},
 PERM #separation {ANY_VALUE},
 PERM #alignment {ANY_VALUE},
 PERM #fill-order {'normal-order'}}),

CASE SUPERIOR ({CompositeHeader | CompositeFooter | VariableCompositeBody
 | CompositeColumnVariable | CompositeColumnFixed | CompositeCommon


```

| SnakingColumns | CompositeFloat} (Layout-path)) OF {

{'270-degrees'}:
    -- H/F layout A1 or B2 when the immediate superior is
    CompositeHeader or CompositeFooter, or --
    -- body layout A when the immediate superior is
    VariableCompositeBody, CompositeColumnVariable,
    CompositeColumnFixed or CompositeCommon, or --
    -- body layout B or C when the immediate superior is
    SnakingColumns or CompositeFloat --
REQ Dimensions      {REQ #horizontal-dimension
                    {REQ #fixed-dimension {ANY_VALUE}
                    |REQ #rule-b {ANY_VALUE}
                    |REQ #maximum-size {'applies'}},
                    REQ #vertical-dimension
                    {REQ #fixed-dimension {ANY_VALUE}
                    |REQ #rule-b {ANY_VALUE}
                    |REQ #maximum-size {'applies'}}},
PERM Layout-path    {'270-degrees' -- H/F layout A1 or body layout A --
                    |'180-degrees' -- H/F layout B2 or body layout C --
                    |'0-degrees'  -- body layout B --

{'180-degrees'}:
    -- H/F layout B1 when the immediate superior is
    CompositeHeader or CompositeFooter, or --
    -- body layout C when the immediate superior is
    VariableCompositeBody, CompositeColumnVariable,
    CompositeColumnFixed or CompositeCommon, or --
    -- body layout A when the immediate superior is
    SnakingColumns or CompositeFloat --
REQ Dimensions      {REQ #horizontal-dimension
                    {REQ #fixed-dimension {ANY_VALUE}
                    |REQ #rule-b {ANY_VALUE}},
                    REQ #vertical-dimension
                    {REQ #fixed-dimension {ANY_VALUE}
                    |REQ #rule-b {ANY_VALUE}
                    |REQ #maximum-size {'applies'}}},
REQ Layout-path     {'180-degrees' -- H/F layout B1 or body layout C --
                    |'270-degrees' -- body layout A --

{' 0-degrees'}:
    -- H/F layout A2 when the immediate superior is
    CompositeHeader or CompositeFooter, or --
    -- body layout B when the immediate superior is
    VariableCompositeBody, CompositeColumnVariable,
    CompositeColumnFixed or CompositeCommon, or --
    -- body layout A when the immediate superior is
    SnakingColumns or CompositeFloat --
REQ Dimensions      {REQ #horizontal-dimension
                    {REQ #fixed-dimension {ANY_VALUE}
                    |REQ #rule-b {ANY_VALUE}
                    |REQ #maximum-size {'applies'}},
                    REQ #vertical-dimension
                    {REQ #fixed-dimension {ANY_VALUE}
                    |REQ #rule-b {ANY_VALUE}
                    |REQ #maximum-size {'applies'}}},
PERM Layout-path    {'270-degrees' -- H/F layout A2 or body layout A --
                    |'0-degrees'  -- body layout B --

```

```

    {'90-degrees'}:      -- body layout B when the immediate superior is
                          SnakingColumns, or --
                          -- body layout B or C when the immediate superior is
                          CompositeFloat --
    REQ Dimensions      {REQ #horizontal-dimension
                          {REQ #fixed-dimension {ANY_VALUE}
                          |REQ #rule-b {ANY_VALUE}
                          |REQ #maximum-size {'applies'}},
                          REQ #vertical-dimension
                          {REQ #fixed-dimension {ANY_VALUE}
                          |REQ #rule-b {ANY_VALUE}}},
    PERM Layout-path    {'0-degrees' -- body layout B --
                          {'180-degrees' -- body layout C --

} }.

REQ Application-comments {REQ #constraint-name {"19"},
                          PERM #external-data {ANY_VALUE}}
SPECIFIC:
CASE $DAC OF {
    $FDA:
        PERM Object-class {OBJECT_CLASS_ID_OF
                            (SourcedContentVariable))
    $FPDA:
        REQ Object-class {OBJECT_CLASS_ID_OF
                            (SourcedContentVariable))
},
REQ Subordinates        {SUB_ID_OF(SpecificBlock)+},
PERM Application-comments {REQ #constraint-name {"19"},
                          PERM #external-data {ANY_VALUE}}
}

```

7.4.3.22 CompositeFixtureVariable

```

:ANY-FRAME-VARIABLE {
GENERIC:
CASE $DAC OF {
    $PDA-FPDA:
        REQ Generator-for-subordinates {$CompositeFixtureVariableGFS},
        REQ Position {REQ #variable-position {
                        PERM #offset {ANY_VALUE},
                        PERM #separation {ANY_VALUE},
                        PERM #alignment {ANY_VALUE},
                        PERM #fill-order {'normal-order'}}},

CASE SUPERIOR (VariableCompositeBody(Layout-path)) OF {
    {'270-degrees'}: -- body layout A --
        REQ Dimensions {REQ #horizontal-dimension
                        {REQ #fixed-dimension {ANY_VALUE}
                        |REQ #maximum-size {'applies'}
                        |REQ #rule-b {ANY_VALUE}},
                        REQ #vertical-dimension
                        {REQ #fixed-dimension {ANY_VALUE}
                        |REQ #rule-b {ANY_VALUE}}},

```

```

        PERM Layout-path      {'270-degrees'|'180-degrees'
                               '|0-degrees'}

{'0-degrees': -- body layout B --
    REQ Dimensions           {REQ #horizontal-dimension
                              {REQ #fixed-dimension {ANY_VALUE}
                              |REQ #rule-b {ANY_VALUE}}},
                              REQ #vertical-dimension
                              {REQ #fixed-dimension {ANY_VALUE}
                              |REQ #maximum-size {'applies'}
                              |REQ #rule-b {ANY_VALUE}}},
    REQ Layout-path          {'0-degrees'|'90-degrees'
                              '|270-degrees'}

{'180-degrees': -- body layout C --
    REQ Dimensions           {REQ #horizontal-dimension
                              {REQ #fixed-dimension {ANY_VALUE}
                              |REQ #rule-b {ANY_VALUE}}},
                              REQ #vertical-dimension
                              {REQ #fixed-dimension {ANY_VALUE}
                              |REQ #maximum-size {'applies'}
                              |REQ #rule-b {ANY_VALUE}}},
    REQ Layout-path          {'180-degrees'|'270-degrees'}
} ).
REQ Application-comments     {REQ #constraint-name {"20"},
                              PERM #external-data {ANY_VALUE}}

SPECIFIC:
CASE $DAC OF {
    $FDA:
        PERM Object-class    {OBJECT_CLASS_ID_OF
                              (CompositeFixtureVariable)}
    $FPDA:
        REQ Object-class     {OBJECT_CLASS_ID_OF
                              (CompositeFixtureVariable)}
    },
REQ Subordinates             {SUB_ID_OF(BasicFloat)+,
                              SUB_ID_OF(FootnoteArea)+,
                              SUB_ID_OF(CompositeArtwork)+,
                              SUB_ID_OF(FormArea)+,
    PERM Application-comments {REQ #constraint-name {"20"},
                              PERM #external-data {ANY_VALUE}}
}

```

7.4.3.23 CompositeFixtureFixed

```

:ANY-FRAME-VARIABLE {
GENERIC:
CASE $DAC OF {
    $PDA-FPDA:
        REQ Generator-for-subordinates {$CompositeFixtureFixedGFS},
        REQ Position                   {REQ #fixed-position
                                        {REQ #horizontal-position {ANY_VALUE},
                                        REQ #vertical-position {ANY_VALUE}}},

```

```

CASE SUPERIOR (FixedCompositeBody(Layout-path)) OF {
  {'270-degrees': -- body layout A --
    REQ Dimensions {REQ #horizontal-dimension
      {REQ #fixed-dimension {ANY_VALUE}
      |REQ #maximum-size {'applies'}},
      REQ #vertical-dimension
      {REQ #fixed-dimension {ANY_VALUE}
      |REQ #rule-b {ANY_VALUE}}},
    PERM Layout-path {'270-degrees'|'180-degrees'
      |'0-degrees'}

  {'0-degrees': -- body layout B --
    REQ Dimensions {REQ #horizontal-dimension
      {REQ #fixed-dimension {ANY_VALUE}
      |REQ #rule-b {ANY_VALUE}},
      REQ #vertical-dimension
      {REQ #fixed-dimension {ANY_VALUE}
      |REQ #maximum-size {'applies'}}},
    REQ Layout-path {'0-degrees'|'90-degrees'
      |'270-degrees'}

  {'180-degrees': -- body layout C --
    REQ Dimensions {REQ #horizontal-dimension
      {REQ #fixed-dimension {ANY_VALUE}
      |REQ #rule-b {ANY_VALUE}},
      REQ #vertical-dimension
      {REQ #fixed-dimension {ANY_VALUE}
      |REQ #maximum-size {'applies'}}},
    REQ Layout-path {'180-degrees'|'270-degrees'}
  } },
  REQ Application-comments {REQ #constraint-name {"21"},
    PERM #external-data {ANY_VALUE}}
SPECIFIC:
  CASE $DAC OF {
    $FDA:
      PERM Object-class {OBJECT_CLASS_ID_OF
        (CompositeFixtureFixed))
    $FPDA:
      REQ Object-class {OBJECT_CLASS_ID_OF
        (CompositeFixtureFixed))
  },
  REQ Subordinates {SUB_ID_OF(BasicFloat)+,
    SUB_ID_OF(FootnoteArea)+,
    SUB_ID_OF(CompositeArtwork)+,
    SUB_ID_OF(FormArea)+,
  PERM Application-comments {REQ #constraint-name {"21"},
    PERM #external-data {ANY_VALUE}}
}

```

7.4.3.24 BasicFixture

```

:ANY-FRAME-VARIABLE {
GENERIC:

```



```

CASE $DAC OF {
  $PDA-FPDA:
    REQ Position {REQ #fixed-position
                  {REQ #horizontal-position {ANY_VALUE},
                  REQ #vertical-position {ANY_VALUE}}},

```

-- Note that values of position may usually be "0" for overlapping figure. --

```

    REQ Dimensions {REQ #horizontal-dimension
                    {REQ #fixed-dimension {ANY_VALUE}
                    {REQ #rule-b {ANY_VALUE}}},
                    REQ #vertical-dimension
                    {REQ #fixed-dimension {ANY_VALUE}
                    {REQ #rule-b {ANY_VALUE}}},

```

```

CASE SUPERIOR ({FixedCompositeBody | CompositeArtwork} (Layout-path)) OF {

```

```

  {'270-degrees'}: -- body layout A --
    PERM Layout-path {'270-degrees'|'180-degrees'}

```

```

  {'0-degrees'}: -- body layout B --
    REQ Layout-path {'0-degrees'|'270-degrees'}

```

```

  {'180-degrees'}: -- body layout C --
    PERM Layout-path {'180-degrees'|'270-degrees'}

```

```

} },
REQ Application-comments {REQ #constraint-name {"22"},
                          PERM #external-data {ANY_VALUE}}

```

SPECIFIC:

```

CASE $DAC OF {
  $FDA:
    PERM Object-class {OBJECT_CLASS_ID_OF(BasicFixture))
  $FPDA:
    REQ Object-class {OBJECT_CLASS_ID_OF(BasicFixture))
  },
  REQ Subordinates {SUB_ID_OF(SpecificBlock)+},
  PERM Application-comments {REQ #constraint-name {"22"},
                             PERM #external-data {ANY_VALUE}}

```

SPECIFIC_AND_GENERIC:

```

  PERM Permitted-categories {ANY_STRING...}
}

```

7.4.3.25 CompositeColumnFixed

:ANY-FRAME-VARIABLE {

GENERIC:

```

CASE $DAC OF {
  $PDA-FPDA:
    REQ Generator-for-subordinates {$CompositeColumnFixedGFS},
    REQ Position {REQ #fixed-position
                  {REQ #horizontal-position {ANY_VALUE},
                  REQ #vertical-position {ANY_VALUE}}},

```

```

CASE SUPERIOR (VariableCompositeBody(Layout-path)) OF {

```

```

    {'270-degrees': -- body layout A --
        REQ Dimensions      {REQ #horizontal-dimension
                             {REQ #fixed-dimension {ANY_VALUE}
                             |REQ #maximum-size {'applies'}},
                             REQ #vertical-dimension
                             {REQ #rule-b {ANY_VALUE}
                             |REQ #maximum-size {'applies'}}},
        PERM Layout-path    {'270-degrees'}

    {'0-degrees': -- body layout B --
        REQ Dimensions      {REQ #horizontal-dimension
                             {REQ #rule-b {ANY_VALUE}
                             |REQ #maximum-size {'applies'}},
                             REQ #vertical-dimension
                             {REQ #fixed-dimension {ANY_VALUE}
                             |REQ #maximum-size {'applies'}}},
        REQ Layout-path     {'0-degrees'}

    {'180-degrees': -- body layout C --
        REQ Dimensions      {REQ #horizontal-dimension
                             {REQ #maximum-size {'applies'}},
                             REQ #vertical-dimension
                             {REQ #fixed-dimension {ANY_VALUE}
                             |REQ #maximum-size {'applies'}}},
        REQ Layout-path     {'180-degrees'}
    } },
    REQ Application-comments {REQ #constraint-name {"23"},
                              PERM #external-data {ANY_VALUE}}
SPECIFIC:
    CASE $DAC OF {
        $FDA:
            PERM Object-class {OBJECT_CLASS_ID_OF
                              (CompositeColumnFixed))
        $FPDA:
            REQ Object-class  {OBJECT_CLASS_ID_OF
                              (CompositeColumnFixed))
    },
    REQ Subordinates         {SUB_ID_OF(BasicFloat)+,
                              SUB_ID_OF(CompositeFloat)+,
                              SUB_ID_OF(FootnoteArea)+,
                              SUB_ID_OF(CompositeFixtureVariable)+,
                              SUB_ID_OF(TableArea)+,
                              SUB_ID_OF(ArrangedContentVariable)+,
                              SUB_ID_OF(SourcedContentVariable)+,
    PERM Application-comments {REQ #constraint-name {"23"},
                              PERM #external-data {ANY_VALUE}}
}

```

7.4.3.26 CompositeColumnVariable

```

:ANY-FRAME-VARIABLE {
GENERIC:
    CASE $DAC OF {
        $PDA-FPDA:

```

```

REQ Generator-for-subordinates    {$CompositeColumnVariableGFS},
REQ Position                      {REQ #variable-position {
                                PERM #offset    {ANY_VALUE},
                                PERM #separation {ANY_VALUE},
                                PERM #alignment  {ANY_VALUE},
                                PERM #fill-order {'normal-order'}}},

```

```

CASE SUPERIOR (VariableCompositeBody(Layout-path)) OF {

```

```

{'270-degrees'}: -- body layout A --

```

```

    REQ Dimensions    {REQ #horizontal-dimension
                        {REQ #fixed-dimension {ANY_VALUE}},
                        REQ #vertical-dimension
                        {REQ #rule-b {ANY_VALUE}
                        |REQ #maximum-size {'applies'}}},
    PERM Layout-path   {'270-degrees'}

```

```

{'0-degrees'}: -- body layout B --

```

```

    REQ Dimensions    {REQ #horizontal-dimension
                        {REQ #rule-b {ANY_VALUE}
                        |REQ #maximum-size {'applies'}},
                        REQ #vertical-dimension
                        {REQ #fixed-dimension {ANY_VALUE}}},
    REQ Layout-path    {'0-degrees'}

```

```

{'180-degrees'}: -- body layout C --

```

```

    REQ Dimensions    {REQ #horizontal-dimension
                        {REQ #rule-b {ANY_VALUE}
                        |REQ #maximum-size {'applies'}},
                        REQ #vertical-dimension
                        {REQ #fixed-dimension {ANY_VALUE}}},
    REQ Layout-path    {'180-degrees'}

```

```

} }.

```

```

REQ Application-comments    {REQ #constraint-name {"24"},
                              PERM #external-data {ANY_VALUE}}

```

SPECIFIC:

```

CASE $DAC OF {

```

```

    $FDA:

```

```

        PERM Object-class    {OBJECT_CLASS_ID_OF
                                (CompositeColumnVariable))

```

```

    $FPDA:

```

```

        REQ Object-class     {OBJECT_CLASS_ID_OF
                                (CompositeColumnVariable))

```

```

    }.

```

```

REQ Subordinates            {SUB_ID_OF(BasicFloat)+,
                              SUB_ID_OF(CompositeFloat)+,
                              SUB_ID_OF(FootnoteArea)+,
                              SUB_ID_OF(CompositeFixtureVariable)+,
                              SUB_ID_OF(TableArea)+,
                              SUB_ID_OF(ArrangedContentVariable)+,
                              SUB_ID_OF(SourcedContentVariable)+},

```

```

PERM Application-comments    {REQ #constraint-name {"24"},
                              PERM #external-data {ANY_VALUE}}

```

```

}

```

196


```

        REQ #vertical-dimension
            {REQ #fixed-dimension {ANY_VALUE}
            |REQ #rule-b {ANY_VALUE}}},

CASE SUPERIOR ((CompositeFixtureVariable | CompositeFixtureFixed)
                (Layout-path)) OF {

    {'270-degrees':
        PERM Layout-path    {'270-degrees'}
    {'0-degrees':
        REQ Layout-path     {'0-degrees'}

    {'180-degrees':
        REQ Layout-path     {'180-degrees'}

    }},
REQ Application-comments    {REQ #constraint-name {"26"},
                             PERM #external-data {ANY_VALUE}}

SPECIFIC:
CASE $DAC OF {
    $FDA:
        PERM Object-class {OBJECT_CLASS_ID_OF(CompositeArtwork)}
    $FPDA:
        REQ Object-class {OBJECT_CLASS_ID_OF(CompositeArtwork)}
    },
REQ Subordinates            {SUB_ID_OF(BasicFixture)+},
PERM Imaging-order          {SUB_ID_OF(BasicFixture)+},
PERM Application-comments   {REQ #constraint-name {"26"},
                             PERM #external-data {ANY_VALUE}}
}

```

7.4.3.29 BasicHeader

```

:ANY-FRAME-FIXED {
GENERIC:
CASE $DAC OF {
    $PDA-FPDA:
        REQ Logical-source {OBJECT_CLASS_ID_OF(CommonContent)}
    },
PERM Layout-path           {'270-degrees' -- H/F layout A1 --
                             {'180-degrees' -- H/F layout B1 --},
REQ Application-comments   {REQ #constraint-name {"27"},
                             PERM #external-data {ANY_VALUE}}

SPECIFIC:
CASE $DAC OF {
    $FDA:
        PERM Object-class {OBJECT_CLASS_ID_OF(BasicHeader)}
    $FPDA:
        REQ Object-class {OBJECT_CLASS_ID_OF(BasicHeader)}
    },
REQ Subordinates           {SUB_ID_OF(SpecificBlock)+},
PERM Application-comments  {REQ #constraint-name {"27"},
                             PERM #external-data {ANY_VALUE}}
}

```

7.4.3.30 BasicFooter

```
:ANY-FRAME-FIXED {
GENERIC:
    CASE $DAC OF {
        $PDA-FPDA:
            REQ Logical-source {OBJECT_CLASS_ID_OF(CommonContent)}
        },
    PERM Layout-path {'270-degrees' -- H/F layout A1 --
                      |'180-degrees' -- H/F layout B1 --},
    REQ Application-comments {REQ #constraint-name {"33"},
                              PERM #external-data {ANY_VALUE}}
SPECIFIC:
    CASE $DAC OF {
        $FDA:
            PERM Object-class {OBJECT_CLASS_ID_OF(BasicFooter)}
        $FPDA:
            REQ Object-class {OBJECT_CLASS_ID_OF(BasicFooter)}
        },
    REQ Subordinates {SUB_ID_OF(SpecificBlock)+},
    PERM Application-comments {REQ #constraint-name {"33"},
                              PERM #external-data {ANY_VALUE}}
}
```

7.4.3.31 BasicBody

```
:ANY-FRAME-FIXED {
GENERIC:
    PERM Layout-path {'270-degrees' -- body layout A --
                      |'0-degrees' -- body layout B --
                      |'180-degrees' -- body layout C --},
    REQ Application-comments {REQ #constraint-name {"28"},
                              PERM #external-data {ANY_VALUE}}
SPECIFIC:
    CASE $DAC OF {
        $FDA:
            PERM Object-class {OBJECT_CLASS_ID_OF(BasicBody)}
        $FPDA:
            REQ Object-class {OBJECT_CLASS_ID_OF(BasicBody)}
        },
    REQ Subordinates {SUB_ID_OF(SpecificBlock)+},
    PERM Application-comments {REQ #constraint-name {"28"},
                              PERM #external-data {ANY_VALUE}}
}
```

7.4.3.32 GenericBlock

```
{
GENERIC:
    REQ Object-type {'block'},
    REQ Object-class-identifier {ANY_VALUE},
    REQ Content-architecture-class {$FC | $FPC | $FPR | $FPG },
    PERM Content-generator {$GENERICBLOCKREF},
}
```

```

PERM Content-portions {CONTENT_ID_OF(Character-content-portion)+
| CONTENT_ID_OF(Raster-graphics-content-portion)
| CONTENT_ID_OF(Geometric-graphics-content-portion) },

PERM Presentation-style {STYLE_ID_OF(PStyle1)
| STYLE_ID_OF(PStyle2)
| STYLE_ID_OF(PStyle3)},

PERM Resource {ANY_VALUE},
REQ Application-comments {REQ #constraint-name {"29"},
PERM #external-data {ANY_VALUE}}

SPECIFIC:
REQ Object-type {'block'},
REQ Object-identifier {ANY_VALUE},
CASE $DAC OF {
    $FDA:
        PERM Object-class {OBJECT_CLASS_ID_OF(GenericBlock)}
    $FPDA:
        REQ Object-class {OBJECT_CLASS_ID_OF(GenericBlock)}
},
PERM Presentation-style {STYLE_ID_OF(PStyle1)
| STYLE_ID_OF(PStyle2)
| STYLE_ID_OF(PStyle3)},
PERM Content-architecture-class {$FC | $FPC | $FPR | $FPG },
CASE GenericBlock (Object-class) OF {
    VOID:
        REQ Content-portions {CONTENT_ID_OF(Character-content-portion)+
|CONTENT_ID_OF(Raster-graphics-content-portion)
|CONTENT_ID_OF(Geometric-graphics-content-portion)}
},
PERM Presentation-attributes {
    PERM #character-attributes {
        PERM #alignment {ANY_VALUE},
        PERM #character-fonts {ANY_VALUE},
        PERM #character-path {ANY_VALUE},
        PERM #character-spacing {ANY_VALUE},
        PERM #character-orientation {ANY_VALUE},
        PERM #code-extension-announcers {$CDEXTEN},
        PERM #first-line-offset {ANY_VALUE},
        PERM #graphic-character-sets {$PERMIT-GRCHAR},
        PERM #graphic-character-subrepertoire {ANY_VALUE},
        PERM #graphic-rendition {$GRAPHICRENDITIONS},
        PERM #itemisation {ANY_VALUE},
        PERM #kerning-offset {ANY_VALUE},
        PERM #line-layout-table {ANY_VALUE},
        PERM #line-progression {ANY_VALUE},
        PERM #line-spacing {ANY_VALUE},
        PERM #pairwise-kerning {ANY_VALUE},
        PERM #formatting-indicator {ANY_VALUE},
        PERM #initial-offset {ANY_VALUE}
    }
},
PERM Application-comments {REQ #constraint-name {"29"},
PERM #external-data{ANY_VALUE}}

SPECIFIC_AND_GENERIC:
PERM Position {REQ #fixed-position
{REQ #horizontal-position {ANY_VALUE},

```

```

        REQ #vertical-position {ANY_VALUE}}),
    PERM Dimensions {REQ #horizontal-dimension
        {REQ #fixed-dimension {ANY_VALUE}},
        REQ #vertical-dimension
        {REQ #fixed-dimension {ANY_VALUE}}},
    PERM Transparency {ANY_VALUE},
    PERM Colour {ANY_VALUE},
    PERM Border {ANY_VALUE},
    PERM User-readable-comments {ANY_STRING},
    PERM User-visible-name {ANY_STRING}
}

```

7.4.3.33 SpecificBlock

```

{
    SPECIFIC:
    REQ Object-type {'block'},
    REQ Object-identifier {ANY_VALUE},
    REQ Content-portions {CONTENT_ID_OF(Character-content-portion)+
        |CONTENT_ID_OF(Raster-graphics-content-portion)
        |CONTENT_ID_OF(Geometric-graphics-content-portion)},
    PERM Position {REQ #fixed-position
        {REQ #horizontal-position {ANY_VALUE},
        REQ #vertical-position {ANY_VALUE}}},
    PERM Dimensions {REQ #horizontal-dimension
        {REQ #fixed-dimension {ANY_VALUE}},
        REQ #vertical-dimension
        {REQ #fixed-dimension {ANY_VALUE}}},
    PERM Presentation-style {STYLE_ID_OF(PStyle1)
        |STYLE_ID_OF(PStyle2)
        |STYLE_ID_OF(PStyle3)
        |STYLE_ID_OF(PStyle4)},
    PERM Content-architecture-class {$FC | $FPC | $FPR | $FPG},
    PERM Presentation-attributes {
        PERM #character-attributes {
            PERM #alignment {ANY_VALUE},
            PERM #character-fonts {ANY_VALUE},
            PERM #character-path {ANY_VALUE},
            PERM #character-spacing {ANY_VALUE},
            PERM #character-orientation {ANY_VALUE},
            PERM #code-extension-announcers {$CDEXTEN},
            PERM #first-line-offset {ANY_VALUE},
            PERM #graphic-character-sets {$PERMIT-GRCHAR },
            PERM #graphic-character-subrepertoire {ANY_VALUE},
            PERM #graphic-rendition {$GRAPHICRENDITIONS},
            PERM #itemisation {ANY_VALUE},
            PERM #kerning-offset {ANY_VALUE},
            PERM #line-layout-table {ANY_VALUE},
            PERM #line-progression {ANY_VALUE},
            PERM #line-spacing {ANY_VALUE},
            PERM #pairwise-kerning {ANY_VALUE},
            PERM #formatting-indicator {ANY_VALUE},
            PERM #initial-offset {ANY_VALUE}
        }
    }
}

```



```

},
PERM Transparency {ANY_VALUE},
PERM Colour {ANY_VALUE},
PERM Border {ANY_VALUE },
PERM User-readable-comments {ANY_STRING},
PERM User-visible-name {ANY_STRING},
PERM Application-comments {REQ #constraint-name {"30"},
PERM #external-data {ANY_VALUE}}
}

```

7.4.3.34 FormArea

ANY-FRAME-VARIABLE {

GENERIC:

CASE \$DAC OF {
\$PDA-FPDA:

```

    REQ Generator-for-subordinates {$FormAreaGFS}},
REQ Position {REQ #variable-position {
    PERM #offset {ANY_VALUE},
    PERM #separation {ANY_VALUE},
    PERM #alignment {ANY_VALUE},
    PERM #fill-order {'normal-order'}}},
REQ Dimensions {REQ #horizontal-dimension
    {REQ #fixed-dimension {ANY_VALUE}},
    REQ #vertical-dimension
    {REQ #fixed-dimension {ANY_VALUE}}},
PERM Layout-path {'270-degrees'},
REQ Application-comments {REQ #constraint-name {"31"},
    PERM #external-data {ANY_VALUE}}

```

SPECIFIC:

CASE \$DAC OF {
\$FDA:

```

    PERM Object-class {OBJECT_CLASS_ID_OF(FormArea)}
    $FPDA:
    REQ Object-class {OBJECT_CLASS_ID_OF(FormArea)}
},
REQ Subordinates {SUB_ID_OF(ArrangedContentFixed)+,
    SUB_ID_OF(FormEntryArea)+,
    SUB_ID_OF(EntryGroupArea)+},
PERM Position {ANY_VALUE},
PERM Dimensions {REQ #horizontal-dimension
    {REQ #fixed-dimension {ANY_VALUE}},
    REQ #vertical-dimension
    {REQ #fixed-dimension {ANY_VALUE}}},
PERM Application-comments {REQ #constraint-name {"31"},
    PERM #external-data {ANY_VALUE}}

```

SPECIFIC_AND_GENERIC:

```

PERM Transparency {ANY_VALUE},
PERM Colour {ANY_VALUE},
PERM Border {ANY_VALUE}
}

```

7.4.3.35 EntryGroupArea

:ANY-FRAME-FIXED {

GENERIC:

```
CASE $DAC OF {
  $PDA-FPDA:
    REQ Generator-for-subordinates    {$EntryGroupAreaGFS}
  },
  REQ Position                        {REQ #fixed-position
                                     {REQ #horizontal-position {ANY_VALUE},
                                      REQ #vertical-position {ANY_VALUE}}},
  REQ Dimensions                      {REQ #horizontal-dimension
                                     {REQ #fixed-dimension {ANY_VALUE}},
                                      REQ #vertical-dimension
                                      {REQ #fixed-dimension {ANY_VALUE}}},
  PERM Layout-path                   {'270-degrees'},
  REQ Application-comments            {REQ #constraint-name {"35"},
                                     PERM #external-data {ANY_VALUE}}
```

SPECIFIC:

```
CASE $DAC OF {
  $FDA:
    PERM Object-class                {OBJECT_CLASS_ID_OF(EntryGroupArea)}
  $FPDA:
    REQ Object-class                 {OBJECT_CLASS_ID_OF(EntryGroupArea)}
  },
  REQ Subordinates                   {SUB_ID_OF(ArrangedContentFixed)+,
                                     SUB_ID_OF(FormEntryArea)+,
                                     SUB_ID_OF(EntryGroupArea)+},
  PERM Position                      {ANY_VALUE},
  PERM Dimensions                    {REQ #horizontal-dimension
                                     {REQ #fixed-dimension {ANY_VALUE}},
                                      REQ #vertical-dimension
                                      {REQ #fixed-dimension {ANY_VALUE}}},
  PERM Imaging-order                 {ANY_VALUE},
  PERM Application-comments          {REQ #constraint-name {"35"},
                                     PERM #external-data {ANY_VALUE}}
```

SPECIFIC_AND_GENERIC:

```
  PERM Transparency                 {ANY_VALUE},
  PERM Colour                       {ANY_VALUE},
  PERM Border                       {ANY_VALUE}
```

}

7.4.3.36 TableArea

:ANY-FRAME-VARIABLE {

GENERIC:

```
CASE $DAC OF {
  $PDA-FPDA:
    REQ Generator-for-subordinates    {$TableAreaGFS}
  },
  REQ Position                       {REQ #variable-position {
                                     PERM #offset    {ANY_VALUE},
                                     PERM #separation {ANY_VALUE},
```

```

        PERM #alignment {ANY_VALUE},
        PERM #fill-order {'normal-order'}}},
REQ  Dimensions
        {REQ #horizontal-dimension
        {REQ #fixed-dimension {ANY_VALUE}},
        REQ #vertical-dimension
        {REQ #rule-b {ANY_VALUE}
        {REQ #fixed-dimension {ANY_VALUE}}},
        {'270-degrees'},
PERM Layout-path
REQ  Application-comments
        {REQ #constraint-name {"36"},
        PERM #external-data {ANY_VALUE}}

SPECIFIC:
CASE $DAC OF {
    $FDA:
        PERM Object-class    {OBJECT_CLASS_ID_OF(TableArea)}
    $FPDA:
        REQ  Object-class    {OBJECT_CLASS_ID_OF(TableArea)}
},
REQ  Subordinates
        {SUB_ID_OF(RowArea)+,
        SUB_ID_OF(TableLabel)+,
        SUB_ID_OF(TableHeader)},
PERM Position
PERM Dimensions
        {REQ #horizontal-dimension
        {REQ #fixed-dimension {ANY_VALUE}},
        REQ #vertical-dimension
        {REQ #fixed-dimension {ANY_VALUE}}},
        {REQ #constraint-name {"36"},
        PERM #external-data {ANY_VALUE}}

SPECIFIC_AND_GENERIC:
PERM Transparency
PERM Colour
PERM Border
        {ANY_VALUE},
        {ANY_VALUE},
        {ANY_VALUE}
}

```

7.4.3.37 TableHeader

```

:ANY-FRAME-VARIABLE {
GENERIC:
CASE $DAC OF {
    $PDA-FPDA:
        REQ  Generator-for-subordinates    {$TableHeaderGFS}
},
REQ  Position
        {REQ #variable-position {
        PERM #offset    {ANY_VALUE},
        PERM #separation {ANY_VALUE},
        PERM #alignment {ANY_VALUE},
        PERM #fill-order {'normal-order'}}},
REQ  Dimensions
        {REQ #horizontal-dimension
        {REQ #fixed-dimension {ANY_VALUE}},
        REQ #vertical-dimension
        {REQ #rule-b {ANY_VALUE}
        {REQ #fixed-dimension {ANY_VALUE}}},
        {'270-degrees'},
PERM Layout-path
REQ  Application-comments
        {REQ #constraint-name {"34"},
        PERM #external-data {ANY_VALUE}}
}

```

SPECIFIC:

CASE \$DAC OF {	
\$FDA:	PERM Object-class {OBJECT_CLASS_ID_OF(TableHeader)}
\$FPDA:	REQ Object-class {OBJECT_CLASS_ID_OF(TableHeader)}
},	
REQ Subordinates	{SUB_ID_OF(SourcedContentFixed)+},
PERM Position	{ANY_VALUE},
PERM Dimensions	{REQ #horizontal-dimension {REQ #fixed-dimension {ANY_VALUE}}, REQ #vertical-dimension {REQ #fixed-dimension {ANY_VALUE}}},
PERM Application-comments	{REQ #constraint-name {"34"}, PERM #external-data {ANY_VALUE}}

SPECIFIC_AND_GENERIC:

PERM Transparency	{ANY_VALUE},
PERM Colour	{ANY_VALUE},
PERM Border	{ANY_VALUE}

}

7.4.3.38 TableLabel

:ANY-FRAME-VARIABLE {

GENERIC:

CASE \$DAC OF {	
\$PDA-FPDA:	REQ Generator-for-subordinates {\$TableLabelGFS}
},	
REQ Position	{REQ #variable-position { PERM #offset {ANY_VALUE}, PERM #separation {ANY_VALUE}, PERM #alignment {ANY_VALUE}, PERM #fill-order {'normal-order'}}},
REQ Dimensions	{REQ #horizontal-dimension {REQ #fixed-dimension {ANY_VALUE}}, REQ #vertical-dimension {REQ #rule-b {ANY_VALUE} {REQ #fixed-dimension {ANY_VALUE}}},
PERM Layout-path	{'270-degrees'},
REQ Application-comments	{REQ #constraint-name {"37"}, PERM #external-data {ANY_VALUE}}

SPECIFIC:

CASE \$DAC OF {	
\$FDA:	PERM Object-class {OBJECT_CLASS_ID_OF(TableLabel)}
\$FPDA:	REQ Object-class {OBJECT_CLASS_ID_OF(TableLabel)}
},	
REQ Subordinates	{SUB_ID_OF(TableLabelContent)+, SUB_ID_OF(CompositeTableLabel)},
PERM Position	{ANY_VALUE},
PERM Dimensions	{REQ #horizontal-dimension {REQ #fixed-dimension {ANY_VALUE}},


```

        REQ #vertical-dimension
            {REQ #fixed-dimension {ANY_VALUE}}},
    PERM Application-comments {REQ #constraint-name {"37"},
        PERM #external-data {ANY_VALUE}}
SPECIFIC_AND_GENERIC:
    PERM Transparency {ANY_VALUE},
    PERM Colour {ANY_VALUE},
    PERM Border {ANY_VALUE}
}

```

7.4.3.39 CompositeTableLabel

```

:ANY-FRAME-VARIABLE {
GENERIC:
    CASE $DAC OF {
        $PDA-FPDA:
            REQ Generator-for-subordinates {$CompositeTableLabelGFS}
    },
    REQ Position {REQ #fixed-position {
        REQ #horizontal-position {ANY_VALUE},
        REQ #vertical-position {ANY_VALUE}}},
    REQ Dimensions {REQ #horizontal-dimension
        {REQ #fixed-dimension {ANY_VALUE}},
        REQ #vertical-dimension
            {REQ #rule-b {ANY_VALUE}
            |REQ #fixed-dimension {ANY_VALUE}
            |REQ #maximum-size {'applies'}}},
    PERM Layout-path {'270-degrees'},
    REQ Application-comments {REQ #constraint-name {"38"},
        PERM #external-data {ANY_VALUE}}
SPECIFIC:
    CASE $DAC OF {
        $FDA:
            PERM Object-class {OBJECT_CLASS_ID_OF
                (CompositeTableLabel)}
        $FPDA:
            REQ Object-class {OBJECT_CLASS_ID_OF
                (CompositeTableLabel)}
    },
    REQ Subordinates {SUB_ID_OF(LabelComponent)+},
    PERM Position {ANY_VALUE},
    PERM Dimensions {REQ #horizontal-dimension
        {REQ #fixed-dimension {ANY_VALUE}},
        REQ #vertical-dimension
            {REQ #fixed-dimension {ANY_VALUE}}},
    PERM Imaging-order {ANY_VALUE},
    PERM Application-comments {REQ #constraint-name {"38"},
        PERM #external-data {ANY_VALUE}}
SPECIFIC_AND_GENERIC:
    PERM Transparency {ANY_VALUE},
    PERM Colour {ANY_VALUE},
    PERM Border {ANY_VALUE}
}

```

7.4.3.40 LabelComponent

```
:ANY-FRAME-VARIABLE {
GENERIC:
    CASE $DAC OF {
        $PDA-FPDA:
            REQ Generator-for-subordinates    {$LabelComponentGFS}
        }.
    REQ Position                            {REQ #variable-position {
                                                PERM #offset {ANY_VALUE},
                                                PERM #separation {ANY_VALUE},
                                                PERM #alignment {ANY_VALUE},
                                                PERM #fill-order {'normal-order'}}},
    REQ Dimensions                          {REQ #horizontal-dimension
                                                {REQ #fixed-dimension {ANY_VALUE}},
                                                REQ #vertical-dimension
                                                {REQ #rule-b {ANY_VALUE}
                                                {REQ #fixed-dimension {ANY_VALUE}
                                                {REQ #maximum-size {'applies'}}}},
    PERM Layout-path                        {'270-degrees'},
    REQ Application-comments                {REQ #constraint-name {"39"},
                                                PERM #external-data {ANY_VALUE}}
SPECIFIC:
    CASE $DAC OF {
        $FDA:
            PERM Object-class    {OBJECT_CLASS_ID_OF(LabelComponent))
        $FPDA:
            REQ Object-class    {OBJECT_CLASS_ID_OF(LabelComponent))
        }.
    REQ Subordinates            {SUB_ID_OF(TableLabelContent)+},
    PERM Position                {ANY_VALUE},
    PERM Dimensions              {REQ #horizontal-dimension
                                    {REQ #fixed-dimension {ANY_VALUE}},
                                    REQ #vertical-dimension
                                    {REQ #fixed-dimension {ANY_VALUE}}},
    PERM Application-comments    {REQ #constraint-name {"39"},
                                    PERM #external-data {ANY_VALUE}}
SPECIFIC_AND_GENERIC:
    PERM Transparency            {ANY_VALUE},
    PERM Colour                  {ANY_VALUE},
    PERM Border                  {ANY_VALUE}
}
```

7.4.3.41 RowArea

```
:ANY-FRAME-VARIABLE {
GENERIC:
    CASE $DAC OF {
        $PDA-FPDA:
            REQ Generator-for-subordinates    {$RowAreaGFS}
        }.
    REQ Position                            {REQ #variable-position {
                                                PERM #offset {ANY_VALUE},
                                                PERM #separation {ANY_VALUE},
```

```

        PERM #alignment {ANY_VALUE},
        PERM #fill-order {'normal-order'}}},
REQ  Dimensions      {REQ #horizontal-dimension
                      {REQ #fixed-dimension {ANY_VALUE}},
                      REQ #vertical-dimension
                      {REQ #rule-b {ANY_VALUE}
                      {REQ #fixed-dimension {ANY_VALUE}}},
PERM  Layout-path    {'270-degrees'},
REQ  Application-comments {REQ #constraint-name {"40"},
PERM #external-data {ANY_VALUE}}

SPECIFIC:
CASE $DAC OF {
  $FDA:
    PERM Object-class {OBJECT_CLASS_ID_OF(RowArea)}
  $FPDA:
    REQ  Object-class {OBJECT_CLASS_ID_OF(RowArea)}
  }.
REQ  Subordinates    {SUB_ID_OF(Cell)+,
                      SUB_ID_OF(SubRowGroup)},
PERM Position        {ANY_VALUE},
PERM Dimensions      {REQ #horizontal-dimension
                      {REQ #fixed-dimension {ANY_VALUE}},
                      REQ #vertical-dimension
                      {REQ #fixed-dimension {ANY_VALUE}}},
PERM Application-comments {REQ #constraint-name {"40"},
PERM #external-data {ANY_VALUE}}

SPECIFIC_AND_GENERIC:
PERM Transparency    {ANY_VALUE},
PERM Colour          {ANY_VALUE},
PERM Border          {ANY_VALUE}
}

```

7.4.3.42 Cell

```

:ANY-FRAME-VARIABLE {
GENERIC:
  REQ  Position      {REQ #fixed-position {
                      REQ #horizontal-position {ANY_VALUE},
                      REQ #vertical-position {ANY_VALUE}}},
  REQ  Dimensions    {REQ #horizontal-dimension
                      {REQ #fixed-dimension {ANY_VALUE}},
                      REQ #vertical-dimension
                      {REQ #rule-b {ANY_VALUE}
                      {REQ #fixed-dimension {ANY_VALUE}
                      {REQ #maximum-size {'applies'}}}},
  PERM Layout-path   {'270-degrees'},
  REQ  Permitted-categories {ANY_STRING...},
      -- category name for tables should be specified --
  REQ  Application-comments {REQ #constraint-name {"41"},
PERM #external-data {ANY_VALUE}}

SPECIFIC:
CASE $DAC OF {
  $FDA:

```

```

        PERM Object-class {OBJECT_CLASS_ID_OF(Cell)}
$FPDA:
        REQ Object-class {OBJECT_CLASS_ID_OF(Cell)}
    },
    REQ Subordinates {SUB_ID_OF(SpecificBlock)},
    PERM Position {ANY_VALUE},
    PERM Dimensions {REQ #horizontal-dimension
        {REQ #fixed-dimension {ANY_VALUE}},
        REQ #vertical-dimension
        {REQ #fixed-dimension {ANY_VALUE}}},
    PERM Imaging-order {ANY_VALUE},
    PERM Application-comments {REQ #constraint-name {"41"},
        PERM #external-data {ANY_VALUE}}
SPECIFIC_AND_GENERIC:
    PERM Transparency {ANY_VALUE},
    PERM Colour {ANY_VALUE},
    PERM Border {ANY_VALUE}
}

```

7.4.3.43 SubRowGroup

```

:ANY-FRAME-VARIABLE {
GENERIC:
    CASE $DAC OF {
        $PDA-FPDA:
            REQ Generator-for-subordinates {$SubRowGroupGFS}
        },
    REQ Position {REQ #fixed-position {
        REQ #horizontal-position {ANY_VALUE},
        REQ #vertical-position {ANY_VALUE}}},
    REQ Dimensions {REQ #horizontal-dimension
        {REQ #fixed-dimension {ANY_VALUE}},
        REQ #vertical-dimension
        {REQ #rule-b {ANY_VALUE}
        {REQ #fixed-dimension {ANY_VALUE}
        {REQ #maximum-size {'applies'}}},
        {'270-degrees'},
    REQ Application-comments {REQ #constraint-name {"42"},
        PERM #external-data {ANY_VALUE}}
SPECIFIC:
    CASE $DAC OF {
        $FDA:
            PERM Object-class {OBJECT_CLASS_ID_OF(SubRowGroup)}
        $FPDA:
            REQ Object-class {OBJECT_CLASS_ID_OF(SubRowGroup)}
        },
    REQ Subordinates {SUB_ID_OF(SubRow)+},
    PERM Position {ANY_VALUE},
    PERM Dimensions {REQ #horizontal-dimension
        {REQ #fixed-dimension {ANY_VALUE}},
        REQ #vertical-dimension
        {REQ #fixed-dimension {ANY_VALUE}}},
    PERM Application-comments {REQ #constraint-name {"42"},
        PERM #external-data {ANY_VALUE}}
}

```



```

SPECIFIC_AND_GENERIC:
  PERM Transparency {ANY_VALUE},
  PERM Colour {ANY_VALUE},
  PERM Border {ANY_VALUE}
}

```

7.4.3.44 SubRow

```

:ANY-FRAME-VARIABLE {
  GENERIC:
    CASE $DAC OF {
      $PDA-FPDA:
        REQ Generator-for-subordinates {$SubRowGFS}
      },
      REQ Position {REQ #variable-position {
        PERM #offset {ANY_VALUE},
        PERM #separation {ANY_VALUE},
        PERM #alignment {ANY_VALUE},
        PERM #fill-order {'normal-order'}}},
      REQ Dimensions {REQ #horizontal-dimension
        {REQ #fixed-dimension {ANY_VALUE}},
        REQ #vertical-dimension
        {REQ #rule-b {ANY_VALUE}
        {REQ #fixed-dimension {ANY_VALUE}
        {REQ #maximum-size {'applies'}}}},
      PERM Layout-path {'270-degrees'},
      REQ Application-comments {REQ #constraint-name {"43"},
        PERM #external-data {ANY_VALUE}}
    }
  SPECIFIC:
    CASE $DAC OF {
      $FDA:
        PERM Object-class {OBJECT_CLASS_ID_OF(SubRow)}
      $FPDA:
        REQ Object-class {OBJECT_CLASS_ID_OF(SubRow)}
      },
      REQ Subordinates {SUB_ID_OF(Cell)+},
      PERM Position {ANY_VALUE},
      PERM Dimensions {REQ #horizontal-dimension
        {REQ #fixed-dimension {ANY_VALUE}},
        REQ #vertical-dimension
        {REQ #fixed-dimension {ANY_VALUE}}},
      PERM Application-comments {REQ #constraint-name {"43"},
        PERM #external-data {ANY_VALUE}}
    }
  SPECIFIC_AND_GENERIC:
    PERM Transparency {ANY_VALUE},
    PERM Colour {ANY_VALUE},
    PERM Border {ANY_VALUE}
}

```

7.4.3.45 TableLabelContent

```

:ANY-FRAME-VARIABLE {
  GENERIC:

```

```

CASE $DAC OF {
  $PDA-FPDA:
    REQ Logical-source {OBJECT_CLASS_ID_OF(CommonContent)},
    REQ Position {REQ #fixed-position
      {REQ #horizontal-position {ANY_VALUE},
      REQ #vertical-position {ANY_VALUE}}},
    REQ Dimensions {REQ #horizontal-dimension
      {REQ #fixed-dimension {ANY_VALUE}},
      REQ #vertical-dimension
      {REQ #fixed-dimension {ANY_VALUE}
      {REQ #maximum-size {'applies'}}},
      {'270-degrees'}
    PERM Layout-path
  },
  REQ Application-comments {REQ #constraint-name {"44"},
    PERM #external-data {ANY_VALUE}}
SPECIFIC:
  CASE $DAC OF {
    $FDA:
      PERM Object-class {OBJECT_CLASS_ID_OF
        (TableLabelContent))
    $FPDA:
      REQ Object-class {OBJECT_CLASS_ID_OF
        (TableLabelContent))
  },
  REQ Subordinates {SUB_ID_OF(SpecificBlock)+},
  PERM Application-comments {REQ #constraint-name {"44"},
    PERM #external-data {ANY_VALUE}}
}

```

7.4.3.46 FormEntryArea

```

:ANY-FRAME-VARIABLE {
GENERIC:
  CASE $DAC OF {
    $PDA-FPDA:
      REQ Position {REQ #fixed-position
        {REQ #horizontal-position {ANY_VALUE},
        REQ #vertical-position {ANY_VALUE}}},
      REQ Dimensions {REQ #horizontal-dimension
        {REQ #fixed-dimension {ANY_VALUE}},
        REQ #vertical-dimension
        {REQ #fixed-dimension {ANY_VALUE}}},
        {'270-degrees'}
      PERM Layout-path
    },
    REQ Application-comments {REQ #constraint-name {"45"},
      PERM #external-data {ANY_VALUE}}
SPECIFIC:
  CASE $DAC OF {
    $FDA:
      PERM Object-class {OBJECT_CLASS_ID_OF(FormEntryArea))
    $FPDA:
      REQ Object-class {OBJECT_CLASS_ID_OF(FormEntryArea))
  },
  REQ Subordinates {SUB_ID_OF(SpecificBlock)},

```

```

    PERM Application-comments      {REQ #constraint-name {"45"},
                                   PERM #external-data {ANY_VALUE}}
SPECIFIC_AND_GENERIC:
    PERM Permitted-categories     {ANY_STRING...}
}

```

7.5 Layout style constituent constraints

7.5.1 Macro definitions

```

DEFINE(SameLayoutObject,"
    REQ {REQ #logical-object {<object-id-expr>::=PREC-OBJ(CURR-OBJ);}
        | REQ #logical-object {'null'}}.
    PERM #layout-object {'page'          -- to layout object type --
                        | ANY_STRING      -- to layout category --
                        | OBJECT_CLASS_ID_OF(ColumnFixed)
                        | OBJECT_CLASS_ID_OF(ColumnVariable)
                        | OBJECT_CLASS_ID_OF(CompositeColumnFixed)
                        | OBJECT_CLASS_ID_OF(CompositeColumnVariable))
")

```

7.5.2 Factor constraints

FACTOR ANY-LAYOUT-STYLE

```

{
    REQ   Layout-style-identifier      {ANY_VALUE}.
    PERM  User-visible-name             {ANY_STRING}.
    PERM  User-readable-comments       {ANY_STRING}
}

```

7.5.3 Constituent constraints

7.5.3.1 LStyle1

:ANY-LAYOUT-STYLE {

-- This layout style is only applicable to composite logical constituent constraints including the Passage, NumberedSegment, Title, Caption, Paragraph, Phrase, Footnote, Figure, Reference, Description, NumberedList, UnNumberedList, DefinitionList, ListItem, ListTerm --

CASE \$GLAS OF{

 \$COMPLETE:

```

    PERM  Indivisibility                {ANY_VALUE}.
    PERM  Layout-object-class           {ANY_VALUE}.
    PERM  New-layout-object             {ANY_VALUE}.
    PERM  Same-layout-object            {$SameLayoutObject}.
    PERM  Synchronization               {ANY_VALUE}

```

VOID:

```

    PERM  Indivisibility                {ANY_STRING --to layout category--

```

PERM	New-layout-object	['page' --to layout object type-- 'null'], { ANY_STRING --to layout category-- 'page' --to layout object type-- 'null'}, REQ { REQ #logical-object { <object-id-expr>::=PREC-OBJ(CURR-OBJ); REQ #logical-object { 'null'}}, PERM #layout-object { ANY_STRING --to layout category-- 'page' --to layout object type-- }, { ANY_VALUE
PERM	Same-layout-object	
PERM	Synchronization	

}}

7.5.3.2 LStyle2

:ANY-LAYOUT-STYLE {

-- This layout style is only applicable to the constituent constraints Number and BodyText --

CASE \$GLAS OF{

 \$COMPLETE:

PERM	Block-alignment	{ ANY_VALUE},
PERM	Concatenation	{ ANY_VALUE},
PERM	Indivisibility	{ ANY_VALUE},
PERM	Layout-category	{ ANY_VALUE},
PERM	Layout-object-class	{ ANY_VALUE},
PERM	New-layout-object	{ ANY_VALUE},
PERM	Offset	{ ANY_VALUE},
PERM	Same-layout-object	{ \$SameLayoutObject},
PERM	Separation	{ ANY_VALUE},
PERM	Synchronization	{ ANY_VALUE}

 VOID:

PERM	Block-alignment	{ ANY_VALUE},
PERM	Concatenation	{ ANY_VALUE},
PERM	Indivisibility	{ ANY_STRING --to layout category-- 'page' --to layout object type-- 'null'},
PERM	Layout-category	{ ANY_VALUE},
PERM	New-layout-object	{ ANY_STRING --to layout category-- 'page' --to layout object type-- 'null'},
PERM	Offset	{ ANY_VALUE},
PERM	Same-layout-object	REQ { REQ #logical-object { <object-id-expr>::=PREC-OBJ(CURR-OBJ); REQ #logical-object { 'null'}}, PERM #layout-object { ANY_STRING --to layout category-- 'page' --to layout object type-- },
PERM	Separation	{ ANY_VALUE},
PERM	Synchronization	{ ANY_VALUE}


```
}}
```

7.5.3.3 LStyle3

```
:ANY-LAYOUT-STYLE{
```

-- This layout style is used only for the constituent constraints CommonText, PageNumber, TableNumber, CurrentInstance, CommonNumber and CommonReference --

```
PERM Block-alignment      {ANY_VALUE}.
PERM Concatenation        {ANY_VALUE}.
PERM Offset                {ANY_VALUE}.
PERM Separation            {ANY_VALUE}
}
```

7.5.3.4 LStyle4

-- This layout style is not used --

7.5.3.5 LStyle5

```
:ANY-LAYOUT-STYLE {
```

-- This layout style is used only for the constituent constraints BodyRaster and BodyGeometric --

```
CASE $GLAS OF{
```

```
  $COMPLETE:
```

```
    PERM Block-alignment      {ANY_VALUE}.
    PERM Layout-category      {ANY_VALUE}.
    PERM Layout-object-class  {ANY_VALUE}.
    PERM New-layout-object    {ANY_VALUE}.
    PERM Offset                {ANY_VALUE}.
    PERM Same-layout-object    {$SameLayoutObject},
    PERM Separation            {ANY_VALUE}.
    PERM Synchronization      {ANY_VALUE}
```

```
  VOID:
```

```
    PERM Block-alignment      {ANY_VALUE}.
    PERM Layout-category      {ANY_VALUE}.
    PERM New-layout-object    {ANY_STRING --to layout category--
                              'page' --to layout object type--
                              'null'},
```

```
    PERM Offset                {ANY_VALUE}.
    PERM Same-layout-object    REQ {REQ #logical-object
                                   {<object-id-expr>::=PREC-OBJ(CURR-OBJ);}
                                   | REQ #logical-object {'null'}}.
    PERM #layout-object
      {ANY_STRING --to layout category--
      'page' --to layout object type--
```

```
    },
```

```
    PERM Separation            {ANY_VALUE}.
    PERM Synchronization      {ANY_VALUE}
```

```
}}
```

7.5.3.6 LStyle6

:ANY-LAYOUT-STYLE {

-- This layout style is used only for the constituent constraint FootnoteText --

CASE \$GLAS OF{

\$COMPLETE:

PERM	Indivisibility	{ANY_VALUE},
PERM	Block-alignment	{ANY_VALUE},
PERM	Concatenation	{ANY_VALUE},
REQ	Layout-category	{\$FOOTNOTECATEGORY},
PERM	Offset	{ANY_VALUE},
PERM	Separation	{ANY_VALUE}

VOID:

PERM	Indivisibility	{ANY_STRING --to layout category-- 'page' --to layout object type-- 'null'},
PERM	Block-alignment	{ANY_VALUE},
PERM	Concatenation	{ANY_VALUE},
REQ	Layout-category	{\$FOOTNOTECATEGORY},
PERM	Offset	{ANY_VALUE},
PERM	Separation	{ANY_VALUE}

}}

7.5.3.7 LStyle7

-- This layout style is not used. --

7.5.3.8 LStyle8

:ANY-LAYOUT-STYLE {

-- This layout style is used only for the constituent constraints CommonRaster and CommonGeometric --

PERM	Block-alignment	{ANY_VALUE},
PERM	Offset	{ANY_VALUE},
PERM	Separation	{ANY_VALUE}

7.5.3.9 LStyle9

:ANY-LAYOUT-STYLE {

-- This layout style is only applicable to the constituent constraint FootnoteNumber --

PERM	Block-alignment	{ANY_VALUE},
REQ	Layout-category	{\$FOOTNOTECATEGORY},

```

PERM Offset {ANY_VALUE},
PERM Separation {ANY_VALUE}
}

```

7.5.3.10 LStyle10

```
:ANY-LAYOUT-STYLE {
```

```
-- This layout style is only applicable to the constituent constraints FootnoteReference and
ReferencedContent --
```

```
CASE $GLAS OF{
```

```
$COMPLETE:
```

```

PERM Block-alignment {ANY_VALUE},
PERM Concatenation {ANY_VALUE},
PERM Indivisibility {ANY_VALUE},
PERM Layout-category {ANY_VALUE},
PERM Offset {ANY_VALUE},
PERM Same-layout-object {$SameLayoutObject}.
PERM Separation {ANY_VALUE}

```

```
VOID:
```

```

PERM Block-alignment {ANY_VALUE},
PERM Concatenation {ANY_VALUE},
PERM Indivisibility {ANY_STRING --to layout category--
|'page' --to layout object type--
|'null'}.

PERM Layout-category {ANY_VALUE},
PERM Offset {ANY_VALUE},
PERM Same-layout-object REQ {REQ #logical-object
{<object-id-expr>::=PREC-OBJ(CURR-OBJ);}
| REQ #logical-object {'null'}}.
PERM #layout-object
{ANY_STRING --to layout category--
|'page' --to layout object type--
}.
PERM Separation {ANY_VALUE}

```

```
}}
```

7.5.3.11 LStyle11

```
:ANY-LAYOUT-STYLE {
```

```
-- This layout style is used only for the constituent constraint FootnoteBody --
```

```
CASE $GLAS OF{
```

```
$COMPLETE:
```

```

PERM Indivisibility {ANY_VALUE},
PERM Same-layout-object {$SameLayoutObject}.
PERM Synchronization {ANY_VALUE}

```

```
VOID:
```

```

PERM Indivisibility {ANY_STRING --to layout category--
|'page' --to layout object type--
}

```

```

    PERM      Same-layout-object
    REQ {REQ #logical-object
        {<object-id-expr>::=PREC-OBJ(CURR-OBJ);}
        | REQ #logical-object {'null'}},
    PERM #layout-object
        {ANY_STRING --to layout category--
         'page' --to layout object type--}
    },
    PERM      Synchronization
    {ANY_VALUE}
}}

```

7.5.3.12 LStyle12

:ANY-LAYOUT-STYLE {

-- This layout style is used only for the constituent constraint Artwork --

CASE \$GLAS OF{

\$COMPLETE:

```

    PERM      Indivisibility
    {ANY_STRING -- to layout category --
     'page' -- to layout object type --
     'null'},

    PERM      Layout-object-class
    PERM      New-layout-object
    {OBJECT_CLASS_ID_OF(CompositeArtwork)},
    {OBJECT_CLASS_ID_OF(CompositeColumnFixed)}
    {OBJECT_CLASS_ID_OF(CompositeColumnVariable)}
    -- to layout object class --
    {ANY_STRING -- to layout category --
     'page' -- to layout object type --
     'null'},
    {ANY_VALUE}

    PERM      Synchronization
VOID:
    PERM      Indivisibility
    {ANY_STRING --to layout category--
     'page' --to layout object type--
     'null'},

    PERM      New-layout-object
    {ANY_STRING --to layout category--
     'page' --to layout object type--
     'null'},

    PERM      Synchronization
    {ANY_VALUE}
}}

```

7.5.3.13 LStyleT1

:ANY-LAYOUT-STYLE {

-- This layout style is used only for the constituent constraint Form --

```

REQ  Layout-object-class      {OBJECT_CLASS_ID_OF(FormArea)}
}

```


7.5.3.14 LStyleT2

:ANY-LAYOUT-STYLE {

-- This layout style is used only for the constituent constraint EntryElement --

-- In the case of Form, the following attribute must be specified. --

REQ Layout-object-class {OBJECT_CLASS_ID_OF(FormEntryArea)}

-- In the case of Table, both of the following attributes must be specified. --

REQ New-layout-object {OBJECT_CLASS_ID_OF(Cell) -- to layout object class --

|ANY_STRING -- to layout category --},

REQ Indivisibility {OBJECT_CLASS_ID_OF(Cell) -- to layout object class --

|ANY_STRING -- to layout category -- }

}

7.5.3.15 LStyleT3

:ANY-LAYOUT-STYLE {

-- This layout style is used only for the constituent constraint EntryGroup --

REQ Layout-object-class {OBJECT_CLASS_ID_OF(EntryGroupArea)}

}

7.5.3.16 LStyleT4

:ANY-LAYOUT-STYLE {

-- This layout style is used only for a logical object class of the constituent constraint Table --

REQ New-layout-object {OBJECT_CLASS_ID_OF(TableArea)},

PERM Indivisibility {OBJECT_CLASS_ID_OF(TableArea)

-- to layout object class --

|ANY_STRING -- to layout category --

|'page' -- to layout object type --

|'null' },

PERM Same-layout-object {\$SameLayoutObject}

}

7.5.3.17 LStyleT5

:ANY-LAYOUT-STYLE {

-- This layout style is used only for the constituent constraint Row --

REQ New-layout-object {OBJECT_CLASS_ID_OF(RowArea)},

PERM Indivisibility {OBJECT_CLASS_ID_OF(RowArea)

-- to layout object class --

|ANY_STRING -- to layout category --

|'page' -- to layout object type --

|'null' }

}

7.5.3.18 LStyleT6

:ANY-LAYOUT-STYLE {

-- This layout style is used only for the constituent constraint TableComponent --

REQ Layout-object-class {OBJECT_CLASS_ID_OF(SubRowGroup)}
}

7.5.3.19 LStyleT7

:ANY-LAYOUT-STYLE {

-- This layout style is used only for the constituent constraint RowComponent --

REQ New-layout-object {OBJECT_CLASS_ID_OF(SubRow)},
PERM Indivisibility {OBJECT_CLASS_ID_OF(SubRow)
-- to layout object class --
|ANY_STRING -- to layout category --
|'null'}
}

7.5.3.20 LStyleT8

:ANY-LAYOUT-STYLE {

-- This layout style is used only for a logical object of the constituent constraint Table --

PERM Indivisibility {OBJECT_CLASS_ID_OF(TableArea)
-- to layout object class --
|ANY_STRING -- to layout category --
|'page' -- to layout object type --
|'null'},
PERM Same-layout-object {\$SameLayoutObject}
}

7.5.3.21 LStyleT9

:ANY-LAYOUT-STYLE {

-- This layout style is used only for the constituent constraints EntryText,EntryRaster,EntryGeometric. --

PERM Block-alignment {ANY_VALUE},
PERM Layout-category {ANY_VALUE},
PERM Offset {ANY_VALUE}
}

7.6 Presentation style constituent constraints

7.6.1 Macro definitions

-- No macro definitions are applicable to this subclause. --

7.6.2 Factor constraints

FACTOR ANY-PRESENTATION-STYLE

```
{
REQ   Presentation-style-identifier    {ANY_VALUE},
PERM  User-readable-comments          {ANY_STRING},
PERM  User-visible-name                {ANY_STRING},
PERM  Border                          {ANY_VALUE},
PERM  Colour                          {ANY_VALUE},
PERM  Transparency                     {ANY_VALUE}
}
```

7.6.3 Constituent constraints

7.6.3.1 PStyle1

:ANY-PRESENTATION-STYLE {

-- This style is used for the constituent constraints BodyText, Number, FootnoteNumber, FootnoteReference, FootnoteText, EntryText, ReferencedContent, GenericBlock and SpecificBlock. --

```
PERM Presentation-attributes {
  PERM #character-attributes {
    PERM #alignment          {ANY_VALUE},
    PERM #character-fonts    {ANY_VALUE},
    PERM #character-orientation {ANY_VALUE},
    PERM #character-path      {ANY_VALUE},
    PERM #character-spacing   {ANY_VALUE},
    PERM #code-extension-announcers {$CDEXTEN},
    PERM #first-line-offset    {ANY_VALUE},
    PERM #graphic-character-sets {$PERMIT-GRCHAR},
    PERM #graphic-character-subrepertoire {ANY_VALUE},
    PERM #graphic-rendition    {$GRAPHICRENDITIONS},
    PERM #indentation          {ANY_VALUE},
    PERM #itemization          {ANY_VALUE},
    PERM #kerning-offset       {ANY_VALUE},
    PERM #line-layout-table    {ANY_VALUE},
    PERM #line-progression     {ANY_VALUE},
    PERM #line-spacing         {ANY_VALUE},
    PERM #orphan-size          {ANY_VALUE},
    PERM #pairwise-kerning     {ANY_VALUE},
    PERM #proportional-line-spacing {ANY_VALUE},
    PERM #widow-size           {ANY_VALUE},
```

```

        PERM #initial-offset          {ANY_VALUE},
        PERM #formatting-indicator   {ANY_VALUE}
    }
}}

```

7.6.3.2 PStyle2

```
:ANY-PRESENTATION-STYLE {
```

```
--      This style is used for the constituent constraints BodyGeometric, CommonGeometric,
EntryGeometric, GenericBlock and SpecificBlock. --
```

```

PERM Presentation-attributes {
    PERM #geometric-graphics-attributes {ANY_VALUE}
}

```

7.6.3.3 PStyle3

```
:ANY-PRESENTATION-STYLE {
```

```
--      This style is used for the constituent constraints BodyRaster, CommonRaster, EntryRaster,
GenericBlock and SpecificBlock. --
```

```

PERM Presentation-attributes {
    PERM #raster-graphics-attributes {
        PERM #pel-path          {ANY_VALUE},
        PERM #line-progression   {ANY_VALUE},
        PERM #pel-spacing        {ANY_VALUE},
        PERM #spacing-ratio      {ANY_VALUE},
        PERM #clipping           {ANY_VALUE},
        PERM #image-dimensions   {ANY_VALUE}
    }
}

```

7.6.3.4 PStyle4

```
:ANY-PRESENTATION-STYLE {
```

```
--      This style is used for the constituent constraints CommonText, PageNumber, TableNumber,
CommonReference, CurrentInstance and SpecificBlock. --
```

```

PERM Presentation-attributes {
    PERM #character-attributes {
        PERM #alignment          {ANY_VALUE},
        PERM #character-fonts     {ANY_VALUE},
        PERM #character-orientation {ANY_VALUE},
        PERM #character-path      {ANY_VALUE},
        PERM #character-spacing    {ANY_VALUE},
        PERM #code-extension-announcers {$CDEXTEN},
        PERM #first-line-offset    {ANY_VALUE},
        PERM #graphic-character-sets {$PERMIT-GRCHAR},
        PERM #graphic-character-subrepertoire {ANY_VALUE},
    }
}

```



```

    PERM #graphic-rendition      {$GRAPHICRENDITIONS},
    PERM #indentation            {ANY_VALUE},
    PERM #itemization            {ANY_VALUE},
    PERM #kerning-offset         {ANY_VALUE},
    PERM #line-layout-table      {ANY_VALUE},
    PERM #line-progression       {ANY_VALUE},
    PERM #line-spacing           {ANY_VALUE},
    PERM #pairwise-kerning       {ANY_VALUE},
    PERM #proportional-line-spacing {ANY_VALUE},
    PERM #initial-offset         {ANY_VALUE},
    PERM #formatting-indicator   {ANY_VALUE}
  }
}

```

7.7 Content portion constituent constraints

7.7.1 Macro definitions

```

DEFINE(T6,          "ASN.1 {2 8 3 7 0}")
DEFINE(T41D,        "ASN.1 {2 8 3 7 1}")
DEFINE(T42D,        "ASN.1 {2 8 3 7 2}")
DEFINE(Bitmap,      "ASN.1 {2 8 3 7 3}")

```

7.7.2 Factor constraints

```

FACTOR ANY-CONTENT {

CASE $DAC OF {
  $FDA :
    REQ Content-identifier-layout {ANY_VALUE}

  $PDA :
    REQ Content-identifier-logical {ANY_VALUE}
    -- This attribute is specified, if the content portion is associated with
    a basic logical object or a basic logical object class. --
    |REQ Content-identifier-layout {ANY_VALUE}
    -- This attribute is specified, if the content portion is associated with
    a basic layout object class. --

  $FPDA :
    REQ Content-identifier-layout {ANY_VALUE},
    REQ Content-identifier-logical {ANY_VALUE}
    -- Both attributes are specified, if the content portion is associated with
    a basic logical object and a basic layout object. --
    |REQ Content-identifier-layout {ANY_VALUE}
    -- This attribute is specified, if the content portion is associated with
    a basic layout object class. --
    |REQ Content-identifier-logical {ANY_VALUE}
    -- This attribute is specified, if the content portion is associated with
    a basic logical object class. --
}

```

```

    }.
    PERM Alternative-representation {ANY_STRING}
}

```

7.7.3 Constituent constraints

7.7.3.1 Character-content-portion

```

:ANY-CONTENT {
  PERM Type-of-coding {ASN.1{2 8 3 6 0}},
  PERM Content-information {CHARACTER
    {
-- Shared Control Functions --
      #CR
      | #GCC {ANY_VALUE}
      | #IGS {ANY_VALUE}
      | #LF
      | #PLD
      | #PLU
      | #SCS {ANY_VALUE}
      | #SGR {$GRAPHICRENDITIONS}
      | #SHS {ANY_VALUE}
      | #SLS {ANY_VALUE}
      | #SRS {ANY_VALUE}
      | #STAB {ANY_VALUE}
      | #SUB
      | #SVS {ANY_VALUE}
      | #VPB {ANY_VALUE}
      | #VPR {ANY_VALUE}

-- Layout Control Functions --
      | #HPB {ANY_VALUE}
      | #HPR {ANY_VALUE}
      | #JFY {0}
      | #SACS {ANY_VALUE}
      | #SRCS {ANY_VALUE}
      | #SSW {ANY_VALUE}

-- Logical Control Functions --
      | #BPH
      | #NBH
      | #PTX {ANY_VALUE}

-- Delimiter Functions --
      | #SOS
      | #ST

-- Space --
      | #SP

-- Code extension control functions --
      | #LS0
    }
  }
}

```

```

| #LS1R
| #LS2R
| #LS3R
| #SS2
| #SS3
| #ESC {$DEG-CORE-G0}
| #ESC {$DEG-646-G0}
| #ESC {$DEG-ANY-G1}
| #ESC {$DEG-ANY-G2}
| #ESC {$DEG-ANY-G3}
| #ESC {$DEG-EMPTY-G1}
|...}
}

```

7.7.3.2 Raster-graphics-content-portion

```

:ANY-CONTENT {
  PERM Type-of-coding          {$T6|$T41D|$T42D|$Bitmap},
  PERM Coding-attributes      {
    PERM #compression          {ANY_VALUE},
    PERM #number-of-lines      {>0},
    REQ  #number-of-pels-per-line {>=0}
  },
  PERM Content-information     {RASTER}
}

```

7.7.3.3 Geometric-graphics-content-portion

```

:ANY-CONTENT {
  PERM Type-of-coding          {ASN.1 {2 8 3 8 0}},
  PERM Content-information     {GEOMETRIC}
}

```

8 Interchange format

For conformance to this profile, the interchange format class A shall be used when applying ODIF, and the interchange format SDIF shall be used when applying ODL in conjunction with SDIF.

NOTE 30 - Interchange format SDIF applies to the ISP only.

8.1 Interchange format class A

8.1.1 Interchange format

The value of the document profile attribute "interchange format" for this interchange format is "if-a". This form of ODIF is defined in [CCITT Recommendation T.415 | ISO 8613-5].

8.1.2 DAP identifier

The value for the document profile attribute "document application profile" for this interchange format is represented by the following object identifier.

ASN.1 { 2 8 4 0 36 0 }

8.1.3 Encoding of application comments

The encoding of the attribute "application comments" is defined in this encoding as an octet string as specified in [CCITT Recommendation T.415 | ISO 8613-5]. This document application profile requires that the encoding within that octet string be in accordance with the ASN.1 syntax specified in the following module definition:

```
FOD_DAPSpecification
DEFINITIONS ::= BEGIN
EXPORTS    Appl-Comm-Encoding;

Appl-Comm-Encoding ::= SEQUENCE {
    constraint-name      [0] IMPLICIT PrintableString OPTIONAL,
    external-data        [1] IMPLICIT OCTETSTRING OPTIONAL }

END
```

8.1.4 Data lengths

The maximum length of data values of the type OCTET STRING, as defined in [ISO 8824 | CCITT Recommendation X.208], in data streams which may be encoded in accordance with this DAP is 32767 octets. If it is required to encode contents octets of greater length than this, constructed type encoding shall be used. That is, data values greater than 32767 in length shall be split into a sequence of strings shorter than 32767, each of which is encoded using a primitive type.

8.2 Interchange format SDIF

NOTE 31 - The subclause 8.2 applies to the ISP only.

8.2.1 Interchange format

The document profile attribute "interchange format" does not apply for this interchange format. This form of the interchange format is defined in Annex E of [CCITT Recommendation T.415 | ISO 8613-5]. [CCITT Recommendations T.416, T.417, and T.418 | ISO 8613-6, -7 and -8] contain additional specifications for this interchange format.

8.2.2 DAP identifier

The value for the attribute "document application profile" for this interchange format is represented by the following object identifier.

ASN.1 { 1 0 11182 0 36 0 }

NOTE 32 - There is no requirement to include a part number arc within the object identifier for the DAP.

8.2.3 Encoding of application comments

The encoding of the attribute "application comments" is defined in data stream conforming to this profile with this encoding with the following DTD definition:

```
<!-- Public document type definition. Typical invocation:
<!DOCTYPE fodapc PUBLIC "ISO/IEC 11182-1 : 1992//DTD
      Application Comments//EN" [ " ]>
-->
<!ELEMENT fodapc . - 0 (external?)>
<!ATTLIST fodapc consname CDATA #IMPLIED>
<!ELEMENT external - 0 {#PCDATA}>
<!ATTLIST external loc ENTITY #CONREF>
```

For example, a typical SUBDOC for representing the "application comments" of a Paragraph then would look like:

```
<!DOCTYPE fodapc PUBLIC "ISO/IEC 11182-1 : 1992//DTD Application Comments//EN">
<fodapc consname=6>
```

If the optionality of the attribute "fodapc" is specified in an earlier portion of the DTD, the invocation may be minimized further because the tag is not needed when "application comments" are not included as is the case here.

The content of external data appearing inline is restricted to parsable data. Referenced external data need not be parsable.

Annex A

Amendments and corrigenda

(This annex forms an integral part of this specification.)

A.1 Amendments

A.1.1 Amendments to the base standard

The amendments applicable to this CCITT Recommendation | ISP includes the ISO 8613 - Amendment 1: 1990. This amendment includes text to be included in ISO 8613-1 as the following annexes:

- Annex E: Use of ISO/IEC 10021 (MOTIS) to interchange documents conforming to ISO 8613;
- Annex F: Document Application Profile proforma and notation;
- Annex G: Conformance testing methodology;
- Annex H: Recording of documents conforming to ISO 8613 on flexible disk cartridges conforming to ISO 9293.

In addition, this amendment addresses the inclusion of the ISO 8613 Technical Corrigenda 1. This ISP does not include the following features of the amendment:

- addendum on security;
- addendum on styles;
- addendum on alternative representation;
- addendum on colour;
- addendum on tiled raster graphics.

A.1.2 Proposed changes to standards due to defects

Proposed Technical Corrigendum No. 44 to [ISO 8613 | T.410 recommendations] dated 15 December 1991 assumed to be ratified by ISO.

A.2 Corrigenda

A.2.1 Corrigenda to the CCITT Recommendation | ISP

There is no corrigendum specific to this CCITT Recommendation | ISP.

Annex B

Recommendations

(This annex does not form an integral part of this specification.)

B.1 Transfer methods for ODA

B.1.1 Conveyance of ODA over CCITT X.400-1984

This recommendation describes how ODA body parts are to be encoded for transmission over a CCITT X.400-1984 service.

An ODA body part is encoded as OdaBodyPart in the definition given below:

```
OdaBodyPart ::= SEQUENCE { OdaBodyPartParameters, OdaData }
OdaBodyPartParameters ::= SET {
    document-application-profile
        [0] IMPLICIT OBJECT IDENTIFIER,
    document-architecture-class
        [1] IMPLICIT INTEGER {
            formatted (0),
            processable (1),
            formatted-processable (2) }
OdaData ::= SEQUENCE OF Interchange-Data-Element
```

NOTE 33 - It is recommended to transfer an ODA document as a single body part with tag 12:

Oda [12] IMPLICIT OCTETSTRING

The content of the octet string is encoded as OdaBodyPart, defined above. However, this is out of the scope of this profile.

B.1.2 Conveyance of ODA over FTAM

This recommendation describes the FTAM Document Type to be used for minimal storage and transfer capabilities of ODA data streams. It is recognized that enhanced capabilities may at some point be added.

When using FTAM to transfer an ODA file, the FTAM-3, "ISO FTAM Unstructured Binary", document type shall be specified.

However, since files that do not contain ODA data streams can have the same document type, it is left up to the user of application programs that remotely access files using FTAM to know that a given file contains an ODA data stream.

B.1.3 Conveyance of ODA over DTAM

This recommendation provides for information concerning the interchange of ODA based documents with DTAM protocols.

DTAM (Document Transfer and Manipulation) is defined in the T.430-Series of recommendations and is, like ODA, an integral part of the T.400-Series of CCITT Recommendations named Open Document Architecture, Transfer and Manipulation.

The T.520-Series of recommendations contain Communication Application Profiles (CAP). Recommendation T.522 describes the Communication Application Profile BT1 for document bulk transfer. Recommendation T.522 is applicable for the Office Document Format Profile (FOD) published in this ISP.

NOTE 34 - The use of BT1 within the end-to-end oriented Telematic Services Telefax 4 and Teletex is described in Recommendation T.561, subclause 7.1 and Recommendation T.562, subclause 7.1.

B.1.4 Conveyance of ODA over flexible disks

The recommended method for interchanging ODA documents between systems by the exchange of magnetically recorded Flexible Disk Cartridges is by the use of an annex to [CCITT Recommendation T.411 | ISO 8613-1] (to be published), *"Recording of Documents Conforming to ISO 8613 on Flexible Cartridges Conforming to ISO 9293"*. This annex provides for recording each ODA document as a separate file as defined by ISO 9293, *"Volume and File Structure of Flexible Disk Cartridges for Information Interchange"*.

NOTE 35 - Document encoded in ODL may be stored such that each SGML ENTITY is recorded in a separate file or in the case of an SDIF encoding, the file may be stored in a single file.

B.2 Font reference

The recommended method for specifying a font reference is to be based on ISO 9541.

Font sizes from 6 to 72 points (100 to 1200 BMU) are intended to be supported by implementation conforming to this recommendation. All other values of font sizes may additionally be supported, but implementations may also support using some form of "fallback".

The minimum font properties and values from ISO 9541 that are to be specified in a Font-Attribute-Set are those specified by the following document application profile notation.

```
Font-Attribute-Set      {  
  
    PERM Font-Name          {ANY_VALUE},  
    PERM Standard-Version   {ANY_VALUE},  
    PERM Data-source        {ANY_VALUE},  
    PERM Design-source      {ANY_VALUE},  
    PERM Font-Family-Name   {ANY_VALUE},  
    PERM Posture             {'upright' | 'italic-forward'},  
    PERM Weight              {'light' | 'medium' | 'bold'},  
    PERM Proportionate-Width {ANY_VALUE},  
    PERM Glyph-Complement   {  
        PERM #Included-Glyph-Collections
```

```

                                {ANY_VALUE},
    PERM #Excluded-Glyph-Collections
                                {ANY_VALUE},
    PERM #Included-Glyphs        {ANY_VALUE},
    PERM #Excluded-Glyphs        {ANY_VALUE}
                                },
    PERM Design-Size              {ANY_VALUE},
    PERM Min-Size                 {
    PERM #Numerator                {100 .. 1200},
    PERM #Denominator              {1}
                                },
    PERM Max-Size                 {
    PERM #Numerator                {100 .. 1200},
    PERM #Denominator              {1}
                                },
    -- BMU Units equivalent to range of 6..72 point sizes --
    PERM Design-Group             {
    PERM #Class                     {ANY_VALUE},
    PERM #Subclass                 {ANY_VALUE},
    PERM #Specific-Group           {ANY_VALUE}
                                },
    PERM Structure                 {ANY_VALUE},
    PERM Writing-Modes             {
    MUL                             {
        REQ #Writing-Mode-Name
                                {ANY_VALUE},
        PERM #Nominal-Escapement-Direction
                                {ANY_VALUE},
        PERM #Escapement-Class
                                {ANY_VALUE},
        PERM #Average-Escapement-X
                                {ANY_VALUE},
        PERM #Average-Escapement-Y
                                {ANY_VALUE}
                                }
        }
    }
}

```

B.3 ISO 8632 (CGM) constraints for this DAP

It is recommended that geometric graphics content information contain only those elements listed in this portion of the document, in addition to the constraints imposed by [CCITT Recommendation T.418 | ISO 8613-8]. It is believed that this subset of the CGM is sufficiently implemented to enable interworking of geometric graphics for application conforming to this document application profile.

Where an element has parameters, recommended constraints on the values are given. The "---" symbol indicates that there is no recommended constraint.

Requirements in ISO 8632 and [CCITT Recommendation T.418 | ISO 8613-8] concerning mandatory elements, parameters shall be fulfilled.

3.3.1 Delimiter elements

No-Op	An arbitrary sequence of n octets. Where $n=0, 1, \dots, 32767$. The sequence of zero or more octets is for padding purposes.
Begin Metafile	Support will be provided for strings with a length up to 254 octets, except for data records which will support strings with a length up to 32767 octets.
End Metafile	
Begin Picture	Support will be provided for strings with a length up to 254 octets, except for data records which will support strings with a length up to 32767 octets.
Begin Picture Body	
End Picture	

B.3.2 Metafile descriptor elements

Metafile Version	1
Metafile Description	Support will be provided for strings with a length up to 254 octets, except for data records which will support strings with a length up to 32767 octets. The METAFILE DESCRIPTION string parameter will be used to include the sub-string "ISO FOD36" to label the content information as conforming to this profile. In addition, generators of content are encouraged to append a sub-string that identifies the company and product that produced the CGM.
VDC Type	--
Integer Precision	16, 32
Real Precision	(0,9,23), (1,16,16)
Index Precision	16
Colour Precision	8, 16
Colour Index Precision	8, 16
Maximum Colour Index	0..63
Colour Value Extent	--
Metafile Element List	--
Metafile Defaults Replacement	--
Font List	All fonts referenced in the metafile shall be defined. Font referencing in FONT LISTS using ISO 9541 names is preferred, but font names may be specified using proprietary font names.
Character Set List	All character sets referenced in the metafile shall be defined in CHARACTER SET LIST. Permissible character sets are the same as for character content architecture.
Character Coding Announcer	--

B.3.3 Picture descriptor elements

Scaling Mode	The Scale Factor parameter of SCALING MODE element is always a 32-bit floating point value, even when the REAL PRECISION has selected fixed point for other real numbers. It is not apparent in ISO 8632 what the precision of this floating point value is when fixed point has been selected. Its precision shall be (0,9,23).
Colour Selection Mode	--
Line Width Specification Mode	--

Marker Size Specification Mode	--
Edge Width Specification Mode	--
VDC Extent	--
Background Colour	--

B.3.4 Control elements

VDC Integer Precision	16, 32
VDC Real Precision	(0,9,23), (1,16,16)
Auxiliary Colour	--
Transparency	Transparent
Clip Rectangle	--
Clip Indicator	--

B.3.5 Graphical primitive elements

Polyline	The minimum support for the length of point lists is 1024 points.
Disjoint Polyline	The minimum support for the length of point lists is 1024 points.
Polymarker	The minimum support for the length of point lists is 1024 points.
Text	Support will be provided for strings with a length up to 254 octets, except for data records which will support strings with a length up to 32767 octets. Format effector control characters are disallowed in the string parameter.
Restricted Text	Support will be provided for strings with a length up to 254 octets, except for data records which will support strings with a length up to 32767 octets. Format effector control characters are disallowed in the string parameter.
Append Text	Support will be provided for strings with a length up to 254 octets, except for data records which will support strings with a length up to 32767 octets. Format effector control characters are disallowed in the string parameter.
Polygon	The minimum support for the length of point lists is 1024 points.
Polygon Set	The minimum support for the length of point lists is 1024 points.
Cell Array	The minimum support for the length of colour lists parameter for the CELL ARRAY element is 1048576. This supports a 1024 x 1024 image.
Rectangle	--
Circle	--
Circular Arc 3 Point	--
Circular Arc 3 Point Close	--
Circular Arc Centre	--
Circular Arc Centre Close	--
Ellipse	--
Elliptical Arc	--
Elliptical Arc Close	--

B.3.6 Attribute elements

Line Bundle Index	--
Line Type	Negative values are prohibited.
Line Width	--

Line Colour	--
Marker Bundle Index	--
Marker Type	Negative values are prohibited.
Marker Size	--
Marker Colour	--
Text Bundle Index	--
Text Font Index	All fonts referenced (indexed by TEXT FONT INDEX) in the metafile shall be defined in FONT LIST either in presentation parameters of [CCITT Recommendation T.410 series ISO 8613] or in ISO 8632.
Text Precision	--
Character Expansion Factor	--
Character Spacing	--
Text Colour	--
Character Height	--
Character Orientation	--
Text Path	--
Text Alignment	--
Character Set Index	All character sets referenced in the metafile (indexed by CHARACTER SET INDEX) shall be defined in CHARACTER SET LIST. The only character sets which may be designated in G0 are ISO 646 IRV or versions of ISO 646. Other character sets shall be designated in G1, G2 or G3.
Alternate Character Set Index	All character sets referenced in the metafile (indexed by ALTERNATE CHARACTER SET INDEX) shall be defined in CHARACTER SET LIST.
Fill Bundle Index	--
Interior Style	--
Fill Colour	--
Hatch Index	Negative values are prohibited.
Pattern Index	1 .. 8
Edge Bundle Index	--
Edge Type	Negative values are prohibited.
Edge Width	--
Edge Colour	--
Edge Visibility	--
Fill Reference Point	--
Pattern Table	The PATTERN TABLE element has an unspecified effect when it appears in a picture subsequent to any graphical primitives. The PATTERN TABLE element shall appear prior to any graphical primitive element to assure that interpreting systems without dynamic pattern update can render the intended effect. The minimum support for the length of the Colour Array parameter for the PATTERN TABLE element is 2048. This will support 8 patterns of 16 x 16, 2 patterns of 32 x 32 or 1 pattern of 32 x 64. All indexes which are used in the metafile shall be defined.
Pattern Size	--
Colour Table Specification	The COLOUR TABLE element has an unspecified effect when it appears in a picture subsequent to any graphical primitives. The COLOUR TABLE element shall appear prior to any graphical primitive elements to assure that interpreting systems without dynamic colour update can render the intended effect. The minimum support for the length of the Colour List parameter in the COLOUR TABLE element is 63. This will support a 64 (0..63)

entry colour table. All indexes which are used in the metafile shall be defined.

--

Aspect Source Flags

B.3.7 External elements

Message

The presentation of message string may not be appropriate for all applications. No requirement for formatted presentation of the message string has been placed on the Interpreter. Only the No Action flag needs to be supported. Support for string lengths up to 254.

Application Data

Support will be provided for strings with a length up to 254 octets, except for data records which will support strings with a length up to 32767 octets.

B.4 Interoperability with SGML applications

NOTE 36 - Annex B.4 applies to the ISP only.

The recommended method for the exchange of documents between Standard Generalized Markup Language (ISO 8879, SGML) based systems and systems based on this ODA document application profile is by means of exchanging a document representation conforming to these agreements in an encoded form of the SGML language known as the Office Document Language (ODL). ODL is a standardized SGML application for representing documents conforming to the ODA base standard. Such a representation may be converted into the Office Document Interchange Format (ODIF) supported by this document application profile.

Annex C

Bibliography

(This annex does not form an integral part of this specification.)

CCITT Recommendation T.50 : 1992 (to be published), *International Reference Alphabet*;

CCITT Recommendation T.51 : 1992 (to be published), *Latin based coded character sets for telematic services*;

CCITT Recommendation T.52 : 1992 (to be published), *Non-Latin coded character sets for telematic services*;

ISO 8571 : 1988, *Information processing systems - Open Systems Interconnection - File transfer, access and management*;

ISO 9070 : 1991, *Information technology - SGML support facilities - Registration procedures for public text owner identifiers*;

ISO/TR 9573 : 1988, *Information processing - SGML technical report - Techniques for using SGML*;

ISO 10021 : 1990, *Information processing systems - Text communication - Message Oriented Text Interchange System*.

Stable Implementation Agreements for Open Systems Interconnection Protocols: Part 17 - Open Document Architecture Level 2 DAP

**Output from the December 1992 Open Systems
Environment Implementors' Workshop (OIW)**

**SIG Chair: Jim Wing, IBM
SIG Editor: Jim Wing, IBM**

**PART 17 - Office Document Architecture
Level 2 DAP**

December 1992 (Stable)

Foreword

This part of the Stable Implementation Agreements was prepared by the Office Document Architecture Special Interest Group (ODASIG) of the Open Systems Environment implementors' Workshop (OIW).

Text in this part has been approved by the Plenary of the above-mentioned Workshop.

Future changes and additions to this version of these implementor Agreements will be published as change pages. Deleted and replaced text will be shown as struckout. New and replacement text will be shown as shaded.

Table of Contents

Part 17 - Office Document Architecture Level 2 DAP 1



Part 17 - Office Document Architecture Level 2 DAP

NOTE - Text for the International Standardized Profile 11181-1 (FOD26) follows this page.

This Agreement provides a Document Application Profile which has been developed through the joint efforts of ODA expert groups within the:

OSE Implementors' Workshop (OIW);

Asia-Oceania Workshop (AOW);

European Workshop for Open Systems (EWOS);

CCITT Study Group VIII.

This effort was conducted through meetings of the Profile Alignment Group for ODA (PAGODA) for the purpose of facilitating the interworking of applications which interchange documents based on [ISO 8613 | T.410 series of CCITT Recommendations]. This Agreement specifies one of the profiles resulting from that work:

ISO/IEC ISP 11181-1 : 1992, Information technology - Standardized Profile FOD36 - Office Document Format : Enhanced document structure - Character, raster graphics and geometric graphics content architectures - Document Application Profile.

This Agreement accepts the entirety of the definitions and provisions of this ISP as the agreement for the OIW Stable Agreements and the standards upon which the specified ISP is based as referenced by the text of the ISP. For this reason the text of the ISP is not reproduced here, but referenced to avoid any doubt as to the official text being agreed upon.



International Standardized Profile (ISP)

FOD26:

**Information technology -
International Standardized Profile FOD26 -
Office Document Format: Enhanced Document Structure -
Character, Raster Graphics and Geometric Graphics
content architectures**

**Part 1:
Document Application Profile**

July 1992

Contents

Foreword.....iv

1. Scope1

2. Normative references..... 2

3. Definitions 4

4. Relationship with other profiles 4

5. Conformance 5

 5.1. Data stream conformance 5

 5.2. Implementation conformance 6

6. Characteristics supported by this document application profile.... 6

 6.1. Overview 6

 6.2. Logical characteristics 9

 6.3. Layout characteristics 17

 6.4. Document layout characteristics 36

 6.5. Content layout and imaging characteristics 44

 6.6. Miscellaneous features 59

 6.7. Document management features 66

7. Specification of constituent constraints 72

 7.1. Introduction 72

 7.2. Document profile constituent constraints 77

 7.3. Logical constituent constraints 86

 7.4. Layout constituent constraints 100

 7.5. Layout style constituent constraints 120

 7.6. Presentation style constituent constraints 124

 7.7. Content portion constituent constraints 126

8. Interchange format 128

 8.1. Interchange format class A 128

 8.2. Interchange format SDIF 129

Annexes

A Amendments and corrigenda 131

B Recommendations 132

C Bibliography 140

List of Figures

Figure 1	Page layout type A	23
Figure 2	Page layout type B	23
Figure 3	Page layout type C	24
Figure 4	Page layout type D	24
Figure 5	Example of body area layout for page layout A	26
Figure 6	Example of body area layout for page layout B	27
Figure 7	Example of body area layout for page layout C and D	28
Figure 8	Examples of the layout of snaking columns	30
Figure 9	Example of synchronized column layout for page layout A .	31
Figure 10	Specification of margins	42
Figure 11	Code extension features (basic case)	47
Figure 12	Code extension features (all possible cases)	48
Figure 13	The 'body' part of the generic logical structure	73
	- the passage and numbered segment levels	
Figure 14	The 'body' part of the generic logical structure	73
	- the paragraph level	
Figure 15	The 'common' part of the generic logical structure	74
Figure 16	The layout structure - document root and page sets	74
Figure 17	The layout structure - the page structure	75
Figure 18	The layout structure - the header and footer	75
	frame structure	

List of Tables

Table 1 Nominal page sizes 21

Table 2 List of number string identifiers 61

Foreword

Development of this document application profile has been done in liaison with several organizations. These include ODA expert groups within the:

- Asia-Oceania Workshop (AOW);
- CCITT Study Group VIII;
- European Workshop for Open Systems (EWOS);
- NIST OSI Implementors Workshop (OIW).

The liaison between these organizations has occurred within the meetings of the Profile Alignment Group for ODA (PAGODA). These meetings have focused on the development of a single set of Internationally aligned ODA document application profiles.

The profile defined in this ISP is a part of the ODA profile taxonomy defined in TR 10000-2, 4.4.4.3 and 5.4.1. This profile is specific to the profile identifier FOD26.

At present, this ISP consists of one part:

- part 1, Document application profile.

Further parts may be added to this ISP.

This part contains three annexes:

- annex A (normative): Amendments and corrigenda
- annex B (informative): Recommendations;
- annex C (informative): Bibliography.

Introduction

The purpose of this International Standardized Profile (ISP) is to facilitate the interworking of applications interchanging documents based on ISO 8613, ODA. This ISP is suitable for interchanging documents in formatted form, processable form or formatted processable form and has been defined in accordance with [ISO 8613-1|CCITT Recommendation T.411]. The format of this profile is in accordance with the standardized proforma and notation defined in Addendum 1-Annex F of [ISO 8613-1|CCITT Recommendation T.411].

International Standardized Profile (ISP)

FOD26:

Information technology -
International Standardized Profile FOD26 -
Office Document Format: Enhanced Document Structure -
Character, Raster Graphics and Geometric Graphics
content architectures

Part 1:
Document Application Profile

1 Scope

This profile specifies interchange formats for the transfer of structured documents between equipment designed for word or document processing. Such documents may contain character, raster graphics and geometric graphics content.

The documents that can be interchanged using this profile range from simple documents to highly structured technical reports, articles and typeset documents such as brochures. This profile provides a comprehensive level of features for the transfer of documents between these systems.

This profile allows documents to be interchanged in the following forms:

- a) formatted form;
- b) processable form;
- c) formatted processable form.

The architecture levels defined for these three forms have matching functionalities so that the interchange formats of a document are convertible from a processable form to any other form.

This profile is independent of the processes carried out in an end system to create, edit or reproduce documents. It is also independent of the means to transfer documents which, for example, may be by means of communication links or storage media.

A document structured in accordance with this ISP is represented for interchange by one of two DAPs. One DAP uses the Office Document Interchange Format (ODIF), the other DAP uses the Office Document Language (ODL), both as defined in ISO 8613-5.

When this document refers to *this profile*, it is referring to either of the document application profiles defined by this specification.

2 Normative References

The following documents contain provisions which, through reference in this text constitute provisions of this Specification. At the time of publication the editions indicated were valid. All documents are subject to revision and parties to agreements based on this Specification are warned against automatically applying any more recent editions of the documents listed below, since the nature of references made by profiles to such documents is that they may be specific to a particular edition. Members of IEC and ISO maintain registers of currently valid International Standards and ISPs. The CCITT secretariat maintains a list of currently valid CCITT recommendations.

ISO 8613-1 : 1989, Information processing - Text and office systems; Office Document Architecture (ODA) and interchange format - Part 1: Introduction and general principles;

ISO 8613-2 : 1989, Information processing - Text and office systems; Office Document Architecture (ODA) and interchange format - Part 2: Document structures;

ISO 8613-4 : 1989, Information processing - Text and office systems; Office Document Architecture (ODA) and interchange format - Part 4: Document profile;

ISO 8613-5 : 1989, Information processing - Text and office systems; Office Document Architecture (ODA) and interchange format - Part 5: Office document interchange format;

ISO 8613-6 : 1989, Information processing - Text and office systems; Office Document Architecture (ODA) and interchange format - Part 6: Character content architectures;

ISO 8613-7 : 1989, Information processing - Text and office systems; Office Document Architecture (ODA) and interchange format - Part 7: Raster graphics content architectures;

ISO 8613-8 : 1989, Information processing - Text and office systems; Office Document Architecture (ODA) and interchange format - Part 8: Geometric graphics content architectures;

ISO 8613-1 : Addendum 1: Information processing - Text and office systems; Office Document Architecture (ODA) and interchange format Part 1: Introduction and general principles, Addendum 1 - Annex F: A Document Application Profile proforma and notation;

CCITT Recommendation T.4 - Standardization of group 3 facsimile apparatus for document transmission (1988);

CCITT Recommendation T.6 - Facsimile coding schemes and coding control functions for group 4 facsimile apparatus (1988);

ISO/IEC 646 : 1991, Information technology - ISO 7-bit coded character set for information interchange;

ISO 8859-1 : 1987, Information processing - 8-bit Single-byte coded graphic character sets - Part 1: Latin alphabet No. 1;

ISO 6937-2 : 1983, Information processing - Coded character sets for text communication - Part 2: Latin alphabetic and non-alphabetic graphic characters;

ISO 6937-2 Addendum 1 : 1989, Information processing - Coded character sets for text communication - Part 2: Latin alphabetic and non-alphabetic graphic characters, Addendum 1;

ISO 2022 : 1986, Information processing - ISO 7-bit and 8-bit coded character sets - Code extension techniques;

ISO 2375 : 1985, Data processing - Procedure for registration of escape sequences;

ISO 7350 : 1984, Text communication - Registration of graphic character subrepertoires;

ISO 8824 : 1987, Information processing systems - Open Systems Interconnection - Abstract Syntax Notation One (ASN.1);

ISO 8825 : 1987, Information processing systems - Open Systems Interconnection - Basic encoding rules for abstract syntax notation one (ASN.1);

ISO 8879 : 1986, Information processing - Text and office systems - Standard Generalized Markup Language (SGML);

ISO 9069 : 1988, Information processing - SGML support facilities - SGML Document Interchange Format (SDIF);

ISP 10610-1 : 1992, Information technology - International Standardized Profile FOD11 - Office Document Format: Simple document structure - Character content architecture only - Part 1: Document Application Profile.

ISP 11182-1 : 1992, Information technology - International Standardized Profile FOD36 - Office Document Format: Extended document structure - Character, Raster Graphics and Geometric Graphics content architectures - Part 1: Document Application Profile.

ISO/IEC TR 10000-1: 1990, Information Technology - Framework and taxonomy of International Standardized Profiles - Part 1: Framework

ISO/IEC TR 10000-2: 1990, Information Technology - Framework and taxonomy of International Standardized Profiles - Part 2 Taxonomy of Profiles

3 Definitions

For the purposes of this Specification, the following definitions apply.

The definitions given in [ISO 8613-1|CCITT Recommendation T.411] are applicable to this profile.

Constituent names

Each constituent that may be included in a document that conforms to this profile has been given a unique name which serves to identify that constituent throughout this profile.

The convention is that full names are used (i.e., no abbreviations are used), two or more words in a name are concatenated and each word begins with a capital. Examples of constituent names used in this profile are BodyText, Footnote, RectoPage and ColumnFixed.

In clause 6 each constituent provided by this profile is underlined once at the point in the text at which the purpose of that constituent is defined. This also serves to identify all the constituents provided by this profile.

The same constituent names are also used in the technical specification in clause 7 so that there is a one-to-one correspondence between the use of these names in clause 6 and 7.

Although the constituent names relate to the purpose of the constituents, the semantics of constituents shall not be implied from the actual names that are used. Also, these names do not appear in an interchanged document but a mechanism for identifying constituents in an interchange document is provided (see 6.6.4). Thus in an application using this profile, the constituents may be known to the user by different names.

4 Relationship with other profiles

This profile belongs to a series of hierarchically related profiles which include FOD11 and FOD36.

The features supported by this profile are a superset of the features supported by the profile FOD11 and thus all data streams that are conformant to FOD11 are also conformant to this profile, apart from the DAP identifier.

Also the features supported by this profile are a subset of the features supported by the profile FOD36 and thus all data streams conformant to this profile are also conformant to FOD36, apart from the DAP identifier.

The specification of the DAPs in this ISP is identical to the specification defined in the CCITT Recommendation T.505 (PM-26), except that PM-26 only defines the use of the ODIF interchange format.

5 Conformance

In order to conform to this profile, a data stream representing a document shall meet the requirements specified in 5.1.

This part of the ISP does not define implementation support requirements. These requirements are defined in other parts that make use of this ISP.

5.1 Data stream conformance

The following requirements apply to the encoding of data streams which conform to this profile:

- a) The data stream shall be encoded either in accordance with the ASN.1 encoding rules defined in ISO 8825 or the SGML encoding rules defined in ISO 8879;
- b) The data stream shall be structured in accordance with the interchange formats defined in clause 8;
- c) The document, as represented by the data stream after resolution of any external references, shall be structured in accordance with one of the documents architecture classes as defined in 6.1 and shall contain all mandatory constituents specified for that class; other constituents may be included, provided that they are permitted for that class, as specified in clause 7;
- d) Each constituent shall contain all those attributes specified as required for that constituent in this profile; other attributes may be specified provided that they are permitted for that constituent;
- e) The attribute values specified shall be within the range of permissible values specified in this profile;
- f) The encoded document shall be constructed in accordance with the abstract document architecture defined in [ISO 8613-2|CCITT Recommendation T.412];
- g) The document shall be structured in accordance with the characteristics and constraints specified in clause 6.

5.2 Implementation conformance

This subclause states the requirements for implementations claiming conformance to this profile.

A conforming receiving implementation shall be capable of receiving either any data streams conforming to this profile structured in accordance with ODIF or any data streams conforming to this profile structured in accordance with ODL or both of these. Receiving usually, but not always, involves recognizing and further processing the data stream elements. The explicit meaning of "receiving" is determined by another part of this ISP.

A receiving system which claims conformance to this DAP shall be capable of handling data streams which are conformant to DAPs that are subordinate to this DAP within the taxonomy described in clause 4, (ie. FOD11).

6 Characteristics supported by this document application profile

This clause describes the characteristics of documents which may be represented by data streams conforming to this profile. This clause also describes how these characteristics are represented in terms of constituent constraints.

6.1 Overview

6.1.1 General

This profile supports the interchange of documents in the following forms:

- processable form, which facilitates the revision of a document by a recipient;
- formatted form, which facilitates the reproduction of a document as intended by the originator;
- formatted processable form, which facilitates the reproduction of a document as intended by the originator or facilitates the revision of a document by a recipient.

In addition this profile supports the interchange of:

- generic-documents;
- document profile.

The constituents that make up these five classes of data stream are defined in 6.1.2 to 6.1.6. Constituents defined as 'required' shall occur in any data stream that conforms to this profile. Constituents listed as 'optional' may or may not be present in the data stream depending on the requirements of the particular data stream.

Note that the constituents that make up a complete document that is conformant to this profile include all those referenced and contained in, if any, resource and external documents (see 6.6.1 and 6.6.2).

6.1.2 Formatted form documents

Required constituents:

- a document profile;
- layout object descriptions representing a specific layout structure.

Optional constituents:

- layout object class descriptions representing a 'factor' generic layout structure;
- presentation styles;
- content portion descriptions.

6.1.3 Processable form documents

Required constituents:

- a document profile;
- logical object class descriptions representing a 'complete' or 'partial' generic logical structure;
- logical object descriptions representing a specific logical structure.

Optional constituents:

- layout object class descriptions representing a 'complete' generic layout structure;
- layout styles;
- presentation styles;
- content portion descriptions.

In the case of processable form documents, when the generic layout structure is not present, additional restrictions are placed on the layout directives that may be included in layout styles. These restrictions are defined in 6.4.3.

6.1.4 Formatted processable form documents

Required constituents:

- a document profile;
- logical object class descriptions representing a 'complete' or 'partial' generic logical structure;
- logical object descriptions representing a specific logical structure;
- layout object class descriptions representing a 'complete' generic layout structure;
- layout object descriptions representing a specific layout structure.

Optional constituents:

- layout styles;
- presentation styles;
- content portion descriptions.

6.1.5 Generic-documents

A generic-document consists of one of the following sets of constituents:

a)

- a document profile;
- logical object class descriptions which represent a 'complete' or 'partial' generic logical structure;
- layout styles whose presence is optional;
- presentation styles whose presence is optional;
- generic content portions whose presence is optional.

b)

- a document profile;
- layout object class descriptions which represent a 'complete' generic layout structure or 'factor' set;
- presentation styles whose presence is optional;
- generic content portions whose presence is optional.

c)

- a document profile;
- logical object class descriptions which represent a 'complete' or 'partial' generic logical structure;
- layout object class descriptions which represent a 'complete' generic layout structure;
- layout styles whose presence is optional;
- presentation styles whose presence is optional;
- generic content portions whose presence is optional.

6.1.6 Document profile

This form of document contains a document profile only.

6.2 Logical characteristics

6.2.1 Introduction

This subclause defines the logical constituent constraints provided by this profile to represent the characteristics of documents containing logical component descriptions.

Different constituent constraints may be used to represent and distinguish parts of a document that have different logical characteristics. This subclause describes the general characteristics and typical uses of the constituent constraints that are provided.

The descriptions of the logical characteristics represented by each of the constituent constraints is provided for guidance only. It is the responsibility of the user to determine how a document is to be represented using the constituents provided. Adherence to these guidelines can enhance the mutual understanding of a document by an originator and a recipient.

6.2.2 Overview of the logical structure

From the logical point of view, the document consists of two parts, namely a 'body' part and a 'common' part.

The 'body' part represents the main content of a document and is intended to be reproduced in the body area of the pages that make up the document.

The 'common' part represents common content that is to be placed in reserved header and footer areas on each page of a document. Header and footer content are independently optional and so may be included in an interchanged document only if required.

6.2.3 Body part of the logical structure

6.2.3.1 DocumentLogicalRoot

DocumentLogicalRoot is a constituent constraint representing the top level in the document logical structure. Its immediate subordinates consist of a sequence of one or more constituent constraints of the type Passage.

The automatic numbering schemes that apply to constituent constraints of the types NumberedSegment and Footnote may be initialised on the DocumentLogicalRoot.

6.2.3.2 Passage

Passage is a constituent constraint that represents the first level of logical subdivision of a document. It may be used to indicate a logical grouping of subordinate parts of a document that are to be regarded as an entity for reading or that have common layout and presentation characteristics. For example:

Passages are typically used to represent:

- the contents to be placed on the title page of a report;
- the front matter in the table of contents or foreword;
- the main matter of the document;
- the back matter, consisting of appendices, glossary or index.

The automatic numbering schemes that apply to subordinate constituent constraints of the types NumberedSegment and Footnote may be initialised on a Passage.

The immediate subordinates of a Passage consist of an optional arbitrary ordered sequence of one or more of the following constituent constraint types:

- Paragraph;
- BodyGeometric;
- BodyRaster;
- BodyText.

These may be optionally followed by one or more constituent constraints of the type NumberedSegment.

A Passage shall have at least one of the above constituent constraint types as a subordinate.

A document may contain several different class definitions of the type Passage, each of which defines the common characteristics of sets of Passages within the document such as their allowed subordinates or layout properties. For example, a class of Passages may be defined which always begin on a new page set.

6.2.3.3 NumberedSegment

NumberedSegment is a constituent constraint that represents a logical subdivision within a document that is distinguished by an identifier. The logical subdivision may be a subdivision of a Passage or another higher level NumberedSegment. The logical subdivision may also have some common layout characteristics.

The automatic numbering schemes that apply to subordinate constituent constraints of the types NumberedSegment and Footnote may be initialised on a Passage.

The immediate subordinates of a NumberedSegment consist of the constituent constraint Number, whose presence is mandatory and serves to carry the identifier of the NumberedSegment. This is followed by an optional arbitrary ordered sequence of one or more of the following constituent constraints:

- Paragraph;
- BodyGeometric;
- BodyRaster;
- BodyText.

These are optionally followed by a sequence of one or more constituent constraints of the type NumberedSegment. Hence a document may contain any number of nested levels of the constituent NumberedSegment.

A NumberedSegment is typically used to represent entities such as chapters, sections, nested sub-sections and appendices which contain an identifier that serves to distinguish that entity for human comprehension.

A document may contain any number of different class definitions of NumberedSegment which define the common characteristics of sets of NumberedSegments, such as their allowed subordinates and layout properties.

Class definitions of NumberedSegments may be recursively defined. In this case only one call of NumberedSegment may be specified, and the <simple-expr> construction in the macro USENUMBERSTRINGS in the bindings attribute of this class shall use the optional ORD construction only. If the recursive class definitions are used for NumberedSegment, the following constraints will also apply. For all levels which reference recursively defined classes:

- numbering format will be the same;
- no initial value other than 1 or re-initialisation of the numbering is possible;
- it is not possible to continue the numbering across Passages.

If class definitions are not recursive in this way, there shall be one and only one class definition for NumberedSegments corresponding to each level of numbering within each Passage. Class definitions may be shared between NumberedSegments belonging to different Passages but they shall then be used at the same level.

6.2.3.4 Number

Number is a constituent constraint that represents the identifier of a NumberedSegment to which it is subordinate. This identifier allows the NumberedSegment to be distinguished within the document for machine processing or human comprehension.

A Number is a basic logical constituent which contains a content generator which, when evaluated, produces the identifier referred to above. This evaluation takes place during the layout process.

The identifiers are structured and consist of a sequence of one or more numerals that allow NumberedSegments at the same or different levels in a document structure to be uniquely distinguished. The numerals may be represented by Arabic or Roman numerals or by their alphabetic equivalent in lower or upper case characters (the number 1 is represented by 'A' etc.). Each numeral in an identifier is distinguished by means of 'separator' characters such as spaces and full stops (the character "period"); a typical example is '6.2.3.4'.

NOTE: The separator may if required be an empty string.

Further details of the structure and generation of the identifiers are given in 6.6.7.

6.2.3.5 Paragraph

Paragraph is a constituent constraint that is a subdivision of a Passage or NumberedSegment. It is typically used to represent the grouping of parts of a document that deals with a single theme or topic. These parts may consist of character, raster graphics and geometric graphics content.

The immediate subordinates of a Paragraph consist of an arbitrary ordered sequence of one or more of the following constituent constraints:

- BodyText;
- BodyRaster;
- BodyGeometric;
- Footnote.

