The National Bureau of Standards was established by an act of Congress on March 3, 1901. The Bureau's overall goal is to strengthen and advance the Nation's science and technology and facilitate their effective application for public benefit. To this end, the Bureau conducts research and provides: (1) a basis for the Nation's physical measurement system, (2) scientific and technological services for industry and government, (3) a technical basis for equity in trade, and (4) technical services to promote public safety. The Bureau's technical work is performed by the National Measurement Laboratory, the National Engineering Laboratory, and the Institute for Computer Sciences and Technology.

THE NATIONAL MEASUREMENT LABORATORY provides the national system of physical and chemical and materials measurement; coordinates the system with measurement systems of other nations; and technical services leading to accurate and uniform physical and chemical standards, and data and technical services to the public problems; conducts research and develops and maintains the ultimate user.

Applied Mathematical Sciences and Engineering and Consumer Products

THE NATIONAL ENGINEERING LABORATORY builds and maintains the ultimate user.

Applied Mathematical Sciences and Engineering and Consumer Products

THE INSTITUTE FOR COMPUTER SCIENCES AND TECHNOLOGY conducts research and provides the technical services leading to accurate and uniform physical and chemical standards, and data and technical services to the public problems; conducts research and develops and maintains the ultimate user.

Programming Science and Technology — Computer Systems Engineering.

1Headquarters and Laboratories at Gaithersburg, MD, unless otherwise noted; mailing address Washington, DC 20234.

2Some divisions within the center are located at Boulder, CO 80303.
Computer Science and Technology

NBS Special Publication 500-88

Software Development Tools

Raymond C. Houghton, Jr.

Center for Programming Science and Technology
Institute for Computer Sciences and Technology
National Bureau of Standards
Washington, DC 20234
Reports on Computer Science and Technology

The National Bureau of Standards has a special responsibility within the Federal Government for computer science and technology activities. The programs of the NBS Institute for Computer Sciences and Technology are designed to provide ADP standards, guidelines, and technical advisory services to improve the effectiveness of computer utilization in the Federal sector, and to perform appropriate research and development efforts as foundation for such activities and programs. This publication series will report these NBS efforts to the Federal computer community as well as to interested specialists in the academic and private sectors. Those wishing to receive notices of publications in this series should complete and return the form at the end of this publication.

National Bureau of Standards Special Publication 500-88
CODEN: XNBSAV

Library of Congress Catalog Card Number: 82-600517
TABLE OF CONTENTS

1. Introduction................................. 1

2. A Profile of Software Tools.................. 3
   2.1 General Classes of Software Tools......... 3
   2.2 Taxonomic View of Software Tools......... 5
   2.3 Environments of Software Tools........... 16
   2.4 Availability of Software Tools.......... 19
   2.5 Literature and Sources on Software Tools.. 20

3. Conclusions................................... 24

4. References.................................... 25

Appendices

Appendix A. Tools by General Classification
Appendix B. Tools by Input Subject
Appendix C. Tools by Transformation Features
Appendix D. Tools by Static Analysis Features
Appendix E. Tools by Dynamic Analysis Features
Appendix F. Tools by User Output
Appendix G. Tools by Machine Output
Appendices (continued)

Appendix H. Tools by Portability
Appendix I. Tools by Source Language
Appendix J. Tools by Hardware
Appendix K. Tools by Software
Appendix L. Tools in the Public Domain
Appendix M. Tools by Information Source
Appendix N. Tool Abstracts
Software Development Tools
Raymond C. Houghton, Jr.

Abstract

As a part of the program to provide information to Federal agencies on the availability, capabilities, limitations, and applications of software development tools, a database of information about existing tools was collected over a 3-year period. The purpose of this report is to present an analysis of the information contained in the database. Various categorizations of the tools are presented in classes listed by their characteristic features. The lists incorporate percentage summaries that are based on the total number of tools for which information is stored in the database. Trends found in the lists are analyzed and discussed. Abstracts of each tool are presented in an appendix.

Key words: programming aids; software automation; software development; software engineering; software testing; software tools.

1. Introduction

The Institute for Computer Sciences and Technology (ICST) within the National Bureau of Standards (NBS) has a mission under Public Law 89-306 (Brooks Act) to develop standards to enable the "economic and efficient purchase, lease, maintenance, operation, and utilization of automatic data processing equipment by Federal departments and agencies." As part of its current program, ICST is studying and evaluating methods that increase the productivity and quality of software procured by the Government and software developed within the Government. A partial solution to achieving higher quality software and increasing programmer productivity appears to lie in the use of the computer technology itself. Automation must be used to serve and augment itself. A recent

-----------------------------
NOTE: Certain commercial products are identified in this report in order to adequately present the subject area. In no case does identification of commercial products imply recommendation or endorsement by the National Bureau of Standards, nor does it imply that the product identified is necessarily the best or the only one available for the purpose.
GAO report [GAO80] endorses this use of automation and concludes that software tools can offer the Federal Government the following:

--- Better management control of computer software development, operation, maintenance, and conversion.

--- Lower costs for computer software development, operation, maintenance, and conversion.

--- Feasible means of inspecting both contractor-developed and in-house-developed computer software for such quality indications as conformance to standards and thoroughness of testing.

To this end, ICST initiated a project to study, evaluate, and make recommendations concerning the use of software development tools. A taxonomy of software tool features was published [Houg81] to provide a means for classifying and evaluating the capabilities of the ever more complex tools available in the marketplace. A survey of the use of software tools [Hech81] by representative software development groups in Government agencies, Government-supported research agencies, and commercial organizations was performed. NBS co-sponsored the tool fair [Houg81a] that was held in conjunction with the Fifth International Conference on Software Engineering.

As a part of the program to provide information to Federal agencies on the availability, capabilities, limitations, and applications of software development tools, a database containing information about existing tools [1] was collected over a 3-year period. Last year, a report [Houg80] was published to review the contents of the database. Since that report, the following work has been completed: (1) information on over 100 tools was added

1] The database contains information about 362 software development tools. There are, however, many more tools than this (for example, see [Sch181], [RCl81], or [SRA81]). A deliberate effort was made to keep the scope of the database within reason, but at the same time include a representative set of tools. Examples of tools that were excluded from the database include: (1) traditional software tools that are normally provided by computer vendors, such as compilers, editors, and operating system utilities; (2) assembly and machine language tools; and (3) tools that are not necessarily oriented to the software development process, such as management information systems and database management systems. However, when a completed submission was received directly from the developer of a tool, the information was almost always included in the database. Further discussion of the sources for tool information is presented in section 2.5.
to the database, (2) all the tools were classified according to the taxonomy of tool features, (3) many retrieval programs were developed, and (4) much of the information was updated.

The purpose of this report is to analyze and to present the current information on software development tools that is contained in the database. Various categorizations of the tools are presented in classes listed by their characteristic features. The lists incorporate percentage summaries that are based on the total number of tools for which information is stored in the database. These lists provide an overview of the database information as well as a means to determine tools of interest for a given category. The appendices provide additional information about specific tools.

2. A Profile of Software Tools

There are many ways in which one can view the characteristics of software tools [Houg81b]. The approach that is taken in this section is to choose several different vantage points. Each of the various points, of course, is based on the inherent structure of the database.

The first vantage point is a coarse functional view of tools. A very simple classification system that consists of only six categories is used. This view is followed by a much more detailed perspective in Section 2.2 that is based on the taxonomy of tool features. In this section, each feature is described and defined. Percentage summaries of tools that are characterized by these features are presented. The third view in Section 2.3 involves the hardware and software characteristics of tools. The fourth and fifth views, Sections 2.4 and 2.5, involve other important considerations about software tools, such as their availability, their sources, and their documentation.

2.1 General Classes of Software Tools

Software development tools have grown in complexity in recent years as have other software applications. Most early tools performed a single function and were very simple to describe. Tutorial papers of this era [Rama75] [Reif75] classified tools usually according to the function that they performed. These early, simpler tools have since given way to more complex tools which have increased functionality. The evolution of tool development is causing major shifts in the types of tools being developed and marketed. For example, formerly one heard a lot about compilers, debuggers, dump analyzers, flowcharters, and editors. Now one hears a lot about
software development systems, application generators, software engineering facilities, program generators, and programming environments. The capabilities of these current tools are much more sophisticated than their earlier counterparts.

Consequently, the tools in the database are not extensively classified according to traditional schemes because of their limitation in describing current technology. The number of categories in the traditional classification scheme was reduced to the following six:

Software Management, Control, and Maintenance Tools (MAC)
Software Modeling and Simulation Tools (SAM)
Requirements/Design Specification and Analysis Tools (RAD)
Program Construction and Generation Tools (GEN)
Source Program Analysis and Testing Tools (TAA)
Software Support System/Programming Environment Tools (ENV)

Since the above classes are not mutually exclusive, this categorization provides only a broad overview of the types of tools currently available.

The list that follows shows the number and percentage of tools [1] in the database that fall into these classes.

<table>
<thead>
<tr>
<th>Class</th>
<th>Number</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>TAA...........</td>
<td>126</td>
<td>34%</td>
</tr>
<tr>
<td>MAC...........</td>
<td>120</td>
<td>33%</td>
</tr>
<tr>
<td>RAD...........</td>
<td>54</td>
<td>14%</td>
</tr>
<tr>
<td>GEN...........</td>
<td>37</td>
<td>10%</td>
</tr>
<tr>
<td>SAM...........</td>
<td>13</td>
<td>3%</td>
</tr>
<tr>
<td>ENV...........</td>
<td>12</td>
<td>3%</td>
</tr>
</tbody>
</table>

Appendix A lists the names of the tools in each class. The two most popular classes are MAC and TAA which, when combined, comprise almost three-quarters of all the tools in the database. The tools in these classes are primarily oriented to the analysis or management of programs written in a specific language. Two of the lesser populated classes, ENV and GEN, contain many tools

[1] In all cases, except where noted, percentage values are based on the total number of tools in the database. The summation of percentages for a given profile is not, in most cases, equal to 100 percent for one of the following reasons: (1) truncation of the fraction, (2) a tool having no information for the profile subject, or (3) a tool being counted more than once within the profile subject. For example, a tool is likely to have more than one static analysis feature.
that have been more recently developed and, in some cases, are still under development. Tools in these classes are popular research areas.

2.2 Taxonomic View of Software Tools

To provide a more useful way of identifying tools of interest, each of the tools in the database is classified according to the taxonomy of tool features. The taxonomy is a hierarchical arrangement of software tool features and is illustrated in Figure 1. The highest level (@) is the most abstract and covers all the features below it. The second level of the taxonomy covers the basic processes of a tool. These are input (in), function (fn), and output (out). At the third level are the classes of tool features. These are subject (I), control input (C), transformation (T), static analysis (S), dynamic analysis (D), user output (U), and machine output (M). At the bottom or feature level of the hierarchy are a total of 53 tool features. The ranges between brackets signify the number of features in each of the classes.

\[
\begin{align*}
\text{@} & \quad \text{[in]} \quad \text{[fn]} \quad \text{[out]} \\
\text{[in]} & \quad \text{[fn]} \quad \text{[out]} \\
\text{[I]} & \quad \text{[C]} \quad \text{T} \quad \text{[S]} \quad \text{[D]} \quad \text{[U]} \quad \text{[M]} \\
\text{[1-4]} & \quad \text{[1-2]} \quad \text{[1-7]} \quad \text{[1-19]} \quad \text{[1-10]} \quad \text{[1-5]} \quad \text{[1-6]}
\end{align*}
\]

Figure 1. Taxonomy of Tool Features

In classifying tools for the database, some liberties were taken to extend the taxonomy when more detailed information was available or when a new feature was discovered. Also, on occasion, some features were further specified to avoid confusion with other features of the same name. For example, "data" which is input was changed to "data input" to avoid confusion with "data" that is output. This experience will influence future revisions of the taxonomy.

In the paragraphs that follow, the taxonomy is used to organize the detailed analysis of the types of tools and tool features that are found in the database. For completeness, definitions of most of the terms in the taxonomy are also given.
INPUT

Tool input features are based on the forms of input which can be provided to a tool. These features fall into two classes, one which is based on how the tool should operate (control input) and the other based on what the tool should operate on (subject). The difference between these classes is clarified further in the paragraphs that follow.

Subject. The subject is usually the main input to a tool. It is the input which is subjected to the main functions performed by a tool. The four types of tool subjects are code, VHLL (very high level language), data, and text. Although the difference between these types is somewhat arbitrary, the taxonomy has very specific definitions for each.

The list that follows shows the number and percentage of tools that fall in these classes.

<table>
<thead>
<tr>
<th>Subject Type</th>
<th>Number</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>CODE INPUT</td>
<td>246</td>
<td>67%</td>
</tr>
<tr>
<td>VHLL INPUT</td>
<td>76</td>
<td>20%</td>
</tr>
<tr>
<td>DATA INPUT</td>
<td>69</td>
<td>19%</td>
</tr>
<tr>
<td>TEXT INPUT</td>
<td>19</td>
<td>5%</td>
</tr>
</tbody>
</table>

Appendix B lists the tools with each of these features and with the extensions defined below for code and VHLL input. Data and text input were not extended further.

Code Input - accepts a program written in a high level, assembly, or object level language. Code is the language form in which most programming solutions are expressed. The tools in this class are, in most cases, further classified according to the specific language that they accept. The list that follows shows the number and percentage of tools classified by language that is accepted as input.

<table>
<thead>
<tr>
<th>Language</th>
<th>Number</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>FORTRAN</td>
<td>110</td>
<td>30%</td>
</tr>
<tr>
<td>COBOL</td>
<td>41</td>
<td>11%</td>
</tr>
<tr>
<td>JOVIAL</td>
<td>15</td>
<td>4%</td>
</tr>
<tr>
<td>STRUCTURED FORTRAN</td>
<td>10</td>
<td>2%</td>
</tr>
<tr>
<td>ASSEMBLY LANGUAGE</td>
<td>8</td>
<td>2%</td>
</tr>
<tr>
<td>BASIC</td>
<td>5</td>
<td>1%</td>
</tr>
<tr>
<td>CICS</td>
<td>4</td>
<td>1%</td>
</tr>
<tr>
<td>OBJECT CODE INPUT</td>
<td>4</td>
<td>1%</td>
</tr>
<tr>
<td>PL/1</td>
<td>3</td>
<td>0%</td>
</tr>
<tr>
<td>IFTRAN</td>
<td>3</td>
<td>0%</td>
</tr>
<tr>
<td>STRUCTURED COBOL</td>
<td>3</td>
<td>0%</td>
</tr>
<tr>
<td>ADA</td>
<td>3</td>
<td>0%</td>
</tr>
<tr>
<td>SRTRAN</td>
<td>3</td>
<td>0%</td>
</tr>
<tr>
<td>PASCAL</td>
<td>2</td>
<td>0%</td>
</tr>
<tr>
<td>COMPASS</td>
<td>2</td>
<td>0%</td>
</tr>
<tr>
<td>RATFOR</td>
<td>2</td>
<td>0%</td>
</tr>
</tbody>
</table>
Almost half of the tools with the code input feature accept FORTRAN and they comprise about one-third of all the tools in the database. However, it is still interesting to note the range of languages accepted by code input tools. The range includes LISP, SNOBOL, BASIC, CHILL, PL/I, ALGOL, APL, ADA, MODULA, and even memory dumps (although one could argue whether a memory dump is code). It is also somewhat disturbing to note the relatively low percentage of COBOL tools since COBOL is the most frequently used programming language. The main reason for this fact is that for COBOL-type applications, the traditional tools have been oriented toward data management problems. A proliferation of tools oriented specifically to the COBOL language (compilers excluded) is a phenomenon that is just recently starting to appear. In fact, approximately 25 percent of the last 60 tools that were added to the database are COBOL tools.

VHLL Input - accepts a program written in a very high level language that is typically not in an executable form. Tools with this feature may define programs, track program requirements throughout their development, or synthesize programs through use of some non-procedural VHLL. The list that follows shows the number and percentage of tools that fall in this class.

<table>
<thead>
<tr>
<th>Description</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>DESIGN SPECIFICATION</td>
<td>23</td>
<td>6%</td>
</tr>
<tr>
<td>REQUIREMENTS SPECIFICATION</td>
<td>13</td>
<td>3%</td>
</tr>
<tr>
<td>DESCRIPTION LANGUAGE</td>
<td>8</td>
<td>2%</td>
</tr>
<tr>
<td>STRUCTURED LANGUAGE</td>
<td>7</td>
<td>1%</td>
</tr>
<tr>
<td>PROGRAM SPECIFICATION</td>
<td>6</td>
<td>1%</td>
</tr>
<tr>
<td>REQUIREMENTS LANGUAGE</td>
<td>6</td>
<td>1%</td>
</tr>
<tr>
<td>DESIGN LANGUAGE</td>
<td>5</td>
<td>1%</td>
</tr>
</tbody>
</table>
Data Input - accepts a string of characters to which meaning is or might be assigned. The input (e.g. raw data) is not in an easily interpreted, natural language form.

Text Input - accepts statements in a natural language form. Certain types of tools are designed to operate on text only (e.g., text editors, document preparation systems) and require no other input except directives or commands. In most cases, these tools are oriented toward the development of documentation. Because the emphasis of the database is on tools that are specific to software development, there are many tools that have text as input that are not included in the database. Examples of excluded tools are general purpose text editors, data base management systems, report generators, and text processors.

Control Input. Tools that have control input features accept statements or data that specify the type of operations and any detail associated with the operations. Unfortunately, features in this area are very difficult to determine from tool descriptions. In a few cases, tools are described as being very interactive and a classification may be assumed. However, without actually seeing samples of the user input, features would have to be guessed. Consequently, very little data was collected and the results are not reported in this publication. This does not mean that control input features are not important. They relate to the user interface, which in many cases determines user acceptance or rejection of a tool. Tool descriptions sorely lack this type of information.

FUNCTION

The features for this class describe the processing functions performed by a tool and fall into three classes: transformation, static analysis, and dynamic analysis.

Transformation. Transformation features describe how the subject is manipulated to accommodate the user's needs. They describe what transformations take place as the input to the tool is processed. The list that follows shows the number and percentage of tools that have transformation features.
Appendix C lists tools with these features and with the extensions defined for translation.

**Formatting** - arranging a program according to predefined or user defined conventions. A tool that "cleans up" a program by making all statement numbers sequential, alphabetizing variable declarations, indenting statements, and making other standardizing changes has this feature.

**Translation** - converting from one language form to another. Tools that have this feature can be further classified with the following extensions: compilation, structure preprocessing, macro expansion and conversion. The list that follows shows the number and percentage of translation tools that have been further classified.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Number</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>STRUCTURE PREPROCESSING</td>
<td>18</td>
<td>4%</td>
</tr>
<tr>
<td>COMPILATION</td>
<td>8</td>
<td>2%</td>
</tr>
<tr>
<td>MACRO EXPANSION</td>
<td>8</td>
<td>2%</td>
</tr>
<tr>
<td>CONVERSION</td>
<td>7</td>
<td>1%</td>
</tr>
</tbody>
</table>

**Instrumentation** - adding sensors and counters to a program for the purpose of collecting information useful for dynamic analysis. Most code analyzers instrument the source code at strategic points in the program to collect execution statistics required for coverage analysis and tuning. See dynamic analysis features.

**Editing** - modifying the content of the subject by inserting, deleting, or moving characters, numbers, or data. Of course, there are a very large number of tools that have this feature that are not included in the database. Many of the ones that are included provide editing to enhance the capability of the user environment. Most tools do not include an editing capability and therefore require users to modify the subject through use of a vendor supplied editor.

**Synthesis** - generating an application or program from a specification or from an intermediate language. Tools that have this feature include application generators, program generators, compiler compilers, and preprocessor generators. Tools with this feature show promise toward increasing programmer productivity; thus there is considerable
research and development going on in this area.

Restructuring - reconstructing and arranging the subject in a new form according to well-defined rules. A tool that generates structured code from unstructured code is an example of a tool with this feature.

Optimization - modifying a program to improve performance, e.g. to make it run faster or to make it use fewer resources. Many vendors' compilers provide this feature. There are many tools that claim this feature, but do not really modify the subject program. Instead, these tools provide data on the results of execution which may be used for tuning purposes. See tuning under dynamic analysis.

Static Analysis. Static analysis features describe operations on the subject without regard to the executability of the subject. They describe the manner in which the subject is analyzed. The list that follows shows the static analysis features with the number of times they occur and their percentages.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>MANAGEMENT</td>
<td>95</td>
<td>26%</td>
</tr>
<tr>
<td>CROSS REFERENCE</td>
<td>61</td>
<td>16%</td>
</tr>
<tr>
<td>SCANNING</td>
<td>56</td>
<td>15%</td>
</tr>
<tr>
<td>AUDITING</td>
<td>30</td>
<td>8%</td>
</tr>
<tr>
<td>DATA FLOW ANALYSIS</td>
<td>22</td>
<td>6%</td>
</tr>
<tr>
<td>CONSISTENCY CHECKING</td>
<td>20</td>
<td>5%</td>
</tr>
<tr>
<td>STATISTICAL ANALYSIS</td>
<td>19</td>
<td>5%</td>
</tr>
<tr>
<td>ERROR CHECKING</td>
<td>19</td>
<td>5%</td>
</tr>
<tr>
<td>STRUCTURE CHECKING</td>
<td>19</td>
<td>5%</td>
</tr>
<tr>
<td>COMPARISON</td>
<td>15</td>
<td>4%</td>
</tr>
<tr>
<td>COMPLETENESS CHECKING</td>
<td>13</td>
<td>3%</td>
</tr>
<tr>
<td>COMPLEXITY MEASUREMENT</td>
<td>12</td>
<td>3%</td>
</tr>
<tr>
<td>TRACKING</td>
<td>12</td>
<td>3%</td>
</tr>
<tr>
<td>INTERFACE ANALYSIS</td>
<td>11</td>
<td>3%</td>
</tr>
<tr>
<td>I/O SPECIFICATION ANALYSIS</td>
<td>5</td>
<td>1%</td>
</tr>
<tr>
<td>TYPE ANALYSIS</td>
<td>3</td>
<td>0%</td>
</tr>
<tr>
<td>COST ESTIMATION</td>
<td>2</td>
<td>0%</td>
</tr>
<tr>
<td>UNITS ANALYSIS</td>
<td>2</td>
<td>0%</td>
</tr>
<tr>
<td>SCHEDULING</td>
<td>2</td>
<td>0%</td>
</tr>
</tbody>
</table>

Appendix D lists the tools with these features and with the extensions for management and statistical analysis.

Management - aiding the management or control of software development. Because of its broad coverage, management is further extended into the following areas:
CONFIGURATION MANAGEMENT............ 20 (5%)
GLOBAL VARIABLE MANAGEMENT........... 18 (4%)
PROJECT MANAGEMENT..................... 18 (4%)
DATA BASE MANAGEMENT................... 13 (3%)
CHANGE CONTROL......................... 11 (3%)
TEST DATA MANAGEMENT................... 10 (2%)
FILES MANAGEMENT....................... 10 (2%)
LIBRARY MANAGEMENT..................... 9 (2%)
VERSION CONTROL......................... 7 (1%)
DOCUMENTATION MANAGEMENT............... 5 (1%)
PERFORMANCE MANAGEMENT................ 3 (0%)
CAPACITY PLANNING...................... 2 (0%)
MANAGEMENT PLANNING.................... 1 (0%)

Cross Reference - referencing entities to other entities by logical means.

Scanning - examining an entity sequentially to identify key areas or structure.

Auditing - conducting an examination to determine whether or not predefined rules have been followed.

Data Flow Analysis - graphical analysis of the sequential patterns of definitions and references of data.

Consistency Checking - determining whether or not each entity is internally consistent in the sense that it contains uniform notation and terminology, and is consistent with its specification.

Statistical Analysis - performing statistical data collection and analysis.

Error Checking - determining discrepancies, their importance, and/or their cause.

Structure Checking - detecting structural flaws within a program (e.g., improper loop nestings, unreferenced labels, unreachable statements, and statements with no successors).

Comparison - determining and assessing differences between two or more items.

Completeness Checking - assessing whether or not an entity has all its parts present and if its parts are fully developed.

Complexity Measurement - determining how complicated an entity (e.g., routine, program, system, etc.) is by evaluating some number of associated characteristics. For example, the following characteristics can impact complexity:
instruction mix, data references, structure/control flow, number of interactions/interconnections, size, and number of computations.

Tracking - tracking the development of an entity through the software life cycle.

Interface Analysis - checking the interfaces between program elements for consistency and adherence to predefined rules and/or axioms.

I/O Specification Analysis - analyzing the input and output specification in a program, usually for the purpose of generating input data.

Type Analysis - evaluating whether or not the domain of values attributed to an entity are properly and consistently defined.

Cost Estimation - assessing the behavior of the variables which impact life cycle cost.

Units Analysis - determining whether or not the units or physical dimensions attributed to an entity are properly defined and consistently used.

Scheduling - assessing the software development schedule and its impact on the software life cycle.

Dynamic Analysis. Dynamic analysis features specify operations that are determined during or after execution takes place. Dynamic analysis features differ from those classified as static by virtue of the fact that they require some form of symbolic or machine execution. They describe the techniques used by the tool to derive meaningful information about a program's execution behavior. The list that follows shows the 10 dynamic analysis features along with the number of times they occur and their percentages.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Occurrences</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>COVERAGE ANALYSIS</td>
<td>40</td>
<td>11%</td>
</tr>
<tr>
<td>TRACING</td>
<td>30</td>
<td>8%</td>
</tr>
<tr>
<td>TUNING</td>
<td>29</td>
<td>8%</td>
</tr>
<tr>
<td>SIMULATION</td>
<td>19</td>
<td>5%</td>
</tr>
<tr>
<td>TIMING</td>
<td>18</td>
<td>4%</td>
</tr>
<tr>
<td>RESOURCE UTILIZATION</td>
<td>17</td>
<td>4%</td>
</tr>
<tr>
<td>SYMBOLIC EXECUTION</td>
<td>10</td>
<td>2%</td>
</tr>
<tr>
<td>ASSERTION CHECKING</td>
<td>9</td>
<td>2%</td>
</tr>
<tr>
<td>REGRESSION TESTING</td>
<td>6</td>
<td>1%</td>
</tr>
<tr>
<td>CONSTRAINT EVALUATION</td>
<td>3</td>
<td>0%</td>
</tr>
</tbody>
</table>
Appendix E lists the tools with these features and with the extensions for tracing.

Coverage Analysis - determining and assessing measures associated with the invocation of program structural elements to determine the adequacy of a test run. Coverage analysis is useful when the user is attempting to execute each statement, branch, path, or iterative structure (i.e., DO loops in FORTRAN) in a program.

Tracing - tracing the historical record of execution of a program. Because of its broad coverage, tracing has been, in many cases, further extended into the following areas:

- **PATH FLOW TRACING**...................... 13 (3%)
- **BREAKPOINT CONTROL**..................... 2 (0%)
- **LOGIC FLOW TRACING**..................... 2 (0%)
- **DATA FLOW TRACING**...................... 1 (0%)

Tuning - determining what parts of a program are being executed the most.

Simulation - representing certain features of the behavior of a physical or abstract system by means of operations performed by a computer.

Timing - reporting actual CPU times associated with a program or its parts.

Resource Utilization - analysis of resource utilization associated with system hardware or software.

Symbolic Execution - reconstructing logic and computations along a program path by executing the path with symbolic, rather than actual values of data.

Assertion Checking - checking of user-embedded statements that assert relationships between elements of a program. An assertion is a logical expression that specifies a condition or relation among the program variables. Checking may be performed with symbolic or run-time data.

Regression Testing - rerunning test cases which a program has previously executed correctly in order to detect errors spawned by changes or corrections made during software development and maintenance.

Constraint Evaluation - generating and/or solving path input or output constraints for determining test input or for proving programs correct.
OUTPUT

Output features provide links from the tool to both the human user and the target machine (where applicable). They describe the types and forms of outputs that are produced by a tool.

User Output. User output features describe the types of information that are returned from the tool to the human user and the forms in which these outputs are presented. There are five user output features. The list that follows shows the number and percentage of tools that fall in these classes.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>LISTINGS</td>
<td>171</td>
<td>(47%)</td>
</tr>
<tr>
<td>TABLES</td>
<td>151</td>
<td>(41%)</td>
</tr>
<tr>
<td>DIAGNOSTICS</td>
<td>84</td>
<td>(23%)</td>
</tr>
<tr>
<td>GRAPHICS</td>
<td>77</td>
<td>(21%)</td>
</tr>
<tr>
<td>USER-ORIENTED TEXT</td>
<td>59</td>
<td>(16%)</td>
</tr>
</tbody>
</table>

Appendix F lists the tools with these features and with the extensions for graphics and user-oriented text.

- **Listings** — output that lists source programs or data and that may be annotated.

- **Tables** — output that is arranged in parallel columns to exhibit a set of facts or relations in a definite, compact and comprehensive form.

- **Diagnostics** — output that simply indicates what software discrepancies have occurred.

- **Graphics** — a graphical presentation with symbols indicating operations, flow, etc. Graphics is further extended into the following areas:

<table>
<thead>
<tr>
<th>Feature</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>FLOW CHARTS</td>
<td>20</td>
<td>(5%)</td>
</tr>
<tr>
<td>HIERARCHICAL TREE</td>
<td>8</td>
<td>(1%)</td>
</tr>
<tr>
<td>DESIGN CHARTS</td>
<td>7</td>
<td>(1%)</td>
</tr>
<tr>
<td>ACTIVITY DIAGRAM</td>
<td>4</td>
<td>(1%)</td>
</tr>
<tr>
<td>CHARTS</td>
<td>2</td>
<td>(0%)</td>
</tr>
<tr>
<td>HIPO CHARTS</td>
<td>1</td>
<td>(0%)</td>
</tr>
<tr>
<td>LINE GRAPHS</td>
<td>1</td>
<td>(0%)</td>
</tr>
<tr>
<td>BAR CHARTS</td>
<td>1</td>
<td>(0%)</td>
</tr>
<tr>
<td>CONTROL MAP</td>
<td>1</td>
<td>(0%)</td>
</tr>
<tr>
<td>HISTOGRAMS</td>
<td>1</td>
<td>(0%)</td>
</tr>
<tr>
<td>MILESTONE CHARTS</td>
<td>1</td>
<td>(0%)</td>
</tr>
<tr>
<td>ACTIVITY DIAGRAMS</td>
<td>1</td>
<td>(0%)</td>
</tr>
<tr>
<td>STRUCTURE CHARTS</td>
<td>1</td>
<td>(0%)</td>
</tr>
</tbody>
</table>
User-Oriented Text - output that is in a natural language form. User-oriented text is further extended into the following areas:

- DOCUMENTATION........................................ 39 (10%)
- REPORTS.................................................. 23 (6%)

Machine Output. Machine output features handle the interface from the tool to a non-human user. The machine output can be directed to a target machine or to another tool for further processing. Machine output features describe what the receiving tool or machine expects as output. The list that follows shows the number and percentage of tools that fall in these classes.

<table>
<thead>
<tr>
<th>Source Code Output</th>
<th>103 (28%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>FORTRAN</td>
<td>45 (12%)</td>
</tr>
<tr>
<td>COBOL</td>
<td>27 (7%)</td>
</tr>
<tr>
<td>JOVIAL</td>
<td>7 (1%)</td>
</tr>
<tr>
<td>PL/1</td>
<td>2 (0%)</td>
</tr>
<tr>
<td>SRTRAN</td>
<td>2 (0%)</td>
</tr>
<tr>
<td>BASIC</td>
<td>1 (0%)</td>
</tr>
<tr>
<td>COMPASS</td>
<td>1 (0%)</td>
</tr>
<tr>
<td>IFTRAN</td>
<td>1 (0%)</td>
</tr>
<tr>
<td>SMAL/80</td>
<td>1 (0%)</td>
</tr>
<tr>
<td>RATFOR</td>
<td>1 (0%)</td>
</tr>
<tr>
<td>ALGOL</td>
<td>1 (0%)</td>
</tr>
<tr>
<td>SFTRAN</td>
<td>1 (0%)</td>
</tr>
<tr>
<td>SIMULA</td>
<td>1 (0%)</td>
</tr>
<tr>
<td>ATOM</td>
<td>1 (0%)</td>
</tr>
<tr>
<td>CMS-2</td>
<td>1 (0%)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Data Output</th>
<th>16 (4%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>OBJECT CODE OUTPUT</td>
<td>9 (2%)</td>
</tr>
<tr>
<td>INTERMEDIATE CODE</td>
<td>4 (1%)</td>
</tr>
<tr>
<td>VHLL OUTPUT</td>
<td>2 (0%)</td>
</tr>
<tr>
<td>PROMPTS</td>
<td>1 (0%)</td>
</tr>
</tbody>
</table>

Note that source code output features have been, in many cases, further classified according to the specific language that is output. Appendix G lists the tools in each class.

Source Code - a program written in a procedural language that must be input to a translation process before execution can take place.

Data - a set of representations of characters or numeric quantities to which meaning has been assigned.

Object Code - a program expressed in machine language which is normally an output of a given translation process.
Intermediate Code - code that is between source code and machine code.

VHLL - a program written in a very high level language.

Prompts - a series of procedural operators that are used to interactively inform the system in which the tool operates that it is ready for the next input.

2.3 The Environment of Software Tools

The environment required by a tool depends on how portable the tool is. The list that follows shows the number and percentage of tools [1] that are considered portable, partially portable, or not portable.

<table>
<thead>
<tr>
<th>PORTABLE</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>YES.......</td>
<td>69 (19%)</td>
<td></td>
</tr>
<tr>
<td>NO........</td>
<td>26 (7%)</td>
<td></td>
</tr>
<tr>
<td>PARTIAL...</td>
<td>6 (1%)</td>
<td></td>
</tr>
</tbody>
</table>

A tool is considered portable if (1) it is written in a portable subset of a language, (2) it is written according to a Federal standard for a language, or (3) it is available on three or more different manufacturers' computers of significantly different architecture. A tool is considered partially portable if it is available on different manufacturers' computers of significantly different architecture or minor modifications are required to move it to other machines. Appendix H lists the portable tools.

Another important environmental factor is the language or languages in which a tool is written. The list that follows shows the number and percentage of tools by source language.

<table>
<thead>
<tr>
<th>LANGUAGE</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>FORTRAN</td>
<td>182 (50%)</td>
<td></td>
</tr>
<tr>
<td>COBOL</td>
<td>48 (13%)</td>
<td></td>
</tr>
<tr>
<td>ASSEMBLY</td>
<td>23 (6%)</td>
<td></td>
</tr>
<tr>
<td>PASCAL</td>
<td>18 (4%)</td>
<td></td>
</tr>
<tr>
<td>PL/I</td>
<td>14 (3%)</td>
<td></td>
</tr>
<tr>
<td>STRUCTURED FORTRAN</td>
<td>11 (3%)</td>
<td></td>
</tr>
<tr>
<td>JOVIAL</td>
<td>10 (2%)</td>
<td></td>
</tr>
</tbody>
</table>

[1] This is an example where little information was given about a subject. 76 percent of the tools in the database lack information about portability.
<table>
<thead>
<tr>
<th>Tool</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>BASIC</td>
<td>9</td>
<td>2%</td>
</tr>
<tr>
<td>COMPASS</td>
<td>7</td>
<td>1%</td>
</tr>
<tr>
<td>BAL</td>
<td>6</td>
<td>1%</td>
</tr>
<tr>
<td>C</td>
<td>5</td>
<td>1%</td>
</tr>
<tr>
<td>SLEUTH</td>
<td>4</td>
<td>1%</td>
</tr>
<tr>
<td>RATFOR</td>
<td>3</td>
<td>0%</td>
</tr>
<tr>
<td>LISP</td>
<td>3</td>
<td>0%</td>
</tr>
<tr>
<td>SCOBOL</td>
<td>2</td>
<td>0%</td>
</tr>
<tr>
<td>IFTRAN</td>
<td>2</td>
<td>0%</td>
</tr>
<tr>
<td>OBJECT</td>
<td>2</td>
<td>0%</td>
</tr>
<tr>
<td>RPG</td>
<td>2</td>
<td>0%</td>
</tr>
<tr>
<td>SIMSCRIPT</td>
<td>2</td>
<td>0%</td>
</tr>
<tr>
<td>ALC</td>
<td>2</td>
<td>0%</td>
</tr>
<tr>
<td>SRTRAN</td>
<td>2</td>
<td>0%</td>
</tr>
<tr>
<td>MACRO-11</td>
<td>1</td>
<td>0%</td>
</tr>
<tr>
<td>SMAL/80</td>
<td>1</td>
<td>0%</td>
</tr>
<tr>
<td>PWS</td>
<td>1</td>
<td>0%</td>
</tr>
<tr>
<td>SNOBAL</td>
<td>1</td>
<td>0%</td>
</tr>
<tr>
<td>SALSIM</td>
<td>1</td>
<td>0%</td>
</tr>
<tr>
<td>PLS</td>
<td>1</td>
<td>0%</td>
</tr>
<tr>
<td>SYML</td>
<td>1</td>
<td>0%</td>
</tr>
<tr>
<td>PPFORT</td>
<td>1</td>
<td>0%</td>
</tr>
<tr>
<td>MODULA</td>
<td>1</td>
<td>0%</td>
</tr>
<tr>
<td>SIMULA</td>
<td>1</td>
<td>0%</td>
</tr>
<tr>
<td>KCL</td>
<td>1</td>
<td>0%</td>
</tr>
<tr>
<td>GIM</td>
<td>1</td>
<td>0%</td>
</tr>
</tbody>
</table>

Appendix I lists the tools by source language. Information on the language dialects is provided for each tool in Appendix N. As one would expect, FORTRAN and COBOL are the most popular tool languages.

The hardware requirements of a tool have the largest impact on the environment. Many tools may require the presence of a specific manufacturer's hardware system. Others have simply been developed on a specific system and their availability on other systems has not been tested [1]. The list that follows profiles the hardware systems that are referenced by the tools in the database.

[1] This fact is evident in the low percentages shown previously for portability.
Appendix J lists the tools by these hardware systems.

In addition to hardware, many tools also require the presence of a specific manufacturer's software systems. The list that follows profiles the software systems that are referenced by the tools in the database.

<table>
<thead>
<tr>
<th>Software System</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>OS/VS</td>
<td>28</td>
<td>(7%)</td>
</tr>
<tr>
<td>OS/MVS</td>
<td>25</td>
<td>(6%)</td>
</tr>
<tr>
<td>OS</td>
<td>23</td>
<td>(6%)</td>
</tr>
<tr>
<td>GCOS</td>
<td>13</td>
<td>(3%)</td>
</tr>
<tr>
<td>RSX-11</td>
<td>11</td>
<td>(3%)</td>
</tr>
<tr>
<td>DOS</td>
<td>11</td>
<td>(3%)</td>
</tr>
<tr>
<td>VMS</td>
<td>10</td>
<td>(2%)</td>
</tr>
<tr>
<td>DOS/VS</td>
<td>8</td>
<td>(2%)</td>
</tr>
<tr>
<td>NOS</td>
<td>8</td>
<td>(2%)</td>
</tr>
<tr>
<td>MULTICS</td>
<td>7</td>
<td>(1%)</td>
</tr>
<tr>
<td>ECL</td>
<td>6</td>
<td>(1%)</td>
</tr>
<tr>
<td>UNIX</td>
<td>6</td>
<td>(1%)</td>
</tr>
</tbody>
</table>
SVS .......... 5 ( 1%)
TSO .......... 5 ( 1%)
EXEC 8 ....... 5 ( 1%)
KRONOS ........ 4 ( 1%)
VM/CMS ....... 3 ( 0%)
OS/MVT ....... 3 ( 0%)
RSTS ......... 3 ( 0%)
SIRIS ......... 3 ( 0%)
TOPS-10/20 .... 2 ( 0%)
SCOPE 3.4 .... 2 ( 0%)
RT-11 ......... 1 ( 0%)
IAS ........... 1 ( 0%)
MASTER ........ 1 ( 0%)
SCP ......... 1 ( 0%)
INTERLISP .... 1 ( 0%)
4JS2 ........... 1 ( 0%)
CDC 6600 SECRE .... 1 ( 0%)

Appendix K lists the tools by these software systems.

2.4 Availability of Software Tools

Many tools have restrictions on their outside availability or are for internal use by a company or research organization. These organizations are willing to share information about their tools for professional reasons, but are not necessarily willing to make their tools generally available. The list that follows shows the number and percentage of tools by their availability to potential users [1].

AVAILABLE

YES....... 187 ( 51%)
NO....... 24 ( 6%)

Many tools are developed for non-commercial purposes. These tools, in many cases, are made available to the general public for no cost or at a nominal cost. When this is the case the

[1] Although one might consider it likely that an organization would make known the availability of a tool, the percentages indicate that this may not be the case. When a tool is developed by a non-marketing group of an organization, the decision to make a tool available depends on proprietary concerns, mission reviews, market studies, and personnel availability. These issues put many tools into an unknown category of public availability.
tools are considered to be in the public domain. The most common repositories for public domain tools are the National Technical Information Service (NTIS) Computer Products Support Group [1], the Federal Software Exchange [2], the Computer Software Management and Information Center (COSMIC) [3], and the many computer user groups that are either directly or indirectly supported by computer hardware vendors. The list that follows shows the number and percentage of tools by their availability in the public domain.

PUBLIC DOMAIN
YES....... 57 (15%)
NO....... 116 (32%)

Appendix L lists the public domain tools.

Other information about the availability of tools is also stored in the database. This information includes restrictions on the availability such as copyright, licenses, legal agreements, etc. It also includes information about support for a tool and the current stage of development. This information can be found listed by tool in Appendix N.

2.5 Tool Literature and Sources

Documentation is the most important source of information for users, developers, and maintainers of software tools. The list that follows includes many of the more common and not so common forms of documentation that are referenced by the tools in the database.


In many cases information about documentation was not available in the sources used for the database. However, it is still worthwhile to note that of all the tool descriptions analyzed, only about half mentioned the availability of a user's manual or a user's guide.

The list that follows includes a sampling of the publications that were cited most often in references to articles available in the open literature. These publications often include articles and papers on software tools.

<table>
<thead>
<tr>
<th>Publication</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACM COMPUTING SURVEYS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BELL SYSTEM TECHNICAL JOURNAL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMUNICATIONS OF THE ACM</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMPUTER</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMPUTER DECISIONS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMPUTERWORLD</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CONGRESSO AICA</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DATAMATION</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ELECTRONIC DESIGN</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IEEE COMPUTER SOCIETY PUBLICATIONS</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
There is a great diversity among the developers of tools. Tool developers can be found within governments, industries, universities, and private homes. The list that follows is a profile of the various types of tool developers.

<table>
<thead>
<tr>
<th>Category</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMMERCIAL RESEARCH ORGANIZATIONS</td>
<td>146</td>
<td>40%</td>
</tr>
<tr>
<td>COMMERCIAL VENDORS OF TOOLS</td>
<td>122</td>
<td>33%</td>
</tr>
<tr>
<td>UNIVERSITIES</td>
<td>27</td>
<td>7%</td>
</tr>
<tr>
<td>US GOVERNMENT AGENCIES</td>
<td>26</td>
<td>7%</td>
</tr>
<tr>
<td>US GOVERNMENT SUPPORTED RESEARCH CENTERS</td>
<td>25</td>
<td>6%</td>
</tr>
<tr>
<td>FOREIGN GOVERNMENTS</td>
<td>10</td>
<td>2%</td>
</tr>
<tr>
<td>INDIVIDUALS</td>
<td>5</td>
<td>1%</td>
</tr>
</tbody>
</table>

It can be seen from the list that a vast majority of software development tools are not necessarily marketed commercially. The largest group of non-vendors are commercial research organizations. These organizations have developed software tools in-house primarily for their own use and do not actively market tools. Government supported research centers are non-profit organizations that perform work primarily under Government contract.
The list that follows shows the information sources for the tools that are included in the database.

<table>
<thead>
<tr>
<th>Source</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>TRW SOFTWARE TOOLS CATALOG [ASDS79]</td>
<td>101</td>
<td>(27%)</td>
</tr>
<tr>
<td>AIAA SURVEY OF SOFT DEV TOOLS [AIAA79]</td>
<td>74</td>
<td>(20%)</td>
</tr>
<tr>
<td>COMPLETED SUBMISSION TO NBS</td>
<td>67</td>
<td>(18%)</td>
</tr>
<tr>
<td>NBS/IEEE/ACM TOOL FAIR [Houg81a]</td>
<td>35</td>
<td>(9%)</td>
</tr>
<tr>
<td>PRODUCT DESCRIPTION</td>
<td>26</td>
<td>(7%)</td>
</tr>
<tr>
<td>RADC-TR-80-13, INTERIM REPORT [Dona80]</td>
<td>25</td>
<td>(6%)</td>
</tr>
<tr>
<td>NOSC SEATECS TOOLS SURVEY [Reif81]</td>
<td>24</td>
<td>(6%)</td>
</tr>
<tr>
<td>TECHNICAL LITERATURE</td>
<td>11</td>
<td>(3%)</td>
</tr>
<tr>
<td>FEDERAL SOFT EXCHANGE CATALOG [GSA80]</td>
<td>6</td>
<td>(1%)</td>
</tr>
<tr>
<td>ADI/CNRS CATALOGUE 1980 [Andr80]</td>
<td>5</td>
<td>(1%)</td>
</tr>
<tr>
<td>BCS TOOLS CATALOG [BCS79]</td>
<td>4</td>
<td>(1%)</td>
</tr>
<tr>
<td>NTIS [NTIS80]</td>
<td>3</td>
<td>(0%)</td>
</tr>
</tbody>
</table>

Appendix M lists the tools by the information sources.

Information for the database was solicited through announcements in trade publications, Commerce Business Daily, and NBS publications. The preferred response from the solicitation was a completed paper submitted to NBS by the developer or user of a tool. When this was the case, the information obtained was usually complete and all the possible records for a tool could be filled and stored in the database. However in most cases, information was referred to NBS through various catalogs, surveys, product descriptions, and technical papers. Consequently, there are occasional gaps in the information. The most notable areas are portability, availability, user interface, and documentation.

Finally, people are the most common sources for information about tools. The database maintains records of people who are contacts for information about tools or who have submitted information about tools. For the sake of curiosity, the geographic location of these people was compiled. The list that follows shows the distribution [1] by country and state.

<table>
<thead>
<tr>
<th>State</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>CALIFORNIA</td>
<td>107</td>
<td>(23%)</td>
</tr>
<tr>
<td>NEW YORK</td>
<td>29</td>
<td>(6%)</td>
</tr>
<tr>
<td>MARYLAND</td>
<td>26</td>
<td>(5%)</td>
</tr>
<tr>
<td>VIRGINIA</td>
<td>25</td>
<td>(5%)</td>
</tr>
<tr>
<td>MASSACHUSETTS</td>
<td>24</td>
<td>(5%)</td>
</tr>
</tbody>
</table>

[1] The percentages are based on the total number of people associated with tools, not on the total number of tools.
<table>
<thead>
<tr>
<th>State/Most Likely Foreign Country</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>PENNSYLVANIA</td>
<td>17</td>
<td>( 3%)</td>
</tr>
<tr>
<td>NEW JERSEY</td>
<td>14</td>
<td>( 3%)</td>
</tr>
<tr>
<td>WASHINGTON</td>
<td>13</td>
<td>( 2%)</td>
</tr>
<tr>
<td>DIST OF COLUMBIA</td>
<td>11</td>
<td>( 2%)</td>
</tr>
<tr>
<td>TEXAS</td>
<td>10</td>
<td>( 2%)</td>
</tr>
<tr>
<td>ILLINOIS</td>
<td>10</td>
<td>( 2%)</td>
</tr>
<tr>
<td>MICHIGAN</td>
<td>8</td>
<td>( 1%)</td>
</tr>
<tr>
<td>OHIO</td>
<td>7</td>
<td>( 1%)</td>
</tr>
<tr>
<td>COLORADO</td>
<td>6</td>
<td>( 0%)</td>
</tr>
<tr>
<td>GEORGIA</td>
<td>6</td>
<td>( 1%)</td>
</tr>
<tr>
<td>OREGON</td>
<td>5</td>
<td>( 1%)</td>
</tr>
<tr>
<td>ALABAMA</td>
<td>5</td>
<td>( 1%)</td>
</tr>
<tr>
<td>FLORIDA</td>
<td>4</td>
<td>( 0%)</td>
</tr>
<tr>
<td>KANSAS</td>
<td>3</td>
<td>( 0%)</td>
</tr>
<tr>
<td>MISSOURI</td>
<td>2</td>
<td>( 0%)</td>
</tr>
<tr>
<td>ARIZONA</td>
<td>2</td>
<td>( 0%)</td>
</tr>
<tr>
<td>TENNESSEE</td>
<td>2</td>
<td>( 0%)</td>
</tr>
<tr>
<td>UTAH</td>
<td>2</td>
<td>( 0%)</td>
</tr>
<tr>
<td>NEW HAMPSHIRE</td>
<td>2</td>
<td>( 0%)</td>
</tr>
<tr>
<td>MINNESOTA</td>
<td>2</td>
<td>( 0%)</td>
</tr>
<tr>
<td>WEST VIRGINIA</td>
<td>2</td>
<td>( 0%)</td>
</tr>
<tr>
<td>NORTH CAROLINA</td>
<td>2</td>
<td>( 0%)</td>
</tr>
<tr>
<td>IOWA</td>
<td>1</td>
<td>( 0%)</td>
</tr>
<tr>
<td>CONNECTICUT</td>
<td>1</td>
<td>( 0%)</td>
</tr>
<tr>
<td>IDAHO</td>
<td>1</td>
<td>( 0%)</td>
</tr>
<tr>
<td>NEW MEXICO</td>
<td>1</td>
<td>( 0%)</td>
</tr>
<tr>
<td>OKLAHOMA</td>
<td>1</td>
<td>( 0%)</td>
</tr>
<tr>
<td>WISCONSIN</td>
<td>1</td>
<td>( 0%)</td>
</tr>
<tr>
<td>COUNTRY</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FRANCE</td>
<td>11</td>
<td>( 2%)</td>
</tr>
<tr>
<td>CANADA</td>
<td>8</td>
<td>( 1%)</td>
</tr>
<tr>
<td>WEST GERMANY</td>
<td>7</td>
<td>( 1%)</td>
</tr>
<tr>
<td>JAPAN</td>
<td>4</td>
<td>( 0%)</td>
</tr>
<tr>
<td>ENGLAND</td>
<td>4</td>
<td>( 0%)</td>
</tr>
<tr>
<td>ISRAEL</td>
<td>3</td>
<td>( 0%)</td>
</tr>
<tr>
<td>SWEDEN</td>
<td>3</td>
<td>( 0%)</td>
</tr>
<tr>
<td>NEW ZEALAND</td>
<td>2</td>
<td>( 0%)</td>
</tr>
<tr>
<td>AUSTRIA</td>
<td>1</td>
<td>( 0%)</td>
</tr>
<tr>
<td>CZECHOSLOVAKIA</td>
<td>1</td>
<td>( 0%)</td>
</tr>
<tr>
<td>ITALY</td>
<td>1</td>
<td>( 0%)</td>
</tr>
<tr>
<td>SOUTH AFRICA</td>
<td>1</td>
<td>( 0%)</td>
</tr>
<tr>
<td>MEXICO</td>
<td>1</td>
<td>( 0%)</td>
</tr>
<tr>
<td>NETHERLANDS</td>
<td>1</td>
<td>( 0%)</td>
</tr>
<tr>
<td>NORWAY</td>
<td>1</td>
<td>( 0%)</td>
</tr>
</tbody>
</table>

As one might expect, California turned out with the highest percentage. The names and addresses of the people who are contacts for tools in the database are printed by tool in Appendix N.
3. Conclusions

As noted in Section 1, the main purposes for compiling information on software tools were to obtain information to support NBS efforts and to transfer information to the Federal Government, industry, and research groups. The database has served these purposes. The features of software tools that appeared in an earlier version of the database have aided the development of the taxonomy of tool features reported in Section 2. The taxonomy is slated for further publications and is being used as a basis for several ongoing NBS efforts.

In the area of information transfer, the database helped the General Services Administration locate Government-owned software tools for inclusion in the Federal Software Exchange Catalog. In addition, information from the database was made available to many people in the Federal Government, including many members of the Federal ADP Users Group. The database was demonstrated at the 5th International Conference on Software Engineering and the National Computer Conference. An earlier report on the database has received wide distribution both inside and outside the Government.

Since the purposes for the database have been accomplished, plans are being made for transferring the database from NBS to the Data and Analysis Center for Software (DACS) [Duva80]. DACS was established in August of 1978 to serve as a central source for information and data on software technology by the Rome Air Development Center. The NBS database will form a basis for the DACS Software Tools Database. It is understood that DACS intends to implement the DACS database under the National Software Works (NSW) and to offer a user-oriented software tool search capability.

Acknowledgments. The author wishes to acknowledge the contributions of others in the development of the NBS Software Tools Database. The need for the database was established at the IEEE Testing and Documentation Workshop, Fort Lauderdale, FL, December 1978 which was attended by Dr. Selden Stewart, Dr. W. Richards Adrion, and Dr. Martha Branstad of NBS. During one of the discussion sessions, it was concluded that a public information exchange on software tools was needed and NBS agreed to initiate its development. Drs. Stewart, Adrion, and Branstad have each, at one time or another, supervised the development of the database. During the initiation of the database, Dr. Donald Deutsch, Dr. Leonard Gallagher, and Mr. Charles Sheppard of NBS provided much assistance in the use of relational models for the tool data. Programs for the database were developed in Pascal/R [Schm80] which is an extension of Pascal developed at the University of Hamburg, West Germany. The extension allows the
programmer to specify and use data relations within the Pascal programming language, which in essence, allows one to establish a relational database. Assistance in populating the database and editing the tool information was provided by Ms. Karen Oakley. Programming assistance was provided by Ms. Sheila Frankel and Ms. Karen Oakley. Review and translation of parts of [Andr80] were provided by Ms. Victoria Roy.

4. References


## APPENDIX A

### TOOLS BY GENERAL CLASSIFICATION

**SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE TOOLS (MAC)**

<table>
<thead>
<tr>
<th>ABS</th>
<th>ACT/1</th>
<th>ADS/CERL</th>
<th>AFS</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALIAS</td>
<td>ASA-PMS</td>
<td>ASC</td>
<td>ASEQ</td>
</tr>
<tr>
<td>AUTDOC</td>
<td>AUTOCOM</td>
<td>AUTOFLOW/TRW</td>
<td>AUTOFLOW(TM)*</td>
</tr>
<tr>
<td>AUTOMATIC DOCUM</td>
<td>AUTORETEST</td>
<td>BLKGEN-BDD</td>
<td>BLKGEN/SPECPN</td>
</tr>
<tr>
<td>BUDGET VS ACTUAL</td>
<td>CADMUS</td>
<td>CALLREF</td>
<td>CAPTURE/MVS(TM)**</td>
</tr>
<tr>
<td>CHECKSUM</td>
<td>COMDIM</td>
<td>COMGEN</td>
<td>COMSORT</td>
</tr>
<tr>
<td>COMLIST/TRW</td>
<td>COMLIST</td>
<td>COMPARE</td>
<td>COMSORT</td>
</tr>
<tr>
<td>COMSTAR</td>
<td>CONFIG</td>
<td>CPA</td>
<td>CPAL</td>
</tr>
<tr>
<td>CROREF</td>
<td>CTC</td>
<td>CUE</td>
<td>DA</td>
</tr>
<tr>
<td>DATAMACS</td>
<td>DCD</td>
<td>DECKBOY COMPAR</td>
<td>DEPCHT</td>
</tr>
<tr>
<td>DICTANL/LOCATE</td>
<td>DIFFS (TM)***</td>
<td>DIRCOM</td>
<td>DOCGEN</td>
</tr>
<tr>
<td>DOCU/TEXT</td>
<td>DOCUMENTER</td>
<td>DOCUMENTER A</td>
<td>DOCUMENT</td>
</tr>
<tr>
<td>DOCUMENTOR</td>
<td>DOSSIER</td>
<td>DPNDCY</td>
<td>EASYTROL</td>
</tr>
<tr>
<td>ESAP</td>
<td>FLOBOL</td>
<td>FLODIA</td>
<td>FLOWGEN</td>
</tr>
<tr>
<td>FORMAN</td>
<td>FORREF</td>
<td>FORTREF</td>
<td>FTNCODER</td>
</tr>
<tr>
<td>GADTR AID</td>
<td>GIM/GIM II</td>
<td>GIRAFF</td>
<td>HARP</td>
</tr>
<tr>
<td>INFORM/REFORM</td>
<td>INSERT</td>
<td>ISUS</td>
<td>JET</td>
</tr>
<tr>
<td>JSDD</td>
<td>LANG INSTRUCTOR</td>
<td>LAYOUT</td>
<td>LEXICON</td>
</tr>
<tr>
<td>LIBARIAN</td>
<td>LIBREF</td>
<td>LOGIFLOW</td>
<td>LOGOS</td>
</tr>
<tr>
<td>LOOK</td>
<td>MEDL-X</td>
<td>MEMORY MNG LIB</td>
<td>MPS</td>
</tr>
<tr>
<td>N5500</td>
<td>N-SQUARED</td>
<td>NUMBER/DEC</td>
<td>ONLINE ASSIST</td>
</tr>
<tr>
<td>PAC II</td>
<td>PDS FLOW</td>
<td>PDSS</td>
<td>PPORT</td>
</tr>
<tr>
<td>PFS</td>
<td>PMCS</td>
<td>PMS IV</td>
<td>PPP</td>
</tr>
<tr>
<td>PROG COMP ANAL</td>
<td>PRONET</td>
<td>PSL</td>
<td>QCM</td>
</tr>
<tr>
<td>QCRT</td>
<td>QUICK-DRAW</td>
<td>REFER</td>
<td>RENAME</td>
</tr>
<tr>
<td>SDP</td>
<td>SLIB</td>
<td>SLIM</td>
<td>SMS</td>
</tr>
<tr>
<td>SMT</td>
<td>SNOOP</td>
<td>SPC</td>
<td>SPEAR</td>
</tr>
<tr>
<td>SPECTRUM-1</td>
<td>SPELL</td>
<td>SPREAD</td>
<td>SPRINT</td>
</tr>
<tr>
<td>TAPS/AM</td>
<td>TDBCOMP</td>
<td>TIDY</td>
<td>TOOLS DATABASE</td>
</tr>
</tbody>
</table>

* AUTOFLOW IS A REGISTERED TRADEMARK OF APPLIED DATA RESEARCH, INC.
** CAPTURE/MVS IS A REGISTERED TRADEMARK OF BGS SYSTEMS, INC.
*** DIFFS (TM) IS A REGISTERED TRADEMARK OF SOFTWARE CONSULTING SERVICES
## Tools by General Classification

### Software Modeling and Simulation Tools (SAM)

<table>
<thead>
<tr>
<th>Tool</th>
<th>Tool</th>
<th>Tool</th>
<th>Tool</th>
</tr>
</thead>
<tbody>
<tr>
<td>AISIM</td>
<td>ASRP</td>
<td>BEST/1 (TM)*</td>
<td>CRYSTAL (TM)*</td>
</tr>
<tr>
<td>DAS</td>
<td>DDPM</td>
<td>DPAD</td>
<td>HARDWARE SIMULA</td>
</tr>
<tr>
<td>MEDL-P</td>
<td>POD</td>
<td>SALSIM</td>
<td>SCERT</td>
</tr>
<tr>
<td>SDVS</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Requirements/Design Specification and Analysis Tools (RAD)

<table>
<thead>
<tr>
<th>Tool</th>
<th>Tool</th>
<th>Tool</th>
<th>Tool</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADF</td>
<td>AFFIRM</td>
<td>ARTS</td>
<td>AUTO-DBO</td>
</tr>
<tr>
<td>AUTOIDEF0</td>
<td>CADSAT</td>
<td>CARA</td>
<td>CBLSHORT</td>
</tr>
<tr>
<td>COBOL/SP</td>
<td>CONFIGURATOR</td>
<td>CRISPFLOW</td>
<td>CS4</td>
</tr>
<tr>
<td>DARTS</td>
<td>DATA DESIGNER</td>
<td>DECA</td>
<td>DQM</td>
</tr>
<tr>
<td>FAME</td>
<td>FOSTRA</td>
<td>IORL</td>
<td>IPDS</td>
</tr>
<tr>
<td>ISDS</td>
<td>LOGICFLOW</td>
<td>MED-SYS</td>
<td>MEDL-R</td>
</tr>
<tr>
<td>MEDL-D</td>
<td>MSL</td>
<td>MTR</td>
<td>NETWORK PLANNER</td>
</tr>
<tr>
<td>PBASIC</td>
<td>PDL</td>
<td>PDS</td>
<td>PERCAM</td>
</tr>
<tr>
<td>PIDGIN-FASP</td>
<td>PSL/PSA</td>
<td>RA</td>
<td>RTT</td>
</tr>
<tr>
<td>SARA</td>
<td>SCG</td>
<td>SCG/DQM</td>
<td>SCHEMACODE</td>
</tr>
<tr>
<td>SCOPE</td>
<td>SDDL</td>
<td>SDL</td>
<td>SDP/MAYDA</td>
</tr>
<tr>
<td>SIGS</td>
<td>SPECLE/DARS</td>
<td>SREM</td>
<td>SREP</td>
</tr>
<tr>
<td>SRIMP</td>
<td>STAG/TEMS</td>
<td>STRUCTURE(S)</td>
<td>SYDIM</td>
</tr>
<tr>
<td>TRANSFOR</td>
<td>XAS8</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Source Program Analysis and Testing Tools (TAA)

<table>
<thead>
<tr>
<th>Tool</th>
<th>Tool</th>
<th>Tool</th>
<th>Tool</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADS</td>
<td>AMPIC</td>
<td>ASSIST-I</td>
<td>ATA-FASP</td>
</tr>
<tr>
<td>ATA-SAI</td>
<td>ATDG</td>
<td>ATTEST</td>
<td>AUDIT</td>
</tr>
<tr>
<td>AUDITOR</td>
<td>BSC</td>
<td>CA</td>
<td>CADA</td>
</tr>
<tr>
<td>CASEGEN</td>
<td>CAVS</td>
<td>CCA</td>
<td>CCREF</td>
</tr>
<tr>
<td>CCS</td>
<td>CENSUS</td>
<td>CGJA</td>
<td>CICS DUMP ANALY</td>
</tr>
<tr>
<td>COBOL/DV</td>
<td>COBOL STRUCT</td>
<td>COBOL TRACING</td>
<td>COBOL/QDM</td>
</tr>
<tr>
<td>COBOL OPTIMIZ</td>
<td>COBOL TESTING</td>
<td>COBOL/CP</td>
<td>COMMAP</td>
</tr>
<tr>
<td>COMSCAN</td>
<td>CORE</td>
<td>COTUNE II</td>
<td>CPA-ADR</td>
</tr>
<tr>
<td>CQD</td>
<td>DAVE</td>
<td>DRIVER</td>
<td>DYNAX</td>
</tr>
<tr>
<td>EAVS</td>
<td>ECA AUTOMATION</td>
<td>EFFIGY</td>
<td>ENFORCE</td>
</tr>
<tr>
<td>EVP</td>
<td>EXPEDITER</td>
<td>FACES</td>
<td>FADEBUG-I</td>
</tr>
<tr>
<td>FAST</td>
<td>FAVS</td>
<td>FCA</td>
<td>FORAN</td>
</tr>
<tr>
<td>FORTRAN TRACING</td>
<td>FORTRAN TESTING</td>
<td>FORTRAN OPTIMIZ</td>
<td>FTN-77 ANALYZER</td>
</tr>
<tr>
<td>FTN ANALYZER</td>
<td>FTNXREF</td>
<td>GENTESTS</td>
<td>GENTEXTS</td>
</tr>
<tr>
<td>GOTO-ANALYZER</td>
<td>HAWKEYE (TM)**</td>
<td>INSTRU</td>
<td>INTERFACE DOCUM</td>
</tr>
<tr>
<td>ITB</td>
<td>JAVS</td>
<td>JIGSAW</td>
<td>JOVIAL TCA</td>
</tr>
<tr>
<td>JOVIAL/J3SC</td>
<td>JOVIAL/VS</td>
<td>JOYCE</td>
<td>LOGIC</td>
</tr>
<tr>
<td>MENTOR</td>
<td>MONITOR</td>
<td>NASA-VATS</td>
<td>NODAL</td>
</tr>
<tr>
<td>NUMBER</td>
<td>OPTIMUS</td>
<td>OPTIMIZER II</td>
<td>OSCYBR</td>
</tr>
<tr>
<td>PACE</td>
<td>PACE-C</td>
<td>PET</td>
<td>PPE</td>
</tr>
<tr>
<td>PREF HDR GEN</td>
<td>PROGLOOK</td>
<td>RADC/FCA</td>
<td>REALIGNMENT SYS</td>
</tr>
<tr>
<td>REFLECT II</td>
<td>REFORM</td>
<td>REFTRAN (TM)***</td>
<td>REL MEAS MODEL</td>
</tr>
<tr>
<td>RISOS TOOLS</td>
<td>RXVP80 (TM)***</td>
<td>SADAT</td>
<td>SAP</td>
</tr>
</tbody>
</table>

* BEST/1 AND CRYSTAL ARE REGISTERED TRADEMARKS OF BGS SYSTEMS, INC.
** HAWKEYE IS A REGISTERED TRADEMARK OF BLACKHAWK DATA CORP.
*** REFTRAN AND RXVP80 ARE REGISTERED TRADEMARKS OF GENERAL RESEARCH CORP.
TOOLS BY GENERAL CLASSIFICATION

<table>
<thead>
<tr>
<th>Software Support System/Programming Environment Tools (Env)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ADA ENVIRONMENT</strong></td>
</tr>
<tr>
<td><strong>FASP</strong></td>
</tr>
<tr>
<td><strong>SEF</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Program Construction and Generation Tools (Gen)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ADA-ATOM</strong></td>
</tr>
<tr>
<td><strong>COGENT</strong></td>
</tr>
<tr>
<td><strong>FOCUS</strong></td>
</tr>
<tr>
<td><strong>JOCIT</strong></td>
</tr>
<tr>
<td><strong>METRAN</strong></td>
</tr>
<tr>
<td><strong>QUIKCODE</strong></td>
</tr>
<tr>
<td><strong>SCOBOL (TM)</strong>*****</td>
</tr>
<tr>
<td><strong>SMMA</strong></td>
</tr>
<tr>
<td><strong>SURGE 72</strong></td>
</tr>
</tbody>
</table>

* SOFTOOL 80 IS A REGISTERED TRADEMARK OF SOFTOOL CORP.
** COPE IS A REGISTERED TRADEMARK OF SOFTWARE CONSULTING SERVICES
*** IFTRAN IS A REGISTERED TRADEMARK OF GENERAL RESEARCH CORP.
**** MARK IV IS A REGISTERED TRADEMARK OF INFORMATICS, INC.
***** SCOBOL IS A REGISTERED TRADEMARK OF SOFTWARE CONSULTING SERVICES
### APPENDIX B

#### TOOLS BY INPUT SUBJECT

<table>
<thead>
<tr>
<th><strong>. . TEXT INPUT</strong></th>
<th><strong>ACT/1</strong></th>
<th><strong>ADF</strong></th>
<th>ARGUS/MICRO</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>FASP</td>
<td>IORL</td>
</tr>
<tr>
<td></td>
<td></td>
<td>LILITH</td>
<td>MEDL-D</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MEDL-X</td>
<td>MPS</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PDL</td>
<td>PDS</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SCHEMACODE</td>
<td>SLIB</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>. . DATA INPUT</strong></th>
<th><strong>ABS</strong></th>
<th><strong>ACT/1</strong></th>
<th>ASC</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>BUDGET VS ACTUA</td>
<td>CADMUS</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CHECKSUM</td>
<td>COMPARE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CS4</td>
<td>DA</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DI-3000</td>
<td>DICTANL/LOCATE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DOCUMENTER</td>
<td>DOCUMENT</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DPAD</td>
<td>DRIVER</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ESAP</td>
<td>FASP</td>
</tr>
<tr>
<td></td>
<td></td>
<td>FTNCODER</td>
<td>GADTR AID</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GRAFMAKER</td>
<td>HARP</td>
</tr>
<tr>
<td></td>
<td></td>
<td>LAYOUT</td>
<td>LOOK</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MENTOR</td>
<td>N5500</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PAC II</td>
<td>PDSS</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PFS</td>
<td>PMCS</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PPP</td>
<td>PRONET</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PWB FOR VAX/VMS</td>
<td>QCM</td>
</tr>
<tr>
<td></td>
<td></td>
<td>RATCODER</td>
<td>REL MEAS</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SARA-H</td>
<td>SARA-IV</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SARA-U</td>
<td>SCERT</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SMT</td>
<td>SPECTRUM-1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SYDIM</td>
<td>SYSXREF</td>
</tr>
<tr>
<td></td>
<td></td>
<td>TIMECS</td>
<td>TOOLPACK</td>
</tr>
<tr>
<td></td>
<td></td>
<td>TRMS</td>
<td>TRWPLT</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>. . CODE INPUT</strong></th>
<th><strong>ADA-ATOM</strong></th>
<th><strong>ADA COMPILER</strong></th>
<th>ADA ENVIRONMENT</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>ADS/CERL</td>
<td>AFFIRM</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ALIAS</td>
<td>AMPIC</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ASA-PMS</td>
<td>ASEQ</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ASSET</td>
<td>ASSIST-I</td>
</tr>
</tbody>
</table>

|                   |             | ADA-ATOM        | ADS             |
|                   |             | ADS/CERL        | AFFIRM          |
|                   |             | ALIAS           | AMPIC           |
|                   |             | ASA-PMS         | ASEQ            |
|                   |             | ASSET           | ASSIST-I        |

|                   |             | ADA-ATOM        | ADS             |
|                   |             | ADS/CERL        | AFFIRM          |
|                   |             | ALIAS           | AMPIC           |
|                   |             | ASA-PMS         | ASEQ            |
|                   |             | ASSET           | ASSIST-I        |
ATA-SAI
AUDIT
AUTOCOM
AUTOMATIC DOCUM
BLKGEN-BDD
CADA
CASEGEN
CCA
CENSUS
CICS DUMP ANALY
COBOL/QDM
COBOL TESTING
COBOL TRACING
COMDIM
COMLIST/TRW
COMSCAN
CORE
CPA-ADR
CROREF
CTC
DATAMACS
DDPM
DOCGEN
DPNDCY
EAVS
ENFORCE
EXPEDITER
FASP
FCA
FLOWGEN
FORREF
FORTREF
FTN-77 ANALYZER
FTNXREF
GOTO-ANALYZER
IFTRAN (TM)
INSTRU
ISUS
JET
JOVIAL/VS
JOYCE
LIBRARIAN
LOGICFLOW
LOGOS
MEMORY MNG LIB
MONITOR
NODAL
OPTIMIZER II
PACE
PDS
PFORT
PROG COMP ANAL
PWB FOR VAX/VMS
RADC/FCA
REFER
ATDG
AUDITOR
AUTOFLOW/TRW
AUTORETEST
BSC
CADMUS
CAVS
CCREF
CGJA
COBOL STRUCT
COBOL/DV
COBOL OPTIMIZ
COBOL/SP
COMGEN/TRW
COMLIST
COMSORT
COTUNE II
CPAL
CS4
CUE
DAVE
DEPCHT
DOCUMENTOR
DRIVER
ECA AUTOMATION
ESAP
FACES
FAST
FLOBOL
FORAN
FORTRAN OPTIMIZ
FORTRAN TESTING
FTN ANALYZER
GENTESTS
HARDWARE SIMULA
INFORM/REFORM
INTERFACE DOCUM
ITB
JIGSAW
JOVIAL/J3SC
JSDD
LIBREF
LOGIC
MAGLE
MENTOR
MSEF
NUMBER
OPTIMUS
PACE-C
PDS FLOW
PPE
PROGLOOK
QUICK-DRAW
RATFOR
REFLECT II
ATTEST
AUTDOC
AUTOFLOW (TM)
BLKGEN/SPECPN
CA
CALLREF
CBLSHORT
CCS
CHILL TRANS
COBOL/ADE
COBOL/SPP
COBOL/CP
COGENT
COMGEN
COMMAP
CONFIG
CPA
CQD
CSPP
DAS
DCD
DIRCOM
DOCUMENTER A
DYNA
EPPFIGY
EVP
FADEBUG-I
FAVS
FLODIA
FORMAN
FORTRAN TRACING
FORTRAN TESTING
FOSTRA
FTNCODER
GIRAFFE
HAWKEYE (TM)
INSERT
ISDS
JAVA
JOCIT
JOVIAL TCA
LANG INSTRUCTOR
LILITH
LOGIFLOW
MEFIA
MODULE ORDERER
NASA-VATS
NUMBER/DEC
OSCYBR
PBASE
PREF HDR GEN
PSL
QUIKCODE
REALIGNMENT SYS
REFORM
<table>
<thead>
<tr>
<th>Tools</th>
<th>Version / Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>REFTRAN (TM)</td>
<td>RENAME</td>
</tr>
<tr>
<td>RXVP80 (TM)</td>
<td>S-FORTTRAN</td>
</tr>
<tr>
<td>SALSIM</td>
<td>SAP</td>
</tr>
<tr>
<td>SCAN/370</td>
<td>SCG/DQM</td>
</tr>
<tr>
<td>SCOPE</td>
<td>SDP</td>
</tr>
<tr>
<td>SEF</td>
<td>SELECT</td>
</tr>
<tr>
<td>SFTRAN3</td>
<td>SLIB</td>
</tr>
<tr>
<td>SMMA</td>
<td>SMS</td>
</tr>
<tr>
<td>SPC</td>
<td>SPEAR</td>
</tr>
<tr>
<td>SPRINT</td>
<td>SPTRAN</td>
</tr>
<tr>
<td>SSA</td>
<td>STAT ENT &amp; EVAL</td>
</tr>
<tr>
<td>STRUCTURING ENG</td>
<td>STRUC1/STRUC2</td>
</tr>
<tr>
<td>SURVAYOR</td>
<td>SUS</td>
</tr>
<tr>
<td>SYMCRS</td>
<td>SYSTEM MONITOR</td>
</tr>
<tr>
<td>TATTLE</td>
<td>TCAT</td>
</tr>
<tr>
<td>TDEM</td>
<td>TEST PREDICTOR</td>
</tr>
<tr>
<td>TFA</td>
<td>THE ENGINE</td>
</tr>
<tr>
<td>TIMER</td>
<td>TOOLPACK</td>
</tr>
<tr>
<td>TRAILBLAZER</td>
<td>TRANSFOR</td>
</tr>
<tr>
<td>UCA</td>
<td>UCSD P-SYSTEM</td>
</tr>
<tr>
<td>VIRTUAL OS</td>
<td>X TO Y TRANS PK</td>
</tr>
<tr>
<td>XPEDITER OS</td>
<td>YACC</td>
</tr>
<tr>
<td>.. BASIC ..................</td>
<td>BSC</td>
</tr>
<tr>
<td>DOCUMENTER</td>
<td>HARDWARE SIMULA</td>
</tr>
<tr>
<td>STAT ENT &amp; EVAL</td>
<td>PBASIC</td>
</tr>
<tr>
<td>.. FORTRAN ...............</td>
<td>ADS/CERL</td>
</tr>
<tr>
<td>ALIAS</td>
<td>AMPIC</td>
</tr>
<tr>
<td>ASEQUE</td>
<td>ATA-SAI</td>
</tr>
<tr>
<td>ATTEST</td>
<td>AUDITOR</td>
</tr>
<tr>
<td>AUTOQ</td>
<td>AUTOCOM</td>
</tr>
<tr>
<td>AUTORETEST</td>
<td>BLKGEN/SPECPN</td>
</tr>
<tr>
<td>CA</td>
<td>CASEGEN</td>
</tr>
<tr>
<td>CENSUS</td>
<td>COMDIM</td>
</tr>
<tr>
<td>COMGEN/TRW</td>
<td>COMLIST/TRW</td>
</tr>
<tr>
<td>COMMAP</td>
<td>COMSCAN</td>
</tr>
<tr>
<td>CONFIG</td>
<td>CPAL</td>
</tr>
<tr>
<td>DAVE</td>
<td>DEPCHT</td>
</tr>
<tr>
<td>DOCGEN</td>
<td>DOCUMENTOR</td>
</tr>
<tr>
<td>DRIVER</td>
<td>DYNA</td>
</tr>
<tr>
<td>ECA AUTOMATION</td>
<td>EVP</td>
</tr>
<tr>
<td>FAST</td>
<td>FAVS</td>
</tr>
<tr>
<td>FLODIA</td>
<td>FLOWGEN</td>
</tr>
<tr>
<td>FORMAN</td>
<td>FORREF</td>
</tr>
<tr>
<td>FORTTRAN OPTIMIZ</td>
<td>FORTRAN TESTING</td>
</tr>
<tr>
<td>FOSTRA</td>
<td>FTN-77 ANALYZER</td>
</tr>
<tr>
<td>FTCODE</td>
<td>FTNXREF</td>
</tr>
<tr>
<td>GOTO-ANALYZER</td>
<td>IFTRAN (TM)</td>
</tr>
<tr>
<td>INSERT</td>
<td>INSTR</td>
</tr>
<tr>
<td>ISUS</td>
<td>ITB</td>
</tr>
<tr>
<td>LANG INSTRUCTOR</td>
<td>LOGICFLOW</td>
</tr>
<tr>
<td>MEMORY MNG LIB</td>
<td>MONITOR</td>
</tr>
<tr>
<td>NUMBER</td>
<td>OSCYBR</td>
</tr>
<tr>
<td>PDS</td>
<td>PET</td>
</tr>
<tr>
<td>PDS</td>
<td>PFORT</td>
</tr>
<tr>
<td>Subject</td>
<td>Tools</td>
</tr>
<tr>
<td>---------------------</td>
<td>------------------------</td>
</tr>
<tr>
<td>PREF HDR GEN</td>
<td>PSL</td>
</tr>
<tr>
<td>RADC/FCA</td>
<td>REFORM</td>
</tr>
<tr>
<td>RENAME</td>
<td>RXVP80 (TM)</td>
</tr>
<tr>
<td>SALSIM</td>
<td>SAP</td>
</tr>
<tr>
<td>SPELL</td>
<td>STRUCTURING ENG</td>
</tr>
<tr>
<td>SUBCRS</td>
<td>SURVAYOR</td>
</tr>
<tr>
<td>SYMCRS</td>
<td>TDEM</td>
</tr>
<tr>
<td>TIDY</td>
<td>TOOLPACK</td>
</tr>
<tr>
<td>TRAILBLAZER</td>
<td>UCA</td>
</tr>
<tr>
<td>X TO Y TRANS PK</td>
<td></td>
</tr>
<tr>
<td>CICS</td>
<td>ADS</td>
</tr>
<tr>
<td>CICS DUMP ANALY</td>
<td>CORE</td>
</tr>
<tr>
<td>COBOL</td>
<td>AUTOFLOW (TM)</td>
</tr>
<tr>
<td>CADA</td>
<td>CAVS</td>
</tr>
<tr>
<td>COBOL TRACING</td>
<td>COBOL OPTIMIZ</td>
</tr>
<tr>
<td>COBOL STRUCT</td>
<td>COBOL/DV</td>
</tr>
<tr>
<td>COBOL/QDM</td>
<td>COBOL/SPP</td>
</tr>
<tr>
<td>COTUNE II</td>
<td>CPA</td>
</tr>
<tr>
<td>CQD</td>
<td>CSPP</td>
</tr>
<tr>
<td>DATAMACS</td>
<td>DCD</td>
</tr>
<tr>
<td>EXPEDITER</td>
<td>FLOBOL</td>
</tr>
<tr>
<td>LANG INSTRUCTOR</td>
<td>LIBREF</td>
</tr>
<tr>
<td>OPTIMIZER II</td>
<td>PSL</td>
</tr>
<tr>
<td>QUIKCODE</td>
<td>REALIGNMENT SYS</td>
</tr>
<tr>
<td>SCOBOL (TM)</td>
<td>SYDOC</td>
</tr>
<tr>
<td>TFA</td>
<td>THE ENGINE</td>
</tr>
<tr>
<td>XPEDITER</td>
<td></td>
</tr>
<tr>
<td>PASCAL</td>
<td>ARGUS/MICRO</td>
</tr>
<tr>
<td>UCSD P-SYSTEM</td>
<td></td>
</tr>
<tr>
<td>PCL</td>
<td>CA</td>
</tr>
<tr>
<td>COMPASS</td>
<td>CCA</td>
</tr>
<tr>
<td>PACE-C</td>
<td></td>
</tr>
<tr>
<td>PL/1</td>
<td>EFFIGY</td>
</tr>
<tr>
<td>QUICK-DRAW</td>
<td>TRAILBLAZER</td>
</tr>
<tr>
<td>JOVIAL</td>
<td>CGJA</td>
</tr>
<tr>
<td>EAVS</td>
<td>ISDS</td>
</tr>
<tr>
<td>JET</td>
<td>JIGSAW</td>
</tr>
<tr>
<td>JOVIAL/J3SC</td>
<td>JOVIAL TCA</td>
</tr>
<tr>
<td>JSDD</td>
<td>PSL</td>
</tr>
<tr>
<td>TAFIRM</td>
<td>TATTLE</td>
</tr>
<tr>
<td>IFTRAN</td>
<td>IFTRAN (TM)</td>
</tr>
<tr>
<td>RXVP80 (TM)</td>
<td>TEVERE-1</td>
</tr>
<tr>
<td>STRUCTURED FORTRAN</td>
<td>LOGOS</td>
</tr>
<tr>
<td>PDS</td>
<td>PDS FLOW</td>
</tr>
<tr>
<td>S-FORTTRAN</td>
<td>SFORT-1</td>
</tr>
<tr>
<td>SPTRAN</td>
<td>SRTRAN.BASELINE</td>
</tr>
</tbody>
</table>
TOOLS BY INPUT SUBJECT

. . . BAL ................................................ LOGICFLOW
. . . SMAL/80 ............................................ SMAL/80
. . . RATFOR ............................................. RATFOR
VIRTUAL OS
. . . LISP ................................................ LANG INSTRUCTOR
SELECT
. . . APL ................................................ LANG INSTRUCTOR
. . . C ..................................................... MSEF
YACC
. . . JCVS ................................................ JOVIAL/VS
. . . HAL/S ............................................... NASA-VATS
. . . DMATRAN .......................................... FAVS
STRUCl/STRUCl2
. . . CHILL ............................................... CHILL TRANS
. . . ALGOL ............................................... LANG INSTRUCTOR
. . . SNOBOL .............................................. LANG INSTRUCTOR
. . . ADA ................................................ ADA-ATOM
ADA COMPILER ADA ENVIRONMENT
. . . OBJECT CODE INPUT ................................ CALLREF
INTERFACE DOCUM MODULE ORDERER PROG COMP ANAL
. . . ASSEMBLY LANGUAGE ................................ AMPIC
ASSIST-I CONFIG LOGICFLOW
MAGLE QUICK-DRAW RISOS TOOLS
XAS8
. . . MEMORY DUMP ....................................... CICS DUMP ANAL
CORE
. . . SRTRAN .............................................. ISUS
ITB SRTRAN.BASELINE
. . . CSL .................................................. CS4
. . . MODULA .............................................. LILITH
. . . MEFIA ............................................... MEFIA
. . . SCOBOL .............................................. SCOBOL (TM)
. . . CMS-2 ............................................... ATA-FASP
<table>
<thead>
<tr>
<th>VHLL INPUT</th>
<th>ADF</th>
</tr>
</thead>
<tbody>
<tr>
<td>AFFIRM</td>
<td>AISIM</td>
</tr>
<tr>
<td>AUTO-DBO</td>
<td>AUTOIDEF0</td>
</tr>
<tr>
<td>CARA</td>
<td>COGENT</td>
</tr>
<tr>
<td>COPE (TM)</td>
<td>CRISPFLOW</td>
</tr>
<tr>
<td>DARTS</td>
<td>DAS</td>
</tr>
<tr>
<td>DDPM</td>
<td>DECA</td>
</tr>
<tr>
<td>FAME</td>
<td>FOCUS</td>
</tr>
<tr>
<td>IFITRAN (TM)</td>
<td>INFORM</td>
</tr>
<tr>
<td>IPDS</td>
<td>ISDS</td>
</tr>
<tr>
<td>JOCIT</td>
<td>LOGICFLOW</td>
</tr>
<tr>
<td>MED-SYS</td>
<td>MEDL-P</td>
</tr>
<tr>
<td>MEDL-R</td>
<td>METTRAN</td>
</tr>
<tr>
<td>MTR</td>
<td>NETWORK PLANNER</td>
</tr>
<tr>
<td>PDS</td>
<td>PERLUETTE</td>
</tr>
<tr>
<td>POD</td>
<td>PROGRAM GENERAT</td>
</tr>
<tr>
<td>RA</td>
<td>RATFOR</td>
</tr>
<tr>
<td>S-FORTRAN</td>
<td>SALSIM</td>
</tr>
<tr>
<td>SCG</td>
<td>SCG/DQM</td>
</tr>
<tr>
<td>SDDL</td>
<td>SDL</td>
</tr>
<tr>
<td>SFORT-1</td>
<td>SFTRAN3</td>
</tr>
<tr>
<td>SNOOP</td>
<td>SOFTOOL 80 (TM)</td>
</tr>
<tr>
<td>SREM</td>
<td>SREP</td>
</tr>
<tr>
<td>STAG/TEMS</td>
<td>STRUCTURIZER</td>
</tr>
<tr>
<td>STRUCTURE(S)</td>
<td>SURGE 72</td>
</tr>
<tr>
<td>TAB40</td>
<td>TRANSFOR</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>PROGRAM SPECIFICATION</th>
<th>COPE (TM)</th>
</tr>
</thead>
<tbody>
<tr>
<td>PROGRAM GENERAT</td>
<td>STRUCTURIZER</td>
</tr>
<tr>
<td>SYSTEM-80</td>
<td>SURGE 72</td>
</tr>
<tr>
<td>TAB40</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>REQUIREMENTS SPECIFICATION</th>
<th>ARTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUTOIDEF0</td>
<td>CARA</td>
</tr>
<tr>
<td>DAS</td>
<td>DARTS</td>
</tr>
<tr>
<td>DDPM</td>
<td>DDPM</td>
</tr>
<tr>
<td>NETWORK PLANNER</td>
<td>RA</td>
</tr>
<tr>
<td>SIGS</td>
<td>SREP</td>
</tr>
<tr>
<td>SREM</td>
<td>SREP</td>
</tr>
<tr>
<td>STAG/TEMS</td>
<td>STAG/TEMS</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>DESIGN SPECIFICATION</th>
<th>ADF</th>
</tr>
</thead>
<tbody>
<tr>
<td>AISIM</td>
<td>AUTO-DBO</td>
</tr>
<tr>
<td>DARTS</td>
<td>DAS</td>
</tr>
<tr>
<td>DDPM</td>
<td>DECA</td>
</tr>
<tr>
<td>IPDS</td>
<td>ISDS</td>
</tr>
<tr>
<td>MSL</td>
<td>PDL</td>
</tr>
<tr>
<td>SCG</td>
<td>SCG/DQM</td>
</tr>
<tr>
<td>SIGS</td>
<td>SNOOP</td>
</tr>
<tr>
<td>STRUCTURE(S)</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SYSTEM SPECIFICATION</th>
<th>CONFIGURATOR</th>
</tr>
</thead>
<tbody>
<tr>
<td>CRYSTAL (TM)</td>
<td>MEDL-P</td>
</tr>
<tr>
<td>POD</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>REQUIREMENTS LANGUAGE</th>
<th>CADSAT</th>
</tr>
</thead>
<tbody>
<tr>
<td>CARA</td>
<td>IORL</td>
</tr>
<tr>
<td>PSL/PSA</td>
<td>SIGS</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>STRUCTURED LANGUAGE</th>
<th>JIGSAW</th>
</tr>
</thead>
</table>
TOOLS BY INPUT SUBJECT

<table>
<thead>
<tr>
<th>RATFOR</th>
<th>S-FORTAN</th>
<th>SFORT-1</th>
</tr>
</thead>
<tbody>
<tr>
<td>SFTRAN3</td>
<td>STRUC1/STRUC2</td>
<td>TRANSFOR</td>
</tr>
</tbody>
</table>

. . . DESIGN LANGUAGE..................... LOGICFLOW
  MED-SYS MTR PDL
  PDS

. . . SPECIFICATION LANGUAGE............. SALSIM

. . . DESCRIPTION LANGUAGE.............. FOCUS
  GENTEXTS INFORM JOCIT
  MARK IV (TM) METRAN PERLUETTE
  YACC

. . . ALGEBRAIC SPECIFICATIONS........... AFFIRM

. . . MODEL SPECIFICATION............... SRIMP
### APPENDIX C

#### TOOLS BY TRANSFORMATION FEATURES

<table>
<thead>
<tr>
<th>Translation</th>
<th>Conversion</th>
<th>Macro Expansion</th>
<th>Structure Preprocessing</th>
<th>Compilation</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADA COMPILER</td>
<td>CTC</td>
<td>CQD</td>
<td>COBOL/CSPP</td>
<td>ADA COMPILER</td>
</tr>
<tr>
<td>ARGUS/MICRO</td>
<td>MAGLE</td>
<td>MSEF</td>
<td>SMAL/80</td>
<td>ARGUS/MICRO</td>
</tr>
<tr>
<td>AUTO-DBO</td>
<td>MEFIA</td>
<td>QUIKCODE</td>
<td>SRC8/80</td>
<td>AUTO-DBO</td>
</tr>
<tr>
<td>CHILL TRANS</td>
<td>OOSYBR</td>
<td>X TO Y TRANS PK</td>
<td>XAS8</td>
<td>CHILL TRANS</td>
</tr>
<tr>
<td>COBOL/CP</td>
<td>SRIMP</td>
<td>QUIKCODE</td>
<td>X TO Y TRANS PK</td>
<td>COBOL/CP</td>
</tr>
<tr>
<td>CQD</td>
<td>SMMA</td>
<td>SOFTOOL 80 (TM)</td>
<td>X TO Y TRANS PK</td>
<td>CQD</td>
</tr>
<tr>
<td>CTC</td>
<td>SRTRAN3</td>
<td>X TO Y TRANS PK</td>
<td>X TO Y TRANS PK</td>
<td>CTC</td>
</tr>
<tr>
<td>FAVS</td>
<td>SPTRAN</td>
<td>X TO Y TRANS PK</td>
<td>X TO Y TRANS PK</td>
<td>FAVS</td>
</tr>
<tr>
<td>INSERT</td>
<td>S-FORTRAN</td>
<td>S-FORTRAN</td>
<td>S-FORTRAN (TM)</td>
<td>INSERT</td>
</tr>
<tr>
<td>JIGSAW</td>
<td>TRANS</td>
<td>S-FORTRAN</td>
<td>S-FORTRAN (TM)</td>
<td>JIGSAW</td>
</tr>
<tr>
<td>LILITH</td>
<td>TRANS</td>
<td>S-FORTRAN</td>
<td>S-FORTRAN (TM)</td>
<td>LILITH</td>
</tr>
<tr>
<td>MEFIA</td>
<td>TRANS</td>
<td>S-FORTRAN</td>
<td>S-FORTRAN (TM)</td>
<td>MEFIA</td>
</tr>
<tr>
<td>RATFOR</td>
<td>TRANS</td>
<td>S-FORTRAN</td>
<td>S-FORTRAN (TM)</td>
<td>RATFOR</td>
</tr>
<tr>
<td>TAB40</td>
<td>TRANS</td>
<td>S-FORTRAN</td>
<td>S-FORTRAN (TM)</td>
<td>TAB40</td>
</tr>
<tr>
<td>VIRTUAL OS</td>
<td>TRANS</td>
<td>S-FORTRAN</td>
<td>S-FORTRAN (TM)</td>
<td>VIRTUAL OS</td>
</tr>
</tbody>
</table>

