TWO RELATIONAL DBMS: A COMPARISON

by

GARY F. GARTEN

B.S., Kansas State University, 1978

B.A., Kansas State University, 1979

B.S., Kansas State University, 1979

AN ABSTRACT OF A MASTER'S DISSERTATION

submitted in partial fulfillment of
of the requirements for the degree of

MASTER OF SCIENCE, COMPUTER SCIENCE

College of Arts and Science

KANSAS STATE UNIVERSITY

Manhattan, Kansas

1987

Acknowledgments

A great many people have contributed in the production of this paper. Unfortunately, only a very few of those will be here.

My wife Becky. She finished her Master's Report from Kansas State University a few months earlier and knows the work involved in researching and writing one.

My parents Fred and Roberta. They saw me through several years of college and the completion of three undergraduate and one graduate degree at Kansas State University. Thank you for all that you did for me.

My major professor, Dr. Elizabeth Unger, Kansas State University Department of Computing and Information Sciences.

The kind and very helpful people at ADR and IBM who provided technical help with the many questions and issues I raised.

Brent Ahsmuhs, a fellow K-State alumni, and his DP Department manager at Forrest T. Jones, KCMO for all of the help with the DB2 miniature system.

And my cat Genny, who sat beside the PC for hours at a time while I worked tirelessly at the word processor typing the text of the paper.

Thanks to all of you. This paper could not have been written without you!

Table of Contents

Table of Contents

Table of Figures

Table of Figures

Chapter 1

Introduction

Data Base Management Systems (DBMS) are playing an increasingly important role in the development of computer systems. There are a great many commercial data base management systems available to chose from, many more in the last 10 years than ever before. Of the 3 major types of data bases (relational, hierarchical and network), this paper will concentrate on two commercially available relational data base management system products for mainframe computers, IBM's DB2 (Data Base 2) and ADR's Datacom/DB.

But first, a little information on relational database management. "Ever since the landmark work of E.F. Codd and C.J. Date in the early 1970s, the concept of relational database technology has engendered considerable excitement, as well as confusion, in the computer industry." [60]

The excitement revolves around the promise of relational technology: "the ability to efficiently access and manipulate data, regardless of the manner in which the data is physically stored." [60] No programmer navigation is required to move through a relational database.

"The manipulative part of the relational model consist of a set of operators known collectively as the relational algebra, together with a relational assignment operator which assigns the value of some arbitrary impression of the algebra to another relation. Each operator of the relational algebra takes either one or two relations as its operand(s) and produces a new relation as its result. Codd originally defined eight such operators, ... union, intersection, difference, Cartesian product, ... select, project, join and divide...". [18]

"According to C. J. Date, the term 'fully relational' refers to two principal components of the relational data base model: the relational data structure, and the relational algebra.... A data base system may be called fully relational if its supports: Relational data

bases (including the concepts of domain and key and the two integrity rules, data integrity and referential integrity); and a language that is at least as powerful as the relational algebra (and that would remain so, even if all facilities for loops and recursion were to be deleted)." [60]

The most fundamental property of a relational data base system is that data is presented to the user as tables - the mathematical name for a table with unique rows is a relation - and that the system provides suitable operators for the manipulation of these tables.

An example of a relational data structure is provided in Figure 1.0.

LASTNAME	WORKDEPT	PHONENO
Smith	E11	2095
Spenser	E21	0972
Geyer	E01	6789
Perez	D21	9001
Haas	A00	3978
Johnson	D21	8953
Nicholls	C01	1793
Thompson	B01	3476
Lutz	D11	0672
Pulaski	D21	7831
Setright	E11	3332
Stern	D11	6423

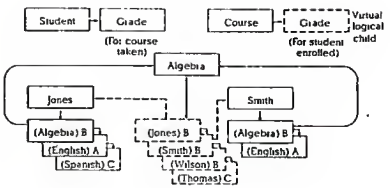
Figure 1.0 Telephone Directory

Figure 1.0 shows only a single table. The table is a telephone directory containing name, work department and telephone number. There is one row for each employee. Each row has three values - one for each column in the table.

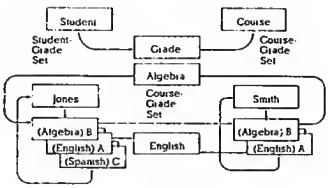
A relational data base is usually composed of many different tables, as shown in Figure 1.1 (the terms relation and table are synonymous). Thus, a relational data base appears as a collection of tables, each consisting of columns and rows. The rows in a table correspond to records in a file and the columns correspond to the fields within such records. "In order for the analogy with a file to be meaningful such a file should have records of one single type only, all occurrences having the same record structure and layout." [60]

The alternatives to presenting data as tables are to present data in the form of hierarchies (as DL/1) or in the form of networks. "A main difference between these alternatives is that in the case of network or hierarchical data structures (see Figure 1.1), the relationships between data are to some extent represented by the struc-

Hierarchical DBMS Structure



Network DBMS Structure



Relational DBMS Structure

Course Relation

Course Number	Course Name
10003	History
10150	Spanish
10152	English
10288	Algebra

Student Relation

Student Number	Student Name
1016	Ruth Davis
1099	Sam Smith
1861	Jan Wilson
1902	Bill Jones
2006	Ceri Thomas

Courses-Taken Relation

Student Name	Course Name	Grade
Ruth Davis	Spanish	A
Bill Jones	Algebra	B
Bill Jones	English	A
Bill Jones	Spanish	C
Sam Smith	Algebra	B
Sam Smith	English	A
Ceri Thomas	English	C
Ceri Thomas	Algebra	C
Jan Wilson	History	A
Jan Wilson	Algebra	B

Figure 1.1 Examples of the Three Models

ture itself - in a network through Owner-Member links, in a hierarchy through Parent-Child links. In relational data structures all relationships between tables are represented purely by the values in the tables.

"In a system logically modeled as a hierarchical or network structure, all potential access paths are explicitly defined by the data base administrator. Any application program that accessed the data base must work through a predefined and preauthorized access path. These paths are a major strength of the hierarchical and network data base." [60]

A relational data base management system is the software package that will provide facilities to access and update tables. As such it includes a language to define, access and update the databases, storage methods to maintain data on disk, utility functions, concurrency control facilities and various service functions.

The relational database is being held up by many as a panacea for the ills currently afflicting modern DP operations: the application backlog, the need to increase

increase development productivity, the complexity and changeability of business requirements and the demands of end users for access to their data.

Database entity terminology can vary widely from vendor to vendor. Appendix C shows the accepted formal relational term and counterparts for each used by the two DBMS vendors under study (ADR Datacom/DB and IBM DB2). For example, the formal term relation is referred to as a table by DB2 and as a file by ADR. [19]

Now, on to Chapter 2 and some detail about the two relational DBMS product families under study. Chapter 3 makes further comparisons through a miniature application system implemented in each environment. Chapter 4 concludes the paper with impressions of the two product families and future work which could be done to supplement the effort of this paper.

Chapter 2

Orientation to the DB2 and Datacom/DB product families

The purpose of this Chapter is to provide the reader with more detailed information about both of the relational DBMS product families under study - IBM's DB2 and ADR's Datacom/DB.

"In the late 1960s and early 1970s, Codd introduced the relational data model as an alternative way of structuring and managing data. Here, data is structured in two-dimensional tables and related by their value only, not by the logical structure of the schema. The term nonnavigational data structure was created because no programmer navigation is required to move through the data structure. In conjunction with the data structure, the relational model suggests data manipulation via a series of set-theoretic operators that help achieve significant economies in programming and end user access to data bases." [44]

IBM's DB2 is the product of the research into the relational model that began at IBM in 1969. DB2 (Data Base 2) is IBM's relational DBMS for large scale (MVS) mainframe data base systems. "Announced in June 1983, generally available in April 1985 after an intensive "field test" program, and recently enhanced (release 2, Feb. 1986) with subtle but important performance improvements. DB2 is a new product, very new. As of fourth quarter 1986 it had approximately 450 users." [16]

"ADR's Datacom/DB was first introduced in 1974. It had approximately 1200 customers as of fourth quarter 1986." [15]

Figure 2.0 provides a quick comparison of the two DBMS product families:

	DB2	Datacom/DB
Vendor	IBM	ADR
Structure	Relational	Quasi-relational*
Hardware	IBM 370 family	IBM 370 family, PC-AT, 3270 PC
Operating System	MVS, MVS/XA	DOS/VSE, VS1, VM, MVS, MVS/XA
High-level non-procedural Langs.	SQL	Ideal
Min. Memory Req.	2.5 - 5 MB	1 - 2.5 MB
Data Dictionary	IMS DB/DC	Data Dictionary
Logical View	Extensive	NO
Data Independ.	Yes	Yes
Query	QMF, SQL, QBE	Data Query
Appl. Pgmg Aid	Cross System Product	Ideal
Graphics support	No	Data Query Graphics
Record/file lock.	Yes	Yes
Automatic recovery	Yes	Yes
Authorization/ security	extensive	extensive

Figure 2.0 Quick Comparison of Features

Networking Facility	SNA	SNA, DNET
Transaction Proc. Facility	CICS, TSO	CICS, Datacom-DC, Roscoe

* Quasi-relational means that many relational algebra operations are provided, but not enough for the relational algebra to be complete.

Figure 2.0 Quick Comparison of Features

Below is a brief description of several products from each DBMS family. The description will be organized in the following general categories:

ADR product descriptions:

ADR Data Base/Data Communications

1. Datacom/DB

ADR/Datacom/DB is a relational database management system designed for on-line and batch applications first installed in 1974. It services concurrent record-at-a-time and set-at-a-time data manipulation requests with comprehensive security and integrity protection, utilizing a hi-speed directory of active definitions, a unique, compound relational index that directs the location of data, and an intelligent access plan optimizer. It is available for both OS and DOS environments.

2. Data Designer

Data Designer is an interactive data base design tool that automates and simplifies data base design.

3. Datacom/DC

Datacom/DC is a reentrant, multi-tasking, multi-threaded TP monitor which controls programs and terminals for on-line data base applications.

4. D-Net and /D-DDB

D-Net and D-DDB support distributed database processing and permit data to be shared transparently across multiple computer systems in a network. D-DDB manages data updating across nodes, and supports replicated and partitioned databases. D-Net provides the connection and transportation services.

ADR Resource Control:

5. Librarian

Librarian is a source program management system for programming, systems, and operations personnel and managers.

6. Datadictionary

Datadictionary is the central control and resource manager for the ADR/Datacom product line. It is integrated and active, managing an organization's entire information resource, ensuring consistency of all definitions and re-

lationships.

ADR On-line Program Development

7. Roscoe

Roscoe is an on-line program development and maintenance system for OS environments.

8. Vollie

Vollie is an on-line program development and maintenance system for DOS/VS(E) environments.

ADR Application Development

9. Ideal

Ideal is a sophisticated, fourth-generation application development system suitable for a broad breadth of on-line and batch applications. It combines a very high-level, structured language with a screen painter, a report writer and change management, prototyping, and testing facilities into a workstation environment.

10. DE-II

An on-line data collection and validation facility with

generalized and user-defined formatted screens.

ADR Advanced Programs and Languages

11. MetaCOBOL

MetaCOBOL is a Cobol language pre-compiler that supports Cobol standards, structured programming, and higher-level language extensions for DB/DC applications.

12. ADR/DL

ADR/DL supports the development and enhancement of Cobol applications through interactive programming facilities, a high-level, structured language consistent with Ideal, and a batch Cobol generator. ADR/DL operates with ADR/ROSCOE and ADR/Vollie, with plans to support IBM's TSO/ISPF. ADR/DL is a high-level extension to the Cobol language designed especially for data base and data communications applications.

ADR Query and Reporting

13. Datareporter

Datareporter is an information retrieval and reporting system for generating management information reports from centralized files.

14. Dataquery

Dataquery supports ad hoc on-line and batch inquiry and reporting. Field value security and user-level processing limits allow a wide audience of users to safely and directly reference the same set of production data for decision making. Other features include updating, graphics, multilingual support, expert and novice modes, and an on-line tutorial. Dataquery is an English-language, relational query system for providing immediate access to information.

ADR Office Automation, Decision Support, Documentation

15. Empire

Empire is a decision support system with integrated color graphics and statistics form business analysis and planning.

16. ETC

ETC is an interactive word processing system for automated preparation, composition, and maintenance of textual material.

17. eMAIL

eMAIL is an interactive electronic mail system for sending, receiving, storing, and managing correspondence quickly and easily.

18. ASC

ASC is a system to automatically produce documentation for system and application-level information.

19. Autoflow II

Autoflow II is an advanced system development tool which provides automatic program documentation and analysis.

ADR Operational Performance

20. LOOK

LOOK is a real-time performance measurement system for improving performance and throughput. Look/Datacom permits the comprehensive performance management in database-oriented shops. In addition to dynamically analyzing activity levels, resource utilization, and response times of Datacom/DB, Look also determines the load and utilization of the entire processor, including CICS, batch, ROSCOE, the operating system, and the hardware.

ADR Transparency

21. ADR's migration software automates the process of porting data from older technologies (VSAM, DL/1, Total) to a relational environment, without the need to convert, recompile, or relink existing applications. Numerous options are available in the new relational environment to productively harness the information, manage the operation, and introduce change, particularly when the applications run as fast or faster than before.

DB2 product descriptions:

DB2 Data Base/Data Communications

1. Database 2 (DB2)

A large-system MVS relational DBMS, first released in 1983. It can be installed with the IMS/VS/DB hierarchical system or configured as a stand-alone DBMS. The DB2 system employes the SQL (Structured Query Language) as its host data base language, and is compatible, to some degrees, with the SQL/DS relational system designed for use the DOS/VS environment.

DB2 Query and reporting

2. Query Management Facility (QMF)

QMF provides interactive data base facilities to users with little of no technical background. It operates against SQL/DS data under VM as well as against DB2 under MVS. Data definition functions are provided through SQL. Features include ad hoc query in SQL or QBE (query by example) languages. Report preparation consists of relation of data for graphic presentation, defining and executing a procedure consisting of a series of query/report functions.

DB2 Application Development

3. Cross System Product/Application Development (CSP/AD) and Cross System Product/Application Execution (CSP/AE)

With CSP/AD and CSP/AE, programming staff can completely define, test, generate, and execute application programs. These productivity tools support multiple development and production environments.

4. Data Base Edit Facility (DBEDIT)

DBEDIT helps you perform data base operations without requiring you to know SQL or data base structure. You can insert, delete, update, or select information from a data base. You can, in short, perform all the data manipulation tasks for which you would otherwise need to know SQL.

DB2 Data Migration, Product Interfacing

5. CSP-DB2 Interface

DB2 offers the powerful data manipulation capabilities of SQL. The Cross System Product offers the ease-of-use features of high-level languages. The CSP-DB2 Interface gives you both. The CSP-DB2 Interface is a PL/1 program

that allows Cross System Product applications to issue SQL statements against DB2 tables.

6. Data Base Migration Aid Utility

Assists the DB2 system of DBA in migrating data and data descriptions between DB2 systems. Data descriptors can also be redefined on an existing DB2 subsystem. The program is a set of nine functions and requires authorization to select data from the DB2 system catalog tables. The Utility is menu driven.

7. Data Extract (DXT)

Extracts data on a periodic or one time basis. The operational data can be in a DL/1 hierarchical database, a VSAM file, a physical sequential file, or a DB2 or SQL/DS-VM relational database. This extracted data can be put into a relational database for easy access and reporting by such programs as Query Management Facility (QMF) and can be moved from different subsystems on the same processor, moved between subsystems on different processors, or stored elsewhere, as defined by the installation's needs. End user dialogs are similar to

those of QMF, allowing users who have no data processing experience to create extract requests.

8. Data Dictionary DB2 Interface

Expands the Dictionary's (DB/DC) base of general functions to support DB2. It is designed to accelerate DB/DC system and application development by supporting these activities by using Dictionary information as the source for creating, modifying, and/or deleting DB2 objects and by using the DB2 catalog as the source for creating and/or modifying Dictionary subjects. A front end dialog under ISPF is optional.

DB2 Resource Control

9. DB/DC Dictionary

The DB/DC dictionary is an IMS productivity aid that consists of five DL/1 physical and five DL/1 logical data bases containing information about the structure of data in five distinct levels. The system performs the following functions: translates dictionary user-input language requests; executes update commands to modify, delete, and add information to the dictionary databases; prepares output reports in either batch or on-line environment;

provides data definitions for Assembler, Cobol, or PL/1 COPY or %INCLUDE libraries.

DB2 Operational Performance

10. DB2 Performance Monitor (DB2PM)

Designed to provide DP managers, system administrators, database administrators, and system programmers with essential information addressing the performance if DB2 data base systems. With DB@PM you can obtain reports providing both system-wide and application-related information showing in varying levels of detail, DB2 performance characteristics during a given interval.

Here is a brief list of products in each DBMS family and prices as of July, 1986. Figure 2.1 shows pricing information for a typical ADR package with the following configuration:

Datacom/DB Kernel
Datadictionary
Datareporter
DataQuery
Ideal

Package price:	Initial charge:	$288,300
	Annual charge:	$ 42,546

Figure 2.1 ADR Package Price, Typical Configuration

Figure 2.2 shows prices for selected ADR products.

		License Purchase ($)	Annual Maint. ($)
Datacom/DB Kernel	DB	116,000	16,240
Datadictionary	DD	39,600	5,545
Datareporter	DR		2,184
Options:			
Dataquery	DQ	34,800	4,872
Ideal (4GL)	ID	97,900	13,705
VSAM Transparency	VT	25,300	3,542
DL/1 Transparency	D1	40,000	5,600
Datasecure	DS	14,100	1,974
Datadesigner	DG	50,600	7,084
D-Net	DN	41,300	5,780
DE-II (data entry)	DE	24,500	3,430

Figure 2.2 ADR Price List, Selected Products

Figure 2.3 shows prices for a typical DB2 package with the following configuration:

SQL (Data Manipulation/definition lang.)
Query Management Facility (QMF)
DB2 Performance Reporting Tool
Data Base Edit
Data Dictionary DB2 Interface

Package price:	Initial charge:	$16,050
	Monthly charge:	$2,675
	(Annual charge:	$32,100)

Figure 2.3 DB2 Package Price, Typical Configuration

Figure 2.4 shows prices for selected DB2 products.