Content from any subordinate basic text objects within a paragraph may be run on one from another (that is, to continue on the same line) by use of the Concatenation feature (see 6.4.2.5). Alternatively, content from subordinates of a paragraph may be separated one from another to give white space between them, using Separation (see 6.4.2.2). This may be used to give an effect similar to that achieved with empty lines of text. Use of empty text lines to achieve white space between areas of text or other content may lead to unintended blank areas adjacent to the leading edge of layout objects (eg at page breaks) whereas the use of Separation avoids this.

Constituents of the type BodyText may be 'concatenated' to form a continuous stream of character content which is laid out as a single unit. Sequences of constituents of the types BodyText and Footnote may be concatenated to represent a stream of character content with embedded footnotes. Multiple embedded footnotes, which may be consecutive without intervening text, may be included in the content.

Another typical use of a Paragraph is to represent a group of document parts that have common layout characteristics. An example is a graphical illustration with associated text which is to be laid out in a particular frame.

6.2.3.6 BodyText, BodyRaster and BodyGeometric

BodyText, BodyRaster and BodyGeometric are constituent constraints which represent the lowest level of logical subdivision of a document. These constituent constraints are subdivisions of Passages, NumberedSegments and Paragraphs. They allow the layout and presentation requirements of different parts of a document to be specified.

These are basic logical constituents that directly refer to content portions that contain character, raster graphics and geometric graphics content respectively. BodyText shall refer to one or more content portions each containing processable, formatted or formatted processable character content. BodyRaster and BodyGeometric shall only refer to a single content portion containing formatted processable raster graphics content or formatted processable geometric graphics content respectively.

Constituents of these types in the generic logical structure may refer to generic content. This provides the means of defining common content within the body part of a document.

6.2.3.7 Footnote

Footnote is a constituent constraint that is a subdivision of a Paragraph and is used to represent footnotes within a document.

A footnote is an amount of content that is logically associated with a particular part of the document body but which is intended to be read and laid out separately from its associated part of the document. Typically, a footnote consists of a footnote identifier, which is embedded within the document body, and the footnote itself, which is laid out elsewhere.

A Footnote is a composite logical constituent whose immediate subordinates consist of the constituent constraint FootnoteReference, which represents the footnote identifier, followed by the constituent constraint FootnoteBody, which represents the footnote itself. Both of these subordinates are mandatory.

6.2.3.8 FootnoteReference

FootnoteReference is a constituent constraint that is used to represent a footnote reference within the body of a document.

FootnoteReference is a basic logical constituent that contains a content generator which when evaluated produces a character string which constitutes the footnote reference referred to above.

The generated character string consists of a label with optional prefix and suffix character strings. The label is used to uniquely identify a particular footnote and may consist of a number which is represented in the form of Arabic or Roman numerals or by an alphabetic equivalent. The number may be automatically generated so that its value is incremented for each successive footnote. Alternatively, the label may consist of a user defined character string.

In a sequence of footnotes, automatic and user defined labels may be freely mixed (giving for example the sequence 1,2*,3,4). If the label consists of a user-defined character string, the automatically generated number sequence is not incremented.

An example of a footnote reference is '(2)' where '(' and ')' are user defined prefix and suffix strings respectively and '2' is the automatically generated label. Another example is 'note⁵' where '5' is the label and 'note' is a prefix string which also contains the control function PLU to enable the label to be represented in the form of a superscript. In this case, a suffix string containing the control function PLD would be required to cause the superscripting to be cancelled before the following text.

The format of the content generator referred to above is described in 6.6.8.

6.2.3.9 FootnoteBody

FootnoteBody is a constituent constraint which represents the content of a footnote.

FootnoteBody is a composite logical constituent whose subordinates consist of the constituent constraint FootnoteNumber, which is mandatory and represents the footnote identifier, followed by a sequence of one or more constituent constraints of the type FootnoteText which represents the footnote content. The identifier referred to above is identical to the corresponding footnote identifier which is embedded in the content of the document body and represented by the constituent constraint FootnoteReference.

The constituents subordinate to FootnoteBody are intended to be laid out separately from the other parts of the document content. When a generic layout structure is specified for the document, these constituents are constrained to be laid out in a FootnoteArea frame (see 6.3.5.9).

6.2.3.10 FootnoteNumber

FootnoteNumber is a constituent constraint that represents the footnote identifier within the footnote body.

This identifier is identical to the content associated with the constituent constraint FootnoteReference but is intended to be laid out so that it immediately precedes the content of the footnote body.

FootnoteNumber is a basic logical constituent that contains a content generator which when evaluated produces the identifier referenced above. The format of this content generator is the same as the content generator that may be specified for the constituent constraint FootnoteReference.

It is required to specify the layout category name 'Footnote' for this constituent. This along with a permitted categories attribute of 'Footnote' on the footnote frame will ensure that this constituent is laid out in a FootnoteArea frame when generic layout structure is specified within the document.

6.2.3.11 FootnoteText

FootnoteText is a constituent constraint that is used to represent the footnote content. It is the lowest logical subdivision of a FootnoteBody.

FootnoteText is a basic logical constituent that references one or more content portions each containing processable, formatted or formatted processable character content.

It is required to specify the layout category name 'Footnote' for this constituent. This along with a permitted categories attribute of 'Footnote' on the footnote frame will ensure that this constituent is laid out in a FootnoteArea frame when generic layout structure is specified within the document.

6.2.4 Common content part of the logical structure

6.2.4.1 CommonContent

CommonContent is a constituent constraint that represents common content that is to be laid out in the header and footer areas of the pages of a document. Common content consists of any combination of character, raster graphics and geometric graphics content.

Any number of constituent constraints of the type `CommonContent` may be contained in a document. `CommonContent` is a composite logical object class whose immediate subordinates consist of an arbitrary ordered sequence of one or more of the following constituent constraints:

- `CommonText`;
- `PageNumber`;
- `CommonRaster`;
- `CommonGeometric`.

When the generic layout structure is present, constituents of the type `CommonContent` and their associated subordinate constituents are constrained to be laid out in frames representing header or footer areas using the 'logical source' mechanism (see 6.3.6).

6.2.4.2 `CommonText`

`CommonText` is a constituent constraint that represents the common character content that is to be laid out in the header or footer area of a document. For example, header or footer content that appears on each page in a sequence of pages may be represented by this constituent.

`CommonText` is a basic logical object class that references one or more content portions each containing processable, formatted and formatted processable character content.

6.2.4.3 `PageNumber`

`PageNumber` is a constituent constraint that represents the common character content that is to be laid out in the header or footer area of a document. This constituent is specifically used when it is required to present a header or footer content which contains an automatically generated page number.

`PageNumber` is a basic logical object class that contains a content generator. This content generator contains a reference to a page number which is automatically evaluated when the document is laid out. This provides the means of representing the page numbers that are displayed on the consecutive pages of a document.

Each page number consists of a single number which may be represented in the form of Arabic or Roman numerals or in its alphabetic equivalent. Page numbering schemes may start at 0 or any value greater than 0.

The format of the content generators is defined in 6.6.6.

6.2.4.4 CommonRaster

CommonRaster is a constituent constraint that represents the common raster graphics content that is to be laid out in the header or footer area of a document. For example, this constraint may be used to represent a logo which is to be laid out on each page of a document.

Common raster is a basic logical object class which references a single content portion containing formatted processable raster graphics content.

6.2.4.5 CommonGeometric

CommonGeometric is a constituent constraint that represents the common geometric graphics content that is to be laid out in the header or footer area of a document. For example, this constraint may be used to represent a graphical icon which is to be laid out on each page of a document.

Commongeometric is a basic logical object class which references a single content portion containing formatted processable geometric graphics content.

6.3 Layout characteristics

This subclause defines the layout constituent constraints provided by this profile to represent the characteristics of documents.

Different constituent constraints may be used to represent and distinguish parts of a document that have different layout characteristics. This subclause describes the general characteristics and typical uses of the constituent constraints that are provided.

The descriptions of the layout characteristics represented by each of the constituent constraints is provided for guidance only. It is the responsibility of the user to determine how a document is to be represented using the constituents provided. Adherence to these guidelines can enhance the mutual understanding of a document by an originator and a recipient.

6.3.1 Overview of the layout characteristics

The document structure allows the document content to be laid out and presented in one or more page sets. Each page set may be used for different parts of the document, for example, the title page, foreword, table of contents, document body and appendices.

Each page set consists of a series of pages. In general, each page may be subdivided into three areas: the body area, which is used to layout the document body; and the header and footer areas, which may be used to layout the common content.

Four page layout types are supported by this profile. Each page layout type specifies how the body, header and footer areas are positioned within each page and how the content may be presented within each of those areas. These four types are referred to as page layouts A, B, C and D and are illustrated in figures 1, 2, 3 and 4 respectively.

It is intended that all applications which use this profile shall support page layout A, whereas support for the other three page layouts may be specified as optional.

Page layout A is used when the character content is to be laid out horizontally (from left to right or from right to left) and from top to bottom within the body area. This layout is typically used for documents written in Latin based, Hebrew and Arabic languages.

Page layout B is used when the character content is to be laid out vertically (bottom to top or top to bottom) and from left to right within the body area. This layout is typically used for documents written in Latin based, Hebrew and Arabic languages in which it is required to layout the content in landscape orientation within the body area of the page.

Page layouts C and D are used when the character content is to be laid out vertically and from right to left within the body area. These layouts are typically used in documents written in languages which use ideograms, such as Japanese and Chinese characters.

The body area may be further sub-divided into areas composed of single and multiple columns and an area may be reserved for footnotes. In addition, the header and footer areas may be sub-divided to allow the representation of different content types.

6.3.2 DocumentLayoutRoot

DocumentLayoutRoot is a constituent constraint that represents the top level in the document layout structure. Its immediate subordinates consist of a sequence of one or more constituents of the type PageSet. The numbering schemes for pages may be initialised on this constituent constraint.

6.3.3 PageSet

PageSet is a constituent constraint that represents a grouping of pages within a document. A PageSet is typically used to represent a part of a document that has different layout requirements from other parts of a document. Also, a PageSet may correspond to a part of a document that has a certain logical significance, for example, a PageSet might represent the front matter in a document or an individual chapter.

Only one level of PageSet is allowed in a document. However, a document may contain any number of class definitions of the type PageSet which may be used, for example, to provide a choice of alternative layouts for different parts of a document or to specify the exact layout requirements for each successive part of a document.

The immediate subordinates of a PageSet consist of a combination of constituent constraints of the types Page, RectoPage and VersoPage, as described in 6.3.4.1.

6.3.4 Page characteristics

6.3.4.1 Page constituents

Three constituent constraints are provided to represent the pages within a document, namely Page, RectoPage and VersoPage.

The only difference in the characteristics of these types of constituent constraints concerns the values that may be specified for the parameter "side of sheet" in the attribute "medium type". In the case of Page, the value of this parameter may be specified as 'recto', 'verso' or 'unspecified'. In the case of RectoPage, the value of this parameter may be specified as 'recto' or 'unspecified'; in the case of VersoPage, the value of this parameter may be specified as 'verso' or 'unspecified'. The values "recto" and "verso" of the "side of sheet" parameter of the "medium type" attribute are non-basic.

The pages that make up a page set consist of an optional initial page which is represented by the constituent constraint Page and which is optionally followed by either:

- a) A sequence of pages represented by the constituent constraint Page. All pages in this sequence shall have the same layout characteristics (see note) but these characteristics may differ from those of the initial page.
- b) A sequence of pages which are intended to be laid out alternatively on the 'recto' and 'verso' (or on the 'verso' and 'recto') sides of the presentation medium and are represented by the constituent constraints RectoPage and VersoPage respectively. All pages in this sequence shall have the same layout characteristics (see note) but these characteristics may differ from those of the initial page.

Pages having the same layout characteristics are pages that have the same page layout (see 6.3.4.5) and for which the body area, header area (if present) and footer area (if present) have the same dimensions and positions within the page (see 6.3.4.3). Pages having the same layout characteristics do not necessarily have the same position on the presentation medium (see 6.3.4.4).

A page set shall contain at least one page.

An initial page is typically used at the beginning of a document or of a section within a document. It may be used, for example, for a title page whose layout requirements differ from the following pages.

The following restrictions also apply to the pages within a page set:

- i) all the pages shall have the same dimensions and orientation (see 6.3.4.2);
- ii) all pages are to be laid out on the same size of presentation medium (see 6.3.4.3).

6.3.4.2 Page dimensions

The dimensions of the pages may be specified as any value (in BMUs) that is equivalent to or less than ISO A3 or ANSI B paper sizes in portrait or landscape orientation. The dimensions may be specified in portrait or landscape orientation. Japanese page sizes B4 and B5 are also supported but the dimensions of these pages lie within the range of dimensions given above.

Dimensions equivalent to or less than the common assured reproduction area of ISO A4 and ANSI-A in portrait or landscape orientation are basic values. Larger page sizes are non-basic and their use shall be indicated in the document profile.

Any default page dimensions may be specified in the document profile subject to the maximum dimensions defined above.

NOTE - The size termed "North American Letter (NAL)" in ISO 8613 (e.g. in ISO 8613-2, clause 7) is in this specification called "ANSI A" in order to be consistent with the other reference to ANSI standard paper sizes.

6.3.4.3 Nominal page sizes

The nominal page sizes that may be specified are listed in table 1. These may be specified in portrait or landscape orientation. All values of nominal page size are non-basic and hence all values used in a document shall be indicated in the document profile.

Any value of nominal page size defined in table 1, subject to the restrictions specified above, may be specified as the default value in the document profile.

Table 1 also includes the recommended assured reproduction area (ARA). Information loss may occur when a document is reproduced if the dimensions of constituent constraints of the type page exceed the ARA for the specified nominal page size.

Table 1 - Nominal page sizes

Page type	Size in inches or millimetres	Size in BMUs	ARA in BMUs
ISO A5	148mm x 210mm	7 015 x 9 920	not defined
ISO A4	210mm x 297mm	9 920 x 14 030	9 240 x 13 200
ISO A3	297mm x 420mm	14 030 x 19 840	13 200 x 18 480
ANSI legal	8.5in. x 14in.	10 200 x 16 800	9 240 x 15 480
ANSI A	8.5in. x 11in.	10 200 x 13 200	9 240 x 12 400
ANSI B	11in. x 17in.	13 200 x 20 400	12 744 x 19 656
Japan-legal	257mm x 364mm	12 141 x 17 196	11 200 x 15 300
Japan-letter	182mm x 257mm	8 598 x 12 141	7 600 x 10 200

6.3.4.4 Page offset

The page offset is the distance of the position of the left and top edges of the page relative to the left and top edges respectively of the presentation medium on which each page is reproduced. Any value of page offset may be specified provided that no part of the page area lies outside the area of the nominal page. Also, page offsets specified for the initial, recto and verso pages within a given page set may differ. The default page offset may be specified in the document profile.

6.3.4.5 Page layout characteristics

6.3.4.5.1 General characteristics

Each page in a document may be subdivided into three rectangular areas, as follows:

- a body area which is reserved for content that belongs to the body part of the document (see 6.3.5);
- a header area which is reserved for common header content (see 6.3.6);
- a footer area which is reserved for common footer content (see 6.3.6).

The body area is mandatory and shall occur on every page in a document. The header and footer areas are both optional.

Also, these three areas shall be entirely contained within the page area and shall not overlap.

Four types of page layout are supported as defined below.

6.3.4.5.2 Page layout A

For page layout A the header and footer areas are placed above and below the body area respectively. The layout paths in the header, body and footer areas are specified as 270 degrees. This type of layout is illustrated in figure 1.

6.3.4.5.3 Page layout B

For page layout B the header and footer areas are placed above and below the body area respectively. The layout path in the body area is specified as 0 degrees; in the header and footer areas the layout paths are specified as 270 degrees. This type of layout is illustrated in figure 2.

6.3.4.5.4 Page layout C

For page layout C the header and footer areas are placed above and below the body area respectively. The layout path in the body area is specified as 180 degrees; in the header and footer areas, the layout paths are specified as 270 degrees. This type of layout is illustrated in figure 3.

6.3.4.5.5 Page layout D

For page layout D the header and footer areas are placed to the right and left of the body area respectively. The layout paths in the header, body and footer areas are all specified as 180 degrees. This type of layout is illustrated in figure 4.

Figure 1 - Page layout type A

Figure 2 - Page layout type B

Figure 3 - Page layout type C

Figure 4 - Page layout type D

6.3.5 Body area characteristics

6.3.5.1 General characteristics

The body area is the area within a page where the main matter of the document, that is the 'body' part of the document, is laid out.

The body area may consist of a single frame into which the content is directly laid out. In this case, the body area is represented by a BasicBody frame.

Alternatively, the body area is subdivided into different rectangular regions to provide for combinations of single or multiple column layout and the layout of footnotes. In this case, the body frame is represented by a VariableCompositeBody frame.

6.3.5.2 BasicBody

BasicBody is a constituent constraint which defines a lowest level frame into which content is directly laid out.

The position and dimensions of this frame are fixed. The layout path specified depends upon the page layout type being used (see 6.3.4.5).

6.3.5.3 VariableCompositeBody

VariableCompositeBody is a constituent constraint that defines a composite frame which contains one or more subordinate variably positioned frames. A VariableCompositeBody frame has a fixed position and fixed dimensions. The layout path specified for this frame depends upon the page layout type being used (see 6.3.4.5).

The immediate subordinates of frames of this type consist of an arbitrary ordered sequence of one or more frames of the following constituent constraints:

- BasicFloat;
- SnakingColumns;
- SynchronizedColumns

It may also contain a single frame of the type FootnoteArea.

The subordinate frames are all variably positioned and have variable dimensions. Thus the relative positions of these frames in the body area may vary and depend upon the positions of other frames (if any) that are placed in the same VariableCompositeBody frame.

The layout path for VariableCompositeBody frames may be specified as 270, 0 or 180 degrees. This determines the page layout type used in the case where VariableCompositeBody represents the entire body area.

Frames of the type BasicFloat, SnakingColumns and SynchronizedColumns are laid out along the layout path specified (in 'normal' positioning fill order). FootnoteArea frames are laid out in the same direction as the body area layout path, but reverse fill order is used.

These frames are constrained to have the same path as the VariableCompositeBody frame to which they are subordinate. Except for frames of the type SnakingColumns.

Figures 5, 6 and 7 provide illustrations of the layout of frames within a VariableCompositeBody frame for the various page layout types.

A choice of subordinate frames of the types listed above may be specified for a VariableCompositeBody frame. Different frame types may be selected using various layout directives (see 6.4) and hence the layout characteristics of the body areas within a page set may change from page to page within a page set.

Figure 5 - Example of body area layout for
page layout A

**Figure 6 - Example of body area layout for
page layout B**

**Figure 7 - Example of body area layout for
page layouts C and D**

6.3.5.4 BasicFloat

BasicFloat is a constituent constraint that defines a lowest level frame that is used to represent a single column area within a body area. A single column area is typically used to layout content in the form of a single column. This is a variably positioned frame.

The dimension of this frame in the direction orthogonal to the layout path of the body area is fixed or defaults to the maximum value allowed within the body area.

The dimensions of this frame in the direction parallel to the layout path of the body area is specified as 'Rule-B'. This dimension is therefore automatically adjusted during the layout process to be the minimum required to contain all the content allocated to the frame.

The layout path specified for this frame is the same as that specified for the body area. Content may only be laid out in this frame in the direction of the layout path specified.

6.3.5.5 SnakingColumns

SnakingColumns is a constituent constraint that defines a composite frame that represents a snaking column area within a body area. A snaking column area is typically used for the layout of one or more columns of content in which the content is allowed to flow freely from one column to the next.

This frame is variably positioned. Its immediate subordinates consist of one or more frames of the type ColumnVariable. Examples of the layout of SnakingColumns frames are given in figure 8.

The dimension of a SnakingColumns frame in the direction orthogonal to the layout path of the body area is fixed or defaults to the maximum value allowed within the body area.

The dimensions of this frame in the direction parallel to the layout path of the body area is specified as 'Rule-B'. This dimension is therefore automatically adjusted to accommodate the subordinate frames which are laid out in it.

The layout path for a SnakingColumns frame may be specified as 0 or 180 degrees in the case of page layout A, 90 or 270 degrees in the case of page layout B, and 270 degrees in the cases of page layouts C and D.

The attribute "balance" may be specified for a SnakingColumns frame to indicate that two or more of the subordinate ColumnVariable frames are to be approximately equal in length in the vertical dimension in the case of page layout A and are to be approximately equal in length in the horizontal dimension in the cases of page layouts B, C and D (see note). Note that "approximately equal" in the context of the "balance" attribute means that the leading edges of the layout objects being balanced are aligned, as closely as possible, to a line orthogonal to the layout path for the objects.

NOTE - The attribute "balance" may be ignored when the subordinate ColumnVariable frames have unequal widths.

Figure 8 - Examples of the layout of snaking columns

6.3.5.6 SynchronizedColumns

SynchronizedColumns is a constituent constraint that defines a composite frame that represents a synchronized columns area within a body area. A synchronized columns area is typically used to represent one or more columns of content such that the content laid out in each column belongs to different layout streams. Thus content laid out in one column is not allowed to flow into the next column.

This type of column layout is typically used when it is required to layout separate amounts of content in parallel with one another such that they are aligned. Examples are the synchronized layout of content belonging to different languages and the layout of a figure in parallel with some text. An example is shown in figure 9.

With regard to positioning and dimensioning, SynchronizedColumns frames have the same characteristics as SnakingColumns frames.

The immediate subordinates of a SynchronizedColumns frame consist of any number of frames of the type ColumnFixed.

The layout path for a SynchronizedColumns frame is 270 degrees for page layout A, 0 degrees for page layout B and 180 degrees for page layouts C and D.

**Figure 9 - Example of synchronized column layout
for page layout A**

6.3.5.7 ColumnVariable

ColumnVariable is a constituent constraint that defines a lowest level frame that is used to represent a column of content within a SnakingColumns frame. This is a frame which is variably positioned.

The dimension of this frame in the direction parallel to the layout path of the superior SnakingColumns frame (that is, the column width) is fixed. The dimensions of different instances of ColumnVariable frames within a given SnakingColumns frame may differ to allow columns of different widths to be specified.

The dimension in the direction orthogonal to the layout path of the superior frame (that is, the column length) may be specified as 'Rule-B' or 'maximum-size'.

The layout path for ColumnVariable frames is 270 degrees in the case of page layout A, 0 degrees in page layout B and 180 degrees in page layouts C and D.

All ColumnVariable frames subordinate to the same SnakingColumns frame shall have the same category name; different names may be used for ColumnVariable frames laid out in different SnakingColumns frames.

6.3.5.8 ColumnFixed

ColumnFixed is a constituent constraint that defines a lowest level frame that is used to represent a column of content within a SynchronizedColumns frame. This is a frame which has a fixed position.

The dimension of this frame in the direction orthogonal to the layout path of the superior SynchronizedColumns frame (that is, the column width) may be fixed or specified as 'maximum size' (see note) in all page layout types. This dimension may differ for different instances of ColumnFixed frames within a given SynchronizedColumns frame to allow columns of different widths to be specified. However, the widths shall be specified such that the columns do not overlap.

The dimension of this frame in the direction parallel to the layout path of the superior frame (that is, the column length) may be specified as 'Rule-B' or 'maximum-size' in the cases of page layouts A and B. In the cases of page layouts C and D, this dimension may only be specified as 'maximum-size'.

The ColumnFixed frames subordinate to a given SynchronizedColumns frame shall have different category names.

The layout path for ColumnFixed frames shall be equal to the layout path of the superior SynchronizedColumns frame.

The content laid out in different ColumnFixed frames within the same SynchronizedColumns frame may be specified as 'synchronized' by using the attribute "synchronization".

NOTE - The value 'maximum size' may only be specified for the right most ColumnFixed frames (relative to the page co-ordinate system) laid out in a SynchronizedColumns frame, to prevent overlapping of the frames.

6.3.5.9 FootnoteArea

FootnoteArea is a constituent constraint that defines a lowest level frame that is used to represent a footnote area within a body area. A footnote area is typically used for the layout of footnotes.

Frames of this type are variably positioned with a positioning fill order specified as 'reverse'. Hence this frame is positioned adjacent to the leading edge of the VariableCompositeBody frame.

The dimension of FootnoteArea frames in the direction orthogonal to the layout path of its superior frame is fixed or specified as 'maximum size'. In the direction of the layout path, the dimension is specified by 'Rule-B' which means that this dimension is automatically adjusted to contain all the content that is allocated to it.

The layout path for FootnoteArea frames is the same as that specified for the body area.

The content that may be laid out in this frame is limited to the content that is associated with basic logical objects which are subordinates of the composite logical object 'FootnoteBody'. To achieve this, the permitted categories attribute of this frame shall specify the category 'Footnote', the same name required on the basic logical objects for footnotes (see 6.2.3.10 and 6.2.3.11).

6.3.6 Header and footer area characteristics

6.3.6.1 General characteristics

The header and footer areas consist of either basic areas or composite areas.

A basic header or footer area is an area into which the content is directly laid out. This type of area is represented by a constituent constraint of the type BasicHeader or BasicFooter respectively.

A composite header or footer area is an area which is subdivided into separate sourced content and arranged content areas to provide greater versatility with regard to the layout of the content. This type of area is represented by a constituent constraint of the type CompositeHeader or CompositeFooter respectively.

In the case of basic header or footer areas, the content allocated to these areas is derived from the common part of the logical structure of a document. In the case of composite header or footer areas, the content may again be derived from the common part of the logical structure of a document but the content may also be derived from common content specified in the generic layout structure.

6.3.6.2 BasicHeader and BasicFooter

BasicHeader and Basic Footer are constituent constraints that define lowest level frames that represent areas within a page that are reserved for common content.

These types of frame have fixed positions and dimensions. The positioning of these frames within a page and the layout paths that may be specified for them depends upon the page layout type used (see 6.3.4.5)

The content that is laid out in these frames is derived, using the logical source mechanism, from the content associated with the composite logical object classes of the type CommonContent.

6.3.6.3 CompositeHeader and CompositeFooter

CompositeHeader and CompositeFooter are constituent constraints that define composite frames that represent areas within a page that are reserved for common content.

These types of frame have fixed positions and dimensions. The positioning of these frames within a page and the layout path that may be specified for them depends upon the page layout type used (see 6.3.4.5)

The subordinates of these frames may consist of either:

- a) any number and combination of variably positioned frames of the types SourcedContentVariable and ArrangedContentVariable.

or

- b) any number and combination of fixed positioned frames of the types SourcedContentFixed and ArrangedContentFixed.

In case b), the subordinate frames may overlap without restriction.

6.3.6.4 SourcedContentVariable

A SourcedContentVariable frame is a constituent constraint that defines a lowest level frame that represents a region within a header footer area that contains common content derived from the generic logical structure. This frame is variably positioned and its layout path is the same as that of the containing header or footer area.

The dimension of this frame in the direction orthogonal to the layout path of the superior frame is fixed or specified as 'maximum-size'. The dimensions of the frame in the direction parallel to the layout path of the superior frame is specified as either fixed or 'Rule-B'.

This frame is required to specify the attribute "logical source" which indicates the particular instance of the constituent constraint `CommonContent` which contains the content to be laid out in this frame.

Typically, this frame is used for the positioning of content which is generated during the layout process, such as a character sequence containing a page number.

6.3.6.5 `ArrangedContentVariable`

An `ArrangedContentVariable` frame is a constituent constraint that defines a lowest level frame that represents a region within a header or footer area that contains pre-defined common content contained in the generic layout structure. This frame is variably positioned and its dimensions are fixed.

This frame references one or more blocks of type `GenericBlock` (see 6.3.7) which contain the content to be laid out in this frame. Thus, this frame is typically used when it is required to layout pre-determined common content.

6.3.6.6 `SourcedContentFixed`

A `SourcedContentFixed` frame is a constituent constraint that defines a lowest level frame that represents a region within a header footer area that contains common content derived from the generic logical structure. This frame has a fixed position and dimensions and its layout path is equal to that of the containing header or footer area.

This frame is required to specify the attribute "logical source" which indicates the particular instance of the constituent constraint `CommonContent` which contains the content to be laid out in this frame.

Thus, as in the case of `SourcedContentVariable` frames, this frame is used for the positioning of content which is generated during the layout process, such as a character sequence containing a page number.

6.3.6.7 `ArrangedContentFixed`

An `ArrangedContentFixed` frame is a constituent constraint that defines a lowest level frame that represents a region within a header or footer area that contains pre-defined common content derived from the generic layout structure. The position and dimensions of this frame are fixed. This frame references one or blocks of type `GenericBlock` (see 6.3.7) which contain the content to be laid out in this frame. Thus this frame is typically used when it is required to layout common content at pre-determined positions in the header or footer areas.

6.3.7 GenericBlock and SpecificBlock

Two types of constituent constraints of the type 'block' are defined, namely GenericBlock and SpecificBlock.

Object classes of the type GenericBlock may occur in the generic layout structure referenced by the "generator for subordinates" of object classes of the types ArrangedContentVariable and ArrangedContentFixed. When the layout process is performed to produce a document in formatted processable form, equivalent blocks may occur in the specific layout structure. Objects of this type are restricted to occur within the header and footer areas of the page.

Objects of the type SpecificBlock may only occur in the specific layout structure. They are created during the document layout process and result from the layout of basic logical objects into lowest level frames that constitute the body, header and footer areas.

6.4 Document layout characteristics

Mechanisms for controlling the allocation of logical constituents to various areas in the layout structure are defined in 6.4.1. Mechanisms for controlling the layout of the content within the allocated areas are defined in 6.4.2.

These mechanisms relate to documents for which a generic layout structure is specified. When a generic layout structure is not present, then these mechanisms are restricted as described in 6.4.3.

6.4.1 Flow controls

Various mechanisms are provided to control the allocation of constituent constraints representing the 'body' parts of the logical structure of a document to pages sets, pages and body areas. These are described in 6.4.1.1, 6.4.1.2 and 6.4.1.3. The mechanisms for controlling the layout of the 'common' parts of a document are described in 6.4.1.4.

6.4.1.1 Allocation of content to page sets

Two methods of allocating the constituent constraints associated with the 'body' part of the document to page sets are provided:

- a) Layout in a nominated page set;
- b) Starting a new page set.

The first method provides the ability to specify that a part of a document is to be laid out entirely within a specified page set. This may be specified for constituent constraints of the types Passage, NumberedSegment and Paragraph using the attribute "layout object class" which specifies the object class identifier of the required class of page set.

The second method provides the ability to specify that the logical objects derived from a particular logical constituent constraint in a document and all subsequent parts of a document are to be laid out starting at the beginning of a new page set. This may be specified for the following logical constituent constraints:

- Passage;
- NumberedSegment;
- Paragraph;
- Number;
- BodyText;
- BodyRaster;
- BodyGeometric.

This is achieved using the attribute "new layout object" which specifies the object class identifier of the required class of page set.

6.4.1.2 Page breaks

This provides the ability to specify that the logical objects derived from a particular logical constituent constraint in a document and all subsequent parts of a document are to be laid out starting at the beginning of a new page. The page specified shall belong to the page set in which the logical objects from the immediately preceding logical constituent constraint is laid out (see note).

This may be specified for the following logical constituent constraints:

- Passage;
- NumberedSegment;
- Paragraph;
- Number;
- BodyText;
- BodyRaster;
- BodyGeometric.

This is achieved using the attribute "new layout object". This attribute may specify the value 'page' indicating that the logical constituent constraint is to be laid out starting on the next available page which may be of any class. Alternatively, the attribute may specify that the logical constituent constraint is to be laid out starting on a page of a particular class; this is achieved by specifying the object class identifier of the required page class.

NOTE - The specification of a page break shall not be used to layout part of a document in a new page set. If a new page set is required, then this shall be explicitly specified as described in 6.4.1.1.

6.4.1.3 Allocation of content to body areas

If the page to which the content is allocated contains a basic body area, then the content is laid out in sequential order in that body area in the form of a single column.

If the page contains a composite body area, then the content is allocated to single, snaking and synchronized columns areas and footnote areas as described below.

6.4.1.3.1 Layout of content into column areas

When laying out content into a composite body area having more than one subordinate frame class (excluding FootnoteArea frame classes), it is necessary to indicate which of the column areas is to be used.

Logical objects of the types Paragraph, Number, FootnoteReference, BodyText, BodyRaster and BodyGeometric may be specified to be laid out in instances of one or more single columns area, snaking columns area or synchronized columns area. This is done by giving each such basic logical component a value of the attribute 'layout-category' which corresponds to the value of the attribute 'permitted-categories' which applies to the lowest level of frame in which the content is to be laid out.

Note that any basic logical objects in the specific logical structure to which this attribute does not apply will be laid out only in a lowest level frame which has the implicit value of the attribute 'permitted-categories'.

The use of layout categories ensures that if there is insufficient area on one page to lay out all of the content allocated to a particular type of area, then the laying out of the content will automatically continue in the same type of area in a succeeding page. Thus content is allowed to flow freely from one page to the next when the type of layout used at the end of one page is the same as that at the beginning of the succeeding page.

It is necessary to ensure the correct use of the mechanism for the layout of independent layout streams. In the absence of additional layout directives, content may be placed in available space within an earlier frame of the specified 'permitted-categories'. If this is not intended, it may be prevented by the use of the attribute 'new-layout-object'.

The attribute 'new-layout-object' may be applied to logical components of the types NumberedSegment, Paragraph, Number, FootnoteReference, BodyText, BodyRaster and BodyGeometric, whenever a change in column layout is required.

The attribute "new layout object" may specify the identifier of the frame class that represents the single, snaking or synchronized column area required. In the case of single or synchronised column areas, the attribute "new layout object" may indicate the category name corresponding to the frame class of the single column area or of any of the columns within the synchronised column area that is required.

When layout occurs in a snaking columns area, column breaks may be indicated by using the attribute "new layout object". This attribute may specify the identifier or the category name of the frame corresponding to the column in which the layout is to continue. However only the use of a category name will ensure that a single column break is always obtained, irrespective of the frame class actually used.

When the layout is to occur in a synchronized columns area, category names are used to control the particular columns that are to be used to layout the logical entities. Each column within a synchronized columns area shall have a different permitted category and each basic logical entity to be laid out in this particular area shall have a category name corresponding to a name allocated to one of the columns. The logical entities allocated to different columns may be aligned using the attribute "synchronization".

6.4.1.3.2 Layout of footnotes

The logical objects derived from basic logical constituent constraints that represent the content belonging to a footnote (i.e FootnoteNumber and FootnoteText) are constrained to be laid out in a footnote area which is represented by a FootnoteArea frame (see 6.3.5.9).

This constraint is specified by means of category names. That is, the logical constituents of the types FootnoteNumber and FootnoteText and layout constituents of the type FootnoteArea are all required to have the category name 'Footnote'.

More than one footnote may be placed in a footnote area within a given body area. In this case the content belonging to the footnotes are laid out sequentially in the footnote area in accordance with their reading order.

If the content belonging to a footnote cannot all be accommodated in the footnote area on one page, then the content may freely flow into the footnote area on the next page. Alternatively, it is possible to specify that a footnote is to be laid out entirely within a particular footnote area. This is achieved using the attribute "indivisibility".

6.4.1.4 Allocation of content to header-footer areas

A header or footer area may be basic or composite (see 6.3.6.1). In the case of a basic area, the frame representing that area specifies the attribute "Logical source" which indicates the particular instance of the constituent constraint of the type CommonContent that is to be laid out in that area. The basic logical constituents subordinate to CommonContent are then laid out in accordance with their sequential order.

In the case of a composite header or footer area (see 6.3.6.3), the area is divided into one or more separate areas, each of which is represented by a lowest level frame. The content allocated to the separate areas may be derived from one of two sources. That is, the content may be pre-defined and represented by one or more blocks which are directly associated with the lowest level frame. Alternatively, the lowest level frame may specify the attribute "logical source" which, as above, indicates the particular logical object of the type CommonContent that is to be laid out in that frame.

6.4.2 Layout of the document content

Various constraints may be specified to control the layout of the content into the body, header and footer areas. These constraints are described below.

6.4.2.1 Margins

The margins are the minimum distances, or offsets, between a part of the document content and the edge of the particular area in which that content is laid out. The margins define the maximum extents of the available area into which the content shall be positioned.

Margins may be specified for any constituent constraint representing a basic logical object; different margin values may be specified for different constituent constraints without restriction.

Four margins may be independently specified for each constituent constraint, namely:

- trailing edge margin;
- leading edge margin;
- right hand edge margin;
- left hand edge margin.

These margins are defined in relationship to the layout path specified for the frame in which the content is to be laid out in (see Figure 10).

Any combination of the above margins may be specified for a particular constituent constraint. These margins are specified by the attribute "offset". Any value may be specified in units of BMUs. If a particular margin is not specified, then it is assumed to be 0 BMUs.

6.4.2.2 Separation

Leading separations is the minimum distance between one basic logical object and the next one, if any, when they are laid out; trailing separation is the minimum distance between one basic logical object and the previous one, if any, when they are laid out. Both may be specified for basic logical components of any constituent constraint types. These distances are specified in BMUs by the attribute "separation". If no value is specified, then the minimum distance is assumed to be 0 BMUs.

6.4.2.3 Indivisibility

Indivisibility provides the means to specify whether or not a logical object derived from a basic or composite logical constituent constraint is allowed to be split over more than one page or over more than one area within a page. It may be specified for constituent constraints of the types Passage, NumberedSegment, Paragraph, Footnote, Number, FootnoteReference and BodyText. The attribute "indivisibility" is used to specify this feature.

6.4.2.4 Same layout object

Same layout object provides the means to specify that the start of the content associated with a logical object and the end of the content associated with the previous logical object are to be laid out within a single layout object. This may be specified for logical objects of the types NumberedSegment, Paragraph, Number, Footnote, FootnoteReference, BodyText, BodyRaster and BodyGeometric. The attribute "same layout object" is used to specify this feature.

Figure 10 - Specification of margins**6.4.2.5 Concatenation**

Concatenation provides the means to specify that the content associated with a logical object derived from a basic logical constituent constraint and the content associated with the logical object derived from previous basic logical constituent constraint are to be regarded as an unbroken stream of content. This may be specified for constituent constraints of the types BodyText, Number, FootnoteReference, FootnoteText, CommonText and PageNumber. The attribute "concatenation" is used to specify this feature.

6.4.2.6 Block alignment

Block alignment allows the content associated with a basic logical entity to be specified as 'left aligned', 'right aligned' or 'centred' within the area in which that content is laid out. Left aligned means that the content is laid out adjacent to the left hand edge margin. Right aligned means that the content is laid out adjacent to the right hand edge margin and centred means that the content is laid out midway between the left and right margins.

This feature may only be specified using the attribute "block alignment" for constituent constraints of the types Number, FootnoteReference, FootnoteText, PageNumber, FootnoteNumber, BodyText and CommonText, when they contain formatted character content, BodyRaster, and BodyGeometric, CommonRaster and CommonGeometric.

6.4.3 Layout controls applicable in the absence of a generic layout structure

In processable form documents the generic layout structure is optional. If the generic layout structure is omitted, then it is the responsibility of the receiver to define an appropriate layout structure. No limitations are placed on the layout structure that is used.

When a generic layout structure is not specified within a processable form document, then restrictions are placed on the layout control functions described in 6.4.1 and 6.4.2 that may be specified within the document. These restrictions are indicated below.

- a) It is not possible to specify that certain logical parts of a document are to be allocated to a given page set or that a part of a document is to be laid out starting in a new page set, as defined in 6.4.1.1.
- b) It is possible to specify page breaks as defined in 6.4.1.2 but it is only possible to indicate that the layout shall begin on a new page. It is not possible to specify a particular page class.
- c) The logical parts of the document that are intended to be laid out in the body area and in the header/footer areas of each page may be distinguished from each other by means of application comments specified for them (see 6.6.4). An exception is that it is not possible to distinguish whether a particular portion of the common content is to be placed in a header or footer area (or both).
- d) It is not possible to indicate the type of layout area to be used to layout each logical constituent in the body part of a document. That is, it is not possible to indicate whether single column or multiple column areas are to be used (see 6.4.1.3.1). This shall be decided by the receiver.
- e) Footnotes within the body part of a document may be distinguished by use of the attribute "application comments". Footnotes are intended to be read and laid out separately from the other logical constituents of the body part (see 6.4.1.3.2). However, it is the responsibility of the receiver to decide how footnotes are laid out.
- f) Margins, separation, indivisibility, same layout object, concatenation, and block alignment, as defined in 6.4.2 may all be specified. Only one restriction applies. Indivisibility (see 6.4.2.3) may be assumed to specify that a logical constituent constraint is not to split over more than one page but indivisibility shall not be specified for other types of layout areas such as single or multiple column areas.

6.5 Content layout and imaging characteristics

A document may contain character, raster graphics and geometric graphics content.

The content architectures that may be specified using the attribute "content architecture class" are formatted character, processable character, formatted processable character, formatted processable raster graphics and formatted processable geometric graphics. Any of these may be specified as the default in the document profile.

6.5.1 Character content

6.5.1.1 Introduction

This subclause defines the features that are applicable to the character content contained in a document and the presentation attributes and control functions that may be used to specify these features. These features may apply to basic logical and layout components unless otherwise indicated.

The default values for the following features may be specified in the document profile:

- graphic character sets;
- graphic character subrepertoire;
- code extension announcers;
- line spacing;
- character spacing;
- character path;
- line progression;
- character orientation;
- graphic rendition, including the parameters values:
default rendition, increased intensity (bold), italicised,
underlined, crossed out, primary font, 1st alternative font, 2nd
alternative font, 3rd alternative font, 4th alternative font, 5th
alternative font, 6th alternative font, 7th alternative font, 8th
alternative font, 9th alternative font, doubly underlined, normal
intensity, not italicised, not underlined, not crossed out;

- line layout table;
- indentation;
- alignment;
- first line offset;
- itemization;
- widow size;
- orphan size;
- character fonts;
- kerning offset;
- proportional line spacing;
- initial offset.

The specification in a document of a non-basic feature by a presentation attribute or control function shall be indicated in the document profile.

6.5.1.2 Character content architecture class

Processable and formatted processable form documents may contain processable, formatted or formatted processable character content. Formatted form documents may contain formatted and formatted processable character content.

When using character content, any number of content portions may be associated with a basic component.

The content information in a content portion may be absent. This is to allow the representation and interchange of documents in which parts of the content may be supplied, for example, during subsequent editing.

6.5.1.3 Character repertoires

The basic character repertoire supported by this profile is composed of the 94 characters of ISO-IR6 (the IRV of ISO 646 revised 1991) plus the character space.

Any other graphic character set which is registered in accordance with ISO 2375 may be designated and invoked at any point in the document provided its use is indicated in the document profile as a non-basic value using the character presentation feature "graphic character sets". No locking shift functions are specified in this presentation feature.

The code extension techniques allowed for the designation and invocation of character sets to the left hand side and right hand side of the 8-bit code table (GL and GR respectively) are defined in 6.5.1.4.

Using these code extension techniques, the graphic character sets designated and/or invoked at the beginning of a content portion containing character content are specified by the presentation attribute "graphic character sets". The graphic character sets may also be changed at any point within a content portion.

The default graphic character sets which apply to the content portions within a document may be specified in the document profile using the presentation attribute "graphic character sets".

If the character set defined in ISO 6937-2 with or without Addendum 1 is designated and invoked, then the use of any of its subrepertoires registered according to ISO 7350 may be specified using the presentation attribute "graphic character repertoire". All sub-repertoires are non-basic and their use shall be indicated in the document profile. The repertoire shall not be changed within a content portion.

NOTES

- 1: The basic character repertoire supported by this profile is not the standard default value specified in [ISO 8613-6|CCITT Recommendation T.416]; hence it may be necessary to specify, in the document profile of a particular document, that this is the default value being used for that document.
- 2: Revised CCITT Recommendations T.50 and T.51 and new CCITT Recommendation T.52 are under preparation. CCITT Recommendations T.50 and T.51 are intended to be completely compatible with ISO 646 revised 1991 (ISO IR-6) and ISO 6937 (under revision) respectively.

6.5.1.4 Code extension techniques

The code extension techniques specified in ISO 2022 may be used subject to the following restrictions:

- i) G0 set: only ISO-IR6 (the IRV of ISO 646 revised 1991), ISO-IR2 (the primary set of ISO 6937-2), or any other version of ISO 646 may be designated for this set; these graphic character sets may only be invoked in GL.
- ii) G1, G2, G3 sets: no restrictions are placed on the character sets that may be designated for these sets; these graphic character sets may only be invoked in GR.
- iii) The locking and single shift functions allowed are as follows:
 - LS0 to invoke the G0 set into GL;
 - LS1R to invoke the G1 set into GR;
 - LS2R to invoke the G2 set into GR;
 - LS3R to invoke the G3 set into GR;
 - SS2 to invoke one character from the G2 set into GL;
 - SS3 to invoke one character from the G3 set into GL.

(Here GL and GR refer to the left and right hand parts respectively of the 8-bit code table)

- iv) When specifying the presentation attribute "graphic character sets", it is necessary to invoke character sets for both GL and GR. Thus an allowed character set shall be designated into G0 (see item i) above) and invoked into GL. It is also necessary to invoke a character set into GR which has been designated into the G1, G2 or G3 set.
- v) The empty set shall be designated into G1 and invoked into GR if no other specific character set is invoked into GR.

The code extension techniques allowed are illustrated in figures 11 and 12.

The announcement and encoding of these functions are to be as specified in ISO 2022.

The code extension techniques that are used or may be used in a basic component shall be specified by the presentation attribute "code extension announcers." The default code extension announcers used throughout a document may be specified in the document profile, using the presentation attribute "code extension announcers".

NOTE - In accordance with [ISO 8613-6|CCITT Recommendation T.416], there is no restriction concerning the number of graphic character sets which may be designated and/or invoked in the presentation attribute "graphic character sets" providing the restrictions defined in this subclause are applied. Hence designation to a particular G set overrides a previous designation to that set and invocation to GL or GR overrides the previous invocation to the GL or GR respectively. Thus the sequential order of designation and/or invocation sequences in the attribute "graphic character sets" is significant.

Figure 11 - Code extension features (basic case)

Figure 12 - Code extension features (all possible cases)

6.5.1.5 Line spacing

Any value of line spacing may be specified. Values of 150, 200, 300 and 400 BMUs are basic; the use of any other value in a document is non-basic and shall be indicated in the document profile.

The line spacing may be specified at the beginning of the content associated with a basic component using the presentation attribute "line spacing". The value may be changed anywhere within the content portion using the control functions SVS and SLS.

6.5.1.6 Character spacing

Any value of character spacing may be specified. Values greater than or equal to 100 are basic; the use of any other value in a document is non-basic and shall be indicated in the document profile.

The character spacing may be specified at the beginning of the content associated with a basic component using the attribute "character spacing". The value may be changed anywhere within a content portion using the control functions SHS or SCS.

NOTES

- 1: A character spacing of 160 BMUs is provided for use with Korean Han-gul characters.
- 2: SHS parameters of 0, 1, 2, 3 and 4 are provided. The use of parameters 5 and 6 may be provided in a future edition of this standard for use with Chinese characters.

6.5.1.7 Character path and line progression

Both horizontal and vertical writing directions may be used within a document. In the case of horizontal writing, the characters progress either from left to right or from right to left across the page and the line progression is from the top of the page to the bottom. In the case of vertical writing, the characters progress from the top of the page to the bottom and the line progression is from the right to the left. The use of these writing directions is restricted by the page layout type used.

For page layout A, only horizontal writing may be used in the body, header and footer areas. Thus, in this case the character path and line progression is specified either as 0 and 270 degrees respectively or 180 and 90 degrees respectively.

For page layout B, again only horizontal writing may be used in the body, header and footer areas. However, in this case the content in the body area is presented for viewing with the page in landscape orientation and the content in the header and footer areas is presented for viewing when the page is in the portrait orientation.

Thus for page layout B, in the body area the character path and line progression is specified either as 90 and 270 degrees respectively or 270 and 90 degrees respectively. In the header and footer areas, the character path and line progression is specified as in page layout A.

For page layout C, only vertical writing may be used in the body area and only horizontal writing may be used in the header and footer areas. Thus in the body area the character path and line progression are specified as 270 and 270 degrees respectively. In the header and footer areas, the character path and line progression is specified as in page layout A.

For page layout D, only vertical writing may be used in the body, header and footer areas. Thus in all these areas, the character path and line progression are specified as 270 and 270 degrees respectively.

A character path value of 0 degrees and a line progression value of 270 degrees are basic values. All other values are non-basic and their use in a document shall be indicated in the document profile.

The values of character path and line progression may be specified at the beginning of the content associated with a basic component using the presentation attributes "character path" and "line progression" respectively. These values cannot be changed within a content portion.

6.5.1.8 Character orientation

The character orientation may be specified as 0 or 90 degrees depending on whether vertical or horizontal writing is used (see 6.5.1.7).

When horizontal writing is used, characters may only be orientated at 0 degrees. When vertical writing is used, characters may be orientated at 90 degrees.

A value of 0 degrees is basic; a value of 90 degrees is non-basic and its use in a document shall be indicated in the document profile.

The value of the character orientation is specified at the beginning of the content associated with a basic component by the presentation attribute "character orientation". This value cannot be changed within the content.

6.5.1.9 Emphasis

The following modes of emphasising graphic characters may be distinguished:

- normal rendition;
- normal intensity;
- increased intensity (bold);
- italicized;
- not italicized;
- underlined;
- doubly underlined;
- not underlined;
- crossed-out;
- not crossed-out.

All the above mentioned modes of emphasis are basic. If no default mode is explicitly specified in the document profile, then the default mode is normal rendition.

The mode of emphasis may be specified at the beginning of the content associated with a basic component using the presentation attribute "graphic rendition". The mode may be changed anywhere within the content using the control function SGR.

The mode of emphasis remains in effect within the content associated with a basic component until changed into a mutually exclusive mode or by the specification of 'normal rendition'. Mutually exclusive modes are normal/increased intensity, italicized/not italicized, underlined/doubly underlined/not underlined and crossed out/not crossed-out. One mode from each mutually exclusive set may be in operation at any point in the document content.

Normal rendition cancels the effect of all modes of emphasis that are currently in operation and specifies that the text shall be displayed in accordance with the default rendition parameters set for the presentation device. Thus, for example, if it is required to ensure that the content is not underlined, then it is necessary to explicitly specify that underlined is not to be used.

6.5.1.10 Tabulation

Tabulation stop positions may be specified at any position along the character path. Each stop is specified by means of the following:

- a) The tabulation position relative to the margin position in the direction opposite to the character path;
- b) An optional alignment qualifier that specifies the type of alignment to be used at the designated tabulation position. The type may be specified as one of the following:
 - start aligned;
 - end aligned;
 - centred;
 - aligned around.

These alignment qualifiers are defined in [ISO 8613-6|CCITT Recommendation T416]. If the alignment qualifier is not explicitly specified, then it is assumed that start aligned is to be used.

Only one set of tabulation stops can be specified to be applicable to the content associated with a basic component. No limit is placed on the number of tabulation stops that may be specified within a given set.

The set of tabulation stop positions associated with the content of a basic component are specified using the presentation attribute "line layout table". Tabulation stop positions are invoked within the content using the control function STAB.

The tabulation reference numbers used in the STAB controls and associated presentation attribute "line layout table" shall be chosen so that, in any given line layout table the reference numbers are unique, sequential in the direction of the character path and do not include leading zeroes.

6.5.1.11 Indentation

Indentation is the distance between the first character on a line of content and the position of the margin that is in the direction opposite to the direction of the character path. Thus the value of the indentation specified determines the line home position (as defined in [ISO 8613-6|CCITT Recommendation T.416]).

Indentation acts as temporary alteration in the position of the offset in the direction opposite to the direction of the character path. When text is formatted, it is intended to be laid out between the indentation position and the margin position in the direction of the character path.

Any value of indentation may be specified for basic logical components using the presentation attribute "indentation". The indentation value shall not be changed within a content portion.

6.5.1.12 Alignment

This feature is concerned with how the first and last characters on each line of character content are to be laid out during the formatting process.

The following values of alignment may be specified as basic:

- start aligned;
- end aligned;
- centred;
- justified.

The semantics of these values are as defined in [ISO 8613-6|CCITT Recommendation T.416].

The presentation attribute "alignment" is used to specify the alignment that is applicable to the content associated with a basic component. The alignment value cannot be changed within a content portion.

6.5.1.13 First line format

This feature specifies how the first line of the content associated with a basic component is to be laid out and provides for the itemisation of paragraphs.

It allows the first character in the content to be positioned at some point along the character path relative to the indentation position (as specified in 6.5.1.11). This point may be in the direction of the character path or in the direction opposite to the direction of the character path relative to the indentation position.

In addition, this feature provides for the specification of an item identifier on the first line. The item identifier is a string of characters that precedes and is separated from the remaining characters that form the first line. The control function CR (Carriage Return) is used as the separator.

The features provided correspond to examples 10.1 to 10.5 shown in figure 10 of [ISO 8613-6|CCITT Recommendation T.416].

First line format is specified by the presentation attributes "first line offset" and "itemisation", and "indentation". Only those values of the attributes which combine to form the examples shown in figure 10 of [ISO 8613-6|CCITT Recommendation T.416] may be used.

6.5.1.14 Widow and orphan sizes

The widow size specifies the minimum number of lines of content that shall be allocated to a following frame or page when the content associated with a basic logical component is laid out such that it flows over two frames or pages. To accommodate this, it may be necessary to move a number of lines of content from one frame or page to the next frame or page.

The orphan size specifies the minimum number of lines of content that shall be placed in the current frame or page when the content associated with a basic logical component is split over two frames or pages. If this minimum cannot be accommodated, then the whole content shall be placed in the next frame or page.

Any value of widow or orphan size may be specified using the presentation attributes "widow size" and "orphan size" respectively.

Widow and orphan size may only be specified for character content placed in body area of pages.

6.5.1.15 Fonts

Any number of fonts may be used within a document. The fonts used in a particular document are specified in the document profile using the attribute "fonts list".

Further information concerning the specification of font references in the document profile is given in Annex B.2.

The fonts that may be used within the content associated with each basic component are specified by the presentation attribute "character fonts". Up to 10 fonts taken from the list specified by the attribute "fonts list" may be specified by the attribute "character fonts".

The font to be used at the start of the content associated with a basic component is specified using the attribute "graphic rendition". The fonts used within the content may be changed using the control function SGR.

The document profile may specify, using the attribute "character fonts", a default set of up to 10 fonts that are applicable to the whole document.

6.5.1.16 Reverse character strings

Bi-directional writing is supported by this profile (see 6.5.1.7). Hence, a string of characters in a content portion associated with a basic component may be specified to be imaged in the reverse direction of the immediately preceding character string. Such strings may be specified by the control function SRS as defined in [ISO 8613-6|CCITT Recommendation T.416].

This control function is provided for cases in which the text belongs to different languages and the character content is written, for example, from left to right or from right to left within the same line of characters, dependent upon the language and/or character set being used.

NOTE - The use of this control function cannot be indicated in the document profile. Thus it is intended that implementations shall ignore this control function when reverse character string layout and presentation is not supported.

6.5.1.17 Kerning offset

A kerning offset value for the content associated with a basic component may be specified using the presentation attribute "kerning offset". It is necessary to specify such a value when certain fonts are invoked to ensure that no part of character images are positioned outside the boundary of the available area.

6.5.1.18 Proportional line spacing.

The use of proportional line spacing may be invoked for the content associated with a basic logical component using the attribute "proportional line spacing". When this invocation occurs, the line spacing between each pair of consecutive lines is determined in an implementation-defined way from the attributes associated with the fonts used within the two lines and may vary from one line to the next. This process is application dependent.

6.5.1.19 Superscripts and subscripts

Superscripts and subscripts may be specified anywhere within the content associated with a basic component by using the control functions PLU and PLD. The use of these control functions shall be in accordance with [ISO 8613-6|CCITT Recommendation T.416].

6.5.1.20 Line breaks

The control functions BPH and NBH may be inserted in processable form character content to indicate where line breaks may occur or may not occur respectively, when the content is laid out.

6.5.1.21 Substitution of characters

The control function SUB is provided to represent characters produced by a local system that cannot be represented by a character within a character set supported by this profile.

6.5.1.22 Initial point

The initial point which is applicable to basic layout components may be specified by the attribute "initial offset". Any value may be specified.

6.5.1.23 Use of control functions

The following is a list of all the control functions and parameter values (where applicable) that may be specified in character content:

- SHS - select character spacing
(allowed parameter values: 0, 1, 2, 3, 4);
- SCS - set character spacing
(allowed parameter values: any);
- SVS - select line spacing
(allowed parameter values: any);
- SLS - set line spacing
(allowed parameter values: any);
- SGR - set graphic rendition
(allowed parameter values: 0, 1, 2, 3, 4, 9-19, 21-24, 29);
- STAB - selective tabulation
(allowed parameter values: any);
- SRS - start reverse string
(allowed parameters: any);
- VPB - line position backward (allowed parameter values: any);
- VPR - line position relative (allowed parameter values: any);
- PLD - partial line down;
- PLU - partial line up;
- BPH - break permitted here;
- NBH - no break here;
- JFY - no justify;

- SUB - substitute character;
- SP - space;
- CR - carriage return;
- LF - line feed;
- SOS - start of string ;
- ST - string terminator
- code extension control functions (see 6.5.1.4).

The use of all these control functions, with the exception of SP, CR, LF, SOS and ST are described in various sections in 6.5.1.

6.5.1.24 Formatting the content

The attribute "formatting indicator" shall not be specified within documents that are conformant with this profile.

6.5.2 Raster graphics content

6.5.2.1 Introduction

This subclause defines the features that are applicable to the raster graphics content contained in a document. These features may apply to basic logical and layout components unless otherwise indicated.

The default values for the following features may be specified in the document profile:

- type of coding;
- compression;
- pel spacing;
- spacing ratio;
- clipping;
- image dimensions.

The specification in a document of a non-basic feature by a presentation or coding attribute or control function shall be indicated in the document profile.

6.5.2.2 Raster graphics content architectures

Only the formatted processable raster graphics content architecture class may be used in documents that conform to this document application profile. This type of content may be used in processable, formatted and formatted processable form documents.

When using raster graphics content, only one content portion may be associated with an object or object class.

The content information in a content portion may be absent. This is to allow the representation and interchange of documents in which parts of the content may be supplied, for example, during subsequent editing.

Also, the scalable or fixed dimension content layout process may be used when laying out and imaging the content depending upon the specification of the presentation attributes "pel spacing" and "imaging dimensions" as described in 6.5.2.6 and 6.5.2.8. Both forms of content layout processes may be used in a single document.

6.5.2.3 Raster graphics encoding methods

The content may be encoded in accordance with the encoding schemes defined in CCITT Recommendations T.4 and T.6. In the case of T.4, either the one-dimensional or two dimensional encoding scheme may be used. Also the 'bit-map encoding' scheme defined in [ISO 8613-7|CCITT Recommendation T.417] may be used. All these forms of encoding may be used in a single document and all are basic. 'Uncompressed' mode of encoding may also be used but as a non-basic feature.

When using the T.4 or T.6 encoding method, the relationship between the order of pels and the order of bits in the octets in the coded data stream shall be such that the first pel in the order of bits is allocated to the least significant bit of an octet. In the case of bit-map encoding, the order of encoding shall be that the first pel is allocated to the most significant bit of an octet.

In a content portion, if content information is specified then it is required that the coding attribute "number of pels per line" is specified; the coding attribute "number of lines" may also be specified. No restriction is placed on the values that may be specified for these coding attributes. Thus this profile places no restriction of the size of the pel arrays that may be used.

The type of encoding method used is specified by the attribute "type of coding". The use of this attribute is non-mandatory. Thus, if this attribute is not specified for a particular content portion and if the content architecture class specified corresponds to the formatted processable raster graphics content architecture class, then the default encoding method is assumed to be T.6.

6.5.2.4 Pel path and line progression

The pel path and line progression supported by this profile are 0 degrees and 270 degrees respectively. This profile does not allow the specification of other values.

6.5.2.5 Clipping

A sub-region within a pel array represented by a content portion associated with a basic component may be defined using the presentation attribute "clipping". No restriction is placed on the use of this attribute.

6.5.2.6 Pel spacing

The pel spacing is the distance in BMUs between any two pels on a line when a pel array is imaged. Any value may be explicitly specified provided that the spacing between pels is not less than 1 BMU. The pel spacing need not be an integer value. Also, the value 'null' may be specified, indicating that the scalable layout process is to be used.

The specification of the value 'null' or spacings of 16, 12, 8, 6, 5, 4, 3, 2, and 1 BMU between adjacent pels are basic. The specification of any other spacing is non-basic and shall be indicated in the document profile.

The pel spacing applicable to content associated with basic logical components is specified by the presentation attribute "pel spacing".

NOTES

1. The basic pel spacing values listed above are equivalent to resolutions of 75, 100, 150, 200, 240, 300, 400, 600 and 1200 pels per 25.4mm respectively when the document is imaged in its specified size.
2. The attribute "pel spacing" specifies two integers, the ratio of which determines the pels spacing. No restriction is placed on the values of these integers.

6.5.2.7 Spacing ratio

The spacing ratio is the ratio between the pel spacing and the line spacing when a pel array is imaged. This ratio is used to determine the line spacing from the pel spacing specified.

No restrictions are placed on the value of this ratio providing that the resultant line spacing is not less than 1 BMU. Also, the line spacing need not be an integral number of BMUs. All values are basic.

The default value may be specified in the document profile. If no default value is explicitly specified then the default value is the ratio 1:1, that is, the line spacing is equal to the pel spacing.

The spacing ratio applicable to the content associated with a basic logical component is specified by the presentation attribute "spacing ratio".

6.5.2.8 Image dimensions

The image dimensions are the constraints to be applied to the size of the image produced when laying out a pel array represented by a content portion associated with a basic logical component.

These constraints are specified for basic logical components by the presentation attribute "image dimensions". The value of this attribute is only taken into account if the value of the attribute "pel spacing" is 'null'.

6.5.3 Geometric graphics content

A document may contain graphic images composed of geometric graphic content encoded as CGM metafiles in accordance with ISO 8632. Each CGM figure shall consist of a single picture only. Each GCM figure may specify its minimum dimensions.

Further information concerning the specification of geometric graphics content information is given in Annex B.

6.6 Miscellaneous features

6.6.1 Resource documents

Object classes of the types BodyText, BodyRaster and BodyGeometric, CommonText, CommonRaster, CommonGeometric and GenericBlock may refer to corresponding constituents in a resource document.

The constituents in the resource document may refer to content portions and to layout and presentation styles that are contained within the resource document. The constituents listed above are the only ones that are allowed to be referenced from another document via the resource attribute: however documents used as a resource document may contain any combination of generic constituents which is conformant to this document application profile.

6.6.2 External documents

In the case of processable and formatted processable, the generic logical structure, and the generic layout structure if present, may be contained in an external document. Note that it is not permitted to exchange one generic structure in the interchanged document whilst referencing the other through the external document.

6.6.3 Border

Border may be specified for all the frame types defined in 6.3.5 and 6.3.6 using the attribute "border". Borders may also be specified in presentation styles. All the features of border specified in [ISO 8613-2|CCITT Recommendation T.412], may be specified. The use of border is a non-basic feature and shall be indicated in the document profile. Border shall not be specified for the constituents GenericBlock and SpecificBlock.

6.6.4 Application comments

Specification of the attribute "application comments" is mandatory for all object classes contained in a document that conforms to this profile. Specification of this attribute is mandatory for all objects that do not refer to an object class. Specification of this attribute is optional for all objects that refer to object classes.

This attribute is structured so that it contains two fields. The first field is mandatory when the attribute is specified and contains a numeric string which uniquely identifies the constituent constraint applicable to the constituent for which the attribute is specified. This facilitates the processing of documents. A list of these identifiers is given in table 2.

NOTES:

- 1: The values of the constituent constraint numeric identifiers are not unique between the logical and layout structures and therefore in order to identify the constituent constraint applicable to a constituent it is necessary to know the structure of which the constituent is a part.
- 2: For constituent constraints that correspond to each other between the hierarchically related profiles to which this profile belongs, the same constituent constraint numeric identifier is specified.

The second field is optional and may contain any information that is relevant to the application or user. The format of the second field is not defined in this profile and the interpretation of this field depends upon a private agreement between the originator and recipient of the document.

The encoding of the attribute "application comments" is defined in 8.1.3. and 8.2.3.

6.6.5 Alternative representation

The content information in a content portion may be replaced by a string of characters specified in the attribute "alternative representation". This attribute may be specified in content portions that contain character, raster graphics or geometric graphics content.

The specification and use of this attribute is optional. The string of characters specified shall belong to the character repertoires indicated in the document profile attribute "alternative representation character sets" (see 6.7.4.3). If the latter attribute is not explicitly specified in the document profile, then the default defined in ISO 8613 is assumed. The control functions CR, LF and SP may also be used within the character string but no other control function is allowed; hence graphic character sets cannot be changed within the alternative representation.

6.6.6 Page Numbering

As described in 6.2.4.3, the constituent constraint PageNumber contains a content generator which may refer to a page number. This content generator is evaluated when the document is laid out and this mechanism provides a means of reproducing the appropriate number of each page of a document.

Table 2 - List of number string identifiers

Logical constituent	Constituent constraint numeric identifier
DocumentLogicalRoot	0
Passage	1
Numberedsegment	2
Number	3
Paragraph	6
Footnote	8
FootnoteNumber	9
FootnoteReference	10
FootnoteBody	11
FootnoteText	12
BodyText	14
BodyRaster	17
BodyGeometric	18
CommonContent	19
CommonText	20
CommonRaster	21
CommonGeometric	22
PageNumber	40
Layout constituent	Constituent constraint numeric identifier
DocumentLayoutRoot	0
PageSet	1
Page	2
RectoPage	3
VersoPage	4
CompositeHeader	5
VariableCompositeBody	7
ColumnFixed	8
ColumnVariable	9
SnakingColumns	10
SynchronizedColumns	11
BasicFloat	12
FootnoteArea	15
ArrangedContentFixed	16
ArrangedContentVariable	17
SourcedContentFixed	18
SourcedContentVariable	19
BasicHeader	27
BasicBody	28
GenericBlock	29
SpecificBlock	30
CompositeFooter	32
BasicFooter	33

The content generator has the following format:

`<string-literal><num-expr><string-literal>`

The format of this content generator is defined in the macro PGNUMBER (see note).

The `<string-literal>` fields are optional and are pre-defined character strings. The basic character repertoire used to specify these strings is the primary character repertoire of ISO 8859-1. Any other character repertoire, and subrepertoire if appropriate, may be used provided that it is designated and invoked by the appropriate code designation and innovation sequences, and indicated in the document profile as a non-basic value. SP and no other control functions may be used in these strings.

The field `<num-expr>` is a reference to a binding PGnum which specifies the number of the page concerned. This binding is initialised at the document layout root or page set level (see the macro INITIALISEPGNUM in 7.4.1) and automatically incremented on each successive page (see macro PAGENUMBER in 7.4.1). By placing initialisation on the layout root, rather than on the pageset class(es), the pagenumbers may be defined to be continued from one pageset to the next.

The content associated with logical object classes of the type PageNumber is laid out in a frame of one of the following types: BasicHeader, BasicFooter, SourcedContentVariable, SourcedContentFixed (see 6.3.6) using the logical source mechanism. Thus when the appropriate frame is being laid out, the field `<num-expr>` in the content generator contained in a logical object class of the type PageNumber is evaluated and this determines the value of the binding PGnum that is associated with the current page being laid out.

The number associated with the binding PGnum is applied to a string function during its evaluation in order to convert the number into a character string. This enables the number to be represented in the form of an Arabic numeric string, an upper or lower case Roman numeric string or an upper or lower case alphabetic string.

Each page class may refer to a different instance of logical object classes of the type PageNumber and this allows different page numbering formats to be used for different parts of the document.

An example of page numbering is "Page X" which consists of two concatenated character strings. The first is the literal character string 'Page' and this is concatenated to a string function denoted by 'X'. When 'X' is evaluated in a particular instance it may, for example, return the character string 'iv', the Roman numeral (lower case) for the number '4'.

NOTE - Unless otherwise stated, the macros referred to in this clause are defined in 7.3.1.

6.6.7 Segment numbering

As described in section 6.2.3.4, the constituent Number contains a content generator which when evaluated during the layout process produces an identifier which serves to identify the Numbered Segment to which the constituent Number belongs.

The format of this identifier is as follows:

```
<pre-str><num-str><suf-str>
```

This format is defined in the macro SEGMENTNUMBER (see note).

The fields <pre-str> and <suf-str> are optional prefix and suffix character strings respectively which may be of any length. The basic character repertoire used to specify these strings is the primary character repertoire of ISO 8859-1. Any other character repertoire, and subrepertoire if appropriate, may be used provided that it is designated and invoked by the appropriate designation and innovation sequences and indicated in the document profile as a non-basic value. No other control functions may be used in these strings.

The field <num-str> is the segment identifier which consists of a single number or a sequence of two or more numbers, each of which is separated by a 'separator'. The separator is a character string and may, for example, consist of a full stop or space. An example of a segment identifier is '6.3.4.2.1'. Thus segment identifiers have the general form:

```
<number>[<separator><number>]...
```

where [...]... indicates optional repetition.

In a document, the prefix and suffix character strings are string literals or carried by the bindings 'prefix -<n>' and 'suffix -<n>' respectively. The separator character strings are carried by bindings of the form <separator-<n>' and the segment identifier <num-str> is carried by the binding 'numberstring-<n>'.

In all these bindings '<n>' is a sequence of one or more digits which indicate the depth of numbering, such that n=1 indicates the number (prefix, suffix, numberstring etc) for the numbered segments immediately subordinate to a Passage, n=2 indicates the number (prefix, suffix, separator etc) for the numbered segments immediately subordinate to the first level of numbered segments, and so on. The level number shall be indicated using the smallest possible number of characters, that is, there shall be no leading zeroes.

These bindings may be initialised at the document logical root, passage or at any numbered segment level to start the numbering scheme sequence at a subordinate level of numbered segment. They may also be re-specified at any level within the numbering scheme. The initialisation of bindings is specified by the macro INITIALISEANY.

The placement of bindings initialisations for numbering schemes is significant. Initial values for numbers-n bindings shall be placed either at the Passage level, or on the NumberedSegment class which is superior to that in which the binding will be referenced. Similarly, prefix and suffixes and separators shall be initialised either at the Passage or at the immediately superior NumberedSegment level to their use. In particular note the "prefix" and "suffix" are not inherited by lower levels in hierarchy (since they belong to the content generator SEGMENTNUMBERS rather than the binding Numberstrings-n). Thus to have concatenation to say "(1).a", lower level shall have a prefix of "(" and separator of ").

In order to evaluate the value of 'numberstring-<n>' for each numbered segment, a number is assigned to each numbered segment at a given level. If the numbered segments are all of the same class then this number may be determined by the ORDINAL numeric function. If they are of different classes, then the number is carried by a binding of the form 'number-<n>'.

A different binding of the type 'number-<n>' is used for each numbered segment level and is initialised at a higher level constituent than the one in which it is used. The number associated with each numbered segment level is automatically incremented for each successive numbered segment (see the macro USENUMBERS).

The binding 'numberstring-<n>' that is applicable to a given level of numbered segment is now constructed as follows:

`<numberstring-x><separator-y><number-z>`

Hence, the segment identifier consists of a concatenation of up to three fields. The field <numberstring-x> is a reference to the segment identifier applicable to the immediately superior level of numbered segment (if any). This identifier is in the form of a character string. The field <separator-y> is a reference to a separator defined at some higher level in the document structure.

The field <number-z> is the number applicable to the given numbered segment whose identifier is being constructed. As indicated above, this number may be determined from an ORDINAL expression or by reference to a binding of the form 'number-<n>' which is specified for the same numbered segment whose identifier is being constructed. In either case, a string function is applied to the number to convert it into a character string. This string function allows the number to be represented in one of the following forms: Arabic number string, upper or lower case Roman numeral string, or upper or lower case alphabetic characters. This construction is defined in the macro USENUMBERSTRINGS.

The constructed binding of the form 'numberstring-<n>' is then available for constructing the identifiers at lower levels of numbered segments. This binding is also referred to in a content generator carried by the constituent Number, which causes the identifier (with optional prefix and suffix strings) to be generated and reproduced when the document is laid out.

NOTE - The macros referred to in this clause are defined in 7.3.1.

6.6.8 Footnote numbering

A footnote number is a character string that identifies a given footnote. The format of this string is as follows:

`<string-literal><num-str><string-literal>`

This format is defined in the macro FNOTENUMBER.

The <string-literal> fields are optional and are pre-defined prefix or suffix character strings. The basic character repertoire used to specify these strings is the primary character repertoire of ISO 8859-1. Any other character

repertoire, and subrepertoire if appropriate, may be used provided that it is designated and invoked by the appropriate designation and innovation sequences, and indicated in the document profile as a non-basic value. No other control functions may be used in these strings.

The field <num-str> is an automatically generated numeral or a user supplied character string that generally serves to identify a particular footnote. Numerals may be represented in the form of Arabic numerals, upper or lower case Roman numerals or upper or lower case Alphabetic characters. Automatically generated footnote numbers are incremented sequentially from an initial value which may be set to any positive value at the beginning of the document and reset at any passage.

A single binding 'fnotenumber' is provide to represent footnote numbers. This may be initialised to any non-negative number at the logical root or on any Passage (see specification of the macro INITIALISEFNOTE).

The footnote number is incremented using a binding expression at each footnote object (see the macro INCFNOTENUMBER). This is then made into a character string using a string function. This value is assigned to the binding 'fnotestring' (see the macro FNOTENUMBERSTRING).

Alternatively, a character string literal may be assigned to the binding 'fnotestring'; this provides the user with the ability to supply particular footnote labels for individual footnotes (see the macro FNOTESTRINGLITERAL).

The constituents FootnoteReference and FootnoteNumber contain content generators whose format is defined by the macro FNOTENUMBER. As indicated above, this format consists of a field represented by <num-str> which has optional prefix and suffix string literals. The field <num-str> consists of a reference to a binding 'fnotestring' which specifies the number of the footnote in the form of a character string.

6.6.9 User readable comments

Information which is to be interpreted as comments relevant to constituents and associated content portions may be specified using the attribute "user readable comments". This information is intended for presentation to humans.

The information consists of a string of characters which shall belong to one of the character repertoires indicated in the document profile attribute "comments character sets" (see 6.7.4.2). If the latter attribute is not explicitly specified, then the default defined in ISO 8613 is assumed. The control functions CR, LF, SP and code extension control functions may also be used within the character string but no other control functions are allowed.

6.6.10 User visible name

Information which may be used to identify constituents within a document may be specified using the attribute "user visible name". This information is intended for presentation to humans, for example, to assist in the editing of documents.

The information consists of a string of characters which shall belong to one of the character repertoires indicated in the document profile attribute "comments character sets" (see 6.7.4.2). If the latter attribute is not explicitly specified, then the default defined in ISO 8613 is assumed. The control functions CR, LF, SP and code extension control functions may also be used within the character string but no other control functions are allowed.

6.7 Document management features

Information relating to the document as a whole is specified in the document profile which is represented by the constituent DocumentProfile. This constituent shall be specified in every document.

The information in the document profile is classified into the following categories:

- i) document constituent information;
- ii) document identification information;
- iii) document default information;
- iv) non-basic characteristics information;
- v) document management information.

The information in the document profile may be of interest to the user or may be used for machine processing of the document.

6.7.1 Document constituent information

This information specifies which constituents are used to represent the document, including constituents that are external to the interchanged document. This information is divided into three categories.

6.7.1.1 Presence of document constituents

This information indicates which constituents are included in the document. That is, this information indicates whether or not the document contains a generic logical structure, a specific logical structure, a generic layout structure, a specific layout structure, layout styles and presentation styles (see note). It is mandatory to specify this information in the document profile.

NOTE - If the generic logical or layout structure is external to the document (see 6.7.1.3), then it is still necessary to indicate that these structures are present and form part of the document.

6.7.1.2 Resource document information

This information consists of a reference to a resource document (see 6.6.1). This is specified by the attribute "resource document". If constituents in the document contain references to object classes in a resource document, then it is mandatory to specify this information in the document profile.

6.7.1.3 External document information

This information consists of a reference to an external document which may consist of a generic logical structure, or both generic layout and generic logical structures (see 6.6.2). If such a reference is required, then it is specified by the attribute "external document class" in the document profile.

6.7.2 Document identification information

This information relates to the identification of the document. This information is divided into six categories.

6.7.2.1 Document application profile information

This information indicates the document application profile to which the document belongs. It is mandatory to specify this information using the attribute "document application profile".

6.7.2.2 Document architecture class information

This information indicates the document architecture class to which the document belongs (see 6.1). It is mandatory to specify this information using the attribute "document architecture class".

6.7.2.3 Content architecture classes information

This information indicates the content architecture classes used in the document (see 6.5.1.2, 6.5.2.2 and 6.5.3). It is mandatory to specify this information using the attribute "content architecture classes".

6.7.2.4 Interchange format class information

This information indicates the interchange format class used to represent the document (see clause 8). It is mandatory to specify this information using the attribute "interchange format class".

6.7.2.5 ODA version information

This information indicates the ISO standard or CCITT Recommendation to which the document conforms. It also specifies a calendar date, which indicates that the document conforms to the version of the ISO standard or CCITT Recommendation and any addenda that are current on that date. It is mandatory to specify this information using the attribute "ODA version".

6.7.2.6 Document reference

This information serves to identify the document. Typically this information is allocated to the document by the creator of the document. The identifier may consist of an ASN.1 object identifier or a string of characters. It is mandatory to specify this information using the attribute "document reference".

6.7.3 Document default information

This information specifies various default values for attributes used in the document. The default values that are allowed are specified in clause 6. The specification of this information is only required when it is required to specify a default value which is other than the standard default value specified in [ISO 8613|T.410 series of CCITT Recommendations].

Default values for the following groups of attributes may be specified:

- document architecture attributes;
- character content attributes;
- raster graphics attributes;
- geometric graphics attributes.

6.7.4 Non-basic characteristics information

This information specifies the non-basic attribute values specified in the document. It is mandatory to specify a non-basic attribute value in the document profile when such a value is used in the document.

The following types of non-basic attribute values may be specified:

- profile character sets;
- comments character sets;
- alternative representation character sets;
- page dimensions;

- medium-types;
- layout paths;
- borders;
- character presentation features;
- raster graphics presentation features;
- raster graphics coding attributes.

Further information concerning document profile, comments and alternative representation character sets is given below.

6.7.4.1 Profile character sets

Some document profile attributes have values consisting of character strings, for example, the document management attributes. The character sets used in these character strings are specified by the document profile attribute "profile character sets".

This attribute "profile character sets" specifies a code extension announcer and designations of character sets, which are subject to the following restrictions:

- i) The code extension announcer shall be 04/03 when specified. This code extension announcer specifies the use of G0 and G1 sets in an 8-bit environment and also the invocation of G0 and G1 sets into GL and GR respectively. Thus, in each attribute to which this attribute applies, invocation shift functions are not necessary, because G0 and G1 sets are implicitly invoked by this code extension announcer.
- ii) G0 set: only ISO-IR6 (the IRV of ISO 646 revised 1991), ISO-IR2 (the primary set of ISO 6937-2), or any other version of ISO 646 may be designated for this set; these graphic character sets are implicitly invoked in GL.
- iii) G1: no restrictions are placed on the graphic character sets that may be designated for this set; these graphic character sets are implicitly invoked in GR.
- iv) The empty set shall be designated into G1 and invoked into GR if no other specific set is invoked in GR.

If the attribute "profile character sets" is not specified, then the default defined in ISO 8613 is assumed.

6.7.4.2 Comments character sets

The character sets assumed to have been designated and optionally invoked at the beginning of the character strings specified by the attributes "user readable comments" (see 6.6.9) and "user visible name" (see 6.6.10) are specified using the document profile attribute "comments character sets".

It also specifies the code extension techniques and the graphic character sets which may be used in the attributes "user readable comments" and "user visible name".

If this attribute is specified, the code extension techniques which may be used in the attributes "user readable comments" and "user visible name" shall be announced by appropriate code extension announcers. The use of G0 set and GL shall always be announced. Other code extension announcers are to be specified according to the requirements of a particular document.

Two kinds of code extension techniques are permitted for this attribute. One is to use GL and GR without shift functions, and the other is to use various character sets by shift functions. The former is rather restricted but no shift functions are necessary in the "user readable comments" and "user visible name".

The same restriction as in 6.7.4.1 is applied in this case. The latter permits various usages of character sets but invocations shall be specified by shift functions in the "user readable comment" and "user visible name".

The same restriction as in 6.5.4 is applied in this case.

All the graphic character sets which may be used in the attribute "user readable comments" and "user visible name" shall be designated in the "comments character sets".

There are no restrictions concerning the number of graphic character sets which are designated and/or invoked in the "comments character sets"; hence designation to the same G set overrides the previous G set.

If the attribute "comments character sets" is not specified, then the default defined in ISO 8613 is assumed.

6.7.4.3 Alternative representation character sets

This attribute specifies the graphic character sets designated and invoked at the beginning of the attribute "alternative representation" other than the standard default graphic character sets.

The restriction on profile character sets described in 6.7.4.1 is also applied. If this attribute is not explicitly specified in the document profile, the default defined in ISO 8613 is assumed.

6.7.5 Fonts list.

This information specifies all the fonts (if any) used in the document. It is specified using the attribute "fonts list". (See ANNEX B.2).

6.7.6 Document management attributes

Document management attributes contain information about the content of the document and its purpose. Information relating to the following may be specified:

- document description (see note);
- dates and times;
- originators;
- other user information;
- external references;
- local file references;
- content attributes;
- security information.

The attributes that may be used to specify this information are defined in [ISO 8613-4|CCITT Recommendation T.414].

The string of characters used in the document management attributes shall belong to the character sets indicated in the document profile attribute "profile character sets" (see 6.7.4.1). If the latter attribute is not explicitly specified in the document profile, then the default character set is the minimum subrepertoire of ISO 6937-2.

The control functions SP, CR and LF may also be used within the character strings but no other control functions are allowed. Hence the graphic character set cannot be changed in the document management attributes.

NOTE - The document description includes the specification of the document reference (see 6.7.2.6).

7 Specification of constituent constraints

This clause specifies the definition of the constituent constraints which may be represented by data streams conforming to this profile.

7.1 Introduction

The structure diagrams illustrating the relationships between the constituents in the logical structures are shown in 7.1.1. The macros indicated on these diagrams are defined in 7.3.1. These macros define the permissible values for the attribute "generator for subordinates" that are applicable to the constituents and, in effect, define the allowed structures that are supported by this profile.

The structure diagrams illustrating the layout structures are shown in 7.1.2. The macros indicated in these diagrams are defined in 7.4.1.

7.1.1.1 Diagrams of relationships of logical constituents

**Figure 13: The 'body' part of the generic logical structure
- the passage and numbered segment levels**

Figure 14: The 'body' part of the generic logical structure - the paragraph level

Figure 15: The 'common' part of the generic logical structure

7.1.2 Diagrams of relationships of layout constituents

Figure 16: The layout structure - document root and page sets

Figure 17: The layout structure - the page structure

Figure 18: The layout structure - the header and footer frame structure

7.1.3 Notation

This clause is written in accordance with the Document Application Profile Proforma and Notation (DAPPN) of ISO 8613-1, Annex F. The following clarifications and minor extensions apply:

1) [Clarification]

The value range definition for the attributes "subordinates" and "imaging order" specify the set of object instances that may occur. The ordering and number (which may be zero) of object instances for the attribute "subordinates" shall be in accordance with the value of the attribute "generator for subordinates" in the respective object class.

2) [Clarification]

The value "ANY_STRING" may include code extension control functions as well as graphic characters.

3) [Extension]

In order to write the specification of the usage of character sets and code extension control functions precisely, the following extensions are applied:

a) Following symbols are introduced to denote shift functions:

Symbol	Shift function	Coded representation
LS0	Locking shift zero	00/15
LS1R	Locking shift one right	ESC 07/14
LS2R	Locking shift two right	ESC 07/13
LS3R	Locking shift three right	ESC 07/12
SS2	Single shift two	08/14
SS3	Single shift three	08/15

b) <escape-sequence> is extended to include shift functions:

```
<escape-sequence>::='ESC'<octet>...[<invocation-control-function>];
<invocation-control-function>::='LS0'|'LS1R'|'LS2R'|'LS3R'|'SS2'|'SS3';
```

c) Data type specification for #ESC in content information is extended as:
<escape-sequence>...

7.2 Document profile constituent constraints

7.2.1 Macro definitions

```

DEFINE(FC, "ASN.1{2 8 2 6 0}" -- formatted character content --)
DEFINE(PC, "ASN.1{2 8 2 6 1}" -- processable character content --)
DEFINE(FPC, "ASN.1{2 8 2 6 2}" -- formatted processable character content --)
DEFINE(FPR, "ASN.1{2 8 2 7 2}" -- formatted processable raster
                                         graphics content --)
DEFINE(FPG, "ASN.1{2 8 2 8 0}" -- formatted processable geometric
                                         graphics content --)

DEFINE(FDA, "{ 'formatted' }")
DEFINE(PDA, "{ 'processable' }")
DEFINE(FPDA, "{ 'formatted-processable' }")
DEFINE(DAC, "DocumentProfile (Document-architecture-class)")

DEFINE(NominalPageSizes, "
    REQ #horizontal-dimension {7015},
    REQ #vertical-dimension {9920}      -- ISO A5 portrait--
|REQ #horizontal-dimension {9920},
    REQ #vertical-dimension {7015}      -- ISO A5 landscape--
|REQ #horizontal-dimension {9920},
    REQ #vertical-dimension {14030}     -- ISO A4 portrait--
|REQ #horizontal-dimension {14030},
    REQ #vertical-dimension {9920}      -- ISO A4 landscape--
|REQ #horizontal-dimension {14030},
    REQ #vertical-dimension {19840}     -- ISO A3 portrait--
|REQ #horizontal-dimension {19840},
    REQ #vertical-dimension {14030}     -- ISO A3 landscape--
|REQ #horizontal-dimension {12141},
    REQ #vertical-dimension {17196}     -- JIS B4 (Japanese legal) portrait--
|REQ #horizontal-dimension {17196},
    REQ #vertical-dimension {12141}     -- JIS B4 (Japanese legal) landscape--
|REQ #horizontal-dimension {8598},
    REQ #vertical-dimension {12141}     -- JIS B5 (Japanese letter) portrait--
|REQ #horizontal-dimension {12141},
    REQ #vertical-dimension {8598}      -- JIS B5 (Japanese letter) landscape--
|REQ #horizontal-dimension {10200},
    REQ #vertical-dimension {16800}     -- ANSI legal portrait--
|REQ #horizontal-dimension {16800},
    REQ #vertical-dimension {10200}     -- ANSI legal landscape--
|REQ #horizontal-dimension {10200},
    REQ #vertical-dimension {13200}     -- ANSI-A portrait--
|REQ #horizontal-dimension {13200},
    REQ #vertical-dimension {10200}     -- ANSI-A landscape--
|REQ #horizontal-dimension {13200},
    REQ #vertical-dimension {20400}     -- ANSI-B portrait--
|REQ #horizontal-dimension {20400},
    REQ #vertical-dimension {13200}     -- ANSI-B landscape--
    ")

```

```

DEFINE(GRAPHICRENDITIONS, "
    {'cancel'|'increased-intensity'
     |'italicised'|'underlined'|'crossed-out'
     |'primary-font'|'first-alternative-font'
     |'second-alternative-font'|'third-alternative-font'
     |'fourth-alternative-font'|'fifth-alternative-font'
     |'sixth-alternative-font'|'seventh-alternative-font'
     |'eighth-alternative-font'|'ninth-alternative-font'
     |'doubly-underlined'|'normal-intensity'
     |'not-italicised'|'not-underlined'|'not-crossed-out'}...
    ")

```

-- Macro defining permissible code extension announcers.
 Note that all the values are basic. --

```

DEFINE(CDEXTEN, " ESC 02/00 05/00,      -- LS0 --
                  [ESC 02/00 05/03],    -- LSR1 --
                  [ESC 02/00 05/05],    -- LSR2 --
                  [ESC 02/00 05/07],    -- LSR3 --
                  [ESC 02/00 05/10],    -- SS2 --
                  [ESC 02/00 05/11]    -- SS3 --
                  ")

```

-- Macro defining code extension announcers for profile default values --

```

DEFINE(DAP-DEFAULT-CDEXTEN, "$CDEXTEN")

```

-- Macros defining final character for designation --

```

DEFINE(FCORE, "04/02 -- A final character designating ISO-IR6
              (the IRV of ISO 646 revised 1991, i.e ASCII) --")

```

```

DEFINE(F646, "-- a final character designating any version of ISO 646
             except, ISO-IR6 --")

```

```

DEFINE(F94S, "-- a final character designating any registered 94 single
             byte graphic character set optionally preceded by one or
             more intermediate characters as defined in Annex C of
             ISO 2022--")

```

```

DEFINE(F94M, "-- a final character designating any registered 94 multi
             byte graphic character set optionally preceded by one or
             more intermediate characters as defined in Annex C of
             ISO 2022--")

```

```

DEFINE(F96S, "-- a final character designating any registered 96 single
             byte graphic character set optionally preceded by one or
             more intermediate characters as defined in Annex C of
             ISO 2022--")

```

```

DEFINE(F96M,    "-- a final character designating any registered 96 multi
                byte graphic character set optionally preceded by one or
                more intermediate characters as defined in Annex C of
                ISO 2022 --")

DEFINE(FEMPTY, "07/14  -- the empty set --")

-- Macros defining a Revision Number of a character set --
DEFINE (REV, "-- An octet between 04/00 and 07/14 which represents a
revision number as defined in ISO2022 --")

-- Macros defining designation sequences --

DEFINE(DEG-CORE-G0, "ESC 02/08 $FCORE")
                -- Designate the 94 characters of ISO-IR6 (the IRV of
                ISO 646 revised 1991) to G0 --
DEFINE(DEG-646-G0, "ESC 02/08 $F646")
                -- Designate any version of ISO 646, except ISO-IR6,
                to G0 --

DEFINE(DEG-ANY-G1, "{[ESC 02/06 REV] {ESC 02/09 $F94S
                |ESC 02/04 02/09 $F94M
                |ESC 02/13 $F96S
                |ESC 02/04 02/13 $F96M]]}")
                -- Designate any character set to G1 --

DEFINE(DEG-ANY-G2, "{[ESC 02/06 REV] {ESC 02/10 $F94S
                |ESC 02/04 02/10 $F94M
                |ESC 02/14 $F96S
                |ESC 02/04 02/14 $F96M]]}")
                -- Designate any character set to G2 --

DEFINE(DEG-ANY-G3, "{[ESC 02/06 REV] {ESC 02/11 $F94S
                |ESC 02/04 02/11 $F94M
                |ESC 02/15 $F96S
                |ESC 02/04 02/15 $F96M]]}")
                -- Designate any character set to G3 --

DEFINE(DEG-EMPTY-G1, "ESC 02/09 $FEMPTY")
                -- Designate the empty set to G1 --

```


-- Macro defining permissible graphic character sets. --

```
DEFINE(PERMIT-GRCHAR, "  {$DEG-CORE-G0 $-LS0
                        |$DEG-646-G0 $-LS0},
                        {{$DEG-ANY-G1 $-LS1R
                        |$DEG-ANY-G2 $-LS2R
                        |$DEG-ANY-G3 $-LS3R}...
                        |$DEG-EMPTY-G1 $-LS1R} ")
```

-- Macro defining graphic character sets for DAP defaults --

```
DEFINE(DAP-DEFAULT-GRCHAR, "$PERMIT-GRCHAR")
```

-- Macro defining basic character sets. Note that this macro is defined for clarification of the specification and is not used in any other part of this DAP specification. --

```
DEFINE(BASIC-GRCHAR, "  $DEG-CORE-G0 $CF-LS0,
                        $DEG-EMPTY-G1 $CF-LS1R ")
```

-- Macro defining non-basic graphic character sets --

```
DEFINE(NON-BASIC-GRCHAR, "  {$DEG-646-G0
                        |$DEG-ANY-G1
                        |$DEG-ANY-G2
                        |$DEG-ANY-G3}... ")
```

-- Macro defining character sets used in document profile attributes --

```
DEFINE(PROFCHAR, "
ESC 02/00 04/03          -- announcement of use of G0 and G1, and
                          invocation into GL and GR respectively, (no shift
                          function are necessary) --
{$DEG-CORE-G0|$DEG-646-G0} -- designate G0 --
{$DEG-ANY-G1|$DEG-EMPTY-G1} -- designate G1 --
```

-- Macro defining comments character sets --

```
DEFINE(COMCHAR, "
-- in the case to use both GL and GR without shift functions --
ESC 02/00 04/03          -- announcement of use of G0 and G1, and
                          invocation into GL and GR respectively. (no shift
                          functions are necessary.) --
{$DEG-CORE-G0|$DEG-646-G0} -- designate G0 --
{$DEG-ANY-G1|$DEG-EMPTY-G1} -- designate G1 --

-- in the case of use of various character sets (shift functions are
necessary) --
{ESC 02/00 05/00,          -- announcement to use G0 and LS0 --
[ESC 02/00 05/03],        -- announcement to use G1 and LS1R --
[ESC 02/00 05/05],        -- announcement to use G2 and LS2R --
[ESC 02/00 05/07],        -- announcement to use G3 and LS3R --
[ESC 02/00 05/10],        -- announcement to use G2 and SS2 --
[ESC 02/00 05/11]    }    -- announcement to use G3 and SS3 --

{$DEG-CORE-G0|$DEG-646-G0} -- designate G0 --
```



```

{{ $DEG-ANY-G1          -- designate G1 --
$DEG-ANY-G2            -- designate G2 --
$DEG-ANY-G3}...       -- designate G3 --
$DEG-EMPTY-G1}

```

-- Macro defining character sets used for alternative representation --

```

DEFINE(ALTCHAR, "$PROFCHAR")

```

7.2.2 Constituent constraints

7.2.2.1 DocumentProfile {

CASE \$DAC OF {

```

    $FDA:  PERM  Generic-layout-structure    {'factor-set'},
           PERM  Specific-layout-structure   {'present'},
           -- shall be present in the case of complete document --
           -- and shall not be present in the case of generic document --
           PERM  Presentation-styles         {'present'}

```

```

    $PDA:  PERM  Generic-layout-structure    {'complete-generator-set'},
           PERM  Generic-logical-structure   {'complete-generator -set
                                             |partial-generator-set'},
           -- shall be present if there is no external document class
           reference --
           PERM  Specific-logical-structure {'present'},
           -- shall be present in the case of complete document --
           -- and shall not be present in the case of generic document --
           PERM  Presentation-styles         {'present'},
           PERM  Layout-styles               {'present'}

```

```

    $FPDA: PERM  Generic-layout-structure    {'complete-generator-set'},
           -- shall be present if there is no external document class
           reference --
           PERM  Specific-layout-structure   {'present'},
           -- shall be present in the case of complete document --
           -- and shall not be present in the case of generic document --
           PERM  Generic-logical-structure   {'complete-generator-set'},
           -- shall be present if there is no external document class
           reference --
           PERM  Specific-logical-structure   {'present'},
           -- shall be present in the case of complete document --
           -- and shall not be present in the case of generic document --
           PERM  Presentation-styles         {'present'},
           PERM  Layout-styles               {'present'}

```

```

    }
    PERM  External-document-class    {ANY_VALUE},

```

```

    PERM  Resource-document          {ANY_VALUE},

```

```

PERM Resources          {MUL{REQ #resource-identifier {ANY_VALUE},
                           REQ #resource-object-class-identifier
                           {ANY_VALUE}}},

```

```

-- document characteristics --

```

```

REQ Document-application-profile  {--see clause 8 for a definition of
                                     the permitted values for this
                                     attribute--},

```

```

PERM Document-application-profile-defaults {

```

```

CASE $DAC OF {
    $FDA :{PERM #content-architecture-class  {$FC|$FPC}}
    $PDA :{PERM #content-architecture-class  {$FC|$PC|$FPC}}
    $FPDA :{PERM #content-architecture-class {$FC|$FPC}}
    },

```

```

PERM #dimensions          {REQ #horizontal-dimension
                             {REQ #fixed-dimension {<=14030}},
                             REQ #vertical-dimension
                             {REQ #fixed-dimension {<=19840}
                              -- up to ISO A3 portrait --
                             |{REQ #horizontal-dimension
                              {REQ #fixed-dimension {<=19840}},
                              REQ #vertical-dimension
                              {REQ #fixed-dimension {<=14030}
                               -- up to ISO A3 landscape --
                              |{REQ #horizontal-dimension
                               {REQ #fixed-dimension {<=13200}},
                               REQ #vertical-dimension
                               {REQ #fixed-dimension {<=20400}
                                -- up to ANSI-B portrait --
                               |{REQ #horizontal-dimension
                                {REQ #fixed-dimension {<=20400}},
                                REQ #vertical-dimension
                                {REQ #fixed-dimension {<=13200}}}
                                -- up to ANSI-B landscape --}},
                             },

```

```

PERM #medium-type          {PERM #nominal-page-size{$NominalPageSizes},
                             PERM #side-of-sheet {ANY_VALUE}},

```

```

PERM #page-position        {ANY_VALUE},

```

```

PERM #layout-path          {'0-degrees'|'180-degrees'|'270-degrees'},

```

```

PERM #type-of-coding       {ASN.1{2 8 3 6 0} -- character encoding --
                             |ASN.1{2 8 3 7 0} -- T.6 encoding --
                             |ASN.1{2 8 3 7 1} -- T.4 one dimensional
                                                    encoding --
                             |ASN.1{2 8 3 7 2} -- T.4 two dimensional
                                                    encoding --

```

```
ASN.1{2 8 3 7 3} -- bitmap encoding --
ASN.1{2 8 3 8 0} -- geometric encoding --},
```

```

PERM #character-content-defaults {
    PERM #alignment                {ANY_VALUE},
    PERM #character-fonts          {ANY_VALUE},
    PERM #character-path           {ANY_VALUE},
    PERM #character-spacing        {ANY_VALUE},
    PERM #character-orientation    {'0-degrees'
                                    |'90-degrees'},
    PERM #code-extension-announcers {$DAP-DEFAULT-CDEXTEN},
    PERM #first-line-offset        {ANY_VALUE},
    PERM #graphic-character-sets   {$DAP-DEFAULT-GRCHAR},
    PERM #graphic-character-subrepertoire {ANY_VALUE},
    PERM #graphic-rendition        {$GRAPHICRENDITIONS},
    PERM #indentation              {ANY_VALUE},
    PERM #initial-offset           {ANY_VALUE},
    PERM #itemisation              {ANY_VALUE},
    PERM #kerning-offset           {ANY_VALUE},
    PERM #line-layout-table        {ANY_VALUE},
    PERM #line-progression         {ANY_VALUE},
    PERM #line-spacing             {ANY_VALUE},
    PERM #orphan-size              {ANY_VALUE},
    PERM #proportional-line-spacing {ANY_VALUE},
    PERM #widow-size               {ANY_VALUE}},

PERM #raster-graphic-content-defaults {
    PERM #clipping                 {ANY_VALUE},
    PERM #image-dimensions         {ANY_VALUE},
    PERM #pel-spacing              {ANY_VALUE},
    PERM #spacing-ratio            {ANY_VALUE},
    PERM #compression              {ANY_VALUE}}},

REQ Document-architecture-class    {$FDA|$PDA|$FPDA},

REQ Content-architecture-classes  {[$FC],[$PC],[$FPC],[$FPR],[$FPG]},

REQ Interchange-format             -- See clause 8 for a definition of
                                   the permitted values for this profile --

REQ Oda-version                    {REQ #standard-or-recommendation{"ISO 8613"},
                                   REQ #publication-date{1992-05-01}},

-- non basic document characteristics --

PERM Profile-character-sets        {$PROFCHAR},

PERM Comments-character-sets       {$COMCHAR},

PERM Alternative-representation-character-sets {$ALTCHAR},
```



```

PERM Page-dimensions {PMUL
    {REQ #horizontal-dimension
        {REQ #fixed-dimension {<=14030}},
    REQ #vertical-dimension
        {REQ #fixed-dimension {12401..19840}}
|{REQ #horizontal-dimension
    {REQ #fixed-dimension {9241..14030}},
    REQ #vertical-dimension
        {REQ #fixed-dimension {<=19840}}
    -- Up to ISO A3 portrait --

|{REQ #horizontal-dimension
    {REQ #fixed-dimension {12401..19840}},
    REQ #vertical-dimension
        {REQ #fixed-dimension {<=14030}}
|{REQ #horizontal-dimension
    {REQ #fixed-dimension {<=19840}},
    REQ #vertical-dimension
        {REQ #fixed-dimension {9241..14030}}
    -- up to ISO A3 landscape --

|{REQ #horizontal-dimension
    {REQ #fixed-dimension {<=13200}},
    REQ #vertical-dimension
        {REQ #fixed-dimension {12401..20400}}
|{REQ #horizontal-dimension
    {REQ #fixed-dimension {9241..13200}},
    REQ #vertical-dimension
        {REQ #fixed-dimension {<=20400}}
    -- up to ANSI-B portrait --

|{REQ #horizontal-dimension
    {REQ #fixed-dimension {12401..20400}},
    REQ #vertical-dimension
        {REQ #fixed-dimension {<=13200}}
|{REQ #horizontal-dimension
    {REQ #fixed-dimension {<=20400}},
    REQ #vertical-dimension
        {REQ #fixed-dimension {9241..13200}} }
    -- up to ANSI-B landscape -- },

-- any value of dimensions which is greater than the common assured
reproduction area of ISO A4 and ANSI-A is non-basic --

PERM Medium-types {PMUL
    {PERM #nominal-page-size{$NominalPageSizes},
    PERM #side-of-sheet{'recto'|'verso'}}},
    -- all values of "medium type" are non-basic --

PERM Layout-paths {{ '0-degrees' | '90-degrees' | '180-degrees' }...},

PERM Borders {ANY_VALUE},

PERM Coding-attributes {

```



```

PERM #raster-graphics-coding-attributes {
  PERM #compression { 'uncompressed' } },

PERM Presentation-features {
  PERM #character-presentation-features {
    PERM #character-orientation { '90-degrees' },
    PMUL { PERM #character-path { '90-degrees'
      | '180-degrees'
      | '270-degrees' } },
    PMUL { PERM #character-spacing { <100|100|160|200 } },
    --only values <100 are required
    to be specified. Values 100, 160
    200 may be specified here for
    upwards compatibility from FOD11--

    PMUL { PERM #graphic-character-sets { $NON-BASIC-GRCHAR } },
    PMUL { PERM #graphic-character-
      subrepertoire { ANY_VALUE } },
    PMUL { PERM #graphic-rendition { 'crossed-out'
      | 'not-crossed-out' } },
    -- values need not be declared. Only permitted for upwards
    compatability from FOD11 --
    PMUL { PERM #line-spacing { ANY_VALUE }
      EXCEPT { 200, 300, 400 } },
    -- value 150 need not be
    declared, the value may be
    specified here for upwards
    compatability from FOD11 --

    PERM #line-progression { '90-degrees' } },

  PERM #raster-graphics-presentation-features {
    PMUL { PERM #pel-spacing { ANY_VALUE }
      EXCEPT { 16, 12, 8, 6, 5, 4, 3, 2, 1 } },

    -- Any value of #pel spaces is permitted as basic --
    -- Basic values of #length are multiples of #pel spaces as listed --

    -- additional document characteristics --

    PERM Fonts-list { PMUL { REQ #font-identifier { ANY_VALUE },
      REQ #font-reference { ANY_VALUE } } },

    -- document management attributes --{

    -- document -description --
    PERM Title { ANY_STRING },
    PERM Subject { ANY_STRING },
    PERM Document-type { ANY_STRING },
    PERM Abstract { ANY_STRING },
    PERM Keywords { ANY_STRING... },
    REQ Document-reference { ANY_VALUE },

```

```

-- dates and times --
PERM Document-date-and-time      {ANY_STRING},
PERM Creation-date-and-time      {ANY_VALUE},
PERM Local-filing-date-and-time  {ANY_STRING},
PERM Expiry-date-and-time        {ANY_STRING},
PERM Start-date-and-time         {ANY_STRING},
PERM Purge-date-and-time         {ANY_STRING},
PERM Release-date-and-time       {ANY_STRING},
PERM Revision-history            {ANY_VALUE},

-- originators --
PERM Organizations               {ANY_STRING...},
PERM Preparers                  {ANY_VALUE},
PERM Owners                     {ANY_VALUE},
PERM Authors                    {ANY_VALUE},

-- other user information --
PERM Copyright                  {ANY_VALUE},
PERM Status                     {ANY_STRING},
PERM User-specific-codes        {ANY_STRING...},
PERM Distribution-list          {ANY_VALUE},
PERM Additional-information      {ANY_VALUE},

-- external references --
PERM References-to-other-documents {ANY_VALUE},
PERM Superseded-documents         {ANY_VALUE},

-- local file references --
PERM Local-file-references        {ANY_VALUE},

-- content attributes --
PERM Document-size               {ANY_INTEGER},
PERM Number-of-pages             {ANY_INTEGER},
PERM Languages                   {ANY_STRING...},

-- security information --
PERM Authorization               {ANY_VALUE},
PERM Security-classification      {ANY_STRING},
PERM Access-rights               {ANY_STRING...}

```

7.3 Logical constituent constraints

7.3.1 Macro definitions

```

DEFINE(DocLogRootGFS, "
<construction-expr>      ::= <construction-term>
                           | <construction-type>;

<construction-term>      ::= <construction-factor>
                           | OPT <construction-factor>
                           | REP <construction-factor>
                           | OPT REP <construction-factor>;

```

```

<construction-type> ::= SEQ({<construction-term>}...)
                        | CHO({<construction-term>}...);

<construction-factor> ::= OBJECT_CLASS_ID_OF(Passage)
                        | <construction-type>;
                        ")

DEFINE(CONSTRAINT-1, "
<constraint-1> ::= <construction-term>
                  | <construction-type>;

<construction-term> ::= <construction-factor>
                        | OPT <construction-factor>
                        | REP <construction-factor>
                        | OPT REP <construction-factor>;

<construction-type> ::= SEQ({<construction-term>}...)
                        | CHO({<construction-term>}...);

<construction-factor> ::= OBJECT_CLASS_ID_OF(Paragraph)
                        | OBJECT_CLASS_ID_OF(BodyText)
                        | OBJECT_CLASS_ID_OF(BodyRaster)
                        | OBJECT_CLASS_ID_OF(BodyGeometric)
                        | <construction-type>;
                        ")

DEFINE(CONSTRAINT-2, "
<constraint-2> ::= OBJECT_CLASS_ID_OF(NumberedSegment)
                        | OPT REP OBJECT_CLASS_ID_OF(NumberedSegment)
                        | REP OBJECT_CLASS_ID_OF(NumberedSegment)
                        | OPT OBJECT_CLASS_ID_OF(NumberedSegment);
                        ")

DEFINE(PassageGFS, "
<construction-expr> ::= <constraint-1>
                        | <constraint-2>
                        | SEQ(<constraint-1><constraint-2>);

$CONSTRAINT-1
$CONSTRAINT-2      ")

DEFINE(NumberedSegmentGFS, "
<construction-expr> ::= SEQ(<constraint-3>[<constraint-1>]
                        [<constraint-2>]);

<constraint-3> ::= OBJECT_CLASS_ID_OF(Number);

$CONSTRAINT-1
$CONSTRAINT-2
                        ")

DEFINE(ParagraphGFS, "
<construction-expr> ::= <construction-term>
                        | <construction-type>;

```



```

<construction-term> ::= <construction-factor>
                        | OPT <construction-factor>
                        | REP <construction-factor>
                        | OPT REP <construction-factor>;

<construction-type> ::= SEQ({<construction-term>}...)
                        | CHO({<construction-term>}...);

<construction-factor> ::= OBJECT_CLASS_ID_OF(BodyText)
                        | OBJECT_CLASS_ID_OF(BodyRaster)
                        | OBJECT_CLASS_ID_OF(BodyGeometric)
                        | OBJECT_CLASS_ID_OF(Footnote)
                        | <construction-type>;

    ")

DEFINE(FootnoteGFS, "
<construction-expr> ::= SEQ(OBJECT_CLASS_ID_OF(FootnoteReference)
                           OBJECT_CLASS_ID_OF(FootnoteBody));

    ")

DEFINE(FootnoteBodyGFS, "
<construction-expr> ::= SEQ(OBJECT_CLASS_ID_OF(FootnoteNumber
                           <constraint-4>);

<constraint-4> ::= OBJECT_CLASS_ID_OF(FootnoteText)
                   | REP(OBJECT_CLASS_ID_OF(FootnoteText))
                   | CHO({OBJECT_CLASS_ID_OF(FootnoteText)}...)
                   | REP CHO({OBJECT_CLASS_ID_OF(FootnoteText)}...);

    ")

DEFINE(CommonContentGFS, "
<construction-expr> ::= <construction-factor>
                        | SEQ(<construction-factor>...);

<construction-factor> ::= OBJECT_CLASS_ID_OF(CommonText)
                        | OBJECT_CLASS_ID_OF(PageNumber)
                        | OBJECT_CLASS_ID_OF(CommonRaster)
                        | OBJECT_CLASS_ID_OF(CommonGeometric);

    ")

DEFINE(N,
    "<n>::={""0""|""1""|""2""|""3""|""4""|""5""|""6""|""7""|""8""|""9""}...;"
    -- any string of characters from the set of characters: '0'...'9' --;

    -- Defines the prefix binding. This binding may be used to associate a string
    literal with an object or object class. In addition, this binding is used to
    prefix text to another binding, such as a segment number, footnote number or
    page number. The instances are differentiated by a suffix number. --

    ")

```



```

DEFINE(PREFIX,      "
<prefix>      ::= ""prefix-""<n>;
$N
      ")

```

-- Defines the suffix binding. This binding may be used to associate a string literal with an object or object class. In addition, this binding is used to suffix text to another binding, such as a segment number, footnote number or page number. The instances are differentiated by a suffix number. --

```

DEFINE(SUFFIX,      "
<suffix>      ::= ""suffix-""<n>;
$N
      ")

```

-- Defines the separator binding. This binding is used to provide a separator character for a hierarchical form of a segment number, footnote number or page number. The instances are differentiated by a suffix number. --

```

DEFINE(SEPARATOR,   "
<separator> ::= "separator-""<n>;
$N
      ")

```

-- Defines the general number binding. This binding may be instantiated for use as the numeric value for use such as in segment number, footnote number or page number bindings. The instances are differentiated by a suffix number. --

```

DEFINE(NUMBER,      "
<number>      ::= ""number-""<n>;
$N
      ")

```

-- Defines the general number string binding. This binding may be instantiated for use as the string value such as for segment number, footnote number or page numbers. The instances are differentiated by a suffix number. --

```

DEFINE(NUMBERSTRING, "
<numberstring> ::= ""numberstring-""<n>;
$N
      ")

```

-- Used to initialise/specify any of the bindings. the bindings defined by this macro are permitted to:

