AD-A221 434

RADC-TR-90-9
Final Technical Report
March 1990

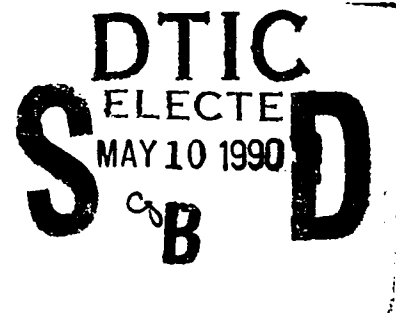# DATA AND ANALYSIS CENTER FOR SOFTWARE

IIT Research Institute

Nancy L. Sunderhaft and Stephen T. Kelly

DTIC
ELECTE
MAY 10 1990
S B D

Rome Air Development Center
Air Force Systems Command
Griffiss Air Force Base, NY 13441-5700

90 05 10 009

This report has been reviewed by the RADC Public Affairs Division (PA) and is releasable to the National Technical Information Services (NTIS) At NTIS it will be releasable to the general public, including foreign nations.
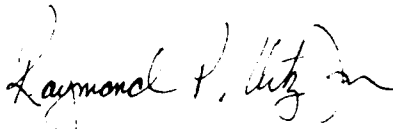
RADC-TR-90-9 has been reviewed and is approved for publication.

APPROVED: *John Palaimo*

JOHN PALAIMO
Project Engineer

APPROVED: *Raymond P. Urtz Jr*

RAYMOND P. URTZ JR
Technical Director
Directorate of Command & Control

FOR THE COMMANDER: *Igor G. Plonisch*

IGOR G. PLONISCH
Directorate of Plans & Programs

REPORT DOCUMENTATION PAGE			*Form Approved* OMB No 0704-0188

1a REPORT SECURITY CLASSIFICATION UNCLASSIFIED		1b RESTRICTIVE MARKINGS N/A	
2a. SECURITY CLASSIFICATION AUTHORITY N/A		3 DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution unlimited.	
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE N/A			
4. PERFORMING ORGANIZATION REPORT NUMBER(S) N/A		5. MONITORING ORGANIZATION REPORT NUMBER(S) RADC-TR-90-9	
6a. NAME OF PERFORMING ORGANIZATION IIT Research Institute	6b. OFFICE SYMBOL (If applicable)	7a. NAME OF MONITORING ORGANIZATION Rome Air Development Center (COEE)	
6c. ADDRESS (City, State, and ZIP Code) New York Technology Center Beeches Technical Campus Rome NY 13440		7b ADDRESS (City, State, and ZIP Code) Griffiss AFB NY 13441-5700	
8a. NAME OF FUNDING/SPONSORING ORGANIZATION DLA-DTIC	8b OFFICE SYMBOL (If applicable) DTIC-AI	9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER F30602-86-C-0111	

8c. ADDRESS (City, State, and ZIP Code) Cameron Station Alexandria VA 22314	10 SOURCE OF FUNDING NUMBERS			
	PROGRAM ELEMENT NO	PROJECT NO	TASK NO	WORK UNIT ACCESSION NO
	65802S	R266	00	02

11. TITLE (Include Security Classification)

DATA AND ANALYSIS CENTER FOR SOFTWARE

12 PERSONAL AUTHOR(S)
Nancy L. Sunderhaft, Stephen T. Kelly

13a. TYPE OF REPORT Final	13b TIME COVERED FROM Apr 86 TO Jun 89	14 DATE OF REPORT (Year, Month, Day) March 1990	15. PAGE COUNT 88

16 SUPPLEMENTARY NOTATION

N/A

17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	Software Engineering Software Technology
12	07		Information Analysis Center
			Scientific & Technical Information

19 ABSTRACT (Continue on reverse if necessary and identify by block number)

The report discusses the operation and enhancements of the Data & Analysis Center for Software during the period from April 1986 to June 1989. Descriptions of the Software Life Cycle Empirical Database and the scientific and technical information database are provided by the DACS, products developed during the period and the technical inquiries received relating to Software Technology. Special studies performed for DACS users on software technology problems and issues are also described.

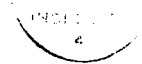| 20 DISTRIBUTION/AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS	21 ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED	
22a NAME OF RESPONSIBLE INDIVIDUAL John Palaimo	22b TELEPHONE (Include Area Code) (315) 330-4063	22c OFFICE SYMBOL RADC (COEE)

DD Form 1473, JUN 86   *Previous editions are obsolete*   SECURITY CLASSIFICATION OF THIS PAGE

FOREWORD

This is the final technical report for the **Data & Analysis Center for Software**, CDRL No. A003, under Contract No. F30602-86-C-0111. This contract is with IIT Research Institute (IITRI) and is sponsored by the Rome Air Development Center (RADC).

i

ABBREVIATIONS

ACEC	Ada Compiler Evaluation Capability
ACS	Ada Compilation System
AFATL	Air Force Armament Laboratory
AFCCE	Air Force Cost Center
AI	Artificial Intelligence
AMC	U.S. Army Materiel Command
AMC MEA	AMC Management Engineering Activity
AMCCOM	U.S. Army Armament, Munitions, and Chemical Command
AMS	Automated Measurement System
ARF	Architecture Research Facility
AVS	Ada Verification System
BSDS	Baseline Software Data System
C3I	Command, Control, Communications and Information
CALSI	Core Set of Assembly Language Instructions
CAMP	Common Ada Missile Packages
CATSS	Cartographic Applications for Tactical and Strategic Systems
CEAC	U.S. Army Cost and Economic Evaluation Center
CECOM	U.S. Army Communications and Electronics Command
COCOMO	Constructive Cost Model
CPM	Contract Program Manger
CPU	Central Processing Unit
CSE	Center for Software Engineering
DACS	Data & Analysis Center for Software
DACSGUIDE	User's Guide to DACS Products and Services
DBMS	Data Base Management System
DGTS	Dynamic Ground Target Simulator
DIDS	Data Item Descriptions
DLSC	Defense Logistics Services Center
DOD	Department of Defense
ESD	Electronic Systems Division
GCOS	General Comprehensive Operating System
GCOS MDQS	GCOS Management Data Query System
IAC	Information Analysis Center
IITRI	IIT Research Institute

JLC	Joint Logistics Commanders
JSSEE	Joint Services Software Engineering Environment
KBSA	Knowledge Based Software Assistant
LCSE	Life-Cycle Software Engineering Center
LCSMM	Life-Cycle System Management Model
MCCR	Mission Critical Computer Resources
MICOM	U.S. Army Missile Command
MIPS	Millions of Instructions per Second
NAC	Naval Avionics Center
NAVSEA	Naval Sea Systems Command
NBS	National Bureau of Standards
O&M	Operation and Maintenance
PAVE/PAWS	PAVE Phased Array Warning System
PC	Personal Computer
PDSS	Post Deployment of Software Support
QCAL	Qualified Contractors Access List
R&D	Research and Development
R&M	Reliability and Maintenance
RADC	Rome Air Development Center
RISC	Reduced Instruction Set Computer
SDS	Software Development Standard
SEBD	Software Engineering Bibliographic Database
SECOMO	Software Engineering Cost Model
SEI	Software Engineering Institute
SEL	Software Engineering Laboratory
SERP	Software Engineering Research Projects
SIGINT	Signal Intelligence
SLED	Software Life-Cycle Empirical Database
SOAR	State-of-the-Art Review
SOW	Statement of Work
SSM	SIGINT Sensor Model
STARS	Software Technology for Adaptable, Reliable Systems
STI	Software Tool Information
STINFO	Scientific and Technical Information
UFP	Unix FORTRAN Preprocessor
UPD	User Profile Database

U.S.	United States
USAF	United States Air Force
VE	Value Engineering
V&V	Verification and Validation

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1.0 INTRODUCTION

1.1 Background

The United States Air Force (USAF) recognized the need for an information analysis center to serve the Government, industrial, and university community as a focal point for software development and experience data. In 1976 the Rome Air Development Center (RADC) contracted with IIT Research Institute (IITRI) to design a center that would acquire, analyze, synthesize, and disseminate information on software engineering technology [DUVA76]. Subsequently, in August 1978, RADC contracted with IITRI to develop such a center, the Data & Analysis Center for Software (DACS). The activities, accomplishments, and history of the DACS during its 36-month pilot period, August 1978 through August 1981, are reported in RADC-TR-81-385, Establishment of the Data & Analysis Center for Software [CARO82].

The DACS was designated a Department of Defense (DOD) Information Analysis Center (IAC) in January 1981 during its pilot period. At the end of this period, IITRI was awarded a contract to operate the DACS for an additional 14-month period. The primary focus of this effort was to provide an orderly transition from a pilot information center to a full-scale IAC; from a center completely supported by Government funds to an IAC whose users are required to contribute to the support of those functions from which they obtain benefit. The activities performed during the transition period are reported in RADC-TR-83-132, Data & Analysis Center for Software: An IAC in Transition [GLOS83].

In December 1982, IITRI won a 36-month contract to continue operating the DACS as a full-scale IAC. The purpose of this contract was to firmly establish the DACS as a fully functional IAC by serving the needs of the software engineering community. This contract was amended to extend the performance to 40 months, concluding April 1986. The activities performed during this operational period are reported in RADC-TR-86-178, Data & Analysis Center for Software [HENN86].

IITRI was again awarded a 36-month contract (later extended to 38 months) in April 1986 to continue operating the DACS as a full-scale IAC. This report addresses the activities of the DACS during this contract period, 15 ʌʌril 1986 to 15 June 1989.

1.2 DACS Objectives

The DACS was established to serve as a focal point for software development and experience data; for the analysis, synthesis, and dissemination of this data; and as a repository or clearinghouse for scientific and technical information concerning the field of software engineering. The DACS provides a centralized, authoritative source for usable data and information concerning software technology. The objectives of this software information analysis center are to:

- Encourage the exchange of software technology information among DOD and other Government agencies, Government contractors, the private sector, and academia

- Support software technology research by providing a centralized source of software life-cycle data

- Increase the productivity of software producers and the quality of the resultant computer software by improving the transfer of software engineering technology

- Assist in diffusing new technology throughout the U.S. industrial base, thereby expanding its capability and competitive posture

- Provide scientific and technical information analysis services to DOD, civil agencies, Government contractors, and the private sector in areas relating to software technology needs, developments, and trends

- Minimize duplication of software technology research

1.3 Report Contents

DACS activities described in this report were performed with these objectives in mind. This report summarizes the activities of the DACS during the most recent contract period. Sections 2 through 10 correspond to tasks outlined in the contract's Statement of Work (SOW).

Section 1.0   Introduction

Section 2.0   Operation and Maintenance of the DACS

Section 3.0   Acquisition of Software Experience Data

Section 4.0   Description of the Scientific and Technical Information Databases

Section 5.0   Discussion of the DACS Data Analysis Program and Related Efforts

Section 6.0   Summary of the Current Awareness Program; Including Newsletters, Bulletins, and Technical Presentations

Section 7.0   Discussion of DACS Products and Services; Including Datasets, Data Compendia, State-of-the-Art Reviews, Bibliographic and Software Tool Searches, and Consulting Services

Section 8.0   Presentation of New DACS Products and Services; Including a Discussion of Improvements Made to Products and Services

Section 9.0   Discussion of Promotional Advertising

Section 10.0   Synopses of the DACS Special Study Efforts Performed During this Contract Period

Section 11.0   Observations and Recommendations

2.0 TASK 1 - OPERATION AND MAINTENANCE OF THE DACS

2.1 Summary of Goals, Technical Progress, and Activities

2.1.1 <u>Goals Set for the DACS</u>

The goals of the DACS during this period were oriented toward:

- Disseminating state-of-the-art information on software technology of general interest to the software engineering community

- Conducting a current awareness program through publication of newsletters and bulletins, presentations at professional seminars and sites of potential clients, and active participation in professional and technical organizations

- Preparing and distributing products and services designed to meet the information needs of the DACS user community

- Preparing a compendium of software engineering standards, including military and industry standards

- Maintaining and expanding the Software Life-Cycle Empirical Database (SLED), which contains data that is descriptive of the development and maintenance process of a variety of software projects

- Maintaining and expanding the Scientific and Technical Information (STINFO) Database to provide up-to-date research materials from which the products and services of the DACS are developed and to have available the most recent information to supply to users

- Expanding the Software Tool Information (STI) Database and offering custom tool searches to DACS users

- Expanding the scope of data analysis activities and disseminating the results of these analysis activities to the software community in readily usable forms

- Continuing to solicit user feedback on the products and services offered by the DACS

- Tracking and tabulating user interactions with the DACS so that the optimum mix of products and services may be provided to users that support the DACS by purchasing its products and services

- Increasing sales of products and services by exploring new methods for the sale of products and services and by developing new products and services

2.1.2 Summary of Accomplishments

The following paragraphs briefly summarize the major accomplishments during the contract period.

A Unify relational database was defined and implemented for the DACS Productivity and Reliability Datasets. Data collected on FORTRAN programs was identified and added to the Software Life-Cycle Empirical Database (SLED) to establish a new software complexity dataset. The SLED was ported from the RADC GCOS system to the dedicated DACS MicroVAX.

The Software Engineering Bibliographic Database (SEBD) was also migrated from the RADC GCOS system to the DACS MicroVAX. The new SEBD database system, coupled with new document processing procedures, reduces data entry time, enhances quality control, improves data consistency, and simplifies the use of the database for responding to technical inquiries. The total number of documents in the DACS Software Engineering Library increased by 2,373 documents and the number of document citations on-line increased to over 9,165 citations. The Software Tool Information (STI) database was also ported from the RADC GCOS system to the DACS MicroVAX.

Four data analysis reports, were produced. The report, Software Reliability Baselines, contains reliability baselines summarizing the Architecture Research Facility (ARF) Error Dataset, an analysis of possible tradeoffs of productivity and reliability, and an investigation into the performance of the Goel-Okumoto Nonhomogeneous Poisson Process Reliability Model. The second analysis report summarizes the results of an effort directed at using the Automated Measurement System (AMS) to collect empirical data on several FORTRAN language software systems and examining complexity data collected by the AMS. The relationships of these metrics were then analyzed. The third data analysis task examined the Goel-Okumoto Software Reliability Model and produced the report, An Empirical Validation of the Goel-Okumoto Software Reliability Model. The final data analysis effort concluded with the development of a report which proposes a method to validate software reliability models for assumption violations with automatically generated data.

The DACS produced and distributed 12 DACS Newsletters and 24 DACS Bulletins as part of the current awareness program. In addition, members of the DACS staff attended and presented papers at various workshops and meetings.

The DACS now distributes source code for an Ada compiler and debugger developed for the Air Force Armament Laboratory (AFATL). In addition, the DACS distributes the AMS, Ada Compiler Evaluation Capability (ACEC), Common Ada Missile Packages (CAMP), and the Ada Compilation System (ACS).

The Products & Services Flier was updated to include information about new DACS products and ordering export-controlled information.

The DACS distributed a total of 4,345 documents/datasets, processed a total of 4,593 technical inquiries, and received $85,981 for products and services during this contract.