### Notes:
- **Translation** includes tools for converting code from one language to another.
- **Conversion** pertains to tools that convert between related languages.
- **Macro Expansion** involves tools that expand macro definitions.
- **Structure Preprocessing** are tools that preprocess structured code.
- **Compilation** encompasses tools for compiling code.
<table>
<thead>
<tr>
<th>. . EDITING</th>
<th>. . RESTRUCTURING</th>
<th>. . INSTRUMENTATION</th>
<th>. . FORMATTING</th>
<th>. . SYNTHESIS</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACT/1</td>
<td>ARGUS/MICRO</td>
<td>CAVS</td>
<td>AUTOMATIC DOCUM</td>
<td>COPE (TM)</td>
</tr>
<tr>
<td>AUTOIDEF0</td>
<td>AUTO-DBO</td>
<td>FOSTRAX</td>
<td>COBOL/QDM</td>
<td>FOCUS</td>
</tr>
<tr>
<td>IFTRAN (TM)</td>
<td>FASP</td>
<td>HAWKEYE (TM)</td>
<td>COBOL/SP</td>
<td>GENTEXTS</td>
</tr>
<tr>
<td>JET</td>
<td>ISUS</td>
<td>HAWKEYE (TM)</td>
<td>COBOL/SPP</td>
<td></td>
</tr>
<tr>
<td>LILITH</td>
<td>LIBRARIAN</td>
<td>HAWKEYE (TM)</td>
<td>COBOL/SPP</td>
<td></td>
</tr>
<tr>
<td>MPS</td>
<td>LOGICFLOW</td>
<td>HAWKEYE (TM)</td>
<td>CAVS</td>
<td></td>
</tr>
<tr>
<td>RATCODER</td>
<td>MEDL-X</td>
<td>HAWKEYE (TM)</td>
<td>CAVS</td>
<td></td>
</tr>
<tr>
<td>SMT</td>
<td>PWB FOR VAX/VMS</td>
<td>HAWKEYE (TM)</td>
<td>CAVS</td>
<td></td>
</tr>
<tr>
<td>THE ENGINE</td>
<td>TOOLPACK</td>
<td>HAWKEYE (TM)</td>
<td>CAVS</td>
<td></td>
</tr>
<tr>
<td>VIRTUAL OS</td>
<td>UCSD P-SYSTEM</td>
<td>HAWKEYE (TM)</td>
<td>CAVS</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>. . EDITING</td>
<td>. . RESTRUCTURING</td>
<td>. . INSTRUMENTATION</td>
<td>. . FORMATTING</td>
<td>. . SYNTHESIS</td>
</tr>
<tr>
<td>ACT/1</td>
<td>ARGUS/MICRO</td>
<td>CAVS</td>
<td>AUTOMATIC DOCUM</td>
<td>COPE (TM)</td>
</tr>
<tr>
<td>AUTOIDEF0</td>
<td>AUTO-DBO</td>
<td>FOSTRAX</td>
<td>COBOL/QDM</td>
<td>FOCUS</td>
</tr>
<tr>
<td>IFTRAN (TM)</td>
<td>FASP</td>
<td>HAWKEYE (TM)</td>
<td>COBOL/SP</td>
<td>GENTEXTS</td>
</tr>
<tr>
<td>JET</td>
<td>ISUS</td>
<td>HAWKEYE (TM)</td>
<td>COBOL/SPP</td>
<td></td>
</tr>
<tr>
<td>LILITH</td>
<td>LIBRARIAN</td>
<td>HAWKEYE (TM)</td>
<td>CAVS</td>
<td></td>
</tr>
<tr>
<td>MPS</td>
<td>LOGICFLOW</td>
<td>HAWKEYE (TM)</td>
<td>CAVS</td>
<td></td>
</tr>
<tr>
<td>RATCODER</td>
<td>MEDL-X</td>
<td>HAWKEYE (TM)</td>
<td>CAVS</td>
<td></td>
</tr>
<tr>
<td>SMT</td>
<td>PWB FOR VAX/VMS</td>
<td>HAWKEYE (TM)</td>
<td>CAVS</td>
<td></td>
</tr>
<tr>
<td>THE ENGINE</td>
<td>TOOLPACK</td>
<td>HAWKEYE (TM)</td>
<td>CAVS</td>
<td></td>
</tr>
<tr>
<td>VIRTUAL OS</td>
<td>UCSD P-SYSTEM</td>
<td>HAWKEYE (TM)</td>
<td>CAVS</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tools</td>
<td>Features</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---------------</td>
<td>-----------</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>INFORM</td>
<td>IPDS</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MARK IV (TM)</td>
<td>METTRAN</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PROGRAM GENERAT</td>
<td>SCOPE</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SURGE 72</td>
<td>SYSTEM-80</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>YACC</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>. . OPTIMIZATION.................................COTUNE II</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>LILITH</td>
<td>MEMORY MNG LIB OPTIMUS</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>OPTIMIZER II</td>
<td>QCM</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## APPENDIX D

### TOOLS BY STATIC ANALYSIS FEATURES

<table>
<thead>
<tr>
<th>. . COMPARISON</th>
<th>CCS</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMPARE</td>
<td>DECKBOY COMPAR</td>
</tr>
<tr>
<td>DRIVER</td>
<td>FADEBUG-I</td>
</tr>
<tr>
<td>MSEF</td>
<td>PROG COMP ANAL</td>
</tr>
<tr>
<td>SCAN/370</td>
<td>SOFTOOL 80 (TM)</td>
</tr>
<tr>
<td>TRAILBLAZER</td>
<td>VIRTUAL OS</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>. . DATA FLOW ANALYSIS</th>
<th>ADF</th>
</tr>
</thead>
<tbody>
<tr>
<td>ATDG</td>
<td>AUDIT</td>
</tr>
<tr>
<td>DARTS</td>
<td>DAVE</td>
</tr>
<tr>
<td>DDPM</td>
<td>FACES</td>
</tr>
<tr>
<td>FAVS</td>
<td>ISUS</td>
</tr>
<tr>
<td>RXVP80 (TM)</td>
<td>SADAT</td>
</tr>
<tr>
<td>SNOOP</td>
<td>SOFTOOL 80 (TM)</td>
</tr>
<tr>
<td>SURVAYOR</td>
<td>TOOLPACK</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>. . INTERFACE ANALYSIS</th>
<th>AUTO-DBO</th>
</tr>
</thead>
<tbody>
<tr>
<td>DAVE</td>
<td>FAST</td>
</tr>
<tr>
<td>INFORM/REFORM</td>
<td>JAVS</td>
</tr>
<tr>
<td>RXVP80 (TM)</td>
<td>SEF</td>
</tr>
<tr>
<td>SYDIM</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>. . CROSS REFERENCE</th>
<th>ADS/CERL</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUTOFLOW (TM)</td>
<td>CALLREF</td>
</tr>
<tr>
<td>CCREF</td>
<td>CICS DUMP ANAL</td>
</tr>
<tr>
<td>COMGEN/TRW</td>
<td>COMGEN</td>
</tr>
<tr>
<td>COMLIST</td>
<td>COMMAP</td>
</tr>
<tr>
<td>CONFIG</td>
<td>CORE</td>
</tr>
<tr>
<td>CROREF</td>
<td>DA</td>
</tr>
<tr>
<td>DCD</td>
<td>DDPM</td>
</tr>
<tr>
<td>DICTANL/LOCATE</td>
<td>DPNDCY</td>
</tr>
<tr>
<td>FAVS</td>
<td>FLOBOL</td>
</tr>
<tr>
<td>FORREF</td>
<td>FORTREF</td>
</tr>
<tr>
<td>GIRAFF</td>
<td>INTERFACE DOCUM</td>
</tr>
<tr>
<td>LEXICON</td>
<td>LIBREF</td>
</tr>
<tr>
<td>PBASIC</td>
<td>PDL</td>
</tr>
<tr>
<td>QUICK-DRAW</td>
<td>REFER</td>
</tr>
<tr>
<td>RISOS TOOLS</td>
<td>RTT</td>
</tr>
<tr>
<td>SARA</td>
<td>SCAN/370</td>
</tr>
<tr>
<td>Feature</td>
<td>Tool 1</td>
</tr>
<tr>
<td>------------------------------</td>
<td>-------------------------</td>
</tr>
<tr>
<td><strong>Complexity Measurement</strong></td>
<td>DARTS</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Completeness Checking</strong></td>
<td>CADSAT</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Cost Estimation</strong></td>
<td>AFFIRM</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Units Analysis</strong></td>
<td>RXVP80 (TM)</td>
</tr>
<tr>
<td><strong>Type Analysis</strong></td>
<td>FAVS</td>
</tr>
<tr>
<td><strong>Statistical Analysis</strong></td>
<td>ASRP</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Auditing</strong></td>
<td>ADS/CERL</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Management

<table>
<thead>
<tr>
<th>Tool</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACT/1</td>
<td>ADA Environment ALIAS</td>
</tr>
<tr>
<td>ARGUS/MICRO</td>
<td>ARTS ASA-PMs</td>
</tr>
<tr>
<td>ASSET</td>
<td>AUTOCOM AUTOIDEF0</td>
</tr>
<tr>
<td>AUTORETEST</td>
<td>BEST/1 (TM) BLKGEN-BDD</td>
</tr>
<tr>
<td>BLKGEN/SPECPN</td>
<td>BUDGET VS ACTUA CAPTURE/MVS (TM)</td>
</tr>
<tr>
<td>CCS</td>
<td>CHECKSUM COMDIM</td>
</tr>
<tr>
<td>COMGEN</td>
<td>COMGEN/TRW COMLIST</td>
</tr>
<tr>
<td>COMSORT</td>
<td>COMSTAR CONFIG</td>
</tr>
<tr>
<td>CPA</td>
<td>CPAL CRYSTAL (TM)</td>
</tr>
<tr>
<td>CS4</td>
<td>DAS DATA DESIGNER</td>
</tr>
<tr>
<td>DDPM</td>
<td>DIRCOM DOSSIER</td>
</tr>
<tr>
<td>DPNDICY</td>
<td>DRIVER EASYTROL</td>
</tr>
<tr>
<td>EVP</td>
<td>FASP FOCUS</td>
</tr>
<tr>
<td>FORMAN</td>
<td>FTNXHRF GIM/GIM II</td>
</tr>
<tr>
<td>HARPO</td>
<td>INFORM ISUS</td>
</tr>
<tr>
<td>LANG INSTRUCTOR</td>
<td>LIBRARIAN MEDL-D</td>
</tr>
<tr>
<td>MEDL-X</td>
<td>MEDL-R MEDL-P</td>
</tr>
<tr>
<td>MSEF</td>
<td>N5500 N-SQUARED</td>
</tr>
<tr>
<td>NETWORK PLANNER</td>
<td>NODAL ONLINE ASSIST</td>
</tr>
<tr>
<td>PAC II</td>
<td>PDS PDSS</td>
</tr>
<tr>
<td>PF</td>
<td>PMCS PMS IV</td>
</tr>
<tr>
<td>PPP</td>
<td>PROG COMP ANAL PRONET</td>
</tr>
<tr>
<td>PSL</td>
<td>PWB FOR VAX/VMS QCRT</td>
</tr>
<tr>
<td>RXVP80 (TM)</td>
<td>SCG/DQM SCHEMACODE</td>
</tr>
<tr>
<td>SDP</td>
<td>SDVS SEF</td>
</tr>
<tr>
<td>SLIB</td>
<td>SMS SNOOP</td>
</tr>
<tr>
<td>SOFTOOL 80 (TM)</td>
<td>SPC SPECTRUM-1</td>
</tr>
<tr>
<td>SPREAD</td>
<td>SREM SUS</td>
</tr>
<tr>
<td>SYDIM</td>
<td>SYMCRS TAFIRM</td>
</tr>
<tr>
<td>TAPS/AM</td>
<td>TOOLPACK TOOLS DATABASE</td>
</tr>
<tr>
<td>TRAILBLAZER</td>
<td>TRMS TRWPLT</td>
</tr>
<tr>
<td>VIRTUAL OS</td>
<td></td>
</tr>
</tbody>
</table>

## Configuration Management

<table>
<thead>
<tr>
<th>Tool</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADA ENVIRONMENT</td>
<td>ASSET AUTOIDEF0</td>
</tr>
<tr>
<td>BLKGEN/SPECPN</td>
<td>CCS CHECKSUM</td>
</tr>
<tr>
<td>COMGEN</td>
<td>CONFIG CPAL</td>
</tr>
<tr>
<td>GIM/GIM II</td>
<td>ISUS PROG COMP ANAL</td>
</tr>
<tr>
<td>PWB FOR VAX/VMS</td>
<td>SDP SDVS</td>
</tr>
<tr>
<td>SEF</td>
<td>SMS SOFTOOL 80 (TM)</td>
</tr>
<tr>
<td>SUS</td>
<td></td>
</tr>
</tbody>
</table>

## Project Management

<table>
<thead>
<tr>
<th>Tool</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BUDGET VS ACTUA</td>
<td>EASYTROL HARP</td>
</tr>
<tr>
<td>N5500</td>
<td>N-SQUARED NETWORK PLANNER</td>
</tr>
<tr>
<td>PAC II</td>
<td>PDSS PF</td>
</tr>
<tr>
<td>PMCS</td>
<td>PMS IV PPP</td>
</tr>
<tr>
<td>PRONET</td>
<td>SPC SPECTRUM-1</td>
</tr>
<tr>
<td>SPREAD</td>
<td>TRMS</td>
</tr>
</tbody>
</table>

## Version Control

<table>
<thead>
<tr>
<th>Tool</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BLKGEN/SPECPN</td>
<td>CHECKSUM LIBRARIAN</td>
</tr>
<tr>
<td>MSEF</td>
<td>SDP SMS</td>
</tr>
</tbody>
</table>
TEST DATA MANAGEMENT

- Asset: Autoretest
- Msef: Nodal
- Se: Tafirm

LIBRARY MANAGEMENT

- Asset: Comstar
- Dossier: Librarian
- Se: Slib

FILES MANAGEMENT

- Argus/micro: Comstar
- Pds: Psl
- Taps/am: Toolpack

DATA BASE MANAGEMENT

- Blkgen/specpn: Cs4
- Focus: Gim/gim ii
- Medl-r: Medl-p
- Medl-d: Srem

CHANGE CONTROL

- Comlist: Das
- Fttxref: Isus
- Scg/dqm: Sms
- Sus

GLOBAL VARIABLE MANAGEMENT

- Autocom: Blkgen/specpn
- Comdim: Comgen
- Comlist: Comsort
- Dpnacy: Evp
- Fttxref: Se
- Sydim: Symcrs

MANAGEMENT PLANNING

- Online assist: Slib

DOCUMENTATION MANAGEMENT

- Act/1

PERFORMANCE MANAGEMENT

- Capture/mvs(tm)

CAPACITY PLANNING

- Capture/mvs(tm)

TRACKING

- Asset: Cadsat
- Das: Med-sys
- Ra: Rtt
- Srep: Stag/tems

ERROR CHECKING

- Atdg: Auditor

TOOLPACK VIRTUAL OS
TOOLS BY STATIC ANALYSIS FEATURES

<table>
<thead>
<tr>
<th>FAME</th>
<th>FASP</th>
<th>FAST</th>
</tr>
</thead>
<tbody>
<tr>
<td>FORAN</td>
<td>ISUS</td>
<td>LOGICFLOW</td>
</tr>
<tr>
<td>MEFIA</td>
<td>PSL/PSA</td>
<td>RXVP80 (TM)</td>
</tr>
<tr>
<td>SFORT-1</td>
<td>SFTRAN3</td>
<td>SRTRAN.BASLINE</td>
</tr>
<tr>
<td>STRUCTURE(S)</td>
<td>SYSTEM MONITOR</td>
<td>THE ENGINE</td>
</tr>
</tbody>
</table>

. . SCHEDULING..........................SLIM SPECTRUM-1

. . STRUCTURE CHECKING.....................ADF
| CAVS   | COBOL STRUCT | FAVS |
| FCA    | IFTRAN (TM)  | ISDS |
| ISUS   | LOGICFLOW    | REFORM |
| RXVP80 (TM) | SARA    | SCAN/370 |
| SDP/MAYDA | SOFTOOL 80 (TM) | STRUCTURING ENG |
| STRUCTURE(S) | THE ENGINE | TOOLPACK |

. . SCANNING...............................AFS
| ASC   | AUTDOC | AUTOFLOW (TM) |
| AUTOFLOW/TRW | AUTOMATIC DOCUM | CADMUS |
| CICS DUMP ANALY | CRISPFLOW | DA |
| DATAMACS | DECA | DOCGEN |
| DOCU/TEXT | DOCUMENTER A | DOCUMENT |
| DOCUMENTER | DOCUMENTOR | DQM |
| ESAP   | FLOBOL  | FLODIA |
| FLOWGEN | FORTREF | GADTR AID |
| INFORM/REFORM | IORL | JET |
| JSDD   | LAYOUT  | LIBRARIAN |
| LOGIFLOW | MPS | MTR |
| NUMBER | PDS FLOW | PIDGIN-FASP |
| PROGLOOK | QUICK-DRAW | REFER |
| RENAME | RXVP80 (TM) | S-FORTRAN |
| SARA   | SCG    | SDDL |
| SDP/MAYDA | SOFTOOL 80 (TM) | SPEAR |
| SPECLE/DARS | SPELL | STRUCTURE(S) |
| SYDOC   | THE ENGINE | UCA |
| UF     |        | |

. . I/O SPECIFICATION ANALYSIS.............COBOL/DV
| DATAMACS | FADDEBUG-I | GENTESTS |
| PREF HDR GEN |        | |

Page D-5
APPENDIX E
TOOLS BY DYNAMIC ANALYSIS FEATURES

. COVERAGE ANALYSIS
  ASSIST-I  ATA-FASP  ATA-SAI
  ATTEST   CADA     CAVS
  CGJA     COBOL TESTING  COTUNE II
  DYNAL    EAVS     FASP
  FAVS     FORTRAN TESTING  FTN-77 ANALYZER
  FTN ANALYZER IFTRAN (TM)  ITB
  JAVS     JIGSAW    JOVIAL TCA
  LOGIC    NODAL    PACE
  PACE-C    PDS      PET
  RXVP80 (TM) SADAT       SOFTOOL 80 (TM)
  TATTLE   TCAT     TDEM
  TEST PREDICTOR TFA    THE ENGINE
  TOOLPACK  TPT    TRAILBLAZER

. ASSERTION CHECKING
  ATA-SAI  CAVS  EFFIGY
  FTN-77 ANALYZER IFTRAN (TM)  IPDS
  RXVP80 (TM) SELECT

. SYMBOLIC EXECUTION
  ASSIST-I  ATTEST  CASEGEN
  EFFIGY    NASA-VATS RXVP80 (TM)
  SADAT     SELECT  TEVERE-1

. SIMULATION
  BEST/1 (TM) CONFIGURATOR CRYSTAL (TM)
  DARTS    DDPM    DPAD
  HARDWARE SIMULA MEDL-P   PERCAM
  POD      SALSIM   SARA
  SCAN/370 SCERT     SDVS
  SLIM     SREM     TAPS/AM

. TUNING
  COBOL OPTIMIZ CUE  FASP
  FAVS     FORTRAN OPTIMIZ FTN-77 ANALYZER
  FTN ANALYZER IFTRAN (TM)  INSERT
  JAVS     MONITOR   NODAL
  POD      PROGLOOK  QCM
TOOLS BY DYNAMIC ANALYSIS FEATURES

RXVP80 (TM) SADAT SARA-H
SARA-U SARA-IV SARA-III
SCAN/370 SMT SOFTOOL 80 (TM)
SPRINT SYSTEM MONITOR TIMECS
TSA/PPE

... RESOURCE UTILIZATION

RESOURCE UTILIZATION

CAPTURE/MVS(TM) CUE DARTS
DDPM HARDWARE SIMULA LOOK
PPE PRONET QCM
REFLECT II SARA-H SARA-IV
SARA-III SARA-U SMT
TSA/PPE

... TIMING

TIMING

COBOL/ADE COTUNE II DARTS
DDPM FASP HARDWARE SIMULA
LOGIC MONITOR PPE
PROGLOOK REFLECT II SMT
SOFTWARE TRACING SOFTWARE TRACING
SOFTOOL 80 (TM) SPRINT TPA
TIMECS TIMER

TRACING

ASSIST-I ATA-FASP ATA-SAI
COBOL/DV COBOL TRACING COBOL/ADE
EAVS EFFIGY EXPEDITER
FORTRAN TRACING FTN-77 ANALYZER IFTRAN (TM)
INSERT INSTRU ITB
JAVS LOGIC MONITOR
RXVP80 (TM) SADAT SCAN/370
SELECT SOFTOOL 80 (TM) TAFIRM
THE ENGINE TOOLPACK TPT
TRAILBLAZER XPEDITER

BREAKPOINT CONTROL

BREAKPOINT CONTROL

ASSIST-I

PATH FLOW TRACING

PATH FLOW TRACING

EAVS FORTRAN TRACING INSERT
INSTRU JAVS LOGIC
MONITOR SADAT SCAN/370
SELECT TAFIRM TRAILBLAZER

DATA FLOW TRACING

DATA FLOW TRACING

INSTRU

LOGIC FLOW TRACING

LOGIC FLOW TRACING

ASSIST-I

INSTRU

REGRESSION TESTING

REGRESSION TESTING

DATAMACS DRIVER EXPEDITER
SEF XPEDITER

CONSTRAINT EVALUATION

CONSTRAINT EVALUATION

ATDG

RXVP80 (TM) TEST PREDICTOR
## APPENDIX F

### TOOLS BY USER OUTPUT

<table>
<thead>
<tr>
<th>. . DIAGNOSTICS</th>
<th>ADA ENVIRONMENT</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADS</td>
<td>AFFIRM</td>
</tr>
<tr>
<td>ASSET</td>
<td>ARGUS/MICRO</td>
</tr>
<tr>
<td>AUDIT</td>
<td>ATDG</td>
</tr>
<tr>
<td>CARA</td>
<td>ATDG</td>
</tr>
<tr>
<td>CHECKSUM</td>
<td>BEST/1 (TM)</td>
</tr>
<tr>
<td>COBOL/ADE</td>
<td>CA</td>
</tr>
<tr>
<td>COMMTP</td>
<td>CAVS</td>
</tr>
<tr>
<td>CPA-ADR</td>
<td>CCA</td>
</tr>
<tr>
<td>CICS DUMP ANALY</td>
<td>COBOL/CP</td>
</tr>
<tr>
<td>COBOL/QDM</td>
<td>COGENT</td>
</tr>
<tr>
<td>COMSCAN</td>
<td>CORE</td>
</tr>
<tr>
<td>CRISTAL (TM)</td>
<td>DAVE</td>
</tr>
<tr>
<td>DECA</td>
<td>EAVS</td>
</tr>
<tr>
<td>ENFORCE</td>
<td>FAST</td>
</tr>
<tr>
<td>FAST</td>
<td>FAVS</td>
</tr>
<tr>
<td>FOCUS</td>
<td>FORAN</td>
</tr>
<tr>
<td>GIRAFF</td>
<td>GOTO-ANALYZER</td>
</tr>
<tr>
<td>HFTRAN (TM)</td>
<td>INFORM</td>
</tr>
<tr>
<td>INSERT</td>
<td>IPDS</td>
</tr>
<tr>
<td>ISUS</td>
<td>JOCIT</td>
</tr>
<tr>
<td>JAVS</td>
<td>JIGSAW</td>
</tr>
<tr>
<td>JOVIAL/VS</td>
<td>JOVIAL TCA</td>
</tr>
<tr>
<td>MARK IV (TM)</td>
<td>MED-SYS</td>
</tr>
<tr>
<td>MED-SYS</td>
<td>NASA-VATS</td>
</tr>
<tr>
<td>PADM</td>
<td>PET</td>
</tr>
<tr>
<td>PREF HDR GEN</td>
<td>PFORT</td>
</tr>
<tr>
<td>PSL/PSA</td>
<td>PWB FOR VAX/VMS</td>
</tr>
<tr>
<td>RXVP80 (TM)</td>
<td>S-FORTRAN</td>
</tr>
<tr>
<td>SCAN/370</td>
<td>SFORT-1</td>
</tr>
<tr>
<td>SNOOP</td>
<td>SREM</td>
</tr>
<tr>
<td>STRUC</td>
<td>SURVAYOR</td>
</tr>
<tr>
<td>SYSTEM MONITOR</td>
<td>TOOLPACK</td>
</tr>
<tr>
<td>TRANSFOR</td>
<td>TSA/PPE</td>
</tr>
<tr>
<td>VIRTUAL OS</td>
<td>VIRTUAL OS</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>. . USER-ORIENTED TEXT</th>
<th>ACT/1</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADS/CERL</td>
<td>ARGUS/MICRO</td>
</tr>
<tr>
<td>ATTEST</td>
<td>ASC</td>
</tr>
<tr>
<td>AUTOIDEF0</td>
<td>AUTO-DBO</td>
</tr>
<tr>
<td>AUTOMATIC DOCUM</td>
<td>CADMUS</td>
</tr>
<tr>
<td>CARA</td>
<td>AUTO-DOC</td>
</tr>
<tr>
<td>COBOL/DV</td>
<td>CA</td>
</tr>
<tr>
<td>CSAV</td>
<td>COBOL/DV</td>
</tr>
<tr>
<td>CROREF</td>
<td>CS4</td>
</tr>
<tr>
<td>DA</td>
<td>DAS</td>
</tr>
<tr>
<td>DEPCHT</td>
<td>DCD</td>
</tr>
<tr>
<td>DECA</td>
<td>DICTANL/LOCATE</td>
</tr>
<tr>
<td>DOCU/TEXT</td>
<td>DOCUMENTER A</td>
</tr>
<tr>
<td>DOCUMENT</td>
<td>DOCUMENTER</td>
</tr>
<tr>
<td>----------------</td>
<td>----------------</td>
</tr>
<tr>
<td>DOSSIER</td>
<td>EASYTROL</td>
</tr>
<tr>
<td>ESAP</td>
<td>FAVS</td>
</tr>
<tr>
<td>GIM/GIM II</td>
<td>INFORM/REFORM</td>
</tr>
<tr>
<td>LANG INSTRUCTOR</td>
<td>MEDL-D</td>
</tr>
<tr>
<td>MEDL-R</td>
<td>MPS</td>
</tr>
<tr>
<td>PDL</td>
<td>PSL</td>
</tr>
<tr>
<td>SARA</td>
<td>SCG/DQM</td>
</tr>
<tr>
<td>SEF</td>
<td>SIGS</td>
</tr>
<tr>
<td>SNOOP</td>
<td>SOFTOOL 80 (TM)</td>
</tr>
<tr>
<td>SREM</td>
<td>SREP</td>
</tr>
<tr>
<td>VIRTUAL OS</td>
<td></td>
</tr>
<tr>
<td></td>
<td>REPORTS</td>
</tr>
<tr>
<td></td>
<td>DA</td>
</tr>
<tr>
<td></td>
<td>DOCU/TEXT</td>
</tr>
<tr>
<td></td>
<td>DOSSIER</td>
</tr>
<tr>
<td></td>
<td>IORL</td>
</tr>
<tr>
<td></td>
<td>MEDL-R</td>
</tr>
<tr>
<td></td>
<td>PSL</td>
</tr>
<tr>
<td></td>
<td>SEF</td>
</tr>
<tr>
<td></td>
<td>SREM</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>DOCUMENTATION</th>
<th>ACT/1</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADS/CERL</td>
<td>ASC</td>
</tr>
<tr>
<td>AUTDOC</td>
<td>AUTO-DBO</td>
</tr>
<tr>
<td>AUTOMATIC DOCUM</td>
<td>CADMUS</td>
</tr>
<tr>
<td>CAVS</td>
<td>COBOL/DV</td>
</tr>
<tr>
<td>CROREF</td>
<td>CS4</td>
</tr>
<tr>
<td>DECA</td>
<td>DEPCHT</td>
</tr>
<tr>
<td>DOCGEN</td>
<td>DOCUMENTER A</td>
</tr>
<tr>
<td>DOCUMENTER</td>
<td>ECA AUTOMATION A</td>
</tr>
<tr>
<td>FAVS</td>
<td>FTNXXREF</td>
</tr>
<tr>
<td>ONLINE ASSIST</td>
<td>PSL</td>
</tr>
<tr>
<td>SCG/DQM</td>
<td>SDF/MAYDA</td>
</tr>
<tr>
<td>SIGS</td>
<td>SNOOP</td>
</tr>
<tr>
<td>SREP</td>
<td>SOFTOOL 80 (TM)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>GRAPHICS</th>
<th>ADF</th>
</tr>
</thead>
<tbody>
<tr>
<td>AFS</td>
<td>AISIM</td>
</tr>
<tr>
<td>ARGUS/MICRO</td>
<td>ASC</td>
</tr>
<tr>
<td>AUTOFLOW/TRW</td>
<td>AUTOFLOW (TM)</td>
</tr>
<tr>
<td>BUDGET VS ACTUA</td>
<td>CADSUT</td>
</tr>
<tr>
<td>CONFIGURATOR</td>
<td>COTUNE II</td>
</tr>
<tr>
<td>CRYSTAL (TM)</td>
<td>DA</td>
</tr>
<tr>
<td>DATA DESIGNER</td>
<td>DCD</td>
</tr>
<tr>
<td>DQM</td>
<td>ESAP</td>
</tr>
<tr>
<td>FLOBOL</td>
<td>FLODIA</td>
</tr>
<tr>
<td>FOCUS</td>
<td>FOSTRA</td>
</tr>
<tr>
<td>GRAFMAKER</td>
<td>HARP</td>
</tr>
<tr>
<td>INFORM</td>
<td>JOYCE</td>
</tr>
<tr>
<td>LAYOUT</td>
<td>LILITH</td>
</tr>
<tr>
<td>LOGIFLOW</td>
<td>LOOK</td>
</tr>
<tr>
<td>MTR</td>
<td>N-SQUARED</td>
</tr>
<tr>
<td>PDS FLOW</td>
<td>PDSS</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>VIRTUAL OS</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>REPORTS</td>
</tr>
<tr>
<td></td>
<td>DA</td>
</tr>
<tr>
<td></td>
<td>DOCU/TEXT</td>
</tr>
<tr>
<td></td>
<td>DOSSIER</td>
</tr>
<tr>
<td></td>
<td>IORL</td>
</tr>
<tr>
<td></td>
<td>MEDL-R</td>
</tr>
<tr>
<td></td>
<td>PSL</td>
</tr>
<tr>
<td></td>
<td>SEF</td>
</tr>
<tr>
<td></td>
<td>SREM</td>
</tr>
<tr>
<td>TOOLS BY USER OUTPUT</td>
<td></td>
</tr>
<tr>
<td>---------------------</td>
<td>-------</td>
</tr>
<tr>
<td>POD</td>
<td>PPP</td>
</tr>
<tr>
<td>PRONET</td>
<td>PSL/PSA</td>
</tr>
<tr>
<td>RISOS TOOLS</td>
<td>RTT</td>
</tr>
<tr>
<td>SARA</td>
<td>SARA-U</td>
</tr>
<tr>
<td>SCAN/370</td>
<td>SCG</td>
</tr>
<tr>
<td>SIGS</td>
<td>SLIM</td>
</tr>
<tr>
<td>SPEAR</td>
<td>SPECLE/DARS</td>
</tr>
<tr>
<td>SREM</td>
<td>SRIMP</td>
</tr>
<tr>
<td>SYDOC</td>
<td>TIMER</td>
</tr>
<tr>
<td>UF</td>
<td></td>
</tr>
<tr>
<td>FLOW CHARTS</td>
<td></td>
</tr>
<tr>
<td>AMPIC</td>
<td>AUTDOC</td>
</tr>
<tr>
<td>CRISPFLOW</td>
<td>DCD</td>
</tr>
<tr>
<td>FLOBOL</td>
<td>FLODIA</td>
</tr>
<tr>
<td>JOYCE</td>
<td>JSDD</td>
</tr>
<tr>
<td>LOGIFLOW</td>
<td>PDS FLOW</td>
</tr>
<tr>
<td>RISOS TOOLS</td>
<td>SNOOP</td>
</tr>
<tr>
<td>UF</td>
<td></td>
</tr>
<tr>
<td>HIPO CHARTS</td>
<td>SNOOP</td>
</tr>
<tr>
<td>BAR CHARTS</td>
<td>SLIM</td>
</tr>
<tr>
<td>LINE GRAPHS</td>
<td>SLIM</td>
</tr>
<tr>
<td>STRUCTURE CHARTS</td>
<td>SCG</td>
</tr>
<tr>
<td>DESIGN CHARTS</td>
<td>DATA DESIGNER</td>
</tr>
<tr>
<td>DQM</td>
<td>GADTR AID</td>
</tr>
<tr>
<td>LOGICFLOW</td>
<td>SPECLE/DARS</td>
</tr>
<tr>
<td>RISOS TOOLS</td>
<td></td>
</tr>
<tr>
<td>UF</td>
<td></td>
</tr>
<tr>
<td>HIERARCHICAL TREE</td>
<td>DARTS</td>
</tr>
<tr>
<td>FAME</td>
<td>GADTR AID</td>
</tr>
<tr>
<td>SCAN/370</td>
<td>SDP/MAYDA</td>
</tr>
<tr>
<td>SYDOC</td>
<td></td>
</tr>
<tr>
<td>CONTROL MAP</td>
<td>FAME</td>
</tr>
<tr>
<td>ACTIVITY DIAGRAM</td>
<td>BUDGET VS ACTUA</td>
</tr>
<tr>
<td>PRONET</td>
<td>SARA-H</td>
</tr>
<tr>
<td>CHARTS</td>
<td>ASC</td>
</tr>
<tr>
<td>CAPTURE/MVS (TM)</td>
<td></td>
</tr>
<tr>
<td>HISTOGRAMS</td>
<td>COTUNE II</td>
</tr>
<tr>
<td>MILESTONE CHARTS</td>
<td>N-SQUARED</td>
</tr>
<tr>
<td>ACTIVITY DIAGRAMS</td>
<td>PROGLOOK</td>
</tr>
<tr>
<td>HIERARCHICAL TREES</td>
<td>GADTR AID</td>
</tr>
<tr>
<td>TABLES</td>
<td>ADS/CERL</td>
</tr>
<tr>
<td>AISIM</td>
<td>ARGUS/MICRO</td>
</tr>
<tr>
<td>Tool</td>
<td>Tool</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-------------------------------</td>
</tr>
<tr>
<td>ASA-PMS</td>
<td>ATA-FASP</td>
</tr>
<tr>
<td>ATDG</td>
<td>BEST/1 (TM)</td>
</tr>
<tr>
<td>CALLREF</td>
<td>CAPTURE/MVS(TM)</td>
</tr>
<tr>
<td>CCREF</td>
<td>CENSUS</td>
</tr>
<tr>
<td>CICS DUMP ANALYZER</td>
<td>COBOL TRACING</td>
</tr>
<tr>
<td>COBOL STRUCT</td>
<td>COBOL/DV</td>
</tr>
<tr>
<td>COBOL OPTIMIZ</td>
<td>COMGEN</td>
</tr>
<tr>
<td>COMLIST/TRW</td>
<td>COMMAP</td>
</tr>
<tr>
<td>CONFIG</td>
<td>CPA-ADR</td>
</tr>
<tr>
<td>CRYSTAL (TM)</td>
<td>CUE</td>
</tr>
<tr>
<td>DARTS</td>
<td>DATA DESIGNER</td>
</tr>
<tr>
<td>DCD</td>
<td>DDPM</td>
</tr>
<tr>
<td>DICTANL/LOCATE</td>
<td>DPAD</td>
</tr>
<tr>
<td>DQMS</td>
<td>DYNALIST</td>
</tr>
<tr>
<td>FAVS</td>
<td>FLOBOL</td>
</tr>
<tr>
<td>FORAN</td>
<td>FORREF</td>
</tr>
<tr>
<td>FORTRAN OPTIMIZ</td>
<td>FORTREF</td>
</tr>
<tr>
<td>FTN-77 ANALYZER</td>
<td>FTN ANALYZER</td>
</tr>
<tr>
<td>GIM/GIM II</td>
<td>GIRAFF</td>
</tr>
<tr>
<td>INFORM</td>
<td>INSERT</td>
</tr>
<tr>
<td>INTERFACE DOCUM</td>
<td>ITB</td>
</tr>
<tr>
<td>JIGSAW</td>
<td>JOVIAL/J3SC</td>
</tr>
<tr>
<td>JOYCE</td>
<td>LEXICON</td>
</tr>
<tr>
<td>LIBREF</td>
<td>LOGIC</td>
</tr>
<tr>
<td>LOOK</td>
<td>MARK IV (TM)</td>
</tr>
<tr>
<td>MEDL-D</td>
<td>MENTOR</td>
</tr>
<tr>
<td>N5500</td>
<td>NODAL</td>
</tr>
<tr>
<td>PBASIC</td>
<td>PDL</td>
</tr>
<tr>
<td>PERCAM</td>
<td>PET</td>
</tr>
<tr>
<td>PMCS</td>
<td>PMS IV</td>
</tr>
<tr>
<td>PROGLook</td>
<td>PRONET</td>
</tr>
<tr>
<td>PWB FOR VAX/VMS</td>
<td>QCM</td>
</tr>
<tr>
<td>QUICK-DRAW</td>
<td>REFER</td>
</tr>
<tr>
<td>REFTRAN (TM)</td>
<td>REL MEAS MODEL</td>
</tr>
<tr>
<td>RTT</td>
<td>RXVP80 (TM)</td>
</tr>
<tr>
<td>SALSIM</td>
<td>SAP</td>
</tr>
<tr>
<td>SARA-III</td>
<td>SARA-U</td>
</tr>
<tr>
<td>SARA-IV</td>
<td>SCAN/370</td>
</tr>
<tr>
<td>SCG/DQMS</td>
<td>SDP</td>
</tr>
<tr>
<td>SLIM</td>
<td>SMT</td>
</tr>
<tr>
<td>SOFTOOL 80 (TM)</td>
<td>SPC</td>
</tr>
<tr>
<td>SPRINT</td>
<td>STAG/TEMS</td>
</tr>
<tr>
<td>STRUCT</td>
<td>SUBCRS</td>
</tr>
<tr>
<td>SYDIM</td>
<td>SYDOC</td>
</tr>
<tr>
<td>SYXREF</td>
<td>TAIFIRM</td>
</tr>
<tr>
<td>TCAT</td>
<td>TDEP</td>
</tr>
<tr>
<td>TFA</td>
<td>TIMECS</td>
</tr>
<tr>
<td>TOOLPACK</td>
<td>TOOLS DATABASE</td>
</tr>
<tr>
<td>TRAILBLAZER</td>
<td>TRMS</td>
</tr>
</tbody>
</table>

**LISTINGS**

<table>
<thead>
<tr>
<th>Tool</th>
<th>Tool</th>
<th>Tool</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACT/1</td>
<td>ADA COMPILER</td>
<td>ADA ENVIRONMENT</td>
</tr>
<tr>
<td>ADF</td>
<td>AFFIRM</td>
<td>ALIAS</td>
</tr>
<tr>
<td>AMPIC</td>
<td>ARGUS/MICRO</td>
<td>ASA-PMS</td>
</tr>
<tr>
<td>ASC</td>
<td>ASEQ</td>
<td>ASRP</td>
</tr>
<tr>
<td>ASSET</td>
<td>ASSIST-I</td>
<td>ATA-FASP</td>
</tr>
<tr>
<td>------------------</td>
<td>------------------</td>
<td>-----------------</td>
</tr>
<tr>
<td>ATA-SAI</td>
<td>ATDG</td>
<td>ATTEST</td>
</tr>
<tr>
<td>AUDIT</td>
<td>AUDITOR</td>
<td>AUTOCOM</td>
</tr>
<tr>
<td>AUTOFLOW (TM)</td>
<td>AUTOFLOW/TRW</td>
<td>AUTOMATIC DOCUM</td>
</tr>
<tr>
<td>AUTORETEST</td>
<td>BLKGEN-BDD</td>
<td>BSC</td>
</tr>
<tr>
<td>CA</td>
<td>CADA</td>
<td>CADSAT</td>
</tr>
<tr>
<td>CALLREF</td>
<td>CASEGEN</td>
<td>CBLSHORT</td>
</tr>
<tr>
<td>CCA</td>
<td>CCREF</td>
<td>CCS</td>
</tr>
<tr>
<td>CGJA</td>
<td>COBOL OPTIMIZ</td>
<td>COBOL TESTING</td>
</tr>
<tr>
<td>COBOL STRUCT</td>
<td>COBOL/ODM</td>
<td>COBOL TRACING</td>
</tr>
<tr>
<td>COGENT</td>
<td>COMGEN/TRW</td>
<td>COMLIST</td>
</tr>
<tr>
<td>COMLIST/TRW</td>
<td>COMPARE</td>
<td>CONFIG</td>
</tr>
<tr>
<td>COPE (TM)</td>
<td>CORE</td>
<td>COTUNE II</td>
</tr>
<tr>
<td>CPAL</td>
<td>CQD</td>
<td>CS4</td>
</tr>
<tr>
<td>CTC</td>
<td>DA</td>
<td>DECKBOY COMPAR</td>
</tr>
<tr>
<td>DIFFS (TM)</td>
<td>DIRCOM</td>
<td>DRIVER</td>
</tr>
<tr>
<td>DYNABENF</td>
<td>EAVS</td>
<td>EFFIGY</td>
</tr>
<tr>
<td>ENFORCE</td>
<td>EVP</td>
<td>EXPEDITER</td>
</tr>
<tr>
<td>FAEBUG-I</td>
<td>FASP</td>
<td>FCA</td>
</tr>
<tr>
<td>FORMAN</td>
<td>FORTRAN OPTIMIZ</td>
<td>FORTRAN TRACING</td>
</tr>
<tr>
<td>FORTRAN TESTING</td>
<td>FTN-77 ANALYZER</td>
<td>FTN ANALYZER</td>
</tr>
<tr>
<td>FTNCODER</td>
<td>GENTEXTS</td>
<td>HARDWARE SIMULA</td>
</tr>
<tr>
<td>HAWKEYE (TM)</td>
<td>IFTRAN (TM)</td>
<td>INSERT</td>
</tr>
<tr>
<td>INSTRU</td>
<td>IPDS</td>
<td>ISDS</td>
</tr>
<tr>
<td>ISUS</td>
<td>ITB</td>
<td>JAVS</td>
</tr>
<tr>
<td>JET</td>
<td>JIGSAW</td>
<td>JOCIT</td>
</tr>
<tr>
<td>JOVIAL TCA</td>
<td>LILITH</td>
<td>LOGICFLOW</td>
</tr>
<tr>
<td>LOGIC</td>
<td>LOGOS</td>
<td>MED-SYS</td>
</tr>
<tr>
<td>MEFIA</td>
<td>MENTOR</td>
<td>MODULE ORDERER</td>
</tr>
<tr>
<td>MONITOR</td>
<td>MSEE</td>
<td>MSL</td>
</tr>
<tr>
<td>NASA-VATS</td>
<td>NODAL</td>
<td>NUMBER/DEC</td>
</tr>
<tr>
<td>NUMBER</td>
<td>PACE-C</td>
<td>PACE</td>
</tr>
<tr>
<td>PBASEIC</td>
<td>PDS</td>
<td>PET</td>
</tr>
<tr>
<td>PFW</td>
<td>PIDGIN-FASP</td>
<td>PPE</td>
</tr>
<tr>
<td>PREFER HDR GEN</td>
<td>PROG COMP ANAL</td>
<td>PSL/PSA</td>
</tr>
<tr>
<td>QUICKCODE</td>
<td>RADC/FCA</td>
<td>REALIGNMENT SYS</td>
</tr>
<tr>
<td>REFORM</td>
<td>REFTTRAN (TM)</td>
<td>RENAME</td>
</tr>
<tr>
<td>RTT</td>
<td>RXVP80 (TM)</td>
<td>SDAT</td>
</tr>
<tr>
<td>SALSIM</td>
<td>SARA</td>
<td>SCAN/370</td>
</tr>
<tr>
<td>SCHEMACODE</td>
<td>SCOBOL (TM)</td>
<td>SDDL</td>
</tr>
<tr>
<td>SDL</td>
<td>SDP</td>
<td>SDP/MAYDA</td>
</tr>
<tr>
<td>SDVS</td>
<td>SELECT</td>
<td>SFORT-1</td>
</tr>
<tr>
<td>SFTRAN3</td>
<td>SOFTOOL 80 (TM)</td>
<td>SPECLE/DARS</td>
</tr>
<tr>
<td>SPELL</td>
<td>SREM</td>
<td>SRIMP</td>
</tr>
<tr>
<td>SRTRAN.BASLINE</td>
<td>STRUCTURIZER</td>
<td>SURVATOR</td>
</tr>
<tr>
<td>SUS</td>
<td>SYDOC</td>
<td>SYMCRS</td>
</tr>
<tr>
<td>SYSTEM MONITOR</td>
<td>SYSXREF</td>
<td>TAB40</td>
</tr>
<tr>
<td>TAIFIRM</td>
<td>TATTLE</td>
<td>TDBCOMP</td>
</tr>
<tr>
<td>TDEM</td>
<td>TEST PREDICTOR</td>
<td>TEVERE-1</td>
</tr>
<tr>
<td>TFA</td>
<td>THE ENGINE</td>
<td>TIDY</td>
</tr>
<tr>
<td>TOOLPACK</td>
<td>TPT</td>
<td>TRAILBLAZER</td>
</tr>
<tr>
<td>TSA/PPE</td>
<td>UCA</td>
<td>UCSD P-SYSTEM</td>
</tr>
<tr>
<td>VIRTUAL OS</td>
<td>XPEDITER</td>
<td></td>
</tr>
</tbody>
</table>
### APPENDIX G

#### TOOLS BY MACHINE OUTPUT

<table>
<thead>
<tr>
<th>Source Code Output</th>
<th>Ada-Atom</th>
<th>Ada-Sai</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assist-I</td>
<td>Fasp</td>
<td>Sai</td>
</tr>
<tr>
<td>Blkgen/specpn</td>
<td>Cada</td>
<td>Cavs</td>
</tr>
<tr>
<td>Cblshort</td>
<td>Cgja</td>
<td>Cobol/cp</td>
</tr>
<tr>
<td>Cobol testing</td>
<td>Cobol optiz</td>
<td>Cobol/spp</td>
</tr>
<tr>
<td>Cobol tracing</td>
<td>Cobol/spp</td>
<td>Coget</td>
</tr>
<tr>
<td>Comgen</td>
<td>Comgen/trw</td>
<td>Cope (tm)</td>
</tr>
<tr>
<td>Cpa</td>
<td>Cpal</td>
<td>Cqd</td>
</tr>
<tr>
<td>Cspp</td>
<td>Ctc</td>
<td>Dyna</td>
</tr>
<tr>
<td>Eavs</td>
<td>Enforce</td>
<td>Fasp</td>
</tr>
<tr>
<td>Favs</td>
<td>Fortran tracing</td>
<td>Fortran testing</td>
</tr>
<tr>
<td>Fortran optimiz</td>
<td>Fostra</td>
<td>Ft77 analyzer</td>
</tr>
<tr>
<td>Ftncoder</td>
<td>Gentexts</td>
<td></td>
</tr>
<tr>
<td>Gentests</td>
<td>Hawkeye (tm)</td>
<td>Iftran (tm)</td>
</tr>
<tr>
<td>Insert</td>
<td>Instru</td>
<td>Ipds</td>
</tr>
<tr>
<td>Isus</td>
<td>Itb</td>
<td>Javs</td>
</tr>
<tr>
<td>Jigsaw</td>
<td>Jovial tca</td>
<td>Logic</td>
</tr>
<tr>
<td>Logicflow</td>
<td>Mefia</td>
<td>Memory mng lib</td>
</tr>
<tr>
<td>Mentor</td>
<td>Metran</td>
<td>Nodal</td>
</tr>
<tr>
<td>Number</td>
<td>Number/dec</td>
<td>Oscybr</td>
</tr>
<tr>
<td>Pace</td>
<td>Pace-c</td>
<td>Pds</td>
</tr>
<tr>
<td>Perluette</td>
<td>Pet</td>
<td>Program generat</td>
</tr>
<tr>
<td>Quikcode</td>
<td>Ratcoder</td>
<td>Ratfor</td>
</tr>
<tr>
<td>Realignment sys</td>
<td>Reform</td>
<td>Rename</td>
</tr>
<tr>
<td>Rxvp80 (tm)</td>
<td>S-fortran</td>
<td>Sara</td>
</tr>
<tr>
<td>Schemacode</td>
<td>Scobol (tm)</td>
<td>Scope</td>
</tr>
<tr>
<td>Sfort-1</td>
<td>Sptran3</td>
<td>Lib</td>
</tr>
<tr>
<td>Smal/80</td>
<td>Sms</td>
<td>Softool 80 (tm)</td>
</tr>
<tr>
<td>Sptran</td>
<td>Srtran.baseline</td>
<td>Structuring eng</td>
</tr>
<tr>
<td>Struct1/struct2</td>
<td>Structurizer</td>
<td>Surge 72</td>
</tr>
<tr>
<td>System-80</td>
<td>Tab40</td>
<td>Tattle</td>
</tr>
<tr>
<td>Tcat</td>
<td>Tdem</td>
<td>Test predictor</td>
</tr>
<tr>
<td>Tfa</td>
<td>The engine</td>
<td>Toolpack</td>
</tr>
<tr>
<td>Tpt</td>
<td>Trailblazer</td>
<td>Transform</td>
</tr>
<tr>
<td>Virtual os</td>
<td>X to y trans pk</td>
<td>Yacc</td>
</tr>
</tbody>
</table>

#### BASIC

| Mefia          |

#### FORTRAN

| Sait           |

---
<table>
<thead>
<tr>
<th>Tools</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMGEN</td>
<td>CPAL</td>
</tr>
<tr>
<td>EAVS</td>
<td>FAVS</td>
</tr>
<tr>
<td>FORTRAN OPTIMIZ</td>
<td>FORTRAN (TM)</td>
</tr>
<tr>
<td>FTNCODER</td>
<td>IFTRAN</td>
</tr>
<tr>
<td>INSTRU</td>
<td>ISUS</td>
</tr>
<tr>
<td>LOGICFLOW</td>
<td>LOGIC</td>
</tr>
<tr>
<td>NODAL</td>
<td>NUMBER</td>
</tr>
<tr>
<td>PACE</td>
<td>PDS</td>
</tr>
<tr>
<td>RATFOR</td>
<td>REFORM</td>
</tr>
<tr>
<td>TVX80 (TM)</td>
<td>SCHEMACODE</td>
</tr>
<tr>
<td>SFTRAN3</td>
<td>SPTRAN</td>
</tr>
<tr>
<td>STRUC1/STRUC2</td>
<td>STRUCTURING ENG</td>
</tr>
<tr>
<td>TDEM</td>
<td>TEST PREDICTOR</td>
</tr>
<tr>
<td>TPT</td>
<td>TRAILBLAZER</td>
</tr>
<tr>
<td>VIRTUAL OS</td>
<td>X TO Y TRANS PK</td>
</tr>
<tr>
<td>COBOL</td>
<td>CADA</td>
</tr>
<tr>
<td>CAVS</td>
<td>CBLSHORT</td>
</tr>
<tr>
<td>COBOL/SPP</td>
<td>COBOL TRACING</td>
</tr>
<tr>
<td>COBOL/SP</td>
<td>COBOL TESTING</td>
</tr>
<tr>
<td>CPA</td>
<td>CQD</td>
</tr>
<tr>
<td>CTC</td>
<td>ENFORCE</td>
</tr>
<tr>
<td>NUMBER/DEC</td>
<td>QUIKCODE</td>
</tr>
<tr>
<td>SCOBOL (TM)</td>
<td>SURGE 72</td>
</tr>
<tr>
<td>TAB40</td>
<td>TCAT</td>
</tr>
<tr>
<td>THE ENGINE</td>
<td>TRAILBLAZER</td>
</tr>
<tr>
<td>COMPASS</td>
<td></td>
</tr>
<tr>
<td>PL/1</td>
<td></td>
</tr>
<tr>
<td>JOVIAL</td>
<td></td>
</tr>
<tr>
<td>EAVS</td>
<td>JAVS</td>
</tr>
<tr>
<td>JOVIAL TCA</td>
<td>LOGICFLOW</td>
</tr>
<tr>
<td>IFTRAN</td>
<td>RXVP80 (TM)</td>
</tr>
<tr>
<td>SMAL/80</td>
<td>SMAL/80</td>
</tr>
<tr>
<td>RATFOR</td>
<td>RATCODER</td>
</tr>
<tr>
<td>ALGOL</td>
<td>TAB40</td>
</tr>
<tr>
<td>SRTRAN</td>
<td>ISUS</td>
</tr>
<tr>
<td>ITB</td>
<td></td>
</tr>
<tr>
<td>SFTRAN</td>
<td></td>
</tr>
<tr>
<td>FOSTRA</td>
<td></td>
</tr>
<tr>
<td>SIMULA</td>
<td></td>
</tr>
<tr>
<td>ATOM</td>
<td></td>
</tr>
<tr>
<td>CMS-2</td>
<td></td>
</tr>
</tbody>
</table>
### Tools by Machine Output

<table>
<thead>
<tr>
<th>ASSEMBLY LANGUAGE</th>
<th>MAGLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>OBJECT CODE OUTPUT</td>
<td></td>
</tr>
<tr>
<td>FASP</td>
<td>CS4</td>
</tr>
<tr>
<td>OPTIMIZER II</td>
<td>MSEF</td>
</tr>
<tr>
<td>SMMA</td>
<td></td>
</tr>
<tr>
<td>DATA OUTPUT</td>
<td>CARA</td>
</tr>
<tr>
<td>CASEGEN</td>
<td></td>
</tr>
<tr>
<td>COBOL/DV</td>
<td></td>
</tr>
<tr>
<td>CS4</td>
<td></td>
</tr>
<tr>
<td>MENTOR</td>
<td></td>
</tr>
<tr>
<td>RTT</td>
<td></td>
</tr>
<tr>
<td>PROMPTS</td>
<td>SARA</td>
</tr>
<tr>
<td>VHLL OUTPUT</td>
<td></td>
</tr>
<tr>
<td>SRIMP</td>
<td></td>
</tr>
<tr>
<td>INTERMEDIATE CODE</td>
<td>ADA COMPILER</td>
</tr>
<tr>
<td>CHILL TRANS</td>
<td></td>
</tr>
<tr>
<td>JOCIT</td>
<td></td>
</tr>
<tr>
<td>CHECKSUM</td>
<td></td>
</tr>
<tr>
<td>COMSTAR</td>
<td></td>
</tr>
<tr>
<td>MARK IV (TM)</td>
<td></td>
</tr>
<tr>
<td>PWB FOR VAX/VMS</td>
<td></td>
</tr>
<tr>
<td>TOOLPACK</td>
<td></td>
</tr>
</tbody>
</table>

| OUTPUT                      |       |
| LILITH                      |       |
| OPTIMUS                     |       |
| XAS8                        |       |
| DATA OUTPUT                 |       |
| CASEGEN                     |       |
| COGENT                      |       |
| INFORM                      |       |
| PSL/PSA                     |       |
| SARA                        |       |
| AUTO-DBO                    |       |
| SRIMP                       |       |
| INTERMEDIATE CODE           | ADA COMPILER |
| CHILL TRANS                 |       |
| JOCIT                       |       |
| CHECKSUM                    |       |
| COMSTAR                     |       |
| MARK IV (TM)                |       |
| PWB FOR VAX/VMS             |       |
| TOOLPACK                    |       |
## APPENDIX H

### PORTABLE TOOLS

#### FULLY PORTABLE

<table>
<thead>
<tr>
<th>ADF</th>
<th>ARGUS/MICRO</th>
<th>ASA-PMS</th>
<th>AUDITOR</th>
</tr>
</thead>
<tbody>
<tr>
<td>BEST/1 (TM)</td>
<td>CAPTURE/MVS(TM)</td>
<td>CAVS</td>
<td>CBLSHORT</td>
</tr>
<tr>
<td>COBOL TESTING</td>
<td>COBOL/CP</td>
<td>COBOL TRACING</td>
<td>COBOL OPTIMIZ</td>
</tr>
<tr>
<td>COMMAPP</td>
<td>COPE (TM)</td>
<td>CSPP</td>
<td>DAVE</td>
</tr>
<tr>
<td>DECKBOY COMPAR</td>
<td>DI-3000</td>
<td>DIFFS (TM)</td>
<td>DOCUMENTER A</td>
</tr>
<tr>
<td>DYNA</td>
<td>ENFORCE</td>
<td>EXPEDITER</td>
<td>FORTRAN OPTIMIZ</td>
</tr>
<tr>
<td>FORTRAN TRACING</td>
<td>FORTRAN TESTING</td>
<td>FTN-77 ANALYZER</td>
<td>FTN ANALYZER</td>
</tr>
<tr>
<td>GOTO-ANALYZER</td>
<td>GRAFMAKER</td>
<td>HAWKEYE (TM)</td>
<td>IFTRAN (TM)</td>
</tr>
<tr>
<td>INTERFACE DOCUM</td>
<td>ISUS</td>
<td>ITB</td>
<td>MAGLE</td>
</tr>
<tr>
<td>MEMORY MNG LIB</td>
<td>METTRAN</td>
<td>N5500</td>
<td>NODAL</td>
</tr>
<tr>
<td>ONLINE ASSIST</td>
<td>P BASIC</td>
<td>PET</td>
<td>POD</td>
</tr>
<tr>
<td>PRONET</td>
<td>PSL/PSA</td>
<td>RATFOR</td>
<td>REFTRAN (TM)</td>
</tr>
<tr>
<td>REL MEAS MODEL</td>
<td>RXVP80 (TM)</td>
<td>SCOBOL (TM)</td>
<td>SDDL</td>
</tr>
<tr>
<td>SDP/MAYDA</td>
<td>SFTRAN3</td>
<td>SLIM</td>
<td>SOFTOOL 80 (TM)</td>
</tr>
<tr>
<td>SPTRAN</td>
<td>SREM</td>
<td>SRIMP</td>
<td>SRTRAN.BASLINE</td>
</tr>
<tr>
<td>STRUCTURIZER</td>
<td>TAB40</td>
<td>TAPS/AM</td>
<td>THE ENGINE</td>
</tr>
<tr>
<td>TOOLPACK</td>
<td>TRAILBLAZER</td>
<td>UCSD P-SYSTEM</td>
<td>VIRTUAL OS</td>
</tr>
<tr>
<td>X TO Y TRANS PK</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#### PARTIALLY PORTABLE

<table>
<thead>
<tr>
<th>AISIM</th>
<th>ARTS</th>
<th>DQM</th>
<th>PWB FOR VAX/VMS</th>
</tr>
</thead>
<tbody>
<tr>
<td>SCG</td>
<td>STRUCTURE(S)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
# APPENDIX I

## TOOLS BY SOURCE LANGUAGE

### BASIC

<table>
<thead>
<tr>
<th>CRISPFLOW</th>
<th>DOCUMENTER</th>
<th>HARDWARE SIMULA</th>
<th>MEFIA</th>
</tr>
</thead>
<tbody>
<tr>
<td>PROGRAM GENERAT</td>
<td>SLIB</td>
<td>SLIB</td>
<td>SLIM</td>
</tr>
<tr>
<td>STAT ENT &amp; EVAL</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### FORTRAN

<table>
<thead>
<tr>
<th>ABS</th>
<th>AISIM</th>
<th>ASEQ</th>
<th>ATTEST</th>
<th>AUTOCOM</th>
<th>AUTORETEST</th>
<th>CA</th>
<th>CARA</th>
<th>CENSUS</th>
<th>COBOL TRACING</th>
<th>COMLIST/TRW</th>
<th>COMSCAN</th>
<th>CONFIGURATOR</th>
<th>CS4</th>
<th>DECA</th>
<th>DICTANL/LOCATE</th>
<th>DOCUMENTOR</th>
<th>DRIVER</th>
<th>FACES</th>
<th>FLODIA</th>
<th>FORMAN</th>
<th>FORTRAN TRACING</th>
<th>FTN ANALYZER</th>
<th>HARP</th>
<th>INTERFACE DOCUM</th>
<th>LOGIC</th>
<th>MED-SYS</th>
<th>METRAN</th>
<th>NETWORK PLANNER</th>
<th>OSCYBR</th>
<th>PDS FLOW</th>
<th>PFORT</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADA COMPILER</td>
<td>ALIAS</td>
<td>ASAP</td>
<td>AUDIT</td>
<td>AUTOFLOW/TRW</td>
<td>BLKGEN/SPECNP</td>
<td>CADMUS</td>
<td>CASEGEN</td>
<td>CHECKSUM</td>
<td>COMDIM</td>
<td>COMLIST</td>
<td>COMSORT</td>
<td>CPAL</td>
<td>DAS</td>
<td>DECKBOY COMPAR</td>
<td>DIRCOM</td>
<td>DPAD</td>
<td>DYNA</td>
<td>FASP</td>
<td>FLOWGEN</td>
<td>FORMAN</td>
<td>FORTRAN TRACING</td>
<td>FTN ANALYZER</td>
<td>HARP</td>
<td>INTERFACE DOCUM</td>
<td>LOGIC</td>
<td>MED-SYS</td>
<td>METRAN</td>
<td>NETWORK PLANNER</td>
<td>OSCYBR</td>
<td>PDS FLOW</td>
<td>PFORT</td>
</tr>
<tr>
<td>Tools by Source Language</td>
<td>Page I-2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>--------------------------</td>
<td>---------</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Tools by Source Language</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>PREF HDR GEN</strong></td>
<td><strong>PROG COMP ANAL</strong></td>
<td><strong>PROGLOOK</strong></td>
<td><strong>PSL/PSA</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RADC/FCA</td>
<td>RATFOR</td>
<td>REFER</td>
<td>REFORM</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>REFTRAN (TM)</td>
<td>RENAME</td>
<td>RISOS TOOLS</td>
<td>RXVP80 (TM)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SALSIM</td>
<td>SARA-IV</td>
<td>SARA-H</td>
<td>SARA-U</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SARA-III</td>
<td>SCG</td>
<td>SCG/DQM</td>
<td>SCHEMACODE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SDP</td>
<td>SDP/MAYDA</td>
<td>SDVS</td>
<td>SFORMAT-1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SFTRAN3</td>
<td>SIGS</td>
<td>SLIM</td>
<td>SOFTOOL 80 (TM)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SPELL</td>
<td>SPREAD</td>
<td>SPRINT</td>
<td>SPTRAN</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SREM</td>
<td>SRIMP</td>
<td>SRTRAN.BASELINE</td>
<td>STRUCTURIZER</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>STRUCT</td>
<td>SUBCRS</td>
<td>SURYAVOR</td>
<td>SUS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SYMCRS</td>
<td>TAB40</td>
<td>TDEM</td>
<td>TEST PREDICTOR</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TIDY</td>
<td>TIMER</td>
<td>TOOLPACK</td>
<td>TPT</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TRAILBLAZER</td>
<td>TRANSFOR</td>
<td>TRWPLT</td>
<td>TSA/PPE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UCA</td>
<td>VIRTUAL OS</td>
<td>W TO Y TRANS PK</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**MACRO-11**

**IORL**

**COBOL**

<table>
<thead>
<tr>
<th>Tools</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADF</td>
<td>CADA</td>
</tr>
<tr>
<td>COBOL/SP</td>
<td>COBOL/SPP</td>
</tr>
<tr>
<td>COBOL TESTING</td>
<td>COBOL/DV</td>
</tr>
<tr>
<td>COTUNE II</td>
<td>CPA</td>
</tr>
<tr>
<td>CTC</td>
<td>CUE</td>
</tr>
<tr>
<td>DOCU/TEXT</td>
<td>EASTRLOL</td>
</tr>
<tr>
<td>HAWKEYE (TM)</td>
<td>LIBREF</td>
</tr>
<tr>
<td>PAC II</td>
<td>PPE</td>
</tr>
<tr>
<td>QUICK-DRAW</td>
<td>QUIKCODE</td>
</tr>
<tr>
<td>SCERT</td>
<td>SCOBOL (TM)</td>
</tr>
<tr>
<td>SNOOP</td>
<td>STRUCTURE(S)</td>
</tr>
<tr>
<td>TFA</td>
<td>THE ENGINE</td>
</tr>
</tbody>
</table>

**PASCAL**

<table>
<thead>
<tr>
<th>Tools</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARGUS/MICRO</td>
<td>AUTO-DBO</td>
</tr>
<tr>
<td>FTNXREF</td>
<td>GENTESTS</td>
</tr>
<tr>
<td>NASA-VATS</td>
<td>RATCODER</td>
</tr>
<tr>
<td>SREP</td>
<td>SYSXREF</td>
</tr>
<tr>
<td>UF</td>
<td>VIRTUAL OS</td>
</tr>
</tbody>
</table>

**ASSEMBLY**

<table>
<thead>
<tr>
<th>Tools</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASSIST-I</td>
<td>AUTOFLOW (TM)</td>
</tr>
<tr>
<td>CUE</td>
<td>FADEBUG-I</td>
</tr>
<tr>
<td>LOGIFLOW</td>
<td>MARK IV (TM)</td>
</tr>
<tr>
<td>PMS IV</td>
<td>PPE</td>
</tr>
<tr>
<td>REFLECT II</td>
<td>SDVS</td>
</tr>
<tr>
<td>SMT</td>
<td>SYDOC</td>
</tr>
</tbody>
</table>

**COMPASS**

<table>
<thead>
<tr>
<th>Tools</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABS</td>
<td>COMSTAR</td>
</tr>
<tr>
<td>PACE-C</td>
<td>PMCS</td>
</tr>
</tbody>
</table>
## Tools by Source Language

### PL/1
- **BSC**
- **JET**
- **SARA**
- **STRUCTURING ENG**
- **DARTS**
- **MEFIA**
- **SPEAR**
- **TPT**
- **EFFIGY**
- **PPP**
- **SRIMP**
- **GENTEXTS**
- **SADAT**
- **SSA**

### JOVIAL
- **CGJA**
- **JOVIAL TCA**
- **TATTLE**
- **JAVS**
- **JSDD**
- **TDBCOMP**
- **JOVIAL/J3SC**
- **SDVS**
- **JOVIAL/VS**
- **TAFIRM**

### IFTRAN
- **EAVS**
- **ISDS**

### SLEUTH
- **CADMUS**
- **CROREF**
- **DEPCHT**
- **FORREF**

### PWS

### GIM/GIM II
- **GIM/GIM II**
- **QUICK-DRAW**

### OBJECT

### STRUCTURED FORTRAN
- **DOCUMENT**
- **LOGOS**
- **SAP/H**
- **FAVS**
- **PDL**
- **SFTRAN3**
- **FOSTRA**
- **S-FORTRAN**
- **LEXICON**
- **SAP**
- **STRUC1/STRUC2**

### BAL
- **CALLREF**
- **DOSSIER**
- **COMSCAN**
- **EXPEDITER**
- **DA**
- **DATAMACS**

### RPG
- **DOCU/TEXT**
- **SPC**

### SMAL/80
- **SMAL/80**

### SIMSCRIPT
- **AISIM**
- **DAS**

### RATFOR
- **INSERT**
- **MONITOR**
- **VIRTUAL OS**
TOOLS BY SOURCE LANGUAGE

SNOBAL
  AMPIC
SALSIM
  DPAD
LISP
  AFFIRM  SELECT  TEVERE-1
ALC
  COTUNE II  SCAN/370
c
  MSEF  ONLINE ASSIST  PWB FOR VAX/VMS  XAS8
PLS
  YACC
DDPM
GIM
  STAG/TEMS
SYMPL
  JOCIT
-PFORT
PBASIC
SRTRAN
ISUS  ITB
MODULA
  LILITH
SIMULA
  GENTEXTS
SCOBOL
  DIFFS (TM)  SCOBOL (TM)
TOOLS BY SOURCE LANGUAGE

KCL
FASP
## APPENDIX J

### TOOLS BY HARDWARE

<table>
<thead>
<tr>
<th>IBM 360/370</th>
<th>DECSYSTEM-10/20</th>
<th>DATA GENERAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACT/1</td>
<td>AFFIRM</td>
<td>MODULE ORDERER</td>
</tr>
<tr>
<td>ARTS</td>
<td>MENTOR</td>
<td></td>
</tr>
<tr>
<td>CALLREF</td>
<td>SDVS</td>
<td></td>
</tr>
<tr>
<td>CICS DUMP ANALY</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMGEN</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CONFIG</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CPA-ADR</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DA</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DCD</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DOCUMENTOR</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EFFIGY</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FLOBOL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GADTR AID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>JIGSAW</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LOOK</td>
<td></td>
<td></td>
</tr>
<tr>
<td>OPTIMIZER II</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PDS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PROGLook</td>
<td></td>
<td></td>
</tr>
<tr>
<td>REFER</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SADAT</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SCAN/370</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SDP</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SPECLE/DARS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>STRUCTURING ENG</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SYDOC</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TPT</td>
<td></td>
<td></td>
</tr>
<tr>
<td>XPEDITOR</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### IBM 360/370

- ACT/1
- ARTS
- CALLREF
- CICS DUMP ANALY
- COMGEN
- CONFIG
- CPA-ADR
- DA
- DCD
- DOCUMENTOR
- EFFIGY
- FLOBOL
- GADTR AID
- JIGSAW
- LOOK
- OPTIMIZER II
- PDS
- PROGLook
- REFER
- SADAT
- SCAN/370
- SDP
- SPECLE/DARS
- STRUCTURING ENG
- SYDOC
- TPT
- XPEDITOR

### DECsystem-10/20

- AFFIRM
- MENTOR
- SDVS

### DATA GENERAL

- MODULE ORDERER

### Tools by Hardware

- IBM 360/370
  - ACT/1
  - ARTS
  - CALLREF
  - CICS DUMP ANALY
  - COMGEN
  - CONFIG
  - CPA-ADR
  - DA
  - DCD
  - DOCUMENTOR
  - EFFIGY
  - FLOBOL
  - GADTR AID
  - JIGSAW
  - LOOK
  - OPTIMIZER II
  - PDS
  - PROGLook
  - REFER
  - SADAT
  - SCAN/370
  - SDP
  - SPECLE/DARS
  - STRUCTURING ENG
  - SYDOC
  - TPT
  - XPEDITOR

- DECsystem-10/20
  - AFFIRM
  - MENTOR
  - SDVS

- DATA GENERAL
  - MODULE ORDERER
TOOLS BY HARDWARE

BURROUGHS B3500
  PAC II
  SEL 32
  PDL  S-FORTRAN

DEC PDP-11
  DOCUMENTER  FOSTRA
  LIBREF  MED-SYS
  NUMBER/DEC  ONLINE ASSIST
  QUIKCODE  RA
  SCG/DQM  SLIB
  YACC

BURROUGHS B6700
  TEVERE-1

TSS
  GIRAFF

TI
  SREP

INTEL 8080/8085
  SMAL/80  XAS8

MODCOMP
  PREF HDR GEN

IBM 3033

CADSAT

NO. 1 ESS

CCS

FACOM 230-60

FADEBUG-I

AMDAHL 470
  AISIM
  SARA-IV
  DDPM

DOM

QCM

SCG

SCG/DQM
### Tools by Hardware

#### ONYX

**PWB for VAX/VMS**

#### APPLE II

**AUTO-DBO**

#### LILITH

#### CDC CYBER

<table>
<thead>
<tr>
<th>AFS</th>
<th>ATA-FASP</th>
<th>AUTOIDEF0</th>
<th>AUTOMATIC DOCUM</th>
</tr>
</thead>
<tbody>
<tr>
<td>CADA</td>
<td>DAS</td>
<td>FAME</td>
<td>FASP</td>
</tr>
<tr>
<td>FTNXREF</td>
<td>MPS</td>
<td>PDS</td>
<td>PIDGIN-FASP</td>
</tr>
<tr>
<td>PMCS</td>
<td>SIGS</td>
<td>SYSXREF</td>
<td>TFA</td>
</tr>
</tbody>
</table>

#### DEC/GT4X

#### IORL

#### CDC 6X00/7X00

<table>
<thead>
<tr>
<th>ABS</th>
<th>ADS/CERL</th>
<th>ARTS</th>
<th>ASEQ</th>
</tr>
</thead>
<tbody>
<tr>
<td>ATA-FASP</td>
<td>AUDIT</td>
<td>AUTOCOM</td>
<td>AUTOIDEF0</td>
</tr>
<tr>
<td>BLKGEN-BDD</td>
<td>BUDGET VS ACTUAL</td>
<td>CA</td>
<td>CADMUS</td>
</tr>
<tr>
<td>CCA</td>
<td>COMDIM</td>
<td>COMGEN/TRW</td>
<td>COMSORT</td>
</tr>
<tr>
<td>COMSTAR</td>
<td>CPAL</td>
<td>DICTANL/LOCATE</td>
<td>DIRCOM</td>
</tr>
<tr>
<td>DOCUMENTOR</td>
<td>DPAD</td>
<td>DRIVER</td>
<td>ESAP</td>
</tr>
<tr>
<td>FASP</td>
<td>FLOBOL</td>
<td>FLODIA</td>
<td>FLOWGEN</td>
</tr>
<tr>
<td>FORAN</td>
<td>FORTREF</td>
<td>FTNCODER</td>
<td>FTNXREF</td>
</tr>
<tr>
<td>GADTR AID</td>
<td>GIRAFF</td>
<td>INFORM/REFORM</td>
<td>INSERT</td>
</tr>
<tr>
<td>JAVS</td>
<td>JIGSAW</td>
<td>JOYCE</td>
<td>LOGIC</td>
</tr>
<tr>
<td>LOGIFLOW</td>
<td>MONITOR</td>
<td>MPS</td>
<td>N-SQUARED</td>
</tr>
<tr>
<td>NASA-VATS</td>
<td>NODAL</td>
<td>PACE-C</td>
<td>PACE</td>
</tr>
<tr>
<td>PDL</td>
<td>PDS FLOW</td>
<td>PDSS</td>
<td>PERCAM</td>
</tr>
<tr>
<td>PFS</td>
<td>PIDGIN-FASP</td>
<td>PMCS</td>
<td>RATCODER</td>
</tr>
<tr>
<td>REFORM</td>
<td>RENAME</td>
<td>RISOS TOOLS</td>
<td>S-FORTTRAN</td>
</tr>
<tr>
<td>SALSIM</td>
<td>SIGS</td>
<td>SPELL</td>
<td>SPREAD</td>
</tr>
<tr>
<td>SPRINT</td>
<td>SREP</td>
<td>STRUCT</td>
<td>SUBCRS</td>
</tr>
<tr>
<td>SURVYOR</td>
<td>SYMCRS</td>
<td>SYSXREF</td>
<td>TEST PREDICTOR</td>
</tr>
<tr>
<td>TIDY</td>
<td>UCA</td>
<td>UF</td>
<td></td>
</tr>
</tbody>
</table>

#### XDS SIGMA X

#### ASSIST-I

**GIRAFF**

#### UNIVAC 11XX

<table>
<thead>
<tr>
<th>ARTS</th>
<th>ATDG</th>
<th>AUTODOC</th>
<th>BLKGEN/SPECPN</th>
</tr>
</thead>
<tbody>
<tr>
<td>CCREF</td>
<td>CENSUS</td>
<td>COBOL STRUCT</td>
<td>COMGEN</td>
</tr>
<tr>
<td>COMPARE</td>
<td>CRISPFLOW</td>
<td>CROREF</td>
<td>CS4</td>
</tr>
<tr>
<td>DATA DESIGNER</td>
<td>DEPCHT</td>
<td>DOCGEN</td>
<td>DOCUMENT</td>
</tr>
<tr>
<td>Tools</td>
<td>Hardware</td>
<td>Tools</td>
<td>Hardware</td>
</tr>
<tr>
<td>--------------</td>
<td>----------------</td>
<td>--------------</td>
<td>----------------</td>
</tr>
<tr>
<td>DPNDCY</td>
<td>FLOBOL</td>
<td>FORREF</td>
<td>FORTREF</td>
</tr>
<tr>
<td>GIM/GIM II</td>
<td>JIGSAW</td>
<td>LEXICON</td>
<td>LOGOS</td>
</tr>
<tr>
<td>NODAL</td>
<td>ONLINE ASSIST</td>
<td>PAC II</td>
<td>PACE</td>
</tr>
<tr>
<td>PDL</td>
<td>PDS</td>
<td>QUICK-DRAW</td>
<td>REALIGNMENT SYS</td>
</tr>
<tr>
<td>S-FORTRAN</td>
<td>SARA-U</td>
<td>SNOOP</td>
<td>STAG/TEMS</td>
</tr>
<tr>
<td>STRUCTURE(S)</td>
<td>TCAT</td>
<td>TDEM</td>
<td>TRWPLT</td>
</tr>
<tr>
<td><strong>Honeywell 6xxx</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ASRP</td>
<td>BSC</td>
<td>CARA</td>
<td>COBOL/SPP</td>
</tr>
<tr>
<td>EASYTROL</td>
<td>FAVS</td>
<td>FCA</td>
<td>FLOBOL</td>
</tr>
<tr>
<td>FORTREF</td>
<td>INSTRU</td>
<td>JOCIT</td>
<td>JOVIAL/J3SC</td>
</tr>
<tr>
<td>JSDD</td>
<td>PSL</td>
<td>QCRT</td>
<td>QUICK-DRAW</td>
</tr>
<tr>
<td>RADC/FCA</td>
<td>SARA-H</td>
<td>SARA</td>
<td>STRUCT1/STRUCT2</td>
</tr>
<tr>
<td>STRUCTURE(S)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Cdc 3xxx</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TAFIRM</td>
<td>TATTLE</td>
<td>TDBCOMP</td>
<td>TIMECS</td>
</tr>
<tr>
<td><strong>Honeywell 6xx</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FCA</td>
<td>FLOBOL</td>
<td>FORTREF</td>
<td>ISDS</td>
</tr>
<tr>
<td>JAVS</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Ec</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TPT</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Dec Vax-11</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ARTS</td>
<td>ATTEST</td>
<td>CS4</td>
<td>FAME</td>
</tr>
<tr>
<td>INFORM</td>
<td>MEDL-D</td>
<td>MEDL-R</td>
<td>MENTOR</td>
</tr>
<tr>
<td>PWB FOR VAX/VMS</td>
<td>SCG</td>
<td>SDDL</td>
<td>SLIB</td>
</tr>
<tr>
<td><strong>Ibm System 3</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SPC</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Hp 85</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PROGRAM GENERAT</td>
<td>SLIM</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>CII-Hb</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GENTESTS</td>
<td>GENTEXTS</td>
<td>MEFIA</td>
<td>MENTOR</td>
</tr>
<tr>
<td>PERLUETTE</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Plessey Micro I</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MEFIA</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## APPENDIX K

### TOOLS BY SOFTWARE

### OS

<table>
<thead>
<tr>
<th>Tool</th>
<th>Function</th>
<th>Tool</th>
<th>Function</th>
<th>Tool</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADS</td>
<td>CICS DUMP ANALY</td>
<td>COBOL/ADE</td>
<td>CORE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CPA-ADR</td>
<td>CTC</td>
<td>CUE</td>
<td>DA</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DOCU/TEXT</td>
<td>EASYTROL</td>
<td>FOCUS</td>
<td>JET</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PMS IV</td>
<td>PROGLOOK</td>
<td>QCM</td>
<td>OCRT</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SARA-III</td>
<td>SARA-IV</td>
<td>SCERT</td>
<td>SYDIM</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TAPS/AM</td>
<td>TRMS</td>
<td>TSA/PPE</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### SVS

<table>
<thead>
<tr>
<th>Tool</th>
<th>Function</th>
<th>Tool</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>CICS DUMP ANALY</td>
<td>FOCUS</td>
<td>LOOK</td>
<td>REFLECT II</td>
</tr>
<tr>
<td>SYDOC</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### VM/CMS

<table>
<thead>
<tr>
<th>Tool</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>EFFIGY</td>
<td>FAME</td>
</tr>
<tr>
<td></td>
<td>FOCUS</td>
</tr>
</tbody>
</table>

### TSO

<table>
<thead>
<tr>
<th>Tool</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS4</td>
<td>DAS</td>
</tr>
<tr>
<td></td>
<td>DDPM</td>
</tr>
<tr>
<td></td>
<td>FOCUS</td>
</tr>
<tr>
<td>SCG/DQM</td>
<td></td>
</tr>
</tbody>
</table>

### RT-11

<table>
<thead>
<tr>
<th>Tool</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>IORL</td>
<td></td>
</tr>
</tbody>
</table>

### OS/VS

<table>
<thead>
<tr>
<th>Tool</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACT/1</td>
<td>ADS</td>
</tr>
<tr>
<td></td>
<td>CADSAT</td>
</tr>
<tr>
<td></td>
<td>CICS DUMP ANALY</td>
</tr>
<tr>
<td></td>
<td>DA</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>COBOL/ADE</td>
<td>CORE</td>
</tr>
<tr>
<td></td>
<td>CUE</td>
</tr>
<tr>
<td></td>
<td>PMS IV</td>
</tr>
<tr>
<td></td>
<td>REFLECT II</td>
</tr>
<tr>
<td></td>
<td>SCERT</td>
</tr>
<tr>
<td></td>
<td>STRUCTURE(S)</td>
</tr>
<tr>
<td></td>
<td>TSA/PPE</td>
</tr>
<tr>
<td>DOCU/TEXT</td>
<td>FOCUS</td>
</tr>
<tr>
<td></td>
<td>LOOK</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>PROGLOOK</td>
<td>QCM</td>
</tr>
<tr>
<td></td>
<td>QCRT</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>SARA-IV</td>
<td>SARA-III</td>
</tr>
<tr>
<td></td>
<td>SCAN/370</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>SCOPE</td>
<td>SMT</td>
</tr>
<tr>
<td></td>
<td>SSA</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>SYDOC</td>
<td>TAPS/AM</td>
</tr>
<tr>
<td></td>
<td>TRMS</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### DOS

<table>
<thead>
<tr>
<th>Tool</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADS</td>
<td>CICS DUMP ANALY</td>
</tr>
<tr>
<td></td>
<td>CORE</td>
</tr>
<tr>
<td></td>
<td>CPA-ADR</td>
</tr>
<tr>
<td>Tools by Software</td>
<td></td>
</tr>
<tr>
<td>-------------------</td>
<td></td>
</tr>
<tr>
<td><strong>CTC</strong></td>
<td><strong>DA</strong></td>
</tr>
<tr>
<td><strong>OPTIMUS</strong></td>
<td><strong>QCRT</strong></td>
</tr>
<tr>
<td><strong>DOS/VS</strong></td>
<td></td>
</tr>
<tr>
<td><strong>ADS</strong></td>
<td><strong>CORE</strong></td>
</tr>
<tr>
<td><strong>DOSSIER</strong></td>
<td><strong>OPTIMUS</strong></td>
</tr>
<tr>
<td><strong>NOS</strong></td>
<td><strong>AUTOIDEFO</strong></td>
</tr>
<tr>
<td><strong>PTNXREF</strong></td>
<td><strong>NASA-VATS</strong></td>
</tr>
<tr>
<td><strong>EXEC 8</strong></td>
<td><strong>CS4</strong></td>
</tr>
<tr>
<td><strong>ARTS</strong></td>
<td><strong>STRUCTURE(S)</strong></td>
</tr>
<tr>
<td><strong>OS/MVS</strong></td>
<td></td>
</tr>
<tr>
<td><strong>ACT/1</strong></td>
<td><strong>AISIM</strong></td>
</tr>
<tr>
<td><strong>CUE</strong></td>
<td><strong>DARTS</strong></td>
</tr>
<tr>
<td><strong>DQM</strong></td>
<td><strong>FOCUS</strong></td>
</tr>
<tr>
<td><strong>LOOK</strong></td>
<td><strong>OSCYBR</strong></td>
</tr>
<tr>
<td><strong>REFLECT II</strong></td>
<td><strong>SCG</strong></td>
</tr>
<tr>
<td><strong>SPEAR</strong></td>
<td><strong>SYDIM</strong></td>
</tr>
<tr>
<td><strong>TSA/PPE</strong></td>
<td></td>
</tr>
<tr>
<td><strong>GCOS</strong></td>
<td><strong>COBOL/SPP</strong></td>
</tr>
<tr>
<td><strong>JOCIT</strong></td>
<td><strong>JOVIAL/J3SC</strong></td>
</tr>
<tr>
<td><strong>QCRT</strong></td>
<td><strong>RADC/FCA</strong></td>
</tr>
<tr>
<td><strong>STRUCTURE(S)</strong></td>
<td></td>
</tr>
<tr>
<td><strong>SCOPE 3.4</strong></td>
<td></td>
</tr>
<tr>
<td><strong>CADA</strong></td>
<td></td>
</tr>
<tr>
<td><strong>UNIX</strong></td>
<td></td>
</tr>
<tr>
<td><strong>MSEF</strong></td>
<td><strong>ONLINE ASSIST</strong></td>
</tr>
<tr>
<td><strong>XAS8</strong></td>
<td><strong>YACC</strong></td>
</tr>
<tr>
<td><strong>MASTER</strong></td>
<td></td>
</tr>
<tr>
<td><strong>TIMECS</strong></td>
<td></td>
</tr>
<tr>
<td><strong>MULTICS</strong></td>
<td></td>
</tr>
<tr>
<td><strong>BSC</strong></td>
<td><strong>GENTESTS</strong></td>
</tr>
<tr>
<td><strong>MENTOR</strong></td>
<td><strong>PERLUETTE</strong></td>
</tr>
</tbody>
</table>
TOOLS BY SOFTWARE

CDC 6600 SECRE
PMCS
VMS
ARTS MEDL-R SDDL
ATTEST MEDL-D SLIB
FAME MENTOR INFORM
INFORM PWB FOR VAX/VMS
SCP
SPC
OS/MVT
LOOK NUMBER REFLECT II
RSTS
DOCUMENTER HARDWARE SIMULA SLIB
INTERLISP
AFFIRM
4JS2
INSTRU
RSX-11
FOSTRA MEDL-R SAP
INFORM MEFIA SAP/H
LIBREF NUMBER/DEC SLIB
MED-SYS QUIKCODE
TOPS-10/20
CS4 TOOLS DATABASE
KRONOS
ATA-FASP SAP/H
FASP PIDGIN-FASP PMCS
SIRIS
GENTEXTS MENTOR PERLUETTE
IAS
INFORM
ECL
CENSUS COMPARE CRISPFLOW DOCUMENT
<table>
<thead>
<tr>
<th>APPENDIX L</th>
<th>TOOLS IN THE PUBLIC DOMAIN</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADA COMPILER</td>
<td>ADA ENVIRONMENT</td>
</tr>
<tr>
<td>AISIM</td>
<td>ASRP</td>
</tr>
<tr>
<td>AUDIT</td>
<td>BLKGEN-BDD</td>
</tr>
<tr>
<td>CGJA</td>
<td>CHILL TRANS</td>
</tr>
<tr>
<td>CQD</td>
<td>CRISPFLOW</td>
</tr>
<tr>
<td>DOCUMENTER</td>
<td>DOCUMENT</td>
</tr>
<tr>
<td>FTN-77 ANALYZER</td>
<td>FTN ANALYZER</td>
</tr>
<tr>
<td>JOVIAL TCA</td>
<td>JSDD</td>
</tr>
<tr>
<td>LOGOS</td>
<td>MAGLE</td>
</tr>
<tr>
<td>PIDGIN-FASP</td>
<td>PREF HDR GEN</td>
</tr>
<tr>
<td>RISOS TOOLS</td>
<td>SAP</td>
</tr>
<tr>
<td>SDDL</td>
<td>SFORT-1</td>
</tr>
<tr>
<td>SREM</td>
<td>STAT ENT &amp; EVAL</td>
</tr>
<tr>
<td>TIDY</td>
<td>TOOLPACK</td>
</tr>
<tr>
<td>VIRTUAL OS</td>
<td></td>
</tr>
</tbody>
</table>
# APPENDIX M

## TOOLS BY INFORMATION SOURCE

### TRW SOFTWARE TOOLS CATALOG

<table>
<thead>
<tr>
<th>TRW Software Tools Catalog</th>
<th>TRW Software Tools Catalog</th>
<th>TRW Software Tools Catalog</th>
<th>TRW Software Tools Catalog</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ABS</strong></td>
<td><strong>AMPIC</strong></td>
<td><strong>ASEQ</strong></td>
<td><strong>AUTOMOTIVE</strong></td>
</tr>
<tr>
<td><strong>ASSIST-I</strong></td>
<td><strong>AUTODOC</strong></td>
<td><strong>AUTOGRAPH</strong></td>
<td><strong>BUDGET VS ACTUAL</strong></td>
</tr>
<tr>
<td><strong>AUTOFLOW/TRW</strong></td>
<td><strong>BLKGEN/SPECPLAN</strong></td>
<td><strong>CABINET</strong></td>
<td><strong>CGR</strong></td>
</tr>
<tr>
<td><strong>CA</strong></td>
<td><strong>CALLREF</strong></td>
<td><strong>COMDIG</strong></td>
<td><strong>COMDIS</strong></td>
</tr>
<tr>
<td><strong>CCA</strong></td>
<td><strong>CHECKSUM</strong></td>
<td><strong>COMLIST</strong></td>
<td><strong>COMLIST</strong></td>
</tr>
<tr>
<td><strong>COMGEN/TRW</strong></td>
<td><strong>COMSTAR</strong></td>
<td></td>
<td><strong>CONFIG</strong></td>
</tr>
<tr>
<td><strong>COMSCAN</strong></td>
<td><strong>CROREF</strong></td>
<td></td>
<td><strong>DATAMACS</strong></td>
</tr>
<tr>
<td><strong>COTUNE II</strong></td>
<td><strong>DEPCHT</strong></td>
<td></td>
<td><strong>DICTAINT/LOCATE</strong></td>
</tr>
<tr>
<td><strong>DCD</strong></td>
<td><strong>DPM</strong></td>
<td></td>
<td><strong>DPNCY</strong></td>
</tr>
<tr>
<td><strong>DOCGEN</strong></td>
<td><strong>ESAP</strong></td>
<td></td>
<td><strong>EVP</strong></td>
</tr>
<tr>
<td><strong>DRIVER</strong></td>
<td><strong>FLOBOL</strong></td>
<td></td>
<td><strong>FLODIA</strong></td>
</tr>
<tr>
<td><strong>FACES</strong></td>
<td><strong>FORREF</strong></td>
<td></td>
<td><strong>FORTREF</strong></td>
</tr>
<tr>
<td><strong>FLOWGEN</strong></td>
<td><strong>HARP</strong></td>
<td></td>
<td><strong>INFORM/REFORM</strong></td>
</tr>
<tr>
<td><strong>GIM/GIM II</strong></td>
<td><strong>LIBRARIAN</strong></td>
<td></td>
<td><strong>LOGIC</strong></td>
</tr>
<tr>
<td><strong>JAVS</strong></td>
<td><strong>LOG N-SQUARED</strong></td>
<td></td>
<td><strong>NODAL</strong></td>
</tr>
<tr>
<td><strong>JAVS</strong></td>
<td><strong>MPS</strong></td>
<td></td>
<td><strong>PACE-C</strong></td>
</tr>
<tr>
<td><strong>LOGIFLOW</strong></td>
<td><strong>PAY</strong></td>
<td></td>
<td><strong>PDSS</strong></td>
</tr>
<tr>
<td><strong>OPTIMIZER II</strong></td>
<td><strong>PDS</strong></td>
<td></td>
<td><strong>PPE</strong></td>
</tr>
<tr>
<td><strong>PDL</strong></td>
<td><strong>PDS FLOW</strong></td>
<td></td>
<td><strong>S-FORTRAN</strong></td>
</tr>
<tr>
<td><strong>PERCAM</strong></td>
<td><strong>PF</strong></td>
<td></td>
<td><strong>SPELL</strong></td>
</tr>
<tr>
<td><strong>QUICK-DRAW</strong></td>
<td><strong>REF</strong></td>
<td></td>
<td><strong>STAG/TEMS</strong></td>
</tr>
<tr>
<td><strong>SALSIM</strong></td>
<td><strong>REFORM</strong></td>
<td></td>
<td><strong>SURVOR</strong></td>
</tr>
<tr>
<td><strong>SALSIM</strong></td>
<td><strong>SDVS</strong></td>
<td></td>
<td><strong>TDCCOMP</strong></td>
</tr>
<tr>
<td><strong>SPREAD</strong></td>
<td><strong>SREP</strong></td>
<td></td>
<td><strong>TRWPLT</strong></td>
</tr>
<tr>
<td><strong>STRUCTURING ENG</strong></td>
<td><strong>SUBCRRS</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>SYMCRS</strong></td>
<td><strong>TATTER</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>TDEM</strong></td>
<td><strong>TRANSFORM</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>UCA</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### COMPLETED SUBMISSION TO NBS

<table>
<thead>
<tr>
<th>COMPLETED SUBMISSION TO NBS</th>
<th>COMPLETED SUBMISSION TO NBS</th>
<th>COMPLETED SUBMISSION TO NBS</th>
<th>COMPLETED SUBMISSION TO NBS</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ADA COMPILER</strong></td>
<td><strong>ADS/CERL</strong></td>
<td><strong>ARTS</strong></td>
<td><strong>ASRP</strong></td>
</tr>
<tr>
<td><strong>ATTEST</strong></td>
<td><strong>AUDITOR</strong></td>
<td><strong>AUTOIDEF0</strong></td>
<td><strong>BEST/1 (TM)</strong></td>
</tr>
<tr>
<td><strong>BSC</strong></td>
<td><strong>CAPTURE/MVS(TM)</strong></td>
<td><strong>CAVI</strong></td>
<td><strong>CHILL TRANS</strong></td>
</tr>
<tr>
<td><strong>COBOL OPTIMIZ</strong></td>
<td><strong>COBOL/SPP</strong></td>
<td><strong>COBOL TESTING</strong></td>
<td><strong>COBOL TRACING</strong></td>
</tr>
<tr>
<td><strong>CONFIGURATOR</strong></td>
<td><strong>DARTS</strong></td>
<td><strong>DAS</strong></td>
<td><strong>DAVE</strong></td>
</tr>
<tr>
<td><strong>DDPM</strong></td>
<td><strong>DEXPEDITE</strong></td>
<td><strong>DOCUMENT</strong></td>
<td><strong>DOCUMENTER A</strong></td>
</tr>
<tr>
<td><strong>DYNA</strong></td>
<td><strong>DIT-3000</strong></td>
<td><strong>FAVS</strong></td>
<td><strong>FOCUS</strong></td>
</tr>
<tr>
<td><strong>FORTRAN OPTIMIZ</strong></td>
<td><strong>FORTRAN TRACING</strong></td>
<td><strong>FORTRAN TESTING</strong></td>
<td><strong>FTN-77 ANALYZER</strong></td>
</tr>
<tr>
<td><strong>FTN ANALYZER</strong></td>
<td><strong>FTNCODER</strong></td>
<td><strong>FTNXREF</strong></td>
<td><strong>GRAFMAKER</strong></td>
</tr>
</tbody>
</table>
## AIAA Survey of Soft Dev Tools

<table>
<thead>
<tr>
<th>Hardware</th>
<th>Simulation</th>
<th>Jocit</th>
<th>Lang Monitor</th>
<th>RADC/PCA</th>
<th>SCG/DQM</th>
<th>Struc1/Struc2</th>
<th>Tpt</th>
<th>Tools</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hardware Simula</td>
<td>HAWKEYE (TM)</td>
<td>JOVIAL/VS</td>
<td>SIMULA</td>
<td>HAWKEYE (TM)</td>
<td>JOVIAL/VS</td>
<td>JOVIAL TCA</td>
<td>JOVIAL/J3SC</td>
<td>INSERT</td>
</tr>
<tr>
<td>Jocit</td>
<td>JOVIAL/VS</td>
<td>JOVIAL/VS</td>
<td>JOVIAL/VS</td>
<td>JOVIAL/VS</td>
<td>JOVIAL/VS</td>
<td>JOVIAL/VS</td>
<td>JOVIAL/VS</td>
<td>INSERT</td>
</tr>
<tr>
<td>Lang Instructor</td>
<td>MAGLE</td>
<td>JOCIT</td>
<td>MAGLE</td>
<td>JOCIT</td>
<td>MAGLE</td>
<td>MAGLE</td>
<td>MAGLE</td>
<td>MAGLE</td>
</tr>
<tr>
<td>Monitor</td>
<td>NETWORK PLANNER</td>
<td>Magnet</td>
<td>NETWORK PLANNER</td>
<td>NETWORK PLANNER</td>
<td>NETWORK PLANNER</td>
<td>NETWORK PLANNER</td>
<td>NETWORK PLANNER</td>
<td>NETWORK PLANNER</td>
</tr>
<tr>
<td>RADC/PCA</td>
<td>RATCODER</td>
<td>JOVIAL/VS</td>
<td>RATCODER</td>
<td>RATCODER</td>
<td>RATCODER</td>
<td>RATCODER</td>
<td>RATCODER</td>
<td>RATCODER</td>
</tr>
<tr>
<td>SCG/DQM</td>
<td>SIGS</td>
<td>JOVIAL/VS</td>
<td>SIGS</td>
<td>SIGS</td>
<td>SIGS</td>
<td>SIGS</td>
<td>SIGS</td>
<td>SIGS</td>
</tr>
<tr>
<td>Struc1/Struc2</td>
<td>STRUCTURIZER</td>
<td>MAGLE</td>
<td>STRUCTURIZER</td>
<td>STRUCTURIZER</td>
<td>STRUCTURIZER</td>
<td>STRUCTURIZER</td>
<td>STRUCTURIZER</td>
<td>STRUCTURIZER</td>
</tr>
<tr>
<td>Tpt</td>
<td>TRAILBLAZER</td>
<td>JOVIAL/VS</td>
<td>TRAILBLAZER</td>
<td>TRAILBLAZER</td>
<td>TRAILBLAZER</td>
<td>TRAILBLAZER</td>
<td>TRAILBLAZER</td>
<td>TRAILBLAZER</td>
</tr>
<tr>
<td>Tools</td>
<td>INTERFACE DOCUM</td>
<td>INTERFACE DOCUM</td>
<td>INTERFACE DOCUM</td>
<td>INTERFACE DOCUM</td>
<td>INTERFACE DOCUM</td>
<td>INTERFACE DOCUM</td>
<td>INTERFACE DOCUM</td>
<td>INTERFACE DOCUM</td>
</tr>
</tbody>
</table>

### AIAA Survey of Soft Dev Tools

<table>
<thead>
<tr>
<th>ADS</th>
<th>AFS</th>
<th>ASA-PMS</th>
<th>AUTOMATIC DOCUM</th>
</tr>
</thead>
<tbody>
<tr>
<td>BLKGEN-BDD</td>
<td>CADA</td>
<td>CENSUS</td>
<td>CGJA</td>
</tr>
<tr>
<td>CICS DUMP ANALY</td>
<td>COGEN</td>
<td>COMPARE</td>
<td>CORE</td>
</tr>
<tr>
<td>CPA</td>
<td>CRISPFLOW</td>
<td>CUE</td>
<td>DA</td>
</tr>
<tr>
<td>DAVE</td>
<td>DIRCOM</td>
<td>DOCU/TEXT</td>
<td>DOCUMENT</td>
</tr>
<tr>
<td>DOSSIER</td>
<td>EASYTROL</td>
<td>FOSTRA</td>
<td>IORL</td>
</tr>
<tr>
<td>ISDS</td>
<td>JET</td>
<td>JSDD</td>
<td>LEXICON</td>
</tr>
<tr>
<td>LOGOS</td>
<td>MED-SYS</td>
<td>METTRAN</td>
<td>MSEE</td>
</tr>
<tr>
<td>N5500</td>
<td>NASA-VATS</td>
<td>OPTIMUS</td>
<td>PET</td>
</tr>
<tr>
<td>PFORT</td>
<td>PMCS</td>
<td>PMS IV</td>
<td>PEP</td>
</tr>
<tr>
<td>PREF HDR GEN</td>
<td>PROG COMP ANAL</td>
<td>PROGLook</td>
<td>PRONET</td>
</tr>
<tr>
<td>QCM</td>
<td>QCRT</td>
<td>REFLECT II</td>
<td>RENAME</td>
</tr>
<tr>
<td>SARA</td>
<td>SAP</td>
<td>SARAV-IV</td>
<td>SARA-III</td>
</tr>
<tr>
<td>SAP</td>
<td>SAP/H</td>
<td>SCERT</td>
<td>SCOPE</td>
</tr>
<tr>
<td>SARAV-IV</td>
<td>SFTRAN3</td>
<td>SLIM</td>
<td>SMAL80</td>
</tr>
<tr>
<td>SFTRAN-1</td>
<td>SM</td>
<td>SPC</td>
<td>SPEAR</td>
</tr>
<tr>
<td>SSA</td>
<td>SYDIM</td>
<td>TAPS/AM</td>
<td>TFA</td>
</tr>
<tr>
<td>TIDY</td>
<td>TIMECS</td>
<td>TRMS</td>
<td>TSA/PPE</td>
</tr>
<tr>
<td>UF</td>
<td>XAS8</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### RAD-TR-80-13, Interim Report

<table>
<thead>
<tr>
<th>Asset</th>
<th>ATDG</th>
<th>ATTEST</th>
<th>CASEGEN</th>
</tr>
</thead>
<tbody>
<tr>
<td>CCS</td>
<td>CSPP</td>
<td>DAVE</td>
<td>EFIGG</td>
</tr>
<tr>
<td>FACES</td>
<td>FADEBUG-I</td>
<td>FAST</td>
<td>FAVS</td>
</tr>
<tr>
<td>FORAN</td>
<td>JAVS</td>
<td>JOYCE</td>
<td>LIBRARIAN</td>
</tr>
<tr>
<td>Optimizer II</td>
<td>PACE</td>
<td>PFORT</td>
<td>REL MEAS MODEL</td>
</tr>
<tr>
<td>SEF</td>
<td>SELECT</td>
<td>SPTRAN</td>
<td>SUS</td>
</tr>
<tr>
<td>System Monitor</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Product Description

<table>
<thead>
<tr>
<th>ACT/1</th>
<th>ADA-ATOM</th>
<th>ASC</th>
<th>CBLSHORT</th>
</tr>
</thead>
<tbody>
<tr>
<td>COBOL/SP</td>
<td>COBOL/ADE</td>
<td>COBOL/QDM</td>
<td>COBOL/DV</td>
</tr>
<tr>
<td>COBOL/CP</td>
<td>CPA-ADR</td>
<td>CRYSTAL (TM)</td>
<td>ECA AUTOMATION</td>
</tr>
<tr>
<td>ENFORCE</td>
<td>INFORM</td>
<td>LOOK</td>
<td>MEDL-R</td>
</tr>
<tr>
<td>MED-L-P</td>
<td>MEDL-X</td>
<td>MEDL-D</td>
<td>REFTRAN (TM)</td>
</tr>
<tr>
<td>SPECTRUM-1</td>
<td>SRTRAN BASELINE</td>
<td>SYDOC</td>
<td>SYSTEM-80</td>
</tr>
<tr>
<td>TCAT</td>
<td>XPEDITOR</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Tool Fair

<table>
<thead>
<tr>
<th>TOOL</th>
<th>FAIR</th>
</tr>
</thead>
<tbody>
<tr>
<td>AFFIRM</td>
<td>AISIM</td>
</tr>
<tr>
<td>COMMAP</td>
<td>CS4</td>
</tr>
</tbody>
</table>
## TOOLS BY INFORMATION SOURCE

<table>
<thead>
<tr>
<th>Information Source</th>
<th>Tool</th>
<th>Tool Description</th>
<th>Category</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>FAME</td>
<td>FTN-77 Analyzer</td>
<td>IFTRAN (TM)</td>
<td>INSTRUMENTS</td>
<td></td>
</tr>
<tr>
<td>ISUS</td>
<td>ITB</td>
<td>LILITH</td>
<td>LOGICFLOWS</td>
<td></td>
</tr>
<tr>
<td>MSEF</td>
<td>ONLINE ASSIST</td>
<td>POD</td>
<td>PSL/PSA</td>
<td></td>
</tr>
<tr>
<td>PWB FOR VAX/VMS</td>
<td>RXVP80 (TM)</td>
<td>SARA</td>
<td>SCG</td>
<td></td>
</tr>
<tr>
<td>SCHEMACODE</td>
<td>SDDL</td>
<td>SDP/MAYDA</td>
<td>SLIM</td>
<td></td>
</tr>
<tr>
<td>SOFTOOL 80 (TM)</td>
<td>SREM</td>
<td>SRIMP</td>
<td>THE ENGINE</td>
<td></td>
</tr>
<tr>
<td>TOOLS DATABASE</td>
<td>UCSD P-SYSTEM</td>
<td>VIRTUAL OS</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### BCS SOFTWARE TOOLS CATALOG
- DECKBOY COMPAR
- GADTR AID
- LAYOUT
- SPECLE/DARS

### FEDERAL SOFT EXCHANGE CATALOG
- COBOL STRUCT
- REALIGNMENT SYS
- FTN ANALYZER
- TIMER
- NUMBER
- OSCYBR

### NTIS
- CQD
- SURGE 72
- TAB40

### ADI/CNRS CATALOGUE 1980
- GENTESTS
- GENETEXTS
- MEFIA
- MENTOR

### NOSC SEATECS TOOLS SURVEY
- ADF
- CADSAT
- DIFFS (TM)
- MSL
- QUIKCODE
- SDL
- ATA-FASP
- COPE (TM)
- FASP
- MTR
- RA
- SLIB
- ATA-SAI
- CTC
- LIBREF
- NUMBER/DEC
- RTT
- SMS
- AUTOFLOW (TM)
- DATA DESIGNER
- MARK IV (TM)
- PIDGIN-FASP
- SCOBOL (TM)
- YACC

### TECHNICAL LITERATURE
- ADA ENVIRONMENT
- GOTO-ANALYZER
- RISOS TOOLS
- AUDIT
- IPDS
- TEVERE-1
- CRYSTAL (TM)
- PBASIC
- TOOLTIPACK
- FASP
- RATFOR
APPENDIX N

TOOL ABSTRACTS

The following is brief definition of each of the possible data elements that can be stored in the database for each tool. This is followed by the data that is currently contained in the database for each tool. If a data element does not appear in the information provided for a tool, it is because no information was available for that element.