- any logical object class
- any logical object
- any layout object class except frame classes and block classes, and
- any layout object except frames and blocks in the case of FPDA and FDA. --

DEFINE(INITIALISEANY, "

```

REQ #binding-identifer{$PREFIX},
REQ #binding-value{ANY_STRING}
REQ #binding-identifer{$SUFFIX},
REQ #binding-value{ANY_STRING}
REQ #binding-identifer{$SEPARATOR},
REQ #binding-value{ANY_STRING}
REQ #binding-identifer{$NUMBER},
REQ #binding-value{>=0}
REQ #binding-identifer{$NUMBERSTRING},
REQ #binding-value{ANY_STRING}

```

")

-- Used to make a simple or compound string out of the number bindings. --

DEFINE(USENUMBERSTRINGS, "

```

REQ #binding-identifer{$NUMBERSTRING},
REQ #binding-value{<string-expr>::=<hierarchic
-expr>|<simple-expr>;}

```

<hierarchic-expr>

```

::= B_REF(SUP(CURR_OBJ))(<numberstring>)
+B_REF(SUP(CURR_OBJ))(<separator>))
+<simple-expr>;

```

<simple-expr>

```

::= MK-STR(B_REF(CURR-OBJ))(<number>))
|U-ALPHA(B_REF(CURR-OBJ))(<number>))
|L-ALPHA(B_REF(CURR-OBJ))(<number>))
|U-ROM(B_REF(CURR-OBJ))(<number>))
|L-ROM(B_REF(CURR-OBJ))(<number>))
|MK-STR(ORD(CURR-OBJ))
|U-ALPHA(ORD(CURR-OBJ))
|L-ALPHA(ORD(CURR-OBJ))
|U-ROM(ORD(CURR-OBJ))
|L-ROM(ORD(CURR-OBJ));

```

\$SEPARATOR

\$NUMBER

}")

-- Used to increment any of the number bindings. --

DEFINE(USENUMBERS, "

```

REQ #binding-identifer{$NUMBER},
REQ #binding-value
{<num-expr>::=INC(B_REF(PREC(CURR_OBJ))(<number>));}

```

\$NUMBER}

-- This string expression is allowed in a content generator for Number to automatically generate text for segment numbers. --

```

DEFINE(SEGMENTNUMBER, "
<string-expr>          ::= [<pre-st>]<num-st>[<suf-st>];
<num-str>              ::= B_REF(SUP(CURR_OBJ))(<numberstring>);
<pre-st>              ::= B_REF(SUP(CURR_OBJ))(<prefix>)
                        | ANY_STRING;
<suf-st>              ::= B_REF(SUP(CURR_OBJ))(<suffix>)
                        | ANY_STRING;

$NUMBERSTRING
$PREFIXES
$SUFFIXES

")

-- Used to initialise fnotenumber and fnotestring bindings. --

DEFINE(INITIALISEFNOTE, "
    REQ #binding-identifier{"fnotenumber"},
    REQ #binding-value{>=0}
")

-- Used to increment fnotenumber binding. --

DEFINE(INCFNOTENUMBER, "
    REQ #binding-identifier{"fnotenumber"},
    REQ #binding-value{<num-expr>:=INC(B_REF(PREC
                                (CURR-OBJ))("fnotenumber"));}
")

-- Used to create a fnotestring from a fnotenumber binding. --

DEFINE(FNOTENUMBERSTRING, "
    REQ #binding-identifier{"fnotestring"},
    REQ #binding-value{<str-expr>:=
        MK-STR(B_REF(CURR-OBJ)("fnotenumber"))
        | U-ALPHA(B_REF(CURR-OBJ)("fnotenumber"))
        | L-ALPHA(B_REF(CURR-OBJ)("fnotenumber"))
        | U-ROM(B_REF(CURR-OBJ)("fnotenumber"))
        | L-ROM(B_REF(CURR-OBJ)("fnotenumber"));}
")

-- Used to reset the footnote number string to a string literal. This provides
a mechanism for setting the footnote number string to something other than a
numeric value. --

DEFINE(FNOTESTRINGLITERAL, "
    REQ #binding-identifier{"fnotestring"},
    REQ #binding-value{ANY_STRING}
")

```

-- This string expression is allowed in a content generator for FootnoteNumber and FootnoteReference to automatically generate text for a footnote number. --

```

DEFINE(FNOTENUMBER, "
<string-expr>          ::= [ANY_STRING]<num-str>[ANY_STRING];

<num-str>               ::= B_REF(SUP(CURR_OBJ))("fnotestring");
                        ")

DEFINE(PGNUMBERS
<string-expr>          ::= [ANY_STRING]{<num-str>}[ANY_STRING];

<num-str>               ::= MK-STR(<numeric-expr>
                                |U-ALPHA(<numeric-expr>)
                                |L-ALPHA(<numeric-expr>)
                                |U-ROM(<numeric-expr>)
                                |L-ROM(<numeric-expr>);

<numeric-expr>          ::= B_REF(SUP(CURR_INST(<class-or-type1>,
                                (CURR_OBJ))))("PGnum")
                                |B_REF(CURR_INST(<class-or-type2>,
                                CURR_OBJ))("PGnum");

<class-or-type1>        ::= 'frame';

<class-or-type2>        ::= 'page'
                                |OBJECT_CLASS_ID_OF(Page)
                                |OBJECT_CLASS_ID_OF(RectoPage)
                                |OBJECT_CLASS_ID_OF(VersoPage);
                        ")

```

7.3.2 Factor constraints

7.3.2.1 FACTOR ANY-LOGICAL {

GENERIC:

REQ	Object-type	{VIRTUAL},
REQ	Object-class-identifier	{ANY_VALUE}

SPECIFIC:

PERM	Object-type	{VIRTUAL},
REQ	Object-identifier	{ANY_VALUE},
REQ	Object-class	{VIRTUAL}

SPECIFIC_AND_GENERIC:

PERM	User-readable-comments	{ANY_STRING},
PERM	User-visible-name	{ANY_STRING}}

7.3.3 Constituent constraints

7.3.3.1 DocumentLogicalRoot: ANY-LOGICAL {

GENERIC:

REQ	Object-type	{'document-logical-root'},
REQ	Generator-for-subordinates	{\$DocLogRootGFS},
REQ	Application-comments	{REQ #constraint-name {"0"},

SPECIFIC:

```

    PERM  Object-type          {'document-logical-root'},
    REQ   Object-class        {OBJECT_CLASS_ID_OF
                                (DocumentLogicalRoot)},

    REQ   Subordinates        {SUB_ID_OF(Passage)+},
    PERM  Application-comments {REQ #constraint-name {"0"},
                                PERM #external-data {ANY_VALUE}}

```

SPECIFIC_AND_GENERIC:

```

    PERM  Bindings             {PMUL{$INITIALISEANY},
                                PERM{$INITIALISEFNOTE}}

```

7.3.3.2 Passage: ANY-LOGICAL {

GENERIC:

```

    REQ   Object-type          {'composite-logical-object'},
    REQ   Generator-for-subordinates {$PassageGFS},
    REQ   Application-comments    {REQ #constraint-name {"1"},
                                PERM #external-data {ANY_VALUE}}

```

SPECIFIC:

```

    PERM  Object-type          {'composite-logical-object'},
    REQ   Object-class        {OBJECT_CLASS_ID_OF(Passage)},
    REQ   Subordinates        {SUB_ID_OF(NumberedSegment)+,
                                SUB_ID_OF(BodyText)+,
                                SUB_ID_OF(BodyRaster)+,
                                SUB_ID_OF(BodyGeometric)+,
                                SUB_ID_OF(Paragraph)+},
    PERM  Application-comments {REQ #constraint-name {"1"},
                                PERM #external-data {ANY_VALUE}}

```

SPECIFIC_AND_GENERIC:

```

    PERM  Layout-style         {STYLE_ID_OF(L-Style1)},
    PERM  Bindings             {PMUL{$INITIALISEANY},
                                PERM{$INITIALISEFNOTE}}

```

7.3.3.3 NumberedSegment: ANY-LOGICAL {

GENERIC:

```

    REQ   Object-type          {'composite-logical-object'},
    REQ   Generator-for-subordinates {$NumberedSegmentGFS},
    REQ   Application-comments    {REQ #constraint-name {"2"},
                                PERM #external-data {ANY_VALUE}},

    REQ   Bindings             {PMUL{$INITIALISEANY},
                                PERM{$USENUMBERS},
                                REQ{$USENUMBERSTRING}}

```

```

    -- The binding USE NUMBERS shall also be present if --
    -- USENUMBERSTRING does not use the ORD option. --

```

SPECIFIC:

```

    PERM  Object-type          {'composite-logical-object'},
    REQ   Object-class        {OBJECT_CLASS_ID_OF(NumberedSegment)},
    REQ   Subordinates        {SUB_ID_OF(Number),
                                SUB_ID_OF(NumberedSegment)+,
                                SUB_ID_OF(BodyText)+,

```

		SUB_ID_OF(BodyRaster)+,
		SUB_ID_OF(BodyGeometric)+,
		SUB_ID_OF(Paragraph)+,
PERM	Bindings	{PMUL{\$INITIALISEANY}},
PERM	Application-comments	{REQ #constraint-name {"2"},
		PERM #external-data {ANY_VALUE}}
SPECIFIC_AND_GENERIC:		
PERM	Layout-style	{STYLE_ID_OF(L-Style4))}

7.3.3.4 Number: ANY-LOGICAL {

GENERIC:		
REQ	Object-type	{'basic-logical-object'},
REQ	Content-generator	{\$SEGMENTNUMBER},
REQ	Application-comments	{REQ #constraint-name {"3"},
		PERM #external-data {ANY_VALUE}}
SPECIFIC:		
PERM	Object-type	{'basic-logical-object'},
REQ	Object-class	{OBJECT_CLASS_ID_OF(Number)},
PERM	Application-comments	{REQ #constraint-name {"3"},
		PERM #external-data {ANY_VALUE}}
SPECIFIC_AND_GENERIC:		
PERM	Layout-style	{STYLE_ID_OF(L-Style2)},
PERM	Presentation-style	{STYLE_ID_OF(P-Style1)},
PERM	Content-architecture-class	{\$FC \$PC \$FPC}}

7.3.3.5 Paragraph: ANY-LOGICAL {

GENERIC:		
REQ	Object-type	{'composite-logical-object'},
REQ	Generator-for-subordinates	{\$ParagraphGFS},
REQ	Application-comments	{REQ #constraint-name {"6"},
		PERM #external-data {ANY_VALUE}}
SPECIFIC:		
PERM	Object-type	{'composite-logical-object'},
REQ	Object-class	{OBJECT_CLASS_ID_OF
		(Paragraph)},
REQ	Subordinates	{SUB_ID_OF(BodyText)+,
		SUB_ID_OF(Footnote)+,
		SUB_ID_OF(BodyRaster)+,
		SUB_ID_OF(BodyGeometric)+},
PERM	Application-comments	{REQ #constraint-name {"6"},
		PERM #external-data {ANY_VALUE}}
SPECIFIC_AND_GENERIC:		
PERM	Layout-style	{STYLE_ID_OF(L-Style4))}

7.3.3.6 BodyText: ANY-LOGICAL {

GENERIC:

REQ	Object-type	{'basic-logical-object'},
PERM	Resource	{ANY_VALUE},
REQ	Application-comments	{REQ #constraint-name {"14"}, PERM #external-data {ANY_VALUE}}

SPECIFIC:

PERM	Object-type	{'basic-logical-object'},
REQ	Object-class	{OBJECT_CLASS_ID_OF(BodyText)},
PERM	Application-comments	{REQ #constraint-name {"14"}, PERM #external-data {ANY_VALUE}}

SPECIFIC_AND_GENERIC:

PERM	Layout-style	{STYLE_ID_OF(L-Style2)},
PERM	Presentation-style	{STYLE_ID_OF(P-Style1)},
PERM	Content-architecture-class	{\$FC \$PC \$FPC},
PERM	Content-portions	{CONTENT_ID_OF(Character-content-portion)+}}

-- if the attribute "content portions" is specified neither in the specific nor in the generic part then the attribute "resource" shall be specified --

7.3.3.7 BodyGeometric: ANY-LOGICAL {

GENERIC:

REQ	Object-type	{'basic-logical-object'},
REQ	Content-architecture-class	{\$FPG},
PERM	Resource	{ANY_VALUE},
REQ	Application-comments	{REQ #constraint-name {"18"}, PERM #external-data {ANY_VALUE}}

SPECIFIC:

PERM	Object-type	{'basic-logical-object'},
REQ	Object-class	{OBJECT_CLASS_ID_OF(BodyGeometric)},
PERM	Content-architecture-class	{\$FPG},
PERM	Application-comments	{REQ #constraint-name {"18"}, PERM #external-data {ANY_VALUE}}

SPECIFIC_AND_GENERIC:

PERM	Layout-style	{STYLE_ID_OF(L-Style5)},
PERM	Presentation-style	{STYLE_ID_OF(P-Style2)},
PERM	Content-portions	{CONTENT_ID_OF(Geometric-graphics-content-portion)}}}

-- if the attribute "content portions" is specified neither in the specific nor in the generic part then the attribute "resource" shall be specified --

7.3.3.8 BodyRaster: ANY-LOGICAL {

GENERIC:

REQ	Object-type	{'basic-logical-object'},
REQ	Content-architecture-class	{\$FPR},
PERM	Resource	{ANY_VALUE},
REQ	Application-comments	{REQ #constraint-name {"17"}, PERM #external-data {ANY_VALUE}}

SPECIFIC:

PERM	Object-type	{'basic-logical-object'},
REQ	Object-class	{OBJECT_CLASS_ID_OF(BodyRaster)},
PERM	Content-architecture-class	{\$FPR},
PERM	Application-comments	{REQ #constraint-name {"17"}, PERM #external-data {ANY_VALUE}}

SPECIFIC_AND_GENERIC:

PERM	Layout-style	{STYLE_ID_OF(L-Style5)},
PERM	Presentation-style	{STYLE_ID_OF(P-Style3)},
PERM	Content-portions	{CONTENT_ID_OF(Raster-graphics-content-portion))}

-- if the attribute "content portions" is specified neither in the specific nor in the generic part then the attribute "resource" shall be specified --

7.3.3.9 Footnote: ANY-LOGICAL {

GENERIC:

REQ	Object-type	{'composite-logical-object'},
REQ	Generator-for-subordinates	{\$FootnoteGFS},
PERM	Bindings	{{SINCFNOTENUMBER, \$FNOTENUMBERSTRING} \$FNOTESTRINGLITERAL},
REQ	Application-comments	{REQ #constraint-name {"8"}, PERM #external-data {ANY_VALUE}}

SPECIFIC:

PERM	Object-type	{'composite-logical-object'},
REQ	Object-class	{OBJECT_CLASS_ID_OF(Footnote)},
REQ	Subordinates	{SUB_ID_OF(FootnoteReference), SUB_ID_OF(FootnoteBody)},
PERM	Bindings	{\$FNOTESTRINGLITERAL},
CASE	Footnote (GENERIC: Bindings)	OF VOID
{REQ	Bindings	{FNOTESTRINGLITERAL},

PERM	Application-comments	{REQ #constraint-name {"8"}, PERM #external-data {ANY_VALUE}}
------	----------------------	--

SPECIFIC_AND_GENERIC:

PERM	Layout-style	{STYLE_ID_OF(L-Style7)}
------	--------------	-------------------------

7.3.3.10 FootnoteReference: ANY-LOGICAL {

GENERIC:

REQ	Object-type	{'basic-logical-object'},
REQ	Content-generator	{\$FNOTENUMBER},
REQ	Application-comments	{REQ #constraint-name {"10"}, PERM #external-data {ANY_VALUE}}

SPECIFIC:

PERM	Object-type	{'basic-logical-object'},
REQ	Object-class	{OBJECT_CLASS_ID_OF (FootnoteReference)},
PERM	Application-comments	{REQ #constraint-name {"10"}, PERM #external-data {ANY_VALUE}}

SPECIFIC_AND_GENERIC:

PERM	Layout-style	{STYLE_ID_OF(L-Style2)},
PERM	Presentation-style	{STYLE_ID_OF(P-Style1)},
PERM	Content-architecture-class	{\$FC \$PC \$FPC}}

7.3.3.11 FootnoteBody: ANY-LOGICAL {

GENERIC:

REQ	Object-type	{'composite-logical-object'},
REQ	Generator-for-subordinates	{\$FootnoteBodyGFS},
REQ	Application-comments	{REQ #constraint-name {"11"}, PERM #external-data {ANY_VALUE}}
PERM	Layout-style	{STYLE_ID_OF (L-Style11)}

SPECIFIC:

PERM	Object-type	{'composite-logical-object'},
REQ	Object-class	{OBJECT_CLASS_ID_OF(FootnoteBody)},
REQ	Subordinates	{SUB_ID_OF(FootnoteNumber, SUB_ID_OF(FootnoteText)+},
PERM	Application-comments	{REQ #constraint-name {"11"}, PERM #external-data {ANY_VALUE}}
PERM	Layout-style	{STYLE_ID_OF (L-Style11)}
	}	

7.3.3.12 FootnoteNumber: ANY-LOGICAL {

GENERIC:

REQ	Object-type	{'basic-logical-object'},
REQ	Content-generator	{\$FNOTENUMBER},
REQ	Application-comments	{REQ #constraint-name {"9"}, PERM #external-data {ANY_VALUE}}
REQ	Layout-style	{STYLE_ID_OF(L-Style9)}

SPECIFIC:

PERM	Object-type	{'basic-logical-object'},
REQ	Object-class	{OBJECT_CLASS_ID_OF(FootnoteNumber)},
PERM	Application-comments	{REQ #constraint-name {"9"}, PERM #external-data {ANY_VALUE}},
PERM	Layout-style	{STYLE_ID_OF(L-Style9)}

SPECIFIC_AND_GENERIC:

PERM Presentation-style {STYLE_ID_OF(P-Style1)},
 PERM Content-architecture-class {\$FC|\$PC|\$FPC}}

7.3.3.13 FootnoteText: ANY-LOGICAL {

GENERIC:

REQ Object-type {'basic-logical-object'},
 REQ Application-comments {REQ #constraint-name {"12"},
 PERM #external-data {ANY_VALUE}},
 REQ Layout-style {STYLE_ID_OF(L-Style6)}

SPECIFIC:

PERM Object-type {'basic-logical-object'},
 REQ Object-class {OBJECT_CLASS_ID_OF(FootnoteText)},
 REQ Content-portions {CONTENT_ID_OF(
 Character-content-portion)+)},
 PERM Application-comments {REQ #constraint-name {"12"},
 PERM #external-data {ANY_VALUE}},
 PERM Layout-style {STYLE_ID_OF(L-Style6)}

SPECIFIC_AND_GENERIC:

PERM Presentation-style {STYLE_ID_OF(P-Style1)},
 PERM Content-architecture-class {\$FC|\$PC|\$FPC}}

7.3.3.14 CommonContent {

GENERIC:

REQ Object-type {'composite-logical-object'},
 REQ Object-class-identifier {ANY_VALUE},
 REQ Generator-for-subordinates {\$CommonContentGFS},
 REQ Application-comments {REQ #constraint-name {"19"},
 PERM #external-data {ANY_VALUE}},
 PERM User-readable-comments {ANY_STRING},
 PERM User-visible-name {ANY_STRING}}

7.3.3.15 CommonText {

GENERIC:

REQ Object-type {'basic-logical-object'},
 REQ Object-class-identifier {ANY_VALUE},
 PERM Content-portion {CONTENT_ID_OF(
 Character content-portion)+)},
 PERM Resource {ANY_VALUE},
 PERM Layout-style {STYLE_ID_OF(L-Style3)},
 PERM Presentation-style {STYLE_ID_OF(P-Style4)},
 PERM Content-architecture-class {\$FC|\$PC|\$FPC},
 REQ Application-comments {REQ #constraint-name {"20"},
 PERM #external-data {ANY_VALUE}},
 PERM User-readable-comments {ANY_STRING},
 PERM User-visible-name {ANY_STRING}}

-- either the attribute "content portions" or "resource" shall be specified
 in the above constituent --

7.3.3.16 PageNumber {

GENERIC:

REQ	Object-type	{'basic-logical-object'},
REQ	Object-class-identifier	{ANY_VALUE},
REQ	Content-generator	{\$PGNUMBERS},
PERM	Layout-style	{STYLE_ID_OF(L-Style3)},
PERM	Presentation-style	{STYLE_ID_OF(P-Style4)},
PERM	Content-architecture-class	{\$FC \$PC \$FPC},
REQ	Application-comments	{REQ #constraint-name {"40"}, PERM #external-data {ANY_VALUE}},
PERM	User-readable-comments	{ANY_STRING},
PERM	User-visible-name	{ANY_STRING}}

7.3.3.17 CommonGeometric {

GENERIC:

REQ	Object-type	{'basic-logical-object'},
REQ	Object-class-identifier	{ANY_VALUE},
PERM	Content-portions	{CONTENT_ID_OF(Geometric-graphics-content-portion))},
PERM	Resource	{ANY_VALUE},
PERM	Layout-style	{STYLE_ID_OF(L-Style8)},
PERM	Presentation-style	{STYLE_ID_OF(P-Style2)},
REQ	Content-architecture-class	{\$FPG},
REQ	Application-comments	{REQ #constraint-name {"22"}, PERM #external-data {ANY_VALUE}},
PERM	User-readable-comments	{ANY_STRING},
PERM	User-visible-name	{ANY_STRING}}

-- either the attribute "content portions" or "resource" shall be specified in the above constituent --

7.3.3.18 CommonRaster {

GENERIC

REQ	Object-type	{'basic-logical-object'},
REQ	Object-class-identifier	{ANY_VALUE},
PERM	Content-portions	{CONTENT_ID_OF(Raster-graphics-content-portion))},
PERM	Resource	{ANY_VALUE},
PERM	Layout-style	{STYLE_ID_OF(L-Style8)},
PERM	Presentation-style	{STYLE_ID_OF(P-Style3)},
REQ	Content-architecture-class	{\$FPR},
REQ	Application-comments	{REQ #constraint-name {"21"}, PERM #external-data {ANY_VALUE}},
PERM	User-readable-comments	{ANY_STRING},
PERM	User-visible-name	{ANY_STRING}}

-- either the attribute "content portions" or "resource" shall be specified in the above constituent --

7.4 Layout constituent constraints

7.4.1 Macro definitions

```

DEFINE(DocLayRootGFS, "
<construction-expr> ::= <construction-term>
                        | <construction-type>;

<construction-term> ::= <construction-factor>
                        | OPT <construction-factor>
                        | REP <construction-factor>
                        | OPT REP <construction-factor>;

<construction-type> ::= SEQ({<construction-term>}...)
                        | CHO({<construction-term>}...);

<construction-factor> ::= OBJECT_CLASS_ID_OF(PageSet)
                        | <construction-type>;
")

DEFINE(PageSetGFS, "
<construction-expr> ::= <PageSet-1>
                        | <PageSet-2>
                        | <PageSet-3>
                        | <SEQ(<PageSet-1><PageSet-2>)
                        | <SEQ(<PageSet-1><PageSet-3>);

<PageSet-1> ::= OBJECT_CLASS_ID_OF(Page)
                | OPT(OBJECT_CLASS_ID_OF(Page));

<PageSet-2> ::= REP(OBJECT_CLASS_ID_OF(Page))
                | OPT REP(OBJECT_CLASS_ID_OF(Page));

<PageSet-3> ::= OPT REP(SEQ(OBJECT_CLASS_ID_OF(RectoPage)
                            OPT(OBJECT_CLASS_ID_OF(VersoPage))))
                | OPT REP(SEQ(OBJECT_CLASS_ID_OF(VersoPage)
                            OPT(OBJECT_CLASS_ID_OF(RectoPage))))
                | REP(SEQ(OBJECT_CLASS_ID_OF(RectoPage)
                            OPT(OBJECT_CLASS_ID_OF(VersoPage))))
                | REP(SEQ(OBJECT_CLASS_ID_OF(VersoPage)
                            OPT(OBJECT_CLASS_ID_OF(RectoPage))));
")

DEFINE(PageGFS, "
<construction-expr> ::= SEQ([<headerarea>]<bodyarea>[<footerarea>])
                        | <bodyarea>;

<headerarea> ::= OBJECT_CLASS_ID_OF(BasicHeader)
                | OBJECT_CLASS_ID_OF(CompositeHeader);

<bodyarea> ::= OBJECT_CLASS_ID_OF(BasicBody)
                | OBJECT_CLASS_ID_OF(VariableCompositeBody);

```



```

<footerarea> ::= OBJECT_CLASS_ID_OF(BasicFooter)
               | OBJECT_CLASS_ID_OF(CompositeFooter);
               ")

DEFINE(VariableCompositeBodyGFS, "
<construction-expr> ::= <construction-term>
                       | <construction-type>
                       | SEQ(<construction-term>, <construction-footnote>)
                       | SEQ(<construction-type>, <construction-footnote>);

<construction-term> ::= <construction-factor>
                       | OPT <construction-factor>
                       | REP <construction-factor>
                       | OPT REP <construction-factor>;

<construction-type> ::= SEQ({<construction-term>}...)
                       | CHO({<construction-term>}...);

<construction-factor> ::= OBJECT_CLASS_ID_OF(BasicFloat),
                          | OBJECT_CLASS_ID_OF(SnakingColumns)
                          | OBJECT_CLASS_ID_OF(SynchronizedColumns)
                          | <construction-type>;

<construction-footnote> ::= OBJECT_CLASS_ID_OF(FootnoteArea)
                          | OPT OBJECT_CLASS_ID_OF(FootnoteArea);
                          ")

DEFINE(SnakingColumnsGFS, "
<construction-expr> ::= SEQ({OBJECT_CLASS_ID_OF(ColumnVariable)}...)
                       | REP OBJECT_CLASS_ID_OF(columnVariable);
                       ")

DEFINE(SynchronizedColumnsGFS, "
<construction-expr> ::= SEQ({OBJECT_CLASS_ID_OF(ColumnFixed)}...);

DEFINE(HeaderFooterGFS, "
<construction-expr> ::= <fixed-common-content-frames>
                       | <variable-common-content-frames>;

<fixed-common-content-frames>
    ::= SEQ({OBJECT_CLASS_ID_OF(SourcedContentFixed)
            | OBJECT_CLASS_ID_OF(ArrangedContentFixed)}...);

<variable-common-content-frames>
    ::= SEQ({OBJECT_CLASS_ID_OF(SourcedContentVariable)
            | OBJECT_CLASS_ID_OF(ArrangedContentVariable)}...);
    ")

DEFINE(PAGENUMBER, "
    REQ #binding-identifier{"PGnum"},
    REQ #binding-value{<num-expr>::=INC(B_REF(PREC(
                                                CURR-OBJ))("PGnum"))};
    ")

```

```

DEFINE(INITIALISEPGNUM, "
    {REQ #binding-identifier{"PGnum"},
    REQ #binding-value{>=-1}}
    ")

DEFINE(PDA-FPDA, "{ 'processable' | 'formatted-processable' }")

```

7.4.2 Factor constraints

7.4.2.1 FACTOR ANY-LAYOUT {

```

GENERIC:
    REQ    Object-type                {VIRTUAL},
    REQ    Object-class-identifier    {ANY_VALUE},
    REQ    Application-comments       {VIRTUAL}
SPECIFIC:
    PERM    Object-type                {VIRTUAL},
    REQ    Object-identifier           {ANY_VALUE},
    CASE $DAC OF {
        $FDA: PERM    Object-class    {VIRTUAL}
        $FPDA: REQ    Object-class    {VIRTUAL}
    },
    REQ    Subordinates                {VIRTUAL},
    PERM    Application-comments       {VIRTUAL}
SPECIFIC_AND_GENERIC:
    PERM    User-readable-comments    {ANY_STRING},
    PERM    User-visible-name         {ANY_STRING}

```

7.4.2.2 FACTOR ANY-PAGE: ANY-LAYOUT {

```

GENERIC:
    REQ    Object-type                {'page'},
    CASE $DAC OF {
        $PDA-FPDA:
    REQ    Generator-for-subordinates {$PageGFS},
    PERM    Bindings                  {$PAGENUMBER}
    },
SPECIFIC:
    PERM    Object-type                {'page'},
    REQ    Subordinates                {SUB_ID_OF(BasicHeader),
    SUB_ID_OF(CompositeHeader),
    SUB_ID_OF(BasicBody),
    SUB_ID_OF(VariableCompositeBody),
    SUB_ID_OF(BasicFooter),
    SUB_ID_OF(CompositeFooter)}

```

SPECIFIC AND GENERIC:

PERM Dimensions

```

{{REQ #horizontal-dimension
  {REQ #fixed-dimension {<=14030}},
  REQ #vertical-dimension
    {REQ #fixed-dimension {<=19840}}}}
-- up to ISO A3 portrait --
|{REQ #horizontal-dimension
  {REQ #fixed-dimension {<=19840}},
  REQ #vertical-dimension
    {REQ #fixed-dimension {<=14030}}},
-- up to ISO A3 landscape --
|{REQ #horizontal-dimension
  {REQ #fixed-dimension {<=13200}},
  REQ #vertical-dimension
    {REQ #fixed-dimension {<=20400}}}}
-- up to ANSI-B portrait --
|{REQ #horizontal-dimension
  {REQ #fixed-dimension {<=20400}},
  REQ #vertical-dimension
    {REQ #fixed-dimension {<=13200}}},
-- up to ANSI-B landscape --,
{ANY_VALUE}}

```

PERM Page-position

```

{ANY_VALUE}}

```

7.4.2.3 FACTOR ANY-FRAME-FIXED: ANY-LAYOUT {

GENERIC:

```

REQ Object-type {'frame'}

```

SPECIFIC:

```

PERM Object-type {'frame'}

```

SPECIFIC AND GENERIC:

```

PERM Position {REQ #fixed-position
  {REQ #horizontal-position {ANY_VALUE},
  REQ #vertical-position {ANY_VALUE}}},
PERM Dimensions {REQ #horizontal-dimension
  {REQ #fixed-dimension {ANY_VALUE}},
  REQ #vertical-dimension
    {REQ #fixed-dimension {ANY_VALUE}}},
PERM Border {ANY_VALUE}}

```

7.4.2.4 FACTOR ANY-FRAME-VARIABLE: ANY-LAYOUT {

GENERIC:

```

REQ Object-type {'frame'}

```

SPECIFIC:

```

PERM Object-type {'frame'},

```

```

CASE $DAC OF {

```

```

  $FPDA: REQ Position {REQ #fixed-position
    {REQ #horizontal-position {ANY_VALUE},
    REQ #vertical-position {ANY_VALUE}}},

```

```

        REQ    Dimensions    {REQ #horizontal-dimension
                                {REQ #fixed-dimension {ANY_VALUE}},
                                REQ #vertical-dimension
                                {REQ #fixed-dimension {ANY_VALUE}}}
    }
SPECIFIC_AND_GENERIC:
CASE    $DAC OF {
    $FDA:  PERM    Position    {REQ #fixed-position
                                {REQ #horizontal-position {ANY_VALUE},
                                REQ #vertical-position {ANY_VALUE}}},
        PERM    Dimensions    {REQ #horizontal-dimension
                                {REQ #fixed-dimension {ANY_VALUE}},
                                REQ #vertical-dimension
                                {REQ #fixed-dimension {ANY_VALUE}}}
    },
    PERM    Border            {ANY_VALUE}

```

7.4.2.5 FACTOR BLOCK {

SPECIFIC:

```

REQ    Object-type            {'block'},
REQ    Object-identifier      {ANY_VALUE},
PERM   Content-architecture-class {$FC|$FPC|$FPR|$FPG},
PERM   Presentation-attributes {
    PERM #character-attributes {
        PERM #alignment          {ANY_VALUE},
        PERM #character-fonts    {ANY_VALUE},
        PERM #character-spacing  {ANY_VALUE},
        PERM #character-orientation {'0-degrees'
                                     |'90-degrees'},
        PERM #character-path      {'0-degrees'
                                     |'90-degrees'
                                     |'180-degrees'
                                     |'270-degrees'},
        PERM #code-extension-announcers {$CDEXTAN},
        PERM #first-line-offset    {ANY_VALUE},
        PERM #graphic-character-sets {$PERMIT-GRCHAR},
        PERM #graphic-character-subrepertoire
                                     {ANY_VALUE},
        PERM #graphic-rendition    {$GRAPHICRENDITIONS},
        PERM #itemisation          {ANY_VALUE},
        PERM #kerning-offset       {ANY_VALUE},
        PERM #line-layout-table    {ANY_VALUE},
        PERM #line-progression     {'90 degrees'|'270-degrees'},
        PERM #line-spacing         {ANY_VALUE},
        PERM #initial-offset       {ANY_VALUE}}}

```



```

PERM  User-readable-comments      {ANY_STRING},
PERM  User-visible-name           {ANY_STRING},
PERM  Position                    {REQ #fixed-position
                                   {REQ #horizontal-position {ANY_VALUE},
                                   REQ #vertical-position {ANY_VALUE}}},
PERM  Dimensions                  {REQ #horizontal-dimension
                                   {REQ #fixed-dimension {ANY_VALUE}},
                                   REQ #vertical-dimension
                                   {REQ #fixed-dimension {ANY_VALUE}}}}

```

7.4.3 Constituent constraints

7.4.3.1 DocumentLayoutRoot: ANY-LAYOUT {

GENERIC:

```

REQ   Object-type                 {'document-layout-root'},
CASE  $DAC OF {
    $PDA-FPDA:
REQ   Generator-for-subordinates  {$DocLayoutRootGFS},
PERM  Bindings                    {$INITIALISEPGNUM}
    },
REQ   Application-comments        {REQ #constraint-name {"0"},
    PERM #external-data {ANY_VALUE}}

```

SPECIFIC:

```

PERM  Object-type                 {'document-layout-root'},
CASE  $DAC OF {
    $FDA:  PERM  Object-class {OBJECT_CLASS_ID_OF
                                (DocumentLayoutRoot)}
    $FPDA: REQ   Object-class {OBJECT_CLASS_ID_OF
                                (DocumentLayoutRoot)}
    },
REQ   Subordinates                {SUB_ID_OF(PageSet)+},
PERM  Application-comments        {REQ #constraint-name {"0"},
    PERM #external-data {ANY_VALUE}}

```

7.4.3.2 PageSet: ANY-LAYOUT {

GENERIC:

```

REQ   Object-type                 {'page-set'},
CASE  $DAC OF {
    $PDA-FPDA:
REQ   Generator-for-subordinates  {$PageSetGFS},
PERM  Bindings                    {$INITIALISEPGNUM}
    },
REQ   Application-comments        {REQ #constraint-name {"1"},
    PERM #external-data {ANY_VALUE}}

```

SPECIFIC:

```

    PERM    Object-type                {'page-set'}
    CASE    $DAC OF {
        $FDA:  PERM  Object-class    {OBJECT_CLASS_ID_OF(PageSet)},
        $FPDA: REQ   Object-class    {OBJECT_CLASS_ID_OF(PageSet)}
    },
    REQ      Subordinates                {SUB_ID_OF(Page)+,
                                         SUB_ID_OF(RectoPage)+,
                                         SUB_ID_OF(VersoPage)+},
    PERM      Application-comments        {REQ #constraint-name {"1"},
                                         PERM #external-data {ANY_VALUE}}

```

7.4.3.3 Page: ANY-PAGE {

GENERIC:

```

    REQ      Application-comments        {REQ #constraint-name {"2"},
                                         PERM #external-data {ANY_VALUE}}

```

SPECIFIC:

```

    CASE    $DAC OF {
        $FDA:  PERM  Object-class    {OBJECT_CLASS_ID_OF(Page)}
        $FPDA: REQ   Object-class    {OBJECT_CLASS_ID_OF(Page)}
    },
    PERM      Application-comments        {REQ #constraint-name {"2"},
                                         PERM #external-data {ANY_VALUE}}

```

SPECIFIC_AND_GENERIC:

```

    PERM      Medium-type                {PERM #nominal-page-size
                                         {$NominalPageSizes},
                                         PERM #side-of-sheet {ANY_VALUE}}

```

7.4.3.4 RectoPage: ANY-PAGE {

GENERIC:

```

    REQ      Application-comments        {REQ #constraint-name {"3"},
                                         PERM #external-data {ANY_VALUE}},
    REQ      Medium-type                {PERM #nominal-page-size
                                         {$NominalPageSizes},
                                         REQ #side-of-sheet
                                         {'recto'|'unspecified'}}

```

SPECIFIC:

```

    CASE    $DAC OF {
        $FDA:  PERM  Object-class    {OBJECT_CLASS_ID_OF(RectoPage)}
        $FPDA: REQ   Object-class    {OBJECT_CLASS_ID_OF(RectoPage)}
    },
    PERM      Application-comments        {REQ #constraint-name {"3"},
                                         PERM #external-data {ANY_VALUE}},
    PERM      Medium-type                {PERM #nominal-page-size
                                         {$NominalPageSizes},
                                         PERM #side-of-sheet
                                         {'recto'|'unspecified'}}

```

7.4.3.5 VersoPage: ANY-PAGE {

GENERIC:

```

REQ    Application-comments    {REQ #constraint-name {"4"},
                                PERM #external-data {ANY_VALUE}},
REQ    Medium-type             {PERM #nominal-page-size
                                {$NominalPageSizes},
                                REQ #side-of-sheet
                                {'verso'|'unspecified'}}

```

SPECIFIC:

```

CASE    $DAC OF {
        $FDA: PERM Object-class {OBJECT_CLASS_ID_OF(VersoPage)}
        $FPDA: REQ Object-class {OBJECT_CLASS_ID_OF(VersoPage)}
    },
PERM    Application-comments    {REQ #constraint-name {"4"},
                                PERM #external-data {ANY_VALUE}},
PERM    Medium-type             {PERM #nominal-page-size
                                {$NominalPageSizes},
                                PERM #side-of-sheet
                                {'verso'|'unspecified'}}}

```

7.4.3.6 BasicBody: ANY-FRAME-FIXED {

GENERIC:

```

PERM    Layout-path            {'270-degrees' -- page layout A --
                                |'0-degrees'   -- page layout B --
                                |'180-degrees' -- page layouts
                                C and D --},
REQ     Application-comments     {REQ #constraint-name {"28"},
                                PERM #external-data {ANY_VALUE}}

```

SPECIFIC:

```

CASE    $DAC OF {
        $FDA: PERM Object-class {OBJECT_CLASS_ID_OF(BasicBody)}
        $FPDA: REQ Object-class {OBJECT_CLASS_ID_OF(BasicBody)}
    },
REQ     Subordinates             {SUB_ID_OF(SpecificBlock)+},
PERM    Application-comments     {REQ #constraint-name {"28"},
                                PERM #external-data {ANY_VALUE}}

```

7.4.3.7 VariableCompositeBody: ANY-FRAME-FIXED {

GENERIC:

```

CASE $DAC OF {
    $PDA-FPDA:
        REQ Generator-for-subordinates
                                {$VariableCompositeBodyGFS},
        PERM Layout-path
                                {'270-degrees' -- page layout A --
                                |'0-degrees' -- page layout B --
                                |'180-degrees' -- page layouts
                                | C and D --}
    },
    REQ Application-comments
                                {REQ #constraint-name {"7"},
                                PERM #external-data {ANY_VALUE}}

```

SPECIFIC:

```

CASE $DAC OF {
    $FDA: PERM Object-class
                                {OBJECT_CLASS_ID_OF
                                (VariableCompositeBody)},
    $FPDA: REQ Object-class
                                {OBJECT_CLASS_ID_OF
                                (VariableCompositeBody)}
    },
    REQ Subordinates
                                {SUB_ID_OF(BasicFloat)+,
                                SUB_ID_OF(SnakingColumns)+,
                                SUB_ID_OF(SynchronizedColumns)+,
                                SUB_ID_OF(FootnoteArea)},
    PERM Application-comments
                                {REQ #constraint-name {"7"},
                                PERM #external-data {ANY_VALUE}}

```

7.4.3.8 BasicFloat: ANY-FRAME-VARIABLE {

GENERIC:

```

CASE $DAC OF {
    $PDA-FPDA:
        REQ Position
                                {REQ #variable-position {
                                PERM #offset {ANY_VALUE},
                                PERM #separation {ANY_VALUE},
                                PERM #alignment {ANY_VALUE},
                                PERM #fill-order {'normal-order'}}},
        PERM Permitted-categories {ANY_STRING}
    CASE SUPERIOR (VariableCompositeBody(Layout-path)) OF {

```



```

{'270-degrees'}: -- page layout A --
    REQ Dimensions      {REQ #horizontal-dimension
                        {REQ #fixed-dimension {ANY_VALUE},
                        |REQ #maximum-size {'applies'}},
                        REQ #vertical-dimension
                        {REQ #rule-b {ANY_VALUE}}},
    PERM Layout-path    {'270-degrees'}
{'0-degrees'}: -- page layout B --
    REQ Dimensions      {REQ #horizontal-dimension
                        {REQ #rule-b {ANY_VALUE}},
                        REQ #vertical-dimension
                        {REQ #fixed-dimension {ANY_VALUE}
                        |REQ #maximum-size {'applies'}}},
    REQ Layout-path     {'0-degrees'}

{'180-degrees'}: -- page layouts C and D --
    REQ Dimensions      {REQ #horizontal-dimension
                        {REQ #rule-b {ANY_VALUE}},
                        REQ #vertical-dimension
                        {REQ #fixed-dimension {ANY_VALUE}
                        |REQ #maximum-size {'applies'}}},
    REQ Layout-path     {'180-degrees'}
    },
REQ Application-comments {REQ #constraint-name {"12"},
                        PERM #external-data {ANY_VALUE}}
SPECIFIC:
CASE $DAC OF {
    $FDA: PERM Object-class {OBJECT_CLASS_ID_OF(BasicFloat)},
    $FPDA: REQ Object-class {OBJECT_CLASS_ID_OF(BasicFloat)}
    },
REQ Subordinates {SUB_ID_OF(SpecificBlock)+},
PERM Application-comments {REQ #constraint-name {"12"},
                        PERM #external-data {ANY_VALUE}}

```

7.4.3.9 SynchronizedColumns: ANY-FRAME-VARIABLE {

```

GENERIC:
CASE $DAC OF {
    $PDA-FPDA:
    REQ Generator-for-subordinates
                        {$SynchronizedColumnsGFS},
    REQ Position      {REQ #variable-position {
                        PERM #offset {ANY_VALUE},
                        PERM #separation {ANY_VALUE},
                        PERM #alignment {ANY_VALUE},
                        PERM #fill-order {'normal-order'}}
    CASE SUPERIOR (VariableCompositeBody(Layout-path)) OF {

```

```
{'270-degrees'}: -- page layout A --
    REQ Dimensions {REQ #horizontal-dimension
                    {REQ #fixed-dimension {ANY_VALUE}
                    |REQ #maximum-size {'applies'}},
                    REQ #vertical-dimension
                    {REQ #rule-b {ANY_VALUE}}},
    PERM Layout-path {'270-degrees'}
```

```
{'0-degrees'}: -- page layout B --
    REQ Dimensions {REQ #horizontal-dimension
                    {REQ #rule-b {ANY_VALUE}},
                    REQ #vertical-dimension
                    {REQ #fixed-dimension {ANY_VALUE}
                    |REQ #maximum-size {'applies'}}},
    REQ Layout-path {'0-degrees'}
```

```
{'180-degrees'}: -- page layouts C and D --
    REQ Dimensions {REQ #horizontal-dimension
                    {REQ #rule-b {ANY_VALUE}},
                    REQ #vertical-dimension
                    {REQ #fixed-dimension {ANY_VALUE}
                    |REQ #maximum-size {'applies'}}},
    REQ Layout-path {'180-degrees'}
```

```
    },
    REQ Application-comments {REQ #constraint-name {"11"},
                              PERM #external-data {ANY_VALUE}}
```

SPECIFIC:

```
    CASE $DAC OF {
        $FDA: PERM Object-class {OBJECT_CLASS_ID_OF
                                  (SynchronizedColumns))
        $FPDA: REQ Object-class {OBJECT_CLASS_ID_OF
                                  (SynchronizedColumns))
    },
    REQ Subordinates {SUB_ID_OF(ColumnFixed)+},
    PERM Application-comments {REQ #constraint-name {"11"},
                              PERM #external-data {ANY_VALUE}}}
```

7.4.3.10 SnakingColumns: ANY-FRAME-VARIABLE {

GENERIC:

```
    CASE $DAC OF {
        $PDA-FPDA:
        REQ Generator-for-subordinates {$SnakingColumnsGFS},
        REQ Position {REQ #variable-position {
                      PERM #offset {ANY_VALUE},
                      PERM #separation {ANY_VALUE},
                      PERM #alignment {ANY_VALUE},
                      PERM #fill-order {'normal-order'}},
        PERM Balance {ANY_VALUE}
        CASE SUPERIOR (VariableCompositeBody(Layout-path)) OF {
```

{'270-degrees'}: -- page layout A --

```

REQ Dimensions {REQ #horizontal-dimension
                {REQ #fixed-dimension {ANY_VALUE}
                |REQ #maximum-size {'applies'}},
                REQ #vertical-dimension
                {REQ #rule-b {ANY_VALUE}}},
REQ Layout-path {'0-degrees'|'180-degrees'}
```

{'0-degrees'}: -- page layout B --

```

REQ Dimensions {REQ #horizontal-dimension
                {REQ #rule-b {ANY_VALUE}},
                REQ #vertical-dimension
                {REQ #fixed-dimension {ANY_VALUE}
                |REQ #maximum-size {'applies'}}},
PERM Layout-path {'90-degrees'|'270-degrees'}
```

{'180-degrees'}: -- page layouts C and D --

```

REQ Dimensions {REQ #horizontal-dimension
                {REQ #rule-b {ANY_VALUE}},
                REQ #vertical-dimension
                {REQ #fixed-dimension {ANY_VALUE}
                |REQ #maximum-size {'applies'}}},
PERM Layout-path {'270-degrees'}
```

},

```

REQ Application-comments {REQ #constraint-name {"10"},
PERM #external-data {ANY_VALUE}}
```

SPECIFIC:

```

CASE $DAC OF {
    $FDA: PERM Object-class {OBJECT_CLASS_ID_OF
                             (SnakingColumns)}
    $FPDA: REQ Object-class {OBJECT_CLASS_ID_OF
                             (SnakingColumns)}
},
REQ Subordinates {SUB_ID_OF(ColumnVariable)+},
PERM Application-comments {REQ #constraint-name {"10"},
PERM #external-data {ANY_VALUE}}
```

7.4.3.11 ColumnVariable: ANY-FRAME-VARIABLE {

GENERIC:

```

CASE $DAC OF {
    $PDA-FPDA:

    PERM Permitted-categories {ANY_STRING},

    REQ Position {REQ #variable-position {
                    PERM #offset {ANY_VALUE},
                    PERM #separation {ANY_VALUE},
                    PERM #alignment {ANY_VALUE},
                    PERM #fill-order {'normal-order'}}
```



```

CASE SUPERIOR (VariableCompositeBody(Layout-path)) OF {

  {'270-degrees'}: -- page layout A --
    REQ Dimensions      {REQ #horizontal-dimension
                        {REQ #fixed-dimension {ANY_VALUE}},
                        REQ #vertical-dimension
                        {REQ #rule-b {ANY_VALUE}
                        |REQ #maximum-size {'applies'}}}},
    PERM Layout-path    {'270-degrees'}

  {'0-degrees'}: -- page layout B --
    REQ Dimensions      {REQ #horizontal-dimension
                        {REQ #rule-b {ANY_VALUE}
                        |REQ #maximum-size {'applies'}}},
                        REQ #vertical-dimension
                        {REQ #fixed-dimension {ANY_VALUE}}},
    REQ Layout-path     {'0-degrees'}

  {'180-degrees'}: -- page layouts C and D --
    REQ Dimensions      {REQ #horizontal-dimension
                        {REQ #rule-b {ANY_VALUE}
                        |REQ #maximum-size {'applies'}}},
                        REQ #vertical-dimension
                        {REQ #fixed-dimension {ANY_VALUE}}}
    REQ Layout-path     {'180-degrees'}
},
REQ Application-comments {REQ #constraint-name {"9"},
                          PERM #external-data {ANY_VALUE}},
SPECIFIC:
CASE $DAC OF {
  $FDA: PERM Object-class {OBJECT_CLASS_ID_OF(ColumnVariable)}
  $FPDA: REQ Object-class {OBJECT_CLASS_ID_OF(ColumnVariable)}
},
REQ Subordinates          {SUB_ID_OF(SpecificBlock)+}},
PERM Application-comments {REQ #constraint-name {"9"},
                          PERM #external-data {ANY_VALUE}}}

```

7.4.3.12 ColumnFixed: ANY-FRAME-VARIABLE {

```

GENERIC:
CASE $DAC OF {
  $PDA-FPDA:

    REQ Permitted-categories {ANY_STRING},

    REQ Position              {REQ #fixed-position
                              {REQ #horizontal-position {ANY_VALUE},
                              REQ #vertical-position {ANY_VALUE}}}

CASE SUPERIOR (VariableCompositeBody(Layout-path)) OF {

```



```
{'270-degrees'}: -- page layout a --
    REQ Dimensions {REQ #horizontal-dimension
                    {REQ #fixed-dimension {ANY_VALUE},
                     |REQ #maximum-size {'applies'}},
                    REQ #vertical-dimension
                    {REQ #rule-b {ANY_VALUE}
                     |REQ #maximum-size {'applies'}}},
    PERM Layout-path {'270-degrees'}
```

```
{'0-degrees'}: -- page layout B --
    REQ Dimensions {REQ #horizontal-dimension
                    {REQ #rule-b {ANY_VALUE}
                     |REQ #maximum-size {'applies'}},
                    REQ #vertical-dimension
                    {REQ #fixed-dimension {ANY_VALUE}
                     |REQ #maximum-size {'applies'}}},
    REQ Layout-path {'0-degrees'}
```

```
{'180-degrees'}: -- page layouts C and D --
    REQ Dimensions {REQ #horizontal-dimension
                    {REQ #maximum-size {'applies'}},
                    REQ #vertical-dimension
                    {REQ #fixed-dimension {ANY_VALUE}
                     |REQ #maximum-size {'applies'}}},
    REQ Layout-path {'180-degrees'}
```

```
}
```

```
},
```

```
REQ Application-comments {REQ #constraint-name {8"},
                          PERM #external-data {ANY_VALUE}}
```

SPECIFIC:

```
CASE $DAC OF {
    $FDA: PERM Object-class {OBJECT_CLASS_ID_OF
                             (ColumnFixed)}
    $FPDA: REQ Object-class {OBJECT_CLASS_ID_OF
                             (ColumnFixed)}
},
REQ Subordinates {SUB_ID_OF(SpecificBlock)+},
PERM Application-comments {REQ #constraint-name {"8"},
                           PERM #external-data {ANY_VALUE}}}
```

7.4.3.13 FootnoteArea: ANY-FRAME-VARIABLE {

GENERIC:

```
CASE $DAC OF {
    $PDA-FPDA:

        REQ Position {REQ #variable-position {
                        PERM #offset {ANY_VALUE},
                        PERM #separation {ANY_VALUE},
                        PERM #alignment {ANY_VALUE},
                        REQ #fill-order {'reverse-order'}}},
```

REQ Permitted-categories {"Footnote"}

CASE SUPERIOR (VariableCompositeBody(Layout-path)) OF {

{'270-degrees'}: -- page layout A --

REQ Dimensions {REQ #horizontal-dimension
 {REQ #fixed-dimension {ANY_VALUE}
 |REQ #maximum-size {'applies'}},
 REQ #vertical-dimension
 {REQ #rule-b {ANY_VALUE}}},
 PERM Layout-path {'270-degrees'}

{'0-degrees'}: -- page layout B --

REQ Dimensions {REQ #horizontal-dimension
 {REQ #rule-b {ANY_VALUE}},
 REQ #vertical-dimension
 {REQ #fixed-dimension {ANY_VALUE}
 |REQ #maximum-size {'applies'}}},
 REQ Layout-path {'0-degrees'}

{'180-degrees'}: -- page layouts C and D --

REQ Dimensions {REQ #horizontal-dimension
 {REQ #rule-b {ANY_VALUE}},
 REQ #vertical-dimension
 {REQ #fixed-dimension {ANY_VALUE}
 |REQ #maximum-size {'applies'}}},
 REQ Layout-path {'180-degrees'}

}
 },

REQ Application-comments {REQ #constraint-name {"15"},
 PERM #external-data {ANY_VALUE}}

SPECIFIC:

CASE \$DAC OF {
 \$FDA: PERM Object-class {OBJECT_CLASS_ID_OF(FootnoteArea)}
 \$FPDA: REQ Object-class {OBJECT_CLASS_ID_OF(FootnoteArea)}
 },
 REQ Subordinates {SUB_ID_OF(SpecificBlock)+},
 PERM Application-comments {REQ #constraint-name {"15"},
 PERM #external-data {ANY_VALUE}}

7.4.3.14 BasicHeader: ANY-FRAME-FIXED {

GENERIC:

CASE \$DAC OF{
 \$PDA-FPDA:
 REQ Logical-source {OBJECT_CLASS_ID_OF(CommonContent)}},
 PERM Layout-path {'270-degrees' -- page layouts A,B,C --
 |'180-degrees' -- page layout D --},
 REQ Application-comments {REQ #constraint-name {"27"},
 PERM #external-data {ANY_VALUE}},

SPECIFIC:

```

CASE    $DAC OF {
        $FDA: PERM  Object-class  {OBJECT_CLASS_ID_OF(BasicHeader)}
        $FPDA: REQ   Object-class  {OBJECT_CLASS_ID_OF(BasicHeader)}
        },
REQ     Subordinates              {SUB_ID_OF(SpecificBlock)+},
PERM    Application-comments      {REQ #constraint-name {"27"},
                                   PERM #external-data {ANY_VALUE}}

```

7.4.3.15 BasicFooter: ANY-FRAME-FIXED {

GENERIC:

```

CASE    $DAC OF{
        $PDA-FPDA:
REQ     Logical-source            {OBJECT_CLASS_ID_OF(CommonContent)}},
PERM    Layout-path              {'270-degrees' -- page layouts A,B,C --
                                   |'180-degrees' -- page layout D --},
REQ     Application-comments      {REQ #constraint-name {"33"},
                                   PERM #external-data {ANY_VALUE}},

```

SPECIFIC:

```

CASE    $DAC OF {
        $FDA: PERM  Object-class  {OBJECT_CLASS_ID_OF(BasicFooter)}
        $FPDA: REQ   Object-class  {OBJECT_CLASS_ID_OF(BasicFooter)}
        },
REQ     Subordinates              {SUB_ID_OF(SpecificBlock)+},
PERM    Application-comments      {REQ #constraint-name {"33"},
                                   PERM #external-data {ANY_VALUE}}

```

7.4.3.16 CompositeHeader: ANY-FRAME-FIXED {

GENERIC:

```

CASE    $DAC OF {
        $PDA-FPDA:
REQ     Generator-for-subordinates {$HeaderFooterGFS}
        },
PERM    Layout-path              {'270-degrees' -- page layouts A,B,C --
                                   |'180-degrees' -- page layout D --},
REQ     Application-comments      {REQ #constraint-name {"5"},
                                   PERM #external-data {ANY_VALUE}}

```

SPECIFIC:

```

CASE    $DAC OF {
        $FDA: PERM  Object-class  {OBJECT_CLASS_ID_OF(CompositeHeader)}
        $FPDA: REQ   Object-class  {OBJECT_CLASS_ID_OF(CompositeHeader)}
        },
REQ     Subordinates              {SUB_ID_OF(SourcedContentFixed)+,
                                   SUB_ID_OF(ArrangedContentFixed)+,
                                   SUB_ID_OF(SourcedContentVariable)+,
                                   SUB_ID_OF(ArrangedContentVariable)+},
PERM    Application-comments      {REQ #constraint-name {"5"},
                                   PERM #external-data {ANY_VALUE}}

```


7.4.3.17 CompositeFooter: ANY-FRAME-FIXED {

GENERIC:

```

CASE $DAC OF {
  $PDA-FPDA:
  REQ Generator-for-subordinates {$HeaderFooterGFS}
    },
  PERM Layout-path {'270-degrees' -- page layouts A,B,C --
    | '180-degrees' -- page layout D --},
  REQ Application-comments {REQ #constraint-name {"32"},
    PERM #external-data {ANY_VALUE}},

```

SPECIFIC:

```

CASE $DAC OF {
  $FDA: PERM Object-class {OBJECT_CLASS_ID_OF(CompositeFooter)}
  $FPDA: REQ Object-class {OBJECT_CLASS_ID_OF(CompositeFooter)}
    },
  REQ Subordinates {SUB_ID_OF(SourcedContentFixed)+,
    SUB_ID_OF(ArrangedContentFixed)+,
    SUB_ID_OF(SourcedContentVariable)+,
    SUB_ID_OF(ArrangedContentVariable)+},
  PERM Application-comments {REQ #constraint-name {"32"},
    PERM #external-data {ANY_VALUE}}

```

7.4.3.18 SourcedContentVariable: ANY-FRAME-VARIABLE {

GENERIC:

```

CASE $DAC OF {
  $PDA-FPDA:

  REQ Logical-source {OBJECT_CLASS_ID_OF(CommonContent)},
  REQ Position {REQ #variable-position {
    PERM #offset {ANY_VALUE},
    PERM #separation {ANY_VALUE},
    PERM #alignment {ANY_VALUE},
    PERM #fill-order {'normal-order'}}}

```

```

CASE SUPERIOR (CompositeHeader|CompositeFooter
  (Layout-path)) OF {

```

```

{'270-degrees'}:

```

```

  REQ Dimensions {REQ #horizontal-dimension
    {REQ #fixed-dimension {ANY_VALUE}
    | REQ #maximum-size {'applies'}},
  REQ #vertical-dimension
    {REQ #fixed-dimension {ANY_VALUE}
    | REQ #rule-b {ANY_VALUE}}},

```

```

  PERM Layout-path {'270-degrees'}
{'180-degrees'}:

```

```

  REQ Dimensions {REQ #horizontal-dimension
    {REQ #fixed-dimension {ANY_VALUE}
    | REQ #rule-b {ANY_VALUE}},
  REQ #vertical-dimension
    {REQ #fixed-dimension {ANY_VALUE}
    | REQ #maximum-size {'applies'}}},

```



```

        REQ  Layout-path      {'180-degrees'}
      }
    }
  REQ  Application-comments  {REQ #constraint-name {"19"},
                              PERM #external-data {ANY_VALUE}}

SPECIFIC:
  CASE  $DAC OF {
    $FDA: PERM  Object-class  {OBJECT_CLASS_ID_OF
                              (SourcedContentVariable)}
    $FPDA: REQ  Object-class  {OBJECT_CLASS_ID_OF
                              (SourcedContentVariable)}
  },
  REQ  Subordinates          {SUB_ID_OF(SpecificBlock)+},
  PERM  Application-comments  {REQ #constraint-name {"19"},
                              PERM #external-data {ANY_VALUE}}

```

7.4.3.19 ArrangedContentVariable: ANY-FRAME-VARIABLE {

```

GENERIC:
  CASE  $DAC OF {
    $PDA-FPDA:

      REQ  Generator-for-subordinates
            {<construction-expr>::=SEQ
            (OBJECT_CLASS_ID_OF(GenericBlock)...)};

      REQ  Position
            {REQ #variable-position {
              PERM #offset {ANY_VALUE},
              PERM #separation {ANY_VALUE},
              PERM #alignment {ANY_VALUE},
              PERM #fill-order {'normal-order'}};

      REQ  Dimensions
            {REQ #horizontal-dimension
              {REQ #fixed-dimension {ANY_VALUE},
              REQ #vertical-dimension
              {REQ #fixed-dimension {ANY_VALUE}};

      },
  REQ  Application-comments  {REQ #constraint-name {"17"},
                              PERM #external-data {ANY_VALUE}};

SPECIFIC:
  CASE  $DAC OF {
    $FDA: PERM  Object-class  {OBJECT_CLASS_ID_OF
                              (ArrangedContentVariable)}
    $FPDA: REQ  Object-class  {OBJECT_CLASS_ID_OF
                              (ArrangedContentVariable)}
  },
  REQ  Subordinates          {SUB_ID_OF(GenericBlock)+},
  PERM  Application-comments  {REQ #constraint-name {"17"},
                              PERM #external-data {ANY_VALUE}}

```

7.4.3.20 SourcedContentFixed: ANY-FRAME-VARIABLE {

GENERIC:

```

CASE $DAC OF {
  $PDA-FPDA:

    REQ Logical-source {OBJECT_CLASS_ID_OF(CommonContent)},
    REQ Position {REQ #fixed-position
                  {REQ #horizontal-position{ANY_VALUE},
                   REQ #vertical-position{ANY_VALUE}}}

    CASE SUPERIOR (CompositeHeader|Compositefooter
                  (Layout-path)) OF {
      {'270-degrees'}:
        REQ Dimension {REQ #horizontal-dimension
                      {REQ #fixed-dimension {ANY_VALUE}},
                      REQ #vertical-dimension
                      {REQ #fixed-dimension {ANY_VALUE}
                       |REQ #rule-b {ANY_VALUE}}},
        PERM Layout-path {'270-degrees'}
      {'180-degrees'}:
        REQ Dimension {REQ #horizontal-dimension
                      {REQ #fixed-dimension {ANY_VALUE}
                       |REQ #rule-b {ANY_VALUE}},
                      REQ #vertical-dimension
                      {REQ #fixed-dimension {ANY_VALUE}}},
        REQ Layout-path {'180-degrees'}
      }
    },
    REQ Application-comments {REQ #constraint-name {"18"},
                              PERM #external-data {ANY_VALUE}}

```

SPECIFIC:

```

CASE $DAC OF {
  $FDA: PERM Object-class {OBJECT_CLASS_ID_OF
                          (SourcedContentFixed)}
  $FPDA: REQ Object-class {OBJECT_CLASS_ID_OF
                          (SourcedContentFixed)}
  },
  REQ Subordinates {SUB_ID_OF(SpecificBlock)+},
  PERM Application-comments {REQ #constraint-name {"18"},
                             PERM #external-data {ANY_VALUE}}

```

7.4.3.21 ArrangedContentFixed: ANY-FRAME-FIXED {

GENERIC:

```

CASE $DAC OF{
  $PDA-FPDA:
    REQ Generator-for-subordinates {SEQ(OBJECT_CLASS_ID_OF
                                         (GenericBlock)...)}},
    REQ Application-comments {REQ #constraint-name {"16"},
                              PERM #external-data {ANY_VALUE}}

```

SPECIFIC:

```
CASE $DAC OF {
$FDA: PERM Object-class {OBJECT_CLASS_ID_OF
                          (ArrangedContentFixed)}
$FPDA: REQ Object-class {OBJECT_CLASS_ID_OF
                         (ArrangedContentFixed)}
},
REQ Subordinates {SUB_ID_OF(GenericBlock)+},
PERM Application-comments {REQ #constraint-name {"16"},
                           PERM #external-data {ANY VALUE}}}
```

7.4.3.22 GenericBlock: BLOCK {

GENERIC:

REQ	Object-class-identifier	{ANY_VALUE}
REQ	Object-type	{'block'},
REQ	Content-architecture-class	{\$FC \$FPC \$FPR \$FPG},
PERM	Resource	{ANY_VALUE},
PERM	Content-portions	{CONTENT_ID_OF (character-content-portion)+ CONTENT_ID_OF (raster-graphics-content-portion) CONTENT_ID_OF (geometric-graphics-content-portion)};
PERM	User-readable-comments	{ANY_STRING},
PERM	User-visible-name	{ANY_STRING},
PERM	Position	{REQ #fixed-position {REQ #horizontal-position{ANY_VALUE}, REQ #vertical-position{ANY_VALUE}}},
PERM	Dimension	{REQ #horizontal-dimension {REQ #fixed-dimension {ANY_VALUE}}, REQ #vertical-dimension {REQ #fixed-dimension {ANY_VALUE}}},
REQ	Application-comments	{REQ #constraint-name {"29"}, PERM #external-data {ANY VALUE}}

SPECIFIC:

```

CASE $DAC OF {
    $FDA: PERM Object-class {OBJECT_CLASS_ID_OF(GenericBlock)}
    $FPDA: REQ Object-class {OBJECT_CLASS_ID_OF(GenericBlock)}
    },
CASE Generic Block (object-class) OF {VOID:
    REQ Content-portions {CONTENT_ID_OF
        (character-content-portion)+,
        CONTENT_ID_OF
        (raster-graphics-content-portion),
        CONTENT_ID_OF
        (geometric-graphics-content-portion)},
    },
PERM Application-comments {REQ #constraint-name {"29"},
    PERM #external-data {ANY VALUE}}}
```

SPECIFIC AND GENERIC:

```

PERM Presentation-style {STYLE_ID_OF(P-Style1)
                          |STYLE_ID_OF(P-Style2)
                          |STYLE_ID_OF(P-Style3)},

```


7.4.3.23 SpecificBlock: BLOCK {

SPECIFIC:

```

    PERM    Presentation-style      {STYLE_ID_OF(P-Style1)
                                     |STYLE_ID_OF(P-Style2)
                                     |STYLE_ID_OF(P-Style3)
                                     |STYLE_ID_OF(P-Style4)},

    REQ     Content-portions        {CONTENT_ID_OF
                                     (character-content-portion)+,
                                     CONTENT_ID_OF
                                     (raster-graphics-content-portion),
                                     CONTENT_ID_OF
                                     (geometric-graphics-content-portion)},

    PERM    Application-comments    {REQ #constraint-name {"30"},
                                     PERM #external-data {ANY_VALUE}}

```

7.5 Layout style constituent constraints

7.5.1 Macro definitions

```

DEFINE(LayoutObjectClasses, "
    OBJECT_CLASS_ID_OF(PageSet)
    |OBJECT_CLASS_ID_OF(Page)
    |OBJECT_CLASS_ID_OF(RectoPage)
    |OBJECT_CLASS_ID_OF(VersoPage)
    |OBJECT_CLASS_ID_OF(BasicBody)
    |OBJECT_CLASS_ID_OF(VariableCompositeBody)
    |OBJECT_CLASS_ID_OF(BasicFloat)
    |OBJECT_CLASS_ID_OF(SnakingColumns)
    |OBJECT_CLASS_ID_OF(SynchronizedColumns)
    |OBJECT_CLASS_ID_OF(ColumnFixed)
    |OBJECT_CLASS_ID_OF(ColumnVariable)
    ")

```

```

DEFINE(SameLayoutObject, "
    REQ #logical-object{<object-id-expr> ::= PREC-OBJ(CURR-OBJ);
    |'null'},
    PERM #layout-object{'page'}
    ")

```

7.5.2 FACTOR: ANY-LAYOUT-STYLE {

```

    REQ     Layout-style-identifier {ANY_VALUE},
    PERM    User-visible-name        {ANY_STRING},
    PERM    User-readable-comments   {ANY_STRING}

```


5.3 Layout style constituent constraints

5.3.1 L-Style1: ANY-LAYOUT-STYLE {

-- this style is used for the constituent Passage only --

```

CASE Document-profile(Generic-layout-structure) OF {
  {'complete-generator-set':
    PERM  Layout-object-class    {OBJECT_CLASS_ID_OF(PageSet)},
    PERM  New-layout-object      {OBJECT_CLASS_ID_OF(PageSet)},
    PERM  Indivisibility         {$LayoutObjectClasses
                                |ANY_STRING|'page'|'null'}
  VOID:
    PERM  Indivisibility         {ANY_STRING|'page'|'null'
                                }}
  -- ANY_STRING is interpreted as representing the name of a layout
  category --

```

5.3.2 L-Style2: ANY-LAYOUT-STYLE {

-- this style is used for the constituents
BodyText, and Number --

```

CASE Document-profile(Generic-layout-structure) OF {
  {'complete-generator-set':
    PERM  Indivisibility         {$LayoutObjectClasses,
                                |ANY_STRING|'page'|'null'},
    PERM  New-layout-object      {$LayoutObjectClasses
                                |ANY_STRING|'page'|'null'}
  VOID:
    PERM  Indivisibility         {ANY_STRING|'page'|'null'},
    PERM  New-layout-object      {ANY_STRING|'page'|'null'
                                }
  PERM  Layout-category         {ANY_STRING},
  PERM  Same-layout-object      {$SameLayoutObject},
  PERM  Concatenation           {ANY_VALUE},
  PERM  Offset                  {ANY_VALUE},
  PERM  Separation              {PERM #leading-edge{ANY_INTEGER},
                                PERM #trailing-edge{ANY_INTEGER}},
  PERM  Block-alignment         {ANY_VALUE},
  PERM  Synchronization        {ANY_VALUE}}

```

5.3.3 L-Style3: ANY-LAYOUT-STYLE {

-- this style is used for the constituents
CommonText and PageNumber --

```

PERM  Concatenation            {ANY_VALUE},
PERM  Offset                   {ANY_VALUE},
PERM  Block-alignment         {ANY_VALUE},
PERM  Separation              {PERM #leading-edge{ANY_INTEGER},
                                PERM #trailing-edge{ANY_INTEGER}}}

```

7.5.3.4 L-Style4: ANY-LAYOUT-STYLE {

-- this style is used for the constituents NumberedSegment
and Paragraph --

```

CASE Document-profile(Generic-layout-structure) OF {
  {'complete-generator-set':
    PERM    Indivisibility      {$LayoutObjectClasses
                                |ANY_STRING| 'page' | 'null'},
    PERM    Layout-object-class {OBJECT_CLASS_ID_OF(Pageset)},
    PERM    New-layout-object   {$LayoutObjectClasses
                                |ANY_STRING| 'page' | 'null'}
  VOID:
    PERM    Indivisibility      {ANY_STRING| 'page' | 'null'},
    PERM    New-layout-object   {ANY_STRING| 'page' | 'null'}
    }
  PERM    Same-layout-object    {$SameLayoutObject},
  PERM    Synchronization       {ANY_VALUE}}

```

-- ANY_STRING is interpreted as representing the name of a layout
category

7.5.3.5 L-Style5: ANY-LAYOUT-STYLE {

-- this style is used for the constituents
BodyRaster and BodyGeometric --

```

CASE Document-profile(Generic-layout-structure) OF {
  {'complete-generator-set':
    PERM    New-layout-object   {$LayoutObjectClasses,
                                |ANY_STRING| 'page' | 'null'}
  VOID:
    PERM    New-layout-object   {ANY_STRING| 'page' | 'null'}
    }

```

-- ANY_STRING is interpreted as representing the name of a layout
category --

```

PERM    Layout-category      {ANY_STRING},
PERM    Offset               {ANY_VALUE},
PERM    Same-layout-object   {$SameLayoutObject},
PERM    Separation           {PERM #leading-edge{ANY_INTEGER},
                             PERM #trailing-edge{ANY_INTEGER}},
PERM    Block-alignment     {ANY_VALUE},
PERM    Synchronization     {ANY_VALUE}}

```

7.5.3.6 L-Style6: ANY-LAYOUT-STYLE {

-- this style is used for the constituent FootnoteText --

```

REQ    Layout-category      {"Footnote"},
PERM   Concatenation        {ANY_VALUE},
PERM   Indivisibility       {OBJECT_CLASS_ID_OF (FootnoteArea)
                             | 'PAGE'
                             | 'NULL'};
PERM   Offset               {ANY_VALUE},
PERM   Block-alignment      {ANY_VALUE},
PERM   Separation           {PERM #leading-edge{ANY_INTEGER},
                             PERM #trailing-edge{ANY_INTEGER}}

```

7.5.3.7 L-Style7: ANY-LAYOUT-STYLE {

-- this style is used for the constituent Footnote only --

```

PERM   Same-layout-object   {$SameLayoutObject}}

```

7.5.3.8 L-Style8: ANY-LAYOUT-STYLE {

-- this style is used for the constituents
CommonRaster and CommonGeometric --

```

PERM   Offset               {ANY_VALUE},
PERM   Block-alignment      {ANY_VALUE},
PERM   Separation           {PERM #leading-edge{ANY_INTEGER},
                             PERM #trailing-edge{ANY_INTEGER}}

```

7.5.3.9 L-Style9: ANY-LAYOUT-STYLE {

-- this style is used for the constituent FootnoteNumber --

```

REQ    Layout-category      {"Footnote"},
PERM   Offset               {ANY_VALUE},
PERM   Block-alignment      {ANY_VALUE},
PERM   Separation           {PERM #leading-edge{ANY_INTEGER},
                             PERM #trailing-edge{ANY_INTEGER}}

```

7.5.3.10 L-Style10: ANY-LAYOUT-STYLE {

-- this style is used for the constituents
FootnoteReference only --

```

CASE Document-profile(Generic-layout-structure) OF {
  {'complete-generator-set':
    PERM   Indivisibility    {$LayoutObjectClasses,
                              | ANY_STRING | 'page' | 'null'}
  VOID:
    PERM   Indivisibility    {ANY_STRING | 'page' | 'null'}
  }

```


PERM	Layout-category	{ANY_STRING},
PERM	Same-layout-object	{\$SameLayoutObject},
PERM	Concatenation	{ANY_VALUE},
PERM	Offset	{ANY_VALUE},
PERM	Separation	{PERM #leading-edge{ANY_INTEGER}, PERM #trailing-edge{ANY_INTEGER}},
PERM	Block-alignment	{ANY_VALUE}

7.5.3.11 L-Style11: ANY_LAYOUT_STYLE{

-- this style is used for the constituent Footnotebody --

PERM	Same-layout-object	{\$SameLayoutObject},
PERM	Indivisibility	{OBJECT_CLASS_ID_OF (FootnoteArea) 'PAGE' 'NULL'};

7.6 Presentation style constituent constraints

7.6.1 Macro definitions

No macro definitions are applicable to this clause.

7.6.2 Factor constraints

7.6.2.1 FACTOR: ANY-PRESENTATION-STYLE {

REQ	Presentation-style-identifier	{ANY_VALUE},
PERM	User-visible-name	{ANY_STRING},
PERM	Border	{ANY_VALUE},
PERM	User-readable-comments	{ANY_STRING}}

7.6.3 Presentation style constituent constraints

7.6.3.1 P-Style1: ANY-PRESENTATION-STYLE {

-- this style is used for the constituents BodyText, Number, FootnoteNumber, FootnoteReference, FootnoteText, GenericBlock and SpecificBlock --

PERM	Presentation attributes	{
	PERM #character-attributes	{
	PERM #alignment	{ANY_VALUE},
	PERM #character-spacing	{ANY_VALUE},
	PERM #character-fonts	{ANY_VALUE},
	PERM #character-orientation	{'0-degrees' '90-degrees'},
	PERM #character-path	{ANY_VALUE},
	PERM #code-extension-announcers	{\$CDEXTAN},
	PERM #first-line-offset	{ANY_VALUE},
	PERM #graphic-character-sets	{\$PERMIT-GRCHAR},


```

    PERM #graphic-character-subrepertoire {ANY_VALUE},
    PERM #indentation                     {ANY_VALUE},
    PERM #itemisation                     {ANY_VALUE},
    PERM #kerning-offset                  {ANY_VALUE},
    PERM #line-progression                 {ANY_VALUE},
    PERM #line-spacing                     {ANY_VALUE},
    PERM #line-layout-table               {ANY_VALUE},
    PERM #orphan-size                     {ANY_VALUE},
    PERM #proportional-line-spacing       {ANY_VALUE},
    PERM #widow-size                       {ANY_VALUE}}}

```

7.6.3.2 P-Style2: ANY-PRESENTATION-STYLE {

-- this style is used for the constituents BodyGeometric, CommonGeometric, Generic Block and Specific Block --

```

    PERM Presentation attributes {
        PERM #geometric-graphics-attributes {
            PERM #picture-dimensions {ANY_VALUE},
            PERM #picture-orientation {ANY_VALUE},
            PERM #text-rendition      {PERM #fonts-list{ANY_VALUE},
                                     PERM #character-set-list
                                     {ANY_VALUE}}}
    }

```

7.6.3.4 P-Style3: ANY-PRESENTATION-STYLE {

-- this style is used for the constituents BodyRaster, CommonRaster, Generic Block and Specific Block --

```

    PERM Presentation attributes {
        PERM #raster-graphics-attributes {
            PERM #image-dimensions {ANY_VALUE},
            PERM #clipping          {ANY_VALUE},
            PERM #pel-spacing       {ANY_VALUE},
            PERM #spacing-ratio    {ANY_VALUE}}}
    }

```

7.6.3.5 P-Style4: ANY-PRESENTATION-STYLE {

-- this style is used for the constituents CommonText, PageNumber and SpecificBlock --

```

    PERM Presentation attributes {
        PERM #character-attributes {
            PERM #alignment {ANY_VALUE},
            PERM #character-spacing {ANY_VALUE},
            PERM #character-fonts {ANY_VALUE},
            PERM #character-orientation {
                '0-degrees'
                | '90-degrees'},
            PERM #character-path {
                '0-degrees'
                | '180-degrees'
                | '270-degrees'},
        }
    }

```

```

    PERM #code-extension-announcers    {$CDEXTAN},
    PERM #first-line-offset            {ANY_VALUE},
    PERM #graphic-character-sets       {$PERMIT-GRCHAR},
    PERM #graphic-character-subrepertoire {ANY_VALUE},
    PERM #graphic-rendition            {$GRAPHICRENDITIONS},
    PERM #indentation                  {ANY_VALUE},
    PERM #itemisation                   {ANY_VALUE},
    PERM #kerning-offset                {ANY_VALUE},
    PERM #line-progression              {ANY_VALUE},
    PERM #line-spacing                  {ANY_VALUE},
    PERM #line-layout-table             {ANY_VALUE},
    PERM #proportional-line-spacing    {ANY_VALUE}}

```

7.7 Content portion constituent constraints

7.7.1 Macro definitions

No macro definitions are applicable to this clause.

7.7.2 Factor constraints

7.7.2.1 FACTOR: ANY-CONTENT {

CASE \$DAC OF {

\$FDA:

REQ Content-identifier-layout {ANY_VALUE}

\$PDA:

REQ Content-identifier-logical {ANY_VALUE}

-- This attribute is specified, if the content portion is associated with a basic logical object or a basic logical object class. --

|REQ Content-identifier-layout {ANY_VALUE}

-- This attribute is specified, if the content portion is associated with a basic layout object class. --

\$FPDA:

REQ Content-identifier-layout {ANY_VALUE},

REQ Content-identifier-logical {ANY_VALUE}

-- Both attributes are specified, if the content portion is associated with a basic logical object and a basic layout object. --

|REQ Content-identifier-layout {ANY_VALUE}

-- This attribute is specified, if the content portion is associated with a basic layout object class. --

|REQ Content-identifier-logical {ANY_VALUE}

-- This attribute is specified, if the content portion is associated with a basic logical object class. --

7.7.3 Content portion constraints

7.7.3.1 Character-content-portion: ANY-CONTENT {

```

PERM  Type-of-coding           {ASN.1{2 8 3 6 0}},
PERM  Alternative-representation {ANY_STRING},
PERM  Content-information
      {CHARACTER {#STAB {ANY_VALUE}
                    #SHS {0,1,2,3,4}
                    #SGR {$GRAPHICRENDITIONS}
                    #SVS {ANY_VALUE}
                    #SLS {ANY_VALUE}
                    #SCS {ANY_VALUE}
                    #SRS {ANY_VALUE}
                    #JFY {0}
                    #CR
                    #LF
                    #VPB
                    #VPR
                    #PLD
                    #PLU
                    #SP
                    #SUB
                    #BPH
                    #NBH
                    #SOS
                    #ST
                    #LS0
                    #LS1R
                    #LS2R
                    #LS3R
                    #SS2
                    #SS3
                    #ESC{$DEG-CORE-G0}
                    #ESC{$DEG-646-G0}
                    #ESC{$DEG-ANY-G1}
                    #ESC{$DEG-ANY-G2}
                    #ESC{$DEG-ANY-G3}
                    #ESC{$DEG-EMPTY-G1}
                    }...}}

```

7.7.3.2 Raster-graphics-content-portion: ANY-CONTENT {

```

PERM  Number-of-lines           {>0},
REQ   Number-of-pels-per-line   {>=0},
PERM  Type-of-coding             {ASN.1{2 8 3 7 0} -- T.6 encoding --
                                |ASN.1{2 8 3 7 1} -- T.4 one-dimensional
                                encoding --
                                |ASN.1{2 8 3 7 2} -- T.4 two dimensional
                                encoding --
                                |ASN.1{2 8 3 7 3} -- bitmap encoding --},
PERM  Compression               {ANY_VALUE},
PERM  Alternative-representation {ANY_STRING},
PERM  Content-information        {RASTER}}

```

7.7.3.3 Geometric-graphics-content-portion: ANY-CONTENT {

```

PERM  Type-of-coding             {ASN.1{2 8 3 8 0}},
PERM  Alternative-representation {ANY_STRING},
PERM  Content-information        {GEOMETRIC}}

```

8 Interchange format

For conformance to this profile, the interchange format class A shall be used when applying ODIF, and the interchange format SDIF shall be used when applying ODL in conjunction with SDIF.

8.1 Interchange format class A

8.1.1 Interchange format

The value of the document profile attribute "Interchange format" for this interchange format is "if-a". This form of ODIF is defined in CCITT Recommendation T.415|ISO 8613-5.

8.1.2 DAP identifier

The value for the document profile attribute "Document application profile" for this interchange format is represented by the following object identifier.

ASN.1 {2 8 4 0 26 0}

8.1.3 Encoding of application comments

The encoding of the attribute "application comments" is defined in this encoding as an octet string as specified in CCITT Recommendation T.415|ISO 8613-5. This document application profile requires that the encoding within that octet string be in accordance with the ASN.1 syntax specified in the following module definition:

```

FOD-DAPSpecification
DEFINITIONS::=BEGIN
EXPORTS Appl-Comm-Encoding;

Appl-Comm-Encoding::=      SEQUENCE{
    constraint-name          [0] IMPLICIT PrintableString OPTIONAL,
    external-data            [1] IMPLICIT OCTET STRING OPTIONAL }
END

```

8.1.4 Data lengths

The maximum length of data values of the type OCTET STRING, as defined in ISO 8824, in data streams which may be encoded in accordance with this DAP is 32767 octets. If it is required to encode an octet string of greater length than this, constructed type encoding shall be used. That is, data values greater than 32767 in length shall be split into a sequence of strings shorter than 32767, each of which is encoded using a primitive type.

8.2 Interchange format SDIF

8.2.1 Interchange format

The document profile attribute "Interchange format" does not apply for this interchange format. This form of ODIF is defined in Annex E of CCITT Recommendation T.415|ISO 8613-5. CCITT Recommendation T.415|ISO 8613-6, -7 and -8 contain additional specifications for interchange format.

8.2.2 DAP identifier

The value for the attribute "Document application profile" for this interchange format is represented by the following object identifier.

ASN.1 {1 0 11181 0 26 0}

NOTE - There is no requirement to include a part number arc within the object identifier for the DAP.

8.2.3 Encoding of application comments

The encoding of the attribute "Application comments" is defined in data stream conforming to this profile with this encoding with the following DTD definition:

```
<!--Public document type definition. Typical invocation:
  <!DOCTYPE fodapc PUBLIC "ISO/IEC 11181-1 : 1992//DTD
    Application Comments//EN" ["]>
  -->
<!ELEMENT fodapc    -0    (externl?)>
<!ATTLIST fodapc    consname CDATA  #IMPLIED>
<!ELEMENT externl   -0    (#PCDATA)>
<!ATTLIST externl   loc    ENTITY   #CONREF>
```

For example, a typical SUBDOC for representing the "application comments" of a Paragraph then would look like:

```
<!DOCTYPE fodapc PUBLIC "ISO/IEC 11181-1 : 1992//DTD Application Comments//EN">
<fodapc consname=6>
```

If the optionality of the attribute "fodapc" is specified in an earlier portion of the DTD, the invocation may be minimised further because the tag is not needed when "application comments" are not included as is the case here.

The content of external data appearing inline is restricted to parsable data. Referenced external data need not be parsable.

ANNEX A**Amendments and corrigenda**

(This annex forms an integral part of this Specification.)

A.1 Amendments**A.1.1 Amendments to the base standard**

The amendments applicable to this ISP includes the [CCITT Recommendation T.411|ISO 8613-1] - Amendment 1: 1990. This amendment includes text to be included in [CCITT Recommendation T.411|ISO 8613-1] as the following annexes:

- Annex E: Use of ISO/IEC 10021 (MOTIS) to interchange documents conforming to CCITT Recommendation T.415|ISO 8613;
- Annex F: Document Application Profile proforma and notation;
- Annex G: Conformance testing methodology;
- Annex H: Recording of documents conforming to [CCITT T.410 series Recommendations|ISO 8613] on flexible disk cartridges conforming to ISO 9293.

In addition, this amendment addresses the inclusion of the CCITT T.410 series Recommendations|ISO 8613 Technical Corregenda 1.

This ISP does not include the following features of the amendment:

- Addendum on security;
- Addendum on styles;
- Addendum on alternative representation.
- Addendum on colour;
- Addendum on tiled raster graphics

A.1.2 Proposed changes to standards due to defects

Proposed Technical Corrigendum No. 44 to [CCITT T.410 series Recommendations|ISO 8613] dated 15 December 1991 assumed to be ratified by ISO.

A.2 Corrigenda**A.2.1 Corrigenda to the ISP**

There is no corrigendum specific to this ISP.

ANNEX B**Recommendations**

(This annex does not form an integral part of this specification.)

B.1 Transfer methods for ODA**B.1.1 Conveyance of ODA over CCITT X.400-1984**

This recommendation describes how ODA body parts are to be encoded for transmission over a CCITT X.400-1984 service.

An ODA body part is encoded as OdaBodyPart in the definition given below:

```
OdaBodyPart ::= SEQUENCE { OdaBodyPartParameters, OdaData }
OdaBodyPartParameters ::= SET {
    document-application-profile
        [0] IMPLICIT OBJECT IDENTIFIER,
    document-architecture-class
        [1] IMPLICIT INTEGER {
            formatted (0),
            processable (1),
            formatted-processable (2) }
OdaData ::= SEQUENCE OF Interchange-Data-Element
```

NOTE - It is recommended to transfer an ODA document as a single body part with tag 12:

Oda [12] IMPLICIT OCTETSTRING

The content of the octet string is encoded as OdaBodyPart, defined above. However, this is out of the scope of this profile.

B.1.2 Conveyance of ODA over FTAM

This recommendation describes the FTAM Document Type to be used for minimal storage and transfer capabilities of ODA data streams. It is recognized that enhanced capabilities may at some point be added.

When using FTAM to transfer an ODA file, the FTAM-3, "ISO FTAM Unstructured Binary", document type shall be specified.

However, since files that do not contain ODA data streams can have the same document type, it is left up to the user of application programs that remotely access files using FTAM to know that a given file contains an ODA data stream.

B.1.3 Conveyance of ODA over DTAM

This recommendation provides for information concerning the interchange of ODA based documents with DTAM protocols.

DTAM (Document Transfer and Manipulation) is defined in the T.430-Series of recommendations and is, like ODA, an integral part of the T.400-Series of CCITT Recommendations named Open Document Architecture, Transfer and Manipulation.

The T.520-Series of recommendations contain Communication Application Profiles (CAP). Recommendation T.522 describes the Communication Application Profile BT1 for document bulk transfer. Recommendation T.522 is applicable for the Office Document Format Profile (FOD) published in this ISP.

NOTE - The use of BT1 within the end-to-end oriented Telematic Services Telefax 4 and Teletex is described in Recommendation T.561, subclause 7.1 and Recommendation T.562, subclause 7.1.

B.1.4 Conveyance of ODA over flexible disks

The recommended method for interchanging ODA documents between systems by the exchange of magnetically recorded Flexible Disk Cartridges is by the use of an annex to ISO 8613-1 (to be published), "Recording of Documents Conforming to ISO 8613 on Flexible Cartridges Conforming to ISO 9293". This annex provides for recording each ODA document as a separate file as defined by ISO 9293, "Volume and File Structure of Flexible Disk Cartridges for Information Interchange".

NOTE - Documents encoded in ODL may be stored such that each SGML ENTITY is recorded in a separate file or in the case of an SDIF encoding, the file may be stored in a single file.

B.2 Font reference

The recommended method for specifying a font reference is to be based on ISO 9541.

Font sizes from 6 to 72 points (100 to 1200 BMU) are intended to be supported by implementation conforming to this recommendation. All other values of font sizes may additionally be supported, but implementations may also support using some form of "fallback".

The minimum font properties and values from ISO 9541 that are to be specified in a Font-Attribute-Set be those specified by the following document application profile notation.

Font-Attribute-Set{

```

PERM      Fontname                {ANY_VALUE},
PERM      Standard-Version        {ANY_VALUE},
PERM      Design-Source           {ANY_VALUE},
PERM      Font-Family-Name        {ANY_VALUE},
PERM      Posture                  {'upright' | 'italic-forward'},
PERM      Weight                   {'light' | 'medium' | 'bold'},
PERM      Proportionate-Width      {ANY_VALUE},
PERM      Glyph-Complement {
    PERM #Included-Glyph-Collections
                                {ANY_VALUE},
    PERM #Excluded-Glyph-Collections
                                {ANY_VALUE},
    PERM #Included-Glyphs         {ANY_VALUE},
    PERM #Excluded-Glyphs         {ANY_VALUE}
},
PERM      Design-Size              {ANY_VALUE},
PERM      Min-Size {
    PERM #Numerator                {100 .. 1200},
    PERM #Denominator              {1}
},
PERM      Max-Size {
    PERM #Numerator                {100 .. 1200},
    PERM #Denominator              {1}
},
-- BMU Units equivalent to range of 6..72 point sizes --
PERM      Design-Group {
    PERM #Class                    {ANY_VALUE},
    PERM #Subclass                 {ANY_VALUE},
    PERM #Specific-Group           {ANY_VALUE}
},
PERM      Structure                {ANY_VALUE},
PERM      Writing-Modes {
    MUL
    PERM #Writing-Mode-Name        {ANY_VALUE},
    PERM #Nominal-Escapement-Direction
                                {ANY_VALUE},
    PERM #Escapement-Class         {ANY_VALUE},
    PERM #Average-Escapement-X     {ANY_VALUE},
    PERM #Average-Escapement-Y     {ANY_VALUE}
}
}

```

B.3 ISO 8632 (CGM) constraints for this DAP

It is recommended that geometric graphics content information contain only those elements listed in this portion of the document, in addition to the constraints imposed by CCITT Recommendation T.415/ISO 8613-8. It is believed that this subset of the CGM is sufficiently implemented to enable interworking of geometric graphics for application conforming to this document application profile.

Where an element has parameters, recommended constraints on the values are given. The "---" symbol indicates that there is no recommended constraint.

Requirements in ISO 8632 and CCITT Recommendation T.415/ISO 8613-8 concerning mandatory elements, parameters shall be fulfilled.

B.3.1 Delimiter elements

No-Op	An arbitrary sequence of n octets. Where n=0, 1, ..., 32 767. The sequence of zero or more octets is for padding purposes.
Begin Metafile	Support will be provided for strings with a length up to 254 octets, except for data records which will support strings with a length up to 32 767 octets.
End Metafile	
Begin Picture	Support will be provided for strings with a length up to 254 octets, except for data records which will support strings with a length up to 32 767 octets.
Begin Picture Body	
End Picture	

B.3.2 Metafile descriptor elements

Metafile Version	1
Metafile Description	Support will be provided for strings with a length up to 254 octets, except for data records which will support strings with a length up to 32 767 octets. The METAFILE DESCRIPTION string parameter will be used to include the sub-string "ISO FOD26" to label the content information as conforming to this profile. In addition, generators of content are encouraged to append a sub-string that identifies the company and product that produced the CGM.

VDC Type	Integer
Integer Precision	16
Real Precision	(0,9,23), (1,16,16)
Index Precision	16
Colour Precision	8, 16
Colour Index Precision	8, 16
Maximum Colour Index	0 .. 63
Colour Value Extent	--
Metafile Element List	--
Metafile Defaults Replacement	--
Font List	All fonts referenced in the metafile <u>shall</u> be defined. Font referencing in FONT LISTS using ISO 9541 names is preferred, but font names <u>may</u> be specified using proprietary font names.
Character Set list	All character sets referenced in the metafile <u>shall</u> be defined in CHARACTER SET LIST. Permissible character sets are the same as for character content architecture.
CHaracter Coding Announcer	--

B.3.3 Picture descriptor elements

Scaling Mode	The Scale Factor parameter of SCALING MODE element is always a 32-bit floating point value, even when the REAL PRECISION has selected fixed point for other real numbers. It is not apparent in ISO 8632 what the precision of this floating point value is when fixed point has been selected. Its precision <u>shall</u> be (0,9,23).
Colour Selection Mode	Indexed
Line Width Specification Mode	Scaled
Marker Size Specification Mode	Scaled
Edge Width Specification Mode	Scaled
VDC Extent	--
Background Colour	--

B.3.4 Control elements

VDC Integer Precision	16
VDC Real Precision	(0,9,23), (1,16,16)
Auxiliary Colour	--
Transparency	Transparent
Clip Rectangle	--
Clip Indicator	--

B.3.5 Graphical primitive elements

Polyline	Support for point lists with up to 255 vertices.
Disjoint Polyline	Support for point lists with up to 255 vertices.
Polymarker	Support for point lists with up to 255 vertices.
Text	Support will be provided for strings with a length up to 254 octets, except for data records which will support strings with a length up to 32 767 octets. Format effector control characters are disallowed in the string parameter.
Restricted Text	Support will be provided for strings with a length up to 254 octets, except for data records which will support strings with a length up to 32 767 octets. Format effector control characters are disallowed in the string parameter.
Append Text	Support will be provided for strings with a length up to 254 octets, except for data records which will support strings with a length up to 32 767 octets. Format effector control characters are disallowed in the string parameter.
Polygon	Support for point lists with up to 255 vertices.
Polygon Set	Support for point lists with up to 255 vertices.
Cell Array	--
Rectangle	--
Circle	--
Circular Arc 3 Point	--
Circular Arc 3 Point Close	--
Circular Arc Centre	--
Circular Arc Centre Close	--
Ellipse	--
Elliptical Arc	--
Elliptical Arc Close	--

B.3.6 Attribute elements

Line Bundle Index	1
Line Type	1-5
Line Width	--
Line Colour	--
Marker Bundle Index	1
Marker Type	1-5
Marker Size	--
Marker Colour	--
Text Bundle Index	1
Text Font Index	All fonts referenced (indexed by TEXT FONT INDEX) in the metafile <u>shall</u> be defined in FONT LIST either in presentation parameters of ISO 8613 or in ISO 8632.
Text Precision	0 (string)
Character Expansion Factor	1.0
Character Spacing	0.0
Text Colour	--
Character Height	--
Character Orientation	--
Text Alignment	--
Character Set Index	All character sets referenced in the metafile (indexed by CHARACTER SET INDEX) <u>shall</u> be defined in CHARACTER SET LIST. The only character sets which <u>may</u> be designated in G0 are ISO 646 IRV or versions of ISO 646. Other character sets <u>shall</u> be designated in G1, G2 or G3.
Alternate Character Set Index	All character sets referenced in the metafile (indexed by ALTERNATE CHARACTER SET INDEX) <u>shall</u> be defined in CHARACTER SET LIST.
Fill Bundle Index	1
Interior Style	--
Fill Colour	--
Hatch Index	--
Pattern Index	1
Edge Bundle Index	1
Edge Type	1
Edge Width	1.0
Edge Colour	--
Edge Visibility	0 (off)
Fill Reference Point	--

Pattern Table

The PATTERN TABLE element has an unspecified effect when it appears in a picture subsequent to any graphical primitives. The PATTERN TABLE element shall appear prior to any graphical primitive element to assure that interpreting systems without dynamic pattern update can render the intended effect. The minimum support for the length of the Colour Array parameter for the PATTERN TABLE element is 2 048. This will support 8 patterns of 16x16, 2 patterns of 32x32 or 1 pattern of 32x64. All indexes which are used in the metafile shall be defined.

Pattern Size

Colour Table Specification

--

The COLOUR TABLE element has an unspecified effect when it appears in a picture subsequent to any graphical primitives. The COLOUR TABLE element shall appear prior to any graphical primitive elements to assure that interpreting systems without dynamic colour update can render the intended effect. The minimum support for the length of the Colour List parameter in the COLOUR TABLE element is 63. This will support a 64 (0..63) entry colour table. All indexes which are used in the metafile shall be defined.

Aspect Source Flags

Individuals

B.3.7 External elements

Message

The presentation of message string may not be appropriate for all applications. No requirement for the formatted presentation of the message string has been placed on the Interpreter. Only the No Action flag needs to be supported. Support for string lengths up to 254.

Application Data

Support will be provided for strings with a length up to 254 octets, except for data records which will support strings with a length up to 32 767 octets.

B.4 Interoperability with SGML applications

The recommended method for the exchange of documents between Standard Generalized Markup Language (ISO 8879, SGML) based systems and systems based on this ODA document application profile is by means of exchanging a document representation conforming to these agreements in an encoded form of the SGML language known as the Office Document Language (ODL). ODL is a standardized SGML application for representing documents conforming to the ODA base standard. Such a representation may be converted into the Office Document Interchange Format (ODIF) supported by this document application profile.

ANNEX C

Bibliography

(This annex does not form an integral part of this Specification)

CCITT Recommendation T.400 : 1988, Introduction to Document Architecture, Transfer and Manipulation;

CCITT Recommendation T.411 : 1988, Open Document Architecture (ODA) and Interchange Format: Introduction and General Principles;

CCITT Recommendation T.412 : 1988, Open Document Architecture (ODA) and Interchange Format: Document Structures;

CCITT Recommendation T.414 : 1988, Open Document Architecture (ODA) and Interchange Format: Document Profile;

CCITT Recommendation T.415 : 1988, Open Document Architecture (ODA) and Interchange Format: Open Document Interchange Format;

CCITT Recommendation T.416 : 1988, Open Document Architecture (ODA) and Interchange Format: Character Content Architecture;

CCITT Recommendation T.417 : 1988, Open Document Architecture (ODA) and Interchange Format: Raster Graphics Content Architecture;

CCITT Recommendation T.418 : 1988, Open Document Architecture (ODA) and Interchange Format: Geometric Graphics Content Architecture;

CCITT Recommendation T.502 : 1991, Document Application Profile PM-11 for the Interchange of Character Content Documents in Processable and Formatted Forms;

CCITT Recommendation T.505 : 1991, Document Application Profile PM-26 for the Interchange of Mixed Content Documents in Processable and Formatted Forms;

CCITT Recommendation T.506 : (to be published) Document Application Profile PM-36 for the Interchange of Enhanced Mixed Content Documents in Processable and Formatted Forms;

CCITT Recommendation T.50 : 1992 (to be published), International Reference Alphabet;

CCITT Recommendation T.51 : 1992 (to be published), Latin based coded character sets for telematic services;

CCITT Recommendation T.52 : 1992 (to be published), Non-Latin coded character sets for telematic services;

ISO 8571 : 1988, Information processing systems - Open Systems Interconnection - File transfer, access and management;

ISO 9070 : 1991, Information technology - SGML support facilities - Registration procedures for public text owner identifiers;

ISO/TR 9573 : 1988, Information processing - SGML technical report - Techniques for using SGML;

ISO 10021 : 1990, Information processing systems - Text communication - Message Oriented Text Interchange System.

Stable Implementation Agreements for Open Systems Interconnection Protocols: Part 18 - Network Management

Output from the December 1992 Open Systems
Environment Implementors' Workshop (OIW)

SIG Chair
SIG Editor

Paul Brusil, The Mitre Corporation
Robert Aronoff, NIST

Foreword

This part of the Stable implementation Agreements was prepared by the Network Management Special Interest Group (NMSIG) of the Open Systems Environment Implementors' Workshop (OIW). See Procedures Manual for Workshop charter.

Text in this part has been approved by the Plenary of the above-mentioned Workshop. This part replaces the previously existing chapter on this subject.

To highlight textual changes since the last Workshop output, additions to the text in this part are marked with shading; deleted text is left in but marked with strikeouts.

Table of Contents

18	Network Management	1
0	Introduction	1
1	Scope	2
1.1	Phased Approach	2
1.1.1	Alignment With Evolving Standards	2
1.1.2	Definition of Phase 1	2
1.1.3	Future Phases	4
2	Normative References	4
3	Status	9
4	Errata	10
5	Management Functions and Services	12
5.1	General Agreements	12
5.1.1	Conventions Used In SMF Agreements	12
5.1.2	General Agreements Referenced By Many SMF Services	13
5.1.2.1	Maximum Length of Notification Identifier	13
5.1.2.2	Maximum Number of SET Items	13
5.1.2.3	Maximum Length of Additional Text	13
5.1.2.4	Use of Additional Info	13
5.2	Object Management Function Agreements	14
5.2.1	General Agreements	14
5.2.2	Specific Agreements	14
5.3	State Management Function Agreements	14
5.3.1	General Agreements	14
5.3.2	Specific Agreements	14
5.4	Attributes For Representing Relationships Agreements	15
5.4.1	General Agreements	15
5.4.2	Specific Agreements	15
5.5	Alarm Reporting Function Agreements	15
5.5.1	General Agreements	15
5.5.2	Specific Agreements	15
5.6	Event Report Management Function Agreements	16
5.6.1	General Agreements	16
5.6.2	Specific Agreements	16
5.7	Log Control Function Agreements	16
5.7.1	General Agreements	16
5.7.2	Specific Agreements	17
5.8	Security Alarm Reporting Function Agreements	17

	5.8.1	General Agreements	17
	5.8.2	Security Alarm Reporting	17
5.9		Security Audit Trail Function Agreements	18
	5.9.1	General Agreements	18
	5.9.2	Security Audit Trail Reporting SMF Service	18
	5.9.2.1	Notifications	19
	5.9.3	Security Audit Trail Record	19
5.10		Objects and Attributes for Access Control Agreements	19
5.11		Accounting Meter Function Agreements	19
5.12		Workload Monitoring Function Agreements	19
5.13		Summarization Function Agreements	20
5.14		Test Management Function Agreements	20
5.15		Confidence and Diagnostic Test Classes Agreements	20
6		Management Communications	20
6.1		Association Policies	20
	6.1.1	Application Context Negotiation	20
	6.1.2	Functional Unit Negotiation	20
	6.1.3	Security Aspects of Associations	21
7		Management Information	21
7.1		The Information Model	21
	7.1.1	Inheritance	21
	7.1.2	Interoperability	21
	7.1.2.1	Interoperability Provided By The Agent System	21
	7.1.2.2	Interoperability Provided By The Manager System	22
	7.1.3	Filter	22
	7.1.4	Management Operations	22
	7.1.5	Deletion of Objects Containing Objects	22
7.2		Guidelines for the Definition of Management Information	22
	7.2.1	Syntactical Definitions of Management Information	22
	7.2.1.1	Attribute Template	22
	7.2.2	Guidelines For Defining Behaviour	23
	7.2.3	Other Guidelines	23
8		Conformance	24
8.1		Introduction	24
8.2		General Requirements of Conformance	24
8.3		Specific Conformance Categories	25
	8.3.1	Management Communication Categories	25
	8.3.2	Management Functions and Services Conformance Categories	25
	8.3.2.1	General Management Capabilities Conformance Category	25
	8.3.2.2	Alarm Reporting and State Management Capabilities Conformance Category	26
	8.3.2.3	Alarm Reporting Capabilities Conformance Category	27
	8.3.2.4	General Event Report Management Conformance Category	27
	8.3.2.5	General Log Control Conformance Category	28

PART 18: Network Management

December 1992 (Stable)

8.3.3	Management Information Conformance Category	29
8.3.3.1	MOCS Proforma	30
8.3.4	Management Application Contexts	30
8.4	Demonstration of Conformance	30
8.4.1	Management Communication	30
8.4.2	Management Functions and Services	31
8.4.3	Management Information	31
 Annex A (informative)		
Management Information Library (MIL)		32
A.1 Introduction		32
A.2 Rules and Procedures		32
A.3 General Guidelines		32
A.4 Harmonized Library		32
A.4.1 Managed Object Classes and Mandatory Packages		32
A.4.1.1	Computer System	32
A.4.1.2	Connection Oriented Transport Protocol Layer Entity	34
A.4.1.3	Connectionless Network Protocol Layer Entity	36
A.4.1.4	OMNIPoint Equipment	38
A.4.1.5	OMNIPoint Network	39
A.4.1.6	Processing Entity	40
A.4.1.7	Transport Connection	41
A.4.2 Conditional Packages		43
A.4.2.1	Addressing Package	43
A.4.2.2	Checksum PDUs Discarded Package	43
A.4.2.3	Contact List Package	43
A.4.2.4	Contact Name Package	44
A.4.2.5	CPU Utilization Package	44
A.4.2.6	Customer List Package	44
A.4.2.7	Customer Name Package	45
A.4.2.8	Function List Package	45
A.4.2.9	Function Name Package	46
A.4.2.10	Incoming Protocol Error Package	46
A.4.2.11	Location Pointer Package	46
A.4.2.12	Manufacturer List Package	47
A.4.2.13	Manufacturer Name Package	47
A.4.2.14	Max PDU Size IV Package	48
A.4.2.15	Max Retransmissions Package	48
A.4.2.16	Memory Size Package	48
A.4.2.17	Memory Utilization Package	49

A.4.2.18 Octets Retransmitted Package	49
A.4.2.19 OMNIPoint Network List Package	50
A.4.2.20 OMNIPoint Network Name Package	50
A.4.2.21 OMNIPoint Version Package	50
A.4.2.22 Outgoing Protocol Error Package	51
A.4.2.23 PDUs Retransmitted Counter Package	51
A.4.2.24 PDUs Retransmitted Threshold Package	52
A.4.2.25 Peripheral List Package	52
A.4.2.26 Peripheral Name Package	53
A.4.2.27 Processing Entity List Package	53
A.4.2.28 Processing Entity Name Package	53
A.4.2.29 Product Label Package	54
A.4.2.30 Retransmission Time Package	54
A.4.2.31 Retransmission Timer Initial Value Package	55
A.4.2.32 Serial Number Package	55
A.4.2.33 Service List Package	56
A.4.2.34 Service Name Package	56
A.4.2.35 Software List Package	56
A.4.2.36 Software Name Package	57
A.4.2.37 System Time Package	57
A.4.2.38 Type Text Package	58
A.4.2.39 Up Time Package	58
A.4.2.40 Usage State Package	59
A.4.2.41 Vendor List Package	59
A.4.3 Name Bindings	59
A.4.3.1 Computer System Name Bindings	59
A.4.3.2 CO Transport Protocol Layer Entity Name Bindings	60
A.4.3.3 CL Network Protocol Layer Entity Name Bindings	60
A.4.3.4 OMNIPoint Equipment Name Bindings	61
A.4.3.5 OMNIPoint Network Name Bindings	61
A.4.3.6 Processing Entity Name Bindings	62
A.4.3.7 Transport Connection Name Bindings	62
A.4.4 Attributes	63
A.4.4.1 Active Connections	63
A.4.4.2 Addressing Size	63
A.4.4.3 Checksum PDUs Discarded Counter	63
A.4.4.4 Computer System Id	63
A.4.4.5 CL Network Protocol Layer Entity Id	64
A.4.4.6 CO Transport Protocol Layer Entity Id	64
A.4.4.7 Contact List	64
A.4.4.8 Contact Name	64
A.4.4.9 CPU Type	65
A.4.4.10 CPU Utilization	65
A.4.4.11 Customer List	65
A.4.4.12 Customer Name	66

A.4.4.13 Endianness	66
A.4.4.14 Function List	66
A.4.4.15 Function Name	66
A.4.4.16 Inactivity Time	67
A.4.4.17 Inactivity Timeout	67
A.4.4.18 Local Network Address	67
A.4.4.19 Local Network Addresses	68
A.4.4.20 Local Transport Addresses	68
A.4.4.21 Local Transport Connection Endpoint	68
A.4.4.22 Location Pointer	68
A.4.4.23 Manufacturer List	69
A.4.4.24 Manufacturer Name	69
A.4.4.25 Max Connections	69
A.4.4.26 Max PDU Size	70
A.4.4.27 Max Retransmissions	70
A.4.4.28 Memory Size	70
A.4.4.29 Memory Utilization	70
A.4.4.30 Network Entity Type	71
A.4.4.31 Network Title	71
A.4.4.32 NPDU Time To Live	71
A.4.4.33 OMNIPoint Equipment List	71
A.4.4.34 OMNIPoint Network List	72
A.4.4.35 OMNIPoint Network Name	72
A.4.4.36 Operating System Information	72
A.4.4.37 PDUs Forwarded Counter	73
A.4.4.38 PDUs Reassembled Ok Counter	73
A.4.4.39 PDUs Reassembled Fail Counter	73
A.4.4.40 PDUs Discarded Counter	73
A.4.4.41 Peripheral List	74
A.4.4.42 Peripheral Name	74
A.4.4.43 Processing Entity List	74
A.4.4.44 Processing Entity Name	75
A.4.4.45 Product Label	75
A.4.4.46 Remote Network Address	75
A.4.4.47 Remote Transport Connection Endpoint	75
A.4.4.48 Retransmission Time	76
A.4.4.49 Retransmission Timer Initial Value	76
A.4.4.50 Serial Number	76
A.4.4.51 Service List	76
A.4.4.52 Service Name	77
A.4.4.53 Software List	77
A.4.4.54 Software Name	77
A.4.4.55 System Time	78
A.4.4.56 Transport Connection Id	78
A.4.4.57 Transport Connection Reference	78
A.4.4.58 Transport Entity Type	78
A.4.4.59 Type Text	79

A.4.4.60 Up Time	79
A.4.4.61 Vendor List	79
A.4.5 Actions	80
A.4.5.1 Activate	80
A.4.5.2 Deactivate	80
A.4.6 Parameters	81
A.4.6.1 Transport Disconnect Cause	81
A.4.7 Syntax Definitions	81
A.4.8 Inheritance & Naming Trees	84
A.4.8.1 Inheritance Tree	84
A.4.8.2 Naming Tree	84
A.5 OIW NMSIG IVMO Definitions	85
A.5.1 Managed Object Classes and Mandatory Packages	85
A.5.1.1 Transport Connection IVMO	85
A.5.1.2 Transport Connection Retransmission IVMO	85
A.5.2 Name Bindings	86
A.5.2.1 Transport Connection IVMO Name Bindings	86
A.5.2.2 Transport Connection Retransmission IVMO Name Bindings	86
A.5.3 Attributes	86
A.5.3.1 Transport Connection IVMO Id	87
A.5.4 Syntax Definitions	87
A.5.5 Inheritance & Naming Trees	87
A.5.5.1 Inheritance Tree	87
A.5.5.2 Naming Tree	87
A.6 OIW NMSIG Shared Management Knowledge (SMK) Definitions	88
Annex B (informative)	
NMSIG Object Identifiers	89
B.1 Introduction	89
B.2 Harmonized MIL Object Identifiers	89
B.2.1 Object Class Object Identifiers	89

B.2.2 Package Object Identifiers	90
B.2.3 Name Bindings Object Identifiers	91
B.2.4 Attribute Object Identifiers	92
B.2.5 Action Object Identifiers	94
B.2.6 Parameter Object Identifiers	95
B.2.7 Response Code Object Identifiers	95
B.2.8 Module Object Identifiers	95
B.3 Phase 1 MIL Object Identifiers	95
B.3.1 Object Class Object Identifiers	95
B.3.2 Name Bindings Object Identifiers	96
B.3.3 Attribute Object Identifiers	96
B.3.4 Module Object Identifiers	96
Annex C (informative)	
MOCS Proforma	97
C.1 Introduction	97
C.2 Symbols, abbreviations, and terms	97
C.3 Instructions for completing the MOCS proforma to produce a MOCS	97
C.4 Statements of Conformance to Managed Object Classes	97
C.4.2 Connection Oriented Transport Protocol Layer Entity MOCS Proforma	105
C.4.3 ConnectionlessNetwork Protocol Layer Entity MOCS Proforma	111
C.4.4 OMNIPoint Equipment MOCS Proforma	117
C.4.5 OMNIPoint Network MOCS Proforma	126
C.4.6 Processing Entity MOCS Proforma	129
C.4.7 Transport Connection MOCS Proforma	137
C.4.8 Transport Connection IVMO MOCS Proforma	141
C.4.9 Transport Connection Retransmission IVMO MOCS Proforma	145

PART 18: Network Management

December 1992 (Stable)

Annex D (normative)

Management Ensemble Annex 149

List of Tables

Table 1 - Agreements on parameter usage pertinent to the Security Alarm Reporting SMF service . .	17
Table 2 - Agreements on parameter usage pertinent to the Security Audit Trail Reporting SMF service	18
Table B.1 - Object identifiers assigned under "nmsig" node	89
Table B.2 - Object identifiers assigned under "x-objectClass" node	90
Table B.3 - Object identifiers assigned under "x-package" node	90
Table B.4 - Object identifiers assigned under "x-nameBinding" node	92
Table B.5 - Object identifiers assigned under "x-attribute" node	92
Table B.6 - Object identifiers assigned under "x-action" node	94
Table B.7 - Object identifiers assigned under "x-parameter" node	95
Table B.8 - Object identifiers assigned under "x-responseCode" node	95
Table B.9 - Object identifiers assigned under "x-module" node	95
Table B.10 - Object identifiers assigned under "y-objectClass" node	96
Table B.11 - Object identifiers assigned under "y-nameBinding" node	96
Table B.12 - Object identifiers assigned under "y-attribute" node	96
Table B.13 - Object identifiers assigned under "y-module" node	96
Table C.4.1.1 - Name Binding Support	98
Table C.4.1.2 - Attribute Support	99
Table C.4.1.3 - Attribute Group Support	101
Table C.4.1.5 - Notification Support	102
Table C.4.2.1 - Name Binding Support	105
Table C.4.2.2 - Attribute Support	106
Table C.4.2.3 - Attribute Group Support	108
Table C.4.2.5 - Notification Support	109
Table C.4.3.1 - Name Binding Support	111
Table C.4.3.2 - Attribute Support	112
Table C.4.3.3 - Attribute Group Support	114
Table C.4.3.5 - Notification Support	115
Table C.4.4.1 - Name Binding Support	118
Table C.4.4.2 - Attribute Support	119
Table C.4.4.3 - Attribute Group Support	121
Table C.4.4.5 - Notification Support	122
Table C.4.5.1 - Name Binding Support	126
Table C.4.5.2 - Attribute Support	127
Table C.4.5.5 - Notification Support	128
Table C.4.6.1 - Name Binding Support	129
Table C.4.6.2 - Attribute Support	130
Table C.4.6.3 - Attribute Group Support	132
Table C.4.6.5 - Notification Support	133
Table C.4.7.1 - Name Binding Support	137
Table C.4.7.2 - Attribute Support	138
Table C.4.7.5 - Notification Support	140
Table C.4.8.1 - Name Binding Support	142
Table C.4.8.2 - Attribute Support	143

PART 18: Network Management

December 1992 (Stable)

Table C.4.8.5 - Notification Support 144

Table C.4.9.1 - Name Binding Support 145

Table C.4.9.2 - Attribute Support 146

Table C.4.9.5 - Notification Support 147

18 Network Management

0 Introduction

Within the community of OSI researchers, users, and vendors, there is a recognized need to address the problems of initiating, terminating, monitoring, and controlling communication activities and assisting in their harmonious operation, as well as handling abnormal conditions. The activities that address these problems are collectively called network management.