The DACS negotiated and implemented the following special study efforts during this contract period:

- Software Engineering Cost Model Enhancement

- Software Engineering and Ada Language Studies: Phases IV & V

- SDS Fragments/Structures; Phase IIIC

- U.S. Air Force Ada/JOVIAL Software Engineering Tools Survey

- Support Services for the RADC Knowledge Based Software Assistant (KBSA) Conference

- Ada Verification System (AVS) Studies

- Development of a Structured Analogy Approach to Software Size and Cost Estimation

- Signal Intelligence (SIGINT) Sensor Processing Dissemination

- Documenting Management of Mission Critical Computer Resources and Battle Automated Systems

- Software Engineering Support for Life-Cycle Software Engineering Centers

- Development of a Software Oriented Value Engineering Program for CECOM Life-Cycle Software Engineering Centers

- A Descriptive Evaluation of Software Sizing Models

- Conceptual Analysis of a Software Engineering Methodology for Naval Sea Systems Command (NAVSEA) Computing Systems

- Support for the 2nd Annual RADC Knowledge Based Software Assistant (KBSA) Conference

- Reliability and Maintainability Standards Development on Software Testing and Evaluation

- Design of a Software Quality and Productivity Laboratory

- Knowledge-Based Front-End to the SECOMO Software Cost Estimation Model

- Development of a Software First Methodology

- Methodology Studies for Real-Time Ada Software

- SECOMO Training

- Support for the 3rd Annual RADC Knowledge Based Software Assistant (KBSA) Conference

- Reduced Instruction Set Computer (RISC) Evaluation

- Support for the 2nd Annual RADC Artificial Intelligence (AI) Technology Fair

- Ada Compiler and Software Engineering Tool Evaluation

- Software Size/Cost Models Seminar Update

- Core Set of Assembly Language Instruction (CALSI) Update

- Management, Review, and Analysis of Army Software Issues

3.0 TASK 2 - ACQUISITION OF SOFTWARE EXPERIENCE DATA

3.1 Introduction

There is a need to collect productivity and failure data on the development, operation, and maintenance of software to support research in the software field. Data is needed which will allow researchers to isolate the factors that contribute significantly to the cost, reliability, and quality of software, to measure achieved reliability, to predict development and maintenance costs, and to track the progress of a software development project. This section contains descriptions of the DACS Data Acquisition Program and the resultant DACS Software Life-Cycle Empirical Database (SLED).

3.2 Data Acquisition

The means used to maintain the Data Acquisition Program consisted of:

- Identifying data sources and acquiring relevant data
- Establishing procedures for automatic submission of data
- Establishing procedures for processing and evaluating data
- Maintaining the computer database

The procedures used to acquire data are based on the documented procedures outlined at length in RADC-TR-81-385 [CAR082]. These procedures have been modified over the operation of the DACS to include standard data exchange formats and development of the <u>DACS Software Engineering Data Collection Package</u> for data acquisition.

3.3 Sources Identified

During this reporting period the DACS began using the Automated Measurement System (AMS) as a data collection vehicle. The AMS automatically collects a subset of the RADC Quality Framework's metrics. The AMS has been applied in-house on three FORTRAN programs. The resulting datasets have been added to the SLED, but are not yet available to DACS users. The three FORTRAN programs include:

- Software Engineering Cost Model (SECOMO). SECOMO, written by IITRI, is a computerized implementation of Dr. Barry Boehm's Constructive Cost Model (COCOMO). The resulting metric was analyzed as a part of the DACS Data Analysis Program.

- Cartographic Applications for Tactical and Strategic Systems (CATSS). The CATSS Sensitivity Model is a laboratory tool developed by IITRI and the Hughes Aircraft Company. Some of the resulting metric data, specifically size data, was analyzed as part of a DACS special study on sizing models.

- Unix FORTRAN Preprocessor (UFP). The UFP was developed as a Unix tool which allows a user to write Pascal-like FORTRAN programs in an extended FORTRAN language. The tool will then process this extended FORTRAN language into compilable FORTRAN-77 code.

3.3.1 Automatic Submission of Data

The DACS received an update of the NASA Software Engineering Laboratory (NASA/SEL) Dataset and numerous updates of the PAVE Phased Array Warning System (PAVE/PAWS) Dataset.

3.4 Accomplishments of the SLED

The DACS began porting the existing datasets in the SLED to the Unify database management system on the DACS MicroVAX. The porting will allow custom datasets to be produced more easily, ease data extraction efforts conducted as part of the DACS Data Analysis Program, and lead to the development of new products. Two datasets, the DACS Productivity Dataset and the Software Reliability Dataset, were completely ported to Unify databases.

During this reporting period the DACS began distributing three SLED datasets on floppy diskettes for IBM-PCs and compatibles:

- DACS Productivity Dataset
- Software Reliability Dataset

• Architecture Research Facility (ARF) Errors Dataset

In addition, the DACS produced the second and third editions of the DACS Data Compendium (March 1986 and 1989) which outline the holdings in the DACS SLED.

3.5 Current Status of the SLED

The DACS SLED currently consists of nine datasets distinguishable by data source, data collection and acquisition methodology, life-cycle phase represented, and data parameters present. Because each set of data was the result of a data collection effort that pursued a specific objective, the resulting datasets differ with regard to:

• The time period represented by projects in a dataset

• The portion of the software life-cycle represented by the data

• The aspects of the software development and/or maintenance processes measured by the data collection activity

• The quality of the data as reflected in the verification and validation procedures used in data collection

• The subsequent analyses supported by the data

The following nine datasets are associated with the SLED. Complete descriptions of the datasets can be found in the references provided.

• The DACS Productivity Dataset contains project level data on over 500 projects; the dataset is maintained as a Unify dataset, and is distributed by the DACS in hardcopy form, on 9-track tape, or on floppy diskettes [NELS78].

• The NASA Software Engineering Laboratory (SEL) Dataset contains very detailed data on over 30 projects conducted by NASA's Goddard Space Flight Center; the dataset is maintained as a GCOS MDQS database, and distributed by the DACS in hardcopy form or on 9-track tape [BASI79].

- The Verification and Validation (V&V) Dataset contains data on almost 1,000 anomaly reports from five software projects; the dataset is maintained as a GCOS MDQS database, and is distributed by the DACS in hardcopy form or on 9-track tape [RADA81].

- The Software Reliability Dataset contains data on over 2,800 failures from 16 projects; the dataset is maintained as a Unify dataset, and is distributed by the DACS in hardcopy form, on 9-track tape, or on floppy diskettes [MUSA79].

- The Architecture Research Facility (ARF) Error Dataset contains data on 117 errors from a system with 253 modules; the dataset is maintained as a GCOS MDQS database. Distributed by the DACS in hardcopy form or on floppy diskettes [ELOV79].

- The PAVE Phased Array Warning System (PAVE/PAWS) Operations and Maintenance (O&M) Dataset consists of data collected by the Program Support Library on the PAVE PAWS system; the dataset is maintained as a GCOS MDQS database, and is distributed by the DACS in hardcopy form or on 9-track tape [IITR82].

- The Baseline Software Data System (BSDS) Dataset contains problem report and module data on six large software projects; the dataset is maintained as a GCOS MDQS database, but it is not currently distributed by the DACS [DUVA79A, DUVA79B].

- The NASA/Ames Error/Fault Dataset contains data on over 3,700 software problem reports from about 35 projects; the dataset is maintained as a GCOS MDQS database, and is distributed by the DACS in hardcopy form or on 9-track tape [DEFE79].

- The Automated Measurement System (AMS) Dataset contains data obtained by running the AMS on three FORTRAN programs; it contains RADC Quality Framework metrics; the dataset is in the AMS database management system format and is not currently available to DACS users.

When analyzing data of this nature, it is important to consider that the datasets were generated at different points in time. Figure 3-1 illustrates the periods of time represented by data in each of the datasets. Each of the datasets contains data from various life-cycle phases as depicted in Figure 3-2. Of these nine distinct datasets, five are available in standard format. The four remaining datasets (ARF, BSDS, AMS, and O&M) have not been processed into a form that is readily usable; only customized versions of three of these datasets are distributed (ARF, BSDS, and O&M). Each of the eight sets of data that are distributed is discussed in detail in The DACS Data Compendium (March 1989).
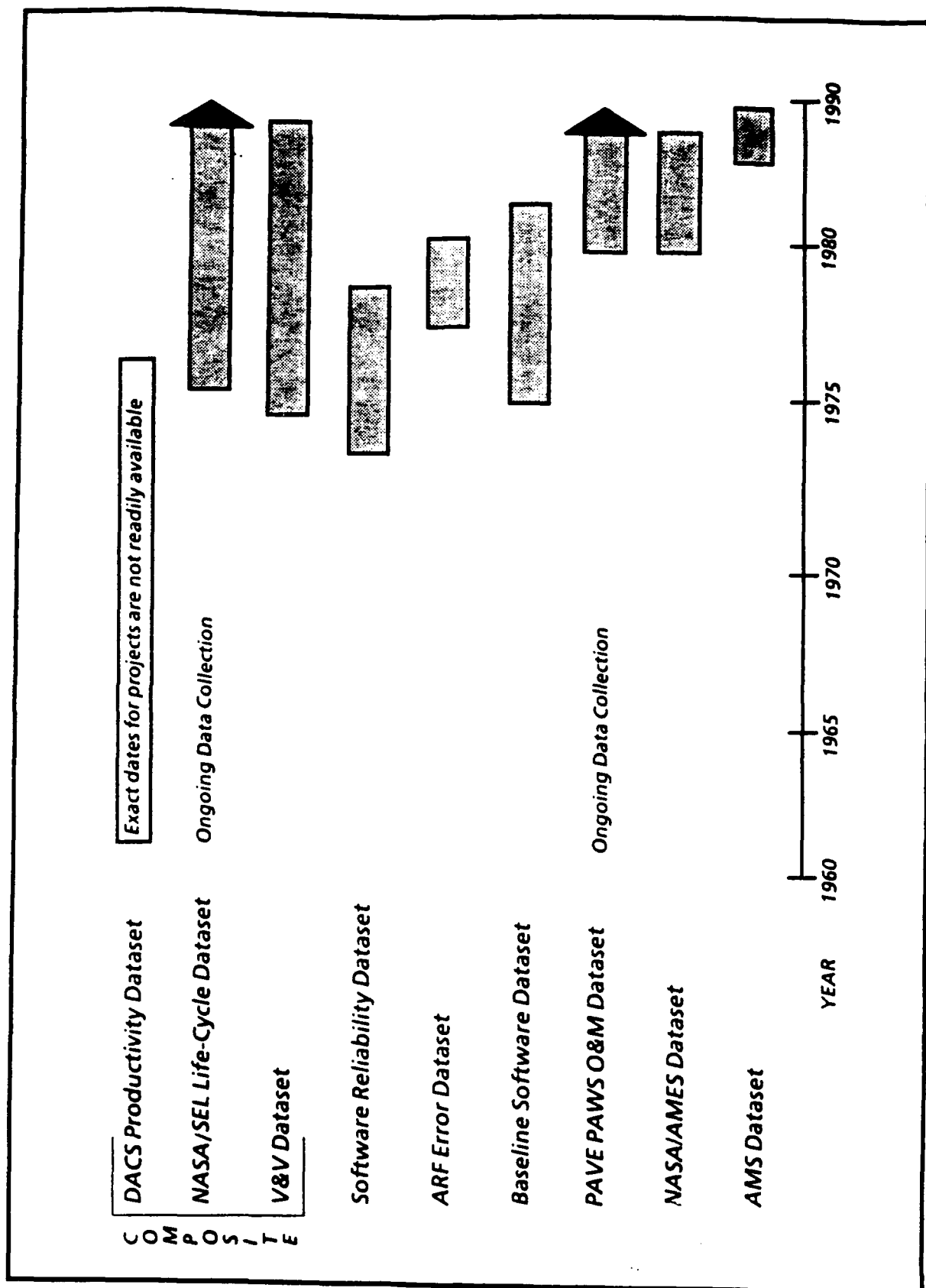
Figure 3-1. Time Periods Represented by SLED Datasets
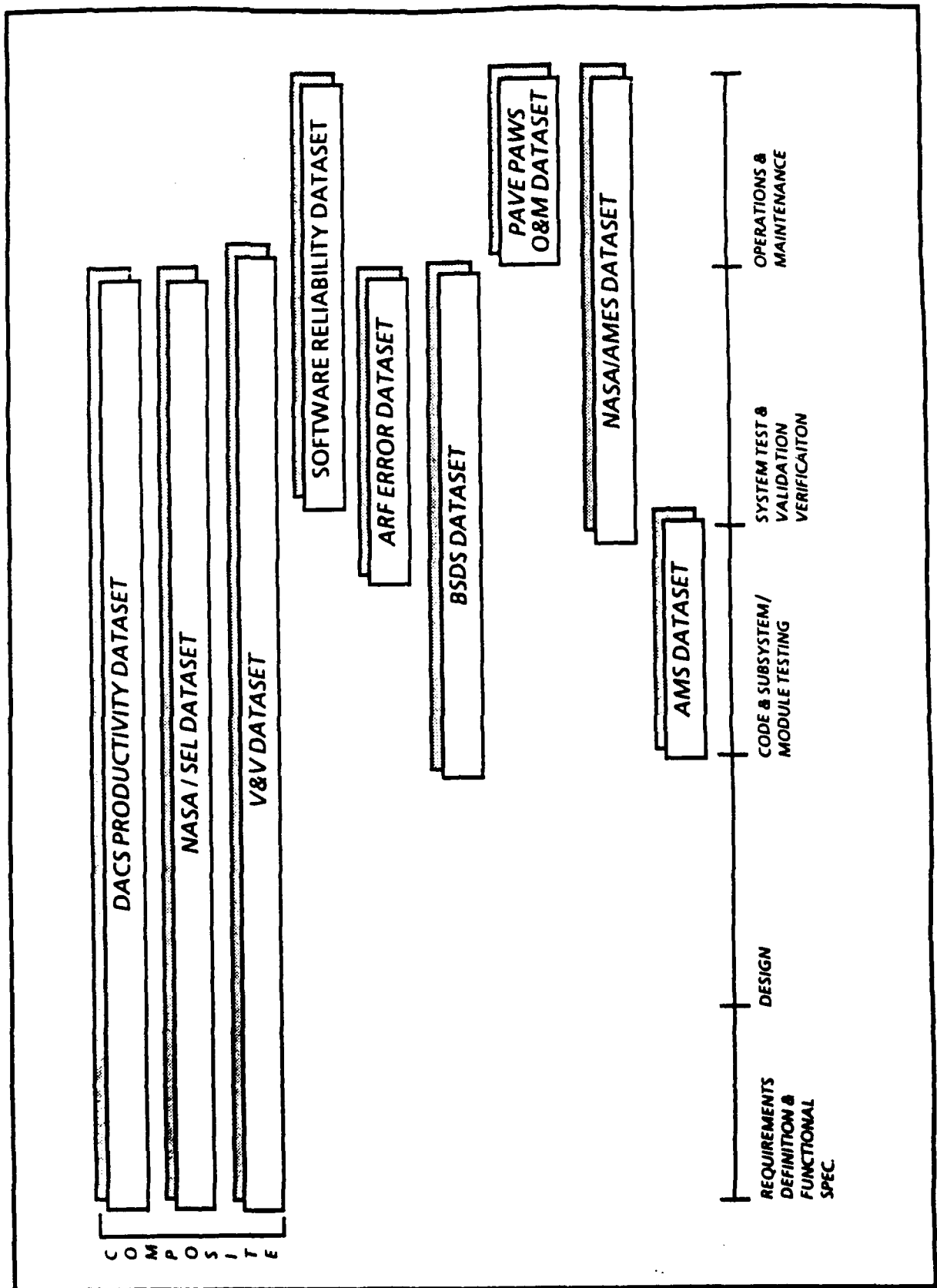
Figure 3-2. Life-Cycle Phases Described by SLED Datasets

4.0 TASK 3 - SCIENTIFIC AND TECHNICAL INFORMATION

4.1 Introduction

Scientific and Technical Information (STINFO) consists of documented information concerning the state-of-the-art and technology aspects of the computer software field. STINFO includes technical reports, trade journal publications, proceedings of conferences and symposia, theses, texts, and product descriptions and specifications. TABLE 4-1 shows the sources from which the software engineering documents are identified.