	1 Time Charge ($)	Initial Charge ($)	Monthly Charge ($)
1. DB2		16,050.00	975.00
2. QMF			1500.00
3. DXT			300.00
4. CSP			1550.00
5. CSP-DB2 Intf	4000.00		1150.00
6. Edit Facility			350.00
7. Mig Aid Util	4000.00		
8. DD DB2 Intf	4000.00		
9. Perf Monitor			975.00

Figure 2.4 DB2 Price List, Selected Products

Figure 2.5 shows abbreviations for DB2 products which were used in Figure 2.4.

DB2	- Database 2
QMF	- Query Management
DXT	- Data Extract
CSP	- Cross System Product
CSP-DB2 Intf	- Cross System Product DB2 Interface
Edit Facility	- DB2 Edit Facility
Mig Aid Util	- Migration Aid Utility
DD DB2 Intf	- Data Dictionary DB2 Interface
Perf Monitor	- Performance Monitor

Figure 2.5 DB2 Product Abbreviations

There are many features about the databases and other related products which are different. Below is a more detailed discussion about each DBMS and selected comparisons. But first, a brief overview of each DBMS system.

DB2 product system overview:

DB2 is a large-system MVS relational DBMS. It was first introduced in 1983 as part of IBM's dual data base strategy. DB2 was to be used as an ad hoc query and data analysis tool, while IMS was to serve as the production DBMS. This plan has been changing as DB2's performance has continually improved.

DB2 can be installed with the IMS/VS/DB hierarchical system or configured as a stand-alone DBMS. The DB2 system employs SQL (Structured Query Language) as its host data base language, and is compatible, to some degrees, with the SQL/DS relational system designed for use with the DOS/VS environment. DB2 provides the following integral database facilities: relational file

structure, views, table space, SQL, data space management, user interface, monitoring and accounting, security and authorization, and data set protection. All data in a DB2 data base is stored in VSAM entry sequenced data sets (EDSD), which can be defined and maintained by the user or automatically maintained by DB2. DB2 supports a relational data model. DB2's data base can be described as a collection of tables. Data is defined in terms of tables and accessed through operations on tables. Data definition, retrieval, manipulation, and control operations are supported by SQL. SQL is a high-level data language available to users through an interactive terminal and through applications written in Cobol, Fortran, PL/1, Basic, or Assembler language. DB2 can be accessed concurrently by the IMS/VS Data Communication Feature, by CICS/OS/VS, by TSO users, and by batch jobs. It operates as an MVS subsystem and is designed to utilize the System/370 Extended Architecture (XA), including 31-bit virtual addressing and large real storage. DB2's architecture provides for very large data bases (up to 64 billion bytes per table). It is supported by a comprehensive set of data base utilities that operate online, including DXT and DB2 Performance Monitor. Its security

and authorization mechanism offers field content security and allows various levels of authority to be delegated to users as appropriate.

ADR Datacom/DB product system overview:

"ADR/Datacom/DB was originally marketed by a small firm that did not have the marketing resources of a major system software house. ADR acquired the company and the DBMS in late 1978. Soon thereafter, ADR spent considerable effort in redesigning and restructuring the architecture of the system, and the growth in the user base since then has been dramatic. In 1985 alone, the product doubled its user base, and should continue to make significant inroads in the market with its flexibility, ease of use, integration with other popular ADR products, and its recently introduced DL/1 Transparency option." [15]

Datacom/DB is a relational database management system designed for on-line and batch applications. Data is stored in tables (also known as logical relations) and accessed as rows and columns. Each database has a single, combined index providing logical relational operations such as select, project, and join. The system provides an interactive end-user language to support direct user access, updating, analysis and reporting.

Datacom/DB programs use dataviews or logical views of data when accessing information. It provides program development, maintenance and information center support for existing and future applications through ADR/Ideal, a fourth generation application development system for new applications and ADR/DL, a comprehensive Cobol application generator for database programs and a complete data manipulation language. Data access is supported by random, sequential or index-only processing. Data integrity is maintained through such features as concurrent update protection, exclusive control of data, automatic transaction backout, complete automatic restart/recovery, program data access security and encryption facilities. Datacom/DB supports a multi-user facility that provides the user with extended ability to access databases concurrently from multiple partitions. Accounting facilities for management control are definable by each installation and its preferred detail level of summarization point. The information is stored in table form, in a system controlled Datacom/DB database. Statistics can be accessed on-line or in batch with the ADR tool set. The Datacom/DB databases are defined through Datadictionary, a central resource manager for the infor-

mation environment. The dictionary provides facilities for definition, design, reporting, auditing and control. ADR's migration software tools, the VSAM or DL/1 Transparencies, migrate current data written for VSAM or DL/1 to Datacom/DB with no modifications to existing application programs.

Below is more detailed information about selected areas for each DBMS:

1. Operating environment

DB2: Minimum memory 2.5 - 5 MB

ADR: Minimum memory 1 - 2.5 MB

Computer/Operating Systems supported:

ADR: IBM system/370, 3000, 4000 and compatible computers. SSX/VSE, DOS/VS, DOS,VSE, OS/VS1, OS/VS2(SVS), OS/VS2(MVS), MVS/XA, AND VM/CMS.

DB2: Any IBM or compatible processor supported by MVS/SP, MVS/XA.

ADR: can be accessed by Datacom/DC (ADR's reentrant, multi-tasking, multi-threaded TP monitor, by CICS and by other telecommunications monitors.

DB2: can be accessed concurrently by IMS/VS Data Communication Feature, by CICS/OS/VS, by TSO users, and by batch jobs.

DB2 was designed to exploit the 31-bit addressing architecture of MVS/XA, but it can also run on MVS/370, which uses 24-bit addressing.

2. File structure

All data in a DB2 data base is stored in VSAM entry sequenced data sets (EDSD), which can be defined and maintained by the user or automatically maintained by DB2.

3. Views of data

The general concept of a DB2 and ADR view of data are the same, a view allows you to present a smaller, simpler

version of a table upon which it is based. It permits you to authorize the use of a view to only specified users, and not every field in the entire table. A view only needs to contain fields to meet the needs of a user/program and not contain any unnecessary fields to complicate or confuse things.

However, a DB2 view can be created dynamically (does not need to previously exist) by a program if the program is authorized to access all fields making up the new view. An ADR dataview cannot be dynamically produced. A DB2 view also very importantly can contain fields from more than one table (i.e. join 2 or more tables), from other views or from a combination of views and tables. Views are used just like tables in SQL data manipulation statements. An ADR dataview currently can only contain fields from one table, not fields from multiple tables. A dataview processor has been promised for a future release which will remedy this situation.

4. Table space/Data space management

DB2 Table Space is user-defined to hold the data base

tables. Each table space is divided into equal-sized units, called pages, which contain table data. A table space consists of one to 64 VSAM entry sequenced data sets (ESDS) and contain up to 64G bytes of information.

Table space is of two types, simple and partitioned. A partitioned table space holds exactly one table. The table space is divided into partitions, and each partition is stored in one VSAM ESDS. The partitions are defined as ranges of an index based on one or multiple columns. Through partitioned table spaces very large tables can be split into manageable units; partitions are independent of each other and can be reorganized and recovered individually.

In addition, partitioned table spaces can be assigned to different storage groups, each of which may be assigned to a different device type. This assignment scheme allows installations to store active data on faster devices than those on which history data is stored. Table spaces which are not partitioned are referred to as "simple" table spaces. A simple table space can hold one or more tables.

Table spaces are physically divided into storage units called pages. Each page holds one or more rows of a table (or multiple tables for simple table spaces). Two page sizes are available, 4K and 32K bytes. The larger page size must be chosen if rows with a length of more than 4K bytes exist in a table.

DB2 uses VSAM for DASD space management and data set cataloging. However, after the data sets are created, they are formatted and used by DB2 and cannot be processed by VSAM services.

At the top of ADR entity model are two related entities, Database and Area. These entities describe the data base and area into which files (tables) are organized. The Area, hierarchically, is the ADR counterpart to the DB2 Table Space. An Area can have one or more Datacom/DB files defined to it. Typically, only one file is related to one area. If more than one file was related to an Area, the data would be co-located, or co-mingled. If two files are nearly always accessed together, the number of I/Os could be cut down by physically locating the data together. These two files should probably be physically

together in the same Area. The records in the Area would resemble:

Record A/Record B/Record A/Record B/Record A/...

ADR Space management option information for an Area's files is also defined at the Area level within Datadictionary when the Area is defined. Options 0-3 can be selected to aid in the management of space within the files. Option selection controls things such as the reclamation of space after a record has been marked for deletion, whether it is important to maintain the native sequence in which records were added to the file or not, whether it is permissible or not to wrap-around to the beginning of the file when an end of file is detected.

ADR's CXX (control) File contains data base control information. It consists of the base segment describing data base characteristics, followed by area and file segments that describe the characteristics of each file, including keys and data elements. The CXX is constructed automatically from definitions in Datadictionary.

5. Data manipulation language

Users of both DB2 and ADR products do not have to navigate their way to data. When the requested data is located, DB2 and ADR returns, updates or deletes the entire collection (termed set-at-a-time processing) of data that met the conditions specified by the user.

SQL, Structured Query Language is used in DB2 for data manipulation. The language includes statements for retrieval, replacement, insertion, and deletion of data.

SQL statements may be issued interactively from a terminal and the results may be browsed. The interactive SQL facility is available to authorized TSO SPF/ISPF users.

SQL statements may be embedded in application programs written in Cobol, PL/1, Fortran, Basic, or Assembler.

SQL also provides Data Definition facilities for creating, changing, and deleting all DB2 objects. Objects are storage groups, table spaces, tables,

indexes, and views. A unique advantage of SQL is that these Data Definition statements may be used in a normal user session and mixed together with other types of statements, such as data manipulation statements. It is not necessary to stop DB2 or to invoke special utility programs to create a table for storing and manipulating some temporary result and drop the table when it is no longer needed.

ADR does not have a direct counterpart to DB2's SQL. ADR does not have one set method of manipulating data, but several products which provide for manipulation of data. First available was data base access by writing a traditional third generation language and embedding it with a low-level CALL interface (ie. CALL DBENTRY parameters). This is similar to DB2's SQL-embedded third language support. ADR's Ideal (standing for Interactive Development Environment for an Application's Life-cycle) is an alternative data manipulation language (4GL) plus an interactive tool for the design, development, and execution of applications in an online environment. Ideal lets the user access and update the data base with

an easy-to-use and very high-level set of statements.

Another ADR data manipulation product is ADR/DL. DL is an interactive, high-level application development and maintenance Cobol workstation. With DL, programmers can access and manipulate Datacom/DB data directly from Cobol applications without coding physical interfaces to the data base. DL has both Cobol-like commands and 4GL-like functions.

ADR announced in the first quarter of 1987 that it will support a subset of SQL data manipulation commands. No further information is available from ADR at this time on SQL support. ADR has recognized SQL as the recognized industry standard data manipulation language and the value of supporting that standard. ADR will introduce SQL support with a selected subset and is expected to expand over time the set of SQL commands it will support.

6. Indexes

DB2 and ADR indexes are very much alike. Both have a B-tree index structure. Each level of the index contains

a pointer to a block of pointers at the next lower level, except the last level, which contains pointers to actual data records. Programs or users accessing data never explicitly specify keys or indexes, indexes are only used by the DBMS. Programs specify field names. The DBMS maps the field or fields back to keys.

DB2 has one index for each key. ADR has 1 index for all keys of each database. Both have cluster keys or indexes. A cluster index or key determines the physical order in which rows are stored. A Cluster key or index is used to store data the way it would be most frequently accessed, for example, employee number, account number, or alphabetically by name. Both ADR and DB2 provide for unique and nonunique indexes/keys.

7. Program preparation

A DB2 application program issuing SQL calls must be precompiled using the DB2 precompiler to create a Data Base Request Module (DBRM). The DBRM is the input to the DB2 BIND process which produces a DB2 application plan that contains an optimized access path for each SQL statement. Besides the access path, the plan also

contains the tables to be accessed and the appropriate locking information. The application program must be compiled and then link-edited.

After programs are written, four steps must be performed before they can be run.

Precompilation: to check SQL syntax, produce a modified source program, and produce a data base request module (DBRM), an intermediate form of an SQL statement.

Compilation: to translate the modified source program using wither a Cobol, PL/1, or Fortran compiler or an assembler.

Bind: to process the DBRM to produce an application plan, the control structure representing one or more SQL statements.

Link-edit: to produce the final object module.

The compiling and link-editing steps are the same in DB2 as they are in any other programming process. The

precompile and bind, however, are unique to DB2.

ADR programs can be written in several different languages, ADR/DL (an extension to Cobol), Ideal (ADR's 4GL) or Cobol, PL/1 or Assembler with embedded DBENTRY calls. When the program is successfully compiled, the program preparation process is finished. Access paths to data are determined dynamically by CBS (Compound Boolean Selection), which is discussed below.

8. Path Selection to data.

When all SQL statements are found to be correct in a program, the binder is authorized to access the data, DB2 builds and stores an application plan that contains information about both the program and the data the program uses.

The major advantage of the precompilation and bind process is that it removes operations that can be done once from the program's normal processing. When the program runs, SQL statements are not translated or semantically checked. Access paths are not selected. All that processing happens only once, before the program

runs.

If the data definitions or indexes that an application plan uses should change, rebinding occurs automatically. For example, suppose an index on a table used in an application plan were dropped. If the program using the plan were run, DB2 would automatically invoke the bind process to rebuild an application plan that did not use the index that had been dropped.

ADR's CBS (Compound Boolean Selection) facility provides full relational selection and ordering capabilities. CBS is totally key insensitive from the programmer viewpoint; selection and ordering can be specified for any fields in the for any fields in the table with full data typing (including floating point). CBS dynamically chooses the most efficient index path to satisfy the access request. If a key is modified, typically no program changes need to be made, CBS will continue to dynamically select the optimal path to data.

Chapter 3

Mini-application system - DB2 vs Datacom/DB

3.0 Introduction

In order to further study how DB2 and ADR products compare, a miniature application system was constructed in both environments. This section details the work involved in taking an existing small application system running in a DB2 environment and duplicating it as closely as possible in an ADR environment.

The mini-system consisted of programs at both extremes: an interactive on-line program and a batch report program and the necessary database, files, data views and other needed entities. This mini-system is part of a sample application system provided by IBM as part of the DB2 software (see Appendix B). It was decided to use this system provided by IBM instead of a system written in-house at the DB2 site because the DB2 site's personnels' programming skills would not be a fac

tor and more importantly, the sensitivity of any of the site's data would be compromised in any way.

The purpose of the mini-application system was to aid the author in demonstrating future comparisons between DB2 and ADR DBMS software.

The batch program produces a report which is several pages of telephone directory listings. Each telephone list is based on card image input (see Figure 3.0).

A line number bar has been provided above the input for the reader's convenience.

```
0        1         2         3         4         5
12345678901234567890123456789012345678901234567890123456789012345678
L*

LJO%

L%SON

LSMITH

LBROWN           ALAN

LBROWN           DAVID

U                                  0002304265
```

Figure 3.0 Batch Program Card Image Input

'LBROWN ALAN', is the fifth card image in Figure 3.0. The 'L' in the first byte indicates the action - 'L' = list, a 'U' in the last card image means update. The above example input indicates for the program to produce a telephone directory listing of all records (using the EMPLOYEE and DEPARTMENT Tables) with a last name of BROWN and first name of ALAN. If no match was found, an appropriate message was produced on the report.

The mini-system's on-line program is used to maintain the database (ie. add, change, delete, and display an index). Figure 3.1 shows the Employee Index screen.

```
                          SELECTING AN EMPLOYEE TO DISPLAY
MAJOR SYSTEM .....: O             ORGANIZATION
ACTION ...........: D             DISPLAY (SHOW)
OBJECT ...........: EM            EMEPLOYEE
SEARCH CRITERIA ..: EN            EMPLOYEE NAME
DATA .............: %

NO   D/ID   DEPARTMENT NAME                    E/ID     EMPLOYEE NAME

01   A00    SPIFFY COMPUTER SERVICE DIV.      000010   CI HASS
02   B01    PLANNING                          000020   ML THOMPSON
03   C01    INFORMATION CENTER                000030   SA KWAN
04   E01    SUPPORT SERVICES                  000050   JB GEYER
05   D11    MANUFACTURING                     000060   IF STERN
06   D21    ADMINISTRATION SYSTEMS            000070   ED PULASKI
07   E11    OPERATIONS                        000090   EW HENDERSON
08   E21    SOFTWARE SUPPORT                  000100   TQ SPENSER
09   A00    SPIFFY COMPUTER SERVICE DIV.      000110   VB LUCCHESI
10   A00    SPIFFY COMPUTER SERVICE DIV.      000120   S  O'CONNELL
11   C01    INFORMATION CENTER                000130   DH QUINTANA

PFK: 02=RESEND  03=END  08=NEXT
```

Figure 3.1 On-line Program Employee Index Screen

The best way to get a comparison of DB2 and ADR DBMS products would have been to have both DB2 and Datacom/DB running at the same site, physically on the same equipment. Since it was not possible to find one data processing shop with both ADR's Datacom/DB and IBM's DB2, the product comparison was conducted at the two sites. Care was given to ensure that the hardware and operating systems at the two sites were as similar as possible. Figure 3.2 compares the pertinent hardware and software at the two sites. Note that both sites have the same CPU and operating system.

The ADR Datacom/DB site selected was Johnson County, Kansas in Olathe, Kansas. The IBM DB2 site selected was Forrest T. Jones, a Kansas City, Missouri insurance company.

	Johnson County (ADR)	Forrest T. Jones (DB2)
Hardware:		
CPU:	IBM 4381	IBM 4381
Memory size:	16 MEG	16 MEG
Software:		
Operating system:	OS MVS/XA	OS MVS/XA
TP Monitor:	VTAM	SNA/VTAM
Pertinent IBM	CICS	CICS
Products:		IMS DL/1
		IMS DB/DC
		DB2
		DXT
		QMF
ADR Products:	Datacom/DB	Librarian
	Data Dictionary	
	Ideal	
	Data Query	
	Data Reporter	
	VSAM Transparency	
	Inter Products Components	
File storage DASD:	IBM 3380's	IBM 3380's

Figure 3.2 Profiles of Test Sites

Now, more about the mini-system. First, the database configuration documentation supplied by the DB2 site is given in Figure 3.3, which shows the relationship between storage group, database, table space, and table. Figure 3.4, shows the Department and Employee table layouts (each is enclosed in a box). Characteristics of fields in each table are shows. Figure 3.5, shows Indexes (keys) for the Department and Employee tables. Fields making up each key are shown. Figure 3.6 shows the Phone View (enclosed in a box), which was used in the DB2 programs. Later in this chapter, entities shown in these figures are discussed in detail.

