NISTIR 4374

THE NAMING FORUM: PROCEEDINGS OF THE IRDS WORKSHOP ON DATA ENTITY NAMING CONVENTIONS

Judith J. Newton
Editor

U.S. DEPARTMENT OF COMMERCE
National Institute of Standards
and Technology
National Computer Systems Laboratory
Gaithersburg, MD 20899

U.S. DEPARTMENT OF COMMERCE
Robert A. Mosbacher, Secretary
NATIONAL INSTITUTE OF STANDARDS
AND TECHNOLOGY
John W. Lyons, Director

NIST

NISTIR 4374

THE NAMING FORUM: PROCEEDINGS OF THE IRDS WORKSHOP ON DATA ENTITY NAMING CONVENTIONS

Judith J. Newton
Editor

U.S. DEPARTMENT OF COMMERCE
National Institute of Standards
and Technology
National Computer Systems Laboratory
Gaithersburg, MD 20899

July 1990

U.S. DEPARTMENT OF COMMERCE
Robert A. Mosbacher, Secretary
NATIONAL INSTITUTE OF STANDARDS
AND TECHNOLOGY
John W. Lyons, Director

Contents

Foreword

As part of the Federal Information Processing Standard (FIPS) Information Resource Dictionary System (IRDS) project, NIST has sponsored a series of workshops intended to produce guidance to the developers and users of the IRDS standard. Groups invited to the workshops have included Federal practitioners of data management, vendors and others. FIPS 156 for IRDS has been developed in close association with the American National Standards Accredited Standards Committee X3H4, which produced ANSI X3.138-1988, Information Resource Dictionary Systems. FIPS 156 adopts for government use ANSI X3.138-1988.

This publication summarizes the major points discussed during speaker's presentations and general discussions at the IRDS Workshop - Naming Convention Forum held at NIST on 16-17 November, 1989. It was cosponsored by the National Computer Systems Laboratory of NIST, and X3H4.

The purposes of the workshop were to bring together data administrators concerned with naming conventions for a networking and discussion session, and to provide guidance to the X3H4.4 Task Group in the development of requirements for a Naming Convention Verification Module for the X3H4 IRDS standard.

After presentations by NIST X3H4 representatives, nine speakers described their implementations of naming conventions. In addition, a demonstration of the WIS/DIM system's automated assistance for naming was presented. Moderated discussions, on topics of concern to the X3H4.4 Task Group and to the data administration community, followed.

The speakers had been identified by NIST and X3H4.4 as individuals who had implemented, or were implementing, naming conventions in their organizations. Those who participated in the discussions had been invited because of their expertise or interest in the data administration field.

Because the presenters in the workshop drew on their personal experience and knowledge, they may have expressed views which do not necessarily reflect those of NIST or ANSI Committee X3H4. Additionally, they sometimes cited specific vendors and commercial products. The inclusion or omission of a particular company or product does not imply either endorsement or criticism by NIST or X3H4.

List of Participants

Those marked with an asterisk delivered presentations.

Major T. Reed Borman*
The Joint Staff

Ms.Twyla Courtot
MITRE Corporation/X3H4

Mr. David Eddy
Global Software, Inc.

Mr. Bruce Habercamp
Headquarters, U.S. Army

Mr. Andrew King
DLA

Ms. Debbie McEver*
USF&G Co.

Mr. Ron Morehead
SRA Inc.

Mr. Barry Newton
NIST

Mr. Bao Nguyen*
Headquarters, U.S. Air Force

Lt. Col. Philip Olson, Jr.*
Headquarters, U.S. Army

Mr. George Ratte*
Department of the Treasury

Mr. Steve Ritzman
MITRE Corporation

Mr. Bruce Rosen
NIST

Mr. Madhu Singh*
Bellcore/X3H4

Mr. Jerry Stewart
Department of the Treasury

Ms. L. Elaine Stricklett*
USF&G Co.

Mr. Steven Cooper
Data Administration, Inc.

Ms. Ivary Doles
U.S. Postal Service

Dr. Alan Goldfine*
NIST/X3H4

Mr. Marco Johnson
GSA

Mr. Donald J. McCaffery
McCaffery and Assoc's/X3H4

Capt. Alan Mick
Hdqtrs, U.S. Air Force

Mr. John Morgan
Data Administration, Inc.

Ms. Judith Newton*
NIST/X3H4

Ms. Barbara Nichols*
Digital Equipment Corp.

Mr. Frank Pardon
Spectrum Technology

Ms. Pat Reed
Smithsonian Institution

Ms. Toni Robinson
CTA, Inc.

Dr. Karen Ryan
Honeywell/X3H4

Mr. Charles Spasaro*
U.S. Postal Service

Mr. David Stowell
DGI

Dr. Anthony J. Winkler*
CTA, Inc./X3H4

Workshop Discussion Summary

A condensed version of the workshop discussion follows. The speakers were asked to address the problems they had encountered and the tools they had used during data entity naming convention implementation. This provoked a lively verbal exchange on this topic both during and after the speakers' presentations. In addition, three topics of special concern to X3H4.4 were addressed: alternate names, thesaurus functionality, and naming semantics.

Problems and Solutions

The speakers were asked to address the problems they had encountered in implementing naming conventions, and how they had surmounted them. An almost universal problem was user resistance. Among the solutions presented were:

> Proving immediate benefits. For example, user queries against names were reduced when standard abbreviations were used.
>
> Proving operational benefits, as in relating improved efficiency in information retrieval to "warfighting" for the military.
>
> Showing inefficiency of present methods; in one organization, analysis showed that 61% of words in names had four characters or less. This refutes the argument that loss of creativity in naming will lead to less meaningful names.

With modern computing systems, putting a dollar value on storage reduction is no longer relevant. The way to convince users to adopt standardized names is to show managers the need for good data for good business decisions.

A special problem in data standardization in the military was raised: since the process takes so long to be developed and approved, officers are reluctant to invest in a process that won't show benefits "on their watch." There is also a lack of Department of Defense policy at higher levels.

Users also had problems with standardization of name components. (Readers unfamiliar with the terms used in the following discussion may refer to "Naming Convention Verification" beginning on page 16.) One group protested that reducing the number of class words increased the size of the name by creating a need for modifiers to fully identify the entity. This is an unavoidable consequence of using a very small set (twelve to fifteen) of very generalized class words. However, the disadvantage is more than compensated for by the increased modularization of definitions and reuse of name components as generic data entities.

Another speaker mentioned users' reluctance to give up their own name components which were synonyms of the standards. A compromise solution was arranged. In the naming tool used by the data dictionary, the abbreviation used in the access name was related to the synonym allowed in the descriptive name. For instance, if the standard is "customer" but the user wants to use "client," the abbreviation CUSTR will be linked to both words. CUSTR will appear in the access name. Either "customer" or "client" may appear in the descriptive name. (Abbreviations are mandatory in the access name for words over a length limit.)

Tool Usage

Three different commercial tools with data naming assistance functions were used by workshop participants. They were used in several capacities:

> To perform analysis on name components, i.e., number of characters.
>
> To maintain a standard abbreviation list.
>
> To assign abbreviations to words automatically.
>
> To identify synonyms or possible synonyms by KWIC or KWOC lists (which must be checked manually).
>
> To automatically check both long forms and abbreviations of words in names.
>
> To check for near-synonyms of name components (thesaurus function).
>
> To perform validity checking on record and schema generation within an active dictionary.

A concern was expressed about semantic checks on name components. Logical contradictions are possible and need to be identified. For instance, a unit of measure component expressing "length in barrels" should be identified by a naming tool as not permissible.

Alternate Names

Several participants expressed a need for separate sets of naming convention rules for alternate names. These separate rule sets would be used for defining names linked to different physical implementations. Thus, a set of rules could be defined for each discrete physical system on which the logical entities of the dictionary are implemented. Individual names could then be verified by the naming tool in accordance with the appropriate rule set for the physical system in which the named entity is utilized.

This concept could be extended to create alternate names generated by application of the rules to the access name. One commercial tool already does this.

Some opposition to this function was expressed by one of the participants, who stated that this policy would encourage pluralism and synonym proliferation when data administrators are trying to emphasize name sharing and synonym reduction. The consensus, however, was that the real world consists of many systems. A tool to create consistently standardized names within physical systems would be useful. Another participant mentioned that his agency contained eight independent design centers, and he needed a tool to help coordinate the names. Consistent physical names would be a good place to start.

Another topic which provoked discussion was the integration of old systems with new ones. Should this even be attempted, or should old systems be ignored? One concern was whether the data model changes when new systems are designed. Are new systems therefore structurally different from old? Or is the data model the last part of the environment to change?

The problem of tracking alternate names is compounded when commercial data is purchased. One commercial insurance package comes with 10,000 of its own names. How can these be controlled? No one had a good answer for this question.

Finally, one participant suggested that the long-term solution for alternate names lies in a standard for compilers and database management systems - standard field lengths, etc. The consensus was that this was not likely to occur in the foreseeable future.

Thesaurus Functionality

A thesaurus is a tool which allows automated identification of synonyms, near-synonyms, broader and narrower terms, and otherwise related terms of name components. It can be used to identify a preferred term when a name component is entered which would be rejected by a name verification tool because it was not on the accepted list. Non-preferred terms are related to preferred terms in separate lists of components. These lists would be created and maintained by the data administrator. Each set of lists will be unique to each organization.

The thesaurus is an attempt to prevent "entity metamorphosis." Its effect is to restrict the set of terms in each name component to the list of allowable terms by providing an alternative to the user entering an unacceptable term; without this functionality a name verification tool would reject an unacceptable name without suggesting an improvement. The thesaurus provides both assistance and control.

Some of the questions raised about the thesaurus were: Should a naming tool provide a "starter set" of terms? Should a natural language thesaurus come with a naming tool? Should it have an interface to a data administrator-supplied word set?

One objective of data sharing is that after some time interval, creation of new names will stop. From that time on, the thesaurus module would be used for synonym identification and avoidance of duplication - "name contraception."

A thesaurus can also be used to assist in the internationalization of data entity names. Among the synonyms linked to each term could be the term's foreign-language equivalent. This is becoming an important issue with the emergence of international standards, international exchange of data and the growth of multinational corporations.

Semantics

It is relatively easy to check the format and content of names by discretely comparing components to lists of allowed words. When meaning is implicit in the juxtaposition of components, or when a component must be related to an entity outside the universe of the data dictionary (i.e., a prime term must match a data model entity if one of a certain set of class terms is used), semantic analysis of the name components must be performed by the naming tool.

The Naming Convention Verification Module (NCVM) design, as envisioned by the Task Group, now has a requirement for supporting the structuring of names based on components and words arranged in a particular order for various contexts. This is the lowest level of semantic analysis necessary to verify relationships between name components.

The Task Group asked the workshop participants to discuss this aspect of NCVM requirements.

An implementor of a commercial naming product stated that the eventual goal of his tool is to locate the semantics of the name within the corporate data model. "Every data entity has a home in the structure."

One of the users present had developed a combination of a thesaurus, text scanner and data model audit trail. The prime word component traces back to the data model.

A discussion raised several related questions: Should non-data element entries be named differently? Should context be embedded in these names? Should they be related to the data model (i.e., with prime words)? What if the data model changes? One participant stated strongly that the data model is the most static thing in the environment after it has become established.

Summary of Requirements from Workshop

Some issues discussed by workshop participants can be translated into requirements for the NCVM. These include:

A standard abbreviation facility.

A check for synonyms of proposed names.

The thesaurus functions of near-synonym identification.

Alternate name convention rule sets.

Semantic analysis of names to prevent logical contradictions.

Separate naming convention rule sets for different data entity types.

These requirements will be discussed in greater detail in the Technical Report to be issued as a result of the Task Group's study, due to be released later this year.

IRDS Status Report

Dr. Alan Goldfine

National Institute of Standards and Technology

THE INFORMATION RESOURCE DICTIONARY SYSTEM (IRDS)
A STATUS REPORT

Alan Goldfine
National Institute of Standards and Technology

The IRDS is a computer software system that provides facilities for recording, storing, and processing information about an organization's significant data and data processing resources. It is a generalization and standardization of commercially available data dictionary/directory systems, and is defined by a series of standard specifications. The IRDS specification was developed as a joint effort of the National Institute of Standards and Technology and Standards Committee X3H4.

The current IRDS specification is a 764 page document. It defines a Command Language and a screen-oriented, menu-driven Panel Interface. It also defines the underlying data model of the IRDS, a variant of the Entity-Relationship approach. The specification also includes the Basic Functional Schema, a "starter set" of IRDS entity-types, relationship-types, and attribute-types.

The IRDS became a voluntary American National Standard (ANSI X3.138-1988) in October, 1988. In 1989, the IRDS was adopted as a Federal Information Processing Standard (FIPS Publication 156).

X3H4 has always recognized the need for a call interface to the IRDS suitable for use by software external to the IRDS. Four such interfaces are being considered:

1. A proposal from Pansophic Software, dpANS X3.185-198x. This draft standard is undergoing public review (through January 8, 1990), and Federal agency review (ended November 15, 1989).

2. A proposal from IBM.

3. A proposal from DEC, "IRDS Extensions in Support of a CASE Environment for Information Interchange."

4. A draft being considered in a committee of the International Organization for Standardization (ISO).

Several other standards in the IRDS family are being developed or are under active consideration in the ANSI arena:

- o <u>The Export/Import File Format</u>--almost ready for public review. This project will produce a standard format for files used in the controlled transfer of dictionary data from one IRDS to another. The format, when official, will complete the specification of the IRD-IRD Interface in the current IRDS standard.

- IRDS Reference Model--under development. This technical report will describe the logical placement of the IRDS in the information systems environment. It will clarify the role of the IRDS, and illustrate the interfaces to software in this environment.

- Naming Convention Verification--under development. This technical report, which we anticipate will serve as the basis of an IRDS Module, will describe facilities to assist data administrators in: storing standard names and their relationships to other, synonymous names; enforcing the organization's rules for the formation of standard names; and producing name analysis reports on demand.

- Schema Integration--under development. This technical report will outline the steps required in synthesizing an integrated data model or conceptual schema from a set of component user views for ultimate placement in an IRDS. It will specify the minimum functionality required for a tool that provided computer-aided support of the model integration process.

- The IRDS in a Distributed Heterogeneous Environment--under development. This technical report will provide a framework for the logical placement of the IRDS in a data administration environment. This framework would clarify the role of the IRDS in a multi-platform networked environment, and will illustrate the interfaces to CASE software, network software, and intelligent device controllers.

The National Institute of Standards and Technology (NIST) is enhancing its IRDS prototype to include a Panel Interface and IRD-IRD Interface capability. The current source code, which is available for outside use and testing, consists of a C program interface to an SQL database, and implements a subset of the IRDS Command Language.

NIST also plans to develop conformance tests for IRDS software. We invite the cooperation of interested vendors and users in this effort.

Several introductory publications on the IRDS or related subjects are available from NIST:

- A Technical Overview of the Information Resource Dictionary System (Second Edition), NBSIR 88-3700, (Revision of NBSIR 85-3164).

- Guide to Information Resource Dictionary System Applications: General Concepts and Strategic Systems Planning, NBS Special Publication 500-152.

- Guide on Data Entity Naming Conventions, NBS Special Publication 500-149.

- Guide to Data Administration, NIST Special Publication 500-173.

For information on obtaining the above publications or the prototype software, or on the conformance testing project, please call Alan Goldfine at 301/975-3252.

FIPS/X3H4 IRDS WORKSHOP
NAMING CONVENTIONS FORUM
November 16, 1989

THE INFORMATION RESOURCE DICTIONARY SYSTEM
A STATUS REPORT

ALAN GOLDFINE
NATIONAL COMPUTER SYSTEMS LABORATORY
NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY
BUILDING 225, ROOM A266
GAITHERSBURG, MD 20899
(301)975-3252

THE IRDS -- A FAMILY OF STANDARDS

o The IRDS (Current Standard)

o The IRDS Services Interface

o The IRDS Export/Import File Format

o The IRDS Reference Model

o Naming Convention Verification

o Integration of IRDS Schemas

o Requirements for an IRDS in a Distributed Heterogeneous Environment

THE IRDS (Current Standard)

o Defines:
- Underlying E/R Data Model
- Command Language
- Panel Interface
- Basic Functional Schema

o Released as ANSI X3.138-1988 in October 1988

o Adopted as FIPS 156
- FIPS became effective September 25, 1989
- Transition period until March 25, 1991
- Copies ($67.95 each) can be ordered from NTIS: (703)487-4650

THE IRDS SERVICES INTERFACE

o An External Software Interface

o Four proposed versions are being considered:

- Pansophic proposal, dpANS X3.185-198x, is currently undergoing public review (through January 8, 1990) and Federal agency review (ended November 15)

- IBM Proposal

- DEC Proposal

- ISO Draft

THE IRDS EXPORT/IMPORT FILE FORMAT

o Specifies the format of files used in the exchange of data between IRDSs

o Completes the specification of the IRD-IRD Interface in the current standard

o May be voted out of the X3H4 committee in January 1990, with full public review later in 1990

THE IRDS REFERENCE MODEL

o Technical Report

o Framework for the logical placement of the IRDS in the information systems environment
 - Clarifies the role of the IRDS in this environment
 - Illustrates the interfaces to software in this environment

o May be voted out of the X3H4 committee in January 1990, with full public review later in 1990

NAMING CONVENTION VERIFICATION

o Technical Report, which may lead to the development of an IRDS module

o Due to be completed in November 1990

INTEGRATION OF IRDS SCHEMAS

o Technical Report, providing guidance to organizations on techniques and methodologies for effectively integrating different IRD schemas

o Due to be completed in early 1991

REQUIREMENTS FOR AN IRDS IN A DISTRIBUTED HETEROGENEOUS ENVIRONMENT

o Technical Report, examining how industry trends impact the logical placement of the IRDS in a data administration environment encompassing an entire enterprise

o Will clarify the role of the IRDS in a distributed heterogeneous environment

o Will illustrate how such environments as CASE and data management may be supported

o Due to be completed in early 1991

IRDS ACTIVITIES AT NIST

o Publications

o IRDS Prototype
- Current source code (C interface to SQL DBMS, implementing a subset of the Command Language) is available for outside use and testing
- Is being extended to include Panel Interface and Export/Import facility

o Development of conformance tests for IRDS implementations
- NIST invites cooperation

Naming Verification Task Group Overview

Ms. Judith Newton

National Institute of Standards and Technology

NAMING CONVENTION VERIFICATION

Judith Newton

National Computer Systems Laboratory
National Institute of Standards and Technology

Introduction

Naming conventions are guidelines for the format and content of logical data entity names, and are enforced by the organization's data administrator. They help to establish consistency of data throughout the organization. This results in greater efficiency through reduced data handling as the number of discrete data elements is reduced, and a reduction in confusion among both staff and management, as communication is enhanced [NEWT87].

When used in the implementation of CASE tools, naming conventions establish logical coherence of data entities as the data is transferred from one tool function to another.

Data entity names can be organized to provide information about the entity's content to human users, as well as to impose order on the chaotic proliferation of uncontrolled synonyms across and within software systems.

The data administrator must be able to verify that all names conform to the established conventions. An understanding of naming convention development is an important prerequisite to this process.

Name Content and Format

Naming conventions assign differing meanings to <u>components</u> of names. A component is defined as one or more consecutive symbols which may or may not be delimited by a connector symbol [MCCA89]. Combining components produces standardized names with predictable arrangements of content and format. For instance, names can reflect the organization of the data both logically, through <u>prime words</u>, and associatively, through <u>class words</u>. (The convention of **prime:modifier:class [P:M:C]** grammar will be used consistently as an example in this paper. Other conventions, such as IBM's "OF" language, are equally valid.) Prime words represent the logical groupings of data, such as all information which describes the concept **employee**; class words describe the basic nature of a class of data, such as **name**, **code**, or **date**.

Data elements, one type of entity, may need a set of class words to fully describe all elements, while other entities such as file or record may need only one. Modifiers, which establish uniqueness of the data entity name, are the third name component.

The content aspect of naming grammar has been discussed above; the other aspect is format - assigning relative or absolute places to name components. Establishing format rules completes the process by which naming consistency is achieved. For instance, if the prime word is always the first word in the name and the class word last, there is no ambiguity in their identification. Searching by logical group (prime word) or basic nature (class word) is greatly simplified.

Guiding Principles

While there may be many rules to be established for a set of naming conventions, there are a few guiding principles to follow while writing those rules:

Clarity - names are as clear as possible to a casual user.

Brevity within uniqueness - names are short while still maintaining uniqueness within the database.

Conformance to rules of syntax - each name is in the proper format. If there are too many names which cannot be made to fit the naming conventions, the rules may be too rigorous.

Context-freedom - each name is free of the physical context in which the data entity is implemented.

Metanaming Structure

The Information Resource Dictionary System (IRDS) [ANSI88] provides a framework for establishing the structure of the names of each entity and the names' relationships to each other, i.e., the metanaming structure. There are three types of names for each entity: access name, descriptive name, and alternate name.

The access and descriptive names are functionally identical, but by providing two names, the IRDS allows them to share the burdens of the guiding principles of clarity and brevity. The access name may be terse, with abbreviations and acronyms but no connectors allowed (for example, EMP-NAME), while the descriptive name allows for a longer and more discursive style (NAME_OF_EMPLOYEE). A user familiar with the database may want to use the access name for retrievals, while a more casual user would prefer the descriptive name. The alternate name may encompass any number of contingencies, such as physical name(s), report header name, and form input name. The majority of this discussion about names is concerned with access name grammar and usage.

One way to develop descriptive names is to cast access names into natural language grammar and add connectors as needed. It is important to retain the prime and class words. For instance, EMPLOYEE-BIRTH-STATE-NAME becomes NAME_OF_BIRTH_STATE_OF_EMPLOYEE.

Use of Naming Conventions

Application of naming conventions assists the data administrator in the analysis of data by (for instance) identification of coupled data elements and their decomposition into atomic data elements, by restructuring data names in which data is mixed in with metadata, and by facilitating data re-use.

The Concept of Data Reusability

Data sharing among all divisions within an organization is crucial to the concept of information as a corporate resource. The increasing employment of tools such as data dictionaries has facilitated the control and management of data across organizational boundaries, but data administrators are faced with the problem of organizing data so that elements can easily be used by different corporate divisions.

Data elements must be readily reusable among applications within an organization. Employment of a methodology to develop a set of generic data elements, and deriving application data elements from them, will establish a reusable collection of structured data. A "building block" approach to the construction of data elements such as the one described below encourages data sharing.

Development of Taxonomy

The Data Classification and Attribution Task Group (X3L8.6) of X3 Technical Committee X3L8, Representations of Data Elements, is developing a taxonomy of data concept classification. In this paper, it will be discussed in terms of the P:M:C convention; others may be used as well. This taxonomy, when complete, will supply a set of discrete, non-overlapping terms of classification for data elements. These class terms can be used in the formation of generic data elements.

Establishing Generic Data Elements

Each generic data element is formed by assigning an adjective to modify the class term, which restricts the meaning of the element, and assigning a set of attributes which serve to define its characteristics. One important attribute is a set of values, called a domain, which represents valid occurrences of the data element in the database.

Because generic data elements are always application-independent, they are the key to data reusability. Their domains and attributes remain constant, "known factors" which become intrinsic parts of all application data elements derived from them. Once defined, a generic data element is combined with application-specific terms to create data elements which have consistent (but not necessarily identical) attributes and domains.

Deriving Application Elements

A set of application-specific terms is derived from the conceptual data model of an organization. These terms are usually names of objects about which an organization collects data; in P:M:C terms, they are prime words. When combined with the generic data element names, and optionally with additional modifiers, they form application data elements.

All of the attributes defined for a generic data element apply to the derived application data elements. This effect is termed "cascading of attributes." Other attributes may be added to restrict the element's characteristics as long as they do not conflict with the cascaded attributes. The domain of an application data element can remain identical to the generic data element or be restricted to a subset of the generic data element's domain.

Thus, generic data elements are freely distributed among applications without the proliferation of conflicting definitions and domains to which uncontrolled synonym usage is prone. Each application's database administrator may customize the data elements to the needs of the application, within the rules. The objective of freely shared information without total chaos will be met.

An Example of Data Element Derivation

Some examples of class terms are: **name**, **code**, and **amount**. A generic data element is formed from the class term **name** by adding the adjective **state**. **State-name** is assigned a set of attributes, including those for definition, logical length, and domain (fig. 1).[1]

This generic data element is then circulated within the organization. The database administrator with responsibility for the personnel system uses the generic element to form an application data element to track employee's state of birth (EMPLOYEE-BIRTH-STATE-NAME). The training officer creates an application data element which records the state in which a company offering training is located (TRAINER-STATE-NAME). Since a company rule forbids travel for training outside an area which encompasses the five states closest to the company's location, the training officer restricts the domain of TRAINER-STATE-NAME to six states out of the original 50 (fig. 2). Both **employee** and **trainer** are objects derived from the organization's conceptual data model. The new application data elements are then passed back to the data administrator for approval and entered into the corporate data dictionary for organization-wide awareness.

Naming Convention Verification

The data administrator has a complex task in the identification of valid names and reduction of inadvertent synonyms (use of different names for the same data entity). It is, however, amenable to automation.

A task group has recently been formed within X3 Technical Committee X3H4, Information Resource Dictionary System, to investigate the feasibility of an optional IRDS module which will verify that input entity names are in conformance with an organization's naming conventions. The task group (X3H4.4, Naming Convention Verification) will, in 1990, produce a Technical Report describing its research and recommendations [NEWT89].

Consensus has already been reached to allow for the description of any naming convention a data administrator chooses to use; the eventual module design must be flexible enough to accommodate the many variations of naming conventions already in use. There must be a process which will describe the conventions of choice to the module, and functions which translate the description into rules for name validation. This version of the module will operate only against the names of entities contained in the IRDS. Future versions may cooperate with the Services Interface to validate names in external software.

[1] The format of these entries is that generated by the "OUTPUT IRD" command of the IRDS Command Language Prototype, developed by the National Computer Systems Laboratory of the National Institute of Standards and Technology [GOLD88/1, GOLD88/2].

These are the requirements the task group has identified.

- The specific name verification rules will be external to the module.
- The module will identify synonyms of name components.
- The module will identify nonstandard names.
- The module will assist in generation of allowable standard names from a given name or given definition based on a set of rules.
- Rule maintenance will be supported.
- Rules may vary for different data object types (for instance, there may be differing name formats among entity-types).
- The module will associate word types within a given context (at least; more semantic functionality may be added).

The scenario for use of the module is as follows:

- The data administrator develops naming conventions for the organization based on user needs and the principles of data modelling.
- The rules governing the formation of names based on these conventions are described to the Naming Convention Verification Module.
- Data entities are entered into the IRDS, invoking the verification module.
- Entity names which are not formed according to the rules are rejected. Alternatives are suggested by the module for the DA's approval.
- Entity names which resemble other names, and could be synonyms, are identified.
- The data administrator modifies the rules as needed.