TABLE 4-1. SOURCES FOR SOFTWARE ENGINEERING DOCUMENTS

DOCUMENT SOURCE	DOCUMENT TYPE	IDENTIFICATION METHOD
National Technical Information Service (NTIS)	Unclassified Reports from Government Agencies	Review of Publication Announcements; NTIS Technical Notes
Defense Technical Information Center (DTIC)	Classified or Limited Distribution reports on R&D performed by DOD and contractors	DTIC Technical Abstracts Bulletin
General Accounting Office (GAO)	Government Agency Investigative Reports	Distribution List
Naval Publications and Forms Center	DOD Standards and Regulations	Catalog
Rome Air Development Center (RADC)	Technical Reports	Distribution List
Software Engineering Institute (SEI)	Technical Reports	Distribution List
Software Engineering Laboratory (SEL)	Technical Reports	Distribution List

Professional Societies
    IEEE
    ACM
	Papers, Conference Proceedings, Journals, and Tutorials	Society Membership; Attendance at Conferences; Review of Publication Catalogs

Periodicals	Journal Articles	Subscriptions
Textbooks	Texts	Catalogs; Book Reviews
Colleges and Universities	Reports, Theses, and Dissertations	Review of Dissertation Abstracts
Information Retrieval Services	Papers, Conference Proceedings, and Journals	Subscription

Also included as STINFO are descriptions of on-going software technology research for which reports may not yet be available. These two types of STINFO serve as input to three information databases maintained by the DACS as shown in Figure 4-1: the Software Engineering Bibliographic Database (SEBD), the Software Engineering Research Projects (SERP) Database, and the Software Tool Information (STI) Database.
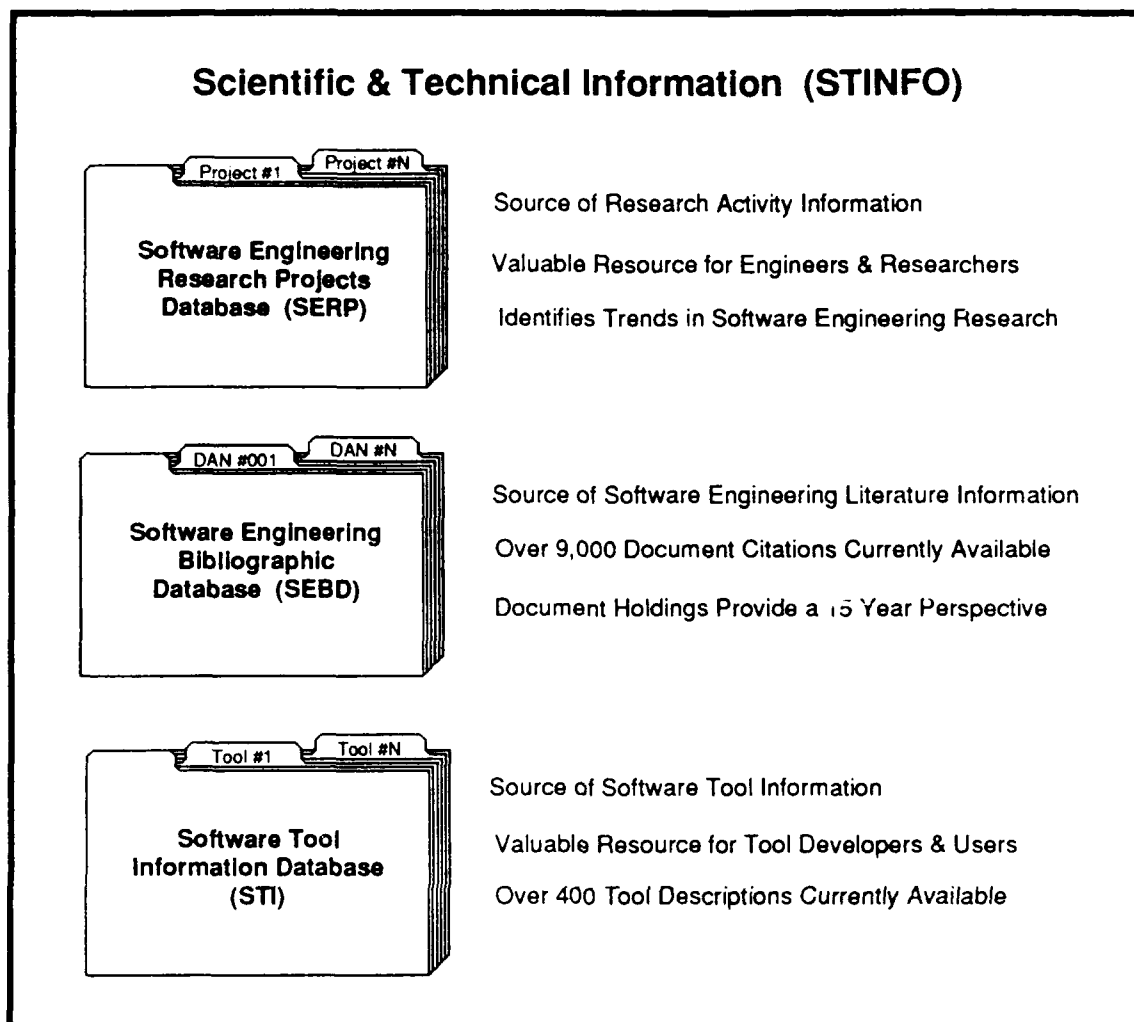
Figure 4-1. The Three STINFO Databases

4.1.1 The Software Engineering Bibliographic Database (SEBD)

The DACS SEBD was established to provide a readily accessible source of comprehensive information on the state-of-the-art in software engineering, as well as a means of channeling that information to software engineers and developers who can apply it in their daily activities of developing, maintaining, and managing software. The bibliographic collection is composed of documents related to software engineering, reliability, cost and quality factors, maintainability, and related topics. The collection is computer-accessible and retrieval of document bibliographic information can be made on any part of the document citations or on assigned keywords.

The SEBD was ported from the RADC GCOS system to the DACS MicroVAX during this contract period. The database system supports on-line indexing by analysts, special quality control reports, automatic data consistency checks and reports, and interactive query capabilities that simplify using the database for responding to technical inquiries.

The procedures used for processing documents were also revised. The new procedures reduce data entry time, enhance quality control, and reduce the turn-around time for document processing. An automated current acquisition listing is produced monthly and circulated to RADC and DACS staff.

Procedures to generate a software engineering bibliography were developed and tested for the SEBD resident on the DACS MicroVAX. The procedures were used to produce the DACS Software Engineering Bibliography; Fifth Annual Supplement and the DACS Software Engineering Bibliography; Sixth Annual Supplement.

The current status of the DACS Software Engineering Library and SEBD for this contract is indicated below:

Documents Ordered	690
Documents Received & Accessioned	1,660
Documents Indexed & Abstracted	2,373
Document Citations Entered On-line	2,373
Total Number of Citations in On-line Database	9,165
Total Number of Documents in Library	9,165

Figure 4-2 displays the composition of the SEBD by document type.

4.1.2 The Software Engineering Research Projects (SERP) Database

The SERP Database provides a computer-accessible source of information about software engineering research. Projects included in this database are those involving software technology research, such as the development or evaluation of programming languages, models, or software tools, and research related to software engineering methodologies, such as modern programming practices.
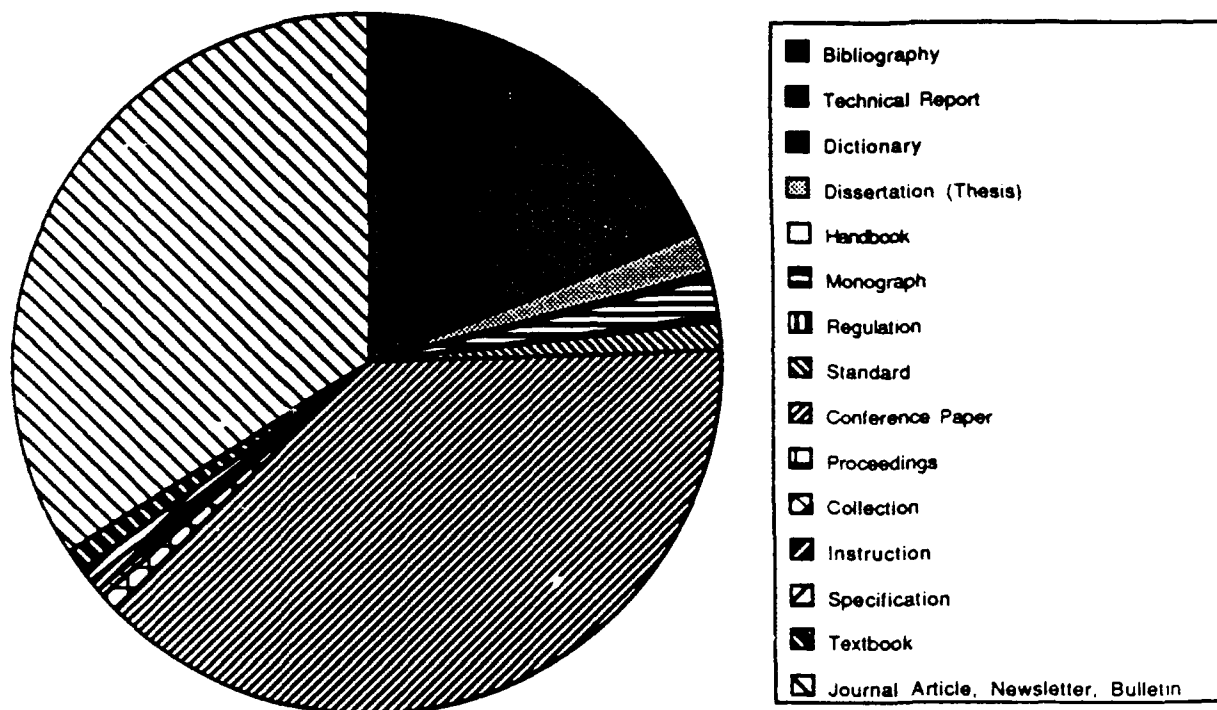
Figure 4-2. Composition of the DACS SEBD

Legend:
- Bibliography
- Technical Report
- Dictionary
- Dissertation (Thesis)
- Handbook
- Monograph
- Regulation
- Standard
- Conference Paper
- Proceedings
- Collection
- Instruction
- Specification
- Textbook
- Journal Article, Newsletter, Bulletin

4.1.3 The Software Tool Information (STI) Database

The National Bureau of Standards (NBS) developed a software tools database which contained information on 250 software tools. When the NBS could no longer support this database, it sought a new facility to support the tools database and make it available to the software engineering community. Recognizing the need for a central source of software development tool information, the DACS acquired NBS's Software Development Tools Database. This database provided the foundation for the DACS STI Database.

To facilitate the collection of information on new software development tools, the DACS developed Software Tool Description Forms. These forms are used in conjunction with an information acquisition program implemented specifically for identifying and acquiring information on new software development tools.

The DACS distributes the <u>Software Life-cycle Tools Directory</u>, which describes tool information contained in the STI Database. During this contract period the STI Database was ported to the DACS MicroVAX.

In addition, the data collection forms used for the STI Database were reviewed to find possible enhancements based on the tool information forms developed for the Air Force Ada/JOVIAL Tool Survey special study. Over 100 new potential sources of tool information were identified. An entirely new database was developed with IITRI funds for the STI. The new system supports both automatic form and letter generation to solicit, confirm, and update tool information.

5.0 TASK 4: DATA ANALYSIS PROGRAM

5.1 Introduction

The DACS contributes to software technology research through data analysis, as well as through special studies, state-of-the-art reviews, and technology assessments. The first Data Analysis Program Plan was initiated in 1981 [TURN81]. Each successive contract further developed that plan by outlining additional tasks to be conducted during a given contract period [ROME83, IITR86].

The research and development objectives for the current Data Analysis Program Plan include:

- Producing products and reports that will permit DACS customers to more easily use data in the SLED by describing what analysis each dataset can support and summarizing crucial relations and distributions in each dataset

- Providing managers with historical baselines for software productivity, cost, reliability, and quality

- Providing insight into the processes involved in producing, managing, and maintaining software

- Identifying those factors which influence the cost, reliability, quality, and complexity of software.

- Validating and developing software engineering models, such as cost, quality, and reliability models

This section provides an overview of the DACS Data Analysis Program Plan and discusses the research and development efforts which resulted from the data analysis task.

5.2 Overview of the DACS Data Analysis Program Plan

The purpose of the DACS Data Analysis Program Plan is to outline a set of analysis tasks which can be performed by the DACS. The SLED serves as primary input to all of the analysis tasks outlined in the plan. Figure 5-1 shows the analysis process that is carried out for each investigation.

Experience with previous data analysis tasks permitted a better assessment of the availability and reliability of data in the SLED and, in turn, made it possible to define new areas in which more relevant and useful results can be obtained. The plan organizes tasks into three major topic areas: Reliability, Cost, and Quality.

5.3 Data Analysis Efforts

During this reporting period the DACS performed data analysis experiments which investigated tools and procedures for measuring the impacts of application areas, software development methodologies, and software characteristics on software reliability.

The DACS Data Analysis Program Plan was submitted to the Contract Program Manager (CPM) for approval. The plan consists of a menu of tasks of which a subset are performed during a given contract, as determined by the DACS Data Analyst and the CPM.

The first data analysis report, <u>Software Reliability Baselines</u> [VIEN87A], analyzed three SLED datasets; the DACS Productivity Dataset, the Software Reliability Dataset, and the Architecture Research Facility (ARF) Dataset. The analysis, detailed in the report, demonstrated the following:

- The DACS Productivity Dataset shows that variations in fault density are not significantly related to variations in productivity. Therefore, methods that increase reliability will not necessarily lower productivity.

Issue Identification

Determine Analysis Objective

Establish Analysis Plan

Determine Data Availability

Yes / No

DATA Acquisition Program

Conduct Analysis

Document Results

Review and Approval by RADC CPM

1) **Issue Identification** - The results of the DACS User Survey conducted during the current DACS contract and the review of current literature and DOD initiatives are utilized to identify issues for analysis.

2) **Determine Analysis Objective** - Once approved, the objectives of the analysis effort together with expected outcome of the investigation are identified.

3) **Establish Analysis Plan** - A plan is developed to address schedules, required resources, milestones and decision points pertinent to the effort. The effort is not pursued until approval of the plan by the RADC CPM.