ACRONYM: Acronym or other short name.

TITLE: Title of tool.

CLASSIFICATION: One of the following:
- Software Management, Control, and Maintenance
- Software Modeling and Simulation
- Requirements/Design Specification and Analysis
- Program Construction and Generation
- Source Program Analysis and Testing
- Software Support System/Programming Environment.

FEATURES: Features provided by the tool ordered according to the taxonomy of tool features.

STAGE OF DEVELOPMENT: One of the following:
- Concept
- Design
- Implemented.

DATE (YYMMDD): YY - year, MM - month, DD - day of development.

IMPLEMENTATION LANGUAGE: The language(s) and dialect(s) in which the tool is written.

TOOL PORTABLE: Whether or not the tool can be transported to other machines.

TOOL SIZE: Number of source statements, disk size, or core memory size.
COMPUTER (OTHER HARDWARE): Hardware manufacturer/identification of machine necessary for use of the tool (Other machine hardware necessary for use of the tool).

OS (OTHER SOFTWARE): Operating system/release necessary for use of the tool. (Other software necessary for use of the tool).

TOOL AVAILABLE: Whether or not the tool can be made available to other potential users.

PUBLIC DOMAIN: Whether or not the tool is in the public domain.

RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): Restrictions on the availability of the tool.

TOOL SUPPORTED: Whether or not the tool is supported.

TOOL SUPPORT: Who supports the tool.

TOOL SUMMARY: Brief, one-paragraph summary of the tool. Clarification of features that the tool provides. Discussion of experience with the tool, such as performance characteristics, projects it was developed or used for, or any other pertinent information.

DOCUMENTATION: Types of available documentation.

DOC LENGTH: Extent of available documentation (page count).

REFERENCES: Articles or publications that discuss the tool and are readily available in the open literature, KEY (AUTHYYC), AUTHOR(S), TITLE, PUBLICATION, DATE.

DEVELOPER: Developer(s) of tool.

CONTACT: Contact(s) for more information about the tool.

INFORMATION SOURCE: Source(s) of the information contained in the database.
ACRONYM: ABS, TITLE: ABSTRACT RETRIEVAL PROGRAM
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE
FEATURES: SUBJECT, DATA INPUT, TRANSFORMATION, EDITING, USER OUTPUT, LISTINGS, STATIC ANALYSIS, MANAGEMENT, CONFIGURATION MANAGEMENT, TEST DATA MANAGEMENT, LIBRARY MANAGEMENT,
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN, COMPASS
COMPUTER (OTHER HARDWARE): CDC 6600/7600
TOOL SUMMARY: THE ABSTRACT RETRIEVAL PROGRAM WILL PROVIDE A MEANS OF MAINTAINING A FILE OF DOCUMENTS AND/OR ANY INFORMATION THAT IS CORRELATED BY SOME IDENTIFICATION CODE (KEYWORD). THE PROGRAM WILL ALLOW FOR THE UPDATING OR EDITING OF THESE FILES AS WELL AS VARIOUS TYPES OF RETRIEVAL LISTINGS. THE ABSTRACT RETRIEVAL PROGRAM IS A VERY SIMPLE KEYWORD SYSTEM. ITS POTENTIAL APPLICATIONS EXTEND FAR BEYOND STORAGE OF DOCUMENT OR PROGRAM ABSTRACTS. FOR EXAMPLE, A KEYWORD SYSTEM CAN BE USED TO CORRELATE TEST REQUIREMENTS STATEMENTS WITH TEST CASE NUMBERS, OR TO CORRELATE PROPOSAL PARAGRAPH NUMBERS WITH SAP/SM PARAGRAPH NUMBERS FOR REQUIREMENTS TRACINGABILITY. ALTHOUGH A LRM-TYPE SYSTEM MAY BE REQUIRED FOR MORE COMPLICATED INDEXING AND STORAGE REQUIREMENTS THE ABSTRACT RETRIEVAL PROGRAM IS A SIMPLE, INEXPENSIVE APPROACH TO SOLVING A CLASS OF STRAIGHTFORWARD KEYWORD PROBLEMS. (UNLIKE LRM, THE PROGRAM IS NOT AN ONLINE KEYWORD SYSTEM, THIS MAY BE A SIGNIFICANT LIMITATION FOR CERTAIN APPLICATIONS.)
DOCUMENTATION: PROGRAM DESCRIPTION REFERENCES: [108579], APPLIED SYSTEMS DESIGN SECTION, TRW DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS CATALOG AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE TOOLS SERIES, 790100
DEVELOPER: TRW, DATA MANAGEMENT SYSTEMS DEPARTMENT CONTACT: MILT HAYASHIDA, TRW, DATA MANAGEMENT SYSTEMS DEPARTMENT, ONE SPACE PARK, REDONDO BEACH, CA, 90278, USA, 213-535-2910
INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG
ACRONYM: ACT/1, TITLE: AN ONLINE APPLICATION FOR DEVELOPING ONLINE APPLICATIONS
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE
FEATURES: SUBJECT, TEXT INPUT, DATA INPUT, TRANSFORMATION, EDITING, USER OUTPUT, USER-ORIENTED TEXT, DOCUMENTATION, LISTINGS, STATIC ANALYSIS, MANAGEMENT, FILES MANAGEMENT, DOCUMENTATION MANAGEMENT, CONTROL, INTERACTIVE,
STAGE OF DEVELOPMENT: IMPLEMENTED
TOOL PURPOSE: NO, TOOL SIZE: FENER THAN 6 CYL ON 3330 DISK COMPUTER (OTHER HARDWARE): IBM 360/370 (IBM 3270 DISPLAY TERMINALS OR EQUIVALENT)
OS (OTHER SOFTWARE): OS/VS (TSO OR CICS, VSAM), OS/VS6 (TSO OR CICS, VSAM)

ACRONYM: ACT/1, TITLE: AN ONLINE APPLICATION FOR DEVELOPING ONLINE APPLICATIONS
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE
FEATURES: SUBJECT, TEXT INPUT, DATA INPUT, TRANSFORMATION, EDITING, USER OUTPUT, USER-ORIENTED TEXT, DOCUMENTATION, LISTINGS, STATIC ANALYSIS, MANAGEMENT, FILES MANAGEMENT, DOCUMENTATION MANAGEMENT, CONTROL, INTERACTIVE,
STAGE OF DEVELOPMENT: IMPLEMENTED
TOOL PURPOSE: NO, TOOL SIZE: FENER THAN 6 CYL ON 3330 DISK COMPUTER (OTHER HARDWARE): IBM 360/370 (IBM 3270 DISPLAY TERMINALS OR EQUIVALENT)
OS (OTHER SOFTWARE): OS/VS (TSO OR CICS, VSAM), OS/VS6 (TSO OR CICS, VSAM)

TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): MARKETED PRODUCT
TOOL SUPPORTED: YES, TOOL SUPPORT: APPLICATION DEVELOPMENT METHODOLOGIES (ADM) LTD.
TOOL SUMMARY: ACT/1 IS USED TO DEVELOP AND MANAGE AN ONLINE APPLICATION THROUGHOUT ITS ENTIRE LIFE CYCLE. ACT/1 PROVIDES INTEGRATED DEVELOPMENT AND PRESERVES FUNCTIONAL INTEGRITY DURING MAINTENANCE. ACT/1 DOES THE FOLLOWING: (1) PERMITS THE INTERACTIVE DESIGN AND CONSTRUCTION OF APPLICATION SCREENS BY SIMPLY KEYING THE LAYOUT DIRECTLY ON THE DISPLAY TERMINAL, (2) PERMITS THE INTERACTIVE SPECIFICATION OF AN APPLICATION DIALOGUE, (3) PERMITS DEVELOPERS TO PROVIDE APPLICATION ROUTINES IN STANDARD LANGUAGES, (4) PROVIDES FOR THE NAMING OF FIELDS ON SCREENS, (5) SUPPORTS APPLICATION SIMULATIONS ("SCENARIOS" AND "DEMO"), (6) PRODUCES PRINTED DOCUMENTATION, (7) MAINTAINS SCREENS, FLOW LOGIC, AND APPLICATION ROUTINES INDEPENDENTLY, (8) AND DIRECTS THE EXECUTION OF THE APPLICATION.
DOCUMENTATION: USER'S GUIDE, INSTALLATION GUIDE DEVELOPER: ART BENJAMIN ASSOCIATES LTD.
CONTACT: K. E. WOODTON, ART BENJAMIN ASSOCIATES LTD., 250 CONSUMERS AVE., 4TH FLOOR, WILLOWDALE, ONTARIO, CANADA, 416-494-9570
INFORMATION SOURCE: PRODUCT DESCRIPTION

ACRONYM: ADA-ATOM, TITLE: THE ADA-ATOM MACHINE
CLASSIFICATION: PROGRAM CONSTRUCTION AND GENERATION
FEATURES: SUBJECT, CODE INPUT, ADA, TRANSFORMATION, TRANSLATION, COMPILED, MACHINE OUTPUT, SOURCE CODE, OUTPUT, ATOM,
STAGE OF DEVELOPMENT: DESIGN
TOOL AVAILABLE: NO
TOOL SUMMARY: THE ADA-ATOM MACHINE WILL BE A HARDWARE INTERPRETER THAT WILL BE DESIGNED USING A BIT SLICE PROCESSOR TO INTERPRET THE ADA-ATOM LANGUAGE. AN ADA COMPILER IS CURRENTLY BEING DESIGNED THAT WILL HAVE A TARGET LANGUAGE CALLED ATOM. THE LANGUAGE ATOM DERIVES ITS COMPLETE SPECIFICATION FROM ATTRIBUTED ACTION SYMBOLS WHICH ARE ADDED TO THE INPUT GRAMMAR SO THAT THE RECOGNISER CORRESPONDING TO THE INPUT GRAMMAR IS TRANSFORMED INTO A TRANSLATOR WHO'S OUTPUT IS A STRING OF ATOMS - THE NAME ADA-ATOM. THE COMPILER WILL HAVE AN INSERT CORRECTABLE LINE CFG FOR ADA AS THE INPUT GRAMMAR UPON WHICH THE SEMANTIC ACTION SYMBOLS FOR ADA WILL BE EMBEDDED IN ORDER TO YIELD THE REQUIRED ATOM LANGUAGE.
DEVELOPER: ARNOLD J. LEVY
CONTACT: ARNOLD J. LEVY, APPLICATION SOFTWARE, 5TH FLOOR, 14579 BOARD OF TRUST, 5TH SIMMONDS ST, JOHANNESBURG, 2001, SOUTH AFRICA, 011)383-3451
INFORMATION SOURCE: PRODUCT DESCRIPTION
ACRONYM: ADA COMPILER, TITLE: ADA COMPILER
CLASSIFICATION: PROGRAM CONSTRUCTION AND GENERATION
FEATURES: SUBJECT, CODE INPUT, ADA, TRANSLATION, TRANSLATION, COMPILETION, MACHINE OUTPUT, INTERMEDIATE CODE, USER OUTPUT, LISTINGS.
STAGE OF DEVELOPMENT: IMPLEMENTED, DATE (YYMMDD): 801200
IMPLEMENTATION LANGUAGE: PORTMAN IV
TOOL SIZE: 936 STATEMENTS
TOOL AVAILABLE: YES, PUBLIC DOMAIN: YES
TOOL SUPPORTED: YES, TOOL SUPPORT: UNIVERSITY OF WAIKATO
TOOL SUMMARY: THE ADA COMPILER IS AN INCREMENTAL INTERACTIVE ADA PROGRAM TRANSLATOR, IT READS ONE STATEMENT AT A TIME AND TRANSFORMS IT INTO THE UNOPTIMISED: OUTPUT CODE. THE COMPILER USES A SYNTAX ANALYSER TO PRODUCE A CONDENSED REVERSED POLISH REPRESENTATION OF EACH STATEMENT, A PASS THROUGH THE REVERSED POLISH NOTATION IS REQUIRED TO GENERATE THE TARGET C.
DOCUMENTATION: TECHNICAL DESCRIPTION (20)
REFERENCES: [PAYNE73], A.J. PAYNE, "SAPS A CRITICAL REVIEW", SPERRY UNIVAC LDC, 730000
DEVELOPER: A.J. PAYNE
CONTACT: A.J. PAYNE, UNIVERSITY OF WAIKATO, HAMILTON, NEW ZEALAND.
INFORMATION SOURCE: COMPLETED SUBMISSION TO NBS

ACRONYM: ADA ENVIRONMENT, TITLE: ADA PROGRAMMING SUPPORT
CLASSIFICATION: SOFTWARE SUPPORT SYSTEM/PROGRAMMING LANGUAGE
FEATURES: SUBJECT, CODE INPUT, ADA, USER OUTPUT, DIAGNOSTICS, LISTINGS, STATIC ANALYSIS, MANAGEMENT, CONFIGURATION MANAGEMENT, ERROR CHECKING, DYNAMIC ANALYSIS, TRACING.
STAGE OF DEVELOPMENT: DESIGN, DATE (YYMMDD): 840000
TOOL AVAILABLE: NO, PUBLIC DOMAIN: YES

ACRONYM: ADF, TITLE: AUTOMATED DESIGN FACILITY
CLASSIFICATION: REQUIREMENTS/DESIGN SPECIFICATION AND ANALYSIS
FEATURES: SUBJECT, TEXT INPUT, VHDL INPUT, DESIGN SPECIFICATION, USER OUTPUT, GRAPHICS, LISTINGS, STATIC ANALYSIS, DATA FLOW ANALYSIS, STRUCTURE CHECKING.
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: COBOL
TOOL PORTABLE: YES, TOOL SIZE: 128 K BYTES
TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): RESTRICTED RIGHTS
TOOL SUPPORTED: YES, TOOL SUPPORT: M. BRYCE ASSOCIATES, INC.
TOOL SUMMARY: THE USER DEFINES REQUIREMENTS AND THEN USES THE ADF TO PERFORM SYSTEM DESIGN. ADF PERFORMS DATA FLOW ANALYSIS, EVALUATES ALL DATA TRANSFORMATIONS, REVIEWS DATA STRUCTURES AND TAKES INTO ACCOUNT INTERFACES WITH OTHER SYSTEMS. USING THE USERS ENVIRONMENT, ADF FULLY SUPPORTS THE "PRIDE" = ASD AUTOMATED SYSTEMS DESIGN METHODOLOGY AND AUTOMATES MANY OF ITS FEATURES. AS AN OPTION TO THE ENVIRONMENT, ADF PROVIDES FOR OUTPUT ANALYSIS AND AUTOMATIC GENERATION OF DESIGN DOCUMENTATION.
DOCUMENTATION: USER'S MANUAL
REFERENCES: [REIFER81], D. J. REIFER AND M. A. MONTGOMERY, "SEATECS SOFTWARE TOOLS SURVEY", RC1-TR-008, REIFER CONSULTANTS, INC., 810330
DEVELOPER: M. BRYCE ASSOCIATES, INC.
CONTACT: M. BRYCE ASSOCIATES, INC., 1248 SPRINGFIELD PIKE, CINCINNATI, OH, 45215, USA, 513-761-8300
INFORMATION SOURCE: NO MSC SEATECS TOOLS SURVEY

ACRONYM: ADB, TITLE: ADVANCED DEBUGGING SYSTEM
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, CICS, USER OUTPUT, DIAGNOSTICS, DYNAMIC ANALYSIS, TRACING, BREAKPOINT CONTROL.
STAGE OF DEVELOPMENT: IMPLEMENTED
TOOL SIZE: CORE: 15K
COMPUTER (OTHER HARDWARE): IBM 360/370
OS (OTHER SOFTWARE): OS, OSVS, DOS, DOS/VS
TOOL LICENSED: YES
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): FOR SALE

ACRONYM: STEN61, VIC STENNING, ET AL., "THE ADA ENVIRONMENT: A PERSPECTIVE", COMPUTER, 810300
REFERENCES: (DO0080), US DEPARTMENT OF DEFENSE, "REQUIREMENTS FOR ADA PROGRAMMING SUPPORT ENVIRONMENTS: "STONEMAN"", OFFICE OF THE UNDERSECRETARY OF DEFENSE, WASHINGTON, DC, 80020
CONTACT: PETE FONASH, ADA JOINT PROGRAM OFFICE, 801 RANDOLPH ST., ARLINGTON, VA, 22303, USA, 202-896-4387
VANCE MAIL: ADA JOINT PROGRAM OFFICE, 801 RANDOLPH ST., ARLINGTON, VA, 22203, USA, 202-896-4387
INFORMATION SOURCE: TECHNICAL LITERATURE

ACRONYM: ADF, TITLE: AUTOMATED DESIGN FACILITY
CLASSIFICATION: REQUIREMENTS/DESIGN SPECIFICATION AND ANALYSIS
FEATURES: SUBJECT, TEXT INPUT, VHDL INPUT, DESIGN SPECIFICATION, USER OUTPUT, GRAPHICS, LISTINGS, STATIC ANALYSIS, DATA FLOW ANALYSIS, STRUCTURE CHECKING.
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: COBOL
TOOL PORTABLE: YES, TOOL SIZE: 128 K BYTES
TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): RESTRICTED RIGHTS
TOOL SUPPORTED: YES, TOOL SUPPORT: M. BRYCE ASSOCIATES, INC.
TOOL SUMMARY: THE USER DEFINES REQUIREMENTS AND THEN USES THE ADF TO PERFORM SYSTEM DESIGN. ADF PERFORMS DATA FLOW ANALYSIS, EVALUATES ALL DATA TRANSFORMATIONS, REVIEWS DATA STRUCTURES AND TAKES INTO ACCOUNT INTERFACES WITH OTHER SYSTEMS. USING THE USERS ENVIRONMENT, ADF FULLY SUPPORTS THE "PRIDE" = ASD AUTOMATED SYSTEMS DESIGN METHODOLOGY AND AUTOMATES MANY OF ITS FEATURES. AS AN OPTION TO THE ENVIRONMENT, ADF PROVIDES FOR OUTPUT ANALYSIS AND AUTOMATIC GENERATION OF DESIGN DOCUMENTATION.
DOCUMENTATION: USER'S MANUAL
REFERENCES: [REIFER81], D. J. REIFER AND M. A. MONTGOMERY, "SEATECS SOFTWARE TOOLS SURVEY", RC1-TR-008, REIFER CONSULTANTS, INC., 810330
DEVELOPER: M. BRYCE ASSOCIATES, INC.
CONTACT: M. BRYCE ASSOCIATES, INC., 1248 SPRINGFIELD PIKE, CINCINNATI, OH, 45215, USA, 513-761-8300
INFORMATION SOURCE: NO MSC SEATECS TOOLS SURVEY

ACRONYM: ADB, TITLE: ADVANCED DEBUGGING SYSTEM
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, CICS, USER OUTPUT, DIAGNOSTICS, DYNAMIC ANALYSIS, TRACING, BREAKPOINT CONTROL.
STAGE OF DEVELOPMENT: IMPLEMENTED
TOOL SIZE: CORE: 15K
COMPUTER (OTHER HARDWARE): IBM 360/370
OS (OTHER SOFTWARE): OS, OSVS, DOS, DOS/VS
TOOL LICENSED: YES
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): FOR SALE
TOOL SUMMARY: ADS (ADVANCED DEBUGGING SYSTEM) IS DESIGNED TO MONITOR AND DEBUG APPLICATION SYSTEMS WRITTEN FOR IBM CICS. SOME OF THE MAJOR ADS FEATURES ARE PROVISION OF A KEYWORD/COMMAND LANGUAGE TO ALLOW ACCESS TO CICS RESOURCE AND INCORPORATION OF A LANGUAGE STRUCTURE SIMILAR TO COBOL. IT ALLOWS A PROGRAMMER TO DYNAMICALLY INSERT SOURCE STATEMENTS INTO EXISTING LOAD MODULES WITHOUT HAVING TO INTERPRET THE OBJECT INSTRUCTIONS. ADS CAN MONITOR THE EXECUTION OF A SELECTED ROUTINE, OR THE WHOLE PATH OF A TRANSACTION TO DETECT AND PREVENT AN ERROR CAUSED BY ILLEGAL FREEWAY EXECUTIONS, BAD FILE OR TERMINAL WRITES, ALTERATION OF CONTROL AREAS, OR LOSS OF ADDRESSABILITY, AND IT WILL SUSPEND THE OFFENDING TASK BEFORE DAMAGE OCCURS. CONTROL OF THE TASK IS GIVEN TO THE PROGRAMMER AT ANY DESIGNATED TERMINAL. THE SYSTEM CAN IDENTIFY STORAGE ADDRESSES WITHIN CICS AND GIVE DESCRIPTIVE HEADINGS THAT RELATE TO THE FIELDS BEING DISPLAYED.

DOCUMENTATION: USER'S MANUAL, MAINTENANCE MANUAL, DEVELOPMENT SPECIFICATION, TEST PLAN

DEVELOPER: GARY BERGMAN ASSOCIATES, INC.

INFORMATION SOURCE: AIAA SURVEY OF SOFT DEV TOOLS

ACRONYM: ADS/CERL TITLE: AUTOMATED DOCUMENTATION SYSTEM

CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE

FEATURES: SUBJECT, CODE INPUT, FORTRAN, CDC FORTRAN, USER OUTPUT, USER-ORIENTED TEXT, DOCUMENTATION, TABLES, STATIC ANALYSIS, CROSS REFERENCE, AUDITING

STAGE OF DEVELOPMENT: IMPLEMENTED, DATE (YYYYMMDD): 790200

IMPLEMENTATION LANGUAGE: FORTRAN CDC

TOOL PORTABLE: NO

COMPUTER (OTHER HARDWARE): CDC 6000/7000

TOOL AVAILABLE: YES, PUBLIC DOMAIN: YES

RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): AVAILABLE FROM NTIS

TOOL SUPPORTED: YES, TOOL SUPPORT: CERL

TOOL SUMMARY: THE ADS SYSTEM PROVIDES "PROGRAM REFERENCE MANUAL" TYPE REPORTS RELIING ON COMMENTS IN THE FORTRAN SOURCE CODE. IT IS INTENDED TO BE INCLUDED FROM THE BEGINNING OF DEVELOPMENT THROUGH IMPLEMENTATION OF THE SYSTEM. THE ADS DOCUMENTATION AND RESULTANT REPORTS PROVIDE BOTH THE MANAGER AND OTHER PROGRAMMERS ON THE PROJECT WITH INFORMATION ON THE DEVELOPMENT STATUS. IT HAS BEEN USED IN SEVERAL SYSTEM DEVELOPED AT CERL.

DOCUMENTATION: USERS MANUAL (84)

DEVELOPER: CONSTRUCTION ENGINEERING RESEARCH LABORATORY (CERL)

CONTACT: LINDA LAWRIE, DEPT OF ARMY, CERL, PO BOX 4006, CHAMPAIGN, IL 61820, USA, 217-352-6351

INFORMATION SOURCE: COMPLETED SUBMISSION TO NBS

ACRONYM: AFFIRM TITLE: A SPECIFICATION AND VERIFICATION SYSTEM

CLASSIFICATION: REQUIREMENTS/DESIGN SPECIFICATION AND ANALYSIS

FEATURES: SUBJECT, CODE INPUT, VHLL INPUT, ALGEBRAIC SPECIFICATIONS, USER OUTPUT, DIAGNOSTICS, LISTINGS, STATIC ANALYSIS, CONSISTENCY CHECKING, TYPE ANALYSIS, DYNAMIC ANALYSIS, ASSERTION CHECKING, FORMAL VERIFICATION

STAGE OF DEVELOPMENT: IMPLEMENTED

IMPLEMENTATION LANGUAGE: LISP INTERLISP

TOOL PORTABLE: DEC SYSTEM 10/20

OS (OTHER SOFTWARE): INTERLISP

TOOL AVAILABLE: YES, PUBLIC DOMAIN: YES

RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): ACCESS PREFERRED OVER ARPNET

TOOL SUPPORTED: YES, TOOL SUPPORT: INFORMATION SCIENCES INSTITUTE


DOCUMENTATION: REFERENCE MANUAL, USERS GUIDE


DEVELOPER: USC INFORMATION SCIENCES INSTITUTE

CONTACT: R. W. ERICKSON, INFORMATION SCIENCES INSTITUTE, 467 ADMIRALITY WAY, MARINA DEL REY, CA, 90291, USA, 213-622-1511

S. L. GERHART, INFORMATION SCIENCES INSTITUTE, 467 ADMIRALITY WAY, MARINA DEL REY, CA, 90291, USA, 213-622-1511

INFORMATION SOURCE: TOOL FAIR

ACRONYM: AFS TITLE: AUTOMATIC FLOWCHARTING SYSTEM

CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE

FEATURES: SUBJECT, CODE INPUT, USER OUTPUT, GRAPHICS, FLOW CHARTS, STATIC ANALYSIS, SCANNING, STAGE OF DEVELOPMENT: DESIGN

IMPLEMENTATION LANGUAGE: FORTRAN

TOOL SIZE: CORE: 63000

COMPUTER (OTHER HARDWARE): CDC CYBER

OS (OTHER SOFTWARE): NOS
TOOL AVAILABLE: NO, PUBLIC DOMAIN: NO

TOOL SUMMARY: A COMMENT-DRIVEN SYSTEM WHICH ALLOWS THE USER TO DEFINE FLOWCHARTS VIA COMMENTS. THIS ALLOWS THE USE OF EFFECTIVELY FLOWCHART "PSEUDO-CODE", EMBEDDED IN COMM.T AT WHATEVER LEVEL OF ABSTRACTION THE USER DESIRES. CONTAINS DIFFERENT BOX SHAPES FOR DIFFERENT PROGRAM CONTROL FEATURES (E.G., BRANCH, TEST, SUBROUTINE CALL), AND ALLOWS THE USER TO SPECIFY THE FORMAT OF THE GENERATED FLOWCHART. THE USER MAY DEFINE HIS/HER OWN FLOWCHART COMMENT CHARACTERS, AND MAY SELECTIVELY SPECIFY WHICH ROUTINES ARE TO BE FLOWCHARTED BY THE AFS.

DOCUMENTATION: USER'S MANUAL

REFERENCES: [SABE77], SABETKA, D., AND W.R. FRANZ, "AUTOMATIC FLOWCHARTING SYSTEM", GENERAL DYNAMICS POMONA DIVISION TECH. MEMO, 577-0-90A, 770907

DEVELOPER: GENERAL DYNAMICS

CONTACT: RICHARD W. MC HARD, GENERAL DYNAMICS, P.O. BOX 2507, POMONA, CA, 91768, USA, 714-629-5111

INFORMATION SOURCE: AIAA SURVEY OF SOFTWARE TOOLS

ACRONYM: AISM, TITLE: AUTOMATED INTERACTIVE SIMULATION MODELING SYSTEM

CLASSIFICATION: SOFTWARE MODELING AND SIMULATION

FEATURES: SUBJECT, VHDL INPUT, DESIGN SPECIFICATION, TRANSFORMATION, USER OUTPUT, GRAPHICS, TABLES, DYNAMIC ANALYSIS, SIMULATION, COMPUTER SYSTEM SIMULATION

STAGE OF DEVELOPMENT: IMPLEMENTED

IMPLEMENTATION LANGUAGE: FORTRAN, SIMSCRIPT II,5

TOOL PORTABLE: PARTIAL

COMPUTER (OTHER HARDWARE): AMDahl 470 (HP2647/48 GRAPHICS TERMINAL)

OS (OTHER SOFTWARE): OS/MVS (PLOT 10, SIMSCRIPT II,5, ABMS (U. OF MICH))

PUBLIC DOMAIN: YES

RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): AVAILABLE JAN 1982 THROUGH USAF/FEED/TOIT, HANSCOM AFB, MASS.

TOOL SUPPORT: NO

TOOL SUMMARY: THE AUTOMATED INTERACTIVE SIMULATION MODELLING SYSTEM (AISM) IS AN OUTGROWTH OF TWO PREVIOUS MODELLING SYSTEMS: THE DESIGN ANALYSIS SYSTEM (DAS), WHICH PROVIDED INTERACTIVE GRAPHICS SUPPORT FOR MODELLING GENERAL FUNCTIONS IN A PROCEDURE-ORIENTED SYSTEM, AND THE DISTRIBUTED DATA PROCESSING MODEL (DDPM), WHICH CONSISTED OF A LIBRARY OF STANDARD, TABLE-DRIVEN COMPUTER SYSTEM MODELS. AISM COMBINES THE FEATURES OF BOTH TO PROVIDE AN INTERACTIVE, GRAPHICS SIMULATION SYSTEM FOR DISTRIBUTED SYSTEM ANALYSIS. AISM PROVIDES THREE TYPES OF INTERFACES FOR MODELLING COMPUTER SYSTEMS. FLOWCHART-ORIENTED PROCESSES REPRESENTING SOFTWARE, INFORMATION FLOW, OR MANY/MACHINE PROCEDURES ARE ENTERED VIA AN INTERACTIVE GRAPHICS LANGUAGE CONSISTING OF EXECUTION CONTROL, RESOURCE ALLOCATION, AND TIMING PRIMITIVES. ARCHITECTURES OF INTERCONNECTED PROCESSORS, CHANNELS, DISKS, TAPES, AND OTHER DEVICES ARE GRAPHICALLY ENTERED VIA AN INTERACTIVE

ARCHITECTURE DESIGNER,

DOCUMENTATION: SYSTEM DESCRIPTION, USERS MANUAL

REFERENCES: [WILL76], R. WILLIS, "DAS - AN AUTOMATED SYSTEM TO SUPPORT DESIGN ANALYSIS", PROC. 12TH ASILOMAR CONFERENCE, 781-796

DEVELOPER: HUGHES AIRCRAFT COMPANY

CONTACT: BILL AUSTELL, HUGHES AIRCRAFT COMPANY, PO BOX 3310, FULLERTON, CA, 92234, USA, 714-732-3232

INFORMATION SOURCE: TOOL FAIR

ACRONYM: AISM, TITLE: AUTOMATED SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE

FEATURES: SUBJECT, CODE INPUT, FORTRAN, USER OUTPUT, LISTINGS, STATIC ANALYSIS, MANAGEMENT, GLOBAL VARIABLE MANAGEMENT

STAGE OF DEVELOPMENT: IMPLEMENTED

IMPLEMENTATION LANGUAGE: FORTRAN

COMPUTER (OTHER HARDWARE): IBM 360/370

TOOL SUMMARY: THIS PROGRAM RENAMES VARIABLES (ACCORDING TO INPUT REQUESTS) IN FORTRAN SOURCE CODE.

DOCUMENTATION: PROGRAM DESCRIPTION

REFERENCES: [AIDS80], APPLIED SYSTEMS DESIGN SECTION, TRW DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS: CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE TOOLS SERIES, 790100

DEVELOPER: TRW

INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG

ACRONYM: AMPIC, TITLE: AMPIC

CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING

FEATURES: SUBJECT, CODE INPUT, FORTRAN, ASSEMBLY LANGUAGE, USER OUTPUT, GRAPHICS, FLOW CHARTS, LISTINGS, DYNAMIC ANALYSIS, SYMBOLIC EXECUTION

STAGE OF DEVELOPMENT: IMPLEMENTED

IMPLEMENTATION LANGUAGE: SNOBAL

COMPUTER (OTHER HARDWARE): IBM 360/370


DOCUMENTATION: USER'S GUIDE

REFERENCES: [AIDS80], APPLIED SYSTEMS DESIGN SECTION, TRW DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS: CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE TOOLS SERIES, 790100
ACRONYM: ARGUS/MICRO
TITLE: SOFTWARE ENGINEERING WORKSTATION
CLASSIFICATION: SOFTWARE SUPPORT SYSTEM/PROGRAMMING ENVIRONMENT
FEATURES: SUBJECT, TEXT INPUT, CODE INPUT, FORTRAN, PASCAL, FORTRAN 77, TRANSLATION, DIAGNOSTICS, USER-ORIENTED TEXT, GRAPHICS, TABLES, LISTINGS, STATIC ANALYSIS, MANAGEMENT, FILES MANAGEMENT, DYNAMIC ANALYSIS, COVERAGE ANALYSIS, STAGE OF DEVELOPMENT: IMPLEMENTED IMPLEMENTATION LANGUAGE: PASCAL
TOOL PORTABLE: YES
TOOL AVAILABLE: NO, PUBLIC DOMAIN: NO
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): UNDER PROTOTYPE DEVELOPMENT, CURRENTLY FOR INTERNAL BOEING USE
TOOL SUPPORTED: YES, TOOL SUPPORT: BOEING COMPUTER SERVICES COMPANY
TOOL SUMMARY: ARGUS ON A MICROCOMPUTER PROVIDES A NUMBER OF CAPABILITIES SIMILAR TO THOSE AVAILABLE WITH ARGUS ON MAIN-FRAME COMPUTERS, BUT ALSO PROVIDES SOME IMPORTANT NEW CAPABILITIES AFFORDABLE WITH THE DEDICATED PROCESSING POWER AND DISPLAY BANDWIDTH OF A MICROCOMPUTER. THROUGH A MENU-DRIVEN INTERFACE, ARGUS PROVIDES ACCESS TO PASCAL AND FORTRAN 77 COMPILERS, LINK EDITOR, DAPPER -- A DYNAMIC ANALYZER FOR PASCAL, PASCAL SOURCE CROSS REFERENCE GENERATOR, AND VED -- A POWERFUL TEXT AND GRAPHICS EDITOR, WHICH HAS BEEN TAILORED FOR THE CREATION OF "DATA FLOW" DIAGRAMS AND VIEWFOILS, IN ADDITION TO MANY OTHER GRAPHICAL CHARTS FOR ARBITRARY DOCUMENTATION PURPOSES. COMPLEMENTING THESE CAPABILITIES IS A SET OF COMPATIBLE PRINT, PLOT AND FILE MANIPULATION UTILITIES.
DOCUMENTATION: REFERENCE MANUAL
REFERENCES: (STUCKI), LEON G., STUCKI AND HARRY D. WALKER, "CONCEPTS AND PROTOTYPES OF ARGUS -- A PROGRESS REPORT", SOFT ENG ENV, ED. HORST HUENKE, NORTH-HOLLAND PUB CO, 810000
DEVELOPER: BOEING COMPUTER SERVICES
CONTACT: LEON G. STUCKI, BOEING COMPUTER SERVICES COMPANY, P.O. BOX 24346 M/S 96-03, SEATTLE, WA, 98124, USA, 206-675-1118
WILLIAM C. KING, BOEING COMPUTER SERVICES, PO BOX 24346, SEATTLE, WA, 98124
INFORMATION SOURCE: TOOL FAIR
ACRONYM: ARTS, TITLE: AUTOMATED REQUIREMENTS TRACEABILITY SYSTEM
CLASSIFICATION: REQUIREMENTS/DESIGN SPECIFICATION AND ANALYSIS
FEATURES: SUBJECT, VHLL INPUT, REQUIREMENTS SPECIFICATION, USER OUTPUT, TABLES, STATIC ANALYSIS, CONSISTENCY CHECKING, MANAGEMENT, DATA BASE MANAGEMENT, TRACKING, STAGE OF DEVELOPMENT: IMPLEMENTED, DATE (YYMMD): 861001 IMPLEMENTATION LANGUAGE: FORTRAN IV
TOOL PORTABLE: PARTIAL
COMPUTER (OTHER HARDWARE): IBM 360/370, CDC 6600/7600, UNIVAC 11X, DEC VAX-11
OS (OTHER SOFTWARE, EXEC 8, VMS
TOOL PORTABLE: PARTIAL
TOOL SUPPORT: LOCKHEED MISSILES SPACE COMPANY INC.
TOOL SUMMARY: ARTS IS A BOOKKEEPING PROGRAM WHICH OPERATES ON A DATA BASE CONSISTING OF SYSTEM REQUIREMENTS AND REQUIREMENT-RELATED ATTRIBUTES. THE MAJOR FUNCTION OF ARTS IS TO PROVIDE RAPID AND ACCURATE TRACEABILITY, UPWARD AND DOWNWARD, IN A REQUIREMENTS STRUCTURE (TREE). TRACEABILITY ALLOWS ASSESSMENT OF THE IMPACT OF CHANGES, ASSURES THAT TOP-LEVEL REQUIREMENTS ARE SATISFIED BY THE LOWER-LEVEL STRUCTURE, FACILITATES GENERATION OF TEST PLANS AND TESTING AGAINST REQUIREMENTS, AND IS ESSENTIAL FOR STRUCTURED DESIGN AND DEVELOPMENT. BY INCLUDING REQUIREMENT-RELATED ATTRIBUTES IN THE DATA BASE, AUTOMATION CAN BE EXTENDED BEYOND TRACEABILITY, FOR EXAMPLE, SCHEDULE DATES FOR VARIOUS PROJECT EVENTS CAN BE INCLUDED, AND EVENTS SCHEDULED TO OCCUR DURING A SPECIFIED INTERVAL CAN BE ASSOCIATED WITH REQUIREMENTS. COMPLETE FLEXIBILITY IS PROVIDED TO THE USER IN DETERMINING THE ATTRIBUTES TO BE INCLUDED IN THE DATA BASE.
DOCUMENTATION: USER'S MANUAL (80 PAGES)
DEVELOPER: LOCKHEED MISSILES AND SPACE CO., INC., SPACE SYSTEMS DIVISION
CONTACT: M. DORFMAN, LOCKHEED MISSILES AND SPACE CO., INC., 1111 LOCKHEED WAY, SUNNYVALE, CA, 94086, USA, 408-742-6306
INFORMATION SOURCE: COMPLETED SUBMISSION TO NBS
ACRONYM: ASAPMS, TITLE: ASA PROJECT MANAGEMENT SYSTEM
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE
FEATURES: SUBJECT, CODE INPUT, USER OUTPUT, TABLES, LISTINGS, STATIC ANALYSIS, MANAGEMENT, PROJECT MANAGEMENT, STAGE OF DEVELOPMENT: IMPLEMENTED IMPLEMENTATION LANGUAGE: FORTRAN IV
TOOL PORTABLE: YES
SIZE: CORE: 30K=CDC, DEC, 20K=IBM
TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): FOR LEASE, FOR SALE
TOOL SUMMARY: THE ASA PROJECT MANAGEMENT SYSTEM IS A SET OF COMPUTER PROGRAMS DEVELOPED TO AID PROJECT MANAGEMENT IN PLANNING AND CONTROLLING PROJECT SCHEDULES. IT IS UTILIZED IN CRITICAL PATH METHOD AND PRECEDENCE JOBS. IT IS DESIGNED AS A COMPLETE MANAGEMENT INFORMATION SYSTEM THAT INCLUDES PROJECT SCHEDULE COMPUTATION, COST CONTROL,
RESOURCE ALLOCATION, AND DIGITAL GRAPHICS. THE SCHEDULING
PROGRAM IS USED TO MAINTAIN THE PROJECT DATA BASE, TO
PERFORM NETWORK DIAGNOSTICS CHECKS, TO COMPUTE THE PROJECT
SCHEDULE, AND TO PRINT A VARIETY OF REPORTS. FIVE PROGRAMS
ARE INTERFACED WITH THE SCHEDULING PROGRAM: 1) THE COST
PROGRAM IS DESIGNED TO COMPARE THE BUDGET ESTIMATES WITH
THE ACTUAL COSTS AND TO EVALUATE THE COST AND PERFORMANCE
OF THE PROJECT; 2) THE RESOURCE PROGRAM PRINTS RESOURCE
LOADING PROGRAMS AND COMPUTES A FEASIBLE SCHEDULE BASED ON
THE LIMITATIONS OF AVAILABLE RESOURCES; 3) THE
MULTI-PROJECT PROGRAM COMBINES THE FILES OF SEVERAL
PROJECTS AND PRINTS COMPANY WIDE REPORTS.

DOCUMENTATION: USER'S MANUAL.
DEVELOPER: ANDREW SIFRES ASSOCIATES
CONTACT: ANDREW SIFRES ASSOCIATES, 104 EAST 10TH STREET, NEW
YORK, NY, 10016, USA, 212-996-0960
INFORMATION SOURCE: AIAA SURVEY OF SOFT DEV TOOLS

ACRONYM: ASC, TITLE: AUTOMATED SYSTEM CHARTER
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE
FEATURES: SUBJECT, DATA INPUT, USER OUTPUT, USER-ORIENTED TEXT, DOCUMENTATION, GRAPHICS, CHARTS, LISTINGS, STATIC ANALYSIS, SCANNING.

STAGE OF DEVELOPMENT: IMPLEMENTED
TOOL AVAILABLE: YES
TOOL SUPPORTED: YES, TOOL SUPPORT: APPLIED DATA RESEARCH

TOOL SUMMARY: ASC IS THE MOST ADVANCED, DYNAMIC REMEDY TO TODAY'S PRESSING NEED FOR EFFECTIVE SYSTEM COMMUNICATION AND DOCUMENTATION. THE VARIOUS HIGH LEVEL CHARTS AND REPORTS AUTOMATICALLY PRODUCED BY ASC CREATE A RELIABLE AND EXTENSIVE SYSTEM INFORMATION NETWORK. IN EFFECT, ASC AUTOMATES THE BURDEN OF PRODUCING AND MAINTAINING TIMELY SYSTEM-LEVEL DOCUMENTATION, WHILE SIMULTANEOUSLY UPGRADING ITS OVERALL QUALITY AND VALUE. FOR EXISTING SYSTEMS, ASC PERMITS THE USER TO EASILY OBTAIN CRITICALLY NEEDED SYSTEM DOCUMENTATION. SUCH INFORMATION IS ESSENTIAL TO A DISCIPLINED AND PRODUCTIVE APPROACH TO SYSTEM MAINTENANCE AND ENHANCEMENT. ASC IS ALSO A VALUABLE TO FOR NEW OR PROPOSED SYSTEMS, STARTING AT THE ELIEST SPECIFICATION AND DESIGN STAGE, AS A SYSTEM EVOLVES, ASC WILL CONTINUALLY ASSURE THAT ALL DOCUMENTATION KEEPS PACE WITH ONGOING DEVELOPMENT ACTIVITY.

DEVELOPER: APPLIED DATA RESEARCH
CONTACT: APPLIED DATA RESEARCH, ROUTE 206 CENTER, CN=8, PRINCETON, NJ, 08540, USA, 609-924-9100
INFORMATION SOURCE: PRODUCT DESCRIPTION

ACRONYM: ASEQ, TITLE: ASEQ
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE
FEATURES: SUBJECT, CODE INPUT, FORTRAN, TRANSFORMATION, FORMATTING, USER OUTPUT, LISTINGS.

STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN
COMPUTER (OTHER HARDWARE): CDC 6600/7X00

TOOL SUMMARY: ASEQ CAN BE USED DURING THE DEVELOPMENT OF FORTRAN CODE TO PRODUCE STATEMENT LABELS IN ASCENDING ORDER, NEW LABELS BEGIN AT 100 AND ARE INCREMENTED BY 10. LABELS ARE RIGHT-JUSTIFIED IN COLUMNS 2 THROUGH 5. ASEQ WILL PUNCH AND/OR LIST A NEW DECK.

DOCUMENTATION: USER'S MANUAL.

REFERENCE: (AS0579), APPLIED SYSTEMS DESIGN SECTION, TRW DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS: CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE TOOLS SERIES, 790100

DEVELOPER: TRW, SEID SOFTWARE PRODUCT ASSURANCE
CONTACT: FRANK INGRISSIA, TRW, SEID SOFTWARE PRODUCT ASSURANCE, ONE SPACE PARK, REDONDO BEACH, CA, 90278, USA, 213-536-3140

INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG

ACRONYM: ARCP, TITLE: AVIONICS SOFTWARE RELIABILITY PREDICTION
CLASSIFICATION: SOFTWARE MODELING AND SIMULATION
FEATURES: SUBJECT, CODE INPUT, USER OUTPUT, LISTINGS, STATISTICAL ANALYSIS, STATISTICAL ANALYSIS, RELIABILITY ANALYSIS.

STAGE OF DEVELOPMENT: IMPLEMENTED, DATE (YYYYMMDD): 790000
IMPLEMENTATION LANGUAGE: FORTRAN
COMPUTER (OTHER HARDWARE): HONEYWELL 6XXX

TOOL AVAILABLE: YES, PUBLIC DOMAIN: YES
TOOL SUPPORTED: YES, TOOL SUPPORT: RADIC/ISI3


DOCUMENTATION: TECHNICAL REPORT (78)

REFERENCE: (RUSH401), C.K. RUSHFORTH, "SOFTWARE RELIABILITY EQUATION UNDER CONDITIONS OF INCOMPLETE INFORMATION", UNIV. OF UTAH, RADIC FINAL TECHNICAL REPORT TO BE PUBLISHED. 0
IMPLEMENTATION LANGUAGE: ASSEMBLY
COMPUTER (OTHER HARDWARE): XDS SIGMA X

TOOL SUMMARY: ASSIST-I IS A COMPUTER PROGRAM WHICH PERFORMS AUTOMATIC TESTING ON THE CORRECTNESS OF ARITHMETIC COMPUTATIONS, STORAGE REFERENCES, AND EXECUTION SEQUENCING WITHIN EVERY POSSIBLE LOGIC PATH OF THE COMPUTER PROGRAM UNDERGOING TEST. IT IS DESIGNED TO OPERATE ON ASSEMBLY LANGUAGE OBTAINED AS OUTPUT FROM A COMPILER OR CODED DIRECTLY IN ASSEMBLY LANGUAGE. THE BASIC CONCEPT OF ASSIST-I IS THAT EVERY COMPUTER PROGRAM CAN BE SECTIONED INTO LOGICAL TEST UNITS (LTU) WHICH CONSIST OF SEPARATE EXECUTABLE STATEMENTS. ASSIST-I PROCESSES THE SUBJECT PROGRAM IN A SEQUENTIAL MANNER, READING ALL BRANCHES OR POTENTIAL BRANCHES, AT THE END OF EACH LTU, AND COMPARING THE COMPUTED RESULTS WITH THE PREDICTED RESULTS AND OUTPUTS AN ERROR MESSAGE INDICATING DISCREPANCIES, IF ANY, AT THE END OF THE SUBJECT PROGRAM. ASSIST-I LISTS EVERY POSSIBLE LOGIC PATH THROUGH THE PROGRAM BY LTU NAME.

DOCUMENTATION: PROGRAM DESCRIPTION
REFERENCES: [ASS059], APPLIED SYSTEMS DESIGN SECTION, TRW DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE TOOLS SERIES, 790100

DEVELOPER: TRW
INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG

ACRONYM: ATA=FASP, TITLE: AUTOMATED TEST ANALYZER=FASP
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, CMS-2, TRANSFORMATION, INSTRUMENTATION, MACHINE OUTPUT, SOURCE CODE OUTPUT, USER OUTPUT, TABLES, LISTINGS, STATIC ANALYSIS, STATISTICAL ANALYSIS, DYNAMIC ANALYSIS, COVERAGE ANALYSIS, TRACING
STAGE OF DEVELOPMENT: IMPLEMENTED
COMPUTER (OTHER HARDWARE): CDC CYBER, CDC 6000/7000 OS (OTHER SOFTWARE): TECHNOS
TOOL AVAILABLE: YES, PUBLIC DOMAIN: YES
TOOL SUPPORTED: YES, TOOL SUPPORT: NAVAL AIR DEVELOPMENT CENTER

TOOL SUMMARY: THE ATA AUTOMATICALLY SCANS THE SOURCE PROGRAM, DETERMINES THE PATHS BETWEEN DECISION POINTS, AND INSTRUMENTS THE SOURCE CODE WITHOUT ALTERING THE INTENDED COMPUTATIONS. THE INSTRUMENTED PROGRAM IS THEN DYNAMICALLY EXECUTED WITH TEST DATA WHILE SPECIAL RUN-TIME ROUTINES RECORD DATA; THE DYNAMIC EXECUTION MAY BE ON THE SOFTWARE EMULATOR OR THE ACTUAL HARDWARE. THE RECORDED DATA IS POST-PROCESSED TO SHOW WHAT PATHS HAVE BEEN TESTED AS WELL AS THE FREQUENCY OF EXECUTION. THE RESULTANT STATISTICS ARE ACCUMULATED OVER MANY TESTS, FORMING A COMPLETE TEST PICTURE, A PROJECT MANAGER CAN ASSESS THE PROGRESS OF THE TESTING EFFORT AND HAS A QUANTITATIVE INDICATION OF THE RISK ASSOCIATED WITH RELEASING SOFTWARE WHICH IS NOT 100% TESTED. IN THIS CASE, 100% TESTED MEANS ALL STATEMENTS HAVE BEEN EXECUTED AT LEAST ONCE.

THE FREQUENCY OF
**ATA-FASP**

**Execution of certain paths provides information to the software engineer for improving the performance of the software.**

**Documentation:** User's Manual, Maintenance Manual

**References:** (Ref1), D. J. Reifer and H. A. Montgomery, "Seatecs Software Tools Survey", RCI-TR-008, Reifer Consultants, Inc., 810330

**Developer:** Naval Air Development Center

**Contact:** J. Bergey, Naval Air Development Center, Advin, Soft., Tech. Div., Code 503, Warminster, PA, 18974, USA, 215-641-3145

**Information Source:** Seatecs Tools Survey

**Acronym:** ATA-SAT, **Title:** Automatic Testing Analysis Tool

**Classification:** Source Program Analysis and Testing

**Features:** Subject, Code Input, FORTRAN, Transformation, Instrumentation, Machine Output, Source Code Output, FORTRAN, User Output, Tables, Listings, Dynamic Analysis, Coverage Analysis, Assertion Checking, Tracing

**Stage of Development:** Implemented

**Implementation Language:** FORTRAN IV

**Computer (Other Hardware):** DEC VAX-11/75

**Tool Supported:** No

**Tool Summary:** The ATA is an automated tool for the dynamic verification of FORTRAN and assembly language programs. ATA aids the user to gather statistics on the correctness of assertions during execution and the usage of decision to decision paths (DD paths). An assertion is a statement that specifies a condition or relation about certain program variables and is placed in a program in the form of a comment. A DD path is a group of statements that begins with an entry point or decision and includes all statements executed until the next decision or exit point. Output options provided by ATA include reporting true or false assertions, cumulative assertion checking, tracing assertion checking, cumulative DD path usage, tracing DD path usage, module tracing, and selective reporting.

**Documentation:** User's Manual

**References:** (Ref1), D. J. Reifer and H. A. Montgomery, "Seatecs Software Tools Survey", RCI-TR-008, Reifer Consultants, Inc., 810330

**Developer:** Science Application, Inc., Contact: R. Downs, Science Application, Inc., 1257 Tasman Drive, Sunnyvale, CA, 94086, USA, 408-734-4162

**Information Source:** Seatecs Tools Survey

**Acronym:** ATDG, **Title:** Automated Test Data Generator

**Classification:** Source Program Analysis and Testing

**Features:** Subject, Code Input, FORTRAN, User Output, Diagnostics, Tables, Listings, Static Analysis, Data Flow Analysis, Error Checking, Dynamic Analysis, Constraint Evaluation

**Stage of Development:** Implemented

**Implementation Language:** FORTRAN

**Computer (Other Hardware):** UNIVAC 1100

**Tool Summary:** The ATDG is an experimental interactive tool with two different functions: The test data generation (TDG) function provides automated support to program testing at the unit level (i.e., a single subroutine, function, or main program) by identification of effective test case paths and the data constraints which must be satisfied to execute these paths; the static error analysis (SEA) function provides a diagnostic capability to supplement the error detection functions of conventional FORTRAN compilers by identification of path-dependent errors (e.g., uninitialized variables, infinite loops, unreachable code). These two functions are performed by analyzing a logic network of the software element using the principles of directed graph theory and dynamic programming. A network is constructed by defining a software element in terms of segments (logic blocks of FORTRAN statements that can be addressed), and by identifying the transfers and connective properties between these segments.

**Documentation:** User Information Note


**Developer:** TRM

**Tool Available:** Yes, Public Domain: Yes

**Tool Summary:** ATTEST is a symbol enhancement system that does test data generation and error analysis. It is composed of 3 major components. The path selection component monitors the branches and statements that have been executed previously and dynamically selects paths to exercise untested portions of the code. The symbolic execution component creates symbolic representations of the

DOCUMENTATION: TECHNICAL REPORTS

REFERENCES: [DONA60], JOHN D. DONAMOO AND DOROTHY SHEARING, "A REVIEW OF SOFTWARE MAINTENANCE TECHNOLOGY", RADC-TR-80-13, INTERIM REPORT, 800200
[CLA62], L. A. CLARKE, "A SYSTEM TO GENERATE TEST DATA AND SYMBOLICALLY EXECUTE PROGRAMS", IEEE TRANSACTIONS OF SOFTWARE ENGINEERING, SE-2, PP. 215-222, 760900
[CLA68], L. A. CLARKE, "AUTOMATIC TEST DATA SELECTION TECHNIQUES", INFOTECH STATE OF THE ART REPORT, SOFTWARE TESTING, VOL. 2, 780900
[CLA68B], L. A. CLARKE, "TESTING: ACHIEVEMENTS AND FRUSTRATIONS", IEEE SECOND INTER. COMPUTER SOFTWARE AND APPL. CONF., PP. 17100

DEVELOPER: DEPARTMENT OF COMPUTER AND INFORN, SCIENCE
CONTACT: L. A. CLARKE, DEPARTMENT OF COMPUTER AND INFORN, SCIENCE, UNIVERSITY OF MASSACHUSETTS, AMHERST, MA, 01003, USA, 413-545-1328

INFORMATION SOURCE: COMPLETED SUBMISSION TO NBS,
RADC-TR-80-13, INTERIM REPORT

ACRONYM: AUDIT, TITLE: AUDIT
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, FORTRAN, USER OUTPUT, DIAGNOSTICS, LISTINGS, STATIC ANALYSIS, DATA FLOW ANALYSIS, AUDITING
STAGE OF DEVELOPMENT: IMPLEMENTED, DATE (YYMMD): 740000
IMPLEMENTATION LANGUAGE: FORTRAN CDC
COMPUTER (OTHER HARDWARE): CDC 6400/7400

PUBLIC DOMAIN: YES

TOOL SUMMARY: AUDIT IS PART OF A PROCUREMENT SYSTEM DEVELOPED BY THE NAVAL SHIP RESEARCH AND DEVELOPMENT CENTER (NSRD) TO EXAMINE CONTRACTOR PRODUCED COMPUTER PROGRAMS. AUDIT ANALYZES A FORTRAN PROGRAM TO INSURE THAT IT CONFORMS TO A SET OF PROGRAMMING STANDARDS AND ISSUES A REPORT DETAILING THE DEVIATIONS FROM THESE STANDARDS. THE AUDIT STANDARDS PRIMARILY INSURE PROGRAM UNIVERSEITY AMONG A CLASS OF COMPILERS AND ARE SIMILAR TO THE AMERICAN NATIONAL STANDARD FOR FORTRAN. AUDIT ALSO DETERMINS IF THE INPUT PROGRAM CONTAINS PATHS ALONG WHICH A VARIABLE MAY BE NEEDED BEFORE IT HAS BEEN DEFINED. TO DO THIS, AUDIT CONSTRUCTS A DIRECTED GRAPH OF THE FORTRAN PROGRAM AND TRACES EACH PATH OF THE GRAPH. A TOTAL FOR THE NUMBER OF PATHS THAT ARE TRACED FOR EACH MODULE IS ALSO PROVIDED.

DOCUMENTATION: TECHNICAL DESCRIPTION

REFERENCES: [CULP63], L. M. CULPEPPER, "A SYSTEM FOR RELIABLE SOFTWARE ENGINEERING", IEEE TRANSACTIONS ON SOFTWARE ENGINEERING, 750600
DEVELOPER: NAVAL SHIP RESEARCH AND DEVELOPMENT CENTER, BETHESDA, MD
CONTACT: L. M. CULPEPPER, NAVAL SHIP RESEARCH AND DEVELOPMENT CTR, BETHESDA, MD, 20034, USA, 301-227-1867
INFORMATION SOURCE: DOD TECHNICAL REPORT

ACRONYM: AUDITOR, TITLE: AUDITOR
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, FORTRAN, USER OUTPUT, DIAGNOSTICS, LISTINGS, STATIC ANALYSIS, AUDITING, ERROR CHECKING
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN

TOOL PORTABLE: YES

TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO

RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): MARKETED COMMERCIALLY

TOOL SUPPORTED: YES, TOOL SUPPORT: SOFTWARE CORPORATION

TOOL SUMMARY: THE AUDITOR AUTOMATICALLY AUDITS FORTRAN PROGRAMS FOR COMPLIANCE WITH USER PROGRAMMING STANDARDS, POOR PROGRAMMING PRACTICES, NONPORTABLE CODE AND DEVIATIONS FROM THE AMERICAN NATIONAL STANDARDS INSTITUTE (ANSI) DEFINITION OF THE FORTRAN LANGUAGE. THIS TOOL ALSO GENERATES A PROGRAM DOCUMENTATION. IN ADDITION, THIS PRODUCT IS A POWERFUL ERROR DETECTOR WHICH TYPICALLY DETECTS MANY ERRORS THAT ESCAPE COMMERCIAL COMPILERS. THIS TOOL REQUIRE NO MODIFICATION OF ANY COMPILER OR APPLICATION PROGRAM, IT SIMPLY ACCEPTS AS INPUT FORTRAN PROGRAMS AND OUTPUTS VARIOUS REPORTS. AN OPTION IS AVAILABLE THAT ALLOWS FORTRAN PROGRAMS FOR 16-BIT WORD MACHINES TO BE CHECKED ON MACHINES WITH 32-BIT WORDS. THIS PRODUCT POSSESES A STRONG MANAGEMENT ORIENTATION AND SERVES AS AN EXCELLENT QUALITY ASSURANCE TOOL SINCE IT PRESENTS SIMPLE SUMMARIES AT THE END OF ITS CLEAR AND DETAILED OUTPUT.

DOCUMENTATION: USERS MANUAL, TECHNICAL REPORT
DEVELOPER: SOFTWARE CORPORATION
CONTACT: CAROL BADDORF, SOFTWARE CORPORATION, 340 SOUTH KELLOGG AVE., GOLETA, CA, 93117, USA, 805-964-0560
INFORMATION SOURCE: COMPLETED SUBMISSION TO NBS

ACRONYM: AUTDOC, TITLE: AUTOMATIC DOCUMENTER
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE
FEATURES: SUBJECT, CODE INPUT, FORTRAN, FORTRAN V, USER OUTPUT, USER-ORIENTED TEXT, DOCUMENTATION, GRAPHICS, FLOW CHARTS,-static ANALYSIS, SCANNING
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN
BEST/1 (TM)

ON QUEUING NETWORK THEORY AND INCLUDES PROPRIETARY EXTENSIONS FOR EFFICIENTLY ANALYZING REALISTIC ASPECTS OF COMPUTER SYSTEMS. FOR EXAMPLE, BEST/1 CAN MODEL PRIORITY SCHEDULING AT THE CPU OR SEPARATED MEMORY QUEUES FOR EACH TIME SHARING AND TRANSACTION PROCESSING WORKLOAD. THE BEST/1 USER IS FULLY INSULATED FROM MATHEMATICAL DETAILS AND INSTEAD FORMULATES MODELS INTERACTIVELY IN TERMS OF COMPUTER SYSTEM CONCEPTS SUCH AS WORKLOAD (UP TO 10), MULTIPROGRAMMING LEVELS, I/O CHANNELS, AND TRANSACTION VOLUMES. FIFTEEN SEPARATE OUTPUT REPORTS LIST THROUGHPUTS, RESPONSE TIMES, UTILIZATIONS, AND OTHER FACTORS IMPORTANT FOR CAPACITY PLANNING AND SYSTEM TUNING.

DOCUMENTATION: USER'S GUIDE (223 PAGES)

REFERENCES: BGS SYSTEMS, INC., "AN INTEGRATED APPROACH TO CAPACITY PLANNING", PROCEEDINGS OF 8TH EUROPEAN CONF ON COMPUTER SYSTEMS, LONDON, ENGLAND, 601000

DEVELOPER: BGS SYSTEMS, INC.

CONTACT: BGS SYSTEMS, INC., 1 UNIVERSITY OFFICE PARK, WALTHAM, MA, 02254, USA, 617-891-0000

INFORMATION SOURCE: COMPLETE SUBMISSION TO NBS

ACRONYM: BLKGEN

CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE

FEATURES: SUBJECT, CODE INPUT, FORTRAN, USER OUTPUT, LISTINGS, STATIC ANALYSIS, MANAGEMENT, GLOBAL VARIABLE MANAGEMENT

STAGE OF DEVELOPMENT: IMPLEMENTED

IMPLEMENTATION LANGUAGE: FORTRAN

COMPUTER (OTHER HARDWARE): IBM 360/370, CDC 6X00/7X00

TOOL AVAILABLE: YES, PUBLIC DOMAIN: YES

TOOL SUMMARY: THIS PROGRAM READS VARIABLE NAMES AND THEIR DIMENSIONS AND USES THIS INFORMATION TO PUNCH A DIRECTORY FOR THE SDF TYPE OF INPUT.

DOCUMENTATION: USER'S MANUAL

DEVELOPER: BRUNSWICK DEFENSE DIV.

CONTACT: JAMES N. CHURCHYARD, BRUNSWICK DEFENSE DIV., 3333 HARBOR BLVD., COSTA MESA, CA, 92626, USA, 714-564-8850

INFORMATION SOURCE: AIAA SURVEY OF SOFTWARE COURTS

ACRONYM: BLKGEN/INDEX

CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE

FEATURES: SUBJECT, CODE INPUT, FORTRAN, FORTRAN V, MACHINE OUTPUT, SOURCE CODE OUTPUT, STATIC ANALYSIS, MANAGEMENT, CONFIGURATION MANAGEMENT, VERSION CONTROL, DATA BASE MANAGEMENT, GLOBAL VARIABLE MANAGEMENT

STAGE OF DEVELOPMENT: IMPLEMENTED

IMPLEMENTATION LANGUAGE: FORTRAN

COMPUTER (OTHER HARDWARE): UNIVAC 11XX

ACRONYM: BEST/1 (TM)

TITLE: AN INTERACTIVE SYSTEM PERFORMANCE MODELING PACKAGE

CLASSIFICATION: SOFTWARE MODELING AND SIMULATION

FEATURES: SUBJECT, DATA INPUT, USER OUTPUT, DIAGNOSTICS, TABLES, STATIC ANALYSIS, MANAGEMENT, PERFORMANCE MANAGEMENT, CAPACITY PLANNING, DYNAMIC ANALYSIS, SIMULATION, RESOURCE UTILIZATION

STAGE OF DEVELOPMENT: IMPLEMENTED, DATE (YYMMD): 760300

TOOL PORTABLE: YES

TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO

RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): MARKETED PRODUCT, TOOL SUPPORTED: YES, TOOL SUPPORT: BGS SYSTEMS, INC.

TOOL SUMMARY: BEST/1 (TM) IS USED TO PREDICT THE RESPONSE TIME AND THROUGHPUT IMPACT OF EACH WORKLOAD DUCE TO CHANGES IN WORKLOAD VOLUME, HARDWARE CONFIGURATION UPDATES, AND ADJUSTMENT OF OPERATING SYSTEM PARAMETERS. BEST/1 IS BASED

REFERENCES: (ASDG79), APPLIED SYSTEMS DESIGN SECTION, TRW DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS: CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE TOOLS SERIES, 790100

DEVELOPER: TRW

CONTACT: J. PARNELL, TRW SYS ENG AND ANAL DEP, ONE SPACE PARK, REDONDO BEACH, CA, 90278, USA, 213-536-1116

INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG

ACRONYM: BSC, TITLE: BASIC STATISTICS COLLECTOR

CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING

FEATURES: SUBJECT, CODE INPUT, BASIC, USER OUTPUT, TABLES, LISTINGS, STATIC ANALYSIS, STATISTICAL ANALYSIS, PROFILE GENERATION

STAGE OF DEVELOPMENT: IMPLEMENTED, DATE (YMMDD): 760000

IMPLEMENTATION LANGUAGE: PL/I

TOOL PORTABLE: YES

COMPUTER (OTHER HARDWARE): HONEYWELL 6XXX

OS (OTHER SOFTWARE): MULTICS

TOOL AVAILABLE: NO

RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): AVAILABILITY ACCORDING TO AIR FORCE MANUAL (AFM) 300-4, PARAGRAPH 11-7A.

TOOL SUPPORTED: YES, TOOL SUPPORT: RADC/ISIS

TOOL SUMMARY: THE BASIC STATISTICS COLLECTOR IS A GROUP OF PROGRAMS WRITTEN IN THE PL PROGRAMMING LANGUAGE AND RESIDING ON THE HIS 6160 MULTICS SYSTEM AT RADC. THIS TOOL FOR THE BASIC PROGRAMMING LANGUAGE IS AN INITIAL ATTEMPT TO DEVELOP A METHOD FOR COLLECTING AND ANALYZING DATA ON THE GENERAL USAGE OF THE BASIC LANGUAGE, THE TOOL PROCESSES PROGRAMS WRITTEN IN BASIC AND STORES THE GENERAL LANGUAGE USE DATA, COLLECTED FROM PROGRAMS, IN A CENTRAL DATA FILE WHICH CONTAINS ACCUMULATIVE DATA FROM ALL PROCESSED PROGRAMS, DOCUMENTATION, TECHNICAL PAPER (SP), RADC TR-76-9, NTIS ACCESS, NO. A023-494 (54), RADC TR-76-9, NTIS ACCESS, NO. A023-494 (54)

REFERENCES: (RADC73), RADC, "BASIC STATISTICS COLLECTOR", RADC-TR-76-9, NTIS ACCESS, NO. A023-494, 760000

DEVELOPER: RADC

CONTACT: DOUGLAS A. WHITE, ROME AIR DEVELOPMENT CENTER (ISIS), GRIFFISS AIR FORCE BASE, NY, 13441, USA, 315-330-2748

BSC

INFORMATION SOURCE: COMPLETED SUBMISSION TO NBS

ACRONYM: BUDGET VS ACTUAL, TITLE: BUDGET VS. ACTUALS PLOT PROGRAM

CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE

FEATURES: SUBJECT, DATA INPUT, USER OUTPUT, GRAPHICS, ACTIVITY DIAGRAM, STATIC ANALYSIS, MANAGEMENT, PROJECT MANAGEMENT

STAGE OF DEVELOPMENT: IMPLEMENTED

IMPLEMENTATION LANGUAGE: FORTRAN

COMPUTER (OTHER HARDWARE): CDC 6600/7800

TOOL SUMMARY: THE BUDGET VS ACTUALS PROGRAMS ACTUAL WEEKLY EXPENDITURES VS. BUDGET WITH VARIANCES NOTED. A SPECIAL FEATURE OF THE PROGRAM IS THAT IT IS BUILT ACCORDING TO THE WORK BREAKDOWN STRUCTURE (WBS). OUTPUT IS AT ANY WBS LEVEL DESIRED. THE ORIGINAL INPUT MUST INCLUDE THE CONTRACT START AND END DATE, THE WORK UNIT OR WBS ELEMENT, BUDGET START AND END DATE, AND THE WEEKLY BUDGET AMOUNT. WEEKLY INPUTS OF ACTUALS FROM THIS POINT ON IS ALL THAT IS REQUIRED.

REFERENCES: (ASDG79), APPLIED SYSTEMS DESIGN SECTION, TRW DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS: CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE TOOLS SERIES, 790100

DEVELOPER: TRW

CONTACT: TOM KAMPE, TRW, ONE SPACE PARK, REDONDO BEACH, CA, 90278, USA, 213-536-0580

INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG

ACRONYM: CA, TITLE: FORTRAN CODE AUDITOR

CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING

FEATURES: SUBJECT, CODE INPUT, FORTRAN, PCL, USER OUTPUT, DIAGNOSTICS, LISTINGS, STATIC ANALYSIS, AUDITING

STAGE OF DEVELOPMENT: IMPLEMENTED

IMPLEMENTATION LANGUAGE: FORTRAN

COMPUTER (OTHER HARDWARE): CDC 6600/7800

TOOL SUMMARY: THE FORTRAN CODE AUDITOR PROGRAM EXAMINES FORTRAN SOURCE CODE FOR ADHERENCE TO A PREDEFINED SET OF 34 SOFTWARE STANDARDS. IT WAS DEVELOPED FOR USE ON THE SYSTEMS TECHNOLOGY PROGRAM, BUT MOST STANDARDS CAN BE APPLIED TO ANY FORTRAN PROGRAM. CODE AUDITOR IS HEAVILY USED BY BOTH PROGRAMMERS AND QUALITY ASSURANCE ENGINEERS. IT HAS BEEN A VALUABLE TOOLS BECAUSE IT PERMITS THE ENFORCEMENT OF A COMPREHENSIVE SET OF CODING STANDARDS WHICH REDUCES THE NUMBER OF SOFTWARE ERRORS, IMPROVES EXECUTION EFFICIENCY, AND IMPROVES READABILITY AND MAINTAINABILITY. THE FORTRAN CODE AUDITOR ACCEPTS EITHER FORTRAN OR PCL SOURCE CODE (PCL) CODE. IT CAN HANDLE UP TO 300 ROUTINES PER RUN PROCESSING ONE AT A TIME. THE STANDARD OUTPUT INCLUDES: A LISTING OF ALL CODE AUDIT VIOLATIONS ASSOCIATED WITH ANY STANDARD VIOLATIONS; A SUMMARY PAGE FOR EACH ROUTINE LISTING ANY VIOLATIONS AND IDENTIFYING THEIR ASSOCIATED CARD NUMBERS.
CALLREF

COMPUTER (OTHER HARDWARE): IBM 360/370

DOCUMENTATION: USER'S GUIDE

REFERENCES: [ASD79], APPLIED SYSTEMS DESIGN SECTION, TRW DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE TOOLS SERIES, 790100

DEVELOPER: TRW DEFENSE SYSTEMS SOFTWARE DEPT.
CONTACT: TOM HEIM, TRW DEFENSE SYSTEMS SOFTWARE DEPT., ONE SPACE PARK, REDONDO BEACH, CA, 90278, USA, 213-536-2864

INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG

ACRONYM: CALLREF, TITLE: A PERFORMANCE ANALYSIS AND REPORTING PACKAGE FOR IBM MVS
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE
FEATURES: SUBJECT, DATA INPUT, USER OUTPUT, GRAPHICS, REPORTS, TABLES, STATIC ANALYSIS, MANAGEMENT, PERFORMANCE MANAGEMENT, CAPACITY PLANNING, DYNAMIC ANALYSIS, RESOURCE UTILIZATION

STAGE OF DEVELOPMENT: IMPLEMENTED, DATE (YY/M/DD): 800300

TOOL PORTABLE: YES
COMPUTER (OTHER HARDWARE): IBM 360/370
OS (OTHER SOFTWARE): OS/MVS

TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO

RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): MARKETED PRODUCT
TOOL SUPPORTED: YES, TOOL SUPPORT: TRW SYSTEM DEVELOPMENT DEPARTMENT

TOOL SUMMARY: CALLREF PROCESSES MVS/MPP RECORDS PRODUCED BY OS/MVS TO DEVELOP INFORMATION FOR DETERMINING CAPTURE RATIO IN CPU SIZING STUDIES, FOR IDENTIFYING HIGH-OVERHEAD AREAS FOR TUNING APPLICATIONS, AND FOR DEVELOPING REPRESENTATIVE BASELINES FOR CAPACITY PLANNING STUDIES USING EITHER BENCHMARKS OR MODELS. THE PACKAGE SEPARATES TOTAL PROCESSING ACTIVITY DURING ANY USER-SELECTED INTERVAL INTO DISTINCT WORKLOADS REPRESENTING BATCH, TSO, IMS, CICS, AND OTHER CATEGORIES. REPORTS PRODUCED BY CALLREF/MVS CONTAIN SUCH INFORMATION AS: (1) OVERHEAD LOADS ON CPU'S, I/O DEVICES, AND CHANNELS FOR SUCH SYSTEM FUNCTIONS AS PMING, SPOOLING AND I/O INTERRUPT HANDLING; (2) PER-WORKLOAD BREAKDOWNS OF OVERHEAD TIME FOR BATCH, TSO, AND OTHER WORKLOAD CATEGORIES; (3) CAPTURE RATIOS FOR EACH WORKLOAD WITH OPTIONS FOR INCLUDING TELECOMMUNICATIONS OVERHEAD, SPOOLING, AND THE LIKE; (4) ACTIVITY PROFILES FOR EACH WORKLOAD THAT INDICATE THE TOTAL SERVICE TIME PER TRANSACTION AT EACH DEVICE AND PROCESSOR.

DOCUMENTATION: USER'S GUIDE (130 PAGES)

CALLREF
PATH GENERATOR, PATH CONSTRAINT GENERATOR, TEST DATA GENERATOR. FSGC generates a data base consisting of the program graph, the symbol table and the internal representation of the source code. The path generator, by partitioning the program graph, generates a minimal set of paths to cover all edges. The path constraint generator uses symbolic execution to produce a set of equality and inequality constraints on the input variables. The test data generator provides a set of inputs which satisfy the constraints and can be used to execute the program. The constraints are solved by means of random number generation and systematic trial and error procedures, with values being assigned to program variables until all constraints are satisfied.

REFERENCES: [RAM275], C. V. RAMAMOORTHY, "TECHNIQUES FOR AUTOMATED TEST DATA GENERATION" PROCEEDINGS NINTH ASILOMAR CONF. ON CIRCUITS, SYS. AND COMP., 751-760 [RAM276], C. V. RAMAMOORTHY, "ON THE AUTOMATED GENERATION OF PROGRAM TEST DATA", IEEE TRANSACTIONS ON SOFTWARE ENG., VOL. 22-2, PP 293-300, 761200

ACRONYM: CAVS; TITLE: COBOL AUTOMATED VERIFICATION SYSTEM
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, COBOL, TRANFORMATION, RESTRUCTURING, INSTRUMENTATION, FORMATTING, MACHINE OUTPUT, SOURCE CODE OUTPUT, COBOL, USER OUTPUT, DIAGNOSTICS, USER-ORIENTED TEXT, DOCUMENTATION, TABLES, STATIC ANALYSIS, DATA STRUCTURES, CODE REVIEW, DYNAMIC ANALYSIS, PROBE GENERATION, STRUCTURE CHECKING, DYNAMIC ANALYSIS, COVERAGE ANALYSIS, ASSERTION CHECKING, TUNING
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: COBOL
TOOL PORTABLE: YES
TOOL AVAILABLE: NO

RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): AVAILABILITY ACCORDING TO AIR FORCE MANUAL (AFM) 300-6, PARAGRAPHS 11-7A.
TOOL SUPPORTED: YES, TOOL SUPPORT: RADC/ISIS
TOOL SUMMARY: CAVS is a software tool that is being developed to enable software systems written in COBOL to be structurally analyzed, documented, and comprehensively tested. CAVS will provide: (1) selective generation of automated reports resulting from source code static analysis. (2) the ability to selectively instrument COBOL source code and accumulate coverage and processing time data in the dynamic analysis mode. (3) the capability of introducing user-defined assertions (input/output, units and logical assertions) to verify program consistency/accuracy. (4) post-execution analysis and test case assistance on an individual (as well as cumulative) basis.

DOCUMENTATION: TECHNICAL PAPER, USER MANUAL, MAINTENANCE MANUAL, SPECIFICATION, TEST PLAN PROCEDURE

CASEGEN

CONTACT: LAWRENCE M. LOMBARDO. ROME AIR DEVELOPMENT CENTER/ISIE, GRIFFISS AFB, NY, 13441, USA, 315-330-7834
INFORMATION SOURCE: COMPLETED SUBMISSION TO NASA

ACRONYM: CBL/SHT; TITLE: A COBOL PRECOMPILE
CLASSIFICATION: REQUIREMENTS/DESIGN SPECIFICATION AND ANALYSIS
FEATURES: SUBJECT, CODE INPUT, COBOL, TRANFORMATION, TRANSLATION, MACRO EXPANSION, FORMATTING, MACHINE OUTPUT, SOURCE CODE OUTPUT, COBOL, USER OUTPUT, LISTINGS
STAGE OF DEVELOPMENT: IMPLEMENTED, DATE (YY/MM/DD): 25/0000
IMPLEMENTATION LANGUAGE: COBOL ANS
TOOL PORTABLE: YES
TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO

RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): MARKETED PRODUCT
TOOL SUPPORTED: YES, TOOL SUPPORT: GENERAL ELECTRONICS
TOOL SUMMARY: CBL/SHT is a valuable programming aid because: (1) CBL/SHT sharply reduces the tedious error-prone longhand of COBOL programming by allowing the programmer to use short-hand abbreviations. (2) CBL/SHT provides fully expanded COBOL source programs with source listing re-aligned for maximum readability. (3) CBL/SHT abbreviation can be dynamically altered by the user who wishes to expand the standard set supplied. (4) CBL/SHT makes maintenance of programs easier since unlike many other pre-compilers, you do not have to use abbreviations, existing programs are accepted as input and are reformatted for easier reading. CBL/SHT processes the input source program, replacing all abbreviations where necessary. A IDENTIFICATION and ENVIRONMENT DIVISION usually require few changes, while the DATA and PROCESS DIVISIONS are the major areas of change.

DEVELOPER: GENERAL ELECTRONICS
CONTACT: GENERAL ELECTRONICS, PO BOX 79, LYONS, IL 60534, USA, 312-467-2797
INFORMATION SOURCE: PRODUCT DESCRIPTION

ACRONYM: CCA; TITLE: COMPASS CODE AUDITOR
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, COMPASS, USER OUTPUT, DIAGNOSTICS, LISTINGS, STATIC ANALYSIS, AUDITING
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN
COMPUTER (OTHER HARDWARE): CDC 6600/7600
TOOL SUMMARY: The COMPASS CODE AUDITOR audits CDC COMPASS code for adherence to 12 software standards, the standards refer primarily to preface COMMENTARY BLOCKS, IN-LINE COMMENTS, COLUMNAR RESTRICTIONS, and instruction types. The COMPASS CODE AUDITOR can handle up to 300 routines per run, and any subset of coding standards may be selectively supported at the user's option. The standard output includes a list of all COMPASS code auditor standards, a list of the user's COMPASS code annotated with any standard violations, a summary page for each routine listing any
VIOLATIONS AND IDENTIFYING THEIR ASSOCIATED CARD NUMBERS;
A MANAGEMENT SUMMARY LISTING EACH ROUTINE NAME, THE NUMBER
OF EXECUTABLE STATEMENTS, THE NUMBER OF TOTAL CARD IMAGES,
The NUMBER OF STANDARD VIOLATIONS, AND A PERFORMANCE INDEX.
DOCUMENTATION: USER'S MANUAL, TEST ANALYSIS REPORT,
REQUIREMENTS MANUAL, DESIGN MANUAL,
REFERENCES: [ADS879], APPLIED SYSTEMS DESIGN SECTION, TRW
DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS
CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE
TOOLS SERIES, 790100
DEVELOPER: TRW, SEID SOFTWARE PRODUCT ASSURANCE
CONTACT: FRANK INGRASTA, TRW, SEID SOFTWARE PRODUCT
ASSURANCE, ONE SPACE PARK, REDONDO BEACH, CA, 90276, USA,
213-536-3140
INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG

ACRONYM: CCREF, TITLE: CCREF
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, FORTRAN, FORTRAN V, USER
OUTPUT, TABLES, LISTINGS, STATIC ANALYSIS, CROSS REFERENCE,
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN
COMPUTER (OTHER HARDWARE): UNIVAC 11XX

TOOL SUMMARY: CCREF AIDS IN THE TASK OF VERIFYING THE
ACCURACY OF THE COMMON DATA BASE AS IT EXISTS ON A GIVEN
SOURCE CODE TAPE, SEVERAL DISPLAYS ARE PRODUCED WHICH
ENABLE THE PROGRAMMER TO QUICKLY IDENTIFY INCONSISTENCIES
FOUND IN THE SOURCE CODE AND DESIGN MANUAL OF ALL COMMON VARIABLES
IN THE PROGRAM. CCREF PROCESSES ALL FORTRAN ELEMENTS
CONTAINING COMMON VARIABLES AND OPTIONALLY PRODUCES ANY OF THE
FOLLOWING DISPLAYS: ALPHABETICAL CROSS-REFERENCE OF COMMON BLOCKS VS.
SYMBOLIC ELEMENTS IN WHICH THEY APPEAR, AND VICE VERSA;
CROSS-REFERENCE OF INDIVIDUAL CELLS OF EACH COMMON ARRAY VS.
THE SYMBOLIC ELEMENTS IN WHICH THEY OCCUR; A COMPLETE ALPHABETICAL CROSS-REFERENCE OF ALL
COMMON VARIABLES VS. THE SYMBOLIC ELEMENTS AND COMMON BLOCKS IN WHICH THEY APPEAR. CCREF OUTPUT IS VERY HELPFUL
IN THE MAINTENANCE AND DEVELOPMENT OF LARGE COMPUTER
PROGRAMS WHOSE COMMON STRUCTURE IS FREQUENTLY CHANGED.