Network management can be viewed as the set of operational and administrative mechanisms necessary to:

- a. bring up, enroll, and/or alter network resources;
- b. keep network resources operational;
- c. fine tune these resources and/or plan for their expansion;
- d. manage the accounting of their usage;
- e. manage their protection from unauthorized use/tampering.

As such, network management is typically concerned with management activities in at least the following five functional areas: configuration management, fault management, performance management, accounting management, and security management. In order to accomplish these management activities, information must be exchanged among open systems.

In Part 18, there are Implementation Agreements (IA's) for providing interoperable OSI management information communication services among OSI systems. Also contained here are agreements on management information. These agreements pertain to the exchange of management information and management commands between open systems operating in a multivendor environment. For example, one goal is to ensure that a management system built by one vendor can manage objects built by another vendor.

1 Scope

The purpose of this Part (Part 18), is to provide implementation agreements that will enable independent vendors to supply customers with a diverse set of networking products that can be managed as part of an integrated environment. Where possible, these agreements are based upon OSI Systems Management standards.

1.1 Phased Approach

Because of the broad scope of the subject, and given that OSI Systems Management standards are still evolving, it is reasonable to assume that a comprehensive set of network management implementation agreements will take a number of years to develop. To arrive at an initial set of implementation agreements in a timely fashion, a phased approach has been adopted.

This phased work approach will result in a series of implementation agreements based on the expanding scope of the OSI Systems Management standards. It is the intention of the NMSIG to define the content of each phase as a compatible superset of the previous Phases to ensure that Phase N products can interact with products based on the implementation agreements of earlier phases.

1.1.1 Alignment With Evolving Standards

In some cases, these phased implementation agreements may be based on DIS standards. As the relevant standards progress from DIS to IS, the agreements will be aligned in future phases.

When a defect is found in any of the management related standards, the reported defect may be technically resolved by the appropriate international technical committee with likely approval by the voting members pending for several months. Since relevant defects can't be ignored in an implementation, these agreements will note defect resolutions which have the tentative approval of the appropriate standards committee. These Interim resolutions will be recorded in clause 4.

Once a defect resolution has been completed by the appropriate standards body, the agreed upon resolution will be incorporated into the next phase of these implementors agreements. If appropriate, a previous phase that relied on an interim resolution will be examined to determine whether errata should be issued to bring the original phase into line with the final resolution.

1.1.2 Definition of Phase 1

As a first step in this phased approach, the NMSIG has targeted an initial set of agreements that provide limited interoperable management in a heterogeneous vendor environment. They are the beginning of a comprehensive set of implementation agreements based on the emerging OSI Systems Management standards. Furthermore, these initial agreements allow the community to gain experience with OSI management standards as they emerge.

The focus of the Phase 1 agreements is to enable a managing process provided by one vendor to Interoperate with an agent process provided by a different vendor to perform limited management on a set of managed objects.

The scope of Phase 1 implementation agreements is the following:

Management Functions:

Object Management Function [OMF],
State Management Function [STMF],
Attributes For Representing Relationships [ARR],
Alarm Reporting Function [ARF],
Event Report Management Function [ERMF].

Management Information:

Information Model, Naming, Guidelines and Templates for Defining Managed Objects

Management Communication:

CMIS/P, Association Policies, and Upper Layer Services Required

Management Objects:

Support Objects required for the above.

Editor's Note: [The relation of the MIL definitions in Annex A of the Working Document to Phase 1 IA's needs to be clarified.]

Conformance Criteria:

Conformance Criteria for the above functionality.

To accomplish these goals in a timely fashion, the following simplifying constraints have been reflected in the Phase 1 agreements:

1. No agreements are provided regarding management domains;
2. These agreements require only the following application service elements: the Association Control Service Element (ACSE), the Common Management Information Service Element (CMISE), Remote Operations Service Element (ROSE), and the System Management Application Service Element (SMASE);
3. These agreements do not require implementation of services defined by the Directory standards;

4. No agreements regarding the security of management are provided.

1.1.3 Future Phases

It is the intention of the NMSIG to freeze the content of Phase 1 when these agreements are progressed to Stable status. Alignment changes required as the standards progress from DIS to IS will be made in future phases.

As standards defining new functionality are progressed, the NMSIG will define future phases incorporating the new functionality as a compatible superset of previous phases.

2 Normative References

The following documents are referenced in the statements of the agreements relating to OSI systems management.

Editor's Note: [Items marked with an asterisk, "*", are ones which, while not cited in the text of this part of the IAs, are included here, nevertheless, to indicate where useful background information can be found.]

- | | |
|-----------|---|
| [ACSEP] | ISO 8650, Information Processing Systems - Open Systems Interconnection - Protocol Specification for the Association Control Service Element (Revised Final Text of DIS 8650), ISO/IEC JTC1/SC21 N2327, 21 April 1988. |
| [ACSES]* | ISO 8649, Information Processing Systems - Open Systems Interconnection - Service Definition for the Association Control Service Element (Revised Final Text of DIS 8649), ISO/IEC JTC1/SC21 N2326, 21 April 1988. |
| [ADDRMVP] | ISO/IEC 9596/DAD 2, Common Management Information Protocol Specification: Addendum 2 (Add/Remove Protocol), ISO/IEC JTC1/SC21, 1 February 1990. |
| [ADDRMVS] | ISO/IEC 9595/DAD 2, Common Management Information Service Definition: Addendum 2 (Add/Remove Service), ISO/IEC JTC1/SC21, 1 February 1990. |
| [ALS]* | ISO/IEC DIS 9545, Information Processing Systems - Open Systems Interconnection - Application Layer Structure, 15 March 1989. |
| [AOM1PT1] | ISO/IEC ISP 11183-1, Information Technology - International Standardized Profiles AOM1n OSI Management - Management Communications - Part 1: Specification of ACSE, Presentation and Session Protocols for the use by ROSE and CMISE, May 1992. |
| [AOM1PT3] | ISO/IEC ISP 11183-3, Information Technology - International Standardized Profiles AOM1n OSI Management - Management Communications - Part 3: CMISE/ROSE for AOM11 - Basic Management Communications, May 1992. |

PART 18: Network Management

December 1992 (Stable)

- [AOM1PT2] ISO/IEC ISP 11183-2, Information Technology - International Standardized Profiles AOM1n OSI Management - Management Communications - Part 2: CMISE/ROSE for AOM12 - Enhanced Management Communications, May 1992.
- [AOM211] pDISP 12060-1, Information Technology - International Standardized Profiles - AOM2n OSI Management - Management Functions - Part 1: AOM211 - General Management Capabilities, July 1992.
- [AOM212] pDISP 12060-2, Information Technology - International Standardized Profiles - AOM2n OSI Management - Management Functions - Part 2: AOM212 - Alarm Reporting and State Management Capabilities, July 1992.
- [AOM213] pDISP 12060-3, Information Technology - International Standardized Profiles - AOM2n OSI Management - Management Functions - Part 3: AOM213 - Alarm Reporting Capabilities, July 1992.
- [AOM221] pDISP 12060-4, Information Technology - International Standardized Profiles - AOM2nn OSI Management - Management Functions - Part 4: AOM221 - General Event Report Management, July 1992.
- [AOM231] pDISP 12060-5, Information Technology - International Standardized Profiles - AOM2n OSI Management - Management Functions - Part 5: AOM231 - General Log Control Profile, July 1992.
- [ARF] ISO/IEC IS 10164-4, Information Technology - Open Systems Interconnection - Systems Management - Part 4: Alarm Reporting Function, ISO/IEC JTC1/SC21 N6359, August 19, 1991.
- [ARR] ISO/IEC IS 10164-3, Information Technology - Open Systems Interconnection - Systems Management - Part 3: Attributes for Representing Relationships, ISO/IEC JTC1/SC21 N5186, September 1991.
- [ASN1]* ISO/IEC 8824, Information Technology - Open System Interconnection - Specification of Abstract Syntax Notation One (ASN.1), ISO/IEC JTC1/SC21 N4720, 30 April 1990.
- [BER] ISO/IEC 8825, Information Technology - Open Systems Interconnection - Specification of Basic Encoding Rules for Abstract Syntax Notation One (ASN.1), ISO/IEC JTC1/SC21 N4721, 30 April 1990.
- [CANGETP] ISO/IEC 9596/DAD 1, Common Management Information Protocol Specification: Addendum 1 (CancelGet Protocol), ISO/IEC JTC1/SC21, 1 February 1990.
- [CANGETS] ISO/IEC 9595/DAD 1, Common Management Information Service Definition: Addendum 1 (CancelGet Service), ISO/IEC JTC1/SC21, 1 February 1990.

PART 18: Network Management

December 1992 (Stable)

[CMIP]	ISO/IEC 9596-1, Information Technology - Open Systems Interconnection - Common Management Information Protocol Specification - Part 1: Specification, 24 November 1990.
[CMIS]	ISO/IEC 9595, Information Technology - Open Systems Interconnection - Common Management Information Service Definition, Common Management Information Service, 24 November 1990.
[DIR]*	ISO 9594 - Information Processing Systems - Open Systems Interconnection - The Directory, 1988.
[DMI]	ISO/IEC IS 10165-2, Information Technology - Open Systems Interconnection - Structure of Management Information - Part 2: Definition of Management Information, ISO/IEC JTC1/SC21 N6363, August 1991.
[ENSCON]*	Forum 025, The "Ensemble" Concepts and Format, Issue 1.0, Network Management Forum, July 1992.
[ERMF]	ISO/IEC IS 10164-5, Information Technology - Open Systems Interconnection - Systems Management - Part 5: Event Report Management Function, ISO/IEC JTC1/SC21 N6360, August 1991.
[FRMWK]*	ISO 7498-4, Information Processing Systems - Open Systems Interconnection - Basic Reference Model - Part 4: Management Framework, 1989.
[GDMO]	ISO/IEC IS 10165-4, Information Technology - Open Systems Interconnection - Structure of Management Information - Part 4: Guidelines for the Definition of Managed Objects, ISO/IEC JTC1/SC21 N6309, July 30, 1991.
[ISPARR3]	pDISP 12059-3, Information Technology - International Standardized Profiles - OSI Management - Common Information for Management Functions - Part 3: Attributes for Representing Relationships, July 1992.
[ISPAR4]	pDISP 12059-4, Information Technology - International Standardized Profiles - OSI Management - Common Information for Management Functions - Part 4: Alarm Reporting, July 1992.
[ISPCOM0]	pDISP 12059-0, Information Technology - International Standardized Profiles - OSI Management - Common Information for Management Functions - Part 0: Common Definitions for Management Function Profiles, July 1992.
[ISPERM5]	pDISP 12059-5, Information Technology - International Standardized Profiles - OSI Management - Common Information for Management Functions - Part 5: Event Report Management, July 1992.
[ISPFM]	ISO/IEC TR 10000-1, Information Technology - Framework and Taxonomy of International Standardized Profiles - Part 1: Framework, ISO/IEC JTC1/SGFS N184, 9 February 1990.

PART 18: Network Management

December 1992 (Stable)

- [ISPLC6] pDISP 12059-6, Information Technology - International Standardized Profiles - OSI Management - Common Information for Management Functions - Part 6: Log Control, July 1992.
- [ISPOM1] pDISP 12059-1, Information Technology - International Standardized Profiles - OSI Management - Common Information for Management Functions - Part 1: Object Management, July 1992.
- [ISPSRVC] ISO/IEC TR 8509, Information Processing Systems - Open Systems Interconnection - Service Conventions, TC97/SC16/1646.
- [ISPSTM2] pDISP 12059-2, Information Technology - International Standardized Profiles - OSI Management - Common Information for Management Functions - Part 2: State Management, July 1992.
- [LCF] ISO/IEC IS 10164-6, Information Technology - Open Systems Interconnection - Systems Management - Part 6: Log Control Function, ISO/IEC JTC1/SC21 N6361, June 1991.
- [MGNM]* CCITT Recommendation M.gnm, Draft Recommendation (M.gnm) Generic Network Information Model, CCITT SGIV, December 3, 1991.
- [MIM] ISO/IEC IS 10165-1, Information Technology - Open Systems Interconnection - Management Information Services - Structure of Management Information - Part 1: Management Information Model, ISO/IEC JTC1/SC21 N6351, June 1991.
- [NMSIG1] OIW Endorsement/Comment on System Management Function Taxonomy (Including Proposed Function Taxonomy), NMSIG-91/164, September 1991.
- [OMF] ISO/IEC IS 10164-1, Information Technology - Open Systems Interconnection - Systems Management - Part 1: Object Management Function, ISO/IEC JTC1/SC21 N5184, September 1991.
- [OP1LIB]* Forum 006, Forum Library - Volume 4: OMNIPoint 1 Definitions, Issue 1.0, Network Management Forum, August 1992.
- [PPS]* ISO/IEC DIS 8823, Information Processing Systems - Open Systems Interconnection - Connection Oriented Presentation Protocol Specification, ISO/IEC JTC1/SC21 N2336, 5 April 1988.
- [PSD]* ISO/IEC Final Text of DIS 8822, Information Processing Systems - Open Systems Interconnection - Connection Oriented Presentation Service Definition, ISO/IEC JTC1/SC21 N2335, 5 April 1988.
- [ROSEP]* ISO/IEC 9072-2 - Information Processing Systems - Text Communications - Remote Operations Part 2: Protocol Specification, 19 September 1989.

PART 18: Network Management

December 1992 (Stable)

- [ROSES]*** ISO/IEC 9072-1, Information Processing Systems - Text Communications - Remote Operations Part 1: Model, Notation and Service Definition, 19 September 1989.
- [SARF]** ISO/IEC IS 10164-7, Information Technology - Open Systems Interconnection - Systems Management - Part 7: Security Alarm Reporting Function, July 1991.
- [SATF]** ISO/IEC DIS 10164-8, Information Technology - Open Systems Interconnection - Systems Management - Part 8: Security Audit Trail Function, ISO/IEC JTC1/SC21 N7039, June 1992.
- [SMO]** ISO/IEC IS 10040, Information Technology - Open Systems Interconnection - Systems Management Overview, ISO/IEC JTC1/SC21 N6353, August 1991.
- [STMF]** ISO/IEC IS 10164-2, Information Technology - Open Systems Interconnection - Systems Management - Part 2: State Management Function, ISO/IEC JTC1/SC21 N5185, September 1991.

3 Status

As of September 1991, the Stable management communications agreements in clause 6 of part 18 and clause 13.7 of part 5 became technically equivalent to DISP 11183. The DISP, however, is a more rigorous statement of specifications. Therefore, it has been the stated intent of the NMSIG to directly reference the ISP 11183, Parts 1 through 3, and all the agreements therein, when the DISP reaches ISP status. Since the DISP has now progressed to ISP 11183 with no technical changes, the NMSIG Stable management communications agreements in clause 6 of part 18 have now been changed to point directly to ISP 11183-1 through -3 [AOM1PT1, AOM1PT2, and AOM1PT2].

(Refer to the Working Implementation Agreements Document for additional status information.)

4 Errata

Editor's Note: ["Defect Report" material (including applicability) may be included here.]

The following table indicates the clause, type, and reference document of technical errata to this part.

Erratum No.	Type & Date Entered	Referenced Document	Clause	Comment
1	Technical 6/91	NMSIG-91/08	6.4.5	This clause, previously clause 6.2.6, was modified and moved to clause 6.4.5 to clarify that it is intended as a support agreement for CMIP rather than a usage agreement for CMIS.
2	Alignment 9/91	NMSIG-91/110 NMSIG-91/113	5	This clause has been updated to reflect alignment changes to the relevant base standards which have just progressed to IS as of August, 1991.
3	Technical 9/91	NMSIG-91/161	6.2.2.2	Move text from clause 5.1.2.1 to more appropriate clause 6.2.2.2 and clarify required support for minimal filter complexity to align with the DISP 11183.
4	Technical 9/91	NMSIG-91/161	6.2.3	Remove unnecessary restrictions on sending CMIP time parameters.
5	Alignment 9/91	NMSIG-91/114	6.1.1	Change reference to required application context support to align with IS version of [SMO].
6	Technical 9/91	NMSIG-91/161	6.3.3.1	Remove clause requiring mandatory attribute list in successful set response because considered redundant information.

Erratum No.	Type & Date Entered	Referenced Document	Clause	Comment
7	Alignment 9/91	NMSIG-91/120	7	Update clause to reflect alignment changes to the relevant base standards which have progressed to IS as of August, 1991.
8	Editorial 9/91	NMSIG-91/161	6.3.6.1	Move clause 6.3.6.1 to more appropriate location at clause 7.1.5.
9	Alignment 3/92	NMSIG-92/066	6.1.3	Update reference because number of clause in other part of OIW Stable Agreements changed.
10	Alignment 6/92	NMSIG-92/093	5.2 - 5.7	Update text to reference appropriate AOM2x pDISPs because have equivalent agreements, but are more rigorous.
11	Alignment 6/92	NMSIG - 92/200	6	Update text to reference ISP 11183 which is technically equivalent with IA text but is more rigorous.
12	Technical 12/92	NMSIG-92/409	A.5.1.2	Modify package name, transportConnectionRetransmissionIV MO-Package, which was incorrectly specified in the CHARACTERIZED BY clause of the MO class definition.
13	Technical 12/92	NMSIG-92/409	A.5.1.2	Modify object ID in REGISTERED AS clause of the MO class definition to register the newly modified MO (see erratum 12).
14	Technical 12/92	NMSIG-92/409	B.3.1	Modify object identifier value in TABLE B.10 to reflect changes in MO definition (see errata 12 and 13).
15	Technical 12/92	NMSIG-92/409	C.4.9	Modify object identifier value in the table to reflect changes in MO definition (see errata 12 and 13).

5 Management Functions and Services

5.1 General Agreements

5.1.1 Conventions Used In SMF Agreements

Each System Management Function defines a set of services referred to in this document as "SMF services." Agreements pertinent to SMF services are provided in the following subclauses. Each subclause contains a series of tables, as follows.

For each SMF service, a normative table references text agreements which constrain the usage and/or value of the associated service parameters. Text agreements defined elsewhere in this document are referenced by clause number. The lack of a row or reference signifies no agreement beyond the base standard.

These tables include codes which specify parameter usage for request, indication, response, and confirmation service primitives. These codes, defined in subclause 1.8.3 of these agreements (Classification of Conformance), in ISO/IEC TR 10000-1 (Framework and Taxonomy of ISPs) [ISPFRM], and in ISO/IEC TR 8509 (Service Conventions) [ISPSRVC], are repeated here for reader convenience:

M	Mandatory
O	Optional
C(p)	If Condition p exists, then parameter is mandatory; otherwise, the parameter is not applicable.
X	Excluded
I	Out Of Scope
	In these agreements, this means that, for the corresponding element,
	* implementations may use it outside the scope of these agreements,
	* conformance tests shall not be provided for it,
	* implementations may conform to other agreements where it is required,
	* no requirements are placed on either transmitter or receiver to support it,
	* receiver actions are unspecified when present.
-	Not Applicable
(=)	The value of the parameter is identical to the corresponding parameter in the interaction described by the preceding related service primitive.
U	The use of the parameter is a service-user option.
P	The parameter is mapped directly onto the corresponding parameters of the CMIS service primitive; refer to subclause 6 for agreements regarding this pass-through parameter.

In addition, the convention "A > B" is used in normative tables to indicate both the usage specified by the base standard (A) and the additional constraint imposed by these agreements (B). This convention is intended to call attention to agreements which modify the usage of a service parameter.

Unless otherwise noted, conditional parameters (C) shall be present according to the conditions defined in [CMIS] and the referenced System Management Function base standard.

5.1.2 General Agreements Referenced By Many SMF Services

The following general agreements pertain to some or all of the System Management Function services defined throughout clause 5. Normative tables for each SMF service reference these general agreements where applicable. These agreements do not apply to SMF services and parameters which do not reference them.

5.1.2.1 Maximum Length of Notification Identifier

To limit implementation complexity, the maximum length of the Notification Identifier parameter shall be 32 bits.

5.1.2.2 Maximum Number of SET Items

To limit implementation complexity, the maximum number of SET items contained within specified SMF service parameters that recipients must be able to process shall be 64.

5.1.2.3 Maximum Length of Additional Text

To limit implementation complexity, the maximum length of the Additional Text parameter which recipients must be able to process shall be 256 octets.

5.1.2.4 Use of Additional Info

Editor's Note: [The Additional Information parameter, described in [ARF] clause 8.1.2.14, includes a "significance indicator." It requires that "[e]ven if the Additional Information parameter is not fully understood, an event report indication shall be issued to the user. Indication that the Additional Information parameter is not fully understood is a local matter."]

5.2 Object Management Function Agreements

5.2.1 General Agreements

These agreements require support for the SMF services defined by the object management standard [OMF].

These agreements also require conformance to the abstract syntaxes identified in clause 11 of the object management standard [OMF] and specified in [DMI], with the exception of event record subclasses. If support for the log control standard [LCF] as described in clause 5.7 is claimed, then all [OMF] event record subclasses shall also be required by these agreements. These agreements permit optional negotiation of the system management functional units specified in clause 10 of [OMF].

5.2.2 Specific Agreements

See [ISPOM1] for specification of agreements for the Object Management Function.

5.3 State Management Function Agreements

5.3.1 General Agreements

These agreements require support for the SMF services defined by the state management standard [STMF].

These agreements also require conformance to the abstract syntaxes identified in clause 11 of the state management standard [STMF] and specified in [DMI], with the exception of event record subclass. If support for the log control standard [LCF] as described in clause 5.7 is claimed, then all [STMF] event record subclasses shall also be required by these agreements. These agreements permit optional negotiation of the State Change Reporting functional unit specified in clause 10 of [STMF].

5.3.2 Specific Agreements

See [ISPSTM2] for specification of agreements for the State Management Function.

5.4 Attributes For Representing Relationships Agreements

5.4.1 General Agreements

These agreements require support for the SMF services defined by the Attributes For Representing Relationships standard [ARR].

These agreements also require conformance to the abstract syntaxes identified in clause 11 of the attributes for representing relationships standard [ARR] and specified in [DMI], with the exception of event record subclass. If support for the log control standard [LCF] as described in clause 5.7 is claimed, then all [ARR] event record subclasses shall also be required by these agreements. These agreements permit optional negotiation of the Relationship Change Reporting functional unit specified in clause 10 of [ARR].

5.4.2 Specific Agreements

See [ISPARR3] for specification of agreements for Attributes for Representing Relationships.

5.5 Alarm Reporting Function Agreements

5.5.1 General Agreements

These agreements require support for the SMF services defined by the alarm reporting standard [ARF].

These agreements also require conformance to the abstract syntaxes identified in clause 11 of the alarm reporting standard [ARF] and specified in [DMI], with the exception of event record subclass. If support for the log control standard [LCF] as described in clause 5.7 is claimed, then all [ARF] event record subclasses shall also be required by these agreements. These agreements permit optional negotiation of the Alarm Reporting functional unit specified in clause 10 of [ARF].

5.5.2 Specific Agreements

See [ISPAR4] for specification of agreements for the Alarm Reporting Function.

5.6 Event Report Management Function Agreements

5.6.1 General Agreements

These agreements require support for the SMF services defined by the event report management standard [ERMF].

These agreements also require conformance to the abstract syntaxes identified in clause 11 of the event report management standard [ERMF] and specified in [DMI]. These agreements permit optional negotiation of the Monitor Event Report Management and Event Report Management functional units specified in clause 10 of [ERMF].

5.6.2 Specific Agreements

See [ISPERM5] for specification of agreements for the Event Report Management Function.

5.7 Log Control Function Agreements

5.7.1 General Agreements

These agreements require the SMF services defined by the log control standard [LCF].

These agreements also require conformance to the abstract syntaxes identified in clause 11 of the log control standard [LCF] and specified in [DMI].

If any other function defined in clause 5 that supports notifications is supported, then any event record subclass defined by that function is required for the log control function.

These agreements permit optional negotiation for log control and monitor log control functional units specified in section 10 of [LCF].

The appropriate CMIS error (i.e., `invalidAttributeValue`) shall be returned for any attempt to set Max log size less than the value of Current log size, except if setting the Max log size to zero. When the Max log size is set to zero, then the maximum log size is unlimited.

5.7.2 Specific Agreements

See [ISPLC6] for specification of agreements for the Log Control Function.

5.8 Security Alarm Reporting Function Agreements

5.8.1 General Agreements

These agreements require support for the SMF services defined by the security alarm reporting standard [SARF].

These agreements also require conformance to the abstract syntaxes identified in clause 11 of the alarm reporting standard [SARF] and specified in [DMI], with the exception of event record subclass. If support for the log control standard [LCF] as described in clause 5.7 is claimed, then all [SARF] event record subclasses shall also be required by these agreements. These agreements permit optional negotiation of the security alarm reporting function as specified in section 10 of [SARF].

5.8.2 Security Alarm Reporting

This subclause provides agreements pertinent to the Security Alarm Reporting SMF service defined by section 9.2 of [SARF]. Subclause 6 provides agreements pertinent to CMIS services and pass-through parameters used by this SMF service.

Table 1 - Agreements on parameter usage pertinent to the Security Alarm Reporting SMF service

SMF Security Alarm Reporting parameter	Req	Rsp	SMF agreements
Event Type	M	C(=)	
Event Information			
Security Alarm Cause	M	-	
Security Alarm Severity	M	-	
Security Alarm Detector	M	-	[1]
Service User	M	-	
Service Provider	M	-	
Notification Identifier	U	-	5.1.2.1

SMF Security Alarm Reporting parameter	Req	Rsp	SMF agreements
Correlated Notifications	U	-	5.1.2.2
Additional Text	U	-	5.1.2.3
Additional Info	U	-	5.1.2.2, 5.1.2.4

- [1] To avoid ambiguity, the Distinguished Name form of this parameter shall be implemented and may be used. Use of Local Distinguished Name and Non-Specific forms are beyond the scope of these agreements. If an implementation is unable to decode or understand the semantics of this parameter, an appropriate CMIS error (i.e., Invalid Attribute Value) shall be returned.

5.9 Security Audit Trail Function Agreements

5.9.1 General Agreements

These agreements require support for the SMF services defined by the security audit trail standard [SATF].

These agreements also require conformance to the abstract syntaxes identified in clause 11.2 of the security audit trail standard [SATF] and specified in [DMI], with the exception of event log record subclass. If support for the log control standard [LCF] as described in clause 5.7 is claimed, then all [SATF] event log record subclasses shall also be required by these agreements.

5.9.2 Security Audit Trail Reporting SMF Service

This subclause provides agreements pertinent to the Security Audit Trail Reporting SMF service defined by section 9.2 of [SATF]. Clause 6 provides agreements pertinent to CMIS services and pass-through parameters used by this SMF service.

Table 2 - Agreements on parameter usage pertinent to the Security Audit Trail Reporting SMF service

SMF Security Audit Trail parameter	Req	Rsp	SMF agreements
Event Type	M	C(=)	5.9.2.1
Event Information			
Service Report Cause	C(1)	-	

SMF Security Audit Trail parameter	Req	Rsp	SMF agreements
Notification Identifier	U	-	5.1.2.3
Correlated Notifications	U	-	5.1.2.4
Additional Text	U	-	5.1.2.5
Additional Info	U>I	-	5.1.2.6

C(1): Mandatory (M) for serviceReport

5.9.2.1 Notifications

These Implementors' Agreements require support for both the serviceReport and usageReport notification types.

5.9.3 Security Audit Trail Record

This subclause is a placeholder for agreements pertaining to the Security Audit Trail Record (SATR) managed object class.

5.10 Objects and Attributes for Access Control Agreements

(Refer to the Working Implementation Agreements Document.)

5.11 Accounting Meter Function Agreements

(Refer to the Working Implementation Agreements Document.)

5.12 Workload Monitoring Function Agreements

(Refer to the Working Implementation Agreements Document.)

5.13 Summarization Function Agreements

(Refer to the Working Implementation Agreements Document.)

5.14 Test Management Function Agreements

(Refer to the Working Implementation Agreements Document.)

5.15 Confidence and Diagnostic Test Classes Agreements

(Refer to the Working Implementation Agreements Document.)

6 Management Communications

This clause covers the agreements pertaining to the use of associations over which to conduct management communications, and agreements for management communication, itself, by reference to ISP 11183 [AOM1PT1, AOM1PT2, and AOM1PT3]. ISP 11183 defines two profiles, AOM11 (Basic Management Communications) [AOM1PT3] and AOM12 (Enhanced Management Communications) [AOM1PT2], and defines upper layer requirements [AOM1PT1] for each of these profiles.

For rigorous specification of the agreements relevant to clause 6, Management Communications, see ISP 11183 [AOM1PT1, AOM1PT2, and AOM1PT3].

6.1 Association Policies

Associations are established using the procedures described in [ACSEP].

6.1.1 Application Context Negotiation

These IAs specify the negotiation of the Systems management application context specified in [SMO]. Other application contexts are outside the scope of these agreements.

6.1.2 Functional Unit Negotiation

These IAs specify that System Management Functional Units are negotiated as specified in [SMO].

6.1.3 Security Aspects of Associations

The ACSE authentication mechanisms and associated data types shall be as defined in clause 9 (Upper Layers Security) of part 12 of the OIW Working Agreements.

Support of ACSE authentication is optional.

7 Management Information

This clause, which is based on ISO standards' documents [MIM] and [GDMO], contains agreements regarding basic concepts and modelling techniques related to management information. It enumerates agreements on (i) the information model (subclause 7.1) and (ii) guidelines for defining management information (subclause 7.2). These agreements apply to developers of contributions to the Management Information Library (MIL). They form a normative part of the standard; hence they must be strictly followed while defining management information. It is not within the scope of this clause to make agreements about specific elements of management information or to define such specific elements of management information. Such definitions and/or agreements can be obtained via the Management Information Library.

7.1 The Information Model

When modelling management information, these agreements require use of [MIM] with the following additional constraints.

7.1.1 Inheritance

The following constraint related to inheritance is enforced in order to remove potential ambiguities:

During the lifetime of a managed object instance, each of its attributes must have a value that is valid for the attribute syntax of that attribute.

7.1.2 Interoperability**7.1.2.1 Interoperability Provided By The Agent System**

Allomorphism, as specified in clause 5.2.3.1 of [MIM], is out of scope. Any other specification within the [MIM] or [GDMO] that refers to allomorphism is also out of scope.

7.1.2.2 Interoperability Provided By The Manager System

The semantics of clause 5.2.3.2 of [MIM] are supported. A manager system can supply the object identifier as specified in clause 7.4.5 of [GDMO] to specify that a managed object should perform an operation as a member of its actual class. The object identifier is intended to be used in requests only, and shall be interpreted by the responder as a requirement to return its real object class value in the response. Agent systems shall support this object identifier as defined in [MIM] 5.2.3.2 and [GDMO] 7.4.5.

7.1.3 Filter

The concept of filter is supported as specified in clause 6. Restrictions on its usage are specified in subclause 6.2.2.2 of these agreements.

7.1.4 Management Operations

An implementation that complies with these agreements shall support management operations as defined in clause 5.3.4 of [MIM] with the following additional clarification.

[MIM] clause 5.3.4.1 (2), [DMI] clause 6.14, and [GDMO] clause 6.1.4 imply that the object class attribute shall not be included in the create request Attribute List parameter. [MIM] states that any conflicting duplicate specifications cause the request to fail.

7.1.5 Deletion of Objects Containing Objects

The error 'Processing Failure' shall be returned if a managed object has existing contained objects and the behavior defined for that object prohibits its deletion unless all contained objects have been deleted.

7.2 Guidelines for the Definition of Management Information

This subclause contains agreements about guidelines for the definition of management information, as specified in [GDMO].

7.2.1 Syntactical Definitions of Management Information**7.2.1.1 Attribute Template**

The following constraint applies to the Attribute Template specified in clause 9.7.2 of [GDMO]:

The BEHAVIOUR construct may be omitted only if a behaviour definition has been inherited from the parent attribute, i.e., the attribute is derived from another attribute whose definition contains a BEHAVIOUR construct.

7.2.2 Guidelines For Defining Behaviour

The following details should be provided in the set of specifications defining a managed object class:

- a) a textual description of the network/system resource(s) the managed object class represents, including their functional role;
- b) a description of the relationships that can occur between different instances of the managed object class being defined, as well as those that can occur between instances of the managed object class being defined and instances of other managed object classes;
- c) a description of the operations that are supported by the managed object class, with precise definition of the effects, side effects if any, constraints, response notifications, failure modes;
- d) specification of how instances of this managed object class are created and deleted, particularly whether they can be created/deleted via the management CREATE/DELETE operations;
- e) a description of notifications that can be generated, the conditions that generate them (e.g., crossing of a threshold), their contents and side-effects, if any. In particular, identify all the attributes that are subject to the AttributeChange and StateChange notifications, if these notifications are supported;
- f) other constraints, including those involving other managed object classes.

7.2.3 Other Guidelines

The Systems Management functions have defined various attributes and events, as indicated in clause 5 of these agreements. Object definers should make use of these attributes and events wherever applicable.

8 Conformance

8.1 Introduction

Clause 8 specifies the conformance requirements for the OIW Network Management Implementation Agreements (IAs). Implementors of products will provide claims of conformance to these requirements. These claims will be in the form of Protocol Implementation Conformance Statements (PICS) and Managed Object Conformance Statements (MOCS). These requirements will also be used to develop test cases which will be used to validate claims of conformance. This clause defines the general conformance requirements and criteria which shall be used as a basis for tests of implementations claiming conformance to these agreements. Dependent conformance requirements are defined in the context in which they are used (e.g., SMF general conformance requirements include CMIP dependent conformance requirements for CMIS services used).

Editor's Note: [The use of the two terms, "general conformance class" and "dependent conformance class", is under review. When a final answer to Question Q1/49.9 (on the long term solution to general and dependent conformance) has been approved, it is intended to clarify and/or correct this conformance section.]

(Refer to the Working Implementation Agreements Document for additional introductory text.)

8.2 General Requirements of Conformance

Conformance for these agreements is designed to specify a well-defined set of management capabilities and features. For the purposes of organization and clarity of these agreements, management has been divided into three categories. Clauses 5 (Management Functions and Services), 6 (Management Communications) and 7 (Management Information) state the agreements which respectively comprise three conformance categories. Within these categories, particular conformance categories are specified which delineate conformance requirements for a well-defined and bounded set of management capabilities and features (e.g., within the Management Functions and Services conformance categories, a conformance category is specified which defines conformance to Alarm Reporting and State Management Services). Once a conformance category is delineated which specifies the set of requirements for that category, tests can be developed to evaluate conformance of products to that conformance category. And finally, for some conformance categories, roles (Manager, Agent, or both) are specified. One or more roles may be supported for those conformance categories to which an implementation is conformant.

The development of conformance categories will enable:

- a) users to define procurement specifications;
- b) vendors to define management capabilities and features;
- c) conformance test houses and others to define test cases.

To be conformant to the IAs, an implementation shall be conformant to at least one of the following categories:

- a) Management Communication;
- b) Management Functions and Services;
- c) Management Information.

Implementations which are conformant to these categories shall comply with the requirements stated in the following clause.

8.3 Specific Conformance Categories

8.3.1 Management Communication Categories

To be conformant to the Management Communication categories, an implementation shall conform to agreements in clause 6. Conformance to management communication also requires an implementor to state which of the management communication profiles specified in clause 6 are supported in the implementation. The implementor's statement of which profile is supported shall be indicated in a CMIP PICS as follows. The implementor shall complete the PICS proforma as specified by one of the profiles specified in clause 6.

Note: [Conformance requirements for these IAs, relating to services required of the upper layers and other ASEs, are discussed in part 5, clause 13.7]

8.3.2 Management Functions and Services Conformance Categories

To be conformant to the Management Functions and Services categories, an implementation shall conform to the agreements in clause 5 on at least one of the categories defined below in either a manager role, an agent role or both roles. [Note: These categories are aligned with the proposed AOM2x Profiles for Systems Management Functions.] [NMSG1] Conformance to agreements in clause 5 requires conformance to referenced ISO standards/CCITT Recommendations and to all other clauses referenced in 5, including dependent conformance to the underlying services required by the SMFs.

The implementor shall state which of the following conformance categories are supported. For each category, the implementor shall complete the related PICS and MOCS proformas to indicate which functional unit(s) and role(s) are supported.

8.3.2.1 General Management Capabilities Conformance Category

Note: [This category corresponds to proposed profile AOM211 [AOM211].]

Conformance to the General Management Capabilities Conformance Category requires general conformance to the Object Management Function [OMF], general conformance to the State Management Function [STMF], general conformance to the Attributes for Representing Relationships Function [ARR], and general conformance to the Alarm Reporting Function [ARF]. To be conformant to the Object Management Function, an implementation shall conform to the requirements stated in [OMF]. In addition, an implementation shall conform to clause 5.2 of these agreements and all other clauses referenced in 5.2. To be conformant to the State Management Function, an implementation shall conform to the requirements stated in [STMF]. In addition, an implementation shall conform to clause 5.3 of these agreements and all other clauses referenced in 5.3. To be conformant with the Attributes for Representing Relationships Function, an implementation shall conform to the requirements stated in [ARR]. In addition, an implementation shall conform to clause 5.4 of these agreements and all other clauses referenced in 5.4. To be conformant to the Alarm Reporting Function, an implementation shall conform to the requirements stated in [ARF]. In addition, an implementation shall conform to clause 5.5 of these agreements and all clauses referenced in 5.5.

8.3.2.2 Alarm Reporting and State Management Capabilities Conformance Category

Note: [This category corresponds to proposed profile AOM212 [AOM212].]

Conformance to the Alarm Reporting and State Management Capabilities Conformance Category requires general conformance to the State Management Function [STMF], general conformance to the Alarm Reporting Function [ARF], and dependent conformance to the Object Management Function [OMF]. To be conformant to the State Management Function, an implementation shall conform to the requirements stated in [STMF]. In addition, an implementation shall conform to clause 5.3 of these agreements and all other clauses referenced in 5.3. To be conformant to the Alarm Reporting Function, an implementation shall conform to the requirements stated in [ARF]. In addition, an implementation shall conform to clause 5.5 of these agreements and all clauses referenced in 5.5.

Dependent conformance to the Object Management Function required by the Alarm Reporting and State Management Capabilities Conformance Category requires support for the PT-SET and PT-GET elements of procedure in clauses 11.1.6 and 11.1.7 of [OMF] in either the agent role, the manager role, or both roles as specified by the implementor in the PICS. In addition, an implementation shall conform to clause 5.2.7 and clause 5.2.9 of these agreements and all clauses referenced in 5.2.7 and 5.2.9. The implementation need only support the PT-SET and PT-GET elements of procedure as applied to the State Management Attributes identified in [STMF] and specified in [DMI]. An implementation shall also conform to the notifications identified in [STMF] and specified in [DMI].

For each role claimed to be supported in the PICS, an implementation shall support the transfer syntax derived from the encoding rules defined in [BER] named [joint-iso-ccitt asn1(1) basic encoding(1)], for the purpose of generating and interpreting the MAPDUs required to support that portion of the "CMIP-PCI" abstract syntax defined in [CMIP] required to support the PT-GET and PT-SET elements of procedure as defined in clauses 11.1.6 and 11.1.7 of [OMF].

The implementation shall support the transfer syntax derived from the encoding rules specified in [BER] named [joint-iso-ccitt asn1(1) basic encoding(1)], for the purpose of generating and interpreting the MAPDUs defined by the abstract data types referenced in 11.2.6 of [STMF].

8.3.2.3 Alarm Reporting Capabilities Conformance Category

Note: [This category corresponds to proposed profile AOM213 [AOM213].]

Conformance to the Alarm Reporting Capabilities Conformance Category requires general conformance to the Alarm Reporting Function [ARF]. To be conformant to the Alarm Reporting Function, an implementation shall conform to the requirements stated in [ARF]. In addition, an implementation shall conform to clause 5.5 of these agreements and all clauses referenced in 5.5.

8.3.2.4 General Event Report Management Conformance Category

Note: [This category corresponds to proposed profile AOM221 [AOM221].]

Conformance to the General Event Report Management Conformance Category requires general conformance to the Event Report Management Function [ERMF], dependent conformance to the Object Management Function [OMF], and dependent conformance to the State Management Function [STMF]. To be conformant to the Event Report Management Function, an implementation shall conform to the requirements stated in [ERMF]. In addition, an implementation shall conform to clause 5.6 of these agreements and all clauses referenced in 5.6.

Dependent conformance to the Object Management Function required by the General Event Report Management Conformance Category requires support for the PT-SET, PT-GET, PT-CREATE, PT-DELETE, object creation reporting, object deletion reporting, and attribute value change reporting elements of procedure in clauses 11.1.1 through 11.1.7 of [OMF] in either the agent role, the manager role, or both roles as specified by the implementor in the PICS. In addition, an implementation shall conform to clause 5.2.2 through clause 5.2.7, and clause 5.2.9 of these agreements and all clauses referenced in these clauses. An implementation shall also conform to the notifications identified in [OMF] and specified in [DMI].

Dependent conformance to the State Management Function required by the General Event Report Management Conformance Category requires support for the state change reporting elements of procedure in clause 11.1 of [STMF] in either the agent role, the manager role, or both roles as specified by the implementor in the PICS. In addition, an implementation shall conform to clause 5.3.2 of these agreements and all clauses referenced by clause 5.3.2. An implementation shall also conform to the notifications identified in [STMF] and specified in [DMI].

For each role claimed to be supported in the PICS, an implementation shall support the transfer syntax derived from the encoding rules defined in [BER] named [joint-iso-ccitt asn1(1) basic encoding(1)], for the purpose of generating and interpreting the MAPDUs required to support that portion of the "CMIP-PCI" abstract syntax defined in [CMIP] required to support the PT-SET, PT-GET, PT-CREATE, PT-DELETE, object creation reporting, object deletion reporting, and attribute value change reporting elements of procedure as defined in clauses 11.1.1 through 11.1.7 of [OMF].

The implementation shall support the transfer syntax derived from the encoding rules specified in [BER] named [joint-iso-ccitt asn1(1) basic encoding(1)], for the purpose of generating and interpreting the MAPDUs defined by the abstract data types referenced in 11.2.5 of [OMF].

For each role claimed to be supported in the PICS, an implementation shall support the transfer syntax derived from the encoding rules defined in [BER] named [joint-iso-ccitt asn1(1) basic encoding(1)], for the purpose of generating and interpreting the MAPDUs required to support that portion of the "CMIP-PCI" abstract syntax defined in [CMIP] required to support the state change reporting elements of procedure as defined in clause 11.1 of [STMF].

The implementation shall support the transfer syntax derived from the encoding rules specified in [BER] named [joint-iso-ccitt asn1(1) basic encoding(1)], for the purpose of generating and interpreting the MAPDUs defined by the abstract data types referenced in 11.2.6 of [STMF].

8.3.2.5 General Log Control Conformance Category

Note: [This category corresponds to proposed profile AOM231 [AOM231].]

Conformance to the Log Control Conformance Category requires general conformance to the Log Control Function [LCF], dependent conformance to the Object Management Function [OMF], dependent conformance to the State Management Function [STMF] and dependent conformance to the Alarm Reporting Function [ARF]. To be conformant to the Log Control Function, an implementation shall conform to the requirements stated in [LCF]. In addition, an implementation shall conform to clause 5.7 of these agreements and all clauses referenced in 5.7.

Dependent conformance to the Object Management Function required by the General Log Control Conformance Category requires support for the PT-SET, PT-GET, PT-CREATE, PT-DELETE, object creation reporting, object deletion reporting, and attribute value change reporting elements of procedure in clauses 11.1.1 through 11.1.7 of [OMF] in either the agent role, the manager role, or both roles as specified by the implementor in the PICS. In addition, an implementation shall conform to clause 5.2.2 through clause 5.2.7, and clause 5.2.9 of these agreements and all clauses referenced in these clauses. An implementation shall also conform to the notifications identified in [OMF] and specified in [DMI].

Dependent conformance to the State Management Function required by the General Log Control Conformance Category requires support for the state change reporting elements of procedure in clause 11.1 of [STMF] in either the agent role, the manager role, or both roles as specified by the implementor in the PICS. In addition, an implementation shall conform to clause 5.3.2 of these agreements and all clauses referenced by clause 5.3.2. An implementation shall also conform to the notifications identified in [STMF] and specified in [DMI].

Dependent conformance to the Alarm Reporting Function required by the General Log Control Conformance Category requires support for the alarm reporting elements of procedure in clause 11.1 of [ARF] in either the agent role, the manager role, or both roles as specified by the implementor in the PICS. In addition, an implementation shall conform to clause 5.5.2 of these agreements and all clauses referenced by clause 5.5.2. An implementation shall also conform to the notifications identified in [ARF] and specified in [DMI].

For each role claimed to be supported in the PICS, an implementation shall support the transfer syntax derived from the encoding rules defined in [BER] named [joint-iso-ccitt asn1(1) basic encoding(1)], for the purpose of generating and interpreting the MAPDUs required to support that portion of the "CMIP-PCI" abstract syntax defined in [CMIP] required to support the PT-SET, PT-GET, PT-CREATE, PT-DELETE, object creation reporting,

object deletion reporting, and attribute value change reporting elements of procedure as defined in clauses 11.1.1 through 11.1.7 of [OMF].

The implementation shall support the transfer syntax derived from the encoding rules specified in [BER] named [joint-iso-ccitt asn1(1) basic encoding(1)], for the purpose of generating and interpreting the MAPDUs defined by the abstract data types referenced in 11.2.5 of [OMF].

For each role claimed to be supported in the PICS, an implementation shall support the transfer syntax derived from the encoding rules defined in [BER] named [joint-iso-ccitt asn1(1) basic encoding(1)], for the purpose of generating and interpreting the MAPDUs required to support that portion of the "CMIP-PCI" abstract syntax defined in [CMIP] required to support the state change reporting elements of procedure as defined in clause 11.1 of [STMF].

The implementation shall support the transfer syntax derived from the encoding rules specified in [BER] named [joint-iso-ccitt asn1(1) basic encoding(1)], for the purpose of generating and interpreting the MAPDUs defined by the abstract data types referenced in 11.2.6 of [STMF].

For each role claimed to be supported in the PICS, an implementation shall support the transfer syntax derived from the encoding rules defined in [BER] named [joint-iso-ccitt asn1(1) basic encoding(1)], for the purpose of generating and interpreting the MAPDUs required to support that portion of the "CMIP-PCI" abstract syntax defined in [CMIP] required to support the alarm reporting elements of procedure as defined in clause 11.1 of [ARF].

The implementation shall support the transfer syntax derived from the encoding rules specified in [BER] named [joint-iso-ccitt asn1(1) basic encoding(1)], for the purpose of generating and interpreting the MAPDUs defined by the abstract data types referenced in 11.2.5 of [ARF].

8.3.3 Management Information Conformance Category

To be conformant to the Management Information Conformance Category, an implementation shall include at least one managed object defined as specified by clause 7. The requirements for managing this managed object shall not conflict with the specifications in clauses 5 and 6. Managed object class definitions shall be provided either in full or by reference. Registered object identifiers shall be associated with any such managed object class definition and supporting definitions (e.g., attributes, name bindings). All mandatory abstract syntaxes and semantics associated with those identifiers shall be used. Note that all managed objects and supporting definitions in Annex A satisfy these conformance requirements.

An implementation is conformant to a managed object class definition if it supports all the mandatory packages specified in the managed object class as well as all associated information (e.g., attributes, notifications, actions, parameters) referenced in these packages and at least one name binding that may be used to support the naming of instances of this managed object class. Although it is not necessary to be conformant to all superior object classes in the containment tree of an instance of a conformant managed object class, all name bindings and naming attributes necessary to access that object instance shall be publicly available.

8.3.3.1 MOCS Proforma

The implementor shall provide a statement specifying which managed object classes are supported. A MOCS proforma shall be completed by the implementor for each managed object class supported.

Editor's Note: [The CD Version of ISO/IEC 10165-6 (Requirements and Guidelines for Implementation Conformance Statement Proformas Associated with Management Information) is now available. MOCS Proformas for each managed object class supported should be developed consistent with 10165-6 [MOCS].]

For each managed object class supported, the following shall be supplied:

- a) a statement of pragmatic constraints (e.g., attribute values/ranges, initial values) supported, unless such constraints are defined in the managed object class definition;
- b) a statement of conditional packages supported;
- c) a statement of role(s) (manager, agent, or both) in which the object class definition is supported.

Editor's Note: [CD 10165-6 does not currently distinguish roles.]

8.3.4 Management Application Contexts

The implementation shall support at least the application context for systems management defined in ISO/IEC 10040 [SMO].

Note: [Such a statement is required by [SMO] clause 7.2.]

Note: [Such a statement is required by part 5, clause 13.7, which discusses conformance requirements for these IAs, as related to services required of the upper layers and other ASEs.]

8.4 Demonstration of Conformance

(Refer to the Working Implementation Agreements Document.)

8.4.1 Management Communication

(Refer to the Working Implementation Agreements Document.)

PART 18: Network Management

December 1992 (Stable)

8.4.2 Management Functions and Services

(Refer to the Working Implementation Agreements Document.)

8.4.3 Management Information

(Refer to the Working Implementation Agreements Document.)

Annex A (informative)

Management Information Library (MIL)

(Refer to the Working Implementation Agreements Document for additional information.)

A.1 Introduction

(Refer to the Working Implementation Agreements Document.)

A.2 Rules and Procedures

(Refer to the Working Implementation Agreements Document.)

A.3 General Guidelines

(Refer to the Working Implementation Agreements Document.)

A.4 Harmonized Library

The definitions specified in this clause can be referenced by using the label "OP1 Library Vol. 1" (e.g., "OP1 Library Vol. 1":computerSystem).

By inclusion of the managed object (MO) definitions and the object identifiers in Annex A and Annex B, respectively, of the Stable Implementors' Agreements (SIAs), these managed object (MO) definitions have become formally registered. Implementors of part 18 of the SIAs do not have to support any of these MOs. However, even though Annex A and Annex B are informative annexes, any implementation that claims to conform to these definitions must treat these definitions as normative and comply with the relevant portions of Annex A.4 and A.5, and Annex B.

A.4.1 Managed Object Classes and Mandatory Packages

A.4.1.1 Computer System

computerSystem MANAGED OBJECT CLASS

DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2 : 1992":top;

CHARACTERIZED BY computerSystemPkg;

CONDITIONAL PACKAGES

peripheralNamePkg	PRESENT IF !an instance supports it and the peripheralListPkg is NOT present!,
peripheralListPkg	PRESENT IF !an instance supports it and the peripheralNamePkg is NOT present!,
processingEntityNamePkg	PRESENT IF !an instance supports it and the processingEntityListPkg is NOT present!,
processingEntityListPkg	PRESENT IF !an instance supports it and the processingEntityNamePkg is NOT present!,

PART 18: Network Management

December 1992 (Stable)

```
systemTimePkg          PRESENT IF !an instance supports it!,
upTimePkg              PRESENT IF !an instance supports it!,
"Rec. M.3100 : 1992":userLabelPackage
usageStatePkg          PRESENT IF !resource can detect usage!;
```

REGISTERED AS {x-objectClass 1};

```
computerSystemPkg      PACKAGE
  BEHAVIOUR computerSystemPkgDefinition,
              computerSystemPkgBehaviour;
  ATTRIBUTES
    computerSystemId GET,
    "Rec. X.721 | ISO/IEC 10165-2 : 1992":operationalState GET,
    "Rec. X.721 | ISO/IEC 10165-2 : 1992":administrativeState GET-REPLACE,
    "Rec. X.721 | ISO/IEC 10165-2 : 1992":alarmStatus GET-REPLACE ADD-REMOVE,
    "Rec. X.721 | ISO/IEC 10165-2 : 1992":availabilityStatus GET;
  ATTRIBUTE GROUPS
    "Rec. X.721 | ISO/IEC 10165-2 : 1992":state
    "Rec. X.721 | ISO/IEC 10165-2 : 1992":administrativeState
    "Rec. X.721 | ISO/IEC 10165-2 : 1992":operationalState
    "Rec. X.721 | ISO/IEC 10165-2 : 1992":alarmStatus
    "Rec. X.721 | ISO/IEC 10165-2 : 1992":availabilityStatus;
  NOTIFICATIONS
    "Rec. X.721 | ISO/IEC 10165-2 : 1992":objectCreation,
    "Rec. X.721 | ISO/IEC 10165-2 : 1992":objectDeletion,
    "Rec. X.721 | ISO/IEC 10165-2 : 1992":attributeValueChange,
    "Rec. X.721 | ISO/IEC 10165-2 : 1992":stateChange,
    "Rec. X.721 | ISO/IEC 10165-2 : 1992":processingErrorAlarm,
    "Rec. X.721 | ISO/IEC 10165-2 : 1992":environmentalAlarm,
    "Rec. X.721 | ISO/IEC 10165-2 : 1992":equipmentAlarm;;
```

```
computerSystemPkgDefinition  BEHAVIOUR
```

DEFINED AS

!The Computer System managed object class represents the aggregate of components which, when considered as a whole, is capable of performing data processing, storage, and retrieval functions. In order to perform its function, the computer system may have a variety of components including processing entities, terminals, disk drives, printers, etc.

The Computer System is intended to represent an aggregation of other objects, and can model either self-contained computer systems or computer systems which are physically distributed, possibly over a wide geographical area. An instance of the Computer System managed object class may have subordinate managed objects representing the individual entities within the computer system. Examples are entities such as disks, operating systems and processing entities.

Since the Computer System may be physically distributed, it is not appropriate to model the computer system managed object class as a subclass of the Equipment managed object class (since Equipment implies a single physical location through its location attribute). However, there can be cases where the Computer System is not physically distributed, in which case a Name Binding allowing Computer System to be named by OMNIPoint Equipment is permissible.

It is not appropriate to model Computer System as a subclass of the DMI System managed object class. Unlike Computer System, the DMI System is a "container" object class which is instantiated in managed systems and exists mainly to name the managed and support objects it makes visible.!

```
computerSystemPkgBehaviour  BEHAVIOUR
```

DEFINED AS

!A value for the computerSystemId attribute can only be provided when the object is created. Furthermore, this attribute value may not change once the managed object has been instantiated. Thus, this attribute is never the subject of an AttributeValueChange Notification.

Conditions under which an AttributeValueChange Notification is emitted are stated in the behaviour of the appropriate package or attribute. In the absence of such a statement in the behaviour, the attribute does not cause an AttributeValueChange notification to be emitted.

All attributeValueChange notifications shall include the Attribute Identifier List parameter.

The stateChange notification is emitted when any of the following attributes change in value: administrativeState, operationalState, and availability status.

The stateChange notification is not emitted when the alarmStatus attribute changes value. (This is to avoid duplication of notifications.)

Since every combination of state attribute values may not be appropriate for particular kinds of computer systems, only appropriate combinations need be supported.

The processingErrorAlarm notification is emitted when the computerSystem resource experiences any of the alarm conditions defined by ISO/IEC 10164-4 (e.g., storage capacity problem, version mismatch, corrupt data, software error, underlying resource unavailable).!;

A.4.1.2 Connection Oriented Transport Protocol Layer Entity

coTransportProtocolLayerEntity MANAGED OBJECT CLASS

DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2 : 1992":top;

CHARACTERIZED BY coTransportProtocolLayerEntityPkg;

CONDITIONAL PACKAGES

manufacturerListPkg	PRESENT IF !an instance supports it and the manufacturerNamePkg is NOT present!,
manufacturerNamePkg	PRESENT IF !an instance supports it and the manufacturerListPkg is NOT present!,
productLabelPkg	PRESENT IF !an instance supports it!,
opVersionPkg	PRESENT IF !an instance supports it!,
serialNumberPkg	PRESENT IF !an instance supports it!,
typeTextPkg	PRESENT IF !an instance supports it!,
upTimePkg	PRESENT IF !an instance supports it!,
incomingProtocolErrorPkg	PRESENT IF !an instance supports it!,
outgoingProtocolErrorPkg	PRESENT IF !an instance supports it!,
checksumPDUsDiscardedPkg	PRESENT IF !an instance supports it!,
maxPDUSizeIVPkg	PRESENT IF !the "OP1 Library Vol. 2 : 1992":transport ConnectionIVMO object class is not used to provide this initial value!,
usageStatePkg	PRESENT IF !resource can detect usage!;

REGISTERED AS {x-objectClass 2};

coTransportProtocolLayerEntityPkg PACKAGE

BEHAVIOUR coTransportProtocolLayerEntityPkgDefinition,
coTransportProtocolLayerEntityPkgBehaviour;

ATTRIBUTES

coTransportProtocolLayerEntityId PERMITTED VALUES SYNTAX-1.GraphicString64 GET,
transportEntityType GET,
localTransportAddresses GET,
activeConnections PERMITTED VALUES SYNTAX-1.Integer32 GET,


```

maxConnections PERMITTED VALUES SYNTAX-1.Integer32 GET,
"Rec. X.721 | ISO/IEC 10165-2 : 1992":administrativeState GET-REPLACE,
"Rec. X.721 | ISO/IEC 10165-2 : 1992":operationalState GET,
"Rec. X.721 | ISO/IEC 10165-2 : 1992":alarmStatus GET-REPLACE ADD-REMOVE,
"Rec. X.721 | ISO/IEC 10165-2 : 1992":outgoingConnectionRequestsCounter
    PERMITTED VALUES SYNTAX-1.Integer32 GET,
"Rec. X.721 | ISO/IEC 10165-2 : 1992":incomingConnectionRequestsCounter
    PERMITTED VALUES SYNTAX-1.Integer32 GET,
"Rec. X.721 | ISO/IEC 10165-2 : 1992":outgoingConnectionRejectErrorCounter
    PERMITTED VALUES SYNTAX-1.Integer32 GET,
"Rec. X.721 | ISO/IEC 10165-2 : 1992":incomingConnectionRejectErrorCounter
    PERMITTED VALUES SYNTAX-1.Integer32 GET,
"Rec. X.721 | ISO/IEC 10165-2 : 1992":outgoingDisconnectErrorCounter
    PERMITTED VALUES SYNTAX-1.Integer32 GET,
"Rec. X.721 | ISO/IEC 10165-2 : 1992":incomingDisconnectErrorCounter
    PERMITTED VALUES SYNTAX-1.Integer32 GET,
"Rec. X.721 | ISO/IEC 10165-2 : 1992":incomingDisconnectCounter
    PERMITTED VALUES SYNTAX-1.Integer32 GET,
"Rec. X.721 | ISO/IEC 10165-2 : 1992":outgoingDisconnectCounter
    PERMITTED VALUES SYNTAX-1.Integer32 GET,
"Rec. X.721 | ISO/IEC 10165-2 : 1992":octetsSentCounter
    PERMITTED VALUES SYNTAX-1.Integer32 GET,
"Rec. X.721 | ISO/IEC 10165-2 : 1992":octetsReceivedCounter
    PERMITTED VALUES SYNTAX-1.Integer32 GET;

```

ATTRIBUTE GROUPS

```

"Rec. X.721 | ISO/IEC 10165-2 : 1992":state
"Rec. X.721 | ISO/IEC 10165-2 : 1992":administrativeState
"Rec. X.721 | ISO/IEC 10165-2 : 1992":operationalState
"Rec. X.721 | ISO/IEC 10165-2 : 1992":alarmStatus;

```

ACTIONS activate, deactivate;

NOTIFICATIONS

```

"Rec. X.721 | ISO/IEC 10165-2 : 1992":objectCreation,
"Rec. X.721 | ISO/IEC 10165-2 : 1992":objectDeletion,
"Rec. X.721 | ISO/IEC 10165-2 : 1992":attributeValueChange,
"Rec. X.721 | ISO/IEC 10165-2 : 1992":stateChange,
"Rec. X.721 | ISO/IEC 10165-2 : 1992":processingErrorAlarm;;

```

coTransportProtocolLayerEntityPkgDefinition BEHAVIOUR

DEFINED AS

!The coTransportProtocolLayerEntity managed object class represents an instantiation of any connection-oriented transport layer protocol (e.g., the ISO Transport Protocol layer or the Internet Transmission Control Protocol (TCP) Layer). The transport protocol layer is layer four of the OSI Reference model. It provides for the transparent transference of data between two peer entities. It relieves its users from any concerns about the detailed way in which supporting communication media are utilized to achieve this transfer. The connection-oriented transport protocol layer entity makes use of a transport connection for the purpose of transferring data.

This is a generally applicable managed object class, in that it does not represent any specific connection-oriented transport protocol - rather it contains characteristics common across various different connection-oriented transport layer protocols. This managed object class is not intended to override any transport layer managed object classes defined in ISO. It provides a high level view of a connection-oriented transport layer protocol and complements the protocol-specific views being defined in the standards.!

coTransportProtocolLayerEntityPkgBehaviour BEHAVIOUR

DEFINED AS

!Conditions under which an attributeValueChange notification is emitted are stated in the behaviour of the appropriate package or attribute. In the absence of such a statement, in the behaviour, the attribute does not cause an attributeValueChange to Be emitted.

The attributeValueChange notification is emitted when any of the following attributes change in value: localTransportAddresses, maxConnections, transportEntityType, and all counter attributes (only when they wrap). All attributeValueChange notifications shall include the Attribute Identifier List parameter. All attributeValueChange notifications which report counter attribute wraps shall contain the maximum counter attribute value in the Old Attribute Value parameter.

The stateChange notification is emitted when any of the following attributes change in value: administrativeState and operationalState.

The processingErrorAlarm notification is emitted when the coTransportProtocolLayerEntity resource experiences any of the alarm conditions defined by ISO/IEC 10164-4 (e.g., storage capacity problem, version mismatch, corrupt data, software error, underlying resource unavailable).!;

This is a generally applicable managed object class, in that it does not represent any specific connection-oriented transport protocol. ISO/IEC 10733 [TLM] defines specific objects for managing OSI transport protocol layer entities.

A.4.1.3 Connectionless Network Protocol Layer Entity

clNetworkProtocolLayerEntity MANAGED OBJECT CLASS

DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2 : 1992": top;

CHARACTERIZED BY clNetworkProtocolLayerEntityPkg;

CONDITIONAL PACKAGES

manufacturerListPkg	PRESENT IF !an instance supports it and the manufacturerNamePkg is NOT present!,
manufacturerNamePkg	PRESENT IF !an instance supports it and the manufacturerListPkg is NOT present!,
productLabelPkg	PRESENT IF !an instance supports it!,
opVersionPkg	PRESENT IF !an instance supports it!,
serialNumberPkg	PRESENT IF !an instance supports it!,
typeTextPkg	PRESENT IF !an instance supports it!,
upTimePkg	PRESENT IF !an instance supports it!;

REGISTERED AS {x-objectClass 3};

clNetworkProtocolLayerEntityPkg PACKAGE

BEHAVIOUR clNetworkProtocolLayerEntityPkgDefinition,
 clNetworkProtocolLayerEntityPkgBehaviour;

ATTRIBUTES

clNetworkProtocolLayerEntityId	PERMITTED VALUES SYNTAX-1.GraphicString64 GET,
networkEntityType	GET,
localNetworkAddresses	GET-REPLACE ADD-REMOVE,
nPDUTimeToLive	PERMITTED VALUES SYNTAX-1.Integer32 GET-REPLACE,
"Rec. X.721 ISO/IEC 10165-2 : 1992":administrativeState	GET-REPLACE,
"Rec. X.721 ISO/IEC 10165-2 : 1992":operationalState	GET,
"Rec. X.721 ISO/IEC 10165-2 : 1992":alarmStatus	GET-REPLACE ADD-REMOVE,
"Rec. X.721 ISO/IEC 10165-2 : 1992":pdusSentCounter	PERMITTED VALUES SYNTAX-1.Integer32 GET,
"Rec. X.721 ISO/IEC 10165-2 : 1992":pdusReceivedCounter	PERMITTED VALUES SYNTAX-1.Integer32 GET,
"Rec. X.721 ISO/IEC 10165-2 : 1992":octetsSentCounter	PERMITTED VALUES SYNTAX-1.Integer32 GET,

```

"Rec. X.721 | ISO/IEC 10165-2 : 1992":octetsReceivedCounter
                                PERMITTED VALUES SYNTAX-1.Integer32 GET,
pdusForwardedCounter  PERMITTED VALUES SYNTAX-1.Integer32 GET,
pdusReasmbldOKCounter PERMITTED VALUES SYNTAX-1.Integer32 GET,
pdusReasmbldFailCounter PERMITTED VALUES SYNTAX-1.Integer32 GET,
pdusDiscardedCounter  PERMITTED VALUES SYNTAX-1.Integer32 GET;

```

ATTRIBUTE GROUPS

```

"Rec. X.721 | ISO/IEC 10165-2 : 1992":state
"Rec. X.721 | ISO/IEC 10165-2 : 1992":administrativeState
"Rec. X.721 | ISO/IEC 10165-2 : 1992":operationalState
"Rec. X.721 | ISO/IEC 10165-2 : 1992":alarmStatus;

```

ACTIONS activate, deactivate;

NOTIFICATIONS

```

"Rec. X.721 | ISO/IEC 10165-2 : 1992":objectCreation,
"Rec. X.721 | ISO/IEC 10165-2 : 1992":objectDeletion,
"Rec. X.721 | ISO/IEC 10165-2 : 1992":attributeValueChange,
"Rec. X.721 | ISO/IEC 10165-2 : 1992":processingErrorAlarm,
"Rec. X.721 | ISO/IEC 10165-2 : 1992":communicationsAlarm,
"Rec. X.721 | ISO/IEC 10165-2 : 1992":stateChange;;

```

clNetworkProtocolLayerEntityPkgDefinition BEHAVIOUR

DEFINED AS

The clNetworkProtocolLayerEntity managed object class represents an instantiation of a connectionless network protocol layer. The network protocol layer provides network services for the transparent transfer of data between peer transport entities. It relieves the transport protocol layer from the need to know anything about the underlying network technologies used to achieve data transfer.

This is a generally applicable managed object class, in that it does not represent any specific connectionless network protocol; instead, it contains characteristics common across various different connectionless network layer protocols. This managed object class is not intended to override any network layer managed object classes defined in ISO. It provides a high level view of a connectionless network layer protocol and complements the protocol-specific views being defined in the standards.

An instance of this managed object class supports only one type of protocol.!

clNetworkProtocolLayerEntityPkgBehaviour BEHAVIOUR

DEFINED AS

!Conditions under which an attributeValueChange notification is emitted are stated in the behaviour of the appropriate package or attribute. In the absence of such a statement, in the behaviour, the attribute does not cause an attributeValueChange to be emitted.

The attributeValueChange notification is emitted when any of the following attributes change in value: networkEntityType, localNetworkAddresses, nPDUTimeToLive, and all counter attributes (only when they wrap). All attributeValueChange notifications shall include the Attribute Identifier List parameter. All attributeValueChange notifications which report counter attribute wraps shall contain the maximum counter attribute value in the Old Attribute Value parameter.

The stateChange notification is emitted when any of the following attributes change in value: administrativeState and operationalState.

The communicationsAlarm notification is emitted when the clNetworkProtocolLayerEntity resource experiences any of the alarm conditions defined by ISO/IEC 10164-4 (e.g., loss of signal, local transmission error, remote transmission error). In particular, this notification is used to report when a data NPDU is discarded for any reason other than network congestion.

The processingErrorAlarm notification is emitted when the clNetworkProtocolLayerEntity resource experiences any of the alarm conditions defined by ISO/IEC 10164-4 (e.g., storage capacity problem, version mismatch, corrupt data, software error, underlying resource unavailable).!;

This is a generally applicable managed object class, in that it does not represent any specific connectionless network protocol. ISO/IEC 10737 [NLM] defines specific objects for managing OSI network protocol layer entities.

A.4.1.4 OMNIPoint Equipment

```
-- This definition is subclassed from CCITT M.3100 Equipment, adding the following items:
--
--      Mandatory AttributeChange, ObjectCreation, ObjectDeletion Notifications
--      Mandatory Environmental, Processing Error, and Equipment Alarm Notifications
--      Mandatory Administrative and Operational State Attributes and State Change Notification
--      CREATE/DELETE operations and behaviours (in name bindings)
--      Conditional Contact, Customer, Function, Manufacturer, OMNIPoint Network, Service, Software
--      and Vendor Name and List Packages
--      Conditional Product and Serial Number Packages
--      Conditional Type Text Package
--      Conditional Location Pointer Package
--
-- ANSI T1M1.5 concerns regarding physical vs. functional modelling were resolved by excluding the Forum
-- R1 Equipment Type attribute from the OMNIPoint definition. The TypeText, FunctionName, and/or
-- FunctionList attributes may be used to carry (as graphic strings or pointers) information concerning
-- the function(s) supported by the physical Equipment. It is expected that Forum R1 to OMNIPoint 1
-- mapping rules will define a translation between Forum R1 EquipmentType enumerations and these OMNIPoint
-- Equipment attributes.
```

opEquipment MANAGED OBJECT CLASS

DERIVED FROM "Rec. M.3100 : 1992":equipment;

CHARACTERIZED BY

```
opEquipmentPkg,
"Rec. M.3100 : 1992":createDeleteNotificationsPackage,
"Rec. M.3100 : 1992":attributeValueChangeNotificationPackage,
"Rec. M.3100 : 1992":stateChangeNotificationPackage,
"Rec. M.3100 : 1992":administrativeOperationalStatesPackage,
"Rec. M.3100 : 1992":environmentalAlarmPackage,
"Rec. M.3100 : 1992":processingErrorAlarmPackage,
"Rec. M.3100 : 1992":equipmentsEquipmentAlarmPackage;
```

CONDITIONAL PACKAGES

contactListPkg	PRESENT IF !an instance supports it and the contactNamePkg is NOT present!,
contactNamePkg	PRESENT IF !an instance supports it and the contactListPkg is NOT present!,
customerListPkg	PRESENT IF !an instance supports it and the customerNamePkg is NOT present!,
customerNamePkg	PRESENT IF !an instance supports it and the customerListPkg is NOT present!,
functionListPkg	PRESENT IF !an instance supports it and the functionNamePkg is NOT present!,
functionNamePkg	PRESENT IF !an instance supports it and the functionListPkg is NOT present!,
locationPointerPkg	PRESENT IF !an instance supports it and the "Rec. M.3100 : 1992": locationNamePackage is NOT present!,
manufacturerListPkg	PRESENT IF !an instance supports it and the manufacturerNamePkg is NOT present!,
manufacturerNamePkg	PRESENT IF !an instance supports it and the


```

opNetworkListPkg      PRESENT IF !an instance supports it and the
                        manufacturerListPkg is NOT present!,
opNetworkNamePkg      PRESENT IF !an instance supports it and the
                        opNetworkNamePkg is NOT present!,
opVersionPkg          PRESENT IF !an instance supports it and the
                        opNetworkListPkg is NOT present!,
                        "Rec. M.3100 : 1992":
                        versionPackage is also present!,
productLabelPkg       PRESENT IF !an instance supports it!,
serialNumberPkg       PRESENT IF !an instance supports it!,
serviceListPkg        PRESENT IF !an instance supports it and the
                        serviceNamePkg is NOT present!,
serviceNamePkg        PRESENT IF !an instance supports it and the
                        serviceListPkg is NOT present!,
softwareListPkg       PRESENT IF !an instance supports it and the
                        softwareNamePkg is NOT present!,
softwareNamePkg       PRESENT IF !an instance supports it and the
                        softwareListPkg is NOT present!,
typeTextPkg          PRESENT IF !an instance supports it!,
usageStatePkg         PRESENT IF !resource can detect usage!,
vendorListPkg         PRESENT IF !an instance supports it and the
                        "Rec. M.3100 : 1992":
                        vendorNamePackage is NOT present!;
```

REGISTERED AS {x-objectClass 4};

opEquipmentPkg PACKAGE

BEHAVIOUR opEquipmentPkgBehaviour;

-- opEquipmentPkgDefinition inherited from Rec. M.3100 Equipment

ATTRIBUTES

"Rec. M.3100 : 1992":equipmentId PERMITTED VALUES SYNTAX-1.EquipmentIdRange GET;

ATTRIBUTE GROUPS

"Rec. X.721 | ISO/IEC 10165-2 : 1992":state

"Rec. X.721 | ISO/IEC 10165-2 : 1992":administrativeState

"Rec. X.721 | ISO/IEC 10165-2 : 1992":operationalState;;

opEquipmentPkgBehaviour BEHAVIOUR

DEFINED AS -- inherited from Rec. M.3100 Equipment, with the following extensions:

!A value for the "Rec. M.3100 : 1992":equipmentId attribute can only be provided when the object is created. Furthermore, this attribute value may not change once the managed object has been instantiated. Thus, this attribute is never the subject of an AttributeValueChange Notification.

Conditions under which an AttributeValueChange Notification is emitted are stated in the behaviour of the appropriate package or attribute. In the absence of such a statement in the behaviour, the attribute does not cause an AttributeValueChange notification to be emitted.

All attributeValueChange notifications shall include the Attribute Identifier List parameter.

The processingErrorAlarm notification (if present) is emitted when the Equipment resource experiences any of the alarm conditions defined by ISO/IEC 10164-4 (e.g., storage capacity problem, version mismatch, corrupt data, software error, underlying resource unavailable).!;

A.4.1.5 OMNIPoint Network

-- This definition is subclassed from Rec. M.3100 Network, adding the following items:

```

--
-- Network Title and associated name binding to Root
-- AttributeChange, ObjectCreation, ObjectDeletion Notifications
-- CREATE/DELETE operations and behaviours (in name bindings)
```

opNetwork MANAGED OBJECT CLASS

DERIVED FROM "Rec. M.3100 : 1992":network;

CHARACTERIZED BY opNetworkPkg;

REGISTERED AS {x-objectClass 5};

opNetworkPkg PACKAGE

BEHAVIOUR opNetworkPkgBehaviour;

-- opNetworkPkgDefinition inherited from Rec. M.3100 Network

ATTRIBUTES

networkTitle GET;

NOTIFICATIONS

"Rec. X.721 | ISO/IEC 10165-2 : 1992":objectCreation,

"Rec. X.721 | ISO/IEC 10165-2 : 1992":objectDeletion,

"Rec. X.721 | ISO/IEC 10165-2 : 1992":attributeValueChange;;

opNetworkPkgBehaviour BEHAVIOUR

DEFINED AS -- inherited from Rec. M.3100 Network, with the following extensions:

!Values for the Network Identifier and Network Title attributes can only be provided when the object is created. Furthermore, these attribute values may not change once the managed object has been instantiated. Thus, they are never the subject of an AttributeValueChange Notification. When NetworkTitle is used for naming, the Network Identifier attribute has a NULL value.

Conditions under which an AttributeValueChange Notification is emitted are stated in the behaviour of the appropriate package or attribute. In the absence of such a statement in the behaviour, the attribute does not cause an AttributeValueChange notification to be emitted. All attributeValueChange notifications shall include the Attribute Identifier List parameter.!

A.4.1.6 Processing Entity

processingEntity MANAGED OBJECT CLASS

DERIVED FROM opEquipment;

CHARACTERIZED BY processingEntityPkg;

CONDITIONAL PACKAGES

addressingPkg PRESENT IF !relevant to the underlying resource!,

cpuUtilizationPkg PRESENT IF !an instance supports it!,

memorySizePkg PRESENT IF !relevant to the underlying resource!,

memoryUtilizationPkg PRESENT IF !an instance supports it!,

upTimePkg PRESENT IF !an instance supports it!;

REGISTERED AS {x-objectClass 6};

processingEntityPkg PACKAGE

BEHAVIOUR processingEntityPkgDefinition,
processingEntityPkgBehaviour;

ATTRIBUTES

cpuType PERMITTED VALUES SYNTAX-1.GraphicString16 GET,

osInfo PERMITTED VALUES SYNTAX-1.OsInfoRange GET;;

processingEntityPkgDefinition BEHAVIOUR

DEFINED AS

!The processingEntity managed object class represents the physical portion of the computer system that performs a processing function, frequently called a Central Processing Unit (CPU). A Processing Entity may be composed of such components as arithmetical logical units (ALU), registers for processing memory, limited storage most often in the form of Random Access Memory (RAM), and various other types of memory used in the processing function. It does not include such components as disk drives, data bases, etc.

Some Processing Entities may have input/output channels, particularly when hardware is shared between elements of the Processing Entity. In other cases, the input/output must be seen as components of a superior managed object, for example a Computer System, or as OMNIPoint Equipment objects shared among several Computer Systems.

The cpuType attribute indicates the type of central processor unit found in the Processing Entity.

The osInfo attribute specifies the names and releases of the supported operating systems.!

processingEntityPkgBehaviour BEHAVIOUR

DEFINED AS

!The AttributeValueChange notification is emitted when any of the following attributes change in value: cpuType or osInfo.!

A.4.1.7 Transport Connection

transportConnection MANAGED OBJECT CLASS

DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2 : 1992": top;

CHARACTERIZED BY transportConnectionPkg;

CONDITIONAL PACKAGES

maxRetransmissionsPkg	PRESENT IF !an instance supports it!,
retransmissionTimePkg	PRESENT IF !an instance supports it!,
retransmissionTimerInitialValuePkg	PRESENT IF !an instance supports it!,
pduRetransmittedCounterPkg	PRESENT IF !an instance supports it!,
octetsRetransmittedPkg	PRESENT IF !an instance supports it!,
pduRetransmittedThresholdPkg	PRESENT IF !an instance supports it!,
outgoingProtocolErrorPkg	PRESENT IF !an instance supports it!,
checksumPDUsDiscardedPkg	PRESENT IF !an instance supports it!,

REGISTERED AS {x-objectClass 7};

transportConnectionPkg PACKAGE

BEHAVIOUR transportConnectionPkgDefinition,
transportConnectionPkgBehaviour;

ATTRIBUTES

transportConnectionId PERMITTED VALUES SYNTAX-1.GraphicString64 GET,
localTransportConnectionEndpoint GET,
remoteTransportConnectionEndpoint GET,
transportConnectionReference PERMITTED VALUES SYNTAX-1.Integer32 GET,
localNetworkAddress GET,
remoteNetworkAddress GET,
inactivityTimeout PERMITTED VALUES SYNTAX-1.Integer32 GET,
inactivityTime PERMITTED VALUES SYNTAX-1.Integer32 GET,
maxPDUSize PERMITTED VALUES SYNTAX-1.Integer32 GET,
"Rec. X.721 | ISO/IEC 10165-2 : 1992":pduSentCounter


```

                                PERMITTED VALUES SYNTAX-1.Integer32 GET,
"Rec. X.721 | ISO/IEC 10165-2 : 1992":pdusReceivedCounter
                                PERMITTED VALUES SYNTAX-1.Integer32 GET,
"Rec. X.721 | ISO/IEC 10165-2 : 1992":octetsSentCounter
                                PERMITTED VALUES SYNTAX-1.Integer32 GET,
"Rec. X.721 | ISO/IEC 10165-2 : 1992":octetsReceivedCounter
                                PERMITTED VALUES SYNTAX-1.Integer32 GET,
"Rec. X.721 | ISO/IEC 10165-2 : 1992":incomingProtocolErrorCounter
                                PERMITTED VALUES SYNTAX-1.Integer32 GET;
NOTIFICATIONS
"Rec. X.721 | ISO/IEC 10165-2 : 1992":objectCreation,
"Rec. X.721 | ISO/IEC 10165-2 : 1992":objectDeletion transportDisconnectCause,
"Rec. X.721 | ISO/IEC 10165-2 : 1992":attributeValueChange;;

```

transportConnectionPkgDefinition BEHAVIOUR**DEFINED AS**

The transportConnection managed object class represents an active transport connection (e.g., an OSI transport connection or a TCP connection). A transport connection is established and used by two peer connection oriented transport protocol layer entities for the purpose of transferring data. A connection oriented transport protocol layer entity may support multiple transport connections.

This is a generally applicable managed object class, in that it does not represent any specific connection-oriented transport protocol; rather it contains characteristics common across various different connection-oriented transport layer protocols. This managed object class is not intended to override any transport layer managed object classes defined in ISO. It provides a high level view of a connection-oriented transport layer protocol and complements the protocol-specific views being defined in the standards.!

transportConnectionPkgBehaviour BEHAVIOUR**DEFINED AS**

An instance of the Transport Connection managed object class is created automatically in response to normal operation of the network. A prerequisite to the creation of a transport connection is the existence of a transport entity (e.g. an instance of the Connection Oriented Transport Protocol Layer Entity) on the open system. When a new Transport Connection instance is created, the "OP1 Library Vol. 2 : 1992":transportConnectionIVMO instance with the same superior may be used to provide initial attribute values for the new instance. Alternatively, the Maximum PDU Size attribute takes on the value of the Maximum PDU Size attribute specified in the superior Transport Protocol Layer Entity managed object instance. Subsequently the Maximum PDU Size attribute may take on another value which applies specifically to the connection represented by the instantiation of the transport connection. This change may occur as the result of peer protocol negotiation.

The Additional Information parameter of the objectDeletion notification may optionally contain a management extension (as defined in DMI) whose identifier is that of the "cause" attribute, whose significance is FALSE, and whose information is "cause" as defined in the associated PARAMETER template.

Conditions under which an attributeValueChange notification is emitted are stated in the behaviour of the appropriate package or attribute. In the absence of such a statement, in the behaviour, the attribute does not cause an attributeValueChange to be emitted.

The attributeValueChange notification is emitted when any of the following attributes change in value: inactivityTimeout, maxPDUSize, and all counter attributes (only when they wrap). All attributeValueChange notifications shall include the Attribute Identifier List parameter. All attributeValueChange notifications which report counter attribute wraps shall contain the maximum counter attribute value in the Old Attribute Value parameter.

Transport Connection will delete itself when the value of the inactivityTime attribute equals that of the inactivityTimeout attribute.!

This is a generally applicable managed object class, in that it does not represent any specific connection-oriented transport protocol. ISO/IEC 10733 [TLM] defines specific objects for managing OSI transport protocol layer entities.

A.4.2 Conditional Packages

A.4.2.1 Addressing Package

addressingPkg **PACKAGE**

BEHAVIOUR addressingPkgDefinition,
 addressingPkgBehaviour;

ATTRIBUTES addressingSize **PERMITTED VALUES SYNTAX-1.AddressingSizeRange GET,**
 endianess **GET;**

REGISTERED AS (x-package 1);

addressingPkgDefinition **BEHAVIOUR**

DEFINED AS

!This package defines the addressing size and endianess which are characteristic of the underlying resource.!

addressingPkgBehaviour **BEHAVIOUR**

DEFINED AS

!!If the AttributeValueChange notification is defined for the managed object class using this package, this notification is emitted when the addressingSize or endianess attributes change value.!

A.4.2.2 Checksum PDUs Discarded Package

checksumPDUsDiscardedPkg **PACKAGE**

BEHAVIOUR checksumPDUsDiscardedPkgDefinition,
 checksumPDUsDiscardedPkgBehaviour;

ATTRIBUTES checksumPDUsDiscardedCounter **PERMITTED VALUES SYNTAX-1.Integer32 GET;**

REGISTERED AS (x-package 2);

checksumPDUsDiscardedPkgDefinition **BEHAVIOUR**

DEFINED AS

!This package reflects the capability of the underlying resource to count the number of well-formed PDUs rejected by the peer entity due to a checksum error.!

checksumPDUsDiscardedPkgBehaviour **BEHAVIOUR**

DEFINED AS

!!If the attributeValueChange notification is defined for the managed object class using this package, this notification is emitted when the checksumPDUsDiscarded attribute wraps.!

A.4.2.3 Contact List Package

contactListPkg **PACKAGE**

BEHAVIOUR contactListPkgDefinition,
contactListPkgBehaviour;

ATTRIBUTES contactList PERMITTED VALUES SYNTAX-1.AnyNamesRange GET-REPLACE ADD-REMOVE;

REGISTERED AS {x-package 3};

contactListPkgDefinition BEHAVIOUR

DEFINED AS

!The Contact List Attribute identifies who (person or organization) should be contacted about the resource.!

contactListPkgBehaviour BEHAVIOUR

DEFINED AS

!If the AttributeValueChange notification is defined for the managed object class using this package, this notification is emitted when the contactList attribute changes value.!

A.4.2.4 Contact Name Package

contactNamePkg PACKAGE

BEHAVIOUR contactNamePkgDefinition,
contactNamePkgBehaviour;

ATTRIBUTES contactName PERMITTED VALUES SYNTAX-1.AnyNameRange GET-REPLACE;

REGISTERED AS {x-package 4};

contactNamePkgDefinition BEHAVIOUR

DEFINED AS

!The Contact Name Attribute identifies who (person or organization) should be contacted about the resource.!

contactNamePkgBehaviour BEHAVIOUR

DEFINED AS

!If the AttributeValueChange notification is defined for the managed object class using this package, this notification is emitted when the contactName attribute changes value.!

A.4.2.5 CPU Utilization Package

cpuUtilizationPkg PACKAGE

BEHAVIOUR cpuUtilizationPkgBehaviour;

ATTRIBUTES cpuUtilization PERMITTED VALUES SYNTAX-1.PercentageRange
GET; -- changed from GET-REPLACE (Forum)

REGISTERED AS {x-package 5};

cpuUtilizationPkgBehaviour BEHAVIOUR

DEFINED AS

!Even if the AttributeValueChange notification is defined for the managed object class using this package, this notification is NOT emitted when the cpuUtilization attribute changes value.!

A.4.2.6 Customer List Package

customerListPkg PACKAGE

 BEHAVIOUR customerListPkgDefinition,
 customerListPkgBehaviour;

 ATTRIBUTES customerList PERMITTED VALUES SYNTAX-1.AnyNamesRange GET-REPLACE ADD-REMOVE;

REGISTERED AS {x-package 6};

customerListPkgDefinition BEHAVIOUR

DEFINED AS

 !The Customer List attribute identifies any customers that are users of the resource.!

customerListPkgBehaviour BEHAVIOUR

DEFINED AS

 !If the AttributeValueChange notification is defined for the managed object class using this package, this notification is emitted when the customerList attribute changes value.!

A.4.2.7 Customer Name Package

customerNamePkg PACKAGE

 BEHAVIOUR customerNamePkgDefinition,
 customerNamePkgBehaviour;

 ATTRIBUTES customerName PERMITTED VALUES SYNTAX-1.AnyNameRange GET-REPLACE;

REGISTERED AS {x-package 7};

customerNamePkgDefinition BEHAVIOUR

DEFINED AS

 !The Customer Name attribute identifies any customer that is a user of the resource.!

customerNamePkgBehaviour BEHAVIOUR

DEFINED AS

 !If the AttributeValueChange notification is defined for the managed object class using this package, this notification is emitted when the customerName attribute changes value.!

A.4.2.8 Function List Package

functionListPkg PACKAGE

 BEHAVIOUR functionListPkgDefinition,
 functionListPkgBehaviour;

 ATTRIBUTES functionList PERMITTED VALUES SYNTAX-1.AnyNamesRange GET-REPLACE ADD-REMOVE;

REGISTERED AS {x-package 8};

functionListPkgDefinition BEHAVIOUR

DEFINED AS

 !The functionList attribute identifies those functions provided by this resource.!

functionListPkgBehaviour BEHAVIOUR

DEFINED AS

!If the AttributeValueChange notification is defined for the managed object class using this package, this notification is emitted when the functionList attribute changes value.!

A.4.2.9 Function Name Package

```
functionNamePkg    PACKAGE

    BEHAVIOUR functionNamePkgDefinition,
                functionNamePkgBehaviour;

    ATTRIBUTES functionName PERMITTED VALUES SYNTAX-1.AnyNameRange GET-REPLACE;

REGISTERED AS {x-package 9};

functionNamePkgDefinition BEHAVIOUR

DEFINED AS
    !The functionName attribute identifies the function provided by this resource.!
```

```
functionNamePkgBehaviour BEHAVIOUR

DEFINED AS
    !If the AttributeValueChange notification is defined for the managed object class using this
    package, this notification is emitted when the functionName attribute changes value.!
```

A.4.2.10 Incoming Protocol Error Package

```
incomingProtocolErrorPkg PACKAGE

    BEHAVIOUR incomingProtocolErrorPkgDefinition,
                incomingProtocolErrorPkgBehaviour;

    ATTRIBUTES
        "Rec. X.721 | ISO/IEC 10165-2 : 1992":incomingProtocolErrorCounter
        PERMITTED VALUES SYNTAX-1.Integer32 GET;

REGISTERED AS {x-package 10};

incomingProtocolErrorPkgDefinition BEHAVIOUR

DEFINED AS
    !This package reflects the capability of the underlying resource to count the number of incoming
    protocol errors detected.!
```

```
incomingProtocolErrorPkgBehaviour BEHAVIOUR

DEFINED AS
    !If the attributeValueChange notification is defined for the managed object class using this
    package, this notification is emitted when the incomingProtocolErrorCounter attribute wraps.!
```

A.4.2.11 Location Pointer Package

```
locationPointerPkg    PACKAGE
    BEHAVIOUR locationPointerPkgDefinition,
                locationPointerPkgBehaviour;

    ATTRIBUTES
        locationPointer GET-REPLACE;

REGISTERED AS {x-package 11};
```


locationPointerPkgDefinition BEHAVIOUR

DEFINED AS

!This package provides managed object instance information for a location (e.g., Hilo Hawaii USA).!;

locationPointerPkgBehaviour BEHAVIOUR

DEFINED AS

!If the attributeValueChange notification is defined for the managed object class using this package, this notification is emitted when the Location Pointer attribute changes value.!;

A.4.2.12 Manufacturer List Package

manufacturerListPkg PACKAGE

BEHAVIOUR manufacturerListPkgDefinition,
manufacturerListPkgBehaviour;

ATTRIBUTES

manufacturerList PERMITTED VALUES SYNTAX-1.AnyNamesRange GET-REPLACE ADD-REMOVE;

REGISTERED AS {x-package 12};

manufacturerListPkgDefinition BEHAVIOUR

DEFINED AS

!This package indicates information about the manufacturer(s) that manufactured the underlying resource!;

manufacturerListPkgBehaviour BEHAVIOUR

DEFINED AS

!If the attributeValueChange notification is defined for the managed object class using this package, this notification is emitted when the ManufacturerList attribute changes value.!;

A.4.2.13 Manufacturer Name Package

manufacturerNamePkg PACKAGE

BEHAVIOUR manufacturerNamePkgDefinition,
manufacturerNamePkgBehaviour;

ATTRIBUTES

manufacturerName PERMITTED VALUES SYNTAX-1.AnyNameRange GET-REPLACE;

REGISTERED AS {x-package 13};

manufacturerNamePkgDefinition BEHAVIOUR

DEFINED AS

!This package indicates information about the manufacturer that manufactured the underlying resource!;

manufacturerNamePkgBehaviour BEHAVIOUR

DEFINED AS

!If the attributeValueChange notification is defined for the managed object class using this package, this notification is emitted when the ManufacturerName attribute changes value.!;

A.4.2.14 Max PDU Size IV Package

```
maxPDUSizeIVPkg PACKAGE
    BEHAVIOUR maxPDUSizeIVPkgDefinition,
              maxPDUSizeIVPkgBehaviour;
```

```
    ATTRIBUTES
        maxPDUSize PERMITTED VALUES SYNTAX-1.Integer32 GET-REPLACE;
REGISTERED AS (x-package 14);
```

```
maxPDUSizeIVPkgDefinition BEHAVIOUR
```

```
DEFINED AS
    !This package provides the initial value for the maximum length of a PDU that can be supported by
    the local layer entity.!
```

```
maxPDUSizeIVPkgBehaviour BEHAVIOUR
```

```
DEFINED AS
    !The Maximum TPDU Size attribute provides the initial value to be used by newly-instantiated
    subordinate Transport Connection managed object instances for the maximum TPDU size to be supported
    on that connection.!
```

A.4.2.15 Max Retransmissions Package

```
maxRetransmissionsPkg PACKAGE
    BEHAVIOUR maxRetransmissionsPkgDefinition,
              maxRetransmissionsPkgBehaviour;
```

```
    ATTRIBUTES
        maxRetransmissions PERMITTED VALUES SYNTAX-1.Integer32 GET;
REGISTERED AS (x-package 15);
```

```
maxRetransmissionsPkgDefinition BEHAVIOUR
```

```
DEFINED AS
    !This package reflects the capability of the underlying transport protocol resource to count the
    maximum number of times a TPDU is to be retransmitted before the transport connection is aborted.!
```

```
maxRetransmissionsPkgBehaviour BEHAVIOUR
```

```
DEFINED AS
    !When a new Transport Connection instance is created containing this package, any "OP1 Library Vol.
    2 : 1992":transportConnectionRetransmissionIVMO instance with the same superior may be used to
    provide initial attribute values for the new instance.!
```

A.4.2.16 Memory Size Package

```
memorySizePkg PACKAGE
    BEHAVIOUR memorySizePkgDefinition,
              memorySizePkgBehaviour;
```

PART 18: Network Management

December 1992 (Stable)

```
ATTRIBUTES    memorySize    PERMITTED VALUES SYNTAX-1.MemorySizeRange    GET;

REGISTERED AS (x-package 16);

memorySizePkgDefinition    BEHAVIOUR

DEFINED AS
    !The memorySize attribute indicates, in kilobytes, the amount of memory available to a Processing
    Entity (irrespective of its current usage).!;

memorySizePkgBehaviour    BEHAVIOUR
```

```
DEFINED AS
    !!If the AttributeValueChange notification is defined for the managed object class using this
    package, this notification is emitted when the memorySize attribute changes value.!
```

A.4.2.17 Memory Utilization Package

```
memoryUtilizationPkg    PACKAGE

    BEHAVIOUR    memoryUtilizationPkgBehaviour;

    ATTRIBUTES    memoryUtilization    PERMITTED VALUES SYNTAX-1.PercentageRange
    GET; -- added in response to Bull comment

REGISTERED AS (x-package 17);

memoryUtilizationPkgBehaviour    BEHAVIOUR

DEFINED AS
    !Even if the AttributeValueChange notification is defined for the managed object class using this
    package, this notification is NOT emitted when the memoryUtilization attribute changes value.!
```

A.4.2.18 Octets Retransmitted Package

```
octetsRetransmittedPkg    PACKAGE

    BEHAVIOUR    octetsRetransmittedPkgDefinition,
    octetsRetransmittedPkgBehaviour;

    ATTRIBUTES
    "Rec. X.721 | ISO/IEC 10165-2 : 1992":octetsRetransmittedErrorCounter
    PERMITTED VALUES SYNTAX-1.Integer32    GET;

REGISTERED AS (x-package 18);

octetsRetransmittedPkgDefinition    BEHAVIOUR

DEFINED AS
    !This package reflects the capability of the underlying transport protocol resource to count the
    number of octets retransmitted.!
```

```
octetsRetransmittedPkgBehaviour    BEHAVIOUR

DEFINED AS
    !!If the attributeValueChange notification is defined for the managed object class using this
    package, this notification is emitted when the octetsRetransmitted attribute wraps.!
```

A.4.2.19 OMNIPoint Network List Package

opNetworkListPkg PACKAGE

BEHAVIOUR opNetworkListPkgDefinition,
opNetworkListPkgBehaviour;

ATTRIBUTES opNetworkList PERMITTED VALUES SYNTAX-1.AnyNamesRange GET-REPLACE ADD-REMOVE;

REGISTERED AS {x-package 19};

opNetworkListPkgDefinition BEHAVIOUR

DEFINED AS

!The opNetworkList attribute indicates what networks use or are dependent on the resource.!

opNetworkListPkgBehaviour BEHAVIOUR

DEFINED AS

!If the AttributeValueChange notification is defined for the managed object class using this package, this notification is emitted when the opNetworkList attribute changes value.!

A.4.2.20 OMNIPoint Network Name Package

opNetworkNamePkg PACKAGE

BEHAVIOUR opNetworkNamePkgDefinition,
opNetworkNamePkgBehaviour;

ATTRIBUTES opNetworkName PERMITTED VALUES SYNTAX-1.AnyNameRange GET-REPLACE;

REGISTERED AS {x-package 20};

opNetworkNamePkgDefinition BEHAVIOUR

DEFINED AS

!The opNetworkName attribute indicates what network uses or is dependent on the resource.!

opNetworkNamePkgBehaviour BEHAVIOUR

DEFINED AS

!If the AttributeValueChange notification is defined for the managed object class using this package, this notification is emitted when the opNetworkName attribute changes value.!

A.4.2.21 OMNIPoint Version Package

opVersionPkg PACKAGE -- refinement of Rec. M.3100 versionPackage

BEHAVIOUR opVersionPkgDefinition,
opVersionPkgBehaviour;

ATTRIBUTES

"Rec. M.3100 : 1992":version

PERMITTED VALUES SYNTAX-1.GraphicString16 GET-REPLACE;

REGISTERED AS {x-package 21};

opVersionPkgDefinition BEHAVIOUR

DEFINED AS

!This package reflects the release version of the underlying resource as an attribute, as defined by "Rec. M.3100 : 1992".!;

opVersionPkgBehaviour BEHAVIOUR

DEFINED AS

!If the attributeValueChange notification is defined for the managed object class using this package, this notification is emitted when the Version attribute changes value.!

A.4.2.22 Outgoing Protocol Error Package

outgoingProtocolErrorPkg PACKAGE

BEHAVIOUR outgoingProtocolErrorPkgDefinition,
outgoingProtocolErrorPkgBehaviour;

ATTRIBUTES

"Rec. X.721 | ISO/IEC 10165-2 : 1992":outgoingProtocolErrorCounter
PERMITTED VALUES SYNTAX-1.Integer32 GET;

REGISTERED AS {x-package 22};

outgoingProtocolErrorPkgDefinition BEHAVIOUR

DEFINED AS

!This package reflects the capability of the underlying resource to count the number of outgoing protocol errors detected. Note that not all resources have this capability.!

outgoingProtocolErrorPkgBehaviour BEHAVIOUR

DEFINED AS

!If the attributeValueChange notification is defined for the managed object class using this package, this notification is emitted when the outgoingProtocolErrorCounter attribute wraps.!

A.4.2.23 PDUs Retransmitted Counter Package

pdusRetransmittedCounterPkg PACKAGE

BEHAVIOUR pdusRetransmittedCounterPkgDefinition,
pdusRetransmittedCounterPkgBehaviour;

ATTRIBUTES

"Rec. X.721 | ISO/IEC 10165-2 : 1992":pdusRetransmittedErrorCounter
PERMITTED VALUES SYNTAX-1.Integer32 GET;

REGISTERED AS {x-package 23};

pdusRetransmittedCounterPkgDefinition BEHAVIOUR

DEFINED AS

!This package reflects the capability of the underlying transport protocol resource to count the number of PDUs retransmitted.!

pdusRetransmittedCounterPkgBehaviour BEHAVIOUR

DEFINED AS

!If the attributeValueChange notification is defined for the managed object class using this package, this notification is emitted when the PDUsRetransmittedCounter attribute wraps.!

A.4.2.24 PDUs Retransmitted Threshold Package

pdusRetransmittedThresholdPkg PACKAGE

BEHAVIOUR pdusRetransmittedThresholdPkgDefinition,
pdusRetransmittedThresholdPkgBehaviour;

ATTRIBUTES

"Rec. X.721 | ISO/IEC 10165-2 : 1992":pdusRetransmittedErrorThreshold GET-REPLACE;

NOTIFICATIONS

"Rec. X.721 | ISO/IEC 10165-2 : 1992":communicationsAlarm;

REGISTERED AS (x-package 24);

pdusRetransmittedThresholdPkgDefinition BEHAVIOUR

DEFINED AS

!This package reflects the capability of the underlying transport protocol resource to threshold the number of PDUs retransmitted.!

pdusRetransmittedThresholdPkgBehaviour BEHAVIOUR

DEFINED AS

!When a new Transport Connection instance is created containing this package, any "OP1 Library Vol. 2 : 1992":transportConnectionRetransmissionIVMO instance with the same superior may be used to provide initial attribute values for the new instance.

If the attributeValueChange notification is defined for the managed object class using this package, this notification is emitted when the pdusRetransmittedThreshold attribute changes in value.!

A.4.2.25 Peripheral List Package

peripheralListPkg PACKAGE

BEHAVIOUR peripheralListPkgDefinition,
peripheralListPkgBehaviour;

ATTRIBUTES

peripheralList PERMITTED VALUES SYNTAX-1.AnyNamesRange GET-REPLACE ADD-REMOVE;

REGISTERED AS (x-package 25);

peripheralListPkgDefinition BEHAVIOUR

DEFINED AS

!The Peripheral List attribute identifies auxiliary devices that are used by the resource (e.g., disk drives, tape drives, printers).!

peripheralListPkgBehaviour BEHAVIOUR

DEFINED AS

!!If the attributeValueChange notification is defined for the managed object class using this package, this notification is emitted when the Peripheral List attribute changes value.!!;

A.4.2.26 Peripheral Name Package

peripheralNamePkg PACKAGE

BEHAVIOUR peripheralNamePkgDefinition,
peripheralNamePkgBehaviour;

ATTRIBUTES

peripheralName PERMITTED VALUES SYNTAX-1.AnyNameRange GET-REPLACE;

REGISTERED AS {x-package 26};

peripheralNamePkgDefinition BEHAVIOUR

DEFINED AS

!!The Peripheral Name attribute identifies an auxiliary device that is used by the resource (e.g., disk drive, tape drive, printer).!!;

peripheralNamePkgBehaviour BEHAVIOUR

DEFINED AS

!!If the attributeValueChange notification is defined for the managed object class using this package, this notification is emitted when the Peripheral Name attribute changes value.!!;

A.4.2.27 Processing Entity List Package

processingEntityListPkg PACKAGE

BEHAVIOUR processingEntityListPkgDefinition,
processingEntityListPkgBehaviour;

ATTRIBUTES

processingEntityList PERMITTED VALUES SYNTAX-1.AnyNamesRange GET-REPLACE ADD-REMOVE;

REGISTERED AS {x-package 27};

processingEntityListPkgDefinition BEHAVIOUR

DEFINED AS

!!The Processing Entity List attribute identifies the processing entities which may be used by the containing object instance but which are not contained in it (i.e., processing entities which are shared among systems).!!;

processingEntityListPkgBehaviour BEHAVIOUR

DEFINED AS

!!If the attributeValueChange notification is defined for the managed object class using this package, this notification is emitted when the Processing Entity List attribute changes value.!!;

A.4.2.28 Processing Entity Name Package

processingEntityNamePkg PACKAGE

BEHAVIOUR processingEntityNamePkgDefinition,
processingEntityNamePkgBehaviour;

ATTRIBUTES
processingEntityName PERMITTED VALUES SYNTAX-1.AnyNameRange GET-REPLACE;

REGISTERED AS {x-package 28};

processingEntityNamePkgDefinition BEHAVIOUR

DEFINED AS

!The Processing Entity Name attribute identifies the processing entity which may be used by the containing object instance but which is not contained in it (i.e., processing entities which are shared among systems).!;

processingEntityNamePkgBehaviour BEHAVIOUR

DEFINED AS

!If the attributeValueChange notification is defined for the managed object class using this package, this notification is emitted when the Processing Entity Name attribute changes value.!;

A.4.2.29 Product Label Package

productLabelPkg PACKAGE

BEHAVIOUR productLabelPkgDefinition,
productLabelPkgBehaviour;

ATTRIBUTES
productLabel PERMITTED VALUES SYNTAX-1.GraphicString32 GET-REPLACE;

REGISTERED AS {x-package 29};

productLabelPkgDefinition BEHAVIOUR

DEFINED AS

!This package allows the product number or identifying string (e.g., model number) of the underlying resource to be reflected as an attribute.!;

productLabelPkgBehaviour BEHAVIOUR

DEFINED AS

!If the attributeValueChange notification is defined for the managed object class using this package, this notification is emitted when the Product Label attribute changes value.!;

A.4.2.30 Retransmission Time Package

retransmissionTimePkg PACKAGE

BEHAVIOUR retransmissionTimePkgDefinition,
retransmissionTimePkgBehaviour;

ATTRIBUTES

retransmissionTime PERMITTED VALUES SYNTAX-1.Integer32 GET;

REGISTERED AS {x-package 30};

retransmissionTimePkgDefinition BEHAVIOUR

DEFINED AS

!This package reflects the capability of the underlying transport protocol resource to present its current retransmission timer value as an attribute.!

retransmissionTimePkgBehaviour BEHAVIOUR

DEFINED AS

!When a new Transport Connection instance is created containing this package, the initial value of this attribute may be provided by the retransmissionTimerInitialValue attribute (if present in the new managed object instance).!

A.4.2.31 Retransmission Timer Initial Value Package

retransmissionTimerInitialValuePkg PACKAGE

BEHAVIOUR retransmissionTimerInitialValuePkgDefinition,
retransmissionTimerInitialValuePkgBehaviour;

ATTRIBUTES

retransmissionTimerInitialValue PERMITTED VALUES SYNTAX-1.Integer32 GET;

REGISTERED AS {x-package 31};

retransmissionTimerInitialValuePkgDefinition BEHAVIOUR

DEFINED AS

!This package reflects the capability of the underlying transport protocol resource to present its initial retransmission timer value as an attribute.!

retransmissionTimerInitialValuePkgBehaviour BEHAVIOUR

DEFINED AS

!When a new Transport Connection instance is created containing this package, any "OP1 Library Vol. 2 : 1992":transportConnectionRetransmissionIVMO instance with the same superior may be used to provide initial attribute values for the new instance.!

A.4.2.32 Serial Number Package

serialNumberPkg PACKAGE

BEHAVIOUR serialNumberPkgDefinition,
serialNumberPkgBehaviour;

ATTRIBUTES

serialNumber PERMITTED VALUES SYNTAX-1.GraphicString32 GET-REPLACE;

REGISTERED AS {x-package 32};

serialNumberPkgDefinition BEHAVIOUR

DEFINED AS

!This package allows the serial number of the underlying resource to be reflected as an attribute.!

serialNumberPkgBehaviour BEHAVIOUR

DEFINED AS

!If the attributeValueChange notification is defined for the managed object class using this package, this notification is emitted when the Serial Number attribute changes value.!

A.4.2.33 Service List Package

serviceListPkg PACKAGE

BEHAVIOUR serviceListPkgDefinition,
serviceListPkgBehaviour;

ATTRIBUTES serviceList PERMITTED VALUES SYNTAX-1.AnyNamesRange GET-REPLACE ADD-REMOVE;

REGISTERED AS {x-package 33};

serviceListPkgDefinition BEHAVIOUR

DEFINED AS

!Service List attribute identifies any services that are supported by the resource.!

serviceListPkgBehaviour BEHAVIOUR

DEFINED AS

!If the AttributeValueChange notification is defined for the managed object class using this package, this notification is emitted when the serviceList attribute changes value.!