4) **Determine Data Availability** - Data availability is determined via the DACS Data Compendium together with knowledge of the contents of the SLED and the types of analysis that each data set will support. If data does not exist, a focused collection effort through the DACS Data Acquisition Program is initiated and coordinated with the RADC CPM.

5) **Conduct Analysis** - The analysis is conducted according to the analysis plan.

6) **Document Results** - The results of the analysis is documented.

7) **Review and Approved by CPM** - The documented results of the analysis are delivered to the RADC CPM for review and approval.

Figure 5-1. Data Analysis Process

- In the DACS Productivity Dataset, both productivity and fault density vary significantly with the programming language used.

- A project from the Software Reliability Dataset was used to compare the performance of the Goel-Okumoto Software Reliability Model when applied to failure data recorded in both CPU and calendar time. The model fit calendar-time data better than CPU-time data. Some interesting effects of aggregating time-between-failure data to failure-count data were also discovered.

- The ARF dataset shows that of the three variables, complexity, size, and function type, function type is the most strongly correlated with the number of faults in a subroutine. Computational and control subroutines tend to contain more faults than data accessing, error handling, and initialization subroutines.

- In the ARF dataset, the three variables complexity, size, and function type are not independent; they all correlate with one another with varying strengths.

The second data analysis report, Software Quality Measurement Using the Automated Measurement System and Operational Data [VIEN87B], was also developed. This report concluded the following:

- The AMS was found to be fairly easy to use. A few bugs were noted, and enhancements were suggested. The most useful enhancement would allow the user to automatically collect data from source code and produce a report summarizing the metrics collected without requiring any knowledge of the backend relational database management system, RIM, or the levels in the quality hierarchy above the metrics level.

- SECOMO was run through the AMS and certain complexity metrics were examined. It was found that all the metrics examined correlated with each other.

- The metrics examined were found to be useful in identifying subroutines that might be of suspiciously low quality. These subroutines show up as outliers on histograms of metric values.

- Data on the PAVE/PAWS radar system was examined to provide a baseline for the reliability of radar systems. The PAVE/PAWS system is in operation at two sites. One site reported a mean-time-between-failures of 44 days, and the other site reported a mean-time-between-failures of 17 days.

- In another radar system, fault density tended to be about 5.2 faults per thousand instructions. This radar system data is a portion of the Baseline Software Data System (BSDS) Dataset in the SLED.

- Coding a module using structured techniques did not seem to affect fault density in the latter system; however, it did significantly alter the type of errors, the phase in which they entered, and the ease with which they could be corrected.

The third data analysis report, An Empirical Validation of the Goel-Okumoto Software Reliability Model [VIEN88B], was also developed. This report concluded the following:

- The Goel-Okumoto Model fits the data in the DACS Software Reliability Dataset quite well. The Kolmogorov-Smirnov test did not identify a statistically significant difference between the model and the data in 14 of the 16 cases. All but one of the 16 projects seemed to experience decreasing failure rates.

- The use of the Newton-Ralphson Method in estimating maximum likelihood parameters of the Goel-Okumoto model did not result in intracTABLE convergence problems. The region of convergence tended to be fairly small and varied in location by an order of magnitude.

- Maximum likelihood parameters exist for 15 of the 16 projects. They also exist in the same cases for the other two software reliability models: Jelinski-Moranda and Musa Execution Time Model.

- The Goel-Okumoto Model's estimates of the number of failures required to expose all the faults in the software seemed to be quite close to the previously published estimates for the Musa Execution Time Model.

- These conclusions indicate that the Goel-Okumoto Model is valid and suggest that the other software reliability models may perform in a comparable manner.

The last data analysis report, <u>A Simulator for Investigating the Robustness of Software Reliability Models to Assumption Violations</u> [VIEN89], outlined a proposed method of investigation in the absence of good data. The report described the need for data with known properties to validate software reliability models:

- A method for generating simulated failure data with known properties

- A high level overview of three experiments for validating various aspects of software reliability models

- A detailed design of one experiment

- A computer program for generating simulated failure data and conducting the experiment

The report contained the following sections:

- Overview

- Reliability concepts and a method of simulated testing based on these concepts

- Reliability model assumptions and real data effects on the models

- Experiments which can be developed with generated data to test model assumptions accurately

- Development of a computer program to generate the data

- Descriptions of other experiments that could be performed.

5-6

6.0 TASK 5 - CURRENT AWARENESS PROGRAM

6.1 Introduction

The Current Awareness Program was maintained and enhanced throughout this reporting period. The program has two purposes, to keep the DACS user community informed of the latest and most significant developments in software technology and engineering, and to inform its current and potential users of products and services offered by the DACS, and the benefits to be realized through use of the DACS. The DACS implements the Current Awareness Program in the following ways:

- Publishing the DACS Newsletter

- Publishing the DACS Bulletin

- Participating in conferences and symposia

- Establishing contacts throughout the software engineering community through active participation in professional organizations

- Placing press releases and announcements describing DACS products and professional activities in trade journals, newspapers, and magazines

- Distributing informational materials designed and developed by the DACS staff

Activities relating to the dissemination of information are discussed in this section; the promotional aspects of the Current Awareness Program are discussed in Section 9.0 under Task 8, Promotional Advertising.

6.2 DACS Newsletter

The DACS Newsletter is the primary means for disseminating current information to the DACS user community. Each issue contains synopses and critiques of significant, newly acquired reports or articles; summaries of new programs; listings of future conferences and symposia; summaries of significant technological breakthroughs and applications; and highlights of other outstanding developments of interest to DACS users. The newsletter is also used as a vehicle to announce new DACS products and programs to its user

community.   In general, one DACS product is featured in each issue, with capsule summaries of other products featured as space permits.

The DACS Newsletter is published quarterly, and is distributed free of charge to government and non-government personnel having an interest in the disciplines served by the DACS.   The newsletter is mailed to over 5,000 contacts is distributed at conferences attended by DACS personnel, and is included in the DACS Information Package.   Twelve issues of the DACS Newsletter were produced during the current reporting period.   Figure 6-1 illustrates the front page of the March 1989 issue of the DACS Newsletter.

During this period the newsletter's layout was redesigned and the production techniques were improved.   These improvements included using desktop publishing software to produce the newsletter, resulting in more efficient production of a higher quality newsletter.

6.3 DACS Bulletin

The DACS Bullet¹ discusses a topic of high interest in greater depth than space allows in the DACS Newsletter.   The bulletin is routinely distributed on a limited basis to RADC personnel, to DACS Participation Plan members, and to DACS Bulletin subscribers.

Twenty-four issues of the DACS Bulletin were published during this reporting period.   Figure 6-2 contains the front page of the June 1989 issue of the DACS Bulletin.   Figure 6-3 contains a sample of the topics discussed in recent DACS Bulletins.

As with the newsletter, the bulletin's layout was redesigned and the production techniques were improved, allowing more efficient production of a a higher quality bulletin.

DACS NEWSLETTER

Software Technology for Adaptable, Reliable Systems (STARS) Program Update

The Software Technology for Adaptable, Reliable Systems (STARS) Program is a major Department of Defense (DoD) software initiative that began in 1982. Its objectives are to achieve dramatic improvements in software quality and to mitigate runaway software costs, especially for embedded and mission critical applications. Over the next five years, the STARS Program's goal is to achieve at least a tenfold increase in software productivity through developments in software engineering technology.

In 1988, the STARS Program transitioned from developing the foundations needed to implement new software engineering principles to building on and applying them to the development of systems-in-the-large. One of the strategies adopted by the STARS Program is to define a "software-first" approach to system development: developing the software for a system independently of its target hardware. With the software-first approach, hardware is selected near the end of the development cycle, with selection criteria based on system needs and software capabilities. The STARS Program will leverage off the capabilities provided by the Ada programming language. A second strategy, that is intended to directly reduce software costs while laying a foundation for software-first, is to develop reusable Ada components. Software-first cannot be a viable development approach unless a critical mass of components which facilitate machine independence is available. Since Ada is a relatively new computer language and good Ada compilers are an even more recent development, these types of components were not available. A first step for the STARS Program, therefore, was to create a foundation set of reusable software components via the STARS Foundations Projects.

The purpose of the foundations research and development program was to build state-of-the-art Ada primitives. The collection of primitives was structured to be adaptable to different application domains. The objective of the program's approach was to achieve economy in Ada software engineering by focusing on common software in a way that facilitates and encourages reuse. Each foundation research project resulted in an executable, unit-tested, documented, primitive capability of Ada, i.e., a reusable Ada component.

All of the foundations products are available to STARS contractors to support exchange of information and STARS goals. Software tools produced under the foundations projects will also be made widely available. STARS contractors will be responsible for integrating the tools into specialized software engineering support environments.

The STARS Foundations Projects consist of approximately 40 projects spanning 12 technical areas. These areas and example projects for each area include: Operating Systems (Universal Filenames and Tailorable Ada Run-time Environment); Database Management Systems (Databases in and for Ada-Supported Information System Management); User Interfaces (Virtual Computer-Aided Instruction Interface); Command Language (Ada Command Environment); Graphics (Computer Graphics/Graphic Kernel System/Ada Graphics); Text Processing (Standard Generalized Markup Language); Network/Communication (Network Protocol); Run-Time Support (Ada Remote Procedure Call); Mission Planning and Optimization (Planning and Optimization Tools); Design, Integration, and Test (Ada Test Support Tool and Ada Embedded System Debugger); Reusability Assistance (Reusability Library Framework); and Other (Pattern Recognition).

The STARS Foundations Projects provides a basis for the next step in the STARS Program. The current activity in the STARS Program is focused on building from the foundations projects, looking at the development of systems-in-the-large. The STARS Program Office has awarded three five-year contracts to continue this work to IBM Federal Systems (Gaithersburg, MD), Unisys (Reston, VA), and Boeing Aerospace (Seattle, WA). These contractors have begun working on an Ada software development environment that will incorporate a software-first life cycle.

The most recent presentation of the STARS Program to industry, the STARS '89 Breakthrough Initiatives Conference, was held 20 March 1989 in Atlanta, GA. This technical briefing and conference was designed to identify and stimulate proposals for innovative "breakthrough" initiative research projects. The focus of the meeting was on identifying opportunities to reduce risk within the STARS system development process. The technical areas emphasized at this conference included: the software-first life cycle; Ada environments; software quality and reliability; prototyping and incremental development; repository operations; and reuse methods. Contact Mr. Arnold R. Beckhardt at the following address for additional information on this conference:

Mr. Arnold R. Beckhardt
Software Engineering Technology, Inc.
2770 Indian River Boulevard
Vero Beach, FL 32960

(407) 569-4444

Figure 6-1. Sample DACS Newsletter

6-3

DACS BULLETIN

Software Life Cycle Support Environment

The Software Life Cycle Support Environment (**SLCSE**) is a computer-based environment of integrated software tools which supports the development and post-deployment phases of the mission critical computer system (MCCS) software life cycle in accordance with DoD-STD-2167A, Defense System Software Development.

SLCSE has been designed using a flexible and evolvable tools-based approach where a virtually unlimited number of tools can be integrated to support various software engineering methods and overall development methodologies. This "methodology independent" tool integration concept provides for the use of off-the-shelf tools as well as tools developed specifically for use within **SLCSE**. Thus, a project manager has flexibility in the choice of tools for use on a particular software development project.
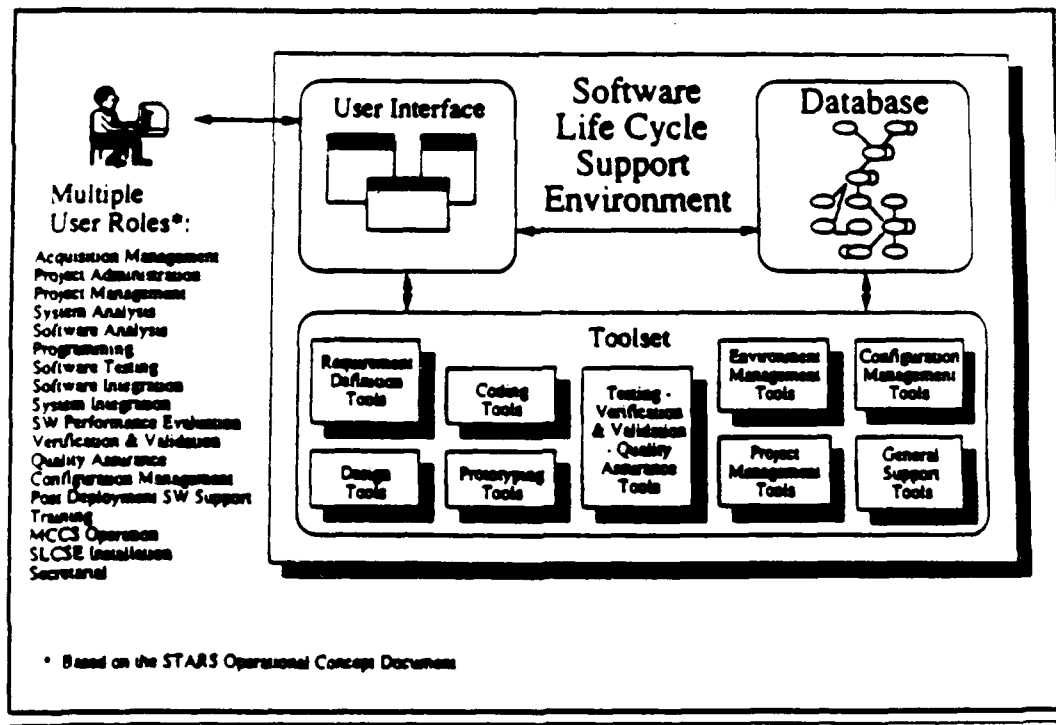
Volume IX Number 4; June 1989

Figure 6-2. Sample DACS Bulletin

6-4

NSIA and National Institute for Quality and Productivity Conferences

DACS Annotated Bibliography: Fourth Annual Supplement

The DACS Measurement Annotated Bibliography

STARS Measurement Survey Report

Ninth Minnowbrook Workshop on Performance Evaluation

RADC Knowledge Based Software Assistant

NASA Software Engineering Laboratory
Eleventh Annual Software Engineering Workshop

Orlando II, Solving the Post-Deployment Software Support (PDSS) Challenge

Software Reliability: Measurement, Prediction, Application

Software Reliability Baselines

Cost Models Overview

Tenth Minnowbrook Workshop on Software Reuse

A Descriptive Evaluation of Software Sizing Models

Assistant for Specifying the Quality of Software (ASQS)

NASA Software Engineering Laboratory
Twelfth Annual Software Engineering Workshop

Software Measurement Models; A DACS State of the Art Report

Software Design Metrics Workshop

The Sixth National Conference on Ada Technology

A Directory of Air Force Ada & JOVIAL Software Engineering Tools

Eleventh Minnowbrook Workshop

3rd Annual RADC Knowledge Based Software Assistant Conference

A Neural Net Research and Application Study

Figure 6-3.   Recent DACS Bulletin Topics

6.4 Technical Presentations

DACS personnel made three technical presentations during this contract. Technical presentations are an effective way to inform the community of the concerns and activities of the DACS and to attract new users. They also serve to identify new areas of concern in the DACS user community, new sources of data for the data acquisition program, and inputs for the newsletters and bulletins.