DOCUMENTATION: COMGEN USER'S GUIDE, 72-FMT-892, COMGEN
PROGRAMMER'S GUIDE, 72-FMT-902
REFERENCES: [ADS879], APPLIED SYSTEMS DESIGN SECTION, TRW
DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS
CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE
TOOLS SERIES, 790100
DEVELOPER: TRW SYS ENG AND ANAL DEP
CONTACT: J. PARNELL, TRW SYS ENG AND ANAL DEP, ONE SPACE
PARK, REDONDO BEACH, CA, 90276, USA, 213-536-1116
INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG

ACRONYM: CCS, TITLE: CHANGE CONTROL SYSTEM
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, USER OUTPUT, LISTINGS, STATIC
ANALYSIS, COMPARISON, MANAGEMENT, CONFIGURATION MANAGEMENT,
CHANGE CONTROL,
STAGE OF DEVELOPMENT: IMPLEMENTED, DATE (YYMMDD): 770110
COMPUTER (OTHER HARDWARE): NO. 1 ESS
TOOL AVAILABLE: YES, TOOL SUPPORT: BELL LABORATORIES
TOOL SUMMARY: THE MAJOR FUNCTIONS OF CCS ARE: CONTROLS
APPLICATION AND APPROVAL STATUS FOR PROGRAM SOURCE CHANGES,
GENERATES AND MAINTAINS OBJECT CODE CHANGES, PERMITS EARLY
SOURCE CODE CHANGES, DETERMINES CHANGE DEPENDENCIES AND
ENFORCES CONCURRENT PROCESSING, SUPPORTS CONTINUOUS
INTEGRATION OF RELEASES IN SEQUENCE, PERMITS
IDENTIFICATION OF ALTERNATIVE SYSTEM VERSIONS, PROVIDES
LANGUAGE/COMPLIER INDEPENDENT PROCESSES, MAINTAINS
CENTRAL, ADMINISTRATIVE DATABASE, PROVIDES STANDARD
ADMINISTRATIVE SYNTAX, THROUGH THE USE OF CCS PROGRAMMERS
HAVE ACCESS TO A STANDARD SET OF SUPPORT PROGRAMS INCLUDING
COMPILERS, EDITORS, AND LOADERS FOR IMPLEMENTING
MODIFICATIONS AND UPDATES TO NO. 1 ESS. CCS IS A
PROGRAMMING SUPPORT INTERFACE PROVIDING AN INTERFACE BETWEEN
PRODUCTION AND MAINTENANCE PROGRAMMERS AND THE NO. 1
ELECTRONIC SWITCHING SYSTEM (ESS) SOFTWARE. NO. 1 ESS IS AN
EXTREMELY LARGE, COMPLEX SYSTEM OPERATING IN A VERY
DYNAMIC ENVIRONMENT.
REFERENCES: [DONA80], JOHN D., DONAHOO AND DOROTHY
SNEARINGER, "A REVIEW OF SOFTWARE MAINTENANCE TECHNOLOGY",
RAFTR-M-13, TECHNICAL REPORT, 800200
[BAUE78], H. A. BAUER, "MANAGING LARGE-SCALE SOFTWARE
DEVELOPMENT WITH AN AUTO CHANGE CONTROL", PROCEEDINGS
COMPSAC 1978, CHICAGO, IL, PP 13-17., 781100
DEVELOPER: BELL LABORATORIES
INFORMATION SOURCE: RADC-TR-80-13, INTERIM REPORT

ACRONYM: CENSUS, TITLE: CENSUS
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, FORTRAN, USER GENERATION,
STATIC ANALYSIS, STATISTICAL ANALYSIS, PROFILE GENERATION,
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN
TOOL SIZE: CORE: 25K
COMPUTER (OTHER HARDWARE): UNIVAC 11XX
OS (OTHER SOFTWARE): ECL
TOOL AVAILABLE: YES, PUBLIC DOMAIN: YES
TOOL SUMMARY: TO READ FORTRAN SOURCE TEXT AND PRODUCE A
REPORT GIVING THE NUMBER OF OCCURRENCES OF ALL OPERANDS,
OPERATORS, LANGUAGE PRIMITIVES, AND STRUCTURES ENCOUNTERED
IN THE TEXT, THE PURPOSE OF THIS DATA IS ANALYSIS OF
PROGRAMS FROM A SOFTWARE SCIENCE POINT-OF-VIEW TO LEARN
HOW TO MEASURE PROGRAM COMPLEXITY AND LOCATE PROGRAM
INVARIANTS.

DOCUMENTATION: USER'S MANUAL
REFERENCES: [ASD79], APPLIED SYSTEMS DESIGN SECTION, TRW DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS: CATALOG AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE TOOLS SERIES, 790100
DEVELOPER: TRW, DEFENSE SYSTEMS SOFTWARE DEPARTMENT
CONTACT: A. J. DESALVIO, TRW, DEFENSE SYSTEMS SOFTWARE DEPARTMENT, ONE SPACE PARK, REDONDO BEACH, CA, 90278, USA, 213-535-3083
INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG

ACRONYM: CHILL TRANS., TITLE: CHILL TRANSLATOR CLASSIFICATION: PROGRAM CONSTRUCTION AND GENERATION FEATURES: SUBJECT, CODE INPUT, CHILL, TRANSLATION, COMPIlATION, MACHINE OUTPUT, INTERMEDIATE CODE,
STAGE OF DEVELOPMENT: IMPLEMENTED, DATE (YMMDD): 800900 IMPLEMENTATION LANGUAGE: FORTRAN IV
TOOL SIZE: 936 STATEMENTS
TOOL AVAILABLE: YES, PUBLIC DOMAIN: YES
TOOL SUPPORTED: YES, TOOL SUPPORTED: UNIVERSITY OF WAIKATO
TOOL SUMMARY: THE CHILL COMPILER IS AN INCREMENTAL INTERACTIVE CHILL PROGRAM TRANSLATOR, IT READS ONE STATEMENT AT A TIME AND TRANSLATES IT INTO THE UNOPTIMISED OUTPUT CODE, THE COMPILER USES A SYNTAX ANALYSE TO PRODUCE A CONDENSED POLISH REPRESENTATION OF EACH STATEMENT, A PASS THROUGH THE REVERSE POLISH NOTATION TO GENERATE THE TARGET CODE.
DOCUMENTATION: TECHNICAL DESCRIPTION (20)
REFERENCES: [PAY73], A. J. PAYNE, "SAPS A CRITICAL REVIEW", SPERRY UNIVAC LTD, 730000
DEVELOPER: A. J. PAYNE
CONTACT: A. J. PAYNE, UNIVERSITY OF WAIKATO, HAMILTON, NEW ZEALAND,
INFORMATION SOURCE: COMPLETED SUBMISSION TO NBS

ACRONYM: CICS DUMP ANAL., TITLE: CICS DUMP ANALYZER CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING FEATURES: SUBJECT, CODE INPUT, CICS, MEMORY DUMP, USER OUTPUT, DIAGNOSTICS, TABLES, STATIC ANALYSIS, CROSS REFERENCE, SCANNING,
STAGE OF DEVELOPMENT: IMPLEMENTED TOOL SIZE: CORE: 50K
COMPUTER (OTHER HARDWARE): IBM 360/370 OS (OTHER SOFTWARE): OS, OS/MVS
TOOL AVAILABLE: YES
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): FOR SALE
TOOL SUMMARY: THE CICS DUMP ANALYZER WAS DESIGNED TO REDUCE THE TIME SPENT IN GOING THROUGH CICS DUMPS IN ORDER TO LOCATE THE REASONS FOR SYSTEM ABENDS. THE DUMP ANALYZER AUTOMATICALLY PERFORMS THE HEXADECIMAL CALCULATIONS, ADDRESS CHAIN SEARCHES, BIT MASK INTERPRETATIONS, AND OTHER TASKS NORMALLY PERformed BY THE PROGRAMMER. FOLLOWING A CICS ABEND, THE DUMP ANALYZER WILL PERFORM A COMPUTER-SPEED INSPECTION AND ANALYSIS OF THE ENTIRE CICS ENVIRONMENT.
THE RESULTS ARE THEN PRESENTED IN AN EASILY ADDRESS
POINTER; 2) VALIDATE ALL SYSTEM QUEUES AND CHAINS; 3)
REPORT ALL SYSTEM MAINTENANCE STATISTICS; 4) INTERPRET
CONTROL BLOCKS AND REPORT THEM USING FAMILIAR CICS
SYMBOLICS AS WELL AS THEIR ENGLISH DESCRIPTION; 5)
INTERPRET ALL SYSTEM BIT VALUES IN ENGLISH; 6) IDENTIFY
THE CONTROL PROGRAM IN WHICH THE ABEND HAS OCCURRED; 7)
COLLECT AND PRINT TOGETHER ALL CONTROL STRUCTURES RELATED
TO A GIVEN TASK, EVEN THROUGH DATA IS DISPERSED THROUGHOUT
MEMORY; 8) PROVIDE A CICS TRACE TABLE SORTED BY TASK; AND
9) PROVIDE INTERPRETED TRACE TABLE
DOCUMENTATION: USER'S MANUAL, MAINTENANCE MANUAL
DEVELOPER: COMMERCIAL SOFTWARE, INC.
INFORMATION SOURCE: AIAA SURVEY OF SOFTWARE TOOLS

ACRONYM: COBOL OPTIMIZ TITLE: COBOL OPTIMIZATION
INSTRUMENTERS
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, COBOL, TRANFORMATION,
INSTRUMENTATION, MACHINE OUTPUT, SOURCE CODE OUTPUT, COBOL,
USER OUTPUT, TABLES, LISTINGS, DYNAMIC ANALYSIS, TUNING.
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN
TOOL PORTABLE: YES
SOFTWARE AVAILABLE: YES, PUBLIC DOMAIN: NO
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): MARKETED PRODUCT
TOOL SUPPORTED: YES, TOOL SUPPORT: SOFTOOL CORPORATION

TOOL SUMMARY: THESE THREE TOOLS THAT AUTOMATICALLY
GENERATE MODULE AND STATEMENT LEVEL EXECUTION-TIME PROFILES
(I.E., REPORTS) OF PROGRAMS, THE PROFILES QUANTIFY OPTIMIZATION
EFFORTS IN DETAIL, THEY SHOW ABSOLUTE AND
RELATIVE EXECUTION TIMES FOR SUBSYSTEMS, MODULES, AND
STATEMENTS AS WELL AS FREQUENCY COUNTS AND OPTIMIZATION
INDICES. THE FIRST TOOL OPERATES AT THE PROGRAM LEVEL,
THE SECOND TOOL OPERATES AT THE PARAGRAPH LEVEL, AND THE
THIRD TOOL OPERATES AT THE STATEMENT LEVEL. THESE TOOLS
REQUIRE NO MODIFICATION OF ANY COMPILER OR APPLICATION
PROGRAM; THEY SIMPLY ACCEPT AS INPUT SOURCE PROGRAMS AND
TEST DATA, AND OUTPUT CLEAR PROFILES, THESE TOOLS PERMIT
TOP-DOWN OPTIMIZATION IN A NATURAL MANNER, THEY POSSESS A
STRONG MANAGEMENT ORIENTATION AND CAN HAVE MUCH IMPACT IN
PROPERLY FOCUSING OPTIMIZATION EFFORTS, THEY SERVE AS AN
EXCELLENT QUALITY ASSURANCE FACILITY.

DOCUMENTATION: USERS MANUAL
DEVELOPER: SOFTOOL CORPORATION
CONTACT: CAROL BADORF, SOFTWARE CORPORATION, 340 SOUTH
KELLOGG AVE., GOLETA, CA, 93117, USA, 805-964-0560
INFORMATION SOURCE: COMPLETED SUBMISSION TO NASA

ACRONYM: COBOL STRUCT TITLE: COBOL STRUCTURED CODE ANALYZER
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, COBOL, USER OUTPUT, TABLES,
LISTINGS, STATIC ANALYSIS, AUDITING, STRUCTURE CHECKING,

STAGE OF DEVELOPMENT: IMPLEMENTED, DATE (YYMMDD): 12/31/80
IMPLEMENTATION LANGUAGE: COBOL
REPORT SIZE: 3,000 STATEMENTS
COMPUTER (OTHER HARDWARE): UNIVAC 110X
TOOL AVAILABLE: YES
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): AVAILABLE FROM
FEDERAL SOFTWER EXCHANGE, FSXEC-600/0118

TOOL SUMMARY: MONITORING THE USE OF STRUCTURED PROGRAMMING
BY MANUFACTURING COBOL SOURCE CODE IS A TIME-CONSUMING
PROCESS, THE STRUCTURED CODE ANALYZER WILL
AUTOMATICALLY DETECT AND PINPOINT OCCURRENCES OF
UNSTRUCTURED COBOL CODE BY A STATIC ANALYSIS OF THE SOURCE
CODE. THE INPUTS IS A COBOL SOURCE PROGRAM WHICH HAS
COMPILED WITH NO ERRORS, THE OUTPUT IS A REPORT CONTAINING
TWO SECTIONS, THE FIRST IS A LISTING OF THE COBOL SOURCE
CODE WHICH WAS ANALYZED, WITH A COMMENT DENOTTING THE TYPE
OF UNSTRUCTURED CODE PRINTED TO THE RIGHT OF THE
CORRESPONDING SOURCE CODE LINE WHERE THE TYPE WAS
FOUND, THE SECOND SECTION IS A SUMMARY PAGE, GIVING THE TOTAL
NUMBER OF OCCURRANCES OF EACH TYPE OF UNSTRUCTURED CODE
FOUND.

REFERENCES: FSXECBOA), GENERAL SERVICES
ADMINISTRATION/NATIONAL TECHNICAL INFORMATION SERVICE,
"FEDERAL SOFTWARE EXCHANGE CATALOG", GSA/ADS/5-C-BO/3,
PB80-904003, BO0900
CONTACT: GSA FEDERAL SOFTWARE EXCHANGE, 2 SKYLINE PL (11TH FL),
5200 LEESEBRO PK, FALLS CHURCH, VA, 22041
703-756-2610
INFORMATION SOURCE: FEDERAL SOFTWARE EXCHANGE CATALOG
COBOL TESTING

OF TESTING, THEY SERVE AS AN EXCELLENT QUALITY ASSURANCE FACILITY WHICH ALLOW MANAGEMENT TO SET, FACILITATE, AND ENFORCE TESTING STANDARDS. THE TESTING INSTRUMENTERS ARE MEMBERS OF SOFTOOL 80, AN INTEGRATED SET OF TOOLS MARKETED BY THE SOFTOOL CORPORATION.

DOCUMENTATION: TECHNICAL REPORTS, USERS MANUAL
DEVELOPER: SOFTOOL CORPORATION
CONTACT: CAROL BADDORF, SOFTOOL CORPORATION, 340 SOUTH KELLOGG AVE., GOLETA, CA, 93117, USA, 805-964-0560
INFORMATION SOURCE: COMPLETED SUBMISSION TO NBS

ACRONYM: COBOL TRACING, TITLE: COBOL TRACING INSTRUMENTERS CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING FEATURES: SUBJECT, CODE INPUT, COBOL, TRANSFORMATION, CLASSIFICATION, MACHINE OUTPUT, SOURCE CODE OUTPUT, COBOL, USER OUTPUT, TABLES, LISTINGS, DYNAMIC ANALYSIS, TRACING, PATH FLOW TRACING.
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN
TOOL PORTABLE: YES
TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): MARKETED PRODUCT
TOOL SUPPORTED: YES, TOOL SUPPORT: SOFTOOL CORPORATION
TOOL SUMMARY: THESE ARE THREE TOOLS THAT AUTOMATICALLY DOCUMENT THE PATH OF PROGRAM CONTROL FLOW FROM MODULE TO SUBPROGRAM. THE FIRST TOOL OPERATES AT THE PROGRAM LEVEL, THE SECOND TOOL OPERATES AT THE PARAGRAPH LEVEL, AND THE THIRD TOOL OPERATES AT THE STATEMENT LEVEL. THESE PRODUCTS OFFER THE SOFTWARE ENGINEER A FLEXIBLE, CONSISTENT, AND EASY TO USE TRACING FACILITY. THESE TOOLS REQUIRE NO MODIFICATION OF ANY COMPILER OR APPLICATION PROGRAM. THEY SIMPLY ACCEPT AS INPUT SOURCE PROGRAMS AND TEST DATA, AND OUTPUT CLEAR TRACE DOCUMENTATION (I.E., PROFILES) WHICH IS FORMATTED AND IN DENTED TO FACILITATE UNDERSTANDING. THEY PERMIT TOP-DOWN TRACING IN A NATURAL MANNER. THE TRACING INSTRUMENTERS ARE MEMBER OF SOFTOOL 80, AN INTEGRATED SET OF TOOLS MARKETED BY SOFTOOL CORPORATION.

DOCUMENTATION: TECHNICAL REPORTS, USERS MANUALS
DEVELOPER: SOFTOOL CORPORATION
CONTACT: CAROL BADDORF, SOFTOOL CORPORATION, 340 SOUTH KELLOGG AVE., GOLETA, CA, 93117, USA, 805-964-0560
INFORMATION SOURCE: COMPLETED SUBMISSION TO NBS

ACRONYM: COBOL/ADE, TITLE: AUTOMATED DEBUGGING ENVIRONMENT FOR COBOL
CLASSIFICATION: SOFTWARE SUPPORT SYSTEM/PROGRAMMING ENVIRONMENT
FEATURES: RUN TIME ANALYSIS, SUBJECT, CODE INPUT, COBOL, USER OUTPUT, DIAGNOSTICS, DYNAMIC ANALYSIS, TIMING, TRACING.
STAGE OF DEVELOPMENT: IMPLEMENTED

COBOL/ADE

COMPUTER (OTHER HARDWARE): IBM 360/370
OS (OTHER SOFTWARE): OS, OS/VS
TOOL AVAILABLE: YES
TOOL SUPPORTED: YES, TOOL SUPPORT: APPLIED DATA RESEARCH
TOOL SUMMARY: COBOL/ADE IS A TOOL DESIGNED TO HELP APPLICATION PROGRAMMERS IN TESTING AND DEBUGGING COBOL PROGRAMS. COBOL/ADE IS RESPONSIVE TO THE DEBUGGING NEEDS OF COBOL PROGRAMMERS AT EVERY LEVEL. DETAILED PROMPTING AND AUTOMATIC DIAGNOSTIC ACTIVITY ASSURES THAT THE NOVICE PROGRAMMER WILL RECEIVE MEANINGFUL RESULTS FROM EVERY DEBUGGING SESSION. SOPHISTICATED DEBUGGING TOOLS ASSURE THAT THE EXPERIENCED COBOL PROGRAMMER WILL BE ABLE TO DEFINE AND ANALYZE EVEN THE MOST INTRICATE PROBLEM. COBOL/ADE APPROACHES DEBUGGING AS A LOGICAL, PROGRAMMABLE TASK. IN A CONTROLLED ENVIRONMENT, COBOL/ADE PROMPTS THE PROGRAMMER TO IDENTIFY SPECIFIC PROBLEM AREAS IN THE PROGRAM AND TO REQUEST THE TYPE OF ANALYSIS TO BE DONE ON EACH. TYPICAL PROBLEMS, SUCH AS ABENDS CAUSED BY BAD DATA ARE AUTOMATICALLY ANALYZED BY COBOL/ADE. PROGRAMMERS AND ONLINE RESOURCES ARE RELEASED FOR OTHER WORK WHILE COBOL/ADE PERFORMS ITS PROBLEMS DETECTION AND ANALYSIS ACTIVITIES IN BACH.
DEVELOPER: APPLIED DATA RESEARCH
CONTACT: APPLIED DATA RESEARCH, ROUTE 206 CENTER, CN 9, PRINCETON, NJ, 08540, USA, 609-924-9100
INFORMATION SOURCE: PRODUCT DESCRIPTION

ACRONYM: COBOL/CP, TITLE: COBOL/CP FOR COBOL-TO-COBOL CONVERSION
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, COBOL, IBM DOS ANSI COBOL (1968), IBM SYSTEM/3 ANSI COBOL, IBM DOS LEVEL D COBOL, HONEYWELL M-200 LEVEL D COBOL, HONEYWELL M-200 LEVEL H COBOL, RCA SPECTRA COBOL, IBM OS LEVEL E COBOL, IBM OS LEVEL F COBOL, TRANSFORMATION, TRANSLATION, CONVERSION, MACHINE OUTPUT, SOURCE CODE OUTPUT, COBOL, IBM DOS ANSI COBOL (1968), IBM OS ANSI COBOL (1968), IBM OS/VS ANSI COBOL (1974), USER OUTPUT, DIAGNOSTICS.
STAGE OF DEVELOPMENT: IMPLEMENTED
TOOL PORTABLE: YES
TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): COMMERCIAL PRODUCT
TOOL SUPPORTED: YES, TOOL SUPPORT: APPLIED DATA RESEARCH
TOOL SUMMARY: COBOL/CP FEATURES: LINE-BY-LINE SYNTAX TRANSLATION, FLANGING OF AREAS REQUIRING MANUAL INTERVENTION, SUMMARIZATION OF CRITICAL REVIEW REQUIREMENTS, GENERATION OF ADVISORY JCL. CONVERSIONS ARE SUPPORTED TO IBM DOS ANSI COBOL (1968), IBM OSI ANSI COBOL (1968), IBM OS/VS ANSI COBOL (1974). PLUS: CONVERSION OF INDEXED, DIRECT AND RELATIVE FILE ACCESS METHODS TO VSAM. COBOL/CP MINIMIZES COBOL-TO-COBOL CONVERSION COSTS, ENHANCES UPGRADE TO THE NEW OPERATING ENVIRONMENT, AND CAN BE ADAPTED TO HANDLE UNIQUE CONVERSION REQUIREMENTS.
ACRONYM: COBOL/DM
TITLE: COBOL PROGRAM DEVELOPMENT MAINTENANCE
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, COBOL, MACHINE OUTPUT, DATA
OUTPUT, USER OUTPUT, USER-ORIENTED TEXT, DOCUMENTATION,
TABLES, STATIC ANALYSIS, I/O SPECIFICATION ANALYSIS,
DYNAMIC ANALYSIS, TRACING,
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: COBOL
TOOL AVAILABLE: YES
TOOL SUPPORTED: YES, TOOL SUPPORT: APPLIED DATA RESEARCH
TOOL SUMMARY: PROGRAMMING AIDS: HIGH-LEVEL FUNCTIONAL VERBS
- 1) DATA MANIPULATION 2) FILE MANAGING 3) TABLE MANAGING
4) REPORT WRITING, 5) DOCUMENTATION: COBOL SHORT FORMS=
RESERVED WORDS, PHRASES AND CLAUSES; DATA NAME PREFIXER,
READABLE ALTERNATIVE TO DATANAME QUALIFICATION, TEST DATA
GENERATOR, USES COBOL FILE AND DATA DESCRIPTIONS -
GENERATES TEST DATA IN PARALLEL WITH PROGRAM DEVELOPMENT;
COBOL/DM ALLOWS AUTOMATION OVER DATA GENERATION; 1) FIELDS
GENERATED AS CONSTANTS, COMPUTER, RANDOM, PRINTABLE, 2) VOLUME
OF TEST DATA UNDER PROGRAM CONTROL; REGISTRATION OF
TEST DATA FOLLOWING MAINTENANCE - DATA GENERATION PARAMETERS
REMAIN IN THE PROGRAM AS COMMENTARY, RUN-TIME DEBUGGING
AID, ABNORMAL TERMINATION ANALYSIS AND REPORTING -
MULIPLE ABNORMALS CAN BE TRAPPED, ANALYZED, LOCATED AND
REPORTED DURING A SINGLE TEST; PROGRAM ACTIVITY DISPLAY
- INPUT, INTERMEDIATE RESULTS AND OUTPUT CONTENTS DISPLAYED
IN ORDER OF TEST EXECUTION;
DEVELOPER: APPLIED DATA RESEARCH, PRINCETON, NJ
CONTACT: APPLIED DATA RESEARCH, ROUTE 206 CENTER, CN-8,
PRINCETON, NJ, 08540, USA, 609-924-9100
INFORMATION SOURCE: PRODUCT DESCRIPTION

ACRONYM: COBOL/DM
TITLE: COBOL QUALITY ASSURANCE,
DOCUMENTATION MAINTENANCE
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, COBOL, TRANSFORMATION,
EDITING, FORMATTING, USER OUTPUT, DIAGNOSTICS, LISTINGS,
STATIC ANALYSIS, AUDITING,
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: COBOL
TOOL AVAILABLE: YES
TOOL SUPPORTED: YES, TOOL SUPPORT: APPLIED DATA RESEARCH
TOOL SUMMARY: COBOL/DM'S PROGRAM STANDARDIZING PROCEEDURES
INCLUDE: 1) AUTOMATIC CONVERSION OF ALTER STATEMENTS TO
MAINTAINABLE COBOL SOURCE CODE 2) CONSISTENT GENERATION OF
COBOL SOURCE FORMAT FOR IMPROVED READABILITY AND
MAINTENANCE, 3) TEXT EDITING CAPABILITIES FOR REVISING
OBSOLETE DATA-NAME AND PROCEDURE-NAME TO MORE MEANINGFUL
TERMS, 4) PREFIXING PROCEDURE-NAME WITH SEQUENCE NUMBERS
FOR EASY REFERENCE, 5) AUTOMATIC "LAUNDERING" OF DATA
DESCRIPTIONS FOR CONSISTENT PHRASE AND CLAUSE CONTENT AND
ORDER.
DEVELOPER: APPLIED DATA RESEARCH
CONTACT: APPLIED DATA RESEARCH, ROUTE 206 CENTER, CN-8,
PRINCETON, NJ, 08540, USA, 609-924-9100
INFORMATION SOURCE: PRODUCT DESCRIPTION
ACRONYM: COBOL/3PP
TITLE: COBOL STRUCTURED PROGRAMMING PRECOMPILER
DESCRIPTION: PROGRAM CONSTRUCTION AND GENERATION
FEATURES: SUBJECT, CODE INPUT, COBOL, STRUCTURED COBOL, ANSI COBOL (1968), TRANSFORMATION, TRANSLATION, STRUCTURE PREPROCESSING, FORMATTING, MACHINE OUTPUT, SOURCE CODE OUTPUT, COBOL
STAGE OF DEVELOPMENT: IMPLEMENTED, DATE (Y/M/D): 740000
IMPLEMENTATION LANGUAGE: COBOL
COMPUTER (OTHER HARDWARE): IBM 360/370, HONEYWELL 6XX
OS (OTHER SOFTWARE): GCOS
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): AVAILABILITY ACCORDING TO AIR FORCE MANUAL (AFM) 20-30-202, PARAGRAPH 11-7A.
TOOL SUPPORT: YES, TOOL SUPPORT: RAD/C/ISIS
TOOL SUMMARY: THE COBOL STRUCTURED PROGRAMMING PRECOMPILER WAS DEVELOPED TO PROCESS AN AUGMENTED ANSI COBOL V.3.23-1968. THE ADDITIONS TO THIS LANGUAGE ARE IN THE FORM OF STRUCTURING VERBS WHICH PERMIT THE PROGRAMMER TO WRITE THE BASIC CONTROL LOGIC FIGURES REQUIRED TO IMPLEMENT STRUCTURED PROGRAMMING FORMS. THE PRECOMPILER IS WRITTEN IN ANS COBOL SUCH THAT CONVERSION TO OTHER COMPUTER SYSTEMS CAN BE ACCOMPLISHED WITH A MINIMUM OF EFFORT.
DOCUMENTATION: USER MANUAL
DEVELOPER: INTERNATIONAL BUSINESS MACHINES CORP.
CONTACT: F. S. JAMISON, RAD/C/ISIS, GRIFFIS AFB, NY, 13441, USA, 315-330-7854
INFORMATION SOURCE: COMPLETED SUBMISSION TO NBS

ACRONYM: COGENT
TITLE: COGENT II COMPILER GENERATOR
CLASSIFICATION: PROGRAM CONSTRUCTION AND GENERATION
FEATURES: SUBJECT, CODE INPUT, VHML INPUT, TRANSFORMATION, SYNTHESIS, MACHINE OUTPUT, SOURCE CODE OUTPUT, DATA OUTPUT, USER OUTPUT, DIAGNOSTICS, LISTINGS
STAGE OF DEVELOPMENT: IMPLEMENTED
TOOL AVAILABLE: YES
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): FOR SALE
TOOL SUMMARY: COGENT IS A PROGRAMMING SYSTEM THAT ACCEPTS A METALANGUAGE DESCRIPTION OF A COMPUTER LANGUAGE, INCLUDING BOTH SYNTAX AND SEMANTICS, AND AUTOMATICALLY PRODUCES A COMPILER FOR THAT LANGUAGE. COGENT IS DESIGNED FOR THE DEVELOPMENT OF COMPILERS FOR INDUSTRY-SPECIFIC LANGUAGES SUCH AS ALGOL, COBOL AND FORTRAN AS WELL AS SPECIAL-PURPOSE LANGUAGES FOR PROCESS CONTROL, SYSTEM PROGRAMMING, INFORMATION UPDATE AND RETRIEVAL LOGIC TESTING, AND OTHER SPECIAL APPLICATIONS. COGENT IS ALSO USEFUL AS A MEANS OF IMPLEMENTING PREPROCESSORS TO STANDARD LANGUAGE COMPILERS AND DEVELOPING PROGRAM CONVERSION AIDS TO TRANSLATE PROGRAMS FROM ONE OPERATING ENVIRONMENT TO ANOTHER. THE SYSTEM IS IMPLEMENTED WITH A TECHNIQUE THAT IS SGAED TO BE HIGHLY MACHINE-INDEPENDENT. COGENT AND THE COMPILERS THAT IT GENERATES CAN BE INSTALLED ON A VARIETY OF COMPUTING EQUIPMENT. COMPILERS GENERATED CAN GENERATE CODE FOR MACHINES OTHER THAN THE MOST AS A MEANS OF IMPLEMENTING CROSS COMPILERS.

ACRONYM: COMGREN
TITLE: COMMON SPECIFICATIONS STATEMENT GENERATOR
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE
FEATURES: SUBJECT, CODE INPUT, FORTRAN, MACHINE OUTPUT, SOURCE CODE OUTPUT, FORTRAN, USER OUTPUT, TABLES, STATIC ANALYSIS, CROSS REFERENCE, MANAGEMENT, CONFIGURATION MANAGEMENT, GLOBAL VARIABLE MANAGEMENT
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN
COMPUTER (OTHER HARDWARE): IBM 360/370, UNIVAC 11XX
TOOL SUMMARY: COMGREN IS DESIGNED TO PROVIDE THE FORTRAN PROGRAMMER WITH AUTOMATED SOFTWARE FOR THE DEVELOPMENT AND MAINTENANCE OF COMPUTER PROGRAMS. THE PRIME CAPABILITIES OF THE SYSTEM ARE AS FOLLOWS: GENERATION AND AUTOMATED INSERTION OF COMMON SPECIFICATION STATEMENTS INTO FORTRAN PROGRAMS; DETERMINATION AND DISPLAY OF ALL PROGRAM COMMON VARIABLES AND COMMON VARIABLES PER SUBROUTINE; AUTOMATED OPERATIONS FOR CONVERTING EXISTING PROGRAMS OVER TO A COMMON COMPATIBLE DATA BASE FOR FUTURE UPDATES AND MAINTENANCE; OUTPUT OF VARIABLE AND SUBROUTINE CROSS-REFERENCES; DEFINITION AND DISPLAY OF SUBROUTINE INTERFACES; PROVISION OF NUMEROS AUTOMATIC PROGRAMMER AID FUNCTIONS; MINIMUM REQUIREMENTS OF THE CAPABILITIES CONTAINED IN THE PROGRAMS BLKGEN, SPECPN, DPNDCY, FORREF, CCREF, AUTODOC, DEPCHT, DOCGEN, DOCEDT.
COMGEN

DOCUMENTATION: USER'S GUIDE, PROGRAMMER'S GUIDE
REFERENCES: [ADS379], APPLIED SYSTEMS DESIGN SECTION, TRW
DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS:
CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE
TOOLS SERIES, 79-1010
DEVELOPER: TRW, SYS ENG AND ANAL DEP
CONTACT: J. PARNELL, TRW SYS ENG AND ANAL DEP, ONE SPACE
PARK, REDONDO BEACH, CA, 90278, USA, 213-536-1116
INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG

ACRONYM: COMGEN/TRW, TITLE: COMMON BLOCK GENERATION PROGRAM
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND
MAINTENANCE
FEATURES: SUBJECT, CODE INPUT, FORTRAN, MACHINE OUTPUT,
SOURCE CODE OUTPUT, USER OUTPUT, LISTINGS, STATIC ANALYSIS,
CROSS REFERENCE, MANAGEMENT, GLOBAL VARIABLE MANAGEMENT,
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN
COMPUTER (OTHER HARDWARE): CDC 6600/7X00

TOOL SUMMARY: THE PURPOSE OF THE COMGEN PROGRAM IS TO
GENERATE THE COMMON BLOCK CAPABILITY OF THE CDC UPDATE SYSTEM.
WORKING ON FORTRAN CODE USING THE TIMESHARE SYSTEM, IT
SPECIALIZES ON JUST THIS ONE ASPECT OF THE UPDATE SYSTEM.
THIS ALLOWS COMGEN TO PROVIDE A FASTER MORE EFFICIENT TOOL
FOR THE MULTIPLE RUNS DURING PROGRAM DEVELOPMENT. THE
USER'S FILES ARE IN A FORMAT FOR CONVERSION TO UPDATE AFTER
DEVELOPMENT IS COMPLETED. IF THAT IS DESIRED, THE FILE
INPUT TO COMGEN CONSISTS OF TWO RECORDS. A MASTER COMMON
CONTAINING ALL OF THE COMMON BLOCKS, IS FIRST.
THE SECOND RECORD CONTAINS THE USER'S SOURCE ROUTINES.
PROCESSING COMMON BLOCKS IDENTIFIED IN THE COMMON BLOCK
RECORD WHICH ARE REQUESTED BY SOURCE ROUTINES.
INSERTED, CREATING AN OUTPUT FILE. THIS ALLOWS
CHANGES TO COMMON BLOCKS TO BE MADE IN ONLY ONE PLACE
ELIMINATING ERRORS DUE TO CHANGES MADE IN SOME ROUTINES
NOT IN OTHERS. A FILE LISTING IS ALSO GENERATED WHICH
IS DATED, PAGE NUMBERED, AND WHICH BEGINS EACH ROUTINE ON A NEW
PAGE.

DOCUMENTATION: TRW IGC 6413.30-096
REFERENCES: [ADS379], APPLIED SYSTEMS DESIGN SECTION, TRW
DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS:
CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE
TOOLS SERIES, 79-1010
DEVELOPER: TRW, SOFTWARE TECHNOLOGY DEPARTMENT
CONTACT: R. L. MAITENZ, TRW, SOFTWARE TECHNOLOGY
DEPARTMENT, ONE SPACE PARK, REDONDO BEACH, CA, 90278, USA,
213-535-3480
INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG

ACRONYM: COMLIST/TRW, TITLE: COMLIST
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND
MAINTENANCE
FEATURES: SUBJECT, CODE INPUT, FORTRAN, USER OUTPUT, TABLES,
LISTINGS, STATIC ANALYSIS, CROSS REFERENCE, MANAGEMENT,
CHANGE CONTROL, GLOBAL VARIABLE MANAGEMENT,
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN
COMPUTER (OTHER HARDWARE): IBM 360/370

TOOL SUMMARY: COMLIST IS A FORTRAN COMPUTER PROGRAM WHICH
PROVIDES A LIST, BY SUBROUTINE, OF ALL COMMON VARIABLES
USED BY A GIVEN PROGRAM. THE OUTPUT FROM COMLIST CAN BE
PRESENTED TO THE REFER UTILITY PROGRAM TO OBTAIN A
CROSS-REFERENCE LISTING IDENTIFYING ALL SUBROUTINES WHICH
USE A PARTICULAR COMMON VARIABLE. SUCH A CROSS-REFERENCE
LISTING IS VALUABLE IN ANALYZING THE EFFECT OF A PROPOSED
SOFTWARE MODIFICATION AND IN DEBUGGING AN ERRORNEOUS
COMPUTER RUN.

DOCUMENTATION: USER'S GUIDE
REFERENCES: [ADS379], APPLIED SYSTEMS DESIGN SECTION, TRW
DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS:
CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE
TOOLS SERIES, 79-1010
DEVELOPER: TRW, DEFENSE SYSTEMS SOFTWARE DEPARTMENT
CONTACT: A. J. DESALVIO, TRW, DEFENSE SYSTEMS SOFTWARE
DEPARTMENT, ONE SPACE PARK, REDONDO BEACH, CA, 90278, USA,
213-536-3083
INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG

ACRONYM: COMLIST, TITLE: COMLIST
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND
MAINTENANCE
COMLIST/TRW

INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG

ACRONYM: COMMAP, TITLE: COMMON BLOCK ANALYSIS TOOL
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, FORTRAN, USER OUTPUT,
DIAGNOSTICS, TABLES, STATIC ANALYSIS, CROSS REFERENCE,
ERROR CHECKING,
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN
TOOL PORTABLE: YES
TOOL AVAILABLE: NO, PUBLIC DOMAIN: NO
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): FOR INTERNAL
BOEING USE
TOOL SUPPORTED: YES, TOOL SUPPORT: BOEING COMPUTER SERVICES
COMPANY

TOOL SUMMARY: COMMAP IS A STATIC ANALYZER FOR FORTRAN
PROGRAMS. OPERATING ON EXISTING SOURCE CODE, IT PRODUCES
A MATRIX CROSS-REFERENCING VARIABLES IN COMMON BLOCKS VERSUS
THE SUBROUTINES THAT USE THEM. THE MATRIX SPECIFIES
WHETHER A VARIABLE IS REFERENCED OR DEFINED WITHIN A
SUBROUTINE. IT ALSO ANALYZES THE INFORMATION IN THE MATRIX
AND REPORTS ON POTENTIAL ERRORS IN THE USE OF VARIABLES
(FOR EXAMPLE, VARIABLES WHICH ARE REFERENCED, BUT NEVER
DEFINED).

REFERENCES: [STUCKI], LEON G. STUCKI AND HARRY D. WALKER,
"CONCEPTS AND PROTOTYPES OF ANGUS — A PROGRESS REPORT", 
SOFTWARE ENV. ED., HORSF HUENEKE, NORTH-HOLLAND PUB CO,
810000

DEVELOPER: BOEING COMPUTER SERVICES
CONTACT: LEON G. STUCKI, BOEING COMPUTER SERVICES COMPANY,
P.O. BOX 28346, K/2, SEATTLE, WA,
206-575-5116

INFORMATION SOURCE: TOOL FAIR

ACRONYM: COMPARE, TITLE: COMPARE
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND
MAINTENANCE
FEATURES: SUBJECT, DATA INPUT, USER OUTPUT, LISTINGS, STATIC
ANALYSIS, COMPARISON,
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN
COMPUTER (OTHER HARDWARE): UNIVAC 11XX
OS (OTHER SOFTWARE): ECL
TOOL AVAILABLE: YES, PUBLIC DOMAIN: YES

TOOL SUMMARY: ALLOWS COMPARISON OF ANY TWO FILES AGAINST
ONE ANOTHER IN ORDER TO DETECT SLIGHT DIFFERENCES, USED
EXTENSIVELY IN NAVIGATION SOFTWARE CERTIFICATION WHEN
VERSIONS OF ABSOLUTES ARE SUPPOSED TO PROVIDE IDENTICAL
OUTPUT FILES. IT HAS THE CAPABILITY TO RESYNCHRONIZE UPON
USER INPUT (INTERACTIVE) TO CONTINUE AFTER DIFFERENCES
ARE DETECTED.

DOCUMENTATION: USER'S MANUAL

COMPARE

DEVELOPER: JET PROPULSION LABORATORY
CONTACT: SANDY PALACIDOS, JET PROPULSION LABORATORY, 4800 OAK
GROVE DRIVE, PASADENA, CA, 91109, USA, 213-344-4884

INFORMATION SOURCE: AIAA SURVEY OF SOFT DEV TOOLS

ACRONYM: COMSCAN, TITLE: COMMON SCANNER ROUTINE
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, FORTRAN, USER OUTPUT,
DIAGNOSTICS, STATIC ANALYSIS, AUDITING,
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN, COBOL, BAL
COMPUTER (OTHER HARDWARE): IBM 360/370

TOOL SUMMARY: THE COMSCAN PROGRAM ANALYZES THE FORTRAN
COMMON DATA BASE PARAMETERS IN EACH ROUTINE OF A FORTRAN
PROGRAM FOR UNIFORMITY.

DOCUMENTATION: USER'S GUIDE

REFERENCES: [ASDS79], APPLIED SYSTEMS DESIGN SECTION, TRW
DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS
CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE
TOOLS SERIES, 790100

DEVELOPER: TRW DEFENSE SYSTEMS SOFTWARE DEPT.
CONTACT: TOM HEIM, TRW, DEFENSE SYSTEMS SOFTWARE DEPT., ONE
SPACE PARK, REDONDO BEACH, CA, 90278, USA,
213-536-2664

INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG

ACRONYM: CONSOMRT, TITLE: COMMON SORT ROUTINE
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND
MAINTENANCE
FEATURES: SUBJECT, CODE INPUT, FORTRAN, USER OUTPUT,
USER-ORIENTED TEXT, DOCUMENTATION, TABLES, STATIC ANALYSIS,
CROSS REFERENCE, MANAGEMENT, GLOBAL VARIABLE MANAGEMENT,
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN
COMPUTER (OTHER HARDWARE): CDC 6500/7X00

TOOL SUMMARY: CONSOMRT READS A FORTRAN SOURCE PROGRAM AND
PROCESSES ALL COMMON, DIMENSION, TYPE AND EQUIVALENCE
STATEMENTS. IT SORTS AND PRINTS OUT COMMON VARIABLES
ACCORDING TO COMMON BLOCK AND INCREASING LOCATION WITHIN
THE BLOCK, THUS, DIFFERENT NAMES FOR SAME LOCATION WILL
APPEAR TOGETHER, PRINTS DIMENSIONS AND TYPE ALONG WITH
LOCATION AND VARIABLE NAME AND LISTS ALL ROUTINES WHERE
THIS INFORMATION IS FOUND. PRINTS ALPHABETIZED LISTING OF
ALL COMMON VARIABLES. IF THE SAME VARIABLE NAME IS USED IN
MORE THAN ONE BLOCK, OR APPEARS AT DIFFERENT LOCATIONS
WITHIN THE SAME BLOCK, MULTIPLE REFERENCES WILL BE GIVEN.
CONSOMRT WILL REPORT, BY NAME, PARAMETERS WHICH ARE
MULTIPLY DEFINED IN COMMON, DEFINED BOTH IN COMMON AND AS
FORMAL PARAMETERS, USED AS VARIABLE DIMENSIONS, BUT ARE NOT
DEFINED IN COMMON, PARAMETERS, EXPLICITLY TYPED MORE THAN
ONCE, DIMENSIONED MORE THAN ONCE, EQUIVALENTED TO COMMON
MORE THAN ONCE.

REFERENCES: [ASDS79], APPLIED SYSTEMS DESIGN SECTION, TRW
DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS
CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE
COMSORT

TOOLS SERIES, 790100
DEVELOPER: TRW, SIMULATION SOFTWARE DEPT
CONTACT: DAVID RICHMOND, TRW, SIMULATION SOFTWARE DEPT, ONE SPACE PARK, REDONDO BEACH, CA, 90278, USA, 213-536-4190
INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG

ACRONYM: COMSTAR
TITLE: COMPUTATIONAL STORAGE AND RETRIEVAL PROGRAM
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE
FEATURES: SUBJECT, DATA INPUT, MACHINE OUTPUT, DATA OUTPUT, STATIC ANALYSIS, MANAGEMENT, LIBRARY MANAGEMENT, FILES MANAGEMENT
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN, COMPASS
COMPUTER (OTHER HARDWARE): CDC 6600/7X00
TOOL SUMMARY: COMSTAR IS A MAJOR DATA STORAGE AND RETRIEVAL SYSTEM. IT IS RELATED TO OTHER MAJOR SYSTEMS SUCH AS GIM, BUT POSSESS UNE CAPABILITIES WHICH MAKE IT PARTICULARLY WELL SUITED TO HANDLE LARGE VOLUME BATCH MODE DATA STORAGE AND RETRIEVAL PROBLEMS. COMSTAR ALLOWS THE USER TO DEFINE HIS DATA STORAGE FILE STRUCTURE, INPUT COMPUTATIONAL FUNCTIONS TO BE PERFORMED, AND OUTPUT DATA FORMATS VIA DATA CARD SPECIFICATIONS, AS WITH ALL DATA STORAGE AND RETRIEVAL SYSTEMS, COMSTAR'S PRINCIPAL IMPORTANCE LIES IN ITS ABILITY TO STRUCTURE AND STORE LARGE VOLUMES OF DATA AND THEN ON REQUEST SORT OR COMBINE THE RAW DATA, COMPIL OR COMPUTE SUPPORTIVE (OR SUMMARY) INFORMATION, AND PRESENT THE REQUESTED OUTPUT IN THE USER'S PREFERRED FORMAT. COMSTAR PROVIDES THE CAPABILITY TO INPUT COMPUTATIONAL REQUESTS AND OUTPUT REQUESTS USING FORTRAN-LIKE STATEMENTS.
DOCUMENTATION: FUNCTIONAL SPECIFICATION
REFERENCES: [A5D679], APPLIED SYSTEMS DESIGN SECTION, TRW DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE TOOLS SERIES, 790100
DEVELOPER: TRW, DATA MANAGEMENT SYSTEMS DEPARTMENT
CONTACT: MILT HAYASHIDA, TRW, DATA MANAGEMENT SYSTEMS DEPARTMENT, ONE SPACE PARK, REDONDO BEACH, CA, 90278, USA, 213-535-2910
INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG

CONFIG

TOOL SUMMARY: GIVEN A MASTER SOURCE TAPE OF FORTRAN AND ASSEMBLY LANGUAGE SOURCE CARD IMAGES, CONFIG GENERATES A LIST IDENTIFYING EACH ROUTINE BY NAME, MODULE NUMBER, Sequence code, document number, number of cards, and the id of the next routine on tape. Two lists are generated, one identifying the routines in the order they appear on tape (sequence code order) and a second in alphabetical order by routine name.
REFERENCES: [A5D679], APPLIED SYSTEMS DESIGN SECTION, TRW DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE TOOLS SERIES, 790100
DEVELOPER: TRW, DEFENSE SYSTEMS SOFTWARE DEPARTMENT
CONTACT: A. J. DESALVIO, TRW, DEFENSE SYSTEMS SOFTWARE DEPARTMENT, ONE SPACE PARK, REDONDO BEACH, CA, 90278, USA, 213-536-5083
INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG

ACRONYM: CONFIGURATOR
TITLE: CONFIGURATOR
CLASSIFICATION: REQUIREMENTS/DESIGN SPECIFICATION AND ANALYSIS
FEATURES: SUBJECT, VHLL INPUT, SYSTEM SPECIFICATION, USER OUTPUT, GRAPHICS, STATIC ANALYSIS, COMPLETENESS CHECKING, CONSISTENCY CHECKING, DYNAMIC ANALYSIS, SIMULATION
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN IV
TOOL SIZE: 2000 STATEMENTS
TOOL AVAILABLE: YES, PUBLIC DOMAIN: YES
TOOL SUPPORTED: YES, TOOL SUPPORT: UNIVERSITY OF WAIKATO
DOCUMENTATION: TECHNICAL DOCUMENTATION (10)
REFERENCES: [PAY75], A. J. PAYNE, "DESIGN OF DISTRIBUTED COMPUTERS", 9TH NZ MATHEMATICS CONGRESS, 7S000
DEVELOPER: A. J. PAYNE
CONTACT: A. J. PAYNE, UNIVERSITY OF WAIKATO, HAMILTON, NEW ZEALAND
INFORMATION SOURCE: COMPLETED SUBMISSION TO NBS

ACRONYM: COPE (TM)
TITLE: COBOL PLUS EXTENSIONS
CLASSIFICATION: PROGRAM CONSTRUCTION AND GENERATION
FEATURES: SUBJECT, VHLL INPUT, PROGRAM SPECIFICATION, TRANSFORMATION, SYNTHESIS, MACHINE OUTPUT, SOURCE CODE OUTPUT, COBOL, USER OUTPUT, LISTINGS
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: COBOL
ACRONYM: COTUNE II, TITLE: COTUNE II
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, COBOL, COBOL F.
TRANSFORMATION, INSTRUMENTATION, OPTIMIZATION, USER OUTPUT,
GRAPHICS, HISTOGRAMS, LISTINGS, DYNAMIC ANALYSIS, COVERAGE
ANALYSIS, TIMING,
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: COBOL, ALC
COMPUTER (OTHER HARDWARE): IBM 360/370
TOOL SUMMARY: COTUNE PROVIDES CRITICAL INFORMATION NECESSARY
IN COBOL PROGRAM DEVELOPMENT, OPTIMIZATION, DEBUGGING,
TESTING AND VALIDATION, MAINTENANCE AND RUN DOCUMENTATION
BY MEANS OF AN EXECUTION-TIME DATA GATHERING PROCESS. ITS
OUTPUT IS THE COMPILER SOURCE LISTING WITH ADDITIONAL COUNTS
SHOWING HOW MANY TIMES EACH STATEMENT WAS EXECUTED, A
NORMALIZED HISTOGRAM SHOWING PERCENTAGE OF CPU TIME SPENT
IN EACH SOURCE STATEMENT AND AN INDICATION OF ANY SOURCE
STATEMENT AT WHICH AN ABEND OCCURRED. ADDITIONALLY,
SUMMARY REPORTS SHOW ALL UNEXECUTED PARAGRAPHS AND WHICH
PARAGRAPHS CONSUMED THE MOST CPU TIME.
DOCUMENTATION: USER'S GUIDE
REFERENCES: [ASD379], APPLIED SYSTEMS DESIGN SECTION, TRW
DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS
CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE
ENGINEERING DIVISION, 1979
DEVELOPER: CAPEX CORPORATION
CONTACT: CAPEX CORPORATION, 2613 N. 3RD ST., PHOENIX, AZ,
85004, USA, 602-264-7241
INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG

ACRONYM: CPA, TITLE: COBOL PROGRAM ARCHIVE
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND
MAINTENANCE
FEATURES: SUBJECT, CODE INPUT, COBOL, MACHINE OUTPUT, SOURCE
CODE OUTPUT, COBOL, STATIC ANALYSIS, MANAGEMENT, LIBRARY
MANAGEMENT, 2K
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: COBOL
COMPUTER (OTHER HARDWARE): IBM 360/370
OS (OTHER SOFTWARE): DOS/V/VS
TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO
TOOL SUMMARY: STORES COBOL PROGRAMS ON TAPE AND ALLOWS THE
RETRIEVAL OF ANY PROGRAM BY A GIVEN SERIAL NUMBER OR DATE
IT ENTERED THE HISTORY FILE.
DEVELOPER: USAF/ALC
CONTACT: T. L. EMERSON, USAF/ALC, 8M=ALC/ADCA8, BLDG 2690,
MCLELLAN AFB, CA, 95652, USA, 916-643-3642
INFORMATION SOURCE: AIAA SURVEY OF SOFT DEV TOOLS

ACRONYM: CPA=ADR, TITLE: CROSS=PROGRAM AUDITOR
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTED
IMPLEMENTATION LANGUAGE: COBOL

TOOL AVAILABLE: YES, PUBLIC DOMAIN: YES

RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): AVAILABLE FROM NTIS, PB-232 024/2

TOOL SUMMARY: THE CPG PREPROCESSOR ASSISTS IN COBOL APPLICATION PROGRAMMING. MACRO-LIKE EXPANSIONS PERMIT THE IDENTIFICATION OF DIVISION, CONFIGURATION SECTION, FILE-CONTROL AND FILE DEFINITION AREAS TO BE CODED IN A SINGLE RECORD. ABBREVIATIONS EXIST FOR FREQUENTLY USED COBOL RESERVED WORDS IN THE DATA AND PROCEDURE DIVISIONS. THE PROGRAMMER MAY ALSO USE COBOL RESERVATION FOR DATA-NAMES, OUTPUT CONSISTS OF A LISTING AND CARD IMAGES FOR THE COBOL COMPILER.

DOCUMENTATION: USER'S MANUAL

REFERENCES: NTIS 860, NATIONAL TECHNICAL INFORMATION SERVICE, "A DIRECTORY OF COMPUTER SOFTWARE AND RELATED TECHNICAL REPORTS", PB80-110232, 800000

INFORMATION SOURCE: NTIS

ACRONYM: CRISPFLOW, TITLE: CRISPFLOW

CLASSIFICATION: REQUIREMENTS/DESIGN SPECIFICATION AND ANALYSIS

FEATURES: SUBJECT: VHLL INPUT, PDL, USER OUTPUT, GRAPHICS, FLOW CHARTS, STATIC ANALYSIS, SCANNING

STAGE OF DEVELOPMENT: IMPLEMENTED

IMPLEMENTATION LANGUAGE: BASIC MINIMAL, FORTRAN

TOOL SIZE: CORE: 40K WORDS

COMPUTER HARDWARE: UNIVAC 11XX

OS (OTHER SOFTWARE): ECL

TOOL AVAILABLE: YES, PUBLIC DOMAIN: YES

TOOL SUMMARY: CONVERTS PDL-LIKE TEXTFILE INTO FLOWCHARTS DRAWN ON COM PLOTTER OF UNIVAC 1108.

DOCUMENTATION: USER'S MANUAL, TECHNICAL PAPER, DEVELOPMENT SPECIFICATION

DEVELOPER: JET PROPULSION LABORATORY

CONTACT: SANDY PALACIOS, JET PROPULSION LABORATORY, 4800 OAK GROVE DRIVE, PASADENA, CA, 91109, USA, 213-344-4864

INFORMATION SOURCE: AIAA SURVEY OF SOFT DEV TOOLS

ACRONYM: CROREF, TITLE: CROSS REFERENCE OF PROGRAM VARIABLES

CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE

FEATURES: SUBJECT: CODE INPUT, FORTRAN, FORTRAN V, USER OUTPUT, USER-ORIENTED TEXT, DOCUMENTATION, TABLES, STATIC ANALYSIS, CROSS REFERENCE

STAGE OF DEVELOPMENT: IMPLEMENTED

IMPLEMENTATION LANGUAGE: FORTRAN, SLEUTH

COMPUTER (OTHER HARDWARE): UNIVAC 11XX

TOOL SUMMARY: CROREF IDENTIFIES CROSS-REFERENCES OF PROGRAM VARIABLES (COMMON AND LOCAL) VS. SYMBOLIC ELEMENTS (SUBROUTINES, MAIN PROGRAM, ETC.). THIS OUTPUT IS NOT ONLY USEFUL IN DEVELOPMENT AND MAINTENANCE OF SOFTWARE SYSTEMS, BUT ADDITIONALLY IS REQUIRED BY DOCUMENTATION STANDARDS.
ACRONYM: CRYSTAL (TM), TITLE: A TOOL FOR ESTIMATING PERFORMANCE OF PROPOSED APPLICATIONS
CLASSIFICATION: SOFTWARE MODELING AND SIMULATION
FEATURES: SUBJECT, VHLL INPUT, SYSTEM SPECIFICATION, USER OUTPUT, DIAGNOSTICS, GRAPHICS, TABLES, STATIC ANALYSIS, MANAGEMENT, PERFORMANCE MANAGEMENT, DYNAMIC ANALYSIS, SIMULATION
STAGE OF DEVELOPMENT: IMPLEMENTED, DATE (YY/MM/DD): 810200
TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): MARKETED PRODUCT
TOOL SUPPORTED: YES, TOOL SUPPORT: BGS SYSTEMS
TOOL SUMMARY: CRYSTAL (TM) IS A SOFTWARE DEVELOPMENT TOOL TO HELP PROJECT MANAGERS, SYSTEM DESIGNERS, AND CAPACITY PLANNERS PREDICT SYSTEM PERFORMANCE BEFORE AND DURING THE DEVELOPMENT PROCESS. CRYSTAL TAKES ESTIMATES OF SOFTWARE STRUCTURE AND MODULE RESOURCE USAGE, AND ALLOWS ACCURATELY PREDICTS A PERFORMANCE PACKAGE TO BE RUN BY OUR PERFORMANCE CALCULATION PACKAGE, BESTI (TM). INITIALLY, THESE ESTIMATES ARE OBVIOUSLY "BALLPARK", BUT EVEN THEN YOU CAN SEE WHAT WILL HAPPEN IF SOME ESTIMATES ARE OFF BY 50% OR EVEN 100% AS THE DEVELOPMENT PROCEEDS, AND SOME MODULES ACTUALLY GET CODED, THESE IMPRECISE ESTIMATES ARE REPLACED BY ACTUAL FIGURES, RESULTING IN SUCCESSION OF IMPROVED PERFORMANCE PREDICTION. IF THE PREDICTED PERFORMANCE IS LESS THAN ACCEPTABLE, THEN THE PROJECT MANAGER HAS ADEQUATE TIME TO TAKE CORRECTIVE ACTION BEFORE A CRISIS DEVELOPS.
DOCUMENTATION: USER'S GUIDE (152)
REFERENCES: (BUZEBI, J. P., BUSSEN, ET. AL., "PREDICTING SOFTWARE PERFORMANCE WITH CRYSTAL (TM)", PROG. 3RD INT CONF ON COMPUTER CAPABILITY, 810400
DEVELOPER: BGS SYSTEMS, INC.
CONTACT: BGS SYSTEMS, INC., 1 UNIVERS ITY OFFICE PARK, WALTHAM, MA, 02254, USA, 617-891-0000
INFORMATION SOURCE: PRODUCT ANNOUNCEMENT, TECHNICAL PAPER
ACRONYM: C94, TITLE: DATABASE DESIGN TOOL
CLASSIFICATION: REQUIREMENTS/DESIGN SPECIFICATION AND ANALYSIS
FEATURES: SUBJECT, DATA INPUT, CODE INPUT, CSL, TRANSFORMATION, TRANSLATION, COMPIILE, MACHINE OUTPUT, OBJECT CODE OUTPUT, DATA OUTPUT, USER OUTPUT, USER-ORIENTED TEXT, DOCUMENTATION, LISTINGS, STATIC ANALYSIS, MANAGEMENT
STAGE OF DEVELOPMENT: IMPLEMENTED, DATE (YY/MM/DD): 750000
IMPLEMENTATION LANGUAGE: FORTRAN
TOOL PORTABLE: NO
COMPUTER (OTHER HARDWARE): IBM 360/370, DEC SYSTEM-10/20,
UNIVAC 1100, DEC VAX-11
OS (OTHER SOFTWARE): TSO, EXEC B, TOPS-10/20
TOOL AVAILABLE: YES
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): CONTACT DABAKONSULT DBK AB
TOOL SUPPORTED: YES, TOOL SUPPORT: DABAKONSULT DBK AB
TOOL SUMMARY: C94 IS A DATABASE DEVELOPMENT PACKAGE, DEVELOPED MAINLY AS A TOOL FOR THE HANDLING AND COMPLEXITY DATABASE SYSTEMS AS WELL AS MINOR PRODUCTION SYSTEMS. C94 INCLUDES AN INTERPRETER FOR AN EASILY LEARNED GENERAL PURPOSE PROGRAMMING LANGUAGE, SPECIALLY DESIGNED FOR HANDLING ASSOCIATIVE DATABASES, IN A C94 ASSOCIATIVE DATABASE INFORMATION IS REPRESENTED AS ASSOCIATIONS BETWEEN ENTITIES. THIS SIMPLE BUT POWERFUL REPRESENTATION FACILITATES HANDLING OF DATA OF ARBITRARY COMPLEXITY, SMALL SELF-CONTAINED PROCEDURES ARE BUILT UP AND STORED IN A PROCEDURE LIBRARY. PROCEDURES CAN CALL EACH OTHER ARBITRARILY, ALSO RECURSIVELY.
REFERENCES: (JAN81), JANNING, BERILD, NACHMENS, "INTRODUCTION TO ASOC, DATABASES AND THE C94 SYSTEM", STUDENTLITETERATUR, LUND, SWEDEN, 810000
(BER1781), BERILD, NACHMENS, "C94-A TOOL FOR DATABASE DESIGN BY UNPOLITICAL SIMULATION", VOL 83, 770000
(AN87), BERILD, NACHMENS, "SOME PRACTICAL APPLICATIONS OF C94 - A DBMS FOR ASSOCIATIVE DATABASES", ARCH AND MODELS IN DATABASE MGT SYS, NIJSSSEN, NORTH-HOLLAND, 770000
(NACK80), NACHMENS, "ASSOCIATIVE DATABASES FOR CHANGING INFORMATION REQUIREMENTS", 13TH HAWAIIAN INTERNATIONAL CONFERENCE ON SYSTEMS SCIENCES, 800000
DEVELOPER: UNIVERSITY OF STOCKHOLM, DABAKONSULT DBK AB
CONTACT: STIG BERILD, UNIVERSITY OF STOCKHOLM, DEPT OF INFORMATION PROCESSING, FACK, STOCKHOLM, 10691, SWEDEN, 46-8-1550100
SAM NACHMENS, DABAKONSULT DBK AB, HUVRASTADGANAN 12, SOLNA, S-171 58, SWEDEN, 08-63 07 30
INFORMATION SOURCE: TOOL FAIR
ACRONYM: CSPP, TITLE: COBOL STRUCTURAL PROGRAMMING PRECOMPILIER
CLASSIFICATION: PROGRAM CONSTRUCTION AND GENERATION
FEATURES: SUBJECT, CODE INPUT, COBOL, STRUCTURED COBOL, ANS COBOL, TRANSFORMATION, TRANSLATION, STRUCTURE PREPROCESSING, MACHINE OUTPUT, SOURCE CODE OUTPUT, COBOL
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: COBOL ANS
TOOL PORTABLE: YES
TOOL SUPPORTED: YES, TOOL SUPPORT: RADC

(AFDA77), AIR FORCE DATA AUTOMATION AGENCY, "COMPENDIUM OF ADS PROJECT MANAGEMENT TOOLS AND TECHNIQUES", 770500.
(DR Event Center), Rome Air Development Center, Rome, NY.

INFORMATION SOURCE: RADC-TR-280-13, INTERIM REPORT.

ACRONYM: CTC, TITLE: COBOL TO COBOL TRANSLATOR
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE.

FEATURES:
- SUBJECT: CODE INPUT, COBOL, TRANSLATION, TRANSLATION, CONVERSION, MACHINE OUTPUT, SOURCE CODE OUTPUT, COBOL, USER OUTPUT, LISTINGS.
- STAGE OF DEVELOPMENT: IMPLEMENTED
- IMPLEMENTATION LANGUAGE: COBOL ANSI
- TOOL SIZE: 100 K MEMORY
- COMPUTER (OTHER HARDWARE): IBM 360/370
- OS (OTHER SOFTWARE): DOS

SOFTWARE: MARKETED PRODUCT
TOOL SUPPORTED: YES, TOOL SUPPORTED: DAWARE, INC.

TOOL SUMMARY: DESIGNED TO AID THE TRANSLATION OF PROGRAMS FROM ONE OPERATING SYSTEM TO ANOTHER, DESIGNED TO PROVIDE A HIGH PERCENTAGE OF CONVERSION OF THE ORIGINAL PROGRAM SYNTAX. PACKAGE CAN BE CUSTOMIZED TO SPECIFIC IN-HOUSE STANDARDS, DIFFERENT VENDORS ALREADY ADDRESSED BY THE TRANSLATOR INCLUDE BURROUGHS, COD, DEC, HONEYWELL, IBM, NCR AND UNIVAC. CURRENTLY RUNS ON A 360/370 COMPUTER USING APPROXIMATELY 120,000 POSITIONS OF CORE MEMORY DISK STORAGE (WORK AREAS), PRINTER AND CARD OR TAPE I/O DEVICES, CAN BE INSTALLED ON ANY COMPUTER THAT SUPPORTS ANSI COBOL AND HAS A SIMILAR CONFIGURATION AS OUTLINED ABOVE.


DEVELOPER: DAWARE, INC.
CONTACT: LYNNANG M. PHILLIPS, DAWARE, INC., 2565 ELMWOOD AVE., BUFFALO, NY, 14217, USA, 716-686-6722

INFORMATION SOURCE: NOSE SEATECS SURVEY.

ACRONYM: CUE, TITLE: CONFIGURATION UTILIZATION EVALUATOR
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE.

FEATURES:
- SUBJECT: CODE INPUT, USER OUTPUT, TABLES, DYNAMIC ANALYSIS, TUNING, RESOURCE UTILIZATION.
- STAGE OF DEVELOPMENT: IMPLEMENTED
- IMPLEMENTATION LANGUAGE: COBOL, ASSEMBLY
- COMPUTER (OTHER HARDWARE): IBM 360/370
- OS (OTHER SOFTWARE): OS, OS/VSE, OS/MVS

SOFTWARE: MARKETED PRODUCT
TOOL SUPPORTED: YES

TOOL SUMMARY: CUE (CONFIGURATION UTILIZATION EVALUATOR) MEASURES COMPUTER HARDWARE/SOFTWARE PERFORMANCE TO ALLOW AN INSTALLATION TO GAIN MAXIMUM EFFICIENCY FROM ITS CURRENT SYSTEM. HARDWARE MEASURES REPORTED BY CUE INCLUDE CPU, CHANNEL, AND DEVICE ACTIVITY. CUE ALSO MEASURES SOFTWARE RESOURCES, SVC LOADING, AND LOGICAL CHANNEL USAGE. THESE MEASUREMENTS ALLOW AN INSTALLATION TO PINPOINT CURRENT SYSTEM BOTTLENECKS AND TO DETERMINE POSSIBLE RECONFIGURATION REQUIREMENTS. THE CUE PACKAGE CONSISTS OF TWO PROGRAMS: AN EXTRACTOR AND AN ANALYZER, WHICH ARE RUN AS SEPARATE JOBS OR JOB STEPS, THE EXTRACTOR SAMPLES SYSTEM ACTIVITY OVER A USER-SPECIFIED TIME PERIOD, AND OUTPUTS ITS SAMPLED INFORMATION TO AN EXTRACTOR DATA SET, THE CUE ANALYZER THEN PROCESSES THE DATA IN THE EXTRACTOR DATA SET TO PRODUCE REPORTS DESCRIBING IN DETAIL THE ACTIVITY THAT WAS SAMPLED BY THE EXTRACTOR. THE EXTRACTOR MUST RUN AS THE HIGHEST PRIORITY JOB IN THE SYSTEM.

DOCUMENTATION: USER'S MANUAL, MAINTENANCE MANUAL
DEVELOPER: BOLLE AND BABBAGE
INFORMATION SOURCE: AIAA SURVEY OF SOFTWARE TOOLS.

ACRONYM: DUE, TITLE: DATA MINER
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE.

FEATURES:
- SUBJECT: DATA INPUT, TRANSCENDENT DATA INPUT, USER-ORIENTED TEXT, REPORTS, GRAPHICS, TABLES, LISTINGS, STATIC ANALYSIS, CROSS REFERENCE, SCANNING.
- STAGE OF DEVELOPMENT: IMPLEMENTED
- IMPLEMENTATION LANGUAGE: BAL
- TOOL SIZE: CORE: 82-108K+DOS, 120-140K+OS
- COMPUTER (OTHER HARDWARE): IBM 360/370
- OS (OTHER SOFTWARE): OS, OS/VSE, OS/MVS, DOS

SOFTWARE: MARKETED PRODUCT
TOOL SUPPORTED: YES, PUBLIC DOMAIN

TOOL SUMMARY: THE DATA MINTER IS A MULTI-PURPOSE INFORMATION RETRIEVAL, ANALYSIS, AND PRESENTATION SYSTEM WITH MANY LEVELS OF LANGUAGE, SO THAT APPLICATION-AREA EXECUTIVES AND PROGRAMMERS CAN PRODUCE REPORTS FROM ONE OR MORE FILES OR DATA BASES. THE BASIC LANGUAGE LEVEL IS COMPLETELY FREE-FORM, WITH IT USERS CAN REQUEST RECORD SELECTION, SORTING, UNLIMITED COMPUTATIONS, AND FORMATTED REPORT ALSO HAVE REQUEST BAR GRAPHS, CROSS-TABULATIONS, ADDRESS LABELS, PREPRINTED FORMS, AND COMPLEX STATISTICAL FUNCTIONS. MORE TECHNICAL LEVELS ALLOW PROGRAMMERS DIRECT CONTROL OVER PROCESSING, INCLUDING IF/GO TO LOGIC. A MACRO PROCESSOR IS INCLUDED TO PERMIT USER.
EXPANSION OF THE SYSTEM'S FEATURES, THE SYSTEM'S OPTIONS ALLOW FOR PROCESSING MULTIPLE FILES SIMULTANEOUSLY AND FOR MATCHING STANDARD FILE STRUCTURES WITH DATA BASE FILES, OPTIONAL GRAPHIC CAPABILITIES INCLUDE POINT PLOTS, DEVIATION GRAPHS, MAPS, ORGANIZATION CHARTS, AND DOCUMENTATION: USER'S MANUAL, TECHNICAL PAPER

DEVELOPER: PROGRAM PRODUCT, INC.

INFORMATION SOURCE: AIAA SURVEY OF SOFT DEV TOOLS

ACRONYM: DARTS, TITLE: DESIGN AID FOR REAL-TIME SYSTEMS
CLASSIFICATION: REQUIREMENTS/DESIGN SPECIFICATION AND ANALYSIS
FEATURES: SUBJECT, VHLL INPUT, REQUIREMENTS SPECIFICATION, DESIGN SPECIFICATION, USER OUTPUT, USER-ORIENTED INTERFACE, REPORTS, STATIC ANALYSIS, CONSISTENCY CHECKING, AUDITING, MANAGEMENT, DATA BASE MANAGEMENT
STAGE OF DEVELOPMENT: IMPLEMENTED, DATE (YYMMDD) 780000
IMPLEMENTATION LANGUAGE: FORTRAN, SIMSCRIPT II.5
TOOL SIZE: CORE: 400K
COMPUTER (OTHER HARDWARE): CDC CYBER (DISK: 2MB)
OS (OTHER SOFTWARE): 730, OS/MVS
TOOL AVAILABLE: YES
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): RESTRICTED TO HUGHES SUBCONTRACTORS OR UPON APPROVAL OF ICAM PROJECT

TOOL SUPPORTED: YES, TOOL SUPPORT: HUGHES

TOOL SUMMARY: DARTS IS A GENERAL SYSTEM MODELING TOOL WITH INTERACTIVE GRAPHICS AND AUTOMATIC MODEL GENERATION AS ITS KEY FEATURES. IT USES ADMS, A GENERALIZED DATA BASE MANAGER FROM PSL/PSA, AS ITS MEANS TO MAINTAIN THE MODEL OR REQUIREMENTS SPECIFICATION. IT HAS AUTOMATIC DOCUMENTATION SUPPORT TOOLS. ITS PRIMARY PURPOSE IT TO ESTABLISH THE FEASIBILITY OF REQUIREMENTS AND, IN PARTICULAR, THE DYNAMIC NATURE OF THE SYSTEM UNDER STUDY, IT HAS BEEN USED FOR MANUFACTURING, OPERATIONS ANALYSIS, COMPUTER SYSTEM DESIGN ANALYSIS, AND FOR SOFTWARE DESIGN ANALYSIS. OTHER IMPORTANT FEATURES ARE: EASY TO LEARN AND USE, INTERACTIVE ANALYSIS VIA ON-LINE PLOTTED GRAPHICS THAT REDUCES ANALYSIS TIME BY AS MUCH AS A FACTOR OF 10.

DOCUMENTATION: USERS GUIDE

REFERENCES: [2UL80], P. A. SZULEWKA, ET. AL., "QA GUIDELINES AND QUALITY METRICS FOR EMBEDDED REAL-TIME SOFTWARE DESIGN", REPORT R-1376, THE C. S. DRAPER LAB., INC., 800500
[FURT81], P. C. FURTEK, J. B. DE WOLF, AND P. BUCHAN, "DARTE: A TOOL FOR SPECIFICATION AND SIMULATION OF REAL-TIME SYSTEMS", PROCEEDINGS OF THE AIAA COMPUTER IN AEROSPACE III CONFERENCE, 811000

DEVELOPER: THE CHARLES DRAPER LABORATORY, INC.

CONTACT: J. BARTON DE WOLF, THE CHARLES DRAPER LAB., INC., TECHNOLOGY SQUARE, CAMBRIDGE, MA, 02139, USA, 617-256-1115

INFORMATION SOURCE: COMPLETED SUBMISSION TO NASA

ACRONYM: DAS, TITLE: DESIGN ANALYSIS SYSTEM
CLASSIFICATION: SOFTWARE MODELING AND SIMULATION
FEATURES: SUBJECT, CODE INPUT, VHLL INPUT, REQUIREMENTS SPECIFICATION, DESIGN SPECIFICATION, USER OUTPUT, USER-ORIENTED TEXT, REPORTS, STATIC ANALYSIS, CONSISTENCY CHECKING, AUDITING, MANAGEMENT, DATA BASE MANAGEMENT
STAGE OF DEVELOPMENT: IMPLEMENTED, DATE (YYMMDD) 780800
IMPLEMENTATION LANGUAGE: FORTRAN, SIMSCRIPT II.5
TOOL SIZE: CORE: 400K
COMPUTER (OTHER HARDWARE): CDC CYBER (DISK: 2MB)
OS (OTHER SOFTWARE): 730, OS/MVS
TOOL AVAILABLE: YES
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): NO

ACRONYM: DATA DESIGNER, TITLE: DATA DESIGNER
CLASSIFICATION: REQUIREMENTS/DESIGN SPECIFICATION AND ANALYSIS
FEATURES: SUBJECT, VHLL INPUT, DESIGN SPECIFICATION, USER OUTPUT, GRAPHICS, DESIGN CHARTS, TABLES, STATIC ANALYSIS, MANAGEMENT, DATA BASE MANAGEMENT
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN
COMPUTER (OTHER HARDWARE): IBM 360/370, UNIVAC 11X
TOOL AVAILABLE: YES
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): MARKETED PRODUCT
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE

FEATURES: SUBJECT, CODE INPUT, COBOL, USER OUTPUT, USER-ORIENTED TEXT, DOCUMENTATION, GRAPHICS, FLOW CHARTS, TABLES, FILE LAYOUTS, STATIC ANALYSIS, DATA FLOW ANALYSIS, CROSS REFERENCE,

STAGE OF DEVELOPMENT: IMPLEMENTED

IMPLEMENTATION LANGUAGE: COBOL

COMPUTER (OTHER HARDWARE): IBM 360/370


DOCUMENTATION: USER'S MANUAL

REFERENCES: (ASD179), APPLIED SYSTEMS DESIGN SECTION, TRW DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS: CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE TOOLS SERIES, 790100

DEVELOPER: BOOL AND BAGGAGE

CONTACT: T. FRED NOBLE, CGA SOFTWARE PRODUCTS GROUP, 1370 PICCARD DRIVE, ROCKVILLE, MD, 20850, USA, 301-948-9600

INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG

ACRONYM: DCD, TITLE: DESIGN EXPRESSION AND CONFIGURATION AIDS

CLASSIFICATION: REQUIREMENTS/DESIGN SPECIFICATION AND ANALYSIS

FEATURES: SUBJECT, VHLL INPUT, DESIGN SPECIFICATION, USER OUTPUT, DIAGNOSTICS, USER-ORIENTED TEXT, DOCUMENTATION, STATIC ANALYSIS, SCANNING

STAGE OF DEVELOPMENT: IMPLEMENTED

IMPLEMENTATION LANGUAGE: FORTRAN

COMPUTER (OTHER HARDWARE): IBM 360/370


DOCUMENTATION: USER'S MANUAL, INSTALLATION INSTRUCTIONS

REFERENCES: (ASD179), APPLIED SYSTEMS DESIGN SECTION, TRW DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS: CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE TOOLS SERIES, 790100

DEVELOPER: BC3, INC.

CONTACT: BC3, INC., P.O. BOX 24346, SEATTLE, WA, 98124, USA, 206-773-5950

INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG

ACRONYM: DECA, TITLE: DECKBOY FILE COMPARATOR

CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE

FEATURES: SUBJECT, DATA INPUT, USER OUTPUT, LISTINGS, STATIC ANALYSIS, COMPARISON,
DI-3000

**FEATURES**: SUBJECT, DATA INPUT, TRANSFORMATION, FORMATTING, USER OUTPUT, GRAPHICS,
STAGE OF DEVELOPMENT: IMPLEMENTED, DATE (YMMDD): 800500
IMPLEMENTATION LANGUAGE: FORTRAN 66
TOOL PORTABLE: YES, TOOL SIZE: 35,000 SOURCE STATEMENTS
TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): LICENSE REQUIRED
TOOL SUPPORTED: YES, TOOL SUPPORT: PRECISION VISUALS, INC.
TOOL SUMMARY: DI-3000 IS AN INTEGRATED SYSTEM OF GRAPHICS SOFTWARE TOOLS, DEVELOPED AS A PACKAGE OF 160 USER-CALLABLE SUBROUTINES. DI-3000 IS BOTH DEVICE AND MACHINE INDEPENDENT. IT OFFERS A COMPREHENSIVE SET OF TOOLS FOR DEVELOPING GRAPHICS APPLICATIONS, INCLUDING FULL COLOR, COMPLETE 3D, AREA FILL AND PATTERNING, GRAPHICS ART QUALITY TEXT, SNAPSHOT DEBUGGING AND A PICTURE FOR DIGITIZING, MENU FUNCTIONS AND COMPUTER-AIDED DESIGN. DI-3000 NETWORK ALSO PERMITS APPLICATION PROGRAM INTERACTION WITH SEVERAL GRAPHICS DEVICES CONCURRENTLY, USING INTELLIGENT DEVICE DRIVERS. DI-3000 TAKES FULL ADVANTAGE OF THE HARDWARE FEATURES OF ANY GRAPHICS DEVICE.
DOCUMENTATION: USER'S GUIDE (334 PAGES), IMPLEMENTATION GUIDE (190 PAGES), REFERENCE CARD
DEVELOPER: PRECISION VISUALS, INC.
CONTACT: PRECISION VISUALS, INC., 250 ARAPAHOE AVE., BOULDER, CO, 80302, USA, 303-449-0806
INFORMATION SOURCE: COMPLETED SUBMISSION TO NBS

DI-3000

**ACRONYM**: DI-3000, TITLE: DI-3000 GRAPHICS SOFTWARE SYSTEM
CLASSIFICATION: PROGRAM CONSTRUCTION AND GENERATION

DECKBOY COMPAR

**STAGE OF DEVELOPMENT**: IMPLEMENTED, DATE (YMMDD): 771100
IMPLEMENTATION LANGUAGE: FORTRAN IV
TOOL PORTABLE: YES
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): PROPERTY OF BCS AND MAY NOT BE AVAILABLE FOR USE OUTSIDE BOEING
TOOL SUMMARY: THE DECKBOY COMPARATOR FINDS DIFFERENCES BETWEEN TWO FILES, INCLUDING LINES THAT EXIST IN ONE FILE BUT NOT THE OTHER AND DIFFERENCES WITHIN LINES CONTAINED IN BOTH FILES. EACH FILE INPUT TO THE COMPARATOR MAY BE SPLIT INTO MODULES AND COMPARED ON A MODULE-BY-MODULE BASIS. DIFFERENCES BETWEEN FILES (VERSIONS) ARE LISTED IN THE ORDER OF DISCOVERY AND IDENTIFIED AS DELETED, ADDED, OR CHANGED. THE COMPARATOR ATTEMPTS TO MAINTAIN LINE SYNCHRONIZATION BETWEEN THE FILES BY CONTINUOUSLY ATTEMPTING TO FIND AN IMAGE IN VERSION 2 MATCHING ONE IN VERSION 1.
DOCUMENTATION: TECHNICAL DESCRIPTION, USERS MANUAL
REFERENCES: [BOWIT], BOEING COMPUTER SERVICES COMPANY, ADVANCED TECHNOLOGY AND APPL. DIV., "AUTOMATED SOFTWARE TOOLS CATALOG", BCS 10236, 790600
DEVELOPER: BOEING COMPUTER SERVICES
CONTACT: GARY KAMPEN, BOEING COMPUTER SERVICES COMPANY, P.O. BOX 24346, SEATTLE, WA, 98124, USA, 206-575-5393
INFORMATION SOURCE: BCS TOOLS CATALOG

ACRONYM: DEPTH, TITLE: DEPTH
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE
FEATURES: SUBJECT, CODE INPUT, FORTRAN, FORTRAN IV, USER OUTPUT, USER-ORIENTED TEXT, DOCUMENTATION, TABLES, STATIC ANALYSIS, CROSS REFERENCE, STAGE OF DEVELOPMENT: IMPLEMENTED IMPLEMENTATION LANGUAGE: FORTRAN, SLLEUTH COMPUTER (OTHER HARDWARE): UNIVAC 11XX
TOOL SUMMARY: DEPTH WILL PRODUCE QUICK AND ACCURATE SUBROUTINE CROSS-REFERENCE INFORMATION. THREE TYPES OF PRINTED OUTPUT ARE: AN ALPHABETICAL CROSS-REFERENCE OF EACH SUBROUTINE AND THOSE IT REFERENCES; AN ALPHABETICAL CROSS-REFERENCE OF EACH SUBROUTINE AND THOSE THAT REFERENCE IT; A PICTORIAL DESCRIPTION OF PROGRAM LINKAGE.
DOCUMENTATION: USER'S GUIDE, PROGRAMMER'S GUIDE
REFERENCES: [AD8957], APPLIED SYSTEMS DESIGN SECTION, TRW DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE TOOLS SERIES, 790100
DEVELOPER: TRW SYSENG AND ANAL DEP
CONTACT: GARY BOEING COMPUTER SERVICES COMPANY, P.O. BOX 24346, SEATTLE, WA, 98124, USA, 213-536-1116
INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG

ACRONYM: DI-3000, TITLE: DI-3000 GRAPHICS SOFTWARE SYSTEM
CLASSIFICATION: PROGRAM CONSTRUCTION AND GENERATION
USER DESCRIPTIVE DATA TO DOCUMENT PRODUCTION RUN STREAMS, FOR OPERATIONS, DOCU/TEXT: 1) DISPLAYS EACH DATA SET WITH KEY ATTRIBUTES, IN ORDER OF OCCURRENCE, THROUGHOUT A SYSTEM; 2) LISTS SYMBOLICS AND THEIR VALUES; 3) PROVIDES RUN SEQUENCE OF JOBS WITHIN SYSTEM, PROCs WITHIN JOB, AND STEPS WITHIN PROC; AND 4) INCLUDES IN ONE LOCATION A LISTING OF REPORTS ISSUED BY A SYSTEM IN OPERATIONAL SEQUENCE. IT ALSO PROVIDES A SIMILAR LISTING FOR INPUT CARD SETS AND CARD SETS PUNCHED, FOR APPLICATION SYSTEMS PROGRAMMING. DOCU/TEXT: 1) PROVIDES THE CAPABILITY TO KEEP SYMBOLICS LEVEL FLOWCHARTS CURRENT AND ACCURATE; 2) LISTS DATA SETS IN ORDER OF FIRST OCCURRENCE AND USAGE THROUGHOUT A SYSTEM, THEREBY PROVIDING INSIGHT INTO CONSEQUENCES OF A PROPOSED FILE CHANGE; 3) LISTS CONTROL STATEMENTS BY STEP; AND 4) ANNOTATES FLOW CHARTS WITH VTD AND CATALOG DATA.

DOCUMENTATION: MAINTENANCE MANUAL
DEVELOPER: GENASYS COMPUTER PROCESSING
INFORMATION SOURCE: AIAA SURVEY OF SOFT DEV TOOLS

ACRONYM: DOCUMENT, TITLE: DOCUMENT
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE
FEATURES: SUBJECT, DATA INPUT, USER OUTPUT, USER-ORIENTED TOOL, REPORTS, STATIC ANALYSIS, SCANNING, STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: STRUCTURED FORTRAN 3
COMPUTER (OTHER HARDWARE): UNIVAC 11X
OS (OTHER SOFTWARE): ECL

TOOL AVAILABLE: YES, PUBLIC DOMAIN: YES

TOOL SUMMARY: THIS PROGRAM READS A SEQUENCE OF SYMBOLIC ELEMENTS AND WRITES OUT SPECIFIC LINES (INDEXED) FROM THESE ELEMENTS PROVIDING SUBROUTINE (HIGH LEVEL) DOCUMENTATION USING ONLY THE COMMENTS INTERNAL TO THE SYMBOLIC ELEMENTS.

DOCUMENTATION: USER'S MANUAL
DEVELOPER: JET PROPULSION LABORATORY
CONTACT: SANDY PALACIOS, JET PROPULSION LABORATORY, 4800 OAK GROVE DRIVE, PASADENA, CA, 91109, USA, 213-344-4864
INFORMATION SOURCE: AIAA SURVEY OF SOFT DEV TOOLS

ACRONYM: DOCUMENTER, TITLE: DOCUMENTER
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE
FEATURES: SUBJECT, DATA INPUT, USER OUTPUT, USER-ORIENTED TEXT, DOCUMENTATION, STATIC ANALYSIS, SCANNING, STAGE OF DEVELOPMENT: IMPLEMENTED, DATE (YYMMDD): 791000
IMPLEMENTATION LANGUAGE: BASIC
TOOL SIZE: 200 STATEMENTS
COMPUTER (OTHER HARDWARE): DEC PDP-11
OS (OTHER SOFTWARE): RSTS

TOOL AVAILABLE: YES, PUBLIC DOMAIN: YES

TOOL SUPPORTED: YES, TOOL SUPPORT: UNIVERSITY OF WAIKATO

TOOL SUMMARY: THIS PROGRAM SOLICITS INFORMATION CONCERNING THE DEVELOPMENT OF A COMPUTER SYSTEM BASED ON THE BSI STANDARD FOR PROGRAM DOCUMENTATION. IF THE USER REPLIES WITH A QUERY, AN EXPLANATION OF THE SECTION IS GIVEN. WHEN A THIRD QUERY IS GIVEN, A MAXIMUM EXPLANATION IS GIVEN, AFTER DATA HAS BEEN COLLECTED FOR A PROJECT IT IS PROCESSED BY A TEXT PROCESSOR TO PRODUCE A GOOD DOCUMENTATION LAYOUT.

DOCUMENTATION: TECHNICAL PAPER (5 PAGES)

REFERENCES: [PAYNE], A. J., PAYNE, "A DOCUMENTATION AID FOR TEACHING AND PRODUCING COMP. SCI. PROJECTS", ACM COMPUTER SCIENCE CONFERENCE, 800000
DEVELOPER: A.J. PAYNE
CONTACT: A.J. PAYNE, UNIVERSITY OF WAIKATO, HAMILTON, NEW ZEALAND
INFORMATION SOURCE: COMPLETED SUBMISSION TO NBS

ACRONYM: DOCUMENTER A, TITLE: DOCUMENTER A
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE
FEATURES: SUBJECT, CODE INPUT, USER OUTPUT, USER-ORIENTED TEXT, DOCUMENTATION, STATIC ANALYSIS, SCANNING, STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN
TOOL POPORTS, STATIC ANALYSIS: NO

TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO

RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): MARKETED COMMERCIALLY

TOOL SUPPORTED: YES, TOOL SUPPORT: SOFTOOL CORPORATION

TOOL SUMMARY: "DOCUMENTER A" AIDS EXPEDITES THE METHODS AND UNIFORM DOCUMENTATION OF SOURCE PROGRAM UNITS. THIS COMPILED-DEPENDENT TOOL PROVIDES TWO BASIC FACILITIES, THE FIRST IS A MECHANISM FOR GENERATING SOURCE PROGRAM DOCUMENTATION UNITS IN THE FORMAT OF A TEMPLATE PREDEFINED BY THE USER, SOURCE PROGRAMS ARE CODED IN A STRAIGHTFORWARD SPICE FORM AND PRESENTED TO "DOCUMENTER A" FOR AUTOMATIC EXPANSION INTO A COMPLETE DOCUMENTATION UNIT AS DETAILED IN THE PREDEFINED TEMPLATE. THE SECOND FACILITY PROVIDED BY THIS TOOL TOLERATES THE RETENTION AND RELOCATION OF STATMENTS FROM THEIR CODED LOCATION TO ANY OTHER LOCATION WITHIN THE SOURCE TEXT. "DOCUMENTER A" IS AVAILABLE FOR A NUMBER OF PROGRAMMING LANGUAGES SUCH AS FORTRAN, COBOL, AND OTHERS. "DOCUMENTER A" SERVES AS AN EXCELLENT QUALITY ASSURANCE INSTRUMENT WHICH ALLOWS TO BUILD, FACILITATE, AND ENFORCE SOURCE PROGRAM DOCUMENTATION STANDARDS.

DEVELOPER: SOFTOOL CORPORATION
CONTACT: CAROL BADDORF, SOFTOOL CORPORATION, 340 SOUTH KELLOGG AVE., GOLETA, CA, 93117, USA, 805-964-0560
INFORMATION SOURCE: COMPLETED SUBMISSION TO NBS

ACRONYM: DOCUMENTER, TITLE: DOCUMENTER
TOOLS SERIES; 790100

DEVELOPER; TRW SYS ENG AND ANAL DEP
CONTACT; J. PARNELL, TRW SYS ENG AND ANAL DEP, ONE SPACE PARK, REDONDO BEACH, CA, 90278, USA, 213-536-1116

INFORMATION SOURCE; TRW SOFTWARE TOOLS CATALOG

ACRONYM: DQM, TITLE: DESIGN QUALITY METRICS
CLASSIFICATION: REQUIREMENTS/DESIGN SPECIFICATION AND ANALYSIS
FEATURES: SUBJECT, VHDL INPUT, DESIGN SPECIFICATION, COMPLEXITY MEASUREMENT, TABLES, STATIC ANALYSIS
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN
TOOL PORTABLE: PARTIAL
COMPUTER (OTHER HARDWARE); Amdahl 470 (HP2647A OR HP2648A GRAPHIC TERMINAL)
OS (OTHER SOFTWARE); OS/MVS (ADBMS AND PLOT-10)
TOOL AVAILABLE; NO, PUBLIC DOMAIN; NO
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.); HUGHES PROPRIETARY

TOOL SUPPORTED; NO

TOOL SUMMARY: ALTHOUGH METHODOLOGIES WITH QUALITATIVE GUIDELINES ABUND, FEW IF ANY, ARE SUPPORTED WITH QUANTITATIVE MEASURABLE MEANS OF ADHERENCE, INDEED, HUGHES' STRUCTURED DESIGN METHODOLOGY FOR SOFTWARE DECOMPOSITION CONTAINS GUIDELINES SUCH AS "MAXIMIZE MODULARITY" AND "MINIMIZE DFTING" - BOTH OF WHICH HAVE NO APPARENT QUANTIFIABLE MEASURE OF COMPLIANCE. THE DESIGN QUALITY METRICS SYSTEM (DQM) IS AN INITIAL STEP TO SOLVE THIS PROBLEM, AS SHOWN ALGORITHMS HAVE BEEN DEVELOPED WHICH, WHEN APPLIED TO THE STRUCTURE CHART OF SOFTWARE MODULES, PRODUCE A QUANTIFICATION OF THE DESIGN USING A PLOT OF COMPLEXITY AS A FUNCTION OF TREE DEPTH. THE RESULTS OF THIS TECHNIQUE HAVE BEEN VERIFIED ON TWO MAJOR SOFTWARE DEVELOPMENT EFFORTS AND HAVE BEEN SHOWN TO CORRELATE CLOSELY WITH THE NUMBER OF ERRORS ENCOUNTERED IN SOFTWARE TESTING. IN OTHER WORDS, DQM QUANTIFIES QUALITATIVE GUIDELINES AND, THEREFORE, PROVIDES A PREDICTIVE TOOL WHICH CAN REDUCE THE NUMBER OF ERRORS AND THE COST.