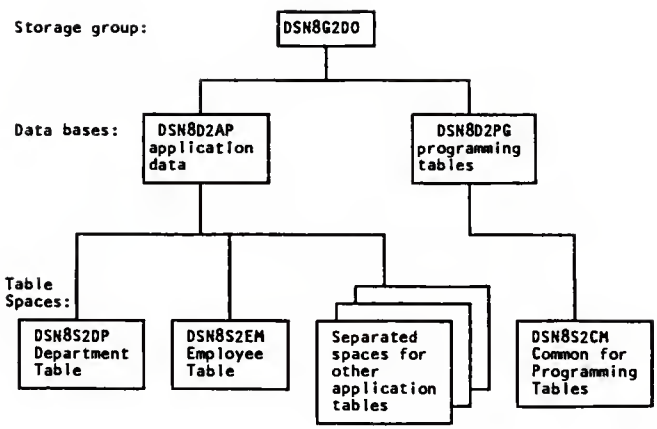


Figure 3.3 DB2 Test Data Base Configuration

CREATOR	TNAME	CNAME	COLNO	COLTYPE	LENGTH	SCALE	NULLS
DSN82	TACTYPE	ACTDESC	3	VARCHAR	20	0	N
DSN82	TACTYPE	ACTKWD	2	CHAR	6	0	N
DSN82	TACTYPE	ACTNO	1	SMALLINT	2	0	N
DSN82	TCONA	CONVID	1	CHAR	16	0	N
DSN82	TCONA	LASTMSG	5	LONGVAR	3574	0	N
DSN82	TCONA	LASTPOS	3	CHAR	254	0	N
DSN82	TCONA	LASTPOSC	4	CHAR	254	0	N
DSN82	TCONA	LASTSCR	2	CHAR	8	0	N
DSN82	TDEPT	ADMRDEPT	4	CHAR	3	0	N
DSN82	TDEPT	DEPTNAME	2	VARCHAR	36	0	N
DSN82	TDEPT	DEPTNO	1	CHAR	3	0	N
DSN82	TDEPT	MGRNO	3	CHAR	6	0	N
DSN82	TDSPTXT	DSPINDEX	1	CHAR	2	0	N
DSN82	TDSPTXT	DSPLINE	3	CHAR	79	0	N
DSN82	TDSPTXT	LINENO	2	CHAR	2	0	N
DSN82	TEMPL	BRTHDATE	11	DECIMAL	6	0	Y
DSN82	TEMPL	EDUCLVL	9	SMALLINT	2	0	Y
DSN82	TEMPL	EMPNO	1	CHAR	6	0	N
DSN82	TEMPL	FIRSTNME	2	VARCHAR	12	0	N
DSN82	TEMPL	HIREDATE	7	DECIMAL	6	0	Y
DSN82	TEMPL	JOBCODE	8	DECIMAL	3	0	Y
DSN82	TEMPL	LASTNAME	4	VARCHAR	15	0	N
DSN82	TEMPL	MIDINIT	3	CHAR	1	0	N
DSN82	TEMPL	PHONENO	6	CHAR	4	0	Y
DSN82	TEMPL	SALARY	12	DECIMAL	8	2	Y
DSN82	TEMPL	SEX	10	CHAR	1	0	Y
DSN82	TEMPL	WORKDEPT	5	CHAR	3	0	N
DSN82	TEMPRAC	ACTNO	3	SMALLINT	2	0	N
DSN82	TEMPRAC	EMENDATE	6	DECIMAL	6	0	Y
DSN82	TEMPRAC	EMPNO	1	CHAR	6	0	N
DSN82	TEMPRAC	EMPTIME	4	DECIMAL	5	2	Y
DSN82	TEMPRAC	EMSTDATE	5	DECIMAL	6	0	Y
DSN82	TEMPRAC	PROJNO	2	CHAR	6	0	N
DSN82	TOPTVAL	ACTION	2	CHAR	1	0	N
DSN82	TOPTVAL	DSPINDEX	11	CHAR	2	0	N
DSN82	TOPTVAL	HEADTXT	6	CHAR	50	0	N
DSN82	TOPTVAL	HELPTXT	9	CHAR	79	0	N
DSN82	TOPTVAL	INFOTXT	8	CHAR	79	0	N
DSN82	TOPTVAL	MAJSYS	1	CHAR	1	0	N
DSN82	TOPTVAL	OBJECT	3	CHAR	2	0	N
DSN82	TOPTVAL	PFKTXT	10	CHAR	79	0	N
DSN82	TOPTVAL	SCRTYPE	5	CHAR	1	0	N
DSN82	TOPTVAL	SELTXT	7	CHAR	50	0	N
DSN82	TOPTVAL	SRCHCRIT	4	CHAR	2	0	N
DSN82	TPROJ	DEPTNO	3	CHAR	3	0	N
DSN82	TPROJ	MAJPROJ	8	CHAR	6	0	N
DSN82	TPROJ	PRENDATE	7	DECIMAL	6	0	Y
DSN82	TPROJ	PROJNAME	2	VARCHAR	24	0	N
DSN82	TPROJ	PROJNO	1	CHAR	6	0	N
DSN82	TPROJ	PRSTAFF	5	DECIMAL	5	2	Y
DSN82	TPROJ	PRSTDATE	6	DECIMAL	6	0	Y
DSN82	TPROJ	RESPEMP	4	CHAR	6	0	N

Figure 3.4 DB2 Department and Employee Table Layouts

IXNAME	IXCREATOR	COLNAME	COLNO	COLSEQ	ORDERING	IBMREQD
XDEPT2	DSN82	ADMRDEPT	3	1	A	N
XDEPT3	DSN82	ADMRDEPT	4	1	A	N
XDEPT1	DSN82	DEPTNO	1	1	A	N
XEMPL1	DSN82	EMPNO	1	1	A	N
XEMPL2	DSN82	WORKDEPT	5	1	A	N
XCONA1	DSN82	CONVID	1	1	A	N
XOPTVAL1	DSN82	MAJSYS	1	1	A	N
XOPTVAL1	DSN82	SCRTYPE	5	5	A	N
XOPTVAL1	DSN82	SRCHCRIT	4	4	A	N
XOPTVAL1	DSN82	OBJECT	3	3	A	N
XOPTVAL1	DSN82	ACTION	2	2	A	N
XPROJ2	DSN82	RESPEMP	4	1	A	N
XPROJ1	DSN82	PROJNO	1	1	A	N
XACTYPE1	DSN82	ACTNO	1	1	A	N
XACTYPE2	DSN82	ACTKWD	2	1	A	N
XPROJAC1	DSN82	ACTNO	2	2	A	N
XPROJAC1	DSN82	ACSTDATE	4	3	A	N
XPROJAC1	DSN82	PROJNO	1	1	A	N
XEMPRAC1	DSN82	PROJNO	2	1	A	N
XEMPRAC1	DSN82	EMPNO	1	4	A	N
XEMPRAC1	DSN82	EMSTDATE	5	3	A	N
XEMPRAC1	DSN82	ACTNO	3	2	A	N
XEMPRAC2	DSN82	EMPNO	1	1	A	N
XDSPTXT1	DSN82	DSPTINDEX	1	1	A	N
XOSPTXT1	DSN82	LINENO	2	2	A	N

Figure 3.5 DB2 Department and Employee Indexes (Keys)

CREATOR	TNAME	CNAME	COLNO	COLTYPE	LENGTH	SCALE	NULLS
DSN82	VEMPLP	EMPLOYEENUMBER	1	CHAR	6	0	N
DSN82	VEMPLP	PHONENUMBER	2	CHAR	4	0	Y
DSN82	VEMPRAC	ACTNO	3	SMALLINT	2	0	N
DSN82	VEMPRAC	EMENDATE	6	DECIMAL	6	0	Y
DSN82	VEMPRAC	EMPNO	1	CHAR	6	0	N
DSN82	VEMPRAC	EMPTIME	4	DECIMAL	5	2	Y
DSN82	VEMPRAC	EMSTDATE	5	DECIMAL	6	0	Y
DSN82	VEMPRAC	PROJNO	2	CHAR	6	0	N
DSN82	VOPTVAL	ACTION	2	CHAR	1	0	N
DSN82	VOPTVAL	OSPINDEX	11	CHAR	2	0	N
DSN82	VOPTVAL	HEADTXT	6	CHAR	50	0	N
DSN82	VOPTVAL	HELPTXT	9	CHAR	79	0	N
DSN82	VOPTVAL	INFOTXT	8	CHAR	79	0	N
DSN82	VOPTVAL	MAJSYS	1	CHAR	1	0	N
DSN82	VOPTVAL	OBJECT	3	CHAR	2	0	N
DSN82	VOPTVAL	PFIXT	10	CHAR	79	0	N
DSN82	VOPTVAL	SCRTYPE	5	CHAR	1	0	N
DSN82	VOPTVAL	SELTXT	7	CHAR	50	0	N
DSN82	VOPTVAL	SRCHCRIT	4	CHAR	2	0	N
DSN82	VPHONE	DEPTNAME	7	VARCHAR	36	0	N
DSN82	VPHONE	DEPTNUMBER	6	CHAR	3	0	N
DSN82	VPHDNE	EMPLOYEENUMBER	5	CHAR	6	0	N
DSN82	VPHONE	FIRSTNAME	2	VARCHAR	12	0	N
DSN82	VPHDNE	LASTNAME	1	VARCHAR	15	0	N
DSN82	VPHONE	MIDDLEINITIAL	3	CHAR	1	0	N
DSN82	VPHONE	PHONENUMBER	4	CHAR	4	0	Y
DSN82	VPROJ	DEPTNO	3	CHAR	3	0	N
DSN82	VPROJ	MAJPROJ	8	CHAR	6	0	N
DSN82	VPROJ	PRENDATE	7	DECIMAL	6	0	Y
DSN82	VPROJ	PROJNAME	2	VARCHAR	24	0	N
DSN82	VPROJ	PROJNO	1	CHAR	6	0	N
DSN82	VPROJ	PRSTAFF	5	DECIMAL	5	2	Y
DSN82	VPROJ	PRSTDATE	6	DECIMAL	6	0	Y
DSN82	VPROJ	RESPEMP	4	CHAR	6	0	N
DSN82	VPROJAC	ACENDATE	5	DECIMAL	6	0	Y
DSN82	VPROJAC	ACSTAFF	3	DECIMAL	5	2	Y
DSN82	VPROJAC	ACSTDATE	4	DECIMAL	6	0	Y
DSN82	VPROJAC	ACTNO	2	SMALLINT	2	0	N
DSN82	VPROJAC	PROJNO	1	CHAR	6	0	N
DSN82	VPROJRE1	FIRSTNME	5	VARCHAR	12	0	N
DSN82	VPROJRE1	LASTNAME	7	VARCHAR	15	0	N
DSN82	VPROJRE1	MAJPROJ	8	CHAR	6	0	N
DSN82	VPROJRE1	MIDINIT	6	CHAR	1	0	N
DSN82	VPROJRE1	PROJDEP	3	CHAR	3	0	N
DSN82	VPROJRE1	PROJNAME	2	VARCHAR	24	0	N
DSN82	VPROJRE1	PROJNO	1	CHAR	6	0	N
DSN82	VPROJRE1	RESPEMP	4	CHAR	6	0	N
DSN82	VPSTROE1	PROJ1NAME	2	VARCHAR	24	0	N
DSN82	VPSTROE1	PROJ1NO	1	CHAR	6	0	N
DSN82	VPSTROE1	PROJ2NAME	8	VARCHAR	24	0	N
DSN82	VPSTROE1	PROJ2NO	7	CHAR	6	0	N
DSN82	VPSTROE1	RESP1FN	4	VARCHAR	12	0	N
DSN82	VPSTROE1	RESP1LN	6	VARCHAR	15	0	N

Figure 3.6 DB2 Phone View

The database configuration from the DB2 site has been presented. The database, tables, indexes, views of data, etc. at the DB2 site were duplicated as closely as possible at the ADR site. Below is database configuration documentation from the ADR site. Figure 3.7 shows, in an indented format, the relationship between database, area, file, record, key, element, and field. Figure 3.8 shows the Employee table layout and keys. Fields making up each table and key are shown. Figure 3.9 shows the Department table layout and keys. Fields making up each table and key are shown. Figures 3.10, 3.11, and 3.12 show the Employee, Department, and Cardin dataviews used in the ADR programs. Entities shown in these figures are discussed in detail later in this chapter.

```
                    DATACOM/DD DATADICTIONARY
                         INDENTED REPORT
          COPYRIGHT OF APPLIED DATA RESEARCH, INCORPORATED

ENTITY-TYPE.... OCCURRENCE.................... VER S DESCRIPTION.....................

DATABASE      MINI-SYSTEM-DB                  DT3 P MINI-SYSTEM DATABASE              DIO

  AREA          MS-CARDIN                       DDI P MINI-SYSTEM ACTION AREA     CRD DIO

    FILE          MS-CARDIN                     DDI P MINI-SYSTEM MAJOR SYST TABLE CRD DIO

      RECORD        MS-CARDIN                   DDI P CARDIN RECORD

        KEY           MS-CARDIN.MS-CARDIN-KEY-I     DDI P CARDIN KEY 1          CRDTH DIO MN
        ELEMENT       MS-CARDIN.MS-CARDIN           DDI P                       CRDIN
        FIELD         MS-CARDIN.CARDIM-KEY          DDI P
        FIELD         MS-CARDIN.IM-ACTION           DDI P
        FIELD         MS-CARDIN.IM-EMPNO            DDI P
        FIELD         MS-CARDIN.IN-FNAME            DDI P
        FIELD         MS-CARDIN.IN-LNAME            DDI P
        FIELD         MS-CARDIN.IN-NEWNO            DDI P

  AREA          MS-DEPARTMENT                   DDI P MINI-SYSTEM DEPARTMENT AREA  DEP DIO

    FILE          MS-DEPARTMENT                 DDI P MINI-SYSTEM DEPARTMENT TABLE DEP DDI

      RECORD        MS-DEPARTMENT               DDI P MINI-SYSTEM DEPARTMENT RECORD

        KEY           MS-DEPARTMENT.MS-DEPART-K-I   DDI P MINI-SYSTEM DEPARTMENT  MGRNO DD2
        KEY           MS-DEPARTMENT.MS-DEPART-K-2   DDI P MINI-SYSTEM DEPARTMENT  AOOPT DD4
        KEY           MS-DEPARTMENT.MS-DEPART-K-3   DDI P MINI-SYSTEM DEPARTMENT  OPTNO DD5 MN
        ELEMENT       MS-DEPARTMENT.MS-DEPARTMENT   DDI P ENTIRE MINI-SYSTEN DEPT  DEPTE
        ELEMENT       MS-DEPARTMENT.MS-DEPARTMENT-2 DDI P DEPTNAME.DEPTNO ELEMENT  DEPT2
        FIELO         MS-DEPARTMENT.ADMRDEPT        DDI P
        FIELO         MS-DEPARTMENT.DEPTNAME        DDI P
        FIELO         MS-DEPARTMENT.DEPTNO          DDI P
        FIELO         MS-DEPARTMENT.MGRNO           DDI P

  AREA          MS-EMPLOYEE                     DDB P MINI-SYSTEM EMPLOYEE AREA    EMP DIO

    FILE          MS-EMPLOYEE                   DDB P MINI-SYSTEM EMPLOYEE TABLE   EMP DD2

      RECORD        MS-EMPLOYEE                 DDB P MINI-SYSTEM EMPLOYEE RECORD

        KEY           MS-EMPLOYEE.MS-EMPLOYEE-K-I   DDB P MINI SYSTEM KEY 1       EMPNO DD1 MN
        KEY           MS-EMPLOYEE.MS-EMPLOYEE-K-2   DDB P MINI SYSTEM EMPLOYEE KE WKOPT DD3
        KEY           MS-EMPLOYEE.MS-EMPLOYEE-K-3   DDB P LASTNAME.FIRSTNNE.MIDIN ENAME DD6
        ELEMENT       MS-EMPLOYEE.MS-EMPLOYEE       DDB P ENTIRE RECORD EMPLOYEE  EMPLE
        ELEMENT       MS-EMPLOYEE.MS-EMPLOYEE-2     DDB P                         NAMNO
        FIELO         MS-EMPLOYEE.BRTHDATE          DDB P
        FIELO         MS-EMPLOYEE.EDUCLVL           DDB P
        FIELO         MS-EMPLOYEE.EMPNO             DDB P
        FIELO         MS-EMPLOYEE.FIRSTNME          DDB P
```

Figure 3.7 ADR Test Data Base Configuration

```
                    ********************************************************
                    *                 DATACOM/DD DATADICTIONARY            *
                    *                     INDENTED REPORT                  *
                    *       COPYRIGHT OF APPLIED DATA RESEARCH, INCORPORATED*
                    ********************************************************

ENTITY-TYPE.... OCCURRENCE..................... VER S DESCRIPTION.....................

          FIELD     MS-EMPLOYEE.FULLNAME          008 P
          FIELD     MS-EMPLOYEE.HIREDATE          008 P
          FIELD     MS-EMPLOYEE.JOBCODE           008 P
          FIELD     MS-EMPLOYEE.LASTNAME          008 P
          FIELD     MS-EMPLOYEE.LINENO            008 P
          FIELD     MS-EMPLOYEE.MIDINIT           008 P
          FIELD     MS-EMPLOYEE.PHONENO           008 P
          FIELD     MS-EMPLOYEE.SALARY            008 P
          FIELD     MS-EMPLOYEE.SEX               008 P
          FIELD     MS-EMPLOYEE.WORKDEPT          008 P
```

Figure 3.7 ADR Test Data Base Configuration

VIEW ALL DISPLAY

BASE NAME: MINI-SYSTEM-DB BASE-ID: 10 RECORD NAME: MS-EMPLOYEE

KEY NAME

MS-EMPLOYEE-K-1
MS-EMPLOYEE-K-2

LV	FIELD NAME	TYPE	LEN	DEC	SIGN	OCC	CL	DESCRIPTION
01	LINENO	C	2			1	S	
01	BRTHDATE	N	5		Y	1	S	
01	EDUCLVL	C	2			1	S	
01	EMPNO	C	6			1	S	
01	FIRSTNME	C	12			1	S	
01	HIREDATE	N	5		Y	1	S	
01	JOBCODE	N	3		Y	1	S	
01	LASTNAME	C	15			1	S	
01	MIDINIT	C	1			1	S	
01	PHONENO	C	4			1	S	
01	SALARY	N	7		Y	1	S	
01	SEX	C	1			1	S	
01	WORKDEPT	C	3			1	S	
01	END					1	C	