DACS personnel attended the second COCOMO/WICOMO User's Group Meeting at Wang Institute, Tyngsboro, Massachusetts on 29-30 May 1986. They demonstrated the Software Engineering Cost Model (SECOMO) as part of the meeting's tool fair and gave a presentation that highlighted SECOMO's features.

On 11-12 June 1986, DACS personnel attended the quarterly Life-cycle Software Engineering (LCSE) meeting at Redstone Arsenal, Alabama where they reported on the state of the art of software sizing and on data collection to support model validation and verification.

DACS personnel attended the Ninth Minnowbrook Workshop on Software Performance Evaluation on 5-8 August 1987 in Blue Mountain Lake, New York. DACS personnel gave a presentation on the validation of the Goel-Okumoto software reliability model.

6.5 Conferences Attended

In addition to the three conferences attended as presenters, DACS personnel attended two additional conferences during this contract period. Conferences are an additional source of technical information and contacts that keep DACS personnel informed on current software technology and issues.

Two DACS staff members attended the Orlando II, Solving the Post Deployment Software Support (PDSS) Challenge conference, which was sponsored by the Joint Logistics Commanders (JLC) Joint Policy Coordinating Group on Computer Resource Management. The purpose of the conference was to review

current DOD PDSS activities and to make specific recommendations relevant to software support issues.

DACS personnel also attended the 9th International Conference on Software Engineering from 31 March to 2 April 1987. Specifically, a tutorial on Software Reusability and a tutorial on Software Environments were of interest to the DACS representatives.

6.6 Summary

As a result of an extensive Current Awareness Program, several benefits have accrued to the DACS:

- DACS personnel have established a network of contacts throughout the software engineering community and are regularly invited to make presentations at conferences and symposia

- DACS personnel have developed an in-depth knowledge of the needs and concerns of the DACS user community; this knowledge is needed to prepare and publish newsletters and bulletins which are both informative and useful

- DACS personnel have developed production and quality control procedures to assure that newsletters and informational materials are of high quality in both appearance and technical content

- Through participation in technical symposia, workshops, and professional activities, DACS personnel have acquired a broad base of information regarding the activities of individuals and organizations in the field of software engineering

7.0 TASK 6 - PRODUCTS & SERVICES PREPARATION AND DISTRIBUTION

7.1 Introduction

This section summarizes the results of the tasks to produce and distribute the DACS products and services. Figure 7-1 illustrates the types of products and services offered by the DACS.

DACS Products & Services

Cost Estimation Models
U.S. Air Force Software Packages
Custom Database Searches
Custom & Standard Empirical Datasets

Technical Inquiries
DACS Participation Plans

DACS Bulletin & DACS Newsletter
Data Compendiums
Information Packages
State-of-the-Art Reports
Software Engineering Bibliographies
Technology Assessments
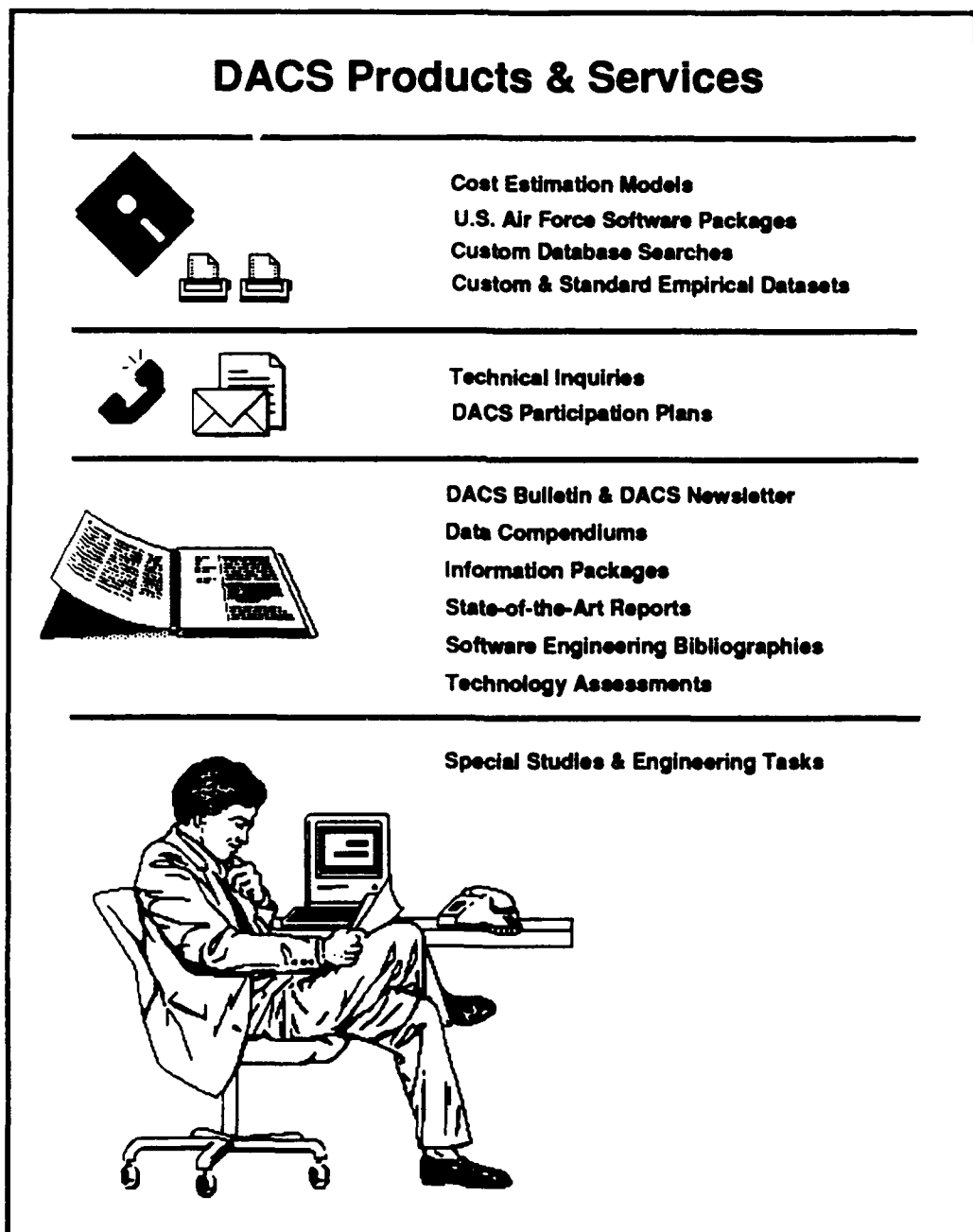
Special Studies & Engineering Tasks

Figure 7-1. Products and Services Offered by the DACS

7.2 Data Services

Computer readable and hard copy subsets of the data contained in the DACS SLED are produced and distributed in response to specific requests to aid in research efforts that require productivity, cost, complexity, error, and change data. Typically, these datasets are used to validate and refine software reliability, maintainability, and estimation models and to aid in additional data analysis studies that require empirical data.

When a dataset is requested, the description of which data items and type, the sorted order, and other information must be known. If the description is not included in the request, DACS personnel contact the user directly. During the pilot period a history of requests was kept. As patterns of use developed, standard options were identified to ease the processing load on engineering personnel. Parameterized procedures were written, allowing clerical personnel to produce data subsets for standard options.

To facilitate distribution of these datasets and their subsets, descriptive literature on SLED datasets is provided to the potential purchaser along with an order form. Upon receipt of the order, the tape or hardcopy listing is generated. The data is then sent along with a data dictionary describing the data elements.

7.2.1 Data Compendium

The DACS Data Compendium is a summarization of the data contained in the SLED. It contains descriptions of the datasets ordering information, and record formats for all the datasets described.

DACS personnel periodically update the compendium to reflect any changes made to the descriptions of data contained in the SLED. The second edition of the DACS Data Compendium was issued in March 1986 and the third edition was produced in March 1989.

7.3 Software Engineering Bibliography

Scientists and engineers depend upon well-designed bibliographies for ready access to previous work and published literature. To be most useful, entries must be comprehensively indexed with terms that are pertinent to and in common use within the community being served. In response to this need, the DACS developed the DACS Thesaurus, a standard word list used for indexing documents in the DACS library, and the DACS has published and disseminated The DACS Annotated Bibliography. The DACS bibliography set currently consists of an initial volume and six supplements. The fifth and sixth supplement were compiled during this contract. Each bibliographic volume contains a citation and abstract for each document, the DACS Thesaurus, a subject index, an author index, and a keyword-in-context listing of the document titles. Instructions for using the bibliography and document ordering information are also provided. The entire set of published bibliographies is available to DACS users.

7.4 Bibliographic Services

Bibliographic inquiries to the DACS are received in many forms: by letter, by telephone call, by visit, or by use of the bibliographic request form contained in The DACS Annotated Bibliography. The DACS conducted 83 bibliographic searches during this contract period. DACS personnel also provided consultation services on requests for bibliographic assistance.

Bibliographic inquiries are processed in a timely manner, with a search strategy generated and an on-line search of the DACS database normally made within one or two days from receipt of an inquiry. Inquiries are tracked to ensure that response time does not exceed ten days; responses may take the form of a letter or listing. Abstracts are supplied with the custom bibliographies, which are reviewed by a member of the DACS technical staff to ensure that the material is pertinent to the subject. The DACS has developed efficient search procedures and has refined them to provide more focused bibliographies in response to specific user requests.

7.5 Custom Software Tool Searches

One service offered by the DACS is the custom search of the Software Tool Information (STI) Database. The information provided on each software tool included: tool title and/or acronym, classification, features/functions, stage/date of development, applicability, implementation language, portability, size, hardware, restrictions, availability, an abstract or summary, documentation, contact, and developer. The DACS performed 51 custom tool searches during the contract reporting period.

7.6 Technical Inquiries

Technical inquiries to the DACS are received by mail, telephone call, or in person and are processed on a daily basis. Information requests range from general inquiries to very specific questions.

Technical inquiries are answered in different ways, depending on the nature of the inquiry and the complexity of the reply. Responses include:

- Custom bibliographic search on the subject of interest

- Summary information based on a preliminary analysis of the subject literature

- Production of a subset of a DACS database

- Distribution of relevant DACS literature

- Referral to other sources of information

Certain inquiries of a very specific nature may not be answerable in terms of a dataset, a bibliographic search, or a published DACS product, but may be answered relatively quickly by a DACS specialist. For such inquiries, the DACS provides engineering services in the form of technical consulting, with additional DACS products included, as appropriate. A total of 4,593 technical inquiries were processed during this contract period. The DACS responds to technical inquiries within ten working days.

7.7 State-of-the-Art Reviews

An effective approach to stimulating user interest in the DACS is through the design, preparation, and distribution of products that provide authoritative sources of information needed throughout the software engineering community. State-of-the-Art Reviews (SOARs) are intended to consolidate and synthesize information on a specific high-interest technology area from multiple sources into a single document containing all pertinent information in a condensed, easy-to-assimilate form. SOARs are most useful when the topic areas they cover have received a substantial amount of attention by scattered researchers (indicating a broad interest), but the research objectives and results have not been previously consolidated.

The DACS produced six SOARs during this contract:

* Software Measurement Models
* Glossary of Software Quality Terms
* FORTRAN Coding Standard
* Verification Techniques Applied to Ada
* Software First System Development Methodology
* Software Quality/Measurement Assessment

7.8 Technology Assessments

Technology assessments are special studies that generally exceed the scope of a technical inquiry response, but are usually shorter than a state-of-the-art review. They are directed toward evaluating and synthesizing the latest available information resulting from recent research and development findings or are comparative assessments based on technical characteristics. For example, a technology assessment may involve the extraction and compilation of specialized data from several independent works to evaluate their differences; it may be a synopsis of pertinent findings from available documentation; or it may compare different approaches to a life-cycle process, using documented specifications and results.

The six technology assessments produced during this reporting period include:

- Review of Software Cost Models
- Issues of Software Portability
- An Evaluation of Software Sizing Models
- Application of Value Engineering
- A Software Reliability MetaModel
- A Review of Software Safety Analysis

8.0 TASK 7 - PREPARING, DISTRIBUTING, AND IMPROVING PRODUCTS AND SERVICES

8.1 Preparing and Distributing Products and Services

A vital component of a responsive Information Analysis Center is its
ability to provide products that reflect the needs of its users. The
following new products and services were introduced to DACS users during this
reporting period:

- Automated Measurement System (AMS): developed for Rome Air
  Development Center (RADC) by the Harris Corporation, the AMS
  collects, stores, analyzes, and generates reports of software
  measurement data.

- Ada Compilation System (ACS): developed for Rome Air Development
  Center (RADC) by Intermetrics, Inc., the ACS is the Air Force's fully
  self-compiled Ada compiler.

- Air Force Armament Laboratory (AFATL) Ada Compiler and Debugger:
  developed for AFATL by Florida State University, the Ada cross-
  compiler's and debugger's source code.

- Software Development Standard (SDS) Fragments/Structures Database and
  Reports: produced as the results of a DACS special study for the
  STARS Joint Program Office that examined the software engineering
  process as prescribed by DOD-STD-2167 to determine the fragments of
  information that are either produced or required by software
  engineering activities.

- DOD-STD-2167A, Software Development Standard Documentation Set:
  produced under direction of the Computer Software Management Subgroup
  of the Joint Logistics Commanders, the set contains the DOD Military
  Standard for Defense System Software Development (DOD-STD-2167A,
  February 1988); the supporting Data Item Descriptions (DIDS, February
  1988); and the DOD Military Standard for the Defense System Software
  Quality Program (DOD-STD-2168, April 1988).

- Software Engineering Cost Model (SECOMO): developed by IIT Research
  Institute (IITRI) for the U.S. Army's Life-Cycle Software Engineering
  Centers (LCSES), SECOMO is an interactive cost estimation tool for
  calculating the total technical and support manpower requirements of
  a LCSE.

- <u>The DACS Measurement Annotated Bibliography</u>: produced by the DACS for Rome Air Development Center, the bibliography contains a collection of citations for 660 software measurement-related documents.

- <u>DACS Data Compendium</u>: the compendium provides a description of the software experience data contained in the DACS' Software Life-Cycle Empirical Database (SLED).

Figure 8-1 contains a complete list of the products and services offered by the DACS.