DOCUMENTATION: USERS MANUAL
REFERENCES; [YIN879], [YIN, B., J. W. WINCHESTER, "SOFTWARE QUALITY METRICS SYSTEM", SECOND INTERNATIONAL CONFERENCE ON MATHEMATICAL MODELING, 790700 [YIN878], [YIN, B., J. W. WINCHESTER, "THE ESTABLISHMENT AND USE OF QUALITY MEASURES TO EVALUATE DESIGN QUALITY", Proc, ACM SOFTWARE QUALITY WORKSHOP, 781100
DEVELOPER; HUGHES AIRCRAFT COMPANY
CONTACT; JAMES W. WINCHESTER, HUGHES AIRCRAFT COMPANY, POST OFFICE BOX 3310, FULLERTON, CA, 92634, USA, 714-732-3232
INFORMATION SOURCE; TOOL FAIR

ACRONYM: DQM, TITLE; AUTOMATED TEST, COMPARE AND MONITOR PROGRAM
CLASSIFICATION; SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, DATA INPUT, CODE INPUT, FORTRAN, EXTENDED LISTING, USER OUTPUT, STATISTICAL ANALYSIS, COMPARISON, MANAGEMENT, TEST GENERATION, DYNAMIC ANALYSIS, REGRESSION TESTING
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN
TOOL PORTABLE; YES
COMPUTER (OTHER HARDWARE); CDC 6600/7000
TOOL SUMMARIZED; USE IN A CONTROLLED I/O ENVIRONMENT SUCH AS ACCESS TO A GLOBAL DATA BASE, IT MAY BE USED FOR DEVELOPMENT TESTING, AND/OR CHECKOUT OF UNIT SOURCE CODE, A UNIT OF CODE MAY CONSIST OF A CONTROL ROUTINE AND A NUMBER OF ROUTINES COMPRISING A FUNCTIONAL MODULE. ANY NUMBER OF TESTING TEST CASES MAY EXECUTED IN A SINGLE RUN. TEST CASES MAY BE ISOLATED BY THE PEER-ZERO OPTION OR INFORMATION MAY BE TRANSMITTED TO SUCCESSIVE CASES IF DESIRED. THE RESULTS OF EACH TEST CASE IS COMPARED TO EXPECTED VALUES AND DISCREPANCIES ARE FLAGGED. INITIAL, FINAL AND EXPECTED VALUES ARE DISPLAYED FOR EACH DISCREPANCY. THE STATEMENT LABEL EXECUTION PATH IS DISPLAYED FOR EACH TEST CASE. AT RUNS END, A SUMMARY IS DISPLAYED INDICATING THE NUMBER OF EXECTIONS OF EACH STATEMENT LABEL наличии UNEXPECTED LABELS. ALSO, A SUMMARY OF THE TEST CASES INDICATING WHICH, IF ANY, COMMENDED ERRORS MATCH SIMILARITIES TO THE AUTOREST TOOL AVAILABLE FOR THE IBM 360 COMPUTERS.

DOCUMENTATION: USER'S BRIEF [ASD579], APPLIED SYSTEM DESIGN SECTION, TRW DEFENSE AND SPACE SYSTEMS GROUP, SOFTWARE TOOLS CATALOG AND RECOMMENDATIONS, TRW AUTOMATED SOFTWARE TOOLS SERIES 790100
DEVELOPER; TRW GUIDANCE SOFTWARE DEPT
CONTACT; H. J. BARROW, TRW GUIDANCE SOFTWARE DEPT, ONE SPACE PARK, REDONDO BEACH, CA, 90278, USA, 213-536-3676
INFORMATION SOURCE; TRW SOFTWARE TOOLS CATALOG

ACRONYM: DYN A, TITLE; DYNAMIC ANALYZER
CLASSIFICATION; SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES; SUBJECT, CODE INPUT, FORTRAN, FORTRAN 66, TRANSFORMATION, INSTRUMENTATION, MACHINE OUTPUT, SOURCE CODE INPUT, FORTRAN, FORTRAN 66, USER OUTPUT, TABLES, LISTINGS, DYNAMIC ANALYSIS, COVERAGE ANALYSIS
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN 77
TOOL PORTABLE; YES
TOOL AVAILABLE; NO, PUBLIC DOMAIN; NO
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.); FOR INTERNAL USE
TOOL SUPPORTED; YES, TOOL SUPPORTED: BOEING COMPUTER SERVICES COMPANY
Dyna is a tool for Fortran programs which allows the user to see the dynamic behavior of a module while it is executing test data. The test data used is that of the users or developers. No modification to either the program or its test data is required. Dyna operates in three steps: during the preprocessing step, probes (additional Fortran statements) are automatically inserted in the source code and the "instrumented source" code is compiled, during the execution step, counts are made of number of times each statement is executed by the test data. The counts may be accumulated with the counts from previous executions, if desired. During the post processing step, the execution data is formatted into reports.


Developer: Boeing Computer Services
Contact: Leon G. Stucki, Boeing Computer Services Company, P.O. Box 24346, S-9S-03, Seattle, WA, 98124, USA, 206-575-5118

Information source: Completed submission to NBS, Tool Fair

EASYTROL

Acronym: EASYTROL, Title: EASYTROL
Classification: Software Management, Control, and Maintenance
Features: Subject, data input, user output, user-oriented text, reports, static analysis, management, project management
Stage of development: Implemented
Implementation language: COBOL
Tool size: Core: 85K
Computer (other hardware): IBM 360/370, Honeywell 6XXX
OS (other software): OS, DOS
Tool available: Yes
Restrictions (copyrights, licenses, etc.): For lease, for sale

Tool summary: EASYTROL is a project management system which provides early warnings and highlights potential trouble spots. EASYTROL allows a phase and task approach to project planning and control. It offers the user the option of overall broad planning at the beginning of the project and the detail planning of each phase just prior to beginning the phase. It also assists in the detailed planning stage. All personnel and equipment utilization schedules must be submitted at the task level. The system design is based on free formatted input. The user is required to enter the hours worked on a specific task for the period. There are ten standard reports, none of which are exception reports. The key report, project status report, presents the project at phase and task levels, and in relation to plan in the areas of man hours, equipment utilization, calendar days, and budget. Other reports present detailed budget information, individual and project level personnel information.

Developer: Keith and Associates, Inc
Information source: AIAA Survey of Soft Dev Tools

Acronym: EAVS, Title: Extensible Automated Verification System
Classification: Source program analysis and testing
Features: Subject, code input, Fortran, JOVIAL, FORTRAN IV, JOVIAL J3B-2, machine output, source code output, Fortran, JOVIAL, user output, diagnostics, listings, dynamic analysis, coverage analysis, tracing, path flow tracing
Stage of development: Implemented
Implementation language: IFTRAN
Computer (other hardware): IBM 360/370

Tool summary: EAVS is a system of compatible tools for analyzing source programs written in either the J3B-2 dialect of the JOVIAL language or IBM FORTRAN IV. EAVS is intended to be applied during program testing to aid in identifying untested paths and specifying test cases that will improve testing coverage. All of this is provided by analysis of program structure, instrumentation of the program with software probes that measure testing coverage, and generation of comprehensive reports which pinpoint paths in the program structure that remain to be exercised. In addition, guidance is provided for the generation of test cases that will assure coverage of the untested portions.

Developer: General Research Corporation
Contact: Carolyn Gannon, General Research Corporation, Santa Barbara, CA, USA, 805-965-7724
Information source: TRW Software Tools Catalog

Acronym: ECA Automation, Title: ECA Automation
Classification: Source program analysis and testing
Features: Test data generation, subject, code input, Fortran, FORTRAN 77, user output, user-oriented text, documentation, quality measurement, static analysis, auditing
Stage of development: Design
Tool available: No, public domain: No
Tool supported: No
**ECA AUTOMATION**

**TOOL SUMMARY:** The system being defined will analyse two software categories: (1) software developed by the agency for which it will provide supervisory control of the production, (2) acquired software for which it will help guarantee that rules and standards imposed by the agency on its subcontractors are followed. For both categories of software, it will also have to supply necessary information in view of maintenance requirements. Thus, the four main functions performed by the system will be: (a) software documentation assistance, (b) respect of rules and standards laid down by the agency, (c) automatic error checking, and (d) generation of sets of test data. A fifth function, using the synthesis of the information obtained will provide a quality measurement of the analysed programs. These different functions will be able to be put in service progressively and be implemented by different tools.

**DEVELOPER:** European Space Agency

**CONTACT:** Michel Poize, European Space Agency, 315, Bureaux de la Colline, Saint-Cloud, CEDEX, 92230, FRANCE.

**INFORMATION SOURCE:** European Space Agency

**ACRONYM:** EFFIGY, TITLE: EFFIGY

**CLASSIFICATION:** Source program analysis and testing

**FEATURES:** Subject, code input, PL/I, user output, listings, dynamic analysis, assertion checking, symbolic execution, tracing, breakpoint control, stage of development, implemented, date (YYMMDD): 730000

**IMPLEMENTATION LANGUAGE:** PL/I

**TOOL PORTABLE:** NO

**COMPUTER (OTHER HARDWARE):** IBM 360/370

**OS (OTHER SOFTWARE):** VM/CMS

**TOOL SUMMARY:** The EFFIGY SYSTEM is an interactive symbolic execution tool incorporating standard debug tools and expanded to include assertion checking, a simple program testing manager and a program verifier. Normal program execution is provided as a special case. EFFIGY accepts one statement at a time, building a symbolic execution tree that defines the paths through the program. A test manager is available for systematically exploring the alternatives presented in the symbolic execution tree. The program verifier generates verification conditions from user supplied assertions in conjunction with the symbolic execution.

**REFERENCES:** DONABY, John D., DONAHOO and DOROTHY SHEARLING, "A REVIEW OF SOFTWARE MAINTENANCE TECHNOLOGY", RADC-TR-80-13, INTERIM REPORT, 8002400

(KING5); J. C. KING, "A NEW APPROACH TO PROGRAM TESTING", PROCEEDINGS OF THE INTERNATIONAL CONFERENCE ON RELIABLE SOFTWARE, 750400

(KING6), J. C. KING, "SYMBOLIC EXECUTION AND PROGRAM TESTING", COMMUNICATIONS OF THE ACM, 760700

**EFFIGY**

**DEVELOPER:** IBM

**INFORMATION SOURCE:** RADC-TR-80-13, INTERIM REPORT

**ACRONYM:** ENFORCE, TITLE: ENFORCE

**CLASSIFICATION:** Source program analysis and testing

**FEATURES:** Subject, code input, COBOL, transformations, formatting, machine output, source code output, COBOL, COBOL 80, user output, diagnostics, listings, static analysis, auditing

**STAGE OF DEVELOPMENT:** IMPLEMENTED

**IMPLEMENTATION LANGUAGE:** COBOL ANSI

**TOOL PORTABLE:** NO

**TOOL AVAILABLE:** YES, PUBLIC DOMAIN: NO

**RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.):** MARKETED PRODUCT

**TOOL SUPPORTED:** YES, TOOL SUPPORT: THE PRODUCTIVITY GROUP, INC. (TPG)

**TOOL SUMMARY:** ENFORCE is a table-driven utility developed to assist systems development, quality assurance, EDP audit and systems maintenance. ENFORCE is an automated tool for program code evaluation and modification, standards enforcement and maintenance through the application of standard and user-supplied COBOL coding rules and conventions. ENFORCE properly aligns, resequences, renames, and expands relational symbols. It will modify data, paragraph, section and file names, as well as detect unacceptables COBOL code. ENFORCE will prepare programs (at your option) for the proposed ANSI COBOL-80 compiler standards.

**DOCUMENTATION:** USER'S MANUAL

**DEVELOPER:** THE PRODUCTIVITY GROUP, INC. (TPG)

**CONTACT:** THE PRODUCTIVITY GROUP, INC. (TPG), 373 FIFTH AVENUE, SUITE 1140, NEW YORK, NY, 10016, USA, 212-724-6657

**INFORMATION SOURCE:** PRODUCT DESCRIPTION

**ACRONYM:** ESAP, TITLE: EVENT SEQUENCE ANALYSIS PROGRAM

**CLASSIFICATION:** Software management, control, and maintenance

**FEATURES:** Subject, data input, code input, user output, user-directed text, documentation, graphics, flow charts, static analysis, scanning

**STAGE OF DEVELOPMENT:** IMPLEMENTED

**IMPLEMENTATION LANGUAGE:** FORTRAN

**COMPUTER (OTHER HARDWARE):** CDC 6500/7X00

**TOOL SUMMARY:** ESAP IS A GENERAL PURPOSE GRAPHICS PROGRAM WHICH AUTOMATICALLY GENERATES PLOTS ACCORDING TO USER SPECIFICATIONS. A WIDE VARIETY OF FIGURES IS AVAILABLE TO THE USER, INCLUDING BASIC FLOWCHARTING SYMBOLS. THESE FIGURES CAN BE ARRANGED IN ANY PATTERN TO REPRESENT SEQUENCE FLOW OR OTHER GRAPHIC IDEAS. THEIR POSITION, SIZE AND CONNECTIONS TO OTHER FIGURES MAY BE SPECIFIED BY THE USER AND ANNOTATIONS MAY BE PLACED WITHIN OR NEAR ANY OF THE FIGURES. DASHED AND DOTTED LINE CAPABILITIES EXIST, AND IN ADDITION TO THE REGULAR PRINT FORMAT, FIVE SPECIAL CHARACTER POINTS ARE AVAILABLE TO
ENHANCE PROPOSALS AND PRESENTATIONS, CACLOP AND VERSATEC PLOTS ARE AVAILABLE. A LARGE SAVINGS CAN BE REALIZED WHEN COMPLETED CHARTS MUST BE CHANGED OR DUPLICATED. FILE CHANGES ARE EASY TO IMPLEMENT AND THE NEW CHARTS ARE GENERATED RAPIDLY FROM THE STORED ESAP FILES.

DOCUMENTATION: USER'S GUIDE

REFERENCES: [A50379], APPLIED SYSTEMS DESIGN SECTION, TRW DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS: CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE TOOLS SERIES, 790100

DEVELOPER: TRW, SOFTWARE TECHNOLOGY DEPARTMENT

CONTACT: R. L. MAITLEN, TRW, SOFTWARE TECHNOLOGY DEPARTMENT, ONE SPACE PARK, REDONDO BEACH, CA, 90278, USA, 213-535-3480

INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG

ACRONYM: EVP, TITLE: EQUIVALENCE VERIFICATION PROGRAM

CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING

FEATURES: SUBJECT CODE INPUT, FORTRAN, FORTRAN IV, USER OUTPUT, LISTINGS, STATIC ANALYSIS, MANAGEMENT, GLOBAL VARIABLE MANAGEMENT.

STAGE OF DEVELOPMENT: IMPLEMENTED

IMPLEMENTATION LANGUAGE: FORTRAN

COMPUTER (OTHER HARDWARE): IBM 360/370

TOOL SUMMARY: OUTPUT FROM THE EQUIVALENCE VERIFICATION PROGRAM IS USED TO VERIFY THE CONSISTENCY OF THOSE GLOBAL SYMBOLS THAT ARE DEFINED BETWEEN SUBROUTINES WITH THE USE OF EQUIVALENCE STATEMENTS. THIS OUTPUT IS USEFUL IN THE DEVELOPMENT OR MAINTENANCE OF SOFTWARE SYSTEMS.

REFERENCES: [A50379], APPLIED SYSTEMS DESIGN SECTION, TRW DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS: CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE TOOLS SERIES, 790100

DEVELOPER: TRW, SOFTWARE TECHNOLOGY DEPARTMENT

CONTACT: R. L. MAITLEN, TRW, SOFTWARE TECHNOLOGY DEPARTMENT, ONE SPACE PARK, REDONDO BEACH, CA, 90278, USA, 213-535-3480

INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG

ACRONYM: EXPEDITER, TITLE: EXPEDITER

CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING

FEATURES: SUBJECT CODE INPUT, COBOL, USER OUTPUT, TABLES, LISTINGS, DYNAMIC ANALYSIS, TRACING, REGRESSION TESTING, STAGE OF DEVELOPMENT: IMPLEMENTED

IMPLEMENTATION LANGUAGE: BAL

TOOL PORTABLE: YES, TOOL SIZE: CORE: 10K-40K

TOOL AVAILABLE: YES

RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): AVAILABILITY ACCORDING TO AIR FORCE MANUAL (AFM) 300-6, PARAGRAPH 11-7A.

TOOL SUPPORTED: YES, TOOL SUPPORT: RADC/158

TOOL SUMMARY: EXPEDITER PROVIDES FACILITIES FOR UNIT TESTING OF MODULES AND PROGRAM COMPONENT TESTING IN THE DEVELOPMENT ENVIRONMENT. IT ALSO INCLUDES FEATURES FOR PROBLEM ISOLATION AND VERIFICATION OF FIXES IN THE MAINTENANCE CONTEXT. NO CHANGES TO SOURCE PROGRAMS ARE REQUIRED. IT IS RESPONSIBLE FOR IMPROVING PRODUCTIVITY IN A COBOL ENVIRONMENT FROM 10 LINES OF PROCEDURE DIVISION CODE PR PROGRAMMER DAY TO 45 LINES.

DOCUMENTATION: USER'S GUIDE (125), REFERENCE CARD (4), SPF TUTORIAL, TSO HELP, PROGRAMMED INSTRUCTION COURSE (200)

DEVELOPER: APPLICATION DEVELOPMENT SYSTEMS, INC.

CONTACT: EDWARD F. HARRIS, APPLICATION DEVELOPMENT SYSTEMS, INC, 1550 MERIDIAN AVENUE, SAN JOSE, CA, 95125, USA, 408-264-2272

INFORMATION SOURCE: COMPLETED SUBMISSION TO NBS

ACRONYM: FACES, TITLE: FORTRAN AUTOMATED CODE EVALUATION SYSTEM

CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING

FEATURES: SUBJECT CODE INPUT, FORTRAN, USER OUTPUT, DIAGNOSTICS, STATIC ANALYSIS, DATA FLOW ANALYSIS, AUDITING, STAGE OF DEVELOPMENT: IMPLEMENTED

IMPLEMENTATION LANGUAGE: FORTRAN

COMPUTER (OTHER HARDWARE): IBM 360/370

TOOL SUMMARY: THE FORTRAN AUTOMATED CODE EVALUATION SYSTEM (FACES) WAS DEVELOPED TO DETECT CODING ERRORS AND UNSOUND COBOL PRACTICE IN FORTRAN SOURCE CODE. THE SYSTEM IS COMPRISED OF A PREPROCESSOR, A PROCESSOR, AND A REPORT GENERATOR. EITHER UNIT MODULES OR INTERRELATED MODULES CAN BE RUN AS A DATA SET FOR FACES. FACES IS ORGANIZED INTO A DRIVER WITH THREE SUBSYSTEM COMPONENTS. THE MAIN DRIVER IS RESPONSIBLE FOR FILE MANIPULATIONS AND INTERPRETING USER COMMANDS. ONE OF THE COMPONENTS IS CALLED THE AUTOMATIC INTERROGATION ROUTINE. ITS PURPOSE IS TO EXAMINE TABLES GENERATED BY A FRONT-END PORTION OF FACES, AND LOOK FOR TYPES OF CODING CONSTRUCTIONS SELECTED BY THE USER. IF THE SPECIFIED CONSTRUCTIONS ARE FOUND, DIAGNOSTIC MESSAGES ARE RECORDED ON THE FLAG FILE. A REPORT GENERATOR GENERATES USER REPORTS, AREAS OF CODING THAT CANNOT BE EFFECTIVELY EVALUATED ARE ALSO REPORTED TO THE USER.

DOCUMENTATION: SYSTEM DESCRIPTION

REFERENCES: [A50379], APPLIED SYSTEMS DESIGN SECTION, TRW DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS: CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE TOOLS SERIES, 790100

(DONAO), JOHN D. DONAO AND DOROTHY SWERINGER, "A REVIEW OF SOFTWARE MAINTENANCE TECHNOLOGY", RADC-TR-80-13, 80 INTERIM REPORT, 80-200

DEVELOPER: COSMIC, UNIVERSITY OF GEORGIA

CONTACT: REX WALKER, COSMIC, UNIVERSITY OF GEORGIA, SUITE 112, BARRON HALL, ATHENS, GEORGIA, 30602, USA, 404-542-3265

INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG, RADC-TR-80-13, INTERIM REPORT

ACRONYM: FADEBUG=I, TITLE: FACOM AUTOMATIC DEBUG

K-42
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: PATH STRUCTURE ANALYSIS, SUBJECT, CODE INPUT, USER OUTPUT, STATIC ANALYSIS, COMPARISON, I/O SPECIFICATION ANALYSIS, STAGE OF DEVELOPMENT: IMPLEMENTED IMPLEMENTATION LANGUAGE: ASSEMBLY
COMPUTER (OTHER HARDWARE): FACOM 230-60
TOOL AVAILABLE: YES
TOOL SUPPORT: YES, TOOL SUPPORT: FUJITSU LTD.
TOOL SUMMARY: FADEBUG-I HAS TWO PRIMARY FUNCTIONS AS A DEBUG AID, COMPARING THE SET OF OUTPUT DATA PRODUCED BY A PROGRAM WITH USER-SPECIFIED OUTPUT DATA IS IDENTIFIED AS ITS MOST IMPORTANT FUNCTION. THE OTHER FUNCTION INVOLVES AUTOMATIC ISOLATION AND DEFINITION OF ALL POSSIBLE EXECUTION PATHS FROM ENTRY TO EXIT IN A PROGRAM MODULE, THESE CAPABILITIES AID IN DETECTING AND REMOVING PROGRAM BUGS, IN THE MODULE TEST STAGE OF PROGRAM DEVELOPMENT THE FOLLOWING AREAS OF DIFFICULTY ARE IDENTIFIED: (1) EXAMINATION AND VERIFICATION OF OUTPUT DATA FROM MODULE TEST EXECUTION, (2) EXAMINATION OF MODULE PROCESSING PATHS FOR LOGICAL ERRORS, (3) EVALUATION OF MODULE LOGIC PATHS FOR OMISSIONS. FADEBUG-I IS DESIGNED TO REDUCE OR ELIMINATE THESE DIFFICULTIES THROUGH ITS TEST FUNCTION OR ROUTE DEFINITION FUNCTION.
ACRONYM: FASP; TITLE: FACILITY FOR AUTOMATED SOFTWARE PRODUCTION
CLASSIFICATION: SOFTWARE SUPPORT SYSTEM/PROGRAMMING ENVIRONMENT
FEATURES: SUBJECT, TEXT INPUT, DATA INPUT, CODE INPUT, TRANSFORMATION, TRANSLATION, EDITING, INSTRUMENTATION, FORMATTING, MACHINE OUTPUT, SOURCE CODE OUTPUT, OBJECT CODE OUTPUT, TEXT OUTPUT, DIAGNOSTICS, LISTINGS, STATIC ANALYSIS, COMPARISON, CROSS REFERENCE, MANAGEMENT, ERROR CHECKING, DYNAMIC ANALYSIS, COVERAGE ANALYSIS, TUNING, TIMING, STAGE OF DEVELOPMENT: IMPLEMENTED IMPLEMENTATION LANGUAGE: FORTRAN, COMPASS, KCL
COMPUTER (OTHER HARDWARE): CDC CYBER, CDC 6X00/7X00
OS (OTHER SOFTWARE): KRONOS
TOOL AVAILABLE: YES, PUBLIC DOMAIN: YES
TOOL SUMMARY: THE FASP IS AN INTEGRATED SET OF SOFTWARE TOOLS WHICH PROVIDES AN ADVANCED PROGRAMMING SYSTEM AND A SOFTWARE MANAGEMENT INFORMATION SYSTEM FOR PRODUCTION AND LIFE CYCLE MAINTENANCE OF PROJECT SOFTWARE. IT PROVIDES SUPPORT FOR THE NAVY'S HIGH ORDER LANGUAGES AND STANDARD MILITARY COMPUTERS AND INCLUDES A COMPREHENSIVE SET OF SOFTWARE ENGINEERING AND MANAGEMENT TOOLS. FASP IS A FUNCTIONALLY ORIENTED SOFTWARE PRODUCTION SYSTEM WHICH CAN BE USED IN EITHER AN INTERACTIVE OR BATCH MODE OF OPERATION AND IS ACCESSIBLE FROM ANY LOCATION THROUGH THE USE OF TERMINALS. FASP SUPPORT CAPABILITIES INCLUDE EDITORS/LIBRARIANS, TRANSLATORS/ PREPROCESSORS, SYSTEM GENERATORS, SOFTWARE EMULATORS/DEBUGGERS, TEST ANALYZERS, DOCUMENTATION: USER'S MANUAL, MAINTENANCE MANUAL REFERENCES: (REIF81), D. J. REIF AND H. A. MONTGOMERY, "SEATEC SOFTWARE TOOLS SURVEY", RCIC-TR-008, REIFER CONSULTANTS, INC., 810300
(STE80), H. G. STUEBIG, "A MODERN FACILITY FOR SOFTWARE PRODUCTION AND MAINTENANCE", PROCEEDINGS OF COMPAC, 811000
DEVELOPER: NAVAL AIR DEVELOPMENT CENTER
TOOL SUMMARY: FAVS, AN INTEGRATED COLLECTION OF COMPUTER PROGRAMS, WAS DEVELOPED FOR THE PURPOSE OF ASSURING THAT SOFTWARE SYSTEMS FORTIFIED IN FORTAN ARE COMPREHENSIVELY TESTED. FAVS PROVIDES (1) STATIC DETECTION OF UNREACHABLE STATEMENTS, SET/USE ERRORS, MODE-CONVERSION ERRORS, AND EXTERNAL REFERENCE ERRORS, (2) A MEANS OF MEASURING THE EFFECTIVENESS OF TEST CASES BY SOURCE CODE INSTRUMENTATION, (3) ASSISTANCE IN THE CONSTRUCTION OF TEST DATA THAT WILL THOROUGHLY EXERCISE THE SOFTWARE, AND (4) AUTOMATED DOCUMENTATION. IN ORDER TO AID IN THE PRODUCTION OF APPLICATION SOFTWARE THAT ADHERES TO MODERN PROGRAMMING TECHNIQUES, FAVS ALSO PROVIDES FOR THE TRANSFORMATION FROM DMATAN (A STRUCTURED EXTENSION OF FORTAN) TO FORTAN AND VICE VERSA.

DOCUMENTATION: VOL. 1, FINAL REPORT (49), VOL. 2, FAVS USER'S MANUAL (50), VOL. 3, DMATAN USER'S GUIDE (46).


DEVELOPER: GENERAL RESEARCH CORPORATION

CONTACT: FRANK S. LAMONICA, RADC/ISIE, GRIFFISS AFB, NY, 13454, USA, 515-370-7949

ACRONYM: FCA, TITLE: FORTRAN CODE AUDITOR

CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING FEATURES: SUBJECT, CODE INPUT, FORTAN, USER OUTPUT, DIAGNOSTICS, LISTINGS, STATIC ANALYSIS, AUDITING, STRUCTURE CHECKING

STAGE OF DEVELOPMENT: IMPLEMENTED

IMPLEMENTATION LANGUAGE: FORTRAN

COMPUTER (OTHER HARDWARE): HONEYWELL 6XX, HONEYWELL 6XX

TOOL SUMMARY: THE FORTRAN CODE AUDITOR, AN AUTOMATED TEST TOOL, IS USED FOR THE COST EFFECTIVE ENFORCEMENT OF FORTRAN PROGRAMMING STANDARDS AND CONVENTIONS APPLICABLE TO THE AIR FORCE SOFTWARE ENVIRONMENT. IT DOES NOT MODIFY CODE, USING PREDEFINED CODING STANDARDS AND CONVENTIONS, IT SIMPLY ADVISES THE USER WHERE THESE STANDARDS AND CONVENTIONS HAVE NOT BEEN ADHERED TO. THE MAJOR ADVANTAGE OF FAVORING AN AUTOMATED AUDITOR OVER MANUALLY METHODS, BECAUSE COST EFFECTIVENESS, IS COMPLETE OBJECTIVITY AND UNAMBIGUITY. THE STANDARDS CAN BE VIEWED AS BEING CODING ENFORCEMENTS IN FOUR AREAS: STANDARDS DEFINING QUANTITY AND PLACEMENT OF COMMENTARY, STANDARDS IDENTIFYING PHYSICAL PLACEMENT AND GROUPING OF CODE ELEMENTS ON THE SOURCE CODE LISTING, STANDARDS LIMITING MODULE SIZE AND PLACING RESTRICTIONS ON THE USE OF CERTAIN INSTRUCTIONS WITH THE END RESULT OF PROVIDING THE OPTIMIZATION OF CODE RELATIVE TO EXECUTION TIME, AND STANDARDS REQUIRING THE USE OF STRICT RULES FOR THE TOP-DOWN DESIGN AND IMPLEMENTATION OF A SYSTEM OF PROGRAMS.
FOCUS

CLASSIFICATION: PROGRAM CONSTRUCTION AND GENERATION
FEATURES: SUBJECT, DATA INPUT, VMLL INPUT, DESCRIPTION LANGUAGE, INFORMATION MANAGER, INTERPRETATION, SYNTHESIS, USER OUTPUT, DIAGNOSTICS, GRAPHICS, TABLES, STATIC ANALYSIS, MANAGEMENT, DATABASE MANAGEMENT

STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN

TOOL PORTABLE: NO
COMPUTER (OTHER HARDWARE): IBM 360/370
OS (OTHER SOFTWARE): D5, IVS, VMS, CMS, TSO, QS/VS, Q/A/MVS
TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): LICENSED IN-HOUSE USE, INFORMATION BUILDERS, USAGE BASED, TYMSHARE

TOOL SUMMARY: FOCUS IS AN INTERACTIVE INFORMATIONAL CONTROL SYSTEM THAT CONTAINS FACILITIES FOR DESCRIBING FILES, FOR ENTERING, CHANGING AND DELETING RECORDS IN FILES, AND FOR REPORTING, GRAPHING, MODELLING AND STATISTICAL ANALYZING DATA FROM FILE INFORMATION. FOCUS CONTAINS MANY DBMS TYPE FACILITIES AND CAN ACCESS DATA FROM IBM'S IMS AND CULLINANE'S IMS DATABASES AS WELL AS FROM FOCUS CREATED FILES. FEATURES INCLUDE: HIERARCHICAL AND RELATIONAL FILE STRUCTURES, INTERACTIVE ENGLISH LANGUAGE REPORT WRITING, GRAPHING, STATISTICS, FILE MANAGEMENT, 1270 FULL SCREEN FORMATTED DATA ENTRY, FINANCIAL MODELLING, INTERFACES TO IMS, IDS, VSAM AND ISAM FILES

DOCUMENTATION: USER'S MANUAL (500), PRIMER

REFERENCES: [COMP79], COMPUTER DECISIONS MAGAZINE, "BANK CUSTOMER QUERIES PUT INTO FOCUS", COMPUTER DECISIONS, 769/000

[COMP79], COMPUTERWORLD, "FOCUS ADOES VM FILE USE", COMPUTERWORLD, 79/07/16

DEVELOPER: INFORMATION BUILDERS
CONTACT: INFORMATION BUILDERS, INC. 294 W. 31ST STREET, NEW YORK, NY, 10001, USA, 212-736-4438

INFORMATION SOURCE: COMPLETED SUBMISSION TO NBS

ACRONYM: FORAN, TITLE: FORAN ANALYZER PROGRAM
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, FORTRAN, USER OUTPUT, DIAGNOSTICS, TABLES, STATIC ANALYSIS, INTERFACE ANALYSIS, CROSS REFERENCE, CONSISTENCY CHECKING, ERROR CHECKING

STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN
COMPUTER (OTHER HARDWARE): CDC 6600/7X00

TOOL AVAILABLE: YES
TOOL SUPPORTED: YES, TOOL SUPPORT: U.S. ARMY ADVANCED RESEARCH CENTER

TOOL SUMMARY: FORAN PERFORMS STATIC ANALYSIS ON SOURCE CODE WRITTEN IN ANY DIALECT OF FORTRAN. USAGE OF PROGRAM LABELS, TAGS, DATA VARIABLES, CONSTANTS, SUBROUTINES, AND OTHER PROGRAM ELEMENTS ARE ANALYZED FOR A MAIN PROGRAM AND ITS RELATED SUBROUTINE COMPONENTS. EACH NAME IS LISTED, SHOWING THE STATEMENT NUMBERS WHERE THE ITEM IS REFERENCED AND HOW IT IS REFERENCED (ASSIGNED, USED, INPUT, OUTPUT, SUBROUTINE CALL, ETC.). FORAN ALSO IDENTIFIES SYMBOLS DEFINED BUT NOT USED, DISCREPANCIES IN VARIABLE TYPE, RECORD DIMENSION, AND NUMBER AND TYPE OF PARAMETERS IN FUNCTIONS AND SUBROUTINES. SYNTAX ERRORS ARE FLAGGED DURING THE ANALYSIS. FORAN'S PRIMARY USE IS TO DETERMINE POSSIBLE COMPUTATION OF LOGIC ERRORS FROM THE STATIC ANALYSIS OF DATA USAGE. IT IS ALSO VALUABLE IN ANALYZING THE EFFECTS OF A PROGRAM MODIFICATION ON DATA USAGE.

REFERENCES: [DONAB01], JOHN O. DONAHUE AND DOROTHY SWEARNERGIR, "A REVIEW OF SOFTWARE MAINTENANCE TECHNOLOGY", RADC-TR-60-13, INTERIM REPORT, 60/2000

[INFIN79], M. FINER, "SOFTWARE DEBUGGING METHODOLOGY", FINAL TECHNICAL REPORT, RADC-TR-79-57, THREE VOLS., 78/400 DEVELOPED BY U.S. ARMY ADVANCED RESEARCH CENTER, MONTGOMERY, ALABAMA

INFORMATION SOURCE: RADC-TR-60-13, INTERIM REPORT

ACRONYM: FORAN, TITLE: FORAN
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE
FEATURES: SUBJECT, CODE INPUT, FORTRAN, USER OUTPUT, LISTINGS, STATIC ANALYSIS, MANAGEMENT, GLOBAL VARIABLE MANAGEMENT

STAGE OF DEVELOPMENT: IMPLEMENTED
THE IMS, IDS, JAM, AND ISAM FORTRAN COMPUTER (OTHER HARDWARE): IBM 360/370


DOCUMENTATION: PROGRAM DESCRIPTION
REFERENCES: [NASI79], APPLIED SYSTEMS DESIGN SECTION, TRW DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS: CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE TOOLS SERIES, 79/0100

DEVELOPER: TRW, DEFENSE SYSTEMS SOFTWARE DEPARTMENT
CONTACT: A. J. DESALVIO, TRW, DEFENSE SYSTEMS SOFTWARE DEPARTMENT, ONE SPACE PARK, REDONDO BEACH, CA, 90278, USA, 213-536-3083

INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG
ACRONYM: FORREF, TITLE: FORTRAN CROSS-REFERENCE
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE
FEATURES: SUBJECT, CODE INPUT, FORTRAN, FORTRAN V, USER OUTPUT, TABLES, STATIC ANALYSIS, CROSS REFERENCE, STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN, SLEUTH
COMPUTER (OTHER HARDWARE): UNIVAC 11XX
TOOL SUMMARY: FORREF PROVIDES THE FORTRAN USER WITH DETAILED CROSS-REFERENCE TABLES OF SELECTED FORTRAN SYMBOIC ELEMENTS. THE THREE PRINCIPAL TYPES OF PRINTED OUTPUT ARE: A VERSUS CROSS-REFERENCE TABLE OF ALL PROGRAM VARIABLES V PROGRAM LINE NUMBERS INCLUDING THE NATURE OF THE CROSS-REFERENCE (E.G., USED, CALLED); A TRANSFER CROSS-REFERENCE LISTING ALL TRANSFER POINTS WITH THE SYMBOIC ELEMENT; AND A STATEMENT NUMBER CROSS-REFERENCE TABLE INDICATE ALL PROGRAM LINE NUMBERS ON WHICH EACH STATEMENT NUMBER IS REFERENCED.
DOCUMENTATION: USER'S GUIDE, PROGRAMMER'S GUIDE
REFERENCES: {ASD079}, APPLIED SYSTEMS DESIGN SECTION, TRW DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE TOOLS SERIES, 790100
DEVELOPER: TRW SYS ENG AND ANAL DEP
CONTACT: J. PARNELL, TRW SYS ENG AND ANAL DEP, ONE SPACE PARK, REDONDO BEACH, CA, 90278, USA, 213-536-1116
INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG
ACRONYM: FORTRAN OPTIMIZ, TITLE: FORTRAN OPTIMIZATION INSTRUMENTERS
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, FORTRAN, TRANSFORMATION, INSTRUMENTATION, MACHINE OUTPUT, SOURCE CODE OUTPUT, FORTRAN, USER OUTPUT, TABLES, LISTINGS, DYNAMIC ANALYSIS, TUNING,
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN
TOOL PORTABLE: YES
TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): MARKETED PRODUCT
TOOL SUPPORTED: YES, TOOL SUPPORT: SOFTWARE CORPORATION
ACRONYM: FORTRAN TRACING, TITLE: FORTRAN TRACING INSTRUMENTERS
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, FORTRAN, TRANSFORMATION, INSTRUMENTATION, MACHINE OUTPUT, SOURCE CODE OUTPUT, FORTRAN, USER OUTPUT, TABLES, LISTINGS, DYNAMIC ANALYSIS, TRACING, PATH FLOW TRACING,
FORTRAN TRACING

STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN
TOOL PORTABLE: YES
TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): MARKETED PRODUCT
TOOL SUPPORTED: YES, TOOL SUPPORT: SOFTOOL CORPORATION
TOOL SUMMARY: THESE ARE TWO TOOLS THAT AUTOMATICALLY DOCUMENT THE PATH OF PROGRAM CONTROL FLOW FROM MODULE (STATEMENT) TO MODULE (STATEMENT). THE FIRST TOOL OPERATES AT THE ROUTINE LEVEL, THE SECOND TOOL OPERATES AT THE STATEMENT LEVEL. THESE PRODUCTS OFFER THE SOFTWARE PROFESSIONAL A FLEXIBLE, CONSISTENT AND EASY TO USE TRACING FACILITY. THESE TOOLS REQUIRE NO MODIFICATION OF ANY COMPILER OR APPLICATION PROGRAM, THEY SIMPLY ACCEPT AS INPUT SOURCE PROGRAMS AND TEST DATA, AND OUTPUT CLEAR TRACE DOCUMENTATION (I.E., PROFILES) WHICH IS FORMATTED AND INDENTED TO FACILITATE UNDERSTANDING. THEY PERMIT TOP-DOWN TRACING IN A NATURAL MANNER. THE TRACING INSTRUMENTERS ARE MEMBERS OF SOFTOOL 80, AN INTEGRATED SET OF TOOLS MARKETED BY SOFTOOL CORPORATION.

DOCUMENTATION: TECHNICAL REPORTS, USERS MANUAL
DEVELOPER: SOFTOOL CORPORATION
CONTACT: CAROL BADDORF, SOFTOOL CORPORATION, 340 SOUTH KELLOGG AVE., GOLETA, CA, 93117, USA, 805-964-0560
INFORMATION SOURCE: COMPLETED SUBMISSION TO NBS

ACRONYM: FORTREF
TITLE: FORTRAN SYMBOLIC NAME CROSS-REFERENCE
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE
FEATURES: SUBJECT, CODE INPUT, FORTRAN, FORTRAN IV, USER OUTPUT, TABLES, STATIC ANALYSIS, CROSS REFERENCE, SCANNING
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN
COMPUTER (OTHER HARDWARE): IBM 360/370, COD 6X00/7X00, UNIVAC 11XX, HONEYWELL 6XX, HONEYWELL 6XX
TOOL SUMMARY: FORTREF PRODUCES FROM FORTRAN SOURCE PROGRAMS A COMPREHENSIVE SYMBOIC NAME CROSS-REFERENCE DICTIONARY TO ASSIST ANALYSTS AND PROGRAMMERS IN DEBUGGING, CHANGING AND CONVERTING THEIR FORTRAN PROGRAMS. IT IS ORGANIZED ALPHABETICALLY BY SYMBOIC NAME AND SHOWS THE ACTUAL FORTRAN STATEMENT IN WHICH THE SYMBOIC NAME APPEARS ALONG WITH THE ASSOCIATED PROGRAM NAME. THE DICTIONARY ALSO INCLUDES SUCH SELECTED FORTRAN STATEMENT NAMES AS READ, WRITE, FORMAT, CALL, ENTRY, STOP, AND RETURN.

DOCUMENTATION: USER'S MANUAL, PROGRAM MANUAL
REFERENCE: "AS08079", APPLIED SYSTEMS DESIGN SECTION, TRW DATA PROCESSING SYSTEMS GROUP, "SOFTWARE TOOLS CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE TOOLS SERIES, 790100
CONTACT: WILLIAM F. DEHAAN, B. D. BOX 101, ROCKAWAY, NJ, 07866, USA, 201-627-6453

FORTRF

INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG
ACRONYM: FOSTRA, TITLE: FORTRAN STRUCTURING AID
CLASSIFICATION: REQUIREMENTS/DESIGN SPECIFICATION AND RESTRUCTURING
FEATURES: SUBJECT, CODE INPUT, FORTRAN, TRANSLATION, TRANSLATION, MACHINE OUTPUT, SOURCE CODE OUTPUT, SFRTRAN, USER OUTPUT, GRAPHICS
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN, STRUCTURED FORTRAN
TOOL SIZE: CORE: 55K
COMPUTER (OTHER HARDWARE): DEC PDP-11
OS (OTHER SOFTWARE): RSX-11
TOOL AVAILABLE: YES, PUBLIC DOMAIN: YES
TOOL SUMMARY: FOSTRA IS A GRAPHICALLY INTERACTIVE SYSTEM USING ELECTRONIC PENCIL THAT IS COGNIZANT OF STRUCTURED CODE AND BLOCK STRUCTURING AND IS USED TO FACILITATE CONVERSION OF FORTRAN TO SFRTRAN. SFRTRAN IS STRUCTURED FORTRAN PREPROCESSOR USED AT JPL.

DOCUMENTATION: TECHNICAL PAPER
DEVELOPER: JET PROPULSION LABORATORY
CONTACT: SANDY PALACIOS, JET PROPULSION LABORATORY, 4800 OAK GROVE DRIVE, PASADENA, CA, 91109, USA, 213-344-4864
INFORMATION SOURCE: AIAA SURVEY OF SOFT DEV TOOLS

ACRONYM: FTM-77 ANALYZER, TITLE: A FORTRAN-77 ANALYZER
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, FORTRAN, FORTRAN 77, TRANSFORMATION, INSTRUMENTATION, MACHINE OUTPUT, SOURCE CODE OUTPUT, FORTRAN 77, USER OUTPUT, TABLES, LISTINGS, STATISTICAL ANALYSIS, STATISTICAL ANALYSIS, PROFILE GENERATION, DYNAMIC ANALYSIS, DYNAMIC ANALYSIS, COVERAGE ANALYSIS, ASSERTION CHECKING, RUN-TIME VERIFICATION, TUNING, TRACING
STAGE OF DEVELOPMENT: IMPLEMENTED DATE (YMMDD): 810500
IMPLEMENTATION LANGUAGE: FORTRAN 77
TOOL PORTABLE: YES
TOOL AVAILABLE: YES, PUBLIC DOMAIN: YES
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): AVAILABLE THROUGH FEDERAL SOFTWARE EXCHANGE CENTER, FSWECC=(NOT ASSIGNED)
TOOL SUPPORTED: NO
TOOL SUMMARY: THE NBS FORTRAN-77 ANALYZER CONSISTS OF TWO PARTS: STATIC ANALYSIS AND DYNAMIC ANALYSIS. STATIC ANALYSIS COLLECTS FROM INDIVIDUAL PROGRAMS FREQUENCY STATISTICS ON FORTRAN STATEMENT TYPES. SUCH ACCUMULATIONS ARE USEFUL FOR LANGUAGE STUDIES, INPUT TO STANDARDS COMMITTEES, SYSTEM BENCHMARK CHARACTERIZATION, AND STUDIES IN SOFTWARE PORTABILITY AND CONVERSION. DYNAMIC ANALYSIS COLLECTS EXECUTION FREQUENCIES OF CODE SEGMENTS (A CODE SEGMENT IS AN EMPTY SEQUENCE OF STATEMENTS WITH A UNIQUE ENTRY AND A SOLE EXIT). INFORMATION ON THE CORRECTNESS OF USER-IMBEDDED ASSERTIONS, AND TRACE DATA, THIS DATA IS USEFUL IN TESTING, DEBUGGING, AND OPTIMIZING PROGRAMS.
FTN-77 ANALYZER

DOCUMENTATION: USER'S MANUAL (95 PAGES), MAINTENANCE MANUAL (313 PAGES), TEST PLAN (91 PAGES), DESIGN SPECIFICATION (95 PAGES), SOFTWARE SUMMARY (1 PAGES), FUNCTIONAL REQUIREMENTS DOCUMENT (46 PAGES)

REFERENCES: [LYON71], LYON AND STILLMAN, "A FORTRAN ANALYZER", NSB TECHNICAL NOTE 869, 741000 [LYON73], LYON, "STATIC LANGUAGE ANALYSIS", NSB TECHNICAL NOTE 797, 731000

DEVELOPER: TRW DEFENSE AND SPACE SYSTEMS GROUP, ONE SPACE PARK, REDONDO BEACH, CA

CONTACT: GSA FEDERAL SOFTWARE EXCHANGE, 2 SKYLINE PL(11TH FL), 5203 LEESBURG PK, FALLS CHURCH, VA, 22041, USA, 703-756-2610

INFORMATION SOURCE: COMPLETED SUBMISSION TO NSB, TOOL FAIR

ACRONYM: FTN ANALYZER, TITLE: FTN ANALYZER
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, FORTRAN, FORTRAN 66, TRANSFORMATION, INSTRUMENTATION, MACHINE OUTPUT, SOURCE CODE OUTPUT, FORTRAN, USER OUTPUT, TABLES, LISTINGS, STATISTICAL ANALYSIS, STATISTICAL ANALYSIS, PROFILE GENERATION, DYNAMIC ANALYSIS, COVERAGE ANALYSIS, TUNING

STAGE OF DEVELOPMENT: IMPLEMENTED, DATE (YYMMDD): 741000

IMPLEMENTATION LANGUAGE: FORTRAN 66

TOOL PORTABLE: YES, TOOL SIZE: 5665 STATEMENTS (60% COMMENTS)

TOOL AVAILABLE: YES, PUBLIC DOMAIN: YES

RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): AVAILABLE THROUGH FEDERAL SOFTWARE EXCHANGE CENTER, FSTEC-77/0264

TOOL SUPPORTED: NO

TOOL SUMMARY: THE ANALYZER CONSISTS OF TWO PARTS: STATIC ANALYSIS AND DYNAMIC ANALYSIS. STATIC ANALYSIS COLLECTS FROM INDIVIDUAL PROGRAMS 118 FREQUENCY STATISTICS ON FORTRAN STATEMENT TYPES AND CAN ACCUMULATE THESE STATISTICS OVER ALL PROGRAMS ANALYZED. SUCH ACCUMULATIONS ARE USEFUL FOR LANGUAGE STUDIES. INPUT TO STANDARDS COMMITTEE, SYSTEM CHARACTERIZATION, AND STUDIES IN SOFTWARE PORTABILITY AND CONVERSION. DYNAMIC ANALYSIS COLLECTS EXECUTION FREQUENCIES OF CODE SEGMENTS (A CODE SEGMENT IS A NON-EMPTY SEQUENCE OF STATEMENTS WITH A UNIQUE ENTRY AND A SOLE EXIT), EXECUTION FREQUENCIES ARE USEFUL FOR OPTIMIZATION AND TESTING OF PROGRAMS.

DOCUMENTATION: TECHNICAL DESCRIPTION (23)

REFERENCES: [LYON74], LYON AND STILLMAN, "A FORTRAN ANALYZER", NSB TECHNICAL NOTE 869, 741000 [STIL75], STILLMAN AND LEE, "SOFTWARE TESTING FOR NETWORK SERVICES", NSB TECHNICAL NOTE 874, 750000 [LYON75], LYON AND STILLMAN, "SIMPLE TRANSFORMS FOR INTERPRETING FORTRAN DECKS", SOFTWARE PRACTICE AND EXPERIENCE, 750000 [LYON73], LYON, "STATIC LANGUAGE ANALYSIS", NSB TECHNICAL NOTE 797, 731000

FTN ANALYZER

(FSEC60), GENERAL SERVICES ADMINISTRATION/NATIONAL TECHNICAL INFORMATION SERVICE, "FEDERAL SOFTWARE EXCHANGE CATALOG", GSA/ADTS/C-80/1, PB80-904001, 800100

DEVELOPER: NATIONAL BUREAU OF STANDARDS (ICST), WASHINGTON, DC 20234

CONTACT: GSA FEDERAL SOFTWARE EXCHANGE, 2 SKYLINE PL(11TH FL), 5203 LEESBURG PK, FALLS CHURCH, VA, 22041, USA, 703-756-2610

INFORMATION SOURCE: COMPLETED SUBMISSION TO NSB, FEDERAL SOFTWARE EXCHANGE CATALOG

ACRONYM: FTNCODER, TITLE: FTNCODER
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE
FEATURES: SUBJECT, DATA INPUT, CODE INPUT, FORTRAN, FORTRAN 66, TRANSFORMATION, FORMATTING, MACHINE OUTPUT, SOURCE CODE OUTPUT, FORTRAN, CDC FORTRAN, USER OUTPUT, LISTINGS,

STAGE OF DEVELOPMENT: IMPLEMENTED

IMPLEMENTATION LANGUAGE: PASCAL

COMPUTER (OTHER HARDWARE): CDC 6600/7000

TOOL AVAILABLE: YES

TOOL SUMMARY: FTNCODER PRODUCES ONE OR MORE "STANDARD" CDC FORTRAN (FTN) PROGRAM MODULE FROM AN INPUT FILE OF LINES CONSISTING OF: MODULE NAME, AND A SHORT COMMENT DESCRIBING THE MODULE'S PURPOSE, EACH OF THE MODULES PRODUCED INCLUDES SUITABLY MARKED COMMENTS FOR THE HEADER (INCLUDING THE PURPOSE), MODULE NO., DATE, AND TIME) AND A PLACE TO INSERT NOTES RELATIVE TO THE MODULE. CONTROL DATA CORPORATION "UPDATE" (REF. A) CONDITIONAL SOURCE CODE BLOCKS ARE WRITTEN AT THE ENTRY AND EXIT OF EACH MODULE TO MARK THE NAME AND TIME OF ENTRY, AND THE TOTAL ELAPSED TIME IN THE MODULE. THESE BLOCKS ARE LOGICALLY DELETABLE WHEN THE PROGRAM UNDER DEVELOPMENT REACHES THE PRODUCTION STAGE; HOWEVER, THEY REMAIN IN THE SOURCE CODE LIBRARY, UNIQUENESS OF TEXT MARKS PRODUCED BY "FTNCODER" ALLOWS EASY LOCATION OF A SPECIFIC PORTION OF THE MODULES WITH A LINE ORIENTED TEXT EDITOR. SUFFICIENT "WHITE SPACE" IS INCLUDED IN EACH MODULE TO ENHANCE READABILITY OF THE SOURCE CODE.

DEVELOPER: NSDC

CONTACT: ROBERT N. ROTH, NSDC, STRUCTURES DEPARTMENT, BETHESDA, MD, 20044, USA, 202-227-1851

INFORMATION SOURCE: COMPLETED SUBMISSION TO NSB

ACRONYM: FTNRCODER, TITLE: FTNCODER
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, FORTRAN, CDC EXTENDED FORTRAN, FORTRAN 66, TRANSFORMATION, DIAGNOSTIC, USER-ORIENTED TEXT, DOCUMENTATION, TABLES, STATIC ANALYSIS, CROSS-REFERENCE, MANAGEMENT, CHANGE CONTROL, GLOBAL VARIABLE MANAGEMENT,

STAGE OF DEVELOPMENT: IMPLEMENTED

IMPLEMENTATION LANGUAGE: PASCAL 6000
IMPLEMENTATION LANGUAGE: PASCAL
COMPUTER (OTHER HARDWARE): IBM 360/370, CII=HB
OS (OTHER SOFTWARE): OS/MVS, GCOS, MULTICS
TOOL SUMMARY: THIS TOOL IS A PROGRAMMED TEST CASE GENERATOR FOR TESTING PROGRAM MODULES AND PARALLEL PROGRAMS. IT HAS THREE MAIN FUNCTIONS: (1) GENERATION OF ALL POSSIBLE CONFIGURATIONS OF THE CALLING PARAMETERS FOR EACH FUNCTION OF THE MODULE BEING TESTED, (2) INTERPRETATION OF TEST CASES WRITTEN IN A SYMBOLIC "COMMAND LANGUAGE" FOR TESTING THE CONNECTION OF FUNCTION, AND (3) EDITING OF TEST RESULTS AND THE MODULE'S OWN DATA.

DOCUMENTATION: USER'S MANUAL

ACRONYM: GENTESTS, TITLE: GENTESTS
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, MACHINE OUTPUT, SOURCE CODE OUTPUT, STATIC ANALYSIS, I/O SPECIFICATION ANALYSIS
STAGE OF DEVELOPMENT: IMPLEMENTED
TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): LICENSE REQUIRED
TOOL SUPPORTED: YES, TOOL SUPPORTS: PRECISION VISUALS, INC.
DESCRIPTION: GRAFMAKER IS AN INTEGRATED SYSTEM OF GRAPHICS
SOFTWARE TOOLS FOR DESIGNING AND VIEWING LINE GRAPHS, BAR
GRAPHS AND PIE CHARTS. BECAUSE GRAFMAKER IS BUILT ON TOP
OF DI-3000, IT IS BOTH DEVICE INDEPENDENT AND MACHINE
INDEPENDENT. GRAFMAKER HAS BEEN DEVELOPED FOR USE BY
FORTAN APPLICATION PROGRAMMERS. A CHART OR GRAPH IS
DEVELOPED AS A MODEL USING A COMBINATION OF "GENERIC"
CAPABILITIES COMMON TO MOST DATA DISPLAY APPLICATIONS
(E.G., ANNOTATION, LEGENDS, TIK MARKS, AXES AND PIE CHART
SEGMENTS). FEATURES INCLUDE EXPLOSION OF PIE CHART
SEGMENTS, METAFILE OUTPUT, BACKGROUND COLOR SELECTION,
MULTIPLE AXES, ARBITRARY AXIS POSITIONING, FILLING BETWEEN
TWO CURVES ON A LINE GRAPH, BACKGROUND GRIDS, LOG AND
NATURAL LOG SCALING OPTIONS AND MULTIPLE CHARTS OR GRAPHS
ON A SINGLE PICTURE.
DOCUMENTATION: USER'S GUIDE (123 PAGES)
CONTACT: PRECISION VISUALS, INC., 250 ARAFAHOE AVE., BOULDER,
CO, 80302, USA, 303-449-0806
INFORMATION SOURCE: COMPLETED SUBMISSION TO NBS
ACRONYM: HARDSWARE SIMULA, TITLE: HARDWARE SIMULATOR
CLASSIFICATION: SOFTWARE MODELING AND SIMULATION
FEATURES: SUBJECT, CODE INPUT, BASIC, USER OUTPUT, LISTINGS,
DYNAMIC ANALYSIS, SIMULATION, RESOURCE UTILIZATION, TIMING,
STAGE OF DEVELOPMENT: IMPLEMENTED, DATE (YYMMDD): 800100
IMPLEMENTATION LANGUAGE: BASIC
TOOL AVAILABLE: YES, OTHER HARDWARE: DEC PDP-11
OS (OTHER SOFTWARE): RSTS
TOOL SUPPORTED: YES, TOOL SUPPORT UNIVERSITY OF WAIKATO
TOOL SUMMARY: THIS MODULE IS APPENDED TO THE MAIN PROGRAM
OF THE USER WHICH CONSISTS OF A SET OF BASIC INSTRUCTIONS AND
FUNCTION CALLS FOR SPECIAL OPERATIONS. THERE IS A
CONFIGURATION PROCEDURE TO DEFINE A SET OF DEVICE
AND MEMORY FILES SO THAT A RANGE OF DEVICES CAN BE SUPPORTED.
ANOTHER PART OF THE CONFIGURATION CONSISTS OF SETTING THE
TIMING AND PRIORITY VALUES OF THE SPECIAL OPERATIONS AND
INTERRUPTS.
DOCUMENTATION: TECHNICAL PAPER (10), USER MANUAL (5)
REFERENCES: [PAYNE80A], A. J. PAYNE, "BASIC CAN BE A USEFUL
SIMULATION LANGUAGE", US SUMMER SIMULATION CONFERENCES,
800000
DEVELOPER: A. J. PAYNE
CONTACT: A. J. PAYNE, UNIVERSITY OF WAIKATO, HAMILTON, NEW
ZELAND
INFORMATION SOURCE: COMPLETED SUBMISSION TO NBS
ACRONYM: HARP, TITLE: HIGH-POWERED ACCOUNTING RESOURCES
PROGRAM
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND
MAINTENANCE
FEATURES: SUBJECT, DATA INPUT, USER OUTPUT, GRAPHICS, DESIGN
IMPLEMENTATION: STATIC ANALYSIS, MANAGEMENT, PROJECT MANAGEMENT,
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN
TOOL SUMMARY: HARP CREATES CHARTS FOR THE GRAPHIC DISPLAY OF
PROJECT SCHEDULE AND RESOURCE INFORMATION. THE SYSTEM HAS
BEEN DESIGNED TO PRESENT TEXTUAL, NUMERIC, AND SCHEDULE
DATA IN A FORMAT WHICH IS TAILORED TO FIT THE REQUIREMENTS
OF A PARTICULAR APPLICATION OR PROJECT. THE HARf SYSTEM
PROVIDES PROJECT MANAGEMENT AND COST CAPABILITIES WHICH ARE
VALUABLE OVER THE ENTIRE LIFE CYCLE A PROJECT INCLUDING
PROPOSAL PRESENTATIONS, CONCEPT DESIGN, DATA VISUALIZATION,
PROJECT SCHEDULE, AND COST CONTROL, AND ENHANCEMENT OF
DELIVERABLE PRODUCTS. SEPARATION OF CHART DESIGN AND CHART
DATA WITHIN THE SYSTEM ENABLES NEW AND SPECIAL CHARTS TO BE
CREATED WITH NO MODIFICATION OF THE DATA BASE CHARTS ARE
ALL RECTANGULAR IN SHAPE, AND DIMENSIONS ARE EXPANDED
AUTOMATICALLY TO ACCOMMODATE ALL ENTERED DATA. THE
POSITION AND NUMBER OF VERTICAL COLUMNS IS DETERMINED BY
THE USER ONCE IN AN INITIAL DESIGN OF A CHART. DATA IS
SEQUENTIALLY ENTERED INTO THIS FORMAT, AND CHARTS MAY BE
PRODUCED WHENEVER UPDATES OCCUR WITHOUT MANIPULATING THE
CHART FORMAT.
DOCUMENTATION: PROGRAM MAINTENANCE MANUAL, INPUT DESCRIPTION,
SUMMARY DESCRIPTION
REFERENCES: [AOS397], APPLIED SYSTEMS DESIGN SECTION, TRW
DEFENSE AND SPACE SYSTEM GROUP, "SOFTWARE TOOLS
CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE
TOOLS SERIES (FEB, 1979)
DEVELOPER: TRW, SOFTWARE TECHNOLOGY DEPT., CONTACT: P. W. BOGLE, TRW, SOFTWARE TECHNOLOGY DEPT., ONE
SPACE PARK, REDONDO BEACH, CA, 90278, USA, 213-355-3480
INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG
ACRONYM: HAWKEYE (TM), TITLE: HAWKEYE UNIVERSAL COBOL
STANDARIZATION
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, COBOL, TRANSFORMATION,
RESTRUCTURING, FORMATTING, MACHINE OUTPUT, SOURCE CODE
OUTPUT, COBOL USER OUTPUT, DIAGNOSTICS, LISTINGS, STATIC
ANALYSIS, AUDITING
STAGE OF DEVELOPMENT: IMPLEMENTED, DATE (YYMMDD): 730000
IMPLEMENTATION LANGUAGE: COBOL ANSI
TOOL PORTABLE: YES, TOOL SIZE: REQUIRES 50 CHAR OF MEMORY
TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): LEASE AGREEMENT,
COPYRIGHTED, "HAWKEYE" IS A REGISTERED TRADEMARK
TOOL SUPPORTED: YES, TOOL SUPPORT: BLACKHAWK DATA CORP
TOOL SUMMARY: HAWKEYE IS A COBOL STANDARDIZER, DESIGNED
PRIMARILY TO OPERATE ON EXISTING COBOL PROGRAMS, PARTICULARLY THOSE WHICH ARE OLD, OVER-AGE, POORLY
DOCUMENTED, AND/OR POORLY MAINTAINED. PERFORMS THE

HAWKEYE (TM)

FOllOWING FUNCTIONs: 1) STANDARDIZES ALIGNMENT OR KEY WORDS IN DATA AND PROCEDURE DIVISION. 2) REPLACES A USER-DEFINED SET OF WORDS WITH ALTERNATIVE WORDS OR PHRASES. CAN DELETE NON-ESSENTIAL WORDS IF DESIRED. 3) NUMBERS, DE-NUMBERS, OR RE- NUMBERS THE PARAGRAPH NAMES, AND ALL REFERENCES THERETO. 4) LIMITS PROCEDURE DIVISION ENTRIES TO ONE VERB PER LINE. 5) STACKS MULTIPLE OPERANDS ONE BELOW THE OTHER. 6) CORRECTS MffbR FORMAT AND PUNCTUATION ERRORS AUTOMATICALLY. 7) INSERTS A BLANK LINE BETWEEN Paragraphs AND 01 - LEVELS, AND SKIPS A PAGE BETWEEN DIVISIONS AND SECTIONS. 8) SINCE HAWKEYE'S DICTIONARIES ARE STORED IN A USER CONTROLLED EXTERNAL FILE, CAN EASILY BE SET UP TO HANDLE NEW COBOL VERBS AND/OR MANUFACTURER'S EXTENSION, INCLUDING ASSEMBLER ESCAPE ROUTINES.

DOCUMENTATION: USER'S MANUAL (41), DESCRIPTIVE BROCHURE (42).

DEVELOPER: BLACKHAWK DATA CORP.

CONTACT: JOHN BRINK, BLACKHAWK DATA CORP., 200 NO. MICHIGAN AVE., CHICAGO, IL, 60601, USA, 312-235-8673

INFORMATION SOURCE: COMPLETED SUBMISSION TO NBS

ACRONYM: IFTRAN (TM), TITLE: A PREPROCESSOR FOR FORTRAN

CLASSIFICATION: PROGRAM CONSTRUCTION AND GENERATION

FEATURES: SUBJECT, CODE INPUT, FORTRAN, IFTRAN, VML INPUT, IFTRAN, TRANSFORMATION, TRANSLATION, STRUCTURE PROGRAMMING, EATING, INSTRUMENTATION, FILES TO BE ADDRESSSED BY IFTRAN, MACHINE OUTPUT, SOURCE CODE OUTPUT, FORTRAN, USER OUTPUT, DIAGNOSTICS, GRAPHICS, LISTINGS, STATIC ANALYSIS, STRUCTURE CHECKING, DYNAMIC ANALYSIS, COVERAGE ANALYSIS, ASSERTION CHECKING, TUNING, TRACING, COMMAND INPUT, COMMANDS, PARAMETERS.

STAGE OF DEVELOPMENT: IMPLEMENTED

IMPLEMENTATION LANGUAGE: FORTRAN

TOOL PORTABLE: YES, TOOL SIZE: 25K WORKS

TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO

RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): LICENSE REQUIRED

TOOL SUPPORTED: YES, TOOL SUPPORT: CORTEX CORPORATION

TOOL SUMMARY: IFTRAN IS AN EXTENSION OF FORTRAN THAT SIMPLIFIES STRUCTURED PROGRAMMING AND TOP-DOWN DESIGN, AIDS SYSTEMATIC TESTING, SUPPORTS A PRACTICAL APPLICATION OF FORMAL VERIFICATION TECHNIQUES, AND AIDS IN SOFTWARE FUTURE TOLERANCE. IFTRAN HAS A CONVENIENT SYNTAX FOR WRITING STRUCTURED PROGRAMMING CONTROL CONSTRUCTS. THE IFTRAN PREPROCESSOR TRANSLATES THE IFTRAN STATEMENTS INTO STANDARD FORTRAN WHILE PASSING ALL OTHER STATEMENTS UNCHANGED TO AN OUTPUT FILE WHICH MAY THEN BE COMPILED BY A FORTRAN COMPILER. A LISTING WHICH AUTOMATICALLY INDENTS THE SOURCE PROGRAM IS ALSO GENERATED. IFTRAN CAN BE USED AS A PROGRAM DESIGN LANGUAGE WHEN ENGLISH IS USED IN THE CONSTRUCTS INSTEAD OF FORTRAN. THE INDEPTED LISTING IS VALUABLE FOR DESIGN Reviews.

DOCUMENTATION: USERS MANUAL

IFTRAN (TM)

DEVELOPER: GENERAL RESEARCH CORP.

CONTACT: WILLIAM R. DE MAN, GENERAL RESEARCH CORP., 5533 HOLLISTER AVE, PO BOX 6770, SANTA BARBARA, CA, 93111, USA, 805-684-7720

INFORMATION SOURCE: TOOL FAIR

ACRONYM: INFORM, TITLE: INF oRM CLASSIFICATION: PROGRAM CONSTRUCTION AND GENERATION

FEATURES: SUBJECT: DATA INPUT, DESCRIPTION LANGUAGE, TRANSFORMATION, SYNTHESIS, MACHINE OUTPUT, DATA OUTPUT, USER OUTPUT, DIAGNOSTICS, GRAPHICS, TABLES, STATIC ANALYSIS, MANAGEMENT, DATA BASE MANAGEMENT.

STAGE OF DEVELOPMENT: IMPLEMENTED

IMPLEMENTATION LANGUAGE: ASSEMBLY

TOOL PORTABLE: NO COMPUTER (OTHER HARDWARE): DEC PDP-11, DEC VAX-11 OS (OTHER SOFTWARE): VMS, RXS-11, IAS

TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO

RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): LICENSE REQUIRED

TOOL SUPPORTED: YES, TOOL SUPPORT: CORTEX CORPORATION

TOOL SUMMARY: INF oRM IMPROVES PROGRAMMING PRODUCTIVITY IN THE DEVELOPMENT OF ONLINE BUSINESS APPLICATIONS (APPLICATIONS WHICH FOCUS ON THE INTERACTION OF MULTIPLE USERS WITH A LARGE OR COMPLEX DATA BASE). FEATURES COMMON TO THE INF oRM TOOL WOULD BE THE SCREEN HANDLING, FLEXIBLE DATA BASE STRUCTURE, DATA BASE ACCESS/INQUIRY, AND AD HOC AND PRODUCTION REPORTING REQUIREMENTS, USING TRADITIONAL APPLICATION TOOLS, THE PROGRAMMER WOULD HAVE TO USE A SCREEN GENERATOR, A JOB CONTROL LANGUAGE, A REPORT WRITER, A DATABASE SYSTEM, STANDARD OPERATING SYSTEM UTILITIES, AS WELL AS A PROCEDURAL LANGUAGE TO DEVELOP THE APPLICATION. THIS FORCES THE PROGRAMMER TO DEVELOP EXPERTISE, NOT ONLY IN A VARIETY OF UNRELATED TOOLS, BUT ALSO IN HOW TO MAKE THEM WORK TOGETHER. THE INF oRM INTEGRATES THE DEVELOPMENT TOOLS, FREEING THE PROGRAMMER TO CONCENTRATE ON THE UNIQUE PROCESSING REQUIREMENTS.

DOCUMENTATION: USERS MANUAL, IMPLEMENTATION GUIDE, HELP DEVELOPER: CORTEX CORPORATION CONTACT: CORTEX CORPORATION, 55 WILLIAM STREET, WELLESLEY, MA, USA, 617-237-2506

INFORMATION SOURCE: PRODUCT DESCRIPTION

ACRONYM: INF oRM/REFORM, TITLE: INF oRM AND REFORM PROGRAMS

CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE

FEATURES: SUBJECT: CODE INPUT, FORTRAN, USER OUTPUT, USER-ORIENTED TEXT, DOCUMENTATION, STATIC ANALYSIS, INTERFACE ANALYSIS, SCANNING.

STAGE OF DEVELOPMENT: IMPLEMENTED

IMPLEMENTATION LANGUAGE: FORTRAN

COMPUTER (OTHER HARDWARE): CDC 6X00/7X00

DOCUMENTATION: USERS MANUAL
INFORM/REFORM

TOOL SUMMARY: INFORM AND REFORM IDENTIFY AND DISPLAY FORTRAN SUBROUTINE/VARIABLE INTERFACES FOR EACH ROUTINE. THE OVERALL DOCUMENTATION IS NOT ONLY USEFUL IN THE DEVELOPMENT AND MAINTENANCE OF SOFTWARE SYSTEMS, BUT ADDITIONALLY IS REQUIRED BY MANY DOCUMENTATION STANDARDS. THESE PROGRAMS ARE NOT VERY LANGUAGE OR MACHINE DEPENDENT BUT ARE HIGHLY DEPENDENT ON CAPABILITIES FOUND IN THE EDITOR AND COMPILER PROGRAMS OF THE TRW/758.

REFERENCES: [TRW758], APPLIED SYSTEMS DESIGN SECTION, TRW DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS: CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE TOOLS SERIES, 790100

DEVELOPER: TRW NAVIGATION SYSTEMS ENG DEP
 COMPUTER: MARRIN H. KLOTTZ, TRW NAVIGATION SYSTEMS ENG DEP, ONE SPACE PARK, REDONDO BEACH, CA, 90278, USA, 213-535-1581
 INFORMATION SOURCE: TRW SOFTWARE TOOL CATALOG

ACRONYM: INSERT, TITLE: INSERT
 CLASSIFICATION: SOFTWARE, MANAGEMENT, CONTROL, AND MAINTENANCE

FEATURES: SUBJECT, CODE INPUT, FORTRAN, FORTRAN 66, MACHINE OUTPUT, SOURCE CODE OUTPUT, FORTRAN, USER OUTPUT, DIAGNOSTICS, TABLES, LISTINGS, DYNAMIC ANALYSIS, TUNING, TRACING, PATH FLOW TRACING

STAGE OF DEVELOPMENT: IMPLEMENTED
 IMPLEMENTATION LANGUAGE: RATFOR

COMPUTER (OTHER HARDWARE): CDC 6000/7000
 TOOL AVAILABLE: YES

TOOL SUMMARY: "INSERT" WAS ORIGINALLY DESIGNED TO INSERT TRAP CODE INTO A LARGE FORTRAN PROGRAM FOR THE ANALYSIS OF SUBMARINE STRUCTURES. "INSERT" READS THE SOURCE CODE AND EXTRACTS THE NAMES OF EACH FORTRAN MODULE FROM THE MODULE HEADER CARD. "INSERT" CONTINUES READING FORTRAN SOURCE CODE UNTIL IT FINDS THE FIRST EXECUTABLE STATEMENT. A NEW "FIRST EXECUTABLE STATEMENT" IS THEN COPIED INTO THE SOURCE OUTPUT FILE, TO BE WRITTEN NAME). THE AUGMENTED STRUCTURAL ANALYSIS PROGRAM WAS THEN COMPILED AND EXECUTED, THE TAPE PRODUCED BY EXECUTION WAS EXAMINED FOR MODULE CALLS, AND THE UNUSED MODULES OF THE PROGRAM WERE DELETED. A SIMPLE MODIFICATION TO "INSERT" WOULD ALLOW IT TO ALSO INSERT TEXT PRECEDING A "RETURN" STATEMENT. THE EXECUTABLE OUTPUT TEXT WOULD BE BRACKETED BY USER INSERTED TRAPS, PARTICULARLY CALLS TO THE "MONITOR" PACKAGE.

DEVELOPER: NSRD

CONTACT: PETER N. ROTH, NSRD, STRUCTURES DEPARTMENT, BETHESDA, MD 20084, USA, 202-227-1851
 INFORMATION SOURCE: COMPLETED SUBMISSION TO NASA

ACRONYM: INSTRU, TITLE: A SOFTWARE SYSTEM THAT INSTRUMENTS DATA FLOW ANALYSIS
 CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING

FEATURES: SUBJECT, CODE INPUT, FORTRAN, TRANSFORMATION, INSTRUMENTATION, MACHINE OUTPUT, SOURCE CODE OUTPUT, FORTRAN, USER OUTPUT, TABLES, LISTINGS, DYNAMIC ANALYSIS, TRACING, PATH FLOW TRACING, DATA FLOW TRACING, LOGIC FLOW TRACING

STAGE OF DEVELOPMENT: IMPLEMENTED
 IMPLEMENTATION LANGUAGE: FORTRAN IV

TOOL PORTABLE: NO
 COMPUTER (OTHER HARDWARE): HONEYWELL 6XXX
 OS (OTHER SOFTWARE): IBM 4352
 TOOL AVAILABLE: YES, PUBLIC DOMAIN: YES
 TOOL SUPPORTED: NO

TOOL SUMMARY: INSTRU IS AN EXPERIMENTAL SOFTWARE TOOL THAT CAN BE USED TO INCREASE THE ERROR-DETECTION CAPABILITY OF A PROGRAM TEST BY INSTRUMENTING THE PROGRAM FOR DATA-FLOW ANOMALY DETECTION AND SYMBOLIC-TRACE GENERATION.

DOCUMENTATION: USERS GUIDE
 REFERENCES: [HUANG78], J. C. HUANG, "PROGRAM INSTRUMENTATION AND SOFTWARE TESTING", COMPUTER, VOL. 11, NO. 4, 760400
 [HUANG79], J. C. HUANG, "PROGRAM INSTRUMENTATION: A TOOL FOR SOFTWARE TESTING", INFOTECH STATE OF THE ART REPORT, SOFTWARE TESTING, VOL. 2, 790000
 [HUANG80], J. C. HUANG, "DETECTION OF DATA FLOW ANOMALY THROUGH PROGRAM INSTRUMENTATION", IEEE TRANSACTIONS ON SOFTWARE ENGINEERING, VOL. 6, NO. 3, 790500
 [HUANG81], J. C. HUANG, "PROGRAM MECHANISM PROGRAM FOR SYMBOLIC-TRACE GENERATION", COMPUTER, VOL. 13, NO. 12, 601200

DEVELOPER: UNIV OF HOUSTON
 CONTACT: J. C. HUANG, UNIVERSITY OF HOUSTON, CENTRAL CAMPUS, DEPT OF COMP SCI, HOUSTON, TEXAS, 77004, USA, 713-749-2856

INFORMATION SOURCE: TOOL FAIR

ACRONYM: INTERFACE DOCUMENT, TITLE: INTERFACE DOCUMENTATION
 CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING

FEATURES: SUBJECT, CODE INPUT, OBJECT CODE INPUT, USER OUTPUT, TABLES, STATIC ANALYSIS, CROSS REFERENCE

STAGE OF DEVELOPMENT: IMPLEMENTED
 IMPLEMENTATION LANGUAGE: FORTRAN

TOOL PORTABLE: YES
 TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO

RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): MARKETED COMMERCIAL

TOOL SUPPORTED: YES, TOOL SUPPORT: SOFTSOURC CORPORATION

TOOL SUMMARY: THIS SOFTWARE PRODUCT ACCEPTS AS INPUT A COLLECTION OF OBJECT MODULES AND AUTOMATICALLY GENERATES CLEAR INFORMATION INDICATING ALL INTERFACES BETWEEN THE OBJECT MODULES. IT PRODUCES, FOR EACH MODULE, AN ANNOTATED LIST OF THE MODULES IT REFERENCES AS WELL AS A LIST OF ALL THE MODULES THAT REFERENCED IT. EXTERNAL DATA ITEMS ARE ALSO DOCUMENTED. THIS PRODUCT IS VERY EASY TO USE, IT ACCEPTS AS INPUT THE SAME OBJECT MODULES THAT AREUSUALLY PRESENTED TO YOUR LINKER (BINDER), THUS IN ORDER TO
GENERATE THE INTERFACE DOCUMENTATION YOU SIMPLE SUBMIT YOUR INPUTS TO THE INTERFACE DOCUMENTER INSTEAD OF THE LINKED TOOL. THE INTERFACE DOCUMENTER IS INDIVIDUAL TO THE LANGUAGE IN WHICH THE PROGRAMS ARE DOCUMENTED. IT WILL GENERATE INTERFACE DOCUMENTATION FOR FORTRAN, COBOL, ASSEMBLER, ETC. THIS TOOL IS A MEMBER OF SOFTOOL, AN INTEGRATED SET OF TOOLS MARKETED BY SOFTOOL CORPORATION.

**DOCUMENTATION:** TECHNICAL REPORTS, USER'S MANUAL

**DEVELOPER:** SOFTOOL CORPORATION

**CONTACT:** CAROL BADDOF, SOFTOOL CORPORATION, 340 SOUTH KELLOGG AVE., GOLETA, CA, 93117, USA, 805-964-0560

**INFORMATION SOURCE:** COMPLETED SUBMISSION TO NBS

**ACRONYM:** IDOR, TITLE:** INPUT/OUTPUT REQUIREMENTS LANGUAGE CLASSIFICATION:** REQUIREMENTS/DESIGN SPECIFICATION AND ANALYSIS

**FEATURES:** SUBJECT, TEXT INPUT, VHLL INPUT, REQUIREMENTS LANGUAGE, TRANSFORMATION, EDITING, USER OUTPUT, USER-ORIENTED TEXT, REPORTS, STATIC ANALYSIS, SCANNING

**STAGE OF DEVELOPMENT:** IMPLEMENTED, DATE (YMMDD): 760000

**IMPLEMENTATION LANGUAGE:** MACRO-1

**COMPUTER (OTHER HARDWARE):** DEC/VT4

**OS (OTHER SOFTWARE):** RT-11

**TOOL AVAILABLE:** YES, PUBLIC DOMAIN: NO

**RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.):** FOR SALE

**TOOL SUMMARY:** STANDALONE, TURN KEY SYSTEM FOR DOCUMENTING SYSTEM DEFINITION, DESIGN AND TESTING REQUIREMENTS ASSOCIATED WITH THE DEVELOPMENT OF ANY COMPUTER BASED SYSTEM IN GENERAL, THE IDOR SYSTEM PROVIDES AN INTERACTIVE GRAPHICS STORAGE AND RETRIEVAL CAPABILITY WITH THE PRECISE SYSTEM REQUIREMENTS CAN BE GENERATED AND MODIFIED WITH AN EDITOR, RETRIEVED BY LIGHT PEN, AND HARD COPIED ON 8 1/2 X 11 INCH PAPER.

**DOCUMENTATION:** TECHNICAL PAPER (USER MANUAL), USER MANUAL, OPERATIONS MANUAL

**REFERENCES:** [EVER78], C. R. EVERHART, "USER EXPERIENCE WITH A FORMALLY DEFINED REQUIREMENTS LANGUAGE IDOR", 2ND US ARMY SOFT SYM, 78025

[EVER78], C. R. EVERHART, "SYSTEM REQUIREMENTS LANGUAGE - FOUNDATION FOR SOFTWARE ENGINEERING", 3RD ANNUAL NASA SUM SOFT ENG WKSHP, 780918

**DEVELOPER:** TELEDYNE BROWN ENGINEERING

**CONTACT:** C. R. EVERHART, TELEDYNE BROWN ENGINEERING, CUMMINGS RESEARCH PARK, HUNTSVILLE, AL, 35807, USA, 205-532-1610

**INFORMATION SOURCE:** AIAA SURVEY OF SOFT DEV TOOLS

**ACRONYM:** IPDS, TITLE:** INTERACTIVE PROGRAM DEVELOPMENT SYSTEM CLASSIFICATION:** REQUIREMENTS/DESIGN SPECIFICATION AND ANALYSIS

**FEATURES:** SUBJECT, VHLL INPUT, DESIGN SPECIFICATION, TRANSFORMATION, SYNTHESIS, MACHINE OUTPUT, SOURCE CODE OUTPUT, USER OUTPUT, DIAGNOSTICS, LISTINGS, DYNAMIC ANALYSIS

**ASSERTION CHECKING, FORMAL VERIFICATION, STAGE OF DEVELOPMENT:** CONCEPT

**TOOL AVAILABLE:** NO

**TOOL SUMMARY:** IPDS IS A TOOL BEING PLANNED BY SEVERAL MEMBERS OF THE COMPUTER SCIENCE DEPARTMENT AT CORNELL UNIVERSITY. THE TOOL WILL BE BASED ON THE PROL SYSTEM DESCRIBED IN A RECENT TECHNICAL REPORT, THE REPORT EXHIBITS THE VIEW THAT CORRECT PROGRAMS ARE MOST NATURALLY PRODUCED AS A RESULT OF CORRECT DEVELOPMENT AND THAT REASONING ABOUT CODE IS NOT THE PROPER PARADIGM FOR SUCH DEVELOPMENT, INSTEAD PROGRAMMERS SHOULD REASON ABOUT SPECIFICATIONS IN A LOGIC FORMULATED FOR CONSTRUCTIVELY PROVING THAT SPECIFICATIONS ARE ACCEPTABLE IMPLEMENTATIONS; FROM THESE PROOFS CODE MAY BE EXTRACTED. THIS APPROACH PROVIDES A UNIFORM VIEW OF BOOLEAN FORMULAS AND PROGRAM SPECIFICATIONS AS PROVABLE PROPOSITIONS, THERE IS NO ASSERTION/CODE DISTINCTIONS BECAUSE PROOFS INCLUDE NEITHER ASSERTIONS NOT CODE. THE PROGRAMMER TRULY CONCENTRATES ON PROOF DEVELOPMENT, NOT ON CODING, EXECUTABLE TEXT IS SIMPLY A VALUABLE PRODUCT OF CORRECT REASONING.

**REFERENCES:** [BATE79], JOSEPH LOUIS BATES, "A LOGIC FOR CORRECT PROGRAM DEVELOPMENT", CORNELL TECH REPORT, TR 85-97

**DEVELOPER:** CORNELL UNIVERSITY

**CONTACT:** JOSEPH LOUIS BATES, CORNELL UNIVERSITY, DEPT OF COMPUTER SCIENCE, ITHACA, NY, 14853, USA

**INFORMATION SOURCE:** CORNELL TECHNICAL REPORT

**ACRONYM:** ISOS, TITLE:** INTEGRATED SOFTWARE DEVELOPMENT SYSTEMS

**CLASSIFICATION:** REQUIREMENTS/DESIGN SPECIFICATION AND ANALYSIS

**FEATURES:** SUBJECT, CODE INPUT, FORTRAN, JOVIAL, VHLL INPUT, DESIGN SPECIFICATION, USER OUTPUT, LISTINGS, STATIC ANALYSIS, COMPLEXITY MEASUREMENT, STRUCTURE CHECKING

**STAGE OF DEVELOPMENT:** DESIGN

**LANGUAGE:** IFTRAN

**TOOL SIZE:** CORE: 128K

**COMPUTER (OTHER HARDWARE):** HONEYWELL 6XX (DISK: 6M)

**OS (OTHER SOFTWARE):** GCOS

**TOOL AVAILABLE:** NO, PUBLIC DOMAIN: NO

**TOOL SUMMARY:** ALLOWS INTERACTIVE CHARTING OF SOFTWARE DESIGN AND ANALYSIS OF CHARTS FOR CONSISTENT STRUCTURING, ALSO ALLOWS SOURCE CODE INPUT (FORTRAN, JOVIAL, PD) AND OTHER BEING DEVELOPED TO ANALYZE THE CODE FOR COMPLEXITY AND STRUCTURE

**DOCUMENTATION:** USER'S MANUAL

**DEVELOPER:** GENERAL ELECTRIC

**CONTACT:** ORIN BARKER, GENERAL ELECTRIC, CSP3=35, SYRACUSE, 13202, USA, 315-456-7111

**INFORMATION SOURCE:** AIAA SURVEY OF SOFT DEV TOOLS
ACRONYM: JOVIAL/TCA, TITLE: JOVIAL J73/I TEST COVERAGE ANALYZER
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: Subject, code input, JOVIAL, JOVIAL J73/I, transformation, instrumentation, machine output, source code output, JOVIAL user output, diagnostics, tables, listings, dynamic analysis, coverage analysis
STAGE OF DEVELOPMENT: IMPLEMENTED, DATE (YMMDD): 760000
IMPLEMENTATION LANGUAGE: JOVIAL J73/I
COMPUTER (OTHER HARDWARE): DEC/DEC SYSTEM-10/20
PUBLIC DOMAIN: YES
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): INFORMALLY DELIVERED TO WRIGHT-PATTERSON AIR FORCE BASE
TOOL SUPPORTED: NO
TOOL SUMMARY: THE TOOL PROVIDES THE CAPABILITY TO MEASURE THE "TESTEDNESS" OF JOVIAL J73/I PROGRAMS. THE TOOL INSTRUMENTS THE J73/I PROGRAM SO THAT WHEN THE PROGRAM EXECUTES, EACH LOGICAL SEGMENT OF THE PROGRAM WILL COUNT THE NUMBER OF TIMES IT WAS EXECUTED. A SUMMARY OF THESE COUNTS WILL BE PRINTED AT THE CONCLUSION OF PROGRAM EXECUTION. THE USER WILL THEN BE ABLE TO IDENTIFY SEGMENTS NOT EXECUTED BY A SUITE OF TEST CASES.
DOCUMENTATION: MAINTENANCE MANUAL (109)
DEVELOPER: BOEING AEROSPACE CO.
CONTACT: ROBERT L. GLASS, BOEING AEROSPACE CO., PO BOX 3999, SEATTLE, WA, 98124, USA, 206-773-0664
INFORMATION SOURCE: COMPLETED SUBMISSION TO NBS

ACRONYM: JOVIAL/J35C TITLE: JOVIAL/J3 STATISTICS COLLECTOR CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: Subject, code input, JOVIAL, JOVIAL J3, user output, diagnostics, tables, static analysis, statistical analysis, profile generation
STAGE OF DEVELOPMENT: IMPLEMENTED, DATE (YMMDD): 770000
IMPLEMENTATION LANGUAGE: JOVIAL J3
TOOL SIZE: CORE: 80K
COMPUTER (OTHER HARDWARE): HONEYWELL 6XXX
OS (OTHER SOFTWARE): GCGS
TOOL AVAILABLE: YES
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): AVAILABILITY ACCORDING TO AIR FORCE MANUAL (AFM) 300-6, PARAGRAPH 11-7A
TOOL SUPPORTED: YES, TOOL SUPPORT: RADC/ISIS
TOOL SUMMARY: THE JOVIAL/J3 STATISTICS COLLECTOR IS A SOFTWARE TOOL DEVELOPED IN-HOUSE BY RADC TO MEASURE USAGE OF VARIOUS CONSTRUCTS AND FEATURES OF THE JOVIAL/J3 HIGHER ORDER COMPUTER PROGRAMMING LANGUAGE. COUNTERS AND PERCENTAGES CAN BE OBTAINED FOR A GIVEN SOURCE PROGRAM REGARDING TYPES OF DATA USED, SIZE AND NATURE OF ARRAYS AND TABLES; TYPES OF STATEMENTS APPEARING; USE AND NESTING OF COMPLEX STATEMENTS; COMMENTS; DEFINE DIRECTIVES; AND OTHER ASPECTS OF SOURCE CODE USAGE BY PROGRAMS. IT IS HOPE THAT THE INFORMATION OBTAINED BY THE STATISTICS COLLECTOR WILL PROVIDE GUIDANCE TOWARDS THE INSTRUCTION OF PROGRAMMERS IN MORE EFFICIENT AND LESS ERROR-PRONE PROGRAMMING METHODS, THE WRITING OF BENEFICIAL CHANGES TO THE LANGUAGE, MACHINE DEPENDENCIES; THE JOVIAL/J3 STATISTICS COLLECTOR IS CURRENTLY EQUIPPED TO RUN ON ANY拱ON READING 600/6600 GCGS COMPUTER SYSTEM, BUT CAN BE MODIFIED TO RUN ON OTHER SYSTEMS.
DOCUMENTATION: TECHNICAL PAPER
DEVELOPER: RADC/ISIS
CONTACT: RICHARD T., SLAVINSKI, ROME AIR DEVELOPMENT CENTER (ISIE), GRIFFISS AIR FORCE BASE, NY, 13441, USA, 315-330-2768
INFORMATION SOURCE: COMPLETED SUBMISSION TO NBS

ACRONYM: JOVIAL/V/S, TITLE: JOVIAL COMPILER VALIDATION SYSTEM CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: Subject, code input, JOVIAL, JCVS, JOVIAL J3, user output, diagnostics, static analysis, auditing
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: JOVIAL J3
TOOL AVAILABLE: YES
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): AVAILABILITY ACCORDING TO AIR FORCE MANUAL (AFM) 300-6, PARAGRAPH 11-7A
TOOL SUPPORTED: YES, TOOL SUPPORT: RADC/ISIS
TOOL SUMMARY: THESE TOOLS ASSIST IN THE TESTING AND DEBUGGING, ACCEPTANCE TESTING, AND MAINTENANCE OF JOVIAL (J3), JOVIAL (J73/I), AND JOVIAL (J3) COMPILERS. THE JOVIAL (J3) VALIDATION SYSTEM TESTS JOVIAL (J3) COMPILERS FOR CONFORMANCE TO M3L=STD-1568 USAF. THE JOVIAL (J73/I) VALIDATION SYSTEM TESTS JOVIAL (J73/I) COMPILERS FOR CONFORMANCE TO M3L=STD-1569 (USAF). THE JOVIAL (J3) VALIDATION SYSTEM TESTS JOVIAL (J3) COMPILERS FOR CONFORMANCE TO M3L=STD-1589A (USAF). WHEREVER POSSIBLE, THE TESTS ARE SELF CHECKING AND THEREFORE REQUIRE MINIMAL EFFORT TO ANALYZE THE RESULTS OF MOST OF THE TEST SEQUENCES. MACHINE DEPENDENCIES: NOT MACHINE DEPENDENCIES ARE SPECIFIED BY DEFINE DECLARATIONS WHICH PRECEDE THE DATA DECLARATIONS AND TEST MODULES. THE USER LOCALIZED THESE DEPENDENCIES TO THIS PRELIMINARY CODE. A FEW ARE NOTED BY COMMENTS IN THE MODULES THEMSELVES.
DOCUMENTATION: REFERENCE AIAA ABSTRACT
DEVELOPER: RADC/ISIS
CONTACT: RADC/ISIS, GRIFFISS AFB, NY, 13441, USA
INFORMATION SOURCE: COMPLETED SUBMISSION TO NBS
JOYCE

FEATURES: SUBJECT, CODE INPUT, FORTRAN, TRANSFORMATION, EDITING, USER OUTPUT, GRAPHICS, FLOW CHARTS, TABLES, STATIC ANALYSIS, CROSS REFERENCE, STAGE OF DEVELOPMENT: IMPLEMENTED, IMPLEMENTATION LANGUAGE: FORTRAN, COMPUTER (OTHER HARDWARE): CDC 6600/7000

TOOL AVAILABLE: YES

TOOL SUPPORTED: YES, TOOL SUPPORT: MCDONNELL DOUGLAS AEROSPACE COMPANY

TOOL SUMMARY: JOYCE IS AN AUTOMATIC STATIC ANALYSIS TOOL FOR FORTRAN PROGRAMS. IT ACCEPTS AS PRIMARY INPUT FORTRAN SOURCE DECKS IN THE FORM OF CARD DECKS OR CDC COMPIL FILEs. THE SOURCE DECKS ARE EDITED AND THE EDITED INFORMATION IS COMBINED TO PRODUCE SEVERAL COMBINATIONS OF DESCRIPTIVE REPORTS, JOYCE COMPIL TABLES OF SYMBOLS AND CROSS REFERENCES OF SYMBOL USAGE WITHIN EACH ROUTINE OF A PROGRAM. THESE SYMBOLS INCLUDE FORTRAN VARIABLE NAMES, THE NAMES OF ANY REFERENCE FUNCTION OR MODULE, ANY ENTRY POINTS, AND ALL I/O FILE REFERENCES. FLOWLISTS ARE PROVIDED IN THE FORM OF MICROFILM FORTRAN LISTINGS WITH ALL TRANSFERS INDICATED BY ARROWS TO THE RIGHT OF THE STATEMENT TEXT AND ALL DO LOOPS INDICATED BY BRACKETS TO THE LEFT.