Full syntax (format) checking is supported. The appropriate amount of semantic (content) interpretation is still under investigation. Word matching, which is sufficient to identify generic data elements, is the minimum level of semantic support.

The technical report to be issued by the task group will include a discussion of the group's research, the functionality identified for the module, and a sample model specification.

Like most design activities, the effort expended in advance of the application of data entity naming conventions will pay off over the life of the enterprise. A tool which will allow all users of the IRDS to verify standard names in a standard way will help immeasurably to accomplish this desirable effect.

```
ENTITY = STATE-NAME
DESCRIPTIVE_NAME = STATE_NAME
ENTITY_TYPE = ELEMENT

                              ATTRIBUTES

DESCRIPTION = The name of a state of the United States of America.

DATA_TYPE = Character-string

LENGTH = 20

DATA_CLASS = Generic data element

ALLOWABLE_VALUE = Alabama
ALLOWABLE_VALUE = Alaska
ALLOWABLE_VALUE = Arizona
        .         .
        .         .
        .         .
ALLOWABLE_VALUE = Wyoming

                             RELATIONSHIPS

employee-birth-state-name ELEMENT_DERIVED_FROM_ELEMENT state-name
trainer-state-name ELEMENT_DERIVED_FROM_ELEMENT state-name
```

Figure 1. Generic Data Entity Description

```
ENTITY = TRAINER-STATE-NAME
DESCRIPTIVE_NAME = STATE_NAME_OF_TRAINER
ENTITY_TYPE = ELEMENT

                              ATTRIBUTES

DESCRIPTION = The name of a state of the United States of America.

DATA_TYPE = Character-string

LENGTH = 20

DATA_CLASS = Application data element

COMMENTS = The state in which a trainer offers training. Restricted
to the six states shown.

ALLOWABLE_VALUE = Delaware
ALLOWABLE_VALUE = District of Columbia
ALLOWABLE_VALUE = Maryland
ALLOWABLE_VALUE = New Jersey
ALLOWABLE_VALUE = New York
ALLOWABLE_VALUE = Virginia


                           GROUP ATTRIBUTES

     IDENTIFICATION_NAMES

     ALTERNATE_NAME = TR-ST-NAME
     ALTERNATE_NAME_CONTEXT = COBOL WORKING FILE


                             RELATIONSHIPS

trainer-state-name ELEMENT_DERIVED_FROM_ELEMENT state-name
```

Figure 2. Application Data Entity Description

REFERENCES

[ANSI88] ANSI X3H4, American National Standard Information Resource Dictionary System, X3.138-1988, American National Standards Institute, New York, 1988.

[GOLD88/1] Goldfine, Alan, and Konig, Patricia, A Technical Overview of the Information Resource Dictionary System, NBSIR 88-3700, National Bureau of Standards, Gaithersburg, MD, January 1988.

[GOLD88/2] Goldfine, Alan, and Kirkendall, Thomasin, The ICST-NBS Information Resource Dictionary System Command Language Prototype, NBSIR 88-3830, National Bureau of Standards, Gaithersburg, MD, August 1988.

[MCCA89] McCaffery, Don, "Metadata Naming Rules and Conventions," X3H4.4 Working Paper 89-24R, June, 1989.

[NEWT87] Newton, Judith, Guide on Data Entity Naming Conventions, NBS Special Publication 500-149, National Bureau of Standards, Gaithersburg, MD, October 1987.

[NEWT89] Newton, Judith, "Proposal to Develop a New X3 Technical Report," X3H4 Working Paper 89-01R, January, 1989.

NAMING CONVENTION VERIFICATION

JUDITH NEWTON

NATIONAL COMPUTER SYSTEMS LABORATORY

NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY

WHAT ARE NAMING CONVENTIONS ?

GUIDELINES FOR FORMAT AND CONTENT OF DATA ENTITY NAMES

ENFORCED BY DATA ADMINISTRATOR

WHAT ARE THEY GOOD FOR ?

CONSISTENCY OF DATA THROUGHOUT ORGANIZATION

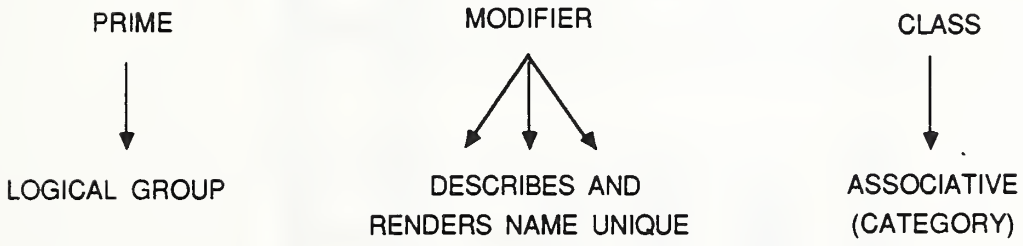
MEANS:

- o GREATER EFFICIENCY - REDUCED DATA HANDLING
  SYNONYM RESOLUTION
- o COST SAVINGS - LESS COMPUTING TIME
- o CONFUSION REDUCTION AMONG STAFF AND MANAGEMENT

GUIDING PRINCIPLES FOR RULE DERIVATION

- CLARITY
- BREVITY WITHIN UNIQUENESS
- CONFORMANCE TO RULES OF SYNTAX
- CONTEXT-FREEDOM

NAME CONTENT AND FORMAT



PRIME → LOGICAL GROUP

MODIFIER → DESCRIBES AND RENDERS NAME UNIQUE

CLASS → ASSOCIATIVE (CATEGORY)

CONTENT - MEANINGS OF NAME COMPONENTS

FORMAT - ARRANGEMENT OF COMPONENTS WITHIN NAME
MAY BE RELATIVE OR ABSOLUTE

I R D S

INFORMATION

RESOURCE

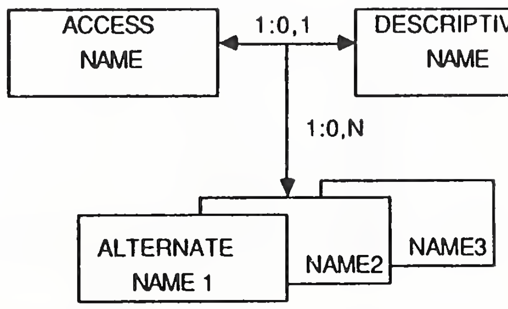
DICTIONARY

SYSTEM

FEDERAL INFORMATION PROCESSING STANDARD

AMERICAN NATIONAL STANDARDS INST. STANDARD

IRDS METANAMING STRUCTURE



METANAME EXAMPLES

ACCESS NAME	DESCRIPTIVE NAME
EMPLOYEE-NAME	NAME_OF_EMPLOYEE

ALTERNATE NAMES

E-NAME
EMP-ID } PHYSICAL FILE NAMES

"NAME: LAST, FIRST, MIDDLE" FORM ENTRY DOCUMENTATION ONLY

WHY DO WE NEED A NAMING CONVENTION VERIFICATION STANDARD?

- O TO ESTABLISH A STANDARD WAY OF DEALING WITH NAMING CONVENTIONS

- O TO PROVIDE CONSISTENCY FOR ALL IRDS USERS

AUTOMATED VERIFICATION

ANSI X3H4.4 TASK GROUP

NAMING CONVENTION VERIFICATION

FLEXIBILITY

A module must allow the data administrator to describe any naming convention variation needed by the organization - the description will be translated into rules for name validation.

TERMS DEFINED

o SYMBOL - ANY ELEMENT OR MEMBER OF A CHARACTER SET

o WORD - ANY RECOGNIZED WORD OF A LANGUAGE, INCLUDING SHORTENED FORMS, USED AS A COMPONENT OF A NAME.

o TERM -ONE OR MORE WORDS OR SYMBOL SETS REFERENCED AS A UNIT, OFTEN ACRONYMS.

o COMPONENT -A WORD OR TERM USED TO COMPARE WITH A SET.

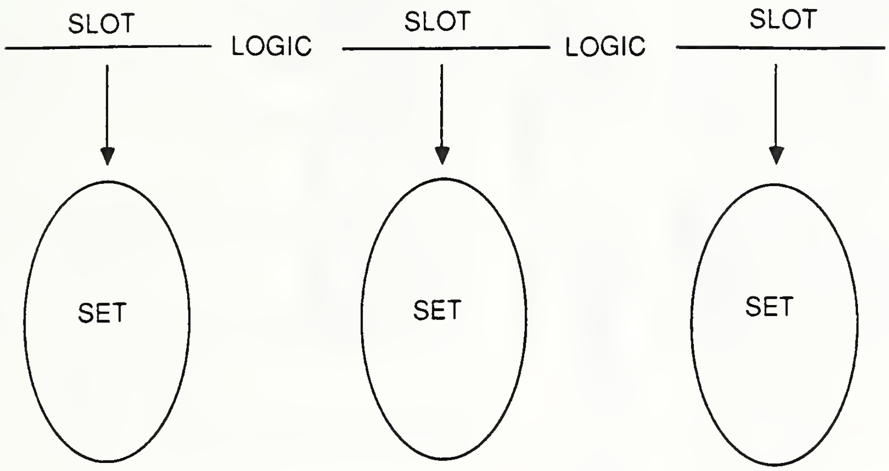
o SET - AN UNORDERED COLLECTION OF WORDS, SYMBOLS, OR TERMS.

o NAME -AN ORDERED SET OF SYMBOLS USED TO IDENTIFY A METADATA ENTITY.

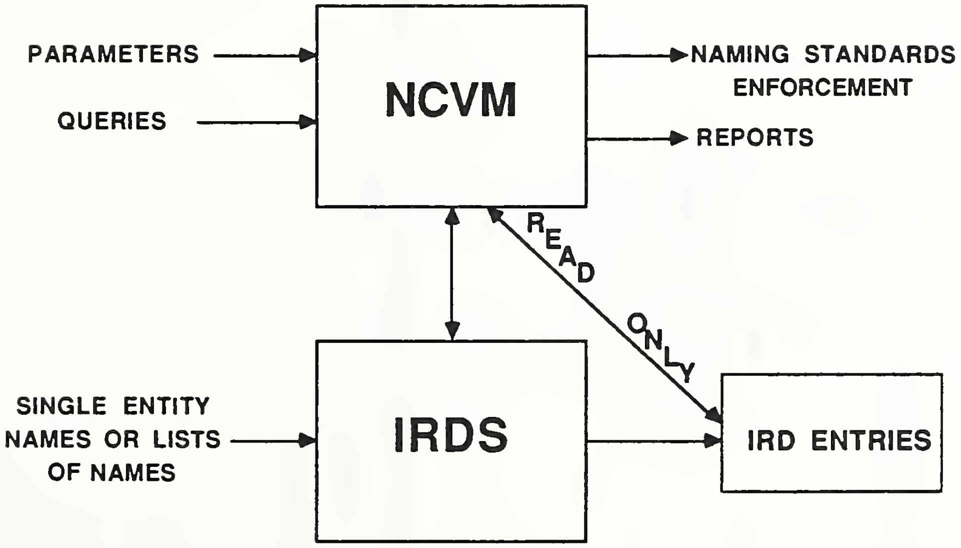
o RULE - (1) IN NAMING, A PROCESS OR PROCEDURE TO FOLLOW WHEN COMPOSING A NAME.

- (2) IN NAME VERIFICATION LOGIC, AN ALGORITHM OR PROCESS APPLIED TO A NAME TO DETERMINE ITS COMPLIANCE WITH A PARTICULAR NAMING CONVENTION.

HOW COMPONENTS ARE RELATED



HOW IT COULD WORK


PARAMETERS
QUERIES
NCVM
NAMING STANDARDS ENFORCEMENT
REPORTS
READ ONLY
SINGLE ENTITY NAMES OR LISTS OF NAMES
IRDS
IRD ENTRIES

NAMING RULE ***PARAMETERS***

WOULD BE A REPRESENTATION OF THE
FORMAT AND CONTENT OF DATA ENTITY NAMES

MAY BE IN THE FORM OF:
- o LIST OF RULES
- o ANSWERS TO QUESTIONS (EXPERT SYSTEM?)
- o FILL-IN-THE-BLANK

QUERIES CAN BE USED TO GENERATE REPORTS FOR NAME OR DATA STRUCTURE ANALYSIS.

NAMING STANDARDS ***ENFORCEMENT:***

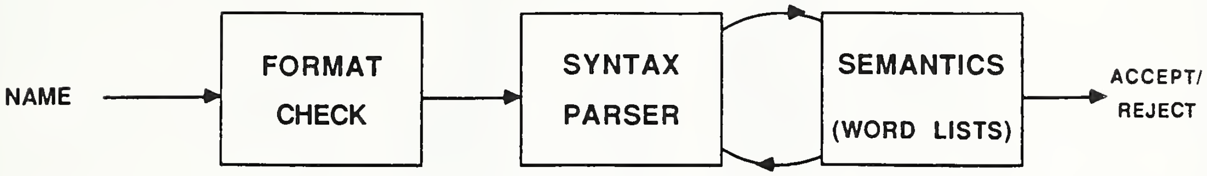
- o CHECKING INPUT NAMES FOR STANDARD GRAMMAR
- o REJECTING NONSTANDARD NAMES, WITH REASONS
- o GENERATION OF STANDARD NAMES WITH DA VERIFICATION

IRDS ***INTERFACE*** :

NAME RETRIEVAL FOR REPORTS

VERIFICATION OF STANDARD FORMAT FOR NEW OR MODIFIED NAMES, AND RETURN.

PROCESSING FLOW OF NCVM



ALLOWABLE CHARACTERS
LENGTH

CONNECTORS/MODIFIERS

WORD FORM

WORD SEPARATORS (PRESENCE)

CAPITALIZATION

WORD LISTS

WORD-BY-WORD IN NAME
MATCH TO WORD LIST

USES TEMPLATE

ABBREVIATIONS/ACRONYMS

WORD ORDER

WORD SEPARATORS-MEANING

COMPONENTS OF THE NCVM COULD INCLUDE:

- FORMAT CHECKER (SUPERFICIAL)
- SYNTAX PARSER
- LIST OF "RESERVED WORDS" FOR VARIOUS SEMANTIC COMPONENTS
- NAME TEMPLATE - DESCRIBES FORMAT

AUTOMATED VERIFICATION

REQUIREMENTS

- o Specific verification rules external to module
- o Identify synonyms of name components
- o Identify nonstandard names
- o Assist in generating allowable standard names
- o Support rule maintenance
- o Support variance of rules by object type
- o Associate word types within a given context
- o Interface with other modules

AUTOMATED VERIFICATION

SCENARIO

- o The data administrator develops naming conventions for the organization.
- o The rules for name formation based on these conventions are described to the Naming Verification Module.
- o Data entities are entered into the IRDS, invoking the verification module.
- o Non-conformant entity names are rejected. Alternative names are suggested for the data administrator's approval.
- o Possible synonyms are identified.
- o The data administrator modifies the rules as needed.

EXPECTATIONS

NOVEMBER 1990 - TECHNICAL REPORT

RECOMMEND MODULE DEVELOPMENT

1990 - 1992? - MODULE DEVELOPMENT

Naming Conventions Presentation

Ms. L. Elaine Stricklett
Ms. Debbie McEver

United States Fidelity and Guaranty Co.

USF&G'S DATA NAMING CONVENTIONS

At present, USF&G's data naming conventions apply only to new systems developed under the System Development Life Cycle. These standards have been developed for items and groups, including modules, records, programs, and called/calling modules.

These naming conventions state that the Corporate Name shall be developed from the definition of the term (use of the "of" language is recommended to aid in establishing the Corporate Name). The Corporate Name (the fully spelled out English name or business term) should contain a class word, prime word, and other modifiers as needed and is to be used in the "Corporate Alias" keyword of DATAMANAGER. A list of USF&G Standard Class-Words is maintained on TSO for easy reference. The member-name for the Corporate Data Dictionary on DATAMANAGER will be derived from the Corporate Name by abbreviating each syllable in the Corporate name. The Standard Abbreviation List will be used to determine the appropriate abbreviations for each syllable. A utility has also been developed using PM/SS to abbreviate automatically each syllable in the Corporate Name.

For ease in establishing data naming conventions, we developed definitions, as well as standards and guidelines, for several terms. Aliases, homonyms, and synonyms are not allowed in new development. Versions are discouraged, but may be necessary.

Aliases are data elements with different data names that reside in the same location in a file. Redefines are a type of alias.

Homonyms are data elements that have the same name, but have different definitions and are used for different purposes in different systems. They may or may not have different formats. For example, ACCOUNT-NUMBER may have very different meanings when used in Accounts Payable or in Accounts Receivable systems.

Synonyms are data elements that have the same name, same logical length, but have different formats (e.g., character or numeric).

Versions are data elements with the same names and definitions, same format (i.e., character or numeric), but with different logical lengths. For example, CUSTOMER-NAME may have a length of 35 in one application and a length of 28 in another.

In addition, we required certain DATAMANAGER keywords to be used to give additional information. Some of these were delivered with the product, others were developed in-house. These include: Description, DB2 Alias, Merge-Date. Other keywords were provided for optional use at the convenience of the user. These include: Notes, Comments, Values, etc.

We encountered some resistance in the user and DP communities when we attempted to implement these conventions. The resistance was based on their belief that there would be no benefit to them; they believed that their creativity in designing their data and data element names would be stifled. They also felt that the use of the abbreviations and classwords made the names meaningless. Most of all, their resistance was due to the fact that the standards meant they would have to change the way they were doing things.

To counter their objections, it was necessary for us to demonstrate the benefits of naming conventions and standards. We did this by showing how the consistency of the usage of data elements could be improved, how the documentation about data could be enhanced. Greater consistency and improved documentation would lead to improved accuracy when querying the dictionary and would also serve to reduce the inventory of data elements.

To counter their objections about the meaninglessness of the data names, we used Data Expediter to produce reports on existing data, showing the number of data element names with syllables containing specific numbers of characters. We found that in existing data elements, over 40% of the data elements contained syllables with four or fewer characters. We used this to show how the existing data elements were no more meaningful than those developed using Data Administration standards.

USF&G

DATA ADMINISTRATION

Standards

- New systems
- Linked to System Life Cycle and Secured Source

Developed for:

- items
- groups
- includes
- records
- programs
- called/calling modules

DATA ADMINISTRATION NAMING CONVENTIONS

Corporate Name

* Class Word, Prime Word, Other Modifiers

* Class Word list on DATAMANAGER

Member Name on Dictionary

* Standard Abbreviation List on DATAMANAGER

Aliases, versions, homonyms, synonyms not allowed in new development.

* Aliases - different data element names, same position in the file.
Same logical and physical lengths, same or different picture

```
Example:      BRANCH-OFFICE-CODE          PIC X(4).
              CLASS-CODE                  PIC X(4).
```

Both are located at position 135 in the same file.

* Versions - data elements with the same names and definitions but different formats

```
Example:      INSUREDS-NAME                   PIC X (35).
              INSUREDS-NAME                   PIC X (28).
```

DATA ADMINISTRATION STANDARDS (cont.)

* Homonyms, same name, different definitions and formats

Example:	ACCOUNT-NUMBER (Accounts Receivable System)	PIC 9(6).
	ACCOUNT-NUMBER (Accounts Payable System)	PIC 9(8).

* Synonyms, same name, same logical length, different pictures

Example:	BRANCH-OFFICE-CODE	PIC X(4).
	BRANCH-OFFICE-CODE	PIC 9(4).

KEYWORDS

Required

Keyword
Merge-Date
Alias
Etc.

Optional

Notes
Comments
Values
Etc.

Available in a dooument from Data Administration

RESISTANCE DUE TO:

o Belief that there is no benefit

o Lack of creativity in data design

o Names are not meaningful

o Means a change in the way they are doing things

OVERCOMING RESISTANCE

Demonstrate benefits of:

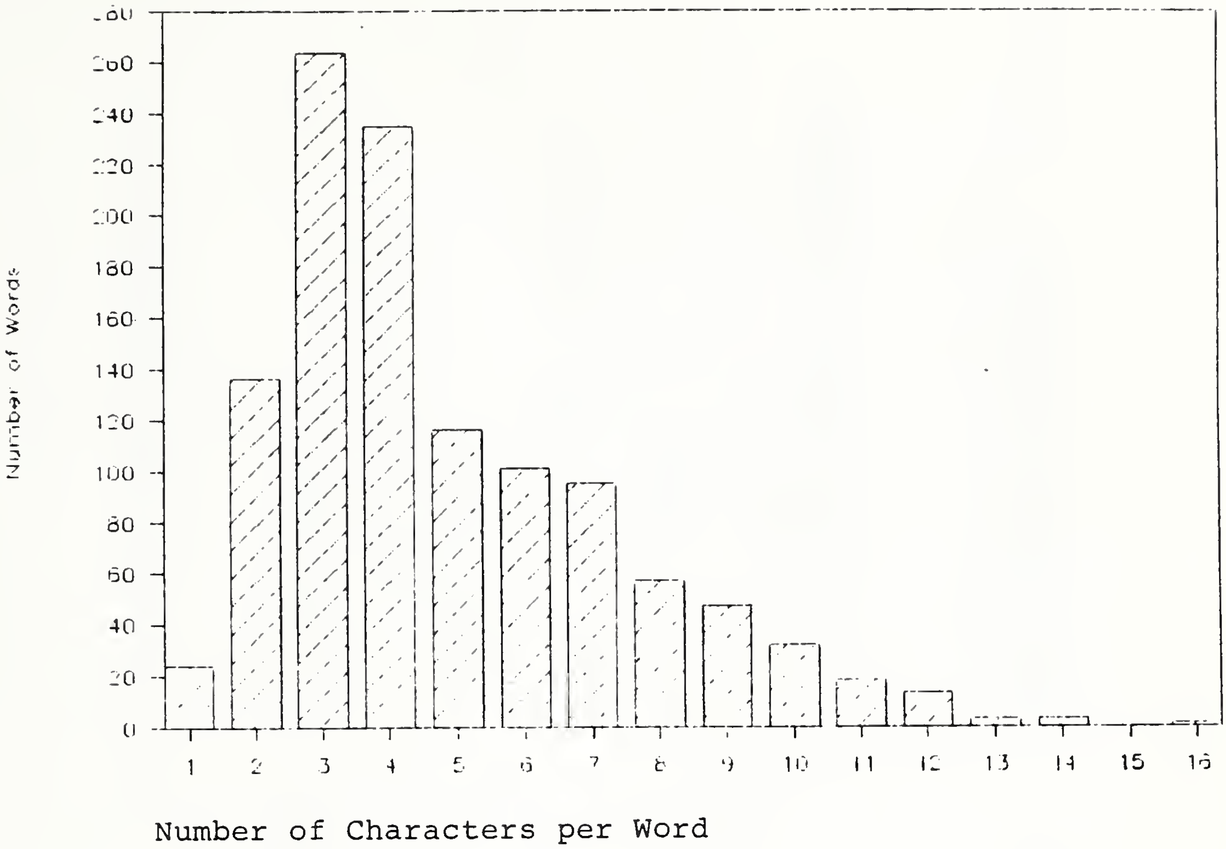
o Improved documentation

o Consistency of data use

o Improved accuracy of data queries

o Reduced inventory

Number of Words Containing N Characters



MANAGEMENT OF DATA ADMINISTRATION STANDARDS AT USF&G

NAMING CONVENTIONS FORUM

NOVEMBER 16-17, 1989

MANAGEMENT OF DATA ADMINISTRATION STANDARDS

- o STANDARDS FOR NAMING MUST BE ESTABLISHED AND ACCEPTED
- o STANDARDIZED VOCABULARY ASSISTS IN CONSTRUCTION
- o ENFORCEMENT OF STANDARDS IS TEDIOUS
- o AUTOMATION OF ENFORCEMENT IS ESSENTIAL

AUTOMATED MANAGEMENT TECHNIQUES

- o PROGRAMMER WORKBENCH
  - PM/SS
  - ISPF
  - MANAGER PRODUCTS
- o DATA EXPEDITER
- o MANAGER PRODUCTS

PROGRAMMER WORKBENCH FACILITY

DATA EXPEDITER

- o DATA ADMINISTRATION OPTIONS
  - STANDARD ABBREVIATION LIST

    ISPF
  - SYLLABLE CHECKING

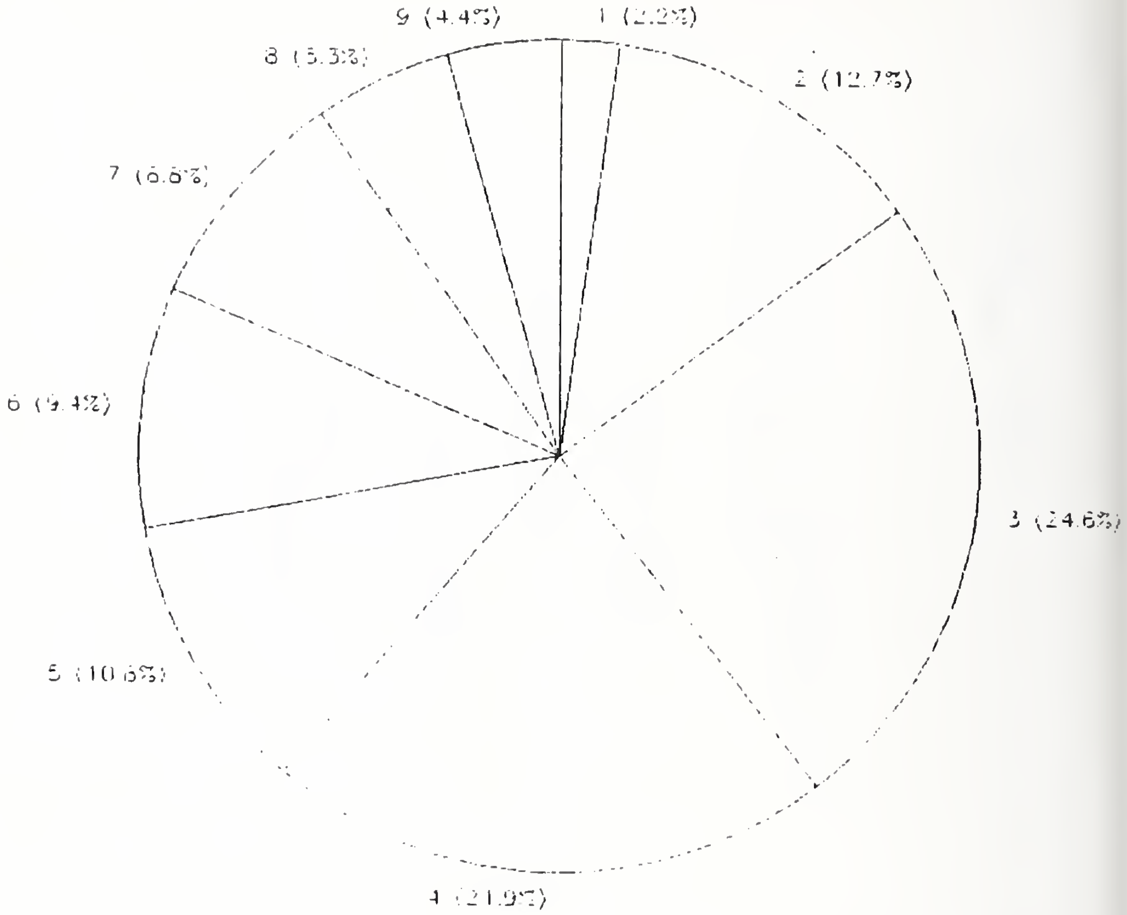
    PM/SS, DATAMANAGER
  - COBOL AND RECORD LAYOUT GENERATION

    DATAMANAGER, ISPF
  - AUTOMATIC ABBREVIATION

    ISPF

- o NAMECHECK OPTION VERIFIES:
  - CLASS WORDS
  - PRIME WORDS
  - SYNONYMS
  - STANDARD ABBREVIATIONS
  - PROPER SEQUENCE/POSITION OF COMPONENT WORDS

Percent Words Containing N Characters



FUTURE MANAGEMENT TECHNIQUES

o CONTROLMANAGER PROCEDURES LANGUAGE

- ACTIVE ENFORCEMENT

- NAMES CHECKED AT DICTIONARY ENTRY TIME

Naming Conventions Presentation

Lt. Col. Philip Olson, Jr

Headquarters, Department of the Army

ARMY DATA MANAGEMENT

and

STANDARDS PROGRAM

AR 25-9

INFORMATION BRIEFING

ARMY DATA MANAGEMENT and STANDARDS PROGRAM

- OBJECTIVES -

- Tackle Integration & Interoperability by managing Information Requirements down to data element level
- Provide a common framework for organizing data and information Army-wide
- Develop policy, standards, guidance and procedures to support current and future data sharing requirements

ARMY DATA MANAGEMENT and STANDARDS PROGRAM

- BASIC ASSUMPTION -

- Data is independent of and maintained separately from the applications that use the data.