8.2 Improvement of DACS Products and Services

The products and services of the DACS are continually reviewed and assessed with the objective of enhancing both the DACS and its products and services. Concerns of the enhancement efforts extend to all facets of the DACS operations, including improving the technical quality of DACS products, increasing the scope of data coverage, improving analysis routines, improving data handling capabilities, operating more efficiently, and increasing services to the user community. During this contract period, significant improvements have been made in several areas. These improvements include:

- Revising the <u>User's Guide to DACS Products & Services</u> (DACSGUIDE) to include a new section which describes the various magnetic tape formats available for selected DACS products, and a new section which explains the procedures to obtain export-controlled technical data from the DACS; the new DACSGUIDE is now included in the <u>DACS Information Package</u>

- Enhancing SECOMO to include an improved user interface, on-line help, and an expanded users manual

- Establishing a procedure for verifying the Defense Logistics Services Center (DLSC) Qualified U.S. Contractor Access List (QCAL) numbers for all DACS users ordering export-controlled materials

Technical Services
Technical Inquiries
Bibliographic Searches
Tool Searches

Dataset Distribution
ARF Error Dataset
Composite Productivity Dataset
Custom Datasets
DACS Productivity Dataset
NASA/AMES Dataset
NASA/SEL Dataset
PAVE PAWS O&M Dataset
Software Reliability Dataset
V&V Dataset

General Information Inquiries
DACS Guide to User Services
DACS Guide to Bibliographic Services
DACS Information Package
Ada Compiler Evaluation Capability (ACEC) Information Package
Ada Compilation System (ACS) Information Package
Air Force Armament Laboratory Ada Compiler (AFATL) Information Package
Automated Measurement System (AMS) Information Package
Common Ada Missile Packages (CAMP) Information Package
Sample Citations from DACS Databases
DACS Newsletter
DACS Bulletin
DACS Newsletter Mailing List
DACS Participation Plan Subscriptions

Software Tools
Software Engineering Cost Model (SECOMO)
Ada Compilation System (ACS)
Ada Compiler Evaluation Capability (ACEC)
Goel-Okumoto Model
Automated Measurement System (AMS)
Common Ada Missile Packages (CAMP) Parts
Common Ada Missile Packages (CAMP) Benchmarks
Common Ada Missile Packages (CAMP) AMPEE
AFATL ADA Compiler/Interactive Debugger

Document Distribution
A Review of Software Maintenance Technology
Ada Compilation Evaluation Capability (ACEC) Documents
Ada Compilation System (ACS) Documents
A Directory of Ada & JOVIAL Software Tools
AIAA Tools Survey
Automated Measurement System (AMS) Documents
Common Ada Missile Packages (CAMP) Documents
DACS Annotated Bibliography
   First Annual Supplement
   Second Annual Supplement
   Third Annual Supplement
   Fourth Annual Supplement
   Fifth Annual Supplement
   Sixth Annual Supplement
DACS Conversion Data Collection Forms
DACS Data Compendium
DACS Glossary
DACS Measurement Annotated Bibliography
DACS Software Engineering Data Collection Package
DACS Productivity Data Collection Forms
DACS Software Tool Description Forms
DOD-STD-2167/2167A Documentation Set
Glossary of Software Quality Terms
NASA/SEL Data Collection Forms
Quantitative Software Models
Software Data Collection and Analysis
Software Life Cycle Tools Directory
Software Measurement Models
Software Quality Standard (DRAFT) DOD-STD-2168
STARS Interim Software Engineering Data Collection Forms
STARS Measurement Survey Summary
Software Sizing Models

Figure 8-1.   The DACS Provides Technical Services, Datasets, Information,
Documents, and Software Tools to its Users

8.3 Continuing Assessment of User Needs

The DACS uses two methods to determine the needs of its users: surveys
to identify user requirements, and historical records of user requests.
These two activities have been pursued to ensure that DACS products and
services meet the needs of the user community.

The most effective mechanism for soliciting users' comments and suggestions has proven to be periodic surveys published as part of the DACS Newsletter. The most recent survey, which appeared in the March 1988 issue, reached over 4,000 users and produced over 300 responses. The survey requested information which characterized respondents by type of organization and job title. In addition, survey respondents were asked to identify and evaluate the DACS products and services which they have used, and to indicate areas of software engineering in which they are interested. Survey responses provided insight into the needs, interests, and concerns of DACS users and will be used to improve existing products and services and foster the introduction of new products and services responsive to DACS users' needs. Figure 8-2 illustrates the survey form used to solicit DACS users' comments and suggestions.

The majority of the March 1988 survey respondents classified themselves as managers working in industrial organizations. Respondents representing Government agencies, primarily DOD, have increased from earlier surveys. Figure 8-3 summarizes these responses; it illustrates the variety of organizational and user types served by the DACS.

The most frequently used DACS product, as reported by the respondents, is the DACS Newsletter. Other products and services in order of popularity include the DACS Bulletin, state-of-the-art reports and technology assessments, software engineering bibliographies, and technical inquiries. Over 75% of the respondents rated the quality of DACS products and services from good to excellent. The software engineering areas of greatest interest to the respondents are software tools, software development techniques, software maintenance techniques, software standards, life-cycle management, and software measurement and metrics.

DATA & ANALYSIS CENTER FOR SOFTWARE
1988 USER SURVEY

We need some advice from you, to help us make the "DACS Newsletter" and other DACS products/services more useful and more interesting to our users. Please take a few moments to share your opinions and observations about the "DACS Newsletter" and other DACS products/services.

1. (a) Please check the item which best describes your government or business activity:

    [ ] Air Force   [ ] Army   [ ] Navy   [ ] Marines
    [ ] Other DOD (Specify): _____
    [ ] Other Government (Specify): _____
    [ ] Academic   [ ] Consultant   [ ] Industry   [ ] Research
    [ ] Other (Specify): _____

    (b) If you checked Academic, Consultant, Industry, or Research, are you a DOD contractor?   [ ] Yes   [ ] No

2. Which title most closely describes your position?

    [ ] Analyst/Programmer   [ ] Consultant   [ ] Instructor   [ ] Researcher
    [ ] Software Engineer   [ ] Manager   [ ] Librarian   [ ] Quality Assurance
    [ ] Other (Specify): _____

3. How did you learn about DACS products and services?

    [ ] "DACS Newsletter" Mailed to Me   [ ] Colleague in My Organization
    [ ] Colleague in Another Organization   [ ] Conference Presentation
    [ ] Visit from DACS Representative
    [ ] Other (Specify): _____

4. Please evaluate the following DACS products and services:

	Excellent	Good	Poor	Never Used
DACS Newsletter				
DACS Bulletin				
Software Engineering Bibs				
Empirical Data Sets (Raw Data)				
Custom Database Searches				
Technical Reports				
Technical Inquiries				
Other (Specify): _____				

5. (a) Have you consulted other sources for the type of software information provided by the DACS?

    [ ] No   [ ] Yes (Specify): _____

    (b) Which statements do you find are true of the other sources?

    [ ] A good complement to DACS products and services.
    [ ] There is some overlap with DACS products and services.
    [ ] There is a great deal of overlap with DACS products and services.
    [ ] Not as satisfactory in amount/quality as DACS products and services.
    [ ] More satisfactory in amount/quality of DACS products and services.

6. (a) Has your organization experienced any cost avoidance or cost saving benefits as a result of using DACS products or services?

    [ ] No   [ ] I do not know   [ ] Yes

    (b) Please describe the benefits your organization has experienced:

    _____

7. Which of the following software engineering areas are of interest to you?

    [ ] Software Measurement/Metrics
    [ ] Software Standards
    [ ] Software Tools
    [ ] Software Development/Maintenance Techniques & Methods
    [ ] Software Life Cycle Management
    [ ] Human Factors
    [ ] Application Technologies
    [ ] Other (Specify): _____

8. What new products or services would you find useful?

    [ ] Software Engineering Training
    [ ] Software Engineering Conferences
    [ ] Assessment Reports on Software Engineering Topics
    [ ] State of the Art Reports on Software Engineering Topics
    [ ] Other (Specify): _____

9. Please evaluate the following regular sections of the "DACS Newsletter":

	Always Read	Sometimes Read	Never Read	Do Not Recall
Sources of Information				
Calendar of Events				
New DACS Products				
Summaries of DACS Products				
Noteworthy				
Conference Summaries				

10. Additional Comments: _____

_____
_____
_____

Figure 8-2. March 1988 User Survey Form

User Types as Indicated by Job Title

Other

Librarian

Consultant

Manager

Instructor

Researcher

Quality
Assurance
Engineer

Analyst/Software
Engineer

Organizational Types

Research   Other

University

Consulting

Industrial

Government
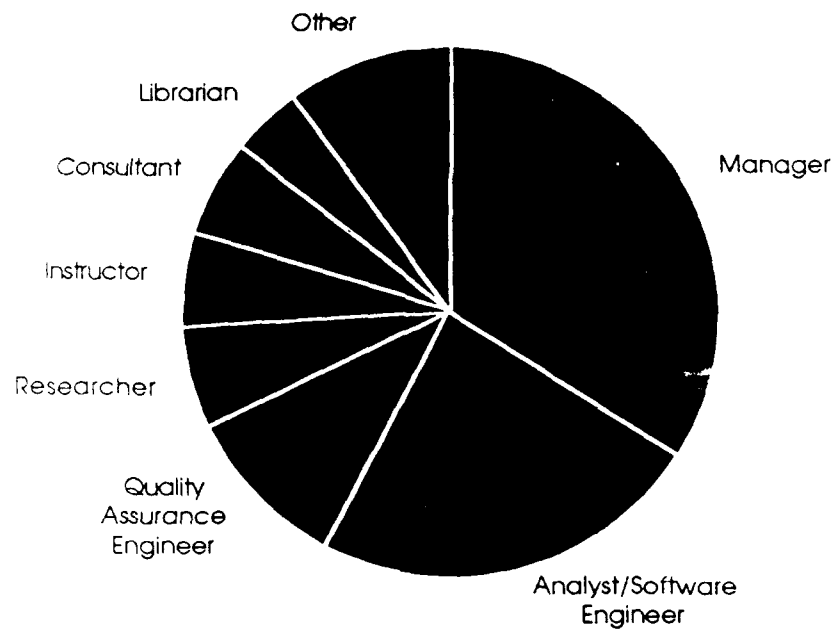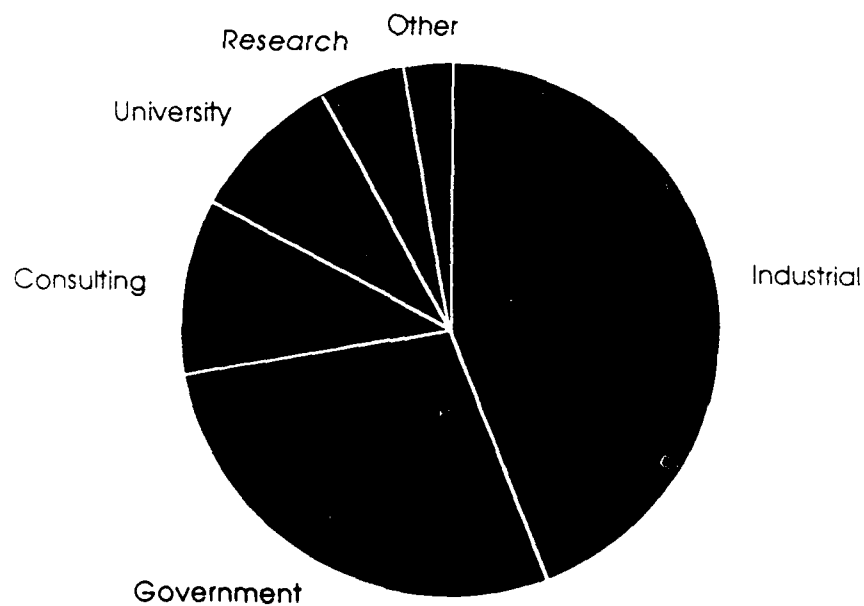
**Figure 8-3.   March 1988 Survey;   User and Organizational Types
Served by the DACS**

Figure 8-4 illustrates the new Information Request Reporting Form, which is used to track and record services performed by the DACS in terms of the user, the type of service provided, and the resources expended. This information is transferred to the User Profile Database (UPD) to accumulate the types of services provided each user. Periodic tabulations of total user interactions with the DACS are generated from the database and are correlated with the results of user surveys to evaluate DACS responsiveness to user needs.

This is an area where a great improvement has been made over previous contracts. Now all service requests are processed on-line as opposed to the previous method of paper forms. The new system is highly automated and provides on-line query capability and order generation. It also provides the DACS with a mailing list capability and the easy production of order statistics.

Figure 8-4. DACS Information Request Reporting Form

9.0 TASK 8 - PROMOTIONAL ADVERTISING

9.1 Promotion to New Users

Since it is unlikely that all potential users of the DACS have been identified or are on the newsletter mailing list, ongoing efforts are necessary to reach potential users. Potential users are defined as organizations or individuals known to participate in research and development or to use materials within the scope of DACS products and services.

Promotion to persons and organizations in the potential user category was accomplished using the following techniques:

- Placement of press releases and announcements in journals, magazines, and newspapers

- Presentations at conferences and symposia

9.1.1 Promotion by Use of Free Publicity

There are several newspapers and magazines circulated free to software engineers that will print press releases and new product information at no charge to the DACS. In addition to sending announcements, fliers, and brochures to several of these journals, editors of other journals and magazines requested information from the DACS on its products and services.

The DACS produced and distributed free promotional fliers with information packages to reach potential users. A flier, shown in Figure 9-1, which provides new users with an overview of the DACS, its mission, and its scope was introduced during this contract period. Figure 9-2 contains a sample of other promotional fliers and brochures that were produced during this contract period.

The DACS Newsletter continued to be used as a regular promotional device. Each newsletter contains book reviews, state-of-the-art surveys, conference announcements, featured DACS products, and capsule summaries of selected products and services. The graph in Figure 9-3 shows user receipts for this

contract period. There is a noticeable increase in receipts when quarterly newsletters or brochures are distributed, indicating that these distributions are effective means of promoting DACS products and services.

Data & Analysis Center for Software

Data & Analysis Center for Software ● RADC/COED ● Griffiss AFB, NY 13441-5700 ● (315) 336-0937 Autovon 587-3396

The Data & Analysis Center for Software (DACS) is a Department of Defense (DoD) Information Analysis Center (IAC) which serves as a center of readily available software engineering technical expertise and information. The DACS was initiated in 1978 and is administered by the Defense Technical Information Center (DTIC) under contract number F30602-86-C-0111. The Rome Air Development Center (RADC) serves as the technical monitor of the DACS and the U.S. Air Force Systems Command serves as its contracting agency. The DACS is operated by IIT Research Institute (IITRI).

MISSION & SCOPE

The DACS was established to provide a focal point for software development and experience data and information within the field of software engineering. Functions of the DACS include acquiring, reviewing, evaluating, storing, analyzing, and disseminating data and Scientific and Technical Information (STINFO) concerning developing and maintaining computer software, software engineering, and software technology. DACS responsibilities include: developing, maintaining, and expanding a database of empirical data collected on software development and maintenance projects; producing and distributing subsets of this database in support of software technology research; maintaining a software technology information base consisting of current reports, articles, and technical papers; analyzing data and producing reports of interest to the DACS user community; producing and distributing software technology assessments and state-of-the-art reports; servicing bibliographic and technical inquiries; and conducting an active user awareness program.

The DACS exists for the purpose of supporting software technology research, assisting in the transition and application of new software technologies, serving as an authoritative source of information concerning the state-of-the-art in software engineering, and serving as a repository of information on the development and maintenance of computer software. The objectives of the DACS include:

Encouraging the exchange of software technology information among DoD, other government agencies, the private sector, and academia

Supporting software technology research by providing a centralized source of software life cycle data, documents, resources, and expertise

Facilitating the use of research project results in a cost-effective manner

Increasing the productivity of software producers and the quality of their products by improving the transfer of software engineering technology

Assisting in distributing new technology throughout the U.S. industrial base, thereby expanding its capability and competitive posture

Providing scientific and technical information analysis services to the DACS user community in areas relating to software technology needs, developments, and trends

Minimizing duplication of software technology research

An Overview of the Data & Analysis Center for Software

Figure 9-1.   The DACS Overview Flier

Common Ada Missile Packages

The Data & Analysis Center for Software (DACS) distributes three Common Ada Missile Packages (CAMP) products developed by the McDonnell Douglas Astronautics Company under contract to the Air Force Armament Laboratory; Eglin Air Force Base, Florida. These products include CAMP Parts, CAMP Armonics Benchmarks, and the CAMP Ada Missile Parts Engineering Expert System (AMPEE).