REFERENCES: [DON68], JOHN D., DONAHOO AND DOROTHY SWEARINGER, "A REVIEW OF SOFTWARE MAINTENANCE TECHNOLOGY", RADC-TR-60-13, INTERIM REPORT, 800200

[STU76], G. STUCKI, "METHODOLOGY FOR PRODUCING AVAILABLE SOFTWARE", NASA CR 147769, TWO VOLUMES, 760300

DEVELOPER: MCDONNELL DOUGLAS AEROSPACE COMPANY

INFORMATION SOURCE: RADC-TR-80-13, INTERIM REPORT

ACRONYM: JSDD, TITLE: JOVIAL STRUCTURED DESIGN DIAGRAMER, CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE

FEATURES: SUBJECT, CODE INPUT, JOVIAL J3, USER OUTPUT, GRAPHICS, FLOW CHARTS, STATIC ANALYSIS, SCANNING, STAGE OF DEVELOPMENT: IMPLEMENTED, IMPLEMENTATION LANGUAGE: JOVIAL J3, COMPUTER (OTHER HARDWARE): HONEYWELL 6XX

OS (OTHER SOFTWARE): GCOS

PUBLIC DOMAIN: YES

TOOL SUMMARY: THE JSDD PRODUCES FLOW INVOCATION DIAGRAMS OF SOFTWARE WRITTEN IN EXTENDED JOVIAL J3, EXTENDED JOVIAL J3 IS STANDARD JOVIAL J3 PLUS CONTEMPORARY FLOW CONTROL CONSTRUCTS (A MULTICS PL/1 PREPROCESSOR FOR TRANSLATING THESE EXTENSIONS INTO JOVIAL J3 IS ALSO AVAILABLE).

DOCUMENTATION: USER'S MANUAL, MAINTENANCE MANUAL, DEVELOPMENT SPECIFICATION, TEST PLAN

REFERENCES: [LRA78], ROME AIR DEVELOPMENT CENTER, "JOVIAL STRUCTURED DESIGN DIAGRAMER - FINAL TECHNICAL REPORT", RADC-TR-78-9 (FOUR VOLUMES), 780200

DEVELOPER: DRAPER LABORATORY

CONTACT: MARK WHITWORTH, DRAPER LABORATORY, 555 TECHNOLOGY SQUARE, CAMBRIDGE, MA, 02139, USA, 617-258-1179

JSDD

INFORMATION SOURCE: AIAA SURVEY OF SOFT DEV TOOLS

ACRONYM: LANG INSTRUCTOR, TITLE: LANGUAGE INSTRUCTOR

CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE

FEATURES: SUBJECT, CODE INPUT, FORTRAN, COBOL, LISP, APL, ALGOL, SNOBOL, USER OUTPUT, USER-ORIENTED TEXT, STATIC ANALYSIS, MANAGEMENT

STAGE OF DEVELOPMENT: IMPLEMENTED, DATE (YMDH): 771200

IMPLEMENTATION LANGUAGE: FORTRAN IV

TOOL AVAILABLE: YES, PUBLIC DOMAIN: YES

TOOL SUPPORTED: YES, TOOL SUPPORT: UNIVERSITY OF WAIKATO

TOOL SUMMARY: THIS INTERACTIVE SYSTEM IS A SET OF PROGRAMS FOR SETTING UP A COMPUTER AIDED INSTRUCTION SYSTEM. THE INSTRUCTIONS FOR USE OF THE SYSTEM ARE CONTAINED IN THE DIALOGUE. THERE IS PROTECTION FOR THE USER WHO CANNOT NORMALLY MODIFY THE INSTRUCTION DATABASE. IF THE USER IS ORGANIZING THE SYSTEM HE CAN UPDATE THE SYSTEM WITH NEW SUBJECTS OR DETAILS ON A SUBJECT. PRESENT SUBJECTS IN THE SYSTEM ARE FORTRAN COBOL, ALGOL LISP, SNOBOL AND APL.

DOCUMENTATION: TECHNICAL, USER

DEVELOPER: B. FERGUSON

CONTACT: A.J. PAYNE, UNIVERSITY OF WAIKATO, HAMILTON, NEW ZEALAND

INFORMATION SOURCE: COMPLETED SUBMISSION TO NBS

ACRONYM: LAYOUT, TITLE: GRAPHICAL REPRESENTATION OF DATA STRUCTURES

CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE

FEATURES: SUBJECT, DATA INPUT, USER OUTPUT, GRAPHICS, STATIC ANALYSIS, SCANNING

STAGE OF DEVELOPMENT: IMPLEMENTED

COMPUTER (OTHER HARDWARE): IBM 360/370

RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): AVAILABILITY OUTSIDE BOEING MAY BE RESTRICTED

TOOL AVAILABLE: YES, TOOL SUPPORT: BOEING COMPUTER SERVICES COMPANY

TOOL SUMMARY: LAYOUT PRODUCES A GRAPHICAL REPRESENTATION OF DATA STRUCTURES IN EXTERNAL FILES OR INTERNAL STORAGE. IT CAPTURES THE RELATIONSHIPS BETWEEN THE FIELD OF THE DATA STRUCTURE AS WELL AS DETAILED INFORMATION ON THE FIELDS INCLUDING NAME, SIZE, TYPE, SOURCE, DERIVATION, AND CONTENTS.

DOCUMENTATION: USERS MANUAL, TECHNICAL NEWSLETTER

REFERENCES: [BCS79], BOEING COMPUTER SERVICES COMPANY, AUTOMATED SOFTWARE TOOLS CATALOG, BCS 10236, 790600

DEVELOPER: BOEING COMPUTER SERVICES COMPANY

CONTACT: GARY KAMPEL, BOEING COMPUTER SERVICES COMPANY, P.O. BOX 24346, SEATTLE, WA, 98124, USA, 206-575-5393

INFORMATION SOURCE: BCS SOFTWARE TOOLS CATALOG
LILITH

TOOL SUMMARY: LILITH IS A TOOL FOR THE CREATION OF PROGRAMS AND DOCUMENTS. IT IS A GENERAL PURPOSE COMPUTER CREATED TO PROVIDE AN OPTIMAL ENVIRONMENT FOR THE PROGRAMMING LANGUAGE MODULA II. THE MERIT OF THIS SYSTEM AS A PROGRAM AND DOCUMENT PREPARATION TOOL RELIES UPON THE FOLLOWING FEATURES OF THE SYSTEM: (1) THE EFFECTIVENESS AND VERSATILITY OF THE LANGUAGE MODULA II (2) THE HIGH PERFORMANCE IN PROGRAM EXECUTION ACHIEVED BY LILITH BECAUSE OF ITS EXTREMELY COMPACT SOURCE CODE AND ITS SPECIALIZED ARCHITECTURE. (3) THE ABILITY TO SHOW ELABORATE IMAGES WITH A HIGH RESOLUTION DISPLAY AND THE ABILITY TO MANIPULATE THEM WITH POWERFUL FIRMWARE-IMPLEMENTED GRAPHIC OPERATIONS. (4) THE EFFICIENCY FOR ENTERING USER COMMANDS AVAILABLE FROM THE COMBINATION OF THE KEYBOARD AND MOUSE AS INPUT DEVICES. (5) THE RELIABILITY OF BOTH SOFTWARE AND HARDWARE RESULTING FROM THE AVOIDANCE OF UNNECESSARY COMPLEXITY.

DEVELOPER: INSTITUTE FÜR INFORMATIK, ETH
CONTACT: RICHARD OHRAN, BRIGHAM YOUNG UNIVERSITY, EE DEPT.
459 CLYDE BUILDING, PROVO, UTAH, 84602, USA, 801-378-0155
INFORMATION SOURCE: TOOL FAIR

ACRONYM: LOGIC, TITLE: LOGIC
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING FEATURES: SUBJECT, SOURCE PROGRAM, MACHINE, OUTPUT, TABLES, LISTINGS, DYNAMIC ANALYSIS, COVERAGE ANALYSIS, TIMING, TRACING, PATHFLOW TRACING.
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN
COMPUTER (OTHER HARDWARE): CDC 7600/7600
TOOL SUMMARY: LOGIC IS A GROUP OF QUALITY ASSURANCE TOOLS (LOVAR, ESTIMATE, PATH, FILTER, COUNTER, TRANMD) THAT USE THE TABLE INFORMATION GENERATED BY PACE TO STATISTICALLY VERIFY COMPLIANCE WITH DETAIL DESIGN SPECIFICATIONS IN FORTRAN PROGRAMS. LOGIC AUTOMATICALLY DETERMINES ALL EXECUTION PATHS, VERIFIES USAGE OF INPUT/OUTPUT VARIABLES, COUNTS THE NUMBER OF CDC 7600 MACHINE INSTRUCTIONS BETWEEN ENTRY AND EXIT POINTS, AND COMPUTES THE AVERAGE AND MAXIMUM RUN TIME FOR A ROUTINE. THE OUTPUT FOR LOGIC IS THE OUTPUT FROM PACE WHICH INCLUDES A LISTING OF THE USER SOURCE CODE AND TABLE INFORMATION. LOGIC IS HIDDEN IN THE FORTRAN EXTENDED LANGUAGE FOR USE ON THE CDC 7600, BUT IT MAY BE USED ON OTHER COMPUTERS.
REFERENCES: [AD0579], APPLIED SYSTEMS DESIGN SECTION, TRW DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS: CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE TOOLS SERIES 79-110
DEVELOPER: TRW, SEID SOFTWARE PRODUCT ASSURANCE CONTACT: FRANK INGRASSIA, TRW, SEID SOFTWARE PRODUCT ASSURANCE, ONE SPACE PARK, REDONDO BEACH, CA, 90278, USA, 213-356-3140
INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG

LOGIC

ACRONYM: LOGICFLOW, TITLE: LOGICFLOW
CLASSIFICATION: REQUIREMENTS/DATA SPECIFICATION AND ANALYSIS FEATURES: SUBJECT, CODE INPUT, FORTRAN, OPTIMIZATION, MACHINE CONTROL, GRAPHICS, FLOW CHARTS,
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN TOOL PORTABLE: NO
COMPUTER (OTHER HARDWARE): IBM 360/370
TOOL AVAILABLE: NO, PUBLIC DOMAIN: NO
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): PARTS OF LOGICFLOW ARE PROPRIETARY
TOOL SUPPORTED: YES TOOL SUPPORT: LOGICON
TOOL SUMMARY: THE LOGICFLOW SYSTEM IS A LOGICON-DEVELOPED SOFTWARE SYSTEM WHICH AIDS THE DESIGN, DEVELOPMENT, ANALYSIS AND DOCUMENTATION OF RELIABLE SOFTWARE. DURING THE DESIGN PHASE, LOGICFLOW ACCEPTS PROGRAM DESIGN LANGUAGE (PDL) AND PRODUCES A GRAPHIC REPRESENTATION OF THAT DESIGN IN FLOWCHART FORMAT FOR EVALUATION. SINCE LOGICFLOW ANALYZES THE SYNTAX OF THE INPUT TO PRODUCE THE GRAPHIC REPRESENTATION AT THIS STAGE IT ALSO CHECKS FOR BASIC LOGIC ERRORS SUCH AS IMPROPER LOOP CONSTRUCTS, METRICS OF WHICH CAN BE OBTAINED BY INVOKING A METRICS EVALUATOR, THIS FEATURE PROVIDES THE ANALYST NORMIZED METRICS ON SELECTED DESIGN CHARACTERISTICS SUCH AS STRUCTURELESSNESS, SIMPLICITY, AND COMPLEXITY. A CROSS-REFERENCE OF ALL DESIGN NAMES CAN BE OBTAINED USING THE CROSS-REFERENCE GENERATOR, THE LOGICFLOW FLOWCHARTER AUTOMATICALLY PRODUCES HIGH QUALITY FLOWCHARTS FROM DESIGN LANGUAGE OR FORTAN SOURCE.
INFORMATION SOURCE: TOOL FAIR

ACRONYM: LOGICFLOW, TITLE: LOGICFLOW
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE FEATURES: SUBJECT, CODE INPUT, USER OUTPUT, GRAPHICS, FLOW CHARTS, STATIC ANALYSIS, SCANNING,
LOGIFLOW

STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN, ASSEMBLY
COMPUTER (OTHER HARDWARE): CDC 6400/7X00

Tool Summary: LOGIFLOW is a semi-automatic functional flowcharting tool which bridges the gap between the completely automatic flowcharters like AUTOFLOW and FLOWGEN and the completely manual approach. LOGIFLOW provides the user complete flexibility in functionally grouping statements to be flowcharted. The user may be as detailed or as general as desired. LOGIFLOW operates in either a batch or remote terminal environment. The flowcharts are produced by CALCOP PLOTTERS, although designed for use primarily with FORTRAN programs, LOGIFLOW may be used to chart any textual material. For any requirement where meaningful, functional flowcharts must be produced, LOGIFLOW is superior to completely automatic flowcharters like AUTOFLOW or FLOWGEN by allowing the intelligent, functional grouping of statements. It is more convenient than the manual approach in that it automatically generates the flow diagram (with the exception of logical branching lines which the user must fill in) according to user supplied directive.

REFERENCES: [ASD 879], APPLIED SYSTEMS DESIGN SECTION, TRW DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS: CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE TOOLS SERIES, 79-100

DEVELOPER: TRW SIMULATION SOFTWARE DEPT.
CONTACT: J. D. OLIVER, TRW SIMULATION SOFTWARE DEPT., ONE SPACE PARK, REDONDO BEACH, CA, 90278, USA, 213-535-0923
INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG

ACRONYM: LOGOS
TITLE: LOGOS
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE
FEATURES: SUBJECT, CODE INPUT, STRUCTURED FORTRAN, TRANSLATION, FORMATTING, USER OUTPUT, TABLES, LISTINGS, STATISTICAL ANALYSIS, CROSS-REFERENCE
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: STRUCTURED FORTRAN 3
COMPUTER (OTHER HARDWARE): UNIVAC 11X
OS (OTHER SOFTWARE): ECL

TOOL AVAILABLE: YES, PUBLIC DOMAIN: YES

TOOL SUMMARY: PROVIDES A COMPLETE VARIABLE INDEX FOR PROGRAMS WRITTEN IN FORTRAN OR SFTTRAN. ALSO WILL TRANSLATE LONG VARIABLE NAMES USED IN SFTTRAN SYMBO LCS INTO NAMES COMPATIBLE WITH BOTH SFTTRAN 3 AND FORTRAN ON THE UNIVAC 11X.

DOCUMENTATION: USER'S MANUAL
DEVELOPER: JET PROPULSION LABORATORY
CONTACT: SANDY PALACIOS, JET PROPULSION LABORATORY, 4600 OAK GROVE DRIVE, PASADENA, CA, 91109, USA, 213-346-4864
INFORMATION SOURCE: AIAA SURVEY OF SOFT DEV TOOLS

LOGOS

ACRONYM: LOGOS
TITLE: LOGOS
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE
FEATURES: SUBJECT, DATA INPUT, USER OUTPUT, GRAPhICS, TABLES, DYNAMIC ANALYSIS, RESOURCE UTILIZATION
STAGE OF DEVELOPMENT: IMPLEMENTED
COMPUTER (OTHER HARDWARE): IBM 360/370
OS (OTHER SOFTWARE): SVS, OS/VS, OS/VMS, OS/MVT

TOOL AVAILABLE: YES

TOOL SUPPORTED: YES, TOOL SUPPORT: APPLIED DATA RESEARCH

TOOL SUMMARY: LOOK IS A FLEXIBLE AND WIDE-RANGING MEASUREMENT TOOL THAT IS CAPABLE OF PROVIDING AN IMMEDIATE AND ACCURATE APPRAISAL OF TODAY'S DYNAMIC COMPUTER ENVIRONMENTS. LOOK'S INFORMATION CAN HELP AVOID COMMON BOTTLENECKS, SUCH AS OPERATIONAL CONFLICTS THAT SERIOUSLY IMPED PRODUCTION SCHEDULES, LOOK OFFERS: (1) DYNAMIC SYSTEM CONTROLS THAT PERMIT YOU TO CHANGE A JOB'S DISPATCHING PRIORITY, (2) THE DISPLAYS THAT SPOTLIGHT WAIT CONDITIONS, THE ALLOCATION OF DEVICES OR DATA SETS, THE CONTENTS OF CONTROL BLOCKS, OR CURRENT CORE USE, (3) PERFORMANCE DISPLAYS THAT CHARACTERIZE CPU USE, I/O ACTIVITY AND PAGING ACTIVITY; OTHER DISPLAYS HELP TO CONTROL THE SRM, THE ASM AND TSO, (4) EXTENDED PERFORMANCE ANALYSIS OPTION THAT ANALYZES AND GRAPHICALLY PRESENTS LONG-TERM DATA TO SPOT-TREND DATA TO SPOT-TREND TRENDS, OR TO PREDICT APPROPRIATE SCHEDULES.

DEVELOPER: APPLIED DATA RESEARCH
CONTACT: APPLIED DATA RESEARCH, ROUTE 206 CENTER, CN-6, PRINCETON, NJ, 08540, USA, 609-924-9101
INFORMATION SOURCE: PRODUCT DESCRIPTION

ACRONYM: MAGLE
TITLE: META ASSEMBLER AND GENERALIZED LINKAGE EDITOR
CLASSIFICATION: PROGRAM CONSTRUCTION AND GENERATION
FEATURES: SUBJECT, CODE INPUT, ASSEMBLY LANGUAGE, TRANSFORMATION, TRANSLATION, CONVERSION, MACHINE OUTPUT, ASSEMBLY LANGUAGE

STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN IV

TOOL PORTABLE: YES, TOOL SIZE: 40,000 STATEMENTS-50% COMMENTS

TOOL AVAILABLE: YES, PUBLIC DOMAIN: YES

TOOL SUPPORTED: YES, TOOL SUPPORT: MCDONnell DOUGLAS ASTRONAUTICS CORP.

TOOL SUMMARY: A GENERAL PURPOSE ASSEMBLER PARAMETERIZED ON COMMON BLOCKS WHICH ARE CONFIGURED BY A GENERAL PURPOSE SYNTAX-DIRECTED PARSER; AN ACCOMPANYING LINKER WHICH PROVIDES MACHINE-INDEPENDENT LINKAGE EDITOR SERVICES. IN ESSENCE, A GENERAL PURPOSE TOOL FOR THE CREATION OF PORTABLE CROSS ASSEMBLERS WITH MINIMAL PROGRAMMING EFFORT; TYPICAL ASSEMBLER GENERATION EFFORT IS 1 TO 5 MAN-DAYS. SIDE EFFECT IS MACHINE-READABLE HIGH ORDER ASSEMBLER
SPECIFICATION
DOCUMENTATION: USER'S MANUAL, MAINTENANCE MANUAL
DEVELOPER: TRW, MCDONNELL DOUGLAS ASTRONAUTICS CORP.
CONTACT: K. V. SMITH, MDAC, DEPT. 259, AEDO, 5301 BOLSA AVE., HUNTINGTON BCH., CA, 92645, USA, 714-896-4155
T. L. MINSLOW, TRW DSSG, P.O. BOX 2229, WARNER ROBINS, GA, 31099, USA, 912-922-3001
INFORMATION SOURCE: COMPLETED SUBMISSION TO NBS
ACRONYM: MARK IV (TM), TITLE: MARK IV (TM)
CLASSIFICATION: PROGRAM CONSTRUCTION AND GENERATION
FEATURES: SUBJECT, DATA INPUT, VHLL INPUT, DESCRIPTION LANGUAGE, TRANSFORMATION, SYNTHESIS, MACHINE OUTPUT, DATA OUTPUT, USER OUTPUT, DIAGNOSTICS, GRAPhICS, TABLES,
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: ASSEMBLY
TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): MARKETED PRODUCT
TOOL SUPPORTED: YES, TOOL SUPPORT: INFORMATICS, INC.
TOOL SUMMARY: MARK IV IS A GENERAL PURPOSE SOFTWARE SYSTEM FOR THE DESIGN, IMPLEMENTATION AND OPERA TION OF DATA PROCESSING APPLICATIONS. IT IS IN USE AT OVER 1500 COMPUTER SITES IN 44 COUNTRIES, DESIGNED AS AN ADJUNCT OR ALTERNATIVE TO COBOL OR PL/1, IT ENHANCES PROGRAMMER PRODUCTIVITY BY CUTTING DEVELOPMENT AND MAINTENANCE TIME AND COSTS. TYPICAL INSTALLATIONS REPORT USES RANGING FROM COMPLEX FINANCIAL AND OPERATION APPLICATIONS TO ONE-TIME REPORTS, ITS EXTENSIVE USE OF AUTOMATIC FUNCTIONS AND COMPLETE DATA INDEPENDENCE HAVE MADE IT THE STANDARD LANGUAGE IN MANY INSTALLATIONS. THE MARK IV PRODUCT LINE CONSISTS OF THREE MODELS WITH DATA BASE AND ON-LINE OPTIONS.
DOCUMENTATION: USER'S MANUAL
REFERENCES: [WEI61], D. J. REIFER AND H. A. MONTGOMERY, "SEATECS SOFTWARE TOOLS SURVEY", RCI=TR-008, REIFER CONSULTANTS, INC., 810300
DEVELOPMENT INFORMATICS, INC.
CONTACT: RON MULENAUX, INFORMATICS, INC., 21050 VANOWEN ST., CANOGA PARK, CA, 91304, USA, 213-887-9121
INFORMATION SOURCE: NO SEATECS TOOLS SURVEY
ACRONYM: MED-SYS, TITLE: MULTI-LEVEL EXPRESSION DESIGN SYSTEM
CLASSIFICATION: REQUIREMENTS/DISIGN SPECIFICATION AND ANALYSIS
FEATURES: SUBJECT, VHLL INPUT, REQUIREMENTS LANGUAGE, DESIGN LANGUAGE, USER OUTPUT, DIAGNOSTICS, LISTINGS, STATIC ANALYSIS, CONSISTENCY CHECKING, TRACKING,
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN
TOOL SIZE: CORE: 30K
COMPUTER (OTHER HARDWARE): DEC PDP-11

OS (OTHER SOFTWARE): RSX-11
TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): FOR LEASE
TOOL SUMMARY: MED-SYS CONSISTS OF A FAMILY OF LOW COST INTERACTIVE SUBSYSTEMS THAT PROVIDE THE USER WITH COMPREHENSIVE SOFTWARE TOOLS FOR THE INITIAL PHASES IN THE SOFTWARE LIFE CYCLE. THIS APPROACH PROVIDES FOR A COMPARISON OF DESIGN QUALITY USING STRUCTURAL DECOMPOSITION AND OTHER VERIFICATION AND VALIDATION TECHNIQUES. MED-SYS CONSISTS OF THREE SYSTEMS: MED-L REQUIREMENTS STATEMENTS, MED-D DESIGN STRUCTURE, AND MED-P PROCEDURAL SPECIFICATION. THE THREE MED-SYS LEVELS MAY BE USED IN AN INTERACTIVE ENVIRONMENT OR AS STAND-ALONE SYSTEMS.
DOCUMENTATION: USER'S MANUAL, MAINTENANCE MANUAL, DEVELOPMENT CONCEPTS, TEST PLAN,
TOOL DEVELOPER: MARTIN MARIETTA AEROSPACE
CONTACT: AMY MUSIERT, MARIETTA MARIETTA AEROSPACE, P.O. BOX 179, CODE 0422, DENVER, CO, 80201, USA, 303-773-3599
INFORMATION SOURCE: AIAA SURVEY OF SOFT DEV TOOLS
ACRONYM: MED-L, TITLE: MULTI-LEVEL EXPRESSION DESIGN SYSTEM
CLASSIFICATION: REQUIREMENTS/DISIGN SPECIFICATION AND ANALYSIS
FEATURES: SUBJECT, TEXT INPUT, VHLL INPUT, DESIGN SPECIFICATION, USER OUTPUT, USER-ORIENTED TEXT, REPORTS, TABLES, STATIC ANALYSIS, CONSISTENCY CHECKING, MANAGEMENT, DATA BASE MANAGEMENT,
STAGE OF DEVELOPMENT: IMPLEMENTED, DATE (YYMMDD): 790000
IMPLEMENTATION LANGUAGE: FORTRAN IV PLUS
TOOL SIZE: 15,000 SOURCE LINES
COMPUTER (OTHER HARDWARE): DEC VAX-11
OS (OTHER SOFTWARE): VMS
PUBLIC DOMAIN: NO
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): CONTACT MARTIN MARIETTA FOR DETAILS
TOOL SUPPORTED: YES, TOOL SUPPORT: MARTIN MARIETTA
TOOL SUMMARY: MED-D IS A COMPONENT OF MED-SYS THAT PROVIDES AN INTERACTIVE REQUIREMENTS LANGUAGE. MED-D IS AN INTERACTIVE HIGH-LEVEL DESIGN TOOL ORIENTED PRIMARILY TOWARD DEFINING AND CAPTURING SOFTWARE STRUCTURES AND COMPONENT PROCEDURE. MED-D IS BASED UPON A CONCEPTUAL FRAMEWORK THAT RECOGNIZES THREE BASIC COMPONENT CATEGORIES CALLED FUNCTIONAL, DATA, AND TEMPORAL OBJECTS, CENTERED ABOUT THESE BASIC DESIGN COMPONENTS ARE MEANS FOR CHARACTERIZING THESE ENTITIES AND THE OBJECTS WITH WHICH THEY ARE ASSOCIATED. IN ORDER TO SUPPORT HIERARCHICAL DEFINITIONS OF ACTIVE DATA STRUCTURES, FUNCTIONAL DECOMPOSITION, AND DISCRETE SCHEDULING, EACH OBJECT IS IMPLICITLY RELATED TO ITSELF. THE CHARACTERISTICS OF FUNCTIONAL OBJECTS ARE REFERRED TO AS PROPERTIES, WHEREAS DATA OBJECTS POSSESS ATTRIBUTES AND TEMPORAL OBJECTS ARE CHANGEPACED WITH RATES. THESE DISTINCTIONS ALLOW FOR A LOGICAL SEGREGATION IN THE VOCABULARY OF THE MED-D LANGUAGE, WHEN MED-D IS
USED IN CONJUNCTION WITH MEDL-R LINKS IN THE DATABASES OCCUR BY WAY OF COMMON ENTITIES.

DOCUMENTATION: USER'S MANUAL
DEVELOPER: MARTIN MARIETTA
CONTACT: MARTIN MARIETTA, PO BOX 179, DENVER, CO, 80201, USA, 303-977-4313
INFORMATION SOURCE: PRODUCT DESCRIPTION

ACRONYM: MEDL-P, TITLE: MULTI-LEVEL EXPRESSION DESIGN LANGUAGE - PROCEDURE
CLASSIFICATION: SOFTWARE MODELING AND SIMULATION
FEATURES: SUBJECT, VML INPUT, SYSTEM SPECIFICATION, STATIC ANALYSIS, MANAGEMENT, DATABASE MANAGEMENT, DYNAMIC ANALYSIS, SIMULATION,

STAGE OF DEVELOPMENT: CONCEPT, DATE (YMD): 03/00
PUBLIC DOMAIN: NO
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): CONTACT MARTIN MARIETTA FOR DETAILS

TOOL SUPPORTED: YES, TOOL SUPPORT: MARTIN MARIETTA

TOOL SUMMARY: MEDL-P WILL BE AN INTERACTIVE PROCESSOR WHICH WILL SUPPORT SOFTWARE ENGINEERS IN THEIR EFFORTS TO CAPTURE AND ANALYZE THE DETAILS OF SOFTWARE SYSTEM CONTROL, WHEREAS MEDL-D IS PRIMARILY CONCERNED WITH THE EXPRESSION OF MODULE-LEVEL DESIGN CONCEPTS AND THE DESCRIPTION OF RELATIONSHIPS BETWEEN DESIGN ENTITIES, MEDL-P WILL BE CONCERNED WITH THE DESCRIPTION OF EXECUTION CONTROL.

MEDL-P WILL THEN BE USED AS AN AID IN THE PROCESS OF DETERMINING WHETHER OR NOT THE SYSTEM (BEING DESIGNED) WILL PERFORM WITHIN EXPECTED PARAMETERS.

DEVELOPER: MARTIN MARIETTA
CONTACT: MARTIN MARIETTA, PO BOX 179, DENVER, CO, 80201, USA, 303-977-4313
INFORMATION SOURCE: PRODUCT DESCRIPTION

ACRONYM: MEDL-R, TITLE: MULTI-LEVEL EXPRESSION DESIGN LANGUAGE - REQUIREMENTS
CLASSIFICATION: REQUIREMENTS/DESIGN SPECIFICATION AND ANALYSIS
FEATURES: SUBJECT, TEXT INPUT, VML INPUT, REQUIREMENTS SPECIFICATION, USER OUTPUT, USER-ORIENTED TEXT, REPORTS, TABLES, STATIC ANALYSIS, COMPLETENESS CHECKING, CONSISTENCY CHECKING, MANAGEMENT, DATABASE MANAGEMENT, TRACKING,

STAGE OF DEVELOPMENT: IMPLEMENTED, DATE (YMD): 03/00
IMPLEMENTATION LANGUAGE: FORTRAN IV PLUS

TOOL SUPPORTED: 10,000 SOURCE LINES OF COMPUTER (OTHER HARDWARE): DEC PDP-11, DEC VAX-11
OS: VMS, RSX-11
PUBLIC DOMAIN: NO
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): CONTACT MARTIN MARIETTA FOR DETAILS

TOOL SUPPORTED: YES, TOOL SUPPORT: MARTIN MARIETTA

TOOL SUMMARY: MEDL-R IS A COMPONENT OF MедSYS THAT PROVIDES AN INTERACTIVE REQUIREMENT LANGUAGE THAT GRADUALLY IMPOSES DISCIPLINE UPON THE PROCESS BY REQUESTING THE USER TO RESPOND TO A FEW RIGOROUS QUESTIONS ABOUT EACH REQUIREMENT, WHILE STILL ALLOWING A GOOD DEAL OF EXPRESSION IN ENGLISH TEXT. THE LANGUAGE RELIES HEAVILY UPON KEYWORDS TO SIMPLIFY THE CHARACTERIZATION OF REQUIREMENTS, EACH OF THE MEDL-R KEYWORDS NOT ONLY PROVIDES AN ASPECT OF CHARACTERIZATION, BUT ALSO IMPLICITLY RELATES REQUIREMENTS WHICH POSSESS COMMON ASPECTS. THIS RESULTS IN A LARGE NUMBER OF IMPLIED INTERRELATIONSHIPS WHICH PROVIDE THE BASIS FOR SUBSEQUENT ANALYSIS OPERATIONS. UNDER MEDL-R, ALL REQUIREMENT INTERRELATIONSHIPS AND INTERDEPENDENCIES ARE RETAINED, AND OBSOLETE ITEMS, THOUGH ARCHIVED, REMAIN AVAILABLE FOR MANIPULATION AND ANALYSIS BY THE USER.

ADDITIONAL MEDL-R FEATURES PROVIDE FOR REQUIREMENT TO DESIGN ITEM TRACEABILITY, SOURCE TO DESTINATION DOCUMENT TRACEABILITY CONSISTENCY AND COMPLETENESS CHECKING, AND DATABASE REPORTING.

DOCUMENTATION: USER'S MANUAL
DEVELOPER: MARTIN MARIETTA
CONTACT: MARTIN MARIETTA, PO BOX 179, DENVER, CO, 80201, USA, 303-977-4313
INFORMATION SOURCE: PRODUCT DESCRIPTION

ACRONYM: MEDL-X, TITLE: MULTI-LEVEL EXPRESSION DESIGN LANGUAGE - TEXT PROCESSING
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE
FEATURES: SUBJECT, TEXT INPUT, TRANSFORMATION, EDITING, USER-ORIENTED TEXT, REPORTS, STATIC ANALYSIS, MANAGEMENT, DATABASE MANAGEMENT,

STAGE OF DEVELOPMENT: DESIGN, DATE (YMD): 03/00
PUBLIC DOMAIN: NO
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): CONTACT MARTIN MARIETTA FOR DETAILS

TOOL SUPPORTED: YES, TOOL SUPPORT: MARTIN MARIETTA

TOOL SUMMARY: MEDL-X WILL PROVIDE THE USER WITH THE ABILITY TO INTERACTIVELY ASSEMBLE, EDIT, ANALYZE, AND PUBLISH THE SOFTWARE DOCUMENTS WHICH WHEN COMBINED, FORM AN INTEGRAL COMPONENT OF THE DEVELOPMENT PROCESS. MEDL-X WILL NOT BE MERELY ANOTHER TEXT EDITOR OR WORD PROCESSOR INSTALLED ON A MINICOMPUTER. THE USEFULNESS AND POWER OF MEDL-X ARE DERIVED FROM ITS ABILITY TO EMPLOY THE INFORMATION CONTAINED WITHIN THE FILES ASSOCIATED WITH THE OTHER MедSYS PROCESSORS IN ADDITION TO ITS OWN DATABASE. ADDITIONALLY, MEDL-X ALLOWS THE TEXT OF STANDARD "BOILERPLATE" PARAGRAPHS TO BE STORED WITHIN A "BOILERPLATE" FILE FOR SUBSEQUENT INCLUSION INTO A GIVEN DOCUMENT. THE FORMAT AND CONTENT OF A GIVEN SOFTWARE DOCUMENT IS DETERMINED BY THE STANDARDS OF THE CUSTOMER OR THE SOFTWARE DEVELOPER. BY ALLOWING THE "RULES" ASSOCIATED WITH A GIVEN DOCUMENT TO BE STORED WITHIN AN EASILY UPDATED FILE, MEDL-X WILL MAINTAIN ITS ABILITY TO SERVE, IRRESPECTIVE OF THE VOLATILITY WHICH MAY aFFECT A GIVEN SET OF STANDARDS.
MEMORY MNG LIB

DESCRIPTION: AUTOMATICALLY MANAGE MEMORY SO AS TO ALLOCATE TO A PROGRAM ONLY THE AMOUNT OF MEMORY SPACE IT NEEDS FOR ITS DATA FOR A GIVEN EXECUTION, AND NO MORE. LARGE MEMORY SAVINGS AND EMBARRASSING OVERFLOWS ARE MINIMIZED. SOFTWARE OFFERS A FAMILY OF DYNAMIC MEMORY MANAGERS. EACH MEMBER OF THE FAMILY EMPLOYS A DIFFERENT STRATEGY FOR THE MANAGEMENT OF MEMORY SPACE. THE STRATEGIES VARY FROM THE VERY SIMPLE TO THE VERY SOPHISTICATED, INCLUDING ONE THAT IMPLEMENTS A VIRTUAL MEMORY FACILITY AND ALLOWS THE PROGRAMMER TO DEFINE THE VIRTUAL ARCHITECTURE (E.G., PAGE SIZE, VIRTUAL SPACE).

DOCUMENTATION: TECHNICAL REPORTS, USERS MANUAL

DEVELOPER: SOFTOOL CORPORATION

CONTACT: H. KELLOGG AVE., GOLETA, CA, 93117, USA, 805-964-0560

INFORMATION SOURCE: COMPLETED SUBMISSION TO NBS

ACRONYM: MEMORY MNG LIB; TITLE: MEMORY MANAGERS LIBRARY
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE
FEATURES: SUBJECT, CODE INPUT, FORTRAN, OPTIMIZATION, MIRROR OUTPUT, SOURCE CODE OUTPUT, FORTRAN
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN
TOOL PORTABLE: YES
TOOL AVAILABLE: YES, PUBLIC DOMAIN; NO
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): MARKETED COMMERCIALLY
TOOL SUPPORT: YES, TOOL SUPPORT: SOFTOOL CORPORATION

TOOL SUMMARY: FORTRAN PROGRAMMERS ARE FORCED, BY THE VERY DEFINITION OF THE FORTRAN LANGUAGE, TO ALLOCATE MEMORY SPACE FOR THEIR DATA IN A WASTEFUL MANNER. HOWEVER, MOST FORTRAN IMPLEMENTATIONS ALLOCATE TO A PROGRAM AN AMOUNT OF MEMORY EQUAL TO THE MAXIMUM THAT MAY EVER BE USED BY THE PROGRAM, RATHER THAN THE AMOUNT NEEDED FOR EACH SPECIFIC RUN. THE PURPOSE OF A DYNAMIC MEMORY MANAGER IS TO

ACRONYM: METRAN; TITLE: META TRANSLATOR
CLASSIFICATION: PROGRAM CONSTRUCTION AND GENERATION
FEATURES: SUBJECT, VMIL INPUT, DESCRIPTION LANGUAGE, OPTIMIZATION, SYNTHESIS, MACHINE OUTPUT, SOURCE CODE OUTPUT,
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN
TOOL PORTABLE: YES, TOOL SIZE: WORDS POP 10, 32+8
TOOL AVAILABLE: NO, PUBLIC DOMAIN: NO

TOOL SUMMARY: THE META TRANSLATOR IS A GENERAL PURPOSE TOOL, WHICH PROVIDES THE USER WITH AUTOMATED TECHNIQUES FOR THE GENERATION OF LANGUAGE SPECIFIC TRANSLATORS. AN ASSOCIATED META-LANGUAGE (METRAN) PROVIDES A DESCRIPTIVE MEDIUM FOR THE USER TO DESCRIBE THE SYNTAX OF THE SOURCE LANGUAGE IN A MODIFIED BNF FORM AND TO SPECIFY TRANSLATIONAL AS WELL AS CODE GENERATIVE SEMANTICS. TRANSLATIONAL DATA STRUCTURES ARE DEFINABLE AT THE METRAN LEVEL WHICH ALLOW THE USER TO DEFINE AND SUBSEQUENTLY EXPLOIT HASH CODED SYMBOL TABLES (WITH ATTRIBUTES) PUSH DOWN=POP UP (FIFO) STACKS LINKED DYNAMIC LISTS OF DATA STRUCTURES; SYNTACTIC PARSING PRIMITIVES ARE PROVIDED SO THAT THE USER NEED NOT BE CONCERNED WITH AN AD HOC CREATION OF A PARSING ALGORITHM. THE BNF-LIKE DESCRIPTION OF THE LANGUAGE IS MAPPED INTO A FORTRAN PARSING PROGRAM WHICH UTILIZES THESE PARSING PRIMITIVES TO PRODUCE A TOP-DOWN RECURSIVE DESCENT PARSER FOR THE LANGUAGE WITH EMBEDDED LINKAGES TO SYSTEM SEMANTIC PROCEDURES OR THE DESIRED USER DEFINED SEMANTIC PROCEDURES.

DOCUMENTATION: USER'S MANUAL
DEVELOPER: MCDOUGAL DOUGLAS ASTRONAUTICS CORP.
CONTACT: K. V. SMITH, MIDAC, DEPT. 239, AEDO, 5301 BOLSA AVE., HUNTINGTON BCH., CA, 92647, USA, 714-896-9155
INFORMATION SOURCE: AIAA SURVEY OF SOFT DEV TOOLS

ACRONYM: MODULE ORDERER, TITLE: MODULE ORDERER
CLASSIFICATION: PROGRAM CONSTRUCTION AND GENERATION
FEATURES: SUBJECT, CODE, INPUT, OBJECT CODE, TRANSFORMATION, FORMATTING, USER OUTPUT, LISTINGS,
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN
COMPUTER (OTHER HARDWARE): DATA GENERAL
TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): MARKETED COMMERCIAL.

TOOL SUPPORTED: YES, TOOL SUPPORT: SOFTWARE CORPORATION.

TOOL SUMMARY: IN THE CASE OF DATA GENERAL AGS SYSTEMS, THE INTERPLAY BETWEEN THE LIBRARY AND THE BINDER IS OF CRUCIAL IMPORTANCE TO SUCCESSFUL SYSTEMS UTILIZATION. THE BINDER REQUIRES THAT ALL MODULES PRESENTED TO IT FOR BINDING, WHICH ORIGINATE IN AN UNSHARED LIBRARY, BE IN FORWARD ORDERING. THIS REQUIREMENT PLACES A CLERICAL BURDEN ON THE USER WHICH, IN MANY Instances, MAKES BINDING WITH USER CREATED LIBRARIES UNNECESSARILY TROUBLESOME AND TIME CONSUMING. THE MODULE ORDERER IS A SOFTWARE PRODUCT THAT ACCEPTS AS INPUT A COLLECTION OF OBJECT MODULES AND AUTOMATICALLY ORGANIZES AND STORES THEM IN A LIBRARY IN FORWARD ORDERING. THE TOOL IS VERY EASY TO USE. THE AUTOMATIC FORWARD ORDERING OF MODULES IS AN IMPORTANT AND BENEFICIAL ENHANCEMENT THAT REMOVES THE USER OF AGS SYSTEMS FROM MUCH CLERICAL AND ERROR-PRONE WORK. THIS PRODUCT IS A MEMBER OF
NETWORK PLANNER

FEATURES: SUBJECT, VHLL INPUT, REQUIREMENTS SPECIFICATION, USER OUTPUT, GRAPHICS, STATIC ANALYSIS, MANAGEMENT, PROJECT MANAGEMENT,

STAGE OF DEVELOPMENT: IMPLEMENTED, DATE (YYMMDD): 80010
IMPLEMENTATION LANGUAGE: FORTRAN IV

TOOL SIZE: 500 STATEMENTS

TOOL AVAILABLE: YES, PUBLIC DOMAIN: YES

TOOL SUPPORTED: YES, TOOL SUPPORT: UNIVERSITY OF WAIKATO

TOOL SUMMARY: THE NETWORK PLANNER IS A MODULE FOR EVALUATING PLANS OF SOFTWARE PROJECTS FOR EFFICIENCY AND COST. IT USES A CPM APPROACH TO EVALUATE ALTERNATIVE PLANS AND GIVES A GRAPHICAL REPRESENTATION OF THE PLANS.

DOCUMENTATION: TECHNICAL DOCUMENTATION (10), USER DOCUMENTATION (5)

REFERENCES: [PAY75], A.J. PAYNE, "DESIGN OF DISTRIBUTED COMPUTERS", 9TH NZ MATHEMATICS CONGRESS, 750000

DEVELOPER: A.J. PAYNE

CONTACT: A.J. PAYNE, UNIVERSITY OF WAIKATO, HAMILTON, NEW ZEALAND

INFORMATION SOURCE: COMPLETED SUBMISSION TO NBS

ACRONYM: NODAL, TITLE: NODE DETERMINATION AND ANALYSIS PROGRAM

CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING FEATURES: SUBJECT, CODE INPUT, FORTRAN, TRANSFORMATION, INSTRUMENTATION, MACHINE OUTPUT, SOURCE CODE OUTPUT, FORTRAN, USER OUTPUT, TABLES, LISTINGS, STATIC ANALYSIS, STATISTICAL ANALYSIS, PROFILE GENERATION, MANAGEMENT, TEST DATA MANAGEMENT, DYNAMIC ANALYSIS, COVERAGE ANALYSIS, TUNING,

STAGE OF DEVELOPMENT: IMPLEMENTED, DATE (YYMMDD): 780600

IMPLEMENTATION LANGUAGE: FORTRAN

TOOL PORTABLE: YES

COMPUTER (OTHER HARDWARE): IBM 360/370, DEC SYSTEM 10/20, CDC 6600/7600, UNIVAC 11X

TOOL AVAILABLE: YES

TOOL SUMMARY: NODAL IS DESIGNED TO AID THE USER IN EXECUTING ALL THE SOURCE CODE AND ALL THE BRANCHES IN TESTING A FORTRAN PROGRAM. IT USES THE TECHNIQUE OF ANALYZING THE CODE OF AN EXISTING PROGRAM AND INSTRUMENTING OR ADDING CODE THAT WILL RECORD THE EXECUTION OF THE PROGRAM'S NODES. A NODE IS DEFINED AS AN ENTRY POINT TO THE SMALLEST CONSEQUENTLY EXECUTABLE STATEMENTS TO WHICH CONTROL CAN BE GIVEN DURING PROGRAM EXECUTION. AT THE NORMAL END OF AN EXECUTION OF THE USER'S INSTRUMENTED PROGRAM, NODAL WILL OBTAIN CONTROL AND PROVIDE INFORMATION ABOUT THE FREQUENCY OF EXECUTION OF EACH NODE. ALSO PROVIDED IS A TEST EFFECTIVENESS RATIO (NODES EXECUTED/NODES IDENTIFIED) FOR EACH ROUTINE, A TEST EFFECTIVENESS RATIO FOR THE ENTIRE PROGRAM, AND A LIST OF THE PROGRAM NODES NOT EXECUTED. NODAL INCREASES ON RUN TIME APPROXIMATELY 8% AND CORE REQUIREMENTS 25%.

NODAL

DOCUMENTATION: TECHNICAL DESCRIPTION

REFERENCES: [A00579], APPLIED SYSTEMS DESIGN SECTION, TRW DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS: CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE TOOLS SERIES, 790100

DEVELOPERS: TRW SOFTWARE TECHNOLOGY DEPARTMENT

CONTACT: R. L. MAITLEN, TRW SOFTWARE TECHNOLOGY DEPARTMENT, ONE SPACE PARK, REDONDO BEACH, CA, 90278, USA, 213-535-3480

INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG

ACRONYM: NUMBER, TITLE: FORTRAN STATEMENT RENUMBERING PROGRAM

CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING FEATURES: SUBJECT, CODE INPUT, FORTRAN, TRANSFORMATION, FORMATTING, MACHINE OUTPUT, SOURCE CODE OUTPUT, FORTRAN, USER OUTPUT, LISTINGS, STATIC ANALYSIS, SCANNING,

STAGE OF DEVELOPMENT: IMPLEMENTED, DATE (YYMMDD): 770000

IMPLEMENTATION LANGUAGE: FORTRAN

COMPUTER (OTHER HARDWARE): IBM 360/370

OS (OTHER SOFTWARE): OS/MVT

TOOL AVAILABLE: YES

RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): AVAILABLE THROUGH THE FEDERAL SOFTWARE EXCHANGE CENTER, FSWEC-77/0203

TOOL SUMMARY: NUMBER IS A FORTRAN PROGRAM FOR RENUMBERING THE STATEMENT NUMBERS OF A FORTRAN PROGRAM INTO A SEQUENTIAL ORDER, USING AN INCREMENT OF TEN. AN OPTION IS PROVIDED FOR VISUAL VERIFICATION OF THE RESULTS. FORMAT STATEMENTS ARE NUMBERED WITH THE OTHER STATEMENTS, NOT SEPARATELY AS IN SOME PROGRAMS.

REFERENCES: [FDEC80], GENERAL SERVICES ADMINISTRATION/NATIONAL TECHNICAL INFORMATION SERVICE, "FEDERAL SOFTWARE EXCHANGE CATALOG", GSA/ADTS/C=80/1, PB80-904001, 800100

CONTACT: GSA FEDERAL SOFTWARE EXCHANGE, 2 SKYLINE PL (11TH FL), 5203 LEESBURG PK, FALLS CHURCH, VA, 22041, 703-756-2610

INFORMATION SOURCE: FEDERAL SOFTWARE EXCHANGE CATALOG

ACRONYM: NUMBER/DEC, TITLE: NUMBER

CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE

FEATURES: SUBJECT, CODE INPUT, COBOL, TRANSFORMATION, FORMATTING, MACHINE OUTPUT, SOURCE CODE OUTPUT, COBOL, USER OUTPUT, LISTINGS,

STAGE OF DEVELOPMENT: IMPLEMENTED

IMPLEMENTATION LANGUAGE: COBOL

TOOL PORTABLE: NO

COMPUTER (OTHER HARDWARE): DEC PDP-11

OS (OTHER SOFTWARE): RSX-11

PUBLIC DOMAIN: NO

RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): RESTRICTED RIGHTS
TOOL SUMMARY: NUMBER IS A PROGRAMMING TOOL FOR COBOL USERS. ITS PRIMARY PURPOSES ARE TO DOCUMENT THE LOCATIONS OF PARAGRAPHS THROUGHOUT A COBOL PROGRAM AND TO REFORMAT THE PROGRAM BY ALIGNING CERTAIN DATA ITEMS. SINCE FORMAT IS HIGHLY STRESSED IN STRUCTURED PROGRAMMING, IT IS IMPORTANT TO BE ABLE TO EASILY MAINTAIN THE FORMATTING ORIGINALLY DESIGNED INTO A PROGRAM. NUMBER AIDS IN THE PROCESS BY MAINTAINING THE ALIGNMENT OF PIC AND VALUE CLAUSES PERMUTATIONS POSSIBLE AND BY THE INSERTION OF BLANK LINES BETWEEN MARGIN ITEMS. E.G., GROUP ITEM FIELDS, SECTIONS AND PARAGRAPHS, OPTIONALLY, THE RECEIVING FIELDS OF MOVE STATEMENTS CAN ALSO BE ALIGNED IN THE PROCEDURE DIVISION.

REFERENCE: [REIF81], G. J. REIFER AND H. A. MONTGOMERY, "SEATEC SOFTWARE TOOLS SURVEY", RCI-TR-008, REIFER CONSULTANTS, INC., 810330

DEVELOPER: DIGITAL EQUIP. CORP., ASK (MLS=2/M44S)=A1000, 146 MAIN ST., MAYNARD MA.

INFORMATION SOURCE: NOC SEATEC TOOLS SURVEY

ACRONYM: ONLINE ASSIST, TITLE: A USER INTERFACE FOR ONLINE ASSISTANCE
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND DOCUMENTATION
FEATURES: SUBJECT, TEXT INPUT, USER OUTPUT, USER-ORIENTED TEXT, DOCUMENTATION, STATIC ANALYSIS, MANAGEMENT, DOCUMENTATION MANAGEMENT
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN, C

TOOL AVAILABLE: YES

COMPUTER (OTHER HARDWARE): DEC PDP-11, UNIVAC 11XX
OS (OTHER SOFTWARE): EXEC B, UNIX

TOOL SUMMARY: FROM THE USER'S POINT OF VIEW, SPECIAL ASSISTANCE FUNCTIONS ARE ALWAYS AVAILABLE TO REQUEST INFORMATION OR TO OBTAIN SUCCESSFULLY MORE DETAILED EXPLANATIONS OF A DISPLAYED MESSAGE. THESE REQUESTS ARE ENTERED THROUGH FUNCTION KEYS OR AS SPECIAL CODES AND DO NOT AFFECT THE INTERPRETATION OF OTHER INPUT. DIFFERENT TYPES OF AIDS MAY BE REQUESTED. FOR INSTANCE, A USER WHO MAKES AN ERROR WHEN RESPONDING TO A SYSTEM PROMPT CAN OBTAIN FURTHER EXPLANATION OF THE ORIGINAL QUESTION, FURTHER EXPLANATION OF THE ERROR MESSAGE, OR EXAMPLES OF CORRECT RESPONSES, FROM THE PROGRAMMER'S POINT OF VIEW. EACH MULTI-LEVEL MESSAGE (CALLED A SCRIPT) IS WRITTEN AS A SEPARATE FILE, IN BOTH IMPLEMENTATIONS, ALL THE SCRIPTS REQUIRED FOR A GIVEN APPLICATION ARE GROUPED TOGETHER IN A SINGLE FILE CALLED THE MESSAGE FILE, ALTHOUGH SINGLE SCRIPTS CAN STILL BE ACCESSED INDEPENDENTLY.

DOCUMENTATION: USERS GUIDE
REFERENCES: [RELL81], REELLES, V. AND L. A. PRICE, "A USER INTERFACE FOR ONLINE ASSISTANCE", 5TH INTERNATIONAL CONFERENCE ON SOFTWARE ENGINEERING, 810000

ONLINE ASSIST

CONTACT: NATHAN RELLES, SPERRY UNIVAC, MS 263, PO BOX 500, BLUE BELL, PA, 19424, USA, 215-542-2307

INFORMATION SOURCE: TOOL FAIR

ACRONYM: OPTIMIZER II, TITLE: OPTIMIZER II
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, COBOL, ANSI COBOL, TRANSFORMATION, OPTIMIZATION, MACHINE OUTPUT, OBJECT CODE OUTPUT
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: ASSEMBLY

TOOL SUMMARY: OPTIMIZER II AUTOMATICALLY IMPROVES THE EFFICIENCY OF THE OBJECT CODE GENERATED BY THE IBM ANSI COBOL COMPILERS. ITS OBJECT LEVEL ANALYSIS PROVIDES SAVINGS IN MAIN PROCESSOR TIME AND MAIN STORAGE REQUIREMENTS WHICH CANNOT BE OBTAINED AT THE SOURCE- LEVEL. SAVINGS OF UP TO 25% IN EXECUTION TIME AND 20-30% IN MEMORY REQUIREMENTS MAY BE ACHIEVED.

DOCUMENTATION: USER'S GUIDE
REFERENCES: [ASDS79], APPLIED SYSTEMS DESIGN SECTION, TRW DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE TOOLS SERIES, 790-100
[IDN80], JOHN D. DONAHOO AND DOROTHY SWARLING, "A REVIEW OF SOFTWARE MAINTENANCE TECHNOLOGY", RAD-TR-80-13, INTERIM REPORT, 800200

DEVELOPER: CEPX CORPORATION

INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG, RAD-TR-80-13, INTERIM REPORT

ACRONYM: OPTIMUS, TITLE: OPTIMUS
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, TRANSFORMATION, OPTIMIZATION, MACHINE OUTPUT, OBJECT CODE OUTPUT
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: ASSEMBLY

TOOL SIZE: CORE: 600=00S, 2K=00S/V8
COMPUTER (OTHER HARDWARE): IBM 360/370
OS (OTHER SOFTWARE): D0S, D0S/V8

TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO

RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): FOR LEASE

TOOL SUMMARY: OPTIMUS IS AN AUTOMATIC THROUGHPUT OPTIMIZER THAT PROVIDES USERS THE FLEXIBILITY TO RUN JOBS IN ANY PARTITION WITHOUT CONCERN FOR PARTITION LOCKOUTS OR SYSTEM DEGRADATION. OPTIMUS ANALYZES MULTI-PROGRAMMING ACTIVITY AT SPECIFIED INTERVALS AND DYNAMICALLY ADJUSTS PARTITION PRIORITIES, GIVING 1/O-BOUND JOBS HIGHER PRIORITY THAN CPU-BOUND JOBS. OPTIMUS DOES NOT REQUIRE ANY SUPERVISOR OR IBM SOFTWARE MODIFICATIONS. IT IS RELEASE-INDEPENDENT AND OPERATES ON PS and Power/V8 AS WELL AS THE VENDOR'S ASAP SPOOLING SYSTEM,
ACRONYM: PACE-C, TITLE: PRODUCT ASSURANCE CONFIDENCE Evaluator-COMPASS
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES/OTHER SUBJECT, CODE INPUT, COMPASS, TRANSFORMATION, INSTRUMENTATION, MACHINE OUTPUT, SOURCE CODE OUTPUT, COMPASS, USER OUTPUT, LISTINGS, DYNAMIC ANALYSIS, COVERAGE ANALYSIS,
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN, COMPASS
COMPUTER (OTHER HARDWARE): CDC 6600/7X00
DOCUMENTATION: USER'S GUIDE
REFERENCES: [ADS579], APPLIED SYSTEMS DESIGN SECTION, TRW DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS: CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE TOOLS SERIES, 790100
DEVELOPER: TRW, SEID SOFTWARE PRODUCT ASSURANCE
CONTACT: FRANK INGRASSIA, TRW, SEID SOFTWARE PRODUCT ASSURANCE, ONE SPACE PARK, REDONDO BEACH, CA, 90278, USA, 213-936-3100
INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG

ACRONYM: PBR, TITLE: PBR
CLASSIFICATION: REQUIREMENTS/DESIGN SPECIFICATION AND ANALYSIS
FEATURES: SUBJECT, TEXT INPUT, VHHL INPUT, DESIGN SPECIFICATION, LANGUAGE: TRANSFORMATION, FORMATTING, MACHINE OUTPUT, USER INPUT, USER-ORIENTED TEXT, REPORTS, TABLES, STATIC ANALYSIS, CROSS REFERENCE, STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: STRUCTURED FORTRAN
COMPUTER (OTHER HARDWARE): IBM 360/370, DEC SYSTEM-10/20, SEL 32, DEC PDP-11, CDC 6600/7X00, UNIVAC 11X
TOOL SUMMARY: THE PROGRAM DESIGN LANGUAGE (PDL) IS A TOOL TO AID IN DESIGNING AND DOCUMENTING A PROGRAM OR SYSTEM OF PROGRAMS. THE LANGUAGE IS SUPPORTED BY A PROCESSOR WHICH IS AVAILABLE FOR A WIDE VARIETY OF COMPUTERS. THE LANGUAGE IS SIMPLE TO LEARN AND SIMPLE TO USE. IT HAS PROVEN EFFECTIVE IN DEVELOPING COMPLETE DESIGNS PRIOR TO THE STARTING OF ANY PROGRAMMING ON A PROJECT. A DESIGN IN PDL IS WRITTEN IN STRUCTURED ENGLISH WHICH IS THEN INPUT TO THE PDL PROCESSOR TO PRODUCE THE PDL CODE. THE PROCESSOR CAN GENERATE A TABLE OF CONTENTS INFORMATION PLUS DESIGNS FOR PROCEDURES (CALLED "SEGMENTS" IN PDL). THE OUTPUT IS A WORKING DESIGN DOCUMENT CONSISTING OF A TABLE OF CONTENTS, A LISTING OF THE SEGMENTS AUTOMATICALLY FORMATTED, A DISPLAY OF THE

THE VERIFIER USED TO CHECK THEIR ADHERENCE; THIS HAS BEEN FOUND USEFUL IN THE PRODUCTION OF PORTABLE NUMERICAL SOFTWARE. SECONDLY, PROGRAMS WRITTEN IN AN EXTENDED DIALD MAY BE PASSED THROUGH THE VERIFIER SO THAT ALL DEVIATIONS FROM THE STANDARD MAY BE FOUND. THIS IS LIKELY TO BE OF VALUE WHEN LARGE PROGRAMS ARE BEING TRANSPORTED AND WILL MINIMIZE THE EFFORT REQUIRED TO ISOLATE NONCONFORMING CONSTRUCTIONS BETWEEN SYSTEMS. FINALLY, INFORMATION PRODUCED BY THE VERIFIER MAY BE USED AS DATA TO FURTHER SOFTWARE TOOLS. IN ADDITION, THE VERIFIER PRODUCES CROSS-REFERENCE TABLES FOR ALL VARIABLE NAMES, ALL FUNCTIONS, ALL LINE NUMBERS USED AS LABELS AND ALL LINES BY STATEMENT TYPE.
REFERENCES: [PANS77], "PROPOSED AMERICAN NATIONAL STANDARD FOR MINIMAL BASIC", PROPOSED AMERICAN NATIONAL STANDARD, 770000
[RYDE74], B. G. RYDER, "THE PPORT VERIFIER", SOFTWARE PRACTICE AND EXPERIENCE, 4, 359-377, 740000
[TEN79], A.C., GENZ, T.R., HOPKINS, "PORTABLE NUMERICAL SOFTWARE FOR MICROCOMPUTERS", PROCEEDINGS OF THE NT9 CONF, AT LIVERPOOL UNIV, 790000
[TEN878], A.R. LAWRENCE, "SCRUB", SYSTEMATICALLY CLEAN AND RENUMBER USERS BASIC, 780000
[HOPK60], T., R., "PBR = A VERIFIER FOR BASIC", SOFTWARE=PRACTICE AND EXPERIENCE, 801000
DEVELOPER: T.R. HOPKINS
CONTACT: T. R. HOPKINS, UNIVERSITY OF KENT AT CANTERBURY, COMPUTING LABORATORY, CANTERBURY, KENT, CT27NF, UK,
INFORMATION SOURCE: SOF-PRAC EXP
PROCEDURE CALLING TREE, AND A CROSS-REFERENCE OF THE
PROCEDURE CALLS, IT IS FORMATTED TO BE EASILY
PHOTO-REDUCED FOR LATER INCLUSION IN A PROJECT WORKBOOK.
THE OUTPUT OF THE PROCESSOR CAN COMPLETELY REPLACE
FLOWCHARTS SUBMITTED.

DOCUMENTATION: PROGRAM DESCRIPTION

REFERENCES: [ASDS79], APPLIED SYSTEMS DESIGN SECTION, TRW
DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS:
CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE
TOOLS SERIES, 790100

DEVELOPER: CAIN, FABER GORDON, INC.

INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG

ACRONYM: PDS, TITLE: PROGRAM DEVELOPMENT SYSTEM
CLASSIFICATION: REQUIREMENTS/DESIGN SPECIFICATION AND
ANALYSIS

FEATURES: SUBJECT, TEXT INPUT, CODE INPUT, FORTRAN,
STRUCTURED FORTRAN, VHLL INPUT, DESIGN LANGUAGE,
TRANSFORMATION, FORMATTING, MACHINE OUTPUT, SOURCE CODE
OUTPUT, FORTRAN, USER OUTPUT, LISTINGS, STATIC ANALYSIS,
MANAGEMENT, TEST DATA MANAGEMENT, FILES MANAGEMENT, DYNAMIC
ANALYSIS, COVERAGE ANALYSIS

STAGE OF DEVELOPMENT: IMPLEMENTED

IMPLEMENTATION LANGUAGE: FORTRAN

COMPUTER (OTHER HARDWARE): IBM 360/370, CDC CYBER, UNIVAC

11X

TOOL SUMMARY: THE PROGRAM DEVELOPMENT SYSTEM IS A MULTIPLE
PURPOSE SOFTWARE PACKAGE DESIGNED TO PROVIDE A SINGLE TOOL
FOR THE STRUCTURED DESIGN AND DEVELOPMENT OF FORTRAN
COMPUTER PROGRAMS. IT IS A PRE-PROCESSOR EXTENDING THE
FORTRAN LANGUAGE TO INCLUDE STRUCTURED CONSTRUCTS AND
PROVIDING A MEANS FOR A CONVERSION OF SUCH STRUCTURED
FORTRAN TO STANDARD FORTRAN. IT IS A FILE CONTROL SYSTEM,
DESIGNED TO REDUCE DEVELOPMENT COSTS AND INCREASE FILE
CONTROL BY AUTOMATICALLY LISTING AND PROCESSING ONLY THOSE
Routines CHANGED SINCE THE PREVIOUS RUN, WHILE MAINTAINING A
COMPLETE SYSTEM FILE. IT IS A TEST TOOL, CAPABLE OF
CREATING AN INSTRUMENTED, COMPILABLE VERSION OF AN ORIGINAL
STRUCTURED FORTRAN PROGRAM FOR THE COLLECTION AND DISPLAY
OF TEST EXECUTION STATISTICS. LISTINGS PROVIDED BY THE
SYSTEM ARE INTENDED TO SHOW THE LOGICAL STRUCTURE OF THE
CODE, AND ARE PRINTED WITH DATE AND FILENAME INFORMATION.

DOCUMENTATION: IN PROGRESS

REFERENCES: [ASDS79], APPLIED SYSTEMS DESIGN SECTION, TRW
DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS:
CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE
TOOLS SERIES, 790100

DEVELOPER: TRW SOFTWARE TECHNOLOGY DEPARTMENT

CONTACT: R. L. MAITLEN, TRW, SOFTWARE TECHNOLOGY
DEPARTMENT, ONE SPACE PARK, REDONDO BEACH, CA, 90278, USA,
213-535-3480

INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG

ACRONYM: PDS, TITLE: PROGRAM DEVELOPMENT SYSTEM
CLASSIFICATION: REQUIREMENTS/DESIGN SPECIFICATION AND
ANALYSIS

FEATURES: SUBJECT, TEXT INPUT, CODE INPUT, FORTRAN,
STRUCTURED FORTRAN, VHLL INPUT, DESIGN LANGUAGE,
TRANSFORMATION, FORMATTING, MACHINE OUTPUT, SOURCE CODE
OUTPUT, FORTRAN, USER OUTPUT, LISTINGS, STATIC ANALYSIS,
MANAGEMENT, TEST DATA MANAGEMENT, FILES MANAGEMENT, DYNAMIC
ANALYSIS, COVERAGE ANALYSIS

STAGE OF DEVELOPMENT: IMPLEMENTED

IMPLEMENTATION LANGUAGE: FORTRAN

COMPUTER (OTHER HARDWARE): IBM 360/370, CDC CYBER, UNIVAC

11X

TOOL SUMMARY: THE PROGRAM DEVELOPMENT SYSTEM IS A MULTIPLE
PURPOSE SOFTWARE PACKAGE DESIGNED TO PROVIDE A SINGLE TOOL
FOR THE STRUCTURED DESIGN AND DEVELOPMENT OF FORTRAN
COMPUTER PROGRAMS. IT IS A PRE-PROCESSOR EXTENDING THE
FORTRAN LANGUAGE TO INCLUDE STRUCTURED CONSTRUCTS AND
PROVIDING A MEANS FOR A CONVERSION OF SUCH STRUCTURED
FORTRAN TO STANDARD FORTRAN. IT IS A FILE CONTROL SYSTEM,
DESIGNED TO REDUCE DEVELOPMENT COSTS AND INCREASE FILE
CONTROL BY AUTOMATICALLY LISTING AND PROCESSING ONLY THOSE
Routines CHANGED SINCE THE PREVIOUS RUN, WHILE MAINTAINING A
COMPLETE SYSTEM FILE. IT IS A TEST TOOL, CAPABLE OF
CREATING AN INSTRUMENTED, COMPILABLE VERSION OF AN ORIGINAL
STRUCTURED FORTRAN PROGRAM FOR THE COLLECTION AND DISPLAY
OF TEST EXECUTION STATISTICS. LISTINGS PROVIDED BY THE
SYSTEM ARE INTENDED TO SHOW THE LOGICAL STRUCTURE OF THE
CODE, AND ARE PRINTED WITH DATE AND FILENAME INFORMATION.

DOCUMENTATION: IN PROGRESS

REFERENCES: [ASDS79], APPLIED SYSTEMS DESIGN SECTION, TRW
DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS:
CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE
TOOLS SERIES, 790100

DEVELOPER: TRW SOFTWARE TECHNOLOGY DEPARTMENT

CONTACT: R. L. MAITLEN, TRW, SOFTWARE TECHNOLOGY
DEPARTMENT, ONE SPACE PARK, REDONDO BEACH, CA, 90278, USA,
213-535-3480

INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG
POSS

POSSIBLE EVENT DATES USING WHATEVER INFORMATION IS AVAILABLE, INCLUDING HOLIDAYS IN CALCULATIONS IF DESIRED. ADDITIONALLY, KEY PROJECT DATES MAY BE ENTERED SEPARATELY WITH AN IDENTIFIER (IE: COD, FCA). USE OF THIS IDENTIFIER TO FIX OTHER PROJECT EVENTS ALLOWS A FLUXXATION OF KEY DATES WITH A MINIMAL EFFECT ON THE DATA BASE. OUTPUT FROM POSS IS PRODUCED ON THE VERSATEC PLOTTER, AND MAY BE SORTED IN A VARIETY OF WAYS. REPORT OUTPUT INCLUDES EVENT TITLES, DELIVERY DATES AND CONTROLLING DATES.

DOCUMENTATION: USER'S MANUAL

REFERENCES: (ADS 879), APPLIED SYSTEMS DESIGN SECTION, TRW DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS: CATALOG AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE TOOLS SERIES, 790100

DEVELOPER: TRW, SOFTWARE TECHNOLOGY DEPARTMENT

CONTACT: R. L. MAILEN, TRW, SOFTWARE TECHNOLOGY DEPARTMENT, ONE SPACE PARK, REDONDO BEACH, CA, 90278, USA, 213-535-3480

INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG

ACRONYM: PERCAM, TITLE: PERFORMANCE AND CONFIGURATION ANALYSIS MODEL

CLASSIFICATION: REQUIREMENTS/DESIGN SPECIFICATION AND ANALYSIS

FEATURES: SUBJECT, DATA INPUT, USER OUTPUT, TABLES, DYNAMIC ANALYSIS, SIMULATION

STAGE OF DEVELOPMENT: IMPLEMENTED

IMPLEMENTATION LANGUAGE: FORTRAN

COMPANY (OTHER HARDWARE): CDC 6600/7700

TOOL SUMMARY: PERCAM IS AN ANALYSIS METHODOLOGY WHICH PROVIDES FOR THE REPRESENTATION OF SYSTEMS IN THE FORM OF EVENT LOGIC TREES (ELTS), AND WHICH SUBSEQUENTLY AUTOMATES THE CONSTRUCTION OF A SIMULATION WHICH OPERATIONAL SEQUENCING LOGIC DUPLICATES THE DESIGNED ELTS. AN ELT IS THE GRAPHIC REPRESENTATION OF A SYSTEM, INCLUDING THOSE FUNCTIONS AND DECISIONS INCLUDED IN THE SYSTEM. BY FORMING A COMPUTER LIBRARY CONTAINING DESCRIPTIONS OF THE PRIMITIVE COMPONENTS OF THIS GRAPHIC REPRESENTATION, A COMPUTER MODEL CAN BE CONSTRUCTED WITH THE LIBRARY COMPONENTS TO SIMULATE THE WORKINGS OF THE PROPOSED SYSTEM ON THE COMPUTER. IN THIS MANNER, PERCAM PROVIDES A TECHNIQUE VALUABLE IN THE DEFINITION AND DEVELOPMENT PHASES OF A PROJECT. A PROPOSED SYSTEM MAY BE DESIGNED GRAPHICALLY, AND A LIBRARY DESIGNED OR AN EXISTING LIBRARY UTILIZED TO REPRESENT THE SYSTEM'S COMPONENTS. SIMULATIONS OF THE TARGET SYSTEM CAN THEN BE CREATED AND THE SYSTEM REVISED UNTIL VERIFIED THROUGH ALTERATIONS IN THE LIBRARY AND ELT DESIGN.

DOCUMENTATION: USER'S MANUAL, PROGRAM DESCRIPTION

REFERENCES: (ADS 879), APPLIED SYSTEMS DESIGN SECTION, TRW DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS: CATALOG AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE TOOLS SERIES, 790100

PERCAM

DEVELOPER: TRW SOFTWARE TOOLS CATALOG

CONTACT: J. FIELDS, TRW SOFTWARE TOOLS CATALOG, ONE SPACE PARK, REDONDO BEACH, CA, 90278, USA, 213-535-2045

INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG

ACRONYM: PERLUTE, TITLE: PERLUTE

CLASSIFICATION: PROGRAM CONSTRUCTION AND GENERATION

FEATURES: SUBJECT, VHDL INPUT, DESCRIPTION LANGUAGE, TRANSFORMATION, SYNTHESIS, MACHINE OUTPUT, SOURCE CODE OUTPUT

STAGE OF DEVELOPMENT: IMPLEMENTED

COMPONENT (OTHER HARDWARE): CII-HB

CATEGORIES SOFTWARE: MULTICS, SIRIS


DOCUMENTATION: TECHNICAL REPORT


CONTACT: M. MAZAUD, INRIA, DOMAINE DE VOLUCOUR-ROCOUENCOURT, BP 105, LE CHESSAY, CEDEX, 78153, FRANCE, (3) 954.90.20

M. C. GAUDEL, INRIA, DOMAINE DE VOLUCOUR-ROCOUENCOURT, BP 105, LE CHESSAY, CEDEX, 78153, FRANCE, (3) 954.90.20

INFORMATION SOURCE: ADI/CNRS CATALOGUE 1980

ACRONYM: PET, TITLE: PROGRAM EVALUATOR AND TESTER

CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING

FEATURES: DATA RANGE ANALYSIS, SUBJECT, CODE INPUT, FORTRAN, TRANSFORMATION, INSTRUMENTATION, MACHINE OUTPUT, SOURCE CODE OUTPUT, FORTRAN, USER OUTPUT, DIAGNOSTICS, TABLES, LISTINGS, STATIC ANALYSIS, STATISTICAL ANALYSIS, PROFILE GENERATION, AUDITING, DYNAMIC ANALYSIS, COVERAGE ANALYSIS

STAGE OF DEVELOPMENT: IMPLEMENTED

IMPLEMENTATION LANGUAGE: FORTRAN

TOOL PORTABLE: YES

TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO

RESTRICTIONS: (COPYRIGHTS, LICENSES, ETC.): FOR SALE
TOOL SUPPORT: YES, TOOL SUPPORT: MCDONNELL-DOUGLAS CORPORATION


DOCUMENTATION: USER MANUAL, SYSTEM DESCRIPTION

REFERENCE: [ADS879], APPLIED SYSTEMS DESIGN SECTION, TRW DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS CATALOG AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE TOOLS SERIES, 790100

DEVELOPER: MCDONNELL-DOUGLAS CORPORATION

CONTACT: J. B. CURCHNEAL. MCDONNELL-DOUGLAS CORPORATION, 5301 BOLSA AVENUE, HUNTINGTON BCH, CA, 92647, USA, 714-896-4155

INFORMATION SOURCE: AIAA SURVEY OF SOFT DEV TOOLS, TRW SOFTWARE TOOLS CATALOG

ACRONYMY: PFORT, TITLE: PFORT CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE

FEATURES: SUBJECT, CODE INPUT, FORTRAN, USER OUTPUT, COMPLETE LISTS OF PROGRAM STATISTICS, LISTINGS, STATIC ANALYSIS, AUDITING, TCP/IP, SOFTWARE MANAGEMENT

IMPLEMENTATION LANGUAGE: FORTRAN

TOOL AVAILABLE: YES, PUBLIC DOMAIN: YES

TOOL SUMMARY: THE PFORT VERIFIER, DEVELOPED AND DISTRIBUTED BY THE BELL RESEARCH LABORATORIES, MURRAY HILL, NEW JERSEY, ANALYZES A FORTRAN PROGRAM AND NOTES THE OCCURRENCES OF PROGRAMMING PRACTICES THAT ARE LIKELY TO BE IMPEDIMENTS TO PORTABILITY.

DOCUMENTATION: USER'S MANUAL, PFORT REFERENCE: [DON680], JOHN O. DONAHOO AND DOROTHY SWAERINGER, "A REVIEW OF SOFTWARE MAINTENANCE TECHNOLOGY", RADC-TR-80-13, INTERIM REPORT, 600200

DEVELOPER: JEF PROPULSION LABORATORY

CONTACT: L.R.M. B. BIRN, BELL LABORATORIES, 600 MOUNTAIN AVENUE, MURRAY HILL, NJ, 07974, USA, 201-582-4300

INFORMATION SOURCE: AIAA SURVEY OF SOFT DEV TOOLS, RADC-TR-80-13, INTERIM REPORT

ACRONYMY: PFS, TITLE: PROJECT FORECASTING SYSTEM CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE

FEATURES: SUBJECT, DATA INPUT, USER OUTPUT, TABLES, STATIC ANALYSIS, MANAGEMENT, PROJECT MANAGEMENT, TCP/IP

STAGE OF DEVELOPMENT: IMPLEMENTED

IMPLEMENTATION LANGUAGE: FORTRAN

COMPUTER (OTHER HARDWARE): CDC 6600/7X00

TOOL SUMMARY: THE PROJECT FORECAST SYSTEM IS A TIMESHARE, COMPUTER BASED SYSTEM WHICH PRICES AND FORMATS COST DATA AND TABULATES AND PLOTS INDEXED WORK BREAKDOWN STRUCTURE CHARTS. THE SYSTEM GREATLY REDUCES THE EFFORT REQUIRED TO SPREAD AND PRICE COST DATA DURING PROJECT BUDGETING AND ESTIMATE/COMPLETE PREPARATIONS. PFS ALLOWS PROJECT PLANNING AND CONTROL PERSONNEL TO RAPIDLY ITERATE THEIR FORECASTS WHILE UPDATING THEIR DATA THROUGH REMOTE TERMINALS. INPUT FOR THE SYSTEM, WHICH MAY COME FROM TERMINAL OR TRW PROJECT REPORTING AND REVIEW (PR-R) SYSTEM, CONSISTS OF COST DATA (JOB NUMBER, COST ELEMENT, COST CENTER, AND COSTS IN DOLLARS, MANHOURS OR MANMONTHS), RATE DATA (BY BURDEN POOL, COST CENTER OR TOTAL), THE ACCOUNTING CALENDAR, AND ACTUAL COSTS (BY JOB NUMBER), OUTPUT INCLUDES A REPORT OF DETAIL COST, SUMMARY COST, FORECAST EXPENDITURE PLAN AND MANPOWER EXPENDITURE PLAN REPORTS AS WELL AS INDEXED WORK BREAKDOWN STRUCTURE CHARTS, THE COST DETAIL REPORTS, ET C., ARE VALUABLE PROJECT/SUBPROJECT OFFICE TOOLS.

DOCUMENTATION: USER'S MANUAL, USER'S GUIDE

REFERENCE: [ADS879], APPLIED SYSTEMS DESIGN SECTION, TRW DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS CATALOG AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE TOOLS SERIES, 790100

DEVELOPER: TRW MGMT SYS PLANNING ANAL

CONTACT: KEN SCHULTZ, TRW MGMT SYS PLANNING ANAL, ONE SPACE PARK, REDONDO BEACH, CA, 90278, USA, 213-535-2326

INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG

ACRONYMY: PIDGIN-FASP, TITLE: PIDGIN-FASP CLASSIFICATION: REQUIREMENTS/DESIGN SPECIFICATION AND ANALYSIS

FEATURES: SUBJECT, VHDL INPUT, DESIGN SPECIFICATION, USER OUTPUT, GRAPHICS, LISTINGS, STATIC ANALYSIS, SCANNING, TCP/IP

STAGE OF DEVELOPMENT: IMPLEMENTED

TOOL PORTABLE: NO

COMPUTER (OTHER HARDWARE): CDC CYBER, CDC 6600/7X00

OS (OTHER SOFTWARE): KRONOS
TOOL AVAILABLE: YES, PUBLIC DOMAIN; YES
TOOL SUPPORTED: YES, TOOL SUPPORT: NAVAL AIR DEVELOPMENT CENTER
TOOL SUMMARY: PIDGIN IS A DESIGN LANGUAGE WITH ASSOCIATED PROCESSOR FOR SPECIFYING AND DOCUMENTING THE SOFTWARE DESIGN OF A PROGRAM. IT IS USED TO SPECIFY SOFTWARE INPUTS, OUTPUTS, AND DOCUMENTS USED TO EXPAND DOCUMENTATION AND PROVIDE TASK-SPECIFIC AND DOCUMENTATION FOR OTHER SOFTWARE. THE LANGUAGE IS MACHINE-DEPENDENT AND INDEPENDENT OF THE PROGRAMMING LANGUAGE IN WHICH THE PROGRAM IS WRITTEN. PIDGIN CAN BE USED TO CREATE A MACHINE-DEPENDENT DESCRIPTION OF THE PROGRAM DESIGN, OR TO DOCUMENT THE PROGRAM DESIGN USING A MACHINE-DEPENDENT LANGUAGE. PIDGIN IS DESIGNED TO BE USED IN CONJUNCTION WITH A SOFT DEVS TOOLS.
DOCUMENTATION: USER'S MANUAL, MAINTENANCE MANUAL
REFERENCES: [REF 81], D. J. REIFER AND H. A. MONTGOMERY, "SEATECS SOFTWARE TOOLS SURVEY", RCI-TR-008, REIFER CONSULTANTS, INC., 610330
DEVELOPER: NAVAL AER DEVELOPMENT CENTER
CONTACT: J. BERGER, NAVAL AIR DEVELOPMENT CENTER, ADVAN, SOFT, TECH, DIV, CODE 503, WARMINSTER, PA, 18974, USA, 215-444-3145
INFORMATION SOURCE: NOSE SEATECS TOOLS SURVEY
ACRONYM: PMCS; TITLE: PROJECT MANAGEMENT AND CONTROL SYST EM
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE
FEATURES: SUBJECT, DATA INPUT, USER OUTPUT, TABLES, STATIC ANALYSIS, MANAGEMENT, PROJECT MANAGEMENT, STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN, COMPASS
 TOOL SIZE: CORE: 165K
COMPUTER (OTHER HARDWARE): CDC CYBER, CDC 6X00/7X00
OS (OTHER SOFTWARE): CDC 6600 SECRE, KRONOS
TOOL AVAILABLE: YES, PUBLIC DOMIAN; NO
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): FOR LEASE, FOR SALE
TOOL SUMMARY: PMCS (PROJECT MANAGEMENT AND CONTROL SYSTEM) IS A CRITICAL PATH PLANNING SYSTEM USED FOR SCHEDULING AND CONTROLLING LARGE COMPLEX PROJECTS. IT ALLOWS THE USER TO QUICKLY IDENTIFY CURRENT AND POTENTIAL PROBLEM AREAS WITHIN A PROJECT SO THAT CORRECTIVE ACTION CAN BE TAKEN. SOME OF THE FEATURES OF THE PACKAGE ARE: UP TO 15,000 ACTIVITIES AND 12,000 EVENTS CAN BE CONTROLLED, MULTIPLE STAND-ALONE START AND END EVENTS WITHIN A NETWORK OR SUBNETWORK CAN BE IDENTIFIED AND SCHEDULED, STANDARD FUNCTIONS, UP TO NINE-MILESTONE LEVELS MAY BE IDENTIFIED, BOTH ARROW DIAGRAM AND PRECEDENCE DIAGRAM NETWORKS ARE INCLUDED, AND RANDOM ASSIGNMENT OF ALPHANUMERIC NODE NUMBER OR NAME IS ALLOWED. SOME OF THE REPORTS AVAILABLE FROM THIS SYSTEM INCLUDE: A DIAGNOSTIC PRINTOUT FOR ERRORS AND INCONSISTENCIES, FLEXIBLE 10-YEAR CALENDAR WITH TWO CALENDAR REPORT FORMATS, AND A REPORT ON TIME PROGRESS AS A PERCENTAGE OF COMPLETENESS, UP TO THREE ESTIMATE (DURATION) FACTORS MAY BE SPECIFIED.
DOCUMENTATION: USER'S MANUAL, MAINTENANCE MANUAL
DEVELOPER: MULTIPLE ACCESS COMPUTER GROUP
INFORMATION SOURCE: AIAA SURVEY OF SOFT DEVS TOOLS
ACRONYM: PMS IV; TITLE: PROJECT MANAGEMENT SYSTEM IV
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE
FEATURES: SUBJECT, DATA INPUT, USER OUTPUT, TABLES, STATIC ANALYSIS, MANAGEMENT, PROJECT MANAGEMENT, STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: ASSEMBLY
 TOOL SIZE: CORE: 44K TO 75K
COMPUTER (OTHER HARDWARE): IBM 360/370
OS (OTHER SOFTWARE): OS, OS/V Q
TOOL AVAILABLE: YES, PUBLIC DOMIAN; NO
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): FOR SALE
TOOL SUMMARY: PMS IV (5734-XP4) IS A MANAGEMENT TOOL DESIGNED TO AID IN THE PLANNING AND CONTROL OF RESOURCES. THE SYSTEM PROVIDES CRITICAL PATH AND GENERAL COST ANALYSES, PERT AND PERT COST CAPABILITIES, PRECEDENCE, AND PRECEDENCE/COST CAPABILITIES, AS WELL AS RESOURCE ALLOCATIONS. A FLEXIBLE ADD-ON AND SUBSTITUTION CAPABILITY ALLOWS FOR A GROWING LIBRARY OF EXECUTABLE SUBROUTINES AND PERMITS THE USER TO TAILOR THE PROGRAM TO THE SPECIFIC REQUIREMENTS OF HIS INSTALLATION. PMS IV OFFERS THREE OPTIONAL FEATURES: AN EXTENDED NETWORK PROCESSOR, A COST PROCESSOR FOR COST ACCOUNTING, AND A RESOURCE ALLOCATION PROCESSOR FOR SCHEDULE ADJUSTMENT BASED UPON RESOURCE AVAILABILITY.
DOCUMENTATION: USER'S MANUAL, MAINTENANCE MANUAL, DEVELOPMENT SPECIFICATION, TEST PLAN
DEVELOPER: IBM
INFORMATION SOURCE: AIAA SURVEY OF SOFT DEVS TOOLS
ACRONYM: POD; TITLE: PERFORMANCE ORIENTED DESIGN
CLASSIFICATION: SOFTWARE MODELING AND SIMULATION
FEATURES: SUBJECT, VHDL INPUT, SYSTEM SPECIFICATION, USER OUTPUT, GRAPHICS, TABLES, DYNAMIC ANALYSIS, SIMULATION, TUNING, STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN
 TOOL PORTABLE: YES
TOOL AVAILABLE: YES, PUBLIC DOMIAN; NO
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): DETAILS FROM BGS REPORT SUPPORTED: YES
TOOL SUMMARY: THE BASIC GOALS OF PERFORMANCE ORIENTED DESIGN MAY BE SUMMARIZED AS FOLLOWS: (A) TO PROVIDE BETTER MANAGEMENT CONTROL DURING SYSTEM DESIGN AND IMPLEMENTATION
BY ENABLING PROJECT MANAGERS TO DEFINE AND VALIDATE PERFORMANCE OBJECTIVES AT EACH STAGE OF THE DEVELOPMENT PROCESS, (B) TO REDUCE TOTAL DEVELOPMENT TIME AND COST BY PROVIDING DESIGNERS AND IMPLEMENTORS WITH EARLY WARNING OF UPCOMING PERFORMANCE PROBLEMS AND BY FOCUSING ATTENTION ON CRITICAL PROBLEM AREAS, (C) TO REDUCE MAINTENANCE COSTS BY PROVIDING MAINTENANCE PERSONNEL WITH PERFORMANCE RELATED INFORMATION WHICH CAN BE CONSULTED WHEN PERFORMANCE PROBLEMS ARISE AFTER THE SYSTEM IS DEPLOYED, (D) TO REDUCE DEVELOPMENT TIME AND COST FOR FUTURE SYSTEMS BY PROVIDING DESIGNERS WITH PERFORMANCE RELATED INFORMATION ABOUT EXISTING SYSTEMS WHICH WILL BE OF DIRECT VALUE IN FUTURE DESIGN EFFORTS.

DOCUMENTATION: USERS MANUAL
DEVELOPER: BGS SYSTEMS
CONTACT: BGS SYSTEMS, INC., 1 UNIVERSITY OFFICE PARK, WALTHAM, MA. 02254, USA. 617-891-0000
INFORMATION SOURCE: TOOL FAIR

ACRONYMS: PPE, TITLE: PROBLEM PROGRAM EVALUATOR
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, USER OUTPUT, LISTINGS, DYNAMIC ANALYSIS, RESOURCE UTILIZATION, TIMING,
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: COBOL, ASSEMBLY
COMPUTER (OTHER HARDWARE): IBM 360/370

TOOL SUMMARY: PPE enables a user to identify time consuming areas of program code or other program characteristics that result in inefficient running times. PPE's approach is to use an in-core resident monitor, the extractor program, which samples the activity of the program and outputs this activity data to an extractor data set. The extractor data set can be recorded on tape or disk. The user's program runs normally, and no recompilation is necessary. The report program, analyzer, produces an easy-to-understand code activity report, based on the extractor data set. The analyzer can be run as the second phase of a job with the extractor or at a later time as a stand alone job.

DOCUMENTATION: PROGRAM DESCRIPTION
REFERENCES: [ASD079], APPLIED SYSTEMS DESIGN SECTION, TRW DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE TOOLS SERIES, 790100
DEVELOPER: TRW OPERATING SYSTEM DEPARTMENT
CONTACT: O. MARTIN ROBINSON, TRW OPERATING SYSTEMS DEPARTMENT.
CONTACT: ONE SPACE PARK, REDONDO BEACH, CA. 90278, USA. 213-535-0862
INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG

ACRONYMS: PPP, TITLE: PROJECT PRE-PLANNER
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE

FEATURES: SUBJECT, DATA INPUT, USER OUTPUT, GRAPHICS, STATIC ANALYSIS, MANAGEMENT, PROJECT MANAGEMENT,
STAGE OF DEVELOPMENT: DESIGN
IMPLEMENTATION LANGUAGE: PL/I
TOOL SIZE: CORE: 20K
COMPUTER (OTHER HARDWARE): IBM 360/370
OS (OTHER SOFTWARE): OS/MVS
TOOL AVAILABLE: NO

TOOL SUMMARY: THIS PROGRAM IS DESIGNED TO INTERACTIVELY CREATE, MODIFY, AND PLOT SOFTWARE DEVELOPMENT PROGRESS STRUCTURES USING A 4014 OR 4015 TESKONIX STORAGE TERMINAL AS A FRONT-END TO PERC OR CPM. A 12" BY 9" WINDOW CAN BE VIEWED AS A SECTION OF A 36" BY 30" PLOTTING SURFACE. THE PLOTTING PROGRAM CAN CREATE A LARGE SIZE CALCOMP PLOT IN COLOR OR BLACK AND WHITE, TOGETHER WITH ANY NUMBER OF SMALL PLOTS. THE Operation OF THE PROGRAM IS CONDUCTED BY TYPING A UNIQUE KEYSTROKE FOR EACH OPERATION. THE PROGRAM CAN GENERATE FIGURES, LINES AND TEXT, ONLY ONE FRAME AT A TIME CAN BE CREATED, SO THAT A DATA SET MUST BE DEFINED AND SAVED FOR EACH DRAWING. THE DATA SET WILL SERVE AS THE INPUT TO A PROGRAM THAT WILL CREATE A CALCOMP 30" WIDE PLOT. THE END PRODUCT IS THEN USED AS AN INPUT INTO STANDARD PERC OR CPM SOFTWARE.

DOCUMENTATION: USER'S MANUAL
DEVELOPER: APPLIED PHYSICS LABORATORY
CONTACT: JOHN H. MANLEY, APPLIED PHYSICS LABORATORY, 1350 HOPKINS ROAD, LAUREL, MD. 20707, USA. 301-953-730
INFORMATION SOURCE: AIAA SURVEY OF SOFT DEV TOOLS

ACRONYMS: PDMR, TITLE: PREFACE HEADER GENERATOR
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, FORTRAN, USER OUTPUT, DIAGNOSTICS, LISTINGS, STATIC ANALYSIS, DATA FLOW ANALYSIS, INTERFACE ANALYSIS, I/O SPECIFICATION ANALYSIS,
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN
COMPUTER (OTHER HARDWARE): MODCOMP
PUBLIC DOMAIN: YES

TOOL SUMMARY: THE HEADER GENERATOR TOOL PRODUCEA A COMPLETE PICTURE OF THE FLOW OF DATA IN AND OUT OF A PROGRAM. THE RESULTING PICTURE CAN BE USED TO VERIFY THAT EACH PROGRAM OF A SYSTEM MEETS ITS INTERFACE SPECIFICATION, ENSURING THAT EACH SUBPROGRAM USES ONLY ITS SPECIFIED INPUTS AND PRODUCES ONLY ITS SPECIFIED OUTPUTS. THE ANALYST CAN USE THE PROGRAM GENERATED OUTPUT DURING THE CODING PHASE OF SYSTEM DEVELOPMENT TO VERIFY THAT A PROGRAM IS USING ONLY THOSE INPUTS AND OUTPUTS THAT HAVE BEEN SPECIFIED IN THE SYSTEM DESIGN. DURING ALL PHASES OF THE SYSTEM'S EXISTENCE, THE OUTPUT CAN BE USED TO VERIFY THAT THE Pream Block Contained In A PROGRAM'S Source CODE HAS BEEN CREATED AND MAINTAINED CORRECTLY. OR, IF NO PREFACE BLOCK EXISTS, THEN THE PROGRAM PROVIDES A STRONG FRAMEWORK FROM WHICH TO CREATE ONE.
DOCUMENTATION: TECHNICAL PAPER
DEVELOPERS: TELEDYNE BROWN ENGINEERING
CONTACT: R. E. ALGER, TELEDYNE BROWN ENGINEERING, CUNNINGS
RESEARCH PARK, HUNTSVILLE, AL, 35807, USA, 205-552-1257
INFORMATION SOURCE: AIAA SURVEY OF SOFT DEV TOOLS

ACRONYM: PROG COMP ANAL TITLE: PROGRAM COMPARISON ANALYSIS
CLASSIFICATION: SOFTWARE, MANAGEMENT, CONTROL, AND MAINTENANCE
FEATURES: SUBJECT, CODE INPUT, OBJECT CODE INPUT, USER
OUTPUT, LISTINGS, STATIC ANALYSIS, COMPARISON, MANAGEMENT,
CONFIGURATION MANAGEMENT,
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN
COMPUTER (OTHER HARDWARE): MDCCOMP
PUBLIC DOMAIN: YES

TOOL SUMMARY: THE PROGRAM COMPARISON ANALYSIS TOOL PROVIDES
THE MEANS TO COMPARE TWO SOFTWARE SYSTEM VERSIONS, AT
EITHER THE SOURCE OR OBJECT LEVEL, TO DETERMINE IF THEY ARE
IDENTICAL. IT SERVES PRIMARILY AS A TOOL FOR CONFIGURATION
MANAGEMENT, DURING THE CODING AND TEST PHASES OF
DEVELOPMENT, THE TOOL MAY BE USED TO VERIFY THAT EACH
DELIVERY AND REVISION OF SOFTWARE RECEIVED, WHETHER IN
SOURCE OR OBJECT FORM, CONTAIN ONLY THOSE MODIFICATIONS
IDENTIFIED, IF THE SYSTEM IS DEPLOYED TO MULTIPLE SITES,
THIS TOOL CAN BE USED DURING THE MAINTENANCE PHASE OF THE
SYSTEM'S EXISTENCE TO VERIFY THAT A PARTICULAR OBJECT
VERSION OF THE SYSTEM IS INSTALLED AT A GIVEN SITE.