VIEW KEYS DISPLAY

BASE NAME: MINI-SYSTEM-DB BASE-ID: 10 RECORD NAME: MS-EMPLOYEE

MS-EMPLOYEE-K-1 KEY CONSISTS OF:

FIELD NAME	TYPE	LEN	DEC	SIGN	OCC	CL	DESCRIPTION
EMPNO	C	6			1	S	

MS-EMPLOYEE-K-2 KEY CONSISTS OF:

FIELD NAME	TYPE	LEN	DEC	SIGN	OCC	CL	DESCRIPTION
WORKDEPT	C	3			1	S	

Figure 3.8 ADR Employee Table Layout and Keys

```
                              VIEW ALL DISPLAY

    BASE NAME: MINI-SYSTEM-DB  BASE-ID:  10    RECORD NAME: MS-DEPARTMENT

  KEY NAME
  ---------------
MS-DEPART-K-1
MS-DEPART-K-2
MS-DEPART-K-3

LV FIELD NAME       TYPE LEN DEC SIGN OCC CL DESCRIPTION
-- ---------------- ---- --- --- ---- --- -- --------------------
01 ADMRDEPT          C     3           1 S
01 DEPTNAME          C    36           1 S
01 DEPTNO            C     3           1 S
01 MGRNO             C     6           1 S
01 END                                 1 C
```

```
                              VIEW KEYS DISPLAY

    BASE NAME: MINI-SYSTEM-DB  BASE-ID:  10    RECORD NAME: MS-DEPARTMENT

MS-DEPART-K-1 KEY CONSISTS OF:
  FIELD NAME       TYPE LEN DEC SIGN OCC CL DESCRIPTION
  ---------------- ---- --- --- ---- --- -- --------------------
  MGRNO             C     6           1 S

MS-DEPART-K-2 KEY CONSISTS OF:
  FIELD NAME       TYPE LEN DEC SIGN OCC CL DESCRIPTION
  ---------------- ---- --- --- ---- --- -- --------------------
  ADMRDEPT          C     3           1 S

MS-DEPART-K-3 KEY CONSISTS OF:
  FIELD NAME       TYPE LEN DEC SIGN OCC CL DESCRIPTION
  ---------------- ---- --- --- ---- --- -- --------------------
  DEPTNO            C     3           1 S
```

Figure 3.9 ADR Department Table Layout and Keys

```
DVW MS-EMPLOYEE-2-010091 PROD              MARCH 5, 1987      09:27:32

DATAVIEW:       MS-EMPLOYEE-2-U VERSION: 008 STATUS:  PROD
                  DATAVIEW: MS-EMPLOYEE-2-U
                  --- ----- ------------------  - - ----- ----- - ----------------------
                  SEQ LEVEL FIELD NAME          T I CH/DG OCCUR K VALUE/REDEF/DEP ON
                  --- ----- ------------------  - - ----- ----- - ----------------------
                  CATALOGED 02/20/87 08:27      DATACOM/DB UPD=YES DBID=010
                    I I     MS-EMPLOYEE-2-U
                    2  2     WORKDEPT           X       3        K
                    3  2     EMPNO              X       6        K
                    4  2     FULLNAME                            P
                    5    3     LASTNAME         X      12        P
                    6    3     FIRSTNME         X      12
                    7    3     MIDINIT          X       1
                    8  2     PHONENO            X       4

              LEGEND:
                    SEQ=SEQUENCE NUMBER
                    T (FIELD TYPE):  X=ALPHANUMERIC, N=NUMERIC, U=UNSIGNED , C=COND. NAME,
                    V=VARIABLE, G=DATE
                    I (INTERNAL NUMERIC TYPE):  P=PACKED, Z=ZONED, B=BINARY
                    CH/DG (CHARACTERS/DIGITS):  NUMBER OF CHARACTERS, INTEGERS, OR INT.DEC
                    K (KEY USAGE):  K=WHOLE KEY, P=PARTIAL KEY (HIGH ORDER POSITION)
                    REDEF=REDEFINITION, DEP ON=DEPENDING ON
```

Figure 3.10 ADR Employee Dataview

```
DVW MS-DEPARTMENT-210021 PROD              MARCH 5, 1987      04:27:22

DATAVIEW:       MS-DEPARTMENT-2 VERSION: 002 STATUS:  PROD
                  DATAVIEW: MS-DEPARTMENT-2
                  --- ----- -------------------  - - ----- ----- - ------------------------
                  SEQ LEVEL FIELD NAME            T I CH/DG OCCUR K VALUE/REDEF/DEP ON
                  --- ----- -------------------  - - ----- ----- - ------------------------
                  CATALOGED 02/16/87 14:02      DATACOM/DB UPD=YES OBJ=010
                    1 1     MS-DEPARTMENT-2
                    2  2     DEPTNAME            X       36
                    3  2     DEPTNO              X        3        K

          LEGEND:
                  SEQ=SEQUENCE NUMBER
                  T (FIELD TYPE):  X=ALPHANUMERIC, N=NUMERIC, U=UNSIGNED , C=COND. NAME,
                  V=VARIABLE, D=DATE
                  I (INTERNAL NUMERIC TYPE):  P=PACKED, Z=ZONED, B=BINARY
                  CH/DG (CHARACTERS/DIGITS):  NUMBER OF CHARACTERS, INTEGERS, OR INT.DEC
                  K (KEY USAGE):  K=WHOLE KEY, P=PARTIAL KEY (HIGH ORDER POSITION)
                  REDEF=REDEFINITION, DEP ON=DEPENDING ON
```

Figure 3.11 ADR Department Dataview

```
DVW MS-CARDIN-U    (001) PROD              MARCH 5, 1987      09:27:11

DATAVIEW:       MS-CARDIN-U     VERSION: 001 STATUS:  PROD
                DATAVIEW: MS-CARDIN-U
                --- ----- --------------------  - - ----- ----- - ----------------------
                SEQ LEVEL FIELD NAME            T I CH/DG OCCUR K VALUE/REDEF/DEP ON
                --- ----- --------------------  - - ----- ----- - ----------------------
                CATALOGED 02/24/87 15:22        DATACOM/DB UPD=YES DBID=010
                  1 1     MS-CARDIN-U                          K
                  2  2     CARDIN-KEY                          K
                  3   3      IN-ACTION          X        1      P
                  4   3      IN-LNAME           X       15
                  5   3      IN-FNAME           X       12
                  6   3      IN-EMPNO           X        6
                  7   3      IN-NEW-O           X        4

        LEGEND:
                SEQ=SEQUENCE NUMBER
                T (FIELD TYPE):  X=ALPHANUMERIC, N=NUMERIC, U=UNSIGNED , C=COND. NAME,
                V=VARIABLE, D=DATE
                I (INTERNAL NUMERIC TYPE):  P=PACKED, Z=ZONED, B=BINARY
                CH/DG (CHARACTERS/DIGITS):  NUMBER OF CHARACTERS, INTEGERS, OR INT.,DEC
                K (KEY USAGE):  K=WHOLE KEY, P=PARTIAL KEY (HIGH ORDER POSITION)
                REDEF=REDEFINITION, DEP ON=DEPENDING ON
```

Figure 3.12 ADR Cardin Dataview

In Figure 3.13, the same DB2 and ADR entities (objects) are shown side-by-side in an indented report manner to allow the reader to judge the similarity of the two. This list of entities was the part implemented for the mini-system.

DB2	ADR
Storage group: DSN8G200	
Data base: DSN8D2AP	Database: Mini-system
Table space: DSN8S2DP	Area: MS-DEPARTMENT
Table: DEPARTMENT	File: MS-DEPARTMENT
	Record: MS-DEPARTMENT
Index: DSN82.XDEPT1	Key: misc. keys
	Element: misc. elms
	Field: misc. flds
Table space: DSN8S2EM	Area: MS-EMPLOYEE
Table: Employee	File: MS-EMPLOYEE
	Record: MS-EMPLOYEE
Index: DSN82.XEMPL1	Key: misc. keys
Index: DSN82.XEMPL2	
	Element: misc. elms
	Field: Misc. flds
	Area: MS-CARDIN
	File: MS-CARDIN
	Record: MS-CARDIN
	Key: misc. keys
	Element: misc. elms
	Field: Misc. flds

Figure 3.13 Side-by-side Comparison

There were a few database duplication problems encountered in the implementing the DB2 database configuration at the ADR site that needed to be handled. The two databases used for the two mini-systems were essentially the same, see Figure 3.13.

DB2 has a Storage Group as its highest level Object. ADR has a Database as its highest level Entity. A DB2 Storage Group is a named set of DASD volumes on which DB2 can be stored.

In the hierarchy of a DB2 database are one or more Tables Spaces logically comprising the database. Each Table Space has one or more table. A table is a relation of rows and columns. Data from one or more table can be co-located in the same Table Space (ie., file A record, file B record, file A record... co-located within the same Table Space). ADR's counterpart to the Table Space is the Area. An Area can contain one or more Files. Data can also be co-located within the same Area. Co-location of data can improve performance if the co-located records are frequently accessed together per

formance may suffer if they are not often accessed together.

3. In the hierarchy of an ADR File (table) are Records, which occur below the ADR file. DB2 has no counterpart the Record. ADR development has talked about eliminating the Record entity type entirely since there is always 1 to 1 relationship between record and file.

4. Below an ADR record is one or more keys, elements, and fields. Each ADR Database has one index which contains all keys for that database. A file can have one or more keys. An element is one or more contiguous fields from the file. There can be one or more elements for each record. An ADR dataview is related to fields by-way-of elements. There is a relationship between an ADR Dataview and one or more elements and there is a relationship between each element and one or more contiguous fields.

DB2's entities are a bit different. An ADR Key is called an Index. A Table can have one or more DB2 Indexes. An ADR database may have but one Index. There is

no counter part for an ADR Element.

DB2 calls fields, columns. A Table will have 1 or more Columns and 1 or more rows (records). DB2 Views of data are "mapped" to fields through SQL definition statements. See the example below:

Figure 3.14 provides a SQL definition of a View called PHONE. which is a view of telephone information. This view is used by the DB2 mini-system batch program. It was created externally to the batch program.

```
CREATE VIEW PHONE AS
  SELECT DEPARTMENT.DEPTNAME, DEPARTMENT.DEPTNUMBER,
         EMPLOYEE.EMPLOYEENUMBER, EMPLOYEE.FIRSTNAME,
         EMPLOYEE.LASTNAME, EMPLOYEE.MIDDLEINITIAL,
         EMPLOYEE.PHONENUMBER
     FROM DEPARTMENT,
          EMPLOYEE
     WHERE DEPARTMENT.DEPTNO =
           EMPLOYEE.WORKDEPT
```

Figure 3.14 SQL Phone View Definition

At the ADR site, with the current software releases, a dataview can contain fields from only a single file. "Logical joins" of files are accomplished by nesting accesses to files (ie., nesting Ideal "FOR _____ dataview" statements, see Figure 3.15). In future software releases, a "dataview processor" has been promised which would allow a single dataview to access fields from multiple files, or views simplifying the programming effort and providing a join of one or more files or views.

3.1 Program conversion problems

There were a few problems that came up in converting the two DB2 programs to Datacom/DB at the ADR site. This section contains a discussion of the problems that were encountered.

Both the DB2 batch report and the on-line program are written in Cobol embedded with SQL statements. Their counter parts were written in ADR/Ideal (not Cobol). Ideal was chosen instead of Cobol for a variety of reasons. First, since ADR products were acquired, Ideal has become the standard language used by the ADR site. Prior to Ideal, PL/1 was the shop standard. A second reason for programming in Ideal was that management at the ADR site preferred that IDEAL be used in this product comparison. The author of this paper works at that site and wanted the full cooperation of management on the project, so the author complied with the request.

Program listings and program output from the DB2 and ADR sites are provided in the Appendices as listed in the Table of Contents.

The DB2 on-line program was streamlined a bit at the ADR site out of practicality to remove irrelevant complication. The part that was streamlined was a series of DB2 programs involved in validation of user input. In timing program response times on the Employee Index, only valid input was used. Timings did not begin until after input was validated, hence the validation was not a factor in the timings and the DB2 validation programs were irrelevant complication.

The user is prompted for 5 things, 4 of which were validated (Major system, Action, Object and Search criteria). Valid input in the DB2 and ADR versions of the program was: "O" for Major System ("O" short for Organization), "D" for action ("D" short for Display), "EM" for Object ("EM" short for EMployee) and "EI" for Search Criteria ("EI" for Employee Id). A series of programs is involved in the DB2 system for this validation. Once valid input was received for these 4 prompts, the

Display Index portion of the on-line program, for example, then read a screen-full of records based on a read-key greater than or equal the user input provided on the fifth line of the screen (ie., An employee id of 000001). Various records were read, data was gathered for each line of the index and the screen was then painted with an the first page of the index. The user could then press one of 3 PF keys to continue. (PF2 to resend the initial screen with all values removed, PF3 to end the program or PF8 to page forward on the index of records).

Data was gathered from two files, the EMPLOYEE file and DEPARTMENT file, in painting the screen.

DB2 SQL has CURSORS to aid in selection of EMPLOYEE and DEPARTMENT records. A cursor is a named control structure used by an application program to point to a row in a table. The position of the row is within some ordered set of rows, and the cursor is used to retrieve rows from the set. ADR has no such aid. Rows had to be selected from the MS-EMPLOYEE and MS-DEPARTMENT files by

nesting FOR statements in Ideal, see Figure 3.15.

```
FOR EACH MS-EMPLOYEE-2 NO UPDATE
     WHERE MS-EMPLOYEE-2.LASTNAME = WK-IN-LNAME
     ORDERED BY WORKDEPT EMPNO
  various statements
  FOR EACH MS-DEPARTMENT-2 NO UPDATE
       WHERE MS-DEPARTMENT-2.DEPTNO =
             MS-DEPARTMENT-2.WORKDEPT
    various statements
  WHEN NONE
    various statements
  ENDFOR  :FOR EACH MS-DEPARTMENT-2
WHEN NONE
  various statements
ENDFOR  :FOR EACH MS-EMPLOYEE-2
```

Figure 3.15 ADR Ideal Nested FOR Statements

Another problem was access of JCL SYSIN card images with ADR's Ideal. One can do so only by calling a non-Ideal (ie., Cobol or Assembler subprogram which accesses the SYSIN file). PL/1 subprogram interfaces were considered not a priority item with the Ideal development staff, and may be added at a later date.

Ideal handles I/O to and from ADR/Datacom/DB files and ADR/Ideal-defined Panels and Reports very well, non-Datacom file I/O is handled only through non-Ideal subprograms. In the mini-system, as is typically done at the ADR site, SYSIN card images were loaded in a Datacom/DB file using an online data entry utility.

ADR Panel definition and Report definition was very easy and flexible. ADR's Report Definition Facility made setting up and modifying a screen very easy. DB2 used CICS BMS (basic mapping support) which makes screen "painting" a very tedious task. DB2 provides no tools to aid in working with BMS, however, screen-painting software can be purchased from various vendors.

3.2 Program Timings

Program timing statistics were kept for each program. The batch and on-line programs were specifically run at times of the day when they were the only programs running and had no competition from any other program for computer resources. The DEPARTMENT file had 6 records in it. The EMPLOYEE file originally had just 32 records, but 1000 records were added at each site bringing the total number of EMPLOYEE records to 1032. Both sites had exactly the same data in their files. Those timings are given in Figure 3.16.

DB2	ADR
Batch report program	Batch report program
CPU time: 14.78 sec.	CPU time: 13.05 sec.
Actual time: .31 min.	Actual time: .25 min.
On-line index display	On-line index display
Index 1st page: 3 sec.	Index 1st page: 2 sec.
PF8, Index 2nd pg: 3 sec.	PF8, Index 2nd pg: 2 sec.

Figure 3.16 Program Timings

Chapter 4

Conclusions and Future Work

4.0 Conclusions

One would think that being a current user of ADR software, the author would be strongly prefer the more familiar ADR software over DB2 DBMS software. Prior to the study, this was the case. After the study of these two commercial DBMS software systems, that was not the case. An appreciation has developed for the potential flexibility and impressive features possessed by DB2 software.

Announced in June 1983, and not generally available until April 1985, DB2 has proved to be a very impressive set of products.

"ADR's DBMS is quasi-relational, utilizing a relational front end that provides end-user friendliness. This is not to say that it does not possess relational

capabilities. DB2 is slow compared to its competitors. However, it is a young product and because it is supported and enhanced by IBM, DB2's prospects for success are excellent...". [16] The logical data structure underlying the front end is hierarchical, which does not meet Date's criteria as being truly relational.

ADR/Datacom/DB is, nevertheless, an extremely popular DBMS. Although not a true representation of a relational system in the mold of IBM's SQL/DS or DB2, ADR/Datacom/DB does come with an impressive list of options and "add-ons" that allow the user to build a DBMS environment conducive to the operations of the installation.

In the second and third chapters of this paper, some differences between DB2 and ADR products were described. Out of all those differences, the most prominent is the view of data. DB2 provides much more flexibility in the construction of its views. DB2 view can be constructed dynamically by a program, if authorized. DB2 views can consist of fields from more than

one table, of fields from other views or from a combination of views and tables. The programmer does not need to be concerned with "logical joins" of 2 or more tables as he does with ADR software. A programmer using ADR software accomplishes "logical joins" of tables by nesting the file accesses. The programmer must be aware of the data that needs to be gathered and processed and whether it comes from one or more tables. A DB2 programmer may be provided with one preconstructed view of data which consists of fields from any combination of tables and/or views. The DB2 programmer's work is reduced in this respect. A dataview processor is promised in a future release of ADR products which should solve this for the ADR programmer.

ADR's fourth generation language, Ideal, provides many conveniences for the programmer which IBM's Cobol-embedded SQL does not have. The task of screen production in on-line programming is simplified greatly. ADR has PDF, a Panel (screen) Definition Facility, which provides separate fill-in screen-design aids which allows the programmer to "paint" sophisticated screen very simply and quickly. DB2 at this point uses CICS BMS (basic

mapping support). Screens can be painted using non-DB2 CICS screen design tools. None is provided with DB2 products.