ARMY DATA MANAGEMENT and STANDARDS PROGRAM

- ASSUMPTION IMPLICATIONS -

- Very long implementation time
- Major training requirements
- Significant management commitment

ARMY DATA MANAGEMENT and STANDARDS PROGRAM

– STANDARDIZATION OF STRUCTURE –

- Domain Concept
- Global View of Data
- Single Element Concept
- Change in Thinking

DATA ELEMENT STANDARDIZATION

- PROGRAM CONSTRUCTS -

- Naming Conventions: Rules used to name standard elements and attributes
- Standard Attributes: Data values used to describe standard elements
- Reference Element: A generic data domain based on a class word
- Data Element: A domain of data values specified by "what" the data is, not how it is used
- Data Element Alias: A non–standard data element currently in use in a system

Naming Syntax

General Format: M:PW:M:CW:Q

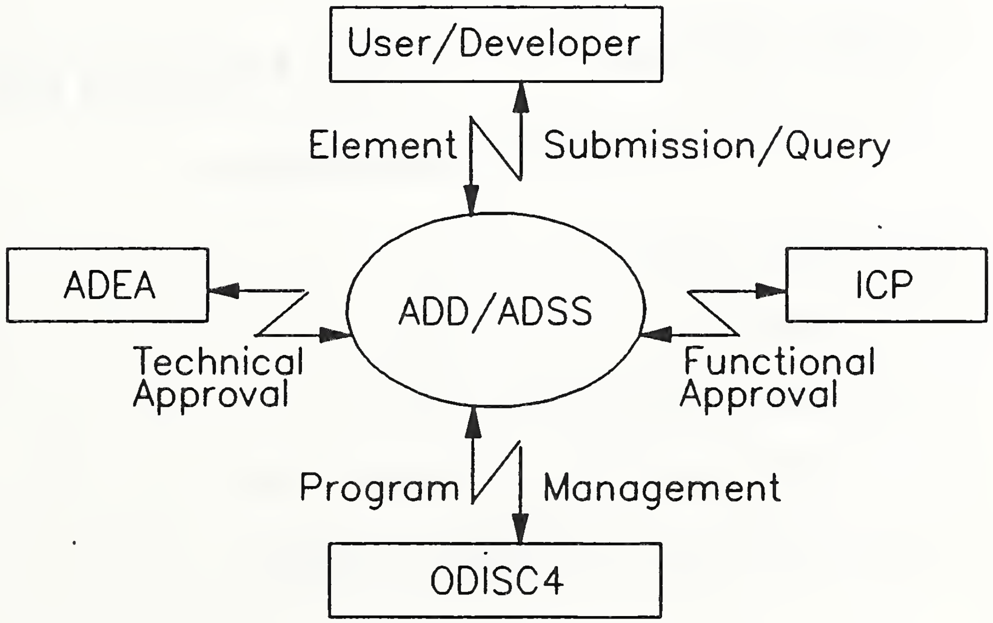
CW = Class Word: A word used to specify the type of information contained in a domain.

M = Modifier: A word which helps to refine, describe or render a name unique for a data element which is not designated a prime or class word.

PW = Prime Word: A word used in a data element name which represents the data grouping to which the data element belongs.

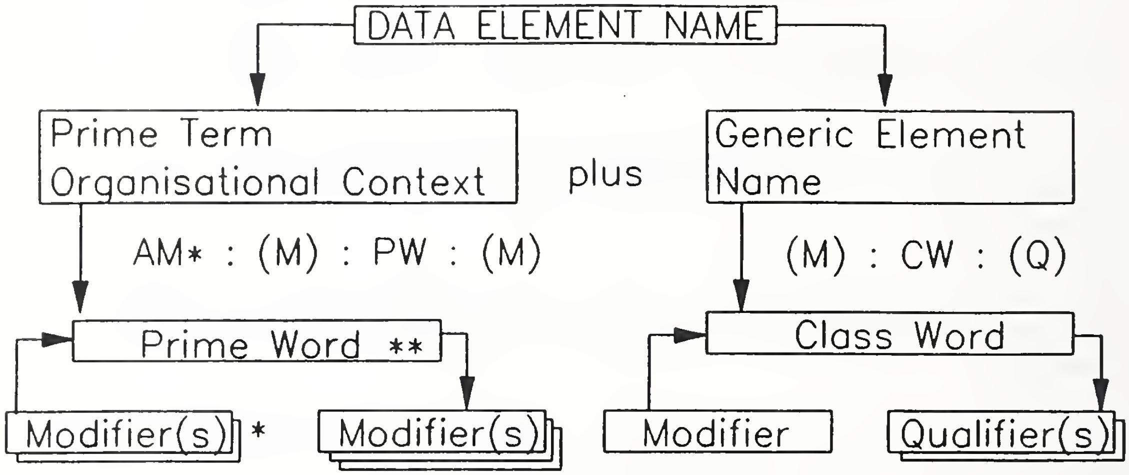
Q = Qualifier A word used with a class word to further describe a characteristic of the information within a domain.

DATA STANDARDIZATION TOOL



DATA ELEMENT NAME CONSTRUCTION

GENERAL FORMAT: M:PW:M:CW:Q



DATA ELEMENT NAME

Prime Term Organisational Context

plus

Generic Element Name

AM* : (M) : PW : (M)

(M) : CW : (Q)

Prime Word **

Class Word

Modifier(s) *

Modifier(s)

Modifier

Qualifier(s)

** The Prime Word does not have a fixed position in the Prime Term.

* The Architectural Modifier may also be the Prime Word.

DATA ELEMENT STANDARDIZATION

- DATA ELEMENT ATTRIBUTE EXAMPLES -

- Information Data Element Name
- Information Element Approval Date
- Information Element Definition Text
- Information Element Domain Definition Text
- Information Element Justification Category

DATA ELEMENT STANDARDIZATION

- STANDARD ELEMENT ATTRIBUTES -

■ Information Reference Element Name

■ Information Element Data Value Name

■ Information Element Definition Text

■ Information Element Domain Definition Text

■ Information Element Approval Date

- Information Data Element Name
- Information Data Element Modifier Name
- Information Data Element Mneumonic Abbreviation
- Information Data Value Type Identifier

□ Information Data Element Alias Name

□ Information Data Element Alias Host System Name

Naming Conventions Presentation

Mr. Charles Spasaro

United States Postal Service

IRDS WORKSHOP

NAMING CONVENTIONS FORUM

U.S. POSTAL SERVICE

PRESENTED BY: CHUCK SPASARO

NOVEMBER 1989

INTRODUCTION

- Background
- Environment
- Software Tools
- Enforcement
- User Reaction
- Management Support

NAMING CONVENTION

- Data Elements
- Language Synonyms
- Other Entities

LESSONS LEARNED

INTRODUCTION

- Background

- Business System Planning Study Completed	Apr. 1980
- Corporate Dictionary Established	Apr. 1983
- Draft Naming Standards Issued	Jun. 1983
- First Formal Naming Standards Issued	Oct. 1986
- Reference Card Created	Feb. 1987
- Current Naming Standards Issued	Apr. 1989

- Environment
  - DBMS'
    - IDMS
    - Oracle
    - (Focus)
  - Application Development Sites
    - New York
    - Raleigh
    - St. Louis
    - Wilkes-Barre
    - Minneapolis
    - San Mateo

- Data Management Functions
  - Headquarters
    - Logical Data Base Design
    - Data Administration
    - Data Element Naming and Definition
  - Data Centers
    - Physical Data Base Design
    - Local Data Administration
    - Data Element Naming and Definition
    - Physical Entity Naming

- Software Tools
  - Directory of Information Resources (DIR)
  - IDMS/IDD
  - ORACLE/SDD

- User Reaction
  - Functional Organizations
    - Some Problems But Generally Supportive
    - Resource Availability Usually an Issue
  - Application Developers
    - Use Standard Names if Available
    - Names Too Long
    - Names Not Application Friendly
    - Standardization Process Slows Progress
- Management Support
  - Top and Middle Management
  - Front-line Supervisors

  - Project Work Station
    - Excelerator
    - ADL IRMA
    - IDEA
    - Utilities
- Enforcement
  - Corporate Dictionary/Field Dictionary Interface System (Data Elements)
    - Automatic Updating of Field Dictionaries
    - Transfer of New Elements to Field Centers.
    - Identification of Mismatches
    - Extraction of Proposed New Elements from Field Centers
  - Local Data Administrator
  - Annual Field Reviews
  - Training
  - Project QA Reviews

NAMING CONVENTION

- Data Elements
  - Maximum Length = 30 characters
  - Must consist of:
    - Subject
    - Suffix (Class Word)
  - Segments must appear in the following order:
    - Subject Qualifier
    - Subject
    - Suffix Qualifier
    - Suffix
  - Suffix must be on approved list
  - Suffix qualifier must be abbreviated on list
  - No intelligence that's subject to change
  - For role elements, qualify root name with role prefix
  - Omit special characters (excluding blanks), prepositions, conjunctions, and articles (unless required to form a term)
  - Full words must be used, except for:
    - Standard suffixes and suffix qualifiers
    - Universal USPS acronyms (e.g., ZIP)
    - When required to meet 30 character limit
  - Use only approved abbreviations, when required
- LANGUAGE SYNONYMS
  - IDMS/COBOL

    Standard name with special characters replaced by dashes (-)
  - ORACLE

    Standard name with special characters replaced by underscores (_)
  - FOCUS

    Manually created 12 character name

- Other Entities
  - Application Name
    - Long Name
    - Short Name
  - Record Name
  - Table name
  - Etc.

LESSONS LEARNED

- Test standards in pilot application before formal issuance
- Consider impact of standards on application developers
- Allow shortened language synonyms developed according to standard algorithm or procedure
- Develop elements (names) in a managed data environment:
  - Logical data base design
  - Data issue resolution
- Automate as much as possible
- Develop workable solutions to:
  - Technical elements versus corporate elements
  - Working storage elements
  - Derived elements
  - Root/role elements

Naming Conventions Presentation

Mr. Bao Nguyen

Headquarters, United States Air Force

AIR FORCE DATA MANAGEMENT AND STANDARDS PROGRAM

POLICY

- AFP 700-50, Volume III, Air Force Communications and Computer Systems Architecture, Data Management (15 December 1989)

This document addresses the data issue to support the Air Force Communications and Computer Systems Architecture.

- AFR 4-29, Air Force Data Management and Standards Program (to be published in Jan 90)

This regulation:

-- Provides the data concept

-- Provides Air Force-wide policy to manage data as a corporate asset

-- Defines organizational responsibilities

-- Provides data naming convention rules to standardize data elements

DATA DICTIONARY

Air Force Corporate Data Dictionary Development

-- Hardware: AT&T 3B2 and ShareBase Data Base Machine

-- Software: Freeform

-- Location: Gunter Air Force Base in Alabama

-- Accessability: 24 hour on-line via DDN or Dial-up

-- IRDS Compliant

MILESTONES

-- H/W and S/W installation	Dec 89
-- Prototype completion	Feb 90
-- Overhauled current DD load	Apr 90
-- Fully operational	May 90
-- Standardized Data Elements Load Start	May 90
-- Data Element Naming Process Automated	Dec 90

USAF DATA MANAGEMENT & STANDARDS PROGRAM

DATA NAMING CONVENTION

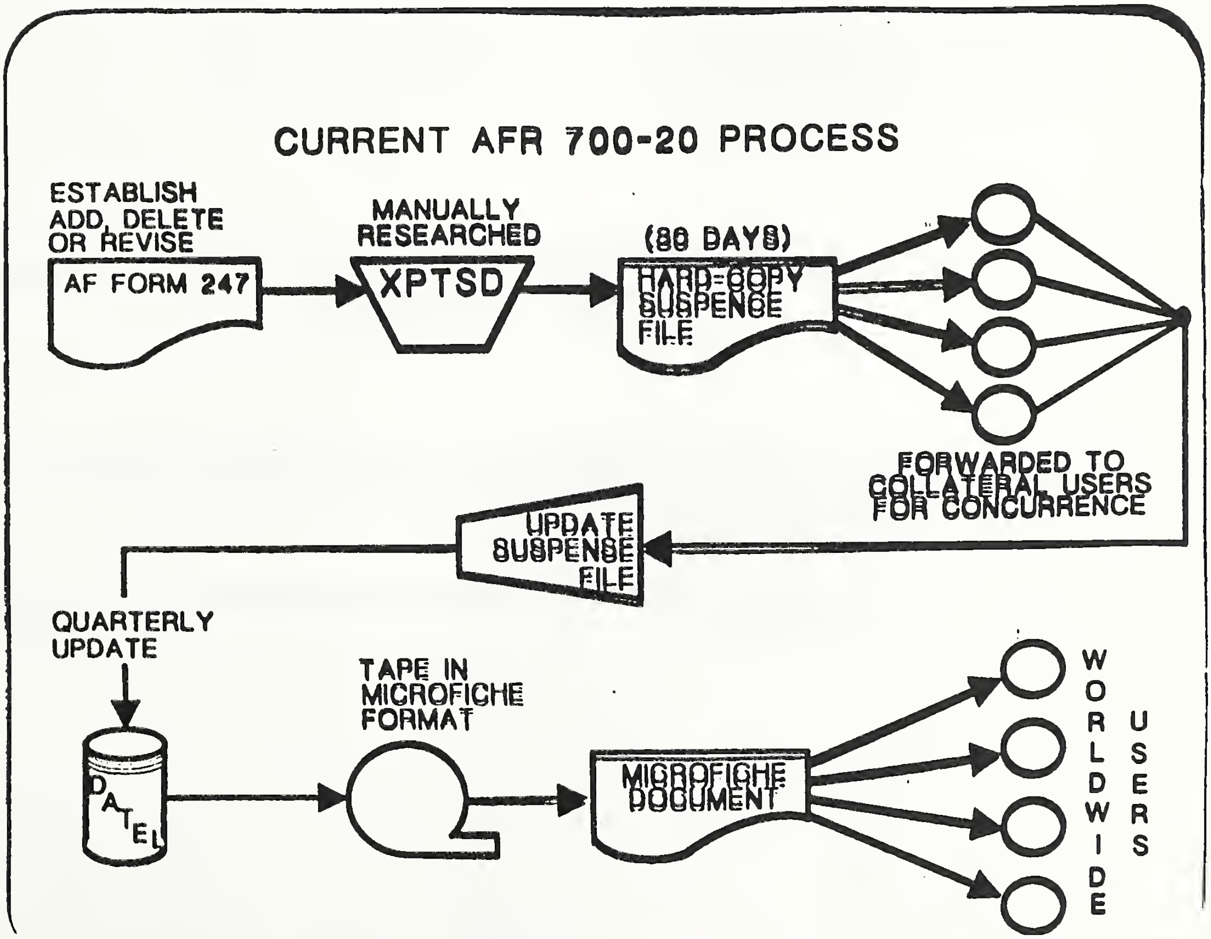
USAF DATA MANAGEMENT & STANDARDS PROGRAM

DATA NAMING CONVENTION - OVERVIEW

- Current situation
- Current efforts

CURRENT DATA STANDARDIZATION ENVIRONMENT

- AFR 700-9 INFORMATION SYSTEMS STANDARDS (POLICY)
- AFR 700-19 COMPUTER SYSTEMS AUTHORIZATION DIRECTORY (SYSTEMS DESCRIPTIONS)
- AFR 700-20 AIR FORCE DATA DICTIONARY (AF STANDARD DATA ELEMENTS AND CODES)



CURRENT AFR 700-20 PROCESS

ESTABLISH ADD, DELETE OR REVISE

AF FORM 247

MANUALLY RESEARCHED

XPTSD

(60 DAYS)

HARD-COPY SUSPENSE FILE

FORWARDED TO COLLATERAL USERS FOR CONCURRENCE

UPDATE SUSPENSE FILE

QUARTERLY UPDATE

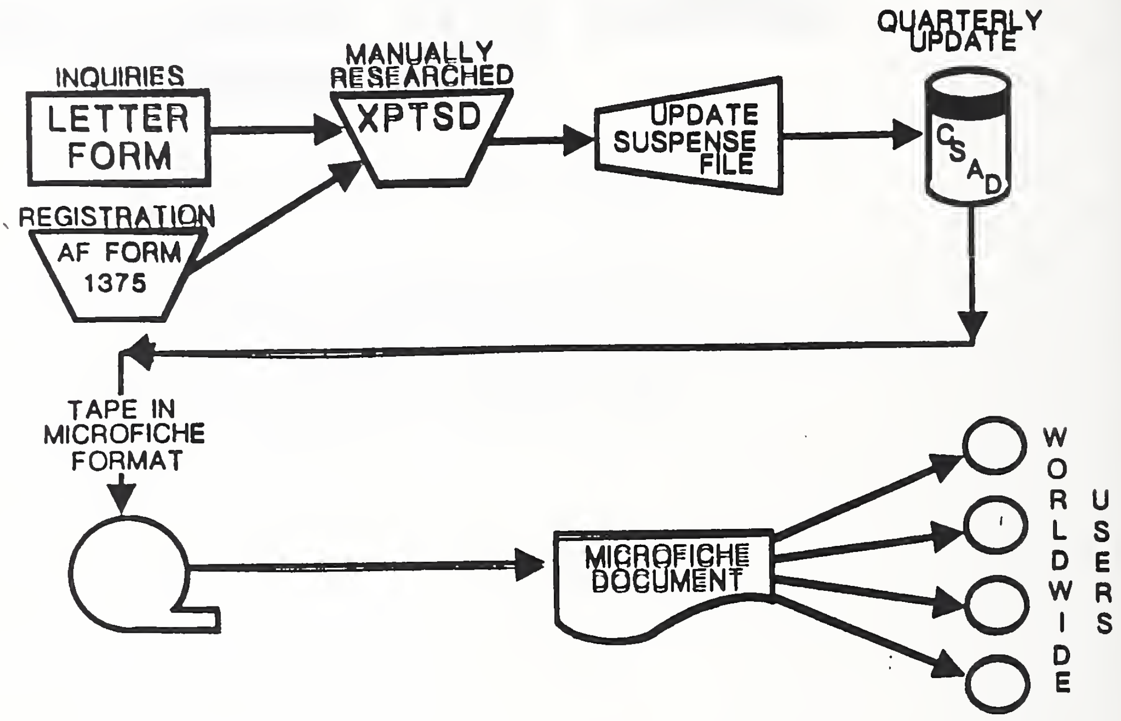
DATEL

TAPE IN MICROFICHE FORMAT

MICROFICHE DOCUMENT

WORLDWIDE USERS

CURRENT AFR 700-19 PROCESS



INQUIRIES

LETTER FORM

REGISTRATION

AF FORM 1375

MANUALLY RESEARCHED

XPTSD

UPDATE SUSPENSE FILE

QUARTERLY UPDATE

CSAD

TAPE IN MICROFICHE FORMAT

MICROFICHE DOCUMENT

WORLDWIDE USERS

USAF DATA MANAGEMENT & STANDARDS PROGRAM

POLICY

- AFP 700-50, VOLUME III, AIR FORCE COMMUNICATIONS AND COMPUTER SYSTEM ARCHITECTURE, DATA MANAGEMENT
- AFR 4-29, AIR FORCE DATA MANAGEMENT AND STANDARDS PROGRAM

USAF DATA MANAGEMENT & STANDARDS PROGRAM

SYNTAX FOR NAMING CONVENTION

- M:PW:M:CW:Q
- M: Max of 4 for PW, 1 for CW
- Q: Optional, Max of 2

USAF DATA MANAGEMENT & STANDARDS PROGRAM

RULES FOR NAMING CONVENTION

13 RULES

- Rule 1: Each generic element name will contain one and only one class word.

  Comments: By restricting the generic element name to one class word, the standard element name will describe only one type of information collected about an object.

- Rule 2: Class words are reserved; i.e., do not be use them as qualifiers or prime terms.

- Rule 3: Each data element name will contain only one prime word and describe only one concept.

  Comments: (1) By requiring a data element name to have one prime term, the data element is formulated to explicitly describe only one concept.

  (2) The end user may optionally use a term from the prime word list as a modifier.

- Rule 4: Use the following sequence of words and format in a data element name: Modifier(s) (if required), Prime Word, Modifier(s) (if required), Class Word, Qualifier(s) (if required).

- Rule 5: Each data element name will include its related generic element name.

- Rule 6: Do not use plurals of prime words unless the plural has a different meaning; never use plurals of class words.

  Comments: Removing plurals from data element names encourages the designer to think in terms of fundamental concepts and increases the possibility that two people will develop the same name to describe identical concepts.

- Rule 7: Use modifiers and qualifiers to fully describe a standard element (up to four modifiers per prime word and one modifier plus two qualifiers per class word).
- Rule 8: Preserve the normal word order of commonly used terms will be in Data Element Alias names (e.g., Port of Debarkation, Department of Defense).
- Rule 9: Apply a unit of measure suffix to the names of all elements that describe a numeric quantity (e.g., Volume-in-Liters).
- Rule 10: Do not use any abbreviations or acronyms in a standard element name.
- Rule 11: Use only alphabetic characters (a-z) in a standard element name.

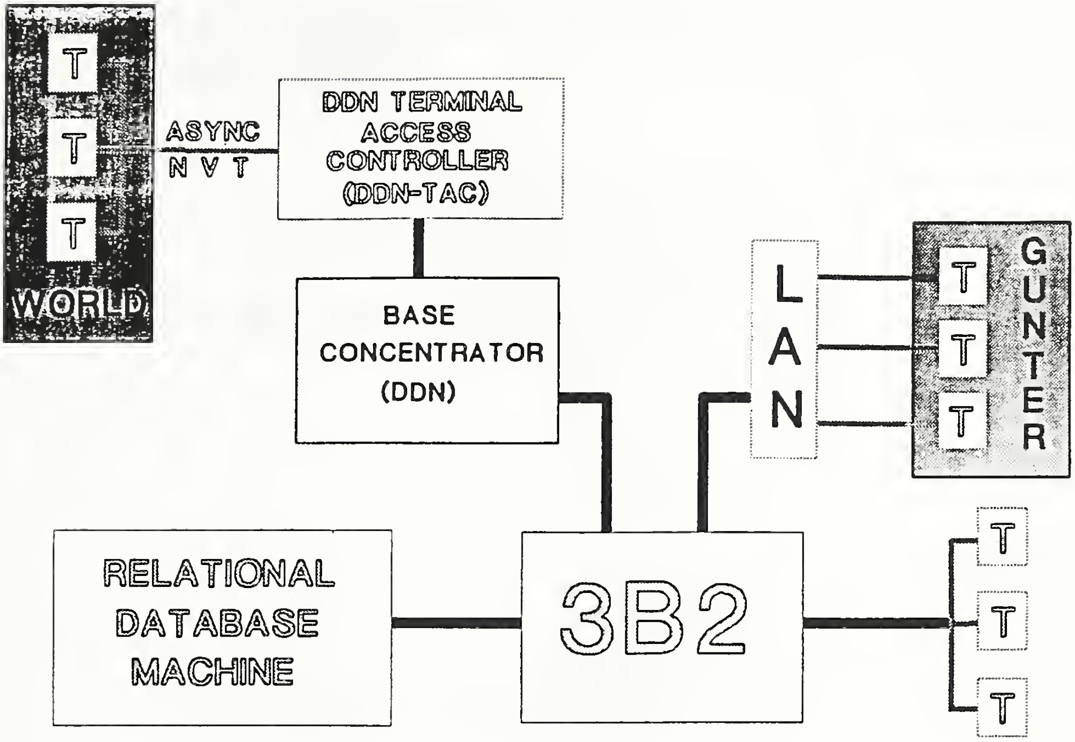
There are two exceptions to rule 11:

(1) Use a hyphen to connect the words in a prime term or generic element name.

(2) Use a number when it is part of a descriptive name e.g., F-16 Fighter.

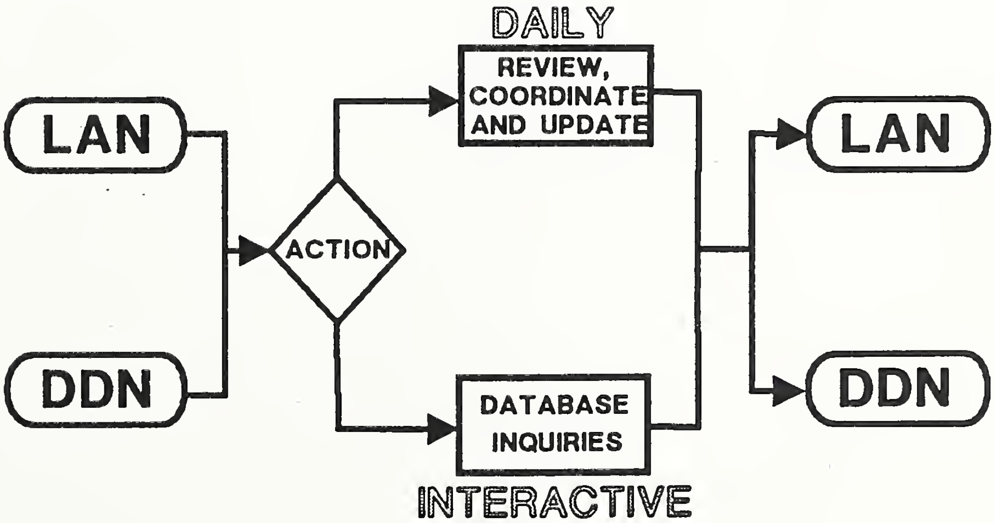
Comments: By permitting only alphabetic characters, standard element developers are encouraged to describe standard element names in terms of what the data is and not how it is stored or used. This rule also improves the probability that different people will develop the same name for identical standard elements.