DOCUMENTATION: TECHNICAL PAPER
DEVELOPER: TELEDYNE BROWN ENGINEERING
CONTACT: R. E. ALGER, TELEDYNE BROWN ENGINEERING, CUNNINGS
RESEARCH PARK, HUNTSVILLE, AL, 35807, USA, 205-552-1257
INFORMATION SOURCE: AIAA SURVEY OF SOFT DEV TOOLS

ACRONYM: PROGLOOK TITLE: PROGLOOK
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, USER OUTPUT, GRAPHICS,
ACTIVITY DIAGRAMS, TABLES, STATIC ANALYSIS, SCANNING,
DYNAMIC ANALYSIS, TUNING, TIMING,
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN IV, ASSEMBLY
COMPUTER (OTHER HARDWARE): IBM 360/370
OS (OTHER SOFTWARE): OS, OS/V3
TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO

RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): FOR SALE

TOOL SUMMARY: PROGLOOK CONSISTS OF TWO PROGRAMS, PROGTIME
AND PROGLOOK, WHICH ENABLE THE USER TO MAKE DETAILED
MEASUREMENTS OF HIS PROGRAM WHILE IT IS RUNNING IT CAN BE
USED TO MEASURE ANY USER PROGRAM THAT CAN BE RUN UNDER
OS/MVT, OS/MFT, OR VS2 REL 1.46. WITH IT, THE USER CAN
ASSIGN WHAT ACTION HE MUST TAKE IN ORDER TO IMPROVE THE
PERFORMANCE OF THE PROGRAM. PROGTIME USES A CONTROL CLOCK
TO SNAP A PICTURE OF ANY PROGRAM RUNNING UNDER IT AND
RECORDS THIS INFORMATION IN A SPECIALLY FORMATTED DATA SET.

DOCUMENTATION: USER'S MANUAL
DEVELOPER: COSMIC
INFORMATION SOURCE: AIAA SURVEY OF SOFT DEV TOOLS

ACRONYM: PROGRAM GENERAT TITLE: PROGRAM GENERATOR
CLASSIFICATION: PROGRAM CONSTRUCTION AND GENERATION
FEATURES: SUBJECT, VHLL INPUT, PROGRAM SPECIFICATION,
TRANSFORMATION, SYNTHESIS, MACHINE OUTPUT, SOURCE CODE
OUTPUT,
STAGE OF DEVELOPMENT: IMPLEMENTED, DATE (YYYYMMDD): 800315
IMPLEMENTATION LANGUAGE: BASIC HP 250 EXTENDED
COMPUTER (OTHER HARDWARE): HP 85 (160K SYS MEMORY, 64K USER
MEMORY)

TOOL SUMMARY: GENERATES SOURCE CODE FOR PROGRAMS TO BE USED
IN SMALL BUSINESS SYSTEMS. THE SYSTEM IS EXPECTED TO
INCREASE OVER-ALL PRODUCTION OF SOFTWARE BY A FACTOR OF
4,
DEVELOPER: LAKESIDE
CONTACT: BILL AUSTIN, LAKESIDE, BOX 860, BROOKS, ALBERTA, TOJ
OJO, CANADA, (TECHNICAL PAPER)
INFORMATION SOURCE: COMPLETED SUBMISSION TO NBS

ACRONYM: PRONET TITLE: PROJECT NETWORK SYSTEM
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE
FEATURES: SUBJECT, DATA INPUT, USER OUTPUT, GRAPHICS,
ACTIVITY DIAGRAM, TABLES, STATIC ANALYSIS, MANAGEMENT,
PROJECT MANAGEMENT, DYNAMIC ANALYSIS, RESOURCE UTILIZATION,
STAGE OF DEVELOPMENT: DESIGN
TOOL PORTABLE: YES

TOOL SUMMARY: THE PRONET (PROJECT NETWORK) SYSTEM IS
DESIGNED TO MONITOR BOTH THE PLANNING AND THE EXECUTION
OF A GIVEN PROJECT. THE DEFINITION OF THIS PROJECT MAY FOLLOW
A FREE FORMAT, OR ELSE, IF THE RELATIONSHIPS BETWEEN
PROJECT ACTIVITIES ARE THEMSELVES DEFINABLE, THE PROJECT
MAY BE DEFINED IN TERMS OF NETWORK PLANNING. IN EITHER
CASE PRONET GENERATES AND GRAPHICALLY DISPLAYS A PERT
CHART ON WHICH IS TRACED THE PROGRESS OF THE PLAN, THE
SYSTEM STORES INFORMATION CONCERNING THE FACTORS INVOLVED
IN ARRIVING AT THE TARGET DATA AND IN ESTIMATING THE COST
OF THE PROJECT IN QUESTION. SHOULD ANY OF THESE FACTORS
CHANGE, THE SYSTEM TAKES THE NECESSARY REVISIONS, THIS
FACILITATES MODIFICATIONS OF THE PROJECT AND ASSISTS THE
TEAM IN DEALING WITH THE UNEXPECTED. SINCE PRONET
IS DESIGNED TO HANDLE BOTH NETWORK AND FREE-FORM PLANNING,
A PROJECT PLAN MAY CONSIST OF ACTIVITIES THAT ARE EITHER RELATED OR INDEPENDENT OF EACH OTHER.

DOCUMENTATION: USER'S MANUAL, MAINTENANCE MANUAL, DEVELOPMENT SPECIFICATION, TEST PLAN

DEVELOPER: SYNET COMPANY

INFORMATION SOURCE: AIAA SURVEY OF SOFT DEV TOOLS

ACRONYM: PSL, TITLE: PROGRAMMING SUPPORT LIBRARY

CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE

FEATURES: STANDARDS ENFORCEMENT, SUBJECT, CODE INPUT, FORTAN, COBOL, JOVIAL, JOVIAL J3, COBOL 68, TRANFORMATION, FORMATTING, USER OUTPUT, USER-ORIENTED TEXT, REPORTS, DOCUMENTATION, STATIC ANALYSIS, AUDITING, MANAGEMENT, LIBRARY MANAGEMENT, FILES MANAGEMENT

STAGE OF DEVELOPMENT: IMPLEMENTED, DATE: 770000

IMPLEMENTATION LANGUAGE: COBOL 68

TOOL PORTABLE: NO, TOOL SIZE: 37K

COMPUTER (OTHER HARDWARE): HONEYWELL 6XXX

OS (OTHER SOFTWARE): GCOS

TOOL AVAILABLE: NO

RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): AVAILABLE ACCORDING TO AIR FORCE MANUAL (AFM) 300-6, PARAGRAPH 11-7A.

TOOL SUPPORTED: YES, TOOL SUPPORT: RADC/ISIS


REFERENCES: [RADC74], RADC, "STRUCTURED PROGRAMMING SERIES", RADC TR-74-300, (15 VOLUMES), 760000

DEVELOPER: ROME AIR DEVELOPMENT CENTER (ISIE)

CONTACT: LAWRENCE M. LOMBARDI, ROME AIR DEVELOPMENT CENTER/ISIE, GRIFFIS AFB, NY, 13441, USA, 315-330-7634

INFORMATION SOURCE: COMPLETED SUBMISSION TO NBS

ACRONYM: PSL/PSA, TITLE: PROBLEM STATEMENT LANGUAGE/PROBLEM STATEMENT ANALYZER

CLASSIFICATION: REQUIREMENTS/DESIGN SPECIFICATION AND ANALYSIS

PSL/PSA

FEATURES: SUBJECT, DATA INPUT, VHLL INPUT, REQUIREMENTS LANGUAGE, PSL, PROBLEM STATEMENT LANGUAGE, MACHINE OUTPUT, DATA OUTPUT, OBJECT CODE OUTPUT, DIAGNOSTICS, TABLES, LISTINGS, STATIC ANALYSIS, CROSS REFERENCE, COMPLETENESS CHECKING, CONSISTENCY CHECKING, ERROR CHECKING, SYNTAX CHECKING

STAGE OF DEVELOPMENT: IMPLEMENTED

IMPLEMENTATION LANGUAGE: FORTRAN

TOOL PORTABLE: YES

TOOL AVAILABLE: YES, PUBLIC DOMAIN

RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): SPONSORSHIP

TOOL SUMMARY: PROBLEM STATEMENT LANGUAGE

PSL/PSA IS A LANGUAGE AND ASSOCIATED PROCESSOR THAT SUPPORTS AN ANALYST IN THE PREPARATION OF REQUIREMENTS DOCUMENT OR A TOP LEVEL DESIGN OF AN INFORMATION PROCESSING SYSTEM. PSL/PSA HAS BEEN DEVELOPED AT THE ISDS PROJECT AT THE UNIVERSITY OF MICHIGAN UNDER THE DIRECTION OF PROFESSOR D. TEICHENROTH. IT IS CURRENTLY AVAILABLE ON A NUMBER OF LARGE SCALE COMPUTERS. TO USE PSL/PSA, THE ANALYST FIRST DESCRIBES A PORTION OF THE TARGET SYSTEM USING THE FORMAL STATEMENTS OF PSL. THESE STATEMENTS ARE THEN ANALYZED BY PSL FOR SYNTACTICAL CORRECTNESS AND ARE ADDED TO A DATA BASE THAT CONTAINS ALL INFORMATION ABOUT THE TARGET SYSTEM. AT THE COMMAND OF THE ANALYST, REPORTS ARE GENERATED BY PSL THAT DESCRIBE DIFFERENT ASPECTS OF THE TARGET SYSTEM. THESE THREE STEPS ARE REPEATED UNTIL ALL ASPECTS OF THE TARGET SYSTEM ARE DESCRIBED IN PSL AND HAVE BEEN ENTERED INTO THE DATA BASE.

DOCUMENTATION: USER'S MANUAL

REFERENCES: [TEICHENROTH], D., HERSHEY III, "PSL/PSA: A COMPUTER-AIDED TECH. FOR STRUC.

DDU, ANALYSIS OF IPS", VOL. 83-3, NO. 1, IEEE TRANSACTIONS ON SOFT. ENG., 770100

CONTACT: CYRIL P. SVOBODA, ADVANCED SYSTEMS TECHNOLOGY CORPORATION, 911 ELMONT ROAD, SUITE 302, GREENBELT, MD, 20770, USA, 301-441-9036

HASAN M. SAYANI, ADVANCED SYSTEMS TECHNOLOGY CORPORATION, 911 ELMONT ROAD, SUITE 302, GREENBELT, MD, 20770, USA, 301-441-9036

DANIEL TEICHENROTH, UNIVERSITY OF MICHIGAN, ISDS PROJECT, 231/443, WEST ENGINERING BLDG, ANN ARBOR, MI, 48109, USA, 313-734-2238

DAVID CALLENDER, JET PROPULSION LABORATORY, 4800 OAK GROVE DRIVE, PASADENA, CA, 91109, USA, 213-344-5365

INFORMATION SOURCE: TOOL FAIR

ACRONYM: PWB FOR VAX/VMS, TITLE: PROGRAMMERS WORKBENCH TOOLS ON VAX/VMS

CLASSIFICATION: SOFTWARE SUPPORT SYSTEM/PROGRAMMING ENVIRONMENT

FEATURES: SUBJECT, DATA INPUT, CODE INPUT, TRANFORMATION, EDITING, MACHINE OUTPUT, CODE OUTPUT, DATA INPUT, USER OUTPUT, DIAGNOSTICS, TABLES, STATIC ANALYSIS, COMPARISON, COMPLETENESS CHECKING, MANAGEMENT.
CONFIDENTIALITY

SOFTWARE

IMPLEMENTATION LANGUAGE: C

TOOL SUPPORT: YES, TOOL SUPPORT: INTERACTIVE SYSTEMS CORP.

TOOL SUMMARY: SOFTWARE DEVELOPERS NEED GOOD TOOLS TO IMPROVE THEIR PRODUCTIVITY. ONE SUCH SET OF TOOLS HAS BEEN AVAILABLE SOME TIME NOW WITH THE PROGRAMMER'S WORKBENCH VERSION OF UNIX RUNNING ON THE DEC PDP-11 SERIES OF COMPUTERS. THE RECENT INTRODUCTION OF THE DEC VAX COMPUTERS WITH THE VMS OPERATING SYSTEM LEFT SOMETHING TO BE DESIRED IN THE AREA OF SOFTWARE TOOLS. A RICH SET OF LANGUAGES HAVE BEEN AND ARE BEING DEVELOPED UNDER THE VMS OPERATING SYSTEM, BUT THERE WAS NOTHING EQUIVALENT TO THE PWB TOOLS. INTERACTIVE SYSTEMS CORPORATION HAS RECENTLY INTRODUCED THE PWB TOOLS ON THE VAX/VMS SYSTEM, THEREBY IMPROVING THE PRODUCTIVITY OF SOFTWARE ENGINEERS ON THIS MACHINE SIGNIFICANTLY. THE TOOLS HAVE BEEN DESIGNED TO WORK EFFECTIVELY IN THE ENVIRONMENT.

CONTACT: HEINZ LYCKLAMA, INTERACTIVE SYSTEMS CORPORATION, 1212 SEVENTH ST., SANTA MONICA, CA, 90401, USA, 213-450-8563

INFORMATION SOURCE: TOOL FAIR

ACRONYMS: QC, QCM, QCRT, TITLE: QCRT CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE

FEATURES: SUBJECT, DATA INPUT, TRANSFORMATION, OPTIMIZATION, USER OUTPUT, TABLES, DYNAMIC ANALYSIS, TUNING, RESOURCE UTILIZATION

STAGE OF DEVELOPMENT: IMPLEMENTED

IMPLEMENTATION LANGUAGE: ASSEMBLY

COMPUTER (OTHER HARDWARE): IBM 360/370, AMDAHL 470

OS (OTHER SOFTWARE): OS/VS/VS, OS/MVS

TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO

RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): FOR SALE

TOOL SUMMARY: QCM (QUANTITATIVE COMPUTER MANAGEMENT) IS A COMPREHENSIVE FAMILY OF FOUR INDEPENDENT BUT INTERRELATED PROGRAMS DESIGNED TO MEASURE, REPORT, AND OPTIMIZE ALL FACETS OF COMPUTER UTILIZATION. THE MODULES ARE: SPM (SYSTEMS PERFORMANCE MODULE) FOR OFF-LINE CAPACITY PERFORMANCE MANAGEMENT AND OVERALL SYSTEM TUNING; SPI (SYSTEMS PERFORMANCE INTERROGATOR) FOR REAL-TIME DIAGNOSTIC PERFORMANCE MANAGEMENT AND SYSTEM PROGRAMMING CAPABILITIES VIA OPERATOR COMMAND; JOB ANALYSIS AND BILLING FOR EQUITABLE, ACCURATE, AND REPEATABLE RESOURCE CHARGEBACK AND JOB ANALYSIS; AND A REGULATOR FOR DYNAMIC THROUGHPUT OPTIMIZATION AND REAL-TIME PRIORITY CONTROL OF JOB

DOCUMENTATION: USER'S MANUAL

DEVELOPERS: THE MANAGEMENT GROUP, INC.

INFORMATION SOURCE: AIAA SURVEY OF SOFT DEV TOOLS

ACRONYMS: QC, QCM, QCRT, TITLE: QCRT CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE

FEATURES: SUBJECT, CODE INPUT, FORTRAN, COBOL, PL/I, ASSEMBLY LANGUAGE, USER OUTPUT, GRAPHICS, FLOW CHARTS, TABLES, STATIC ANALYSIS, CROSS REFERENCE, SCANING, TREE STRUCTURES, AND ISO COMPATIBILITY

IMPLEMENTATION LANGUAGE: COBOL, OBJECT
Quick-Draw

Computer (Other Hardware): IBM 360/370, UNIVAC 11XX, HONEYWELL 6XX

Tool Summary: Quick-Draw is a syntax analysis system that scans COBOL, BAL, AUTO CODER, FORTRAN, and PL/1 source programs and automatically produces flowcharts, program diagnostics, and a series of related cross-references. It is configuration independent with no manual intervention or special preparation required.


Information Source: TRW Software Tools Catalog

Acronym: QuickCode
Title: QuickCode

Classification: Program construction and generation

Computer (Other Hardware): DEC PDP-11
OS (Other Software): RSX-11

Public Domain: No

Restrictions (Copyrights,Licenses, Etc.): Restricted Rights

Tool Supported: Yes, Tool Support: Digital Equipment Corporation

Tool Summary: QuickCode is a programming aid for the users of COBOL. As a pre-compiler translator, QuickCode performs many of the redundant, non-creative tasks involved with programming and substantially reduces the coding required. Having internally defined and abbreviated over 125 of the most frequently used reserved words, the basic structure of COBOL is reduced to a set of easily remembered mnemonics which are as easy to read as the reserved words they represent. Key features of QuickCode include: internal and external data name dictionaries, multiple word expressions, M5 of COBOL and QuickCode statements and formatting of programs.


Developer: Digital Equip. Corp., Ask (MLS=2/M46)=A0180, 146 Main St., Maynard, MA

Information Source: NOSC Seatools Survey

Acronym: RA
Title: Requirements Automation

Classification: Requirements/Design Specification and Analysis

Features: Subject, VML Input, Requirements Specification, User Output, User-Oriented Text, Reports, Static Analysis, Completeness Checking, Consistency Checking, Tracking, Stage of Development, Implemented

Computer (Other Hardware): DEC PDP-11
OS (Other Software): UNIX

Tool Available: Yes

Restrictions (Copyrights, Licenses, etc.): Availability according to Air Force Manual (AFM) 300-6, Paragraph 11-74

Tool Supported: Yes, Tool Support: RADC/ISEE

Tool Summary: The FORTRAN code auditor, an automated test tool, is used for the cost effective enforcement of FORTRAN programming standards and conventions appropriate to the Air Force software environment. It does not modify code, using predefined coding standards and conventions, it simply advises the user where these standards and conventions have not been adhered to. The major advantage
OF FAVORING AN AUTOMATED AUDITOR OVER MANUAL METHODS, BESIDES COST EFFECTIVENESS, IS COMPLETE OBJECTIVITY AND UNAMBIGUITY. THE STANDARDS CAN BE VIEWED AS BEING CODING ENFORCEMENTS IN FOUR AREAS: (1) DOCUMENTATION STANDARDS - STANDARDS DEFINING QUANTITY AND PLACEMENT OF COMMENTARY TO ENHANCE PROGRAM READABILITY AND COMPREHENSIBILITY (2) FORMATTING STANDARDS - STANDARDS IDENTIFYING PHYSICAL PLACEMENT AND GROUPING OF CODE ELEMENTS ON THE SOURCE CODE LISTING, (3) DESIGN STANDARDS - STANDARDS LIMITING MODULE SIZE AND PLACING RESTRICTIONS ON THE USE OF CERTAIN INSTRUCTIONS WITH THE END RESULT OF PROVIDING AN OPTIMIZATION OF CODE RELATIVE TO EXECUTION TIME, (4) DEVELOPMENT IMPLEMENTED LANGUAGE: PASCAL

IMPLEMENTATION DESCRIPTION: RACED/ISLE, GRIFFISS AFB, NY, 13441, USA, 315-330-7834

INFORMATION SOURCE: COMPLETED SUBMISSION TO NBS

ACRONYM: RATCODER, TITLE: RACED/ISLE, CLASSIFICATION: PROGRAM CONSTRUCTION AND GENERATION

FEATURES: SUBJECT, DATA INPUT, TRANSFORMATION, EDITING, FORMATTING, MACHINE OUTPUT, SOURCE CODE OUTPUT, RATFORD, STAGE OF DEVELOPMENT: IMPLEMENTED

IMPLEMENTATION LANGUAGE: PASCAL

COMPUTER (OTHER HARDWARE): CDC 6600/7000

TOOL AVAILABLE: YES

TOOL SUMMARY: RATCODER PRODUCES ONE OR MORE STANDARDS RATFORD PROGRAMS FROM AN INPUT FILE OF LINES CONSISTING OF: MODULE NAME, AND A SHORT COMMENT DESCRIBING THE MODULE'S PURPOSE, EACH OF THE MODULES PRODUCED INCLUDES SUITABLY MARKED COMMENTS FOR THE HEADER (INCLUDING THE PURPOSE), MODULE AUTHOR, DATE AND TIME, AND A PLACE TO INSERT NOTES RELATIVE TO THE MODULE. CONTROL DATA CORPORATION "UPDATE" (REF. A) CONDITIONAL SOURCE CODE BLOCKS ARE WRITTEN AT THE ENTRY AND EXIT OF EACH MODULE TO MARK THE NAME AND TIME OF ENTRY, AND THE TOTAL ELAPSED TIME IN THE MODULE. THESE BLOCKS ARE LOGICALLY DELETABLE WHEN THE PROGRAM UNDER DEVELOPMENT REACHES THE PRODUCTION STAGE; HOWEVER, THEY REMAIN IN THE SOURCE CODE. LIBRARY. UNIQUENESS OF TEXT MARKS PRODUCED BY 'RATCODER' ALLOWS EASY LOCATION OF A SPECIFIC PORTION OF THE MODULES WITH A LINE ORIENTED TEXT EDITOR, SUFFICIENT "WHITE SPACE" IS INCLUDED IN EACH MODULE TO ENHANCE READABILITY OF THE SOURCE CODE. THE DEVELOPER HAS USED THIS PROGRAM TO DEVELOP SEVERAL "RATFORD" PROGRAMS.

DEVELOPER: NSRC

CONTACT: PETER S. ROTH, NSRC, STRUCTURES DEPARTMENT, BETHESDA, MD, 20084, USA, 202-277-1651

INFORMATION SOURCE: COMPLETED SUBMISSION TO NBS

ACRONYM: RATFORD, TITLE: RATIONAL FORTRAN TRANSLATOR, CLASSIFICATION: PROGRAM CONSTRUCTION AND GENERATION

FEATURES: SUBJECT, CODE INPUT, STRUCTURED FORTRAN, RATFORD, VMLL INPUT, STRUCTURED LANGUAGE, TRANSFORMATION, TRANSLATION, STRUCTURE PREPROCESSING, MACHINE OUTPUT, SOURCE CODE OUTPUT, FORTRAN

STAGE OF DEVELOPMENT: IMPLEMENTED DATE (YYMMD): 760000

IMPLEMENTATION LANGUAGE: FORTRAN

TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO

RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): COPYRIGHT 1976 BY BELL TELEPHONE LABORATORIES, INC., AND YOURDON, INC.

TOOL SUMMARY: THE PRIMARY PURPOSE OF RATFORD IS TO MAKE FORTRAN A BETTER PROGRAMMING LANGUAGE, FOR BOTH WRITING AND EXPLAINING, BY PERMITTING AND ENCOURAGING READABLE AND WELL-STRUCTURED PROGRAMS, THIS IS DONE BY PROVIDING THE CONTROL STRUCTURES THAT ARE UNAVAILABLE IN BARE FORTRAN, AND BY IMPROVING THE "COSMETICS" OF THE LANGUAGE. THE CONTROL FLOW STRUCTURES ARE IF-ELSE, WHILE, DO, BREAK, NEXT, FOR, REPEAT-UNTIL, AND STATEMENT GROUPING WITH BRACKETS. RATFORD IS TRANSLATED INTO FORTRAN, THUS RETAINING THE ADVANTAGES OF FORTRAN - UNIVERSALITY, PORTABILITY, AND RELATIVE EFFICIENCY - WHILE AT THE SAME TIME CONCEALING ITS WORST DRAWBACKS.

DOCUMENTATION DESCRIPTION: RATFOR

REFERENCES: (KERN76), BRIAN W. KERNIGAN AND P. J. PLUGER, "SOFTWARE TOOLS," ADDISON-WESLEY PUBLISHING CO., 760000

DEVELOPER: ADDISON-WESLEY PUBLISHING CO.

INFORMATION SOURCE: KERN76

ACRONYM: RALIGNMENT SYS, TITLE: COBOL REALIGNMENT SYSTEM CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING

FEATURES: SUBJECT, CODE INPUT, COBOL, TRANSFORMATION, FORMATTING, MACHINE OUTPUT, SOURCE CODE OUTPUT, COBOL, USER OUTPUT, LISTINGS

STAGE OF DEVELOPMENT: IMPLEMENTED DATE (YYMMD): 790000

IMPLEMENTATION LANGUAGE: COBOL

TOOL SIZE: 3,100 STATEMENTS

COMPUTER (OTHER HARDWARE): UNIVAC 11XX

LIBRARY AVAILABLE: YES

RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): AVAILABLE THROUGH FEDERAL SOFTWARE EXCHANGE, FSWE-C-79-0114

TOOL SUMMARY: COBOL PROGRAMS ARE OFTEN DIFFICULT AND TIME-CONSUMING TO DEBUG AND/OR MAINTAIN DUE TO THE LACK OF READABILITY. THIS IS ESPECIALLY TRUE WHEN THERE ARE NESTED IF-ELSE STATEMENTS WITHOUT CLEAR AND CONSISTENT INDENTATION OR WHEN THEIR ARE SEVERAL DIFFERENT STYLes OF RECORD DESCRIPTIONS. THE COBOL REALIGNMENT SYSTEM SOLVES THIS PROBLEM BY REALIGNING A COBOL PROGRAM INTO A MORE READABLE AND STANDARD FORM, THE INPUT TO THE SYSTEM IS A COBOL SOURCE PROGRAM WHICH HAS BEEN COMPILED WITHOUT ERRORS, THE OUTPUT IS A REALIGNED SOURCE PROGRAM.

REFERENCES: (FSWE-C-79-0114), GENERAL SERVICES ADMINISTRATION/NATIONAL TECHNICAL INFORMATION SERVICE, "FEDERAL SOFTWARE EXCHANGE CATALOG," GSA/ADTS/C-80/1.
REALIGNMENT SYS

P880=904001, 800100
CONTACT: GSA FEDERAL SOFTWARE EXCHANGE, 2 SKYLINE PL (11TH FL), 5203 LEESBURG PK, FALLS CHURCH, VA, 22041, 703-756-2610
INFORMATION SOURCE: FEDERAL SOFTWARE EXCHANGE CATALOG

ACRONYM: REFERR TITLE: REFERR PROGRAM
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE
FEATURES: SUBJECT, CODE INPUT, USER OUTPUT, TABLES, STATIC ANALYSIS, CROSS REFERENCE, SCANNING
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN IV
COMPUTER (OTHER HARDWARE): IBM 360/370

TOOL SUMMARY: THE REFERR PROGRAM OPERATES ON COMPILER-PRODUCED DATA TO AUTOMATICALLY PRODUCE CROSS-REFERENCE MAPS OF SUBROUTINE USAGES AND HIERARCHICAL DISPLAYS OF SUBROUTINE CALLING STRUCTURES, SINCE THIS DATA IS NOT NORMALLY PRODUCED BY VENDOR-SUPPLIED OPERATING SYSTEMS, REFERR CAN BE USED EXTENSIVELY FOR PROGRAM ANALYSIS AND DOCUMENTATION.

DOCUMENTATION: USER'S GUIDE
REFERENCES: [ADS079], APPLIED SYSTEMS DESIGN SECTION, TRW DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS: CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE TOOLS SERIES, 700100

DEVELOPER: TRW DEFENSE SYSTEMS SOFTWARE DEPARTMENT
CONTACT: CLARKE LUCAS, TRW DEFENSE SYSTEMS SOFTWARE DEPARTMENT, ONE SPACEN PARK, REDONDO BEACH, CA, 90278, USA, 213-535-0426

INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG

ACRONYM: REFILLII TITLE: REFILL II
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, USER OUTPUT, TABLES, DYNAMIC ANALYSIS, RESOURCE UTILIZATION, TIMING
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: ASSEMBLY
COMPUTER (OTHER HARDWARE): IBM 360/370
OS (OTHER SOFTWARE): SVS, 03/VS, 03/MVS, OS/MVT

TOOL AVAILABLE: YES
PUBLIC DOMAIN: NO
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): FOR LEASE, FOR SALE


DOCUMENTATION: USER'S MANUAL, MAINTENANCE MANUAL, DEVELOPMENT SPECIFICATION, TEST PLAN DEVELOPMENT CYCLES: INC.

INFORMATION SOURCE: AIAA SURVEY OF SOFTWARE TOOLS

ACRONYM: REFTRAN (TM) TITLE: REFTRAN (TM) SOFTWARE TOOLS
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, FORTRAN, FORTRAN IV, FORTRAN 77, USER OUTPUT, TABLES, LISTINGS, STATIC ANALYSIS, CROSS REFERENCE,
REFTRAN (TM)

STAGE OF DEVELOPMENT: IMPLEMENTED, DATE (YYMMD): 670000
IMPLEMENTATION LANGUAGE: FORTRAN
TOOL PORTABLE: YES
TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): MARKETED PRODUCT
TOOL SUPPORTED: YES, TOOL SUPPORT: GENERAL RESEARCH CORPORATION

TOOL SUMMARY: REFTRAN (TM) PRODUCES A COMPREHENSIVE CROSS-REFERENCE DICTIONARY OF SYMBOLIC NAMES FROM FORTRAN SOURCE PROGRAMS. THIS DICTIONARY PROVIDES ANALYSIS AND PROGRAMMERS WITH A DEPTH OF DOCUMENTATION HERTOFOR UNAVAILABLE, AND IS AN INVALUABLE AID IN THE EVERYDAY TASKS OF UNDERSTANDING, CHANGING, DEBUGGING, DOCUMENTING, AND CONVETING FORTRAN PROGRAMS. THE REFTRAN DICTIONARY IS ORGANIZED ALPHABETICALLY BY SYMBOLIC NAME AND SHOWS THE ACTUAL FORTRAN STATEMENT IN WHICH THE SYMBOLIC NAME APPEARS ALONG WITH THE NAME OF THE PROGRAM UNIT IN WHICH THE STATEMENT APPEARS. AN OUTSTANDING FEATURE OF REFTRAN (TM) IS THE CAPABILITY OF INCLUDING SELECTED FORTRAN "KEYWORDS" SUCH AS DATA, READ, WRITE, FORMAT, ENTRY, CALL, RETURN, AND STOP AS SYMBOLIC NAMES IN THE DICTIONARY.

DOCUMENTATION: USER'S MANUAL, PROGRAMMER'S MANUAL
DEVELOPER: WILLIAM R. DE HAAN
CONTACT: WILLIAM R. DE HAAN, GENERAL RESEARCH CORP., 5363 HOLLISTER AVE., PO BOX 6770, SANTA BARBARA, CA, 93111, USA, 805-966-7726
INFORMATION SOURCE: PRODUCT ANNOUNCEMENT

ACRONYM: REL MEAS MODEL, TITLE: RELIABILITY MEASUREMENT MODEL
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, DATA INPUT, USER OUTPUT, TABLES, STATISTICAL ANALYSIS, RELIABILITY ANALYSIS
STAGE OF DEVELOPMENT: IMPLEMENTED
TOOL PORTABLE: YES
TOOL AVAILABLE: YES


REFERENCES: [DONAB01], JOHN D. DONANOO AND DOROTHY SWERINGER, "A REVIEW OF SOFTWARE MAINTENANCE TECHNOLOGY", RADC-TR-80-13, INTERIM REPORT, 800228
[DONAB01], JOHN D. DONANOO AND DOROTHY SWERINGER, "A REVIEW OF SOFTWARE MAINTENANCE TECHNOLOGY", RADC-TR-80-13, INTERIM REPORT

DEVELOPER: BELL LABORATORIES, WHIPPANY, NEW JERSEY
INFORMATION SOURCE: RADC-TR-80-13, INTERIM REPORT

ACRONYM: RISOS, TITLE: RISOS
CLASSIFICATION: SOFTWARE DOCUMENTATION, MAINTENANCE, AND TESTING
FEATURES: SUBJECT, CODE INPUT, FORTRAN, TRANSFORMATION, EDITS, MACHINE OUTPUT, SOURCE CODE OUTPUT, FORTRAN, USER OUTPUT, LISTINGS, STATIC ANALYSIS, SCANNING
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN
COMPUTER (OTHER HARDWARE): IBM 360/370, CDC 6600/7600
DEVELOPER: RISOS DIV.
CONTACT: JAMES N. CHURCHYARD, RISOS DIV., 3333 HARBOR BLVD., COSTA MESA, CA, 92626, USA, 714-546-8030
INFORMATION SOURCE: AIAA SURVEY OF SOFTWARE TOOLS

ACRONYM: RISOS TOOLS, TITLE: RISOS TOOLS
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, ASSEMBLY LANGUAGE, USER OUTPUT, GRAPHICS, FLOW CHARTS, TABLES, STATIC ANALYSIS, CROSS REFERENCE, STATISTICAL ANALYSIS, PROFILE GENERATION
STAGE OF DEVELOPMENT: IMPLEMENTED, DATE (YYMMD): 760300
IMPLEMENTATION LANGUAGE: FORTRAN, CDC
DEVELOPER: NO
COMPUTER (OTHER HARDWARE): CDC 6600/7600
PUBLIC DOMAIN: YES

TOOL SUMMARY: THE RISOS TOOLS WERE DEVELOPED BY THE RESEARCH IN SECURED OPERATING SYSTEMS (RISOS) PROJECT AT LAWRENCE LIVERMORE LABORATORY. THESE TOOLS WERE DEVELOPED TO ANALYZE ASSEMBLY LANGUAGE PROGRAMS AND CONSIST OF THE FOLLOWING THREE ANALYTIC TOOLS: (1) SAP (STATISTICAL ANALYSIS PROGRAM) COUNTS ALL OCCURRENCES OF SPECIFIED SYMBOLS WITHIN A GIVEN MODULE, (2) SPAN (SOURCE PROGRAM ALTERATION MODULE) IDENTIFIES THE CONTROL FLOW IN A MODULE BY INSERTING COMMENTS PRIOR TO STATEMENTS THAT ARE BRANCHED TO AND STATEMENTS THAT CAUSE A BRANCH, AND (3) CRISP (CROSS REFERENCE INTERFACE AND SEARCH PROGRAM) IS A CHARACTER STRING PROCESSING ROUTINE THAT LOCATES INSTRUCTION PATTERNS BY USING THE NUMBER. THE RISOS TOOLS HAVE THE DRAWBACK THAT THEY REQUIRE FOUR FILES TO WORK PROPERLY. IN ORDER TO CREATE THOSE FILES, THE USER MUST EITHER USE AN
AVAILABLE PARSER OR MODIFY A PROTOTYPE PARSER FOR THE ASSEMBLY LANGUAGE THAT IS BEING ANALYZED.

DOCUMENTATION: TECHNICAL DESCRIPTION

REFERENCES: [CHI75], CHIN, J. S., "ANALYTIC TOOLS THAT AID IN THE TEST AND EVALUATION OF OS SECURITY", LAWRENCE LIVERMORE LABORATORY, REPT. UCRL-76434, 750500.

[WEB75], WEBB, D. A., ET. AL., "RISOS ANALYTICAL TOOL DESCRIPTION MANUAL, PART 2: PROGRAM SOURCE LISTINGS", LAWRENCE LIVERMORE LABORATORY, REPT. UCRL-51810 PT. 2, 750500.

[FRIC75], FRICKER, W. G., ET. AL., "RISOS ANALYTICAL TOOL DESCRIPTION MANUAL, PART 1: PROGRAM DESCRIPTION", LAWRENCE LIVERMORE LABORATORY, REPT. UCRL-51810 PT. 1, 750500.

DEVELOPER: LAWRENCE LIVERMORE LABORATORY

INFORMATION SOURCE: FRIC75

ACRONYM: RTT
TITLE: REQUIREMENTS TRACING TOOL

CLASSIFICATION: REQUIREMENTS/DESIGN SPECIFICATION AND ANALYSIS

FEATURES: SUBJECT, DATA INPUT, VHL INPUT, REQUIREMENTS SPECIFICATION, MACHINE OUTPUT, DATA OUTPUT, USER OUTPUT, GRAPHICS, HIERARCHICAL TREE, TABLES, LISTINGS, STATIC ANALYSIS, CROSS REFERENCE, TRACKING.

STAGE OF DEVELOPMENT: IMPLEMENTED

COMPUTER (OTHER HARDWARE): DEC SYSTEM=10/20

TOOL SUMMARY: RTT IS A TOOL TO SUPPORT THE AUTOMATED TRACING OF REQUIREMENTS. THIS TOOL ALSO AIDS IN THE IDENTIFICATION OF SYSTEM'S AREAS AFFECTED BY SPECIFICATION CHANGES AND SUPPORTS TEST PLANNING ACTIVITIES. THE TOOL BUILD A HIERARCHICAL REQUIREMENTS DATA BASE FROM USER INPUTS WHICH ARE MANUALLY EXTRACTED FROM APPROPRIATE DOCUMENTATION. THESE INPUTS INCLUDE A STATEMENT OF THE REQUIREMENT, ASSIGNMENT OF HIERARCHICAL LEVEL, UNIQUE IDENTIFICATION AND KEYWORD ASSIGNMENT. RTT PROVIDES THE CAPABILITY TO MERGE THE INPUTS FROM SEVERAL USERS INTO A MASTER DATA BASE AFTER PROVIDING FORMAT ERROR CHECKING. IT GENERATE USER SELECTABLE OUTPUT REPORTS SUCH AS MASTER REQUIREMENTS LISTINGS, LISTINGS BY KEYWORD, CROSS REFERENCES LISTINGS BY KEYWORDS, ROOT AND DERIVED REQUIREMENTS, AND INDEXES TO SOURCE DOCUMENTS.

REFERENCES: [REIF81], D. J. REIFER AND H. A. MONTGOMERY, "SEATEC SOFTWARE TOOLS SURVEY", RCI-TR-008, REIFER CONSULTANTS, INC., 810330

DEVELOPER: SAI COMSYSTMS

CONTACT: J. MILLER, SAI COMSYSTMS, 201 CAMINO DEL RIO
SOUTH, SAN DIEGO, CA, 93152, USA, 714-871-7500

INFORMATION SOURCE: NASC SEATEC TOOLS SURVEY

ACRONYM: RXV9 (TM) TITLE: A SOFTWARE ANALYSIS AND TESTING TOOL

CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING

FEATURES: SUBJECT, CODE INPUT, FORTRAN, IFTRAN, TRANSFORMATION, TRANSLATION, STRUCTURE, PREPROCESSING, RESTRUCTURING, INSTRUMENTATION, FORMATTING, MACHINE OUTPUT,

SOURCE CODE OUTPUT, FORTRAN, IFTRAN, USER OUTPUT, DIAGNOSTICS, GRAPHICS, TABLES, LISTINGS, STATISTICAL ANALYSIS, DATA FLOW ANALYSIS, INTERFACE ANALYSIS, CROSS REFERENCE, COMPLEXITY MEASUREMENT, COMPLETENESS CHECKING, CONSISTENCY CHECKING, UNITS ANALYSIS, TYPE ANALYSIS, STATISTICAL ANALYSIS, MANAGEMENT, ERROR CHECKING, STRUCTURE CHECKING, SCANNING, DYNAMIC ANALYSIS, COVERAGE ANALYSIS, ASSERTION CHECKING, SYMBOLIC EXECUTION, TUNING, TRACING, CONSTRAINT EVALUATION.

STAGE OF DEVELOPMENT: IMPLEMENTED

IMPLEMENTATION LANGUAGE: FORTRAN

TOOL PORTABLE: YES, TOOL SIZE: 50K WORDS

TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO

RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): LICENSE

TOOL SUPPORTED: YES, TOOL SUPPORT: GENERAL RESEARCH CORP.

TOOL SUMMARY: RXVP80 IS A SYSTEM OF TOOLS WHICH PERFORM A NUMBER OF FUNCTIONS SUCH AS DOCUMENTATION, ANALYSIS, AND TEST ASSISTANCE, FOR CODE WRITTEN IN COMMON DIALECTS OF FORTRAN INCLUDING FORTRAN 77. THE HEART OF THE SYSTEM IS A LIBRARY CAPABLE OF STORING THE RESULTS OF ANALYSIS OF VERY LARGE PROGRAMS (>100,000 SOURCE LINES), THE SYSTEM PERFORMS MUCH OF ITS ANALYSIS ON AN INTERNAL REPRESENTATION OF THE PROGRAM AS A DIRECTED GRAPH, ONE OF THE PRIMARY FEATURES OF RXVP80 IS ITS ABILITY TO ANALYZE ONLY THE NEW OR CHANGED MODULES OF A PROGRAM, USING THE STORED LIBRARY TO CHECK INTERFACES.

INFORMATION SOURCE: TOOL FAIR

ACRONYM: S-FORTRAN
TITLE: S-FORTRAN

CLASSIFICATION: PROGRAM CONSTRUCTION AND GENERATION

FEATURES: SUBJECT, CODE INPUT, STRUCTURED FORTRAN, VHL INPUT, STRUCTURED LANGUAGE, TRANSLATION, TRANSLATION, STRUCTURE, PREPROCESSING, MACHINE OUTPUT, SOURCE CODE OUTPUT, USER OUTPUT, DIAGNOSTICS, STATIC ANALYSIS, SCANNING.

STAGE OF DEVELOPMENT: IMPLEMENTED

IMPLEMENTATION LANGUAGE: STRUCTURED FORTRAN

COMPUTER (OTHER HARDWARE): IBM 360/370, SEL 32, CDC 6600/7000, UNIVAC 11XX

TOOL SUMMARY: THE S-FORTRAN LANGUAGE, DEVELOPED BY CAINE, FABER GORDON, INC., IS AN EXTENSION OF FORTRAN WHICH ALLOWS EASIER, EFFICIENT AND RELIABLE STRUCTURE PROGRAMMING IN A FORTRAN ENVIRONMENT. THE LANGUAGE RESULTS FROM THE ADJUNCTION OF A CAREFULLY CHOSEN SET OF CONTROL STRUCTURES TO EXISTING FORTRAN. THE S-FORTRAN LANGUAGE IS IMPLEMENTED USING AN S-FORTRAN TO FORTRAN TRANSLATOR, THE PURPOSE OF THE TRANSLATOR IS TO LIST THE INPUT PROGRAM AUTOMATICALLY IDENTIFYING POSSIBLE ERRORS AND PRINT APPROPRIATE

D IAGNOSTICS, AND PRODUCE EQUIVALENT FORTRAN PROGRAMS THAT...
S-FORTRAN CAN SUBSEQUENTLY BE COMPILED AND EXECUTED. DOCUMENTATION: LANGUAGE REFERENCE GUIDE REFERENCES: [ADS079], APPLIED SYSTEMS DESIGN SECTION, TRW DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS: CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE TOOLS SERIES, 790100 DEVELOPER: CAXNE, FARNER GORDON, INC. INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG


INFORMATION SOURCE: AIAA Survey of Soft Dev Tools


TOOL PORTABLE: NO, TOOL SIZE: 25000 LINES OF PL/I SOURCE COMPUTER (OTHER HARDWARE): HONEYWELL 6XXX

OS (OTHER SOFTWARE): MULTICS TOOL AVAILABLE: YES, PUBLIC DOMAIN: YES TOOL SUPPORTED: NO

TOOL SUMMARY: SARHAM (SYSTEM ARCHITECT'S APPRENTICE) IS A COMPUTER-AIDED DESIGN SYSTEM CURRENTLY UNDER DEVELOPMENT AT MIT, WHICH SUPPORTS A STRUCTURED MULTI-LEVEL DESIGN METHODOLOGY FOR THE DESIGN OF HARDWARE OR SOFTWARE SYSTEMS. IT COMPRISSES A NUMBER OF LANGUAGE PROCESSORS AND TOOLS FOR ASSISTING DESIGNERS USING THE SARAH METHODOLOGY, TOGETHER WITH A USER-INTERFACE CAPABILITY FOR ASSISTING DESIGNERS USING THE SARAH SYSTEM. THE SARAH SYSTEM IS IMPLEMENTED IN THE MIT MULTICS SYSTEM AND IS READILY ACCESSIBLE THROUGH ARPANET OR TELNET.


INFORMATION SOURCE: TOOL FAIR


TOOL SUMMARY: THE SARAH (SYSTEM ANALYSIS AND RESOURCE ACCOUNTING) SYSTEMS ENABLE USERS TO MEASURE COMPUTER SYSTEM CAPACITY AND ESTABLISH REASONABLE PRODUCTION PERFORMANCE STANDARDS. SARAH IS DESIGNED FOR USE WITH HONEYWELL COMPUTERS. THE PRIMARY INPUT SOURCE FOR SARAH IS HONEYWELL'S STATISTICAL COLLECTION FILE (SCF). THE SYSTEM PROCESSES THE RAW SCF DATA, ANALYZES IT AGAINST PRE-ESTABLISHED STANDARDS, AND OUTPUT VARIOUS ANALYSIS REPORT THE SYSTEMS PRODUCE THE FOLLOWING REPORTS: GENERAL SYSTEM OVERVIEW REPORTS, TOP-DOWN MANAGEMENT REPORTS, JOB MIX CHARACTERISTICS AND RESOURCE UTILIZATION, PERFORMANCE INDICATORS, GRAPHICAL DATA DISPLAYS SHOWING RESOURCE DISTRIBUTIONS, AND EXCEPTION REPORTS. SARAH IS ALSO PROVIDED REPORTS ON TSS ACTIVITY. A GENERAL PURPOSE REPORT WRITER AND A MANAGEMENT REPORTING SYSTEM ARE AVAILABLE AS OPTIONS.

DEVELOPER: BOEING COMPUTER SERVICES

INFORMATION SOURCE: AIAA SURVEY OF SOFT DEV TOOLS

SARA-U

STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTAN
COMPUTER (OTHER HARDWARE): UNIVAC 11XX
OS (OTHER SOFTWARE): EXEC 8
TOOL AVAILABLE: YES
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): FOR SALE

TOOL SUMMARY: THE SARA-U (SYSTEMS ANALYSIS AND RESOURCE ACCOUNTING) SYSTEM ENABLES USERS TO MEASURE COMPUTER SYSTEM CAPACITY AND ESTABLISH REASONABLE PRODUCTION PERFORMANCE STANDARDS. SARA-U IS DESIGNED FOR USE WITH THE UNIVAC'S MASTER LOG FILE (MLF) AS ITS INPUT SOURCE. THE SYSTEM PROCESSES THE RAW MLF DATA, ANALYZES IT AGAINST PRE-DEFINED STANDARD, AND OUTPUTS VARIOUS ANALYSIS REPORTS. THE SYSTEM PRODUCES THE FOLLOWING REPORTS: GENERAL SYSTEM OVERVIEW REPORTS, TOP-DOWN MANAGEMENT REPORTS, JOB MIX CHARACTERISTICS AND RESOURCE UTILIZATION, PERFORMANCE INDICATORS, GRAPHICAL DATA DISPLAY SHOWING RESOURCE DISTRIBUTIONS, AND EXCEPTION REPORTS. A GENERAL PURPOSE REPORT WRITER AND A MANAGEMENT REPORTING SYSTEM ARE AVAILABLE AS OPTIONS.

DEVELOPER: BOEING COMPUTER SERVICES, INC.
INFORMATION SOURCE: AIAA SURVEY OF SOFT DEV TOOLS

ACRONYM: SARA-U
TITLE: SARA-U
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, DATA INPUT, USER OUTPUT, TABLES, DYNAMIC ANALYSIS, TUNING, RESOURCE UTILIZATION,
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTAN
TOOL SIZE: CORE: 200-700K
COMPUTER (OTHER HARDWARE): IBM 360/370, ANDAH 470
OS (OTHER SOFTWARE): OS/VS
TOOL AVAILABLE: YES
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): FOR LEASE, FOR SALE

SARA-III

TOOL SUMMARY: THE SARA-III (SYSTEMS ANALYSIS AND RESOURCE ACCOUNTING) PACKAGES ARE COMPUTER CAPACITY MANAGEMENT TOOLS. SARA-III PROVIDES COMPREHENSIVE REPORTS DEPICTING BOTH HARDWARE AND SYSTEM CHARACTERISTICS AND SYSTEM BEHAVIOR AT EACH LEVEL OF MULTIPROGRAMMING. THE REPORTS ASSIST THE USER IN SETTING STANDARDS FOR LEVELS OF BATCH AND TSO PERFORMANCE AND PROVIDE THE NECESSARY VISIBLE FOR PERFORMANCE CONTROL. THE SYSTEM PROCESSES APPLICATION ANALYSIS REPORTS THAT ANALYZE AND EVALUATE JOBS, STEPS, AND PROGRAMS. IN ADDITION, SARA-III PRODUCES GRAPHIC DISPLAYS SHOWING PEAKS AND LOW SYSTEM PERFORMANCE LEVELS, USER ANALYSIS REPORTS SHOWING THE IMPACT OF USERS ON THE DATA PROCESSING ENVIRONMENT, AND SHIFT ANALYSIS REPORTS THAT REPORT SYSTEM ACTIVITY BY USER-DEFINED SHIFTS.

DOCUMENTATION: USER'S MANUAL, MAINTENANCE MANUAL, DEVELOPMENT SPECIFICATION, TEST PLAN
DEVELOPER: BOEING COMPUTER SERVICES, INC.
INFORMATION SOURCE: AIAA SURVEY OF SOFT DEV TOOLS

ACRONYM: SARA-III
TITLE: SARA-III
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, DATA INPUT, USER OUTPUT, TABLES, DYNAMIC ANALYSIS, TUNING, RESOURCE UTILIZATION,
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN
COMPUTER (OTHER HARDWARE): IBM 360/370, ANDAH 470
OS (OTHER SOFTWARE): OS/VS
TOOL AVAILABLE: YES
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): FOR LEASE, FOR SALE

TOOL SUMMARY: THE SARA-III (SYSTEMS ANALYSIS AND RESOURCE ACCOUNTING) PACKAGES ARE COMPUTER CAPACITY MANAGEMENT TOOLS. SARA-III PROVIDES COMPREHENSIVE REPORTS DEPICTING BOTH HARDWARE AND SYSTEM CHARACTERISTICS AND SYSTEM BEHAVIOR AT EACH LEVEL OF MULTIPROGRAMMING. THE REPORTS ASSIST THE USER IN SETTING STANDARDS FOR LEVELS OF BATCH AND TSO PERFORMANCE AND PROVIDE THE NECESSARY VISIBLE FOR PERFORMANCE CONTROL. THE SYSTEM PROCESSES APPLICATION ANALYSIS REPORTS THAT ANALYZE AND EVALUATE JOBS, STEPS, AND PROGRAMS. IN ADDITION, SARA-III PRODUCES GRAPHIC DISPLAYS SHOWING PEAKS AND LOW SYSTEM PERFORMANCE LEVELS, USER ANALYSIS REPORTS SHOWING THE IMPACT OF USERS ON THE DATA PROCESSING ENVIRONMENT, AND SHIFT ANALYSIS REPORTS THAT REPORT SYSTEM ACTIVITY BY USER-DEFINED SHIFTS.

DOCUMENTATION: USER'S MANUAL, MAINTENANCE MANUAL, DEVELOPMENT SPECIFICATION, TEST PLAN
DEVELOPER: BOEING COMPUTER SERVICES, INC.
INFORMATION SOURCE: AIAA SURVEY OF SOFT DEV TOOLS

ACRONYM: SARA-III
TITLE: SARA-III
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, DATA INPUT, USER OUTPUT, TABLES, DYNAMIC ANALYSIS, TUNING, RESOURCE UTILIZATION,
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN
COMPUTER (OTHER HARDWARE): IBM 360/370, ANDAH 470
OS (OTHER SOFTWARE): OS/VS
TOOL AVAILABLE: YES
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): FOR LEASE, FOR SALE
BUT CONCISE HIERARCHICAL TRACE OF EVERY PATH ENCOUNTERED DURING THE SIMULATION PROCESS, SCAN/370’S EXCEPTION REPORTS PROVIDE AN EARLY WARNING SYSTEM WHICH HIGHLIGHTS SPECIFIC EVENTS WHICH COULD ADVERSELY AFFECT THE PROGRAM’S EXECUTION.

DOCUMENTATION: USER’S GUIDE, IMPLEMENTATION GUIDE DEVELOPER: GROUP OPERATIONS, INCORPORATED CONTACT: M. W. MORGAN, GROUP OPERATIONS, INCORPORATED, 1110 VERMONT AVE, NW, WASHINGTON, DC, 20005, USA, 202-687-5420 INFORMATION SOURCE: COMPLETED SUBMISSION TO NBS


TOOL SUMMARY: SCERT IS A COMPUTER SYSTEM SIMULATION PACKAGE DESIGNED TO SIMULATE ALMOST ANY WORKING ENVIRONMENT, INCLUDING REAL-TIME, VIRTUAL STORAGE, TIME-SHARING, MULTIPROGRAMMING, MULTIPROCESSING, DATA BASE MANAGEMENT, AND DATA COMMUNICATIONS, BASICALLY, SCERT UTILIZES HIGHLY DETAILED DESCRIPTIONS OF CONFIGURATIONS AND APPLICATIONS PROGRAMMING SPECIFICATIONS, AND BREAKS THEM DOWN INTO INDIVIDUAL PROCESSING EVENTS, THEN FROM THE WORKLOAD DEFINITION AND THE FACTOR LIBRARY, TIMING AND UTILIZATION INFORMATION IS COMPUTED FOR EQUIPMENT COMPONENTS AND FOR CONTROL SOFTWARE SUCH AS OPERATING SYSTEMS AND/OR INPUT/OUTPUT CONTROL ROUTINES, THEREBY PROVIDING DETAILED THROUGHPUT INFORMATION REGARDING TOTAL SYSTEM PERFORMANCE.

DOCUMENTATION: USER’S MANUAL, MAINTENANCE MANUAL, DEVELOPMENT SPECIFICATION, TEST PLAN DEVELOPER: PERFORMANCE SYSTEMS, INC. INFORMATION SOURCE: AIAA SURVEY OF SOFT DEV TOOLS


RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): HUGHES PROPRIETARY TOOL SUPPORTED: NO

TOOL SUMMARY: THE STRUCTURE CHART GRAPHICS SUBSYSTEM (SCG) WAS DEVELOPED TO SOLVE THE FOLLOWING FUNDAMENTAL SOFTWARE DESIGN PROBLEM: ALTHOUGH EFFECTIVE DESIGN GUIDELINES HAVE BEEN ESTABLISHED WHICH REDUCE THE OVERALL LIFE CYCLE COST OF SOFTWARE, CONSISTENT AND THOROUGH APPLICATION OF THESE GUIDELINES ACROSS ALL DEVELOPMENT EFFORTS IS HAMPERED BY A LENGTHY TECHNOLOGY TRANSFER TIME AND THE RARE AVAILABILITY OF EXPERT DESIGNERS. THE SCG IS A FIRST GENERATION FEASIBILITY DEMONSTRATION OF AN APPROACH WHICH ADDRESSES THIS PROBLEM, THE USER INTERACTS WITH SCG THROUGH A GRAPHICS TERMINAL USING DESIGN COMMANDS SUCH AS "PLACE A MODULE", "CONNECT TWO MODULES", AND "REDEFINE GRAPHICS COMMANDS SUCH AS "JUMP TO MODULE X", "REPOSITION SUBTREE TO (X,Y)" TO HAVE A LOCALIZED DESIGN VIEW. THE USER CAN REQUEST EITHER PAGE-SIZED CALCOMP COPY OR WALL CHARTS OF HIS ENTIRE DESIGN.

DOCUMENTATION: USERS MANUAL DEVELOPER: HUGHES AIRCRAFT COMPANY CONTACT: JAMES W. WINCHESTER, HUGHES AIRCRAFT COMPANY, POST OFFICE BOX 3310, FULLERTON, CA, 92640, USA, 714-732-3232 INFORMATION SOURCE: TOOL FAIR


OS (OTHER SOFTWARE): OS/VS, OS/MVS RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): AGREEMENT REQUIRED.

TOOL SUPPORTED: YES, TOOL SUPPORT: HUGHES AIRCRAFT CO.

TOOL SUMMARY: SCG PROVIDES AN INTERACTIVE GRAPHICS INTERFACE FOR DEVELOPING, MODIFYING AND DOCUMENTING SOFTWARE DESIGN STRUCTURE CHARTS DEVELOPED UNDER THE STRUCTURED DESIGN METHODOLOGY ORIGINALLY PROPOSED BY L. CONSTANTINE IN 1974. DLM USES THE SCG DATA BASE TO ESTABLISH MEASURES OF QUALITY FOR STRUCTURED DESIGNS IN PARTICULAR, PLOTS OF COMPLEXITY AND TREE PURITY AS A FUNCTION OF TREE DEPTH, FAN IN AND FAN OUT INFORMATION, AND THE TESTABILITY MODIFYABILITY OF DESIGN. AN AUTOMATIC LAYOUT FEATURE HAS JUST BEEN COMPLETED WHICH RESTRUCTURES A
SARA-Ill

COMPUTER (OTHER HARDWARE): IBM 360/370, AMDAL 470
OS (OTHER SOFTWARE): CS, OS/VSE
TOOL AVAILABLE: YES
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): FOR LEASE, FOR SALE

TOOL SUMMARY: THE SARA (SYSTEMS ANALYSIS AND RESOURCE ACCOUNTING) PACKAGES ARE COMPUTER CAPACITY MANAGEMENT TOOLS. SARA III PROVIDES COMPREHENSIVE REPORTS DESCRIBING JOB MIX CHARACTERISTICS AND SYSTEM BEHAVIOR AT EACH LEVEL OF MULTIPROGRAMMING. THE REPORTS ASSIST THE USER IN SETTING STANDARDS FOR LEVELS OF BATCH AND IS0 PERFORMANCE AND PROVIDE THE NECESSARY VISIBILITY FOR PERFORMANCE CONTROL. THE SYSTEM PRODUCES APPLICATION ANALYSIS REPORTS THAT ANALYZE AND EVALUATE JOBS, STEPS, AND PROGRAMS. IN ADDITION, SARA III PRODUCES GRAPHIC DISPLAYS SHOWING PEAK AND NOMINAL SYSTEM PERFORMANCE LEVELS, USER ANALYSIS REPORTS SHOWING THE IMPACT OF USERS ON THE DATA PROCESSING ENVIRONMENT, AND SHIFT ANALYSIS REPORTS THAT REPORT SYSTEM ACTIVITY BY USER-DEFINED SLOTS.

DOCUMENTATION: USER'S MANUAL, MAINTENANCE MANUAL, DEVELOPMENT SPECIFICATION, TEST PLAN
DEVELOPER: BOEING COMPUTER SERVICES, INC.
INFORMATION SOURCE: AIAA SURVEY OF SOFT DEV TOOLS

ACRONYM: SARA-IV, TITLE: SARA-IV
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, DATA INPUT, USER OUTPUT, TABLES, DYNAMIC ANALYSIS, TUNING, RESOURCE UTILIZATION
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN
TOOL SIZE: CORE: 200-700K
COMPUTER (OTHER HARDWARE): IBM 360/370, AMDAL 470
OS (OTHER SOFTWARE): CS, OS/VSE
TOOL AVAILABLE: YES
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): FOR LEASE, FOR SALE

TOOL SUMMARY: THE SARA (SYSTEMS ANALYSIS AND RESOURCE ACCOUNTING) PACKAGES ARE COMPUTER CAPACITY MANAGEMENT TOOLS. SARA IV IS DESIGNED TO ADDRESS THOSE PROBLEMS UNIQUE TO MVS. SARA IV INCLUDES ALL OF THE FEATURES OF SARA III PLUS MF/I AND RMF ANALYSIS CAPABILITIES. SARA IV ACCEPTS MF/I AND RMF RECORD AND PREPARES THEM FOR PROCESSING BY THE SARA REPORT WRITER.

DOCUMENTATION: USER'S MANUAL, MAINTENANCE MANUAL, DEVELOPMENT SPECIFICATION, TEST PLAN
DEVELOPER: BOEING COMPUTER SERVICES, INC.
INFORMATION SOURCE: AIAA SURVEY OF SOFT DEV TOOLS

ACRONYM: SARA-U, TITLE: SARA-U
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, DATA INPUT, USER OUTPUT, GRAPHICS, TABLES, DYNAMIC ANALYSIS, TUNING, RESOURCE UTILIZATION

SARA-U

STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN
COMPUTER (OTHER HARDWARE): UNIVAC 11XX
OS (OTHER SOFTWARE): EXEC 8
TOOL AVAILABLE: YES
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): FOR LEASE

TOOL SUMMARY: THE SARA (SYSTEMS ANALYSIS AND RESOURCE ACCOUNTING) PACKAGES ENABLE USERS TO MEASURE COMPUTER SYSTEM CAPACITY AND ESTABLISH REASONABLE PRODUCTION PERFORMANCE STANDARDS. SARA-U IS DESIGNED FOR USE WITH THE UNIVAC COMPUTERS. THE PRIMARY INPUT SOURCE FOR SARA-U IS UNIVAC'S MASTER LOG FILE (MLF) AS ITS INPUT SOURCE. THE SYSTEM PROCESS THE MLF DATA, ANALYZE AGAINST ESTABLISHED STANDARDS, AND OUTPUT VARIOUS ANALYSIS REPORTS. THE SYSTEM PRODUCES THE FOLLOWING REPORTS: GENERAL SYSTEM OVERVIEW REPORTS, TOP-DOWN MANAGEMENT REPORTS, JOB MIX CHARACTERISTICS AND RESOURCE UTILIZATION, PERFORMANCE INDICATORS, GRAPHICAL DATA DISPLAYS SHOWING RESOURCE DISTRIBUTIONS, AND EXCEPTION REPORTS. A GENERAL PURPOSE REPORT WRITER AND A MANAGEMENT REPORTING SYSTEM ARE AVAILABLE AS OPTIONS.

DEVELOPER: BOEING COMPUTER SERVICES
INFORMATION SOURCE: AIAA SURVEY OF SOFT DEV TOOLS

ACRONYM: SCAN/370, TITLE: SCAN/370
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, COBOL, USER OUTPUT, DIAGNOSTICS, GRAPHICS, HIERARCHICAL TREE, TABLES, LISTINGS, STATIC ANALYSIS, COMPARISON, CROSS REFERENCE, STRUCTURE CHECKING, IMPLEMENTATION LANGUAGE: COBOL, ALC
TOOL PORTABLE: NO, TOOL SIZE: 10,000 STATEMENTS
COMPUTER (OTHER HARDWARE): IBM 360/370
OS (OTHER SOFTWARE): OS/VS, OS/VSE
TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): MARKETED PRODUCT
TOOL SUPPORTED: YES, TOOL SUPPORT: GROUP OPERATIONS, INCORPORATED

TOOL SUMMARY: SCAN/370 IS AN INNOVATIVE ANALYSIS TOOL DESIGNED TO SPEED THE DEVELOPMENT AND MAINTENANCE OF COBOL PROGRAMS AND IMPROVE OPERATIONAL RELIABILITY. SCAN/370 USES AN ADVANCED SIMULATION TECHNIQUE WHICH ANALYZES THE SOURCE CODE AND WALKS THROUGH EVERY POSSIBLE PROCESSING PATH WHICH CAN OCCUR DURING THE PROGRAM'S EXECUTION. THE INFORMATION THAT IS GATHERED DURING THIS WALK-THROUGH PROCESS IS NOT DATA DEPENDENT AND, HENCE, COVERS EVERY POSSIBLE PROCESSING CONDITION. SCAN/370 INTERSEPARATE CLEAR AND CONCISE NARRATIVE COMMENTS THROUGHOUT THE PROCEDURE DIVISION OF THE COMPILE LISTING TO PROVIDE CRITICAL ANALYSIS INFORMATION AS AN INTEGRAL PART OF THE PROGRAMMER'S WORKING DOCUMENT. SCAN/370 BACKS UP ITS NARRATIVE COMMENTS IN THE SOURCE LISTING WITH A COMPLETE
BUT CONCISE HIERARCHICAL TRACE OF EVERY PATH ENCOUNTERED
DURING THE SIMULATION PROCESS, SCAN/370'S EXCEPTION
REPORTS PROVIDE AN EARLY WARNING SYSTEM WHICH HIGHLIGHTS
SPECIAL EVENTS WHICH COULD ADVERSELY AFFECT THE PROGRAM'S
EXECUTION.

DOCUMENTATION: USER'S GUIDE, IMPLEMENTATION GUIDE
DEVELOPER: GROUP OPERATIONS, INCORPORATED
CONTACT: M. W. MORGAN, GROUP OPERATIONS, INCORPORATED, 1110
VERMONT AVE, NW, WASHINGTON, DC, 20005, USA, 202-887-5420
INFORMATION SOURCE: COMPLETED SUBMISSION TO NBS

ACRONYM: SCERT, TITLE: SCERT
CLASSIFICATION: SOFTWARE MODELING AND SIMULATION
FEATURES: SUBJECT, DATA INPUT, USER OUTPUT, TABLES, DYNAMIC
ANALYSIS, SIMULATION
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: COBOL
TOOL SIZE: CORE: 150K
COMPUTER (OTHER HARDWARE): IBM 360/370
OS (OTHER SOFTWARE): OS, OS/VS

SCG

RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.):
HUGHES

PROCUTARY

TOOL SUPPORTED: NO

TOOL SUMMARY: THE STRUCTURE CHART GRAPHICS SUBSYSTEM (SCG)
WAS DEVELOPED TO SOLVE THE FOLLOWING FUNDAMENTAL SOFTWARE
DESIGN PROBLEM: ALTHOUGH EFFECTIVE DESIGN GUIDELINES HAVE
BEEN ESTABLISHED WHICH REDUCE THE OVERALL LIFE CYCLE COST
OF SOFTWARE, CONSISTENT AND THOROUGH APPLICATION OF THESE
GUIDELINES ACROSS ALL DEVELOPMENT EFFORTS IS HAMPERED BY A
LENGTHY TECHNOLOGY TRANSFER TIME AND THE RARE AVAILABILITY
OF EXPERT DESIGNERS, THE SCG IS A FIRST GENERATION
FEASIBILITY DEMONSTRATION OF AN APPROACH WHICH ADDRESSES
THIS PROBLEM, THE USER INTERACTS WITH SCG THROUGH A
GRAPHICS TERMINAL USING DESIGN COMMANDS SUCH AS "PLACE A
PROGRAM", "CONNECT TWO MODULES", AND USER GRAPHICS COMMANDS
SUCH AS "JUMP TO MODULE X", REPOSITION SUBTREE TO (X,Y)",
HAVING CAPTURED HIS DESIGN IN A DATA BASE, THE USER CAN
REQUEST EITHER PAGE-SIZED CALCOMP COPIES OR WALL CHARTS OF
HIS ENTIRE DESIGN.

DOCUMENTATION: USERS MANUAL
DEVELOPER: HUGHES AIRCRAFT COMPANY
CONTACT: JAMES W. WINCHESTER, HUGHES AIRCRAFT COMPANY, POST
OFFICE BOX 3310, FULLERTON, CA, 92634, USA, 714-732-3232
INFORMATION SOURCE: TOOL FAIR

ACRONYM: SCG/ODM, TITLE: STRUCTURE CHART GRAPHICS/DESIGN
QUALITY METRICS
CLASSIFICATION: REQUIREMENTS/DESIGN SPECIFICATION AND
ANALYSIS
FEATURES: STANDARDS ENFORCEMENT, TEST DATA GENERATION,
PROGRAM STRUCTURE CHECKING, SOFTWARE QUALITY EVALUATION,
SUBJECT, CODE INPUT, VM/P INPUT, DESIGN SPECIFICATION, USER
OUTPUT, USER-ORIENTED TEXT, REPORTS, DOCUMENTATION, TABLES,
STATIC ANALYSIS, CROSS REFERENCE, COMPLEXITY MEASUREMENT,
CONSISTENCY CHECKING, AUDITING, MANAGEMENT, CHANGE CONTROL,
STAGE OF DEVELOPMENT: IMPLEMENTED, DATE (YYMMD): 790600
IMPLEMENTATION LANGUAGE: FORTRAN
TOOL SIZE: CORE: 250KB
COMPUTER (OTHER HARDWARE): DEC PDP-11, Amdahl 470 (DLM)
MB
OS (OTHER SOFTWARE): TSO, OS/MVS

RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): AGREEMENT
REQUIRED
TOOL SUPPORTED: YES, TOOL SUPPORT: HUGHES AIRCRAFT CO.

TOOL SUMMARY: SCG PROVIDES AN INTERACTIVE GRAPHICS INTERFACE
FOR DEVELOPING, MODIFYING AND DOCUMENTING SOFTWARE DESIGN
STRUCTURES CHARTS DEVELOPED UNDER THE STRUCTURED DESIGN
METODOLOGY ORIGINALLY PROPOSED BY L. CONSTANTINE IN 1974.
DM USES THE SCG DATA BASE TO ESTABLISH MEASURES OF QUALITY
FOR STRUCTURED DESIGNS; IN PARTICULAR, PLOTT OF COMPLEXITY
AND TREE PURITY AS A FUNCTION OF TREE DEPTH, IN AND OUT
 INFORMATION, OF THE TESTABILITY/MODIFIABILITY OF A DESIGN. AN AUTOMATIC LAYOUT
FEATURE HAS JUST BEEN COMPLETED WHICH RESTRUCTURES A
SCOBOL (TM)

STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: COBOL, SCOBOL
TOOL PORTABLE: YES
TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO
REQUIREMENTS (COPYRIGHTS, LICENSES, ETC.): MARKETED PRODUCT
TOOL SUPPORTED: YES, TOOL SUPPORT: SOFTWARE CONSULTING SERVICES
TOOL SUMMARY: SCOBOL IS A PROGRAMMING LANGUAGE FOR STRUCTURED PROGRAMMING. IT IS A SUPERSET OF COBOL. IT GIVES YOU EVERYTHING THAT COBOL DOES PLUS THE CONTROL STRUCTURES YOU NEED FOR STRUCTURED CODING. THE CENTRAL PART OF THE SCOBOL SYSTEM IS A PRECOMPILER PROGRAM. IT READS A PROGRAM THAT YOU HAVE WRITTEN IN THE SCOBOL LANGUAGE AND TRANSLATES IT INTO A COBOL PROGRAM READY TO BE COMPILED BY YOUR COMPILER. A PROGRAMMER'S MOST IMPORTANT TOOL IS HIS PROGRAMMING LANGUAGE. IT AFFECTS NOT ONLY THE WAY HE CODES BUT THE WAY HE THINKS ABOUT A PROBLEM. SCOBOL MAKES IT POSSIBLE FOR PROGRAMMERS TO WRITE CORRECT PROGRAMS USING THE PROPER CONTROL STRUCTURES OF STRUCTURED PROGRAMMING WITHOUT THE LABORIOUS AND ERROR-PRONE HAND TRANSLATION REQUIRED TO DO THIS IN COBOL.

DOCUMENTATION: USER'S MANUAL
REFERENCES: [REIF81], D. J., REIFER AND H. A., MONTGOMERY, "SEATETS SOFTWARE TOOLS SURVEY", RCI-TR-008, REIFER CONSULTANTS, INC., 810330
DEVELOPER: SOFTWARE CONSULTING SERVICES
CONTACT: M. CICHELLI, SOFTWARE CONSULTING SERVICES, 910 WHITTIER DRIVE, ALLENTOWN, PA, 18103, USA, 215-797-9690
INFORMATION SOURCE: NOBC SEATETS TOOLS SURVEY

ACRONYM: SCOBOL (TM), TITLE: SCOBOL (TM)
CLASSIFICATION: REQUIREMENTS/DESIGN SPECIFICATION AND ANALYSIS
FEATURES: SUBJECT, CODE INPUT, CICS, TRANSLATION, SYNTHESIS, MACHINE OUTPUT, SOURCE CODE OUTPUT, STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: COBOL
COMPUTER (OTHER HARDWARE): IBM 360/370
OS (OTHER SOFTWARE): OS/VS, DOS/VS
TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO
TOOL SUMMARY: SCOPE IS A CICS PROGRAM GENERATOR THAT ENABLES COBOL PROGRAMMERS WITH MINIMAL OR NO CICS TRAINING TO CODE CICS APPLICATION PROGRAMS. THE SYSTEM GENERATES COMPLETE COBOL=CICS SOURCE PROGRAMS AND ASSOCIATED SCREEN MAPPINGS FROM A SIMPLE INPUT LANGUAGE. CONVERSATION BETWEEN THE TERMINAL OPERATOR AND THE PROGRAM IS HANDLED STEP BY STEP TO REDUCE CPU LOADS AND RESPONSE TIME. SCOPE OBTAINS AND RELEASES MAIN STORAGE AND DISK SPACE AS NECESSARY, IT PROVIDES BASE ADDRESSES, MAPPING SUPPORT, DUMMY SECTIONS, ETC. EXTENSIVE DOCUMENTATION IS PROVIDED. SCOPE ALSO INCLUDES AN INTERACTIVE DEBUGGING FEATURE CALLED TRACE.

DOCUMENTATION: USER'S MANUAL
REFERENCES: [REIF81], D. J., REIFER AND H. A., MONTGOMERY, "SEATETS SOFTWARE TOOLS SURVEY", RCI-TR-008, REIFER CONSULTANTS, INC., 810330
DEVELOPER: SOFTWARE CONSULTING SERVICES
CONTACT: M. CICHELLI, SOFTWARE CONSULTING SERVICES, 910 WHITTIER DRIVE, ALLENTOWN, PA, 18103, USA, 215-797-9690
INFORMATION SOURCE: NOBC SEATETS TOOLS SURVEY

ACRONYM: SCOBOL (TM), TITLE: SCOBOL (TM)
CLASSIFICATION: PROGRAM CONSTRUCTION AND GENERATION
FEATURES: SUBJECT, CODE INPUT, COBOL, SCOBOL, TRANSFORMATION, TRANSLATION, STRUCTURE PREPROCESSING, FORMATTING, MACHINE OUTPUT, SOURCE CODE OUTPUT, COBOL, USER OUTPUT, LISTINGS,
DOCUMENTATION: USER'S MANUAL, MAINTENANCE MANUAL, DEVELOPMENT SPECIFICATION, TEST PLAN
DEVELOPER: DATACREN CORPORATION
INFORMATION SOURCE: AIAA SURVEY OF SOFT DEV TOOLS

ACRONYM: SDL, TITLE: SOFTWARE DESIGN AND DOCUMENTATION LANGUAGE
CLASSIFICATION: REQUIREMENTS/DESIGN SPECIFICATION AND ANALYSIS
FEATURES: SUBJECT, VHLL INPUT, SDL, TRANSFORMATION, FORMATTING, USER OUTPUT, LISTINGS, STRUCTURED LISTINGS, STATIC ANALYSIS, SCANNING, KEY WORD SCANNING, STRUCTURE SCANNING,
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: PASCAL
TOOL PORTABLE: YES
COMPUTER (OTHER HARDWARE): DEC VAX-11
OS (OTHER SOFTWARE): VMS
TOOL AVAILABLE: YES, PUBLIC DOMAIN: YES
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): AVAILABLE FROM COSMIC
TOOL SUPPORTED: NO

TOOL SUMMARY: SDL IS A LANGUAGE AND ASSOCIATED PROCESSOR THAT IS ORIENTED TOWARD SUPPORTING SOFTWARE DEVELOPMENT. THE TOOL PROCESSOR ACCEPTS AN INPUT OF A FILE OF SOURCE STATEMENTS, STRUCTURES THESE SOURCE STATEMENTS ACCORDING TO THE GENERAL RULES OF STRUCTURED PROGRAMMING, AND PRODUCES VARIOUS REPORTS INCLUDING A STRUCTURED LIST OF THE ORIGINAL SOURCE STATEMENTS. ONE OF ITS MAJOR FEATURES IS THE ABILITY TO SELECT KEYWORDS THAT ARE APPROPRIATE FOR THE APPLICATION OF THE USER. THE PROPOSED SCENARIO WOULD ALLOW AN ICSC ATTENDEE TO CREATE A VERY BRIEF INPUT FILE AND THEN HAVE THIS INPUT FILE PROCESSED AND THE OUTPUT RETURNED TO THE ATTENDEE THROUGH A HARD COPY TERMINAL. THE ONLY CONSTRAINT WOULD BE THE SIZE OF THE INPUT FILE.

DOCUMENTATION: REFERENCE GUIDE
REFERENCES: [CALL60], CALLENDER, CLARKSON, AND FRASIER, "AN APPLICATION OF SDL", JPL REPORT 80-16, 800000
DEVELOPER: JET PROPULSION LABORATORY
CONTACT: BARRY COOPER, JET PROPULSION LABORATORY, 4600 OAK GROVE DRIVE, PASADENA, CA, 91109, USA, 213-354-6159
INFORMATION SOURCE: TOOL FAIR

TOOL SUMMARY: SDL IS A DESIGN LANGUAGE FOR DEFINING THE DESIGN STRUCTURE OF A SOFTWARE SYSTEM. IT IS ASSOCIATED WITH A SOFTWARE DESIGN ANALYZER (SDA) THAT PROVIDES AUTOMATED SUPPORT TO AID IN THE LOGIC AND ACCURACY DESIGN PRIOR TO IMPLEMENTATION OR PROGRAMMING CONSIDERATIONS. FIRST THE OBJECT IS DEFINED, WITH UNIQUE NAMES FROM ALLOWABLE TYPES SUCH AS SUBSYSTEMS, MODULES, OPERATORS, COLLECTIONS OF DATA, ETC. AND THE RELATIONSHIP AMONG THE OBJECTS IS DEFINED FROM ALLOWABLE TYPES SUCH AS REFERENCES, COMPOSITION OF, HAS SUBPARTS, ETC. THE USER ALSO ATTACHES INFORMATION TO EACH OBJECT IN THE FORM OF COMMENT ENTRIES OR VALUES OF SPECIFIC PROPERTIES. THE SDA ADDS (UPDATES) NEW OBJECTS TO THE DATABASE AND ADDS RELATIONSHIPS, ATTRIBUTES, CLASSIFICATIONS, AND TEXT AS SPECIFIED. SDA PROVIDES A LISTING OF THE INPUT SPECIFICATION AND A FORMATTED LISTING WITH CROSS-REFERENCE INFORMATION.

DEVELOPER: UNIVERSITY OF MICHIGAN
CONTACT: BARRY COOPER, UNIVERSITY OF MICHIGAN, ANN ARBOR, MI, 48109, USA, 313-763-2238
INFORMATION SOURCE: ICSC SEATOS SOFTWARE SURVEY

ACRONYM: SDC, TITLE: SOURCE DIRECTORY PROGRAM CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE
FEATURES: SUBJECT, CODE INPUT, USER OUTPUT, TABLES, LISTINGS, STATIC ANALYSIS, MANAGEMENT, CONFIGURATION MANAGEMENT, VERSION CONTROL
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN
COMPUTER (OTHER HARDWARE): IBM 360/370
TOOL SUMMARY: ONE OF THE PROBLEMS ENCOUNTERED DURING THE DEVELOPMENT AND MAINTENANCE OF A LARGE SCALE SOFTWARE SYSTEM IS ACCURATELY DETERMINING THE CURRENT SOFTWARE CONFIGURATION. THIS INVOLVES THE IDENTIFICATION OF EACH ACCOUNTABLE ELEMENT IN THE SYSTEM (SUBROUTINE, MODULE, PROGRAM, DATA BASE ENTRY, ETC.), AND ITS CURRENT MODIFICATION LEVEL, ALTHOUGH FREQUENTLY DONE BY HAND, THIS IDENTIFICATION PROCESS CAN EASILY TAKE ON MAMMOTH PROPORTIONS AS THE NUMBER OF IDENTIFIABLE ENTITIES GROWS INTO THE HUNDREDS AND THE THOUSANDS. THE SOURCE DIRECTORY PROGRAM WAS WRITTEN TO AUTOMATE THE IDENTIFICATION OF EACH SUBROUTINE IN THE CURRENT SYSTEM CONFIGURATION. THE PRINTED OUTPUT PROVIDES THE PROJECT WITH A CONCISE ACCURATE RECORD OF EACH ROUTINE NAME, CURRENT VERSION NUMBER (MODIFICATION LEVEL) AND DOCUMENT NUMBER (CORRESPONDING TO THE SOURCE LISTING) AND NUMBER OF CARDS IN THE ROUTINE. EVERY PROJECT HAS TO REPORT THIS DATA IN SOME FORM.
CLASSIFICATION: SOFTWARE MODELING AND SIMULATION
FEATURES: SUBJECT, CODE INPUT, USER OUTPUT, TABLES, LISTINGS, STATIC ANALYSIS, MANAGEMENT, CONFIGURATION MANAGEMENT, DYNAMIC ANALYSIS, SIMULATION, STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN, COBOL, ASSEMBLY, JOVIAL J73
COMPUTER (OTHER HARDWARE): DEC SYSTEM 10/20

TOOL SUMMARY: SDVS IS AN INTEGRATED SOFTWARE SYSTEM TO SUPPORT DESIGN, CODING, TEST AND MAINTENANCE OF DIGITAL AVIONICS INFORMATION SYSTEM (DAIS) MISCELLANEOUS SOFTWARE (PROTOTYPE OFP). IT PROVIDES THE FOLLOWING CAPABILITIES: A SIMULATION OF THE DAIS PROCESSORS AND DATA BUS FOR DEVELOPING M ISSION SOFTWARE WITHOUT USING THE ACTUAL HARDWARE; AUTOMATIC CONFIGURATION MANAGEMENT OF M ISSION SOFTWARE; AUTOMATIC CONTROL OF SIMULATION RUNS; EDITING AND PROCESSING OF DATA GENERATED BY THE SIMULATION; AND SIMPLER, EASY USER LANGUAGE WHICH ALLOWS THE SOFTWARE DEVELOPER, THE TEST ENGINEER, AND THE PROJECT MANAGER TO COMMUNICATE WITH THE SYSTEM.

DOCUMENTATION: SYSTEM DESCRIPTION
REFERENCES: [AS0379], APPLIED SYSTEMS DESIGN SECTION, TRW DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS: CATALOG AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE TOOLS SERIES, 790100

DEVELOPER: TRW SOFTWARE TECHNOLOGY DEPT
CONTACT: HAL HART, TRW SOFTWARE TECHNOLOGY DEPT, ONE SPACE PARK, REDONDO BEACH, CA, 90278, USA, 213-536-1781

INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG

CLASSIFICATION: SOFTWARE ENGINEERING FACILITY
FEATURES: SUBJECT, CODE INPUT, USER OUTPUT, USER-ORIENTED TEXT, REPORTS, DOCUMENTATION, STATIC ANALYSIS, INTERFACE ANALYSIS, MANAGEMENT, CONFIGURATION MANAGEMENT, TEST MANAGEMENT, LIBRARY MANAGEMENT, GLOBAL VARIABLE MANAGEMENT, DYNAMIC ANALYSIS, REGRESSION TESTING, STAGE OF DEVELOPMENT: IMPLEMENTED

TOOL SUPPORTED: YES, TOOL SUPPORT: NAVAL AIR DEVELOPMENT CENTER, WARMINSTER, PA

TOOL SUMMARY: AN SEF CONSISTS OF A SOFTW ARE ENGINEERING DATABASE, STRUCTURED DATA MANAGEMENT SYSTEM (SDMS), STRUCTURED DATA LANGUAGE PROCESSOR (SDL P) AND A SET OF SUBSYSTEMS WHICH PROVIDE SPECIALIZED SUPPORT TO SOFTWARE DEVELOPMENT. ITS PRIMARY OBJECTIVE IS TO PROVIDE A CAPABILITY FOR AUTOMATED CAPTURE OF AS MUCH SOFTWARE DEVELOPMENT DATA AS POSSIBLE FOR THE SOFTWARE ENGINEERING DATABASE. IN ADDITION, THE SEF, THROUGH THE STRUCTURED DATA LANGUAGE, PROVIDES A COMMON PROTOCOL FOR INVOLVING THE SUBSYSTEM PROCESSORS. THESE PROCESSORS MAY BE LINKED DIRECTLY TO THE SEF AS AN "SEF COMPATIBLE PROCESSOR."

REFERENCES: [ID04801], JOHN D. DONAGHO AND DOROTHY T. STAFFORD, "A REVIEW OF SOFTWARE MAINTENANCE TECHNOLOGY", RADC-TR-80-13, INTERIM REPORT, 800200

CLASSIFICATION: REQUIREMENTS/DESIGN SPECIFICATION AND ANALYSIS
FEATURES: SUBJECT, VHDL INPUT, TRANSFORMATION, FORMATTING, USER OUTPUT, USER-ORIENTED TEXT, DOCUMENTATION, GRAPHICS, HIERARCHICAL TREE, LISTINGS, STATIC ANALYSIS, CROSS REFERENCE, STRUCTURE CHECKING, SCANNING,

STAGE OF DEVELOPMENT: IMPLEMENTED

IMPLEMENTATION LANGUAGE: FORTRAN 66

TOOL AVAILABLE: YES, TOOL SIZE: 64 KB

TOOL SUPPORT: YES, TOOL SUPPORT: MAYDA SOFTWARE ENGINEERING

TOOL SUMMARY: SDP IS A VERY HIGH LEVEL LANGUAGE (VHDL) THAT APPLIES SOFTWARE ENGINEERING METHODOLOGIES SUCH AS TOP-DOWN DESIGN, STRUCTURED DESIGN, AND DATA ABSTRACTION TO THE DESIGN OF SYSTEMS. THE DESIGNER EXPRESSES HIS IDEAS IN A STRUCTURED ENGLISH-LIKE LANGUAGE. SDP PRODUCES DOCUMENTS WHICH DISPLAY HIS DESIGN IN A CLEAR AND READABLE MANNER SO THAT ALL THOSE INVOLVED CAN EASILY UNDERSTAND THE SOLUTION. SDP FORMATS THE DESIGN MODELS, PROVIDING FULL CROSS-REFERENCING OF THE MODULES AND DATA ITEMS, AND A TREE REPRESENTING THE HIERARCHICAL SEQUENCE OF REFERENCES (CALLING TREE). IN ADDITION, SDP PROVIDES A MECHANISM FOR DESIGNING THE CONTROL OF SYNCHRONIZATION BETWEEN PROCESSES. ADDITIONAL FEATURES INCLUDE TEXT MODULES, INCLUSION OF EXTERNAL MODULES, INTERFACE DEFINITION, USER-DEFINED KEYWORDS, AND PARAMETERIZED MODULE NAMING. SDP PERFORMS SEVERAL CONSISTENCY CHECKS ON THE DESIGN SUCH AS PROPER USE OF CONTROL PRIMITIVES.

DOCUMENTATION: TECHNICAL PAPER, PROGRAMMER'S GUIDE

DEVELOPER: MAYDA SOFTWARE ENGINEERING

CONTACT: MAYDA SOFTWARE ENGINEERING, PO BOX 1389, REMOVOT, 76113, ISRAEL, 054-565554

INFORMATION SOURCE: TOOL FAIR

ACRONYM: SDVS, TITLE: SOFTWARE DESIGN AND VERIFICATION SYSTEM

REFERENCES: [AD05800], DONCH, "A REVIEW OF SOFTWARE MAINTENANCE TECHNOLOGY", RADC-TR-80-13, INTERIM REPORT, 800200

CLASSIFICATION: SOFTWARE ENGINEERING FACILITY
FEATURES: SUBJECT, CODE INPUT, USER OUTPUT, USER-ORIENTED TEXT, DOCUMENTATION, STATIC ANALYSIS, INTERFACE ANALYSIS, MANAGEMENT, CONFIGURATION MANAGEMENT, TEST MANAGEMENT, LIBRARY MANAGEMENT, GLOBAL VARIABLE MANAGEMENT, DYNAMIC ANALYSIS, REGRESSION TESTING, STAGE OF DEVELOPMENT: IMPLEMENTED

TOOL SUPPORTED: YES, TOOL SUPPORT: NAVAL AIR DEVELOPMENT CENTER, WARMINSTER, PA

TOOL SUMMARY: AN SEF CONSISTS OF A SOFTWARE ENGINEERING DATABASE, STRUCTURED DATA MANAGEMENT SYSTEM (SDMS), STRUCTURED DATA LANGUAGE PROCESSOR (SDLP) AND A SET OF SUBSYSTEMS WHICH PROVIDE SPECIALIZED SUPPORT TO SOFTWARE DEVELOPMENT. ITS PRIMARY OBJECTIVE IS TO PROVIDE A CAPABILITY FOR AUTOMATED CAPTURE OF AS MUCH SOFTWARE DEVELOPMENT DATA AS POSSIBLE FOR THE SOFTWARE ENGINEERING DATABASE. IN ADDITION, THE SEF, THROUGH THE STRUCTURED DATA LANGUAGE, PROVIDES A COMMON PROTOCOL FOR INVOLVING THE SUBSYSTEM PROCESSORS. THESE PROCESSORS MAY BE LINKED DIRECTLY TO THE SEF AS AN "SEF COMPATIBLE PROCESSOR."

REFERENCES: [ID04801], JOHN D. DONAGHO AND DOROTHY T. STAFFORD, "A REVIEW OF SOFTWARE MAINTENANCE TECHNOLOGY", RADC-TR-80-13, INTERIM REPORT, 800200

CLASSIFICATION: REQUIREMENTS/DESIGN SPECIFICATION AND ANALYSIS
FEATURES: SUBJECT, VHDL INPUT, TRANSFORMATION, FORMATTING, USER OUTPUT, USER-ORIENTED TEXT, DOCUMENTATION, GRAPHICS, HIERARCHICAL TREE, LISTINGS, STATIC ANALYSIS, CROSS REFERENCE, STRUCTURE CHECKING, SCANNING,

STAGE OF DEVELOPMENT: IMPLEMENTED

IMPLEMENTATION LANGUAGE: FORTRAN 66

TOOL AVAILABLE: YES, TOOL SIZE: 64 KB

TOOL SUPPORT: YES, TOOL SUPPORT: MAYDA SOFTWARE ENGINEERING

TOOL SUMMARY: SDP IS A VERY HIGH LEVEL LANGUAGE (VHDL) THAT APPLIES SOFTWARE ENGINEERING METHODOLOGIES SUCH AS TOP-DOWN DESIGN, STRUCTURED DESIGN, AND DATA ABSTRACTION TO THE DESIGN OF SYSTEMS. THE DESIGNER EXPRESSES HIS IDEAS IN A STRUCTURED ENGLISH-LIKE LANGUAGE. SDP PRODUCES DOCUMENTS WHICH DISPLAY HIS DESIGN IN A CLEAR AND READABLE MANNER SO THAT ALL THOSE INVOLVED CAN EASILY UNDERSTAND THE SOLUTION. SDP FORMATS THE DESIGN MODELS, PROVIDING FULL CROSS-REFERENCING OF THE MODULES AND DATA ITEMS, AND A TREE REPRESENTING THE HIERARCHICAL SEQUENCE OF REFERENCES (CALLING TREE). IN ADDITION, SDP PROVIDES A MECHANISM FOR DESIGNING THE CONTROL OF SYNCHRONIZATION BETWEEN PROCESSES. ADDITIONAL FEATURES INCLUDE TEXT MODULES, INCLUSION OF EXTERNAL MODULES, INTERFACE DEFINITION, USER-DEFINED KEYWORDS, AND PARAMETERIZED MODULE NAMING. SDP PERFORMS SEVERAL CONSISTENCY CHECKS ON THE DESIGN SUCH AS PROPER USE OF CONTROL PRIMITIVES.