Another Ideal time-saver is report production capabilities. Ideal provides RDF, a Report Definition Facility, to aid in the production of routine batch report production. Things such as column headings, page numbers, summary totals, level breaks, lines per page, etc. are handled very by RDF. The programmer defines the Report separately from the program and lists the report in the Resource Section of his Ideal program (along with any Dataviews, Sub programs or Panels used). DB2 provides a report writer, but the DB2 site in section 3's mini-system had not yet purchased it, so depended on an application programmer's report producing skills using Cobol.

Nothing is defined dynamically within an ADR program. All dataviews, indexes, etc. must be defined outside of the program. With the proper authorization, DB2 provides the flexibility of dynamic definition of entities within application programs. However, there are

some limitations in updating data when using views containing fields from more than 1 table. For non-update processing of data, however, this does remove some burden from the application programmer.

"4GLs (fourth generation languages) are easier for a programmer to use than Cobol. Novice programmers with very little commercial experience can learn Ideal 'and run with it' and become very productive very quickly. Independent software vendors will have to respond to this by modifying their tools to work with DB2. It's not difficult to speculate that soon you will be able to buy DB2 from IBM and acquire...Ideal from ADR to work with it." [54]

ADR in early 1987 announced support of DB2 and initially, a selected subset of SQL. ANSI (American National Standards Institute) recently recognized SQL as an industry data manipulation language standard.

All data in a DB2 data base is stored in VSAM entry sequenced data sets (ESDS), ADR uses Datacom /DA (direct access format) files.

DB2 product documentation, which is extensive, presents DB2 in a well-organized and straightforward manner and as such should be considered an additional product strength. ADR could use a some improvement in this area. ADR should pattern their documentation after DB2's.

The fact that DB2 can coexist with and complement IMS/VS/DB and allows concurrent access to data is a definite advantage. However, at this point DB2 is still being used primarily as an adjunct to the IMS/VS/DB, which causes DB2 to function less as a full relational DBMS, and more as a data language that gives IMS/VS/DB users relational capability. "DB2 coexists with IMS and can share data via an extraction module. Users can move from IMS to DB2 and still protect their investment in IMS applications". 16

"Except for SQL/DS, DB2 and a few others, most of the products offer an interpretive data sublanguage facility. To improve efficiency and obtain a higher performance, relational systems should support data sublanguage compilers instead of interpreters. In

addition to this, they should provide a precompiler for support of the callable procedural languages such as Cobol. This reduces the run-time system overhead and thus improves performance." 16

DB2 uses IMS DB/DC as its data dictionary. IBM has obviously not "re-invented the wheel" in a few areas of its software; VSAM files are used for data storage and IMS's DB/DC data dictionary is used for DB2. DB/DC would need to be purchased.

"Relational systems utilize a higher-level language for separating the translation from source code from the execution of target object code. Consequently, they require more computing resources for the translation to be effective, and human productivity is increased at the expense of the additional computing resources necessary." 16 With labor costs steadily rising and hardware costs dropping, this hardly appears to be a problem.

Figure 3.16 shows timings were kept for programs in the miniature application system programming effort of chapter 3. The purpose of the timings was to make

further comparison of the two DBMS product families through on-line and batch programs written in each environment. In this test there was very little difference in the timings for either set of programs, on-line or batch. The ADR programs did execute a bit faster than their counter parts.

It was interesting to have the timings as close as they were. The two test sites were selected because of their similarities in operating environments (see Figure 3.2, profiles of test sites). Conditions at the DB2 site were duplicated at the ADR site as closely as possible. Yet differences, as discussed in detail in chapter 3, did exist (ie., Differences in the data base configurations, programming languages used, dynamic verses static access paths to data, etc.). On the small scale in which this test product comparison was conducted, approximately 1000 Employee and 10 Department file records were processed, timings were almost the same. On a much larger scale the differences in the timings may have been much greater. However, the mini-application programming effort of chapter 3 was an excellent vehicle in discussing DBMS product family differences in detail.

4.1 Future Work

It was mentioned before that ADR has announced a "support commitment for such industry standards ... as SQL, DB2, SQL/DS,... to equip users with a foundation for investment protection while being positioned for technology refresh". [16] ADR plans to start with a "subset of SQL". These plans were just announced in early 1987. No further details are available. The SQL subset will very probably contain no data definition statements. It will very likely contain only selected data manipulation statements. Providing SQL support will allow one to write a program access either a DB2 database or ADR database or both within the same program.

Future work could include a study of the subset of SQL that ADR chooses to support. Details of the interfacing products should be documented as well. A performance comparison similar to the effort in this paper could be done. Programs could be written demonstrating performance comparisons between ADR's SQL subset and DB2's data manipulation language.

ADR and DB2 DBMS products continue to evolve to meet their customer's needs. The versions of software for each family of products will need to provide downward support of existing customer database configurations, but likely will contain many new and exciting enhancements for the customer.

Bibliography

1. ADR Manual No. DB46-DB-00, Data Base Administration, 1986.

2. ADR Manual No. DB46-DB-01, Data Base Design, 1986.

3. ADR Manual No. DD13-DD-00, Data Dictionary Administration, 1986.

4. ADR Manual No. ID13-ID-00, Ideal Administrator, 1986.

5. Anonymous, IBM upgrades, repositions DB2; Product's role poses threat to independent DBMS vendors, Computerworld, July 9, 1986, pg 15 focus.

6. Babcock, Charles, A tale of two IBM DBMS, Computerworld, July 28, 1986, pg 19,22.

7. Carlyle R. Emmett, System Software: Calling All Data, Datamation, October 15, 1985, pg 40-42.

8. Carlyle, R. Emmett, Software: Whose Turn to Cry?, Datamation, April 15, 1986, pg 28,32.

9. Codd, E.F., Is your DBMS really relational?, Computerworld, October 14, 1985, pg 1-9 indepth.

10. Codd, E.F., Does your DBMS run by the rules?, Computerworld, October 21, 1985, pg 49-60.

11. Codd, E.F., The real strengths of relational systems, Computerworld, Feburary 6, 1984, pg 44-45.

12. Curtice, Robert M. and Casey, William, Database: What's in store?, Datamation, December 1, 1985, pg 83-84.

13. Data Decisions, IBM Corp Database 2 (DB2), Relational Data Base System Profile/Evaluation, July 1985, pg 1-4.

14. Data Decisions, Applied Data Research, ADR/DATACOM/DB, Database Management System Profile/Evaluation, August 1985, pg 1-3.

15. Datapro Research Corporation, ADR/Datacom/DB Profile/Evaluation, January 1987, pg 1-5.

16. Datapro Research Corporation, IBM DB2 (Database 2) Profile/Evaluation, January 1987, pg 1-5.

17. Date, C.J., How Relational Systems Perform, Computerworld, Feb 13, 1984, pg 15-20.

18. Date, C.J., A Guide to DB2, July 1985.

19. Date, C.J., Relational Data Base Selected Writings, 1986.

20. Farin, Jeff and Nazario, Amor, DBMS Basics, Infosystems, June 1986, pg 42-47.

21. Gallant, John, Relational sell really straight?, Computerworld, November 1, 1985, pg 47, 54.

22. Gallant, John, IBM, independents trade views on DBMS, Computerworld, November 25, 1985, pg 8.

23. Gallant, John, Strained relations: DBMS debate turns bitter, Computerworld, January 13, 1986, pg 1, 8.

24. Gallant, John, Sideline view: Users want resolution to relational debase, Computerworld, January 20, 1986, pg 1, 7.

25. Gallant, John, Blue horizon: Users rate DB2 high, await new version, Computerworld, Feburary 3, 1986, pg 1, 14-15.

26. Gallant, John, IBM pushes relational, updates DB2/IBM to support IMS full function for the 'Indefinite Future', Computerworld, February 10, 1986, pg 1, 4-5.

27. Gerrard, Steven, A pragmatic response to relational rules, Computerworld, January 27, 1986, pg 61, 64, 66.

28. Gillin, Paul, Industry Pioneer Speaks; Strategy Positioning for Mainframe Software Independents, Computerworld, July 23, 1984, pg 12-13.

29. Hamilton, Rosemary, ADR deal: Rivals downplay importance, users look to future, Computerworld, November 15, 1985, pg 5.

30. Hessinger, Paul E., DB2 complementing IMS: IBM's data base strategy for tomorrow, Computerworld, April 9, 1984, pg 9-30 indepth.

31. Hessinger, Paul, Integrated Architecture for Integrated Systems; Key to Success with DB2, Computerworld, Dec 7, 1986. pg 40-46.

32. Hessinger, Paul, DB2 and IMS, Computerworld, December 4, 1985, pg 57-68.

33. IBM Alert, Query Management Facility (QMF) Version 2 Release 1, November 1986, pg 1-10.

34. IBM Alert, Cross System Product/Application Development Version 3 and Cross System Product/Application Execution Version 3, September 12, 1986, pg 1-17.

35. IBM Manual No. GC26-4073-2, IBM Database2 General Information, 1986.

36. IBM Manual No. GC24-1582, IBM Database 2 Concepts and Facilities Guide, 1986.

37. IBM Manual No. GC24-1581, IBM Database 2 Relational Concepts, 1986.

38. IBM Manual No. GC24-1600-0, IBM Database 2 Performance Design and Tuning Guide, 1986.

39. IBM Manual No. SC26-4077-2, IBM Database 2 Database Planning and Administration Guide, 1986.

40. IBM Manual No. SC26-4130-00, Development Guide Relational Applications, 1986.

41. IBM Programming Announcement, Data Extract Version 2 Release 1, February 4, 1986, pg 1-3.

42. IBM Programming Announcement, IBM DB2 Release 2 Announced with Planned Availability March 7, 1986, February 4, 1986, pg 1-8.

43. IBM Software Catalog, Data Systems, Database 2 (DB2), 1986, pg 1-5.

44. IBM Systems Journal, Vol 23, No. 2, 1984, pg 98-219.

45. Kull, David, DB2: Build for Comfort and Speed?, Computer Decisions, October 8, 1985, pg 48-56, 104.

46. Kull, David, A Relational Standard Arrives?, Computer Decisions, Jan 2, 1986, pg 44-46.

47. Kull, David, New DB2 pushing IMS into grave, Computer Decisions, Mar 25, 1986, pg 18-24.

48. Kull, David, Lack of integrity shackles relational DBMS productivity, Computer Decisions, Dec 2, 1986, pg 16-17, 20.

49. McCusker, Tom, IBM bounces back, Datamation, July 1, 1985, pg 32, 34, 39, 42.

50. McEnaney, Maura, Codd: Relational approach to DBMS embraced by users, Computerworld, May 6, 1985, pg 41, 50.

51. Millsap, Ed, Sloan, Ken, and Gerrard, Steve, Relational DBMS, Computerworld, March 4, 1985, pg 1-12 indepth.

52. Perry, Robert L., Relational DBMS takes off, Computer Decisions, Feb 12, 1985, pg 106-108, 110, 114-125.

53. Pfrenzinger, Steven, IBM's DB2: An Analysis, Computerworld, Aug 19, 1985, pg 49, 54.

54. Snyders, Jan, DBMS: A Mature Technology with No Alternative, Infosystems, March 1986, pg 57, 60, 63, 65.

55. Stevens, Lawrence, A Helping hand for SQL, Computer Decisions, Oct 7, 1986, pg 38-39.

56. Rae, Sharon Gamble, ICP Interviews: E.F. Codd, Business Software Review, Oct 1985, pg 57-60.

57. Verity, John W., IBM's DB2 Gets Big Push, Datamation, Mar 15, 1986, pg 24, 26.

58. Verity, John W., Mainframe Survey IBM's DB2 DBMS to make gains, Datamation, May 1, 1986, pg 72.

59. White, Lee, 3 Users - Relational DBMS - Different Experiences, Jul 9, 1986, pg 18-21.

60. Wood, David, Relational systems meet the real world, Data Management, July 1985, pg 10, 12, 14-15.

61. Young, John, Relational databases - benefits and drawbacks, Data Processing (GB), July/Aug 1986, pg 312-313.

Appendix A

ADR Programs and Output

TELEPHONE DIRECTORY

LAST NAME	FIRST NAME	INITIAL	PHONE NUMBER	EMPLOYEE NUMBER	WORK DEPT	WORK NAME
NAAS	CHRISTINE	I	3978	000010	A00	SPIFFY COMPUTER SERVICE DIV.
LUCCHESI	VINCENZO	G	3490	000110	A00	SPIFFY COMPUTER SERVICE DIV.
O'CONNELL	SEAN		2167	000120	A00	SPIFFY COMPUTER SERVICE DIV.
THOMPSON	MICHAEL	L	3476	000020	B01	PLANNING
KWAN	SALLY	A	4738	000030	C01	INFORMATION CENTER
QUINTANA	DELORES	M	4578	000130	C01	INFORMATION CENTER
NICHOLLS	HEATHER	A	1793	000140	C01	INFORMATION CENTER
STERN	IRVING	F	5423	000060	D11	MANUFACTURING SYSTEMS
ADAMSON	BRUCE		4510	000150	D11	MANUFACTURING SYSTEMS
PIANKA	ELIZABETH	R	3782	000160	D11	MANUFACTURING SYSTEMS
YOSHIMURA	MASATOSHI	J	2890	000170	D11	MANUFACTURING SYSTEMS
SCOUTTEN	MARILYN	S	1682	000180	D11	MANUFACTURING SYSTEMS
WALKER	JAMES	H	2986	000190	D11	MANUFACTURING SYSTEMS
BROWN	DAVID		4501	000200	D11	MANUFACTURING SYSTEMS
JONES	WILLIAM	T	0942	000210	D11	MANUFACTURING SYSTEMS
LUTZ	JENNIFER	K	0672	000220	D11	MANUFACTURING SYSTEMS
PULASKI	EVA	D	7831	000010	D21	ADMINISTRATION SYSTEMS
JEFFERSON	JAMES	J	4265	000230	D21	ADMINISTRATION SYSTEMS
MARINO	SALVATORE	M	3780	000240	D21	ADMINISTRATION SYSTEMS
SMITH	DANIEL	S	0961	000250	D21	ADMINISTRATION SYSTEMS
JOHNSON	SYBIL	V	8953	000260	D21	ADMINISTRATION SYSTEMS
PEREZ	MARIA	L	9001	000270	D21	ADMINISTRATION SYSTEMS
GEYER	JOHN	B	6789	000050	E01	SUPPORT SERVICES
HENDERSON	EILEEN	W	5498	000090	E11	OPERATIONS
SCHNEIDER	ETHEL	R	8997	000280	E11	OPERATIONS
PARKER	JOHN	R	4502	000290	E11	OPERATIONS
SMITH	PHILIP	X	2095	000300	E11	OPERATIONS

ADR - Batch program output

TELEPHONE DIRECTORY

LAST NAME	FIRST NAME	INITIAL	PHONE NUMBER	EMPLOYEE NUMBER	WORK DEPT	WORK NAME
SETRIGHT	MAUDE	F	3332	000310	E11	OPERATIONS
SPENSER	THEODORE	Q	0972	000100	E21	SOFTWARE SUPPORT
LEE	RAMLAL		2103	000320	E21	SOFTWARE SUPPORT
GOUNOT	JASON	R	5698	000330	E21	SOFTWARE SUPPORT
GOUNOT	JASON	R	5698	000340	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300000	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300001	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300002	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300003	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300004	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300005	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300006	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300007	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300008	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300009	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300010	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300011	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300012	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300013	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300014	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300015	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300016	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300017	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300018	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300019	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300020	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300021	E21	SOFTWARE SUPPORT

TELEPHONE DIRECTORY

LAST NAME	FIRST NAME	INITIAL	PHONE NUMBER	EMPLOYEE NUMBER	WORK DEPT	WORK NAME
DOE	JOHN	E	0000	300022	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300023	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300024	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300025	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300026	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300027	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300028	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300029	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300030	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300031	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300032	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300033	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300034	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300035	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300036	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300037	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300038	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300039	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300040	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300041	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300042	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300043	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300044	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300045	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300046	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300047	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300048	E21	SOFTWARE SUPPORT

TELEPHONE DIRECTORY

LAST NAME	FIRST NAME	INITIAL	PHONE NUMBER	EMPLOYEE NUMBER	WORK DEPT	WORK NAME
DOE	JOHN	E	0000	300049	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300050	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300051	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300052	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300053	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300054	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300055	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300056	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300057	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300058	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300059	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300060	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300061	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300062	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300063	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300064	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300065	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300066	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300067	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300068	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300069	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300070	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300071	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300072	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300073	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300074	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300075	E21	SOFTWARE SUPPORT

TELEPHONE DIRECTORY

LAST NAME	FIRST NAME	INITIAL	PHONE NUMBER	EMPLOYEE NUMBER	WORK DEPT	WORK NAME
DOE	JOHN	E	0000	300076	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300077	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300078	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300079	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300080	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300081	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300082	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300083	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300084	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300085	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300086	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300087	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300088	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300089	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300090	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300091	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300092	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300093	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300094	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300095	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300096	E21	SOFTWARE SUPPORT
DOF	JOHN	E	0000	300097	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300098	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300099	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300100	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300101	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300102	E21	SOFTWARE SUPPORT

TELEPHONE DIRECTORY

LAST NAME	FIRST NAME	INITIAL	PHONE NUMBER	EMPLOYEE NUMBER	WORK DEPT	WORK NAME
DOE	JOHN	E	0000	300103	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300104	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300105	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300106	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300107	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300108	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300109	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300110	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300111	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300112	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300113	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300114	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300115	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300116	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300117	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300118	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300119	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300120	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300121	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300122	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300123	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300124	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300125	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300126	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300127	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300128	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300129	E21	SOFTWARE SUPPORT

TELEPHONE DIRECTORY

LAST NAME	FIRST NAME	INITIAL	PHONE NUMBER	EMPLOYEE NUMBER	WORK DEPT	WORK NAME
DOE	JOHN	E	0000	300994	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300995	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300996	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300997	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300998	E21	SOFTWARE SUPPORT
DOE	JOHN	E	0000	300999	E21	SOFTWARE SUPPORT