The CAMP Parts are 444 reusable Ada components organized into 35 Top-Level Computer Software Components (TLCSCs) which contain 137,000 source lines of Ada code (including comments, package specifications, package bodies, and test code). CAMP Parts are distributed on two ANSI standard labeled 9-track 1600 bpi tapes. The Version Description Document and the User's Guide for the Missile Software Parts of the Common Ada Missile Packages (CAMP) Project are shipped with the CAMP Parts tapes.

The CAMP Armonics Benchmarks are used to evaluate Ada and processor implementations in the armonics domain. The Benchmarks represent typical armonics applications and include missile operational parts as well as support parts from the mathematical domain. The tests establish the correctness of compiler implementations and measure performance in size and speed of generated code. The CAMP Armonics Benchmarks are distributed on one ANSI standard labeled 9-track 1600 bpi tape. The CAMP Armonics Benchmarks Suite, and the CAMP Armonics Benchmarks Suite Inventory are shipped with the CAMP Armonics Benchmarks tapes.

The CAMP Ada Missile Parts Engineering Expert System (AMPEE) provides mechanisms for identifying potentially applicable Ada software parts, obtaining specific information about the parts, and generating Ada components based on the catalogued parts. These mechanisms correspond to the three main AMPEE System functions: parts identification, parts cataloguing, and component construction. AMPEE runs on a Symbolics 3620 under Genera (Version 7.1) and requires the Automated Reasoning Tool (ART; Version 3.0)*. AMPEE is distributed on two ANSI standard labeled 9-track 1600 bpi tapes. AMPEE System and the Software User's Manual for the Ada Missile Parts Engineering Expert System (AMPEE) of the Common Ada Missile Packages (CAMP) Project are shipped with the CAMP AMPEE tapes.

NOTE: The ART is not available from the DACS, but may be ordered from the Inference Corporation, (213) 417-7997.

Common Ada Missile Packages (CAMP) Information Sheet & Ordering Instructions

Automated Measurement System

The Data & Analysis Center for Software (DACS) distributes the Automated Measurement System (AMS) software and documentation developed by the Harris Corporation under contract to the Rome Air Development Center (RADC/COEE); Griffiss Air Force Base, New York.

The AMS collects, stores, analyzes, and reports software measurement data for use by software acquisition and project personnel. This tool is based on the RADC Software Quality Framework data collection worksheets for software quality measurement. The AMS runs under the VMS operating system on DEC VAX systems.

Automated Measurement System (AMS) Highlights

The AMS supports all phases of the software life cycle (as defined in DoD-STD-2167, Defense Systems Software Development Standard).

The framework underlying the AMS is a set of software quality factors: Correctness, Efficiency, Flexibility, Integrity, Interoperability, Maintainability, Portability, Reliability, Reusability, Survivability, and Verifiability.

The AMS automatically collects quality data from the Systems Requirements Evaluation Methodology (SREM), Software Design and Documentation Language (SDDL), Fortran 77, and Ada.

AMS data entry is controlled by a General Forms Manager.

The AMS collects resource expenditures, problem reports, and quality data.

AMS reporting facilities include bar graphs, Kiviat charts, historical trend graphs, and a variety of detailed and summary reports.

The AMS User Interface is menu driven by the novice user or command language driven by the expert user.

Metrics, forms, reports, and the underlying framework details can be easily modified.

Automated Measurement System (AMS) Information Sheet & Ordering Instructions

Figure 9-2. Two Sample Promotional Fliers
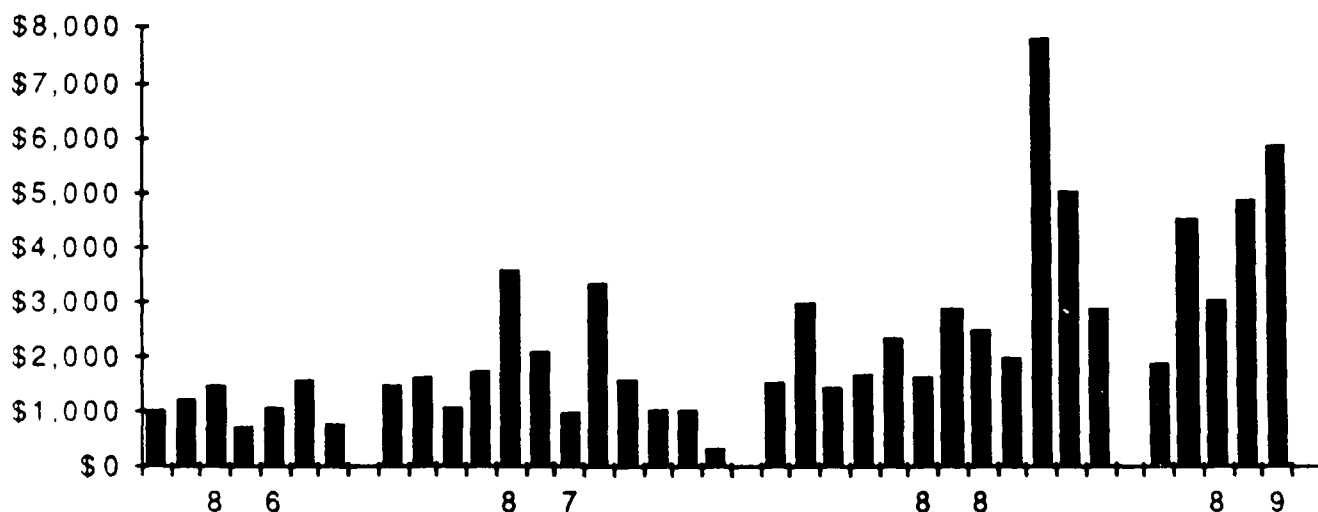
Figure 9-3. DACS User Receipts April 1986 to June 1989

9.1.2 Presentations at Conferences and Symposia

This topic is more fully discussed under Task 5, Current Awareness. It is only noted here that presentations at conferences and symposia provide an excellent opportunity for individual discussions, as well as personal communication with groups of potential users concerning the DACS and its products and services.

10.0 TASK 9 – SPECIAL STUDIES AND PROJECTS

10.1 Introduction

There are many problems related to software technology that can be solved through the full service capabilities provided by the DACS. Many of these are sizable, requiring a substantial expenditure of engineering resources to accomplish. The DACS can make a significant contribution in this area to increase the productivity of software engineers and researchers and at the same time make efficient use of accumulated information and resources.

The DACS can best serve its user community by identifying those areas for which there is a high need for the expertise and other resources of the DACS and performing special studies tailored to the individual needs of particular organizations. Figure 10-1 features the brochure developed during this contract period to describe the process of initiating special study efforts to potential users of this service. The following subsections describe the special studies performed by the DACS during this contract.

Software Engineering Cost Model Enhancement, U.S. Army Missile Command (MICOM), IITRI Project A06173

The purpose of this effort was to assist the Battlefield Automation Management Directorate in refining the Software Engineering Cost Model (SECOMO) to increase its usability in the U.S. Army Missile Command (MICOM) software engineering environment.

DACS Special Tasking Procedure

INTRODUCTION

The Data & Analysis Center for Software (DACS) is a Department of Defense (DoD) Information Analysis Center (IAC) which serves as a center of readily available software engineering technical expertise and information. The DACS was initiated in 1978 and is administered by the Defense Technical Information Center (DTIC) under Contract Number F30602-89-C-0111. The Rome Air Development Center (RADC) serves as the technical monitor of the DACS and the U.S. Air Force Systems Command serves as its contracting agency.

The DACS is mandated to pursue and conduct special tasks for DoD Agencies. These tasks must be relevant to the DACS' primary field of interest, software engineering, which encompasses a broad technical area. These special tasks are separately funded by the Sponsoring Agency.

The DACS is operated by IIT Research Institute (IITRI), which provides the technical and administrative staff for conducting most of the special tasks undertaken through the DACS. Other organizations or consultants, however, can be tasked to provide expertise or facilities required to address specific problems through subcontracts.

Since the contract under which the DACS operates is tailored to accommodate special tasks beyond the basic Center operations, the procedure necessary to initiate such a task is quite simple, requiring a minimum of contractual documentation and turn-around time.

PROCEDURE

The procedure to be followed by a DoD Agency in initiating a special task through the DACS is a four step process. The procedure is outlined below; see the other side of this page for a visual presentation of this procedure.

STEP 1: Representatives from the DACS and the Sponsoring Agency meet to discuss the technical tasks, approach, timeframe, and available funding for a proposed special task.

The DACS prepares a Draft Proposal (Project Plan) and submits it to the Sponsoring Agency for review and acceptance. Each proposal will contain an approach, cost estimate, performance schedule, and a list of deliverables.

After acceptance of the proposal by the Sponsoring Agency, copies of the Draft Proposal are delivered to the RADC Contract Officer's Technical Representative (COTR), Mr. John Palaimo, for consideration.

STEP 2: The Sponsoring Agency initiates its funding action and prepares a DD Form 448, Military Interdepartmental Purchase Request (MIPR). See the attached sampl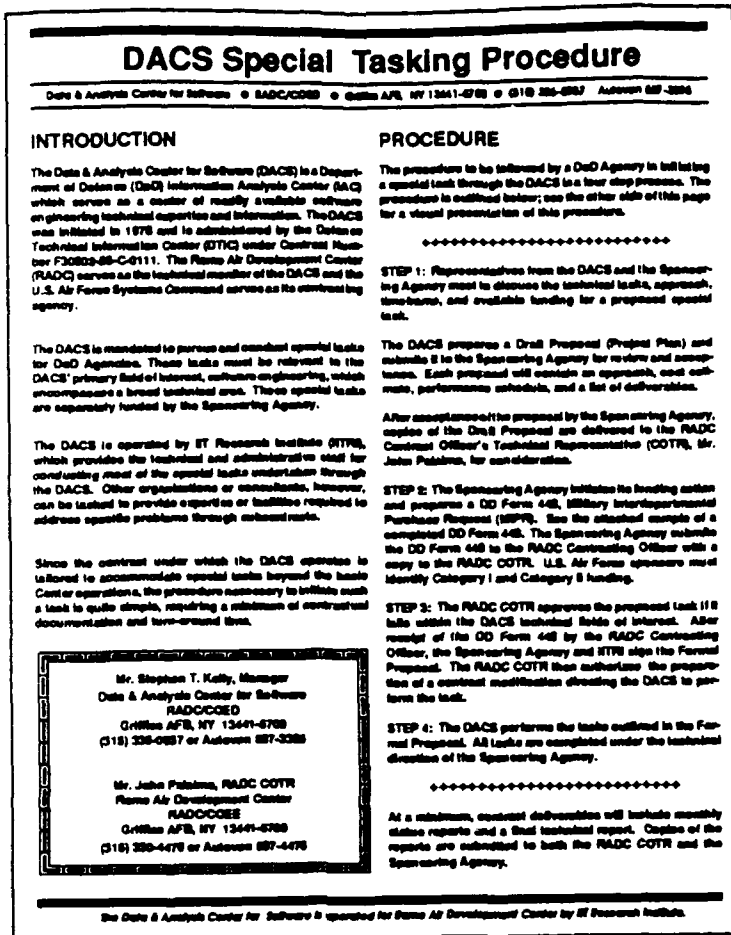e of a completed DD Form 448. The Sponsoring Agency submits the DD Form 448 to the RADC Contracting Officer with a copy to the RADC COTR. U.S. Air Force sponsors must identify Category I and Category II funding.

STEP 3: The RADC COTR approves the proposed task if it falls within the DACS technical fields of interest. After receipt of the DD Form 448 by the RADC Contracting Officer, the Sponsoring Agency and IITRI sign the Formal Proposal. The RADC COTR then authorizes the preparation of a contract modification directing the DACS to perform the task.

STEP 4: The DACS performs the tasks outlined in the Formal Proposal. All tasks are completed under the technical direction of the Sponsoring Agency.

At a minimum, contract deliverables will include monthly status reports and a final technical report. Copies of the reports are submitted to both the RADC COTR and the Sponsoring Agency.

Mr. Stephen T. Kelly, Manager
Data & Analysis Center for Software
RADC/COED
Griffiss AFB, NY 13441-5700
(315) 336-0937 or Autovon 587-3395

Mr. John Palaimo, RADC COTR
Rome Air Development Center
RADC/COEE
Griffiss AFB, NY 13441-5700
(315) 330-4476 or Autovon 587-4476

The Data & Analysis Center for Software is operated for Rome Air Development Center by IIT Research Institute.

Figure 10-1. DACS Special Tasking Procedures Brochure

Software Engineering and Ada Language Studies; Phases IV & V, U.S. Army Materiel Command (AMC) Battlefield Automation Office, IITRI Projects A06175 and A06207

The DACS developed a version of the Constructive Cost Model (COCOMO) for use at AMC's Life-Cycle Software Engineering Centers (LCSEs). The Software Engineering Cost Model (SECOMO) was tailored to the LCSE environment and extended to predict indirect development costs. The DACS has developed a number of enhancements and modifications to this model to reflect the current LCSE concept and operational requirements. In addition, software cost estimation training seminars were presented to the centers to ensure proper utilization of SECOMO.

<u>SDS Fragments/Structures: Phase IIIC</u>, STARS Joint Program Office, IITRI Project A06177

In an effort performed for the Software Technology for Adaptable, Reliable Systems (STARS) Joint Program Office, the DACS applied its standards knowledge to the software life-cycle by performing a thorough examination of the software life-cycle and its products, as specified by DOD-STD-2167, its associated Data Item Descriptions (DIDs), and the STARS Interim Data Collection Forms. The DACS examined the top-level data items required and developed consistent definitions for data fragments and attributes of fragments, and developed a mapping of data fragments into life-cycle activities.

<u>U.S. Air Force Ada/JOVIAL Software Engineering Tools Survey</u>, Air Force Systems Command (AFSC) Electronic Systems Division (ESD), IITRI Project A06178

In an effort for ESD, the DACS surveyed, classified, analyzed and evaluated information on software life-cycle tools used throughout the Air Force to support systems developed in the Ada and JOVIAL programming languages.

<u>Support Services for the RADC Knowledge Based Software Assistant (KBSA) Conference</u>, U.S. Air Force, Rome Air Development Center (RADC/COES), IITRI Project A06180

This effort provided support for the preparation and execution for the first Knowledge Based Software Assistant (KBSA) Conference sponsored by the Rome Air Development Center. The goals of this meeting were to make the community aware of RADC's efforts in the KBSA and automated software development/software maintenance areas, and to encourage the exchange of technical information and data on Government-funded efforts in these areas.

<u>Ada Verification System (AVS) Studies</u>; Defense Communications Engineering Center, IITRI Project A06184

The objective of this study was to develop an integrated, coherent set of recommendations conforming to an Ada Verification System (AVS), which will provide a foundation for the high-level design of an optimal AVS. An accurate list of Ada constructs and the existing verification technology for each were developed.