- Rule 12: Do not use names of organizations, computer or information systems, directives, forms, screens, or reports in standard element names.
- Rule 13: Do not use titles of blocks, rows, or columns of screens, reports, or listings in standard element names unless those titles satisfy rules 1-11.



AIR FORCE CORPORATE DATA DICTIONARY (AFCDD)

PROPOSED AFR 700-19/AFR 700-20 PROCESS



USAF DATA MANAGEMENT & STANDARDS PROGRAM

PROBLEMS

- LACK OF DOD POLICY
- DATA ELEMENT NAMES CAN BE TOO LONG

Naming Conventions Presentation and WIS/DIM Demonstration

Major Reed Borman

The Joint Staff

JCS
DATA ADMINISTRATION
IN
WWMCCS

MAJOR REED BORMAN

JCS J-7

(AV) 225-7745
COMM (202)695-7745

NAMING CONVENTIONS

DATA
ELEMENT
NAMING
CONVENTIONS

NAMING CONVENTIONS

- ◘ JCS PUB 19, ANNEX M, APPENDIX D CONTAINS 13 NAMING CONVENTION RULES
- ◘ FOLLOWS THE GUIDELINES SET BY THE NATIONAL INSTITUTE FOR STANDARDS AND TECHNOLOGY (NIST)

NAMING CONVENTIONS

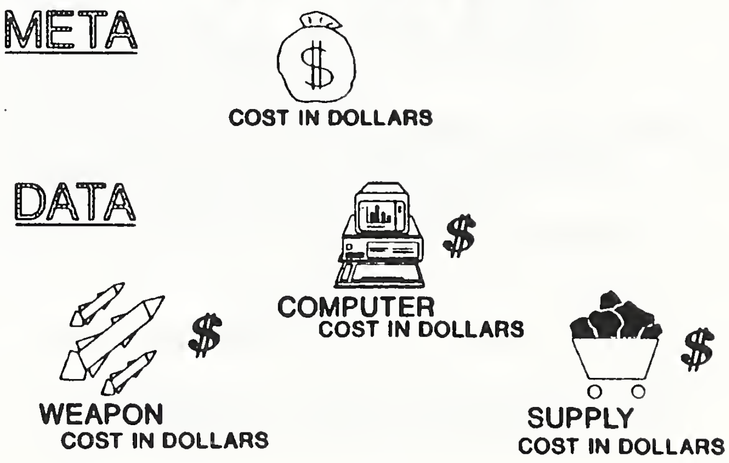
DEFINITIONS

PRIME TERM: WORD THAT DESCRIBES THE TYPE OF OBJECT BEING DESCRIBED

CLASS TERM: A WORD THAT IDENTIFIES THE TYPE OF DATA THAT DESCRIBES AN OBJECT

META DATA: DATA ABOUT DATA

ELEMENTS



META

COST IN DOLLARS

DATA

COMPUTER
COST IN DOLLARS

WEAPON
COST IN DOLLARS

SUPPLY
COST IN DOLLARS

META ELEMENT

- MODEL & REFERENCE FOR SIMILAR DATA ELEMENTS
- DESCRIBES DATA ELEMENTS' DOMAIN
- DESCRIBES TARGET STANDARD

NAMING CONVENTIONS

RULES

1. META-ELEMENTS HAVE ONE CLASS TERM
2. CLASS TERMS ARE RESERVED
3. DATA ELEMENT NAMES WILL CONTAIN ONE PRIME TERM & DESCRIBE ONE CONCEPT
4. SEQUENCE OF WORDS (MODIFIER, PRIME TERM, MODIFIERS, CLASS TERM, & UNITS OF MEASURE
5. DE NAMES WILL INCLUDE A CLASS TERM
6. NO PLURALS OF CLASS/PRIME TERMS
7. QUALIFIERS WILL BE USED

NAMING CONVENTIONS

RULES (CONT'D)

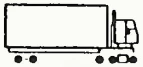
8. ORDER OF COMMON TERMS PRESERVED
9. UNIT OF MEASURE WILL BE ADDED WHERE APPROPRIATE
10. NO ABBREVIATIONS OR ACRONYMS
11. ONLY ALPHABETIC CHARACTERS PERMITTED
12. NAMES OF ORGANIZATIONS NOT PERMITTED
13. TITLES OF SCREENS NOT PERMITTED

NAMING CONVENTION

- WHAT? --RULES
  - ✓ FORMAT
  - ✓ SEQUENCE
  - ✓ VOCABULARY
- WHY? --ENSURE CONSISTENCY
  - ✓ DATA ELEMENT NAMES
  - ✓ SHORT NAMES - MNEMONICS
- HOW?

WORDS/TERMS

- PRIME
  - ✓ Object described



- CLASS
  - ✓ Type of information

- UNIT OF MEASURE
  - ✓ For quantitative data

NAMING CONVENTION

MODIFIER(S)-PRIME-MODIFIER(S)-CLASS-UNITS

Naming Conventions Presentation

Mr. George Ratte

United States Treasury

Automated Naming Interface to IRDS

George P. Ratte
Financial Management Service
U.S. Department of the Treasury

The U.S. Department of Treasury, Financial Management Service (FMS) has implemented an online repository to assist with the management of data as corporate asset. The controlled repository, referred to as the Information Resource Dictionary System (IRDS), is to assist with the management of the data asset and accomplish the following specific objectives within FMS:

> Provide comprehensive and standardized documentation of computer systems and ensure its quality, authenticity and timeliness.
>
> Facilitate and assist in the logical integration of data into the data base environment.
>
> Provide better planning and strategic information in support of operations and data processing systems development.
>
> Make information about the organization's data and systems resources more available to the users, both within data processing and within the user departments.
>
> Support the FMS systems development methodology.

Basic facilities of the IRDS include the production of reports for all documentation requirements, interrogation capabilities, and source language generation for COBOL, Assembler, DB/2, and Cullinets' Integrated Database Management System (IDMS). These facilities are available online. However, implementation within the systems development life cycle is not yet in place.

We are currently in the process of modifying existing standards and developing new standards for the IRDS. These standards include procedures for using the IRDS, and the data element naming facility.

The IRDS has been implemented using front-end software known as HUGO and automated naming facility called $NAME. This interface creates a more friendly environment for data entry into the IRDS and provides a mechanism for standardizing the meta-entity naming process.

Our initial implementation of $NAME was to control and standardize names for low-level objects such as data elements and groups.

These objects require greater control, in terms of identification and naming, than other objects, due to the large number of these within the organization. The names which are generated for these objects will also be used in COBOL applications.

$NAME control is currently being added to other object types such as records, reports and documents. We are currently undecided as to whether the $NAME facility should be used in naming all object types or only for selected object types.

HARDWARE/SOFTWARE ENVIRONMENT:

0 IBM 3084, MVS/XA Operating System
0 TSO/ISPF
o DATAMANAGER Release 7.0 (MSP Inc.)
o HUGO/ISPF Front-end (Global Software Inc.)
0 $NAME Automated Naming Interface (Global Software Inc.)

CAPABILITIES:

General capabilities of the automated naming interface include the following:

0 The interface generates unique names for meta-entity objects based on the business name, or short description of the object. The business name is optionally stored under a named alias (which is indexed) or attribute of the object (not indexed). FMS elected to store the business name as a text attribute called BUSINESS-NAME, rather than use one of the alias name slots. The system currently limits the number of alias names for an object to 16.

o It enforces naming rules based on a standard abbreviation list consisting of PRIME words, CLASS words, MODIFIER words, and NULL words (stop words). There can be multiple abbreviation lists to accommodate different naming schemes. Class words and prime words can be made mandatory or optional. Class words and prime words can vary, depending on object type.

- It enforces the positioning of the term within the resulting name. This is usually applicable to class words and prime words where the designated position is either first or last.

- It provides tailoring of naming rules by meta-entity type. An installation can select which object types, or groups of object types, are to be subject to $NAME processing.

- It provides the capability to generate alias names for objects and to specify the characteristics of the alias name based on alias type (COBOL, SQL, ASSEMBLER, etc). It optionally enforces uniqueness rules for data object instance aliases across data object instances.

- It automatically catalogues terms and abbreviations as key words for objects, enabling retrieval by subject area, terms, key words or acronyms.

- It performs redundancy checking against dictionary objects with similar term combinations in the name. If an object is found with the same or similar term combinations, the user is given the opportunity to review the existing object prior to continuing with the add function.

EXAMPLE:

A typical scenario for adding a new data element is as follows:

USER: Logs on to the system and selects "I" for IRDS.

SYSTEM: Verifies logon ID with IRDS profile. This profile establishes access authority within the IRDS, for each user.

USER: Selects ADD/REVISE function.
Selects object type DATA ELEMENT, and enters a free-form business name for the object.

SYSTEM: Validates each term of the business name against the standard abbreviation list.
Checks for an occurrence of a prime term and one class term.
Issues error message if above criteria not met,
OR
Displays the generated data element name and indicates that there is a possible duplicate in the IRDS,

OR

Displays and accepts the generated data element name and displays the appropriate attributes for selection.

USER: Either

1. Aborts the operation, or
2. Browses the possible duplicate object, or
3. Selects the desired optional attributes for the add operation. (Attributes can be made optional or mandatory).

POTENTIAL PROBLEMS:

If the automated naming tool is introduced after an IRDS is partially populated, there may be many ramifications. Many names will need to be changed, in addition to references to these names. These name changes may also effect existing applications outside the IRDS. Attempting to retro-fit existing applications, due to the introduction of an automated naming tool may not be acceptable. The organization may not be willing to undertake a conversion and/or rename effort to bring all object names in synchronization with the naming facility.

An automated naming interface may be ineffective when objects are added with an inappropriate business name. This is very often the case since many objects are named by someone in data processing or someone not knowing the appropriate definition. To alleviate this problem an owner or custodian must be designated for each object. The business name would require approval by the owner or custodian of the object before it is added to the IRDS.

The $NAME interface stores a special hash code with each object to enable it to perform its own redundancy checking against the IRDS. If the naming facility or the IRDS allows this hash code to be inadvertently removed, or changed for objects, then the naming interface would be less effective.

The $NAME interface does not have semantic support nor the ability to associate an object to a higher level object. It is somewhat rule based, but not comparable to artificially intelligent (Al) systems. Rules are established by supplying appropriate parameters for Assembler language macros, which require re-compilation after making changes. Establishing the rules and making changes can become fairly complex and cumbersome for the data administrator. An automated naming interface should be fairly easy to implement, tailor, and maintain.

SUMMARY:

An automated naming interface to the IRDS is an important tool for the Data Administrator. This is particularly true during the start-up operation and initial population of an IRDS. It controls the naming process and performs much of the research required to determine the pre-existence of an object. This capability should be available from the start. If an IRDS is in place within an organization, prior to introducing an automated naming tool, the consequence may not be so desirable. I think it would be appropriate for the IRDS to provide an audit and conversion capability for the data administrator.

Tools available today appear to be dependent upon the particular IRDS software used. $NAME, for example, only works with Datamanager. It relies on the user exit interface and issues Datamanager queries against the IRDS prior to determining the appropriate response to the user. It may be desirable for the software to interface with multiple repositories, possibly operating at different locations, using different repository software.

The introduction of standards pertaining to the implementation of an automated naming interface may do much to alleviate some of the data administrators' problems in the future.

AUTOMATED NAMING INTERFACE
TO
INFORMATION RESOURCE DICTIONARY SYSTEM

U.S. DEPT OF TREASURY
FINANCIAL MANAGEMENT SERVICE
DATA ADMINISTRATION BRANCH

George P. Ratte

November 16, 1989

FMS PROGRAMS AND INITIATIVES:

- CENTRAL ACCOUNTING AND REPORTING FOR ENTIRE GOVERNMENT
- MANAGE PAYMENTS & COLLECTIONS
  (750 Million Payments Per Year)
  (1 Trillion dollars in Collections)
  (1 Million Claims Per Year)
- MANAGE DIRECT DEPOSIT,
  ELECTRONIC CERTIFICATION,
  ELECTRONIC FUNDS TRANSFER
  AND OTHER GOVERNMENT FINANCIAL SYSTEMS

OUTLINE

- GENERAL DESCRIPTION
    - TOOLS USED
    - GENERAL CAPABILITY
    - TAILORING OPTIONS
- DEMO FROM ONLINE SESSION
- USER REACTION

TOOLS USED

- DATAMANAGER
  (Manager Software Products Inc.)
- HUGO FRONT-END SYSTEM
  (Global Software Inc.)
- $NAME AUTOMATED NAMING PROGRAM
  (Global Software Inc.)

GENERAL CAPABILITY

- VALIDATES TERMS AND ABBREVIATIONS.
  Each term in the supplied business name is validated against the installation's list of approved terms, acronyms and abbreviations.

- GENERATES UNIQUE NAMES FOR META-ENTITY OBJECTS.
  Using the standard terms derived from the lookup, the system searches the repository to determine if the same terms, possibly in a different order, have been used to describe a previous object.

- ENFORCEMENT OF CLASS WORD/PRIME WORD RULES.
  Enforces the positioning of the class word and prime word within the resulting name.

- NAMING RULES CAN BE TAILORED BY META-ENTITY TYPE.
  Different object types can have different abbreviation rules.

- GENERATES ALIAS NAMES FOR OBJECTS.
  Can specify the characteristics of alias names and optionally enforce uniqueness rules for data object instance aliases across data object instances.

- AUTOMATICALLY CATALOGUES TERMS, ABBREVIATIONS, USER LOGON ID, OBJECT TYPE, PROJECT.
  Maintained as part of the Datamanager Catalog clause.

TAILORING OPTIONS

- o CONTROL MAXIMUM SIZE OF ABBREVIATED TERM

- o CONTROL INDEXING OF TERMS AND ABBREVIATIONS (KWIC and KWOC entries)

- o CLASS WORD/PRIME WORD MANDATORY OR OPTIONAL

- o CONNECTOR CHARACTER (Hyphen, Underscore)

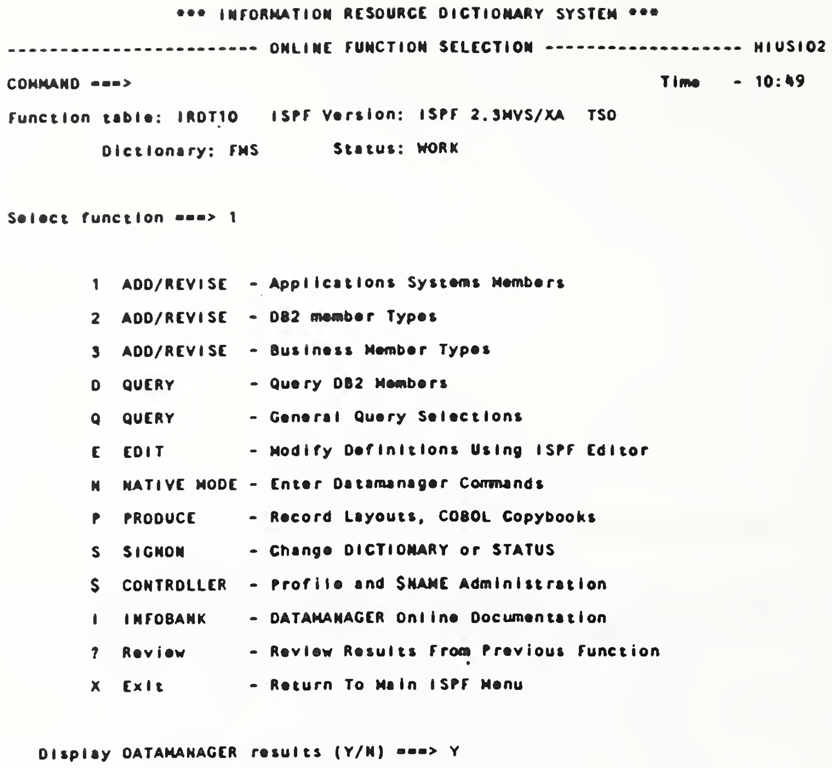
- o WHEN TO ABBREVIATE
  - Always
  - Only When Too Long
  - From Left to Right When Too Long

- o MAXIMUM LENGTH OF GENERATED NAME

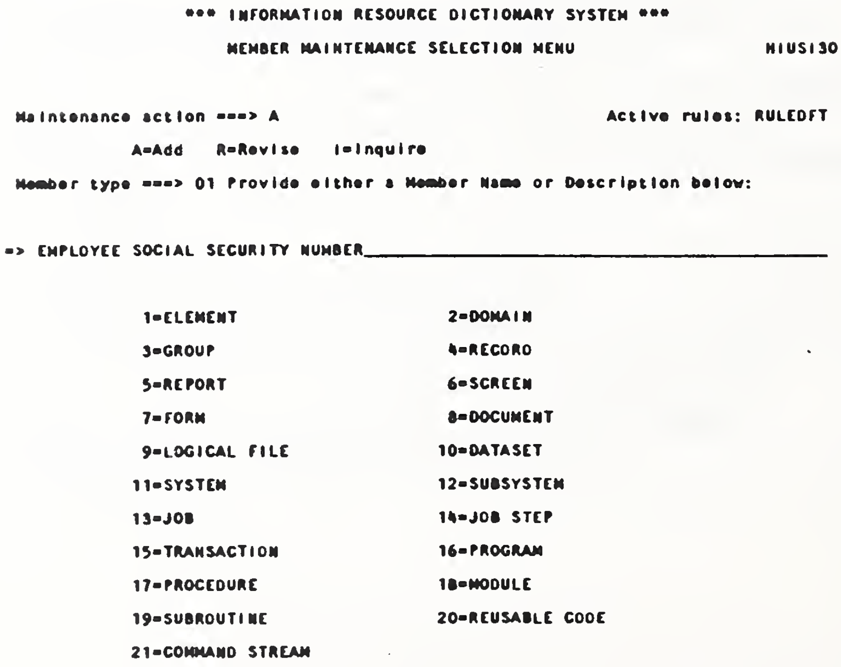
- o ALLOW NUMBERS (Y/N)

- o PERFORM REDUNDANCY SEARCH (Y/N)

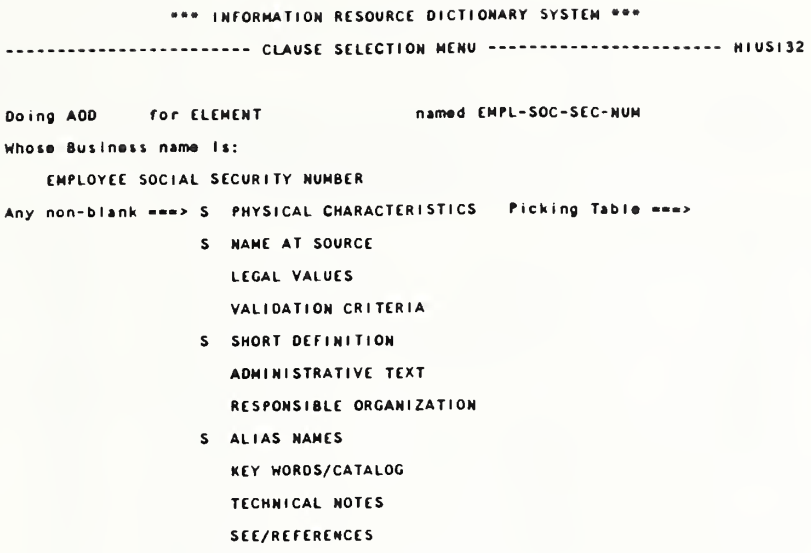
- o PLACEMENT OF BUSINESS NAME
  - Not Stored
  - Stored as Alias Name
  - Stored Under Text Attribute.



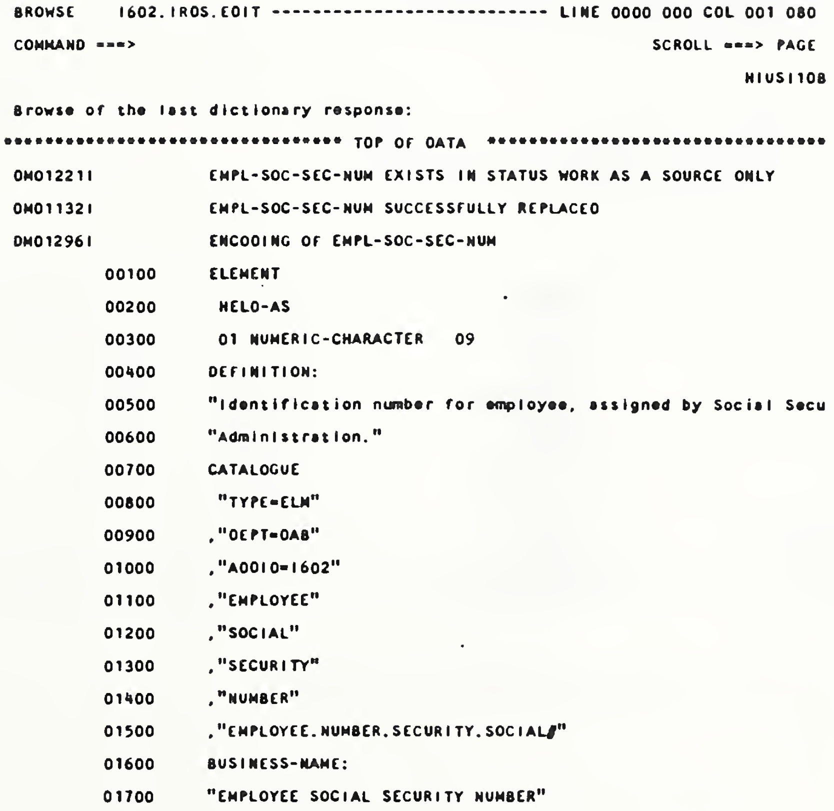
```
                    *** INFORMATION RESOURCE DICTIONARY SYSTEM ***
------------------------ ONLINE FUNCTION SELECTION ------------------- HIUSI02
COMMAND ===>                                                      Time   - 10:49
Function table: IRDT10   ISPF Version: ISPF 2.3MVS/XA  TSO
         Dictionary: FMS        Status: WORK


Select function ===> 1


        1 ADD/REVISE  - Applications Systems Members
        2 ADD/REVISE  - DB2 member Types
        3 ADD/REVISE  - Business Member Types
        D QUERY       - Query DB2 Members
        Q QUERY       - General Query Selections
        E EDIT        - Modify Definitions Using ISPF Editor
        N NATIVE MODE - Enter Datamanager Commands
        P PRODUCE     - Record Layouts, COBOL Copybooks
        S SIGNON      - Change DICTIONARY or STATUS
        $ CONTROLLER  - Profile and $NAME Administration
        I INFOBANK    - DATAMANAGER Online Documentation
        ? Review      - Review Results From Previous Function
        X Exit        - Return To Main ISPF Menu

   Display DATAMANAGER results (Y/N) ===> Y
```



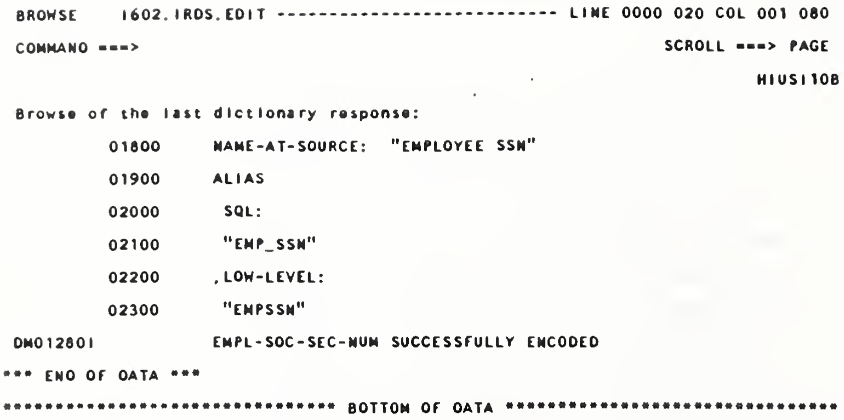
```
                 *** INFORMATION RESOURCE DICTIONARY SYSTEM ***
                     MEMBER MAINTENANCE SELECTION MENU                HIUSI30

Maintenance action ===> A                                 Active rules: RULEDFT
           A=Add   R=Revise   I=Inquire
Member type ===> 01 Provide either a Member Name or Description below:

=> EMPLOYEE SOCIAL SECURITY NUMBER_________________________________________

             1=ELEMENT                      2=DOMAIN
             3=GROUP                        4=RECORD
             5=REPORT                       6=SCREEN
             7=FORM                         8=DOCUMENT
             9=LOGICAL FILE                10=DATASET
            11=SYSTEM                      12=SUBSYSTEM
            13=JOB                         14=JOB STEP
            15=TRANSACTION                 16=PROGRAM
            17=PROCEDURE                   18=MODULE
            19=SUBROUTINE                  20=REUSABLE CODE
            21=COMMAND STREAM
```



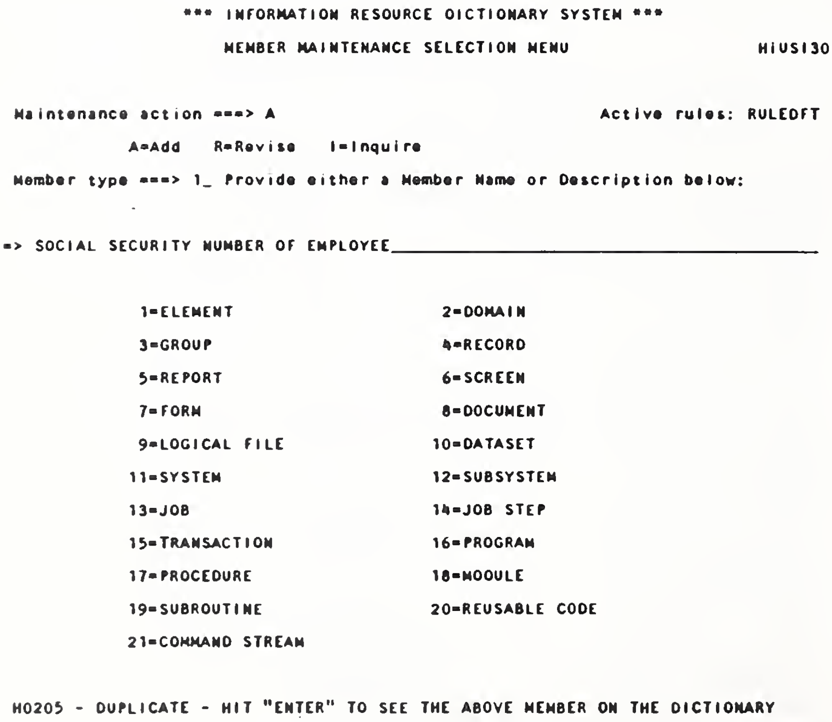
```
              *** INFORMATION RESOURCE DICTIONARY SYSTEM ***
------------------------ CLAUSE SELECTION MENU ---------------------- HIUS132

Doing ADD     for ELEMENT              named EMPL-SOC-SEC-NUM
Whose Business name Is:
    EMPLOYEE SOCIAL SECURITY NUMBER
Any non-blank ===> S  PHYSICAL CHARACTERISTICS   Picking Table ===>
                   S  NAME AT SOURCE
                      LEGAL VALUES
                      VALIDATION CRITERIA
                   S  SHORT DEFINITION
                      ADMINISTRATIVE TEXT
                      RESPONSIBLE ORGANIZATION
                   S  ALIAS NAMES
                      KEY WORDS/CATALOG
                      TECHNICAL NOTES
                      SEE/REFERENCES
```