TELEPHONE DIRECTORY

LAST NAME	FIRST NAME	INITIAL	PHONE NUMBER	EMPLOYEE NUMBER	WORK DEPT	WORK NAME
THOMPSON	MICHAEL	L	3475	000020	B01	PLANNING
ADAMSON	BRUCE		4510	000150	011	MANUFACTURING SYSTEMS
JEFFERSON	JAMES	J	4265	000230	021	ADMINISTRATION SYSTEMS
JOHNSON	SYBIL	V	8953	000260	D21	ADMINISTRATION SYSTEMS
HENDERSON	EILEEN	W	5498	000090	E11	OPERATIONS

```
                    TELEPHONE DIRECTORY

LAST NAME       FIRST NAME   INITIAL PHONE   EMPLOYEE WORK  WORK
                                     NUMBER   NUMBER   DEPT  NAME

DSN80081 - NO EMPLOYEE FOUND IN TABLE
```

TELEPHONE DIRECTORY

LAST NAME	FIRST NAME	INITIAL	PHONE NUMBER	EMPLOYEE NUMBER	WORK DEPT	WORK NAME
BROWN	DAVID		4501	000200	D11	MANUFACTURING SYSTEMS

TELEPHONE DIRECTORY

LAST NAME	FIRST NAME	INITIAL	PHONE NUMBER	EMPLOYEE NUMBER	WORK DEPT	WORK NAME
JONES	WILLIAM	T	0942	000210	011	MANUFACTURING SYSTEMS
JOHNSON	SYBIL	V	8953	000260	021	ADMINISTRATION SYSTEMS

TELEPHONE DIRECTORY

LAST NAME	FIRST NAME	INITIAL	PHONE NUMBER	EMPLOYEE NUMBER	WORK DEPT	WORK NAME
SMITH	DANIEL	S	0961	000250	D21	ADMINISTRATION SYSTEMS
SMITH	PHILIP	X	2095	000300	E11	OPERATIONS

```
---------- TELEPHONE DIRECTORY ----------

LAST NAME       FIRST NAME   INITIAL PHONE   EMPLOYEE WORK WORK
                                     NUMBER NUMBER   DEPT NAME

DSN8DD4I - EMPLOYEE SUCCESSFULLY UPDATED
```

```
PGM  CCFA958I   001   TEST    SYS:#IO    FEBRUARY 25, 1987    18:29:01

IDENTIFICATION: CCFA958I          VERSION: 001 STATUS:  TEST

                PROGRAM CCFA958I

                CREATED              02/23/87                    BY GARTEN
                LAST MODIFIED        02/25/87 AT 16:50           BY GARTEN
                LAST COMPILED        02/25/87 AT 16:51

                RUN STATUS:                PRIVATE
                SHORT DESCRIPTION:
                LANGUAGE:  IDEAL           TARGET DATE __ __ __ ACTUAL DATE __ __ __

                DESCRIPTION:
```

```
PGM  CCFA958I   001   TEST    SYS:610    FEBRUARY 25, 1987    18:29:01

RESOURCES:      CCFA958I          VERSION: 001 STATUS:  TEST

                DATAVIEW          VER   PANEL     VER   REPORT    VER   PROGRAM   VER  SYS

                MS-CARDIN-U       0001                  CCFA90R1  0001  CCFA95V1  0001
                MS-DEPARTMENT-2   0002                                  CCFA99VI  0005
                MS-EMPLOYEE-2     0006
```

```
PGN  CCFA95BI   001   TEST    SYS:$ID    FEBRUARY 25, 1987    18:29:01

DATAVIEW:        NS-CARDIN-U      VERSION: 001 STATUS:  PROD
                   DATAVIEW: NS-CARDIN-U
                   --- ----- ---------------  - - ----- ----- - ----------------
                   SEQ LEVEL FIELD NAME        T I CH/DG OCCUR K VALUE/REDEF/DEP ON
                   --- ----- ---------------  - - ----- ----- - ----------------
                   CATALOGED 02/24/87 15:22    DATACON/DB UPD=YES DBID=010
                     1 1     NS-CARDIN-U                      K
                     2 2      CARDIN-KEY                      K
                     3  3      IN-ACTION          X       1    P
                     4  3      IN-LNAME           X      15
                     5  3      IN-FNAME           X      12
                     6  3      IN-ENPNO           X       6
                     7  3      IN-NEWNO           X       4

              LEGEND:
                   SEQ=SEQUENCE NUMBER
                   T (FIELD TYPE):  X=ALPHANUMERIC, N=NUMERIC, U=UNSIGNED , C=COND. NAME,
                   V=VARIABLE, D=DATE
                   I (INTERNAL NUMERIC TYPE):  P=PACKED, Z=ZONED, B=BINARY
                   CH/DG (CHARACTERS/DIGITS):  NUMBER OF CHARACTERS, INTEGERS, OR INT.DEC
                   K (KEY USAGE):  K=WHOLE KEY, P=PARTIAL KEY (HIGH ORDER POSITION)
                   REDEF=REDEFINITION, DEP ON=DEPENDING ON
```

```
PGN  CCFA95BI   001   TEST    SYS:&ID    FEBRUARY 25, 1987    18:29:01

DATAVIEW:       MS-DEPARTMENT-2 VERSION: 002 STATUS:  PROD
                  DATAVIEW: MS-DEPARTMENT-2
                  --- ----- ---------------  - - ----- ----- - ---------------------
                  SEQ LEVEL FIELD NAME       T I CN/DG OCCUR K VALUE/REDEF/DEP ON
                  --- ----- ---------------  - - ----- ----- - ---------------------
                  CATALOGED 02/16/87 14:02     DATACOM/DB UPD=YES DBID=DID
                    1  1    MS-DEPARTMENT-2
                    2  2    DEPTNAME         X     36
                    3  2    DEPTNO           X      3       K

            LEGEND:
                  SEQ=SEQUENCE NUMBER
                  T (FIELD TYPE):  X=ALPHANUMERIC, N=NUMERIC, U=UNSIGNED , C=COND. NAME,
                  V=VARIABLE, D=DATE
                  I (INTERNAL NUMERIC TYPE):  P=PACKED, Z=ZONED, B=BINARY
                  CN/DG (CHARACTERS/DIGITS):  NUMBER OF CHARACTERS, INTEGERS, OR INT.DEC
                  K (KEY USAGE):  K=WHOLE KEY, P=PARTIAL KEY (NIGH ORDER POSITION)
                  REDEF=REDEFINITION, DEP ON=DEPENDING ON
```

```
PGM  CCFA95BI   DD1   TEST    SYS:SID    FEBRUARY 25, 1987    18:29:01

DATAVIEW:       MS-EMPLOYEE-2   VERSION: 006 STATUS:  PROD
                DATAVIEW: MS-EMPLOYEE-2
                --- ----- ------------------  - - ----- ----- - ----------------------
                SEQ LEVEL FIELD NAME          T I CH/DG OCCUR K VALUE/REDEF/DEP ON
                --- ----- ------------------  - - ----- ----- - ----------------------
                CATALOGED 02/16/87 15:45      DATACOM/DB UPD=YES DBID=DID
                  1 1     MS-EMPLOYEE-2
                  2  2    WORKDEPT            X       3       K
                  3  2    EMPNO               X       6       K
                  4  2    FIRSTNME            X      12
                  5  2    LASTNAME            X      15       P
                  6  2    MIDINIT             X       1
                  7  2    PHONENO             X       4

          LEGEND:
                SEQ=SEQUENCE NUMBER
                T (FIELD TYPE):  X=ALPHANUMERIC, N=NUMERIC, U=UNSIGNED , C=COND. NAME,
                V=VARIABLE, D=DATE
                I (INTERNAL NUMERIC TYPE):  P=PACKED, Z=ZONED, B=BINARY
                CH/DG (CHARACTERS/DIGITS):  NUMBER OF CHARACTERS, INTEGERS, OR INT.DEC
                K (KEY USAGE):  K=WHOLE KEY, P=PARTIAL KEY (HIGH ORDER POSITION)
                REDEF=REDEFINITION, DEP ON=DEPENDING ON
```

```
PGM  CCFA95BI   001   TEST    SYS:$IO    FEBRUARY 25, 1987    18:29:01

WORKING DATA:   CCFA95BI          VERSION: 001 STATUS:  TEST

           SEQ     LEVEL  FIELD NAME              T  I  CH/OG  OCCUR  VAL/COMMENT/REDEF/DEP ON/COPY
        ------     -----  ----------------------  -  -  -----  -----  ------------------------------
        000300  1         WK-IN-LNAME-LEN         N        3
        000301  1         WK-IN-FNAME-LEN         N        3
        000303  1         WK-LENGTH               N        3
        000305  1         WK-BLANK-POS            N        2
        000306  1         WK-BLANK-POS-1          N        2
        000307  1         WK-START-POS            N        2
        000308  1         WK-INPUT-LEN            N        2
        000309  1         WK-LASTNAME             X       15
        000310  1         WK-IN-LNAME             V       15
        000311  1         WK-IN-FNAME             V       12
        000312  1         WK-PERCENT-POS          N        2
        000500
        000501  1         ERROR-MSG-LINE
        000502     2       EM-OSNB                X        4      *OSNB*
        004900     2       EM-MSGCODE             X        4
        004901     2       EM-TEXT                X       80
        004902
        004903  1         RPT-99-BLANKS          X       99
        004904
        005000  1         RPT-DETAIL-LINE
        005100     2       RPT-LNAME              X       15
        005101     2       RPT-FILLER-1           X        2
        005200     2       RPT-FNAME              X       12
        005201     2       RPT-FILLER-2           X        1
        005300     2       RPT-MIDDLEINIT         X        1
        005301     2       RPT-FILLER-3           X        7
        005400     2       RPT-PHONENO            X        4
        005401     2       RPT-FILLER-4           X        3
        005500     2       RPT-EMPNO              X        6
        005501     2       RPT-FILLER-5           X        3
        005600     2       RPT-WORKDEPT           X        3
        005601     2       RPT-FILLER-6           X        2
        005700     2       RPT-DEPTNAME           X       36

     LEGEND:
          T (FIELD TYPE):  X=ALPHANUMERIC, N=NUMERIC, U=UNSIGNED, C=COND.NAME, F=FLAG,
                           V=VARIABLE, D=DATE
          I (INTERNAL NUMERIC TYPE):  BLANK OR P=PACKED, Z=ZONED, B=BINARY
          CH/OG (CHARACTERS/DIGITS):  NUMBER OF CHARACTERS, INTEGERS, OR INTEGERS.DECIMALS
          REDEF= REDEFINITION, DEP ON= DEPENDING ON
```

```
PGN  CCFA95B1   DD1   TEST    SYS:$1D    FEBRUARY 25, 1987    18:29:01

REPORT:          CCFA9OR1          VERSION: 001 STATUS:  TEST

          IDENTIFICATION:

                    REPORT NAME  CCFA9CR1

                    CREATED            02/12/87                    BY GARTEN
                    LAST NODIFIED      02/21/87 AT 11:47           BY GARTEN

                    SNORT DESCRIPTION

                    DESCRIPTION:
                      _____________________________________________________
                      _____________________________________________________
                      _____________________________________________________
                      _____________________________________________________
                      _____________________________________________________

          REPORT PARAMETER:

                    LINES PER PAGE ON PRINTOUT                 D6D  (1 THRU 250)
                    REPORT WIDTH                               I32  (40 THRU 23D)
                    SPACING BETWEEN LINES                      2    (I THRU 3)
                    SPACING BETWEEN COLUNNS                    02   (0 THRU 66 OR A=AUTOMATIC)
                    SUNNARIES ONLY                             N    (Y=YES,N=NO)
                    COLUMN NEADINGS DESIRED                    Y    (Y=YES,N=NO)
                    COLUMN NEADINGS INDICATION                 N    (U=UNDERSCORE,N=NONE,D=DASNES)
                    CONTROL BREAK NEADING                      N    (Y=YES,N=NO)
                    CONTROL BREAK FDDTING                      Y    (Y=YES,N=NO)
                    GROUP CONTINUATION AT TOP OF PAGE          N    (Y=YES,N=NO)
                    ANNDTATED COUNT IN CONTROL FOOTINGS        N    (Y=YES,N=NO)
                    REPORT FINAL SUNNARY TITLE                 N    (Y=YES,N=NO)
                      SPACING BEFORE SUMMARY                   I    (I THRU 9 = LINES,P=NEW PAGE)
                      TITLE    _____________________________________

                    DATE
                      POSITION                 ND   (NO=NONE, BR=BOT.RIGHT, BL=BOT.LEFT, BC=BOT.CTR.,
                                                      TR=TOP RIGHT, TL=TOP LEFT, TC=TOP CENTER)
                      FORMAT                   NN/DD/YY

                    PAGE NUNBERS
                      POSITION                 NO
                      FORNAT                   H    (D=DIGITS ONLY, H=WITN NYPNENS, P= PAGE        NNN)

                    PAGE HEADING
                      NEADING                  _____________________________________
                      POSITION                 C    (C=CENTER, L=LEFT JUSTIFY, R=RIGNT JUSTIFY)
```

```
PGM  CCFA95B1   001   TEST    SYS:510    FEBRUARY 25, 1987    18:29:01

REPORT:          CCFA90R1         VERSIUN: 001 STATUS:  TEST

          PAGE HEADING:
                    ------ ------------------------------------------------ -- --- ---------------
                                                                            COLUMN
                                                                            W  T
                    SEO    FIELD NAME, LITERAL, FUNCTION,                   10 A    EOIT PATTERN
                           OR ARITHMETIC EXPRESSION                         TH B
                    ------ ------------------------------------------------ -- --- ---------------
                    000300 '---------------------------- TELEPHONE DIRECTORY'
                    000400 '------------------------------'
                    000401                                                        L02
                    000600 'LAST NAME'
                    000700 'FIRST NAME'                                           018
                    000800 'INITIAL'                                              031
                    000900 'PHONE'                                                039
                    001000 'EMPLOYEE'                                             046
                    001100 'WORK'                                                 055
                    001101 'WORK'                                                 060
                    001102                                                        L01
                    001103 'NUMBER'                                               039
                    001200 'NUMBER'                                               046
                    001201 'OEPT'                                                 055
                    001300 'NAME'                                                 060
                    001301                                                        L01

          LEGENO:
                    SEQ=SEQUENCE NUMBER
                    WIOTH: 0-99=USER OEFINEO
                    TAB: +NN=RELATIVE SPACING,  NNN=ABSOLUTE SPACING, LNN=VERTICAL SPACING,
                         P=TOP OF NEW PAGE
```

```
PGM  CCFA950I   001   TEST    SYS:$1D    FEBRUARY 25, 1987    18:29:01

REPORT:        CCFA90R1        VERSION: 001 STATUS:  TEST

     DETAIL:
                                                  SORT  BREAK  FUNCTION COLUMN
                                                  L A   L S I  T N H A  H W  T
                    FIELD NAME, LITERAL,          V /   V K N  D A I V  D ID A  EDIT
          SEQ       FUNCTION, OR                  L D   L P D  T X N G  G TH B  PATTERN        HEADING
          ---- ------------------------------     - -   - - -  - - - -  - -- -  -------------- ----------
          001401 RPT-DETAIL-LINE                                          U 99

     LEGEND:
          LVL=LEVEL: 1=MAJOR, 2-9=MINOR
          A/D=ASCENDING/DESCENDING:  A=ASCENDING, D=DESCENDING
          SKP=SKIP: P=TOP OF NEW PAGE,  0-9=NUMBER OF LINES
          IND=INDICATION: OMITTED/G=NEW GROUP,  N=NONE, R=REPEATED
          TOT=TOTAL: A=ANNOTATED, S=SINGLE SUMMARY LINE
          MAX=MAXIMUM: A=ANNOTATED, S=SINGLE SUMMARY LINE
          MIN=MINIMUM: A=ANNOTATED, S=SINGLE SUMMARY LINE
          AVG=AVERAGE: A=ANNOTATED, S=SINGLE SUMMARY LINE
          HDG=HEADING: N=NONE, U=USER DEFINED

                         *** REPORT INTERNAL RECORD SIZE    95 BYTES
```

```
PGM  CCFA95BI   001   TEST    SYS:$IO    FEBRUARY 25, 1987    18:29:01

PROCEDURE:    CCFA95BI          VERSION:  001  STATUS:  TEST

 SEQ  STATEMENT
 100 :IDEAL BATCH PGM CCFA95BI - FOR MINI-SYSTEM DEMO
 200 :
 300 <<MAIN>> PROCEDURE
 500
 600     FOR EACH MS-CARDIN-U NO UPDATE
 700
 701     SET RPT-DETAIL-LINE = RPT-99-BLANKS
1400
1500       DO PROCESS-INPUT
1600     WHEN NONE
1700       PRODUCE CCFA90R1
1800       RELEASE REPORT CCFA90R1
1900     ENDFOR
2000
2100
2200 :-------------------------------------------------------------------
2300 : PROCESS INPUT CARD IMAGES
2400 :-------------------------------------------------------------------
2500 <<PROCESS-INPUT>> PROCEDURE
2600
2700     SELECT FIRST ACTION
2800     WHEN IN-ACTION = 'L'
2900        DO LIST-FUNCTION
3000     WHEN IN-ACTION = 'U'
3100        DO PHONE-UPDATE
3200     WHEN OTHER
3300        MOVE '0681' TO EN-MSGCODE
3400        CALL CCFA95VI USING EN-MSGCODE EN-TEXT
3500        MOVE ERROR-MSG-LINE TO RPT-DETAIL-LINE
3600        PRODUCE CCFA90R1
3700        RELEASE REPORT CCFA90R1
3800     ENDSEL
3900
4000 ENDPROC    :PROCESS-INPUT PROC
4100
4200 :-------------------------------------------------------------------
4300 : LIST EMPLOYEES
4400 :-------------------------------------------------------------------
4500
4600 <<LIST-FUNCTION>> PROCEDURE
4700     :NO LAST NAME GIVEN
4800     IF IN-LNAME = $SPACES
4900        MOVE '%' TO WK-IN-LNAME
5000     ELSE
5100        :NULL
5200     ENDIF
5300
```

```
PGM  CCFA95BI   001   TEST    SYS:$IO    FEBRUARY 25, 1987    18:29:01

PROCEDURE:    CCFA95BI          VERSION:  001  STATUS:  TEST

SEQ  STATEMENT
5400    :NO FIRST NAME GIVEN
5500    IF IN-FNAME = $SPACES
5600       MOVE '%' TO WK-IN-FNAME
5700    ELSE
5800       :NULL
5900    ENDIF
6000
6100    :LIST ALL EMPLOYEES
6200    IF IN-LNAME = '*'
6300      DO LIST-ALL
6400    ELSE
6500      :TRIM TRAILING BLANKS FROM LAST NAME
6600      SET WK-PERCENT-POS       = $LENGTH(IN-LNAME)
6800      SET WK-IN-LNAME = $TRIM(IN-LNAME,RIGHT=' ')
6900      SET WK-PERCENT-POS       = $LENGTH(WK-IN-LNAME)
7100
7200      :TRIM TRAILING BLANKS FROM FIRST NAME
7300      SET WK-IN-FNAME = $TRIM(IN-FNAME,RIGHT=' ')
7400
7600      :COUNT %'S, ALL WE NEED IS 1 TO KNOW WHAT TO DO
7700       MOVE 0 TO WK-PERCENT-POS
7800       SET WK-PERCENT-POS = $INDEX(IN-LNAME,SEARCH='%')
8000
8100       IF WK-PERCENT-POS > 0
8200             :IF NO %'S, LIST SPECIFIC NAMES
8300             :ELSE LIST GENERIC NAMES
8400         DO LIST-GENERIC
8500       ELSE
8600         DO LIST-SPECIFIC
8700       ENDIF
8800    ENDIF
8900 ENDPROC   :LIST-FUNCTION
9000
9100 :-------------------------------------------------------------------
9200 : LIST ALL EMPLOYEES
9300 :-------------------------------------------------------------------
9400 <<LIST-ALL>> PROCEDURE
9500
9600    FOR EACH MS-EMPLOYEE-2 NO UPDATE
9700          ORDERED BY WORKDEPT ENPNO
9800       SET RPT-LNAME          = LASTNAME
9900       SET RPT-FNAME          = FIRSTNME
10000      SET RPT-MIDDLEINIT     = MIDINIT
10100      SET RPT-PHONENO        = PHONENO
10200      SET RPT-ENPNO          = EMPNO
10300      SET RPT-WORKDEPT       = WORKDEPT
10400
```

```
PGM  CCFA95DI   001   TEST    SYS:DIO    FEBRUARY 25, 1987    18:29:01

PROCEDURE:    CCFA95DI          VERSION:  001  STATUS:  TEST

------ ------------------------------------------------------------------------
 SEQ  STATEMENT
------ ------------------------------------------------------------------------
10500          FOR EACH MS-DEPARTMENT-2 NO UPDATE
10600              WHERE MS-DEPARTMENT-2.DEPTNO =
10700                    MS-EMPLOYEE-2.WORKDEPT
10800            SET RPT-DEPTNAME = DEPTNAME
10900            PRODUCE CCFA90R1
11000          WHEN NONE
11100            SET RPT-DETAIL-LINE =
11200                'NO DEPARTMENT RECORDS FOUND - CONTACT PROGRAMMER'
11300            PRODUCE CCFA90R1
11400          ENDFOR
11500
11600      WHEN NONE
11700          MOVE '0008I' TO EM-MSGCODE
11800          CALL CCFA95VI USING EM-MSGCODE EM-TEXT
11900          MOVE ERROR-MSG-LINE TO RPT-DETAIL-LINE
12000          PRODUCE CCFA90R1
12100      ENDFOR
12200
12300      RELEASE REPORT CCFA90R1
12400  ENDPROC
12500
12600  *-------------------------------------------------------------------
12700  * LIST EMPLOYEES GIVEN GENERIC KEY
12800  *-------------------------------------------------------------------
12900
13000  <<LIST-GENERIC>> PROCEDURE
13100      *FIND OUT IF HAVE LEADING OR TRAILING %
13200
13300      IF WK-PERCENT-POS > 1
13400          DO NAME-BEGIN-WITH
13500      ELSE
13600          DO NAME-END-WITH
13700      ENDIF
13800
13900  ENDPROC
14000
14100
14200  *-------------------------------------------------------------------
14300  * LIST EMPLOYEES WHOSE LASTNAME BEGINS WITH...
14400  *-------------------------------------------------------------------
14500
14600  <<NAME-BEGIN-WITH>> PROCEDURE
14700    :IF  JOX
14800    SET WK-IN-LNAME-LEN = $LENGTH(WK-IN-LNAME) - 1
14900    SET WK-IN-LNAME = $SUBSTR(WK-IN-LNAME,START=1,
15000                              LENGTH=WK-IN-LNAME-LEN)
15300
```

```
PGM  CCFA9581   001   TEST    SYS:$10    FEBRUARY 25, 1987    18:29:01

PROCEDURE:   CCFA9581          VERSION:  001  STATUS:  TEST

 SEQ  STATEMENT
15400   FOR EACH MS-EMPLOYEE-2 NO UPDATE
15500 :       WHERE MS-EMPLOYEE-2.LASTNAME >= WK-IN-LNAME
15600         ORDERED BY WORKDEPT EMPNO
15700
15800     IF $SUBSTR(LASTNAME,START=1,LENGTH=WK-IN-LNAME-LEN) = WK-IN-LNAME
15900       SET RPT-LNAME          = LASTNAME
16000       SET RPT-FNAME          = FIRSTNME
16100       SET RPT-MIDDLEINIT     = MIDINIT
16200       SET RPT-PHONENO        = PHONENO
16300       SET RPT-EMPNO          = EMPNO
16400       SET RPT-WORKDEPT       = WORKDEPT
16500
16600       FOR FIRST MS-DEPARTMENT-2 NO UPDATE
16700             WHERE MS-DEPARTMENT-2.DEPTNO =
16800                   MS-EMPLOYEE-2.WORKDEPT
16900
17000         SET RPT-DEPTNAME = DEPTNAME
17100         PRODUCE CCFA90R1
17200       WHEN NONE
17300         SET RPT-DETAIL-LINE =
17400             'NO DEPARTMENT RECORDS FOUND - CONTACT PROGRAMMER'
17500         PRODUCE CCFA90R1
17600       ENDFOR   :FOR EACH MS-DEPARTMENT-2
17700
17800     ENDIF
17900       WHEN NONE
18000         MOVE '0091' TO EM-MSGCODE
18100         CALL CCFA95V1 USING EM-MSGCODE EM-TEXT
18200         MOVE ERROR-MSG-LINE TO RPT-DETAIL-LINE
18300         PRODUCE CCFA90R1
18400   ENDFOR    :FOR EACH MS-EMPLOYEE-2
18500
18600   RELEASE REPORT CCFA90R1
18700 ENDPROC    :NAME-BEGIN-WITH
18800
18900 :-------------------------------------------------------------------
19000 : LIST EMPLOYEES WHOSE LASTNAME ENDS WITH ...
19100 :-------------------------------------------------------------------
19200
19300 <<NAME-END-WITH>> PROCEDURE
19400
19700   SET WK-IN-LNAME-LEN = $LENGTH(WK-IN-LNAME) - 1
19900   SET WK-IN-LNAME     = $SUBSTR(WK-IN-LNAME,START=2,
20000                                 LENGTH=WK-IN-LNAME-LEN)
20100
20200   FOR EACH MS-EMPLOYEE-2 NO UPDATE
20300   :     WHERE MS-EMPLOYEE-2.LASTNAME >= WK-IN-LNAME
```

```
PGM  CCFA958I   001   TEST    SYS:$IO    FEBRUARY 25, 1987    18:29:01

PROCEDURE:    CCFA958I          VERSION:  001  STATUS:  TEST

 SEQ  STATEMENT
------ ------------------------------------------------------------------------
20400       ORDERED BY WORKDEPT EMPNO
20800     SET WK-BLANK-POS-I  = $INDEX(LASTNAME,SEARCH=' ')
21000     SET WK-START-POS    = WK-BLANK-POS-1  -  3
21300     IF $SUBSTR(MS-EMPLOYEE-2.LASTNAME,START=WK-START-POS,
21400                LENGTH=3) = WK-IN-LNAME
21500
21600       SET RPT-LNAME           = LASTNAME
21700       SET RPT-FNAME           = FIRSTNME
21800       SET RPT-MIDDLEINIT      = MIDINIT
21900       SET RPT-PHONENO         = PHONENO
22000       SET RPT-EMPNO           = EMPNO
22100       SET RPT-WORKDEPT        = WORKDEPT
22200
22300       FOR EACH MS-DEPARTMENT-2 NO UPDATE
22400           WHERE MS-DEPARTMENT-2.DEPTNO =
22500                 MS-EMPLOYEE-2.WORKDEPT
22600         SET RPT-DEPTNAME = DEPTNAME
22700         PRODUCE CCFA90R1
22800       WHEN NONE
22900         SET RPT-DETAIL-LINE =
23000             'NO DEPARTMENT RECORDS FOUND - CONTACT PROGRAMMER'
23100         PRODUCE CCFA90R1
23200       ENDFOR   :FOR EACH MS-DEPARTMENT-2
23300     ENDIF
23400
23500   WHEN NONE
23600     MOVE '0081' TO SM-MSGCODE
23700     CALL CCFA95VI USING EM-MSGCODE SM-TEXT
23800     MOVE ERROR-MSG-LINE TO RPT-DETAIL-LINE
23900     PRODUCE CCFA90R1
24000
24100   ENDFOR    :FOR EACH MS-EMPLOYEE-2
24200
24300   RELEASE REPORT CCFA90R1
24500 ENDPROC  :NAME-END-WITH
24600
24700 :-------------------------------------------------------------------------
24800 : LIST EMPLOYEES MATCHING SPECIFIC NAME PROVIDED
24900 :-------------------------------------------------------------------------
25000 <<LIST-SPECIFIC>> PROCEDURE
25001
25004   IF $SUBSTR(WK-IN-FNAME,START=1,LENGTH=1) = '%' OR $SPACES
25005     :USE JUST THE LAST NAME, NOT THE FIRST
25006
25007     FOR EACH MS-EMPLOYEE-2 NO UPDATE
25008         ORDERED BY WORKDEPT EMPNO
25009         WHERE MS-EMPLOYEE-2.LASTNAME = WK-IN-LNAME
```