A.4.2.34 Service Name Package

serviceNamePkg PACKAGE

BEHAVIOUR serviceNamePkgDefinition,
serviceNamePkgBehaviour;

ATTRIBUTES serviceName PERMITTED VALUES SYNTAX-1.AnyNameRange GET-REPLACE;

REGISTERED AS {x-package 34};

serviceNamePkgDefinition BEHAVIOUR

DEFINED AS

!Service Name attribute identifies any service that is supported by the resource.!

serviceNamePkgBehaviour BEHAVIOUR

DEFINED AS

!If the AttributeValueChange notification is defined for the managed object class using this package, this notification is emitted when the serviceName attribute changes value.!

A.4.2.35 Software List Package

softwareListPkg PACKAGE

BEHAVIOUR softwareListPkgDefinition,

softwareListPkgBehaviour;

ATTRIBUTES softwareList PERMITTED VALUES SYNTAX-1.AnyNamesRange GET-REPLACE ADD-REMOVE;

REGISTERED AS {x-package 35};

softwareListPkgDefinition BEHAVIOUR

DEFINED AS

!The Software List attribute identifies those software components that run on or are considered part of the resource.!

softwareListPkgBehaviour BEHAVIOUR

DEFINED AS

!If the AttributeValueChange notification is defined for the managed object class using this package, this notification is emitted when the softwareList attribute changes value.!

A.4.2.36 Software Name Package

softwareNamePkg PACKAGE

BEHAVIOUR softwareNamePkgDefinition,
softwareNamePkgBehaviour;

ATTRIBUTES softwareName PERMITTED VALUES SYNTAX-1.AnyNameRange GET-REPLACE;

REGISTERED AS {x-package 36};

softwareNamePkgDefinition BEHAVIOUR

DEFINED AS

!The Software Name attribute identifies the software component that runs on or are considered part of the resource.!

softwareNamePkgBehaviour BEHAVIOUR

DEFINED AS

!If the AttributeValueChange notification is defined for the managed object class using this package, this notification is emitted when the softwareName attribute changes value.!

A.4.2.37 System Time Package

systemTimePkg PACKAGE

BEHAVIOUR systemTimePkgDefinition,
systemTimePkgBehaviour;

ATTRIBUTES

systemTime PERMITTED VALUES SYNTAX-1.Integer32 GET;

REGISTERED AS {x-package 37};

systemTimePkgDefinition BEHAVIOUR

DEFINED AS

!This package records the current time clocked by the resource.!

systemTimePkgBehaviour BEHAVIOUR

DEFINED AS

!The attribute contained in this package is never the subject of an attribute value change notification. Even if the AttributeValueChange notification is defined for the managed object class using this package, this notification is NOT emitted when the systemTime attribute changes value.!

A.4.2.38 Type Text Package**typeTextPkg PACKAGE**

BEHAVIOUR typeTextPkgDefinition,
typeTextPkgBehaviour;

ATTRIBUTES

typeText PERMITTED VALUES SYNTAX-1.GraphicString32 GET-REPLACE;

REGISTERED AS {x-package 38};

typeTextPkgDefinition BEHAVIOUR**DEFINED AS**

!This package serves to supplement and refine individual managed object class attributes!;

typeTextPkgBehaviour BEHAVIOUR**DEFINED AS**

!If the attributeValueChange notification is defined for the managed object class using this package, this notification is emitted when the Type Text attribute changes value.!

A.4.2.39 Up Time Package**upTimePkg PACKAGE**

BEHAVIOUR upTimePkgDefinition,
upTimePkgBehaviour;

ATTRIBUTES

upTime PERMITTED VALUES SYNTAX-1.Integer32 GET;

REGISTERED AS {x-package 39};

upTimePkgDefinition BEHAVIOUR**DEFINED AS**

!This package records the elapsed time during which the underlying resource has been enabled.!

upTimePkgBehaviour BEHAVIOUR**DEFINED AS**

!The attribute contained in this package is never the subject of an attribute value change notification. Even if the AttributeValueChange notification is defined for the managed object class using this package, this notification is NOT emitted when the upTime attribute changes value.!

A.4.2.40 Usage State Package

usageStatePkg PACKAGE

BEHAVIOUR usageStatePkgDefinition,
usageStatePkgBehaviour;

ATTRIBUTES

"Rec. X.721 | ISO/IEC 10165-2 : 1992":usageState GET;

ATTRIBUTE GROUPS

"Rec. X.721 | ISO/IEC 10165-2 : 1992":state

"Rec. X.721 | ISO/IEC 10165-2 : 1992":usageState;

REGISTERED AS {x-package 40};

usageStatePkgDefinition BEHAVIOUR

DEFINED AS

!This package specifies the Usage State of the underlying resource, to be included in resources which are able to detect whether or not they are currently in use.!

usageStatePkgBehaviour BEHAVIOUR

DEFINED AS

!If the stateChange notification is defined for the managed object class using this package, this notification is emitted when the usageState attribute changes value.!

A.4.2.41 Vendor List Package

vendorListPkg PACKAGE

BEHAVIOUR vendorListPkgDefinition,
vendorListPkgBehaviour;

ATTRIBUTES vendorList PERMITTED VALUES SYNTAX-1.AnyNamesRange GET-REPLACE ADD-REMOVE;

REGISTERED AS {x-package 41};

vendorListPkgDefinition BEHAVIOUR

DEFINED AS

!The Vendor List attribute identifies the organization(s) from which the resource was obtained (e.g., purchased, leased, etc.)!

vendorListPkgBehaviour BEHAVIOUR

DEFINED AS

!If the AttributeValueChange notification is defined for the managed object class using this package, this notification is emitted when the vendorList attribute changes value.!

A.4.3 Name Bindings**A.4.3.1 Computer System Name Bindings**

```
computerSystem-system      NAME BINDING
  SUBORDINATE OBJECT CLASS  computerSystem AND SUBCLASSES;
  NAMED BY
  SUPERIOR OBJECT CLASS    "Rec. X.721 | ISO/IEC 10165-2 : 1992":system;
  WITH ATTRIBUTE            computerSystemId;
  CREATE WITH-REFERENCE-OBJECT, WITH-AUTOMATIC-INSTANCE-NAMING;
  DELETE ONLY-IF-NO-CONTAINED-OBJECTS;

REGISTERED AS {x-nameBinding 1};
```

```
computerSystem-opNetwork    NAME BINDING
  SUBORDINATE OBJECT CLASS  computerSystem AND SUBCLASSES;
  NAMED BY
  SUPERIOR OBJECT CLASS    opNetwork AND SUBCLASSES;
  WITH ATTRIBUTE            computerSystemId;
  CREATE WITH-REFERENCE-OBJECT, WITH-AUTOMATIC-INSTANCE-NAMING;
  DELETE ONLY-IF-NO-CONTAINED-OBJECTS;

REGISTERED AS {x-nameBinding 2};
```

```
computerSystem-computerSystem NAME BINDING
  SUBORDINATE OBJECT CLASS  computerSystem AND SUBCLASSES;
  NAMED BY
  SUPERIOR OBJECT CLASS    computerSystem AND SUBCLASSES;
  WITH ATTRIBUTE            computerSystemId;
  CREATE WITH-REFERENCE-OBJECT, WITH-AUTOMATIC-INSTANCE-NAMING;
  DELETE ONLY-IF-NO-CONTAINED-OBJECTS;

REGISTERED AS {x-nameBinding 3};
```

A.4.3.2 CO Transport Protocol Layer Entity Name Bindings

```
coTransportProtocollayerEntity-computerSystem NAME BINDING
  SUBORDINATE OBJECT CLASS  coTransportProtocollayerEntity AND SUBCLASSES;
  NAMED BY
  SUPERIOR OBJECT CLASS    computerSystem AND SUBCLASSES;
  WITH ATTRIBUTE            coTransportProtocollayerEntityId;
REGISTERED AS {x-nameBinding 4};
```

```
coTransportProtocollayerEntity-system NAME BINDING
  SUBORDINATE OBJECT CLASS  coTransportProtocollayerEntity AND SUBCLASSES;
  NAMED BY
  SUPERIOR OBJECT CLASS    "Rec. X.721 | ISO/IEC 10165-2 : 1992": system AND SUBCLASSES;
  WITH ATTRIBUTE            coTransportProtocollayerEntityId;
REGISTERED AS {x-nameBinding 5};
```

```
coTransportProtocollayerEntity-opEquipment NAME BINDING
  SUBORDINATE OBJECT CLASS  coTransportProtocollayerEntity AND SUBCLASSES;
  NAMED BY
  SUPERIOR OBJECT CLASS    opEquipment AND SUBCLASSES;
  WITH ATTRIBUTE            coTransportProtocollayerEntityId;
REGISTERED AS {x-nameBinding 6};
```

A.4.3.3 CL Network Protocol Layer Entity Name Bindings

```
clNetworkProtocollayerEntity-computerSystem NAME BINDING
  SUBORDINATE OBJECT CLASS  clNetworkProtocollayerEntity AND SUBCLASSES;
  NAMED BY
```

```

SUPERIOR OBJECT CLASS computerSystem AND SUBCLASSES;
WITH ATTRIBUTE clNetworkProtocolLayerEntityId;
REGISTERED AS {x-nameBinding 7};

clNetworkProtocolLayerEntity-system NAME BINDING
SUBORDINATE OBJECT CLASS clNetworkProtocolLayerEntity AND SUBCLASSES;
NAMED BY
SUPERIOR OBJECT CLASS
"Rec. X.721 | ISO/IEC 10165-2 : 1992": system AND SUBCLASSES;
WITH ATTRIBUTE clNetworkProtocolLayerEntityId;
REGISTERED AS {x-nameBinding 8};

clNetworkProtocolLayerEntity-opEquipment NAME BINDING
SUBORDINATE OBJECT CLASS clNetworkProtocolLayerEntity AND SUBCLASSES;
NAMED BY
SUPERIOR OBJECT CLASS opEquipment AND SUBCLASSES;
WITH ATTRIBUTE clNetworkProtocolLayerEntityId;
REGISTERED AS {x-nameBinding 9};

```

A.4.3.4 OMNIPoint Equipment Name Bindings

```

opEquipment-computerSystem NAME BINDING
SUBORDINATE OBJECT CLASS opEquipment AND SUBCLASSES;
NAMED BY
SUPERIOR OBJECT CLASS computerSystem AND SUBCLASSES;
WITH ATTRIBUTE "Rec. M.3100 : 1992":equipmentId;
CREATE WITH-REFERENCE-OBJECT, WITH-AUTOMATIC-INSTANCE-NAMING;
DELETE ONLY-IF-NO-CONTAINED-OBJECTS;

REGISTERED AS {x-nameBinding 10};

opEquipment-system NAME BINDING
SUBORDINATE OBJECT CLASS opEquipment AND SUBCLASSES;
NAMED BY
SUPERIOR OBJECT CLASS "Rec. X.721 | ISO/IEC 10165-2 : 1992":system;
WITH ATTRIBUTE "Rec. M.3100 : 1992":equipmentId;
CREATE WITH-REFERENCE-OBJECT, WITH-AUTOMATIC-INSTANCE-NAMING;
DELETE ONLY-IF-NO-CONTAINED-OBJECTS;

REGISTERED AS {x-nameBinding 11};

opEquipment-equipment NAME BINDING
SUBORDINATE OBJECT CLASS opEquipment AND SUBCLASSES;
NAMED BY
SUPERIOR OBJECT CLASS "Rec. M.3100 : 1992":equipment AND SUBCLASSES;
WITH ATTRIBUTE "Rec. M.3100 : 1992":equipmentId;
CREATE WITH-REFERENCE-OBJECT, WITH-AUTOMATIC-INSTANCE-NAMING;
DELETE ONLY-IF-NO-CONTAINED-OBJECTS;

REGISTERED AS {x-nameBinding 12};

opEquipment-opNetwork NAME BINDING
SUBORDINATE OBJECT CLASS opEquipment AND SUBCLASSES;
NAMED BY
SUPERIOR OBJECT CLASS opNetwork AND SUBCLASSES;
WITH ATTRIBUTE "Rec. M.3100 : 1992":equipmentId;
CREATE WITH-REFERENCE-OBJECT, WITH-AUTOMATIC-INSTANCE-NAMING;
DELETE ONLY-IF-NO-CONTAINED-OBJECTS;

REGISTERED AS {x-nameBinding 13};

```

A.4.3.5 OMNIPoint Network Name Bindings

-- The following name bindings are defined, in addition to those
-- inherited from Rec. M.3100 Network (which do not include CREATE/DELETE):

```
network-opNetwork-1      NAME BINDING
  SUBORDINATE OBJECT CLASS opNetwork AND SUBCLASSES;
  NAMED BY
  SUPERIOR OBJECT CLASS "Rec. M.3100 : 1992":network AND SUBCLASSES;
  WITH ATTRIBUTE "Rec. M.3100 : 1992":networkId;
  CREATE WITH-REFERENCE-OBJECT, WITH-AUTOMATIC-INSTANCE-NAMING;
  DELETE ONLY-IF-NO-CONTAINED-OBJECTS;
```

REGISTERED AS {x-nameBinding 14};

```
network-opNetwork-2      NAME BINDING
  SUBORDINATE OBJECT CLASS opNetwork AND SUBCLASSES;
  NAMED BY
  SUPERIOR OBJECT CLASS "Rec. M.3100 : 1992":network AND SUBCLASSES;
  WITH ATTRIBUTE networkTitle;
  CREATE WITH-REFERENCE-OBJECT, WITH-AUTOMATIC-INSTANCE-NAMING;
  DELETE ONLY-IF-NO-CONTAINED-OBJECTS;
```

REGISTERED AS {x-nameBinding 15};

```
opNetwork-root           NAME BINDING
  SUBORDINATE OBJECT CLASS opNetwork AND SUBCLASSES;
  NAMED BY SUPERIOR OBJECT CLASS "Rec. X.660 | ISO/IEC 9834-1 : 1992":root;
  WITH ATTRIBUTE networkTitle;
  CREATE WITH-REFERENCE-OBJECT, WITH-AUTOMATIC-INSTANCE-NAMING;
  DELETE ONLY-IF-NO-CONTAINED-OBJECTS;
```

REGISTERED AS {x-nameBinding 16};

A.4.3.6 Processing Entity Name Bindings

-- processingEntity-opEquipment NAME BINDING
-- processingEntity-computerSystem NAME BINDING
-- both inherited from opEquipment, no additional bindings required.

A.4.3.7 Transport Connection Name Bindings

```
transportConnection-coTransportProtocolLayerEntity NAME BINDING
  SUBORDINATE OBJECT CLASS transportConnection AND SUBCLASSES;
  NAMED BY
  SUPERIOR OBJECT CLASS coTransportProtocolLayerEntity AND SUBCLASSES;
  WITH ATTRIBUTE transportConnectionId;
  BEHAVIOUR transportConnectionNBBehaviour;
  DELETE DELETES-CONTAINED-OBJECTS;
REGISTERED AS {x-nameBinding 17};
```

transportConnectionNBBehaviour BEHAVIOUR

DEFINED AS

!The expected real effect of the DELETE operation when applied to an instance of the transport connection managed object class is that the underlying transport connection resource is aborted.!

A.4.4 Attributes**A.4.4.1 Active Connections**

activeConnections ATTRIBUTE

WITH ATTRIBUTE SYNTAX SYNTAX-1.IntegerBase;
MATCHES FOR EQUALITY, ORDERING;
BEHAVIOUR activeConnectionsBehaviour;

REGISTERED AS {x-attribute 1};

activeConnectionsBehaviour BEHAVIOUR

DEFINED AS

!The activeConnections attribute specifies the number of currently active transport connections (i.e., the number of transport connections which are in the open state [as defined for the underlying protocol machine], updated upon each connection establishment and release).!;

A.4.4.2 Addressing Size

addressingSize ATTRIBUTE

WITH ATTRIBUTE SYNTAX SYNTAX-1.AddressingSizeBase;
MATCHES FOR EQUALITY, ORDERING;
BEHAVIOUR addressingSizeBehaviour;

REGISTERED AS {x-attribute 2};

addressingSizeBehaviour BEHAVIOUR

DEFINED AS

!The Addressing Size attribute indicates the number of bits which represent an address to the Processing Entity's central processing unit (CPU).!;

A.4.4.3 Checksum PDUs Discarded Counter

checksumPDUsDiscardedCounter ATTRIBUTE

DERIVED FROM
"Rec. X.721 | ISO/IEC 10165-2 : 1992": counter;
BEHAVIOUR checksumPDUsDiscardedCounterBehaviour;

REGISTERED AS {x-attribute 3};

checksumPDUsDiscardedCounterBehaviour BEHAVIOUR

DEFINED AS

!The attribute specifies the number of well-formed PDUs rejected by the peer entity due to a checksum error.!;

A.4.4.4 Computer System Id

computerSystemId ATTRIBUTE

WITH ATTRIBUTE SYNTAX SYNTAX-1.GraphicStringBase;
MATCHES FOR EQUALITY, SUBSTRINGS;
BEHAVIOUR computerSystemIdBehaviour;

REGISTERED AS {x-attribute 4};

computerSystemIdBehaviour BEHAVIOUR

DEFINED AS

!The computerSystemId attribute is the distinguishing attribute for the computerSystem managed object class.!

A.4.4.5 CL Network Protocol Layer Entity Id

clNetworkProtocolLayerEntityId ATTRIBUTE

WITH ATTRIBUTE SYNTAX SYNTAX-1.GraphicStringBase;
MATCHES FOR EQUALITY, SUBSTRINGS;
BEHAVIOUR clNetworkProtocolLayerEntityIdBehaviour;

REGISTERED AS {x-attribute 5};

clNetworkProtocolLayerEntityIdBehaviour BEHAVIOUR

DEFINED AS

!The clNetworkProtocolLayerEntityId attribute is the distinguishing attribute for the clNetworkProtocolLayerEntity managed object class.!

A.4.4.6 CO Transport Protocol Layer Entity Id

coTransportProtocolLayerEntityId ATTRIBUTE

WITH ATTRIBUTE SYNTAX SYNTAX-1.GraphicStringBase;
MATCHES FOR EQUALITY, SUBSTRINGS;
BEHAVIOUR coTransportProtocolLayerEntityIdBehaviour;

REGISTERED AS {x-attribute 6};

coTransportProtocolLayerEntityIdBehaviour BEHAVIOUR

DEFINED AS

!The coTransportProtocolLayerEntityId attribute is the distinguishing attribute for the coTransportProtocolLayerEntity managed object class.!

A.4.4.7 Contact List

contactList ATTRIBUTE

WITH ATTRIBUTE SYNTAX SYNTAX-1.AnyNamesBase;
MATCHES FOR SET-COMPARISON, SET-INTERSECTION;
BEHAVIOUR contactListBehaviour;

REGISTERED AS {x-attribute 7};

contactListBehaviour BEHAVIOUR

DEFINED AS

!The Contact List attribute provides managed object instance information for one or more contacts. The following object classes (or any of their subclasses or allomorphic classes) are valid as contacts: "OP1 Library Vol. 4":Contact.

The SET-COMPARISON and/or SET-INTERSECTION matching rules may not be supported by some managed object instances which include this attribute.!

A.4.4.8 Contact Name

contactName **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX SYNTAX-1.AnyNameBase;
MATCHES FOR EQUALITY, SUBSTRINGS;
BEHAVIOUR contactNameBehaviour;

REGISTERED AS (x-attribute 8);

contactNameBehaviour **BEHAVIOUR**

DEFINED AS

!The Contact Name attribute provides information for one person or organization who can be contacted about the resource.!

A.4.4.9 CPU Type

cpuType **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX SYNTAX-1.GraphicStringBase;
MATCHES FOR EQUALITY, SUBSTRINGS;
BEHAVIOUR cpuTypeBehaviour;

REGISTERED AS (x-attribute 9);

cpuTypeBehaviour **BEHAVIOUR**

DEFINED AS

!The Central Processor Unit (CPU) Type attribute indicates the type of central processor unit found in a Processing Entity.!

A.4.4.10 CPU Utilization

cpuUtilization **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX SYNTAX-1.IntegerBase;
MATCHES FOR EQUALITY, ORDERING;
BEHAVIOUR cpuUtilizationBehaviour;

REGISTERED AS (x-attribute 10);

cpuUtilizationBehaviour **BEHAVIOUR**

DEFINED AS

!The cpuUtilization attribute specifies, as a percentage, the overall utilization of all central processor units found in a processing entity. The percentage is expressed as an integer with permissible values in the range of 0 to 100.!

A.4.4.11 Customer List

customerList **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX SYNTAX-1.AnyNamesBase;
MATCHES FOR SET-COMPARISON, SET-INTERSECTION;
BEHAVIOUR customerListBehaviour;

REGISTERED AS (x-attribute 11);

customerListBehaviour **BEHAVIOUR**

DEFINED AS

!The Customer List attribute provides managed object instance information about one or more customers. The following classes (or any of their subclasses or allomorphic classes) are valid as customers: "OP1 Library Vol. 4":Customer.

The SET-COMPARISON and/or SET-INTERSECTION matching rules may not be supported by some managed object instances which include this attribute.!

A.4.4.12 Customer Name

customerName ATTRIBUTE

WITH ATTRIBUTE SYNTAX SYNTAX-1.AnyNameBase;
MATCHES FOR EQUALITY, SUBSTRINGS;
BEHAVIOUR customerNameBehaviour;

REGISTERED AS {x-attribute 12};

customerNameBehaviour BEHAVIOUR

DEFINED AS

!The Customer Name attribute provides information about one customer.!

A.4.4.13 Endianess

endianess ATTRIBUTE

WITH ATTRIBUTE SYNTAX SYNTAX-1.Endianess;
MATCHES FOR EQUALITY, ORDERING;
BEHAVIOUR endianessBehaviour;

REGISTERED AS {x-attribute 13};

endianessBehaviour BEHAVIOUR

DEFINED AS

!The Endianess attribute indicates the bit order (big endian, little endian) used by the Processing Entity's central processing unit (CPU).!

A.4.4.14 Function List

functionList ATTRIBUTE

WITH ATTRIBUTE SYNTAX SYNTAX-1.AnyNamesBase;
MATCHES FOR SET-COMPARISON, SET-INTERSECTION;
BEHAVIOUR functionListBehaviour;

REGISTERED AS {x-attribute 14};

functionListBehaviour BEHAVIOUR

DEFINED AS

!The Function List attribute provides managed object instance information about one or more functions. The following managed object classes (or any of their subclasses or allomorphic classes) are valid as functions: "OP1 Library Vol. 4":Function.

The SET-COMPARISON and/or SET-INTERSECTION matching rules may not be supported by some managed object instances which include this attribute.!

A.4.4.15 Function Name

functionName **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX SYNTAX-1.AnyNameBase;
MATCHES FOR EQUALITY, SUBSTRINGS;
BEHAVIOUR functionNameBehaviour;

REGISTERED AS {x-attribute 15};

functionNameBehaviour **BEHAVIOUR**

DEFINED AS

!The Function Name attribute provides information about one function.!

A.4.4.16 Inactivity Time

inactivityTime **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX SYNTAX-1.HundredthsOfSec;
MATCHES FOR EQUALITY, ORDERING;
BEHAVIOUR inactivityTimeBehaviour;

REGISTERED AS {x-attribute 16};

inactivityTimeBehaviour **BEHAVIOUR**

DEFINED AS

!This attribute specifies the amount of time (in 1/100ths of a second) that the transport connection has been inactive.!

A.4.4.17 Inactivity Timeout

inactivityTimeout **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX SYNTAX-1.HundredthsOfSec;
MATCHES FOR EQUALITY, ORDERING;
BEHAVIOUR inactivityTimeoutBehaviour;

REGISTERED AS {x-attribute 17};

inactivityTimeoutBehaviour **BEHAVIOUR**

DEFINED AS

!This attribute specifies the maximum amount of time (in 1/100ths of a second) that the transport connection can remain enabled when there is no activity (i.e., data flow) on it. A value of 0 for this attribute indicates that an inactivity timeout is not supported on the transport connection.!

A.4.4.18 Local Network Address

localNetworkAddress **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX SYNTAX-1.Address;
MATCHES FOR EQUALITY, SUBSTRINGS;
BEHAVIOUR localNetworkAddressBehaviour;

REGISTERED AS {x-attribute 18};

localNetworkAddressBehaviour **BEHAVIOUR**

DEFINED AS

!The localNetworkAddress attribute identifies the local network address supported by a network protocol layer entity (e.g., local IP address for TCP or the local NSAP address for OSI).!

A.4.4.19 Local Network Addresses

localNetworkAddresses **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX SYNTAX-1.NetworkAddresses;
MATCHES FOR SET-COMPARISON, SET-INTERSECTION;
BEHAVIOUR localNetworkAddressesBehaviour;

REGISTERED AS {x-attribute 19};

localNetworkAddressesBehaviour **BEHAVIOUR**

DEFINED AS

!The localNetworkAddresses attribute identifies the local network addresses supported by a network protocol layer entity (e.g., local IP address for TCP or the local NSAP address for OSI).

Set comparison and/or set intersection matching rules may not be supported by some managed object instances which include this attribute.!

A.4.4.20 Local Transport Addresses

localTransportAddresses **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX SYNTAX-1.TransportAddresses;
MATCHES FOR SET-COMPARISON, SET-INTERSECTION;
BEHAVIOUR localTransportAddressesBehaviour;

REGISTERED AS {x-attribute 20};

localTransportAddressesBehaviour **BEHAVIOUR**

DEFINED AS

!The localTransportAddresses attribute specifies the set of local transport addresses (e.g., local TSAP identifiers) that a connection oriented transport protocol layer entity provides to its users. A transport address consists of a transport connection endpoint and a network address.

Set comparison and/or set intersection matching rules may not be supported by some managed object instances which include this attribute.!

A.4.4.21 Local Transport Connection Endpoint

localTransportConnectionEndpoint **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX SYNTAX-1.Address;
MATCHES FOR EQUALITY, SUBSTRINGS;
BEHAVIOUR localTransportConnectionEndpointBehaviour;

REGISTERED AS {x-attribute 21};

localTransportConnectionEndpointBehaviour **BEHAVIOUR**

DEFINED AS

!This attribute identifies the local transport connection endpoint (e.g., the source port for TCP or the local t-selector for OSI Transport protocol).!

A.4.4.22 Location Pointer

locationPointer **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX SYNTAX-1.ObjectInstance;
MATCHES FOR EQUALITY;

BEHAVIOUR locationPointerBehaviour;

REGISTERED AS (x-attribute 22);

locationPointerBehaviour BEHAVIOUR

DEFINED AS

!The Location Pointer attribute provides managed object instance information for a location (e.g., Hilo Hawaii USA). The following managed object classes (or any of their subclasses or allomorphic classes) are valid as locations: "OP1 Library Vol. 4":Location.!

A.4.4.23 Manufacturer List

manufacturerList ATTRIBUTE

WITH ATTRIBUTE SYNTAX SYNTAX-1.AnyNamesBase;
MATCHES FOR SUBSTRINGS, SET-COMPARISON, SET-INTERSECTION;
BEHAVIOUR manufacturerListBehaviour;

REGISTERED AS (x-attribute 23);

manufacturerListBehaviour BEHAVIOUR

DEFINED AS

!The manufacturerList attribute indicates information about the manufacturer(s) that manufactured the underlying resource. This attribute contains object instance name(s) for "OP1 Library Vol. 4":manufacturer (or any subclass or allomorphic class).

Set comparison and/or set intersection matching rules may not be supported by some managed object instances which include this attribute.!

A.4.4.24 Manufacturer Name

manufacturerName ATTRIBUTE

WITH ATTRIBUTE SYNTAX SYNTAX-1.AnyNameBase;
MATCHES FOR EQUALITY, SUBSTRINGS;
BEHAVIOUR manufacturerNameBehaviour;

REGISTERED AS (x-attribute 24);

manufacturerNameBehaviour BEHAVIOUR

DEFINED AS

!The manufacturerName attribute indicates information about the manufacturer(s) that manufactured the underlying resource. This attribute contains descriptive text.!

A.4.4.25 Max Connections

maxConnections ATTRIBUTE

WITH ATTRIBUTE SYNTAX SYNTAX-1.IntegerBase;
MATCHES FOR EQUALITY, ORDERING;
BEHAVIOUR maxConnectionsBehaviour;

REGISTERED AS (x-attribute 25);

maxConnectionsBehaviour BEHAVIOUR

DEFINED AS

!The maxConnections attribute specifies the maximum number of simultaneously active/open transport connections that can be supported by the transport protocol layer entity.!

A.4.4.26 Max PDU Size

maxPDUSize ATTRIBUTE

WITH ATTRIBUTE SYNTAX SYNTAX-1.IntegerBase;
MATCHES FOR EQUALITY, ORDERING;
BEHAVIOUR maxPDUSizeBehaviour;

REGISTERED AS {x-attribute 26};

maxPDUSizeBehaviour BEHAVIOUR

DEFINED AS

!The maxPDUSize attribute specifies the maximum length of a PDU that can be supported by the local layer entity.!

A.4.4.27 Max Retransmissions

maxRetransmissions ATTRIBUTE

WITH ATTRIBUTE SYNTAX SYNTAX-1.IntegerBase;
MATCHES FOR EQUALITY, ORDERING;
BEHAVIOUR maxRetransmissionsBehaviour;

REGISTERED AS {x-attribute 27};

maxRetransmissionsBehaviour BEHAVIOUR

DEFINED AS

!This attribute specifies the maximum number of times a TPDU is to be retransmitted before the transport connection is aborted.!

A.4.4.28 Memory Size

memorySize ATTRIBUTE

WITH ATTRIBUTE SYNTAX SYNTAX-1.MemorySizeBase;
MATCHES FOR EQUALITY, ORDERING;
BEHAVIOUR memorySizeBehaviour;

REGISTERED AS {x-attribute 28};

memorySizeBehaviour BEHAVIOUR

DEFINED AS

!The Memory Size attribute indicates, in kilobytes, the amount of memory available to a Processing Entity (irrespective of its current usage).!

A.4.4.29 Memory Utilization

memoryUtilization ATTRIBUTE

WITH ATTRIBUTE SYNTAX SYNTAX-1.IntegerBase;

MATCHES FOR EQUALITY, ORDERING;
BEHAVIOUR memoryUtilizationBehaviour;

REGISTERED AS {x-attribute 29};

memoryUtilizationBehaviour BEHAVIOUR

DEFINED AS
!The memoryUtilization attribute specifies, as a percentage, the overall utilization of amount of memory available to a processing entity. The percentage is expressed as an integer with permissible values in the range of 0 to 100.!

A.4.4.30 Network Entity Type

networkEntityType ATTRIBUTE

WITH ATTRIBUTE SYNTAX SYNTAX-1.NetworkEntityType;
MATCHES FOR EQUALITY;
BEHAVIOUR networkEntityTypeBehaviour;

REGISTERED AS {x-attribute 30};

networkEntityTypeBehaviour BEHAVIOUR

DEFINED AS
!The networkEntityType attribute indicates the type of the network protocol layer entity.!

A.4.4.31 Network Title

networkTitle ATTRIBUTE

DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2 : 1992":systemTitle;
BEHAVIOUR networkTitleBehaviour;

REGISTERED AS {x-attribute 31};

networkTitleBehaviour BEHAVIOUR

DEFINED AS
!The Network Title is one of the distinguishing attribute the network managed object class for use as described in clause 6.3 of [MIM]!;

A.4.4.32 NPDU Time To Live

nPDUTimeToLive ATTRIBUTE

WITH ATTRIBUTE SYNTAX SYNTAX-1.IntegerBase;
MATCHES FOR EQUALITY, ORDERING;
BEHAVIOUR nPDUTimeToLiveBehaviour;

REGISTERED AS {x-attribute 32};

nPDUTimeToLiveBehaviour BEHAVIOUR

DEFINED AS
!This attribute specifies the maximum amount of time (in units of 10 ms) that an NPDU can exist in the network. This attribute is used to limit the lifetime of NPDUs during unstable network situations.!

A.4.4.33 OMNIPoint Equipment List

opEquipmentList ATTRIBUTE

WITH ATTRIBUTE SYNTAX SYNTAX-1.AnyNamesBase;
MATCHES FOR SET-COMPARISON, SET-INTERSECTION;
BEHAVIOUR opEquipmentListBehaviour;

REGISTERED AS {x-attribute 33};

opEquipmentListBehaviour BEHAVIOUR

DEFINED AS

!The OMNIPoint Equipment List attribute provides managed object instance information about one or more pieces of opEquipment. The following classes (or any of their subclasses or allomorphic classes) are valid as equipment: OMNIPoint Equipment.

The SET-COMPARISON and/or SET-INTERSECTION matching rules may not be supported by some managed object instances which include this attribute.!

A.4.4.34 OMNIPoint Network List

opNetworkList ATTRIBUTE

WITH ATTRIBUTE SYNTAX SYNTAX-1.AnyNamesBase;
MATCHES FOR SET-COMPARISON, SET-INTERSECTION;
BEHAVIOUR opNetworkListBehaviour;

REGISTERED AS {x-attribute 34};

opNetworkListBehaviour BEHAVIOUR

DEFINED AS

!The OMNIPoint Network List attribute shall provide managed object instance information about a set of networks. The following object classes (or any of their subclasses or allomorphic classes) are valid as networks: OMNIPoint Network.

The SET-COMPARISON and/or SET-INTERSECTION matching rules may not be supported by some managed object instances which include this attribute.!

A.4.4.35 OMNIPoint Network Name

opNetworkName ATTRIBUTE

WITH ATTRIBUTE SYNTAX SYNTAX-1.AnyNameBase;
MATCHES FOR EQUALITY, SUBSTRINGS;
BEHAVIOUR opNetworkNameBehaviour;

REGISTERED AS {x-attribute 35};

opNetworkNameBehaviour BEHAVIOUR

DEFINED AS

!The OMNIPoint Network Name attribute shall provide information about a network.!

A.4.4.36 Operating System Information

osInfo ATTRIBUTE

WITH ATTRIBUTE SYNTAX SYNTAX-1.OsInfoBase;
MATCHES FOR EQUALITY, SET-COMPARISON, SET-INTERSECTION;
BEHAVIOUR osInfoBehaviour;

REGISTERED AS {x-attribute 36};

osInfoBehaviour BEHAVIOUR

DEFINED AS

!The Operating System Information attribute specifies the names and releases of the supported operating systems.

The SET-COMPARISON and/or SET-INTERSECTION matching rules may not be supported by some managed object instances which include this attribute.!

A.4.4.37 PDUs Forwarded Counter

pdusForwardedCounter ATTRIBUTE

DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2 : 1992": counter;
BEHAVIOUR pdusForwardedCounterBehaviour;

REGISTERED AS {x-attribute 37};

pdusForwardedCounterBehaviour BEHAVIOUR

DEFINED AS

!This attribute specifies the number of valid incoming PDUs which were forwarded (transmitted as outgoing PDUs) to another destination. This attribute does not count incoming PDUs which were delivered to a local service user.!

A.4.4.38 PDUs Reassembled Ok Counter

pdusReasmbldOKCounter ATTRIBUTE

DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2 : 1992": counter;
BEHAVIOUR pdusReasmbldOKCounterBehaviour;

REGISTERED AS {x-attribute 38};

pdusReasmbldOKCounterBehaviour BEHAVIOUR

DEFINED AS

!This attribute specifies the number of PDUs that were reassembled successfully by a protocol layer entity.!

A.4.4.39 PDUs Reassembled Fail Counter

pdusReasmbldFailCounter ATTRIBUTE

DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2 : 1992": counter;
BEHAVIOUR pdusReasmbldFailCounterBehaviour;

REGISTERED AS {x-attribute 39};

pdusReasmbldFailCounterBehaviour BEHAVIOUR

DEFINED AS

!This attribute specifies the number of valid PDUs received by a protocol layer entity but discarded due to reassembly failure. This attribute counts only incoming PDUs which were recognized as valid segments of an SDU, but which were discarded during reassembly (for example, due to reassembly time expiration).!

A.4.4.40 PDUs Discarded Counter

pdusDiscardedCounter ATTRIBUTE

DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2 : 1992": counter;
BEHAVIOUR pdusDiscardedCounterBehaviour;

REGISTERED AS {x-attribute 40};

pdusDiscardedCounterBehaviour BEHAVIOUR

DEFINED AS

!This attribute specifies the number of invalid PDUs received and discarded by a protocol layer entity.!;

A.4.4.41 Peripheral List

peripheralList **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX SYNTAX-1.AnyNamesBase;
MATCHES FOR EQUALITY, SET-COMPARISON, SET-INTERSECTION;
BEHAVIOUR peripheralListBehaviour;

REGISTERED AS {x-attribute 41};

peripheralListBehaviour **BEHAVIOUR**

DEFINED AS

!The Peripheral List attribute provides managed object instance information for peripheral devices accessible by a resource.

The Peripheral List attribute identifies the auxiliary devices that are used by a resource. This includes things such as disk drives, tape drives, printers, etc.

The following object classes (or their subclasses or allomorphic classes) are valid processing entities: OMNIPoint Equipment.

The SET-COMPARISON and/or SET-INTERSECTION matching rules may not be supported by some managed object instances which include this attribute.!

A.4.4.42 Peripheral Name

peripheralName **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX SYNTAX-1.AnyNameBase;
MATCHES FOR EQUALITY, SUBSTRINGS;
BEHAVIOUR peripheralNameBehaviour;

REGISTERED AS {x-attribute 42};

peripheralNameBehaviour **BEHAVIOUR**

DEFINED AS

!The Peripheral Name attribute provides information for peripheral devices accessible by a resource.

The Peripheral Name attribute identifies an auxiliary devices that is used by a resource. This includes things such as disk drives, tape drives, printers, etc.!

A.4.4.43 Processing Entity List

processingEntityList **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX SYNTAX-1.AnyNamesBase;
MATCHES FOR EQUALITY, SET-COMPARISON, SET-INTERSECTION;
BEHAVIOUR processingEntityListBehaviour;

REGISTERED AS {x-attribute 43};

processingEntityListBehaviour **BEHAVIOUR**

DEFINED AS

!The Processing Entity List attribute specifies the processing entities which may be used by the containing object instance but which are not contained in (i.e., processing entities which are shared among systems). The following object classes (or their subclasses or allomorphic classes) are valid processing entities: processingEntity.

The SET-COMPARISON and/or SET-INTERSECTION matching rules may not be supported by some managed object instances which include this attribute.!

A.4.4.44 Processing Entity Name

processingEntityName ATTRIBUTE

WITH ATTRIBUTE SYNTAX SYNTAX-1.AnyNameBase;
MATCHES FOR EQUALITY, SUBSTRINGS;
BEHAVIOUR processingEntityNameBehaviour;

REGISTERED AS {x-attribute 44};

processingEntityNameBehaviour BEHAVIOUR

DEFINED AS

!The Processing Entity Name attribute specifies the processing entity which may be used by the containing object instance but which is not contained in (i.e., processing entities which are shared among systems).!;

A.4.4.45 Product Label

productLabel ATTRIBUTE

WITH ATTRIBUTE SYNTAX SYNTAX-1.GraphicStringBase;
MATCHES FOR EQUALITY, SUBSTRINGS;
BEHAVIOUR productLabelBehaviour;

REGISTERED AS {x-attribute 45};

productLabelBehaviour BEHAVIOUR

DEFINED AS

!The productLabel attribute specifies the product number or identifying string (e.g., model number) of the underlying resource.!

A.4.4.46 Remote Network Address

remoteNetworkAddress ATTRIBUTE

WITH ATTRIBUTE SYNTAX SYNTAX-1.Address;
MATCHES FOR EQUALITY, SUBSTRINGS;
BEHAVIOUR remoteNetworkAddressBehaviour;

REGISTERED AS {x-attribute 46};

remoteNetworkAddressBehaviour BEHAVIOUR

DEFINED AS

!The remoteNetworkAddress attribute identifies the remote network address of a transport connection (e.g., remote IP address for TCP or the remote NSAP address for OSI).!;

A.4.4.47 Remote Transport Connection Endpoint

remoteTransportConnectionEndpoint ATTRIBUTE

WITH ATTRIBUTE SYNTAX SYNTAX-1.Address;
MATCHES FOR EQUALITY, SUBSTRINGS;
BEHAVIOUR remoteTransportConnectionEndpointBehaviour;

REGISTERED AS {x-attribute 47};

remoteTransportConnectionEndpointBehaviour BEHAVIOUR

DEFINED AS

!This attribute identifies the remote transport connection endpoint (e.g., the destination port for TCP or the remote t-selector for OSI Transport protocol).!;

A.4.4.48 Retransmission Time

retransmissionTime ATTRIBUTE

WITH ATTRIBUTE SYNTAX SYNTAX-1.HundredthsOfSec;
MATCHES FOR EQUALITY, ORDERING;
BEHAVIOUR retransmissionTimeBehaviour;

REGISTERED AS {x-attribute 48};

retransmissionTimeBehaviour BEHAVIOUR

DEFINED AS

!This attribute specifies the current value (in 1/100ths of a second) of the retransmission timer used by a transport connection.!;

A.4.4.49 Retransmission Timer Initial Value

retransmissionTimerInitialValue ATTRIBUTE

WITH ATTRIBUTE SYNTAX SYNTAX-1.HundredthsOfSec;
MATCHES FOR EQUALITY, ORDERING;
BEHAVIOUR retransmissionTimerInitialValueBehaviour;

REGISTERED AS {x-attribute 49};

retransmissionTimerInitialValueBehaviour BEHAVIOUR

DEFINED AS

!This attribute specifies the initial value (in 1/100ths of a second) of the retransmission timer used by a transport connection.!;

A.4.4.50 Serial Number

serialNumber ATTRIBUTE

WITH ATTRIBUTE SYNTAX SYNTAX-1.GraphicStringBase;
MATCHES FOR EQUALITY, SUBSTRINGS;
BEHAVIOUR serialNumberBehaviour;

REGISTERED AS {x-attribute 50};

serialNumberBehaviour BEHAVIOUR

DEFINED AS

!The serialNumber attribute provides the serial number of the underlying resource.!;

A.4.4.51 Service List

serviceList **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX SYNTAX-1.AnyNamesBase;
MATCHES FOR SET-COMPARISON, SET-INTERSECTION;
BEHAVIOUR serviceListBehaviour;

REGISTERED AS {x-attribute 51};

serviceListBehaviour **BEHAVIOUR**

DEFINED AS

!The Service List attribute provides managed object instance information about one or more services. The following object classes (or any of their subclasses or allomorphic classes) are valid as services: "OP1 Library Vol. 4":Service.

The SET-COMPARISON and/or SET-INTERSECTION matching rules may not be supported by some managed object instances which include this attribute.!

A.4.4.52 Service Name

serviceName **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX SYNTAX-1.AnyNameBase;
MATCHES FOR EQUALITY, SUBSTRINGS;
BEHAVIOUR serviceNameBehaviour;

REGISTERED AS {x-attribute 52};

serviceNameBehaviour **BEHAVIOUR**

DEFINED AS

!The Service Name attribute provides information about one service.!

A.4.4.53 Software List

softwareList **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX SYNTAX-1.AnyNamesBase;
MATCHES FOR SET-COMPARISON, SET-INTERSECTION;
BEHAVIOUR softwareListBehaviour;

REGISTERED AS {x-attribute 53};

softwareListBehaviour **BEHAVIOUR**

DEFINED AS

!The Software List attribute identifies those software components that run on or are considered part of the equipment. (There is no corresponding managed object class at this time.)

The SET-COMPARISON and/or SET-INTERSECTION matching rules may not be supported by some managed object instances which include this attribute.!

A.4.4.54 Software Name

softwareName **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX SYNTAX-1.AnyNameBase;
MATCHES FOR EQUALITY, SUBSTRINGS;
BEHAVIOUR softwareNameBehaviour;

REGISTERED AS {x-attribute 54};

softwareNameBehaviour BEHAVIOUR

DEFINED AS

!The Software Name attribute identifies the software component that runs on or are considered part of the equipment.!

A.4.4.55 System Time

systemTime ATTRIBUTE

WITH ATTRIBUTE SYNTAX SYNTAX-1.GeneralTime;
MATCHES FOR EQUALITY, ORDERING;
BEHAVIOUR systemTimeBehaviour;

REGISTERED AS {x-attribute 55};

systemTimeBehaviour BEHAVIOUR

DEFINED AS

!The systemTime attribute specifies the current time clocked at the resource.!

A.4.4.56 Transport Connection Id

transportConnectionId ATTRIBUTE

WITH ATTRIBUTE SYNTAX SYNTAX-1.GraphicStringBase;
MATCHES FOR EQUALITY, SUBSTRINGS;
BEHAVIOUR transportConnectionIdBehaviour;

REGISTERED AS {x-attribute 56};

transportConnectionIdBehaviour BEHAVIOUR

DEFINED AS

!The transportConnectionId attribute is the distinguishing attribute for the transportConnection managed object class.!

A.4.4.57 Transport Connection Reference

transportConnectionReference ATTRIBUTE

WITH ATTRIBUTE SYNTAX SYNTAX-1.IntegerBase;
MATCHES FOR EQUALITY;
BEHAVIOUR transportConnectionReferenceBehaviour;

REGISTERED AS {x-attribute 57};

transportConnectionReferenceBehaviour BEHAVIOUR

DEFINED AS

!This attribute identifies the local transport connection reference that is established by the two transport connection endpoints (e.g., the local socket number for TCP or the local connection reference for OSI).!

A.4.4.58 Transport Entity Type

transportEntityType ATTRIBUTE

WITH ATTRIBUTE SYNTAX SYNTAX-1.TransportEntityType;
MATCHES FOR EQUALITY;

BEHAVIOUR transportEntityTypeBehaviour;

REGISTERED AS (x-attribute 58);

transportEntityTypeBehaviour BEHAVIOUR

DEFINED AS

!The transportEntityType attribute indicates the type of the transport protocol layer entity.!

A.4.4.59 Type Text

typeText ATTRIBUTE

WITH ATTRIBUTE SYNTAX SYNTAX-1.GraphicStringBase;

MATCHES FOR EQUALITY, SUBSTRINGS;

BEHAVIOUR typeTextBehaviour;

REGISTERED AS (x-attribute 59);

typeTextBehaviour BEHAVIOUR

DEFINED AS

!The typeText attribute serves to supplement and refine individual managed object class attributes. If none of the named items defined for the "type" attribute are appropriate, or the "type" attribute requires refinement, the typeText attribute contains supplemental information.!

A.4.4.60 Up Time

upTime ATTRIBUTE

WITH ATTRIBUTE SYNTAX SYNTAX-1.IntegerBase;

MATCHES FOR EQUALITY, ORDERING;

BEHAVIOUR upTimeBehaviour;

REGISTERED AS (x-attribute 60);

upTimeBehaviour BEHAVIOUR

DEFINED AS

!The upTime attribute specifies the time interval (in seconds) that has elapsed since the entity's operational state changed to "enabled", or since the time that the entity was created in the "enabled" state.!

A.4.4.61 Vendor List

vendorList ATTRIBUTE

WITH ATTRIBUTE SYNTAX SYNTAX-1.AnyNamesBase;

MATCHES FOR SET-COMPARISON, SET-INTERSECTION;

BEHAVIOUR vendorListBehaviour;

REGISTERED AS (x-attribute 61);

vendorListBehaviour BEHAVIOUR

DEFINED AS

!The Vendor List attribute provides managed object instance information about a set of vendor organizations. The following classes (or any of their subclasses or allomorphic classes) are valid as vendors: "OP1 Library Vol. 4":Vendor.

The SET-COMPARISON and/or SET-INTERSECTION matching rules may not be supported by some managed object instances which include this attribute.!

A.4.5 Actions

A.4.5.1 Activate

-- Copied from ISO/IEC DIS 10737, should be replaced by reference to standard
-- definition when/if this ACTION is registered in a final IS version.

activate ACTION

BEHAVIOUR activateBehaviour;
MODE CONFIRMED;
WITH REPLY SYNTAX SYNTAX-1.ActivateActionReply;

REGISTERED AS { x-action 1 };

activateBehaviour BEHAVIOUR

DEFINED AS

!This action initializes the operation of the resource. As a result of the action, the sequence of operations necessary to cause the resource to enter its operational mode shall be initiated. These may include, for example, checks against attribute constraint violation and checks on the validity of relationship attributes (cross-layer and other). If these operations are successfully initiated, the administrative state (if present) shall be changed to "unlocked" and the value "successResponse" shall be returned in the responseCode parameter of the action reply. If these operations cannot be successfully initiated, the value "failureResponse" shall be returned, together with a failure reason parameter describing the reason for the failure. On successful completion of these operations, the operational state shall have the value "enabled". Depending upon the current state of the resource when the action is attempted, some or all of the above operations may be unnecessary.!

A.4.5.2 Deactivate

-- Copied from ISO/IEC DIS 10737, should be replaced by reference to standard
-- definition when/if this ACTION is registered in a final IS version.

deactivate ACTION

BEHAVIOUR deactivateBehaviour;
MODE CONFIRMED;
WITH REPLY SYNTAX SYNTAX-1.ActivateActionReply;

REGISTERED AS { x-action 2 };

deactivateBehaviour BEHAVIOUR

DEFINED AS

!This action terminates the operation of the resource. As a result of the action, the sequence of operations necessary to cause the resource to cease operation shall be initiated. If these operations are successfully initiated, the administrative state (if present) shall be changed to "locked" and the value "successResponse" shall be returned in the responseCode parameter of the action reply. If these operations cannot be successfully initiated, the value "failureResponse" shall be returned, together with a failure reason parameter describing the reason for the failure.

On successful completion of these operations, the operational state shall have the value "disabled". Depending upon the current state of the resource when the action is attempted, some or all of the above operations may be unnecessary.!

A.4.6 Parameters

A.4.6.1 Transport Disconnect Cause

transportDisconnectCause PARAMETER

CONTEXT EVENT-INFO;
WITH SYNTAX SYNTAX-1.Cause;
BEHAVIOUR causeBehaviour;

REGISTERED AS { x-parameter 1 };

causeBehaviour BEHAVIOUR

DEFINED AS

!This parameter specifies the reason why a transport connection was deleted. It may be included in the Additional Information parameter of the objectDeletion notification.!

A.4.7 Syntax Definitions

SYNTAX-1 { x-module 1 }

DEFINITIONS IMPLICIT TAGS ::= BEGIN

IMPORTS DistinguishedName FROM InformationFramework {joint-iso-ccitt ds(5) modules(1)}

informationFramework(1)} ObjectInstance FROM CMIP-1 {joint-iso-ccitt ms(9) cmip(1) modules(0) protocol(3)};

-- EXPORTS everything

-- The following OIDs are allocated from the OIW NMSIG registration arc,
-- for use in registering harmonized OIW/NMF definitions.

nmsig	OBJECT IDENTIFIER ::= { iso identified-organization(3) oiw(14) nmsig(2) }
op1LibraryVol1	OBJECT IDENTIFIER ::= { nmsig 2 }
x-module	OBJECT IDENTIFIER ::= { op1LibraryVol1 0 }
x-objectClass	OBJECT IDENTIFIER ::= { op1LibraryVol1 1 }
x-package	OBJECT IDENTIFIER ::= { op1LibraryVol1 2 }
x-nameBinding	OBJECT IDENTIFIER ::= { op1LibraryVol1 3 }
x-attribute	OBJECT IDENTIFIER ::= { op1LibraryVol1 4 }
x-attributeGroup	OBJECT IDENTIFIER ::= { op1LibraryVol1 5 }
x-parameter	OBJECT IDENTIFIER ::= { op1LibraryVol1 6 }
x-action	OBJECT IDENTIFIER ::= { op1LibraryVol1 7 }
x-notification	OBJECT IDENTIFIER ::= { op1LibraryVol1 8 }
x-responseCode	OBJECT IDENTIFIER ::= { op1LibraryVol1 9 }

-- By convention, the postfix "base" is used when defining base types which appear
-- as syntax labels in ATTRIBUTE templates and the postfix "range" is used when defining
-- constrained types which appear as syntax labels in PERMITTED VALUES clauses.

```

ActivateActionReply ::= SEQUENCE {
    responseCode      OBJECT IDENTIFIER,
    responseArgs      SET OF Parameter OPTIONAL
}

-- OBJECT IDENTIFIER values used with ActivateActionReply --
failureResponse OBJECT IDENTIFIER ::= { x-responseCode 1 }
successResponse  OBJECT IDENTIFIER ::= { x-responseCode 2 }

Address ::= OCTET STRING

AddressingSizeBase ::= CHOICE {
    unknown      NULL,
    addressingSize IntegerBase -- measured in bits
}

AddressingSizeRange ::= CHOICE {
    unknown      NULL,
    addressingSize IntegerBase (1..64) -- measured in bits
}

AnyNamesBase ::= SET OF ObjectInstance
AnyNameBase  ::= GraphicStringBase

AnyNamesRange ::= SET SIZE(0..64) OF ObjectInstance
AnyNameRange  ::= GraphicString64

Cause ::= SEQUENCE {
    who INTEGER {
        unknown      (0),
        user          (1),
        provider      (2)
    },
    why INTEGER {
        unknown      (0),
        excessiveIdle (1),
        excessiveRetransmissions (2)
    }
}

Endianess ::= ENUMERATED {
    big      (1),
    little   (2)
}

EquipmentIdRange ::= CHOICE {
    -- based on "Rec. M.3100 : 1992" ASN.1 Module NameType
    numericName Integer32,
    pString      GraphicString64
}

GeneralTime ::= GeneralizedTime

GraphicStringBase ::= GraphicString
GraphicString16  ::= GraphicStringBase(SIZE(0..16))

```



```

GraphicString32      ::= GraphicStringBase(SIZE(0..32))
GraphicString64      ::= GraphicStringBase(SIZE(0..64))

HundredthsOfSec      ::= IntegerBase

IntegerBase          ::= INTEGER
Integer32            ::= IntegerBase(0..4294967295)

MemorySizeBase       ::= CHOICE (
    unknown NULL,
    size          IntegerBase -- measured in kilobytes
)

MemorySizeRange ::= CHOICE (
    unknown NULL,
    size          Integer32   -- measured in kilobytes
)

NetworkEntityType    ::= INTEGER ( other      (0),
    oSI-clnp          (1),
    internet-IP       (2)
) (0..255)

NetworkAddresses     ::= SET OF Address

OsInfoBase           ::= SET OF SEQUENCE
    (
        osName      GraphicStringBase,
        osRelease   GraphicStringBase
    )

OsInfoRange          ::= SET OF SEQUENCE
    (
        osName      GraphicString64,
        osRelease   GraphicString64
    )

Parameter            ::= SEQUENCE (
    paramId      OBJECT IDENTIFIER,
    paramInfo    ANY DEFINED BY paramId
)

PercentageRange      ::= IntegerBase (0..100)

TransportAddresses   ::= SET OF TransportAddress

TransportAddress      ::= SEQUENCE (
    transportConnectionEndpoint Address,
    networkAddress           Address
)

```

```

TransportEntityType ::= INTEGER ( other      (0),
                                oSI-TP      (1),
                                tCP          (2),
                                sNA          (3)
                                ) (0..255)

```

END

A.4.8 Inheritance & Naming Trees

This section provides graphic depictions for the inheritance and naming trees that are defined in the previous sections.

A.4.8.1 Inheritance Tree

```

top --- opEquipment ----- processingEntity
      |
      +--- computerSystem
      |
      +--- opNetwork
      |
      +--- coTransportProtocolLayerEntity
      |
      +--- clNetworkProtocolLayerEntity
      |
      +--- transportConnection

```

A.4.8.2 Naming Tree

```

root --- opNetwork --- opNetwork
      |               |
      |               +--- computerSystem
      |               +--- opEquipment
      |
      +--- DMI system --- coTransportProtocolLayerEntity -- transportConn
                        |
                        +--- clNetworkProtocolLayerEntity
                        |
                        +--- opEquipment --- opEquipment
                                      |
                                      +--- processingEntity
                                      |
                                      +--- coTransportProtocolLayerEntity -- transportConn
                                      |
                                      +--- clNetworkProtocolLayerEntity
                        |
                        +--- computerSystem --- computerSystem
                                      |
                                      +--- processingEntity
                                      |
                                      +--- opEquipment

```

```

+-- coTransportProtocolLayerEntity -- transportConn
|
+-- clNetworkProtocolLayerEntity

```

A.5 OIW NMSIG IVMO Definitions

The definitions specified in this clause can be referenced by using the label "OP1 Library Vol. 2" (e.g., "OP1 Library Vol. 2":transportConnectionIVMO).

A.5.1 Managed Object Classes and Mandatory Packages

A.5.1.1 Transport Connection IVMO

transportConnectionIVMO MANAGED OBJECT CLASS

DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2 : 1992":top;
 CHARACTERIZED BY transportConnectionIVMO-Package;

REGISTERED AS {y-objectClass 1};

transportConnectionIVMO-Package PACKAGE

BEHAVIOUR transportConnectionIVMO-behaviour;

ATTRIBUTES

transportConnectionIVMOId GET,

"OP1 Library Vol. 1":inactivityTimeout PERMITTED VALUES SYNTAX-1.Integer32 GET-REPLACE,

"OP1 Library Vol. 1":maxPDUSize PERMITTED VALUES SYNTAX-1.Integer32 GET-REPLACE;

NOTIFICATIONS

"Rec. X.721 | ISO/IEC 10165-2 : 1992":objectCreation,

"Rec. X.721 | ISO/IEC 10165-2 : 1992":objectDeletion,

"Rec. X.721 | ISO/IEC 10165-2 : 1992":attributeValueChange;;

transportConnectionIVMO-behaviour BEHAVIOUR

DEFINED AS

This managed object class is an IVMO (Initial Value Managed Object class). It represents the collection of characteristic attributes which supply default and initially advertised attribute values to be used by instances of the Transport Connection managed object class when they are created. There can be only one instance of the Transport Connection IVMO managed object class for each instance of the CO Transport Protocol Layer Entity managed object class. Each Transport Connection IVMO instance may provide initial attribute values for newly-created Transport Connection instances with the same superior.

The Attribute List parameter of the ObjectCreation notification shall contain all the attributes of the created transport connection IVMO instance.

The Attribute List parameter of the ObjectDeletion notification shall contain all the attributes of the deleted transport connection IVMO instance.

Attributes that are subject to the AttributeValueChange notification are : "OP1 Library Vol. 1":inactivityTimeout, "OP1 Library Vol. 1":maxPDUSize. All attributeValueChange notifications shall include the Attribute Identifier List parameter.1;

A.5.1.2 Transport Connection Retransmission IVMO

transportConnectionRetransmissionIVMO MANAGED OBJECT CLASS

DERIVED FROM transportConnectionIVMO;
CHARACTERIZED BY transportConnectionRetransmissionIVMO-Package;

REGISTERED AS {y-objectClass 3};

transportConnectionRetransmissionIVMO-Package PACKAGE

BEHAVIOUR transportConnectionIVMO-behaviour;

ATTRIBUTES

"OP1 Library Vol. 1":maxRetransmissions PERMITTED VALUES SYNTAX-1.Integer32 GET-REPLACE,
"OP1 Library Vol. 1":retransmissionTimerInitialValue
PERMITTED VALUES SYNTAX-1.Integer32 GET-REPLACE;;

transportConnectionRetransmissionIVMO-behaviour BEHAVIOUR

DEFINED AS

!This managed object class is an IVMO (Initial Value Managed Object class). It represents the collection of characteristic attributes which supply default and initially advertised attribute values to be used by instances of the Transport Connection managed object class that support retransmission, when they are created. There can be only one instance of the Transport Connection Retransmission IVMO managed object class for each instance of the CO Transport Protocol Layer Entity managed object class. Each Transport Connection Retransmission IVMO instance may provide initial attribute values for newly-created Transport Connection instances with the same superior.

Attributes, additional to those inherited from the transport connection IVMO managed object class, that are subject to the AttributeValueChange notification are : "OP1 Library Vol. 1":maxRetransmissions, "OP1 Library Vol. 1":retransmissionTimerInitialValue.!

A.5.2 Name Bindings

A.5.2.1 Transport Connection IVMO Name Bindings

transportConnectionIVMO-coTransportProtocolLayerEntity NAME BINDING

SUBORDINATE OBJECT CLASS transportConnectionIVMO AND SUBCLASSES;

NAMED BY

SUPERIOR OBJECT CLASS "OP1 Library Vol. 1":coTransportProtocolLayerEntity AND SUBCLASSES;

WITH ATTRIBUTE transportConnectionIVMOId;

REGISTERED AS {y-nameBinding 1};

A.5.2.2 Transport Connection Retransmission IVMO Name Bindings

transportConnectionRetransmissionIVMO-coTransportProtocolLayerEntity NAME BINDING

SUBORDINATE OBJECT CLASS transportConnectionRetransmissionIVMO
AND SUBCLASSES;

NAMED BY

SUPERIOR OBJECT CLASS "OP1 Library Vol. 1":coTransportProtocolLayerEntity AND SUBCLASSES;

WITH ATTRIBUTE transportConnectionIVMOId;

REGISTERED AS {y-nameBinding 2};

A.5.3 Attributes

A.5.3.1 Transport Connection IVMO Id

```

transportConnectionIVMOId ATTRIBUTE
  WITH ATTRIBUTE SYNTAX SYNTAX-1.GraphicStringBase;
  MATCHES FOR EQUALITY, SUBSTRINGS;
  BEHAVIOUR transportConnectionIVMOIdBehaviour;

REGISTERED AS      {y-attribute 1};

transportConnectionIVMOIdBehaviour BEHAVIOUR

DEFINED AS      !This attribute is the distinguishing attribute for the managed object class
                transportConnectionIVMO.!!;

```

A.5.4 Syntax Definitions

```

SYNTAX-2 ( y-module 1 )
DEFINITIONS IMPLICIT TAGS ::= BEGIN

--  EXPORTS everything

--  The following OIDs are allocated from the OIW NMSIG registration arc,
--  for use in registering OIW NMSIG MIL definitions.

nmsig      OBJECT IDENTIFIER ::= { iso identified-organization(3) oiw(14) nmsig(2) }
op1LibraryVol2 OBJECT IDENTIFIER ::= { nmsig 1 }
y-module   OBJECT IDENTIFIER ::= { op1LibraryVol2 0 }
y-objectClass OBJECT IDENTIFIER ::= { op1LibraryVol2 1 }
y-package  OBJECT IDENTIFIER ::= { op1LibraryVol2 2 }
y-nameBinding OBJECT IDENTIFIER ::= { op1LibraryVol2 3 }
y-attribute OBJECT IDENTIFIER ::= { op1LibraryVol2 4 }
y-attributeGroup OBJECT IDENTIFIER ::= { op1LibraryVol2 5 }
y-parameter OBJECT IDENTIFIER ::= { op1LibraryVol2 6 }
y-action    OBJECT IDENTIFIER ::= { op1LibraryVol2 7 }
y-notification OBJECT IDENTIFIER ::= { op1LibraryVol2 8 }

END

```

A.5.5 Inheritance & Naming Trees

This section provides graphic depictions for the inheritance and naming trees that are defined in the previous sections.

A.5.5.1 Inheritance Tree

```

top ---- transportConnectionIVMO ----- transportConnectionRetransmissionIVMO

```

A.5.5.2 Naming Tree

```

coTransportProtocolLayerEntity -+-- transportConnection

```

```
|
+-- transportConnectionIVMO
|
+-- transportConnectionRetransmissionIVMO
```

A.6 OIW NMSIG Shared Management Knowledge (SMK) Definitions

(Refer to the Working Implementation Agreements Document.)

Annex B (informative)

NMSIG Object Identifiers

(Refer to the Working Implementation Agreements Document for additional information.)

B.1 Introduction

This Annex (B) specifies object identifier component values which are globally unambiguous. These object identifiers are to be used when referencing NMSIG-specified information objects. As defined in Part 6 of these agreements, the OIW has assigned the following object identifier for use by the NMSIG:

{ iso(1) identified-organization(3) oiw(14) nmsig(2) }

The following object identifiers are assigned under the { iso identified-organization oiw nmsig } node, labelled "nmsig".

Table B.1 - Object identifiers assigned under "nmsig" node

Identifier	Value	Reference
op1LibraryVol2	1	A.5
op1LibraryVol1	2	A.4

By inclusion of the managed object (MO) definitions and the object identifiers in Annex A and Annex B, respectively, of the Stable Implementors' Agreements (SIAs), these managed object (MO) definitions have become formally registered. Implementors of part 18 of the SIAs do not have to support any of these MOs. However, even though Annex A and Annex B are informative annexes, any implementation that claims to conform to these definitions must treat these definitions as normative and comply with the relevant portions of Annex A.4 and A.5, and Annex B.

B.2 Harmonized MIL Object Identifiers

Harmonized MIL Object Identifiers are assigned under the "nmsig" node as follows:

```

nmsig      OBJECT IDENTIFIER ::= { iso identified-organization(3) oiw(14) nmsig(2) }
op1LibraryVol1 OBJECT IDENTIFIER ::= { nmsig 2 }
x-module   OBJECT IDENTIFIER ::= { op1LibraryVol1 0 }
x-objectClass OBJECT IDENTIFIER ::= { op1LibraryVol1 1 }
x-package  OBJECT IDENTIFIER ::= { op1LibraryVol1 2 }
x-nameBinding OBJECT IDENTIFIER ::= { op1LibraryVol1 3 }
x-attribute OBJECT IDENTIFIER ::= { op1LibraryVol1 4 }
x-attributeGroup OBJECT IDENTIFIER ::= { op1LibraryVol1 5 }
x-parameter OBJECT IDENTIFIER ::= { op1LibraryVol1 6 }
x-action   OBJECT IDENTIFIER ::= { op1LibraryVol1 7 }
x-notification OBJECT IDENTIFIER ::= { op1LibraryVol1 8 }
x-responseCode OBJECT IDENTIFIER ::= { op1LibraryVol1 9 }

```

B.2.1 Object Class Object Identifiers

The following object identifiers are assigned under the { x-objectClass } node:

Table B.2 - Object identifiers assigned under "x-objectClass" node

Reference	Identifier	Value
A.4.1.1	computerSystem	1
A.4.1.2	coTransportProtocolLayerEntity	2
A.4.1.3	clNetworkProtocolLayerEntity	3
A.4.1.4	opEquipment	4
A.4.1.5	opNetwork	5
A.4.1.6	processingEntity	6
A.4.1.7	transportConnection	7

B.2.2 Package Object Identifiers

The following object identifiers are assigned under the { x-package } node:

Table B.3 - Object identifiers assigned under "x-package" node

Reference	Identifier	Value
A.4.2.1	addressingPkg	3
A.4.2.2	checksumPDUsDiscardedPkg	2
A.4.2.3	contactListPkg	3
A.4.2.4	contactNamePkg	4
A.4.2.5	cpuUtilizationPkg	5
A.4.2.6	customerListPkg	6
A.4.2.7	customerNamePkg	7
A.4.2.8	functionListPkg	8
A.4.2.9	functionNamePkg	9
A.4.2.10	incomingProtocolErrorPkg	10
A.4.2.11	locationPointerPkg	11
A.4.2.12	manufacturerListPkg	12
A.4.2.13	manufacturerNamePkg	13
A.4.2.14	maxPDUSizeIVPkg	14
A.4.2.15	maxRetransmissionsPkg	15

Reference	Identifier	Value
A.4.2.16	memorySizePkg	16
A.4.2.17	memoryUtilizationPkg	17
A.4.2.18	octetsRetransmittedPkg	38
A.4.2.19	opNetworkListPkg	19
A.4.2.20	opNetworkNamePkg	20
A.4.2.20	opVersionPkg	21
A.4.2.20	outgoingProtocolErrorPkg	20
A.4.2.29	pdusRetransmittedCounterPkg	23
A.4.2.20	pdusRetransmittedThresholdPkg	28
A.4.2.20	peripheralListPkg	29
A.4.2.26	peripheralNamePkg	21
A.4.2.27	processingEntityListPkg	27
A.4.2.26	processingEntityNamePkg	28
A.4.2.29	productLabelPkg	29
A.4.2.10	retransmissionTimePkg	30
A.4.2.31	retransmissionTimerInitialValuePkg	31
A.4.2.38	serialNumberPkg	38
A.4.2.33	serviceListPkg	33
A.4.2.34	serviceNamePkg	38
A.4.2.35	softwareListPkg	35
A.4.2.36	softwareNamePkg	36
A.4.2.37	systemTimePkg	30
A.4.2.38	typeTextPkg	38
A.4.2.39	upTimePkg	39
A.4.2.40	usageStatePkg	40
A.4.2.41	vendorListPkg	41

B.2.3 Name Bindings Object Identifiers

The following object identifiers are assigned under the { x-nameBinding } node:

Table B.4 - Object identifiers assigned under "x-nameBinding" node

Reference	Identifier	Value
A.4.3.2	computerSystem-system	6
A.4.3.2	computerSystem-opNetwork	2
A.4.3.2	computerSystem-computerSystem	3
A.4.3.3	coTransportProtocolLayerEntity-computerSystem	6
A.4.3.3	coTransportProtocolLayerEntity-system	6
A.4.3.3	coTransportProtocolLayerEntity-opEquipment	6
A.4.3.4	clNetworkProtocolLayerEntity-computerSystem	7
A.4.3.4	clNetworkProtocolLayerEntity-system	3
A.4.3.4	clNetworkProtocolLayerEntity-opEquipment	6
A.4.3.5	opEquipment-computerSystem	10
A.4.3.5	opEquipment-system	11
A.4.3.5	opEquipment-equipment	12
A.4.3.5	opEquipment-opNetwork	13
A.4.3.4	network-opNetwork-1	10
A.4.3.6	network-opNetwork-2	15
A.4.3.6	opNetwork-root	16
A.4.3.8	transportConnection-coTransportProtocolLayerEntity	17

B.2.4 Attribute Object Identifiers

The following object identifiers are assigned under the { x-attribute } node:

Table B.5 - Object identifiers assigned under "x-attribute" node

Reference	Identifier	Value
A.4.4.1	activeConnections	1
A.4.4.2	addressingSize	2
A.4.4.3	checksumPDUsDiscardedCounter	3
A.4.4.4	computerSystemId	4
A.4.4.5	clNetworkProtocolLayerEntityId	5
A.4.4.6	coTransportProtocolLayerEntityId	6
A.4.4.7	contactList	7

Reference	Identifier	Value
A.4.4.8	contactName	8
A.4.4.9	cpuType	9
A.4.4.10	cpuUtilization	10
A.4.4.11	customerList	11
A.4.4.12	customerName	12
A.4.4.13	endianess	13
A.4.4.14	functionList	14
A.4.4.15	functionName	15
A.4.4.16	inactivityTime	16
A.4.4.17	inactivityTimeout	17
A.4.4.18	localNetworkAddress	18
A.4.4.19	localNetworkAddresses	19
A.4.4.20	localTransportAddresses	20
A.4.4.21	localTransportConnectionEndpoint	21
A.4.4.22	locationPointer	22
A.4.4.23	manufacturerList	23
A.4.4.24	manufacturerName	24
A.4.4.25	maxConnections	25
A.4.4.26	maxPDUSize	26
A.4.4.27	maxRetransmissions	27
A.4.4.28	memorySize	28
A.4.4.29	memoryUtilization	29
A.4.4.30	networkEntityType	30
A.4.4.31	networkTitle	31
A.4.4.32	npduTimeToLive	32
A.4.4.33	opEquipmentList	33
A.4.4.34	opNetworkList	34
A.4.4.35	opNetworkName	35
A.4.4.36	osInfo	36
A.4.4.37	pdusForwardedCounter	37
A.4.4.38	pdusReasmbldOkCounter	38
A.4.4.39	pdusReasmbldFailCounter	39

Reference	Identifier	Value
A.4.4.40	pdusDiscardedCounter	40
A.4.4.41	peripheralList	41
A.4.4.42	peripheralName	42
A.4.4.43	processingEntityList	43
A.4.4.44	processingEntityName	44
A.4.4.45	productLabel	45
A.4.4.46	remoteNetworkAddress	46
A.4.4.47	remoteTransportConnectionEndpoint	47
A.4.4.48	retransmissionTime	48
A.4.4.49	retransmissionTimerInitialValue	49
A.4.4.50	serialNumber	50
A.4.4.51	serviceList	51
A.4.4.52	serviceName	52
A.4.4.53	softwareList	53
A.4.4.54	softwareName	54
A.4.4.55	systemTime	55
A.4.4.56	transportConnectionId	56
A.4.4.57	transportConnectionReference	57
A.4.4.58	transportEntityType	58
A.4.4.59	typeText	59
A.4.4.60	upTime	60
A.4.4.61	vendorList	61

B.2.5 Action Object Identifiers

The following object identifiers are assigned under the { x-action } node:

Table B.6 - Object identifiers assigned under "x-action" node

Reference	Identifier	Value
A.4.5.1	activate	1
A.4.5.2	deactivate	2

B.2.6 Parameter Object Identifiers

The following object identifiers are assigned under the { x-parameter } node:

Table B.7 - Object identifiers assigned under "x-parameter" node

Reference	Identifier	Value
A.4.6.1	transportDisconnectCause	1

B.2.7 Response Code Object Identifiers

The following object identifiers are assigned under the { x-responseCode } node:

Table B.8 - Object identifiers assigned under "x-responseCode" node

Reference	Identifier	Value
A.4.7	failureResponse	1
A.4.7	successResponse	2

B.2.8 Module Object Identifiers

The following object identifiers are assigned under the { x-module } node:

Table B.9 - Object identifiers assigned under "x-module" node

Reference	Identifier	Value
A.4.7	SYNTAX-1	1

B.3 Phase 1 MIL Object Identifiers

Phase 1 MIL Object Identifiers are assigned under the "nmsig" node as follows:

```

op1LibraryVol2  OBJECT IDENTIFIER ::= { nmsig 1 }
y-module        OBJECT IDENTIFIER ::= { op1LibraryVol2 0 }
y-objectClass   OBJECT IDENTIFIER ::= { op1LibraryVol2 1 }
y-package       OBJECT IDENTIFIER ::= { op1LibraryVol2 2 }
y-nameBinding   OBJECT IDENTIFIER ::= { op1LibraryVol2 3 }
y-attribute     OBJECT IDENTIFIER ::= { op1LibraryVol2 4 }
y-attributeGroup OBJECT IDENTIFIER ::= { op1LibraryVol2 5 }
y-parameter     OBJECT IDENTIFIER ::= { op1LibraryVol2 6 }
y-action        OBJECT IDENTIFIER ::= { op1LibraryVol2 7 }
y-notification  OBJECT IDENTIFIER ::= { op1LibraryVol2 8 }

```

B.3.1 Object Class Object Identifiers

The following object identifiers are assigned under the { y-objectClass } node:

Table B.10 - Object identifiers assigned under "y-objectClass" node

Reference	Identifier	Value
A.5.1.1	transportConnectionIVMO	1
A.5.1.2	transportConnectionRetransmissionIVMO	3 [See note below]

Note: [Previous version (value 2) has been deprecated in favor of this version (value 3).]

B.3.2 Name Bindings Object Identifiers

The following object identifiers are assigned under the { y-nameBinding } node:

Table B.11 - Object identifiers assigned under "y-nameBinding" node

Reference	Identifier	Value
A.5.2.1	transportConnectionIVMO-coTransportProtocolLayerEntity	1
A.5.2.2	transportConnectionRetransmissionIVMO-coTransportProtocolLayerEntity	2

B.3.3 Attribute Object Identifiers

The following object identifiers are assigned under the { y-attribute } node:

Table B.12 - Object identifiers assigned under "y-attribute" node

Reference	Identifier	Value
A.5.3.1	transportConnectionIVMOId	1

B.3.4 Module Object Identifiers

The following object identifiers are assigned under the { y-module } node:

Table B.13 - Object identifiers assigned under "y-module" node

Reference	Identifier	Value
A.5.4	SYNTAX-2	1

Annex C (informative)

MOCS Proforma

C.1 Introduction

The purpose of this MOCS proforma is to provide a mechanism for a supplier of an implementation of these agreements which claims conformance to a managed object class to provide conformance information in a standard form.

C.2 Symbols, abbreviations, and terms

The MOCS proforma contained in this Annex is comprised of information in a tabular format in accordance with the guidelines presented in ISO/IEC 9646-2 [ATSS] and ISO/IEC 10165-6 [MICS].

The following common notations, defined in ISO/IEC 9646-2, are used for the status column.

c	conditional
m	mandatory
o	optional
x	prohibited
-	not applicable

The following common notations, defined in ISO/IEC 9646-2, are used for the support column.

Ig	the item is ignored (i.e., processed syntactically but not semantically)
N	not implemented
Y	implemented
-	not applicable

C.3 Instructions for completing the MOCS proforma to produce a MOCS

The supplier of the implementation shall enter an explicit statement in each of the boxes provided using the notation described in clause C.2. Additional instructions are provided in ISO/IEC 10165-6, Annex B.

C.4 Statements of Conformance to Managed Object Classes

This clause contains a MOCS Proforma for each managed object class defined in Annex A of these agreements, and registered by Annex B of these agreements.

C.4.1 Computer System MOCS Proforma

Managed Object class template label	Value of Object identifier for class
"OP1 Library Vol. 1":computerSystem	(1 3 14 2 2 1 1)

Are all mandatory features of the class supported?

Yes _____ No _____

Table C.4.1.1 - Name Binding Support

Index	Name Binding Template Label	Value of Object Identifier for Name Binding	Superior Object Class Template Label	Status	Support	Status		Support		Additional Information
						c r e a t e	d e l e t e	c r e a t e	d e l e t e	
C.4.1.1.1	computerSystem-system	{ 1 3 14 2 2 3 1 }	"Rec. X.721 ISO/IEC 10165-2 : 1992":system	o		m	m			
C.4.1.1.2	computerSystem-opNetwork	{ 1 3 14 2 2 3 2 }	opNetwork	o		m	m			
C.4.1.1.3	computerSystem-computerSystem	{ 1 3 14 2 2 3 3 }	computerSystem	o		m	m			

Table C.4.1.2 - Attribute Support

Index	Attribute Template Label	Value of Object Identifier for Attribute	Status					Support					Additional Information	
			set by create	get	replace	add	remove	set by create default	set by create	get	replace	add		remove
C.4.1.2.1	peripheralName	(1 3 14 2 2 4 42)	c3	c3	c3	x	x	x						
C.4.1.2.2	peripheralList	(1 3 14 2 2 4 41)	c4	c4	c4	c4	c4	x						
C.4.1.2.3	processingEntityName	(1 3 14 2 2 4 44)	c5	c5	c5	x	x	x						
C.4.1.2.4	processingEntityList	(1 3 14 2 2 4 43)	c6	c6	c6	c6	c6	x						
C.4.1.2.5	systemTime	(1 3 14 2 2 4 55)	x	c0	x	x	x	x						
C.4.1.2.6	upTime	(1 3 14 2 2 4 44)	x	c0	x	x	x	x						
C.4.1.2.7	"Rec. M.3100 : 1992":userLabel	(0 0 13 3100 0 7 50)	c0	c0	c0	x	x	x						
C.4.1.2.8	"Rec. X.721 ISO/IEC 10165-2 : 1992":usageState	(2 9 3 2 7 35)	x	c7	x	x	x	x						
C.4.1.2.9	"Rec. X.721 ISO/IEC 10165-2 : 1992":operationalState	(2 9 3 2 7 35)	x	m	x	x		x						
C.4.1.2.10	"Rec. X.721 ISO/IEC 10165-2 : 1992":administrativeState	(2 9 3 2 7 31)	m	m	m	x		x						
C.4.1.2.11	"Rec. X.721 ISO/IEC 10165-2 : 1992":alarmStatus	(2 9 3 2 7 32)	m	m	m	m	m	x						
C.4.1.2.12	"Rec. X.721 ISO/IEC 10165-2 : 1992":availabilityStatus	(2 9 3 2 7 33)	x	m	x	x	x	x						
C.4.1.2.13	computerSystemId	(1 3 14 2 2 4 4)	m	m	x	x	x	x						
C.4.1.2.14	"Rec. X.721 ISO/IEC 10165-2 : 1992":allomorphs	(2 9 3 2 7 50)	c1	c1	x	x	x	x						
C.4.1.2.15	"Rec. X.721 ISO/IEC 10165-2 : 1992":nameBinding	(2 9 3 2 7 63)	m	m	x	x	x	x						
C.4.1.2.16	"Rec. X.721 ISO/IEC 10165-2 : 1992":objectClass	(2 9 3 2 7 65)	m	m	x	x	x	x						
C.4.1.2.17	"Rec. X.721 ISO/IEC 10165-2 : 1992":packages	(2 9 3 2 7 66)	c2	c2	x	x	x	x						

c0 = m if an instance supports it, else -

c1 = m if an object supports allomorphism, else -

c2 = m if any any registered package (other than this package) has been instantiated, else -

c3 = m if an instance supports it and the peripheralListPkg is NOT present, else x
c4 = m if an instance supports it and the peripheralNamePkg is NOT present, else x
c5 = m if an instance supports it and the processingEntityListPkg is NOT present, else x
c6 = m if an instance supports it and the processingEntityNamePkg is NOT present, else x
c7 = m if a resource can detect usage, else -

Table C.4.1.3 - Attribute Group Support

Index	Attribute Group Template Label	Value of Object Identifier for Attribute Group	Status		Support		Additional Information
			g e t	s e t	g e t	s e t	
			t o d e f a u l t	t o d e f a u l t			
C.4.1.3.1	"Rec. X.721 ISO/IEC 10165-2 : 1992":state	(2 9 3 2 8 1)	m	x			

Table C.4.1.5 - Notification Support

Index	Notification Type Label	Value of Notification Type Identifier	Status	Support		Add Info	Sub-Index	Notification Field Name Label	Value of OID of Attribute associated with Field	Status	Support	Additional Information
				con	non							
C.4.1.5.1	"Rec. X.721 ISO/IEC 10165-2 : 1992": attributeValueChange	{ 2 9 3 2 10 1 }	m				C.4.1.5.1.1	additionalInformation	{ 2 9 3 2 70 6 }			
							C.4.1.5.1.2	additionalText	{ 2 9 3 2 70 7 }			
							C.4.1.5.1.3	attributeIdentifierList	{ 2 9 3 2 70 8 }			
							C.4.1.5.1.4	attributeValueChangeDefinition	{ 2 9 3 2 70 12 }			
							C.4.1.5.1.5	correlatedNotifications	{ 2 9 3 2 70 12 }			
							C.4.1.5.1.6	notificationIdentifier	{ 2 9 3 2 70 16 }			
							C.4.1.5.1.2	sourceIndicator	{ 2 9 3 2 70 26 }			
C.4.1.5.2	"Rec. X.721 ISO/IEC 10165-2 : 1992": objectCreation	{ 2 9 3 2 10 6 }	m				C.4.1.5.2.1	additionalInformation	{ 2 9 3 2 70 6 }			
							C.4.1.5.2.2	additionalText	{ 2 9 3 2 70 7 }			
							C.4.1.5.2.3	attributeList	{ 2 9 3 2 70 9 }			
							C.4.1.5.2.4	correlatedNotifications	{ 2 9 3 2 70 12 }			
							C.4.1.5.2.5	notificationIdentifier	{ 2 9 3 2 70 16 }			
							C.4.1.5.2.6	sourceIndicator	{ 2 9 3 2 70 26 }			
C.4.1.5.3	"Rec. X.721 ISO/IEC 10165-2 : 1992": objectDeletion	{ 2 9 3 2 10 7 }	m				C.4.1.5.3.1	additionalInformation	{ 2 9 3 2 70 6 }			
							C.4.1.5.3.2	additionalText	{ 2 9 3 2 70 7 }			
							C.4.1.5.3.3	attributeList	{ 2 9 3 2 70 9 }			
							C.4.1.5.3.4	correlatedNotifications	{ 2 9 3 2 70 12 }			
							C.4.1.5.3.5	notificationIdentifier	{ 2 9 3 2 70 16 }			

Index	Notification Type Label	Value of Notification Type Identifier	Status	Support		Add Info	Sub-Index	Notification Field Name Label	Value of OID of Attribute associated with Field	Status	Support	Additional Information
				Conf	Non							
							C.4.1.5.3.6	sourceIndicator	(2 9 3 2 7 0 26)			
C.4.1.5.4	"Rec. X.721 ISO/IEC 10165-2 : 1992": stateChange	(2 9 3 2 10 14)	m				C.4.1.5.4.1	additionalInformation	(2 9 3 2 7 0 6)			
							C.4.1.5.4.2	additionalText	(2 9 3 2 7 0 7)			
							C.4.1.5.4.3	attributeIdentifierList	(2 9 3 2 7 0 8)			
							C.4.1.5.4.4	correlatedNotifications	(2 9 3 2 7 0 12)			
							C.4.1.5.4.5	notificationIdentifier	(2 9 3 2 7 0 16)			
							C.4.1.5.4.6	sourceIndicator	(2 9 3 2 7 0 26)			
							C.4.1.5.4.7	stateChangeDefinition	(2 9 3 2 7 0 28)	m		
C.4.1.5.5	"Rec. X.721 ISO/IEC 10165-2 : 1992": processingError Alarm	(2 9 3 2 10 10)	m				C.4.1.5.5.1	additionalInformation	(2 9 3 2 7 0 6)			
							C.4.1.5.5.2	additionalText	(2 9 3 2 7 0 7)			
							C.4.1.5.5.3	backUpObject	(2 9 3 2 7 0 41)			
							C.4.1.5.5.4	backedUpStatus	(2 9 3 2 7 0 11)			
							C.4.1.5.5.5	correlatedNotifications	(2 9 3 2 7 0 12)			
							C.4.1.5.5.6	notificationIdentifier	(2 9 3 2 7 0 16)			
							C.4.1.5.5.7	perceivedSeverity	(2 9 3 2 7 0 17)	m		
							C.4.1.5.5.8	probableCause	(2 9 3 2 7 0 18)	m		
							C.4.1.5.5.9	proposedRepairActions	(2 9 3 2 7 0 19)			
							C.4.1.5.5.10	specificProblems	(2 9 3 2 7 0 27)			
							C.4.1.5.5.11	stateChangeDefinition	(2 9 3 2 7 0 28)			

Index	Notification Type Label	Value of Notification Type Identifier	Status	Support		Add Info	Sub-Index	Notification Field Name Label	Value of OID of Attribute associated with Field	Status	Support	Additional Information
				conf	non							
							C.4.1.5.5.12	thresholdInfo	(2 9 3 2 7 29)			
							C.4.1.5.5.13	trendIndication	(2 9 3 2 7 30)			
C.4.1.5.6	"Rec. X.721 ISO/IEC 10165-2 : 1992": environmentalAlarm	(2 9 3 2 10 3)	m				C.4.1.5.6.1	additionalInformation	(2 9 3 2 7 6)			
							C.4.1.5.6.2	additionalText	(2 9 3 2 7 7)			
							C.4.1.5.6.3	backUpObject	(2 9 3 2 7 41)			
							C.4.1.5.6.4	backedUpStatus	(2 9 3 2 7 11)			
							C.4.1.5.6.5	correlatedNotifications	(2 9 3 2 7 12)			
							C.4.1.5.6.6	notificationIdentifier	(2 9 3 2 7 16)			
							C.4.1.5.6.7	perceivedSeverity	(2 9 3 2 7 17)	m		
							C.4.1.5.6.8	probableCause	(2 9 3 2 7 18)	m		
							C.4.1.5.6.9	proposedRepairActions	(2 9 3 2 7 19)			
							C.4.1.5.6.10	specificProblems	(2 9 3 2 7 27)			
							C.4.1.5.6.11	stateChangeDefinition	(2 9 3 2 7 28)			
							C.4.1.5.6.12	thresholdInfo	(2 9 3 2 7 29)			
							C.4.1.5.6.13	trendIndication	(2 9 3 2 7 30)			
C.4.1.5.7	"Rec. X.721 ISO/IEC 10165-2 : 1992": equipmentAlarm	(2 9 3 2 10 4)	m				C.4.1.5.7.1	additionalInformation	(2 9 3 2 7 6)			
							C.4.1.5.7.2	additionalText	(2 9 3 2 7 7)			
							C.4.1.5.7.3	backUpObject	(2 9 3 2 7 41)			
							C.4.1.5.7.4	backedUpStatus	(2 9 3 2 7 11)			

Index	Notification Type Label	Value of Notification Type Identifier	Status	Support		Add Info	Sub-Index	Notification Field Name Label	Value of OID of Attribute associated with Field	Status	Support	Additional Information
				c o n f	n o n							
							C.4.1.5	correlatedNotifications	{ 2 9 3 2 7 12 }	o		
							C.4.1.6	notificationIdentifier	{ 2 9 3 2 7 16 }	o		
							C.4.1.7	perceivedSeverity	{ 2 9 3 2 7 17 }	m		
							C.4.1.8	probableCause	{ 2 9 3 2 7 18 }	m		
							C.4.1.9	proposedRepairActions	{ 2 9 3 2 7 19 }	o		
							C.4.1.10	specificProblems	{ 2 9 3 2 7 27 }	o		
							C.4.1.11	stateChangeDefinition	{ 2 9 3 2 7 28 }	o		
							C.4.1.12	thresholdInfo	{ 2 9 3 2 7 29 }	o		
							C.4.1.13	trendIndication	{ 2 9 3 2 7 30 }	o		

C.4.2 Connection Oriented Transport Protocol Layer Entity MOCS Proforma

Managed Object class template label	Value of Object identifier for class
"OP1 Library Vol. 1":coTransportProtocolLayerEntity	{ 1 3 14 2 2 1 2 }

Are all mandatory features of the class supported? Yes _____ No _____

Table C.4.2.1 - Name Binding Support

Index	Name Binding Template Label	Value of Object Identifier for Name Binding	Superior Object Class Template Label	Status	Support	Status		Support		Additional Information
						c r e a t e	d e l e t e	c r e a t e	d e l e t e	
C.4.2.1.1	coTransportProtocolLayerEntity-computerSystem	{ 1 3 14 2 2 3 4 }	computerSystem	o		x	x			
C.4.2.1.2	coTransportProtocolLayerEntity-system	{ 1 3 14 2 2 3 5 }	"Rec. X.721 ISO/IEC 10165-2 : 1992":system	o		x	x			
C.4.2.1.3	coTransportProtocolLayerEntity-opEquipment	{ 1 3 14 2 2 3 6 }	opEquipment	o		x	x			

Table C.4.2.2 - Attribute Support

Index	Attribute Template Label	Value of Object Identifier for Attribute	Status					Support					Additional Information	
			set by create	get	replace	add	remove	set by create default	get	replace	add	remove		set by default
C.4.2.2.1	manufacturerList	{ 1 3 14 2 2 4 23 }	c3	c3	c3	c3	c3	x						
C.4.2.2.2	manufacturerName	{ 1 3 14 2 2 4 24 }	c4	c4	c4	x	x	x						
C.4.2.2.3	productLabel	{ 1 3 14 2 2 4 45 }	c0	c0	c0	x	x	x						
C.4.2.2.4	"Rec. M.3100 : 1992":version	{ 0 0 13 3100 0 7 52 }	c0	c0	c0	x	x	x						
C.4.2.2.5	serialNumber	{ 1 3 14 2 2 4 50 }	c0	c0	c0	x	x	x						
C.4.2.2.6	typeText	{ 1 3 14 2 2 4 59 }	c0	c0	c0	x	x	x						
C.4.2.2.7	upTime	{ 1 3 14 2 2 4 60 }	x	c0	x	x	x	x						
C.4.2.2.8	"Rec. X.721 ISO/IEC 10165-2 : 1992":incomingProtocolErrorCounter	{ 2 9 3 2 7 77 }	x	c0	x	x	x	x						
C.4.2.2.9	"Rec. X.721 ISO/IEC 10165-2 : 1992":outgoingProtocolErrorCounter	{ 2 9 3 2 7 85 }	x	c0	x	x	x	x						
C.4.2.2.10	checksumPDUsDiscardedCounter	{ 1 3 14 2 2 4 3 }	x	c0	x	x	x	x						
C.4.2.2.11	maxPDUSize	{ 1 3 14 2 2 4 26 }	c5	c5	c5	x	x	x						
C.4.2.2.12	"Rec. X.721 ISO/IEC 10165-2 : 1992":usageState	{ 2 9 3 2 7 39 }	x	c6	x	x	x	x						
C.4.2.2.13	transportEntityType	{ 1 3 14 2 2 4 58 }	x	m	x	x	x	x						
C.4.2.2.14	localTransportAddresses	{ 1 3 14 2 2 4 20 }	x	m	x	x	x	x						
C.4.2.2.15	activeConnections	{ 1 3 14 2 2 4 1 }	x	m	x	x	x	x						
C.4.2.2.16	maxConnections	{ 1 3 14 2 2 4 25 }	x	m	x	x	x	x						
C.4.2.2.17	"Rec. X.721 ISO/IEC 10165-2 : 1992":outgoingConnectionRequestsCounter	{ 2 9 3 2 7 82 }	x	m	x	x	x	x						
C.4.2.2.18	"Rec. X.721 ISO/IEC 10165-2 : 1992":incomingConnectionRequestsCounter	{ 2 9 3 2 7 74 }	x	m	x	x	x	x						

Index	Attribute Template Label	Value of Object Identifier for Attribute	Status						Support						Additional Information
			set by create	get	replace	add	remove	set to default	set by create	get	replace	add	remove	set to default	
C.4.2.2.19	"Rec. X.721 ISO/IEC 10165-2 : 1992": outgoingConnectionRejectErrorCounter	(2 9 3 2 7 81)	x	m		x	x	x							
C.4.2.2.22	"Rec. X.721 ISO/IEC 10165-2 : 1992": incomingConnectionRejectErrorCounter	(2 9 3 2 7 73)	x	m	x	x	x	x							
C.4.2.2.21	"Rec. X.721 ISO/IEC 10165-2 : 1992": outgoingDisconnectErrorCounter	(2 9 3 2 7 76)	x	m		x	x	x							
C.4.2.2.22	"Rec. X.721 ISO/IEC 10165-2 : 1992": incomingDisconnectErrorCounter	(2 9 3 2 7 76)	x	m	x	x	x	x							
C.4.2.2.23	"Rec. X.721 ISO/IEC 10165-2 : 1992": outgoingDisconnectCounter	(2 9 3 2 7 83)	x	m	x	x	x	x							
C.4.2.2.26	"Rec. X.721 ISO/IEC 10165-2 : 1992": incomingDisconnectCounter	(2 9 3 2 7 78)	x	m		x	x	x							
C.4.2.2.25	"Rec. X.721 ISO/IEC 10165-2 : 1992": octetsSentCounter	(2 9 3 2 7 35)	x	m	x	x	x	x							
C.4.2.2.26	"Rec. X.721 ISO/IEC 10165-2 : 1992": octetsReceivedCounter	(2 9 3 2 7 78)	x	m		x	x	x							
C.4.2.2.27	"Rec. X.721 ISO/IEC 10165-2 : 1992": operationalState	(2 9 3 2 7 35)	x	m		x	x	x							
C.4.2.2.23	"Rec. X.721 ISO/IEC 10165-2 : 1992": administrativeState	(2 9 3 2 7 31)	m	m	m	x	x	x							
C.4.2.2.23	"Rec. X.721 ISO/IEC 10165-2 : 1992": alarmStatus	(2 9 3 2 7 32)	m	m	m	m	m	x							
C.4.2.2.30	coTransportProtocolLayerId	(1 3 14 2 2 4 6)	m	m	x	x	x	x							
C.4.2.2.31	"Rec. X.721 ISO/IEC 10165-2 : 1992": allomorphs	(2 9 3 2 7 50)	c1	c1		x	x	x							
C.4.2.2.32	"Rec. X.721 ISO/IEC 10165-2 : 1992": nameBinding	(2 9 3 2 7 50)	m	m		x	x	x	x						
C.4.2.2.33	"Rec. X.721 ISO/IEC 10165-2 : 1992": objectClass	(2 9 3 2 7 65)	m	m	x	x	x	x							
C.4.2.2.34	"Rec. X.721 ISO/IEC 10165-2 : 1992": packages	(2 9 3 2 7 66)	c2	c2	x	x	x	x							

c0 = m if an instance supports it, else -
c1 = m if an object supports allormorphism, else -
c2 = m if any any registered package (other than this package) has been instantiated, else -
c3 = m if an instance supports it and the manufacturerNamePkg is NOT present, else x
c4 = m if an instance supports it and the manufacturerListPkg is NOT present, else x
c5 = m if the "OP1 Library Vol. 2 : 1992":transportConnectionIVMO object class is not used to provide this initial value, else x
c6 = m if resource can detect usage, else -

Table C.4.2.3 - Attribute Group Support

Index	Attribute Group Template Label	Value of Object Identifier for Attribute Group	Status		Support		Additional Information
			g e t	s e t	g e t	s e t	
			t o d e f	a u l t	t o d e f	a u l t	
C.4.2.3.1	"Rec. X.721 ISO/IEC 10165-2 : 1992":state	(2 9 3 2 8 1)	m	x			

Table C.4.2.5 - Notification Support

Index	Notification Type Label	Value of Notification Type Identifier	Status	Support		Add Info	Sub-Index	Notification Field Name Label	Value of OID of Attribute associated with Field	Status	Support	Additional Information
				con	firm							
C.4.2.5.1	"Rec. X.721 ISO/IEC 10165-2 : 1992": attributeValueChange	(2 9 3 2 10 1)	m				C.4.2	additionalInformation	(2 9 3 2 706)			
							.5.1.1					
							C.4.2	additionalText	(2 9 3 2 707)			
							.5.1.2					
							C.4.2	attributeIdentifierList	(2 9 3 2 708)			
							.5.1.3					
							C.4.2	attributeValueChangeDefinition	(2 9 3 2 710)	m		
C.4.2.5.2	"Rec. X.721 ISO/IEC 10165-2 : 1992": objectCreation	(2 9 3 2 10 6)	m				.5.1.4					
							C.4.2	correlatedNotifications	(2 9 3 2 712)			
							.5.1.5					
							C.4.2	notificationIdentifier	(2 9 3 2 716)			
							.5.1.6					
							C.4.2	sourceIndicator	(2 9 3 2 726)			
							.5.1.7					
C.4.2.5.3	"Rec. X.721 ISO/IEC 10165-2 : 1992": objectDeletion	(2 9 3 2 10 7)	m				C.4.2	additionalInformation	(2 9 3 2 706)			
							.5.2.1					
							C.4.2	additionalText	(2 9 3 2 707)			
							.5.2.2					
							C.4.2	attributeList	(2 9 3 2 709)			
							.5.2.3					
							C.4.2	correlatedNotifications	(2 9 3 2 712)			
C.4.2.5.4							.5.2.4					
							C.4.2	notificationIdentifier	(2 9 3 2 716)			
							.5.2.5					
							C.4.2	sourceIndicator	(2 9 3 2 726)			
							.5.2.6					
C.4.2.5.5							C.4.2	additionalInformation	(2 9 3 2 706)			
							.5.3.1					
							C.4.2	additionalText	(2 9 3 2 707)			
							.5.3.2					
							C.4.2	attributeList	(2 9 3 2 709)			
C.4.2.5.6							.5.3.3					
							C.4.2	correlatedNotifications	(2 9 3 2 712)			
							.5.3.4					
C.4.2.5.7							C.4.2	notificationIdentifier	(2 9 3 2 716)			
							.5.3.5					

Index	Notification Type Label	Value of Notification Type Identifier	Status	Support		Add Info	Sub-Index	Notification Field Name Label	Value of OID of Attribute associated with Field	Status	Support	Additional Information
				conf	on							
							C.4.2 .5.3. 6	sourceIndicator	(2 9 3 2 7 0 26)			
C.4.2.5.4	"Rec. X.721 ISO/IEC 10165-2 : 1992": stateChange	(2 9 3 2 10 14)	m				C.4.2 .5.4. 1	additionalInform ation	(2 9 3 2 7 0 6)			
							C.4.2 .5.4. 2	additionalText	(2 9 3 2 7 0 7)			
							C.4.2 .5.4. 3	attributeIdentif ierList	(2 9 3 2 7 0 8)			
							C.4.2 .5.4. 4	correlatedNotifi cations	(2 9 3 2 7 0 12)			
							C.4.2 .5.4. 5	notificationIden tifier	(2 9 3 2 7 0 16)			
							C.4.2 .5.4. 6	sourceIndicator	(2 9 3 2 7 0 26)			
							C.4.2 .5.4. 7	stateChangeDefin ition	(2 9 3 2 7 m 28)			
C.4.2.5.5	"Rec. X.721 ISO/IEC 10165-2 : 1992": processingError Alarm	(2 9 3 2 10 10)	m				C.4.2 .5.5. 1	additionalInform ation	(2 9 3 2 7 0 6)			
							C.4.2 .5.5. 2	additionalText	(2 9 3 2 7 0 7)			
							C.4.2 .5.5. 3	backUpObject	(2 9 3 2 7 0 41)			
							C.4.2 .5.5. 4	backedUpStatus	(2 9 3 2 7 0 11)			
							C.4.2 .5.5. 5	correlatedNotifi cations	(2 9 3 2 7 0 12)			
							C.4.2 .5.5. 6	notificationIden tifier	(2 9 3 2 7 0 16)			
							C.4.2 .5.5. 7	perceivedSeverit y	(2 9 3 2 7 m 17)			
							C.4.2 .5.5. 8	probableCause	(2 9 3 2 7 m 19)			
							C.4.2 .5.5. 9	proposedRepairAc tions	(2 9 3 2 7 0 19)			
							C.4.2 .5.5. 10	specificProblems	(2 9 3 2 7 0 27)			
							C.4.2 .5.5. 11	stateChangeDefin ition	(2 9 3 2 7 0 28)			

Index	Notification Type Label	Value of Notification Type Identifier	Status	Support	Add Info	Sub-Index	Notification Field Name Label	Value of OID of Attribute associated with Field	Status	Support	Additional Information
				c o n f							
						C.4.2.5.5.12	thresholdInfo	{ 2 9 3 2 7 29 }			
						C.4.2.5.5.13	trendIndication	{ 2 9 3 2 7 30 }			

C.4.3 ConnectionlessNetwork Protocol Layer Entity MOCS Proforma

Managed Object class template label	Value of Object identifier for class
"OP1 Library Vol. 1":clNetworkProtocolLayerEntity	{ 1 3 14 2 2 1 3 }

Are all mandatory features of the class supported? Yes _____ No _____

Table C.4.3.1 - Name Binding Support

Index	Name Binding Template Label	Value of Object Identifier for Name Binding	Superior Object Class Template Label	Status	Support	Status	Support	Additional Information
						c r e a t e	d e l e t e	
C.4.3.1.1	clNetworkProtocolLayerEntity-computerSystem	{ 1 3 14 2 2 3 7 }	computerSystem	o		x	x	
C.4.3.1.2	clNetworkProtocolLayerEntity-system	{ 1 3 14 2 2 3 8 }	"Rec. X.721 ISO/IEC 10165-2 : 1992":system	o		x	x	
C.4.3.1.3	clNetworkProtocolLayerEntity-opEquipment	{ 1 3 14 2 2 3 9 }	opEquipment	o		x	x	

Table C.4.3.2 - Attribute Support

Index	Attribute Template Label	Value of Object Identifier for Attribute	Status					Support					Additional Information
			set by create	get by replace	read by move	read by delete	set by create	get by replace	read by move	read by delete			
C.4.3.2.1	manufacturerList	{ 1 3 14 2 2 4 23 }	c3	c3	c3	c3	c3	x					
C.4.3.2.2	manufacturerName	{ 1 3 14 2 2 4 24 }	c4	c4	c4	x	x	x					
C.4.3.2.3	productLabel	{ 1 3 14 2 2 4 45 }	c0	c0	c0	x	x	x					
C.4.3.2.4	"Rec. M.3100 : 1992":version	{ 0 0 13 3100 0 7 52 }	c0	c0	c0	x	x	x					
C.4.3.2.5	serialNumber	{ 1 3 14 2 2 4 50 }	c0	c0	c0	x	x	x					
C.4.3.2.6	typeText	{ 1 3 14 2 2 4 59 }	c0	c0	c0	x	x	x					
C.4.3.2.7	upTime	{ 1 3 14 2 2 4 45 }	x	c0	x	x	x	x					
C.4.3.2.8	networkEntityType	{ 1 3 14 2 2 4 30 }	x	m	x	x	x	x					
C.4.3.2.9	localNetworkAddresses	{ 1 3 14 2 2 4 19 }	x	m	m	m	m	x					
C.4.3.2.10	npduTimeToLive	{ 1 3 14 2 2 4 32 }	m	m	m	x	x	x					
C.4.3.2.11	"Rec. X.721 ISO/IEC 10165-2 : 1992":pdusSentCounter	{ 2 9 3 2 7 88 }	x	m	x	x	x	x					
C.4.3.2.12	"Rec. X.721 ISO/IEC 10165-2 : 1992":pdusReceivedCounter	{ 2 9 3 2 7 86 }	x	m	x	x	x	x					
C.4.3.2.13	"Rec. X.721 ISO/IEC 10165-2 : 1992":octetsSentCounter	{ 2 9 3 2 7 80 }	x	m		x	x	x					
C.4.3.2.14	"Rec. X.721 ISO/IEC 10165-2 : 1992":octetsReceivedCounter	{ 2 9 3 2 7 78 }	x	m		x	x	x					
C.4.3.2.15	pdusForwardedCounter	{ 1 3 14 2 2 4 37 }	x	m	x	x	x	x					
C.4.3.2.16	pdusReasmblDOKCounter	{ 1 3 14 2 2 4 38 }	x	m	x	x	x	x					
C.4.3.2.17	pdusReasmblFailCounter	{ 1 3 14 2 2 4 39 }	x	m	x	x	x	x					
C.4.3.2.18	pdusDiscardedCounter	{ 1 3 14 2 2 4 40 }	x	m	x	x	x	x					
C.4.3.2.19	"Rec. X.721 ISO/IEC 10165-2 : 1992":operationalState	{ 2 9 3 2 7 35 }	x	m		x	x	x					
C.4.3.2.20	"Rec. X.721 ISO/IEC 10165-2 : 1992":administrativeState	{ 2 9 3 2 7 31 }	m	m	m	x	x	x					

Index	Attribute Template Label	Value of Object Identifier for Attribute	Status					Support					Additional Information		
			s e t b y c r e a t e	g e t	r e p l a c e	a d d	r e m o v e	s e t t o d e f a u l t	s e t b y c r e a t e	g e t	r e p l a c e	a d d		r e m o v e	s e t t o d e f a u l t
C.4.3.2.21	"Rec. X.721 ISO/IEC 10165-2 : 1992": alarmStatus	(2 9 3 2 7 32)	m	m	m	m	m	x							
C.4.3.2.22	clNetworkProtocolLayerId	(1 3 14 2 2 4 5)	m	m	x	x	x	x							
C.4.3.2.23	"Rec. X.721 ISO/IEC 10165-2 : 1992": allomorphs	(2 9 3 2 7 50)	c1	c1	x	x	x	x							
C.4.3.2.24	"Rec. X.721 ISO/IEC 10165-2 : 1992": nameBinding	(2 9 3 2 7 63)	m	m		x	x	x							
C.4.3.2.25	"Rec. X.721 ISO/IEC 10165-2 : 1992": objectClass	(2 9 3 2 7 65)	m	m	x	x	x	x							
C.4.3.2.26	"Rec. X.721 ISO/IEC 10165-2 : 1992": packages	(2 9 3 2 7 66)	c2	c2	x	x	x	x							

c0 = m if an instance supports it, else -

c1 = m if an object supports allomorphism, else -

c2 = m if any any registered package (other than this package) has been instantiated, else -

c3 = m if an instance supports it and the manufacturerNamePkg is NOT present, else x

c4 = m if an instance supports it and the manufacturerListPkg is NOT present, else x

Table C.4.3.3 - Attribute Group Support

Index	Attribute Group Template Label	Value of Object Identifier for Attribute Group	Status		Support		Additional Information
			get	set	get	set	
				to default		to default	
C.4.3.3.1	"Rec. X.721 ISO/IEC 10165-2 : 1992":state	(2 9 3 2 8 1)	m	x			

Table C.4.3.5 - Notification Support

Index	Notification Type Label	Value of Notification Type Identifier	Status	Support		Add Info	Sub-Index	Notification Field Name Label	Value of OID of Attribute associated with Field	Status	Support	Additional Information
				Conf	Non							
C.4.3.5.1	"Rec. X.721 ISO/IEC 10165-2 : 1992": attributeValueChange	{ 2 9 3 2 10 1 }	m				C.4.3.5.1.1	additionalInformation	{ 2 9 3 2 70 6 }			
							C.4.3.5.1.2	additionalText	{ 2 9 3 2 70 7 }			
							C.4.3.5.1.3	attributeIdentifierList	{ 2 9 3 2 70 8 }			
							C.4.3.5.1.4	attributeValueChangeDefinition	{ 2 9 3 2 70 10 }	m		
							C.4.3.5.1.5	correlatedNotifications	{ 2 9 3 2 70 12 }			
							C.4.3.5.1.6	notificationIdentifier	{ 2 9 3 2 70 16 }			
							C.4.3.5.1.7	sourceIndicator	{ 2 9 3 2 70 26 }			
C.4.3.5.2	"Rec. X.721 ISO/IEC 10165-2 : 1992": objectCreation	{ 2 9 3 2 10 6 }	m				C.4.3.5.2.3	additionalInformation	{ 2 9 3 2 70 6 }			
							C.4.3.5.2.2	additionalText	{ 2 9 3 2 70 7 }			
							C.4.3.5.2.3	attributeList	{ 2 9 3 2 70 9 }			
							C.4.3.5.2.4	correlatedNotifications	{ 2 9 3 2 70 12 }			
							C.4.3.5.2.5	notificationIdentifier	{ 2 9 3 2 70 16 }			
C.4.3.5.3	"Rec. X.721 ISO/IEC 10165-2 : 1992": objectDeletion	{ 2 9 3 2 10 7 }	m				C.4.3.5.3.1	additionalInformation	{ 2 9 3 2 70 6 }			
							C.4.3.5.3.2	additionalText	{ 2 9 3 2 70 7 }			
							C.4.3.5.3.3	attributeList	{ 2 9 3 2 70 9 }			
							C.4.3.5.3.4	correlatedNotifications	{ 2 9 3 2 70 12 }			

Index	Notification Type Label	Value of Notification Type Identifier	Status	Support		Add Info	Sub-Index	Notification Field Name Label	Value of OID of Attribute associated with Field	Status	Support	Additional Information
				con	firm							
							C.4.3.5	notificationIdentifier	{ 2 9 3 2 7 0 16 }			
							C.4.3.6	sourceIndicator	{ 2 9 3 2 7 0 26 }			
C.4.3.5.4	"Rec. X.721 ISO/IEC 10165-2 : 1992": stateChange	{ 2 9 3 2 10 14 }	m				C.4.3.1	additionalInformation	{ 2 9 3 2 7 0 6 }			
							C.4.3.2	additionalText	{ 2 9 3 2 7 0 7 }			
							C.4.3.3	attributeIdentifierList	{ 2 9 3 2 7 0 8 }			
							C.4.3.4	correlatedNotifications	{ 2 9 3 2 7 0 12 }			
							C.4.3.5	notificationIdentifier	{ 2 9 3 2 7 0 16 }			
							C.4.3.6	sourceIndicator	{ 2 9 3 2 7 0 26 }			
							C.4.3.7	stateChangeDefinition	{ 2 9 3 2 7 0 28 }			
C.4.3.5.5	"Rec. X.721 ISO/IEC 10165-2 : 1992": processingError Alarm	{ 2 9 3 2 10 10 }	m				C.4.3.1	additionalInformation	{ 2 9 3 2 7 0 6 }			
							C.4.3.2	additionalText	{ 2 9 3 2 7 0 7 }			
							C.4.3.3	backUpObject	{ 2 9 3 2 7 0 41 }			
							C.4.3.4	backedUpStatus	{ 2 9 3 2 7 0 11 }			
							C.4.3.5	correlatedNotifications	{ 2 9 3 2 7 0 12 }			
							C.4.3.6	notificationIdentifier	{ 2 9 3 2 7 0 16 }			
							C.4.3.7	perceivedSeverity	{ 2 9 3 2 7 0 17 }	m		
							C.4.3.8	probableCause	{ 2 9 3 2 7 0 18 }	m		
							C.4.3.9	proposedRepairActions	{ 2 9 3 2 7 0 19 }			
							C.4.3.10	specificProblems	{ 2 9 3 2 7 0 27 }			

Index	Notification Type Label	Value of Notification Type Identifier	Status	Support	Add Info	Sub-Index	Notification Field Name Label	Value of OID of Attribute associated with Field	Status	Support	Additional Information
							C.4.3.5.5.11 stateChangeDefinition	(2 9 3 2 7 28)			
							C.4.3.5.5.12 thresholdInfo	(2 9 3 2 7 29)			
							C.4.3.5.5.13 trendIndication	(2 9 3 2 7 30)			
C.4.3.5.6	"Rec. X.721 ISO/IEC 10165-2 : 1992": communicationsAlarm	(2 9 3 2 10 2)	m				C.4.3.5.6.1 additionalInformation	(2 9 3 2 7 6)			
							C.4.3.5.6.2 additionalText	(2 9 3 2 7 7)			
							C.4.3.5.6.6 backupObject	(2 9 3 2 7 41)			
							C.4.3.5.6.6 backedUpStatus	(2 9 3 2 7 11)			
							C.4.3.5.6.5 correlatedNotifications	(2 9 3 2 7 12)			
							C.4.3.5.6.6 notificationIdentifier	(2 9 3 2 7 16)			
							C.4.3.5.6.1 perceivedSeverity	(2 9 3 2 7 17)	m		
							C.4.3.5.6.6 probableCause	(2 9 3 2 7 18)	m		
							C.4.3.5.6.9 proposedRepairActions	(2 9 3 2 7 16)			
							C.4.3.5.6.10 specificProblems	(2 9 3 2 7 28)			
							C.4.3.5.6.11 stateChangeDefinition	(2 9 3 2 7 28)			
							C.4.3.5.6.12 thresholdInfo	(2 9 3 2 7 29)			
							C.4.3.5.6.13 trendIndication	(2 9 3 2 7 30)			

C.4.4 OMNIPoint Equipment MOCS Proforma

Managed Object class template label	Value of Object identifier for class
"OP1 Library Vol. 1":opEquipment	(1 3 14 2 2 1 4)

Are all mandatory features of the class supported?