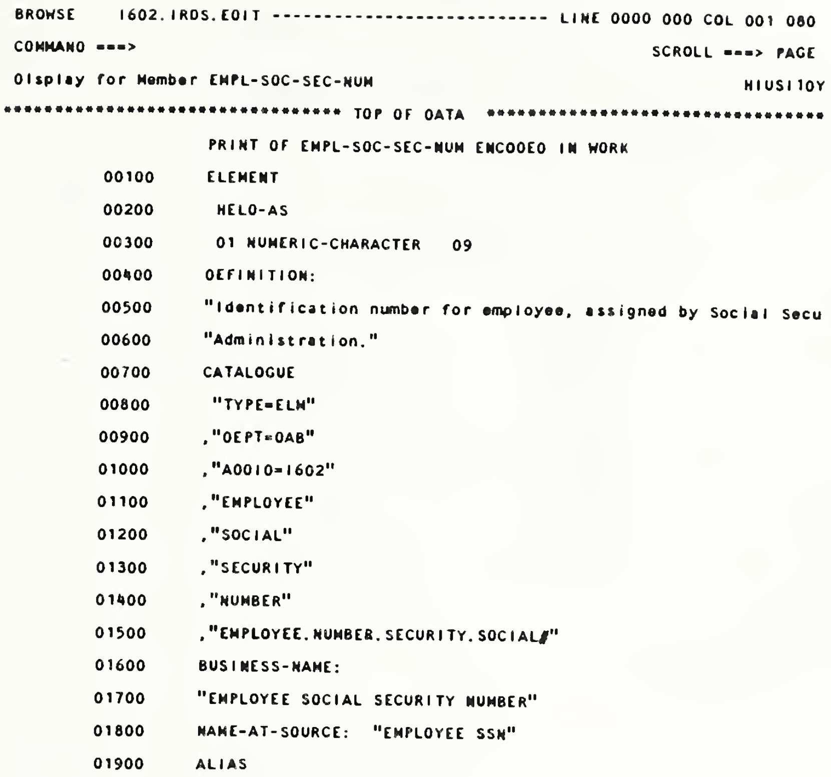
```
BROWSE    1602.IRDS.EDIT -------------------------- LINE 0000 000 COL 001 080
COMMAND ===>                                                 SCROLL ===> PAGE
                                                                     HIUS110B
Browse of the last dictionary response:
******************************** TOP OF DATA ********************************
DM012211          EMPL-SOC-SEC-NUM EXISTS IN STATUS WORK AS A SOURCE ONLY
DM011321          EMPL-SOC-SEC-NUM SUCCESSFULLY REPLACED
DM012961          ENCODING OF EMPL-SOC-SEC-NUM
         00100    ELEMENT
         00200     HELD-AS
         00300     01 NUMERIC-CHARACTER   09
         00400    DEFINITION:
         00500    "Identification number for employee, assigned by Social Secu
         00600    "Administration."
         00700    CATALOGUE
         00800     "TYPE=ELM"
         00900    ,"DEPT=DAB"
         01000    ,"A0010=1602"
         01100    ,"EMPLOYEE"
         01200    ,"SOCIAL"
         01300    ,"SECURITY"
         01400    ,"NUMBER"
         01500    ,"EMPLOYEE.NUMBER.SECURITY.SOCIAL/"
         01600    BUSINESS-NAME:
         01700    "EMPLOYEE SOCIAL SECURITY NUMBER"
```



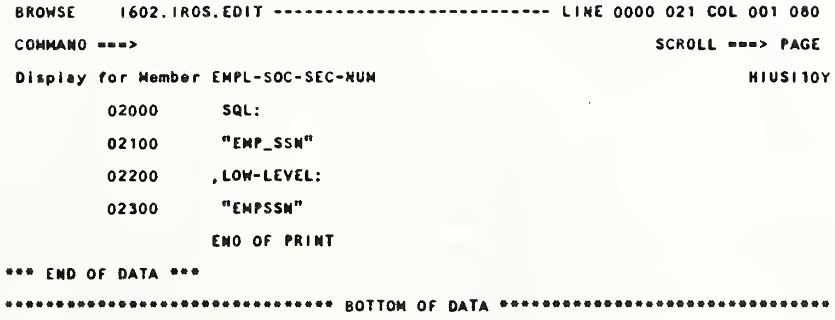
```
BROWSE    1602.IRDS.EDIT --------------------------- LINE 0000 020 COL 001 080
COMMANO ===>                                                 SCROLL ===> PAGE
                                                                     HIUSI10B
Browse of the last dictionary response:
         01800     NAME-AT-SOURCE:  "EMPLOYEE SSN"
         01900     ALIAS
         02000      SQL:
         02100      "EMP_SSN"
         02200     ,LOW-LEVEL:
         02300      "EMPSSN"
DM012801           EMPL-SOC-SEC-NUM SUCCESSFULLY ENCODED
*** ENO OF OATA ***
******************************** BOTTOM OF OATA ********************************
```



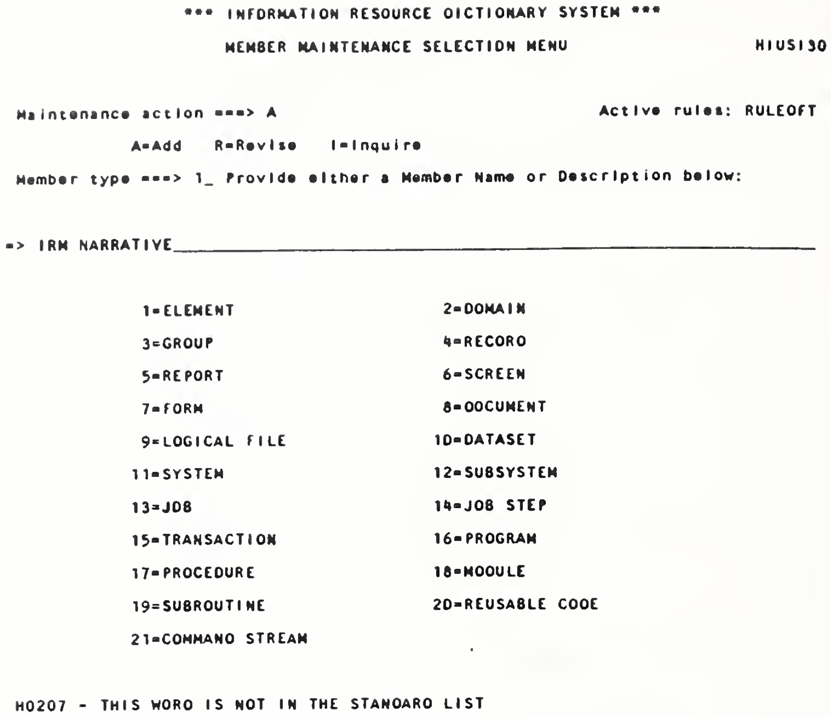
```
              *** INFORMATION RESOURCE OICTIONARY SYSTEM ***
                  MEMBER MAINTENANCE SELECTION MENU                HIUSI30

Maintenance action ===> A                           Active rules: RULEDFT
          A=Add   R=Revise   I=Inquire
Member type ===> 1_ Provide either a Member Name or Description below:

=> SOCIAL SECURITY NUMBER OF EMPLOYEE________________________________________

           1=ELEMENT                    2=DOMAIN
           3=GROUP                      4=RECORD
           5=REPORT                     6=SCREEN
           7=FORM                       8=DOCUMENT
           9=LOGICAL FILE              10=DATASET
          11=SYSTEM                    12=SUBSYSTEM
          13=JOB                       14=JOB STEP
          15=TRANSACTION               16=PROGRAM
          17=PROCEDURE                 18=MOOULE
          19=SUBROUTINE                20=REUSABLE CODE
          21=COMMAND STREAM

H0205 - DUPLICATE - HIT "ENTER" TO SEE THE ABOVE MEMBER ON THE DICTIONARY
```



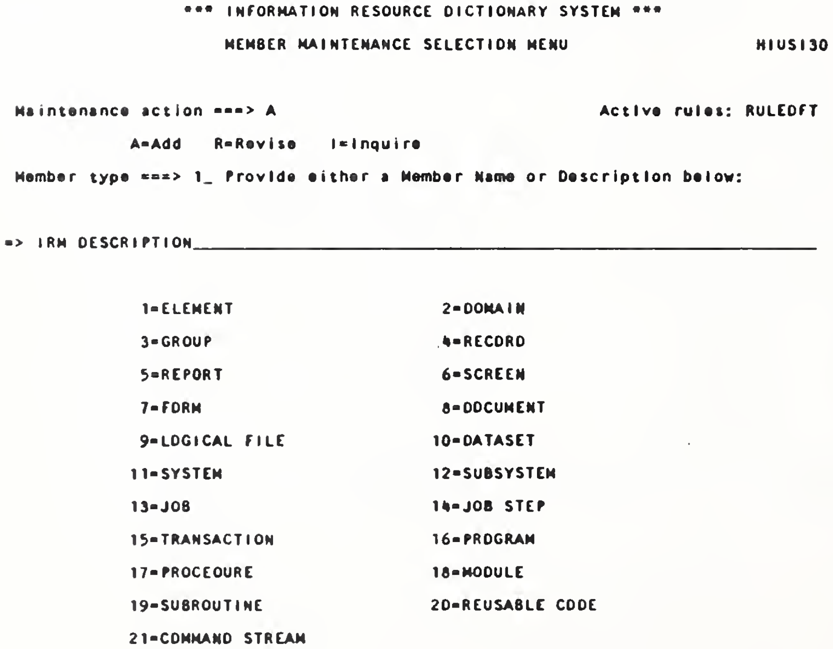
```
BROWSE    1602.IRDS.EOIT -------------------------- LINE 0000 000 COL 001 080
COMMANO ===>                                                SCROLL ===> PAGE
Olsplay for Member EMPL-SOC-SEC-NUM                                  HIUSI10Y
******************************* TOP OF OATA *******************************
                PRINT OF EMPL-SOC-SEC-NUM ENCOOEO IN WORK
        00100   ELEMENT
        00200    HELO-AS
        00300    01 NUMERIC-CHARACTER   09
        00400   OEFINITION:
        00500   "Identification number for employee, assigned by Social Secu
        00600   "Administration."
        00700   CATALOGUE
        00800    "TYPE=ELM"
        00900   ,"OEPT=OAB"
        01000   ,"A0010=1602"
        01100   ,"EMPLOYEE"
        01200   ,"SOCIAL"
        01300   ,"SECURITY"
        01400   ,"NUMBER"
        01500   ,"EMPLOYEE.NUMBER.SECURITY.SOCIAL#"
        01600   BUSINESS-NAME:
        01700   "EMPLOYEE SOCIAL SECURITY NUMBER"
        01800   NAME-AT-SOURCE:  "EMPLOYEE SSN"
        01900   ALIAS
```



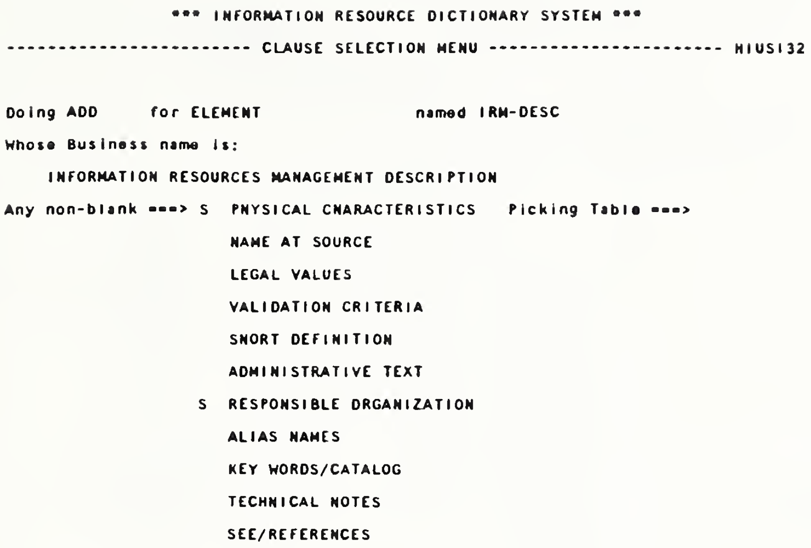
```
BROWSE    1602.IROS.EDIT -------------------------- LINE 0000 021 COL 001 080
COMMANO ===>                                                SCROLL ===> PAGE
Display for Member EMPL-SOC-SEC-NUM                                  HIUSI10Y
        02000     SQL:
        02100      "EMP_SSN"
        02200    ,LOW-LEVEL:
        02300      "EMPSSN"
                 ENO OF PRINT
*** END OF DATA ***
****************************** BOTTOM OF DATA ******************************
```



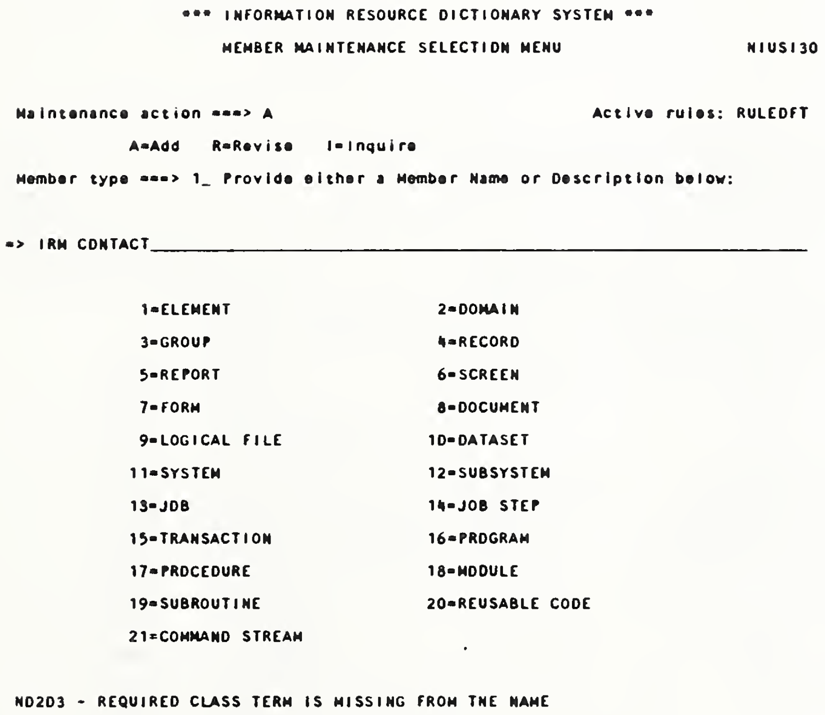
```
            *** INFORMATION RESOURCE DICTIONARY SYSTEM ***
                MEMBER MAINTENANCE SELECTION MENU                HIUSI30

Maintenance action ===> A                          Active rules: RULEDFT
           A=Add    R=Revise    I=Inquire
Member type ===> 1_ Provide either a Member Name or Description below:

=> IRM NARRATIVE___________________________________________________________

            1=ELEMENT                       2=DOMAIN
            3=GROUP                         4=RECORD
            5=REPORT                        6=SCREEN
            7=FORM                          8=DOCUMENT
            9=LOGICAL FILE                 10=DATASET
           11=SYSTEM                       12=SUBSYSTEM
           13=JOB                          14=JOB STEP
           15=TRANSACTION                  16=PROGRAM
           17=PROCEDURE                    18=MODULE
           19=SUBROUTINE                   20=REUSABLE CODE
           21=COMMAND STREAM

H0207 - THIS WORD IS NOT IN THE STANDARD LIST
```



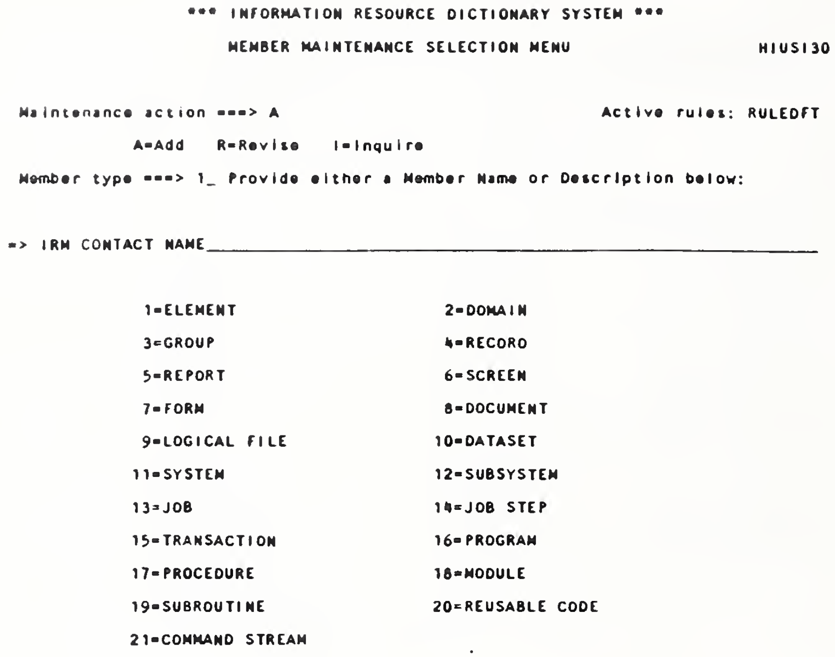
```
            *** INFORMATION RESOURCE DICTIONARY SYSTEM ***
                MEMBER MAINTENANCE SELECTION MENU                HIUSI30

Maintenance action ===> A                          Active rules: RULEDFT
           A=Add    R=Revise    I=Inquire
Member type ===> 1_ Provide either a Member Name or Description below:

=> IRM DESCRIPTION_________________________________________________________

            1=ELEMENT                       2=DOMAIN
            3=GROUP                         4=RECORD
            5=REPORT                        6=SCREEN
            7=FORM                          8=DOCUMENT
            9=LOGICAL FILE                 10=DATASET
           11=SYSTEM                       12=SUBSYSTEM
           13=JOB                          14=JOB STEP
           15=TRANSACTION                  16=PROGRAM
           17=PROCEDURE                    18=MODULE
           19=SUBROUTINE                   20=REUSABLE CODE
           21=COMMAND STREAM
```



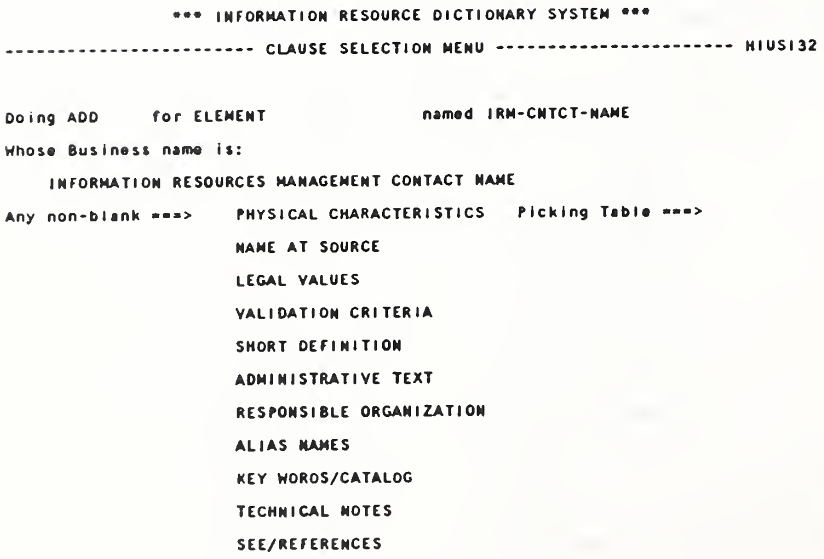
```
                  *** INFORMATION RESOURCE DICTIONARY SYSTEM ***
------------------------ CLAUSE SELECTION MENU ----------------------- NIUSI32

Doing ADD     for ELEMENT               named IRM-DESC
Whose Business name is:
    INFORMATION RESOURCES MANAGEMENT DESCRIPTION
Any non-blank ===> S  PNYSICAL CNARACTERISTICS   Picking Table ===>
                      NAME AT SOURCE
                      LEGAL VALUES
                      VALIDATION CRITERIA
                      SNORT DEFINITION
                      ADMINISTRATIVE TEXT
                   S  RESPONSIBLE DRGANIZATION
                      ALIAS NAMES
                      KEY WORDS/CATALOG
                      TECHNICAL NOTES
                      SEE/REFERENCES
```



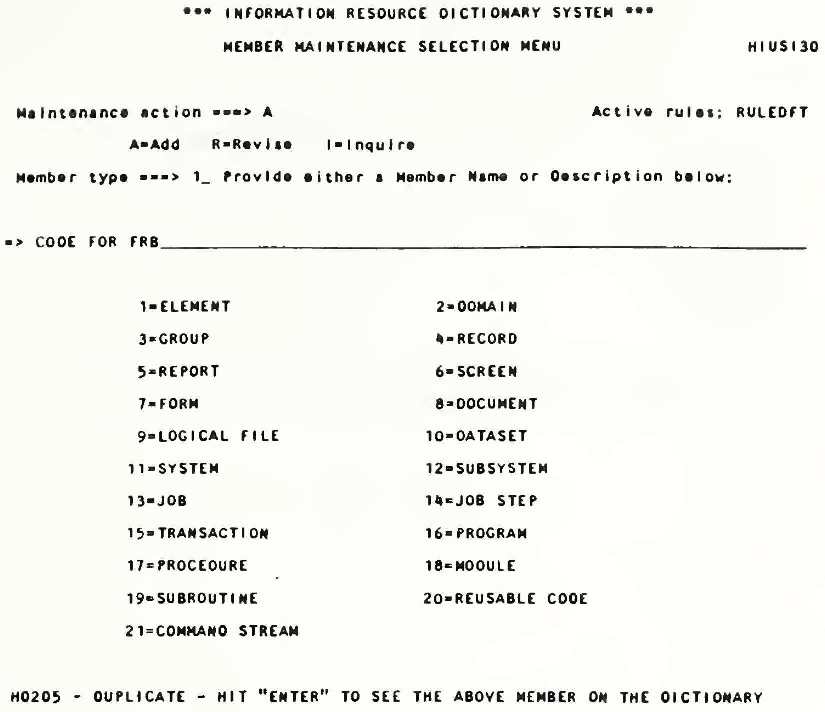
```
                 *** INFORMATION RESOURCE DICTIONARY SYSTEM ***
                     MEMBER MAINTENANCE SELECTIDN MENU                NIUSI30

Maintenance action ===> A                          Active rules: RULEDFT
           A=Add    R=Revise    I=Inquire
Member type ===> 1_ Provide either a Member Name or Description below:

=> IRM CDNTACT____________________________________________________________

            1=ELEMENT                        2=DOMAIN
            3=GROUP                          4=RECORD
            5=REPORT                         6=SCREEN
            7=FORM                           8=DOCUMENT
            9=LOGICAL FILE                  1D=DATASET
           11=SYSTEM                        12=SUBSYSTEM
           13=JDB                           14=JOB STEP
           15=TRANSACTION                   16=PRDGRAM
           17=PRDCEDURE                     18=MDDULE
           19=SUBROUTINE                    20=REUSABLE CODE
           21=COMMAND STREAM

ND2D3 - REQUIRED CLASS TERM IS MISSING FROM TNE NAME
```



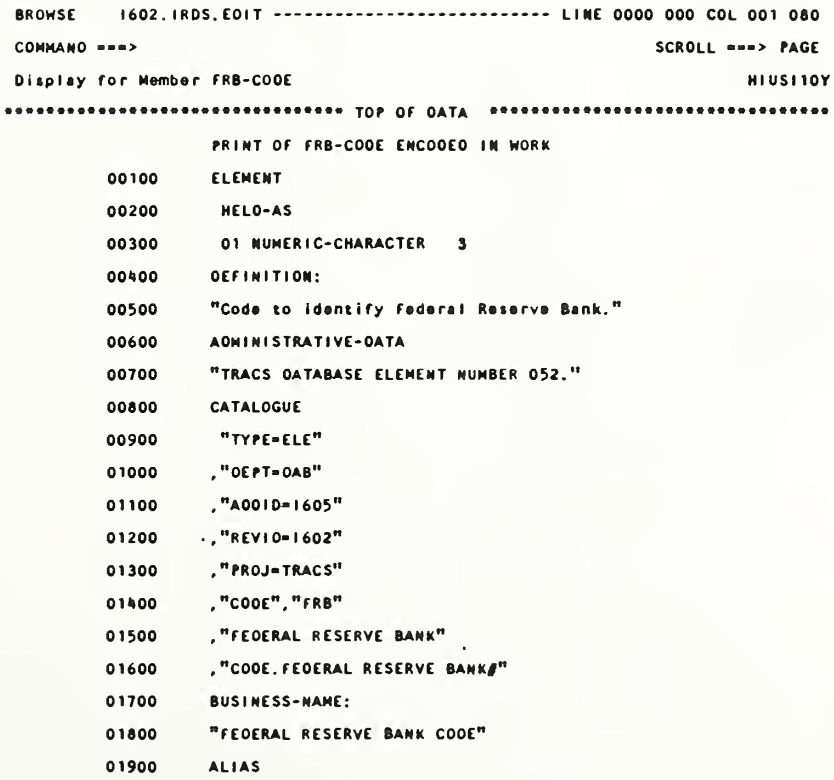
```
                 *** INFORMATION RESOURCE DICTIONARY SYSTEM ***
                     MEMBER MAINTENANCE SELECTION MENU              HIUSI30

   Maintenance action ===> A                             Active rules: RULEDFT
              A=Add    R=Revise    I=Inquire
   Member type ===> 1_ Provide either a Member Name or Description below:

  => IRM CONTACT NAME____________________________________________________________

                1=ELEMENT                     2=DOMAIN
                3=GROUP                       4=RECORO
                5=REPORT                      6=SCREEN
                7=FORM                        8=DOCUMENT
                9=LOGICAL FILE               10=DATASET
               11=SYSTEM                     12=SUBSYSTEM
               13=JOB                        14=JOB STEP
               15=TRANSACTION                16=PROGRAM
               17=PROCEDURE                  18=MODULE
               19=SUBROUTINE                 20=REUSABLE CODE
               21=COMMAND STREAM
```