PROCEDURE: CCFA958I VERSION: 001 STATUS: TEST

```
SEQ  STATEMENT
-----------------------------------------------------------------------------
25010
25011       SET RPT-LNAME              = LASTNAME
25012       SET RPT-FNAME              = FIRSTNME
25013       SET RPT-NIOOLEINIT         = MIOINIT
25014       SET RPT-PHONENO            = PHONENO
25015       SET RPT-ENPNO              = EMPNO
25016       SET RPT-WORKDEPT           = WORKDEPT
25017
25018       FOR EACH NS-DEPARTMENT-2 NO UPDATE
25019           WHERE NS-DEPARTMENT-2.DEPTNO =
25020                 MS-EMPLOYEE-2.WORKDEPT
25021         SET RPT-DEPTNAME = DEPTNAME
25022         PRODUCE CCFA90RI
25023       WHEN NONE
25024         SET RPT-DETAIL-LINE =
25025             'NO DEPARTMENT RECORDS FOUND - CONTACT PROGRAMMER'
25026         PRODUCE CCFA90RI
25027
25028       ENDFOR   :FOR EACH NS-DEPARTMENT-2
25029
25030     WHEN NONE
25031       MOVE 'DDBI' TO EN-NSGCODE
25032       CALL CCFA95VI USING EN-NSGCODE EM-TEXT
25033       MOVE ERROR-MSG-LINE TO RPT-DETAIL-LINE
25034       PRODUCE CCFA90RI
25035     ENDFOR   :FOR EACH NS-ENPLOYEE-2
25036
25100
25101   ELSE
25200   FOR EACH NS-EMPLOYEE-2 NO UPDATE
25300       ORDERED BY WORKDEPT ENPNO
25400       WHERE NS-ENPLOYEE-2.LASTNAME = IN-LNAME AND
25500             MS-ENPLOYEE-2.FIRSTNME = IN-FNAME
25600
25700     SET RPT-LNAME              = LASTNAME
25800     SET RPT-FNAME              = FIRSTNME
25900     SET RPT-NIOOLEINIT         = MIOINIT
26000     SET RPT-PHONENO            = PHONENO
26100     SET RPT-EMPNO              = EMPNO
26200     SET RPT-WORKDEPT           = WORKDEPT
26300
26400     FOR EACH MS-DEPARTMENT-2 NO UPDATE
26500         WHERE MS-DEPARTNENT-2.DEPTNO =
26600               MS-ENPLOYEE-2.WORKDEPT
26700       SET RPT-DEPTNAME = DEPTNAME
26800       PRODUCE CCFA90RI
26900     WHEN NONE
```

```
PGM  CCFA958I   001   TEST    SYS:$IO    FEBRUARY 25, 1987    18:29:01

PROCEDURE:    CCFA958I          VERSION:  001  STATUS:  TEST

 SEQ  STATEMENT
------ ------------------------------------------------------------------------
27000       SET RPT-DETAIL-LINE =
27100           'NO DEPARTMENT RECORDS FOUND - CONTACT PROGRAMMER'
27200       PRODUCE CCFA90R1
27300
27400     ENDFOR    :FOR EACH MS-DEPARTMENT-2
27500
27600     WHEN NONE
27700       MOVE '0081' TO EM-MSGCODE
27800       CALL CCFA95VI USING EM-MSGCODE EM-TEXT
27900       MOVE ERROR-MSG-LINE TO RPT-DETAIL-LINE
28000       PRODUCE CCFA90R1
28100   ENDFOR    :FOR EACH MS-EMPLOYEE-2
28101   ENDIF
28200
28300   RELEASE REPORT CCFA90R1
28400 ENDPROC   :LIST-SPECIFIC
28500
28600  :-------------------------------------------------------------------
28700  : UPDATE EMPLOYEE PHONE NUMBERS
28800  :-------------------------------------------------------------------
28900
29000 <<PHONE-UPDATE>> PROCEDURE
29100
29200   FOR EACH MS-EMPLOYEE-2
29300         ORDERED BY WORKDEPT EMPNO
29400           WHERE MS-EMPLOYEE-2.EMPNO = IN-EMPNO
29500       SET RPT-PHONENO          = '4265'
29600       MOVE '0041' TO EM-MSGCODE
29700       CALL CCFA95VI USING EM-MSGCODE EM-TEXT
29800       MOVE ERROR-MSG-LINE TO RPT-DETAIL-LINE
29900       PRODUCE CCFA90R1
30000   WHEN NONE
30100       MOVE '0681' TO EM-MSGCODE
30200       CALL CCFA95VI USING EM-MSGCODE EM-TEXT
30300       MOVE ERROR-MSG-LINE TO RPT-DETAIL-LINE
30400       PRODUCE CCFA90R1
30500   ENDFOR
30600
30700   RELEASE REPORT CCFA90R1
30800
30900 ENDPROC    :PHONE-UPDATE
31000
31100 :-------------------------------------------------------------------------
31200 : ERROR PROCEDURE TRAPS AND PRINTS ERRORS NOT TRAPPED ABOVE
31300 :-------------------------------------------------------------------------
31400  <<ERROR>> PROCEDURE
31500     CALL CCFA99VI USING INPUT "IO"
```

```
PGM  CCFA958I   001   TEST    SYS:$IO    FEBRUARY 25, 1987    18:29:01

PROCEDURE:    CCFA958I          VERSION:  001  STATUS:  TEST

      SEQ  STATEMENT
-----------------------------------------------------------------------
31600       QUIT RUN
31700  ENDPROC   !ERROR
31800
```

```
PGM  CCFA959I   001   TEST     SYS:$ID    FEBRUARY 25, 1987    18:29:01

                                        COMPILER  DIAGNOSTICS  AND  SUMMARY

PROGRAM HAS BEEN SUCCESSFULLY COMPILED

NO ERROR MESSAGE(S) FLAGGED IN THIS COMPILATION
NO WARNING MESSAGE(S) FLAGGED IN THIS COMPILATION
NO ADVISORY MESSAGE(S) FLAGGED IN THIS COMPILATION

COMPILED PROGRAM STATISTICS:

- NUMBER OF SHAREABLE      MEMBERS  OF THE PROGRAM =          001
- LENGTH OF SHAREABLE      PORTIONS OF THE PROGRAM =        7,012  BYTES
- NUMBER OF NON-SHAREABLE MEMBERS  OF THE PROGRAM =          001
- LENGTH OF NON-SHAREABLE PORTIONS OF THE PROGRAM =        5,812  BYTES
```

```
                          SELECTING AN EMPLOYEE TO DISPLAY
MAJOR SYSTEM .....: O             ORGANIZATION
ACTION ...........: D             DISPLAY (SHOW)
OBJECT ...........: EM            EMEPLOYEE
SEARCH CRITERIA ..: EN            EMPLOYEE NAME
DATA .............: %

NO   D/ID   DEPARTMENT NAME                          E/ID     EMPLOYEE NAME

01   A00    SPIFFY COMPUTER SERVICE DIV.            000010   CI HASS
02   B01    PLANNING                                000020   ML THOMPSON
03   C01    INFORMATION CENTER                      000030   SA KWAN
04   E01    SUPPORT SERVICES                        000050   JB GEYER
05   D11    MANUFACTURING                           000060   IF STERN
06   D21    ADMINISTRATION SYSTEMS                  000070   ED PULASKI
07   E11    OPERATIONS                              000090   EW HENDERSON
08   E21    SOFTWARE SUPPORT                        000100   TQ SPENSER
09   A00    SPIFFY COMPUTER SERVICE DIV.            000110   VG LUCCHESI
10   A00    SPIFFY COMPUTER SERVICE DIV.            000120   S  O'CONNELL
11   C01    INFORMATION CENTER                      000130   DM QUINTANA

PFK: 02=RESEND  03=END  08=NEXT
```

```
                          EMPLOYEE ADD
MAJOR SYSTEM .....: O
ACTION ...........: A
OBJECT ...........: EM
SEARCH CRITERIA ..: EN
DATA .............: 000099

    EMPLOYEE    ID              :  000099
                FIRST NAME      :  ____________
                MIDDLE INITIAL  :  _
                LAST NAME       :  ___________
                WORK DEPT ID    :  ___
                PHONE NUMBER    :  ____


PFK: 02=RESEND  03=END  08=NEXT
```

```
                                    EMPLOYEE ERASE
MAJOR SYSTEM .....: O
ACTION ...........: E
OBJECT ...........: EM
SEARCH CRITERIA ..: EN
DATA .............: 000030

    EMPLOYEE    ID                :   000030
                FIRST NAME        :   JANE
                MIDDLE INITIAL    :   E
                LAST NAME         :   DOE
                WORK DEPT ID      :   E21
                PHONE NUMBER      :   0000



PFK: 02=RESEND  03=END


                                    EMPLOYEE UPDATE
MAJOR SYSTEM .....: O
ACTION ...........: U
OBJECT ...........: EM
SEARCH CRITERIA ..: EN
DATA .............: 000030

    EMPLOYEE    ID                :   000030
                FIRST NAME        :   JANE__________
                MIDDLE INITIAL    :   E
                LAST NAME         :   DOE_________
                WORK DEPT ID      :   E21
                PHONE NUMBER      :   0000
```