Yes _____

No _____

Table C.4.4.1 - Name Binding Support

Index	Name Binding Template Label	Value of Object Identifier for Name Binding	Superior Object Class Template Label	Status	Support	Status		Support		Additional Information
						c	d	c	d	
C.4.4.1.1	"Rec. M.3100 : 1992": equipment-equipment	(0 0 13 3100 0 6 10)	"Rec. M.3100 : 1992":equipment	o		m	m			
C.4.4.1.2	"Rec. M.3100 : 1992": equipment-ManagedElement	(0 0 13 3100 0 6 9)	"Rec. M.3100 : 1992": managedElement	o		m	m			
C.4.4.1.3	opEquipment- computerSystem	(1 3 14 2 2 3 10)	computerSystem	o		m	m			
C.4.4.1.4	opEquipment-system	(1 3 14 2 2 3 11)	"Rec. X.721 : ISO/IEC 10165-2 : 1992":system	o		m	m			
C.4.4.1.5	opEquipment-equipment	(1 3 14 2 2 3 12)	"Rec. M.3100 : 1992":equipment	o		m	m			
C.4.4.1.6	opEquipment-opNetwork	(1 3 14 2 2 3 13)	opNetwork	o		m	m			

Table C.4.4.2 - Attribute Support

Index	Attribute Template Label	Value of Object Identifier for Attribute	Status					Support					Additional Information
			set by create	get	replace	add	remove	set by create	get	replace	add	remove	
C.4.4.2.1	"Rec. M.3100 : 1992":affectedObjectList	{ 0 0 13 3100 0 7 2 }	x	c0	x	x	x	x					
C.4.4.2.2	"Rec. M.3100 : 1992":currentProblemList	{ 0 0 13 3100 0 7 17 }	x	c0	x	x	x	x					
C.4.4.2.3	"Rec. M.3100 : 1992":locationName	{ 0 0 13 3100 0 7 27 }	c0	c0	c0	x	x	x					
C.4.4.2.4	"Rec. M.3100 : 1992":replaceable	{ 0 0 13 3100 0 7 34 }	x	m	x	x	x	x					
C.4.4.2.5	"Rec. M.3100 : 1992":userLabel	{ 0 0 13 3100 0 7 50 }	c0	c0	c0	x	x	x					
C.4.4.2.6	"Rec. M.3100 : 1992":vendorName	{ 0 0 13 3100 0 7 51 }	c0	c0	c0	x	x	x					
C.4.4.2.7	"Rec. M.3100 : 1992":version	{ 0 0 13 3100 0 7 52 }	c14	c14	c14	x	x	x					
C.4.4.2.8	contactList	{ 1 3 14 2 2 4 7 }	c3	c3	c3	x	x	x					
C.4.4.2.9	contactName	{ 1 3 14 2 2 4 8 }	c4	c4	c4	c4	c4	x					
C.4.4.2.12	customerList	{ 1 3 14 2 2 4 11 }	c5	c5	c5	x	x	x					
C.4.4.2.19	customerName	{ 1 3 14 2 2 4 12 }	c6	c6	c6	c6	c6	x					
C.4.4.2.12	functionList	{ 1 3 14 2 2 4 14 }	c7	c7	c7	x	x	x					
C.4.4.2.13	functionName	{ 1 3 14 2 2 4 15 }	c8	c6	c6	c6	c6	x					
C.4.4.2.14	locationPointer	{ 1 3 14 2 2 4 22 }	c9	c9	c9	x	x	x					
C.4.4.2.15	manufacturerList	{ 1 3 14 2 2 4 23 }	c10	c10	c10	x	x	x					
C.4.4.2.16	manufacturerName	{ 1 3 14 2 2 4 24 }	c11	c11	c11	c11	c11	x					
C.4.4.2.17	opNetworkList	{ 1 3 14 2 2 4 14 }	c12	c12	c12	x	x	x					
C.4.4.2.16	opNetworkName	{ 1 3 14 2 2 4 35 }	c13	c13	c13	c13	c13	x					
C.4.4.2.19	productLabel	{ 1 3 14 2 2 4 45 }	c0	c0	c0	x	x	x					
C.4.4.2.20	serialNumber	{ 1 3 14 2 2 4 50 }	c0	c0	c0	x	x	x					
C.4.4.2.21	serviceList	{ 1 3 14 2 2 4 51 }	c15	c15	c15	x	x	x					
C.4.4.2.22	serviceName	{ 1 3 14 2 2 4 52 }	c16	c16	c16	c16	c16	x					
C.4.4.2.23	softwareList	{ 1 3 14 2 2 4 53 }	c17	c17	c17	x	x	x					

Index	Attribute Template Label	Value of Object Identifier for Attribute	Status					Support					Additional Information		
			set by create	get	replace	add	remove	set by create default	set by create	get	replace	add		remove	set by create default
C.4.4.2.24	softwareName	(1 3 14 2 2 4 54)	c18	c18	c18	c18	c18	x							
C.4.4.2.25	typeText	(1 3 14 2 2 4 59)	c0	c0	c0	x	x	x							
C.4.4.2.26	"Rec. X.721 ISO/IEC 10165-2 : 1992": usageState	(2 9 3 2 7 31)	x	c19		x	x	x							
C.4.4.2.27	vendorList	(1 3 14 2 2 4 61)	c20	c20	c20	c20	c20	x							
C.4.4.2.28	"Rec. X.721 ISO/IEC 10165-2 : 1992": operationalState	(2 9 3 2 7 33)	x	m	x	x	x	x	x						
C.4.4.2.29	"Rec. X.721 ISO/IEC 10165-2 : 1992": administrativeState	(2 9 3 2 7 31)	m	m	m	x	x	x							
C.4.4.2.30	"Rec. M.3100 : 1992": alarmStatus	(0 0 13 3100 0 7 6)	x	c	x	x	x	x							
C.4.4.2.31	"Rec. X.721 ISO/IEC 10165-2 : 1992": availabilityStatus	(2 9 3 2 7 33)	x	m		x	x	x							
C.4.4.2.32	"Rec. M.3100 : 1992": equipmentId	(0 0 13 3100 0 7 20)	m	m	x	x	x	x							
C.4.4.2.33	"Rec. X.721 ISO/IEC 10165-2 : 1992": allomorphs	(2 9 3 2 7 50)	c1	c1	x	x	x	x							
C.4.4.2.34	"Rec. X.721 ISO/IEC 10165-2 : 1992": nameBinding	(2 9 3 2 7 63)	m	m	x	x	x	x							
C.4.4.2.35	"Rec. X.721 ISO/IEC 10165-2 : 1992": objectClass	(2 9 3 2 7 65)	m	m	x	x	x	x							
C.4.4.2.36	"Rec. X.721 ISO/IEC 10165-2 : 1992": packages	(2 9 3 2 7 66)	c2	c2	x	x	x	x							

```

c0 = m if an instance supports it, else -
c1 = m if an object supports allomorphism, else -
c2 = m if any any registered package (other than this package) has been instantiated, else -
c3 = m if an instance supports it and the contactNamePkg is NOT present, else x
c4 = m if an instance supports it and the contactListPkg is NOT present, else x
c5 = m if an instance supports it and the customerNamePkg is NOT present, else x
c6 = m if an instance supports it and the customerListPkg is NOT present, else x
c7 = m if an instance supports it and the functionNamePkg is NOT present, else x
c8 = m if an instance supports it and the functionListPkg is NOT present, else x
c9 = m if an instance supports it and the "Rec. M.3100 : 1992":locationNamePackage is NOT present,
else x
c10= m if an instance supports it and the manufacturerNamePkg is NOT present, else x
c11= m if an instance supports it and the manufacturerListPkg is NOT present, else x
c12= m if an instance supports it and the opNetworkNamePkg is NOT present, else x
c13= m if an instance supports it and the opNetworkListPkg is NOT present, else x
c14= m if "Rec. M.3100 : 1992":versionPackage is also present and if an instance supports it, else -
c15= m if an instance supports it and the serviceNamePkg is NOT present, else x
c16= m if an instance supports it and the serviceListPkg is NOT present, else x
c17= m if an instance supports it and the softwareNamePkg is NOT present, else x
c18= m if an instance supports it and the softwareListPkg is NOT present, else x
c19= m if a resource can detect usage, else -
c20= m if an instance supports it and the "Rec. M.3100 : 1992":vendorNamePackage is NOT present,
else x

```

Table C.4.4.3 - Attribute Group Support

Index	Attribute Group Template Label	Value of Object Identifier for Attribute Group	Status		Support		Additional Information
			g e t	s e t	g e t	s e t	
C.4.4.3.1	"Rec. X.721 ISO/IEC 10165-2 : 1992":state	(2 9 3 2 8 1)	m	x			

Table C.4.4.5 - Notification Support

Index	Notification Type Label	Value of Notification Type Identifier	Status	Support		Additional Info	Sub-Index	Notification Field Name Label	Value of OID of Attribute associated with Field	Status	Support	Additional Information
				conform	non							
C.4.4.5.1	"Rec. X.721 ISO/IEC 10165-2 : 1992": attributeValueChange	(2 9 3 2 10 1)	m				C.4.4.5.1.1	additionalInformation	(2 9 3 2 7 6)			
							C.4.4.5.1.2	additionalText	(2 9 3 2 7 7)			
							C.4.4.5.1.3	attributeIdentifierList	(2 9 3 2 7 8)			
							C.4.4.5.1.4	attributeValueChangeDefinition	(2 9 3 2 7 10)			
							C.4.4.5.1.5	correlatedNotifications	(2 9 3 2 7 12)			
							C.4.4.5.1.6	notificationIdentifier	(2 9 3 2 7 16)			
							C.4.4.5.1.7	sourceIndicator	(2 9 3 2 7 26)			
C.4.4.5.2	"Rec. X.721 ISO/IEC 10165-2 : 1992": objectCreation	(2 9 3 2 10 6)	m				C.4.4.5.2.1	additionalInformation	(2 9 3 2 7 6)			
							C.4.4.5.2.2	additionalText	(2 9 3 2 7 7)			
							C.4.4.5.2.3	attributeList	(2 9 3 2 7 9)			
							C.4.4.5.2.4	correlatedNotifications	(2 9 3 2 7 12)			
							C.4.4.5.2.5	notificationIdentifier	(2 9 3 2 7 16)			
							C.4.4.5.2.6	sourceIndicator	(2 9 3 2 7 26)			
C.4.4.5.3	"Rec. X.721 ISO/IEC 10165-2 : 1992": objectDeletion	(2 9 3 2 10 7)	m				C.4.4.5.3.1	additionalInformation	(2 9 3 2 7 6)			
							C.4.4.5.3.2	additionalText	(2 9 3 2 7 7)			
							C.4.4.5.3.3	attributeList	(2 9 3 2 7 9)			
							C.4.4.5.3.4	correlatedNotifications	(2 9 3 2 7 12)			
							C.4.4.5.3.5	notificationIdentifier	(2 9 3 2 7 16)			

Index	Notification Type Label	Value of Notification Type Identifier	Status	Support		Add Info	Sub-Index	Notification Field Name Label	Value of OID of Attribute associated with Field	Status	Support	Additional Information
				conf	non							
							C.4.4.5.3.6	sourceIndicator	{ 2 9 3 2 7 0 26 }			
C.4.4.5.4	"Rec. X.721 ISO/IEC 10165-2 : 1992": stateChange	{ 2 9 3 2 10 14 }	m				C.4.4.5.4.1	additionalInfo	{ 2 9 3 2 7 0 6 }			
							C.4.4.5.4.2	additionalText	{ 2 9 3 2 7 0 7 }			
							C.4.4.5.4.3	attributeIdentifierList	{ 2 9 3 2 7 0 8 }			
							C.4.4.5.4.4	correlatedNotifications	{ 2 9 3 2 7 0 12 }			
							C.4.4.5.4.5	notificationIdentifier	{ 2 9 3 2 7 0 16 }			
							C.4.4.5.4.6	sourceIndicator	{ 2 9 3 2 7 0 26 }			
							C.4.4.5.4.2	stateChangeDefinition	{ 2 9 3 2 7 0 28 }	●		
C.4.4.5.5	"Rec. X.721 ISO/IEC 10165-2 : 1992": communications Alarm	{ 2 9 3 2 10 2 }	m				C.4.4.5.5.1	additionalInfo	{ 2 9 3 2 7 0 6 }			
							C.4.4.5.5.2	additionalText	{ 2 9 3 2 7 0 7 }			
							C.4.4.5.5.3	backUpObject	{ 2 9 3 2 7 0 41 }			
							C.4.4.5.5.4	backedUpStatus	{ 2 9 3 2 7 0 11 }			
							C.4.4.5.5.5	correlatedNotifications	{ 2 9 3 2 7 0 12 }			
							C.4.4.5.5.6	notificationIdentifier	{ 2 9 3 2 7 0 16 }			
							C.4.4.5.5.7	perceivedSeverity	{ 2 9 3 2 7 0 18 }	●		
							C.4.4.5.5.8	probableCause	{ 2 9 3 2 7 0 18 }	●		
							C.4.4.5.5.9	proposedRepairActions	{ 2 9 3 2 7 0 18 }			
							C.4.4.5.5.10	specificProblems	{ 2 9 3 2 7 0 27 }			
							C.4.4.5.5.11	stateChangeDefinition	{ 2 9 3 2 7 0 28 }			

Index	Notification Type Label	Value of Notification Type Identifier	Status	Support		Add Info	Sub-Index	Notification Field Name Label	Value of OID of Attribute associated with Field	Status	Support	Additional Information
				conform	non							
							C.4.4.5.5.1	thresholdInfo	{ 2 9 3 2 7 29 }			
							C.4.4.5.5.13	trendIndication	{ 2 9 3 2 7 30 }			
C.4.4.5.6	"Rec. X.721 ISO/IEC 10165-2 : 1992": processingErrorAlarm	{ 2 9 3 2 10 10 }	m				C.4.4.5.5.1	additionalInformation	{ 2 9 3 2 7 6 }			
							C.4.4.5.5.2	additionalText	{ 2 9 3 2 7 7 }			
							C.4.4.5.5.3	backUpObject	{ 2 9 3 2 7 41 }	m		
							C.4.4.5.5.8	backedUpStatus	{ 2 9 3 2 7 11 }			
							C.4.4.5.5.5	correlatedNotifications	{ 2 9 3 2 7 12 }			
							C.4.4.5.5.6	notificationIdentifier	{ 2 9 3 2 7 16 }			
							C.4.4.5.5.9	perceivedSeverity	{ 2 9 3 2 7 17 }	m		
							C.4.4.5.5.8	probableCause	{ 2 9 3 2 7 18 }			
							C.4.4.5.5.9	proposedRepairActions	{ 2 9 3 2 7 18 }			
							C.4.4.5.5.10	specificProblems	{ 2 9 3 2 7 27 }			
							C.4.4.5.5.11	stateChangeDefinition	{ 2 9 3 2 7 28 }			
							C.4.4.5.5.12	thresholdInfo	{ 2 9 3 2 7 29 }			
							C.4.4.5.5.13	trendIndication	{ 2 9 3 2 7 30 }			
C.4.4.5.7	"Rec. X.721 ISO/IEC 10165-2 : 1992": environmentalAlarm	{ 2 9 3 2 10 3 }	m				C.4.4.5.6.1	additionalInformation	{ 2 9 3 2 7 6 }			
							C.4.4.5.6.2	additionalText	{ 2 9 3 2 7 7 }			
							C.4.4.5.6.3	backUpObject	{ 2 9 3 2 7 41 }			
							C.4.4.5.6.4	backedUpStatus	{ 2 9 3 2 7 11 }			

Index	Notification Type Label	Value of Notification Type Identifier	Stat us	Support		Add Info	Sub- Index	Notification Field Name Label	Value of OID of Attr Type associated with Field	Stat us	Support	Additional Informati on
				con f	on n							
							C.4.4 .5.6. 5	correlatedNoti fications	(2 9 3 2 7 o 12)			
							C.4.4 .5.6. 6	notificationId entifier	(2 9 3 2 7 o 16)			
							C.4.4 .5.6. 7	perceivedSever ity	(2 9 3 2 7 m 17)			
							C.4.4 .5.6. 8	probableCause	(2 9 3 2 7 m 18)			
							C.4.4 .5.6. 9	proposedRepair Actions	(2 9 3 2 7 o 17)			
							C.4.4 .5.6. 10	specificProble ms	(2 9 3 2 7 o 27)			
							C.4.4 .5.6. 11	stateChangeDef inition	(2 9 3 2 7 o 28)			
							C.4.4 .5.6. 12	thresholdInfo	(2 9 3 2 7 o 29)			
							C.4.4 .5.6. 13	trendIndicatio n	(2 9 3 2 7 o 30)			
C.4.4.5.8	"Rec. X.721 ISO/IEC 10165- 2 : 1992": equipmentAlarm	(2 9 3 2 10 4)	m				C.4.4 .5.7. 14	additionalInfo rmation	(2 9 3 2 7 o 6)			
							C.4.4 .5.7. 15	additionalText	(2 9 3 2 7 o 7)			
							C.4.4 .5.7. 16	backUpObject	(2 9 3 2 7 o 41)			
							C.4.4 .5.7. 17	backedUpStatus	(2 9 3 2 7 o 18)			
							C.4.4 .5.7. 18	correlatedNoti fications	(2 9 3 2 7 o 12)			
							C.4.4 .5.7. 19	notificationId entifier	(2 9 3 2 7 o 16)			
							C.4.4 .5.7. 20	perceivedSever ity	(2 9 3 2 7 m 17)			
							C.4.4 .5.7. 21	probableCause	(2 9 3 2 7 o 18)			
							C.4.4 .5.7. 22	proposedRepair Actions	(2 9 3 2 7 o 17)			
							C.4.4 .5.7. 23	specificProble ms	(2 9 3 2 7 o 27)			

Index	Notification Type Label	Value of Notification Type Identifier	Status	Support		Add Info	Sub-Index	Notification Field Name Label	Value of OID of Attribute associated with Field	Status	Support	Additional Information
				conformance	non							
							C.4.4.11	stateChangeDefinition	{ 2 9 3 2 7 28 }			
							C.4.4.12	thresholdInfo	{ 2 9 3 2 7 29 }			
							C.4.4.13	trendIndication	{ 2 9 3 2 7 30 }			

C.4.5 OMNIPoint Network MOCS Proforma

Managed Object class template label	Value of Object identifier for class
"OP1 Library Vol. 1":opNetwork	{ 1 3 14 2 2 1 5 }

Are all mandatory features of the class supported? Yes _____ No _____

Table C.4.5.1 - Name Binding Support

Index	Name Binding Template Label	Value of Object Identifier for Name Binding	Superior Object Class Template Label	Status	Support	Status		Support		Additional Information
						create	delete	create	delete	
C.4.5.1.1	"Rec. M.3100 : 1992": network-network	{ 0 0 13 3100 0 6 17 }	"Rec. M.3100 : 1992": network	o		x	x			
C.4.5.1.2	network-opNetwork-1	{ 1 3 14 2 2 3 14 }	"Rec. M.3100 : 1992": network	o		m	m			
C.4.5.1.3	network-opNetwork-2	{ 1 3 14 2 2 3 15 }	"Rec. M.3100 : 1992": network	o		m	m			
C.4.5.1.4	opNetwork-root	{ 1 3 14 2 2 3 16 }	"Rec. X.600 ISO/IEC 9834-1 : 1992":root	o		m	m			

Table C.4.5.2 - Attribute Support

Index	Attribute Template Label	Value of Object Identifier for Attribute	Status					Support					Additional Information
			s e t b y c r e a t e	g e t	r e p l a c e	a d d	r e m o v e	s e t b y c r e a t e	g e t	r e p l a c e	a d d	r e m o v e	
C.4.5.2.1	"Rec. M.3100 : 1992":networkId	{ 0 0 13 3100 0 7 3 }	m	m	x	x	x	x					
C.4.5.2.2	"Rec. M.3100 : 1992":userLabel	{ 0 0 13 3100 0 7 50 }	c0	c0	c0	x	x	x					
C.4.5.2.3	networkTitle	{ 1 3 14 2 2 4 31 }	m	m	x	x	x	x					
C.4.5.2.4	"Rec. X.721 ISO/IEC 10165-2 : 1992":allomorphs	{ 2 9 3 2 7 50 }	c1	c1	x	x	x	x					
C.4.5.2.5	"Rec. X.721 ISO/IEC 10165-2 : 1992":nameBinding	{ 2 9 3 2 7 63 }	m	m	x	x	x	x					
C.4.5.2.6	"Rec. X.721 ISO/IEC 10165-2 : 1992":objectClass	{ 2 9 3 2 7 65 }	m	m	x	x	x	x					
C.4.5.2.7	"Rec. X.721 ISO/IEC 10165-2 : 1992":packages	{ 2 9 3 2 7 66 }	c2	c2	x	x	x	x					

c0 = m if an instance supports it, else -

c1 = m if an object supports allomorphism, else -

c2 = m if any any registered package (other than this package) has been instantiated, else -

Table C.4.5.5 - Notification Support

Index	Notification Type Label	Value of Notification Type Identifier	Status	Support		Add Info	Sub-Index	Notification Field Name Label	Value of OID of Attribute associated with Field	Status	Support	Additional Information
				con	fon							
C.4.5.5.1	"Rec. X.721 ISO/IEC 10165-2 : 1992": attributeValueChange	{ 2 9 3 2 10 1 }	m				C.4.5.5.1.3	additionalInformation	{ 2 9 3 2 70 6 }			
							C.4.5.5.1.2	additionalText	{ 2 9 3 2 70 8 }			
							C.4.5.5.1.3	attributeIdentifierList	{ 2 9 3 2 70 8 }			
							C.4.5.5.1.4	attributeValueChangeDefinition	{ 2 9 3 2 70 10 }			
							C.4.5.5.1.5	correlatedNotifications	{ 2 9 3 2 70 12 }			
							C.4.5.5.1.3	notificationIdentifier	{ 2 9 3 2 70 16 }			
							C.4.5.5.1.2	sourceIndicator	{ 2 9 3 2 70 26 }			
C.4.5.5.2	"Rec. X.721 ISO/IEC 10165-2 : 1992": objectCreation	{ 2 9 3 2 10 6 }	m				C.4.5.5.2.1	additionalInformation	{ 2 9 3 2 70 6 }			
							C.4.5.5.2.2	additionalText	{ 2 9 3 2 70 7 }			
							C.4.5.5.2.3	attributeList	{ 2 9 3 2 70 9 }			
							C.4.5.5.2.4	correlatedNotifications	{ 2 9 3 2 70 12 }			
							C.4.5.5.2.5	notificationIdentifier	{ 2 9 3 2 70 16 }			
							C.4.5.5.2.6	sourceIndicator	{ 2 9 3 2 70 26 }			
C.4.5.5.3	"Rec. X.721 ISO/IEC 10165-2 : 1992": objectDeletion	{ 2 9 3 2 10 7 }	m				C.4.5.5.3.1	additionalInformation	{ 2 9 3 2 70 6 }			
							C.4.5.5.3.2	additionalText	{ 2 9 3 2 70 8 }			
							C.4.5.5.3.1	attributeList	{ 2 9 3 2 70 9 }			
							C.4.5.5.3.4	correlatedNotifications	{ 2 9 3 2 70 12 }			
							C.4.5.5.3.5	notificationIdentifier	{ 2 9 3 2 70 16 }			

Index	Notification Type Label	Value of Notification Type Identifier	Status	Support	Add Info	Sub-Index	Notification Field Name Label	Value of OID of Attribute associated with Field	Status	Support	Additional Information
				c o n f	n o n	C.4.5 .5.3. 6	sourceIndicator	{ 2 9 3 2 7 0 26 }			

C.4.6 Processing Entity MOCS Proforma

Managed Object class template label	Value of Object identifier for class
"OP1 Library Vol. 1":processingEntity	{ 1 3 14 2 2 1 6 }

Are all mandatory features of the class supported? Yes _____ No _____

Table C.4.6.1 - Name Binding Support

Index	Name Binding Template Label	Value of Object Identifier for Name Binding	Superior Object Class Template Label	Status	Support	Status				Additional Information
						c r e a t e	d e l e t e	c r e a t e	d e l e t e	
C.4.6.1.1	"Rec. M.3100 : 1992": equipment-equipment	{ 0 0 13 3100 0 6 10 }	"Rec. M.3100 : 1992":equipment	o		m	m			
C.4.6.1.2	"Rec. M.3100 : 1992": equipment-ManagedElement	{ 0 0 13 3100 0 6 9 }	"Rec. M.3100 : 1992": managedElement	o		m	m			
C.4.6.1.3	opEquipment- computerSystem	{ 1 3 14 2 2 3 10 }	computerSystem	o		m	m			
C.4.6.1.4	opEquipment-system	{ 1 3 14 2 2 3 11 }	"Rec. X.721 ISO/IEC 10165-2 : 1992":system	o		m	m			
C.4.6.1.5	opEquipment-equipment	{ 1 3 14 2 2 3 12 }	"Rec. M.3100 : 1992":equipment	o		m	m			
C.4.6.1.6	opEquipment-opNetwork	{ 1 3 14 2 2 3 13 }	opNetwork	o		m	m			

Table C.4.6.2 - Attribute Support

Index	Attribute Template Label	Value of Object Identifier for Attribute	Status					Support					Additional Information
			set by create	get	replace	add	remove	set by create default	get	replace	add	remove	
C.4.6.2.1	"Rec. M.3100 : 1992":affectedObjectList	{ 0 0 13 3100 0 7 2 }	x	c0	x	x	x	x					
C.4.6.2.7	"Rec. M.3100 : 1992":currentProblemList	{ 0 0 13 3100 0 7 17 }	x	c0	x	x	x	x					
C.4.6.2.3	"Rec. M.3100 : 1992":locationName	{ 0 0 13 3100 0 7 27 }	c0	c0	c0	x	x	x					
C.4.6.2.4	"Rec. M.3100 : 1992":replaceable	{ 0 0 13 3100 0 7 34 }	x	m	x	x	x	x					
C.4.6.2.5	"Rec. M.3100 : 1992":userLabel	{ 0 0 13 3100 0 7 50 }	c0	c0	c0	x	x	x					
C.4.6.2.6	"Rec. M.3100 : 1992":vendorName	{ 0 0 13 3100 0 7 51 }	c0	c0	c0	x	x	x					
C.4.6.2.7	"Rec. M.3100 : 1992":version	{ 0 0 13 3100 0 7 50 }	c14	c14	c14	x	x	x					
C.4.6.2.8	contactList	{ 1 3 14 2 2 4 7 }	c3	c3	c3	x	x	x					
C.4.6.2.9	contactName	{ 1 3 14 2 2 4 8 }	c4	c4	c4	c4	c4	x					
C.4.6.2.10	customerList	{ 1 3 14 2 2 4 11 }	c5	c5	c5	x	x	x					
C.4.6.2.11	customerName	{ 1 3 14 2 2 4 12 }	c8	c8	c8	c8	c6	x					
C.4.6.2.12	functionList	{ 1 3 14 2 2 4 14 }	c7	c7	c7	x	x	x					
C.4.6.2.13	functionName	{ 1 3 14 2 2 4 35 }	c8	c8	c8	c8	c8	x					
C.4.6.2.14	locationPointer	{ 1 3 14 2 2 4 22 }	c9	c9	c9	x	x	x					
C.4.6.2.15	manufacturerList	{ 1 3 14 2 2 4 23 }	c10	c10	c10	x	x	x					
C.4.6.2.16	manufacturerName	{ 1 3 14 2 2 4 24 }	c11	c11	c11	c11	c11	x					
C.4.6.2.17	opNetworkList	{ 1 3 14 2 2 4 34 }	c12	c12	c12	x	x	x					
C.4.6.2.18	opNetworkName	{ 1 3 14 2 2 4 35 }	c13	c13	c13	c13	c13	x					
C.4.6.2.19	productLabel	{ 1 3 14 2 2 4 45 }	c0	c0	c0	x	x	x					
C.4.6.2.20	serialNumber	{ 1 3 14 2 2 4 50 }	c0	c0	c0	x	x	x					
C.4.6.2.21	serviceList	{ 1 3 14 2 2 4 51 }	c15	c15	c15	x	x	x					
C.4.6.2.22	serviceName	{ 1 3 14 2 2 4 50 }	c16	c16	c16	c16	c16	x					
C.4.6.2.23	softwareList	{ 1 3 14 2 2 4 53 }	c17	c17	c17	x	x	x					

Index	Attribute Template Label	Value of Object Identifier for Attribute	Status					Support					Additional Information	
			set by create	get	replace	add	remove	set by create default	get	replace	add	remove		set by create default
C.4.6.2.24	softwareName	(1 3 14 2 2 4 54)	c18	c18	c18	c18	c18	x						
C.4.6.2.25	typeText	(1 3 14 2 2 4 59)	c0	c0	c0	x	x	x						
C.4.6.2.26	"Rec. X.721 ISO/IEC 10165-2 : 1992": usageState	(2 9 3 2 7 35)	x	c19		x	x	x						
C.4.6.2.27	vendorList	(1 3 14 2 2 4 61)	c20	c20	c20	c20	c20	x						
C.4.6.2.28	"Rec. X.721 ISO/IEC 10165-2 : 1992": operationalState	(2 9 3 2 7 35)	x	m		x	x	x						
C.4.6.2.29	"Rec. X.721 ISO/IEC 10165-2 : 1992": administrativeState	(2 9 3 2 7 31)	m	m	m	x	x	x						
C.4.6.2.30	"Rec. M.3100 : 1992": alarmStatus	(0 0 13 3100 0 7 6)	x	c	x	x	x	x						
C.4.6.2.31	"Rec. X.721 ISO/IEC 10165-2 : 1992": availabilityStatus	(2 9 3 2 7 33)	x	m	x	x	x	x						
C.4.6.2.32	"Rec. M.3100 : 1992": equipmentId	(0 0 13 3100 0 7 20)	m	m	x	x	x	x						
C.4.6.2.33	"Rec. X.721 ISO/IEC 10165-2 : 1992": allomorphs	(2 9 3 2 7 50)	c1	c1		x	x	x						
C.4.6.2.34	"Rec. X.721 ISO/IEC 10165-2 : 1992": nameBinding	(2 9 3 2 7 63)	m	m		x	x	x						
C.4.6.2.35	"Rec. X.721 ISO/IEC 10165-2 : 1992": objectClass	(2 9 3 2 7 65)	m	m		x	x	x						
C.4.6.2.36	"Rec. X.721 ISO/IEC 10165-2 : 1992": packages	(2 9 3 2 7 66)	c2	c2		x	x	x						
C.4.6.2.37	addressingSize	(1 3 14 2 2 4 2)	x	c21	x	x	x	x						
C.4.6.2.38	endianess	(1 3 14 2 2 4 13)	x	c21	x	x	x	x						
C.4.6.2.39	cpuUtilization	(1 3 14 2 2 4 10)	x	c0	x	x	x	x						
C.4.6.2.40	memorySize	(1 3 14 2 2 4 28)	x	c21	x	x	x	x						
C.4.6.2.41	memoryUtilization	(1 3 14 2 2 4 29)	x	c0	x	x	x	x						
C.4.6.2.42	upTime	(1 3 14 2 2 4 60)	x	c0	x	x	x	x						
C.4.6.2.43	cpuType	(1 3 14 2 2 4 9)	x	m	x	x	x	x						
C.4.6.2.44	osInfo	(1 3 14 2 2 4 36)	x	m	x	x	x	x						

c0 = m if an instance supports it, else -
 c1 = m if an object supports allomorphism, else -
 c2 = m if any any registered package (other than this package) has been instantiated, else -
 c3 = m if an instance supports it and the contactNamePkg is NOT present, else x
 c4 = m if an instance supports it and the contactListPkg is NOT present, else x
 c5 = m if an instance supports it and the customerNamePkg is NOT present, else x
 c6 = m if an instance supports it and the customerListPkg is NOT present, else x
 c7 = m if an instance supports it and the functionNamePkg is NOT present, else x
 c8 = m if an instance supports it and the functionListPkg is NOT present, else x
 c9 = m if an instance supports it and the "Rec. M.3100 : 1992":locationNamePackage is NOT present,
 else x
 c10= m if an instance supports it and the manufacturerNamePkg is NOT present, else x
 c11= m if an instance supports it and the manufacturerListPkg is NOT present, else x
 c12= m if an instance supports it and the opNetworkNamePkg is NOT present, else x
 c13= m if an instance supports it and the opNetworkListPkg is NOT present, else x
 c14= m if "Rec. M.3100 : 1992":versionPackage is also present and if an instance supports it, else -
 c15= m if an instance supports it and the serviceNamePkg is NOT present, else x
 c16= m if an instance supports it and the serviceListPkg is NOT present, else x
 c17= m if an instance supports it and the softwareNamePkg is NOT present, else x
 c18= m if an instance supports it and the softwareListPkg is NOT present, else x
 c19= m if a resource can detect usage, else -
 c20= m if an instance supports it and the "Rec. M.3100 : 1992":vendorNamePackage is NOT present,
 else x
 c21= m if relevant to the underlying resource, else -

Table C.4.6.3 - Attribute Group Support

Index	Attribute Group Template Label	Value of Object Identifier for Attribute Group	Status		Support		Additional Information
			g e t	s e t	g e t	s e t	
C.4.6.3.1	"Rec. X.721 ISO/IEC 10165-2 : 1992":state	{ 2 9 3 2 8 1 }	m	x			

Table C.4.6.5 - Notification Support

Index	Notification Type Label	Value of Notification Type Identifier	Status	Support		Add Info	Sub-Index	Notification Field Name Label	Value of OID of Attribute associated with Field	Status	Support	Additional Information
				con	firm							
C.4.6.5.1	"Rec. X.721 ISO/IEC 10165-2 : 1992": attributeValueChange	(2 9 3 2 10 1)	m				C.4.6.5.1.1	additionalInformation	(2 9 3 2 706)			
							C.4.6.5.1.2	additionalText	(2 9 3 2 707)			
							C.4.6.5.1.3	attributeIdentifierList	(2 9 3 2 708)			
							C.4.6.5.1.4	attributeValueChangeDefinition	(2 9 3 2 710)	m		
							C.4.6.5.1.5	correlatedNotifications	(2 9 3 2 712)			
							C.4.6.5.1.6	notificationIdentifier	(2 9 3 2 716)			
							C.4.6.5.1.7	sourceIndicator	(2 9 3 2 726)			
C.4.6.5.2	"Rec. X.721 ISO/IEC 10165-2 : 1992": objectCreation	(2 9 3 2 10 6)	m				C.4.6.5.2.1	additionalInformation	(2 9 3 2 706)			
							C.4.6.5.2.2	additionalText	(2 9 3 2 709)			
							C.4.6.5.2.3	attributeList	(2 9 3 2 709)			
							C.4.6.5.2.6	correlatedNotifications	(2 9 3 2 712)			
							C.4.6.5.2.5	notificationIdentifier	(2 9 3 2 716)			
							C.4.6.5.2.6	sourceIndicator	(2 9 3 2 726)			
C.4.6.5.3	"Rec. X.721 ISO/IEC 10165-2 : 1992": objectDeletion	(2 9 3 2 10 7)	m				C.4.6.5.3.1	additionalInformation	(2 9 3 2 706)			
							C.4.6.5.3.2	additionalText	(2 9 3 2 707)			
							C.4.6.5.3.3	attributeList	(2 9 3 2 709)			
							C.4.6.5.3.4	correlatedNotifications	(2 9 3 2 712)			
							C.4.6.5.3.5	notificationIdentifier	(2 9 3 2 716)			

Index	Notification Type Label	Value of Notification Type Identifier	Status	Support	Add Info	Sub-Index	Notification Field Name Label	Value of OID of Attribute associated with Field	Status	Support	Additional Information
							C.4.6sourceIndicator .5.3.r 6	(2 9 3 2 7 0 26)			
C.4.6.5.4	"Rec. X.721 ISO/IEC 10165- 2 : 1992": stateChange	(2 9 3 2 10 14)	m				C.4.6additionalInfo .5.4.r 1	(2 9 3 2 7 0 6)			
							C.4.6additionalText .5.4. 2	(2 9 3 2 7 0 7)			
							C.4.6attributeIdent .5.4.ifierList 3	(2 9 3 2 7 0 8)			
							C.4.6correlatedNoti .5.4.fications 4	(2 9 3 2 7 0 12)			
							C.4.6notificationId .5.4.entifier 5	(2 9 3 2 7 0 16)			
							C.4.6sourceIndicator .5.4.r 6	(2 9 3 2 7 0 26)			
							C.4.6stateChangeDef .5.4.inition 2	(2 9 3 2 7 0 26)	m		
C.4.6.5.5	"Rec. X.721 ISO/IEC 10165- 2 : 1992": communications Alarm	(2 9 3 2 10 2)	m				C.4.6additionalInfo .5.5.r 1	(2 9 3 2 7 0 6)			
							C.4.6additionalText .5.5. 2	(2 9 3 2 7 0 7)			
							C.4.6backUpObject .5.5. 1	(2 9 3 2 7 0 41)			
							C.4.6backedUpStatus .5.5. 4	(2 9 3 2 7 0 11)			
							C.4.6correlatedNoti .5.5.fications 5	(2 9 3 2 7 0 12)			
							C.4.6notificationId .5.5.entifier 6	(2 9 3 2 7 0 16)			
							C.4.6perceivedSever .5.5.ity 7	(2 9 3 2 7 0 17)	●		
							C.4.6probableCause .5.5. 8	(2 9 3 2 7 0 18)	m		
							C.4.6proposedRepair .5.5.Actions 9	(2 9 3 2 7 0 19)			
							C.4.6specificProble .5.5.ms 10	(2 9 3 2 7 0 27)			
							C.4.6stateChangeDef .5.5.inition 11	(2 9 3 2 7 0 28)			

PART 18: Network Management

December 1992 (Stable)

Index	Notification Type Label	Value of Notification Type Identifier	Status	Support		Add Info	Sub-Index	Notification Field Name Label	Value of OID of Attribute associated with Field	Status	Support	Additional Information
				con	firm							
							C.4.6.5.5.12	thresholdInfo	(2 9 3 2 7 29)			
							C.4.6.5.5.13	trendIndication	(2 9 3 2 7 30)			
C.4.6.5.6	"Rec. X.721 ISO/IEC 10165-2 : 1992": processingErrorAlarm	(2 9 3 2 10 10)	m				C.4.6.5.5.6	additionalInformation	(2 9 3 2 7 6)			
							C.4.6.5.5.2	additionalText	(2 9 3 2 7 7)			
							C.4.6.5.5.3	backUpObject	(2 9 3 2 7 41)			
							C.4.6.5.5.8	backedUpStatus	(2 9 3 2 7 11)			
							C.4.6.5.5.5	correlatedNotifications	(2 9 3 2 7 12)			
							C.4.6.5.5.6	notificationIdentifier	(2 9 3 2 7 16)			
							C.4.6.5.5.7	perceivedSeverity	(2 9 3 2 7 17)			
							C.4.6.5.5.8	probableCause	(2 9 3 2 7 18)			
							C.4.6.5.5.6	proposedRepairActions	(2 9 3 2 7 19)			
							C.4.6.5.5.10	specificProblems	(2 9 3 2 7 27)			
							C.4.6.5.5.11	stateChangeDefinition	(2 9 3 2 7 23)			
							C.4.6.5.5.12	thresholdInfo	(2 9 3 2 7 29)			
							C.4.6.5.5.13	trendIndication	(2 9 3 2 7 30)			
C.4.6.5.7	"Rec. X.721 ISO/IEC 10165-2 : 1992": environmentalAlarm	(2 9 3 2 10 3)	m				C.4.6.5.6.1	additionalInformation	(2 9 3 2 7 6)			
							C.4.6.5.6.2	additionalText	(2 9 3 2 7 7)			
							C.4.6.5.6.3	backUpObject	(2 9 3 2 7 41)			
							C.4.6.5.6.4	backedUpStatus	(2 9 3 2 7 11)			

Index	Notification Type Label	Value of Notification Type Identifier	Status	Support		Additional Info	Sub-Index	Notification Field Name Label	Value of OID of Attribute associated with Field	Status	Support	Additional Information
				conf	non							
							C.4.6.5	correlatedNotifications	{ 2 9 3 2 7 m 12 }			
							C.4.6.6	notificationIdentifier	{ 2 9 3 2 7 o 30 }			
							C.4.6.8	perceivedSeverity	{ 2 9 3 2 7 m 17 }			
							C.4.6.8	probableCause	{ 2 9 3 2 7 m 18 }			
							C.4.6.9	proposedRepairActions	{ 2 9 3 2 7 o 19 }			
							C.4.6.10	specificProblems	{ 2 9 3 2 7 m 27 }			
							C.4.6.11	stateChangeDefinition	{ 2 9 3 2 7 o 28 }			
							C.4.6.12	thresholdInfo	{ 2 9 3 2 7 o 29 }			
							C.4.6.10	trendIndication	{ 2 9 3 2 7 o 30 }			
C.4.6.5.8	"Rec. X.721 ISO/IEC 10165-2 : 1992": equipmentAlarm	{ 2 9 3 2 10 4 }	m				C.4.6.11	additionalInformation	{ 2 9 3 2 7 m 6 }			
							C.4.6.12	additionalText	{ 2 9 3 2 7 o 7 }			
							C.4.6.13	backUpObject	{ 2 9 3 2 7 m 41 }			
							C.4.6.16	backedUpStatus	{ 2 9 3 2 7 o 11 }			
							C.4.6.15	correlatedNotifications	{ 2 9 3 2 7 m 12 }			
							C.4.6.16	notificationIdentifier	{ 2 9 3 2 7 o 30 }			
							C.4.6.17	perceivedSeverity	{ 2 9 3 2 7 m 17 }			
							C.4.6.18	probableCause	{ 2 9 3 2 7 m 18 }			
							C.4.6.17	proposedRepairActions	{ 2 9 3 2 7 o 19 }			
							C.4.6.10	specificProblems	{ 2 9 3 2 7 o 27 }			

Index	Notification Type Label	Value of Notification Type Identifier	Status	Support	Add Info	Sub-Index	Notification Field Name Label	Value of OID of Attribute associated with Field	Status	Support	Additional Information
				conf							
						C.4.6.5.7.11	stateChangeDefinition	(2 9 3 2 7 28)			
						C.4.6.5.7.12	thresholdInfo	(2 9 3 2 7 29)			
						C.4.6.5.7.13	trendIndicator	(2 9 3 2 7 30)			

C.4.7 Transport Connection MOCS Proforma

Managed Object class template label	Value of Object identifier for class
"OP1 Library Vol. 1":transportConnection	(1 3 14 2 2 1 7)

Are all mandatory features of the class supported? Yes _____ No _____

Table C.4.7.1 - Name Binding Support

Index	Name Binding Template Label	Value of Object Identifier for Name Binding	Superior Object Class Template Label	Status	Support	Status	Support	Additional Information
						create	delete	
C.4.7.1.1	transportConnection-coTransportProtocolLayerEntity	(1 3 14 2 2 3 17)	coTransportProtocolLayerEntity			x	m	

Table C.4.7.2 - Attribute Support

Index	Attribute Template Label	Value of Object Identifier for Attribute	Status					Support					Additional Information	
			set by create	get	replace	add	remove	set by create default	set by create	replace	add	remove		set by default
C.4.7.2.1	maxRetransmissions	{ 1 3 14 2 2 4 27 }	x	c0	x	x	x	x						
C.4.7.2.2	retransmissionTime	{ 1 3 14 2 2 4 48 }	x	c0	x	x	x	x						
C.4.7.2.3	retransmissionTimerInitialValue	{ 1 3 14 2 2 4 49 }	x	c0	x	x	x	x						
C.4.7.2.4	"Rec. X.721 ISO/IEC 10165-2 : 1992": pduRetransmittedErrorCounter	{ 2 9 3 2 7 87 }	x	c0	x	x	x	x						
C.4.7.2.5	"Rec. X.721 ISO/IEC 10165-2 : 1992": octetsRetransmittedErrorCounter	{ 2 9 3 2 7 79 }	x	c0	x	x	x	x						
C.4.7.2.6	"Rec. X.721 ISO/IEC 10165-2 : 1992": pduRetransmittedErrorThreshold	{ 2 9 3 2 7 102 }	x	c0	c0	x	x	x						
C.4.7.2.7	"Rec. X.721 ISO/IEC 10165-2 : 1992": outgoingProtocolErrorCounter	{ 2 9 3 2 7 85 }	x	c0	x	x	x	x						
C.4.7.2.8	checksumPDUsDiscardedCounter	{ 1 3 14 2 2 4 3 }	x	c0	x	x	x	x						
C.4.7.2.9	localTransportConnectionEndpoint	{ 1 3 14 2 2 4 21 }	x	m	x	x	x	x						
C.4.7.2.10	remoteTransportConnectionEndpoint	{ 1 3 14 2 2 4 47 }	x	m	x	x	x	x						
C.4.7.2.11	transportConnectionReference	{ 1 3 14 2 2 4 57 }	x	m	x	x	x	x						
C.4.7.2.12	localNetworkAddress	{ 1 3 14 2 2 4 18 }	x	m	x	x	x	x						
C.4.7.2.13	remoteNetworkAddress	{ 1 3 14 2 2 4 46 }	x	m	x	x	x	x						
C.4.7.2.14	inactivityTimeout	{ 1 3 14 2 2 4 17 }	x	m	x	x	x	x						
C.4.7.2.15	inactivityTime	{ 1 3 14 2 2 4 16 }	x	m	x	x	x	x						
C.4.7.2.16	maxPDUSize	{ 1 3 14 2 2 4 26 }	x	m	x	x	x	x						
C.4.7.2.17	"Rec. X.721 ISO/IEC 10165-2 : 1992": pduSentCounter	{ 2 9 3 2 7 88 }	x	m	x	x	x	x						
C.4.7.2.18	"Rec. X.721 ISO/IEC 10165-2 : 1992": pduReceivedCounter	{ 2 9 3 2 7 86 }	x	m	x	x	x	x						

Index	Attribute Template Label	Value of Object Identifier for Attribute	Status					Support					Additional Information	
			set by create	get	replace	add	remove	set by create default	get	replace	add	remove		set to default
C.4.7.2.19	"Rec. X.721 ISO/IEC 10165-2 : 1992": octetsSentCounter	(2 9 3 2 7 80)	x	m	x	x	x	x						
C.4.7.2.20	"Rec. X.721 ISO/IEC 10165-2 : 1992": octetsReceivedCounter	(2 9 3 2 7 78)	x	m	x	x	x	x						
C.4.7.2.21	"Rec. X.721 ISO/IEC 10165-2 : 1992": incomingProtocolErrorCounter	(2 9 3 2 7 77)	x	m	x	x	x	x						
C.4.7.2.22	transportConnectionId	(1 3 14 2 2 4 56)	x	m	x	x	x	x						
C.4.7.2.23	"Rec. X.721 ISO/IEC 10165-2 : 1992": allomorphs	(2 9 3 2 7 50)	x	c1	x	x	x	x						
C.4.7.2.24	"Rec. X.721 ISO/IEC 10165-2 : 1992": nameBinding	(2 9 3 2 7 63)	x	m	x	x	x	x						
C.4.7.2.25	"Rec. X.721 ISO/IEC 10165-2 : 1992": objectClass	(2 9 3 2 7 65)	x	m	x	x	x	x						
C.4.7.2.26	"Rec. X.721 ISO/IEC 10165-2 : 1992": packages	(2 9 3 2 7 66)	x	c2	x	x	x	x						

c0 = m if an instance supports it, else -

c1 = m if an object supports allomorphism, else -

c2 = m if any any registered package (other than this package) has been instantiated, else -

Table C.4.7.5 - Notification Support

Index	Notification Type Label	Value of Notification Type Identifier	Status	Support		Additional Info	Sub-Index	Notification Field Name Label	Value of OID of Attribute associated with Field	Status	Support	Additional Information
				con	firm							
C.4.7.5.1	"Rec. X.721 ISO/IEC 10165-2 : 1992": attributeValueChange	{ 2 9 3 2 10 1 }	m				C.4.7.5.1.1	additionalInformation	{ 2 9 3 2 70 6 }			
							C.4.7.5.1.2	additionalText	{ 2 9 3 2 70 7 }			
							C.4.7.5.1.3	attributeIdentifierList	{ 2 9 3 2 70 8 }			
							C.4.7.5.1.4	attributeValueChangeDefinition	{ 2 9 3 2 70 10 }			
							C.4.7.5.1.5	correlatedNotifications	{ 2 9 3 2 70 12 }			
							C.4.7.5.1.6	notificationIdentifier	{ 2 9 3 2 70 16 }			
							C.4.7.5.1.7	sourceIndicator	{ 2 9 3 2 70 26 }			
C.4.7.5.2	"Rec. X.721 ISO/IEC 10165-2 : 1992": objectCreation	{ 2 9 3 2 10 6 }	m				C.4.7.5.2.1	additionalInformation	{ 2 9 3 2 70 8 }			
							C.4.7.5.2.2	additionalText	{ 2 9 3 2 70 7 }			
							C.4.7.5.2.3	attributeList	{ 2 9 3 2 70 9 }			
							C.4.7.5.2.4	correlatedNotifications	{ 2 9 3 2 70 12 }			
							C.4.7.5.2.5	notificationIdentifier	{ 2 9 3 2 70 16 }			
							C.4.7.5.2.6	sourceIndicator	{ 2 9 3 2 70 26 }			
C.4.7.5.3	"Rec. X.721 ISO/IEC 10165-2 : 1992": objectDeletion	{ 2 9 3 2 10 7 }	m				C.4.7.5.3.1	additionalInformation	{ 2 9 3 2 70 6 }			
							C.4.7.5.3.2	additionalText	{ 2 9 3 2 70 7 }			
							C.4.7.5.3.3	attributeList	{ 2 9 3 2 70 9 }			
							C.4.7.5.3.4	correlatedNotifications	{ 2 9 3 2 70 12 }			
							C.4.7.5.3.5	notificationIdentifier	{ 2 9 3 2 70 16 }			

Index	Notification Type Label	Value of Notification Type Identifier	Status	Support	Add Info	Sub-Index	Notification Field Name Label	Value of OID of Attribute associated with Field	Status	Support	Additional Information
						C.4.7.5.3.6	sourceIndicator	{ 2 9 3 2 7 0 26 }			
C.4.7.5.4	"Rec. X.721 ISO/IEC 10165-2 : 1992": communications Alarm	{ 2 9 3 2 10 2 }	c0			C.4.7.5.4.1	additionalInformation	{ 2 9 3 2 7 0 6 }			
						C.4.7.5.4.2	additionalText	{ 2 9 3 2 7 0 7 }			
						C.4.7.5.4.3	backUpObject	{ 2 9 3 2 7 0 41 }			
						C.4.7.5.4.4	backedUpStatus	{ 2 9 3 2 7 0 11 }			
						C.4.7.5.4.5	correlatedNotifications	{ 2 9 3 2 7 0 12 }			
						C.4.7.5.4.6	notificationIdentifier	{ 2 9 3 2 7 0 16 }			
						C.4.7.5.4.7	perceivedSeverity	{ 2 9 3 2 7 0 17 }			
						C.4.7.5.4.8	probableCause	{ 2 9 3 2 7 0 18 }			
						C.4.7.5.4.9	proposedRepairActions	{ 2 9 3 2 7 0 19 }			
						C.4.7.5.4.10	specificProblems	{ 2 9 3 2 7 0 27 }			
						C.4.7.5.4.11	stateChangeDefinition	{ 2 9 3 2 7 0 28 }			
						C.4.7.5.4.12	thresholdInfo	{ 2 9 3 2 7 0 29 }			
						C.4.7.5.4.13	trendIndicator	{ 2 9 3 2 7 0 30 }			

c0 = m if instance supports it, else -

C.4.8 Transport Connection IVMO MOCS Proforma

Managed Object class template label	Value of Object identifier for class
"OP1 Library Vol. 2":transportConnectionIVMO	{ 1 3 14 2 1 1 1 }

Are all mandatory features of the class supported?

Yes _____ No _____

Table C.4.8.1 - Name Binding Support

Index	Name Binding Template Label	Value of Object Identifier for Name Binding	Superior Object Class Template Label	Status	Support	Status		Support		Additional Information
						create	delete	create	delete	
C.4.8.1.1	transportConnectionIVMO-coTransportProtocolLayerEntity	{ 1 3 14 2 1 3 1 }	"OP1 Library Vol. 1": coTransportProtocolLayerEntity	o		x	x			

Table C.4.8.2 - Attribute Support

Index	Attribute Template Label	Value of Object Identifier for Attribute	Status					Support					Additional Information		
			set by create	get	replace	add	remove	set by default	set by create	get	replace	add		remove	set by default
C.4.8.2.1	"OP1 Library Vol. 1":inactivityTimeout	(1 3 14 2 2 4 17)	m	m	m	x	x	x							
C.4.8.2.2	"OP1 Library Vol. 1":maxPDUSize	(1 3 14 2 2 4 26)	m	m	m	x	x	x							
C.4.8.2.3	transportConnectionIVMOld	(1 3 14 2 1 4 1)	x	m	x	x	x	x							
C.4.8.2.4	"Rec. X.721 ISO/IEC 10165-2 : 1992":allomorphs	(2 9 3 2 7 50)	x	c1	x	x	x	x							
C.4.8.2.5	"Rec. X.721 ISO/IEC 10165-2 : 1992":nameBinding	(2 9 3 2 7 63)	x	m	x	x	x	x							
C.4.8.2.6	"Rec. X.721 ISO/IEC 10165-2 : 1992":objectClass	(2 9 3 2 7 65)	x	m	x	x	x	x							
C.4.8.2.7	"Rec. X.721 ISO/IEC 10165-2 : 1992":packages	(2 9 3 2 7 66)	x	c2	x	x	x	x							

c1 = m if an object supports allomorphism, else -

c2 = m if any any registered package (other than this package) has been instantiated, else -

Table C.4.8.5 - Notification Support

Index	Notification Type Label	Value of Notification Type Identifier	Status	Support		Add Info	Sub-Index	Notification Field Name Label	Value of OID of Attr Type associated with Field	Status	Support	Additional Information
				con	fon							
C.4.8.5.1	"Rec. X.721 ISO/IEC 10165-2 : 1992": attributeValueChange	{ 2 9 3 2 10 1 }	m				C.4.8.5.1	additionalInformation	{ 2 9 3 2 7 9 }	o		
							C.4.8.5.2	additionalText	{ 2 9 3 2 7 7 }	o		
							C.4.8.5.3	attributeIdentifierList	{ 2 9 3 2 7 7 }	o		
							C.4.8.5.4	attributeValueChangeDefinition	{ 2 9 3 2 7 10 }	m		
							C.4.8.5.5	correlatedNotifications	{ 2 9 3 2 7 12 }	o		
							C.4.8.5.6	notificationIdentifier	{ 2 9 3 2 7 16 }	o		
							C.4.8.5.7	sourceIndicator	{ 2 9 3 2 7 26 }	o		
C.4.8.5.2	"Rec. X.721 ISO/IEC 10165-2 : 1992": objectCreation	{ 2 9 3 2 10 6 }	m				C.4.8.5.1	additionalInformation	{ 2 9 3 2 7 6 }	o		
							C.4.8.5.2	additionalText	{ 2 9 3 2 7 7 }	o		
							C.4.8.5.3	attributeList	{ 2 9 3 2 7 9 }	o		
							C.4.8.5.4	correlatedNotifications	{ 2 9 3 2 7 12 }	o		
							C.4.8.5.5	notificationIdentifier	{ 2 9 3 2 7 16 }	o		
							C.4.8.5.6	sourceIndicator	{ 2 9 3 2 7 26 }	o		
C.4.8.5.3	"Rec. X.721 ISO/IEC 10165-2 : 1992": objectDeletion	{ 2 9 3 2 10 7 }	m				C.4.8.5.1	additionalInformation	{ 2 9 3 2 7 6 }	o		
							C.4.8.5.2	additionalText	{ 2 9 3 2 7 7 }	o		
							C.4.8.5.3	attributeList	{ 2 9 3 2 7 9 }	o		
							C.4.8.5.4	correlatedNotifications	{ 2 9 3 2 7 12 }	o		
							C.4.8.5.5	notificationIdentifier	{ 2 9 3 2 7 16 }	o		

Index	Notification Type Label	Value of Notification Type Identifier	Status	Support	Add Info	Sub-Index	Notification Field Name Label	Value of OID of Attr Type associated with Field	Status	Support	Additional Information
				conf		C.4.8.5.3.6	sourceIndicator	(2 9 3 2 7 26)			

C.4.9 Transport Connection Retransmission IVMO MOCS Proforma

Managed Object class template label	Value of Object identifier for class
"OP1 Library Vol. 2":transportConnectionRetransmissionIVMO	(1 3 14 2 1 1 3)

Are all mandatory features of the class supported? Yes _____ No _____

Table C.4.9.1 - Name Binding Support

Index	Name Binding Template Label	Value of Object Identifier for Name Binding	Superior Object Class Template Label	Status	Support	Status		Support		Additional Information
						create	delete	create	delete	
C.4.9.1.1	transportConnectionIVMO-coTransportProtocolLayerEntity	(1 3 14 2 1 3 1)	"OP1 Library Vol. 1":coTransportProtocolLayerEntity			x	x			
C.4.9.1.2	transportConnectionRetransmissionIVMO-coTransportProtocolLayerEntity	(1 3 14 2 1 3 2)	"OP1 Library Vol. 1":coTransportProtocolLayerEntity			x	x			

Table C.4.9.2 - Attribute Support

Index	Attribute Template Label	Value of Object Identifier for Attribute	Status					Support					Additional Information		
			s e t b y c r e a t e	g e t	r e p l a c e	a d d	r e m o v e	s e t b y c r e a t e	s e t b y c r e a t e	g e t	r e p l a c e	a d d		r e m o v e	s e t b y c r e a t e
C.4.9.2.1	"OP1 Library Vol. 1":maxRetransmissions	(1 3 14 2 2 4 27)	m	m	m	x	x	x							
C.4.9.2.8	"OP1 Library Vol. 1":retransmissionTimerInitial Value	(1 3 14 2 2 4 49)	m	m	m	x	x	x							
C.4.9.2.3	"OP1 Library Vol. 1":inactivityTimeout	(1 3 14 2 2 4 17)	m	m	m	x	x	x							
C.4.9.2.4	"OP1 Library Vol. 1":maxPDUSize	(1 3 14 2 2 4 26)	m	m	m	x	x	x							
C.4.9.2.5	transportConnectionIVMOId	(1 3 14 2 1 4 1)	x	m	x	x	x	x							
C.4.9.2.6	"Rec. X.721 ISO/IEC 10165-2 : 1992":allomorphs	(2 9 3 2 7 50)	x	c1	x	x	x	x							
C.4.9.2.7	"Rec. X.721 ISO/IEC 10165-2 : 1992":nameBinding	(2 9 3 2 7 63)	x	m	x	x	x	x							
C.4.9.2.8	"Rec. X.721 ISO/IEC 10165-2 : 1992":objectClass	(2 9 3 2 7 65)	x	m	x	x	x	x							
C.4.9.2.9	"Rec. X.721 ISO/IEC 10165-2 : 1992":packages	(2 9 3 2 7 66)	x	c2	x	x	x	x							

c1 = m if an object supports allomorphism, else -

c2 = m if any any registered package (other than this package) has been instantiated, else -

Table C.4.9.5 - Notification Support

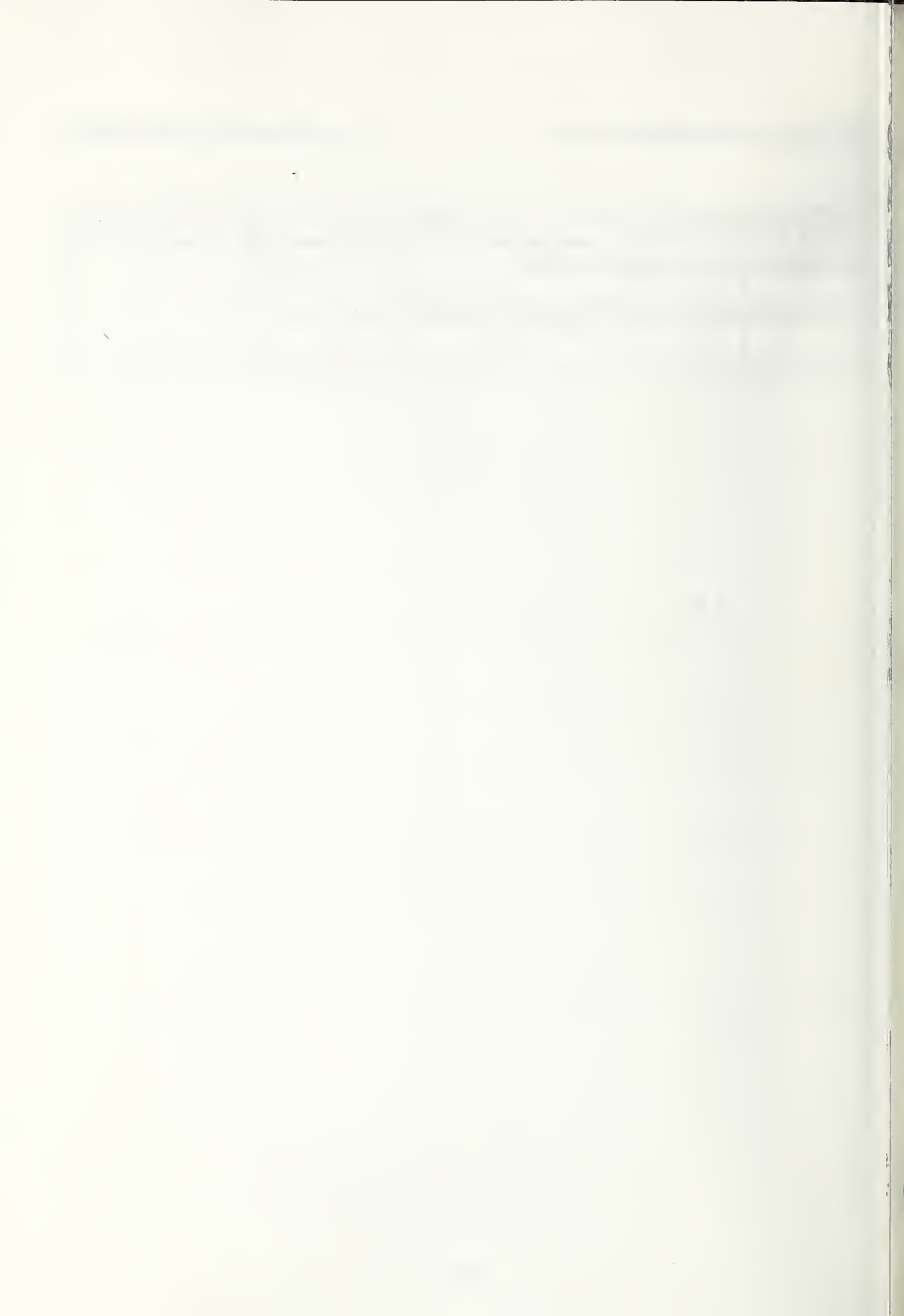
Index	Notification Type Label	Value of Notification Type Identifier	Status	Support		Add Info	Sub-Index	Notification Field Name Label	Value of OID of Attribute associated with Field	Status	Support	Additional Information
				con	non							
C.4.9.5.1	"Rec. X.721 ISO/IEC 10165-2 : 1992": attributeValue Change	(2 9 3 2 10 1)	m				C.4.9.5.1.1	additionalInformation	(2 9 3 2 70 6)			
							C.4.9.5.1.2	additionalText	(2 9 3 2 70 7)			
							C.4.9.5.1.3	attributeIdentifierList	(2 9 3 2 70 8)			
							C.4.9.5.1.4	attributeValueChangeDefinition	(2 9 3 2 70 10)	m		
							C.4.9.5.1.5	correlatedNotifications	(2 9 3 2 70 12)			
							C.4.9.5.1.6	notificationIdentifier	(2 9 3 2 70 16)			
							C.4.9.5.1.7	sourceIndicator	(2 9 3 2 70 26)			
C.4.9.5.2	"Rec. X.721 ISO/IEC 10165-2 : 1992": objectCreation	(2 9 3 2 10 6)	m				C.4.9.5.2.1	additionalInformation	(2 9 3 2 70 6)			
							C.4.9.5.2.2	additionalText	(2 9 3 2 70 7)			
							C.4.9.5.2.3	attributeList	(2 9 3 2 70 9)			
							C.4.9.5.2.4	correlatedNotifications	(2 9 3 2 70 12)			
							C.4.9.5.2.5	notificationIdentifier	(2 9 3 2 70 16)			
							C.4.9.5.2.6	sourceIndicator	(2 9 3 2 70 26)			
C.4.9.5.3	"Rec. X.721 ISO/IEC 10165-2 : 1992": objectDeletion	(2 9 3 2 10 7)	m				C.4.9.5.3.1	additionalInformation	(2 9 3 2 70 6)			
							C.4.9.5.3.2	additionalText	(2 9 3 2 70 7)			
							C.4.9.5.3.3	attributeList	(2 9 3 2 70 9)			
							C.4.9.5.3.4	correlatedNotifications	(2 9 3 2 70 12)			
							C.4.9.5.3.5	notificationIdentifier	(2 9 3 2 70 16)			

Index	Notification Type Label	Value of Notification Type Identifier	Stat us	Support		Add Info	Sub- Index	Notification Field Name Label	Value of OID of Attr Type associated with Field	Stat us	Support	Additional Information
				con f	on n							
							C.4.9 .5.3. 6	sourceIndicato r	(2 9 3 2 7 0 26)			

Annex D (normative)

Management Ensemble Annex

(Refer to the Working Implementation Agreements Document.)



Stable Implementation Agreements for Open Systems Interconnection Protocols: Part 19 - Remote Database Access

**Output from the December 1992 OSE Implementors'
Workshop**

SIG Chair: Peter Eng, IBM Canada

SIG Editor: Joel Berson, Santosh Hasani, Digital Equipment Corp.

Foreword

This part of the Stable Implementation Agreements was prepared by the Remote Database Access Special Interest Group (RDA SIG) of the Open Systems Environment Implementors' Workshop (OIW). See Procedure Manual for Workshop Charter.

Text in this part has been approved by the Plenary of the Workshop. This part replaces the previously existing part on this subject.

Future changes and additions to this version of these Implementation Agreements will be published as change pages. Deleted and replaced text will be shown as ~~strikeout~~. New and replacement text will be shown as shaded.

The text in this part contains a set of Remote Database Access (RDA) Implementation Agreements intended to serve in lieu of an International Standardized Profile (ISP) for RDA. It is the aim of the OIW RDA SIG to pursue alignment of this part with a future RDA ISP. When the ISP is complete, this part will be revised to refer to the ISP, and to only highlight additional practices and North American regional requirements.

Table of Contents

Part 19 - Remote Database Access	1
0 Introduction	1
1 Scope	1
2 Status	1
3 Normative references	2
4 Definitions and abbreviations	2
5 Structure of RDA standards	2
6 SQL specialization	3
6.1 Service parameter limits/agreements	3
6.1.1 Dialogue initialization functional unit	5
6.1.1.1 R-Initialize service	5
6.1.1.1.1 R-Initialize request	6
6.1.1.1.2 R-Initialize result	8
6.1.1.1.3 R-Initialize error	8
6.1.1.2 R-Synchronize APDU	9
6.1.2 Dialogue termination functional unit	9
6.1.2.1 R-Terminate service	9
6.1.2.1.1 R-Terminate request	9
6.1.2.1.2 R-Terminate result	9
6.1.2.1.3 R-Terminate error	10
6.1.3 Transaction management functional unit	10
6.1.3.1 R-BeginTransaction service	10
6.1.3.1.1 R-BeginTransaction request	10
6.1.3.1.2 R-BeginTransaction error	11
6.1.3.2 R-Commit service	11
6.1.3.2.1 R-Commit request	11
6.1.3.2.2 R-Commit result	11
6.1.3.2.3 R-Commit error	12
6.1.3.3 R-Rollback service	12
6.1.3.3.1 R-Rollback request	12
6.1.3.3.2 R-Rollback result	12
6.1.3.3.3 R-Rollback error	12
6.1.4 Cancel functional unit	13
6.1.4.1 R-Cancel service	13
6.1.4.1.1 R-Cancel request	13
6.1.4.1.2 R-Cancel result	14
6.1.4.1.3 R-Cancel error	14
6.1.5 Status functional unit	15

6.1.5.1	R-Status service	15
6.1.5.1.1	R-Status request	15
6.1.5.1.2	R-Status result	16
6.1.5.1.3	R-Status error	16
6.1.6	Resource handling functional unit	17
6.1.6.1	R-Open service	17
6.1.6.1.1	R-Open request	17
6.1.6.1.2	R-Open result	19
6.1.6.1.3	R-Open error	19
6.1.6.2	R-Close service	20
6.1.6.2.1	R-Close request	20
6.1.6.2.2	R-Close result	21
6.1.6.2.3	R-Close error	21
6.1.7	Immediate execution DBL functional unit	22
6.1.7.1	R-ExecuteDBL service	22
6.1.7.1.1	R-ExecuteDBL request	22
6.1.7.1.2	R-ExecuteDBL result	24
6.1.7.1.3	R-ExecuteDBL error	25
6.1.8	Stored Execution DBL Functional Unit	26
6.1.8.1	R-DefineDBL Service	26
6.1.8.1.1	R-DefineDBL request	26
6.1.8.1.2	R-DefineDBL result	27
6.1.8.1.3	R-DefineDBL error	27
6.1.8.2	R-InvokeDBL Service	28
6.1.8.2.1	R-InvokeDBL request	28
6.1.8.2.2	R-InvokeDBL result	29
6.1.8.2.3	R-InvokeDBL error	30
6.1.8.3	R-DropDBL Service	31
6.1.8.3.1	R-DropDBL request	31
6.1.8.3.2	R-DropDBL result	31
6.1.8.3.3	R-DropDBL error	32
6.2	Limits for common parameters	33
6.2.1	SQLDataTypeDescriptor	33
6.2.2	SQLDBLException	35
6.2.3	SQLValue	35
6.3	Other limits and agreements	37
6.3.1	Operation limits and agreements	37
6.3.2	Recommended practices	38
6.4	Rules for Profiles	38
6.4.1	Application contexts	38
6.4.1.1	RDA basic application context	38
6.4.1.2	Profiles	38
6.4.1.2.1	Rationale	38
6.4.1.2.2	Immediate Execution	39
6.4.1.2.3	Stored execution	39
6.4.1.2.4	Status	39
6.4.1.2.5	Cancel	40
6.4.2	RDA TP application context	40

Part 19 - Remote Database Access

December 1992 (Stable)

Annex A (normative)

RDA SIG object identifiers	41
---	-----------

List of Tables

Table 1 - Parameters for R-Initialize request	6
Table 2 - Parameters for R-Initialize result response	8
Table 3 - Parameters for R-Initialize error response	8
Table 4 - Parameters for R-Terminate request	9
Table 5 - Parameters for R-Terminate result response	9
Table 6 - Parameters for R-Terminate error response	10
Table 7 - Parameters for R-BeginTransaction request	10
Table 8 - Parameters for R-BeginTransaction error response	11
Table 9 - Parameters for R-Commit request	11
Table 10 - Parameters for R-Commit result response	11
Table 11 - Parameters for R-Commit error response	12
Table 12 - Parameters for R-Rollback request	12
Table 13 - Parameters for R-Rollback result response	12
Table 14 - Parameters for R-Rollback error response	13
Table 15 - Parameters for R-Cancel request	13
Table 16 - Parameters for R-Cancel result response	14
Table 17 - Parameters for R-Cancel error response	14
Table 18 - Parameters for R-Status request	15
Table 19 - Parameters for R-Status result response	16
Table 20 - Parameters for R-Status error response	17
Table 21 - Parameters for R-Open request	18
Table 22 - Parameters for R-Open result response	19
Table 23 - Parameters for R-Open error response	19
Table 24 - Parameters for R-Close request	21
Table 25 - Parameters for R-Close result response	21
Table 26 - Parameters for R-Close error response	21
Table 27 - Parameters for R-ExecuteDBL request	23
Table 28 - Parameters for R-ExecuteDBL result response	24
Table 29 - Parameters for R-ExecuteDBL error response	25
Table 30 - Parameters for R-DefineDBL request	26
Table 31 - Parameters for R-DefineDBL result	27
Table 32 - Parameters for R-DefineDBL error	27
Table 33 - Parameters for R-InvokeDBL request	28
Table 34 - Parameters for R-InvokeDBL result	29
Table 35 - Parameters for R-InvokeDBL error	30
Table 36 - Parameters for R-DropDBL request	31
Table 37 - Parameters for R-DropDBL result	31
Table 38 - Parameters for R-DropDBL error	32
Table 39 - Parameters for SQLDataTypeDescriptor	33
Table 40 - Parameters for SQLDBLException	35
Table 41 - Parameters for SQLValue	36
Table 42 - Object Identifiers defined by RDA	41

Part 19 - Remote Database Access

0 Introduction

This part defines Implementation Agreements based on ISO Remote Database Access, as defined in ISO/IEC 9579. That specification has two parts. Part 1 defines the RDA Generic Model, Service, and Protocol; part 2 defines the RDA SQL Specialization.

1 Scope

This implementation agreement addresses interaction between an application program and a remote database server. The database server is an open system that provides database storage facilities and supplies database processing services to clients at other open systems.

The RDA service-provider supplies the protocol for RDA client interaction with an RDA server. The RDA client initiates an RDA dialogue and requests RDA operations to be performed by the RDA server on behalf of an application program. The RDA server, located within the RDA database server, provides database services to RDA clients.

More specifically, this document describes implementation agreements in the following areas:

- a) limitations on values of parameters of the RDA protocol;
- b) other restrictions on operations of an RDA client or server;
- c) profiles.

These agreements presently include only the RDA SQL Specialization Basic Application Context.

2 Status

The following clauses were moved to the Stable Implementation Agreements in December 1992 :

- 0 Introduction
- 1 Scope
- 2 Status
- 3 Normative references
- 4 Definitions and abbreviations
- 5 Structure of RDA standards
- 6 SQL specialization

Annex A (normative), RDA SIG object identifiers

3 Normative references

The following documents contain provisions which, through reference in this text, constitute provisions of these Implementation Agreements. At the time of publication, the editions indicated were valid. All documents are subject to revision, and parties to agreements based on these Implementation Agreements are warned against automatically applying any more recent editions of the documents listed below since the nature of references made by the Implementation Agreements to such documents is that they may be specific to a particular edition. Members of IEC and ISO maintain registers of currently valid International Standards, and CCITT maintains published editions of its current recommendations.

- [1] ISO/IEC 9579-1 *Information Processing Systems - Open Systems Interconnection - Remote Database Access - Part 1: Generic Model, Service, and Protocol.*
- [2] ISO/IEC 9579-2 *Information Processing Systems - Open Systems Interconnection - Remote Database Access - Part 2: SQL Specialization.*
- [3] ISO/IEC TR10000-1: 1990(E) *Information Technology - Framework and Taxonomy of International Standardized Profiles - Part 1: Framework.*
- [4] ISO/IEC TR10000-2: 1990(E) *Information Technology - Framework and Taxonomy of International Standardized Profiles - Part 2: Taxonomy.*
- [5] ISO/IEC 8859-1:1987 *Information Processing - 8-bit single-byte coded graphic characters sets - Part 1: Latin alphabet No. 1.*

4 Definitions and abbreviations

Definitions and abbreviations are given in the standards listed in clause 3.

5 Structure of RDA standards

The complete specification of an RDA Service is only obtained by the combination of two standards:

- a) the RDA Generic Model, Service, and Protocol (ISO/IEC 9579-1), which specifies general capabilities of any RDA service; and
- b) an RDA Specialization, which applies to a particular type of database language and augments the RDA Generic standard.

Since RDA Specializations "complete" an RDA specification, these Implementation Agreements are based on RDA Specializations rather than on the RDA Generic standard.

At present there is only one RDA Specialization, the RDA SQL Specialization (ISO/IEC 9579-2).

6 SQL specialization

6.1 Service parameter limits/agreements

This subclause states the limits on the values of RDA parameters for the RDA Basic Application Context.

NOTE - Limits for the RDA TP Application Context will be defined at a later time.

A tabular format is used to describe the usage of each RDA parameter and its limit. Limits vary among implementations. This subclause defines the range of parameter values that developers may safely assume all OIW-compliant systems support.

Each parameter table indicates the following:

- a) the name of the parameter;
- b) whether it is sent in the request or response service primitive under these Implementation Agreements;
- c) whether it is received in the indication or confirm service primitive under these Implementation Agreements;
- d) the limitations on the parameter.

The Parameter column gives the name of the parameter or type, as defined in either the service parameter tables or the ASN.1 for the protocol data units of ISO/IEC 9579-1 and 9579-2.

NOTE - Parameter names begin with a lowercase letter, and type names begin with an uppercase letter.

The req (ind) column states whether the parameter is present in the request (indication) service primitive event.

The rsp (cnf) column states whether the parameter is present in the response (confirm) service primitive event.

The Limitation column gives the limits on the parameter value in addition to any limits imposed by the RDA standard. The maximum value indicated for the limit imposed by these Implementation Agreements is a min-max limit. This means that an OIW-compliant implementation must support minimally at least the min-max value. An implementation can support values beyond the min-max value but it can not expect other implementations to do the same. Hence, an implementation should stay within the min-max limit when it is Interoperating with another implementation.

If a parameter value is outside the range specified in the Limitation column for that parameter, then the behavior is outside the scope of these Implementation Agreements.

Each RDA parameter is listed on a separate line. Some parameters are structures, composed of subparameters. The structure is shown by the bullet (•) symbols in the parameter column. A parameter

name preceded by bullets is a subparameter of the nearest entry above it that has one fewer bullet. In the example below, parameterA and parameterB are subparameters of the structure parameter parameterX, and parameterC is a subparameter of parameterB :

parameterX
•parameterA
•parameterB
••parameterC

The presence of subparameters is always dependent on the presence of the parameter that they appear under (for example, an optional parameter may have subparameters; if the parameter is not supplied, then no subparameters may be supplied).

Under each req, ind, rsp, or cnf column, a code is used to specify the type of usage of each RDA parameter. For the sake of explanation, let x represent the entry under columns req, rsp, ind, and cnf.

If x is M, the parameter is mandatory.

If x is U, the parameter is a user option and may or may not be provided, depending upon the requirements of the user.

If x is C, the parameter is conditional and subject to rules stated in the parameter description in ISO/IEC 9579-1 and 9579-2.

If x is S, the parameter is a mandatory selection from a collection of two or more possible parameters. The parameters that make up this collection are indicated in the parameter table as follows:

- a) each parameter in the collection is specified with the code "S";
- b) the name of each parameter in the collection has the same number of bullets preceding it in the parameter column in the table; and
- c) either
 - 1) each parameter has no bullets preceding it in the table; or
 - 2) each parameter is part of the same structure parameter.

If x is I, the parameter is out of the scope of this profile. The parameter is optional in the base standard but is not used by this profile, or the parameter is not used by the RDA SQL specialization. The parameter may be present. If present, it is ignored or processed according to local procedures; it does not cause an error.

A blank code indicates the parameter is never present.

If x includes (=), the parameter is semantically equivalent to the parameter in the service primitive to its left

in the table.

6.1.1 Dialogue initialization functional unit

6.1.1.1 R-Initialize service

6.1.1.1.1

R-Initialize request

Table 1 - Parameters for R-Initialize request

Parameter	req	ind	Limitation
operationID	M	M(=)	An INTEGER with value greater than 0.
dialogueID	M	M(=)	See subparameter(s) below.
•dialogueIDClient-Invocation		M	See subparameter(s) below.
••aP-title	M	M(=)	No additional limitation.
••aE-qualifier	M	M(=)	No additional limitation.
••aP-invocationID	M	M(=)	No additional limitation.
••aE-invocationID	M	M(=)	No additional limitation.
•dialogueIDSuffix	M	M(=)	An OCTET STRING from 1 through 64 octets long.
identityOfUser	M	M(=)	A VisibleString from 1 through 64 characters long.
userAuthenticationData	I	I(=)	The client may provide an IA5String from 1 through 255 characters long, an OCTET STRING from 1 through 255 octets long, or a BIT STRING from 1 through 2040 bits long.
controlServiceData- Requested (default value: FALSE)	M	M(=)	No additional limitation.
functionalUnitsRequested	M	M(=)	No additional limitation.
sQLInitializeArgument	U	C(=)	See subparameter(s) below.
•sQLConformanceLevel- Default	U	C(=)	An OBJECT IDENTIFIER from 2 through 16 elements.
•userData	I	I(=)	An OCTET STRING from 1 through 255 octets long.

6.1.1.1.2 R-Initialize result

Table 2 - Parameters for R-Initialize result response

Parameter	rsp	cnf	Limitation
operationID	M	M(=)	An INTEGER with value greater than 0.
controlServiceData	C	C(=)	See subparameter(s) below.
•controlServicesAllowed (default value: TRUE)	M	M(=)	No additional limitation.
•controlAuthentication-Data	I	I(=)	The server may provide an IA5String from 1 through 255 characters long, an OCTET STRING from 1 through 255 octets long, or a BIT STRING from 1 through 2040 bits long.
functionalUnitsAllowed	M	M(=)	No additional limitation.
sQLInitializeResult	U	C(=)	See subparameter(s) below.
•userData	I	I(=)	An OCTET STRING from 1 through 255 octets long.

6.1.1.1.3 R-Initialize error

Table 3 - Parameters for R-Initialize error response

Parameter	rsp	cnf	Limitation
operationID	M	M(=)	An INTEGER with value greater than 0.
accessControlViolation	S	S(=)	No additional limitation.
duplicateDialogueID	S	S(=)	No additional limitation.
invalidSequence		S	See subparameter(s) below.
•diagnosticInformation		U	No additional limitation.

Parameter	rsp	cnf	Limitation
operationAborted	S	S(=)	See subparameter(s) below.
•errorType (default value:transient)	M	M(=)	No additional limitation.
•diagnosticinformation	U	C(=)	A VisibleString, from 1 through 254 characters long.
userAuthenticationFailure	S	S(=)	No additional limitation.

6.1.1.2 R-Synchronize APDU

The R-Synchronize-RI APDU has no parameters.

6.1.2 Dialogue termination functional unit

6.1.2.1 R-Terminate service

6.1.2.1.1 R-Terminate request

Table 4 - Parameters for R-Terminate request

Parameter	req	ind	Limitation
operationID	M	M(=)	An INTEGER with value greater than 0.

6.1.2.1.2 R-Terminate result

Table 5 - Parameters for R-Terminate result response

Parameter	rsp	cnf	Limitation
operationID	M	M(=)	An INTEGER with value greater than 0.

6.1.2.1.3 R-Terminate error

Table 6 - Parameters for R-Terminate error response

Parameter	rsp	cnf	Limitation
operationID	M	M(=)	An INTEGER with value greater than 0.
duplicateOperationID	S	S(=)	No additional limitation.
invalidSequence		S	See subparameter(s) below.
•diagnosticInformation		U	No additional limitation.
operationAborted	S	S(=)	See subparameter(s) below.
•errorType(default value: transient)	M	M(=)	No additional limitation.
•diagnosticInformation	U	C(=)	A VisibleString from 1 through 254 characters long.
serviceNotNegotiated	S	S(=)	No additional limitation.

6.1.3 Transaction management functional unit

6.1.3.1 R-BeginTransaction service

6.1.3.1.1 R-BeginTransaction request

Table 7 - Parameters for R-BeginTransaction request

Parameter	req	ind	Limitation
operationID	M	M(=)	An INTEGER with value greater than 0.

6.1.3.1.2 R-BeginTransaction error

Table 8 - Parameters for R-BeginTransaction error response

Parameter	rsp	cnf	Limitation
operationID	M	M(=)	An INTEGER with value greater than 0.
duplicateOperationID	S	S(=)	No additional limitation.
invalidSequence		S	See subparameter(s) below.
•diagnosticInformation		U	No additional limitation.
operationAborted	S	S(=)	See subparameter(s) below.
•errorType(default value: transient)	M	M(=)	No additional limitation.
•diagnosticInformation	U	C(=)	A VisibleString, from 1 through 254 characters long.
serviceNotNegotiated	S	S(=)	No additional limitation.

6.1.3.2 R-Commit service

6.1.3.2.1 R-Commit request

Table 9 - Parameters for R-Commit request

Parameter	req	ind	Limitation
operationID	M	M(=)	An INTEGER with value greater than 0.

6.1.3.2.2 R-Commit result

Table 10 - Parameters for R-Commit result response

Parameter	rsp	cnf	Limitation
operationID	M	M(=)	An INTEGER with value greater than 0.
transactionResult	M	M(=)	No additional limitation.

6.1.3.2.3 R-Commit error**Table 11 - Parameters for R-Commit error response**

Parameter	rsp	cnf	Limitation
operationID	M	M(=)	An INTEGER with value greater than 0.
duplicateOperationID	S	S(=)	No additional limitation.
invalidSequence		S	See subparameter(s) below.
•diagnosticInformation		U	No additional limitation.

6.1.3.3 R-Rollback service**6.1.3.3.1 R-Rollback request****Table 12 - Parameters for R-Rollback request**

Parameter	req	ind	Limitation
operationID	M	M(=)	An INTEGER with value greater than 0.

6.1.3.3.2 R-Rollback result**Table 13 - Parameters for R-Rollback result response**

Parameter	rsp	cnf	Limitation
operationID	M	M(=)	An INTEGER with value greater than 0.

6.1.3.3.3 R-Rollback error

Table 14 - Parameters for R-Rollback error response

Parameter	rsp	cnf	Limitation
operationID	M	M(=)	An INTEGER with value greater than 0.
duplicateOperationID	S	S(=)	No additional limitation.
invalidSequence		S	See subparameter(s) below.
•diagnosticInformation		U	No additional limitation.

6.1.4 Cancel functional unit

6.1.4.1 R-Cancel service

6.1.4.1.1 R-Cancel request

Table 15 - Parameters for R-Cancel request

Parameter	req	Ind	Limitation
operationID	M	M(=)	An INTEGER with value greater than 0.
controlledDialogue	U	C(=)	See subparameter(s) below.
•dialogueID	M	M(=)	See subparameter(s) below.
••dialogueIDClient-Invocation	U	C(=)	See subparameter(s) below.
•••aP-title	M	M(=)	No additional limitation.
•••aE-qualifier	M	M(=)	No additional limitation.
•••aP-invocationID	M	M(=)	No additional limitation.
•••aE-InvocationID	M	M(=)	No additional limitation.
••dialogueIDSuffix	M	M(=)	An OCTET STRING from 1 through 64 octets long.

Parameter	req	ind	Limitation
•controlAuthenticationData	M	M(=)	The client may provide an IA5String from 1 through 255 characters long, an OCTET STRING from 1 through 255 octets long, or a BIT STRING from 1 to 2040 bits long.
listOfOperationID	U	C(=)	This list may contain from 1 through 32 entries of OperationID.
•OperationID	C	C(=)	An INTEGER with value greater than 0.

6.1.4.1.2 R-Cancel result

Table 16 - Parameters for R-Cancel result response

Parameter	rsp	cnf	Limitation
operationID	M	M(=)	An INTEGER with value greater than 0.

6.1.4.1.3 R-Cancel error

Table 17 - Parameters for R-Cancel error response

Parameter	rsp	cnf	Limitation
operationID	M	M(=)	An INTEGER with value greater than 0.
controlAuthenticationFailure	S	S(=)	No additional limitation.
controlServices-NotAllowed	S	S(=)	No additional limitation.
dialogueIDUnknown	S	S(=)	No additional limitation.
duplicateOperationID	S	S(=)	No additional limitation.
invalidSequence		S	See subparameter(s) below.
•diagnosticInformation		U	No additional limitation.
operationAborted	S	S(=)	See subparameter(s) below.

Parameter	rsp	cnf	Limitation
•errorType (default value: transient)	M	M(=)	No additional limitation.
•diagnosticInformation	U	C(=)	A VisibleString, from 1 through 254 characters long.
serviceNotNegotiated	S	S(=)	No additional limitation.

6.1.5 Status functional unit

6.1.5.1 R-Status service

6.1.5.1.1 R-Status request

Table 18 - Parameters for R-Status request

Parameter	req	ind	Limitation
operationID	M	M(=)	An INTEGER with value greater than 0.
controlledDialogue	U	C(=)	See subparameters below.
•dialogueID	M	M(=)	See subparameter(s) below.
••dialogueIDClient-Invocation	U	C(=)	See subparameter(s) below.
•••aP-title	M	M(=)	No additional limitation.
•••aE-qualifier	M	M(=)	No additional limitation.
•••aP-invocationID	M	M(=)	No additional limitation.
•••aE-invocationID	M	M(=)	No additional limitation.
••dialogueIDSuffix	M	M(=)	An OCTET STRING from 1 through 64 octets long.

Parameter	req	ind	Limitation
•controlAuthenticationData	M	M(=)	The client may provide an IA5String from 1 through 255 characters long, an OCTET STRING from 1 through 255 octets long, or a BIT STRING from 1 to 2040 bits long.
listOfOperationID	U	C(=)	This list may contain from 1 through 32 entries of OperationID.
•OperationID	C	C(=)	An INTEGER with value greater than 0.

6.1.5.1.2 R-Status result

Table 19 - Parameters for R-Status result response

Parameter	rsp	cnf	Limitation
operationID	M	M(=)	An INTEGER with value greater than 0.
listOfStatusInformation	C	C(=)	This list may contain from 1 through 32 entries of StatusInformation.
•StatusInformation	U	C(=)	See subparameter(s) below.
••operationID	M	M(=)	An INTEGER with value greater than 0.
••operationStatus	M	M(=)	See choice(s) below.
•••operationIDUnknown	S	S(=)	No additional limitation.
•••awaitingExecution	S	S(=)	No additional limitation.
•••executing	S	S(=)	No additional limitation.
•••finished	S	S(=)	No additional limitation.
•••cancelled	S	S(=)	No additional limitation.
•••aborted	S	S(=)	A VisibleString with value from 1 through 254 characters long.

6.1.5.1.3 R-Status error

Table 20 - Parameters for R-Status error response

Parameter	rsp	cnf	Limitation
operationID	M	M(=)	An INTEGER with value greater than 0.
controlAuthenticationFailure	S	S(=)	No additional limitation.
controlServices-NotAllowed	S	S(=)	No additional limitation.
dialogueIDUnknown	S	S(=)	No additional limitation.
duplicateOperationID	S	S(=)	No additional limitation.
invalidSequence		S	See subparameter(s) below.
•diagnosticInformation		U	No additional limitation.
operationAborted	S	S(=)	See subparameter(s) below.
•errorType (default value: transient)	M	M(=)	No additional limitation.
•diagnosticInformation	U	C(=)	A VisibleString, from 1 through 254 characters long.
serviceNotNegotiated	S	S(=)	No additional limitation.

6.1.6 Resource handling functional unit

6.1.6.1 R-Open service

6.1.6.1.1 R-Open request

Table 21 - Parameters for R-Open request

Parameter	req	ind	Limitation
operationID	M	M(=)	An INTEGER with value greater than 0.
dataResourceHandle	M	M(=)	An INTEGER with value greater than 0.
parentDataResourceHandle	I	I(=)	An INTEGER with value geater than 0.
dataResourceName	U	C(=)	A VisibleString, from 1 through 254 characters long.
sQLAccessControlData	I	I(=)	The client may provide an IA5String from 1 through 255 characters long, an OCTET STRING from 1 through 255 octets long, or a BIT STRING from 1 through 2040 bits long.
sQLUsageMode	U	C(=)	No additional limitation.
sQLOpenArgument	U	C(=)	See subparameter(s) below.
•charSet	U	C(=)	An OBJECT IDENTIFIER from 2 through 16 elements long.
•sQLConformanceLevel	U	C(=)	An OBJECT IDENTIFIER from 2 through 16 elements long.

6.1.6.1.2 R-Open result

Table 22 - Parameters for R-Open result response

Parameter	rsp	cnf	Limitation
operationID	M	M(=)	An INTEGER with value greater than 0.
sQLOpenResult	U	C(=)	See subparameter(s) below.
•charSet	U	C(=)	An OBJECT IDENTIFIER from 2 through 16 elements long.
•charSetNotSupported (default value : FALSE)	U	C(=)	No additional limitation.
•SQLConformanceLevel	C	C(=)	An OBJECT IDENTIFIER from 2 through 16 elements long.

6.1.6.1.3 R-Open error

Table 23 - Parameters for R-Open error response

Parameter	rsp	cnf	Limitation
operationID	M	M(=)	An INTEGER with value greater than 0.
dataResourceAlreadyOpen	S	S(=)	See subparameter(s) below.
•dataResourceHandle	M	M(=)	An INTEGER with value greater than 0.
dataResourceNameNot-Specified	S	S(=)	No additional limitation.
dataResourceNotAvailable	S	S(=)	See subparameters below.
•errorType (default value : transient)	M	M(=)	No additional limitation.
•diagnosticInformation	U	C(=)	A VisibleString, from 1 through 254 characters long.
dataResourceUnknown	S	S(=)	No additional limitation.

Parameter	rsp	cnf	Limitation
duplicateDataResource-Handle	S	S(=)	No additional limitation.
duplicateOperationID	S	S(=)	No additional limitation.
InvalidSequence		S	See subparameter(s) below.
•diagnosticInformation		U	No additional limitation.
operationAborted	S	S(=)	See subparameter(s) below.
•errorType(default value: transient)	M	M(=)	No additional limitation.
•diagnosticInformation	U	C(=)	A VisibleString, from 1 through 254 characters long.
operationCancelled	S	S(=)	No additional limitation.
parentDataResource-Unknown	I	I(=)	No additional limitation.
serviceNotNegotiated	S	S(=)	No additional limitation.
sQLOpenError	S	S(=)	See choice(s) below.
•invalidSQLConformance-Level	S	S(=)	No additional limitation.
•sQLAccessControl-Violation	S	S(=)	No additional limitation.
•sQLDatabaseResource-AlreadyOpen	S	S(=)	No additional limitation.
•rDATransactionOpen	S	S(=)	No additional limitation.

6.1.6.2 R-Close service

6.1.6.2.1 R-Close request

Table 24 - Parameters for R-Close request

Parameter	req	ind	Limitation
operationID	M	M(=)	An INTEGER with value greater than 0.
listOfDataResourceHandle	U	C(=)	This list shall contain one entry of DataResourceHandle.
•DataResourceHandle	U	C(=)	An INTEGER with value greater than 0.

6.1.6.2.2 R-Close result

Table 25 - Parameters for R-Close result response

Parameter	rsp	cnf	Limitation
operationID	M	M(=)	An INTEGER with value greater than 0.
listOfCloseExceptions	U	C(=)	This list shall contain one entry of CloseException.
•CloseException	M	M(=)	See subparameter(s) below.
••dataResourceHandle	M	M(=)	An INTEGER with value greater than 0.
••closeException	M	M(=)	See choice(s) below.
•••dataResourceHandle-Unknown	S	S(=)	No additional limitation.

6.1.6.2.3 R-Close error

Table 26 - Parameters for R-Close error response

Parameter	rsp	cnf	Limitation
operationID	M	M(=)	An INTEGER with value greater than 0.
duplicateOperationID	S	S(=)	No additional limitation.
invalidSequence		S	See subparameter(s) below.
•diagnosticInformation		U	No additional limitation.

operationAborted	S	S(=)	See subparameter(s) below.
•errorType(default value: transient)	M	M(=)	No additional limitation.
•diagnosticInformation	U	C(=)	A VisibleString, from 1 through 254 characters long.
operationCancelled	S	S(=)	No additional limitation.
serviceNotNegotiated	S	S(=)	No additional limitation.
sQLCloseError	S	S(=)	See choice(s) below.
•rDATransactionOpen	S	S(=)	No additional limitation.

6.1.7 Immediate execution DBL functional unit

6.1.7.1 R-ExecuteDBL service

6.1.7.1.1 R-ExecuteDBL request

Table 27 - Parameters for R-ExecuteDBL request

Parameter	req	ind	Limitation
operationID	M	M(=)	An INTEGER with value greater than 0.
dataResourceHandle	U	C(=)	An INTEGER with value greater than 0.
sQLDBLStatement	M	M(=)	See subparameter(s) below.
•statementText	M	M(=)	An OCTET STRING, from 1 through 4000 octets long.
•charSet	U	C(=)	An OBJECT IDENTIFIER, from 2 through 16 elements long.
sQLDBLArgument-Specification	U	C(=)	This parameter may contain 1 through 100 entries of SQLDataTypeDescriptor. See table 39, Parameters for SQLDataTypeDescriptor.
sQLDBLResultSpecification	U	C(=)	This parameter may contain 1 through 100 entries of SQLDataTypeDescriptor. See table 39, Parameters for SQLDataTypeDescriptor.
dBLArguments	U	C(=)	See choice(s) below.
•singleArgument	S	S(=)	See subparameter(s) below.
••repetitionCount(default value: 1)	M	M(=)	An INTEGER with value from 1 through 64.
••sQLDBLArgumentValues	C	C(=)	This parameter may contain from 1 through 100 entries of SQLValue. See table 41, Parameters for SQLValue.
•multipleArgument	S(=)	S(=)	See subparameter(s) below.
••listOfSQLDBLArgument-Values	M	M(=)	This list may contain from 1 through 64 entries of SQLDBLArgumentValues.
•••SQLDBLArgumentValues	C	C(=)	This parameter may contain from 1 through 100 entries of SQLValue. See table 41, Parameters for SQLValue.

6.1.7.1.2 R-ExecuteDBL result

Table 28 - Parameters for R-ExecuteDBL result response

Parameter	rsp	cnf	Limitation
operationID	M	M(=)	An INTEGER with value greater than 0.
sQLDBLResult-Specification	C	C(=)	This parameter may contain 1 through 100 entries of SQLDataTypeDescriptor. See table 39, Parameters for SQLDataTypeDescriptor.
listOfResultValues	U	C(=)	This list may contain 1 through 64 entries of ResultValues.
•ResultValues	M	M(=)	See subparameter(s) below.
••sQLDBLException	M	M(=)	See table 40, Parameters for sQLDBLException.
••sQLDBLResultValues	C	C(=)	This parameter may contain 1 through 100 entries of SQLValue. See table 41, Parameters for SQLValue.

6.1.7.1.3 R-ExecuteDBL error

Table 29 - Parameters for R-ExecuteDBL error response

Parameter	rsp	cnf	Limitation
operationID	M	M(=)	An INTEGER with value greater than 0.
badRepetitionCount	S	S(=)	No additional limitation.
dataResourceHandle-NotSpecified	S	S(=)	No additional limitation.
dataResourceHandle-Unknown	S	S(=)	No additional limitation.
duplicateOperationID	S	S(=)	No additional limitation.
invalidSequence		S	See subparameter(s) below.
•diagnosticInformation		U	No additional limitation.
noDataResourceAvailable	S	S(=)	No additional limitation.
operationAborted	S	S(=)	See subparameter(s) below.
•errorType(default value: transient)	M	M(=)	No additional limitation.
•diagnosticInformation	U	C(=)	A VisibleString, from 1 through 254 characters long.
operationCancelled	S	S(=)	No additional limitation.
serviceNotNegotiated	S	S(=)	No additional limitation.
transactionRolledBack	S	S(=)	No additional limitation.
sQLExecuteDBLError	S	S(=)	See choice(s) below.
•sQLUsageModeViolation	S	S(=)	No additional limitation.
•sQLDBLTransaction-StatementNotAllowed	S	S(=)	No additional limitation.

Parameter	rsp	cnf	Limitation
•sQLDBLArgumentCount-Mismatch	S	S(=)	No additional limitation.
•sQLDBLArgumentType-Mismatch	S	S(=)	No additional limitation.
•sQLDBLNoCharSet	S	S(=)	No additional limitation.
•hostIdentifierError	S	S(=)	No additional limitation.
•rDATransactionNotOpen	S	S(=)	No additional limitation.

6.1.8 Stored Execution DBL Functional Unit

6.1.8.1 R-DefinedBL Service

6.1.8.1.1 R-DefinedBL request

Table 30 - Parameters for R-DefinedBL request

Parameter	req	ind	Limitation
operationID	M	M(=)	An INTEGER with value greater than 0.
commandHandle	M	M(=)	An INTEGER with value greater than 0.
dataResourceHandle	U	C(=)	An INTEGER with value greater than 0.
sQLDBLStatement	M	M(=)	See subparameter(s) below.
•statementText	M	M(=)	An OCTET STRING from 1 through 4000 octets long.
•charSet	U	C(=)	An OBJECT IDENTIFIER from 2 through 16 elements long.

sQLDBLArgument-Specification	U	C(=)	This parameter may contain 1 through 100 entries of SQLDataTypeDescriptor. See table 39, Parameters for SQLDataTypeDescriptor.
sQLDBLResult-Specification	U	C(=)	This parameter may contain 1 through 100 entries of SQLDataTypeDescriptor. See table 39, Parameters for SQLDataTypeDescriptor.

6.1.8.1.2 R-DefinedBL result

Table 31 - Parameters for R-DefinedBL result

Parameter	res	cnf	Limitation
operationID	M	M(=)	An INTEGER with value greater than 0.
sQLDBLResult-Specification	C	C(=)	This parameter may contain 1 through 100 entries of SQLDataTypeDescriptor. See table 39, Parameters for SQLDataTypeDescriptor.
sQLDBLException	C	C(=)	See table 40, Parameters for sQLDBLException.

6.1.8.1.3 R-DefinedBL error

Table 32 - Parameters for R-DefinedBL error

Parameter	rsp	cnf	Limitation
operationID	M	M(=)	An INTEGER with value greater than 0.
dataResourceHandle-NotSpecified	S	S(=)	No additional limitation.
dataResourceHandle-Unknown	S	S(=)	No additional limitation.
duplicateCommandHandle	S	S(=)	No additional limitation.

Parameter	rsp	cnf	Limitation
duplicateOperationID	S	S(=)	No additional limitation.
invalidSequence		S	See subparameter(s) below.
•diagnosticInformation		U	No additional limitation.
noDataResourceAvailable	S	S(=)	No additional limitation.
operationAborted	S	S(=)	See subparameter(s) below.
•errorType(default value: transient)	M	M(=)	No additional limitation.
•diagnosticInformation	U	C(=)	A VisbleString, from 1 through 254 characters long.
operationCancelled	S	S(=)	No additional limitation.
serviceNotNegotiated	S	S(=)	No additional limitation.
sQLDefineDBLError	S	S(=)	See choice(s) below.
•sQLDBLNoCharSet	S	S(=)	No additional limitation.
•sQLDBLTransaction- StatementNotAllowed	S	S(=)	No additional limitation.
•sQLUsageModeViolation	S	S(=)	No additional limitation.
•hostIdentifierError	S	S(=)	No additional limitation.

6.1.8.2 R-InvokeDBL Service

6.1.8.2.1 R-InvokeDBL request

Table 33 - Parameters for R-InvokeDBL request

Parameter	req	ind	Limitation
-----------	-----	-----	------------

operationID	M	M(=)	An INTEGER with value greater than 0.
commandHandle	M	M(=)	An INTEGER with value greater than 0.
dbLArguments	U	C(=)	See choice(s) below.
•singleArgument	S	S(=)	See subparameter(s) below.
••repetitionCount (Default Value: 1)	M	M(=)	An INTEGER with value from 1 through 64.
••sQLDBLArgumentValues	C	C(=)	This parameter may contain 1 through 100 entries of SQLValue. See table 41, Parameters for SQLValue.
•multipleArgument	S	S(=)	See subparameter(s) below.
••listOfSQLDBLArgument- Values	M	M(=)	This list may contain 1 through 64 entries of SQLDBLArgumentValues.
•••SQLDBLArgument- Values	C	C(=)	This parameter may contain 1 through 100 entries of SQLValue. See table 41, Parameters for SQLValue.

6.1.8.2.2 R-InvokeDBL result

Table 34 - Parameters for R-InvokeDBL result

Parameter	res	cnf	Limitation
operationID	M	M(=)	An INTEGER with value greater than 0.
sQLDBLResult- Specification	C	C(=)	This parameter may contain 1 through 100 entries of SQLDataTypedescriptor. See table 39, Parameters for SQLDataTypeDescriptor.
lIstOfResultValues	M	M(=)	This list may contain 1 through 64 entries of ResultValues.
•ResultValues	M	M(=)	See subparameter(s) below.

••SQLDBLException	M	M(=)	See table 40, Parameters for SQLDBLException.
••SQLDBLResultValues	C	C(=)	This parameter may contain 1 through 100 entries of SQLValue. See table 41, Parameters for SQLValue.

6.1.8.2.3 R-InvokeDBL error

Table 35 - Parameters for R-InvokeDBL error

Parameter	rsp	cnf	Limitation
operationID	M	M(=)	An INTEGER with value greater than 0.
badRepetitionCount	S	S(=)	No additional limitation.
commandHandleUnknown	S	S(=)	No additional limitation.
duplicateOperationID	S	S(=)	No additional limitation.
invalidSequence		S	See subparameter(s) below.
•diagnosticInformation		U	No additional limitation.
operationAborted	S	S(=)	See subparameter(s) below.
•errorType(default value: transient)	M	M(=)	No additional limitation.
•diagnosticInformation	U	C(=)	A VisibleString, from 1 through 254 characters long.
operationCancelled	S	S(=)	No additional limitation.
serviceNotNegotiated	S	S(=)	No additional limitation.
transactionRolledBack	S	S(=)	No additional limitation.
sQLInvokeDBLError	S	S(=)	See choice(s) below.

Parameter	rsp	cnf	Limitation
•sQLUsageModeViolation	S	S(=)	No additional limitation.
•sQLDBLArgumentType-Mismatch	S	S(=)	No additional limitation.
•sQLDBLArgumentCount-Mismatch	S	S(=)	No additional limitation.
•rDATransactionNotOpen	S	S(=)	No additional limitation.

6.1.8.3 R-DropDBL Service

6.1.8.3.1 R-DropDBL request

Table 36 - Parameters for R-DropDBL request

Parameter	req	ind	Limitation
operationID	M	M(=)	An INTEGER with value greater than 0.
listOfCommandHandle	U	C(=)	This list may contain from 1 through 32 entries of CommandHandle.
•CommandHandle	C	C(=)	An INTEGER with value greater than 0.

6.1.8.3.2 R-DropDBL result

Table 37 - Parameters for R-DropDBL result

Parameter	res	cnf	Limitation
operationID	M	M(=)	An INTEGER with value greater than 0.
listOfDropDBLExceptions	U	C(=)	This list may contain 1 through 32 entries of DropDBLException.
•DropDBLException	M	M(=)	See subparameter(s) below.

••commandHandle	M	M(=)	An INTEGER with value greater than 0.
••dropDBLException	M	M(=)	See subparameter(s) below.
•••commandHandle-Unknown	S	S(=)	No additional limitation.

6.1.8.3.3 R-DropDBL error

Table 38 - Parameters for R-DropDBL error

Parameter	rsp	cnf	Limitation
operationID	M	M(=)	An INTEGER with value greater than 0.
duplicateOperationID	S	S(=)	No additional limitation.
invalidSequence		S	See subparameter(s) below.
•diagnosticInformation		U	No additional limitation.
operationAborted	S	S(=)	See subparameter(s) below.
•errorType(default value: transient)	M	M(=)	No additional limitation.
•diagnosticInformation	U	C(=)	A VisibleString, from 1 through 254 characters long.
operationCancelled	S	S(=)	No additional limitation.
serviceNotNegotiated	S	S(=)	No additional limitation.

6.2 Limits for common parameters

This clause describes the parameters of the common RDA Types, those shared by more than one RDA Service.

Table 39 describes the parameters for `SQLDataTypeDescriptor`.

Table 40 describes the parameters for `SQLDBLException`.

Table 41 describes the parameters for `SQLValue`. The range of values for each type of data item in this table is implied by the values of the parameters of the corresponding type of `SQLDataTypeDescriptor` in Table 39.