<u>Development of a Structured Analogy Approach to Software Size and Cost Estimation</u>, U.S. Army Cost and Economic Analysis Center (CEAC), IITRI Project A06189

The objective of this study was to provide a foundation that cost analysts can use to arrive at an appropriate software cost estimate for the development of C^3I systems. The uncertainty present in software cost estimating stems from a lack of analyst experience in developing software cost estimates and of definitive information during the concept and design phases of software development, which generally results in size estimates of low validity.

<u>Signal Intelligence (SIGINT) Sensor Processing Dissemination</u>, U.S. Air Force, Rome Air Development Center (RADC/IRAP), IITRI Project A06192

The objective of this study was to continue the software quality evaluation of the Dynamic Ground Target Simulator (DGTS) and the Signal Intelligence (SIGINT) support software, with emphasis on real-time concurrent processing and its impact on the software quality of the two systems. In addition, this effort identified the software engineering requirements for interfacing SIGINT software and DGTS with the RADC Battle Management Laboratory.

Documenting Management of Mission Critical Computer Resources and Battle Automated Systems, U.S. Army Communications & Electronics Command (CECOM), IITRI Project A06195

The objective of this effort was to describe Department of the Army management activities for mission critical computer resources in battlefield automated systems. This description, in the form of an addendum to an existing Army document, Army Pamphlet 11-25, "Life-Cycle System Management for Army Systems," describes management activities throughout the entire software life-cycle to assist different levels of management with the specific oversight activities occurring and various milestones in the system development.

Software Engineering Support for Life-Cycle Software Engineering Centers, U.S. Army Armament, Munitions, and Chemical Command (AMCCOM), IITRI Project A06196

The objectives of this project were to assist the U.S. Army Armament, Munitions and Chemical Command (AMCCOM) in refining, calibrating and validating the Software Engineering Cost Model (SECOMO) for AMCCOM's environment, analyze indirect software costs and determine relationships to direct software development and maintenance costs, and investigate software sizing models to improve software cost estimation.

Development of a Software Oriented Value Engineering Program for CECOM Life-Cycle Software Engineering Centers, U.S. Army Armament, Munitions, and Chemical Command (AMCCOM), IITRI Project A06198

The objective of this effort was to develop a value engineering program that can be applied to CECOM's Life-Cycle Software Engineering Centers (LCSE). A draft CECOM LCSEC Value Engineering Program was developed and candidate applications for the Value Engineering Program were identified during this effort.

A Descriptive Evaluation of Software Sizing Models, U.S. Air Force Cost Center (AFCCE), IITRI Project A06205

The DACS performed an effort for the U.S. Air Force Cost Center in reviewing and critiquing software sizing models. The analysis included an evaluation of the hardware requirements, required inputs, underlying methodology, and accessibility of each model. To gain additional insight into the strengths and weaknesses of the models, they were exercised against an established Air Force system.

Conceptual Analysis of a Software Engineering Methodology for Naval Sea Systems Command (NAVSEA) Computing Systems, Naval Sea Systems Command (NAVSEA), IITRI Project A06210

The objective of this study was to review the Naval Sea Systems Command's (NAVSEA) information resources to identify those methods, tools, and techniques that stand to maximize effectiveness and efficiency of use of those resources. The approach was to document existing state-of-the-art and state-of-practice followed by distillation of the findings into a recommended suite of methods, tools, and techniques for NAVSEA. NAVSEA had particular interest in assessment of Ada as a higher order language for their use.

Support for the 2nd Annual RADC Knowledge Based Software Assistant (KBSA) Conference, U.S. Air Force, Rome Air Development Center (RADC/COES), IITRI Project A06211

The DACS provided support for the 2nd Annual Knowledge Based Software Assistant (KBSA) Conference held 18-20 August 1987 in Utica, New York. Eighty-five (85) participants from 34 industrial and Government agencies met to review progress on the RADC research program for developing a knowledge-based life-cycle paradigm for large software projects. The paradigm is based on formalization and machine capture of all software-related decisions and subsequently applying knowledge-based reasoning to assist with decision making.

Reliability and Maintainability Standards Development on Software Testing and Evaluation, Naval Avionics Center, IITRI Project A06217

The objective of this study was to develop a company standard for the Naval Avionics Center (NAC) that covers software testing. This standard is one of many NAC Reliability and Maintainability (R&M) standards developed for all aspects of hardware and software R&M.

NAC R&M standards describes NAC-specific tasks and activities, identifies organizational responsibilities within NAC, and provides tutorial information to aid the organizations in performing their assigned tasks. Tasks are derived from Military and Defense Department standards and regulations.

The R&M standard on Software Test & Evaluation contains testing activities and responsibilities for each phase of the development life-cycle in accordance with DOD-STD-2167. Other topics addressed in the standard and tutorial sections are test documentation, test methodologies, test tools, test case design techniques, and test execution procedures.

Design of a Software Quality and Productivity Laboratory, U.S. Air Force, Rome Air Development Center (RADC/COEE), IITRI Project A06218

This special study was performed for RADC, Command and Control Directorate, Software Engineering Branch, to develop the requirements, specifications and design for a software quality and productivity laboratory. The study also identified measurement related experiments and performed the experiments to gain additional insight into the requirements of the laboratory.

Knowledge-Based Front-End to the SECOMO Software Cost Estimation Model, U.S. Army Materiel Command (AMC)/U.S. Army Communications & Electronics Command (CECOM), IITRI Project A06224

The objective of this effort was the development of a knowledge-based front-end to the SECOMO software cost estimation system. The SECOMO system was developed by the DACS under earlier special studies for AMC.

The knowledge-based front-end to SECOMO incorporates two tiers of expert assistance into the SECOMO model. The first tier assists the user in selecting the correct mode and correct values for the 17 cost drivers and provides instructions on how to properly count lines of code. The second tier performs counting checks among parameters and interacts with the user to resolve discrepancies and ambiguities.

Development of a Software First Methodology, U.S. Army Communications & Electronics Command (CECOM), IITRI Project A06225

This effort was performed for the Army Communications Electronics Command (CECOM) to develop a methodology for software development. The methodology embraces the concepts and methods of a system development in which software requirements are developed before hardware requirements are established.

The software first system development methodology uses existing tools and components of other methodologies where appropriate. The principle goal of software first is to reduce total life-cycle costs for major software-dependent DOD systems. This is the same goal as that of many recent and current methodologies. Therefore, other methodologies have been reviewed to determine which concepts, components, techniques, or tools already exist that can be utilized by software first.

Methodology Studies for Real-Time Ada Software, U.S. Army Communications & Electronics Command (CECOM), IITRI Project A06226

This effort was performed for the Army Communications and Electronics Command (CECOM) to develop a methodology for software development. This effort detailed the requirements of a software development using the Ada programming language for real-time software applications.

The study provided a set of criteria for evaluating any software development methodology proposed for real-time systems. These criteria were organized on the basis of the software development life-cycle; requirements analysis, preliminary design, detailed design, code and unit test, and integration and test.

SECOMO Training, U.S. Army Communications & Electronics Command (CECOM), IITRI Project A06256

This effort was performed for the U.S. Army Communications & Electronic Command (CECOM) Center for Software Engineering. The purpose of this effort was to increase understanding of cost estimation technology by presenting a three-day training session on the COCOMO and SECOMO cost estimation models to CECOM personnel.

The training session was held at Fort Monmouth, New Jersey. The course consisted of individual sections on SECOMO and COCOMO. Following the course presentation, DACS personnel remained on site for two days to answer any question and provide consultation on the models.

Support for the 3rd Annual RADC Knowledge Based Software Assistant (KBSA) Conference, U.S. Air Force, Rome Air Development Center (RADC/COES), IITRI Project A06258

The scope of this effort was to provide support and technical advice for all the activities involved in organizing and presenting the 3rd Annual RADC

Knowledge Based Software Assistant (KBSA) Conference. The purpose of the conference is to facilitate the transfer of KBSA technology and information. Over 75 participants attended the conference held 21-24 August 1988 in Utica, New York.

Reduced Instruction Set Computer (RISC) Evaluation, U.S. Air Force, Rome Air Development Center (RADC), IITRI Project A06262

This effort was performed for RADC to determine the effort required to develop an Ada Environment to be used on RADC's Reduced Instruction Set Computers (RISC) Program. The results of this effort included a briefing and a final report, "VAX/MIPS Ada Compiler Systems," detailing the results of the investigation.

Support for the 2nd Annual RADC Artificial Intelligence (AI) Technology Fair, U.S. Air Force, Rome Air Development Center (RADC/COES), IITRI Project A06271

The DACS provided support and technical advice for all the activities involved in organizing and presenting the 2nd RADC Artificial Intelligence Technology Fair. The purpose of the fair is to facilitate the transfer of AI technology and information within the RADC contractor community. The 2nd RADC Artificial Intelligence Technology Fair was held 15-16 November 1988 in Utica, New York.

<u>Ada Compiler and Software Engineering Tool Evaluation</u>, U.S. Army Communications & Electronics Command (CECOM), IITRI Project A06287

This effort was performed for the U.S. Army Communications and Electronics Command (CECOM) Advanced Software Technology (AST) Division. The scope of this effort involved evaluating software tools. The first objective of this effort was to provide the AST on-site support in the evaluation of Ada compilers at the AST Laboratory. The second objective was to evaluate various software engineering tools for application in CECOM's Center for Software Engineering (CSE).

<u>Software Size/Cost Models Seminar Update</u>, U.S. Air Force Cost Center (AFCCE), IITRI Project A06292

The purpose of this effort was to continue development and provide updated information on software size and cost models. The initial development of a presentation was funded by the AFCCE as part of a previous DACS special study. The initial presentation was provided to over 100 participants in 1987. Under this effort, this presentation was modified to cover changes in software cost and software size estimation models. New and updated information was included to make the presentation current.

<u>Core Set of Assembly Language Instruction (CALSI) Update</u>, U.S. Air Force, Rome Air Development Center (RADC/COEE), IITRI Project A06298

The purpose of this effort was to continue supporting RADC's Reduced Instruction Set Computer (RISC) program. This was accomplished through the update of a draft document, "Core Set of Assembly Language Instruction (CALSI) for MIPS-based Microprocessors."

Management, Review, and Analysis of Army Software Issues, U.S. Army Materiel Command (AMC), IITRI Project A06301

This task focused on providing the necessary technical expertise and advice in addressing the issues faced by the U.S. Army Software Task Force, located at Fort Belvoir. The activities which this effort directly supported included the review of materials from existing software issue studies, analysis of this and any additional information, and developing a comprehensive report.

11.0 OBSERVATIONS AND RECOMMENDATIONS

During this period the DACS continued to operate as a full-scale IAC. Substantial improvement was made in the automation of the User Profile Database (UPD) for order tracking, statistics production, and mailing use. A number of the SLED holdings were also converted to an on-site on-line database.

11.1 Observations

The following observations can be made about the DACS and the associated activities required to operate the DACS:

- During the period the lack of easily available data to augment the SLED has become evident

- The need for a centralized source of information concerning software issues such as quality, measurement, and engineering practices has increased, however the requests for such information from the DACS has maintained a constant level

- The capabilities utilizing an on-line database system have accomplished substantial improvements. The automation of many activities has improved the capabilities of the DACS. The cost of these improvements has been incurred and the payback is beginning to be realized

11.2 Recommendations

The following recommendations are made in light of the observations and operation of the DACS:

- Continue operating the DACS, but focus its goals in accordance with the level of funding available for core activities

- Actively develop mechanisms to acquire data and actively seek new data sources

- Invest in core activities that yield salable products with a large market to promote the potential for self-sufficiency of operation or continued expansion funded by user receipts

REFERENCES

[BASI79]   Basili, Victor R.; Zelkowitz, Marvin V.; McGarry, Frank E.;
             Reiter, Robert W.; Truszkowski, Walter F.; Weiss, David L.   The
             Software Engineering Laboratory.   TR-535, SEL-1; Greenbelt, MD:
             NASA Goddard Space Flight Center; May 1977.

[CARO82]   Caron, Gary, et al., Establishment of the Data and Analysis
             Center for Software (DACS), Rome Air Development Center, RADC-TR-
             81-385, January 1982, 186 p.

[DEFE79]   Defeo, Pio V., Software V&V Tools - An Assessment Program.
             Proceedings from the Fourth Summer Software Engineering Workshop,
             NASA Goddard Space Flight Center, Greenbelt, MD, November 1989.

[DUVA76]   Duvall, Lorraine M., Software Data Repository Study, Rome Air
             Development Center, RADC-TR-76-387, Griffiss Air Force Base, NY,
             December 1976.

[DUVA79A]   Duvall, Lorraine M., Baseline Software Data System. System
             Description, Vol. 1.   RADC-TR-79-185.   Rome Air Development
             Center: Griffiss Air Force Base; July 1979.

[DUVA79B]   Duvall, Lorraine M., Curtis, Christine.   Baseline Software Data
             System.   Database Reference Manual, Vol. II.   RADC-TR-79-185.
             Rome Air Development Center: Griffiss Air Force Base; July 1979.

[ELOV79]   Elovits, Honey S., An Experiment in Software Engineering: The
             Architecture Research Facility as a Case Study. Proceedings from
             the 4th international conference on Software Engineering;
             September 1979.

[GLOS83]   Gloss-Soler, S.A., Data and Analysis Center for Software: An IAC
             in Transition, Rome Air Development Center, RADC-TR-83-132, June
             1983, 106 p.

[HENN86]   Henniger, Karen., Data and Analysis Center for Software: Final
             Technical Report.   Rome Air Development Center RADC-TR-86-178,
             November 1986.

[IITR82]   IIT Research Institute.   Analysis of PAVE/PAWS O&M Data.   Final
             Technical Report.   IIT Research Institute, March 1982: 43 p.

[IIT86]   DACS Data Analysis Plan.   IIT Research Institute; August 1986.

[MUSA79]   Musa, John D., Validity of the Execution Time Theory of Software
             Reliability.   IEEE Transactions on Reliability.   August 1979.

[NELS78]   Nelson, Richard.   Software Data Collection and Analysis.   Rome
             Air Development Center: Griffiss Air Force Base; September 1978.

[RADA81]   Radatz, Jane W., Analysis of IV&V Data. RADC-TR-81-145.   Rome Air
             Development Center: Griffiss AFB; June 1981: 140 p.

[ROME83]   Romeu, Jorge L., Data Analysis Program Plan for the Data and
             Analysis Center for Software.   Data & Analysis Center for
             Software; January 1983: 26 p.

[TURN81]   Turner, Christopher, Data Analysis Plan for the Data & Analysis
             Center for Software.   IIT Research Institute; December 1981.

[TURN82]   Turner, Christopher, The DACS Data Compendium.   Data & Analysis
             Center for Software; 1982.

[VIEN87A,   Vienneau, Robert.   Software Reliability Baselines.   IIT Research
             Institute; June 1987.

[VIEN87B,   Vienneau, Robert.   Software Quality Measurement Using the
             Automated Measurement System and Operating Data.   IIT Research
             Institute; March 1989.

[VIEN88]   Vienneau, Robert. An Empirical Validation of the Goel-Okumoto
            Software Reliability Model. IIT Research Institute; April 1988.

[VIEN89]   Vienneau, Robert. A Simulator for Investigating the Robustness
            of Software Reliability Models to Assumption Violations.   IIT
            Research Institute; January 1989.