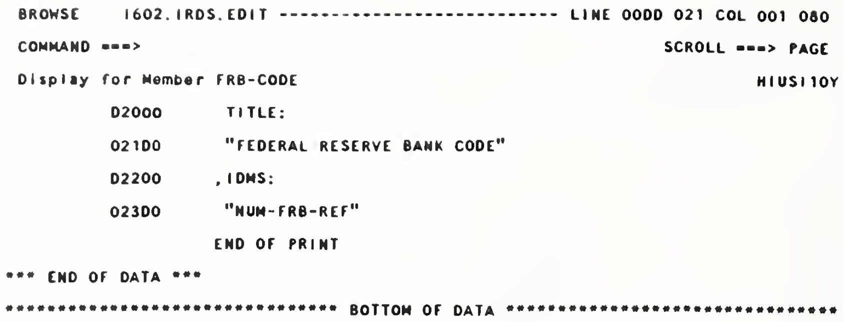
```
               *** INFORMATION RESOURCE DICTIONARY SYSTEM ***
------------------------ CLAUSE SELECTION MENU ----------------------- HIUSI32

Doing ADD      for ELEMENT              named IRM-CNTCT-NAME
Whose Business name is:
    INFORMATION RESOURCES MANAGEMENT CONTACT NAME
Any non-blank ===>      PHYSICAL CHARACTERISTICS    Picking Table ===>
                        NAME AT SOURCE
                        LEGAL VALUES
                        VALIDATION CRITERIA
                        SHORT DEFINITION
                        ADMINISTRATIVE TEXT
                        RESPONSIBLE ORGANIZATION
                        ALIAS NAMES
                        KEY WOROS/CATALOG
                        TECHNICAL NOTES
                        SEE/REFERENCES
```



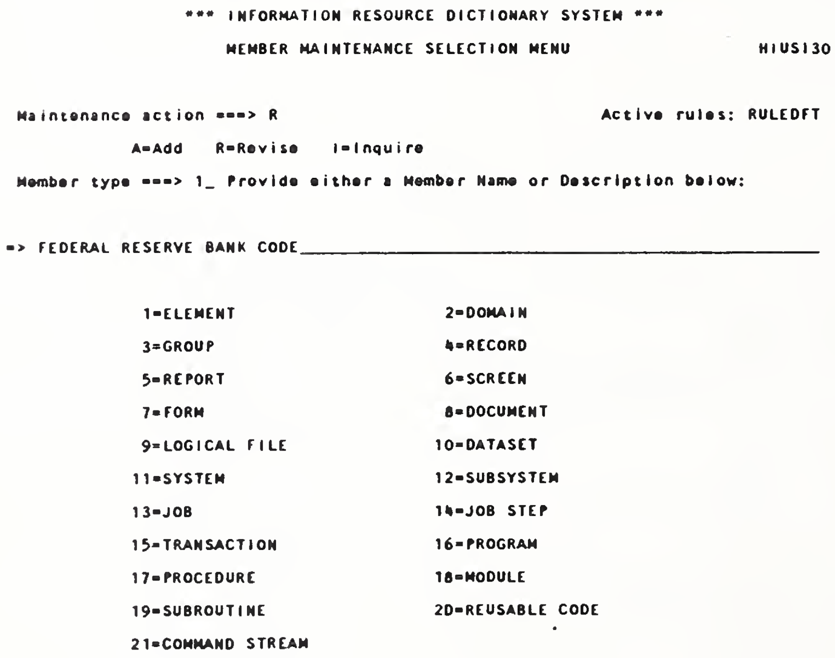
```
                 *** INFORMATION RESOURCE OICTIONARY SYSTEM ***
                     MEMBER MAINTENANCE SELECTION MENU                  HIUSI30

 Maintenance action ===> A                                  Active rules: RULEDFT
            A=Add   R=Revise   I=Inquire
 Member type ===> 1_ Provide either a Member Name or Oescription below:

=> COOE FOR FRB________________________________________________________________

              1=ELEMENT                          2=OOMAIN
              3=GROUP                            4=RECORD
              5=REPORT                           6=SCREEN
              7=FORM                             8=DOCUMENT
              9=LOGICAL FILE                    10=OATASET
             11=SYSTEM                          12=SUBSYSTEM
             13=JOB                             14=JOB STEP
             15=TRANSACTION                     16=PROGRAM
             17=PROCEOURE                       18=MOOULE
             19=SUBROUTINE                      20=REUSABLE COOE
             21=COMMANO STREAM

 H0205 - OUPLICATE - HIT "ENTER" TO SEE THE ABOVE MEMBER ON THE OICTIONARY
```



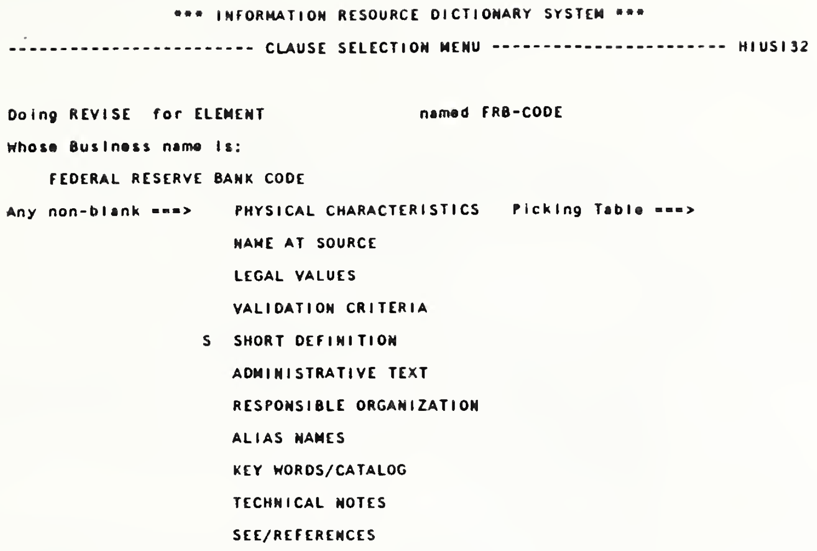
```
 BROWSE    1602.IRDS.EOIT --------------------------- LINE 0000 000 COL 001 080
 COMMANO ===>                                                  SCROLL ===> PAGE
 Display for Member FRB-COOE                                          HIUSI10Y
******************************** TOP OF OATA ********************************
                  PRINT OF FRB-COOE ENCOOEO IN WORK
          00100   ELEMENT
          00200    HELO-AS
          00300    01 NUMERIC-CHARACTER   3
          00400   OEFINITION:
          00500   "Code to identify Federal Reserve Bank."
          00600   AOMINISTRATIVE-OATA
          00700   "TRACS OATABASE ELEMENT NUMBER 052."
          00800   CATALOGUE
          00900    "TYPE=ELE"
          01000   ,"OEPT=OAB"
          01100   ,"AOOID=1605"
          01200   ,"REVIO=1602"
          01300   ,"PROJ=TRACS"
          01400   ,"COOE","FRB"
          01500   ,"FEOERAL RESERVE BANK"
          01600   ,"COOE.FEOERAL RESERVE BANK#"
          01700   BUSINESS-NAME:
          01800   "FEOERAL RESERVE BANK COOE"
          01900   ALIAS
```



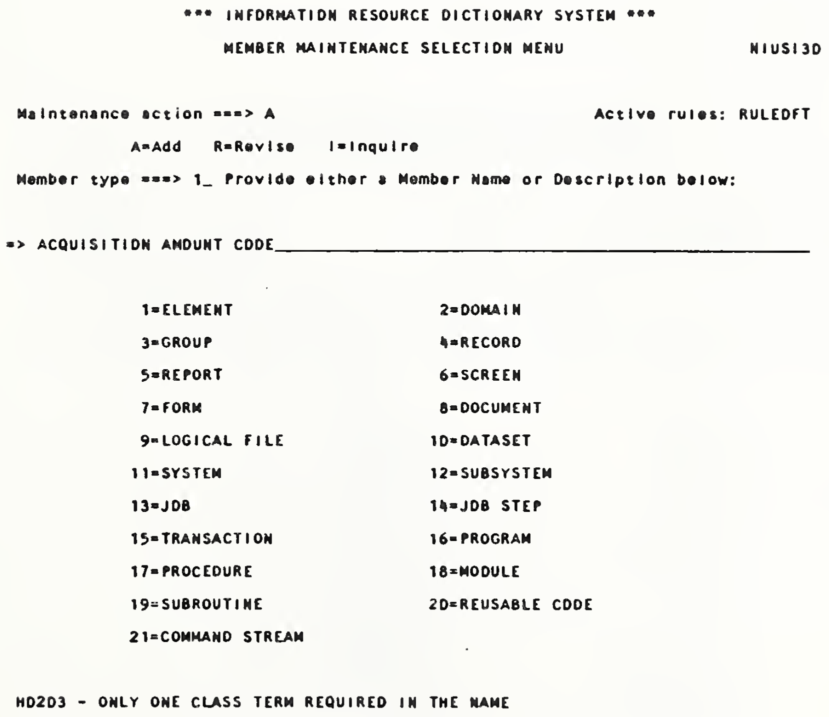
```
BROWSE    1602.IRDS.EDIT --------------------------- LINE 000D 021 COL 001 080
COMMAND ===>                                                 SCROLL ===> PAGE
Display for Member FRB-CODE                                          HIUSI10Y
        D2000      TITLE:
        021D0      "FEDERAL RESERVE BANK CODE"
        D2200     ,IDMS:
        023D0      "NUM-FRB-REF"
                  END OF PRINT
*** END OF DATA ***
******************************** BOTTOM OF DATA ********************************
```



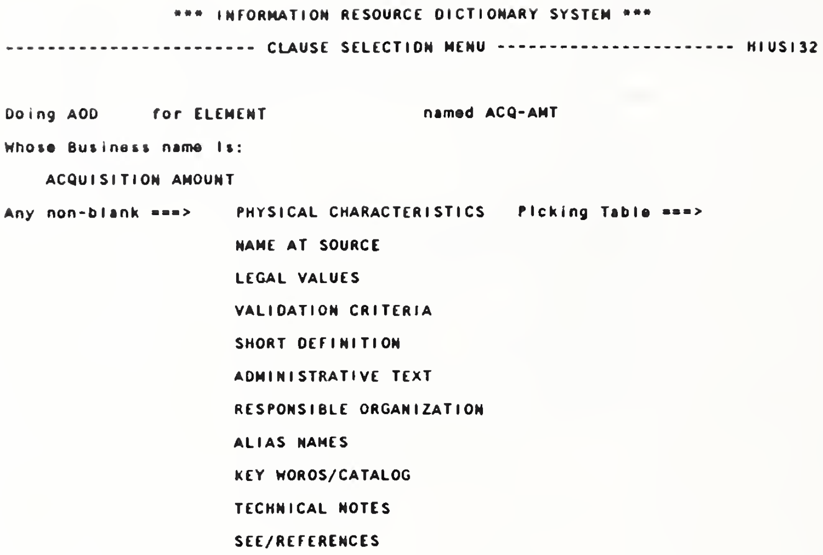
```
                 *** INFORMATION RESOURCE DICTIONARY SYSTEM ***
                    MEMBER MAINTENANCE SELECTION MENU                HIUS130

 Maintenance action ===> R                            Active rules: RULEDFT
           A=Add    R=Revise    I=Inquire
 Member type ===> 1_ Provide either a Member Name or Description below:

=> FEDERAL RESERVE BANK CODE_____________________________________________

            1=ELEMENT                     2=DOMAIN
            3=GROUP                       4=RECORD
            5=REPORT                      6=SCREEN
            7=FORM                        8=DOCUMENT
            9=LOGICAL FILE               10=DATASET
           11=SYSTEM                     12=SUBSYSTEM
           13=JOB                        14=JOB STEP
           15=TRANSACTION                16=PROGRAM
           17=PROCEDURE                  18=MODULE
           19=SUBROUTINE                 2D=REUSABLE CODE
           21=COMMAND STREAM
```



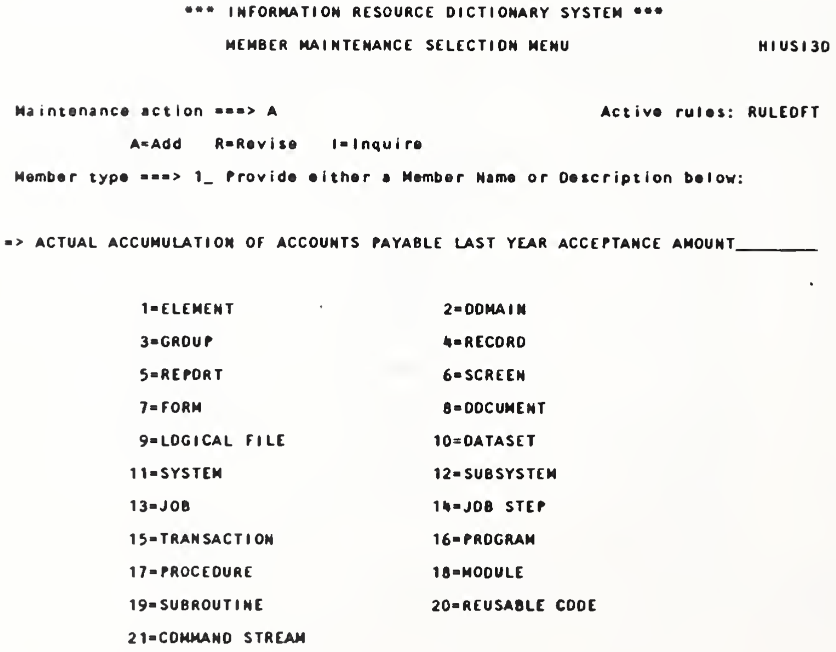
```
                 *** INFORMATION RESOURCE DICTIONARY SYSTEM ***
------------------------ CLAUSE SELECTION MENU ----------------------- HIUSI32

Doing REVISE  for ELEMENT              named FRB-CODE
Whose Business name is:
    FEDERAL RESERVE BANK CODE
Any non-blank ===>    PHYSICAL CHARACTERISTICS   Picking Table ===>
                      NAME AT SOURCE
                      LEGAL VALUES
                      VALIDATION CRITERIA
                   S  SHORT DEFINITION
                      ADMINISTRATIVE TEXT
                      RESPONSIBLE ORGANIZATION
                      ALIAS NAMES
                      KEY WORDS/CATALOG
                      TECHNICAL NOTES
                      SEE/REFERENCES
```



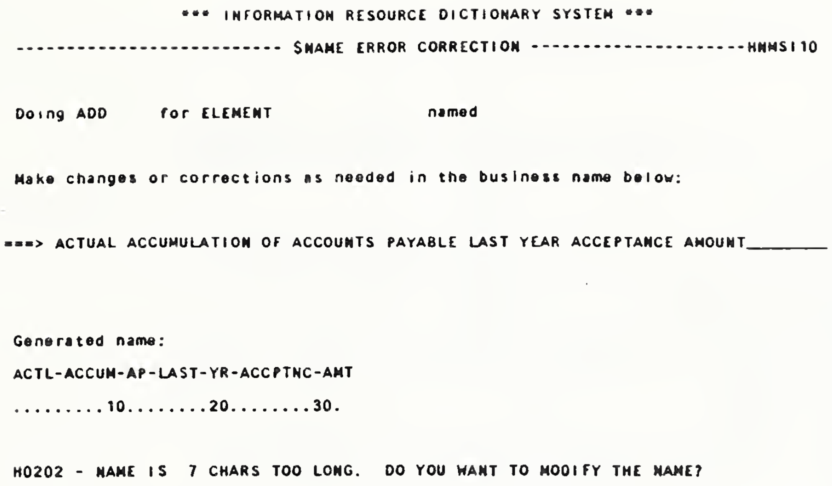
```
                 *** INFDRMATIDN RESOURCE DICTIONARY SYSTEM ***
                     MEMBER MAINTENANCE SELECTIDN MENU                 HIUSI3D

 Maintenance action ===> A                               Active rules: RULEDFT
           A=Add    R=Revise    I=Inquire
 Member type ===> 1_ Provide either a Member Name or Description below:

=> ACQUISITIDN AMDUNT CDDE____________________________________________________

             1=ELEMENT                    2=DOMAIN
             3=GROUP                      4=RECORD
             5=REPORT                     6=SCREEN
             7=FORM                       8=DOCUMENT
             9=LOGICAL FILE              1D=DATASET
            11=SYSTEM                    12=SUBSYSTEM
            13=JDB                       14=JDB STEP
            15=TRANSACTION               16=PROGRAM
            17=PROCEDURE                 18=MODULE
            19=SUBROUTINE                2D=REUSABLE CDDE
            21=COMMAND STREAM

 HD2D3 - ONLY ONE CLASS TERM REQUIRED IN THE NAME
```



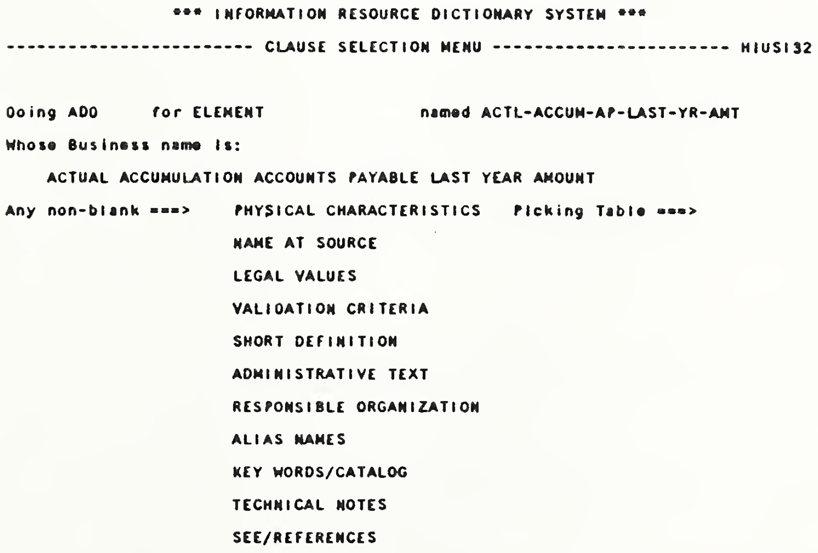
```
                   *** INFORMATION RESOURCE DICTIONARY SYSTEM ***
------------------------ CLAUSE SELECTION MENU ----------------------- HIUSI32

Doing ADD     for ELEMENT                named ACQ-AMT
Whose Business name Is:
    ACQUISITION AMOUNT
Any non-blank ===>     PHYSICAL CHARACTERISTICS    Picking Table ===>
                       NAME AT SOURCE
                       LEGAL VALUES
                       VALIDATION CRITERIA
                       SHORT DEFINITION
                       ADMINISTRATIVE TEXT
                       RESPONSIBLE ORGANIZATION
                       ALIAS NAMES
                       KEY WORDS/CATALOG
                       TECHNICAL NOTES
                       SEE/REFERENCES
```



```
                   *** INFORMATION RESOURCE DICTIONARY SYSTEM ***
                       MEMBER MAINTENANCE SELECTION MENU                 HIUSI30

 Maintenance action ===> A                               Active rules: RULEDFT
            A=Add    R=Revise    I=Inquire
 Member type ===> 1_ Provide either a Member Name or Description below:

=> ACTUAL ACCUMULATION OF ACCOUNTS PAYABLE LAST YEAR ACCEPTANCE AMOUNT________

              1=ELEMENT                     2=DOMAIN
              3=GROUP                       4=RECORD
              5=REPORT                      6=SCREEN
              7=FORM                        8=DOCUMENT
              9=LOGICAL FILE               10=DATASET
             11=SYSTEM                     12=SUBSYSTEM
             13=JOB                        14=JOB STEP
             15=TRANSACTION                16=PROGRAM
             17=PROCEDURE                  18=MODULE
             19=SUBROUTINE                 20=REUSABLE CODE
             21=COMMAND STREAM
```

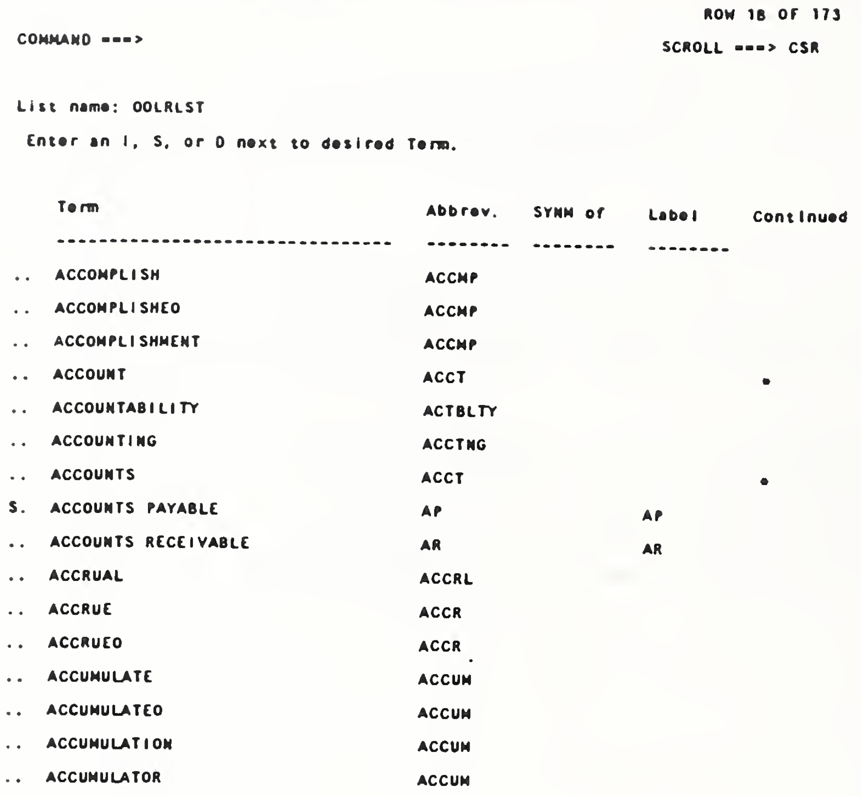


```
                *** INFORMATION RESOURCE DICTIONARY SYSTEM ***
-------------------------- $NAME ERROR CORRECTION ---------------------HNMSI10

 Doing ADD     for ELEMENT              named

 Make changes or corrections as needed in the business name below:

===> ACTUAL ACCUMULATION OF ACCOUNTS PAYABLE LAST YEAR ACCEPTANCE AMOUNT________

 Generated name:
 ACTL-ACCUM-AP-LAST-YR-ACCPTNC-AMT
 .........10........20........30.

 H0202 - NAME IS  7 CHARS TOO LONG.  DO YOU WANT TO MODIFY THE NAME?
```

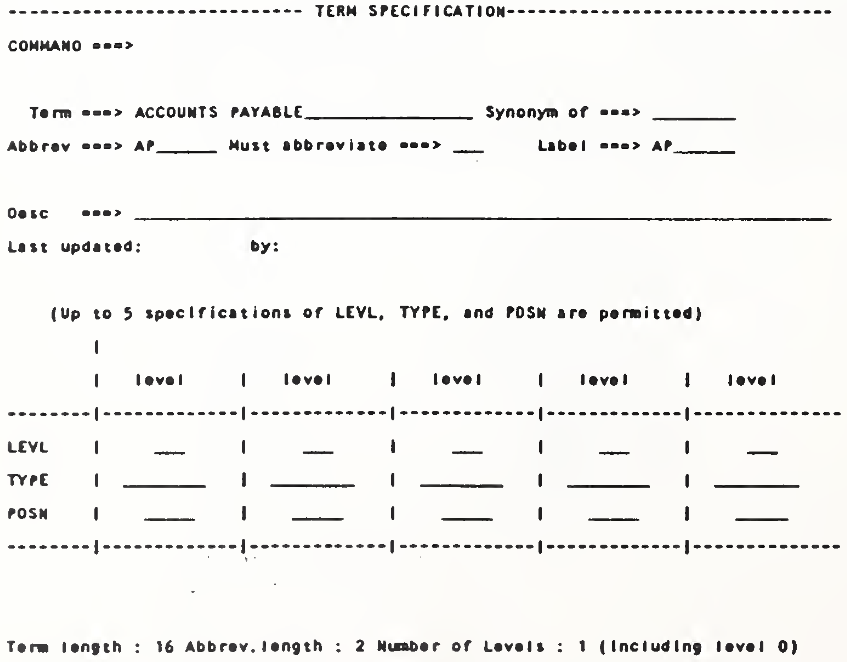


```
                *** INFORMATION RESOURCE DICTIONARY SYSTEM ***
------------------------ CLAUSE SELECTION MENU ----------------------- HIUSI32

 Doing ADD     for ELEMENT              named ACTL-ACCUM-AP-LAST-YR-AMT
 Whose Business name is:
     ACTUAL ACCUMULATION ACCOUNTS PAYABLE LAST YEAR AMOUNT
 Any non-blank ===>     PHYSICAL CHARACTERISTICS   Picking Table ===>
                        NAME AT SOURCE
                        LEGAL VALUES
                        VALIDATION CRITERIA
                        SHORT DEFINITION
                        ADMINISTRATIVE TEXT
                        RESPONSIBLE ORGANIZATION
                        ALIAS NAMES
                        KEY WORDS/CATALOG
                        TECHNICAL NOTES
                        SEE/REFERENCES
```

(PARTIAL DISPLAY OF STANDARD ABBREVIATION LIST)



```
                                                                 ROW 18 OF 173
 COMMAND ===>                                                   SCROLL ===> CSR

 List name: OOLRLST
  Enter an I, S, or D next to desired Term.

     Term                                Abbrev.   SYNM of    Label     Continued
     --------------------------------    --------  --------   --------
 ..  ACCOMPLISH                          ACCMP
 ..  ACCOMPLISHEO                        ACCMP
 ..  ACCOMPLISHMENT                      ACCMP
 ..  ACCOUNT                             ACCT                                •
 ..  ACCOUNTABILITY                      ACTBLTY
 ..  ACCOUNTING                          ACCTNG
 ..  ACCOUNTS                            ACCT                                •
 S.  ACCOUNTS PAYABLE                    AP                   AP
 ..  ACCOUNTS RECEIVABLE                 AR                   AR
 ..  ACCRUAL                             ACCRL
 ..  ACCRUE                              ACCR
 ..  ACCRUEO                             ACCR
 ..  ACCUMULATE                          ACCUM
 ..  ACCUMULATEO                         ACCUM
 ..  ACCUMULATION                        ACCUM
 ..  ACCUMULATOR                         ACCUM
```



```
 --------------------------- TERM SPECIFICATION------------------------------
 COMMANO ===>

   Term ===> ACCOUNTS PAYABLE________________ Synonym of ===> ________
 Abbrev ===> AP______ Must abbreviate ===> ___     Label ===> AP______

 Oesc   ===> ____________________________________________________________
 Last updated:           by:

     (Up to 5 specifications of LEVL, TYPE, and POSN are permitted)
          |
          |  level      |  level      |  level      |  level      |  level
 --------|-------------|-------------|-------------|-------------|--------------
 LEVL     |    ___      |    ___      |    ___      |    ___      |    ___
 TYPE     |  _______    |  _______    |  _______    |  _______    |  _______
 POSN     |   _____     |   _____     |   _____     |   _____     |   _____
 --------|-------------|-------------|-------------|-------------|--------------


 Term length : 16 Abbrev.length : 2 Number of Levels : 1 (Including level 0)
```

USER REACTION

- DIFFERENT USERS WANT DIFFERENT ABBREVIATIONS FOR THE SAME TERM

- CHANGE IN CONCATENATION OF TERMS
  Users request new acronyms or new set of terms for concatenation which effect existing names.
  i.e.
  Before: YEAR TO DATE BUDGET AMOUNT = YR-TO-DTE-BUDGT-AMT
  After: YEAR TO DATE BUDGET AMOUNT = YTD-BUDGT-AMT

- DON'T LIKE NAMES WITH CONNECTORS
  Too computer like, or doesn't sound right the way the terms are arranged.