6.2.1 `SQLDataTypeDescriptor`

Table 39 - Parameters for `SQLDataTypeDescriptor`

Parameter	req	ind	rsp	cnf	Limitation
nullable (default value: true)			M	M(=)	No additional limitation.
colName			M	M(=)	A <code>VisibleString</code> , from 1 through 18 characters long.
typeDescriptor	M	M(=)	M	M(=)	See choice(s) below.
•characterType	S	S(=)	S	S(=)	See subparameter(s) below.
••charSet	U	C(=)	U	C(=)	An <code>OBJECT IDENTIFIER</code> , from 2 through 16 elements long.
••length	M	M(=)	M	M(=)	An <code>INTEGER</code> with value from 1 through 240. It represents the maximum number of characters in the data value. Trailing (padded) spaces are included in the length, but a trailing null byte is not.
••fixedLength-Encoding	M	M(=)	M	M(=)	No additional limitation.
•numericType	S	S(=)	S	S(=)	See subparameter(s) below.

Parameter	req	Ind	rsp	cnf	Limitation
••precision	M	M(=)	M	M(=)	This parameter shall contain a value within the range of 1 through 15.
••scale	M	M(=)	M	M(=)	This parameter shall contain a value within the range of 0 through the value of precision.
•decimalType	S	S(=)	S	S(=)	See subparameter(s) below.
••precision	M	M(=)	M	M(=)	This parameter shall contain a value within the range of 1 through 15.
••scale	M	M(=)	M	M(=)	This parameter shall contain a value within the range of 0 through the value of precision.
•integerType	S	S(=)	S	S(=)	See subparameter(s) below.
••precision	M	M(=)	M	M(=)	If precisionBase is decimal, precision must be in the range 1 through 9. If precisionBase is binary, precision must be in the range 1 through 31.
••precisionBase	M	M(=)	M	M(=)	No additional limitation.
•smallIntType	S	S(=)	S	S(=)	See subparameter(s) below.
••precision	M	M(=)	M	M(=)	If precisionBase is decimal, precision must be in the range 1 through 4. If precisionBase is binary, precision must be in the range 1 through 15.
••precisionBase	M	M(=)	M	M(=)	No additional limitation.
•floatType	S	S(=)	S	S(=)	See subparameter(s) below.

Parameter	req	ind	rsp	cnf	Limitation
••mantissa-Precision	M	M(=)	M	M(=)	An INTEGER with value from 1 through 20.
••maxExponent	M	M(=)	M	M(=)	An INTEGER with value from 0 through 38.
•realType	S	S(=)	S	S(=)	See subparameter(s) below.
••mantissa-Precision	M	M(=)	M	M(=)	An INTEGER with value from 1 through 20.
••maxExponent	M	M(=)	M	M(=)	An INTEGER with value from 0 through 38.
•double-PrecisionType	S	S(=)	S	S(=)	See subparameter(s) below.
••mantissa-Precision	M	M(=)	M	M(=)	An INTEGER with value from 1 through 30.
••maxExponent	M	M(=)	M	M(=)	An INTEGER with value from 0 through 38.

6.2.2 SQLDBException

Table 40 - Parameters for SQLDBException

Parameter	rsp	cnf	Limitation
sqlSTATE	C	C(=)	No additional limitation.
sqlCODE	C	C(=)	No additional limitation.
sqlErrorText	U	C(=)	A VisibleString from 1 through 254 characters long.

6.2.3 SQLValue

Table 41 - Parameters for SQLValue

Parameter	req	ind	rsp	cnf	Limitation
dataItem	U	C(=)	U	C(=)	See choice(s) below.
•characterItem	S	S(=)	S	S(=)	An OCTET STRING from 1 through 240 characters.
•numericItem	S	S(=)	S	S(=)	An INTEGER with absolute value less than 10^{15} . Note that the maximum requires a 7-octet INTEGER value.
•decimalItem	S	S(=)	S	S(=)	An INTEGER with absolute value less than 10^{15} . Note that the maximum requires a 7-octet INTEGER value.
•integerItem	S	S(=)	S	S(=)	An INTEGER with absolute value less than 10^9 , if precisionBase is decimal. An INTEGER with absolute value less than 2^{31} , if precisionBase is binary.
•smallIntItem	S	S(=)	S	S(=)	An INTEGER with absolute value less than 10^4 , if precisionBase is decimal. An INTEGER with absolute value less than 2^{15} , if precisionBase is binary.
•floatItem	S	S(=)	S	S(=)	A REAL with the value 0 or absolute value in the range 10^{-38} through 10^{38} inclusive.
•realItem	S	S(=)	S	S(=)	A REAL with the value 0 or absolute value in the range 10^{-38} through 10^{38} inclusive.

Parameter	req	ind	rsp	cnf	Limitation
•double-PrecisionItem	S	S(=)	S	S(=)	A REAL with the value 0 or absolute value in the range 10**-38 through 10**38 inclusive.
Indicator	U	C(=)	U	C(=)	No additional limitation.

6.3 Other limits and agreements

6.3.1 Operation limits and agreements

Operation limits and agreements follow:

a) OIW-compliant RDA Implementations shall conform to the following agreements stated in part 5, Upper layers:

- 1) the specific ASE requirements stated in clause 13.8, Remote Database Access; and
- 2) all other agreements that pertain to the Association Control Service Element, Presentation, and Session, including those for ASN.1 encoding;

b) OIW-compliant RDA implementations shall be able to process at a minimum RDA Application Protocol Data Units of 30,000 octets. It is recommended, however, that Presentation user data not be restricted in size;

c) The maximum number of outstanding RDA operations is 32;

d) If an RDA server receives an abort event after an R-BeginTransaction indication and before an R-Commit indication, then it should roll back the current transaction;

e) As a minimum, the character set ISO 8859-1 shall be supported. The OIW-defined object identifier for this character set is :

```
{ iso (1) identified-organization (3) oiw (14)
  rda-sig (9) character-sets (1) oiw-latin-1 (1) abstract-syntax (1) }
```

NOTE - This object identifier will be deprecated at such time that ISO defines an object identifier for ISO 8859-1.

6.3.2 Recommended practices

An implementation should document any limitation on the number of schema statements that the server permits within a single transaction. For maximum interoperability, a client should never mix schema statements and data statements within a transaction and should restrict the number of schema statements within a transaction to one.

6.4 Rules for Profiles

An implementation conformant to a profile shall implement all the functional units of that profile; it may additionally implement other functional units without being nonconformant.

If a functional unit is included in a profile, all the services of that functional unit, as specified in the ISO/IEC 9579-1 and 9579-2, shall be included.

An implementation conforming to a given profile may accept a dialogue whose functional units conform to a different profile without being nonconformant.

6.4.1 Application contexts

This subclause specifies agreements that apply to individual application contexts.

6.4.1.1 RDA basic application context

This subclause specifies agreements that apply to the RDA Basic Application Context.

6.4.1.2 Profiles

This subclause specifies which functional units combine to form each profile.

6.4.1.2.1 Rationale

The minimum requirement is to be able to execute SQL Statements. Therefore all profiles include the "Immediate Execution DBL" functional unit, and one profile (Immediate Execution) includes just this minimum capability.

Additional capabilities may be required for defining and invoking DBL statements. Therefore the "Stored Execution DBL" functional unit is required only in a separate profile (Stored Execution) defined specifically for this capability.

For the RDA Control Services, executing control services on the current dialogue requires different and probably simpler capabilities than executing control services on other dialogues. Therefore additional profiles are defined for controlling other dialogues.

The R-Status Service implies only inquiry about the controlled dialogue, while the R-Cancel Service requires the ability to modify the controlled dialogue. Since different authorization or access control permissions may be required, it is useful to separate R-Status from R-Cancel. Therefore two separate profiles (Status and Cancel) are defined for controlled dialogue capabilities.

Since an implementation may include functional units and services beyond those required, only this minimum set of four profiles is defined.

6.4.1.2.2 Immediate Execution

This profile requires support of immediate execution of SQL Statements. It requires the following functional units:

- a) dialogue initialization;
- b) dialogue termination;
- c) transaction management;
- d) resource handling; and
- e) Immediate execution DBL.

6.4.1.2.3 Stored execution

This profile requires support of execution of stored SQL Statements. It requires the following functional units:

- a) dialogue initialization;
- b) dialogue termination;
- c) transaction management;
- d) resource handling;
- e) Immediate execution DBL; and
- f) stored execution DBL.

6.4.1.2.4 Status

This profile requires support of the R-Status service on other dialogues. It requires the following functional units:

- a) dialogue initialization, specifically including the following parameters:

- 1) controlServiceDataRequested; and
- 2) controlServiceData;
- b) dialogue termination;
- c) transaction management;
- d) status, specifically including the following parameters:
 - 1) controlledDialogue;
- e) resource handling; and
- f) immediate execution DBL.

6.4.1.2.5 Cancel

This profile requires support of the R-Cancel service on other dialogues. It requires the following functional units:

- a) dialogue initialization, specifically including the following parameters:
 - 1) controlServiceDataRequested; and
 - 2) controlServiceData;
- b) dialogue termination;
- c) transaction management;
- d) cancel, specifically including the following parameters:
 - 1) controlledDialogue;
- e) resource handling; and
- f) immediate execution DBL.

6.4.2 RDA TP application context

No text.

Annex A (normative)

RDA SIG object identifiers

Table 42 lists the object identifiers defined by this part of the Implementation Agreements.

Table 42 - Object Identifiers defined by RDA

Object Identifier	Reference
iso (1) identified-organization (3) oiw (14) rda-sig (9) character-sets (1) oiw-latin-1 (1) abstract-syntax (1)	6.3.1

Stable Implementation Agreements for Open Systems Interconnection Protocols: Part 20 - Manufacturing Message Specification (MMS)

**Output from the December 1992 Open Systems
Environment Implementors' Workshop (OIW)**

**SIG Chair: John Baier, Allen-Bradley
SIG Editor: Neal Laurance, Ford**

Foreword

This part of the Stable Implementation Agreements was prepared by the Manufacturing Message Specification (MMS) Special Interest Group (MMSSIG) of the Open Systems Environment Implementors' Workshop (OIW). See Procedures Manual for Workshop charter.

Text in this part has been approved by the Plenary of the above-mentioned Workshop. No significant technical change has occurred in this part since it was previously presented.

Future changes and additions to this version of these implementor Agreements will be published as change pages. Deleted and replaced text will be shown as ~~strikeout~~. New and replacement text will be shown as shaded.

Table of Contents

Part 20 - Manufacturing Message Specification (MMS)	1
0 Introduction	1
1 Scope	1
2 Field of Application	1
2.1 General	1
2.2 Phase 1 agreements	1
3 Normative References	2
4 Definitions	3
5 Corrigenda and addenda	3
6 Status	3
7 General agreements	3
7.1 Max supported PDU size	3
7.2 FileName	4
8 Service-specific agreements	4
8.1 Environment and general management	4
8.1.1 Initiate	4
8.1.1.1 Negotiation of MMS abstract syntaxes	4
8.1.1.2 Max serv outstanding	4
8.1.1.3 Local detail calling	5
8.1.1.4 Local detail called	5
8.2 VMD support	5
8.3 Domain management	6
8.3.1 List of capabilities	6
8.3.2 Initiate Download Sequence service	6
8.3.3 Download Segment service	6
8.3.4 Terminate Download Sequence service	6
8.3.5 Initiate Upload Sequence service	6
8.3.6 Upload Segment service	6
8.3.7 Get Domain Attributes service	7
8.3.8 Get Capability List service	7
8.4 Program Invocation management	7
8.4.1 Start service	7
8.4.2 Stop service	7
8.4.3 Resume service	7
8.4.4 Reset service	7
8.5 Variable access	7

PART 20 - MMS

December 1992 (Stable)

8.5.1	Scattered access	7
8.5.2	Floating point	8
8.6	Semaphore management	8
8.7	Operator communication	8
8.8	Event management	8
8.9	Journal management	8

Annex A (normative)

Backwards compatibility agreements		9
A.1	Introduction	9
A.2	Backwards compatibility agreements for calling MMS Implementations	10
A.3	Backwards compatibility agreements for called MMS Implementations	10
A.4	General backwards compatibility agreements	11
A.4.1	VMD logical status	11

Annex B (normative)

DIS 9506 Modifications Required for Backwards Compatibility		12
B.1	Introduction	12
B.2	References	12
B.3	General	12
B.3.1	Implementation base	12
B.3.2	Rules of extensibility	12
B.4	Modifications to the protocol definitions	12
B.4.1	Page 39, Section 7.5.2 of DIS 9506-2	12
B.4.2	Page 49, Section 7.6.4, DIS 9506-2	13
B.4.3	Page 95, Section 12.2.1 of DIS 9506-2	13
B.4.4	Page 96, Section 12.3.1 of DIS 9506-2	13
B.4.5	Page 98, Section 12.4.2 of DIS 9506-2	14
B.4.6	Page 138, Section 15.14 of DIS 9506-2	14
B.4.7	Page 166, Section 17.10 of DIS 9506-2	14
B.5	Behavioral requirements	15
B.5.1	Filenames	15
B.5.2	Identify service	15
B.5.3	Initiate service	15
B.5.3.1	Minimum segment size	15
B.5.3.2	Maximum segment size	15
B.5.4	Abstract syntax name	16
B.5.5	Application context name	16
B.5.6	Minor version number	16
B.6	Parameter CBB subset	16
B.7	Service subset	17

List of Tables

Table 1 - Phase 1 Services	2
Table 2 - MMS Service Subset	17

Part 20 - Manufacturing Message Specification (MMS)

0 Introduction

This section defines Implementors Agreements based on Manufacturing Message Specification (MMS), as defined in ISO/IEC 9506. ISO/IEC 9506 has two parts. Part 1 defines the Virtual Manufacturing Device (VMD), its subordinate abstract objects, and the services on these objects. Part 2 defines the protocol. Future parts may define companion standards.

MMS, as described in ISO/IEC 9506, is based on the following ISO documents: ACSE Service and Protocol (ISO 8649, ISO 8650), Presentation Service and Protocol (ISO 8822, ISO 8823), ASN.1 Abstract Syntax Notation and Basic Encoding Rules (ISO 8824, ISO 8825), and Session Service and Protocol (ISO 8326, ISO 8327). These services and protocols are defined architecturally in the OSI Reference Model (ISO 7498). These agreements provide detailed guidance for the Implementor, and eliminate ambiguities in interpretations.

An A-Profile based on MMS and these agreements can be used over any T-Profile (see ISO TR 10000) specifying the OSI connection-mode transport service, or the transport profiles used in support of MAP (Manufacturing Automation Protocol), TOP (Technical and Office Protocols), or US GOSIP.

1 Scope

2 Field of Application

2.1 General

Work on implementation agreements will proceed in phases based upon grouping of services and/or contexts. Implementations are not constrained from implementing services or contexts not addressed by the current set of stable agreements. Future phases of work may affect such implementations.

2.2 Phase 1 agreements

These agreements will be based on a subset of MMS services and protocol listed in table 1 and defined in ISO/IEC 9506-1 and ISO/IEC 9506-2.

Table 1 - Phase 1 Services

Initiate
Conclude
Reject
Abort
Status
GetNameList
Identify
UnsolicitedStatus
GetCapabilityList
InitiateDownloadSequence
DownloadSegment
TerminateDownloadSequence
InitiateUploadSequence
UploadSegment
TerminateUploadSequence
DeleteDomain
GetDomainAttributes
Read
Write
InformationReport
GetVariableAccessAttributes
Input
Output
CreateProgramInvocation
DeleteProgramInvocation
Start
Stop
Resume
Reset
Kill
GetProgramInvocationAttributes

3 Normative References

- [1] ISO/IEC 9506-1: 1990 - *Industrial automation systems - Manufacturing Message Specification Part 1: Service definition*
- [2] ISO/IEC 9506-2: 1990 - *Industrial automation systems - Manufacturing Message Specification Part 2: Protocol specification*

4 Definitions

The definitions given in ISO/IEC 9506-1 are applicable to this document.

In addition the following definitions are used in this document:

MMS Implementation a term used to describe a system which conforms to ISO/IEC 9506, acting either as client or server, when it is unnecessary to distinguish between the MMS-user and the MMS-provider.

MMSpdu Any valid value of the MMSpdu abstract data type as defined in Clause 7 of ISO/IEC 9506-2, except for the initiate-RequestPDU, initiate-ResponsePDU, and initiate-ErrorPDU choices, encoded in the negotiated transfer syntax.

5 Corrigenda and addenda

(Refer to Working Agreements.)

6 Status

(Refer to Working Agreements.)

7 General agreements

7.1 Max supported PDU size

The max_mms_pdu_size is defined as the maximum number of octets in an MMSpdu encoded using the negotiated transfer syntax. This size shall apply to all MMSpdu's with the exception of the initiate-Request PDU, initiate-Response PDU, and initiate-Error PDU. The max_mms_pdu_size shall be negotiated during connection initiation using the Local Detail Calling and Local Detail Called parameters of the MMS initiate service.

The negotiated max_mms_pdu_size shall be applied as follows:

- a) Any received MMSpdu whose length is less than or equal to the negotiated max_mms_pdu_size shall be properly parsed and processed.
- b) An MMS Implementation should not send an MMSpdu whose size exceeds the negotiated max_mms_pdu_size. If an MMS implementation sends an MMSpdu that exceeds the negotiated max_mms_pdu_size, then it shall be prepared to receive a reject pdu. Should an MMS Implementation receive an MMSpdu that exceeds the negotiated max_mms_pdu_size, it shall either reject the MMSpdu or accept the MMSpdu as if no size violation had occurred and perform the expected processing.
- c) If an MMS implementation is unable to send a service response because the response would exceed the max_mms_pdu_size, then it shall return a Service response (-) with an error class of SERVICE and an error code of OTHER.
- d) When rejecting an MMSpdu because it exceeds the negotiated max_mms_pdu_size, an MMS Implementation shall use a Reject PDU Type of PDU-ERROR and a Reject Code of INVALID-PDU in the resulting reject pdu.

7.2 FileName

Restrictions for the use of the type FileName in the MMS Abstract Syntax are specified in section 9.1 of part 9 of these agreements.

8 Service-specific agreements

8.1 Environment and general management

8.1.1 Initiate

8.1.1.1 Negotiation of MMS abstract syntaxes

On the A-ASSOCIATE response, the MMS responder shall not accept more than one presentation context derived from an MMS abstract syntax. For this agreement, the term "MMS abstract syntax" shall represent an abstract syntax from the set containing the abstract syntax defined in clause 19 of ISO/IEC 9506-2 and abstract syntaxes defined by MMS companion standards.

NOTE - There are technical problems with describing operation in multiple MMS abstract syntaxes over a single association. These problems have been identified as of 9/90, and form the basis of the prior agreement.

8.1.1.2 Max serv outstanding

An MMS implementation which intends to conform only with the Client Conformance Requirements for Requester CBBs shall:

- a) propose one or greater for the value of the Proposed Max Serv Outstanding Called parameter in the Initiate service when initiating the application association (calling);
- b) offer one or greater for the value of the Negotiated Max Serv Outstanding Calling parameter in the Initiate service when receiving the application association initiation (called).

An MMS implementation which intends to conform to one or more Server Conformance Requirements for Responder CBBs shall:

- a) propose one or greater for the value of the Proposed Max Serv Outstanding Calling parameter in the initiate service when initiating the application association (calling);
- b) offer one or greater for the value of the Negotiated Max Serv Outstanding Called parameter in the Initiate service when receiving the application association initiation (called).

8.1.1.3 Local detail calling

The local detail calling parameter in the initiate request primitive shall specify the `max_mms_pdu_size` guaranteed to be supported by the calling MMS implementation. If the local detail calling parameter is absent from the request primitive, then the calling MMS implementation guarantees support for an unlimited `max_mms_pdu_size`.

If present in the request or Indication primitives, the local detail calling parameter shall not be less than 64; however, it is recommended that at least 512 octets be supported.

8.1.1.4 Local detail called

The local detail called parameter in the initiate response shall specify the negotiated `max_mms_pdu_size` for the application association.

If the local detail calling parameter was omitted in the indication primitive, then the local detail called parameter:

- a) may be omitted from the response, indicating that the calling MMS Implementation and the called MMS implementation are prepared to support an unbounded `max_mms_pdu_size`;
- b) may be specified in the response, indicating a requirement to support the specified value for `max_mms_pdu_size`.

If the local detail calling parameter was included in the request, then this parameter shall be present in the response and its value shall be less than or equal to the value of the local detail calling parameter of the request.

If present in the response, the local detail called parameter shall not be less than 64; however, it is recommended that at least 512 octets be supported.

8.2 VMD support

8.3 Domain management

8.3.1 List of capabilities

Only one capability shall be described in each Visible String of the SEQUENCE OF.

8.3.2 Initiate Download Sequence service

The List of Capability parameter shall follow the limitations of 8.3.1.

The syntax and semantics of the capabilities shall be defined by the Server in the PICS. Any deviation from the defined syntax and semantics shall be grounds for the Server to return a service error with Error Class = RESOURCE and Error Code = CAPABILITY-UNKNOWN.

8.3.3 Download Segment service

A client that receives a Download Segment indication after issuing a Download Segment Result(+) with the MoreFollows parameter equal to FALSE or after issuing a Download Segment Result(-) shall issue either a service error, specifying an Error Class = SERVICE and an Error Code = PRIMITIVES-OUT-OF-SEQUENCE, or an Abort request.

8.3.4 Terminate Download Sequence service

If a client receives a Terminate Download Sequence indication in which the Discard parameter is absent and the client has not issued a Download Segment response with the More Follows parameter = FALSE for that Domain, it shall behave as if it had received a Terminate Download Sequence indication with the Discard parameter present with error class = VMD-STATE and error code = DOMAIN-TRANSFER-PROBLEM. It is then up to the client application to determine the true state of the Domain and take any recovery action.

8.3.5 Initiate Upload Sequence service

The List of Capability parameter shall follow the limitations of 8.3.1.

8.3.6 Upload Segment service

A server that receives an Upload Segment Indication for an Upload State Machine for which it has issued an Upload Segment Result(-) or an Upload Segment Result(+) with the MoreFollows parameter equal to FALSE, shall issue either a service error, specifying an Error Class = SERVICE and an Error Code = PRIMITIVES-OUT-OF-SEQUENCE, or an Abort request.

8.3.7 Get Domain Attributes service

The List of Capability parameter shall follow the limitations of 8.3.1.

8.3.8 Get Capability List service

The List of Capability parameter shall follow the limitations of 8.3.1.

8.4 Program Invocation management**8.4.1 Start service**

A ProgramInvocationState of non-existent shall be returned in a Result(-) when a request to Start a non-existent Program Invocation is received.

8.4.2 Stop service

A ProgramInvocationState of non-existent shall be returned in a Result(-) when a request to Stop a non-existent Program Invocation is received.

8.4.3 Resume service

A ProgramInvocationState of non-existent shall be returned in a Result(-) when a request to Resume a non-existent Program Invocation is received.

8.4.4 Reset service

A ProgramInvocationState of non-existent shall be returned in a Result(-) when a request to Reset a non-existent Program Invocation is received.

8.5 Variable access

8.5.1 Scattered access

It is strongly recommended that for services which use variable access, a Variable List Name or List of Variable be used instead of Scattered Access.

No Implementations shall be required to propose or accept the VSCA Parameter CBB.

8.5.2 Floating point

It is strongly recommended for services which use floating point types or values, that the choice of floating-point in the Data and TypeSpecification productions be used instead of the real choice.

No Implementations shall be required to propose or accept the REAL parameter CBB.

8.6 Semaphore management

Semaphore services are not considered in Phase 1.

8.7 Operator communication

No Operator Communication agreements have been identified to date.

8.8 Event management

Event Management services are not considered in Phase 1.

8.9 Journal management

Journal Management services are not considered in Phase 1.

Annex A (normative)

Backwards compatibility agreements**A.1 Introduction**

There is an installed base of real DIS 9506 based implementations. Providing support for application connectivity to both DIS and IS is desired as a migration strategy. These implementation agreements will allow IS based implementations to interoperate with DIS based implementations as described in ANNEX B. To achieve this backwards compatibility, the IS implementation shall support all of the agreements in this section.

It was found that the Abstract Syntax name object identifiers of both DIS and IS were identical. Therefore, the use of zero as the version number allows differentiation between an IS and a DIS based implementation. Since the abstract syntax name object identifier of any companion standard is different from that used by the DIS implementations, DIS implementations cannot interoperate with IS based implementations in a companion standard context.

NOTES

- 1 The value of zero is a valid value for this parameter in the DIS and not in the IS.
- 2 There are three types of implementations when considering MMS backwards compatibility.

- IMP-1:** An Implementation based on DIS 9506 as described in Annex B;
- IMP-2:** An Implementation based on IS 9506 with no backwards compatibility agreements applied;
- IMP-3:** An Implementation based on IS 9506 which includes the backwards compatibility agreements of this annex. Such an implementation can dynamically negotiate to interoperate with an IMP-1, an IMP-2, or an IMP-3 implementation.

Since the value of the minor version number is zero for DIS-based implementations, and is one or greater for implementations of ISO/IEC 9506, this value can be used to differentiate between IMP-1 and IMP-2. An IMP-3 system can interoperate with either of these systems. If an IMP-3 is the Calling system, it will offer a value of one (or greater) for the proposed version number. An IMP-1 system will respond with a value of the negotiated version number of zero, using the negotiation procedure defined in ISO/IEC 9506. The IMP-3 system will accept this response. If the IMP-3 system is the Called system and has received an initiate request with a value of zero for the proposed version number (from an IMP-1 system), it will respond with a value of zero for the negotiated version number. By following this procedure, an IMP-3 can interoperate with an IMP-2 or with another IMP-3 viewed as an IMP-2. After association context establishment, an IMP-3 system shall behave as either an IMP-1 or an IMP-2 system as appropriate on that particular association. The remainder of this section describes additional agreements which change an IMP-2 implementation into an IMP-3 implementation.

A.2 Backwards compatibility agreements for calling MMS implementations

A calling MMS Implementation shall be capable of receiving and supporting a negotiatedVersionNumber parameter in the Initiate Service confirm of zero.

A calling MMS Implementation which has received a negotiatedVersionNumber parameter in the Initiate Service confirm of zero shall support the modifications described in A.4.

A calling MMS Implementation shall be capable of receiving an Application Context Name parameter value appropriate to an IMP-1 or IMP-2 in the A-Associate confirm.

A calling MMS Implementation which has received a negotiatedVersionNumber of zero shall be capable of receiving and supporting an Initiate Response which has been encoded according to the modifications described in Appendix B, specifically the capability of receiving and supporting a negotiatedParameterCBB containing exactly 7 bits.

If a calling MMS Implementation receives an Initiate confirm primitive with a negotiatedVersionNumber parameter equal to zero, the calling MMS Implementation shall support the VLIS conformance building block if the Implementation claims support for any service which contains one or more parameters which indicate the VLIS CBB in its service definition.

A.3 Backwards compatibility agreements for called MMS implementations

A called MMS Implementation shall be capable of receiving and supporting a proposedVersionNumber parameter in the Initiate Service Indication of zero.

A called MMS Implementation which has received a proposedVersionNumber parameter in the Initiate Service Indication of zero shall support the modifications in A.4.

A called MMS Implementation shall be capable of receiving an Application Context Name parameter appropriate to an IMP-1 or IMP-2 in the A-Associate Indication.

A called MMS Implementation shall be capable of receiving and supporting an Initiate Request which has been encoded according to the modifications described in Appendix B, specifically the capability of receiving and supporting a proposedParameterCBB containing exactly 7 bits.

If a called MMS Implementation receives an Initiate Indication primitive with a proposedVersionNumber parameter equal to zero, the called MMS implementation shall support the VLIS conformance building block if the Implementation claims support for any service which contains one or more parameters which indicate the VLIS CBB in its service definition.

A.4 General backwards compatibility agreements

A.4.1 VMD logical status

If the current VMD State is SUPPORT-SERVICES-ALLOWED and the association minor version number is zero, then the vmdLogicalStatus parameter shall have a value of STATE-CHANGES-ALLOWED in a Status response or in an unsolicitedStatus request.

Annex B (normative)

DIS 9506 Modifications Required for Backwards Compatibility

B.1 Introduction

This annex is an integral part of this part. It documents the modifications to DIS 9506 required to describe implementations for which the agreements of this part provide backwards compatibility. This annex as applied to DIS 9506 is referred to as Version 0.

B.2 References

- [1] *MMS/1 Manufacturing Message Specification - ISO DIS 9506 - Service Definition, December 1987*
- [2] *MMS/2 Manufacturing Message Specification - ISO DIS 9506 - Protocol Specification, December 1987*
- [3] *NBS OSI Implementors Workshop Agreements - December 1987*

B.3 General

B.3.1 Implementation base

Version 0 is based upon Reference [3] in B.2 as it applies to MMS.

B.3.2 Rules of extensibility

The following sentence is appended to the last paragraph in section 8.2.1.1.5.2 Proposed Parameter CBB and the last paragraph in section 8.2.1.2.5.2 Negotiated Parameter CBB of DIS 9506-1.

"Any additional bits shall be ignored."

B.4 Modifications to the protocol definitions

B.4.1 Page 39, Section 7.5.2 of DIS 9506-2

CHANGE	
reportEventEnrollmentStatus	[60] IMPLICIT ReportEventEnrollmentStatus-Request,
TO	
reportEventEnrollmentStatus	[60] ReportEventEnrollmentStatus-Request,

B.4.2 Page 49, Section 7.6.4, DIS 9506-2

CHANGE	
ApplicationReference	::= SEQUENCE { ap-title ISO-8650-ACSE-1.AP-title OPTIONAL, ap-invocation-id ISO-86 50-ACSE-1.AP-invocation-id OPTIONAL, ae-qualifier ISO-8650-ACSE-1.AE-qualifier OPTIONAL, ae-invocation-id ISO-8650-ACSE-1.AE-invocation-id OPTIONAL }
TO	
ApplicationReference	::= SEQUENCE { ap-title [0] OBJECT IDENTIFIER OPTIONAL, ap-invocation-id [1] INTEGER OPTIONAL, ae-qualifier [2] INTEGER OPTIONAL, ae-invocation-id [3] INTEGER OPTIONAL }

B.4.3 Page 95, Section 12.2.1 of DIS 9506-2

CHANGE	
structure	[2] IMPLICIT SEQUENCE OF SEQUENCE {
TO	
structure	[2] IMPLICIT SEQUENCE {

B.4.4 Page 96, Section 12.3.1 of DIS 9506-2

CHANGE	
named	[4] IMPLICIT SEQUENCE {
TO	
named	[5] IMPLICIT SEQUENCE {

B.4.5 Page 98, Section 12.4.2 of DIS 9506-2

CHANGE	
generalized-time	[10] IMPLICIT GeneralizedTime,
TO	
generalized-time	[11] IMPLICIT GeneralizedTime,

B.4.6 Page 138, Section 15.14 of DIS 9506-2

CHANGE	
additionalDetail	[9] IMPLICIT EE-Additional-Detail OPTIONAL
TO	
additionalDetail	[9] EE-Additional-Detail OPTIONAL

B.4.7 Page 166, Section 17.10 of DIS 9506-2

CHANGE the transfer syntax object identifier value from	
{ iso asn1(1) basic-encoding(1) }	
TO	
{ joint-iso-ccitt asn1(1) basic-encoding(1) }	

B.5 Behavioral requirements

B.5.1 Filenames

File Names are specified in accordance with the NBS Implementors' agreements for FTAM Reference [3] in B.2.

B.5.2 Identify service

In the identify service, the vendor, model and revision fields may be of any length, but only the first 64, 16, and 16 octets respectively are treated as significant.

B.5.3 Initiate service

An MMS Client will:

- a) propose 1 or greater for the value of the Proposed Max Serv Outstanding Called parameter in the initiate service when initiating the application association (calling);
- b) offer 1 or greater for the value of the Negotiated Max Serv Outstanding Calling parameter in the initiate service when receiving the application association initiation (called).

An MMS Server will:

- a) propose 1 or greater for the value of the Proposed Max Serv Outstanding Calling parameter in the initiate service when initiating the application association (calling);
- b) offer 1 or greater for the value of the Negotiated Max Serv Outstanding Called parameter in the initiate service when receiving the application association initiation (called).

B.5.3.1 Minimum segment size

MMS implementations are able to parse and process 512 octets of MMSPDU as they are encoded in ASN.1 basic encoding rules.

B.5.3.2 Maximum segment size

The Max Segment Size is defined as the maximum number of octets in an MMSPDU encoded using the negotiated transfer syntax. This size will apply to all MMSPDU's with the exception of the Initiate-Request PDU, Initiate-Response PDU, and the Initiate-Error PDU. The max segment size will be negotiated during connection initiation using the Proposed Max Segment Size and Negotiated Max Segment Size parameters of the MMS initiate service.

The Max Segment Size will be applied as follows:

- a) Any received MMSpdu which is less than or equal to the Max Segment Size will be properly parsed and processed;
- b) An MMS implementation will not send an MMSpdu whose size exceeds the Max Segment Size.

B.5.4 Abstract syntax name

The ASN.1 object identifier value for the abstract syntax name will be the same as specified on page 166, section 17.10 of DIS 9506-2.

B.5.5 Application context name

The ASN.1 object identifier value for the application context name will be the same as specified on page 166, section 17.11 of DIS 9506-2.

An MMS Implementation ignores the Application Context Name in the A-Associate indication and the A-Associate confirm.

B.5.6 Minor version number

The Minor Version Number is zero.

B.6 Parameter CBB subset

The following subset of MMS Parameter CBBs were considered during preparation of this annex:

- a) STR1;
- b) NEST;
- c) VADR;
- d) VNAM.

B.7 Service subset

The following subset of MMS services were considered during preparation of this annex.

Table 2 - MMS Service Subset

Initiate	Output
Conclude	TakeControl
Cancel	RelinquishControl
Status	ReportSemaphoreStatus
GetNameList	ReportPoolSemaphoreStatus
Identify	ReportSemaphoreEntryStatus
UnsolicitedStatus	CreateProgramInvocation
GetCapabilityList	DeleteProgramInvocation
InitiateDownloadSequence	Start
DownloadSegment	Stop
TerminateDownloadSequence	Resume
InitiateUploadSequence	Reset
UploadSegment	Kill
TerminateUploadSequence	GetProgramInvocationAttributes
RequestDomainDownload	ObtainFile
RequestDomainUpload	GetEventConditionAttributes
LoadDomainContent	ReportEventConditionStatus
StoreDomainContent	GetAlarmSummary
DeleteDomain	ReadJournal
GetDomainAttributes	WriteJournal
Read	InitializeJournal
Write	CreateJournal
InformationReport	DeleteJournal
GetVariableAccessAttributes	ReportJournalStatus
Input	

Stable Implementation Agreements for Open Systems Interconnection Protocols: Part 21 - Character Set

Output from the December 1992 Open Systems
Environment Implementors' Workshop (OIW)

Foreword

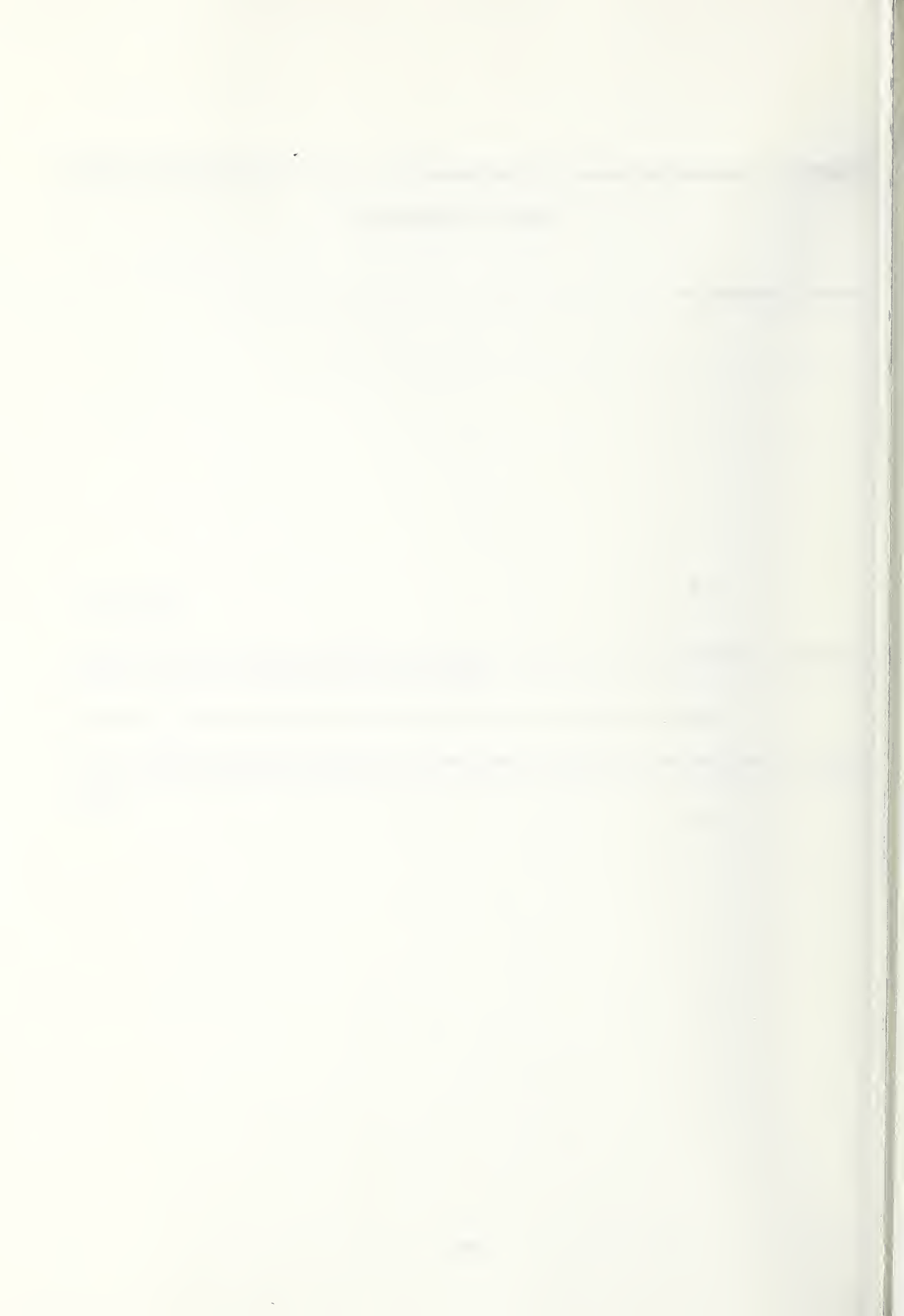
This part of the Stable Implementation Agreements was prepared by the Workshop Chair of the Open Systems Environment Implementors' Workshop (OIW).

Text in this part has been approved by the Plenary of the above-mentioned Workshop.

Future changes and additions to this version of these Implementor Agreements will be published as change pages. Deleted and replaced text will be shown as struckout. New and replacement text will be shown as shaded.

Table of Contents

Part 21 - Character Set 1



Part 21 - Character Set

Editor's Note - Refer to the Working Implementation Agreements for information on this text.



Stable Implementation Agreements for Open Systems Interconnection Protocols: Part 22 - ODA Image DAP

**Output from the December 1992 Open Systems
Environment Implementors' Workshop**

**SIG Chair: James Wing, IBM
SIG Editor: Frank Spielman, NIST**

Foreword

This part of the Stable Implementation Agreements was prepared by the Office Document Architecture (ODA) Special Interest Group (SIG) of the Open Systems Environment Implementors' Workshop (OIW). Development of this document application profile has been done in liaison with several organizations. These include the DoD Computer-aided Acquisition and Logistic Support (CALS) Office, Navy's David Taylor Research Center, and the ad-hoc Tiling Task Group.

This document application profile is intended to be suitable for the interchange of large format raster images which may be annotated with character, raster, or geometric revisions.

This part contains four annexes:

- a) annex A (normative): Amendments and corrigenda;
- b) annex B (informative): Recommended practices;
- c) annex C (informative): References to other standards and registers;
- d) annex D (informative): Supplementary information on attributes.

Future changes and additions to this version of these Implementor Agreements will be published as a new part. Deleted and replaced text will be shown as struckout. New and replacement text will be shown as shaded.

This part uses a convention of double and single quotes that has been established by ISO for use in the ODA base standard and related document application profiles. The convention is to use within the text double quotes to accentuate ODA attribute names and single quotes to accentuate values for those attributes.

Table of Contents

Part 22 - ODA Image DAP	1
0 Introduction	1
1 Scope and field of applications	1
2 Normative references	2
2.1 ISO	2
2.2 CCITT	3
3 Definitions and terminology	3
3.1 Definitions	3
3.2 Constituent names	4
4 Relationship to other DAPs	4
5 Conformance	4
5.1 Data stream conformance	4
5.2 Implementation conformance	5
6 Characteristics supported by this DAP	5
6.1 Overview	5
6.2 Logical constituents	7
6.3 Layout constituents	7
6.3.1 Overview of the layout characteristics	7
6.3.2 DocumentLayoutRoot	8
6.3.3 Page characteristics	8
6.3.3.1 CompositePage	8
6.3.3.2 Page dimensions	8
6.3.3.3 Nominal page sizes	9
6.3.4 OriginalImage	9
6.3.5 RevisionAnnotation	9
6.3.6 SpecificBlock	9
6.3.7 GenericBlock	10
6.4 Document layout characteristics	12
6.5 Content layout and imaging control	12
6.5.1 Raster graphics content	13
6.5.1.1 Introduction	13
6.5.1.2 Raster graphics content architecture	13
6.5.1.3 Raster graphics encoding methods	13
6.5.1.4 Raster presentation	14
6.5.2 Character content	15
6.5.2.1 Character content architecture class	15
6.5.2.2 Character repertoires	15
6.5.2.3 Code extension techniques	15

6.5.2.4	Line spacing	16
6.5.2.5	Character spacing	16
6.5.2.6	Character path and line progression	16
6.5.2.7	Character orientation	17
6.5.2.8	Emphasis	17
6.5.2.9	Tabulation	18
6.5.2.10	Alignment	18
6.5.2.11	Fonts	19
6.5.2.12	Reverse character strings	19
6.5.2.13	Superscripts and subscripts	19
6.5.2.14	Substitution of characters	19
6.5.2.15	Use of control functions	19
6.5.3	Geometric graphics content	20
6.6	Miscellaneous features	20
6.6.1	Resource documents	20
6.6.2	Application comments	21
6.7	Document management features	21
7	Specification of constituent constraints	22
7.1	Document profile constraints	22
7.1.1	Macro definitions	22
7.1.2	Constituent constraints	29
7.1.2.1	DocumentProfile	29
7.2	Logical constituent constraints	33
7.3	Layout constituent constraints	33
7.3.1	Macro definitions	33
7.3.2	Factor constraints	33
7.3.3	Constituent constraints	33
7.3.3.1	DocumentLayoutRoot	33
7.3.3.2	CompositePage	33
7.3.3.3	OriginalImage	34
7.3.3.4	RevisionAnnotation	34
7.3.3.5	SpecificBlock	35
7.3.3.6	GenericBlock	36
7.4	Layout style constraints	37
7.5	Presentation style constraints	37
7.5.1	Macro definitions	37
7.5.2	Factor constraints	38
7.5.3	Presentation style constituent constraint	38
7.5.3.1	PStyle1	38
7.5.3.2	PStyle2	38
7.5.3.3	PStyle3	39
7.6	Content portion constraints	39
7.6.1	Macro definitions	39
7.6.2	Factor constraints	39
7.6.3	Constituent constraints	39
7.6.3.1	Character content portion	39
7.6.3.2	Raster graphics content portion	40
7.6.3.3	Geometric graphics content portion	41

PART 22 - ODA Image DAP

December 1992 (Stable)

7.7	Additional usage constraints	41
8	Interchange format	41
8.1	Interchange format ODIF (class A)	41
8.1.1	Interchange format	41
8.1.2	DAP Identifier	42
8.1.3	Encoding of application comments	42
8.2	Interchange format SDIF	42
8.2.1	Interchange format	42
8.2.2	DAP Identifier	42
8.2.3	Encoding of application comments	42
8.3	Encoding of raster content information	43
Annex A (normative)		
Amendments and corrigenda		44
A.1	Amendments	44
A.1.1	Amendments to the base standard	44
A.2	Corrigenda	44
A.2.1	Corrigenda to this DAP	44
Annex B (informative)		
Recommended practices		45
B.1	Transfer methods for ODA	45
B.1.1	Conveyance of ODA over CCITT X.400-1984	45
B.1.2	Conveyance of ODA over FTAM	45
B.1.3	Conveyance of ODA over DTAM	46
B.1.4	Conveyance of ODA over flexible disks	46
B.2	Font reference	46
B.3	ISO 8632 (CGM) constraints for this DAP	47
B.3.1	Delimiter elements	47
B.3.2	Metafile descriptor elements	48
B.3.3	Picture descriptor elements	48
B.3.4	Control elements	48
B.3.5	Graphical primitive elements	48
B.3.6	Attribute elements	49
B.3.7	External elements	50
B.4	Interoperability with SGML applications	51
Annex C (informative)		
References to other standards and registers		52
Annex D (informative)		
Supplementary information on attributes		54

List of Figures

Figure 1 - Constituents 6

Figure 2 - Document layout structure 7

List of Tables

Table 1	Dimensions for various page sizes	11
Table 2	Layout attributes	12
Table D.1	Content coding attributes	54
Table D.2	Presentation attributes	54
Table D.3	Document profile attributes	55

Part 22 - ODA Image DAP

0 Introduction

This is the definition of a specification for an Open Document Architecture (ODA) Document Application Profile (DAP) named ODA Image DAP. This DAP is suitable for Interchanging documents in formatted form. The documents contain primarily raster graphics images but may also contain character and geometric graphics content portions.

There are two DAP object identifiers supporting this DAP with the only difference being in the encoding of the data stream. One uses the ASN.1 based ODIF encoding. The other uses the SGML/SDIF based ODL encoding. When this document refers to *this profile*, it is referring to this specification regardless of which DAP identifier may be selected to create the data stream.

The DAP is defined in accordance with ISO 8613-1 and CCITT T.411 and follows the standardized proforma and notation defined in ISO 8613-1 Annex F. The DAP is based on ODA as defined in ISO 8613 and the Tiled Raster Graphics Addendum to ISO 8613, Part 7.

1 Scope and field of applications

This DAP specifies an interchange format suitable for transfer of structured documents between equipment designed for raster processing. The documents supported by this DAP are based on a paradigm of an electronic engineering drawing or illustration. Such documents contain one or more pages. Each page consists of a base image in the form of a bi-tonal raster graphics, character, or geometric graphics content. This base image may be further annotated with character, raster graphics or geometric graphics content. These latter content portions serve to provide revision control for the engineering drawing or illustration. There is no restriction on the minimum size of the base image.

This document defines a DAP that allows large format raster documents to be interchanged in a formatted form in accordance with ISO 8613.

It is assumed that, when negotiation is performed by the service using this DAP, all non-basic values are subject to negotiation.

This DAP is independent of the processes carried out in an end system to create, edit, or reproduce raster documents. It is also independent of the means to transfer the document which, for example, may be by means of communication links or exchanged storage media.

The features of a document that can be interchanged using this DAP fall into the following categories:

- a) Page format features - these concern how the layout of each page of a document will appear when reproduced;
- b) Raster graphics layout and imaging features - these concern how the document content will appear within pages of the reproduced document;
- c) Raster graphics coding - these concern the raster graphics representations and control functions that make up the document raster graphics content.

2 Normative references

The following references are required in order to implement this DAP:

2.1 ISO

- [1] ISO 8613-1 : 1989, *information processing - Text and Office Systems; Open Document Architecture (ODA) and Interchange Format - Part 1: Introduction and General Principles*;
- [2] ISO 8613-2 : 1989, *information processing - Text and Office Systems; Open Document Architecture (ODA) and Interchange Format - Part 2: Document Structures*;
- [3] ISO 8613-4 : 1989, *information processing - Text and Office Systems; Open Document Architecture (ODA) and interchange Format - Part 4: Document Profile*;
- [4] ISO 8613-5 : 1989, *information processing - Text and Office Systems; Open Document Architecture (ODA) and interchange Format - Part 5: Open Document interchange Format*;
- [5] ISO 8613-6 : 1989, *Information processing - Text and Office Systems; Open Document Architecture (ODA) and Interchange Format - Part 6: Character Content Architecture*;
- [6] ISO 8613-7 : 1989, *information processing - Text and Office Systems; Open Document Architecture (ODA) and Interchange Format - Part 7: Raster Graphics Content Architectures*;
- [7] ISO 8613-8 : 1989, *Information processing - Text and Office Systems; Open Document Architecture (ODA) and interchange Format - Part 8: Geometric Graphics Content Architectures*;
- [8] ISO 8613-1 : 1991, *information processing - Text and Office Systems; Open Document Architecture (ODA) and interchange Format - Part 1: Annex F - A Document Application Profile Proforma and Notation*;
- [9] ISO 8613-7 : (to be published), *information processing - Text and Office Systems; Office Document Architecture (ODA) and interchange Format - Part 7: Amendment - Tiled Raster Graphics Addendum to ISO 8613, Part 7*;
- [10] ISO 8613-7 : (to be published), *Information processing - Text and Office Systems; Office Document Architecture (ODA) and interchange Format - Part 7: Amendment - Additional Bit Order Mapping Addendum*;
- [11] ISO 646 : 1990, *Information processing - ISO 7-bit coded character sets for information interchange*;
- [12] ISO 8859-1 : 1983, *information processing - 8-bit Single-byte coded graphic character sets - Part 1: Latin alphabet No. 1*;
- [13] ISO 6937-2 : 1983, *information processing - Coded character sets for text communication - Part 2:*

Latin alphabet and non-alphabetic characters;

- [14] ISO 2022 : 1986, *Information processing - ISO 7-bit and 8-bit coded character sets - Code extension techniques*;
- [15] ISO 7350 : 1984, *Text communication - Registration of graphic character subrepertoires*;
- [16] ISO 8824 : 1987, *Information Processing Systems - Open Systems Interconnection - Specification of Abstract Syntax Notation One (ASN.1)*;
- [17] ISO 8825 : 1987, *Information Processing Systems - Open Systems Interconnection - Specification of Basic Encoding Rules for Abstract Syntax Notation One (ASN.1)*;
- [18] ISO 8879 : 1986, *Information processing - Text and office systems - Standard Generalized Markup Language (SGML)*;
- [19] ISO 8879 : 1986, *Information processing - Text and office systems - Standard Generalized Markup Language (SGML), Amendment 1*;
- [20] ISO 9069 : 1988, *Information processing - SGML support facilities - SGML Document Interchange Format (SDIF)*.

2.2 CCITT

- [19] Recommendation T.4 : 1988, *Standardization of Group 3 Facsimile Apparatus for Document Transmission*.
- [20] Recommendation T.6 : 1988, *Facsimile Coding Schemes and Coding Control Functions for Group 4 Facsimile Apparatus*.

3 Definitions and terminology

3.1 Definitions

The definitions given in ISO 8613-1 are applicable to this document.

3.2 Constituent names

Each constituent that may be included in a document that conforms to this profile has been given a unique name which serves to identify that constituent throughout this profile.

The convention is that full names are used (i.e., no abbreviations are used), two or more words in a name are concatenated and each word begins with a capital. Examples of constituent names used in this profile are CompositePage, DocumentLayoutRoot, and SpecificBlock.

In clause 6, each constituent provided by this profile is underlined once at the point in the text at which the purpose of that constituent is defined. This also serves to identify all the constituents provided by this profile.

The same constituent names are also used in the technical specification in clause 7 so that there is a one-to-one correspondence between the use of these names in clauses 6 and 7.

Although the constituent names relate to the purpose of the constituents, the semantics of constituents must not be implied from the actual names that are used. Also, these names do not appear in an interchanged document but a mechanism for identifying constituents in an interchange document is provided. Thus in an application using this profile, the constituents may be known to the user by different names.

4 Relationship to other DAPs

Functionally, this DAP is a functional superset of the CCITT Recommendation T.503, A Document Application Profile for the Interchange of Group 4 Facsimile Documents.

5 Conformance

In order to conform to this DAP, a data stream representing a document must meet the requirements specified in 5.1.

The requirements for implementations that originate and/or receive data streams conforming to this DAP are specified in 5.2.

5.1 Data stream conformance

The following requirements apply to the encoding of data streams that conform to these agreements:

- a) The data stream shall be encoded in accordance with the ASN.1 encoding rules defined in ISO 8825 or the SGML grammar and syntax of ISO 8879;
- b) The data stream shall be structured in accordance with the interchange format defined in clause 8;
- c) The document shall be structured in accordance with only the formatted document

architecture class specified in clause 7. In addition, the document shall contain all mandatory constituents specified for that class and may optionally contain constituents permitted for that class as specified in clause 7;

- d) Each constituent shall contain all those attributes specified as required for that constituent in this profile. Other attributes may be specified provided they are permitted for that constituent;
- e) The attributes shall have values within the range of permissible values specified in this profile;
- f) The encoded document shall be structured in accordance with the abstract document architecture defined in ISO 8613-2;
- g) The encoded document shall be structured in accordance with the characteristics defined in clause 6 and shall contain only those features defined in clause 6.

5.2 Implementation conformance

This clause states the requirements for implementations claiming conformance to this DAP.

A conforming receiving implementation must be capable of receiving *either* any data streams conforming to this profile structured in accordance with ODIF *or* any data streams conforming to this profile structured in accordance with ODL *or* both of these. Receiving usually, but not always, involves recognizing and further processing the data stream elements.

6 Characteristics supported by this DAP

This clause describes the characteristics of documents that can be represented by data streams conforming to this profile. This clause also describes how these characteristics are represented in terms of divisional components of the data streams.

6.1 Overview

This DAP describes the features of ISO 8613 that are needed to support the interchange of documents containing images. It specifies interchange formats for the transfer of structured documents with simple layout structures.

This DAP describes documents that can be interchanged in the formatted form, which facilitates the reproduction of a document as intended by the originator.

The content within the document forming the original or base image(s) may be formatted processable raster graphics, formatted processable geometric graphics, and/or formatted character. This is intended to facilitate the reproduction of the document content as intended by the originator or allows the reformatting of the document content.

The content allowed within the document to annotate revisions to the base image(s) may also be formatted processable raster graphics, formatted processable geometric graphics, and/or formatted character.

This clause describes the layout features that can be represented in documents conforming to this DAP. The features are described in terms that are typical of the user-perceived capabilities and semantics found in a raster document interchange environment.

For the purpose of interchange, a document is represented as a collection of constituents, each of which is represented by a set of attributes. The constituents that make up a formatted document are defined below in this clause and are illustrated in figure 1.

Document Profile
Generic Layout Structure (Optional)
Presentation Style (Optional)
Specific Layout Structure
Content Portion Description

Figure 1 - Constituents

Constituents defined as **required** must occur in any document that conforms to this profile. Constituents listed as **optional** may or may not be present in the document, depending on the requirements of the particular document.

The required constituents include:

- a) a document profile;
- b) layout object descriptions representing a specific layout structure;
- c) content portion description.

The only optional constituents are presentation style and generic layout structure.

6.2 Logical constituents

Not applicable.

6.3 Layout constituents

This clause describes the features of the layout objects that can be represented in documents conforming to this DAP.

6.3.1 Overview of the layout characteristics

The document structure allows the document content to be laid out and presented in one or more pages. Each page in a document may consist of a single raster graphics content. This would be the case for an original image of an engineering drawing, illustration, or other raster scanned image. Optionally, each page in a document may consist of an original image which contains raster graphics, geometric graphics, and/or character content, with additional character, raster graphics or geometric graphics content, representing a set of revision annotations of the original image.

A specific layout structure of the document conforming to this application profile consists of a four-level hierarchy consisting of a document layout root, composite pages, frames, and blocks. The document can consist of multiple composite pages where each page represents a single image including any revision annotations. The composite pages consist of frames which in turn contain blocks containing the content associated with the base image and the revision annotation.

Figure 2 is an illustration of the features of the document layout structure supported by this DAP.

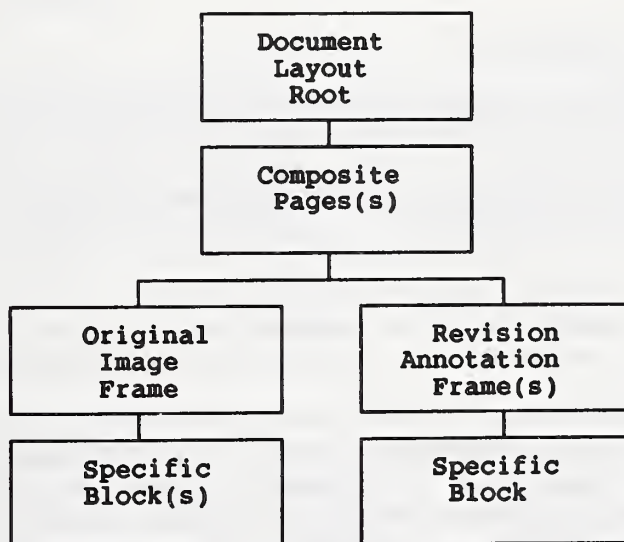


Figure 2 - Document layout structure

6.3.2 DocumentLayoutRoot

A DocumentLayoutRoot is the top level in a document layout structure. A DocumentLayoutRoot consists of a sequence of one or more CompositePage constituent constraints.

6.3.3 Page characteristics

Only one constituent constraint is provided to present pages within a document.

A document consists of a sequence of one or more composite pages. In a document's composite page, two types of frames are used to position content information on the page. One frame type is used to position the content representing the original image on the page. Only one frame of this type is allowed per page, but it may contain any number of raster graphics, geometric graphics, or character content portions. The second frame type is used to position a character, raster graphics or geometric graphics content representing a revision annotation on the page. There may be one or more of the frames containing a revision annotation.

A document may consist of multiple pages of different sizes. Each page may be either landscape or portrait orientation. Both orientations are permitted in the document.

6.3.3.1 CompositePage

A CompositePage is a constituent constraint which defines a composite-page that corresponds to the page area used for presenting the sequence of an OriginalImage frame and zero or more RevisionAnnotation frames.

6.3.3.2 Page dimensions

A wide variety of page dimensions are supported including large format raster documents. The dimensions of the pages may be specified as any value, in BMU measurement units, including the larger sizes produced from foldout-size images and roll paper. These sizes apply to both portrait and landscape orientations. The page sizes include: ISO A0-A5, ANSI A-K, Japanese legal and letter, foldouts 27.94 cm (11 in.) X 34.56 cm (14 in.) and 27.94 cm (11 in.) X 43.18 cm (17 in.), and 27.94 cm (11 in.) roll paper.

Dimensions equivalent to or less than the common assured reproduction area (CARA) of ISO A4 and North American Letter (NAL) in portrait or landscape orientation are basic values. Larger page sizes including those produced from roll paper are non-basic and their use must be indicated in the document profile (See table 2).

The default dimensions are the CARA of North American Letter (A). Any default page dimensions may be specified in the document profile subject to the maximum dimensions defined above by using the "page dimensions" attribute. The "page position" attribute may be used to specify the position of the pel array image on the page. Although actual page dimensions may be used allowing for the raster content to completely fill a page leaving no borders, it is advised that the assured reproduction area (ARA) listed in table 1 be used wherever feasible. See 7.3 of ISO 8613-2 for general rules for positioning pages on presentation surfaces.

6.3.3.3 Nominal page sizes

The nominal page sizes that may be specified are listed in table 1. In addition, 11 inch roll paper of any length is supported. These may be specified in portrait or landscape orientations. All values of nominal page size are non-basic and hence all values used in a document must be indicated in the document profile using the "medium type" attribute (See table 2).

Any of the nominal page sizes defined in table 1, subject to the restriction specified above, may be specified as the default value in the document profile.

Table 1 also includes the recommended ARA. Information loss may occur when a document is reproduced if the dimensions of the CompositePage exceed the ARA for the specified nominal page size.

6.3.4 OriginalImage

An OriginalImage is a constituent constraint which defines a lowest level frame used for laying out the original image of an engineering drawing, illustration or other image. This frame contains one or more SpecificBlocks each of which may contain one of a character content portion, a raster graphics content portion, or a geometric graphics content portion. Note that there must be exactly one OriginalImage frame on each page.

This type of frame has a fixed position and dimensions. The position, if not specified, defaults to the origin of the page. The dimensions, if not specified, default to the maximum size that can be achieved for the position within the area of the page.

6.3.5 RevisionAnnotation

A RevisionAnnotation is a constituent constraint which defines a lowest level frame used for laying out the revision annotation associated with the original image. This frame contains a single SpecificBlock containing either a character content portion, a raster graphics content portion or a geometric graphics content portion.

This type of frame has a fixed position and dimensions. This provision provides for the capability of positioning of revision annotation anywhere on the page. Registration of revision annotation over a portion of the original image, as in revision artwork, is accomplished using this capability.

6.3.6 SpecificBlock

A SpecificBlock is a constituent constraint which defines a basic layout object used to position and image the content portions associated with either an OriginalImage or RevisionAnnotation frame.

The position of the block is fixed and defaults to the origin of the superior frame. The dimensions default to the maximum size that can be achieved for the position within the area of the superior frame.

6.3.7 GenericBlock

GenericBlock is a constituent constraint which defines a layout object class which can define content that is common and can be referenced throughout the document. Any content type (raster, character, or geometric graphics) can be defined using this technique.

Table 1 Dimensions for various page sizes

Page type	Size	Size (BMU)	ARA (BMU)
- Metric			
ISO-A5	148mm x 210mm	7015 x 9920	not defined
ISO-A4	210mm x 297mm	9920 x 14030	9240 x 13200
ISO-A3	297mm x 420mm	14030 x 19840	13200 x 18480
ISO-A2	420mm x 594mm	19840 x 28060	18898 x 27118
ISO-A1	594mm x 841mm	28060 x 39680	26173 x 37843
ISO-A0	841mm x 1189mm	39680 x 56120	37843 x 54283
- ANSI, North American (NA)			
NA-A	8.5in x 11in	10200 x 13200	9240 x 12400
NA-B	11in x 17in	13200 x 20400	12744 x 19656
NA-C	17in x 22in	20400 x 26400	19500 x 25800
NA-D	22in x 34in	26400 x 40800	25800 x 39600
NA-E	34in x 44in	40800 x 52600	39600 x 52200
NA-F	28in x 40in	33600 x 48000	32400 x 47400
NA-G	11in x 90in	13200 x 108000	12400 x 106800
NA-H	28in x 143in	33600 x 171600	31400 x 170400
NA-J	34in x 176in	40800 x 211200	39600 x 210000
NA-K	40in x 143in	48000 x 171600	47400 x 170400
NA-Legal	8.5in x 14in	10200 x 16800	9240 x 15480
- Foldouts			
Small	11in x 14in	13200 x 16800	12744 x 15480
NA-B	11in x 17in	13200 x 20400	12744 x 19656
- Japan			
Legal	257mm x 364mm	12141 x 17196	11200 x 15300
Letter	182mm x 257mm	8598 x 12141	7600 x 10200

Tutorial Note - These page sizes are for the portrait orientation.

Table 2 Layout attributes

Attributes	Basic values	Default values	Non-basic values
Page dimensions **	CARA NA A, ISO A4	CARA NA-A	ARA NA B-K, ISO A0-A3, Japan legal, 11" Roll Paper
Medium-type ** (Nominal page size)	None	None	NA A-K, ISO A0-A5, Japan letter & legal, 11" Roll Paper

Tutorial Note - See table 1 **

6.4 Document layout characteristics

This DAP provides only for formatted documents. Hence, no provision is made for constraining the document layout process other than as implied in the formatted documents supported by this DAP. In particular, these formatted documents are characterized by the following:

- Documents containing only composite pages;
- Documents may contain one or more pages;
- Pages may vary by orientation within a document;
- As a minimum, each page contains a single raster graphics, geometric graphics, or character content portion, representing the original image;
- Each page may additionally contain one or more character, raster graphics or geometric graphics content portions representing revision annotation;
- Content is positioned within fixed position and dimension frames.

6.5 Content layout and imaging control

A document is modelled as an original image with optional revision annotation(s). The original image and the revision annotation(s) may be represented by either character, raster graphics, or geometric graphics content portions, as specified in ISO 8613-6, ISO 8613-7 and ISO 8613-8, respectively.

The content architectures that may be specified using the attribute "content architecture class" are formatted character, formatted processable raster graphics and formatted processable geometric graphics. Any of

the above contents may be specified as the default in the document profile.

6.5.1 Raster graphics content

6.5.1.1 Introduction

This clause defines the features that are applicable to the raster graphics content.

The default values for the following features may be specified in the document profile:

- a) type of coding (required);
- b) compression;
- c) pel path;
- d) line progression;
- e) pel spacing;
- f) spacing ratio.

The specification in a document of a non-basic value by a presentation or coding attribute must be indicated in the document profile.

6.5.1.2 Raster graphics content architecture

The formatted processable raster graphics content architecture is supported by this DAP and will frequently be the primary content architecture in a document.

In a composite page, multiple content portions may be associated with the original image, whereas only one content portion may be associated with a given revision annotation.

6.5.1.3 Raster graphics encoding methods

The content may be encoded in accordance with the encoding schemes defined in CCITT Recommendations T.4 and T.6. In the case of T.4, either the one-dimensional or two dimensional encoding scheme may be used. Also the 'bit-map encoding' scheme defined in ISO 8613-7 may be used. All these forms of encoding may be used in a single document and all are basic values. 'Uncompressed' mode of encoding may also be used but only as a non-basic value.

In a content portion, it is required that the coding attribute "number of pels per line" is specified. The coding attribute "number of lines" may also be specified. No restriction is placed on the values that may be specified for these coding attributes. This profile places no constraints on the size of the pel arrays that may be used.

The type of coding method used is specified by the attribute "type of coding". The use of this attribute is mandatory in the "document architecture defaults" of the document profile to define the default value of either 'T.6 encoding' (untiled), 'T.6 encoding - MSB' (untiled), or 'tiled encoding'. The use of this attribute in the description of the content portions is non-mandatory. If this attribute is not specified for a particular content portion, then the default value specified in the "document architecture defaults" of the document profile is used.

If the tiled encoding method is used, the default value of 512 for the "number of pels per tile line" and "number of lines per tile" must be used. No other values are supported, therefore these two attributes do not need to be specified. If the "tile types" attribute is not present, then all tiles will be T.6 encoded. If it is present, then there must be a value specified for each tile in which case only 'null background', 'null foreground', 'T.6 encoded', 'T.6 encoded - MSB', or 'bitmap encoded' values are supported. The T.4 encodings are not supported. There are no restrictions on the use of the "tilling offset" attribute other than that specified in ISO 8613-7 Addendum.

See table D.1, Annex D, for a tabulated list of the attributes and their basic, default, and non-basic values.

6.5.1.4 Raster presentation

Raster presentation is controlled by the presentation attributes specified in ISO 8613-7. This DAP provides for additional constraints on these presentation attributes as specified below.

The basic values for the attribute "pel path" supported by this profile are 0 and 90 degrees. The "pel path" values of 180 and 270 degrees are non-basic.

The basic values for the attribute "line progression" supported by this profile is 270 degrees. The "line progression" value of 90 degrees is non-basic.

Any value may be explicitly specified for pel spacing provided that the spacing between pels is not less than 1 BMU. The pel spacing need not be an integer value. The value of 'null' may not be specified because the scalable layout process is not supported. The specification of the spacings of 16, 12, 8, 6, 5, 4, 3, 2, and 1 BMU between adjacent pels are basic. The specification of any other spacing is non-basic and must be specified in the document profile.

NOTES

1 The basic pel spacing values listed above are equivalent to resolutions of 75, 100, 150, 200, 240, 300, 400, 600, and 1200 pels per 25.4mm respectively when the BMU is interpreted as 1/1200 inch.

2 The attribute "pel spacing" specifies two integers, the ratio of which determines the pel spacing. No restriction is placed on the values of these integers.

There are no restrictions on the use of the "clipping" attribute. The "image dimensions" attribute is not supported.

There are no restrictions placed on the value of the "spacing ratio" attribute providing that the resultant line spacing is not less than 1 BMU. Also, the line spacing need not be an integral number of BMUs. All values are basic.

See table D.2, Annex D, for a tabulated list of the attributes and their basic, default, and non-basic values.

6.5.2 Character content

The formatted character content is permitted in this DAP for use in either the original image or in revision annotations of that original image.

The specification in a document of a non-basic feature by a presentation attribute or control function must be indicated in the document profile.

6.5.2.1 Character content architecture class

When using character content, only one content portion may be associated with a basic component. The content information in a content portion must be present.

6.5.2.2 Character repertoires

The basic character set supported by this profile is the primary character set of ISO 8859-1. This must be designated to the G0 set and invoked to the GL. Any other graphic character set which is registered in accordance with ISO 2375 may be designated and invoked at any point in the document provided its use is announced in the document profile as a non-basic value using the character presentation attribute "graphic character sets". No locking shift functions are specified in this presentation attribute. The default graphic character sets which apply to the content portions within a document can be specified in the document profile using the presentation attribute "graphic character sets".

Using code extension techniques, the graphic character sets designated and/or invoked at the beginning of a content portion containing character content are specified using the presentation attribute "graphics character sets".

If the character set defined in ISO 6937-2 is designated and invoked, then the use of any sub-repertoire registered according to ISO 7350 may be specified. All sub-repertoires are non-basic and their use must be indicated in the document profile.

6.5.2.3 Code extension techniques

The code extension techniques specified in ISO 2022 may be used subject to the following restrictions:

- a) G0 set: only the primary character sets of ISO 6937-2, ISO 8859-X (where ISO 8859-X corresponds to any finalized part of ISO 8859) and a version of ISO 646 may be designated for this set; these character sets may only be invoked in GL;
- b) G1, G2, G3 sets: no restrictions are placed on the character sets that may be designated for these sets; these sets may only be invoked in GR;
- c) The locking and single shift functions allowed should be restricted to the following:

LS0 for the G0 set

LS1R for the G1 set

LS2R for the G2 set

LS3R for the G3 set

SS2

SS3;

d) When specifying the presentation attribute "graphic character sets", it is necessary to invoke character sets for both GL and GR. Thus an allowed character set must be designated into G0, as specified above, and invoked into GR. It is also necessary to invoke a character set into GR which has been designated into G1, G2 or G3 sets;

e) The empty set should be designated and invoked in GR if no other specific set is invoked into GR.

The announcement and encoding of these functions are to be as specified in ISO 2022.

6.5.2.4 Line spacing

Any value of line spacing may be specified. Values of 150, 200, 300 and 400 BMUs are basic; the use of any other value in a document is non-basic and must be indicated in the document profile. The line spacing may be specified at the beginning of the content associated with a basic component using the presentation attribute "line spacing". The value may be changed anywhere within the content portion using the control functions SVS and SLS.

6.5.2.5 Character spacing

Any value of character spacing may be specified. Values greater than or equal to 100 are basic; the use of any other value in a document is non-basic and must be indicated in the document profile. The character spacing may be specified at the beginning of the content associated with a basic component using the attribute "character spacing". The value may be changed anywhere within a content portion using the control functions SHS or SCS.

6.5.2.6 Character path and line progression

Both horizontal and vertical writing directions may be used within a character content. In the case of horizontal writing, the characters progress either from left to right or from right to left across the page and the line progression is from the top of the page to the bottom. In the case of vertical writing, the characters progress from the top of the page to the bottom and the line progression is from the right to the left. The values of character path and line progression may be specified at the beginning of the content associated with a basic component using the presentation attributes "character path" and "line progression",

respectively. These values cannot be changed within a content portion.

6.5.2.7 Character orientation

The character orientation may be specified as 0 or 90 degrees depending on whether vertical or horizontal writing is used. When vertical writing is used, characters are normally orientated at 0 degrees. When horizontal writing is used, characters may be orientated at 0 or 90 degrees. A value of 0 degrees is basic; a value of 90 degrees is non-basic and its use in a document must be indicated in the document profile. The value of the character orientation is specified at the beginning of the content associated with a basic component by the presentation attribute "character orientation". This value cannot be changed within the content.

6.5.2.8 Emphasis

The following modes of emphasising graphic characters may be distinguished:

- a) normal rendition;
- b) normal intensity;
- c) increased intensity (bold);
- d) italicised;
- e) not italicised;
- f) underlined;
- g) doubly underlined;
- h) not underlined;
- i) crossed-out;
- j) not crossed-out.

All the above modes of emphasis are basic. If no default mode is explicitly specified in the document profile, then the default mode is normal rendition. The mode of emphasis may be specified at the beginning of the content associated with a basic component using the presentation attribute "graphic rendition". The mode may be changed anywhere within the content using the control function SGR. The mode of emphasis remains in effect within the content associated with a basic component until changed into a mutually exclusive mode or by the specification of 'normal rendition'. Mutually exclusive modes are normal/increased intensity, italicized/not italicized, underlined/not underlined and crossed out/not crossed-out. One mode from each mutually exclusive set may be in operation at any point in the document content. Normal rendition cancels the effect of all methods of emphasis that are currently in operation and specifies that the text should be displayed in accordance with the default rendition parameters set for the presentation device. Thus, for example, if it is required to ensure that the content is not underlined, then it is necessary to

explicitly specify that underlined is not to be used.

6.5.2.9 Tabulation

Tabulation stop positions may be specified at any character position along the character path. Each stop is specified by means of the following:

- a) The tabulation position relative to the margin position in the direction opposite to the character path;
- b) An alignment qualifier that specifies the type of alignment to be used at the designated tabulation position. The type may be specified as one of the following:

start aligned;

end aligned;

centered;

aligned around.

These alignment qualifiers are defined in ISO 8613-6. If the alignment qualifier is not explicitly specified, then it is assumed that start aligned is to be used. Only one set of tabulation stops can be specified to be applicable to the content associated with a basic component. No limit is placed on the number of tabulation stops that can be specified within a given set. The set of tabulation stop positions associated with the content of a basic component are specified using the presentation attribute "line layout table". Tabulation stop positions are invoked within the content using the control function STAB.

6.5.2.10 Alignment

This feature is concerned with how the first and last characters on each line of character content is to be laid out during the formatting process. The following values of alignment may be specified:

- a) start aligned;
- b) end aligned;
- c) centred;
- d) justified.

The semantics of these values are as defined in ISO 8613-6. The presentation attribute "alignment" is used to specify the alignment that is applicable to the content associated with a basic component. The alignment value cannot be changed within a content portion.

6.5.2.11 Fonts

Any number of fonts may be used within a document. The fonts used in a particular document are specified in the document profile using the attribute "font list". Further information concerning the specification of font references in the document profile is given in Annex B. The fonts that may be used within the content associated with each basic component are specified by the presentation attribute "character fonts". Up to 10 fonts taken from the list specified by the attribute "font list" may be specified by the attribute "character fonts". The font to be used at the start of the content associated with a basic component is specified using the attribute "graphic rendition". The fonts used within the content may be changed using the control function *SGR*.

6.5.2.12 Reverse character strings

Bi-directional writing is supported by this profile. Hence, a string of characters in a content portion associated with a basic component may be specified to be imaged in the reverse direction of the immediately preceding character string. Such strings can be specified by the control function *SRS* as defined in ISO 8613-6. This control function is provided for cases in which the text belongs to different languages and the character content is written, for example, from left to right or from right to left within the same line of characters, dependent upon the language and/or character set being used.

NOTE - The use of this control function cannot be indicated in the document profile. Thus it is intended that implementations should ignore this control function when reverse character string layout and presentation is not supported.

6.5.2.13 Superscripts and subscripts

Superscripts and subscripts may be specified anywhere within the content associated with a basic component by using the control functions 'PLU' and 'PLD'. The use of these control functions shall be in accordance with ISO 8613-6.

6.5.2.14 Substitution of characters

The control function 'SUB' is provided to represent characters produced by a local system that cannot be represented by a character within a character set supported by this profile.

6.5.2.15 Use of control functions

The following is a list of all the control functions and parameter values (where applicable) that may be specified in character content:

- a) SHS - set horizontal spacing;
- b) SCS - set character spacing;
- c) SVS - set vertical spacing;

- d) SLS - set line spacing;
- e) SGR - set graphic rendition;
- f) STAB - selective tabulation (allowed parameter values: any);
- g) SRS - start reverse string (allowed parameters: any);
- h) PLD - partial line down;
- i) PLU - partial line up;
- j) SUB - substitute character;
- k) SP - space;
- l) CR - carriage return;
- m) LF - line feed;
- n) - code extension control functions .

6.5.3 Geometric graphics content

The formatted processable graphics content is permitted in this DAP for use in either the original image or in the revision annotation of that image. Such geometric graphics content is encoded as CGM (Computer Graphics Metafile) metafiles in accordance with ISO 8632 and ISO 8613-8. Each CGM figure must consist of a single picture only.

Further information concerning the specification of geometric graphics content information is given in Annex B.

6.6 Miscellaneous features

6.6.1 Resource documents

A GenericBlock may refer to a corresponding constituent in a resource document. The GenericBlock in the resource document may refer to content portions and to presentation styles that are contained within the resource document. These are the only constituents that may appear in a resource document.

6.6.2 Application comments

Specification of the attribute "application comments" is optional. When used in conjunction with the "type of coding" of 'tiled encoding', it contains a sequence of positive integers, one for each tile in the content portion. The sequence of integers is a set of indices representing the octet offsets to the beginning of the respective tiles, starting from the beginning of the "content information". A tile index of zero(0) indicates that the respective tile is null. The integers will be sequenced in the same order as the tiles. The tiles will be sequenced primarily in the pel path and secondarily in the line progression direction as defined by the presentation attributes.

6.7 Document management features

Every document interchanged in accordance with this DAP must include a document profile containing information which relates to the document as a whole.

The features specified by the document profile are listed below. A definition of the information contained in these features is given in the corresponding attribute definitions in ISO 8613-4.

Document constituent information:

- a) specific layout structure;
- b) generic layout structure;
- c) presentation styles (optional);
- d) resource document information (optional).

Document characteristics:

- a) document application profile;
- b) document application profile defaults;
- c) document architecture class;
- d) content architecture class;
- e) interchange format class;
- f) ODA version date;
- g) raster graphics content defaults.

Non-basic document characteristics:

- a) page dimensions;

- b) medium type;
- c) raster graphics presentation features.

Document management attributes:

- a) document description (see note 1);
- b) dates and times;
- c) originators;
- d) other user information;
- e) external references;
- f) local file references;
- g) content attributes;
- h) security information.

NOTE - The document description includes the specification of the document reference.

The attributes applicable to the document profile are defined in table D.3, Annex D.

7 Specification of constituent constraints

7.1 Document profile constraints

7.1.1 Macro definitions

-- General macros --

DEFINE(FDA, "{formatted}")

DEFINE(DAC,"DocumentProfile (Document-architecture-class)")

DEFINE(FC," ASN.1{2 8 2 6 0}") -- Character formatted --

DEFINE(FPR," ASN.1{2 8 2 7 2}") -- Raster graphics formatted processable --

DEFINE(FPG," ASN.1{2 8 2 8 0}") -- Geometric graphics formatted processable --

-- Basic page dimensions. --

DEFINE(BasicPageDimension,"

REQ #horizontal-dimension {REQ #fixed-dimension { 1..9240 }},

REQ #vertical-dimension {REQ #fixed-dimension { 1..12400 }},

| REQ #horizontal-dimension {REQ #fixed-dimension { 1..12400 }},

REQ #vertical-dimension {REQ #fixed-dimension { 1..9240 }},

")

-- Any size equal to or smaller than CARA (Common Assured Reproduction Area) of ISO A4 and NA A. Both Portrait and Landscape may be specified. --

-- Non-basic page dimensions. --

DEFINE(NonBasicPageDimensions,"

{REQ #horizontal-dimension {REQ #fixed-dimension {1..39680}},

REQ #vertical-dimension {REQ #fixed-dimension {12401..56120}}}

| {REQ #horizontal-dimension {REQ #fixed-dimension {9241..39680}},

REQ #vertical-dimension {REQ #fixed-dimension {1..56120}}}

-- up to ISO A0 portrait --

| {REQ #horizontal-dimension {REQ #fixed-dimension {1..56120}},

REQ #vertical-dimension {REQ #fixed-dimension {9241..39680}}}

| {REQ #horizontal-dimension {REQ #fixed-dimension {12401..56120}},

REQ #vertical-dimension {REQ #fixed-dimension {1..39680}}}

-- up to ISO A0 landscape --

| {REQ #horizontal-dimension {REQ #fixed-dimension {1..48000}},

REQ #vertical-dimension {REQ #fixed-dimension {12401..211200}}}

| {REQ #horizontal-dimension {REQ #fixed-dimension {9241..48000}},

REQ #vertical-dimension {REQ #fixed-dimension {1..211200}}}

-- up to ANSI J/K portrait --

| {REQ #horizontal-dimension {REQ #fixed-dimension {1..211200}},

REQ #vertical-dimension {REQ #fixed-dimension {9241..48000}}}

| {REQ #horizontal-dimension {REQ #fixed-dimension {12401..211200}},

REQ #vertical-dimension {REQ #fixed-dimension {1..48000}}}

-- up to ANSI J/K landscape --

| {REQ #horizontal-dimension {REQ #fixed-dimension {1..12141}},

REQ #vertical-dimension {REQ #fixed-dimension {12401..17196}}}

| {REQ #horizontal-dimension {REQ #fixed-dimension {9241..12141}},

REQ #vertical-dimension {REQ #fixed-dimension {1..17196}}}

-- up to Japanese legal portrait --

| {REQ #horizontal-dimension {REQ #fixed-dimension {1..17196}},

REQ #vertical-dimension {REQ #fixed-dimension {9241..12141}}}

| {REQ #horizontal-dimension {REQ #fixed-dimension {12401..17196}},

REQ #vertical-dimension {REQ #fixed-dimension {1..12141}}}

-- up to Japanese legal landscape --

| {REQ #horizontal-dimension {REQ #fixed-dimension {13200}},

REQ #vertical-dimension {REQ #fixed-dimension {>= 16801}}}

-- Any portrait size larger than the typical foldout size (11 in x 14 in) including 11 Inch roll paper. --

| {REQ #horizontal-dimension {REQ #fixed-dimension {>= 16801}},

REQ #vertical-dimension {REQ #fixed-dimension {13200}}}

- Any landscape size larger than the typical foldout size (14 in x 11 in) including 11 inch roll paper --")

DEFINE(PermissiblePageDimensions,"

```
{REQ #horizontal-dimension {REQ #fixed-dimension {1..39680}},
REQ #vertical-dimension {REQ #fixed-dimension {1..56120}}}}
-- up to ISO A0 portrait --
| {REQ #horizontal-dimension {REQ #fixed-dimension {1..56120}},
REQ #vertical-dimension {REQ #fixed-dimension {1..39680}}}}
-- up to ISO A0 landscape --
| {REQ #horizontal-dimension {REQ #fixed-dimension {1..48000}},
REQ #vertical-dimension {REQ #fixed-dimension {1..211200}}}}
-- up to ANSI J/K portrait --
| {REQ #horizontal-dimension {REQ #fixed-dimension {1..211200}},
REQ #vertical-dimension {REQ #fixed-dimension {1..48000}}}}
-- up to ANSI J/K landscape --
| {REQ #horizontal-dimension {REQ #fixed-dimension {1..12141}},
REQ #vertical-dimension {REQ #fixed-dimension {1..17196}}}}
-- up to Japanese legal portrait --
| {REQ #horizontal-dimension {REQ #fixed-dimension {1..17196}},
REQ #vertical-dimension {REQ #fixed-dimension {1..12141}}}}
-- up to Japanese legal landscape --
```

")

DEFINE(NominalPageSizes,"

-- ISO Page Sizes --

```
REQ #horizontal-dimension {7015}, REQ #vertical-dimension {9920}
-- ISO A5 Portrait --
| REQ #horizontal-dimension {9920}, REQ #vertical-dimension {7015}
-- ISO A5 Landscape --
| REQ #horizontal-dimension {9920}, REQ #vertical-dimension {14030}
-- ISO A4 Portrait --
| REQ #horizontal-dimension {14030}, REQ #vertical-dimension {9920}
-- ISO A4 Landscape --
| REQ #horizontal-dimension {14030}, REQ #vertical-dimension {19840}
-- ISO A3 Portrait --
| REQ #horizontal-dimension {19840}, REQ #vertical-dimension {14030}
-- ISO A3 Landscape --
| REQ #horizontal-dimension {19840}, REQ #vertical-dimension {28060}
-- ISO A2 Portrait --
| REQ #horizontal-dimension {28060}, REQ #vertical-dimension {19840}
-- ISO A2 Landscape --
| REQ #horizontal-dimension {28060}, REQ #vertical-dimension {39680}
-- ISO A1 Portrait --
| REQ #horizontal-dimension {39680}, REQ #vertical-dimension {28060}
-- ISO A1 Landscape --
| REQ #horizontal-dimension {39680}, REQ #vertical-dimension {56120}
-- ISO A0 Portrait --
```


| REQ #horizontal-dimension {56120}, REQ #vertical-dimension {39680}
-- ISO A0 Landscape --

-- ANSI Page Sizes --

| REQ #horizontal-dimension {10200}, REQ #vertical-dimension {13200}
-- ANSI A Portrait --

| REQ #horizontal-dimension {13200}, REQ #vertical-dimension {10200}
-- ANSI A Landscape --

| REQ #horizontal-dimension {10200}, REQ #vertical-dimension {16800}
-- ANSI Legal Portrait --

| REQ #horizontal-dimension {16800}, REQ #vertical-dimension {10200}
-- ANSI Legal Landscape --

| REQ #horizontal-dimension {13200}, REQ #vertical-dimension {20400}
-- ANSI B Portrait --

| REQ #horizontal-dimension {20400}, REQ #vertical-dimension {13200}
-- ANSI B Landscape --

| REQ #horizontal-dimension {20400}, REQ #vertical-dimension {26400}
-- ANSI C Portrait --

| REQ #horizontal-dimension {26400}, REQ #vertical-dimension {20400}
-- ANSI C Landscape --

| REQ #horizontal-dimension {26400}, REQ #vertical-dimension {40800}
-- ANSI D Portrait --

| REQ #horizontal-dimension {40800}, REQ #vertical-dimension {26400}
-- ANSI D Landscape --

| REQ #horizontal-dimension {40800}, REQ #vertical-dimension {52800}
-- ANSI E Portrait --

| REQ #horizontal-dimension {52800}, REQ #vertical-dimension {40800}
-- ANSI E Landscape --

| REQ #horizontal-dimension {33600}, REQ #vertical-dimension {48000}
-- ANSI F Portrait --

| REQ #horizontal-dimension {48000}, REQ #vertical-dimension {33600}
-- ANSI F Landscape --

| REQ #horizontal-dimension {13200}, REQ #vertical-dimension {108000}
-- ANSI G Portrait --

| REQ #horizontal-dimension {108000}, REQ #vertical-dimension {13200}
-- ANSI G Landscape --

| REQ #horizontal-dimension {33600}, REQ #vertical-dimension {171600}
-- ANSI H Portrait --

| REQ #horizontal-dimension {171600}, REQ #vertical-dimension {33600}
-- ANSI H Landscape --

| REQ #horizontal-dimension {40800}, REQ #vertical-dimension {211200}
-- ANSI J Portrait --

| REQ #horizontal-dimension {211200}, REQ #vertical-dimension {40800}
-- ANSI J Landscape --

| REQ #horizontal-dimension {48000}, REQ #vertical-dimension {171600}
-- ANSI K Portrait --

| REQ #horizontal-dimension {171600}, REQ #vertical-dimension {48000}
-- ANSI K Landscape --

-- Foldouts --

```
| REQ #horizontal-dimension {13200}, REQ #vertical-dimension {16800}
      -- Foldout Portrait --
| REQ #horizontal-dimension {16800}, REQ #vertical-dimension {13200}
      -- Foldout Landscape --
| REQ #horizontal-dimension {13200}, REQ #vertical-dimension {> = 16801}
-- Any portrait size larger than the typical foldout size (11 In x 14 In) including 11 Inch roll paper --
| REQ #horizontal-dimension {> = 16801}, REQ #vertical-dimension {13200}
-- Any landscape size larger than the typical foldout size (14 In x 11 In) including 11 Inch roll paper --
")
```

-- Macro defining permissible code extension announcers --

```
DEFINE(CDEXTEN, " ESC 02/00 05/00,      -- LSR --
      [ESC 02/00 05/03],      -- LSR1 --
      [ESC 02/00 05/05],      -- LSR2 --
      [ESC 02/00 05/07],      -- LSR3 --
      [ESC 02/00 05/10],      -- SS2 --
      [ESC 02/00 05/11]      -- SS3 --
")
```

-- Macro defining permitted graphic renditions --

```
DEFINE(GRAPHICRENDITIONS "
      {'cancel' | 'increased-intensity'
      | 'italicised' | 'underlined' | 'crossed-out'
      | 'primary-font' | 'first-alternative-font'
      | 'second-alternative-font' | 'third-alternative-font'
      | 'fourth-alternative-font' | 'fifth-alternative-font'
      | 'sixth-alternative-font' | 'seventh-alternative-font'
      | 'eighth-alternative-font' | 'ninth-alternative-font'
      | 'doubly-underlined' | 'normal-intensity'
      | 'not-italicised' | 'not-underlined' | 'not-crossed-out'}...
")
```

-- Macros defining final character for designation --

```
DEFINE(FCORE, "04/02 -- the 94 characters of the IRV of ISO 646
      (revised 1990) (i.e., ASCII) --")
```

```
DEFINE(F646, "-- a final character designating any version of ISO 646
      except 04/02 --")
```

```
DEFINE(F94S, "-- a final character designating any registered 94 single
      byte graphic character set --")
```


PART 22 - ODA Image DAP

December 1992 (Stable)

DEFINE(F94M, "-- a final character designating any registered 94 multi
byte graphic character set --")

DEFINE(F96S, "-- a final character designating any registered 96 single
byte graphic character set --")

DEFINE(F96M, "-- a final character designating any registered 96 multi
byte graphic character set --")

DEFINE(FEMPTY, "07/14 -- the empty set --")

-- Macros defining designation sequences --

DEFINE(DEG-CORE-GO, "ESC 02/08 \$FCORE")
-- Designate the 94 characters of the IRV of
ISO 646 to G0 --

DEFINE(DEG-646-GO, "ESC 02/08 \$F646")
-- Designate any version of ISO 646, except 04/02,
to G0 --

DEFINE(DEG-ANY-G1, "{ESC 02/09 \$F94S
|ESC 02/04 02/09 \$F94M
|ESC 02/13 \$F96S
|ESC 02/04 02/13 \$F96M}")
-- Designate any character set to G1 --

DEFINE(DEG-ANY-G2, "{ESC 02/10 \$F94S
|ESC 02/04 02/10 \$F94M
|ESC 02/14 \$F96S
|ESC 02/04 02/14 \$F96M}")
-- Designate any character set to G2 --

DEFINE(DEG-ANY-G3, "{ESC 02/11 \$F94S
|ESC 02/04 02/11 \$F94M
|ESC 02/15 \$F96S
|ESC 02/04 02/15 \$F96M}")
-- Designate any character set to G3 --

DEFINE(DEG-EMPTY-G1, "ESC 02/09 \$FEMPTY")
-- Designate the empty set to G1 --

-- Macros defining shift functions --

DEFINE(LSO, "00/15") -- locking shift invoking G0 to GL --

DEFINE(LS1R, "ESC 07/14") -- locking shift invoking G1 to GR --

PART 22 - ODA Image DAP

December 1992 (Stable)

DEFINE(LS2R, "ESC 07/13") -- locking shift Invoking G2 to GR --

DEFINE(LS3R, "ESC 07/14") -- locking shift Invoking G3 to GR --

DEFINE(SS2, "08/14") -- single shift Invoking G2 to GL --

DEFINE(SS3, "08/15") -- single shift Invoking G3 to GL --

-- Macro defining permissible graphic character sets. --

```
DEFINE(PERMIT-GRCHAR, " {$DEG-CORE-G0 $LS0
    |$DEG-646-G0 $LS0},
    {$DEG-ANY-G1 $LS1R
    |$DEG-ANY-G2 $LS2R
    |$DEG-ANY-G3 $LS3R}...
    |{$DEG-EMPTY-G1 $LS1R} ")
```

-- Macro defining default graphic character sets --

DEFINE(DAP-DEFAULT-GRCHAR, "\$PERMIT-GRCHAR")

-- Macro defining basic character sets. Note that this macro is defined for clarification of the specification and is not to be used in any other part of this DAP specification. --

```
DEFINE(BASIC-GRCHAR, " $DEG-CORE-G0 $LS0,
    $DEG-EMPTY-G1 $LS1R ")
```

-- Macro defining non-basic character sets --

```
DEFINE(NON-BASIC-GRCHAR, " {$DEG-646-G0
    |$DEG-ANY-G1
    |$DEG-ANY-G2
    |$DEG-ANY-G3}... ")
```

-- Macro defining character sets used in document profile attributes --

```
DEFINE(PROFCHAR, " {$DEG-CORE-G0 $LS0,
    |$DEG-646-G0 $LS0},
    {$DEG-ANY-G1 $LS1R
    |$DEG-ANY-G2 $LS2R
    |$DEG-ANY-G3 $LS3R
    |$DEG-EMPTY-G1 $LS1R} ")
```

PART 22 - ODA Image DAP

December 1992 (Stable)

-- Macro defining comments character sets --

```
DEFINE(COMCHAR, " {ESC 02/00 05/00,      -- LS0 --
                [ESC 02/00 05/03],      -- LSR1 --
                [ESC 02/00 05/05],      -- LSR2 --
                [ESC 02/00 05/07],      -- LSR3 --
                [ESC 02/00 05/10],      -- SS2 --
                [ESC 02/00 05/11]},      -- SS3 --
                {$DEG-CORE-G0 [LS0]
                |$DEG-646-G0 [LS0]},
                {{$DEG-ANY-G1 [$LS1R]
                |$DEG-ANY-G2 [$LS2R]
                |$DEG-ANY-G3 [$LS3R]}...
                |$DEG-EMPTY-G1 $LS1R}} ")
```

-- Macro defining character sets used for alternative representation --

```
DEFINE(ALTCHAR, "$PROFCHAR")
```

7.1.2 Constituent constraints

7.1.2.1 DocumentProfile

{

-- Presence of document constituents --

REQ	Specific-layout-structure	{'present'},
PERM	Generic-layout-structure	{'factor-set'},
PERM	Presentation-styles	{'present'},
PERM	Resource-document	{ANY_VALUE},
PERM	Resources	{MUL {REQ #resource-identifier {ANY_VALUE},
		REQ #resource-object-class-identifier {ANY_VALUE}}},

-- Document characteristics --

REQ	Document-application-profile	{-- See clause 8 for a definition of the permitted values for this attribute. --},
-----	------------------------------	--

REQ	Document-application-profile-defaults	{
-----	---------------------------------------	---

-- Document architecture defaults --

REQ	#content-architecture-class	{\$FPR},
PERM	#dimensions	{\$PermissiblePageDimensions},
PERM	#medium-type	{

PERM #nominal-page-size {\$NominalPageSizes},
 PERM #side-of-sheet {ANY_VALUE}},

-- Any permitted medium type. Both landscape and portrait may be specified. --

REQ #type-of-coding {ASN.1 {2 8 3 7 0} -- T6 encoding --
 | ASN.1 {2 8 3 7 5} -- tiled encoding --
 | ASN.1 {2 8 3 7 6} -- T6 encoding - MSB -- },

PERM #page-position {ANY_VALUE},

PERM raster-graphics-contents-defaults {
 PERM #pel-path {ANY_VALUE},
 PERM #line-progression {ANY_VALUE},
 PERM #pel-spacing {REQ #length {ANY_VALUE},
 REQ #pel-spaces {ANY_VALUE}},

PERM #spacing-ratio
 {REQ #line-spacing-value {ANY_VALUE},
 REQ #pel-spacing-value {ANY_VALUE}},

PERM #compression {ANY_VALUE}},

PERM #geometric-graphics-content-defaults {ANY_VALUE},

PERM #character-content-defaults {
 PERM #alignment {ANY_VALUE},
 PERM #character-spacing {ANY_VALUE},
 PERM #character-fonts {ANY_VALUE},
 PERM #character-orientation {'0-degrees' | '90-degrees'},
 PERM #character-path {'0-degrees' | '90-degrees' | '180-degrees' |
 '270-degrees'},

PERM #code-extension-announcers {\$CDEXTEN},

PERM #graphic-character-sets {\$PERMIT-GRCHAR},

PERM #graphic-character-subrepertoire {ANY_VALUE},

PERM #graphic-rendition {\$GRAPHICRENDITIONS},

PERM #line-progression {'90-degrees' | '270-degrees'},

PERM #line-spacing {ANY_VALUE},

PERM #line-layout-table {ANY_VALUE}},

-- End of document architecture defaults --

REQ Document-architecture-class {\$FDA},

REQ Content-architecture-classes {{\$FPR | \$FPG | \$FC}...},

REQ Interchange-format-class {-- This attribute required only for ODIF interchange. See
 clause 8 for a definition of the permitted values for this
 attribute. --},

REQ ODA-version

{REQ #standard-or-recommendation {'ISO 8613'},
 REQ #publication-date {'1991-12-31'}},

-- This date represents the date that this DAP was approved. This is the only
 -- approved value, however, the date will be changed if the DAP is significantly
 -- revised. If the date is revised, use of the new date is required only when the
 -- additional functionality is being used. That is, legacy products may continue to

PART 22 - ODA Image DAP

December 1992 (Stable)

-- support the earlier DAP.

-- Non-basic document characteristics --

```
PERM Profile-character-sets      {$PROFCHAR},
PERM Comments-character-sets    {$COMCHAR},
PERM Alternative-representation-character-sets {$ALTCHAR},
PERM Page-dimensions            {MUL {$NonBasicPageDimensions}},
PERM Medium-types               {MUL{
  PERM #nominal-page-size      {$NominalPageSizes},
  PERM #side-of-sheet          {ANY_VALUE}}},
  -- All values of "medium type" are non-basic --
PERM Coding-attributes          {
  REQ #raster-graphics-coding-attributes {
    REQ #compression {'uncompressed'}}},
PERM Presentation-features      {
  PERM #character-presentation-features { MUL {
    | PERM #character-orientation      {'90-degrees'}
    | PERM #character-path              {'90-degrees', '180-degrees', '270-degrees'}
    | PERM #graphic-character-sets      {ANY_VALUE} EXCEPT {$BASIC-GRCHAR}
    | PERM #graphic-character-subrepertoire {>0}
    | PERM #line-spacing                {ANY_VALUE} EXCEPT {150,200,300,400}
    | PERM #line-progression            {'90-degrees'}}}
  PERM #Raster-graphics-presentation-features { MUL {
    | PERM #pel-path                    {'180-degrees' |
                                          '270-degrees'}
    | PERM #line-progression            {'90-degrees'}
    | PERM #pel-spacing                 {REQ #length {ANY_VALUE} EXCEPT {16,12,8,6,5,4,3,2,1},
                                          REQ #pel-spaces {ANY_VALUE} EXCEPT {1}}
    | PERM #spacing-ratio               {REQ #line-spacing-value {ANY_VALUE} EXCEPT {1},
                                          REQ #pel-spacing-value {ANY_VALUE} EXCEPT {1}}}}},
```

-- End of Non-basic characteristics --

-- Additional document characteristics --

```
PERM Fonts-list      {MUL {REQ #font-identifier {ANY_VALUE},
                          REQ #font-reference {ANY_VALUE}}},
```

-- The format of the parameter "font-reference" is defined in annex B --

-- Document management attributes --

-- Document description --

```
PERM Title      {ANY_STRING},
PERM Subject    {ANY_STRING},
PERM Document-type {ANY_STRING},
PERM Abstract   {ANY_STRING},
```

PART 22 - ODA Image DAP

December 1992 (Stable)

PERM	Keywords	{ANY_VALUE},
REQ	Document-reference	{ANY_VALUE},
- Dates and times -		
PERM	Document-date-and-time	{ANY_STRING},
PERM	Creation-date-and-time	{ANY_STRING},
PERM	Local-filing-date-and-time	{ANY_STRING},
PERM	Expiry-date-and-time	{ANY_STRING},
PERM	Start-date-and-time	{ANY_STRING},
PERM	Purge-date-and-time	{ANY_STRING},
PERM	Release-date-and-time	{ANY_STRING},
PERM	Revision-history	{ANY_VALUE},
--Originators -		
PERM	Organizations	{ANY_STRING},
PERM	Preparers	{ANY_VALUE},
PERM	Owners	{ANY_VALUE},
PERM	Authors	{ANY_VALUE},
- Other user information -		
PERM	Copyright	{ANY_VALUE},
PERM	Status	{ANY_STRING},
PERM	User-specific-codes	{ANY_STRING},
PERM	Distribution-list	{ANY_VALUE},
PERM	Additional-Information	{ANY_VALUE},
- External references -		
PERM	References-to-other-documents	{ANY_VALUE},
PERM	Superseded-documents	{ANY_VALUE},
- Local file references -		
PERM	Local-file-references	{ANY_VALUE},
- Content attributes -		
PERM	Document-size	{ANY_VALUE},
PERM	Number-of-pages	{ANY_INTEGER},
PERM	Languages	{ANY_STRING},
- Security Information --		
PERM	Authorization	{ANY_VALUE},
PERM	Security-classification	{ANY_STRING},
PERM	Access-rights	{ANY_STRING}
}		

7.2 Logical constituent constraints

No logical constituents applicable in this clause.

7.3 Layout constituent constraints

7.3.1 Macro definitions

```

DEFINE(CHAR,"      CONTENT_ID_OF(Character-content-portion)")
DEFINE(RAST,"      CONTENT_ID_OF(Raster-graphics-content-portion)")
DEFINE(GEOM,"      CONTENT_ID_OF(Geometric-graphics-content-portion)")

```

7.3.2 Factor constraints

```

FACTOR:      ANY-LAYOUT      {

SPECIFIC:
PERM Object-type              {VIRTUAL},
REQ  Object-identifier        {ANY_VALUE},
PERM Subordinates             {VIRTUAL},
PERM User-visible-name        {ANY_VALUE},
PERM User-readable-comments   {ANY_VALUE},
}

```

7.3.3 Constituent constraints

7.3.3.1 DocumentLayoutRoot

```

DocumentLayoutRoot: ANY-LAYOUT {

SPECIFIC:
REQ  Object-type              { 'document-layout-root' },
REQ  Subordinates             { SUB_ID_OF(CompositePage) + }
}

```

7.3.3.2 CompositePage

```

CompositePage:      ANY-LAYOUT {

SPECIFIC:

```

PART 22 - ODA Image DAP

December 1992 (Stable)

REQ	Object-type	{'page'},
REQ	Subordinates	{SUB_ID_OF(OriginalImage),
		{SUB_ID_OF(RevisionAnnotation) + }},
PERM	Dimensions	{\$PermissiblePageDimensions},
PERM	Page-position	{ANY_VALUE},
PERM	Medium-type	{REQ #nominal-page-size {\$NominalPageSizes},
		REQ #side-of-sheet {ANY_VALUE}},
PERM	Imaging-order	{ANY_VALUE},
PERM	Application-comments	{ANY_VALUE}
		}

7.3.3.3 OriginalImage

OriginalImage:	ANY-LAYOUT	{
SPECIFIC:		
REQ	Object-type	{'frame'},
REQ	Subordinates	{SUB_ID_OF(SpecificBlock) + },
PERM	Position	{REQ #fixed-position
		{REQ #horizontal-position {ANY_VALUE},
		REQ #vertical-position {ANY_VALUE}}},
PERM	Dimensions	{REQ #horizontal-dimension
		{REQ #fixed-dimension{ANY_VALUE}},
		REQ #vertical-dimension
		{REQ #fixed-dimension{ANY_VALUE}}},
PERM	Application-comments	{ANY_VALUE}
		}

7.3.3.4 RevisionAnnotation

RevisionAnnotation:	ANY-LAYOUT	{
SPECIFIC:		
REQ	Object-type	{'frame'},
REQ	Subordinates	{SUB_ID_OF(SpecificBlock)},
PERM	Position	{REQ #fixed-position
		{REQ #horizontal-position {ANY_VALUE},
		REQ #vertical-position {ANY_VALUE}}},
PERM	Dimensions	{REQ #horizontal-dimension
		{REQ #fixed-dimension{ANY_VALUE}},
		REQ #vertical-dimension
		{REQ #fixed-dimension{ANY_VALUE}}},
PERM	Application-comments	{ANY_VALUE}}

7.3.3.5 SpecificBlock

SpecificBlock: {

SPECIFIC:

```

REQ  Object-type           {'block'},
REQ  Object-identifier     {ANY_VALUE},
REQ  Content-portions     {$CHAR | $RAST | $GEOM},
PERM Position              {REQ #fixed-position {
                                REQ #horizontal-position {ANY_VALUE},
                                REQ #vertical-position {ANY_VALUE}}},
PERM Dimensions            {REQ #horizontal-dimension
                                {REQ #fixed-dimension{ANY_VALUE}},
                                REQ #vertical-dimension
                                {REQ #fixed-dimension{ANY_VALUE}}},
PERM Object-class          {OBJECT CLASS ID OF(GenericBlock)},
PERM Content-architecture-class {$FC | $FPR | $FPG},
PERM Transparency          {'transparent' | 'opaque'},
PERM Colour                {'colourless' | 'white'},
PERM User-readable-comments {ANY_STRING},
PERM User-visible-name     {ANY_STRING},
PERM Application-comments  {ANY_VALUE},
                                -- See 8.1.3 and 8.2.3 --
PERM Presentation-style    {STYLE_ID_OF(PStyle1) | STYLE_ID_OF(PStyle2) |
                                STYLE_ID_OF(PStyle3),
                                -- PStyle1 for character content, PStyle2 for geometric, & PStyle3 for raster --
PERM Presentation-attributes {

```

CASE SpecificBlock(Content-portions) OF {

{\$CHAR}:

```

    {PERM #character-attributes {
        PERM #alignment          {ANY_VALUE},
        PERM #character-spacing  {ANY_VALUE},
        PERM #character-fonts    {ANY_VALUE},
        PERM #character-orientation {'0-degrees' | '90-degrees'},
        PERM #character-path      {'0-degrees' | '90-degrees' | '180-degrees' |
                                   '270-degrees'},
        PERM #code-extension-announcers {$CDEXTEN},
        PERM #graphic-character-sets {$PERMIT-GRCHAR},
        PERM #graphic-character-subrepertoire {ANY_VALUE},
        PERM #graphic-rendition    {$GRAPHICRENDITIONS},
        PERM #iline-progression    {'90-degrees' | '270-degrees'},
        PERM #iline-spacing        {ANY_VALUE},
        PERM #iline-layout-table   {ANY_VALUE},
    }}

```

{\$RAST}:

```

    {PERM #raster-graphics-attributes {

```

```

    PERM #Pel-path {ANY_VALUE},
    PERM #Line-progression {ANY_VALUE},
    PERM #Pel-spacing {ANY_VALUE},
    PERM #Spacing-ratio {REQ #line-spacing-value {ANY_VALUE},
    REQ #pel-spacing-value {ANY_VALUE}},
    PERM #Clipping {ANY_VALUE}}

```

{ \$GEOM }:

```

    { PERM #geometric-graphics-attributes {
      PERM #picture-dimensions {ANY_VALUE},
      PERM #picture-orientation {ANY_VALUE},
      PERM #text-rendition {PERM #fonts-list {ANY_VALUE},
      PERM #character-set-lists {ANY_VALUE}}}

```

}}}

7.3.3.6 GenericBlock

GenericBlock {

GENERIC:

```

REQ Object-type {'block'},
REQ Content-portions {$CHAR | $RAST | $GEOM},
PERM Position {REQ #fixed-position {
  REQ #horizontal-position {ANY_VALUE}
  REQ #vertical-position {ANY_VALUE}}}

```

```

PERM Dimensions {REQ #horizontal-dimension
  {REQ #fixed-dimension {ANY_VALUE}},
  REQ #vertical-dimension
  {REQ #fixed-dimension {ANY_VALUE}}},
  {ANY_VALUE},

```

```

REQ Object-class-identifier {ANY_VALUE},
PERM Resource {$FC | $FPR | $FPG},
PERM Content-architecture-class {'transparent' | 'opaque'},
PERM Transparency {'colourless' | 'white'},
PERM Colour {ANY_STRING},
PERM User-readable-comments {ANY_STRING},
PERM User-visible-name {ANY_STRING},
PERM Application-comments {ANY_VALUE},

```

-- See 8.2 --

```

PERM Presentation-style {STYLE_ID_OF(PStyle1) | STYLE_ID_OF(PStyle2) |
  STYLE_ID_OF(PStyle3),
  -- PStyle1 for character content, PStyle2 for geometric, & PStyle3 for raster
PERM Presentation-attributes {

```

CASE GenericBlock(Content-portions) OF {

{ \$CHAR }:

```

    { PERM #character-attributes {
      PERM #alignment {ANY_VALUE},

```

```

    PERM #character-spacing      {ANY_VALUE},
    PERM #character-fonts       {ANY_VALUE},
    PERM #character-orientation {'0-degrees' | '90-degrees'},
    PERM #character-path        {'0-degrees' | '90-degrees' | '180-degrees' |
                                '270-degrees'},
    PERM #code-extension-announcers {$CDEXTEN},
    PERM #graphic-character-sets {$PERMIT-GRCHAR},
    PERM #graphic-character-subrepertoire {ANY_VALUE},
    PERM #graphic-rendition      {$GRAPHICRENDITIONS},
    PERM #line-progression       {'90-degrees' | '270-degrees'},
    PERM #line-spacing           {ANY_VALUE},
    PERM #line-layout-table      {ANY_VALUE},

```

```

}}

```

```

{$RAST}:

```

```

    {PERM #raster-graphics-attributes {
        PERM #Pel-path                {ANY_VALUE},
        PERM #Line-progression         {ANY_VALUE},
        PERM #Pel-spacing              {ANY_VALUE},
        PERM #Spacing-ratio            {REQ #line-spacing-value {ANY_VALUE},
                                        REQ #pel-spacing-value {ANY_VALUE}}
        PERM #Clipping                {ANY_VALUE}}}

```

```

{$GEOM}:

```

```

    {PERM #geometric-graphics-attributes {
        PERM #picture-dimensions       {ANY_VALUE},
        PERM #picture-orientation       {ANY_VALUE},
        PERM #text-rendition            {PERM #fonts-list {ANY_VALUE},
                                        PERM #character-set-lists {ANY_VALUE}}}

```

```

}}

```

7.4 Layout style constraints

No layout style constraints applicable in this clause.

7.5 Presentation style constraints

7.5.1 Macro definitions

No macro definitions are applicable to this clause.