DOCUMENTATION: TECHNICAL PAPER, PROGRAMMER'S GUIDE

DEVELOPER: MAYDA SOFTWARE ENGINEERING

CONTACT: MAYDA SOFTWARE ENGINEERING, PO BOX 1389, REMOVOT, 76113, ISRAEL, 054-565554

INFORMATION SOURCE: TOOL FAIR

ACRONYM: SDVS, TITLE: SOFTWARE DESIGN AND VERIFICATION SYSTEM

REFERENCES: [AD05800], DONCH, "A REVIEW OF SOFTWARE MAINTENANCE TECHNOLOGY", RADC-TR-80-13, INTERIM REPORT, 800200
ACRONYM: SELECT, TITLE: SYMBOLIC EXECUTION LANGUAGE TO ENABLE COMPREHENSIVE TESTING
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, LISP, LISP SUBSET, USER OUTPUT, LISTINGS, DYNAMIC ANALYSIS, ASSERTION CHECKING, SYMBOLIC EXECUTION, TRACING, PATH FLOW TRACING
STATE OF DEVELOPMENT: IMPLEMENTED, DATE (YYYY-MM-DD): 740000
IMPLEMENTATION LANGUAGE: LISP
TOOL SUPPORTED: YES, TOOL SUPPORT: STANFORD RESEARCH INSTITUTE

TOOL SUMMARY: SELECT IS A SYMBOLIC EXECUTION TOOL WHICH IS INTENDED TO BE A COMPROMISE BETWEEN AN AUTOMATED PROGRAM PROVING SYSTEM AND AN AD HOC DEBUGGING PRACTICE EXPERIMENTALLY, SELECT INCLUDES: (A) SEMANTIC ANALYSIS OF PROGRAMS, (B) CONSTRUCTION OF INPUT DATA CONSTRAINTS TO COVER SELECTED PROGRAM PATHS, (C) IDENTIFICATION OF SOME UNFACILILABLE PROGRAM PATHS, (D) AUTOMATIC DETERMINATION OF ACTUAL (REAL NUMBER) INPUT DATA TO DRIVE THE TEST PROGRAM THROUGH SELECTED PATHS, (E) EXECUTION (ACTUAL OR SYMBOLIC) OF THE TEST PROGRAM WITH OPTIMAL INTERMEDIATE ASSERTIONS AND OUTPUT ASSERTIONS, (F) GENERATION OF SIMPLIFIED EXPLANATIONS FOR THE VALUES OF ALL PROGRAM VARIABLES, IN TERMS OF SYMBOLIC INPUT VALUES, AND (G) PATH ANALYSIS FOR EACH POTENTIALLY EXECUTABLE PATH OR FOR A USER-SELECTED SUBSET OF PATHS, MULTIPLE EXECUTIONS OF A LOOP WITH A PATH ARE DEFINED AS SEPARATE PATHS, PRODUCING A POTENTIALLY INFINITE NUMBER OF DISTINCT PATHS. THE NUMBER OF LOOP TRAVERSALS MAY BE CONstrained BY THE USER.
REFERENCES: [DONA60], JOHN D. DONAHOO AND DOROTHY SWEARINGER, "A REVIEW OF SOFTWARE MAINTENANCE TECHNOLOGY", RADC-TR-80-13, INTERIM REPORT, 900200 [BOYE75], R. S. BOYER, "SELECT A FORMAL SYS. FOR TESTING DEBUG. PROG. BY SYMBOLIC EXECUTION", PROCEEDINGS INTER. CONF. ON RELIABLE SOFTWARE, PP 234-245, 750400
DEVELOPER: STANFORD RESEARCH INSTITUTE
INFORMATION SOURCE: RADC-TR-80-13, INTERIM REPORT

ACRONYM: SFORT-1, TITLE: STRUCTURED FORTRAN PREPROCESSOR
CLASSIFICATION: PROGRAM CONSTRUCTION AND GENERATION
FEATURES: SUBJECT, CODE INPUT, STRUCTURED FORTRAN, VHLL INPUT, STRUCTURED LANGUAGE, TRANSFORMATION, TRANSLATION, STRUCTURE PREPROCESSING, FUNDAMENTAL MACHINE OUTPUT, SOURCE CODE OUTPUT, FORTRAN, USER OUTPUT, DIAGNOSTICS, LISTINGS, STATIC ANALYSIS, ERROR CHECKING
STATE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN

TOOL SIZE: CORE: 72K
COMPUTER (OTHER HARDWARE): IBM 360/370
TOOL AVAILABLE: YES, PUBLIC DOMAIN: YES
TOOL SUMMARY: THIS IS A TYPICAL STRUCTURED FORTRAN PREPROCESSOR THAT SUPPORTS TWO ADDITIONAL CONSTRUCTS TO THE STANDARD FORTRAN LANGUAGE: DOWHILE, IF THEN ELSE, NONSTATEMENT EXECUTIONS INCLUDING THE ENDFOR AND ENDOF AND ARE GENERATED BY THE PREPROCESSOR, THIS PREPROCESSOR ALSO PERFORMS FORMATTING, ERROR CHECKING AND REPORTING, THE EXTENT OF WHICH IS CONTROLLED BY INPUT PARAMETERS.
DOCUMENTATION: USER'S MANUAL, MAINTENANCE MANUAL, DEVELOPMENT CERTIFICATION, TEST PLAN
DEVELOPER: NASA/GODDARD SPACE FLIGHT CENTER
CONTACT: F. E. MCGARRY, NASA/GODDARD SPACE FLIGHT CENTER, CODE 582.1, GREENBELT, MD, 20771, USA, 301-344-5048
INFORMATION SOURCE: AIAA SURVEY OF SOFT DEV TOOLS

ACRONYM: SFTR3, TITLE: SFTR3 STRUCTURED FORTRAN PREPROCESSOR
CLASSIFICATION: PROGRAM CONSTRUCTION AND GENERATION
FEATURES: SUBJECT, CODE INPUT, STRUCTURED FORTRAN, VHLL INPUT, STRUCTURED LANGUAGE, TRANSFORMATION, TRANSLATION, STRUCTURE PREPROCESSING, MACHINE OUTPUT, SOURCE CODE OUTPUT, FORTRAN, USER OUTPUT, DIAGNOSTICS, LISTINGS, STATIC ANALYSIS, ERROR CHECKING
STAGE OF DEVELOPMENT: DESIGN
IMPLEMENTATION LANGUAGE: FORTRAN, STRUCTURED FORTRAN 3
TOOL PORTABLE: YES
TOOLS AVAILABLE: NO, PUBLIC DOMAIN: YES
DOCUMENTATION: USER'S MANUAL
DEVELOPER: JET PROPULSION LABORATORY
CONTACT: S. PALACIOS, JET PROPULSION LABORATORY, 4800 OAK GROVE DRIVE, PASADENA, CA, 91109, USA, 213-344-4880
INFORMATION SOURCE: AIAA SURVEY OF SOFT DEV TOOLS

ACRONYM: SIGS, TITLE: SAMM INTERACTIVE GRAPHICS SYSTEM
CLASSIFICATION: REQUIREMENTS/DESIGN SPECIFICATION AND ANALYSIS
FEATURES: SUBJECT, VHLL INPUT, REQUIREMENTS SPECIFICATION, DESIGN SPECIFICATION, REQUIREMENTS LANGUAGE, USER OUTPUT, USER-ORIENTED TEXT, DOCUMENTATION, GRAPHICS, STATIC ANALYSIS, COMPLETE CHECKING,
SMMA

DOCUMENTATION: USER'S MANUAL
DEVELOPER: JET PROPULSION LABORATORY
CONTACT: SANDY PALACIOS, JET PROPULSION LABORATORY, 4800 OAK GROVE DRIVE, PASADENA, CA, 91109, USA, 213-344-7372
INFORMATION SOURCE: AIAA SURVEY OF SOFT DEV TOOLS

ACRONYM: SMMA, TITLE: STRUCTURED MACROS FOR THE MOCOMP ASSEMBLER
CLASSIFICATION: PROGRAM CONSTRUCTION AND GENERATION
FEATURES: SUBJECT, CODE INPUT, TRANSFORMATION, MACRO EXPANSION, MACHINE OUTPUT, OBJECT CODE OUTPUT
STAGE OF DEVELOPMENT: DESIGN
IMPLEMENTATION LANGUAGE: ASSEMBLY
COMPUTER (OTHER HARDWARE): MOCOMP
TOOL AVAILABLE: NO, PUBLIC DOMAIN: YES
TOOL SUMMARY: THE STRUCTURED MACROS ARE A SET OF MACROS DEVELOPED TO EXPLICITLY SUPPORT THE CONCEPTS OF STRUCTURED PROGRAMMING. THEY HAVE BEEN USED WITH THE MOCOMP ASSEMBLER AND MAY PRODUCE REENTRANT CODE. THE MACROS ARE IN THE FINAL STAGE OF TESTING.
ACRONYM: SNOOP, TITLE: SNOOP
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE
FEATURES: SUBJECT, VHDL INPUT, DESIGN SPECIFICATION, USER OUTPUT, DIAGNOSTICS, USER-ORIENTED TEXT, DOCUMENTATION, GRAPHICS, FLOW CHARTS, HIPO CHARTS, TABLES, STATIC ANALYSIS, DATA FLOW ANALYSIS, CROSS REFERENCE, MANAGEMENT, GLOBAL VARIABLE MANAGEMENT.
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: COBOL
COMPUTER (OTHER HARDWARE): UNIVAC 11XX
DOCUMENTATION: INTERNAL REPORT (59)
CONTACT: J. W. ESCH, SPERRY UNIVAC, P.O. BOX 3525, ST. PAUL, MN, 55165, USA, 612-456-2222
INFORMATION SOURCE: COMPLETED SUBMISSION TO NBS
ACRONYM: SOFTool 80 (TM), TITLE: A METHODOLOGY AND A COMPREHENSIVE SET OF TOOLS FOR THE 1980s
CLASSIFICATION: SOFTWARE SUPPORT SYSTEM/PROGRAMMING ENVIRONMENT
FEATURES: SUBJECT, CODE INPUT, VHLL INPUT, TRANSCRIPTION, TRANSLATION, INSTRUCTIONAL MATERIAL, TEXT DOCUMENTATION, TABLES, LISTINGS, STATIC ANALYSIS.
COMPARISON, DATA FLOW ANALYSIS, INTERFACE ANALYSIS, COMPLEXITY MEASUREMENT, COMPLETENESS CHECKING, AUDITING, MANAGEMENT, CONFIGURATION MANAGEMENT, STRUCTURE CHECKING, SCANNING, DYNAMIC ANALYSIS, COVERAGE ANALYSIS, TESTING, TUNING.
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN
TOOL PORTABLE: YES, TOOL SIZE: 2K - 27K BYTES
TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): PROPRIETARY
TOOL SUPPORTED: YES, TOOL SUPPORT: SOFTool CORPORATION
TOOL SUMMARY: SOFTool 80 is an integrated methodology and supporting tools for software management, development, and maintenance. Conceptually, SOFTool 80 allows a user to interactively create in an "APPLICATION GENERATOR MODE" A SUBSTANTIAL PORTION OF A NEW APPLICATION, TYPICALLY, OVER 50% OF THE REQUIRED CODE. THE USER THEN, WITH THE ASSISTANCE OF AN ELABORATE COLLECTION OF SOFTWARE TOOLS, COMPLETES THE PROGRAM AND GENERATES A DELIVERABLE PRODUCT. THE AREAS SUPPORTED WITH EXISTING TOOLS INCLUDE: STRUCTURED PROGRAMMING AT A LEVEL THAT MATCHES DESIGN DOCUMENTS, EXTENSIVE DIAGNOSTICS, CODE AUDITING, PORTABILITY, DOCUMENTATION, TRACING, TESTING, TIME AND SPACE OPTIMIZATION, MANAGEMENT VISIBILITY, STANDARDS AND QUALITY CONTROL ARE GIVEN EXPERT SUPORT, EXPERIENCE IN THE USE OF DESCRIPTION 1.0 INDICATES THAT FIVEFOLD IMPROVEMENTS, OVER CONVENTIONAL APPROACHES, IN THE AMOUNT AND QUALITY OF THE SOFTWARE CREATED ARE READILY ATTAINED.
DOCUMENTATION: TECHNICAL PAPER, REFERENCE MANUALS
CONTACT: SOFTool CORPORATION, 340 S. KELLOGG, GOLETA, CA, 93117, USA, 805-964-0560
INFORMATION SOURCE: TOOL FAIR
ACRONYM: SPC, TITLE: SYSTEM 3 PROJECT CONTROL
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE
FEATURES: SUBJECT, CODE INPUT, USER OUTPUT, TABLES, STATIC ANALYSIS, MANAGEMENT, PROJECT MANAGEMENT.
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: RPG II
TOOL SIZE: CORE: 12K
COMPUTER (OTHER HARDWARE): IBM SYSTEM 3
OS (OTHER SOFTWARE): SPC
TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): FOR SALE
TOOL SUMMARY: PROJECT CONTROL FOR THE SYSTEM 3 MODEL 6 (5798 ADJ.) PROVIDES A BASIC TOOL TO HELP MANAGEMENT IN THE PLANNING, SUPERVISING, AND CONTROLLING OF PROJECT-ORIENTED WORK BY THE CRITICAL PATH METHOD. THIS METHOD IS A TECHNIQUE TO PLAN, SCHEDULE, SUPERVISE, AND CONTROL ANY PROJECT WHICH IS COMPOSED OF A SERIES OF TIME-RELATED EVENTS, EACH EVENT HAS A TIME DURATION AND A RELATIONSHIP TO OTHER EVENTS WHICH MUST FINISH BEFORE IT CAN START AND
TO EVENTS WHICH CANNOT START UNTIL IT IS FINISHED, A START DATE IS ASSIGNED TO THE FIRST EVENT, PERMITTING START DATES TO BE CALCULATED FOR ALL THE OTHER EVENTS. IN ADDITION TO CRITICAL PATH ANALYSIS, THE SYSTEM PROVIDES THE CAPABILITY FOR SUMMARIZING PREPARED RESOURCE INFORMATION, RESOURCE REPORTING ALLOWS USERS TO DISPLAY THE STATUS OF UP TO FOUR DIFFERENT RESOURCES PER WORK ITEM, COMPARING THE ESTIMATED RESOURCES AGAINST ACTUAL RESOURCES USED. A PROGRESS REPORT SHOWS THE CURRENT STATUS OF ALL WORK ITEMS, DISPLAYING THOSE THAT ARE ON OR BEHIND SCHEDULE.

DOCUMENTATION: USER'S MANUAL, MAINTENANCE MANUAL, DEVELOPMENT SPECIFICATION, TEST PLAN

DEVELOPER: IBM
INFORMATION SOURCE: AIAA SURVEY OF SOFTWARE TOOLS

ACRONYM: SPEAR, TITLE: STRUCTURED PROGRAMMING EVALUATION AND AUTOPROCESSOR ROUTINE
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE
FEATURES: SUBJECT, CODE INPUT, JOVIAL, JOVIAL J38-2, USER OUTPUT, GRAPHICS, FLOW CHARTS, STATIC ANALYSIS, SCANNING, STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: PL/I
COMPUTER (OTHER HARDWARE): IBM 360/370
OS (OTHER SOFTWARE): OS/VS
TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO

TOOL SUMMARY: PROCESSES THE JOVIAL J38-2 LANGUAGE TO PRODUCE FLOWCHARTS DIRECTLY FROM THE CONTROL STRUCTURES OF THE LANGUAGE. J38-2 HAS TWO TYPES OF COMPLEXES, ONE OF WHICH FLOWCHARTS AND ONE OF WHICH DOES NOT RESULTING IN CORRECT AND READABLE LINE PRINTER PRODUCED FLOWCHARTS. ALSO PROVIDES LIMITED CHECKING FOR STRUCTURED PROGRAMMING CONSTRUCTS.

DOCUMENTATION: USER'S MANUAL, MAINTENANCE MANUAL
DEVELOPER: GENERAL DYNAMICS
CONTACT: L. C. KLOS, GENERAL DYNAMICS, P.O. BOX 746, FORT WORTH, TX, 76101, USA, 817-732-4811
INFORMATION SOURCE: AIAA SURVEY OF SOFTWARE TOOLS

ACRONYM: SPECLE/DARS, TITLE: SPECIFICATION LANGUAGE/DESIGN ANALYSIS AND REPORTING SYSTEM
CLASSIFICATION: REQUIREMENTS/DESIGN SPECIFICATION AND ANALYSIS
FEATURES: SUBJECT, VHDL INPUT, DESIGN SPECIFICATION, USER OUTPUT, GRAPHICS, DESIGN CHARTS, LISTINGS, STATIC ANALYSIS, COMPLETENESS CHECKING, SCANNING, STAGE OF DEVELOPMENT: IMPLEMENTED
COMPUTER (OTHER HARDWARE): IBM 360/370
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): AVAILABILITY OUTSIDE OF BCS MAY BE RESTRICTED
TOOL SUPPORTED: YES, TOOL SUPPORTED: BOEING COMPUTER SERVICES COMPANY

TOOL SUMMARY: SPECLE/DARS (SPECIFICATION LANGUAGE/DESIGN ANALYSIS AND REPORTING SYSTEM) IS A TOOL WHICH AIDS IN THE DESIGN DEVELOPMENT AND DOCUMENTATION PROCESS. SPECLE IS A DESIGN SPECIFICATION LANGUAGE USED TO DESCRIBE THE DESIGN OF A TARGET SYSTEM. DARS IS A COORDINATED SET OF ANALYSIS REPORT GENERATING PROGRAMS THAT PROCESS THE SPECLE DESCRIPTION OF THE DESIGN, THERE ARE CURRENTLY FOUR TOOLS WHICH MAKE UP THE DARS SYSTEM: (1) GADTR, (2) SAMMDF, (3) LAYOUT, AND (4) SPEDUC. SPECLE/DARS CAN CAPTURE A MODEL OF THE SYSTEM TO ANY LEVEL OF DETAIL, IT CAN THEN PRODUCE A VARIETY OF REPORTS AND GRAPHICS WHICH WILL ASSIST IN DESIGN, COMMUNICATION, AND REVIEW. THE USE OF SPECLE/DARS PROVIDES SEVERAL ADVANTAGES OVER MANUAL TECHNIQUES, INCLUDING: STANDARD DESIGN DESCRIPTIONS, ENFORCE PRECISION OF MEANING, ALLOW AUTOMATED DESIGN VERIFICATION, EASE OF MODIFICATION, AND PROVIDE INCREASED PROGRESS VISIBILITY.

DOCUMENTATION: USER'S MANUAL
REFERENCES: [BC879], ASSOCIATED SOFTWARE TOOLS CATALOG, BCS 10236, 700800
DEVELOPER: BOEING COMPUTER SERVICES COMPANY
CONTACT: GARY KAMPS, ASSOCIATED SOFTWARE TOOLS CATALOG, BCS SOFTWARE TOOLS CATALOG

ACRONYM: SPECTRUM-1, TITLE: A COMPLETE PROJECT MANAGEMENT SYSTEM
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE
FEATURES: SUBJECT, DATA INPUT, USER OUTPUT, USER-ORIENTED TEXT, REPORTS, GRAPHICS, STATIC ANALYSIS, MANAGEMENT, CHANGE CONTROL, DOCUMENTATION MANAGEMENT, PROJECT MANAGEMENT, SCHEDULING
STAGE OF DEVELOPMENT: IMPLEMENTED
TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): MARKETED PRODUCT
TOOL SUPPORTED: YES, TOOL SUPPORTED: SPECTRUM INTERNATIONAL INC.

TOOL SUMMARY: SPECTRUM-1 IS A COMPLETE PROJECT MANAGEMENT SYSTEM THAT CONTAINS AND INTEGRATES THE FOLLOWING FIVE ESSENTIAL ELEMENTS: (1) A MASTER SYSTEMS PLAN THAT DOCUMENTS THE NEW SYSTEM BACKLOG FOR THE NEXT TWO TO FIVE YEARS AND A MANAGEMENT REVIEW BOARD THAT SETS PRIORITIES IN THIS BACKLOG USING STANDARD BUSINESS PROCEDURES, (2) A PROJECT PLANNING AND CONTROL COMPONENT THAT CONTAINS ESTIMATING GUIDELINES, PROCEDURES FOR SCHEDULE PREPARATION AND TIME REPORTING, (3) A SET OF SYSTEM DESIGN GUIDELINES TO CAUSE NEEDED DESIGN TASKS TO BE PERFORMED IN A REQUIRED SEQUENCE, (4) A CONTROL PROCESS THAT CAUSES DOCUMENTATION TO BE PRODUCED AS A BYPRODUCT OF THE DESIGN EFFORT, (5) A CHANGE CONTROL METHOD THAT ENSURES THAT DOCUMENTATION IS KEPT UP-TO-DATE AND THAT CHANGES ARE THOROUGHLY TESTED BEFORE BEING PLACED INTO PRODUCTION.
SPREAD

SPREAD also computes the corresponding manpower levels required to accomplish the development effort. Total costs and manloading may be broken down and segregated either by individual routines, development phase spread costs to achieve a work milestone, activity, pricing category, WBS account, or virtually any combination thereof. Additionally, other cost factors such as Computed hours or other direct cost expenditures may be noted.


DEVELOPER: TRW

CONTACT: J. E. Green, TRW, One Space Park, Redondo Beach, CA, 90278, USA, 213-536-4012

INFORMATION SOURCE: TRW Software Tools Catalog

SPREAD

SPREAD computes the corresponding manpower levels required to accomplish the development effort. Total costs and manloading may be broken down and segregated either by individual routines, development phase spread costs to achieve a work milestone, activity, pricing category, WBS account, or virtually any combination thereof. Additionally, other cost factors such as Computed hours or other direct cost expenditures may be noted.

DOCUMENTATION: USER'S MANUAL


DEVELOPER: TRW

CONTACT: J. E. Green, TRW, One Space Park, Redondo Beach, CA, 90278, USA, 213-536-4012

INFORMATION SOURCE: TRW Software Tools Catalog

SPREAD

SPREAD computes the corresponding manpower levels required to accomplish the development effort. Total costs and manloading may be broken down and segregated either by individual routines, development phase spread costs to achieve a work milestone, activity, pricing category, WBS account, or virtually any combination thereof. Additionally, other cost factors such as Computed hours or other direct cost expenditures may be noted.

DOCUMENTATION: USER'S MANUAL


DEVELOPER: TRW

CONTACT: J. E. Green, TRW, One Space Park, Redondo Beach, CA, 90278, USA, 213-536-4012

INFORMATION SOURCE: TRW Software Tools Catalog

SPREAD

SPREAD computes the corresponding manpower levels required to accomplish the development effort. Total costs and manloading may be broken down and segregated either by individual routines, development phase spread costs to achieve a work milestone, activity, pricing category, WBS account, or virtually any combination thereof. Additionally, other cost factors such as Computed hours or other direct cost expenditures may be noted.

DOCUMENTATION: USER'S MANUAL


DEVELOPER: TRW

CONTACT: J. E. Green, TRW, One Space Park, Redondo Beach, CA, 90278, USA, 213-536-4012

INFORMATION SOURCE: TRW Software Tools Catalog

SPREAD

SPREAD computes the corresponding manpower levels required to accomplish the development effort. Total costs and manloading may be broken down and segregated either by individual routines, development phase spread costs to achieve a work milestone, activity, pricing category, WBS account, or virtually any combination thereof. Additionally, other cost factors such as Computed hours or other direct cost expenditures may be noted.

DOCUMENTATION: USER'S MANUAL


DEVELOPER: TRW

CONTACT: J. E. Green, TRW, One Space Park, Redondo Beach, CA, 90278, USA, 213-536-4012

INFORMATION SOURCE: TRW Software Tools Catalog

SPREAD

SPREAD computes the corresponding manpower levels required to accomplish the development effort. Total costs and manloading may be broken down and segregated either by individual routines, development phase spread costs to achieve a work milestone, activity, pricing category, WBS account, or virtually any combination thereof. Additionally, other cost factors such as Computed hours or other direct cost expenditures may be noted.

DOCUMENTATION: USER'S MANUAL


DEVELOPER: TRW

CONTACT: J. E. Green, TRW, One Space Park, Redondo Beach, CA, 90278, USA, 213-536-4012

INFORMATION SOURCE: TRW Software Tools Catalog
ACRONYM: SPRINT

CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING

FEATURES: SUBJECT, CODE INPUT, STRUCTURED FORTRAN, TRANSFORMATION, TRANSLATION, STRUCTURE PREPROCESSING, MACHINE OUTPUT, SOURCE CODE OUTPUT, FORTRAN.

STAGE OF DEVELOPMENT: IMPLEMENTED

IMPLEMENTATION LANGUAGE: FORTRAN

TOOL PORTABLE: YES, TOOL SIZE: CORE: 16,000

TOOL SUPPORTED: YES, TOOL SUPPORT: HONEYWELL

TOOL SUMMARY: THE SPTRAN CONVERTER ALLOWS A USER TO APPLY STRUCTURED PROGRAMMING CONCEPTS BY CODING IN A LANGUAGE THAT HAS THE STRUCTURED AND FREE-FORMAT FEATURES OF PL/I, BUT APPEARS SIMILARLY LIKE FORTRAN. THIS LANGUAGE, CALLED SPTRAN, WAS DESIGNED SUCH THAT IT COMBINES THOSE FEATURES OF FORTRAN AND PL/I MOST OFTEN USED FOR SCIENTIFIC PROGRAMMING APPLICATIONS. FREE-FORM SPTRAN INPUT IS TRANSLATED INTO ANSI FORTRAN AND THE ORIGINAL SPTRAN SOURCE CODE IS RETAINED AS FORTRAN REMARKS. SPTRAN ADVANTAGES ARE DESCRIBED AS FOLLOWS: (1) PROVIDES ACTUAL EXPERIENCE IN ENCODING AND EXECUTING STRUCTURED PROGRAMS, (2) ENABLES TIMELY CREATION OF FORTRAN PROGRAMS, (3) RESULTS IN ANNOTATED PROGRAM LISTINGS THAT ARE EASY TO READ AND CAN SUPPLEMENT DOCUMENTATION, AND (4) PROVIDES EARLY TRADITION TO STRUCTURED CONSTRUCTS AND PRINCIPLES.

DOCUMENTATION: USER MANUAL, INSTALLATION MANUAL


DEVELOPER: HONEYWELL

INFORMATION SOURCE: RADC-TR-80-13, INTERIM REPORT

ACRONYM: SREM

CLASSIFICATION: REQUIREMENTS/DESIGN SPECIFICATION AND ANALYSIS

FEATURES: SUBJECT, VHLL INPUT, REQUIREMENTS STATEMENT LANGUAGE, RSL, USER OUTPUT, DIAGNOSTICS, USER-ORIENTED TEXT, REPORTS, GRAPHICS, LISTINGS, STATIC ANALYSIS, COMPLETENESS CHECKING, CONSISTENCY CHECKING, MANAGEMENT, DATA BASE MANAGEMENT, DYNAMIC ANALYSIS, SIMULATION

STAGE OF DEVELOPMENT: IMPLEMENTED

IMPLEMENTATION LANGUAGE: FORTRAN, PASCAL

TOOL PORTABLE: YES

TOOL AVAILABLE: YES, PUBLIC DOMAIN: YES

RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): NO RESTRICTIONS WITHIN THE USA

TOOL SUPPORTED: YES, TOOL SUPPORT: TRW DEFENSE AND SPACE SYSTEMS GROUP, HUNTSVILLE, AL

TOOL SUMMARY: SOFTWARE REQUIREMENTS ENGINEERING METHODOLOGY (SREM) WAS DEVELOPED IN RESPONSE TO CONTINUING AND INCREASING DIFFICULTIES IN DEVELOPING COMPLEX, LARGE, REAL-TIME SOFTWARE FOR BALLISTIC MISSILE DEFENSE (BMD) SYSTEMS IN THE EARLY 1970S. SREM IS A FORMAL STEP-BY-STEP PROCESS FOR DEFINING DATA PROCESSING REQUIREMENTS, IT PROVIDES THE MEANS TO THOROUGHLY EVALUATE THE ADEQUACY OF SYSTEM REQUIREMENTS TOWARDS THE GOAL OF ATTAINING GOOD SOFTWARE SPECIFICATIONS FOR ANY SYSTEM PRIOR TO DESIGN AND CODING. ITS GOAL IS TO REDUCE SOFTWARE DEVELOPMENT COST AND SCHEDULE RISK, IN ADDITION TO THE STEP-BY-STEP REQUIREMENTS ENGINEERING TECHNIQUES, SREM INCLUDES A MACHINE-PROCESSABLE "ENGLISH-LIKE" REQUIREMENTS STATEMENT LANGUAGE (RSL) AND A REQUIREMENTS ENGINEERING VALIDATION SYSTEM (REV3S) TO AUTOMATICALLY PROCESS THE REQUIREMENTS STATEMENTS, AND TO PERFORM A WIDE RANGE OF ANALYSES AND SIMULATIONS ON ITS CENTRALIZED DATA BASE.

REFERENCES: [ALFO77], M. W. ALFORD, "A REQUIREMENTS ENGINEERING METHODOLOGY FOR REAL-TIME REAL-TIME PROCESSING"., IEEE TRANSACTIONS ON SOFTWARE ENGINEERING, VOL. SE-3, NO. 1, 770100 [ALFO78], M. W. ALFORD, "SOFTWARE REQUIREMENTS ENGINEERING METHODOLOGY (SREM)", CSMAC 78 PROCEEDINGS, PP. 352-359, 780000

DEVELOPER: TRW

CONTACT: ROBERT H. HOFFMAN, TRW INC., HUNTSVILLE FACILITY, 200 GOVERNORS DRIVE WEST, HUNTSVILLE, ALABAMA, 35805, USA, 205-637-3950

INFORMATION SOURCE: TOOL FAIR

ACRONYM: SREM

CLASSIFICATION: REQUIREMENTS/DESIGN SPECIFICATION AND ANALYSIS

FEATURES: REQUIREMENTS SIMULATION, SUBJECT, VHLL INPUT, REQUIREMENTS SPECIFICATION, USER OUTPUT, USER-ORIENTED TEXT, DOCUMENTATION, STATIC ANALYSIS, CONSISTENCY CHECKING, TRACKING

STAGE OF DEVELOPMENT: IMPLEMENTED

IMPLEMENTATION LANGUAGE: PASCAL

COMPUTER (OTHER HARDWARE): TI, CDC 6X00/7X00

TOOL SUMMARY: SREM IS A TOOL THAT PERFORMS A TECHNIQUE FOR DEFINING SPECIFYING SOFTWARE REQUIREMENTS. THE TECHNIQUE IS BUILT UPON A LANGUAGE, REQUIREMENTS STATEMENT LANGUAGE (RSL), READABLE BOTH BY A COMPUTER OR MAN, AND A SET OF TOOLS TERMINED COLLECTIVELY THE REQUIREMENTS ENGINEERING AND VALIDATION SYSTEM (REV3S), THE TOOLS PROVIDE FOR RETENTION OF ALL REQUIREMENTS IN A RELATIONAL DATA BASE FROM WHICH DOCUMENTATION, CONSISTENCY ANALYSES, AND SIMULATIONS MAY BE CONSTRUCTED AUTOMATICALLY, THE METHODOLOGY SYSTEMATICALLY DEVELOPS THE SPECIFICATION FROM SOURCE DOCUMENTATION AT THE
ACRONYM: SRTRAN, BASELINE, TITLE: BASELINE STANDARD VERSION OF SRTRAN PREPROCESSOR SYSTEM
CLASSIFICATION: PROGRAM CONSTRUCTION AND GENERATION
FEATURES: SUBJECT, CODE INPUT, STRUCTURED FORTRAN, SRTRAN, TRANSFORMATION, TRANSLATION, STRUCTURE PREPROCESSING, FORMATTING, MACHINE OUTPUT, SOURCE CODE OUTPUT, FORTRAN, USER OUTPUT, DIAGNOSTICS, LISTINGS, STATIC ANALYSIS, ERROR CHECKING.
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN
TOOL PORTABLE: YES
TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): MARKETED PRODUCT
TOOL SUPPORTED: YES, TOOL SUPPORT: SOFTWARE RESEARCH ASSOCIATES

SYSTENM LEVEL, DOCUMENTING OMISSIONS AND ERRORS OF THE SOURCE MATERIALS IN THE PROCESS. THE PRODUCED REQUIREMENTS ARE PROBABLY CONSISTENT, AND MAY BE VALIDATED AGAINST SYSTEM OBJECTIVES THROUGH THE GENERATED SIMULATION. THE ENTITIES ARE SUBJECT TO SYSTEMATIC MANAGEMENT THROUGH DEFINABLE AND MILESTONES SUPPORTED BY REVIEWS.

DOCUMENTATION: SYSTEM DESCRIPTION
REFERENCES: [ASD379], APPLIED SYSTEMS DESIGN SECTION, TRW DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE TOOLS SERIES, 790100

DEVELOPER: TRW
INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG

ACRONYM: SRIMP, TITLE: SOFTWARE REQUIREMENTS INTEGRATED MODELING PROGRAM
CLASSIFICATION: REQUIREMENTS/DESIGN SPECIFICATION AND ANALYSIS
FEATURES: SUBJECT, VHLL INPUT, MODEL SPECIFICATION, TRANSFORMATION, TRANSLATION, CONVERSION, EDITING, FORMATTING, MACHINE OUTPUT, VHLL OUTPUT, PSL USER OUTPUT, GRAPHICS, HIERARCHICAL TREE, ACTIVITY DIAGRAM, LISTINGS, STATIC ANALYSIS, DATA FLOW ANALYSIS, CONSISTENCY CHECKING
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN, PL/I
TOOL PORTABLE: YES
TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): CONTACT GRUMMAN AEROSPACE CORP. FOR DETAILS

TOOL SUPPORTED: YES, TOOL SUPPORT: GRUMMAN AEROSPACE CORP.

TOOL SUMMARY: THE SOFTWARE REQUIREMENTS INTEGRATED MODELING PROGRAM (SRIMP) AIDS THE CONCEPTUAL AND REQUIREMENTS DEFINITION PHASES OF THE SOFTWARE DEVELOPMENT PROCESS. THE SRIMP METHODOLOGY WAS DEVELOPED BY SYNTHESIZING THE BETTER FEATURES OF A NUMBER OF WIDELY ACCEPTED REQUIREMENTS TECHNIQUES, AUGMENTED WITH A NUMBER OF GRUMMAN DEVELOPED INNOVATIONS. THE RESULTANT SYSTEM PROVIDES THE USER WITH A FORMAL LANGUAGE TO EXPRESS SPECIFICATIONS THROUGH THE IDENTIFICATION OF OBJECTS AND ACCEPTABLE RELATIONSHIPS. TOP-DOWN DECOMPOSITION IS ENFORCED BY A METHOD WHICH RESULTS IN STRUCTURED INTRA SYSTEM INTERFACES, THUS MINIMIZING MODEL COMPLEXITY. HUMAN FACTORS ENGINEERING, WHICH PLAYED A MAJOR ROLE IN THE DEVELOPMENT OF SRIMP, WAS RESPONSIBLE FOR THE PRODUCTION OF AN EASY-TO-USE TOOL. THE USER IS LED THROUGH A SERIES OF PROMPTS WHERE CONSTANT MONITORING IS PERFORMED TO ENSURE THAT THE USER PROVIDES VALID RESPONSES.

DEVELOPER: GRUMMAN AEROSPACE CORP.

CONTACT: STEPHANIE WHITE, GRUMMAN AEROSPACE CORP., TECHNOLOGY DIVISION, 4015 4023, BETHPAGE, NY, 11714, USA, 516-575-6493

INFORMATION SOURCE: TOOL FAIR

ACRONYM: SSA, TITLE: SNEAK SOFTWARE ANALYSIS
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, USER OUTPUT, DIAGNOSTICS, STATIC ANALYSIS, AUDITING.
STAGE OF DEVELOPMENT: DESIGN
IMPLEMENTATION LANGUAGE: PL/I

TOOL SIZE: CODE: 300KB
COMPUTER (OTHER HARDWARE): IBM 360/370 (DISK: 19,8KB, TAPE (1) 9 TRACK)

OS (OTHER SOFTWARE): OS/VS
TOOL AVAILABLE: NO, PUBLIC DOMAIN: NO

TOOL SUMMARY: A COMPUTERIZED TECHNIQUE FOR IMPROVING THE EFFICIENCY AND RELIABILITY OF COMPUTER PROGRAMS, THE PROCESS INVOLVES THE USE OF MATHEMATICAL GRAPH THEORY, ELECTRICAL SNEAK THEORY, AND COMPUTERIZED SEARCH ALGORITHMS ON A COMPUTER PROGRAM TO IDENTIFY SOFTWARE PROBLEMS THAT MAY NOT BE FOUND IN TESTING. THESE TYPES OF PROBLEMS ARE
FORTRAN V PROGRAMS INTO STRUCTURED PROGRAM FORM, STRUCTRAN-2 PROVIDES THE ABILITY TO TRANSLATE UNSTRUCTURED CONTROL FORMS INTO LOGICALLY EQUIVALENT STRUCTURED CONTROL FORMS. DMA REPLACES FORTRAN CONTROL STATEMENTS WITH FIVE STRUCTURED COMMANDS: (1) IF... THEN... ELSE... END IF, (2) DO WHILE... END WHILE, (3) DO UNTIL... END UNTIL, (4) CASE OF... CASE... CASE ELSE... END CASE, (5) BLOCK (NAME)... END BLOCK (NAME).

DOCUMENTATION: FINAL TECH REPORT (28), STRUCTRAN I USERS MANUAL (21), STRUCTRAN I DESIGN IMP (14), STRUCTRAN II USERS MANUAL (16), STRUCTRAN II SYSTEM DESIGN IMP MANUAL (56).

DEVELOPER: GENERAL RESEARCH CORPORATION
INFORMATION SOURCE: COMPLETED SUBMISSION TO NBS

ACRONYM: STRUCT, TITLE: STRUCT
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, USER OUTPUT, DIAGNOSTICS, TABLES, STATIC ANALYSIS, AUDITING
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN
COMPUTER (OTHER HARDWARE): CDC 6600/7X00

TOOL SUMMARY: STRUCT AUDITS SOURCE CODE TRANSFER TABLE FOR COMPLIANCE TO STRUCTURED PROGRAMMING STANDARDS. ALLOWABLE SEGMENT STRUCTURES INCLUDE: SEQUENCE, IF - THEN - ELSE, DO UNTIL, DO WHILE, CASE AND ESCAPE. OUTPUTS INCLUDE THE PACE SEGMENT TRANSFER TABLE FOR EACH SUBROUTINE, AND MANAGEMENT SUMMARY OF ALL ROUTINES FOR THE WHOLE PROGRAM. INPUT FOR THIS PROGRAM IS NOT SOURCE CODE, BUT A SEGMENT TRANSFER TABLE GENERATED BY PACE. INPUT IS A SEGMENT TRANSFER TABLE PRODUCED BY THE PACE PROGRAM. NO MORE THAN 1000 TRANSFERS PER SUBROUTINE.

REFERENCES: [A80979], APPLIED SYSTEMS DESIGN SECTION, TRW DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS: CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE TOOLS SERIES, 790100

DEVELOPER: TRW, SEID SOFTWARE PRODUCT ASSURANCE
CONTACT: FRANK INGRASSIA, TRW, SEID SOFTWARE PRODUCT ASSURANCE, ONE SPACE PARK, REDONDO BEACH, CA 90278, USA, 213-536-3100
INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG

ACRONYM: STRUCTURE(S), TITLE: STRUCTURE(S)
CLASSIFICATION: REQUIREMENTS/DESIGN SPECIFICATION AND ANALYSIS
FEATURES: SUBJECT, VHLL INPUT, DESIGN SPECIFICATION, USER OUTPUT, GRAPHICS, DESIGN CHARTS, STATIC ANALYSIS, CROSS REFERENCE, ERROR CHECKING, STRUCTURE CHECKING, SCANNING
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: COBOL

COMPUTER (OTHER HARDWARE): IBM 360/370, UNIVAC 11XX, HONEYWELL 6XX

TOOL SUMMARY: STRUCTURE(S) IS A SOFTWARE PACKAGE THAT AUTOMATES THE PRODUCTION AND MAINTENANCE OF WARNER/DRUM DIAGRAMS. IT IS USED TO AID THE MANAGER, ANALYST, DESIGNER, PROGRAMMER, AND PROGRAM LIBRARIAN WITH THE CONTROL, ANALYSIS, DESIGN, CONSTRUCTION, MAINTENANCE, AND DOCUMENTATION OF SYSTEMS. VERSION 3.0 OFFERS THE ADDITION OF CONDITIONALS TO THE DIAGRAM, MORE USER-SPECIFIED OPTIONS, IMPROVED ERROR HANDLING, SUPPORT FOR IBM LASER PRINTER OUTPUT, AND FORMS FOR ASYNCHRONOUS TERMINALS. AND NEW SUPPORT MATERIAL. STRUCTURE(S) IS AVAILABLE IN IBM OS/COBOL, SOURCE CODE IS PROVIDED AND CAN BE MODIFIED WITH VENDOR ASSISTANCE WHEN POSSIBLE TO RUN ON MOST MACHINES HAVING A COBOL COMPILER.

DOCUMENTATION: USER'S GUIDE (125 PAGES), IMPLEMENTATION GUIDE (109 PAGES)

DEVELOPER: KEN ORR AND ASSOCIATES, INC.
CONTACT: ROBERT OTTE, KEN ORR AND ASSOCIATES, INC., 715 E. 8TH, TOPEKA, KS 66607, USA, 913-233-2349
INFORMATION SOURCE: COMPLETED SUBMISSION TO NBS

ACRONYM: STRUCTURING ENG, TITLE: STRUCTURING ENGINE
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, FORTRAN, ANSI FORTRAN, MACHINE OUTPUT, SOURCE CODE OUTPUT, FORTRAN, STATIC ANALYSIS, STRUCTURE CHECKING, TRANSFORMATION
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: PL/I

COMPUTER (OTHER HARDWARE): IBM 360/370

TOOL SUMMARY: THE STRUCTURING ENGINE IS A SET OF SOFTWARE TOOLS CAPABLE OF AUTOMATICALLY TRANSFORMING UNSTRUCTURED PROGRAMS INTO EQUIVALENT STRUCTURED PROGRAMS. THIS TRANSFORMATION ENHANCES THE CAPABILITIES OF PROGRAMMING TEAMS BY PROVIDING PROGRAMS THAT ARE EASIER TO UNDERSTAND, SAFER TO MODIFY, AND LESS COSTLY TO MAINTAIN. PROGRAMS OF ARBITRARY COMPLEXITY AND SIZE CAN BE PROCESSED PROVIDED THAT THEY CAN BE COMPILED WITHOUT ERRORS BY THE COMPILER FOR WHICH THEY WERE WRITTEN. THE STRUCTURED PROGRAMS ARE WRITTEN IN S-FORTRAN, AN EXTENSION AND IMPROVEMENT OF THE FORTRAN LANGUAGE DEVELOPED BY CFRG, INC. TO SUPPORT STRUCTURED CODING IN FORTRAN.

DOCUMENTATION: GENERAL INFORMATION MANUAL
REFERENCES: [A80879], APPLIED SYSTEMS DESIGN SECTION, TRW DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS: CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE TOOLS SERIES, 790100

DEVELOPER: CAINE, FABER GOORD, INC.
INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG
ACRONYM: SURGE 72
TITLE: SURGE 72 COBOL GENERATOR
CLASSIFICATION: PROGRAM CONSTRUCTION AND GENERATION
FEATURES: SUBJECT, VHL INPUT, PROGRAM SPECIFICATION, TRANFORMATION, SYNTHESIS, MACHINE OUTPUT, SOURCE CODE OUTPUT
STAGE OF DEVELOPMENT: IMPLEMENTED
DATE (YMD): 720000

ACRONYM: SURGE 72
TITLE: SURGE 72 COBOL GENERATOR
CLASSIFICATION: PROGRAM CONSTRUCTION AND GENERATION
FEATURES: SUBJECT, VHL INPUT, PROGRAM SPECIFICATION, TRANSFORMATION, SYNTHESIS, MACHINE OUTPUT, SOURCE CODE OUTPUT, COBOL
STAGE OF DEVELOPMENT: IMPLEMENTED
DATE (YMD): 720000

IMPLEMENTATION LANGUAGE: COBOL

TOOL AVAILABLE: YES, PUBLIC DOMAIN: YES
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): AVAILABLE FROM NTIS, PB 214-256

TOOL SUMMARY: THE SOURCE PROGRAM IS A NON-PROPRIETARY COMPUTER PROGRAM IN COBOL WHICH GENERATES APPROPRIATE DATA PROCESSING LOGIC AND PRODUCES COBOL SOURCE PROGRAMS FROM SIMPLE DESCRIPTIVE PARAMETER CARDS, AS A MACHINE-INDEPENDENT AND APPLICATION-INDEPENDENT SOFTWARE PACKAGE, IT PROVIDES PROGRAMMER AND NON-PROGRAMMER PERSONNEL WITH A SHORTCUT APPROACH TO FILE SORTING, SELECTIVE RETRIEVAL, AND TABULAR REPORT GENERATION, INCLUDING MULTIPLE LEVELS OF TOTALS AND A CAPABILITY FOR CERTAIN OTHER KINDS OF COMPUTATION.

DOCUMENTATION: TECHNICAL PAPER, USER'S GUIDE
REFERENCES: NTIS 801, NATIONAL TECHNICAL INFORMATION SERVICE, "A DIRECTORY OF COMPUTER SOFTWARE AND RELATED TECHNICAL REPORTS", PB 80-110232, 80000
INFORMATION SOURCE: NTIS

ACRONYM: SURVEYOR
TITLE: SET AND USE OF ROUTINE VARIABLES ANALYSIS PROGRAM
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, FORTRAN, USER OUTPUT, DIAGNOSTICS, LISTINGS, STATIC ANALYSIS, DATA FLOW ANALYSIS
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN
COMPUTER (OTHER HARDWARE): IBM 360/370, CDC 6600/7X00

TOOL SUMMARY: THE SURVEYOR PROGRAM IS AN AUTOMATED SOFTWARE VERIFICATION TOOL WHICH WILL PERFORM THE NECESSARY STRUCTURAL PATH ANALYSIS TO IDENTIFY ALMOST ALL POSSIBLE PATHS IN A ROUTINE, AND WHERE IN A ROUTINE VARIABLES ARE COMPUTED AND USED. BASED ON PATH AND USAGE DATA COLLECTED, THE SURVEYOR PROGRAM WILL EVALUATE EACH LOCAL VARIABLE TO IDENTIFY WHICH ARE COMPUTED AND NOT USED, WHICH ARE COMPUTED AND USED, WHICH ARE USED IN COMPUTATIONS WITHOUT BEING GIVEN A VALUE, THE PATH FOR WHICH A VARIABLE MAY BE USED WITHOUT BEING GIVEN A VALUE; EVALUATE EACH GLOBAL VARIABLE TO IDENTIFY EQUIVALENCES NOT REQUIRED, COMMON BLOCKS NOT REQUIRED, AND SUBROUTINE PARAMETER VARIABLES NOT REQUIRED; AND PROVIDE DETAILED INFORMATION ON THE AMOUNT AND TYPE OF SOFTWARE PROCESSED.

REFERENCES: [ASD79], DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS: CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE TOOLS SERIES, 790100
DEVELOPER: TRW, SOFTWARE TECHNOLOGY DEPARTMENT
CONTACT: R. L. MAITLEN, TRW, SOFTWARE TECHNOLOGY DEPARTMENT, ONE SPACE PARK, REDONDO BEACH, CA 90278, USA, 213-535-3480
INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG
SYMCRS

INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG

ACRONYM: SYMCRS
TITLE: SYSTEM MONITOR
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, USER OUTPUT, DIAGNOSTICS, LISTINGS, STATIC ANALYSIS, ERROR CHECKING, DYNAMIC ANALYSIS, TUNING, STAGE OF DEVELOPMENT: IMPLEMENTED
TOOL AVAILABLE: YES

TOOL SUMMARY: SYSTEM MONITOR IS A SOFTWARE SYSTEM WHICH HAS BEEN DEVELOPED TO PROVIDE ERROR DETECTION, ERROR CONTAINMENT, AND FUNCTIONAL RECOVERY SUPPORT TO APPLIED SOFTWARE AT THE PROGRAM MODULE AND SYSTEM LEVELS. IT IS COMPRISED OF FIVE COMPONENTS: INTERNAL PROCESS SUPERVISOR, EXTERNAL PROCESS SUPERVISOR, INTERFACE SUPERVISOR, SYSTEM MONITOR KERNEL, AND MAINTENANCE PROGRAM.


INFORMATION SOURCE: RADC-TR-80-15, INTERIM REPORT

ACRONYM: SYM-80
TITLE: AN AUTOMATED COBOL GENERATION SYSTEM
CLASSIFICATION: PROGRAM CONSTRUCTION AND GENERATION
FEATURES: SUBJECT, VHLL INPUT, PROGRAM SPECIFICATION, TRANSFORMATION, SYNTHESIS, MACHINE OUTPUT, SOURCE CODE OUTPUT, COBOL, COBOL 74
STAGE OF DEVELOPMENT: IMPLEMENTED
TOOL AVAILABLE: YES, PUBLIC DOMAIN
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): LICENSE REQUIRED

TOOL SUPPORTED: YES, TOOL SUPPORT: PHOENIX SYSTEMS, INC.

ACRONYM: TAFIRM, TITLE: TAFIRM
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, JOVIAL, JOVIAL IV, USER OUTPUT, TABLES, LISTINGS, STATIC ANALYSIS, MANAGEMENT, TEST DATA MANAGEMENT, DYNAMIC ANALYSIS, TRACING, PATH FLOW TRACING
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: JOVIAL
COMPUTER (OTHER HARDWARE): CDC 3XXX
TOOL SUMMARY: TAFIRM IS AN AUTOMATED SOFTWARE DEVELOPMENT TOOL WHICH WAS DESIGNED TO SHORTEN THE SCHEDULE FOR SOFTWARE DEVELOPMENT AND ALSO TO REDUCE COSTS THROUGH EARLY IDENTIFICATION OF LOGIC AND COMPUTATIONAL ERRORS. TAFIRM AUTOMATICALLY PERFORMS AND EVALUATES A SET OF TESTS ON OTHER COMPUTER PROGRAMS DURING THE DEVELOPMENT PHASE. IT PERFORMS A LOGIC PATH ANALYSIS, EXECUTES ALL THE SOURCE STATEMENTS, AND CHECKS THE ACCURACY OF EVERY COMPUTATION. IT OPERATES EFFECTIVELY WITH DETAILED INPUT DATA OR WITH NO TEST DATA SPECIFIED. TAFIRM CONSISTS OF THREE MODULES AND A DRIVER PROGRAM. THE TASKS WHICH ARE PERFORMED ARE: STRIP TEST DATA FROM THE SOURCE CODE TAPE; LOAD THE SPECIFIED CODING UNITS FROM THE MASTER TAPE AND EXECUTE THEM WITH THE CONTROLLED DATA FROM THE SOURCE CODE TAPE; PRINT REPORTS ON THE TEST RESULTS; AND OUTPUT A TABULATION OF ALL POSSIBLE LOGICAL PATHS INTO AND OUT OF THE TEST UNITS.
DOCUMENTATION: USER'S MANUAL
REFERENCES: [ASD85], APPLIED SYSTEMS DESIGN SECTION, TRW DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS: CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE TOOLS SERIES, 790100
DEVELOPER: TRW
INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG

ACRONYM: TAPS/AM, TITLE: TERMINAL APPLICATIONS PROCESSING SYSTEM/APPLICATIONS MANAGER
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE
FEATURES: SUBJECT, DATA INPUT, TRANSFORMATION, EDITING, FORMATTING, MACHINE OUTPUT, USER OUTPUT, USER-ORIENTED TEXT, DOCUMENTATION, STATIC ANALYSIS, CROSS REFERENCE, MANAGEMENT, FILES MANAGEMENT, DATA BASE MANAGEMENT, DYNAMIC ANALYSIS, SIMULATION, CONTROL, INTERACTIVE
STAGE OF DEVELOPMENT: IMPLEMENTED
TOOL PORTABLE: YES
OS (OTHER SOFTWARE): CS, OS/VS, DOS
TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO
RESTRICTIONS (COPIRIGHTS, LICENSES, ETC.): FOR LEASE, FOR SALE
TOOL SUMMARY: TAPS/AM (TERMINAL APPLICATIONS PROCESSING SYSTEM/APPLICATIONS MANAGER) IS A SUPPORT PRODUCT DESIGNED TO INCREASE PRODUCTIVITY IN DEVELOPING AND MAINTAINING

ACRONYM: TAPS/AM, TITLE: TERMINAL APPLICATIONS PROCESSING SYSTEM/APPLICATIONS MANAGER
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE
FEATURES: SUBJECT, DATA INPUT, TRANSFORMATION, EDITING, FORMATTING, MACHINE OUTPUT, USER OUTPUT, USER-ORIENTED TEXT, DOCUMENTATION, STATIC ANALYSIS, CROSS REFERENCE, MANAGEMENT, FILES MANAGEMENT, DATA BASE MANAGEMENT, DYNAMIC ANALYSIS, SIMULATION, CONTROL, INTERACTIVE
STAGE OF DEVELOPMENT: IMPLEMENTED
TOOL PORTABLE: YES
OS (OTHER SOFTWARE): CS, OS/VS, DOS
TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO
RESTRICTIONS (COPIRIGHTS, LICENSES, ETC.): FOR LEASE, FOR SALE
TOOL SUMMARY: TAPS/AM (TERMINAL APPLICATIONS PROCESSING SYSTEM/APPLICATIONS MANAGER) IS A SUPPORT PRODUCT DESIGNED TO INCREASE PRODUCTIVITY IN DEVELOPING AND MAINTAINING

ON-LINE SYSTEMS, THE SYSTEM PROVIDES ROUTINES TO PERFORM STANDARD FUNCTIONS COMMONLY PROGRAMMED INTO MOST ON-LINE APPLICATIONS. IT DRIVES THESE ROUTINES AND THE USER APPLICATION CODE THROUGH A GENERALIZED TABLE STRUCTURE CREATED FROM INFORMATION DERIVED FROM INPUT DATA SHEETS. TAPS FEATURES: MACRO LANGUAGE, DATA ELEMENT DICTIONARY, ON-LINE TESTING SIMULATOR FOR BATCH MODE, SCREEN AND FILE RECOVERY, AND AN ABILITY TO PROGRAM IN HIGHER-LEVEL LANGUAGES. STANDARDIZED FACILITIES INCLUDE: SIGN-ON/SIGN-OFF, APPLICATION SELECTION, TRANSACTION MENU SELECTION AND TERMINAL OPERATOR INTERRUPT, AUTOMATIC CAPABILITIES INCLUDE: DATA INQUIRY, COLLECTION, AND PAGING; SCREEN PROCESSING; DATA FORMAT EDITING; AND DOCUMENTATION. THE TAPS/AM PROGRAMMING OPTION PROVIDES A COMBINATION COMMUNICATIONS MONITOR AND APPLICATIONS MANAGER FOR IBM SYSTEMS.
DOCUMENTATION: USER'S MANUAL
DEVELOPER: DECISION STRATEGY CORP.
INFORMATION SOURCE: AAA SURVEY OF SOFT DEV TOOLS

ACRONYM: TATTLE, TITLE: TATTLE
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, JOVIAL, TRANSFORMATION, INSTRUMENTATION, MACHINE OUTPUT, SOURCE CODE OUTPUT, JOVIAL, USER OUTPUT, TABLES, LISTINGS, DYNAMIC ANALYSIS, COVERAGE ANALYSIS
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: JOVIAL
COMPUTER (OTHER HARDWARE): CDC 3XXX
TOOL SUMMARY: TATTLE MONITORS SOURCE STATEMENT EXECUTIONS DURING FREE-RUNNING END-TO-END TESTS. IT MAINTAINS EITHER (DEPENDING ON USER OPTION) A YES/NO INDICATOR WHICH IS SET WHEN A STATEMENT IS FIRST EXECUTED, OR ELSE AN ACTUAL CUMULATIVE EXECUTION FREQUENCY COUNT, FOR EVERY SOURCE STATEMENT IN EVERY PROGRAM SUBJECT TO TATTLE MONITORING, UPON REQUEST, IT WILL DISPLAY CUMULATIVE EXECUTION FREQUENCIES (OR YES/NO INDICATORS) AT THE STATEMENT LEVEL, I.E., NEXT TO EACH STATEMENT, IN A RE-LISTING OF THE SOURCE CODE.
DOCUMENTATION: DETAILED DESIGN DOCUMENTATION
REFERENCES: [ASD87], APPLIED SYSTEMS DESIGN SECTION, TRW DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS: CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE TOOLS SERIES, 790100
DEVELOPER: TRW
INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG

ACRONYM: TATTLE, TITLE: TATTLE
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, COBOL, TRANSFORMATION, INSTRUMENTATION, MACHINE OUTPUT, SOURCE CODE OUTPUT, COBOL, USER OUTPUT, TABLES, DYNAMIC ANALYSIS, COVERAGE ANALYSIS

INFO SOURCE: NTIS
STAGE OF DEVELOPMENT: IMPLEMENTED
DATE (YMMDD): 800700
COMPUTER (OTHER HARDWARE): IBM 360/370, UNIVAC 11XX
TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): LICENSE FEE INCLUDES ONE YEAR OF MAINTENANCE
TOOL SUPPORTED: YES, TOOL SUPPORT: SOFTWARE RESEARCH ASSOCIATES

TOOL SUMMARY: THE TEST COVERAGE ANALYSIS TOOL (TCAT) AUTOMATICALLY PROCESSES A COBOL PROGRAM AND PRODUCES A LOGICALLY Equivalent VERSION WHICH INCLUDES SPECIAL RUNTIME SUBROUTINE CALLS, THE TCAT SYSTEM OPERATES ON STANDARD COBOL FOR THE IBM AND UNIVAC ENVIRONMENTS AND IS ACCOMPANIED BY DETAILED DOCUMENTATION AS WELL AS A USER-SUPPORTIVE MAINTENANCE AND UPGRADE SUPPORT.

REFERENCES: [ICOMP81], "SYSTEM ANALYZES COBOL COVERAGE", COMPUTERWORLD, 800714

DEVELOPER: SOFTWARE RESEARCH ASSOCIATES
CONTACT: SOFTWARE RESEARCH ASSOCIATES, PO BOX 2432, SAN FRANCISCO, CA, 94122, USA, 415-957-1441
INFORMATION SOURCE: COMPUTERWORLD, SRA

ACRONYM: TCAT
TITLE: TCAT PROGRAM
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE

FEATURES: SUBJECT, CODE INPUT, USER OUTPUT, LISTINGS, STATIC ANALYSIS, COMPARISON,

STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: JOVIAL J4
COMPUTER (OTHER HARDWARE): CDC 33XX

TOOL DESCRIPTION: TCAT COMPARES AND SUMMARIZES THE DIFFERENCES BETWEEN TWO DATA BASES, WHERE ONE DATA BASE IS ON TAPE AND THE OTHER IS ACTIVE ON DISK. THE IMPORTANCE OF AUTOMATIC DATA COMPARISON IN EVALUATION OF THE EFFECT OF CHANGES (BOTH CODING CHANGES AND PARAMETRIC CHANGES) IS THAT IT SAVES MANY ENGINEERING MAN-HOURS OTHERWISE WASTED ON MANUAL DATA COMPARISONS. IT PROVIDES MORE ACCURATE COMPARISON THAN POSSIBLE MANUALLY, AND ENABLES THE ENGINEER TO FOCUS HIS TIME AND ATTENTION ON ANALYSIS OF THE DIFFERENCES REPORTED. THIS CAPABILITY IS NEEDED ON PRACTICALLY A DAILY BASIS DURING A SOFTWARE TEST PROCESS.

DOCUMENTATION: USER'S MANUAL
REFERENCES: [ICOMP81], "SYSTEM ANALYZES COBOL COVERAGE", COMPUTERWORLD, 800714

DEVELOPER: TRW, OPERATIONAL SOFTWARE OPERATIONS M
CONTACT: DAVID E. HEINE, TRW, OPERATIONAL SOFTWARE OPERATIONS M, ONE SPACE PARK, REDONDO BEACH, CA, 90278, USA, 213-535-3480
INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG

ACRONYM: TCAT
TITLE: TEST DATA EFFECTIVENESS MEASUREMENT SUBSYSTEM
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING FEATURES: SUBJECT, CODE INPUT, FORTRAN, FORTRAN V, INSTRUMENTATION, MACHINE OUTPUT, SOURCE CODE OUTPUT, FORTRAN, USER OUTPUT, TABLES, LISTINGS

STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN
COMPUTER (OTHER HARDWARE): CDC 6600/700

TOOL SUMMARY: TEST PREDICTOR IS A QUALITY ASSURANCE TOOL USED TO AUTOMATICALLY GENERATE TEST CASE INPUTS FOR FORTRAN COMPUTER PROGRAMS TO GUARANTEE THAT ALL BRANCHES WILL HAVE BEEN TESTED. IF A PARTICULAR BRANCH IS IMPOSSIBLE TO TEST, THE TEST PREDICTOR WILL SO INFORM THE USER. TEST PREDICTOR WILL AUTOMATICALLY GENERATE TEST INPUTS FOR A PARTICULAR PATH SELECTED BY THE USER, AND IT WILL IDENTIFY ALL NON-REPEATED PATHS THROUGH A BLOCK OF SOFTWARE, USING THIS TOOL, QUALITY ASSURANCE CAN INDEPENDENTLY TEST SELECTED ROUTINE, AND CAN ANALYZE SEQUENCES OF EVENTS LEADING TO FAULFUL ERRONEOUS CONDITIONS. INPUTS FOR TEST PREDICTOR ARE THE USER'S FORTRAN SOURCE CODE AND ORIGUTALLY, ANY SPECIFIED VARIABLE DATA RANGES, OUTPUTS
TEST PREDICTOR

INCLUDE A LISTING OF THE USER'S SOURCE CODE ANNOTATED WITH SEGMENT NUMBERS, A LIST OF ALL NON-REPEETITIVE PATHS THROUGH EACH ROUTINE, AND A LIST OF THE MINIMUM NUMBER OF TEST CASES NECESSARY TO EXECUTE ALL SEGMENTS AND THE ASSOCIATED DATA RANGE FOR EACH INPUT VARIABLE.

DOCUMENTATION: NONE

REFERENCES: [ASD55], APPLIED SYSTEMS DESIGN SECTION, TRW DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS", CATALOGUE AND RECOMMENDATIONS, TRW AUTOMATED SOFTWARE TOOLS SERIES, 7901000

DEVELOPER: TRW, SEID SOFTWARE PRODUCT ASSURANCE

CONTACT: FRANK INGRASSIA, TRW, SEID SOFTWARE PRODUCT ASSURANCE, ONE SPACE PARK, REDONDO BEACH, CA, 90276, USA, 213-536-3140

INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG

ACRONYM: TEVE=1, TITLE: A SOFTWARE SYSTEM FOR PROGRAMS TESTING AND VERIFICATION

CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING FEATURES: SUBJECT, CODE INPUT, IFTRAN, USER OUTPUT, LISTINGS, DYNAMIC ANALYSIS, SYMBOLIC EXECUTION,

STAGE OF DEVELOPMENT: IMPLEMENTED, DATE (YYMDDD): 810000

IMPLEMENTATION LANGUAGE: LISP 1.5

COMPUTER (OTHER HARDWARE): DEC PDP-11 (64K CORE MEMORY), BURROUGHS B6700

TOOL AVAILABLE: YES

TOOL SUMMARY: TEVE=1 IS A SOFTWARE SYSTEM INTENDED TO BE USED FOR VALIDATION OF WELL-STRUCTURED PROGRAMS WRITTEN IN AN ALGOL-LIKE LANGUAGE WHICH ALLOWS THE USE OF ONLY THE THREE BASIC CONSTRUCTS OF STRUCTURED PROGRAMMING (ASSIGNMENT, WHILE-DO, IF-THEN-ELSE) PLUS INPUT-OUTPUT STATEMENTS, BOTH THE PROGRAM USED TO DERIVE TEST CASES AND THE PROGRAM USED TO AID PROGRAM PROOF ARE BASED ON THE "WEakest CONDITION" THEORY APPLIED TO WELL-STRUCTURED PROGRAMS IN ORDER TO DERIVE "PATH-PREDICATES", LOGICAL REQUIREMENTS WHICH MUST BE FILLED BY TEST DATA IF THE VERIFICATION OF THE PROGRAM EXECUTION IS TO FOLLOW A PARTICULAR PATH, THE SYSTEM IS INTENDED FOR USE AS PART OF THE TESTING AND VERIFICATION OF THE SOFTWARE FOR A REACTOR SAFETY SYSTEM.

DOCUMENTATION: SYSTEM DESCRIPTION (8)

REFERENCES: [BLOBO97], S. BOLOGNA, "TEVE=1: A SOFTWARE SYSTEM FOR PROGRAMS TESTING AND VERIFICATION", CONGRESSO AICA '79, 791000

DEVELOPER: CNEN CSN=CASACCIA (ITALY), RISO NATIONAL LABORATORY (DENMARK)

CONTACT: SANDRO BOLOGNA, CNEN CSN=CASACCIA, C. P. 2400, ROMA, A. D., ITALY, 1-300-6948

J. R. TAYLOR, RISO NATIONAL LABORATORY, RISO, DENMARK, INFORMATION SOURCE: CONGRESSO AICA '79

ACRONYM: TFA, TITLE: TRACE FREQUENCY ANALYZER

CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING

FEATURES: SUBJECT, CODE INPUT, COBOL, TRANSLATION, INSTRUMENTATION, MACHINE OUTPUT, SOURCE CODE OUTPUT, COBOL, USER OUTPUT, TABLES, LISTINGS, DYNAMIC ANALYSIS, COVERAGE ANALYSIS, TIMING,

STAGE OF DEVELOPMENT: IMPLEMENTED

IMPLEMENTATION LANGUAGE: COBOL

TOOL SIZE: CORE: 60000

COMPUTER (OTHER HARDWARE): CDC CYBER

QS (OTHER SOFTWARE): SCOPE 3.4

TOOL AVAILABLE: YES

TOOL SUMMARY: THIS PROGRAM USES THE COBOL TRACE FILE TO ANALYZE EACH PARAGRAPH SHOWING THE NUMBER OF TIMES IT WAS ENTERED AND THE EXECUTION TIME OF THE PARAGRAPH. THE TRACE FILE IS PRODUCED WHEN THE TRACE OPTION IS USED AT COMPILATION TIME, THE PARAGRAPHS WHICH ARE HIGH IN FREQUENCY AND TIME ACCUMULATION WILL ENABLE THE PROGRAMMER TO DIRECT OPTIMIZING EFFORTS TO THOSE PARAGRAPHS THAT MIGHT NEED IT THE MOST.

DOCUMENTATION: TECHNICAL PAPER

DEVELOPER: USAF/ALC

CONTACT: BILL SHIRLEY, USAF/ALC, SM=ALC/ACDAB, BLDG 2608, MCCLELLAN AFB, CA, 95652, USA, 916-649-3642

INFORMATION SOURCE: AIAA SURVEY OF SOFT DEV TOOLS

ACRONYM: THE ENGINE, TITLE: COBOL STRUCTURING ENGINE

CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING FEATURES: SUBJECT, CODE INPUT, COBOL, TRANSFORMATION, EDITING, PRODUCTION, INSTRUMENTATION, FORMATTING, MACHINE OUTPUT, SOURCE CODE OUTPUT, COBOL, USER OUTPUT, LISTINGS, STATIC ANALYSIS, ERROR CHECKING, STRUCTURE CHECKING, SCANNING, DYNAMIC ANALYSIS, COVERAGE ANALYSIS, TRACING,

STAGE OF DEVELOPMENT: IMPLEMENTED

IMPLEMENTATION LANGUAGE: COBOL

TOOL PORTABLE: YES, TOOL SIZE: 400K

TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.), SERVICE CONTRACT

TOOL SUPPORT: YES, TOOL SUPPORT: CATALYST CORPORATION

TOOL SUMMARY: STRUCTURED RETROFIT IS A SOFTWARE TOOLS BASED METHODOLOGY. IT INTRODUCES TODAY'S STRUCTURED PROGRAMMING METHODOLOGIES INTO EXISTING SYSTEMS IN ORDER TO METE FUTURE DEMANDS FOR CHANGE, ITS FOCUS IS ON APPLICATION SYSTEMS MAINTENANCE. TODAY MAINTENANCE CONSUMES 70% OF THE TYPICAL MIS BUDGET. THE STRUCTURED PROGRAMMING METHODOLOGIES CUT MAINTENANCE COSTS BY A RATIO OF 3 TO 1. STRUCTURED RETROFIT ALLOWS AN ORGANIZATION TO INTRODUCE THE STRUCTURED METHODOLOGIES INTO SYSTEMS DEVELOPED WITHOUT THEM IN A STRAIGHTFORWARD, FAST, AND RELIABLE MANNER. RETROFIT TAKES WORKING COBOL PROGRAMS—WRITTEN AND MAINTAINED BY VARIOUS PROGRAMMERS IN A VARIETY OF STYLES—AND RESTRUCTURES AND REFORMATS THEM MECHANICALLY. IT GIVES THE CODE A CONSISTENT STRUCTURE AND FORMAT, MAKING IT MORE READABLE, UNDERSTANDABLE, AND MAINTAINABLE.
THE ENGINE

APPLICATION OF THE STRUCTURED PROGRAMMING CONSTRUCTS TO EXISTING "SPAGHETTI" CODE.

REFERENCES: (LYON80), M.J. Lyons, "STRUCTURED RETROFIT - 1980", PROCEEDINGS OF SHARE 55, ATLANTA, GA, 800600
(MILL80), J. C. Miller, "STRUCTURED RETROFIT", TECHNIQUES OF PROGRAM AND SYSTEMS MAINTENANCE, ETHNOTECH, 800000
(LYON81), J. J. Lyons, "SALVAGING YOUR SOFTWARE ASSET", PROCEEDINGS OF THE NATIONAL COMPUTER CONFERENCE, 810500

DEVELOPERS: CATALYST CORP.
CONTACT: MICHAEL J. Lyons, CATALYST CORPORATION, 433 SOUTH KENSINGTON, LA GRANGE, IL, 60525, USA, 312-352-5422

INFORMATION SOURCE: TOOL FAIR

ACRONYM: TIDY, TITLE: TIDY
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE
FEATURES: SUBJECT, CODE INPUT, FORTRAN, TRANSFORMATION, FORMATTING, USER OUTPUT, LISTINGS,
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN
COMPUTER (OTHER HARDWARE): IBM 360/370, CDC 6600/7600
TOOL AVAILABLE: YES, PUBLIC DOMAIN: YES

TOOL SUMMARY: THIS PROGRAM PROVIDES A SIMPLE WAY TO CLEAN UP FORTRAN PROGRAMS BY MAKING ALL STATEMENT NUMBERSSEQUENTIAL, LABELING AND SEQUENCING THE CARDS.

DOCUMENTATION: USER'S MANUAL
DEVELOPERS: BRUNSWICK DEFENSE DIV.
CONTACT: JAMES N. CHURCHYARD, BRUNSWICK DEFENSE DIV., 3333 HARBOR BLVD., COSTA MESA, CA, 92626, USA, 714-546-6830
INFORMATION SOURCE: AIAA SURVEY OF SOFT DEV TOOLS

ACRONYM: TIMECS, TITLE: TASK INTERNAL MICRO EXECUTION CHRONOGRAPH SYSTEM
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, DATA INPUT, USER OUTPUT, TABLES, STATIC ANALYSIS, STATISTICAL ANALYSIS, DYNAMIC ANALYSIS, TUNING, TIMING,
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: COBOL, COMPASS
TOOL SIZE: CORE: 5K
COMPUTER (OTHER HARDWARE): CDC 3XXX
OS (OTHER SOFTWARE): MASTER
TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO

RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): FOR SALE
TOOL SUMMARY: TIMECS (TASK INTERNAL MICRO EXECUTION CHRONOGRAPH SYSTEM IS A JOB PERFORMANCE EVALUATION SYSTEM. TIMECS IDENTIFIES INEFFICIENT TASKS WITHIN A JOB AND THE INEFFICIENT ROUTINES WITHIN THAT TASK. A STATISTICS EXTRACTION SYSTEM AND ANALYSIS/REPORT SYSTEM ARE THE 2 BASIC ELEMENTS OF TIMECS. THE STATISTICS EXTRACTION SYSTEM IS A SOFTWARE MONITOR THAT COLLECTS WAIT STATUS, EXECUTION TIMES AND TASK OVERLAY INITIATION, AND PROVIDES STATISTICS FOR ALL TASKS DURING THE JOB SAMPLE PERIOD. THE ANALYSIS/REPORT SYSTEM GENERATES 4 REPORT FORMATS FROM A

TAPE GENERATED BY THE STATISTICS SYSTEM. 2 JOB SUMMARY REPORTS AND 2 TASK REPORTS FOR EACH TASK AND OVERLAY ARE GENERATED.

DOCUMENTATION: USER'S MANUAL
DEVELOPER: TECHANALYSIS CORPORATION
INFORMATION SOURCE: AIAA SURVEY OF SOFT DEV TOOLS

ACRONYM: TIMECS, TITLE: PROGRAM EXECUTION TIMER
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, USER OUTPUT, GRAPHICS, TABLES, DYNAMIC ANALYSIS, TIMING,
STAGE OF DEVELOPMENT: IMPLEMENTED
DATE (YMMDD): 770000
IMPLEMENTATION LANGUAGE: FORTRAN
TOOL SIZE: 2000 STATEMENTS
COMPUTER (OTHER HARDWARE): IBM 360/370
OS (OTHER SOFTWARE): 03/MVS

TOOL AVAILABLE: YES

RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): AVAILABLE THROUGH FEDERAL SOFTWARE EXCHANGE CENTER, FSEC=77/3373

TOOL SUMMARY: TIMECS IS A SYSTEM FOR ANALYZING AN EXECUTING PROGRAM'S UTILIZATION OF CPU TIME. TIMECS TAKES A PROGRAM, WITH ITS DATA AND PRODUCE A CPU TIME ACCOUNTING BY STORAGE LOCATION FOR ITS EXECUTION ON THAT PARTICULAR DATA. PRINTED AND PLOTTED REPORTS ARE PRODUCED THAT SHOW THE AMOUNT OF CPU TIME SPENT IN EACH AREA OF THE PROGRAM'S STORAGE, THE NUMBER AND SIZE OF THE AREAS TO BE MONITORED ARE DETERMINED BY USER INPUT.

REFERENCES: (FSEC80), GENERAL SERVICES ADMINISTRATION/NATIONAL TECHNICAL INFORMATION SERVICE, "FEDERAL SOFTWARE EXCHANGE CATALOG", GSA/ADTS/C-80/1, PBB80-904001, 800100
CONTACT: GSA FEDERAL SOFTWARE EXCHANGE, 2 SKYLINE PL (11TH FL), 3200 LEESBURG PK, FALLS CHURCH, VA, 22041, 703-756-2610

INFORMATION SOURCE: FEDERAL SOFTWARE EXCHANGE CATALOG

ACRONYM: TOOLPACK, TITLE: AN INTEGRATED COLLECTION OF TOOLS FOR MATHEMATICAL SOFTWARE
CLASSIFICATION: SOFTWARE SUPPORT SYSTEM/PROGRAMMING ENVIRONMENT
FEATURES: SUBJECT, DATA INPUT, CODE INPUT, FORTRAN, TRANSFORMATION, EDITING, FORMATTING, MACHINE OUTPUT, SOURCE CODE OUTPUT, FORTRAN DATA OUTPUT, USER OUTPUT, DIAGNOSTICS, TABLES, LISTINGS, STATIC ANALYSIS, DATA FLOW ANALYSIS, CROSS REFERENCE, MANAGEMENT, FILES MANAGEMENT, STRUCTURE CHECKING, DYNAMIC ANALYSIS, COVERAGE ANALYSIS, TRACING,
STAGE OF DEVELOPMENT: DESIGN
DATE (YMMDD): 830000
IMPLEMENTATION LANGUAGE: FORTRAN

TOOL PORTABILITY: NO
PUBLIC DOMAIN: YES

TOOL SUMMARY: TOOLPACK IS AN APPROACH BEING TAKEN TO CONFIGURE A SET OF TOOL CAPABILITIES WHOSE GOAL IS THE SUPPORT OF MATHEMATICAL SOFTWARE DEVELOPMENT MAINTENANCE
STAGE OF DEVELOPMENT: IMPLEMENTED, DATE (YYMMDD): 790900
IMPLEMENTATION LANGUAGE: FORTRAN
TOOL PORTABLE: YES
TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO
RESTRICTIONS: (COPYRIGHTS, LICENSES, ETC.): LICENSE AVAILABLE THROUGH TASC
TOOL SUPPORTED: YES, TOOL SUPPORT: TASC
TOOL SUMMARY: Trailblazer is a family of tools for improving the quality of software and reducing the cost of software development and maintenance. Tools within the Trailblazer family operate by analyzing the structure of source programs and modifying the source to report on information about operating behavior of the programs. Trailblazer tools are useful to software developers, software development managers, quality assurance specialists, and security managers. Trailblazer testing analysis provides summary and detailed pictures of how thoroughly a set of tests has exercised software. The detailed information can be used to increase testing coverage by creating tests that execute unexecuted steps or to build a comprehensive regression test library that can be selectively executed whenever changes are made to the system. The Trailblazer change analyzer reports on differences between an old and new version of a system's source code.
(HOLTZ), M. A., HOLTHOUSE, M. J. HATCH, "Experience with Automated Testing Analysis", Computer, 790600
DEVELOPER: TASC
CONTACT: MARK A. HOLTHOUSE, TASC, ONE JACOB WAY, READING, MA, 01867, USA, 617-944-6850
INFORMATION SOURCE: COMPLETED SUBMISSION TO NBS
ACRONYM: TRANSFOR, TITLE: TRANSLATOR FOR STRUCTURED FORTRAN
CLASSIFICATION: REQUIREMENTS/DESIGN SPECIFICATION AND ANALYSIS
FEATURES: SUBJECT, CODE INPUT, STRUCTURED FORTRAN, VHIL INPUT, STRUCTURED LANGUAGE, TRANSFORMATION, TRANSLATION, STRUCTURE PREPROCESSING, MACHINE OUTPUT, SOURCE CODE OUTPUT, FORTRAN, USER OUTPUT, DIAGNOSTICS.
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN
TOOL SUMMARY: TRANSFOR (TRANSLATOR FOR STRUCTURED FORTRAN) is a pre-processor that extends the FORTRAN language for structured programming. It allows the programmer to write efficient, structured code free of GO TO statements. The use of statement numbers is eliminated and the nesting of control structures enforced. TRANSFOR and FORTRAN
TRANSFOR STATEMENTS ARE COMPLETELY COMPATIBLE AND CAN BE FREELY INTERMIXED. THE RESULTANT SOFTWARE IS MORE READABLE BECAUSE OF THE AUTOMATIC INDENTATION OF CONTROL STATEMENTS WHICH NESTING IS DEPICTED. SNEAK PATHS AND UNDESIRED EXECUTION SEQUENCES ARE RARE, SINCE THE PROGRAMMER IS ABLE TO SEE PRIMARILY STRAIGHTFORWARD LOGIC.
REFERENCES: (ADSO97), APPLIED SYSTEMS DESIGN SECTION, TRW DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS, CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE TOOLS CENTER, 790100
DEVELOPER: BCS, INC.
INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG
ACRONYM: TRMS, TITLE: TECHNICAL REQUIREMENTS MANAGEMENT SYSTEM
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE
FEATURES: SUBJECT, DATA INPUT, USER OUTPUT, TABLES, STATIC ANALYSIS, MANAGEMENT, PROJECT MANAGEMENT
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: COBOL
TOOL SIZE: CORE: 56K
COMPUTER (OTHER HARDWARE): IBM 360/370
OS (OTHER SOFTWARE): OS, OS/V3
TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO
TOOL SUMMARY: TRMS is designed to monitor critical milestones and to provide documentation at the project level. It allows for top-down event/activity identification and scheduling, maintains data on all critical documents, monitors incorporation of design changes, and provides an "as designed" parts list as well as a comparison of the "as designed - as built" parts lists. The system is composed of six modules: requirements identification, planning and scheduling, document status, parts data, configuration verification, and as designed - as built. Each uses a single data base and has the capability of stand-alone operation. The modules are designed to operate in an on-line mode using video display equipment, are configured to have batch reporting and batch back-up capabilities, and are constructed to meet the applicable military specifications required of DOD contractors.
DEVELOPER: MARTIN MARIETTA DATA SYSTEMS
INFORMATION SOURCE: AIAA SURVEY OF SOFT DEV TOOLS
ACRONYM: TRWPLT, TITLE: PROGRAM TRWPLT
CLASSIFICATION: SOFTWARE MANAGEMENT, CONTROL, AND MAINTENANCE
FEATURES: SUBJECT, DATA INPUT, USER OUTPUT, GRAPHICS, STATIC ANALYSIS, MANAGEMENT, DOCUMENTATION MANAGEMENT,
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN V
COMPUTER (OTHER HARDWARE): UNIVAC 11XX

TOOL SUMMARY: TRWPLT IS A GENERAL PLOT PROGRAM -- EASY TO
USE, ABLE TO RAPIDLY PLOT LARGE AMOUNTS OF DATA, ABLE TO
PRODUCE HIGH QUALITY VULNERABLE CALCOPM OR MACHINE
PLOTS. TRWPLT IS A REAL TIMESAVER, IT IS A VALUABLE AND
POWERFUL TOOL FOR PROJECTS WHICH MUST EXHIBIT AND ANALYZE
LARGE AMOUNTS OF DATA AND PREPARE PLOT-TYPE DISPLAYS FOR
INCLUSION IN DOCUMENTATION.

DOCUMENTATION: USER'S MANUAL
REFERENCES: [ASD879], APPLIED SYSTEMS DESIGN SECTION, TRW
DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS:
CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE
TOOLS SERIES, 790100
DEVELOPER: TRW HOUSTON FACILITY
INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG

ACRONYM: TSA/PPE, TITLE: TOTAL SYSTEM ANALYZER/PROBLEM
PROGRAM EVALUATOR
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, USER OUTPUT, DIAGNOSTICS,
LISTINGS, DYNAMIC ANALYSIS, TUNING, RESOURCE UTILIZATION,
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN ASSEMBLY
COMPUTER (OTHER HARDWARE): IBM 360/370
OS (OTHER SOFTWARE): OS/VS/VS/OS/MVS
TOOL AVAILABLE: YES, PUBLIC DOMAIN: NO

TOOL SUMMARY: TSA/PPE TOTAL SYSTEM ANALYZER/PROBLEM PROGRAM
EVALUATOR PERFORMS SYSTEM ANALYSIS, WORKLOAD ANALYSIS, AND
PROBLEM ANALYSIS TO IDENTIFY AREAS OF HIGH CPU UTILIZATION
FOR SYSTEM AND PROGRAM MODULES, THE PACKAGE CONSISTS OF
FIVE PROGRAM: THE TSA EXTRACTOR, THE TSA ANALYZER, THE TSA
WORKLOAD ANALYZER, THE PPE EXTRACTOR, AND THE PPE ANALYZER.
THE TSA EXTRACTOR SAMPLES THE ACTIVITY OF SYSTEM AND
PROGRAM MODULES SIMULTANEOUSLY, THE TSA ANALYZER TABULATES
THE COLLECTED DATA INTO A MODULE USAGE SUMMARY REPORT
LISTING ALL THE MODULES MEASURED AND THEIR CORRESPONDING
CPU USAGE AND ACTIVITY, THE PPE ANALYZER UTILIZES THE DATA
COLLECTED BY THE EXTRACTOR AND PRODUCES A SET OF
INSTRUCTION LOCATIONS HISTOGRAMS FOR ANY PROGRAM OR SYSTEM
MODULES SPECIFIED BY THE USER.

DEVELOPER: Boole and Babbage
INFORMATION SOURCE: AIAA SURVEY OF SOFT DEV TOOLS

ACRONYM: UCA, TITLE: UNITS CONSISTENCY ANALYSIS
CLASSIFICATION: SOURCE PROGRAM ANALYSIS AND TESTING
FEATURES: SUBJECT, CODE INPUT, FORTRAN, USER OUTPUT,
DIAGNOSTICS, LISTINGS, STATIC ANALYSIS, UNITS ANALYSIS,
SCANNING,

STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: FORTRAN
COMPUTER (OTHER HARDWARE): CDC 6600/7600

TOOL SUMMARY: THE UNITS CONSISTENCY ANALYSIS PROGRAM IS A
QUALITY ASSURANCE TOOL WHICH SCANS A FORTRAN PROGRAM AND
ITS ASSOCIATED DATA BASE (FOR ACCESS TO VARIABLE UNITS
DEFINITION) TO DETERMINE IF THE UNITS OF THE VARIABLES USED
IN THE PROGRAM ARE CONSISTENT WITHIN THE SOURCE STATEMENTS.
THE PROGRAM SYMBOILICALLY INTERPRETS THE EQUATIONS
REFERRING THE VARIABLES TO ASSURE THAT THE UNITS ARE CORRECT
AND CONSISTENT WITHIN THE EQUATIONS, FOR EXAMPLE,
IN THE C = A * B IF VARIABLES A AND B WERE DEFINED TO BE IN
UNITS OF FEET, VARIABLE C OUGHT TO BE DEFINED IN UNITS OF
SQUARE FEET. IF THE DIAGNOSTIC WOULD BE PRINTED, OPTIONAL
INPUTS ALLOW THE USER TO OVERRIDE THE THE UNITS SPECIFICATION
IN THE DATA BASE, AND ADD UNITS DEFINITIONS FOR VARIABLES NOT FOUND IN THE DATA BASE.

DOCUMENTATION: USER'S MANUAL, REQUIREMENTS MANUAL, DESIGN
MANUALS
REFERENCES: [ASD879], APPLIED SYSTEMS DESIGN SECTION, TRW
DEFENSE AND SPACE SYSTEMS GROUP, "SOFTWARE TOOLS:
CATALOGUE AND RECOMMENDATIONS", TRW AUTOMATED SOFTWARE
TOOLS SERIES, 790100
DEVELOPER: TRW, SEID SOFTWARE PRODUCT ASSURANCE
COMPANY: P-BANK ENTERPRISE, TRW, PSEID SOFTWARE PRODUCT
ASSURANCE, ONE SPACE PARK, REDONDO BEACH, CA, 90278, USA,
213-536-3140
INFORMATION SOURCE: TRW SOFTWARE TOOLS CATALOG

ACRONYM: UCSF P-SYSTEM, TITLE: A PORTABLE SOFTWARE
ENVIRONMENT FOR SMALL COMPUTERS
CLASSIFICATION: PROGRAM CONSTRUCTION AND GENERATION
FEATURES: SUBJECT, CODE INPUT, FORTRAN, PASCAL, FORTRAN 77,
TRANSFORMATION, TRANSLATION, COMPILATION, EDITING, USER
OUTPUT, LISTINGS,
STAGE OF DEVELOPMENT: IMPLEMENTED
IMPLEMENTATION LANGUAGE: PASCAL UCSF
TOOL PORTABLE: YES

TOOL AVAILABLE: YES, PUBLIC DOMAIN: YES
RESTRICTIONS (COPYRIGHTS, LICENSES, ETC.): LICENSE
TOOL SUPPORTED: YES, TOOL SUPPORT: SOFTECH MICROSYSTEMS

TOOL SUMMARY: THE UCSF P-SYSTEM IS A STAND-ALONE PROGRAM
DEVELOPMENT AND EXECUTION ENVIRONMENT FOR SMALL COMPUTERS,
ITS FACILITIES INCLUDE TEXT EDITORS AND FILE MANAGEMENT
UTILITIES, AS WELL AS COMPILERS (UCSF PASCAL, FORTRAN 77
AND BASIC), MACRO CROSS-ASSEMBLERS AND A LINKAGE EDITOR,
THE P-SYSTEM IS A PORTABLE SOFTWARE ENVIRONMENT
INDEPENDENT FROM THE HOST PROCESSOR AND ITS PERIPHERAL
DEVICES, THE FOUNDATION FOR THIS PORTABILITY IS THE UCSF
P-MACHINE, IT IS A SIMPLE IDEALIZED STACK COMPUTER WHICH
CAN BE IMPLEMENTED BY DIRECT HARDWARE SUPPORT OR BY AN
INTERPRETER EXECUTING IN THE MACHINE LANGUAGE OF THE HOST
COMPUTER, THE COMPLETE SOFTWARE ENVIRONMENT IS WRITTEN IN UCSF PASCAL,
COMPILLED TO P-CODE AND THEN EXECUTED BY THE P-MACHINE.
Computer Science and Technology: Software Development Tools

Raymond C. Houghton, Jr.

NATIONAL BUREAU OF STANDARDS
DEPARTMENT OF COMMERCE
WASHINGTON, D.C. 20234

Same as above

Library of Congress Catalog Card Number: 82-600517

As a part of the program to provide information to Federal agencies on the availability, capabilities, limitations, and applications of software development tools, a database of information about existing tools was collected over a 3-year period. The purpose of this report is to present an analysis of the information contained in the database. Various categorizations of the tools are presented in classes listed by their characteristic features. The lists incorporate percentage summaries that are based on the total number of tools for which information is stored in the database. Trends found in the lists are analyzed and discussed. Abstracts of each tool are presented in an appendix.

programming aids; software automation; software development; software engineering; software testing; software tools

U.S. DEPT. OF COMM. BIBLIOGRAPHIC DATA SHEET (See instructions)

1. PUBLICATION OR REPORT NO. 2. Performing Organ. Report No. 3. Publication Date

NBS SP 500-88 March 1982

4. TITLE AND SUBTITLE

5. AUTHOR(S)

Raymond C. Houghton, Jr.

6. PERFORMING ORGANIZATION (If joint or other than NBS, see instructions)

7. Contract/Grant No.

NATIONAL BUREAU OF STANDARDS
DEPARTMENT OF COMMERCE
WASHINGTON, D.C. 20234

8. Type of Report & Period Covered

Final

9. SPONSORING ORGANIZATION NAME AND COMPLETE ADDRESS (Street, City, State, ZIP)

10. SUPPLEMENTARY NOTES

Library of Congress Catalog Card Number: 82-600517

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)

12. KEY WORDS (Six to twelve entries; alphabetical order; capitalize only proper names; and separate key words by semicolons)

13. AVAILABILITY

14. NO. OF PRINTED PAGES

193

15. Price

$6.50


Order From National Technical Information Service (NTIS), Springfield, VA. 22161
ANNOUNCEMENT OF NEW PUBLICATIONS ON COMPUTER SCIENCE & TECHNOLOGY

Superintendent of Documents,
Government Printing Office,
Washington, D. C. 20402

Please add my name to the announcement list of new publications to be issued in the series: National Bureau of Standards Special Publication 500-.

Name _____________________________________________________________
Company __________________________________________________________
Address ____________________________________________________________
City ______________________ State __________ Zip Code __________

(Notification key N-503)
NBS TECHNICAL PUBLICATIONS

PERIODICALS

JOURNAL OF RESEARCH—The Journal of Research of the National Bureau of Standards reports NBS research and development in those disciplines of the physical and engineering sciences in which the Bureau is active. These include physics, chemistry, engineering, mathematics, and computer sciences. Papers cover a broad range of subjects, with major emphasis on measurement methodology and the basic technology underlying standardization. Also included from time to time are survey articles on topics closely related to the Bureau's technical and scientific programs. As a special service to subscribers each issue contains complete citations to all recent Bureau publications in both NBS and non-NBS media. Issued six times a year. Annual subscription: domestic $18; foreign $22.50. Single copy, $4.25 domestic; $5.35 foreign.

NONPERIODICALS

Monographs—Major contributions to the technical literature on various subjects related to the Bureau's scientific and technical activities.

Handbooks—Recommended codes of engineering and industrial practice (including safety codes) developed in cooperation with interested industries, professional organizations, and regulatory bodies.

Special Publications—Include proceedings of conferences sponsored by NBS, NBS annual reports, and other special publications appropriate to this grouping such as wall charts, pocket cards, and bibliographies.

Applied Mathematics Series—Mathematical tables, manuals, and studies of special interest to physicists, engineers, chemists, biologists, mathematicians, computer programmers, and others engaged in scientific and technical work.

National Standard Reference Data Series—Provides quantitative data on the physical and chemical properties of materials, compiled from the world's literature and critically evaluated. Developed under a worldwide program coordinated by NBS under the authority of the National Standard Data Act (Public Law 90-396).

NOTE: The principal publication outlet for the foregoing data is the Journal of Physical and Chemical Reference Data (JPCRD) published quarterly for NBS by the American Chemical Society (ACS) and the American Institute of Physics (AIP). Subscriptions, reprints, and supplements available from ACS, 1155 Sixteenth St., NW, Washington, DC 20036.

Building Science Series—Disseminates technical information developed at the Bureau on building materials, components, systems, and whole structures. The series presents research results, test methods, and performance criteria related to the structural and environmental functions and the durability and safety characteristics of building elements and systems.

Technical Notes—Studies or reports which are complete in themselves but restrictive in their treatment of a subject. Analogous to monographs but not so comprehensive in scope or definitive in treatment of the subject area. Often serve as a vehicle for final reports of work performed at NBS under the sponsorship of other government agencies.

Voluntary Product Standards—Developed under procedures published by the Department of Commerce in Part 10, Title 15, of the Code of Federal Regulations. The standards establish nationally recognized requirements for products, and provide all concerned interests with a basis for common understanding of the characteristics of the products. NBS administers this program as a supplement to the activities of the private sector standardizing organizations.

Consumer Information Series—Practical information, based on NBS research and experience, covering areas of interest to the consumer. Easily understandable language and illustrations provide useful background knowledge for shopping in today's technological marketplace.


Order the following NBS publications—FIPS and NBSIR's—from the National Technical Information Services, Springfield, VA 22161.


NBS Interagency Reports (NBSIR)—A special series of interim or final reports on work performed by NBS for outside sponsors (both government and non-government). In general, initial distribution is handled by the sponsor; public distribution is by the National Technical Information Services, Springfield, VA 22161, in paper copy or microfiche form.