- DON'T LIKE THE NAME THAT WAS GENERATED
  The user's name or business name should be the dictionary name rather than the abbreviated name.

Naming Conventions Presentation

Ms. Barbara Nichols

Digital Equipment Corporation

Data Description and Naming

Standards, Procedures, and Guidelines

Barbara Nichols
Digital Equipment Corporation

Presentation to the Naming Conventions Forum
National Institute of Standards and Technology
November 16-17, 1989

Agenda

- Description of Digital's Naming Conventions
- Tools Used
- Enforcement
- User Reaction
- Degree of Management Support
- Current Issues

Description of Digital's Naming Conventions

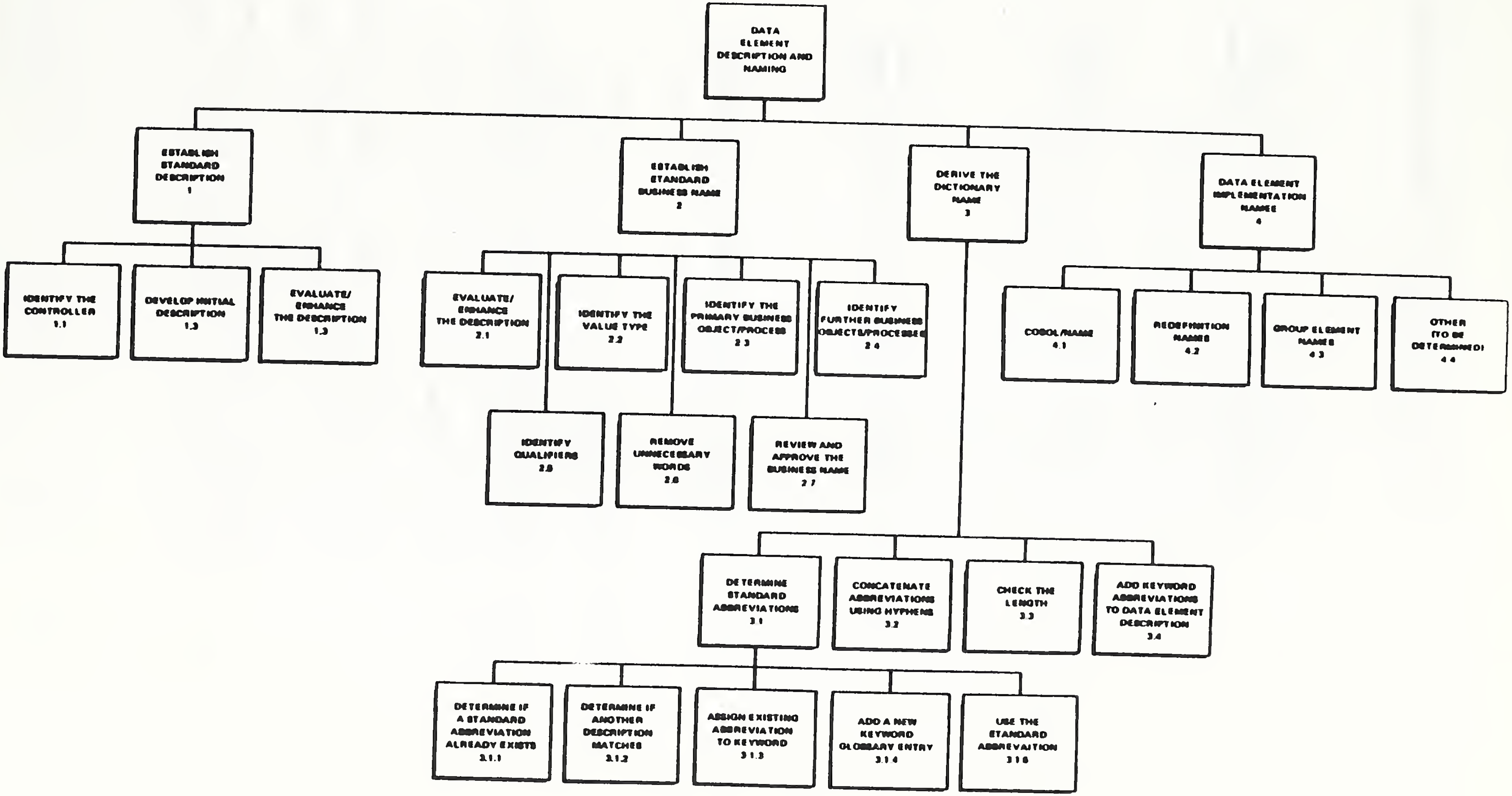
1. Establish a Standard Description

2. Establish a Standard Business Name

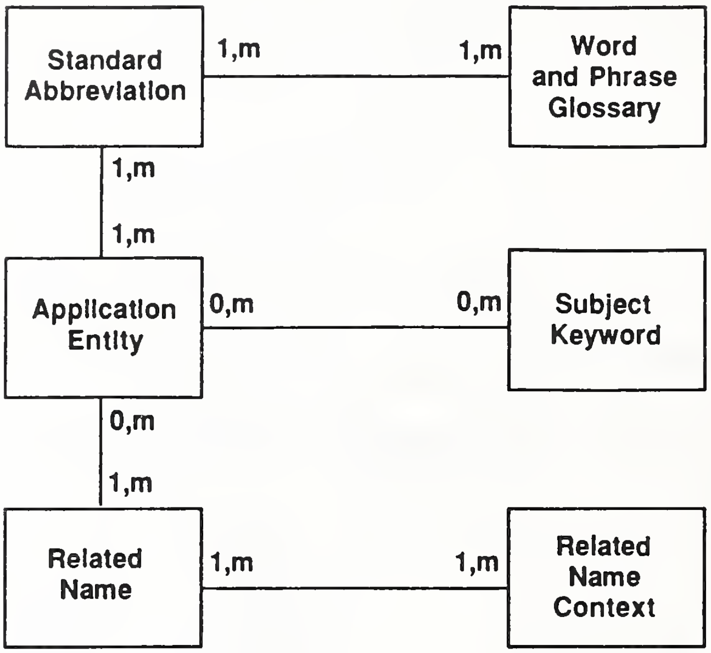
3. Derive the Dictionary Name (Std Abbreviated Name)

4. Establish Data Element Implementation Names

FIGURE 1
DIGITAL EQUIPMENT CORPORATION
DATA ELEMENT DESCRIPTION AND NAMING PROCEDURE


DATA ELEMENT DESCRIPTION AND NAMING
ESTABLISH STANDARD DESCRIPTION 1
IDENTIFY THE CONTROLLER 1.1
DEVELOP INITIAL DESCRIPTION 1.2
EVALUATE/ ENHANCE THE DESCRIPTION 1.3
ESTABLISH STANDARD BUSINESS NAME 2
EVALUATE/ ENHANCE THE DESCRIPTION 2.1
IDENTIFY THE VALUE TYPE 2.2
IDENTIFY THE PRIMARY BUSINESS OBJECT/PROCESS 2.3
IDENTIFY FURTHER BUSINESS OBJECTS/PROCESSES 2.4
IDENTIFY QUALIFIERS 2.5
REMOVE UNNECESSARY WORDS 2.6
REVIEW AND APPROVE THE BUSINESS NAME 2.7
DERIVE THE DICTIONARY NAME 3
DETERMINE STANDARD ABBREVIATIONS 3.1
DETERMINE IF A STANDARD ABBREVIATION ALREADY EXISTS 3.1.1
DETERMINE IF ANOTHER DESCRIPTION MATCHES 3.1.2
ASSIGN EXISTING ABBREVIATION TO KEYWORD 3.1.3
ADD A NEW KEYWORD GLOSSARY ENTRY 3.1.4
USE THE STANDARD ABBREVAITION 3.1.5
CONCATENATE ABBREVIATIONS USING HYPHENS 3.2
CHECK THE LENGTH 3.3
ADD KEYWORD ABBREVIATIONS TO DATA ELEMENT DESCRIPTION 3.4
DATA ELEMENT IMPLEMENTATION NAMES 4
COBOL NAME 4.1
REDEFINITION NAMES 4.2
GROUP ELEMENT NAMES 4.3
OTHER (TO BE DETERMINED) 4.4

Digital's Standard Naming Model



Some Model/Process Highlights

- Names derived from Business Description
- Synonym/Homonym resolution through Word and Phrase Glossary: Meanings associated with abbreviation, not full word or phrase
- Related Names must be unique within Context
- Related Names pertain only to a subset of entity-types
- Phrases may be used in naming and abbreviated
- Redundancy checking done on "name components"
- Subject Class not a mandatory component of Object Name

Tools Used

- Word and Phrase Glossary Maintenance:

  BASIS - Text Retrieval Tool.
  Searched words and meanings of words.
  Purpose: synonym and homonym control.

- Application Entity Redundancy:

  Query against name components in Dictionary Database.
  Uncover opportunities for re-use.

Enforcement / Management Support

- Good acceptance throughout the Data Administration function
- Introduced at Digital in December 1984
- Incorporated into the Data Standardization Process
- Data Description and Naming Workshop
- Enforcement breaks down at Implementation Names

User Reaction

- Names are "too long"
- Fear loss of use of familiar terms
- Standard sometimes mis-applied:

  "Customer Order Financial Institution Credit Authorization Identifier"

  "US Government Person Multipurpose Unique Identifier"

- Realization of the benefit of preciseness
- Salvation when Element Names developed outside of model context
- Glossary maintenance is burdensome
- Rigorous Glossary facilitates redundancy checking

Current Issues

- Naming of Application Entities in Context
- Need for Subject Class when rigorous naming within context?
- Integration of toolset to include:
  Word and Phrase Glossary Maintenance,
  Redundancy Intelligence
  Naming Standard Enforcement
- Internationalization

Naming Conventions Presentation

Mr. Madhu S. Singh

Bell Communications Research (Bellcore)

Bellcore Data Naming Guidelines

A logical data name is a conceptual (or business) name used by an organization to carry out its intended business. At the logical level, the primary requirements of data naming are clarity of meaning and uniqueness, above all else. In general, logical data names at Bellcore may be described as below.

1. Logical Data Names

<u>Corporate Data Names</u>

A corporate data name is the designated label of a data type recognized consistently across applications. There are two types of corporate data names: (1) Corporate Logical Data Name (CLDN) and (2) Corporate Logical Access Name (CLAN).

Corporate Logical Data Names (Full or Descriptive Names)

A corporate logical data name (CLDN) is the authentic data type name that will reside in the Bellcore IRD. A corporate logical data name provides much semantics of the data and takes precedence whenever there is a conflict in expressing the meaning of a data type used in different applications. (Semantics describes the functional meaning attached to a data type.) Consistent use of the single CLDN to represent the same business data across all applications will improve data sharing.

Corporate Logical Access Names

Corporate logical access names (also at times referred to simply as access names) are data type names that conform to the ANSI X3.138 Standard and will be used most frequently to access data type in the Information Resource Dictionary (IRD) due to their shorter length (easy to enter and remember). A corporate logical access name is not a unique identifier of a data instance in a database, however, it uniquely distinguishes a data type in the IRD. ANSI X3. 138 Information Resource Dictionary System (IRDS) requires that each data type name must have an access name. In many cases, an access name may be a CLDN if the CLDN is 32 characters or less.

Access names also include user names (e.g., panel, screen, and interface names), acronyms, and abbreviated corporate logical data names. An access name is unique and can be assigned to only one entity type, relationship type, or attribute type within the IRD.

Aliases (Abbreviations/Acronyms)

Because of the diversity of sources, a particular data type may be referenced by different names in different sources. However, an alias is unique only within a given context. A "context-alias" combination points to one and only one corporate logical data name. As mentioned earlier, context may be the name of a system, service, interface, or application.

Standard acronyms and abbreviated names are also considered to be aliases of CLDNs. A list of the standard abbreviations/acronyms will be stored in the Bellcore IRD.

2. Data Name Construction

Corporate Logical Data Name Construction

Corporate Logical Data Names include entity, relationship, and attribute type names and their construction is discussed as below.

Naming Entity Types and Relationship Types

A logical entity type or relationship type name consists of a prime term.

A prime term consists of two components: (1) prime word and (2) modifying word.

Prime words explicitly describe the meaning of concepts (or "things") useful to the enterprise (e.g., employee, project, supplier, circuit, customer, invoice).

Add one or more modifying words to prime word to describe the entity type to make the entity type unique within an application (e.g., purchasing department, funded project, terminal block, shared trunk circuit). In many instances, however, modifying words may not be required to fully construct an entity type name.

Guidelines For Selecting Components

o Ensure that each entity type or relationship type name has a prime word. Assign the prime word that most explicitly describes the meaning of the data type being named.

o Avoid using a class word as a prime word component of a name.

o Do not use words that are articles i.e., a, an, the), pronouns, prepositions, and conjunctions.

o If required, assign modifying words to uniquely describe the data within an application.

- o Do not use intelligence that is subject to change as a component of a data type name (i.e., organization's name).

Guidelines For Sequencing the Components Within a Data Name

- o Place modifying words, starting from left, followed by prime word, respectively.
- o If there are two or more modifying words, arrange them in sequence, from left to right, from most generic to most unique.

Naming Attribute Types

An attribute type name will have the following two components: (1) class term and (2) prime term.

Class Term

A class term consists of: (1) class word and (2) modifying word.

Class words are the key words that most explicitly describe attribute types. A class word describes a basic property of the data element and can be recognized by asking the question: "what" the data is, not "how" it is used. A few examples of class words include name, count, date, and amount.

Modifying words provide additional meaning to class words. For example, in birth date and pay amount, the words birth and pay are modifying words.

Prime Term

A prime term is a name component of an entity type or relationship type which possesses the attribute type being named. A few examples of the prime term are employee, customer, purchase order, and authorization approval. Keeping entity/relationship type name as a part of attribute type name provides a context to an environment in which a class word is being used. The context is important, since it may have profound influence on the meaning of the data.

Guidelines For Selecting Components

- o Select a class word from the class word list that most explicitly describes the meaning of the attribute type being named.

  It is important that an attribute type is assigned to an appropriate class word. This will help querying on the classes in the data dictionary to determine if a data element has been previously named.

o Ensure that each attribute type name has a prime term (entity type or relationship type name). Assign an entity type or relationship type name to which the attribute type being named belongs to.

o Assign one and only one class word to each attribute type name.

o Avoid using class words as prime or modifying words.

o Assign modifying words, if required, to uniquely identify an attribute type name. An attribute type name must be unique within an application.

o Do not use words that are articles (i.e., a, an, the), pronouns, prepositions, or conjunctions.

Guidelines For Sequencing the Components Within a Data Name

o Place prime term (entity type or relationship type name), starting from left, followed by class word, respectively.

o Place the modifying word, if any, between prime term and class word.

o If there are two or more modifying words, arrange them in sequence, from left to right, from most generic to most unique.

<u>Corporate Logical Access Name Construction</u>

Access names are either shortened CLDN or external names and can be managed more efficiently. Access names that pertain to entity types, relationship types, and attribute types have similar construction to that of entity type, relationship type, and attribute type names.

Guidelines For Selecting Components

o Ensure that every access name has a prime term.

o Select class words for attribute type access names from the approved class word list only.

o Avoid using a class word as a part of an entity type or relationship type access name.

o Do not use words that are articles (ie., a, an, the), pronouns, prepositions, and conjunctions.

- o Do not use intelligence that is subject to change as a component of data type name (i.e., organization's name).

Guidelines For Sequencing the Components Within a Data Name

- o Place first prime term, starting from left, followed by class word.

- o Place modifying words between prime term and class word.

- o If there are two or more modifying words, arrange them in sequence, from left to right, from most generic to most unique.

Constructing Aliases

An alias mapped to any corporate logical data name (CLDN) in the IRD must be associated with its context. A context name should be separated from a data name using a period and words should be separated using underscores, respectively.

DATA NAMING CONVENTIONS FORUM

National Institute of Standards and Technology

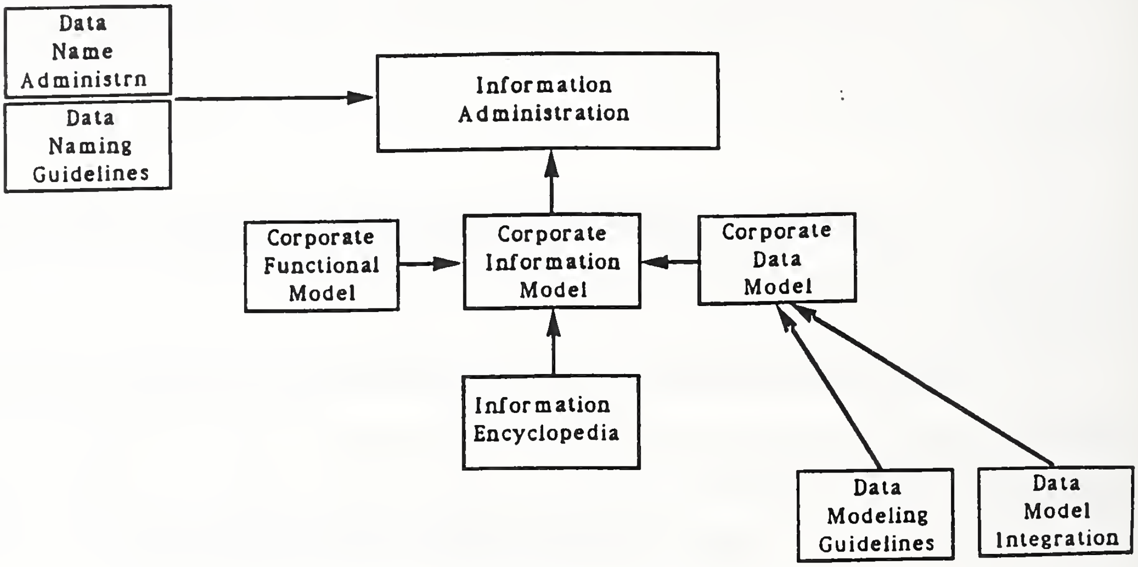
November 16 - 16, 1989

BELLCORE DATA NAMING GUIDELINES

Madhu S. Singh

201-829-3306

DATA NAME ADMINISTRATION AT BELLCORE



DATA NAME ADMINISTRATION AT BELLCORE

Mission:

- Standardize data type names across systems
- Improve communication among data users
- Increase sharability of data
- Reduce data maintenance and management cost

How:

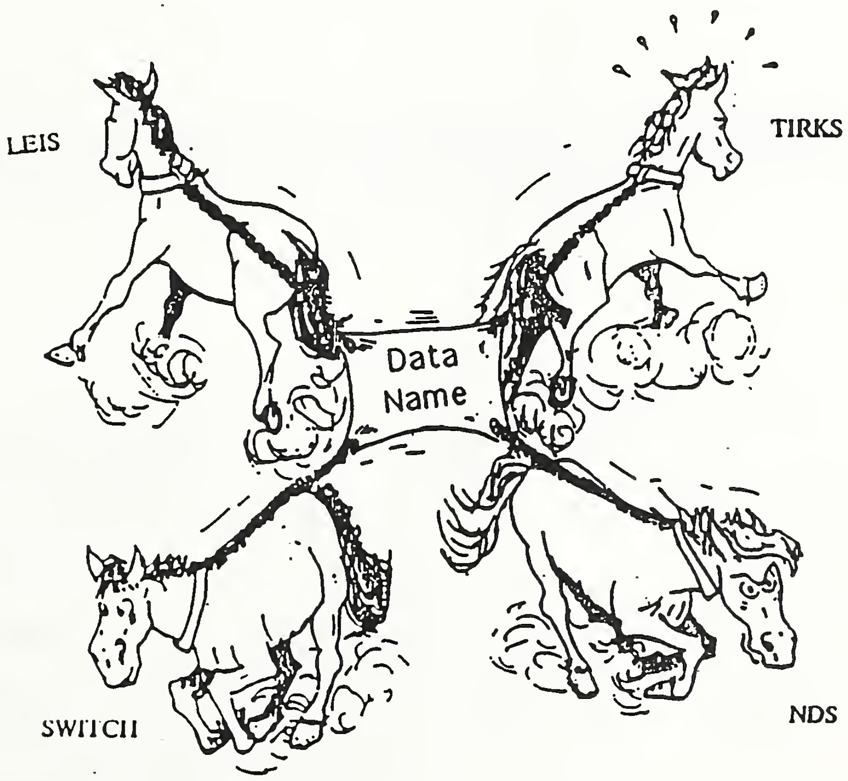
- Developing Bellcore data naming guidelines
- Developing framework for data name management
- Interfacing with different stakeholders in data naming arena

BELLCORE DATA NAMING GUIDELINES

Issues:

- Inconsistent data naming across systems
- Many data name types
- Numerous homonyms and synonyms
- Mechanization of data name administration

DATA NAMING GUIDELINES



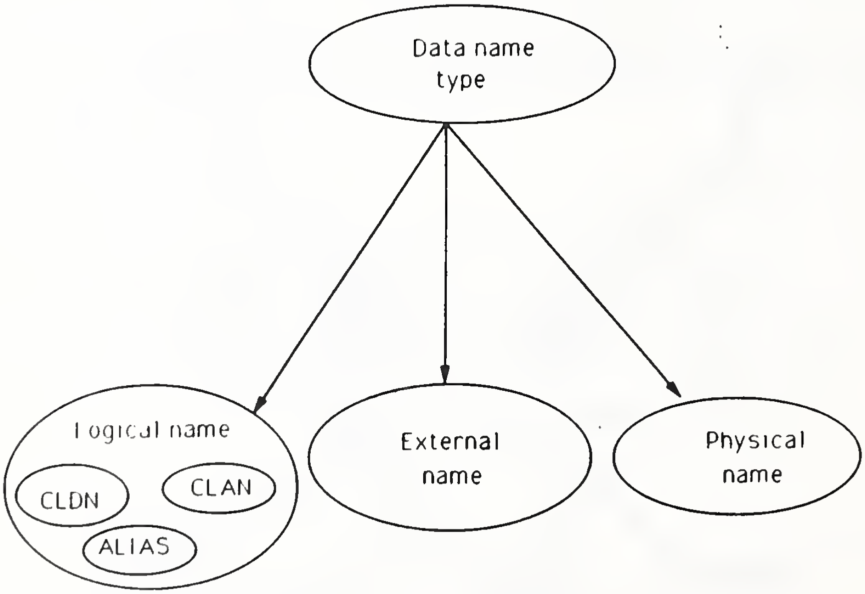
BELLCORE DATA NAMING GUIDELINES

Goal:

- Standardize data names throughout Bellcore
- Guide the creation of the most meaningful name
- Standardize acronyms and abbreviations used across applications.
- Eliminate redundant/inconsistent data names, e.g., homonyms and synonyms

BELLCORE DATA NAMING GUIDELINES

Data Name Types



Data name type

Logical name

CLDN

CLAN

ALIAS

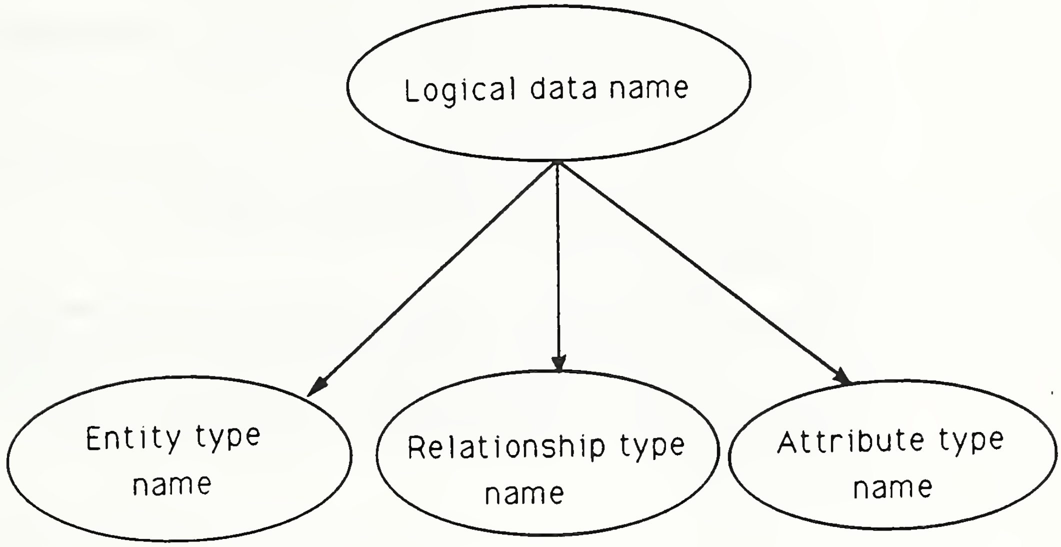
External name

Physical name

```
GLDN-Corporate
Logical Data Name
CLAN-Corporate
Logical Access Name
```

BELLCORE DATA NAMING GUIDELINES

Logical Data Name



BELLCORE DATA NAMING GUIDELINES

Corporate Logical Data Names

Rules:

- Data Names Consist of Components (e.g., employee name, employee birth date)
- Format and Size of Data Name
  - No Limit on Length of CLDN
  - Logical access names are limited to 30 characters
- Ordering or Sequencing of Data Name Components
  - Ordering is left to right
  - Most generic to most specific

DATA NAMING GUIDELINES

Guidelines for Naming Entity Type

- Format Guidelines
  - An entity name must appear in full English. Abbreviations and acronyms are not allowed as an entity name
  - An entity name must be singular noun form (employee, customer)
  - All letters be in lower case
  - An underscore "_" should be used as a delimiter between the parts of a single name component (e.g., sub_assembly_model)
  - A period "." should be used as a separator between the context phrase and prime phrase components of an entity type name (e.g., sub_assembly_model.parts_configuration) No other special characters are permitted as a part of entity names.