7.5.2 Factor constraints

```

FACTOR:      ANY-PRESENTATION-STYLE  {

REQ  Presentation-style-identifier      {ANY_VALUE},
PERM  User-readable-comments            {ANY_STRING},
PERM  User-visible-name                 {ANY_STRING},
}

```

7.5.3 Presentation style constituent constraint**7.5.3.1 PStyle1**

```

PStyle1:      ANY-PRESENTATION-STYLE  {

```

```

    -- This style is used for character content --

```

```

PERM  Presentation-attributes          {
    PERM  #character-attributes         {
        PERM  #alignment                {ANY_VALUE},
        PERM  #character-spacing        {ANY_VALUE},
        PERM  #character-fonts          {ANY_VALUE},
        PERM  #character-orientation    {'0-degrees' | '90-degrees'},
        PERM  #character-path           {'0-degrees' | '90-degrees' | '180-degrees' |
                                          '270-degrees'},
        PERM  #code-extension-announcers {$CDEXTEN},
        PERM  #graphic-character-sets   {$PERMIT-GRCHAR},
        PERM  #graphic-character-subrepertoire {ANY_VALUE},
        PERM  #graphic-rendition        {$GRAPHICRENDITIONS},
        PERM  #line-progression          {'90-degrees' | '270-degrees'},
        PERM  #line-spacing              {ANY_VALUE},
        PERM  #line-layout-table         {ANY_VALUE}}
    }
}

```

7.5.3.2 PStyle2

```

PStyle2:      ANY-PRESENTATION-STYLE  {

```

```

    -- This style is used for geometric graphics content --

```

```

PERM  Presentation-attributes {
    PERM  #geometric-graphics-attributes {
        PERM  #picture-dimensions      {ANY_VALUE},
        PERM  #picture-orientation      {ANY_VALUE},
        PERM  #text-rendition           {PERM #fonts-list{ANY_VALUE},
    }
}

```



```

    PERM #character-set-list{ANY_VALUE}}}}
}

```

7.5.3.3 PStyle3

PStyle3: ANY-PRESENTATION-STYLE {

-- This style is used for raster graphics content --

```

PERM Presentation-attributes {
  PERM #raster-graphics-attributes {
    PERM #pel-path {ANY_VALUE},
    PERM #line-progression {ANY_VALUE},
    PERM #pel-spacing {REQ #length {ANY_VALUE},
                      REQ #pel-spaces {ANY_VALUE}},
    PERM #spacing-ratio {REQ #line-spacing-value {ANY_VALUE},
                       REQ #pel-spacing-value {ANY_VALUE}},
    PERM #clipping {ANY_VALUE}}
}

```

7.6 Content portion constraints

7.6.1 Macro definitions

DEFINE(TILED," ASN.1{2 8 3 7 5}") -- Tiled raster encoding --

7.6.2 Factor constraints

No factor constraints are applicable to this clause.

7.6.3 Constituent constraints

7.6.3.1 Character content portion

```

Character-content-portion {
  REQ Content-identifier-layout {ANY_VALUE},
  PERM Type-of-coding {ASN.1{2 8 3 6 0}},
  PERM Alternative-representation {ANY_STRING},
  PERM Content-information
    {CHARACTER, {#STAB {ANY_VALUE}
                  |#SHS {ANY_VALUE}
                  |#SGR {$GRAPHICRENDITIONS}

```

```

| #SVS {ANY_VALUE}
| #SLS {ANY_VALUE}
| #SCS {ANY_VALUE}
| #SRS {ANY_VALUE}
| #CR
| #LF
| #PLD
| #PLU
| #SP
| #SUB
| #LS0
| #LS1R
| #LS2R
| #LS3R
| #SS2
| #SS3
| # $DEG-CORE-G0
| # $DEG-646-G0
| # $DEG-ANY-G1
| # $DEG-ANY-G2
| # $DEG-ANY-G3
| # $DEG-EMPTY-G1
| ...}

```

```

}

```

7.6.3.2 Raster graphics content portion

Raster-graphics-content-portion {

REQ Content-identifier-layout

PERM Type-of-coding

```

{ ANY_VALUE },
| ASN.1{2 8 3 7 0} -- T.6 encoding --
| ASN.1{2 8 3 7 1} -- T.4 one dimensional --
| ASN.1{2 8 3 7 2} -- T.4 two dimensional --
| ASN.1{2 8 3 7 3} -- bitmap encoding --
| ASN.1{2 8 3 7 5} -- tiled encoding --
| ASN.1{2 8 3 7 6} -- T.6 encoding - MSB --
| ASN.1{2 8 3 7 7} -- T.4 one dimensional - MSB --
| ASN.1{2 8 3 7 8} -- T.4 two dimensional - MSB -- },

```

PERM Coding-attributes

REQ #raster-graphics-coding-attributes

PERM #compression

PERM #number-of-lines

REQ #number-of-pels-per-line

CASE Raster-graphics-content-portion (Type-of-coding) OF {

```

{ $TILED }:
| PERM #number-of-pels-per-tile-line {512},
| PERM #number-of-lines-per-tile {512},
| PERM #tiling-offset {ANY_VALUE},
| PERM #tile-types {'null background' |

```

```

PERM Alternative-representation    {ANY_STRING},
PERM Content-information           {RASTER}
}

```

```

'null foreground' |
'T.6 encoded' |
'bitmap encoded' |
'T.6 encoded - MSB' }}}},

```

7.6.3.3 Geometric graphics content portion

```

Geometric-graphics-content-portion {
REQ Content-identifier-layout      {ANY_VALUE},
PERM Type-of-coding                {ASN.1{2 8 3 8 0}},
PERM Alternative-representation    {ANY_VALUE},
PERM Content-information           {GEOMETRIC}
}

```

7.7 Additional usage constraints

No other usage constraints are currently defined.

8 Interchange format

Two Interchange formats are supported by this profile. The Interchange format ODIF (class A) can be used by applications requiring a binary encoding based on ASN.1. The Interchange Format SDIF can be used by applications requiring a SGML based clear text encoding. This latter interchange format is an SGML application, called Office Document Language (ODL). For the purposes of interchange, the ODL ENTITIES are placed in an ASN.1 wrapper, as defined by SDIF. Each encoding form has inherent advantages. Conversion of document encoded in one Interchange format into the other should not produce the loss of semantic document information.

8.1 Interchange format ODIF (class A)

8.1.1 Interchange format

The value of the document profile attribute "interchange format" for this interchange format is 'if-a'. This form of ODIF is defined in ISO 8613-5.

The encoding is in accordance with the Basic Encoding Rules for Abstract Syntax Notation One (ASN.1), as defined in ISO 8825.

8.1.2 DAP identifier

The value for the document profile attribute "document application profile" for this interchange format is represented by the following object identifier.

Editor's Note - To be supplied.

8.1.3 Encoding of application comments

ISO 8613-5 define the encoding of the attribute "application comments" as an octet string. For SpecificBlock, this DAP requires that the encoding within that octet string be in accordance with the ASN.1 syntax specified in the following module definition.

```
NIST-DAPSpecification
DEFINITIONS ::= BEGIN
EXPORTS Object-Appl-Comm-Encoding;

Object-Appl-Comm-Encoding ::= SEQUENCE OF INTEGER
END
```

8.2 Interchange format SDIF**8.2.1 Interchange format**

The document profile attribute "interchange format" does not apply for this interchange format. The SDIF encoding of ODA is defined in Annex E of ISO 8613-5. In addition, ISO 8613-6, -7, and -8 contain additional specifications for this encoding of ODA.

8.2.2 DAP identifier

The value for this attribute "document application profile" for this interchange format is represented by the following object identifier.

Editor's Note - To be supplied.

8.2.3 Encoding of application comments

For SpecificBlock, the encoding of the attribute "application comments" is defined in a data stream conforming to this profile with the following DTD definition:

<!-- The following set of declarations may be invoked by using a public entity as follows:

<!DOCTYPE odaac Public "-//USA-OIW//DTD SGML ENCODING ODA APPLICATION COMMENTS//EN">

-->

<i-- NOTE: To parse the following Document Type Declaration Subset, place the Document Type declaration" <!DOCTYPE odaac [" at the beginning of the file and "]">" at the end of the file. -->

<!ELEMENT odaac - - (objappc)+ >

<i-- Object application comment -->

<!ELEMENT objappc - O (#PCDATA)>

8.3 Encoding of raster content information

The encoding of raster content information in the bitmap encoding scheme is that specified in 9.3 of the raster graphics content architecture part of ISO 8613-7, that is, the first pel in the order of bits is allocated to the most significant bit of an octet. The encoding of the code words in the CCITT Recommendation T.4 and T.6 encoding schemes may be done in either the up or down bit order. The bit order is specified by the attributes "type of coding" or "tile types". The attribute "tile types" is used only when the value for "type of coding" is 'tiled encoded'. For the up order, it is such that the first or only bit of the first code word shall be placed in the least significant bit of the first octet. Subsequent bits of the first and following code words are placed in the direction of more significant bits in the first and following octets. For the down order, it is such that the first or only bit of the first code word shall be placed in the most significant bit (MSB) of the first octet. Subsequent bits of the first and following code words are placed in the direction of least significant bits in the first and following octets.

Annex A (normative)

Amendments and corrigenda

A.1 Amendments

A.1.1 Amendments to the base standard

The amendments applicable to this DAP includes the ISO 8613 - Amendment 1: 1990. This amendment includes text to be included in ISO 8613-1 as the following annexes:

- a) Annex E: Use of ISO/IEC 10021 (MOTIS) to Interchange documents conforming to ISO 8613;
- b) Annex F: Document application profile proforma and notation;
- c) Annex G: Conformance testing methodology;
- d) Annex H: Recording of documents conforming to ISO 8613 on flexible disk cartridges conforming to ISO 9293.

In addition, this amendment addresses the inclusion of the ISO 8613 Technical Corrigenda 1.

This DAP does not include the following features of the amendment:

- a) Addendum on security;
- b) Addendum on styles;
- c) Addendum on alternative representation.

Additionally, this DAP includes features from the Tiled Raster Graphics Addendum to ISO 8613-7, ISO/IEC JTC1/SC18/WG5 901, dated September 1990, and the Additional Bit Order Mapping Addendum to CCITT Rec. T.417/ISO 8613-7, ISO/IEC JTC 1/WG 3, dated July 1991. A new version of ISO 8613-7 which also will incorporate the Colour Addendum is scheduled to be issued in 1993.

A.2 Corrigenda

A.2.1 Corrigenda to this DAP

There are no corrigenda to this DAP.

Annex B (informative)

Recommended practices**B.1 Transfer methods for ODA****B.1.1 Conveyance of ODA over CCITT X.400-1984**

This recommendation describes how ODA body parts are to be encoded for transmission over a CCITT X.400-1984 service.

An ODA body part is encoded as OdaBodyPart in the definition given below:

```
OdaBodyPart ::= SEQUENCE { OdaBodyPartParameters, OdaData }
OdaBodyPartParameters ::= SET {
    document-application-profile
        [0] IMPLICIT OBJECT IDENTIFIER,
    document-architecture-class
        [1] IMPLICIT INTEGER {
            formatted (0),
            processable (1),
            formatted-processable (2) }
OdaData ::= SEQUENCE OF Interchange-Data-Element
```

NOTE - It is recommended to transfer an ODA document as a single body part with tag 12:

Oda [12] IMPLICIT OCTETSTRING

The content of the octet string is encoded as OdaBodyPart, defined above. However, this is out of the scope of this profile.

B.1.2 Conveyance of ODA over FTAM

This recommendation describes the File Transfer, Access, and Management (FTAM) Document Type to be used for minimal storage and transfer capabilities of ODA data streams. It is recognized that enhanced capabilities may at some point be added.

When using FTAM to transfer an ODA file, the FTAM-3, "ISO FTAM Unstructured Binary", document type should be specified. However, since files that do not contain ODA data streams can have the same document type, it is left up to the user of application programs that remotely access files using FTAM to know that a given file contains an ODA data stream.

B.1.3 Conveyance of ODA over DTAM

This recommendation provides for information concerning the interchange of ODA based documents with Document Transfer and Manipulation (DTAM) protocols.

DTAM is defined in the T.430-Series of recommendations and is, like ODA, an integral part of the T.400-Series of CCITT Recommendations named *Open Document Architecture, Transfer and Manipulation*.

The T.520-Series of recommendations contain *Communication Application Profiles (CAP)*. Recommendation T.522 describes the Communication Application Profile BT1 for document bulk transfer. Recommendation T.522 is applicable for the Office Document Format Profile (FOD) published in this ISP.

NOTE - The use of BT1 within the end-to-end oriented Telematic Services Telefax 4 and Teletex is described in 7.1 of Recommendation T.561 and 7.1 of Recommendation T.562.

B.1.4 Conveyance of ODA over flexible disks

The recommended method for interchanging ODA documents between systems by the exchange of magnetically recorded Flexible Disk Cartridges is by the use of an annex to ISO 8613-1 (to be published), *Recoding of Documents Conforming to ISO 8613 on Flexible Cartridges Conforming to ISO 9293*. This annex provides for recording each ODA document as a separate file as defined by ISO 9293, *Volume and File Structure of Flexible Disk Cartridges for Information Interchange*.

NOTE - Document encoded in ODL can be stored such that each SGML ENTITY is recorded in a separate file or in the case of an SDIF encoding, the file can be stored in a single file.

B.2 Font reference

The recommended method for specifying a font reference is to be based on ISO 9541. Such a reference is to be specified by the following ASN.1 encoding.

```

Fonts-Reference ::= SET {
  user-visible-name          (0) IMPLICIT Comment-String OPTIONAL,
  user-readable-comment      (1) IMPLICIT Comment-String OPTIONAL,
  reference-attributes       (2) IMPLICIT SET OF SET {
    precedence-number        (0) IMPLICIT INTEGER OPTIONAL,
    attributes                (1) IMPLICIT Font-Attribute-Set,
    user-readable-comment     (2) IMPLICIT Comment-String OPTIONAL }
}

```

Font sizes from 6 to 72 points (100 to 1200 BMU) are intended to be supported by implementation conforming to this informative recommendation. All other values of font sizes may additionally be supported, but implementations may also support using some form of "fallback".

The minimum font properties and values from ISO 9541 that are to be specified in a Font-Attribute-Set be those specified by the following document application profile notation.

```

Font-Attribute-Set {
  PERM  Fontname          {ANY_VALUE},

```


PERM	Standardversion	{-- To be supplied --},
PERM	Densource	{ANY_VALUE},
PERM	Fontfamily	{ANY_VALUE},
PERM	Posture	{'upright' 'italic-forward'},
PERM	Weight	{'light' 'medium' 'bold'},
PERM	Propwidth	{ANY_VALUE},
PERM	Glyphcomp	{
	PERM #inclglyphcols	{ANY_VALUE},
	PERM #exclglyphcols	{ANY_VALUE},
	PERM #inclglyphs	{ANY_VALUE},
	PERM #exclglyphs	{ANY_VALUE }},
PERM	Dnsize	{ANY_VALUE},
PERM	Minsize	{
	PERM #numerator	{100 .. 1200},
	PERM #denominator	{1} },
PERM	Maxsize	{
	PERM #numerator	{100 .. 1200},
	PERM #denominator	{1} },
	-- BMU Units equivalent to range of 6..72 point sizes --	
PERM	Dsngroup	{
	PERM #group-code	{ANY_VALUE},
	PERM #subgroup-code	{ANY_VALUE},
	PERM #specific-group-code	{ANY_VALUE }},
PERM	Structure	{ANY_VALUE},
PERM	Wrmodes	{
	PERM #wrmodename	{ANY_VALUE},
	PERM #nomescdir	{'0-degrees' '90-degrees' '180-degrees' '270-degrees'},
	PERM #esclass	{ANY_VALUE},
	PERM #avgescx	{ANY_VALUE},
	PERM #avgescy	{ANY_VALUE } }
	}	

B.3 ISO 8632 (CGM) constraints for this DAP

It is recommended that geometric graphics content information contain only those elements listed in this portion of the document, in addition to the constraints imposed by ISO 8613-8. It is believed that this subset of the CGM is sufficiently implemented to enable interworking of geometric graphics for application conforming this document application profile.

Where an element has parameters, recommended constraints on the values are given. The "--" symbol indicates that there is no recommended constraint.

Requirements in ISO 8632 and ISO 8613-8 concerning mandatory elements, parameters must be fulfilled.

B.3.1 Delimeter elements

No-Op	See Note 1
Begin Metafile	See Note 2
End Metafile	
Begin Picture	See Note 2
Begin Picture Body	--
End Picture	

B.3.2 Metafile descriptor elements

Metafile Version	1
Metafile Description	See Notes 2, 3
VDC Type	--
Integer Precision	8, 16
Real Precision	(0,9,23), (1,16,16)
Index Precision	16
Colour Precision	8, 16
Colour Index Precision	8, 16
Maximum Colour Index	--
Colour Value Extent	--
Metafile Element List	--
Font List	--
Character Set List	See Note 5
Character Coding Announcer	0, (basic-7-bit), (basic-8-bit)

B.3.3 Picture descriptor elements

Scaling Mode	See Note 6
Colour Selection Mode	--
Line Width Specification Mode	--
Marker Size Specification Mode	--
Edge Width Specification Mode	--
VDC Extent	--
Background Colour	--

B.3.4 Control elements

VDC Integer Precision	16, 32
VDC Real Precision	(0,9,23), (1,16,16)
Auxiliary Colour	--
Transparency	--
Clip Rectangle	--
Clip Indicator	--

B.3.5 Graphical primitive elements

Polyline	See Note 7
Disjoint Polyline	See Note 7
Polymarker	See Note 7
Text	See Note 2

PART 22 - ODA Image DAP

December 1992 (Stable)

Restricted Text	See Notes 2, 8
Append Text	See Notes 2, 8
Polygon	See Note 7
Polygon Set	See Note 7
Cell Array	See Note 9
Rectangle	--
Circle	--
Circular Arc 3 Point	--
Circular Arc 3 Point Close	--
Circular Arc Centre	--
1Circular Arc Centre Close	--
Ellipse	--
Elliptical Arc	--
Elliptical Arc Close	--

B.3.6 Attribute elements

Line Bundle Index	1-5
Line Type	1-5
Line Width	positive
Line Colour	--
Marker Bundle Index	1-5
Marker Type	1-5
Marker Size	--
Marker Colour	--
Text Bundle Index	1-5
Text Font Index	--
Text Precision	--
Character Expansion Factor	--
Character Spacing	--
Text Colour	--
Character Height	positive
Character Orientation	--
Text Path	--
Text Alignment	--
Character Set Index	--
Alternate Character Set Index	--
Fill Bundle Index	1-5
Interior Style	--
Fill Colour	--
Hatch Index	1-6
Pattern Index	1 .. 8, nx 1-16, ny 1-16
Edge Bundle Index	1-5
Edge Type	1-5
Edge Width	positive
Edge Colour	--
Edge Visibility	--

Fill Reference Point	—
Pattern Table	See Notes 10, 11
Pattern Size	—
Colour Table Specification	See Notes 12, 13
Aspect Source Flags	—

B.3.7 External elements

Message	No action
Application Data	See Note 2

NOTE -

1. An arbitrary sequence of n octets. Where $n=0, 1, \dots, 32767$. The sequence of zero or more octets is for padding purposes.
2. The string occurring in the parametric list of this element shall not contain more than 254 characters, except for data records where the string shall not contain more than 32767 characters.
3. There will be exactly one METAFILE DESCRIPTION element in the metafile. The METAFILE DESCRIPTION string parameter will be used to include the sub-string "ISO FCG13" to label the content information as conforming to this agreement. In addition, the METAFILE DESCRIPTION element should include a sub-string that identifies the generator of this metafile, including company, product, and product version.
4. The only character sets that may be specified are those specified for character content portions. Refer to 7.1, Document Profile Constraints, for further detail on which character sets are supported by this document application profile. The default character set for geometric graphics content is the same as the default character set for character content architecture.
5. The Scale Factor parameter of SCALING MODE element is always a 32-bit floating point value, even when the REAL PRECISION has selected fixed point for other real numbers. It is not apparent in ISO 8632 what the precision of this floating point value is when fixed point has been selected. Its precision shall be (0,9,23).
6. The maximum number of points of this element shall be 1024.
7. The complete restricted text string, including any appended text, shall be included in a metafile conforming to this agreement. The complete restricted text string shall be scaled isotropically such that the specified aspect ratio for the text is not distorted and the string fits into the text extent parallelogram. String of parameters shall not contain any control characters except as allowed by and necessary to implement the character set switching modes which can be selected by basic values of CHAR CODE ANNOUNCER.
8. The maximum number of colour values that can appear in the colour list parameter for the CELL ARRAY element shall be 1048576 (one 1024 x 1024 image).
9. The PATTERN TABLE element shall appear prior to any graphical primitive element to assure that interpreting systems without dynamic pattern update can render the intended effect. Once a given pattern representation is specified and used, it shall not be respecified.
10. Colour Array parameter for the PATTERN TABLE element is 2048. This will support 8 patterns of 16x16. The maximum number of colour values that can appear in a colour array parameter shall be 256 for each PATTERN TABLE element (one 16 x 16 pattern) and 2048 for the complete pattern table itself (eight 16 x 16 patterns).
11. The COLOUR TABLE element shall appear prior to any graphical primitive elements to assure that interpreting systems without dynamic colour update can render the intended effect. Once a given colour representation is

specified and used, it shall not be respecified. For Indexed colour selection, either background colour or all colour indexes in the metafile shall have their representations specified or none shall. Colour indexes shall be specified by the COLOUR TABLE element. Background colour shall be specified either by the BACKGROUND COLOUR element or the the colour Index 0. For direct colour selection, either the background colour or the colour of each displayed primitive shall be explicitly specified, or none shall be specified. In other words, either all colours shall be defaulted or none shall be defaulted.

12. The maximum number of colour values that can appear in the Colour List parameter for the COLOUR TABLE element is 64. This will support a 63 entry colour table.

B.4 Interoperability with SGML applications

The recommended method for the exchange of documents between Standard Generalized Markup Language (ISO 8879, SGML) based systems and systems based on this ODA document application profile is by means of exchanging a document representation conforming to these agreements in an encoded form of the SGML language known as the Office Document Language (ODL). ODL is a standardized SGML application for representing documents conforming to the ODA base standard. Such a representation can be converted into the Office Document Interchange Format (ODIF) supported by this document application profile.

Annex C (informative)

References to other standards and registers

- [1] CCITT Recommendation T.400 : 1988, Introduction to Document Architecture, Transfer and Manipulation;
- [2] CCITT Recommendation T.411 : 1988, Open Document Architecture (ODA) and Interchange Format: Introduction and General Principles;
- [3] CCITT Recommendation T.412 : 1988, Open Document Architecture (ODA) and Interchange Format: Document Structures;
- [4] CCITT Recommendation T.414 : 1988, Open Document Architecture (ODA) and Interchange Format: Document Profile;
- [5] CCITT Recommendation T.415 : 1988, Open Document Architecture (ODA) and Interchange Format: Open Document Interchange Format;
- [6] CCITT Recommendation T.416 : 1988, Open Document Architecture (ODA) and Interchange Format: Character Content Architecture;
- [7] CCITT Recommendation T.417 : 1988, Open Document Architecture (ODA) and Interchange Format: Raster Graphics Content Architecture;
- [8] CCITT Recommendation T.418 : 1988, Open Document Architecture (ODA) and Interchange Format: Geometric Graphics Content Architecture;
- [9] CCITT Recommendation T.502 : 1990, Document Application Profile PM-11 for the Interchange of Character Content Documents in Processable and Formatted Forms;
- [10] CCITT Recommendation T.503 : 1984, Document Application Profile for the Interchange of Group 4 Facsimile Documents;
- [11] CCITT Recommendation T.505 : 1990, Document Application Profile PM-26 for the Interchange of Enhanced Mixed Content Documents in Processable and Formatted Forms;
- [12] ISO 8571 : 1988, Information processing systems - Open Systems Interconnection - File transfer, access and management;
- [13] ISO 9070 : 1990, Information processing - SGML support facilities - Registration procedures for public owner identifiers;
- [14] ISO/TR 9573 : 1988, Information processing - SGML technical report - Techniques for using SGML;
- [15] ISO 10021 : (to be published), Information processing systems - Text communication - Message Oriented Text Interchange System;

PART 22 - ODA Image DAP

December 1992 (Stable)

- [16]** ISP FOD11 : (to be published), Office document format profile for the interchange of basic function character content document in processable and formatted forms;
- [17]** ISP FOD26 : (to be published), Office document format profile for the interchange of enhanced function mixed content documents in processable and formatted forms;
- [18]** ISP FOD36 : (to be published), Office document format profile for the interchange of extended function mixed content documents in processable and formatted forms;
- [19]** MIL-R-28002A : 1990, MILITARY SPECIFICATION, RASTER GRAPHICS REPRESENTATION IN BINARY FORMAT, REQUIREMENTS FOR.

Annex D (informative)**Supplementary information on attributes****Table D.1 Content coding attributes**

Attributes	Basic values	Default values	Non-basic values
Number-of-pels-per-line	any positive integer	None	None
Number-of-lines	any positive integer	None	None
Tiling-offset*	(any non-negative integer < 512, any non-negative integer < 512)	(0,0)	None
Tile-types*	T.6 encoded, bitmap encoded, null background, null foreground, T.6 encoded - MSB	T.6 encoded	None
Type-of-coding	T.6 encoding (untiled), bitmap (untiled), tiled encoded, T.4 1D encoding, T.4 2D encoding, T.6 encoding - MSB (untiled), T.4 1D encoding - MSB, T.4 2D encoding - MSB	T.6 encoding, T.6 encoding - MSB, tiled encoding **	None

Tutorial Note - * Only used if "type of coding" is 'tiled encoded'

Tutorial Note - ** As specified in the document profile

Table D.2 Presentation attributes

Attributes	Basic values	Default values	Non-basic values
Pel-path	0, 90 deg	0 deg	180, 270 deg
Line-progression	270 deg	270 deg	90 deg
Pel-spacing	16, 12, 8, 6, 5, 4, 3, 2, 1 BMU	4 BMU (300)	Any value except 'null'
Clipping	Two Coordinate Pairs (any non-negative integer, any non-negative integer)	(0,0), (N-1, L-1)	None

Table D.3 Document profile attributes

Attribute	Class	Permissible values
Specific-layout-structure	m	present
Presentation-styles	nm	present
Document-characteristics	M	
Document-architecture-class	m	formatted
Document-application-profile	m	{– See clause 8 for a definition of the permitted values for this attribute. –}
Content-architecture-classes	m	{2 8 2 7 2}, {2 8 2 8 0}, {2 8 2 6 0}
Interchange-format-class	m	A
ODA-version	m	ISO 8613, 1991-12-31
Document-architecture-defaults	M	
Content-architecture-class	m	formatted processable raster graphics
Type-of-coding	m	T.6 encoding, tiled encoding, T.6 encoding - MSB
Page-dimensions	nm	See list in table 1, (Default value is NA-A, 9240 x 13200 BMU)
Medium-types	nm	See list in table 1, (Default value is NA-A, 9240 x 13200 BMU)
Page-position	nm	any coordinate pair within page
Raster-gr-content-defaults	NM	
Pel-path	nm	0, 90, 180, 270 degrees (0 is normal default)
Line-progression	nm	90, 270 degrees (270 is normal default)
Pel-spacing	nm	16, 12, 8, 6, 5, 4, 3, 2, or 1 BMU (Normal default is 4 BMU)
Spacing Ratio	nm	Any value
Non-basic-doc-characteristics	NM	
Page-dimensions	nm	See table 1
Medium-types	nm	See table 1

Table D.3 Document profile attributes (concluded)

Attribute	Class	Permissible values
Raster-gr-presentation-features	NM	
Pel-path	nm	180, 270 degrees
Line-progression	nm	90 degrees
Pel-spacing	nm	16, 12, 8, 6, 5, 4, 3, 2, or 1 BMU
Document-management-attributes	M	
Document Reference	m	Any string of characters

The following notation is used in the class column of this table:

- a) m mandatory attribute
- b) nm non-mandatory attribute
- c) d defaultable attribute

Capital letters (M, NM, and D) are used for groups of attributes.

Stable Implementation Agreements for Open Systems Interconnection Protocols: Part 23 - ODA Raster DAP

Output from the December 1992 Open Systems
Environment Implementors' Workshop (OIW)

SIG Chair: **James Wing, IBM**
SIG Editor: **Frank Spielman, NIST**

Foreword

This part of the Stable Implementation Agreements was prepared by the Office Document Architecture (ODA) Special Interest Group (SIG) of the Open Systems Environment Implementors' Workshop (OIW). Development of this document application profile has been done in liaison with several organizations. These include the DoD Computer-aided Acquisition and Logistic Support (CALS) Office, Navy's David Taylor Research Center, and the ad-hoc Tiling Task Group.

This document application profile is intended to be suitable for the interchange of large format raster images.

This part contains four annexes:

- a) annex A (normative): Amendments and corrigenda;
- b) annex B (informative): Recommended practices;
- c) annex C (informative): References to other standards and registers;
- d) annex D (informative): Supplementary Information on attributes.

Future changes and additions to this version of these Implementor Agreements will be published as a new part. Deleted and replaced text will be shown as struckout. New and replacement text will be shown as shaded.

This part uses a convention of double and single quotes that has been established by ISO for use in the ODA base standard and related document application profiles. The convention is to use within the text double quotes to accentuate ODA attribute names and single quotes to accentuate values for those attributes.

Table of Contents

Part 23 - ODA Raster DAP	1
0 Introduction	1
1 Scope and field of applications	1
2 Normative references	2
2.1 ISO	2
2.2 CCITT	3
3 Definitions and terminology	3
3.1 Definitions	3
3.2 Constituent names	3
4 Relationship to other DAPs	4
5 Conformance	4
5.1 Data stream conformance	4
5.2 Implementation conformance	5
6 Characteristics supported by this DAP	5
6.1 Overview	5
6.2 Logical constituents	6
6.3 Layout constituents	6
6.3.1 Overview of the layout characteristics	6
6.3.2 DocumentLayoutRoot	6
6.3.3 Page characteristics	7
6.3.3.1 CompositePage	7
6.3.3.2 Page dimensions	7
6.3.3.3 Nominal page sizes	8
6.3.4 ImageFrame	8
6.3.5 SpecificBlock	8
6.4 Document layout characteristics	10
6.5 Content layout and Imaging control	10
6.5.1 Raster graphics content	11
6.5.1.1 Introduction	11
6.5.1.2 Raster graphics content architecture	11
6.5.1.3 Raster graphics encoding methods	11
6.5.1.4 Raster presentation	12
6.6 Miscellaneous features	13
6.7 Document management features	13
7 Specification of constituent constraints	14
7.1 Document profile constraints	14
7.1.1 Macro definitions	14

7.1.2	Constituent constraints	17
7.1.2.1	DocumentProfile	17
7.2	Logical constituent constraints	19
7.3	Layout constituent constraints	19
7.3.1	Macro definitions	19
7.3.2	Factor constraints	20
7.3.3	Constituent constraints	20
7.3.3.1	DocumentLayoutRoot	20
7.3.3.2	CompositePage	20
7.3.3.3	ImageFrame	20
7.3.3.4	SpecificBlock	21
7.4	Layout style constraints	21
7.5	Presentation style constraints	22
7.5.1	Macro definitions	22
7.5.2	Factor constraints	22
7.5.3	Presentation style constituent constraint	22
7.5.3.1	PStyle	22
7.6	Content portion constraints	22
7.6.1	Macro definitions	22
7.6.2	Factor constraints	23
7.6.3	Constituent constraints	23
7.6.3.1	Raster graphics content portion	23
7.7	Additional usage constraints	24
8	Interchange format	24
8.1	Interchange format ODIF (class A)	24
8.1.1	Interchange format	24
8.1.2	DAP identifier	24
8.1.3	Encoding of application comments	24
8.2	Interchange format SDIF	25
8.2.1	Interchange format	25
8.2.2	DAP identifier	25
8.2.3	Encoding of application comments	25
8.3	Encoding of raster content Information	25
Annex A (normative)		
Amendments and corrigenda		27
A.1	Amendments	27
A.1.1	Amendments to the base standard	27
A.2	Corrigenda	27
A.2.1	Corrigenda to this DAP	27
Annex B (informative)		
Recommended practices		28
B.1	Transfer methods for ODA	28
B.1.1	Conveyance of ODA over CCITT X.400-1984	28
B.1.2	Conveyance of ODA over FTAM	28

PART 23 - ODA Raster DAP	December 1992 (Stable)
B.1.3 Conveyance of ODA over DTAM	29
B.1.4 Conveyance of ODA over flexible disks	29
B.2 Interoperability with SGML applications	29
Annex C (Informative)	
References to other standards and registers	30
Annex D (Informative)	
Supplementary information on attributes	31
Annex E (Informative)	
Register index	34

List of Figures

Figure 1 - Constituents 6

Figure 2 - Document layout structure 7

List of Tables

Table 1 - Dimensions for various page sizes	9
Table 2 - Layout attributes	10
Table D.1 - Content coding attributes	31
Table D.2 - Presentation attributes	31
Table D.3 - Document profile attributes	32
Table E.1 - Object identifiers	34

Pe

0

Thi

Pro

Thi

Thi

the

and

DA

Thi

Im

sta

In

1

Thi

de

ele

co

mi

Thi

for

It i

su

Thi

do

me

Thi

Part 23 - ODA Raster DAP

0 Introduction

This is the definition of a specification for an Open Document Architecture (ODA) Document Application Profile (DAP) named ODA Raster DAP. This DAP is suitable for interchanging documents in formatted form. The documents contain only raster graphics images.

There are two DAP object identifiers supporting this DAP with the only difference being in the encoding of the data stream. One uses the ASN.1 based ODIF encoding. The other uses the SGML/SDIF based ODL encoding. When this document refers to *this profile*, it is referring to this specification regardless of which DAP identifier may be selected to create the data stream.

This DAP has been prepared by the ODA Special Interest Group (SIG) of the Open Systems Environment Implementors' Workshop (OIW). The DAP is defined in accordance with ISO 8613-1 and follows the standardized proforma and notation defined in ISO 8613-1 Annex F. The DAP is based on ODA as defined in ISO 8613 and the Tiled Raster Graphics Addendum to ISO 8613, Part 7.

1 Scope and field of applications

This DAP specifies an interchange format suitable for transfer of structured documents between equipment designed for raster processing. The documents supported by this DAP are based on a paradigm of an electronic engineering drawing or illustration. Such documents contain one or more pages. Each page consists of an image in the form of a bi-tonal raster graphics content. There is no restriction on the minimum size of the image.

This document defines a DAP that allows large format raster documents to be interchanged in a formatted form in accordance with ISO 8613.

It is assumed that, when negotiation is performed by the service using this DAP, all non-basic values are subject to negotiation.

This DAP is independent of the processes carried out in an end system to create, edit, or reproduce raster documents. It is also independent of the means to transfer the document which, for example, may be by means of communication links or exchanged storage media.

The features of a document that can be interchanged using this DAP fall into the following categories:

- a) Page format features - these concern how the layout of each page of a document will appear when reproduced;
- b) Raster graphics layout and imaging features - these concern how the document content will appear within pages of the reproduced document;
- c) Raster graphics coding - these concern the raster graphics representations and control functions that make up the document raster graphics content.

2 Normative references

The following references are required in order to implement this DAP:

2.1 ISO

- [1] ISO 8613-1 : 1989, *Information processing - Text and Office Systems; Open Document Architecture (ODA) and Interchange Format - Part 1: Introduction and General Principles*;
- [2] ISO 8613-2 : 1989, *Information processing - Text and Office Systems; Open Document Architecture (ODA) and Interchange Format - Part 2: Document Structures*;
- [3] ISO 8613-4 : 1989, *Information processing - Text and Office Systems; Open Document Architecture (ODA) and Interchange Format - Part 4: Document Profile*;
- [4] ISO 8613-5 : 1989, *Information processing - Text and Office Systems; Open Document Architecture (ODA) and Interchange Format - Part 5: Open Document Interchange Format*;
- [5] ISO 8613-7 : 1989, *Information processing - Text and Office Systems; Open Document Architecture (ODA) and Interchange Format - Part 7: Raster Graphics Content Architectures*;
- [6] ISO 8613-1 : 1991, *Information processing - Text and Office Systems; Open Document Architecture (ODA) and Interchange Format - Part 1: Annex F - A Document Application Profile Proforma and Notation*;
- [7] ISO 8613-7 : (to be published), *Information processing - Text and Office Systems; Office Document Architecture (ODA) and Interchange Format - Part 7: Amendment - Tiled Raster Graphics Addendum to ISO 8613, Part 7*;
- [8] ISO 8613-7 : (to be published), *Information processing - Text and Office Systems; Office Document Architecture (ODA) and Interchange Format - Part 7: Amendment - Additional Bit Order Mapping Addendum*;
- [9] ISO 8824 : 1987, *Information Processing Systems - Open Systems Interconnection - Specification of Abstract Syntax Notation One (ASN.1)*;
- [10] ISO 8825 : 1987, *Information Processing Systems - Open Systems Interconnection - Specification of Basic Encoding Rules for Abstract Syntax Notation One (ASN.1)*;
- [11] ISO 8879 : 1986, *Information processing - Text and office systems - Standard Generalized Markup Language (SGML)*;
- [12] ISO 8879 : 1986, *Information processing - Text and office systems - Standard Generalized Markup Language (SGML), Amendment 1*;
- [13] ISO 9069 : 1988, *Information processing - SGML support facilities - SGML Document Interchange Format (SDIF)*.

2.2 CCITT

- [14] Recommendation T.4 : 1988, *Standardization of Group 3 Facsimile Apparatus for Document Transmission*.
- [15] Recommendation T.6 : 1988, *Facsimile Coding Schemes and Coding Control Functions for Group 4 Facsimile Apparatus*.

3 Definitions and terminology

3.1 Definitions

The definitions given in ISO 8613-1 are applicable to this document.

3.2 Constituent names

Each constituent that may be included in a document that conforms to this profile has been given a unique name which serves to identify that constituent throughout this profile.

The convention is that full names are used (i.e., no abbreviations are used), two or more words in a name are concatenated and each word begins with a capital. Examples of constituent names used in this profile are CompositePage, DocumentLayoutRoot, and SpecificBlock.

In clause 6, each constituent provided by this profile is underlined once at the point in the text at which the purpose of that constituent is defined. This also serves to identify all the constituents provided by this profile.

The same constituent names are also used in the technical specification in clause 7 so that there is a one-to-one correspondence between the use of these names in clauses 6 and 7.

Although the constituent names relate to the purpose of the constituents, the semantics of constituents must not be implied from the actual names that are used. Also, these names do not appear in an Interchanged document but a mechanism for identifying constituents in an interchange document is provided. Thus in an application using this profile, the constituents may be known to the user by different names.

4 Relationship to other DAPs

Functionally, this DAP is a functional superset of the CCITT Recommendation T.503, A Document Application Profile for the Interchange of Group 4 Facsimile Documents. This DAP is a functional subset of Part 22 - ODA Image DAP.

5 Conformance

In order to conform to this DAP, a data stream representing a document must meet the requirements specified in 5.1.

The requirements for implementations that originate and/or receive data streams conforming to this DAP are specified in 5.2.

5.1 Data stream conformance

The following requirements apply to the encoding of data streams that conform to these agreements:

- a) The data stream shall be encoded in accordance with the ASN.1 encoding rules defined in ISO 8825 or the SGML grammar and syntax of ISO 8879;
- b) The data stream shall be structured in accordance with the interchange format defined in clause 8;
- c) The document shall be structured in accordance with only the formatted document architecture class specified in clause 7. In addition, the document shall contain all mandatory constituents specified for that class and may optionally contain constituents permitted for that class as specified in clause 7;
- d) Each constituent shall contain all those attributes specified as required for that constituent in this profile. Other attributes may be specified provided they are permitted for that constituent;
- e) The attributes shall have values within the range of permissible values specified in this profile;
- f) The encoded document shall be structured in accordance with the abstract document architecture defined in ISO 8613-2;
- g) The encoded document shall be structured in accordance with the characteristics defined in clause 6 and shall contain only those features defined in clause 6.

5.2 Implementation conformance

This clause states the requirements for Implementations claiming conformance to this DAP.

A conforming receiving Implementation must be capable of receiving *either* any data streams conforming to this profile structured in accordance with ODIF or any data streams conforming to this profile structured in accordance with ODL or both of these. Receiving usually, but not always, involves recognizing and further processing the data stream elements.

6 Characteristics supported by this DAP

This clause describes the characteristics of documents that can be represented by data streams conforming to this profile. This clause also describes how these characteristics are represented in terms of divisional components of the data streams.

6.1 Overview

This DAP describes the features of ISO 8613 that are needed to support the interchange of documents containing only raster graphics content. It specifies interchange formats for the transfer of structured documents with simple layout structures.

This DAP describes documents that can be interchanged in the formatted form, which facilitates the reproduction of a document as intended by the originator.

Only one category of content is allowed within the document, that is, a raster graphics content in the formatted processable form. This is intended to facilitate the reproduction of the document content as intended by the originator.

This clause describes the layout features that can be represented in documents conforming to this DAP. The features are described in terms that are typical of the user-perceived capabilities and semantics found in a raster document interchange environment.

For the purpose of interchange, a document is represented as a collection of **constituents**, each of which is represented by a set of attributes. The constituents that make up a formatted document are defined below in this clause and are illustrated in figure 1.

Constituents defined as **required** must occur in any document that conforms to this profile. Constituents listed as **optional** may or may not be present in the document, depending on the requirements of the particular document.

The required constituents include:

- a) a document profile;
- b) layout object descriptions representing a specific layout structure;

c) content portion description.

The only optional constituent is the presentation style.

Document Profile
Presentation Style (Optional)
Specific Layout Structure
Content Portion Description

Figure 1 - Constituents

6.2 Logical constituents

Not applicable.

6.3 Layout constituents

This clause describes the features of the layout objects that can be represented in documents conforming to this DAP.

6.3.1 Overview of the layout characteristics

The document structure allows the document content to be laid out and presented in one or more pages. Each page in a document consists of only a single raster graphics content representing an engineering drawing, illustration, or other raster scanned image.

A specific layout structure of the document conforming to this application profile consists of a four-level hierarchy consisting of a document layout root, composite pages, frames, and blocks. The document can consist of multiple composite pages where each page represents a single image. Each composite page consists of a frame which in turn contains a block containing the content associated with the image.

Figure 2 is an illustration of the features of the document layout structure supported by this DAP.

6.3.2 DocumentLayoutRoot

A DocumentLayoutRoot is the top level in a document layout structure. A DocumentLayoutRoot consists of a sequence of one or more CompositePage constituent constraints.

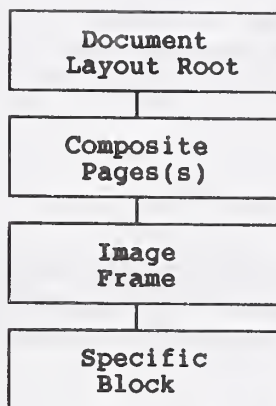


Figure 2 - Document layout structure

6.3.3 Page characteristics

Only one constituent constraint is provided to present pages within a document.

A document consists of a sequence of one or more composite pages. In a document's composite page, a frame is used to position a single raster graphics content representing the image on the page.

A document may consist of multiple pages of different sizes. Each page may be either landscape or portrait orientation. Both orientations are permitted in the document.

6.3.3.1 CompositePage

A CompositePage is a constituent constraint which defines a composite page that corresponds to the page area used for presenting the sequence of an ImageFrame frame.

6.3.3.2 Page dimensions

A wide variety of page dimensions are supported including large format raster documents. The dimensions of the pages may be specified as any value, in BMU measurement units, including the larger sizes produced from foldout-size images and roll paper. These sizes apply to both portrait and landscape orientations. The page sizes include: ISO A0-A5, ANSI A-K, Japanese legal and letter, foldouts 27.94 cm (11 in.) X 35.56 cm (14 in.) and 27.94 cm (11 in.) X 43.18 cm (17 in.), and 27.94 cm (11 in.) roll paper. See table 1.

Dimensions equivalent to or less than the common assured reproduction area (CARA) of ISO A4 and North American Letter (NAL) in portrait or landscape orientation are basic values. Larger page sizes including those produced from roll paper are non-basic and their use must be indicated in the document profile (See table 2).

The default dimensions are the CARA of North American Letter (A). Any default page dimensions may be specified in the document profile subject to the maximum dimensions defined above by using the "page dimensions" attribute. The "page position" attribute may be used to specify the position of the pel array

Image on the page. Although actual page dimensions may be used allowing for the raster content to completely fill a page leaving no borders, it is advised that the assured reproduction area (ARA) listed in table 1 be used wherever feasible. See 7.3 of ISO 8613-2 for general rules for positioning pages on presentation surfaces.

6.3.3.3 Nominal page sizes

The nominal page sizes that may be specified are listed in table 1. In addition, 11 inch roll paper of any length is supported. These may be specified in portrait or landscape orientations. All values of nominal page size are non-basic and hence all values used in a document must be indicated in the document profile using the "medium type" attribute (See table 2).

Any of the nominal page sizes defined in table 1, subject to the restriction specified above, may be specified as the default value in the document profile.

Table 1 also includes the recommended ARA. Information loss may occur when a document is reproduced if the dimensions of the CompositePage exceed the ARA for the specified nominal page size.

6.3.4 ImageFrame

An ImageFrame is a constituent constraint which defines a lowest level frame used for laying out the image of an engineering drawing, illustration, or other raster scanned image. This frame contains a single SpecificBlock containing a raster graphics content portion. Note that there must be exactly one ImageFrame on each page and one block in the frame.

The frame has a fixed position that is equal to the origin of the page. The vertical and horizontal dimensions of this frame are fixed and equal to the maximum size that can be achieved for the position within the area of the page.

6.3.5 SpecificBlock

A SpecificBlock is a constituent constraint which defines a basic layout object used to position and image the content portions associated with an ImageFrame.

The position of the block is fixed and defaults to the origin of the superior frame. The dimensions default to the maximum size that can be achieved for the position within the area of the superior frame.

Table 1 - Dimensions for various page sizes

Page type	Size	Size (BMU)	ARA (BMU)
- Metric			
ISO-A5	148mm x 210mm	7015 x 9920	not defined
ISO-A4	210mm x 297mm	9920 x 14030	9240 x 13200
ISO-A3	297mm x 420mm	14030 x 19840	13200 x 18480
ISO-A2	420mm x 594mm	19840 x 28060	18898 x 27118
ISO-A1	594mm x 841mm	28060 x 39680	26173 x 37843
ISO-A0	841mm x 1189mm	39680 x 56120	37843 x 54283
- ANSI, North American (NA)			
NA-A	8.5in x 11in	10200 x 13200	9240 x 12400
NA-B	11in x 17in	13200 x 20400	12744 x 19656
NA-C	17in x 22in	20400 x 26400	19500 x 25800
NA-D	22in x 34in	26400 x 40800	25800 x 39600
NA-E	34in x 44in	40800 x 52800	39600 x 52200
NA-F	28in x 40in	33600 x 48000	32400 x 47400
NA-G	11in x 90in	13200 x 108000	12400 x 106800
NA-H	28in x 143in	33600 x 171600	31400 x 170400
NA-J	34in x 176in	40800 x 211200	39600 x 210000
NA-K	40in x 143in	48000 x 171600	47400 x 170400
NA-Legal	8.5in x 14in	10200 x 16800	9240 x 15480
- Foldouts			
Small	11in x 14in	13200 x 16800	12744 x 15480
NA-B	11in x 17in	13200 x 20400	12744 x 19656
- Japan			
Legal	257mm x 364mm	12141 x 17196	11200 x 15300
Letter	182mm x 257mm	8598 x 12141	7600 x 10200

Tutorial Note - These page sizes are for the portrait orientation.

Table 2 - Layout attributes

Attributes	Basic values	Default values	Non-basic values
Page dimensions **	CARA NA A, ISO A4	CARA NA-A	ARA NA B-K, ISO A0-A3, Japan legal, 11" Roll Paper
Medium-type ** (Nominal page size)	None	None	NA A-K, ISO A0-A5, Japan letter & legal, 11" Roll Paper

Tutorial Note - See table 1 **

6.4 Document layout characteristics

This DAP provides only for formatted documents. Hence, no provision is made for constraining the document layout process other than as implied in the formatted documents supported by this DAP. In particular, these formatted documents are characterized by the following:

- a) Documents containing only composite pages;
- b) Documents may contain one or more pages;
- c) Pages may vary by orientation within a document;
- d) Each page contains a single raster graphics content portion representing the image;
- e) Content is positioned within fixed position and dimension frames.

6.5 Content layout and imaging control

A document is modelled as an image represented by a raster graphics content portion, as specified in ISO 8613-7.

The only content architecture that may be specified using the attribute "content architecture class" is formatted processable raster graphics. The formatted processable raster graphics content must be specified as the default in the document profile.

6.5.1 Raster graphics content

6.5.1.1 Introduction

This clause defines the features that are applicable to the raster graphics content.

The default values for the following features may be specified in the document profile:

- a) type of coding (required);
- b) compression;
- c) pel path;
- d) line progression;
- e) pel spacing;
- f) spacing ratio.

The specification in a document of a non-basic value by a presentation or coding attribute must be indicated in the document profile.

6.5.1.2 Raster graphics content architecture

The formatted processable raster graphics content is the only content architecture class supported by this DAP and is the only default content architecture class that can be specified in the document profile.

In a composite page, only one content portion can be associated with the image.

6.5.1.3 Raster graphics encoding methods

The content may be encoded in accordance with the encoding schemes defined in CCITT Recommendations T.4 and T.6. In the case of T.4, either the one-dimensional or two-dimensional encoding scheme may be used. Also the bitmap encoding scheme defined in ISO 8613-7 may be used. All these forms of encoding may be used in a single document and all are basic values. 'Uncompressed' mode of encoding may also be used but only as a non-basic value.

In a content portion, it is required that the coding attribute "number of pels per line" be specified. The coding attribute "number of lines" may also be specified. No restriction is placed on the values that may be specified for these coding attributes. This profile places no constraints on the size of the pel arrays that may be used.

The type of coding method used is specified by the attribute "type of coding". The use of this attribute is mandatory in the "document architecture defaults" of the document profile to define the default value of either 'T.6 encoding' (untiled), 'T.6 encoding - MSB' (untiled), or 'tiled encoding'. The use of this attribute

In the description of the content portions is non-mandatory. If this attribute is not specified for a particular content portion, then the default value specified in the "document architecture defaults" of the document profile is used.

If the tiled encoding method is used, the default value of 512 for the "number of pels per tile line" and "number of lines per tile" must be used. No other values are supported, therefore these two attributes do not need to be specified. If the "tile types" attribute is not present, then all tiles will be T.6 encoded. If it is present, then there must be a value specified for each tile in which case only 'null background', 'null foreground', 'T.6 encoded', 'T.6 encoded - MSB', or 'bitmap encoded' values are supported. The T.4 encodings are not supported. There are no restrictions on the use of the "tiling offset" attribute other than that specified in ISO 8613-7 Addendum.

See table D.1, Annex D, for a tabulated list of the attributes and their basic, default, and non-basic values.

6.5.1.4 Raster presentation

Raster presentation is controlled by the presentation attributes specified in ISO 8613-7. This DAP provides for additional constraints on these presentation attributes as specified below.

The basic values for the attribute "pel path" supported by this profile are 0 and 90 degrees. The "pel path" values of 180 and 270 degrees are non-basic.

The basic values for the attribute "line progression" supported by this profile is 270 degrees. The "line progression" value of 90 degrees is non-basic.

Any value may be explicitly specified for pel spacing provided that the spacing between pels is not less than 1 BMU. The pel spacing need not be an integer value. The value of 'null' may not be specified because the scalable layout process is not supported. The specification of the spacings of 16, 12, 8, 6, 5, 4, 3, 2, and 1 BMU between adjacent pels are basic. The specification of any other spacing is non-basic and must be specified in the document profile.

NOTES

1 The basic pel spacing values listed above are equivalent to resolutions of 75, 100, 150, 200, 240, 300, 400, 600, and 1200 pels per 25.4mm respectively when the BMU is interpreted as 1/1200 inch.

2 The attribute "pel spacing" specifies two integers, the ratio of which determines the pel spacing. No restriction is placed on the values of these integers.

There are no restrictions on the use of the "clipping" attribute. The "image dimensions" attribute is not supported.

There are no restrictions placed on the value of the "spacing ratio" attribute providing that the resultant line spacing is not less than 1 BMU. Also, the line spacing need not be an integral number of BMUs. All values are basic.

See table D.2, Annex D, for a tabulated list of the attributes and their basic, default, and non-basic values.

6.6 Miscellaneous features

Specification of the attribute "application comments" is optional. When used in conjunction with the "type of coding" of 'tiled encoding', it contains a sequence of positive integers, one for each tile in the content portion. The sequence of integers is a set of indices representing the octet offsets to the beginning of the respective tiles, starting from the beginning of the "content information". A tile index of zero(0) indicates that the respective tile is null. The integers will be sequenced in the same order as the tiles. The tiles will be sequenced primarily in the pel path and secondarily in the line progression direction as defined by the presentation attributes.

6.7 Document management features

Every document interchanged in accordance with this DAP must include a document profile containing information which relates to the document as a whole.

The features specified by the document profile are listed below. A definition of the information contained in these features is given in the corresponding attribute definitions in ISO 8613-4.

Document constituent information:

- a) specific layout structure;
- b) presentation styles (optional).

Document characteristics:

- a) document application profile;
- b) document application profile defaults;
- c) document architecture class;
- d) content architecture class;
- e) interchange format class;
- f) ODA version date;
- g) raster graphics content defaults.

Non-basic document characteristics:

- a) page dimensions;
- b) medium type;
- c) raster graphics presentation features.

Document management attributes:

- a) document description (only document reference supported).

The attributes applicable to the document profile are defined in table D.3, Annex D.

7 Specification of constituent constraints

7.1 Document profile constraints

7.1.1 Macro definitions

-- General macros --

```
DEFINE(FDA, "{ 'formatted' }")
```

```
DEFINE(DAC, "DocumentProfile (Document-architecture-class)")
```

```
DEFINE(FPR, "ASN.1{2 8 2 7 2}") -- Raster formatted processable --
```

-- Basic page dimensions. --

```
DEFINE(BasicPageDimension,
```

```
REQ #horizontal-dimension {REQ #fixed-dimension { 1..9240 }},
```

```
REQ #vertical-dimension {REQ #fixed-dimension { 1..12400 }},
```

```
| REQ #horizontal-dimension {REQ #fixed-dimension { 1..12400 }},
```

```
REQ #vertical-dimension {REQ #fixed-dimension { 1..9240 }},
```

```
")
```

-- Any size equal to or smaller than CARA (Common Assured Reproduction Area) of ISO A4 and NA A. Both Portrait and Landscape may be specified. --

-- Non-basic page dimensions. --

```
DEFINE(NonBasicPageDimensions,
```

```
{REQ #horizontal-dimension {REQ #fixed-dimension {1..39680}},
```

```
REQ #vertical-dimension {REQ #fixed-dimension {12401..56120}}}
```

```
| {REQ #horizontal-dimension {REQ #fixed-dimension {9241..39680}},
```

```
REQ #vertical-dimension {REQ #fixed-dimension {1..56120}}}
```

```
-- up to ISO A0 portrait --
```

```
| {REQ #horizontal-dimension {REQ #fixed-dimension {1..56120}},
```

```
REQ #vertical-dimension {REQ #fixed-dimension {9241..39680}}}
```

```
| {REQ #horizontal-dimension {REQ #fixed-dimension {12401..56120}},
```

```
REQ #vertical-dimension {REQ #fixed-dimension {1..39680}}}
```

```
-- up to ISO A0 landscape --
```

```
| {REQ #horizontal-dimension {REQ #fixed-dimension {1..48000}},
```

```
REQ #vertical-dimension {REQ #fixed-dimension {12401..211200}}}
```

```
| {REQ #horizontal-dimension {REQ #fixed-dimension {9241..48000}},
```

```
REQ #vertical-dimension {REQ #fixed-dimension {1..211200}}}
```



```

-- up to ANSI J/K portrait --
| {REQ #horizontal-dimension {REQ #fixed-dimension {1..211200}}},
REQ #vertical-dimension {REQ #fixed-dimension {9241..48000}}}
| {REQ #horizontal-dimension {REQ #fixed-dimension {12401..211200}}},
REQ #vertical-dimension {REQ #fixed-dimension {1..48000}}}
-- up to ANSI J/K landscape --
| {REQ #horizontal-dimension {REQ #fixed-dimension {1..12141}}},
REQ #vertical-dimension {REQ #fixed-dimension {12401..17196}}}
| {REQ #horizontal-dimension {REQ #fixed-dimension {9241..12141}}},
REQ #vertical-dimension {REQ #fixed-dimension {1..17196}}}
-- up to Japanese legal portrait --
| {REQ #horizontal-dimension {REQ #fixed-dimension {1..17196}}},
REQ #vertical-dimension {REQ #fixed-dimension {9241..12141}}}
| {REQ #horizontal-dimension {REQ #fixed-dimension {12401..17196}}},
REQ #vertical-dimension {REQ #fixed-dimension {1..12141}}}
-- up to Japanese legal landscape --
| {REQ #horizontal-dimension {REQ #fixed-dimension {13200}}},
REQ #vertical-dimension {REQ #fixed-dimension {> = 16801}}}
-- Any portrait size larger than the typical foldout size (11 in x 14 in) including 11 Inch roll paper. --
| {REQ #horizontal-dimension {REQ #fixed-dimension {> = 16801}}},
REQ #vertical-dimension {REQ #fixed-dimension {13200}}}
-- Any landscape size larger than the typical foldout size (14 in x 11 in) including 11 Inch roll paper --
")

```

DEFINE(PermissiblePageDimensions,"

```

{REQ #horizontal-dimension {REQ #fixed-dimension {1..39680}}},
REQ #vertical-dimension {REQ #fixed-dimension {1..56120}}}
-- up to ISO A0 portrait --
| {REQ #horizontal-dimension {REQ #fixed-dimension {1..56120}}},
REQ #vertical-dimension {REQ #fixed-dimension {1..39680}}}
-- up to ISO A0 landscape --
| {REQ #horizontal-dimension {REQ #fixed-dimension {1..48000}}},
REQ #vertical-dimension {REQ #fixed-dimension {1..211200}}}
-- up to ANSI J/K portrait --
| {REQ #horizontal-dimension {REQ #fixed-dimension {1..211200}}},
REQ #vertical-dimension {REQ #fixed-dimension {1..48000}}}
-- up to ANSI J/K landscape --
| {REQ #horizontal-dimension {REQ #fixed-dimension {1..12141}}},
REQ #vertical-dimension {REQ #fixed-dimension {1..17196}}}
-- up to Japanese legal portrait --
| {REQ #horizontal-dimension {REQ #fixed-dimension {1..17196}}},
REQ #vertical-dimension {REQ #fixed-dimension {1..12141}}}
-- up to Japanese legal landscape --
")

```

DEFINE(NominalPageSizes,"

```

-- ISO Page Sizes --
REQ #horizontal-dimension {7015}, REQ #vertical-dimension {9920}

```

```

-- ISO A5 Portrait --
| REQ #horizontal-dimension {9920}, REQ #vertical-dimension {7015}
-- ISO A5 Landscape --
| REQ #horizontal-dimension {9920}, REQ #vertical-dimension {14030}
-- ISO A4 Portrait --
| REQ #horizontal-dimension {14030}, REQ #vertical-dimension {9920}
-- ISO A4 Landscape --
| REQ #horizontal-dimension {14030}, REQ #vertical-dimension {19840}
-- ISO A3 Portrait --
| REQ #horizontal-dimension {19840}, REQ #vertical-dimension {14030}
-- ISO A3 Landscape --
| REQ #horizontal-dimension {19840}, REQ #vertical-dimension {28060}
-- ISO A2 Portrait --
| REQ #horizontal-dimension {28060}, REQ #vertical-dimension {19840}
-- ISO A2 Landscape --
| REQ #horizontal-dimension {28060}, REQ #vertical-dimension {39680}
-- ISO A1 Portrait --
| REQ #horizontal-dimension {39680}, REQ #vertical-dimension {28060}
-- ISO A1 Landscape --
| REQ #horizontal-dimension {39680}, REQ #vertical-dimension {56120}
-- ISO A0 Portrait --
| REQ #horizontal-dimension {56120}, REQ #vertical-dimension {39680}
-- ISO A0 Landscape --

```

-- ANSI Page Sizes --

```

| REQ #horizontal-dimension {10200}, REQ #vertical-dimension {13200}
-- ANSI A Portrait --
| REQ #horizontal-dimension {13200}, REQ #vertical-dimension {10200}
-- ANSI A Landscape --
| REQ #horizontal-dimension {10200}, REQ #vertical-dimension {16800}
-- ANSI Legal Portrait --
| REQ #horizontal-dimension {16800}, REQ #vertical-dimension {10200}
-- ANSI Legal Landscape --
| REQ #horizontal-dimension {13200}, REQ #vertical-dimension {20400}
-- ANSI B Portrait --
| REQ #horizontal-dimension {20400}, REQ #vertical-dimension {13200}
-- ANSI B Landscape --
| REQ #horizontal-dimension {20400}, REQ #vertical-dimension {26400}
-- ANSI C Portrait --
| REQ #horizontal-dimension {26400}, REQ #vertical-dimension {20400}
-- ANSI C Landscape --
| REQ #horizontal-dimension {26400}, REQ #vertical-dimension {40800}
-- ANSI D Portrait --
| REQ #horizontal-dimension {40800}, REQ #vertical-dimension {26400}
-- ANSI D Landscape --
| REQ #horizontal-dimension {40800}, REQ #vertical-dimension {52800}
-- ANSI E Portrait --
| REQ #horizontal-dimension {52800}, REQ #vertical-dimension {40800}

```

```

-- ANSI E Landscape --
| REQ #horizontal-dimension {33600}, REQ #vertical-dimension {48000}
-- ANSI F Portrait --
| REQ #horizontal-dimension {48000}, REQ #vertical-dimension {33600}
-- ANSI F Landscape --
| REQ #horizontal-dimension {13200}, REQ #vertical-dimension {108000}
-- ANSI G Portrait --
| REQ #horizontal-dimension {108000}, REQ #vertical-dimension {13200}
-- ANSI G Landscape --
| REQ #horizontal-dimension {33600}, REQ #vertical-dimension {171600}
-- ANSI H Portrait --
| REQ #horizontal-dimension {171600}, REQ #vertical-dimension {33600}
-- ANSI H Landscape --
| REQ #horizontal-dimension {40800}, REQ #vertical-dimension {211200}
-- ANSI J Portrait --
| REQ #horizontal-dimension {211200}, REQ #vertical-dimension {40800}
-- ANSI J Landscape --
| REQ #horizontal-dimension {48000}, REQ #vertical-dimension {171600}
-- ANSI K Portrait --
| REQ #horizontal-dimension {171600}, REQ #vertical-dimension {48000}
-- ANSI K Landscape --

-- Foldouts --

| REQ #horizontal-dimension {13200}, REQ #vertical-dimension {16800}
-- Foldout Portrait --
| REQ #horizontal-dimension {16800}, REQ #vertical-dimension {13200}
-- Foldout Landscape --
| REQ #horizontal-dimension {13200}, REQ #vertical-dimension {> = 16801}
-- Any portrait size larger than the typical foldout size (11 in x 14 in) including 11 inch roll paper --
| REQ #horizontal-dimension {> = 16801}, REQ #vertical-dimension {13200}
-- Any landscape size larger than the typical foldout size (14 in x 11 in) including 11 inch roll paper --
")

```

7.1.2 Constituent constraints

7.1.2.1 DocumentProfile

```

{
-- Presence of document constituents --

REQ   Specific-layout-structure    {'present'},
PERM  Presentation-styles         {'present'},

-- Document characteristics --

```


REQ Document-application-profile {-- See clause 8 for a definition of the permitted values for this attribute. --},

REQ Document-application-profile-defaults {

-- Document architecture defaults --

REQ #content-architecture-class {\$FPR},
 PERM #dimensions {\$PermissiblePageDimensions},
 PERM #medium-type {
 PERM #nominal-page-size {\$NominalPageSizes},
 PERM #side-of-sheet {ANY_VALUE}},

-- Any permitted medium type. Both landscape and portrait may be specified. --

REQ #type-of-coding {ASN.1 {2 8 3 7 0} -- T6 encoding --
 | ASN.1 {2 8 3 7 5} -- tiled encoding --
 | ASN.1 {2 8 3 7 6} -- T6 encoding - MSB -- },

PERM #page-position {ANY_VALUE},
 PERM raster-graphics-contents-defaults {
 PERM #pel-path {ANY_VALUE},
 PERM #line-progression {ANY_VALUE},
 PERM #pel-spacing {REQ #length {ANY_VALUE},
 REQ #pel-spaces {ANY_VALUE}},
 PERM #spacing-ratio
 {REQ #line-spacing-value {ANY_VALUE};
 REQ #pel-spacing-value {ANY_VALUE}},
 PERM #compression {ANY_VALUE}},

REQ Document-architecture-class {\$FDA},

REQ Content-architecture-classes {\$FPR},

REQ Interchange-format-class {-- This attribute required only for ODIF interchange. See clause 8 for a definition of the permitted value for this attribute. --},

REQ ODA-version

{REQ #standard-or-recommendation {'ISO 8613'},

REQ #publication-date {'1991-12-31'}},

-- This date represents the date that this DAP was approved. This is the only
 -- approved value, however, the date will be changed if the DAP is significantly
 -- revised. If the date is revised, use of the new date is required only when the
 -- additional functionality is being used. That is, legacy products may continue to
 -- support the earlier DAP.

-- Non-basic document characteristics --

PERM Page-dimensions {MUL {\$NonBasicPageDimensions}},
 PERM Medium-types {MUL{
 PERM #nominal-page-size {\$NominalPageSizes},
 PERM #side-of-sheet {ANY_VALUE}}},


```

-- All values of "medium type" are non-basic --
PERM Coding-attributes {
  REQ #raster-graphics-coding-attributes {
    REQ #compression {'uncompressed'}}},

PERM Presentation-features {
  PERM #Raster-graphics-presentation-features { MUL {
    | PERM #pel-path {'180-degrees' |
    '270-degrees'}
    | PERM #line-progression {'90-degrees'}
    | PERM #pel-spacing {REQ #length {ANY_VALUE}
    EXCEPT {16,12,8,6,5,4,3,2,1},
    REQ #pel-spaces {ANY_VALUE}
    EXCEPT {1}}}
    | PERM #spacing-ratio
    {REQ #line-spacing-value {ANY_VALUE} EXCEPT
    {1},
    REQ #pel-spacing-value {ANY_VALUE} EXCEPT
    {1}}}},

```

```

-- Document management attributes --

```

```

-- Document description --
REQ Document-reference {ANY_VALUE}

```

```

}

```

7.2 Logical constituent constraints

No logical constituents applicable in this clause.

7.3 Layout constituent constraints

7.3.1 Macro definitions

```

DEFINE(RAST," CONTENT_ID_OF(Raster-graphics-content-portion)")

```

7.3.2 Factor constraints

```

FACTOR          ANY-LAYOUT      {

SPECIFIC:
PERM Object-type          {VIRTUAL},
REQ  Object-identifier    {ANY_VALUE},
PERM Subordinates         {VIRTUAL},
PERM User-visible-name    {ANY_VALUE},
PERM User-readable-comments {ANY_VALUE}
}

```

7.3.3 Constituent constraints**7.3.3.1 DocumentLayoutRoot**

```

DocumentLayoutRoot:    ANY-LAYOUT {

SPECIFIC:
REQ  Object-type          { 'document-layout-root' },
REQ  Subordinates         { SUB_ID_OF(CompositePage) + }
}

```

7.3.3.2 CompositePage

```

CompositePage:          ANY-LAYOUT {

SPECIFIC:
REQ  Object-type          { 'page' },
REQ  Subordinates         { SUB_ID_OF(ImageFrame) },
PERM Dimensions           { $PermissiblePageDimensions },
PERM Page-position        { ANY_VALUE },
PERM Medium-type          { PERM #nominal-page-size { $NominalPageSizes },
                           PERM #side-of-sheet { ANY_VALUE } },
PERM Application-comments { ANY_VALUE }
}

```

7.3.3.3 ImageFrame

```

ImageFrame:             ANY-LAYOUT {

SPECIFIC:
REQ  Object-type          { 'frame' },
REQ  Subordinates         { SUB_ID_OF(SpecificBlock) },
PERM Application-comments { ANY_VALUE }
}

```

}

7.3.3.4 SpecificBlock

SpecificBlock

{

SPECIFIC:

REQ Object-type

{ 'block' },

REQ Object-identifier

{ ANY_VALUE },

REQ Content-portions

{ \$RAST },

PERM Position

{ REQ #fixed-position {

REQ #horizontal-position { ANY_VALUE },

REQ #vertical-position { ANY_VALUE } } },

PERM Dimensions

{ REQ #horizontal-dimension

{ REQ #fixed-dimension { ANY_VALUE } },

REQ #vertical-dimension

{ REQ #fixed-dimension { ANY_VALUE } } },

PERM Content-architecture-class

{ \$FPR },

PERM User-readable-comments

{ ANY_STRING },

PERM User-visible-name

{ ANY_STRING },

PERM Application-comments

{ ANY_VALUE },

-- See 8.1.3 and 8.2.3 --

PERM Presentation-style

{ STYLE_ID_OF(PStyle) },

-- PStyle for raster content --

PERM Presentation-attributes

{

PERM #raster-graphics-attributes

{

PERM #pel-path

{ ANY_VALUE },

PERM #line-progression

{ ANY_VALUE },

PERM #pel-spacing

{ REQ #length { ANY_VALUE },

REQ #pel-spaces { ANY_VALUE } },

PERM #spacing-ratio

{ REQ #line-spacing-value { ANY_VALUE },

REQ #pel-spacing-value { ANY_VALUE } },

PERM #clipping

{ ANY_VALUE } }

}

7.4 Layout style constraints

No layout style constraints applicable in this clause.

7.5 Presentation style constraints

7.5.1 Macro definitions

No macro definitions are applicable to this clause.

7.5.2 Factor constraints

```

FACTOR    ANY-PRESENTATION-STYLE  {

REQ  Presentation-style-identifier    {ANY_VALUE},
PERM User-readable-comments           {ANY_STRING},
PERM User-visible-name                {ANY_STRING}
}

```

7.5.3 Presentation style constituent constraint

7.5.3.1 PStyle

```

PStyle:    ANY-PRESENTATION-STYLE  {

    -- This style is used for raster graphics content --

PERM Presentation-attributes        {
    PERM #raster-graphics-attributes {
        PERM #pel-path                {ANY_VALUE},
        PERM #line-progression        {ANY_VALUE},
        PERM #pel-spacing              {REQ #length {ANY_VALUE},
                                        REQ #pel-spaces {ANY_VALUE}},
        PERM #spacing-ratio           {REQ #line-spacing-value {ANY_VALUE},
                                        REQ #pel-spacing-value {ANY_VALUE}},
        PERM #clipping                {ANY_VALUE}}}
    }
}

```

7.6 Content portion constraints

7.6.1 Macro definitions

```

DEFINE(TILED,"    ASN.1{2 8 3 7 5}") -- Tiled raster encoding --

```


7.6.2 Factor constraints

No factor constraints are applicable to this clause.

7.6.3 Constituent constraints**7.6.3.1 Raster graphics content portion**

```

Raster-graphics-content-portion      {
REQ  Content-identifier-layout      {ANY_VALUE},
PERM Type-of-coding                 {
    ASN.1{2 8 3 7 0} -- T.6 encoding --
    | ASN.1{2 8 3 7 1} -- T.4 one dimensional --
    | ASN.1{2 8 3 7 2} -- T.4 two dimensional --
    | ASN.1{2 8 3 7 3} -- bitmap encoding --
    | ASN.1{2 8 3 7 5} -- tiled encoding --
    | ASN.1{2 8 3 7 6} -- T.6 encoding - MSB --
    | ASN.1{2 8 3 7 7} -- T.4 one dimensional - MSB --
    | ASN.1{2 8 3 7 8} -- T.4 two dimensional - MSB -- },

PERM Coding-attributes               {
    REQ  #raster-graphics-coding-attributes {
        PERM #compression                {ANY_VALUE},
        PERM #number-of-lines            {>0},
        REQ  #number-of-pels-per-line    {>0},
        CASE Raster-graphics-content-portion (Type-of-coding) OF {
            {$TILED}: {PERM #number-of-pels-per-tile-line {512},
                        PERM #number-of-lines-per-tile   {512},
                        PERM #tiling-offset              {ANY_VALUE},
                        PERM #tile-types                  {'null background' |
                                                            'null foreground' |
                                                            'T.6 encoded' |
                                                            'bitmap encoded' |
                                                            'T.6 encoded - MSB'}}}},

PERM Alternative-representation      {ANY_STRING},
PERM Content-information             {RASTER}
}

```

7.7 Additional usage constraints

No other usage constraints are currently defined.

8 Interchange format

Two interchange formats are supported by this profile. The interchange format ODIF (class A) can be used by applications requiring a binary encoding based on ASN.1. The Interchange Format SDIF can be used by applications requiring a SGML based clear text encoding. This latter interchange format is an SGML application, called Office Document Language (ODL). For the purposes of interchange, the ODL ENTITIES are placed in an ASN.1 wrapper, as defined by SDIF. Each encoding form has inherent advantages. Conversion of document encoded in one interchange format into the other should not produce the loss of semantic document information.

8.1 Interchange format ODIF (class A)

8.1.1 Interchange format

The value of the document profile attribute "interchange format" for this Interchange format is 'if-a'. This form of ODIF is defined in ISO 8613-5.

The encoding is in accordance with the Basic Encoding Rules for Abstract Syntax Notation One (ASN.1), as defined in ISO 8825.

8.1.2 DAP identifier

The value for the document profile attribute "document application profile" for this interchange format is represented by the following object identifier.

iso (1) identified-organization (3) oiw (14) odasig (11) image-appl (1) raster-dap-odif (1)

8.1.3 Encoding of application comments

ISO 8613-5 define the encoding of the attribute "application comments" as an octet string. For SpecificBlock, this DAP requires that the encoding within that octet string be in accordance with the ASN.1 syntax specified in the following module definition.

```
NIST-DAPSpecification
DEFINITIONS ::= BEGIN
EXPORTS Object-Appl-Comm-Encoding;

Object-Appl-Comm-Encoding ::= SEQUENCE OF INTEGER
END
```

8.2 Interchange format SDIF

8.2.1 Interchange format

The document profile attribute "Interchange format" does not apply for this Interchange format. The SDIF encoding of ODA is defined in Annex E of ISO 8613-5. In addition, ISO 8613-7 contains additional specifications for this encoding of ODA.

8.2.2 DAP identifier

The value for this attribute "document application profile" for this Interchange format is represented by the following object identifier.

iso (1) identified-organization (3) oiw (14) odasig (11) Image-appl (1) raster-dap-sdif (2)

8.2.3 Encoding of application comments

For SpecificBlock, the encoding of the attribute "application comments" is defined in a data stream conforming to this profile with the following DTD definition:

<!-- The following set of declarations may be invoked by using a public entity as follows:

```
<!DOCTYPE odaac Public "-//USA-OIW//DTD SGML ENCODING ODA APPLICATION COMMENTS//EN">
-->
```

<!-- NOTE: To parse the following Document Type Declaration Subset, place the Document Type declaration "<!DOCTYPE odaac [" at the beginning of the file and "]">" at the end of the file. -->

```
<!ELEMENT odaac - - (objappc)+ >
```

```
<!-- Object application comment -->
```

```
<!ELEMENT objappc - O (#PCDATA)>
```

8.3 Encoding of raster content information

The encoding of raster content information in the bitmap encoding scheme is that specified in 9.3 of the raster graphics content architecture part of ISO 8613-7, that is, the first pel in the order of bits is allocated to the most significant bit of an octet. The encoding of the code words in the CCITT Recommendation T.4 and T.6 encoding schemes may be done in either the **up** or **down** bit order. The bit order is specified by the attributes "type of coding" or "tile types". The attribute "tile types" is used only when the value for "type of coding" is 'tiled encoded'. For the **up** order, it is such that the first or only bit of the first code word shall be placed in the least significant bit of the first octet. Subsequent bits of the first and following code words are placed in the direction of more significant bits in the first and following octets. For the **down** order, it is such that the first or only bit of the first code word shall be placed in the most significant bit (MSB) of the

first octet. Subsequent bits of the first and following code words are placed in the direction of least significant bits in the first and following octets.

Annex A (normative)

Amendments and corrigenda

A.1 Amendments

A.1.1 Amendments to the base standard

The amendments applicable to this DAP includes the ISO 8613 - Amendment 1: 1990. This amendment includes text to be included in ISO 8613-1 as the following annexes:

- a) Annex E: Use of ISO/IEC 10021 (MOTIS) to interchange documents conforming to ISO 8613;
- b) Annex F: Document application profile proforma and notation;
- c) Annex G: Conformance testing methodology;
- d) Annex H: Recording of documents conforming to ISO 8613 on flexible disk cartridges conforming to ISO 9293.

In addition, this amendment addresses the Inclusion of the ISO 8613 Technical Corrigenda 1.

This DAP does not include the following features of the amendment:

- a) Addendum on security;
- b) Addendum on styles;
- c) Addendum on alternative representation.

Additionally, this DAP includes features from the Tiled Raster Graphics Addendum to ISO 8613-7, ISO/IEC JTC1/SC18/WG5 901, dated September 1990, and the Additional Bit Order Mapping Addendum to CCITT Rec. T.417|ISO 8613-7, ISO/IEC JTC 1/WG 3, dated July 1991. A new version of ISO 8613-7 which also will incorporate the Colour Addendum is scheduled to be issued in 1993.

A.2 Corrigenda

A.2.1 Corrigenda to this DAP

There are no corrigenda to this DAP.

Annex B (informative)

Recommended practices**B.1 Transfer methods for ODA****B.1.1 Conveyance of ODA over CCITT X.400-1984**

This recommendation describes how ODA body parts are to be encoded for transmission over a CCITT X.400-1984 service.

An ODA body part is encoded as OdaBodyPart in the definition given below:

```
OdaBodyPart ::= SEQUENCE { OdaBodyPartParameters, OdaData }
OdaBodyPartParameters ::= SET {
    document-application-profile
        [0] IMPLICIT OBJECT IDENTIFIER,
    document-architecture-class
        [1] IMPLICIT INTEGER {
            formatted (0),
            processable (1),
            formatted-processable (2) }
OdaData ::= SEQUENCE OF Interchange-Data-Element
```

NOTE - It is recommended to transfer an ODA document as a single body part with tag 12:

Oda [12] IMPLICIT OCTETSTRING

The content of the octet string is encoded as OdaBodyPart, defined above. However, this is out of the scope of this profile.

B.1.2 Conveyance of ODA over FTAM

This recommendation describes the File Transfer, Access, and Management (FTAM) Document Type to be used for minimal storage and transfer capabilities of ODA data streams. It is recognized that enhanced capabilities may at some point be added.

When using FTAM to transfer an ODA file, the FTAM-3, "ISO FTAM Unstructured Binary", document type should be specified. However, since files that do not contain ODA data streams can have the same document type, it is left up to the user of application programs that remotely access files using FTAM to know that a given file contains an ODA data stream.

B.1.3 Conveyance of ODA over DTAM

This recommendation provides for information concerning the interchange of ODA based documents with Document Transfer and Manipulation (DTAM) protocols.

DTAM is defined in the T.430-Series of recommendations and is, like ODA, an integral part of the T.400-Series of CCITT Recommendations named *Open Document Architecture, Transfer and Manipulation*.

The T.520-Series of recommendations contain *Communication Application Profiles (CAP)*. Recommendation T.522 describes the Communication Application Profile BT1 for document bulk transfer. Recommendation T.522 is applicable for the Office Document Format Profile (FOD) published in this ISP.

NOTE - The use of BT1 within the end-to-end oriented Telematic Services Telefax 4 and Teletex is described in 7.1 of Recommendation T.561 and 7.1 of Recommendation T.562.

B.1.4 Conveyance of ODA over flexible disks

The recommended method for interchanging ODA documents between systems by the exchange of magnetically recorded Flexible Disk Cartridges is by the use of an annex to ISO 8613-1 (to be published), *Recoding of Documents Conforming to ISO 8613 on Flexible Cartridges Conforming to ISO 9293*. This annex provides for recording each ODA document as a separate file as defined by ISO 9293, *Volume and File Structure of Flexible Disk Cartridges for Information Interchange*.

NOTE - Document encoded in ODL can be stored such that each SGML ENTITY is recorded in a separate file or in the case of an SDIF encoding, the file can be stored in a single file.

B.2 Interoperability with SGML applications

The recommended method for the exchange of documents between Standard Generalized Markup Language (ISO 8879, SGML) based systems and systems based on this ODA document application profile is by means of exchanging a document representation conforming to these agreements in an encoded form of the SGML language known as the Office Document Language (ODL). ODL is a standardized SGML application for representing documents conforming to the ODA base standard. Such a representation can be converted into the Office Document Interchange Format (ODIF) supported by this document application profile.

Annex C (informative)

References to other standards and registers

CCITT Recommendation T.400 : 1988, Introduction to Document Architecture, Transfer and Manipulation;

CCITT Recommendation T.411 : 1988, Open Document Architecture (ODA) and Interchange Format: Introduction and General Principles;

CCITT Recommendation T.412 : 1988, Open Document Architecture (ODA) and Interchange Format: Document Structures;

CCITT Recommendation T.414 : 1988, Open Document Architecture (ODA) and Interchange Format: Document Profile;

CCITT Recommendation T.415 : 1988, Open Document Architecture (ODA) and Interchange Format: Open Document Interchange Format;

CCITT Recommendation T.417 : 1988, Open Document Architecture (ODA) and Interchange Format: Raster Graphics Content Architecture;

CCITT Recommendation T.503 : 1984, Document Application Profile for the Interchange of Group 4 Facsimile Documents;

ISO 8571 : 1988, Information processing systems - Open Systems Interconnection - File transfer, access and management;

ISO 9070 : 1990, Information processing - SGML support facilities - Registration procedures for public owner identifiers;

ISO/TR 9573 : 1988, Information processing - SGML technical report - Techniques for using SGML;

ISO 10021 : (to be published), Information processing systems - Text communication - Message Oriented Text Interchange System;

ISP FOD26 : (to be published), Office document format profile for the Interchange of enhanced function mixed content documents in processable and formatted forms;

ISP FOD36 : (to be published), Office document format profile for the Interchange of extended function mixed content documents in processable and formatted forms;

MIL-R-28002A : 1990, MILITARY SPECIFICATION, RASTER GRAPHICS REPRESENTATION IN BINARY FORMAT, REQUIREMENTS FOR.

Annex D (informative)**Supplementary information on attributes****Table D.1 - Content coding attributes**

| Attributes | Basic values | Default values | Non-basic values |
|-------------------------|---|---|------------------|
| Number-of-pels-per-line | any positive integer | None | None |
| Number-of-lines | any positive integer | None | None |
| Tiling-offset* | (any non-negative integer < 512, any non-negative integer < 512) | (0,0) | None |
| Tile-types* | T.6 encoded, bitmap encoded, null background, null foreground, T.6 encoded - MSB | T.6 encoded | None |
| Type-of-coding | T.6 encoding (untiled), bitmap (untiled), tiled encoded, T.4 1D encoding, T.4 2D encoding, T.6 encoding - MSB (untiled), T.4 1D encoding - MSB, T.4 2D encoding - MSB | T.6 encoding, T.6 encoding - MSB, tiled encoding ** | None |

Tutorial Note - * Only used if "type of coding" is 'tiled encoded'

Tutorial Note - ** As specified in the document profile

Table D.2 - Presentation attributes

| Attributes | Basic values | Default values | Non-basic values |
|------------------|---|-------------------|-------------------------|
| Pel-path | 0, 90 deg | 0 deg | 180, 270 deg |
| Line-progression | 270 deg | 270 deg | 90 deg |
| Pel-spacing | 16, 12, 8, 6, 5, 4, 3, 2, 1 BMU | 4 BMU (300) | Any value except 'null' |
| Clipping | Two Coordinate Pairs (any non-negative integer, any non-negative integer) | (0,0), (N-1, L-1) | None |

Table D.3 - Document profile attributes

| Attribute | Class | Permissible values |
|--------------------------------|-------|---|
| Specific-layout-structure | m | present |
| Presentation-styles | nm | present |
| Document-characteristics | M | |
| Document-architecture-class | m | formatted |
| Document-application-profile | m | {- See clause 8 for a definition of the permitted values for this attribute. -} |
| Content-architecture-classes | m | {2 8 2 7 2} |
| Interchange-format-class | m | A |
| ODA-version | m | ISO 8613, 1991-12-31 |
| Document-architecture-defaults | M | |
| Content-architecture-class | m | formatted processable raster graphics |
| Type-of-coding | m | T.6 encoding, tiled encoding, T.6 encoding - MSB |
| Page-dimensions | nm | See list in table 1, (Default value is NA-A, 9240 x 13200 BMU) |
| Medium-types | nm | See list in table 1, (Default value is NA-A, 9240 x 13200 BMU) |
| Page-position | nm | any coordinate pair within page |
| Raster-gr-content-defaults | NM | |
| Pel-path | nm | 0, 90, 180, 270 degrees (0 is normal default) |
| Line-progression | nm | 90, 270 degrees (270 is normal default) |
| Pel-spacing | nm | 16, 12, 8, 6 5, 4, 3, 2, 1 BMU, (Normal default is 4 BMU) |
| Spacing Ratio | nm | Any value |
| Non-basic-doc-characteristics | NM | |
| Page-dimensions | nm | See table 1 |
| Medium-types | nm | See table 1 |

Table D.3 - Document profile attributes (concluded)

| Attribute | Class | Permissible values |
|---------------------------------|-------|---|
| Raster-gr-presentation-features | NM | |
| Pel-path | nm | 180, 270 degrees |
| Line-progression | nm | 90 degrees |
| Pel-spacing | nm | Any value except 16, 12, 8, 6, 5, 4, 3, 2, or 1 BMU |
| Document-management-attributes | M | |
| Document Reference | m | Any string of characters |

The following notation is used in the class column of this table:

- a) m mandatory attribute
- b) nm non-mandatory attribute
- c) d defaultable attribute

Capital letters (M, NM, and D) are used for groups of attributes.

Annex E (informative)

Register index

Table E.1 - Object identifiers

| Object identifier | Reference |
|--|-----------|
| iso (1) identified-organization (3) oiw (14) odasig (11)
image-appl (1) raster-dap-odif (1) | 8.1.2 |
| iso (1) identified-organization (3) oiw (14) odasig (11)
image-appl (1) raster-dap-sdif (2) | 8.2.2 |

Stable Implementation Agreements for Open Systems Interconnection Protocols: Part 24 - Conformance Testing

**Output from the December 1992 Open Systems
Environment Implementors' Workshop (OIW)**

SIG Chair: Eva Kuiper, Hewlett Packard
Workshop Editor: Brenda Gray, NIST

Foreword

This part of the Stable Implementation Agreements was prepared by the Conformance Testing Special Interest Group (CTSIG) of the Open Systems Environment implementors' Workshop (OIW).

Text in this part has been approved by the Plenary of the above-mentioned Workshop.

Future changes and additions to this version of these Implementor Agreements will be published as a new part. Deleted and replaced text will be shown as struck. New and replacement text will be shown as shaded.

Table of Contents

| | |
|--|----------|
| Part 24 - Conformance Testing | 1 |
| 0 Introduction | 1 |
| 1 Scope | 1 |
| 2 Normative References | 1 |
| 3 Status | 1 |
| 4 Errata | 1 |
| 5 Guidelines on Interpretation of Disputed Test Cases | 1 |
| 6 Guidelines on the Choice of PICS | 1 |
| 7 CT SIG Resolution for FTAM | 2 |
| 8 Guidelines for PCTR Test Campaign Summary | 2 |

List of Tables

Table 1 - Guidelines for the use of the "Selected" and "Not Run" columns in a PCTR 3

Part 24 - Conformance Testing

0 Introduction

(Refer to Working Implementation Agreements Document)

1 Scope

(Refer to Working Implementation Agreements Document)

2 Normative References

(Refer to Working Implementation Agreements Document)

3 Status

This material is current as of December 18, 1992.

4 Errata

Errata will be reflected in replacement pages of Version 6, Stable Document.

5 Guidelines on Interpretation of Disputed Test Cases

(Refer to Working Implementation Agreements Document)

6 Guidelines on the Choice of PICS

(Refer to Working Implementation Agreements Document)

7 CT SIG Resolution for FTAM

(Refer to Working Implementation Agreements Document)

8 Guidelines for PCTR Test Campaign Summary

The following table provides guidelines for handling the "selected", "not run" and "observations" columns in a Protocol Conformance Test Report. In the following table, the "criterion" column is taken with the "test case status" column to give the expected contents of the three PCTR columns. Note that this table does not contain all possible permutations of PICS support answer, IUT behavior, and test case status. The primary focus of this table is to provide PCTR guidelines for the permutations of "test case status".

Table 1 - Guidelines for the use of the "Selected" and "Not Run" columns in a PCTR

| Criterion | PCTR
selected
column | PCTR
not run
column | PCTR
observation
column |
|---|----------------------------|---------------------------|--|
| I. TEST DESELECTED | | | |
| 1. feature not implemented | | deselect | < empty> (f) |
| 2. feature not applicable | | deselect | < empty> (f) |
| | | | |
| II. TEST SELECTED; TEST PURPOSE NOT ACHIEVED | | | |
| 1. ATS defect; no error in standard | | select | not run (g)
ATS error (a,c) |
| 2. ATS defect; error or ambiguity in standard | | select | not run (g)
ATS error (a,c) |
| 3. ETS defect | | select | not run (g)
ETS error (a,b,c) |
| 4. No defect; Inconclusive verdict | | select | run
manual analysis (c,d) |
| | | | |
| III. TEST SELECTED; TEST PURPOSE ACHIEVED | | | |
| 1. Test is defect-free | | select | run
< empty> (e) |
| 2. ETS/ATS defect; workaround available | | select | run
manual analysis alternate verdict (b,c,d) |

See Notes listed below:

a) This criteria includes any test where the test purpose cannot be achieved by an IUT exhibiting valid behavior.

b) This criteria includes any test where the ETC attempts to accomplish the test purpose in an overly restrictive way, resulting in an Inconclusive test case verdict, even though the IUT exhibits valid behavior. Manual analysis Instructions are provided by the MOT supplier for such tests. These instructions are used by the test lab to determine whether the test purpose can be achieved. If the analysis shows that the test purpose cannot be achieved, the PCTR indicates this test as "Not Run".

If the analysis shows that the test purpose can be achieved, the PCTR indicates this test as "Run".

c) The Observation must include the MOT supplier's Defect Report number or a reference to official MOT documentation (such as the release notes or test specifications) or corrective action. If the MOT supplier does not agree that the test case is defective, then the observation must include a standard and/or profile justification. If the defect is IUT specific, the observation must include a test²-specific justification, or a reference to one. The observation must fully and adequately explain why the IUT's behavior is valid.

d) The Observation must include a test-specific justification of verdict. Analysis of IUT behavior reveals no evidence of non-conformance and no known impact on interworking. Manual analysis also reveals no evidence of ATC/ETC defect.

e) An observation is needed only if the test case verdict is not Pass.

f) It is recommended that the observation column contain a reference to the test case selection expression used to de-select the test case. If the observation column is empty for a particular item, which has been de-selected, the default meaning is that the item was de-selected based on the PICS. If the test is deselected for PIXIT reasons, the reference is mandatory.

g) The term "not run" as used here is not the normal English usage of the term. See ISO/IEC 9646-5 for the meaning of the "run" column of the PCTR. When "not run" appears in the run column of the PCTR it indicates a situation in which there is an ATS or ETS error.

Stable Implementation Agreements for Open Systems Interconnection Protocols: Part 25 - Health Care

Output from the December 1992 Open Systems
Environment Implementors' Workshop (OIW)

SIG Chair:
Workshop Editor:

John J. Harrington, Hewlett Packard
Brenda Gray, NIST

Foreword

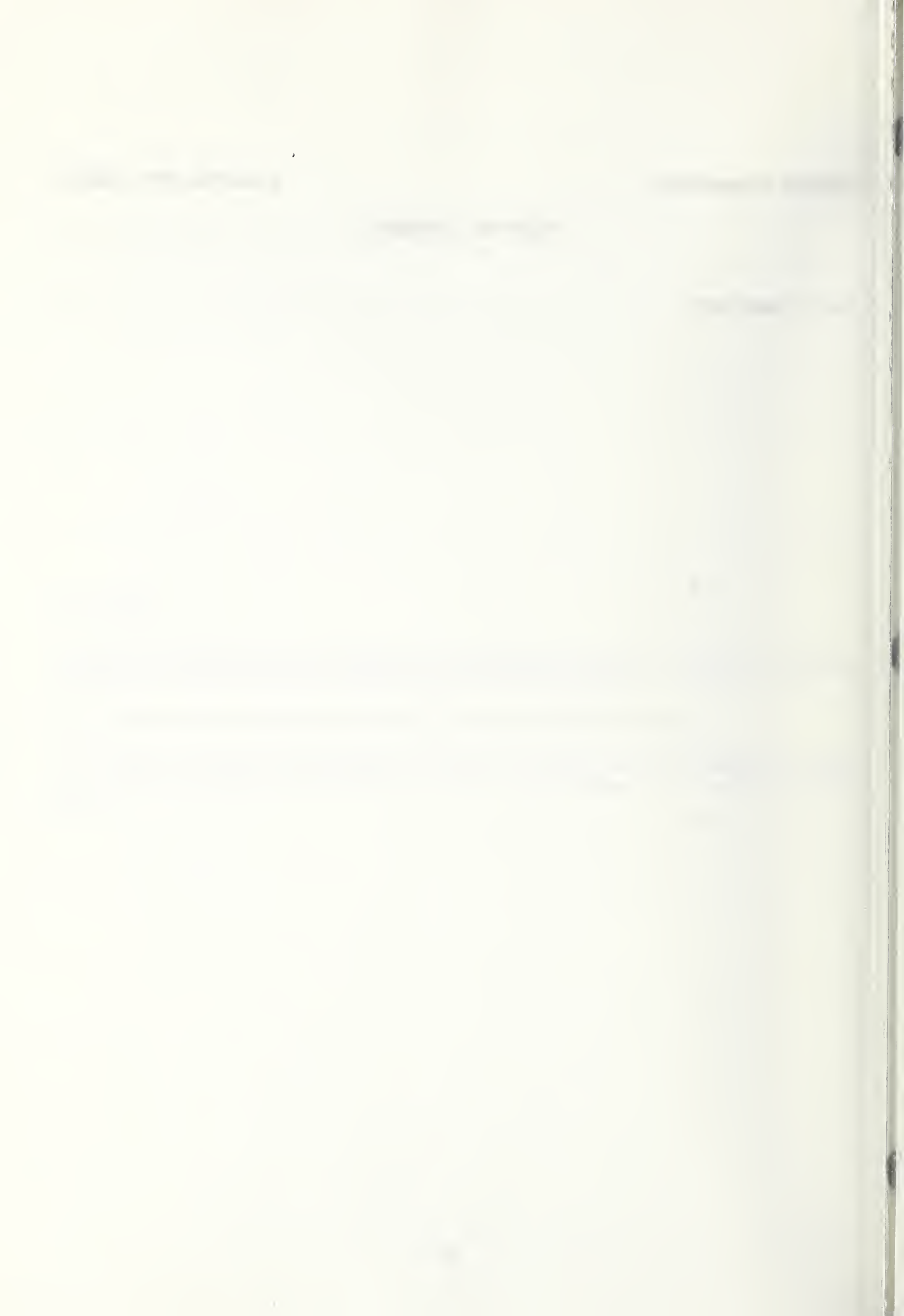
This part of the Stable Implementation Agreements was prepared by the Health Care Special Interest Group (HCSIG) of the Open Systems Environment Implementors' Workshop (OIW).

Text in this part has been approved by the Plenary of the above-mentioned Workshop.

Future changes and additions to this version of these Implementor Agreements will be published as a new part. Deleted and replaced text will be shown as struck. New and replacement text will be shown as shaded.

Table of Contents

Part 25 - Health Care 1



Part 25 - Health Care

Editor's Note - This part is reserved for future stable Health Care agreements. These agreements may be found in the aligned part of the Working Implementation Agreements document. When these agreements become stable, they will be moved into Part 25.

Stable Implementation Agreements for Open Systems Interconnection Protocols: Part 26 - Open Systems Environment

**Output from the December 1992 Open Systems
Environment Implementors' Workshop (OIW)**

SIG Chair: Jerry Johnson, Dept. of Inf. Resources, State of Texas
SIG Editor: Robert Lynch, Digital Equipment Corporation

Foreword

This part of the Stable Implementation Agreements was prepared by the Open Systems Environment Technical Committee (OSE TC) of the Open Systems Environment Implementors' Workshop (OIW).

This text was approved by the Plenary of the Workshop.

Future changes and additions to this version of these implementor Agreements will be published as a new part with redline (shaded) for next text and stikeout (---) for deleted text.

Table of Contents

Part 26 - Open Systems Environment 1



Part 26 - Open Systems Environment

Editor's Note - Future text on this subject will be transferred from the Working Agreements Document.

Stable Implementation Agreements for Open Systems Interconnection Protocols: Part 27 - Office Document Architecture Level 2 to Level 3 Migration DAP

Output from the December 1992 Open Systems
Environment Implementors' Workshop (OIW)

Acting SIG Chair: **Jon Stewart, Digital Equipment Corporation**
SIG Editor: **Patrick Stephens, Digital Equipment Corporation**

Foreword

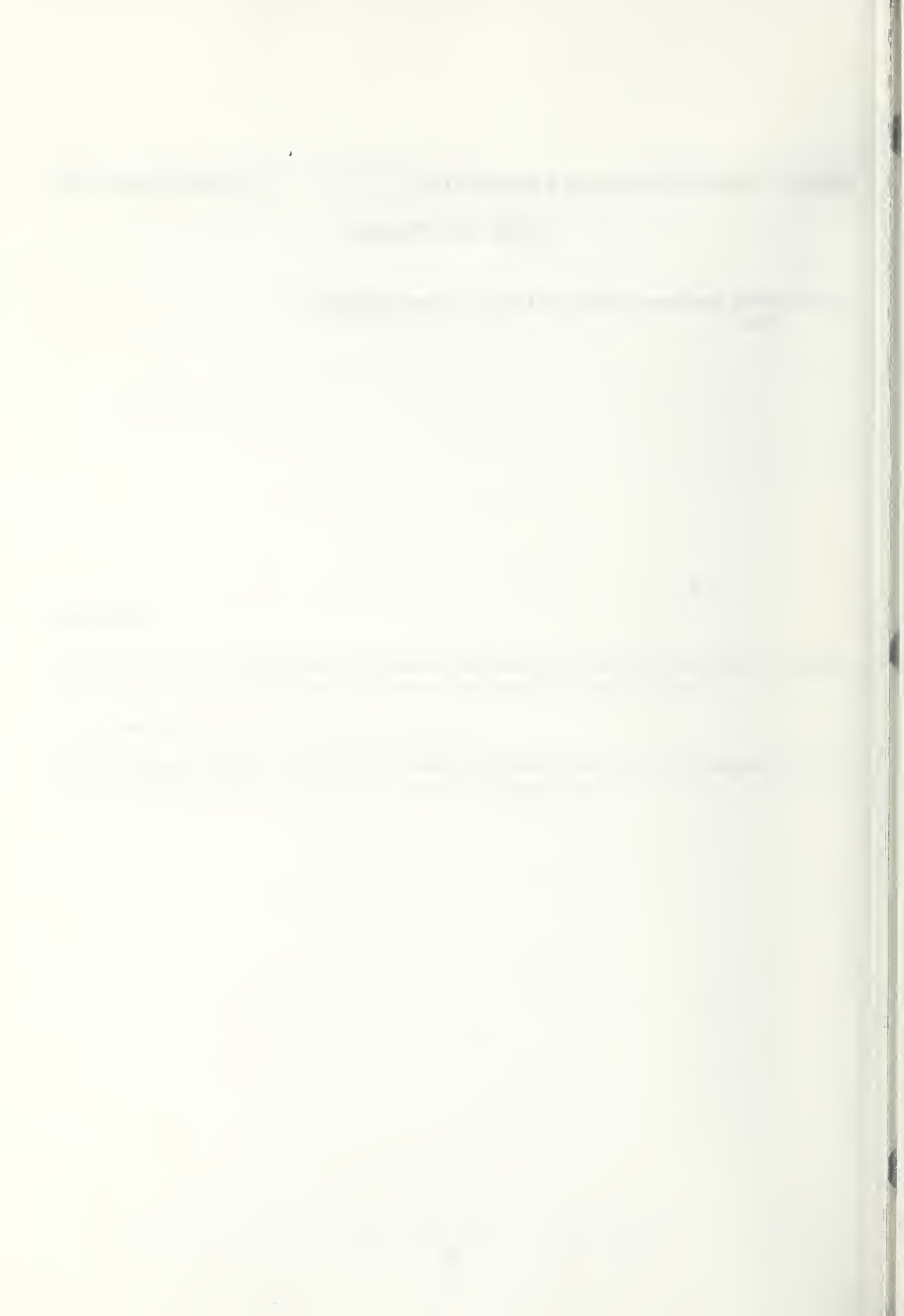
This part of the Stable Implementation Agreements was prepared by the Office Document Architecture Group (ODASIG) of the Open Systems Environment Implementors' Workshop (OIW).

This text was approved by the Plenary of the Workshop.

Future changes and additions to this version of these Implementor Agreements will be published as a new part with redline (shaded) for next text and strikeout (---) for deleted text.

Table of Contents

Part 27 - Office Document Architecture Level 2 to Level 3 Migration
DAP **1**



Part 27 - Office Document Architecture Level 2 to Level 3 Migration DAP

Editor's Note - Text for the Level 2 to Level 3 Migration DAP will be moved here from the Working Agreements whenever it becomes stable.

Stable Implementation Agreements for Open Systems Interconnection Protocols: Part 28 - 1993 Edition Directory Services Protocols

Output from the December 1992 Open Systems
Environment Implementors' Workshop (OIW)

SIG Chair: Kenneth J. Rossen, SHL Systemhouse
SIG Editor: Michael Ransom, NIST

Foreword

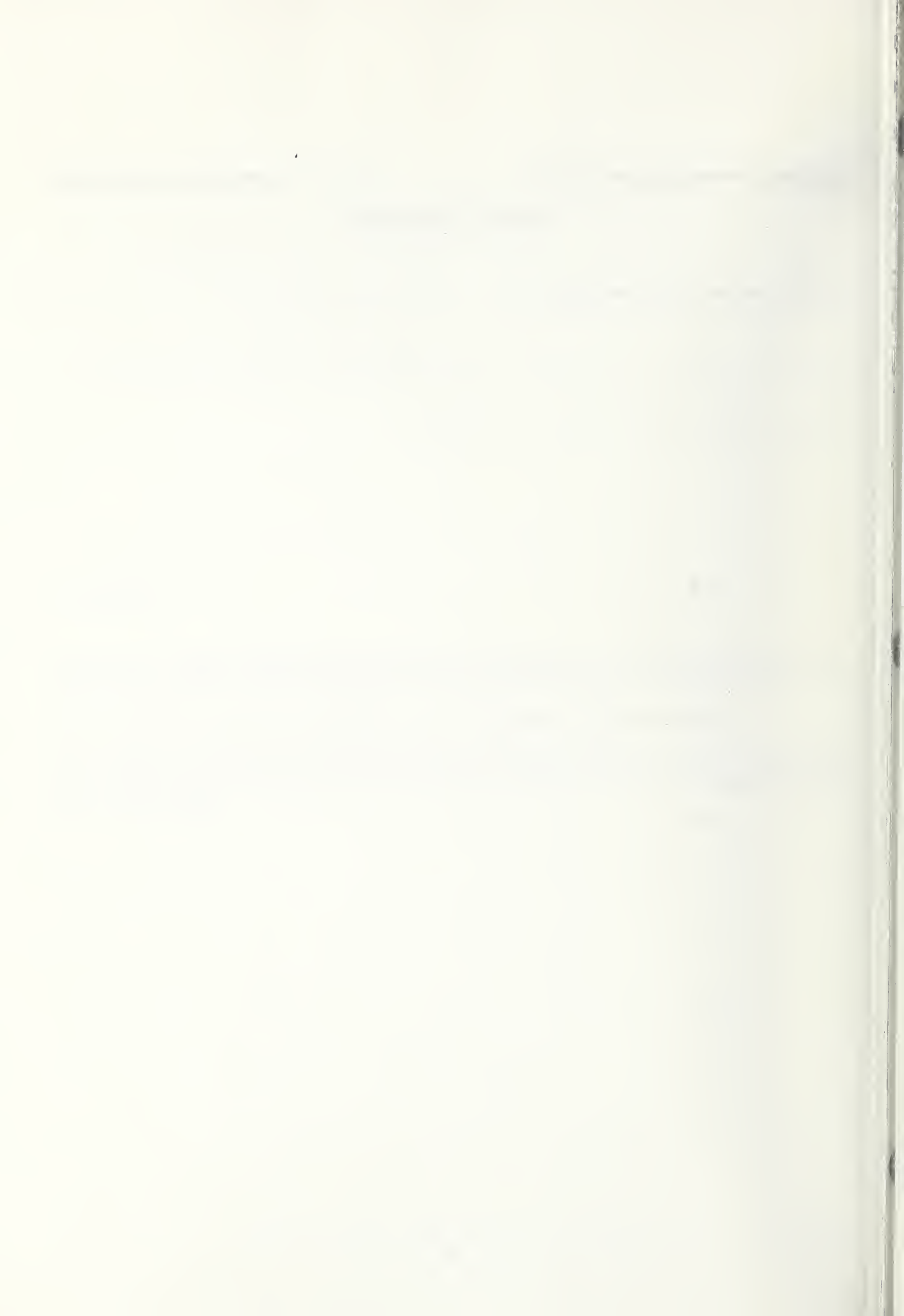
This part of the Stable Implementation Agreements was prepared by the Directory Services Special Interest Group (DSSIG) of the Open Systems Environment Implementors' Workshop (OIW).

Text in this part has been approved by the Plenary of the above mentioned Workshop.

Future changes and additions to this version of these Implementor Agreements will be published as change pages. Deleted and replaced text will be shown as ~~strikeout~~. New and replacement text will be shown as shaded.

Table of Contents

Part 28 - Directory Services Protocols 1



Part 28 - Directory Services Protocols

Editor's Note - This part is reserved for future stable text relating to the 1993 Edition Directory Services Protocols. Refer to the aligned section of the Working Document.

**ANNOUNCEMENT OF NEW PUBLICATIONS ON
COMPUTER SYSTEMS TECHNOLOGY**

**Superintendent of Documents
Government Printing Office
Washington, DC 20402**

Dear Sir:

**Please add my name to the announcement list of new publications to be issued in
the series: National Institute of Standards and Technology Special Publication 500-.**

Name _____

Company _____

Address _____

City _____ **State** _____ **Zip Code** _____

(Notification key N-503)

MEMORANDUM FOR THE RECORD

DATE: 10/10/54

SUBJECT: [Illegible]

1. [Illegible]

2. [Illegible]

3. [Illegible]

4. [Illegible]

5. [Illegible]

6. [Illegible]

7. [Illegible]

8. [Illegible]

9. [Illegible]

10. [Illegible]

NIST Technical Publications

Periodical

Journal of Research of the National Institute of Standards and Technology—Reports NIST research and development in those disciplines of the physical and engineering sciences in which the Institute is active. These include physics, chemistry, engineering, mathematics, and computer sciences. Papers cover a broad range of subjects, with major emphasis on measurement methodology and the basic technology underlying standardization. Also included from time to time are survey articles on topics closely related to the Institute's technical and scientific programs. Issued six times a year.

Nonperiodicals

Monographs—Major contributions to the technical literature on various subjects related to the Institute's scientific and technical activities.

Handbooks—Recommended codes of engineering and industrial practice (including safety codes) developed in cooperation with interested industries, professional organizations, and regulatory bodies.

Special Publications—Include proceedings of conferences sponsored by NIST, NIST annual reports, and other special publications appropriate to this grouping such as wall charts, pocket cards, and bibliographies.

Applied Mathematics Series—Mathematical tables, manuals, and studies of special interest to physicists, engineers, chemists, biologists, mathematicians, computer programmers, and others engaged in scientific and technical work.

National Standard Reference Data Series—Provides quantitative data on the physical and chemical properties of materials, compiled from the world's literature and critically evaluated. Developed under a worldwide program coordinated by NIST under the authority of the National Standard Data Act (Public Law 90-396). NOTE: The Journal of Physical and Chemical Reference Data (JPCRD) is published bimonthly for NIST by the American Chemical Society (ACS) and the American Institute of Physics (AIP). Subscriptions, reprints, and supplements are available from ACS, 1155 Sixteenth St., NW., Washington, DC 20056.

Building Science Series—Disseminates technical information developed at the Institute on building materials, components, systems, and whole structures. The series presents research results, test methods, and performance criteria related to the structural and environmental functions and the durability and safety characteristics of building elements and systems.

Technical Notes—Studies or reports which are complete in themselves but restrictive in their treatment of a subject. Analogous to monographs but not so comprehensive in scope or definitive in treatment of the subject area. Often serve as a vehicle for final reports of work performed at NIST under the sponsorship of other government agencies.

Voluntary Product Standards—Developed under procedures published by the Department of Commerce in Part 10, Title 15, of the Code of Federal Regulations. The standards establish nationally recognized requirements for products, and provide all concerned interests with a basis for common understanding of the characteristics of the products. NIST administers this program in support of the efforts of private-sector standardizing organizations.

Consumer Information Series—Practical information, based on NIST research and experience, covering areas of interest to the consumer. Easily understandable language and illustrations provide useful background knowledge for shopping in today's technological marketplace.

Order the above NIST publications from: Superintendent of Documents, Government Printing Office, Washington, DC 20402.

Order the following NIST publications—FIPS and NISTIRs—from the National Technical Information Service, Springfield, VA 22161.

Federal Information Processing Standards Publications (FIPS PUB)—Publications in this series collectively constitute the Federal Information Processing Standards Register. The Register serves as the official source of information in the Federal Government regarding standards issued by NIST pursuant to the Federal Property and Administrative Services Act of 1949 as amended, Public Law 89-306 (79 Stat. 1127), and as implemented by Executive Order 11717 (38 FR 12315, dated May 11, 1973) and Part 6 of Title 15 CFR (Code of Federal Regulations).

NIST Interagency Reports (NISTIR)—A special series of interim or final reports on work performed by NIST for outside sponsors (both government and non-government). In general, initial distribution is handled by the sponsor; public distribution is by the National Technical Information Service, Springfield, VA 22161, in paper copy or microfiche form.

U.S. Department of Commerce

National Institute of Standards and Technology

Gaithersburg, MD 20899

Official Business

Penalty for Private Use \$300

DATE DUE

[illegible]

Demco, Inc. 38-293