DATA NAMING GUIDELINES

Guidelines for Naming Relationship Type

- Format Guidelines
  - A relationship name must be in full English. Abbreviations/acronyms are not allowed
  - A relationship name must be singular noun or gerund (e.g., employing, offering, and managing)
  - All letters should be in lower case
  - An underscore "_" should be used to separate the parts of a single name components
  - A period "." should be used to separate the context phrase and prime phrase components of a relationship name
  - No other special characters are permitted as part of a relationship name

DATA NAMING GUIDELINES

Guidelines for Naming Attribute Type

- Format Guidelines
  - An attribute name must be in full English. Abbreviation/acronyms are not allowed
  - An attribute name must be a singular noun
  - All letters should be in lower case
  - An underscore "_" should be used as a separator between the parts of a singler name component (e.g., birth_date)
  - A period "." should be usd to separate the context phrase and class phrase components in an attribute name (e.g., regular_customer.birth_date)
  - No other special characters are permitted

DATA NAMING GUIDELINES

Guidelines for Naming Corporate Logical Access Name

- Format Guidelines
  - Do not use access names longer than 30 characters
  - Use standard abbreviation and acronyms if they exist for name components
  - Abbreviate names longer than 12 characters if there are no standard abbreviations.
- Guidelines for Selecting Components
  - Every access name must have a "context phrase" abbreviated to 3 characters long.
  - Key name for an attribute access name must be selected from "Class" name list

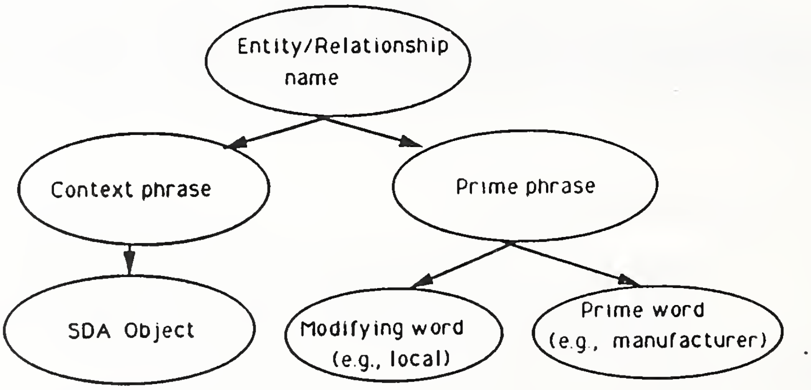
DATA NAMING GUIDELINES

Guidelines for Corporate Logical Access Name (Continued)

- Ordering of Components
  - Starting from left, the first 3 characters should be assigned to context phrase, followed by key name
  - Place qualifying names between context phrase and key name
  - For two or more qualifying names, arrange them from left to right to most generic to most specific.

LOGICAL DATA NAME STRUCTURE

Entity/Relationship Names

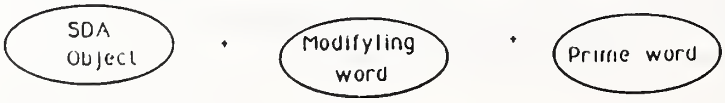


Examples
:

SDA Object
.
Prime word

catalog.item_manufacturer

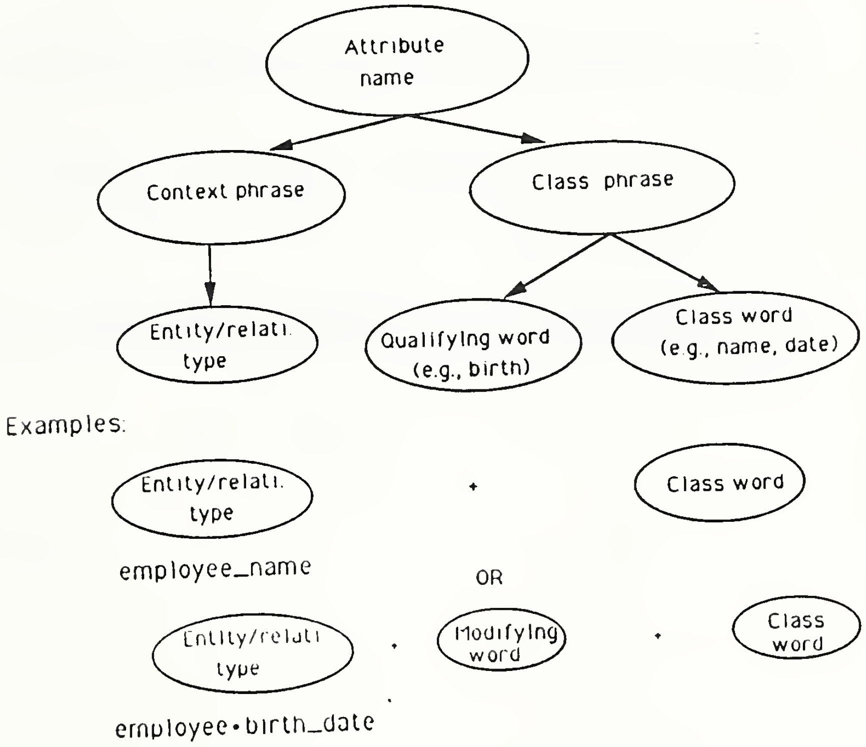
OR



catalog_item•local_manufacturer

BELLCORE DATA NAMING GUIDELINES

Summary



BELLCORE DATA NAMING GUIDELINES

Stakeholders

- IRDS Standards committee
- Language Standards organizations
- Bellcore Client Companies
- ISO/CCITT Standards committees

BELLCORE DATA NAMING GUIDELINES

Summary

- Consistency in data naming is most vital to achieve the Bellcore mission of treating and sharing data as a corporate resource.
- Working with Bell Operating Companies, Operations Technology and ANSI/X3H4 to develop appropriate data naming guidelines.
- Ensuring conformation of data naming guidelines by making them as a integral part of the data administration policies and procedures.

Naming Conventions Presentation

Space Station Naming

Dr. Anthony J. Winkler
CTA, Inc.

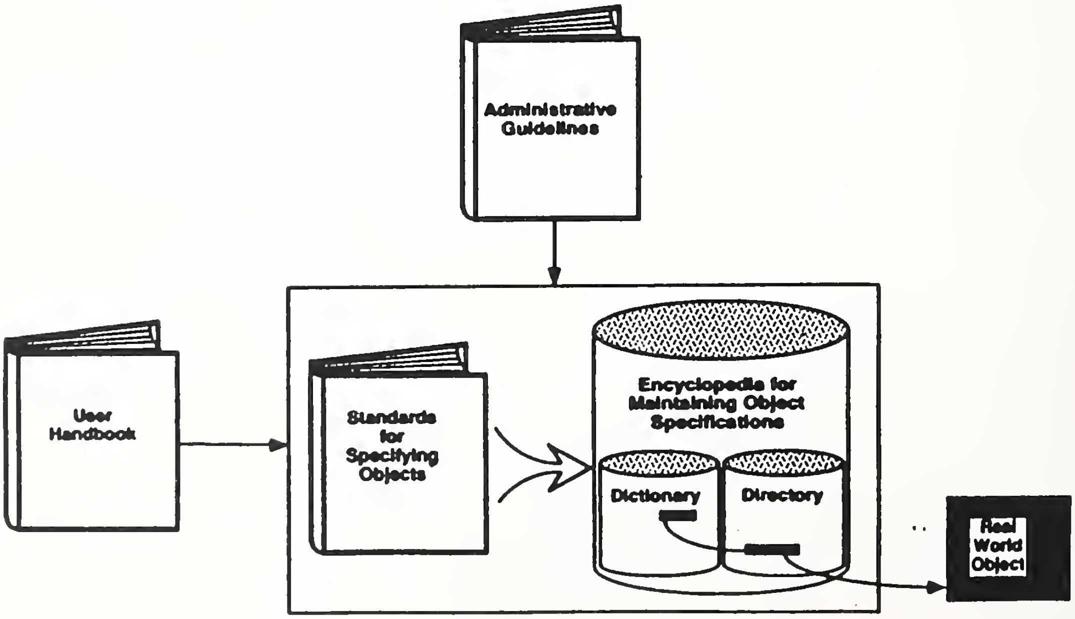
CTA
INCORPORATED

A PROPOSAL FOR OBJECT NAMING STANDARDS

JOINT X3H4.4/NIST NAMING CONVENTION FORUM

DR. JERRY WINKLER
CTA INCORPORATED
NOVEMBER 16, 1989

Relationship of Standards


Administrative Guidelines
User Handbook
Standards for Specifying Objects
Encyclopedia for Maintaining Object Specifications
Dictionary
Directory
Real World Object

OBJECT NAMING REQUIREMENTS

- THE SPACE STATION FREEDOM PROGRAM (SSFP) NAMING STANDARDS WERE DEVELOPED TO SATISFY SIX PRINCIPAL NAMING NEEDS WITHIN THE SSFP:

  (1) TO PROVIDE A MEANS FOR GLOBALLY AND UNAMBIGUOUSLY IDENTIFYING SSFP OBJECTS.

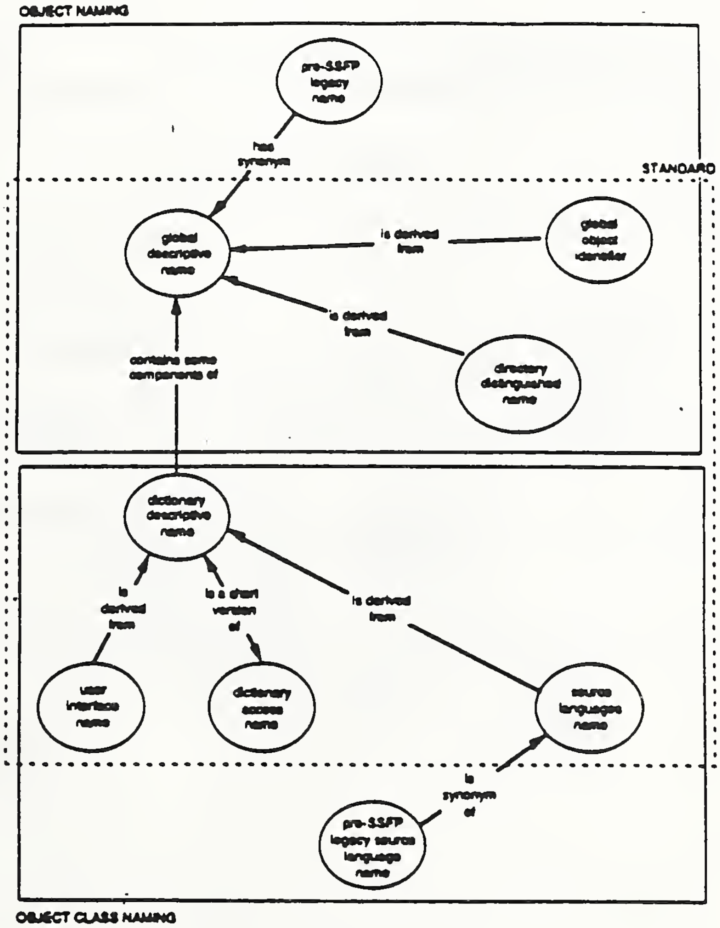
  (2) TO PROVIDE A METHOD FOR NAMING OBJECTS WHICH CAN BE USED IN MORE THAN ONE LOCATION IN THE SSFP, WITHOUT LOSING THE IDENTITY OF THE OBJECT.

  (3) TO PRECISELY DESCRIBE THE OBJECT BY ITS NAME

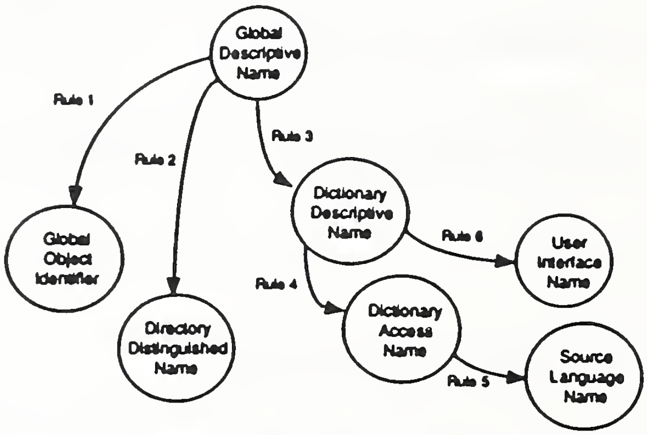
  (4) TO BE ABLE TO RELATE SIMILAR OBJECTS TO EACH OTHER BY THEIR NAME

  (5) TO PROVIDE AN APPROACH FOR NAMING AN OBJECT INDEPENDENT OF ITS PHYSICAL IMPLEMENTATION

  (6) TO SUPPORT DIFFERENT TYPES OF NAMES FOR OBJECTS

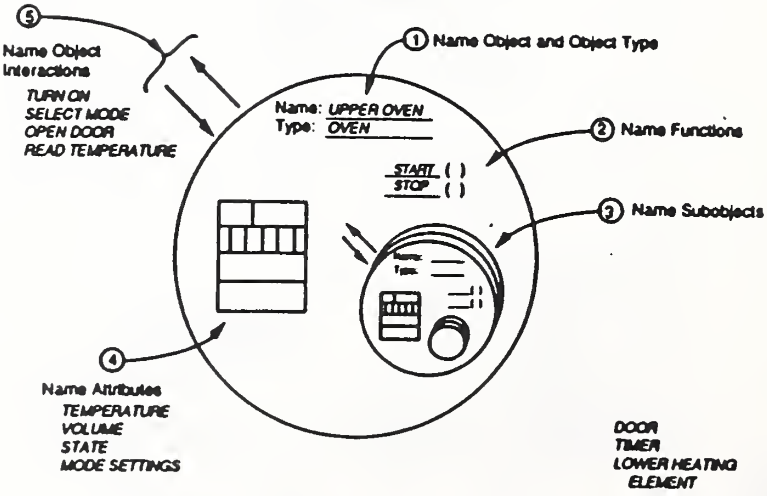


Mapping between the GDN and the other Name Types



Mappings Between the GDN and Other Name Types
Figure 4-1

Object Description



CTA INCORPORATED

GLOBAL NAME TERMS

GLOBAL NAME TERMS ARE USED TO CONSTRUCT THE GLOBAL DESCRIPTIVE NAME

EACH GLOBAL NAME TERM, OR "TERM," REPRESENTS A CERTAIN ASPECT OF THE OBJECT

EACH TERM IS DEFINED AS ONE OR MORE WORDS USED AS A SINGLE WORD

GLOBAL NAME TERMS ARE DIVIDED INTO AN ADMINISTRATIVE NAME PART AND A TECHNICAL NAME PART

CTA INCORPORATED

TECHNICAL NAME TERMS

VERB (VERB)

THE VERB IS USED TO IDENTIFY THE ACTION OR OPERATION PERFORMED BY CERTAIN EXECUTABLE OR FUNCTIONAL OBJECTS.

OBJECT ABSTRACTION (OAB)

THE OAB IS THE PRINCIPAL FUNCTIONAL IDENTITY OF AN OBJECT

OBJECT ABSTRACTIONN ATTRIBUTE (OAA)

THE OAA MODIFIES THE OBJECT ABSTRACTION TO MAKE IT MORE SPECIFIC

OBJECT ABSTRACTION LEVEL (OAL)

THE OAL TERM IDENTIFIES THE LEVEL OF PROCESSING OF THE OBJECT

OBJECT ATTRIBUTE VALUE (OAV)

THE OAV TERM IS USED TO QUALIFY THE OBJECT ABSTRACTION BY SPECIFYING WHAT KIND OF OBJECT ABSTRACTION IT IS

CTA INCORPORATED

TECHNICAL NAME TERMS (CONCLUDED)

INSTANCE COMPONENT

THE INSTANCE QUALIFIER TERM IS THE SPECIFIC INSTANCE IDENTIFIER OF THE OBJECT

CODED INSTANCE (CI) IDENTIFIERS SUCH AS A NUMBER CAN BE USED TO DIFFERENTIATE INSTANCES, AS CAN CHARACTER STRING INSTANCE (SI) IDENTIFIERS SUCH AS "PRIMARY"

CTA INCORPORATED

GLOBAL DESCRIPTIVE NAME SYNTAX SUMMARY

LEGEND:

<...>	=	naming component
::=	=	"is defined as"
[...]	=	optional
+	=	one or more occurrences of the preceding naming component
|	=	"or"
--	=	comment (to end of line)

GLOBAL DESCRIPTIVE NAME

```
<GDN> ::  = [<ACT>]<ANE><TNE> -- Global Descriptive Name

<ACT>::   = <VERB>                   -- Action
```

Administrative Name Expression

```
<ANE> ::  = <SSFP>                   -- "SSFP"
            <SSFP>.<SEP>             -- System/Element/Payload
            <SSFP>.<SEP>.<SS>        -- Subsystem
```

CTA INCORPORATED

GLOBAL DESCRIPTIVE NAME SYNTAX SUMMARY

Technical Name Expression

```
-- Technical Name Expression
   <TNE>::    = <OIE>                       -- Object Instance Expression
                  <OIE>.<TNE>

-- Object Instance Expression
   <OIE>:: = (<OC>)                         -- Object Class  (Possible Dictionary Name)
             <SI>(<OC>)
             (<OC>)<CI>

-- Object Class
   <OC> :: = [<ACT>](<OCT>)                 -- Object Class Tree  (Probably Dictionary Name

-- Object Class Tree
   <OCT> ::= <AOC>                          -- Administrative Object Class
             <AOC>(<OCT>)

-- Administrative Object Class
   <AOC> ::= (<OCE>) |                      -- Object Class Expression
             <ANP>(<OCE>)                   -- Administrative Name Part
```

CTA INCORPORATED

GLOBAL DESCRIPTIVE NAME SYNTAX SUMMARY

```
-- Object Class Expression
   <OCE>:: = <OAB> |                        -- Object Abstraction      (Subtype or Type)
             <OAV>(<OCE>) |                 -- Object Attribute Value
             (<OCE>)<OAQ>                   -- Object Abstraction Qualifier

-- Object Abstraction Qualifier
   <OAQ:>::= <OAA> |                        -- Object Abstraction Attribute
             <OAA><OAL>                     -- Object Abstraction Level

-- Administrative Name Part
   <ANP>::=  <SSFP>|
             [<SSFP>.]<SEP> |
             [<SSFP>.][<SEP>.]<SS>
```

Elementary Name Components:

```
<CI>      :: =  <term>        -- Instance as a Coded Identifier
<OAB>     :: =  <term>        -- Object Abstraction
<OAL>     :: =  <term>        -- Object Abstraction Level
<OAA>     :: =  <term>        -- Object Abstraction Attribute
<OAV>     :: =  <term>        -- Object Attribute Value
<SEP>     :: =  <term>        --  System/Element/Payload
<SI>      :: =  <term>        -- Instance as a String Identifier
<SS>      :: =  <term>        -- Subsystem
<SSFP>    :: =  <term>        -- "SSFP"
<VERB>    :: =  <term>        -- active verb
<term>    : =   <word> |
                <group of words treated as a single word>
```

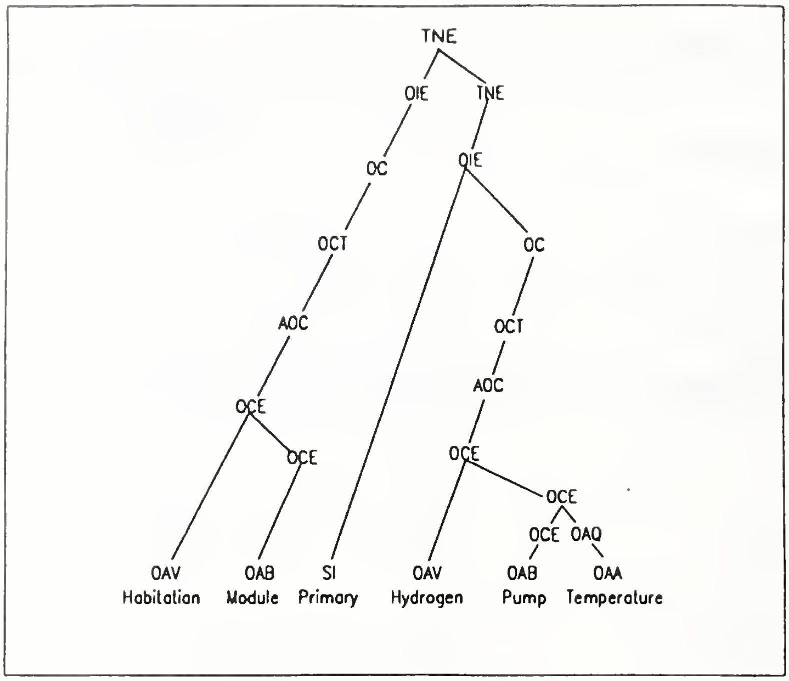


Figure 7-2 Graphical Example of Deriving a TNE for an Attribute of an Object

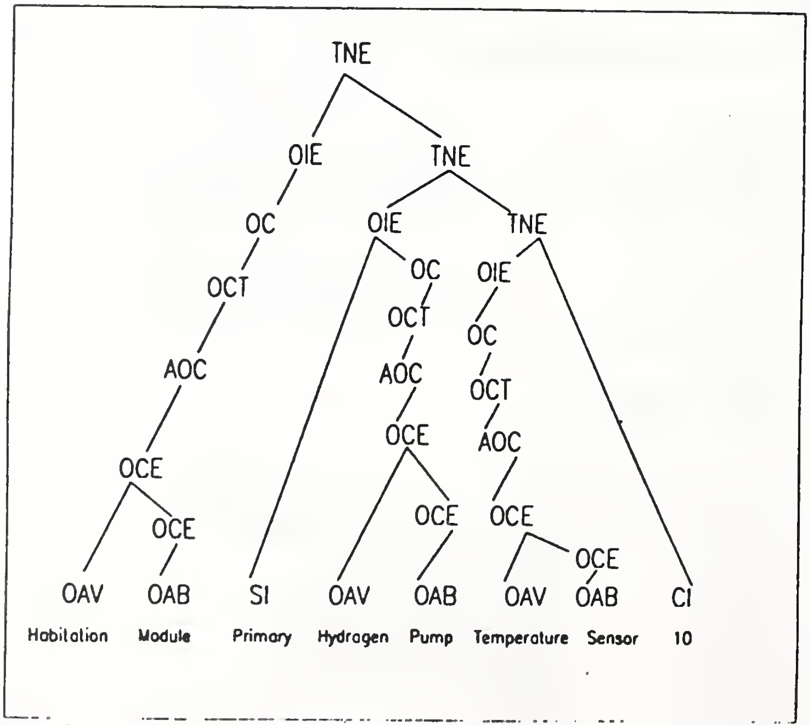


Figure 7-1 Deriving a Technical Name Expression for an Object

General Bibliography

ANSI, American National Standard Guidelines for Thesaurus Structure, Construction, and Use, ANSI Z39.19-1980, June, 1980.

ANSI, American National Standard X3.138-1988, Information Resource Dictionary System, American National Standards Institute, New York, 1988.

Goldfine, A. H. and Konig, P. A., A Technical Overview of the Information Resource Dictionary System (Second Edition), NBSIR 88-3700, National Bureau of Standards, Gaithersburg, MD, January, 1988.

Goldfine, A. H., Using the Information Resource Dictionary System Command Language (Second Edition), NBSIR 88-3701, National Bureau of Standards, Gaithersburg, MD, January, 1988.

Law, M. H., Guide to Information Resource Dictionary Applications: General Concepts and Strategic Systems Planning, NBS Special Publication 500-152, National Bureau of Standards, Gaithersburg, MD, April, 1988.

Newton, J. J., Guide on Data Entity Naming Conventions, NBS Special Publication 500-149, National Bureau of Standards, Gaithersburg, MD, October, 1987.

Rosen, B.K. and Law, M. H., Guide to Data Administration, NIST Special Publication 500-173, National Institute of Standards and Technology, Gaithersburg, MD, October, 1989.

NIST-114A
(REV. 3-89)

U.S. DEPARTMENT OF COMMERCE
NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY

BIBLIOGRAPHIC DATA SHEET

1. PUBLICATION OR REPORT NUMBER
NISTIR 4374

2. PERFORMING ORGANIZATION REPORT NUMBER

3. PUBLICATION DATE
JULY 1990

4. TITLE AND SUBTITLE
The Naming Forum: Proceedings of the IRDS Workshop on Data Entity Naming Conventions

5. AUTHOR(S)
Judith J. Newton, Editor

6. PERFORMING ORGANIZATION (IF JOINT OR OTHER THAN NIST, SEE INSTRUCTIONS)
U.S. DEPARTMENT OF COMMERCE
NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY
GAITHERSBURG, MD 20899

7. CONTRACT/GRANT NUMBER

8. TYPE OF REPORT AND PERIOD COVERED

9. SPONSORING ORGANIZATION NAME AND COMPLETE ADDRESS (STREET, CITY, STATE, ZIP)
NIST, ANS X3H4

10. SUPPLEMENTARY NOTES

□ DOCUMENT DESCRIBES A COMPUTER PROGRAM; SF-185, FIPS SOFTWARE SUMMARY, IS ATTACHED.

11. ABSTRACT (A 200-WORD OR LESS FACTUAL SUMMARY OF MOST SIGNIFICANT INFORMATION. IF DOCUMENT INCLUDES A SIGNIFICANT BIBLIOGRAPHY OR LITERATURE SURVEY, MENTION IT HERE.)

As part of the Federal Information Processing Standard (FIPS) Information Resource Dictionary System (IRDS)project, NIST has sponsored a series of workshops intended to produce guidance to the developers and users of the IRDS standard. The FIPS IRDS has been developed in close association with the American National Standard (ANS) Accredited Committee X3H4, which produced ANS X3.138-1988, Information Resource Dictionary Systems. The two standards are virtually identical.

The document summarizes the major points discussed both during speaker's presentations and general discussions at the IRDS Workshop-Naming Convention Forum held at NIST on 16-17 November, 1989. It was cosponsored by the Information Systems Engineering Division, National Computer Systems Laboratory, NIST, and X3H4.

The purposes of the workshop were to bring together data administrators concerned with naming conventions for a networking and discussion session, and to provide guidance to the X3H4.4 Task Group in the development of requirements for a Naming Convention Verification Module for the X3H4 IRDS standard.

12. KEY WORDS (6 TO 12 ENTRIES; ALPHABETICAL ORDER; CAPITALIZE ONLY PROPER NAMES; AND SEPARATE KEY WORDS BY SEMICOLONS)
ANS; data; data administration; data dictionary system; data management; data standards; FIPS, information; information management; IRDS; IRM; naming conventions.

13. AVAILABILITY

☒ UNLIMITED

□ FOR OFFICIAL DISTRIBUTION. DO NOT RELEASE TO NATIONAL TECHNICAL INFORMATION SERVICE (NTIS).

□ ORDER FROM SUPERINTENDENT OF DOCUMENTS, U.S. GOVERNMENT PRINTING OFFICE, WASHINGTON, DC 20402.

☒ ORDER FROM NATIONAL TECHNICAL INFORMATION SERVICE (NTIS), SPRINGFIELD, VA 22161.

14. NUMBER OF PRINTED PAGES
137

15. PRICE
A07

ELECTRONIC FORM