ADR - Online program screens

```
PGM  CCFA92CI   001   TEST    SYS:$ID    MARCH 4, 1987        18:44:00

IDENTIFICATION: CCFA92CI        VERSION: 001 STATUS:  TEST

                PROGRAM CCFA92CI

                CREATED                 02/20/87                BY GARTEN
                LAST MODIFIED   03/04/87 AT 18:43        BY GARTEN
                LAST COMPILED   03/04/87 AT 18:27
                RUN STATUS:             PRIVATE
                SHORT DESCRIPTION:
                LANGUAGE:  IDEAL        TARGET DATE __ __ __    ACTUAL DATE __ __ __
                DESCRIPTION:
```

```
PGM  CCFA92CI   001   TEST    SYS:$IO    MARCH 4, 1987        18:44:00

RESOURCES:      CCFA92CI          VERSION: 001 STATUS:  TEST
                ---------------- ----- -------- ---- -------- ---- -------- ---- ------
                DATAVIEW         VER   PANEL    VER  REPORT   VER  PRDGRAM  VER  SYS
                ---------------- ----- -------- ---- -------- ---- -------- ---- ------

                MS-DEPARTMENT-U  0002  FA9201I  0001 --------  ---  CCFA99VI 0005  ---
                MS-EMPLOYEE-2-U  0008  -------- ---- -------- ---- -------- ---- ---
```

```
PGM  CCFA97CI    OD1    TEST     SYS:SID     MARCH 4, 1987          18:44:DD

DATAVIEW:        MS-DEPARTMENT-U VERSION: DD2 STATUS:  PROD
                   DATAVIEW: MS-DEPARTMENT-U
                   --- ----- ---------------------- - - ----- ----- - ----------------------
                   SEQ LEVEL FIELD NAME             T I CH/DG OCCUR K VALUE/REDEF/DEP ON
                   --- ----- ---------------------- - - ----- ----- - ----------------------
                   CATALOGED 02/16/87 14:02     DATACOM/DB UPD=YES DBID=DID
                     1 1       MS-DEPARTMENT-U
                     2  2      ADMRDEPT         X       3         K
                     3  2      DEPTNAME         X      36
                     4  2      DEPTNO           X       3         K
                     5  2      MGRNO            X       6         K

                 LEGEND:
                   SEQ=SEQUENCE NUMBER
                   T (FIELD TYPE):  X=ALPHANUMERIC, N=NUMERIC, U=UNSIGNED , C=COND. NAME,
                   V=VARIABLE, D=DATE
                   I (INTERNAL NUMERIC TYPE):  P=PACKED, Z=ZONED, B=BINARY
                   CH/DG (CHARACTERS/DIGITS):  NUMBER OF CHARACTERS, INTEGERS, OR INT.DEC
                   K (KEY USAGE):  K=WHOLE KEY, P=PARTIAL KEY (HIGH ORDER POSITION)
                   REDEF=REDEFINITION, DEP ON=DEPENDING ON
```

```
PGM  CCFA92CI   001   TEST    SYS:$IO    MARCH 4, 1987        18:44:00

DATAVIEW:       MS-ENPLOYEE-2-U VERSION: 008 STATUS:  PROD
                  DATAVIEW: NS-ENPLOYEE-2-U
                  --- ----- -------------------  - - ------ ----- - ---------------------
                  SEQ LEVEL FIELD NAME           T I CH/DG OCCUR K VALUE/REDEF/DEP ON
                  --- ----- -------------------  - - ------ ----- - ---------------------
                  CATALOGED 02/20/87 08:27      DATACON/DB UPD=YES DBID=010
                    1 1       MS-ENPLOYEE-2-U
                    2  2      WORKDEPT           X      3        K
                    3  2      ENPNO              X      6        K
                    4  2      FULLNAME                           P
                    5   3       LASTNAME         X     12        P
                    6   3       FIRSTNME         X     15
                    7   3       MIDINIT          X      1
                    8  2      PHONENO            X      4

              LEGEND:
                  SEQ=SEQUENCE NUMBER
                  T (FIELD TYPE):  X=ALPHANUNERIC, N=NUNERIC, U=UNSIGNED , C=COND. NANE,
                  V=VARIABLE, D=DATE
                  I (INTERNAL NUNERIC TYPE):  P=PACKED, Z=ZONED, B=BINARY
                  CH/DG (CHARACTERS/DIGITS):  NUMBER OF CHARACTERS, INTEGERS, OR INT.DEC
                  K (KEY USAGE):  K=WHOLE KEY, P=PARTIAL KEY (HIGH ORDER POSITION)
                  REDEF=REDEFINITION, DEP ON=DEPENDING ON
```

```
PGM  CCFA92CI   001   TEST    SYS:510    MARCH 4, 1987          18:44:00

PANEL:          FA9201I          VERSION: 001 STATUS:  TEST

          IDENTIFICATION:
                    PANEL NAME FA9201I

                    CREATED              02/21/87                    BY GARTEN
                    LAST MODIFIED        03/04/87 AT 18:21           BY GARTEN

                    RUN STATUS PRIVATE
                    SHORT DESCRIPTION

                    DESCRIPTION:
```

```
PGM  CCFA92CI   001   TEST    SYS:$ID    MARCH 4, 1987        18:44:00

PANEL:          FA92011I          VERSION: 001 STATUS:  TEST

  FACSIMILE:
  |....+....1....+....2....+....3....+....4....+....5....+....6....+....7....+....8|
  | SELECTING AN EMPLOYEE TO DISPLAY                                               |
  | MAJOR SYSTEM . . .:                                                            |
  | ACTION . . . . . .:                                                            |
  | OBJECT . . . . . .:                                                            |
  | SEARCH CRITERIA. .:                                                            |
  | DATA . . . . . .:                                                              |
  |                                                                                |
  |                                                                                |
  |                                                                                |
  | PFK: 02=RESEND 03=END                                                          |
  |....+....1....+....2....+....3....+....4....+....5....+....6....+....7....+....8|
```

```
PGM  CC=A92CI    001   TEST    SYS:&ID     MARCH 4, 1987        18:44:00

PANEL:          FA9201I         VERSION: 001 STATUS:  TEST

                LAYOUT:
                |....+....1....+....2....+....3....+....4....+....5....+....6....+....T....+....8|
                |¬SELECTING AN EMPLOYEE TO DISPLAY;                                              |
                |1                                                                               |
                |¬MAJOR SYSTEM . . .:¬ ; ¬                                                       :|
                |2                   3   4                                                       |
                |¬ACTION . . . . . .:¬ ; ¬                                                       ;|
                |5                   6   7                                                       |
                |¬OBJECT . . . . . .:¬  ; ¬                                                      ;|
                |8                   9   10                                                      |
                |¬SEARCH CRITERIA. .:¬  ; ¬                                                      ;|
                |11                  12  13                                                      |
                |¬DATA . . . . . .:¬                                                             ;|
                |14                  15                                                          |
                |¬                                                                               ;|
                |16                                                                              |
                |                                                                                |
                |¬                                                                               ;|
                |17                                                                              |
                |a                                                a_____________________________:|
                |19                                               20                             |
                |                                                                                |
                |¬PFK: 02=RESEND 03=END¬                                                         ;|
                |21                  22                                                          |
                |....+....1....+....2....+....3....+....4....+....5....+....6....+....T....+....8|
```

```
PGM  CCFA92CI   001   TEST    SYS:#IO    MARCH 4, 1987        18:44:00

PANEL:          FA9201I          VERSION: 001 STATUS:  TEST

        FIELD SUMMARY TABLE:

                SEQ LV FIELD NAME        ATTR    T LEN  IN.OP OCC COMMENTS
                1  2                     PSL     X  32            SELECTING AN EMPLOYEE
                2  2                     PSL     X  19            MAJOR SYSTEM . . .:
                3  2  MAJSYS            UALE    X   1
                4  2  MAJSYS-MSG        PSL     X  52
                5  2                     PSL     X  19            ACTION . . . . . .:
                6  2  ACTION            UALE    X   1
                7  2  ACTION-MSG        PSL     X  52
                8  2                     PSL     X  19            OBJECT . . . . . .:
                9  2  OBJECT            UALE    X   2
                10 2  OBJECT-MSG        PSL     X  52
                11 2                     PSL     X  19            SEARCH CRITERIA. .:
                12 2  SEARCH            UALE    X   2
                13 2  SEARCH-MSG        PSL     X  52
                14 2                     PSL     X  19            DATA . . . . . .:
                15 2  USER-INPUT        UALE    X  58
                16 2  MSGLINE           PSL     X  78
                17 2  COLUMN-HEADS      PSH     X  78
                18 2                     G       X
                19  3 POL-HEADER        PSL     X  38
                20  3 POL-INPUT         UAL     X  39
                21 2                     PSL     X  21            PFK: 02=RESEND 03=END
                22 2  PFK-MSG           PSL     X  55

        LEGEND:
                SEQ=SEQUENCE NUMBER
                ATTR=SCREEN ATTRIBUTES: U=UNPROT   H=HIGHLIGHT   A=327X ALPHANUMERIC
                                        P=PROT     I=INVISIBLE   N=327X NUMERIC
                                        S=SKIP     L=LOW-LIGHT   E=ENSURE INPUT
                                                                 C=CURSOR
                T=FIELD TYPE: X=ALPHANUMERIC, N=NUMERIC
                IN.DP=INTEGER-PLACES.DECIMAL-PLACES
                OCC=NUMBER OF OCCURRENCES

        PANEL PARAMETERS:

                START      FIELD SYMBOL ¬       NEW    FIELD SYMBOL +
                END        FIELD SYMBOL ;       DELETE FIELD SYMBOL *
                REPEATING GROUP SYMBOL @

                INPUT  FILL CHARACTER  $        (S=SPACE, L=LOWVAL, Z=ZEROS, U=_, OTHER=ITSELF)
                OUTPUT FILL CHARACTER  U        (S=SPACE, L=LOWVAL, U=_, OTHER=ITSELF)
                NON-DISPLAY CHARACTER  S        (S=SPACE, OTHER=AS SPECIFIED)
                ERROR FILL CHARACTER   *        (AS SPECIFIED)
                CASE TRANSLATION       U        (U=UPPER, M=MIXED)

                REQUIRED               N        (Y=YES, N=NO)
```

```
PGM  CCFA92CI   001   TEST    SYS:=IO    MARCH 4, 1987        18:44:00

PANEL:          FA92011         VERSION: 001 STATUS:  TEST

        PANEL PARAMETERS:

                ERROR HANDLING              B       (N=NONE, *=FILL W/ERRORFILL, H=HIGH INTENSITY,)
                                                    (B=BOTH: H IF ILLEGAL VALUE & * IF RQD MISSING)

                PF1=HELP, PF3=CLARIFY       Y       (Y=YES, N=NO)
                PF7=SCR -, PF8=SCR +        N       (Y=YES, N=NO)
                 PF10=SCR TOP, PF11=SCR BOT

                EDIT-RULE ERROR PROC        C       (C=CLARIFY COMMAND, A=APPLICATION)
                PROCESS APPL ON SCROLL      Y       (Y=YES, N=NO)
                HELP PANEL NAME  ________           VERSION ___
                PREFIX PANEL NAME ________          VERSION ___
                SUFFIX PANEL NAME ________          VERSION ___
```

```
PGM  CCFA92CI   001   TEST    SYS:SID    MARCH 4, 1987        18:44:00

PANEL:          FA92011          VERSION: 001 STATUS:  TEST

        INPUT RULES:

                  SEQ FIELD NAME          E R MINIMUM          MAXIMUM          J I C MN D N A C N
                                          H Q VALUE            VALUE            S F S OP S S C O F
                  --- ------------------- - - ---------------- ---------------- - - - -- - - - - -
                    1                     B N                                   L S U              N
                    2                     B N                                   L S U              N
                    3 NAJSYS              B N                                   L S U              N
                    4 NAJSYS-NSG          B N                                   L S U              N
                    5                     B N                                   L S U              N
                    6 ACTION              B N                                   L S U              N
                    7 ACTION-NSG          B N                                   L S U              N
                    8                     B N                                   L S U              N
                    9 OBJECT              B N                                   L S U              N
                   10 OBJECT-NSG          B N                                   L S U              N
                   11                     B N                                   L S U              N
                   12 SEARCH              B N                                   L S U              N
                   13 SEARCH-MSG          B N                                   L S U              N
                   14                     B N                                   L S U              N
                   15 USER-INPUT          B N                                   L S U              N
                   16 NSGLINE             B N                                   L S U              N
                   17 COLUMN-NEADS        B N                                   L S U              N
                   18
                   19 PDL-HEADER          B N                                   L S U              N
                   20 PDL-INPUT           B N                                   L S U              N
                   21                     B N                                   L S U              N
                   22 PFK-NSG             B N                                   L S U              N

        LEGEND:
                  SEQ=SEQUENCE NUMBER
                  E/N=ERROR HANDLING: N=NONE, *=FILL WITH *, N=HIGH INTENSITY,
                                      B=BOTH (N IF ILLEGAL VALUE, * IF REQUIRED FIELD MISSING)
                  R/Q=REQUIRED FIELD: Y=YES, N=NO, C=CONDITIONAL
                  J/S=JUSTIFICATION: N=NONE, L=LEFT, R=RIGHT, A=ALIGN BY DECIMAL POINT
                  I/F=INPUT FILL CHARACTER:
                        S=SPACES, L=LOW-VALUES, Z=ZEROES, U=UNDERSCORE, OTHER=ITSELF
                  C/S=CASE: U=UPPER CASE, N=NIXED CASE
                  MN/OP=MINIMUM REQUIRED DECIMALS (FOR JUSTIFICATION=A)
                  D/S=ALLOW DIGIT SEPARATOR: Y=YES, N=NO
                  M/S=ALLOW MINUS SIGN:      Y=YES, N=NO
                  A/C=ALLOW CURRENCY SYMBOL: Y=YES, N=NO
                  C/D=CHECK DIGIT: N=NONE, T=MODULO 10, E=MODULO 11
                  M/F=MUST FILL: Y=YES, N=NO
```

```
PGM  CCFA92CI   001   TEST    SYS:$IO    MARCH 4, 1987        18:44:00

PANEL:          FA9201I          VERSION: 001 STATUS:  TEST

        OUTPUT RULES:

                  SEQ FIELD NAME       EDIT PATTERN                    O   E
                                                                       F C H
                  --- ---------------- ------------------------------ - - -
                    1                                                  U N N
                    2                                                  U N N
                    3 MAJSYS                                           S N N
                    4 MAJSYS-MSG                                       S N N
                    5                                                  U N N
                    6 ACTION                                           S N N
                    7 ACTION-MSG                                       S N N
                    8                                                  U N N
                    9 OBJECT                                           S N N
                   10 OBJECT-MSG                                       S N N
                   11                                                  U N N
                   12 SEARCH                                           S N N
                   13 SEARCH-MSG                                       S N N
                   14                                                  U N N
                   15 USER-INPUT                                       S N N
                   16 MSGLINE                                          S N N
                   17 COLUMN-HEADS                                     S N N
                   18
                   19 PDL-HEADER                                       S N N
                   20 PDL-INPUT                                        U N N
                   21                                                  S N N
                   22 PFK-MSG                                          S N N

        LEGEND:
                  SEQ=SEQUENCE NUMBER
                  O/F=OUTPUT FILL CHARACTER:
                       S=SPACES, L=LOW-VALUES, Z=ZEROES, U=UNDERSCORE, OTHER=ITSELF
                  C=COLOR:
                       N=NEUTRAL, B=BLUE, R=RED, P=PINK, G=GREEN
                       T=TURQUOISE, Y=YELLOW, W=WHITE/BLACK
                  E/H=EXTENDED HIGHLIGHTING:
                       N=NONE, B=BLINK, R=REVERSE VIDEO, U=UNDERSCORE
```

```
PGM  CCFA92CI   001   TEST    SYS:$IO    MARCH 4, 1987        18:44:00

WORKING DATA:  CCFA92CI        VERSION: 001 STATUS:  TEST

SEQ     LEVEL  FIELD NAME              T I CH/OG OCCUR VAL/COMMENT/REDEF/DEP ON/COPY
------  -----  -------------------     - - -----  ----- ------------------------------
000101  1      WK-FNAME-BYTE-I         X       1
000200  1      WK-MOLD-INPUT           X      50        '                             '
000201  1      WK-MOLD-NAME            X      50        '                             '
000202  1      WK-MOLD-KEY             X      50        '                             '
000203  1      WK-READ-KEY             X      50        '                             '
000204  1      WK-CTR                  N       4        :VALUE 1-12 COUNTER
000205  1      WK-PNL-CTR              N       4        :NUMERIC  PNL INDEX
000206  1      WK-RESEND-CTR           N       2        :RESEND COUNTER
000207  1      WK-VALID-INPUT          X       1
000208  1      WK-REC-DISP             X       1
000209  1      WK-GOOD-EDIT            X       1        'F'
000210  1      WK-GOOD-COMPARE         X       1
000211  1      WK-SAVE-EMPNO           X       6
000212  1      WK-SAVE-LNAME           X      12
000213  1      WK-SAVE-FNAME           X      15
000214  1      WK-SAVE-MIDINIT         X       1
000215  1      WK-SAVE-PHONENO         X       4
000216  1      WK-SAVE-WKDEPT          X       3
000300
000400  1      WK-DETAIL-LINE
000500  2       WK-DL-NDX-ALPN         X       4
000600  2       WK-FILLER-1            X       2
000700  2       WK-DL-WORKDEPT         X       3
000800  2       WK-FILLER-2            X       3
000900  2       WK-DL-DEPTNAME         X      36
001000  2       WK-FILLER-3            X       2
001100  2       WK-DL-EMPNO            X       6
001200  2       WK-FILLER-4            X       3
001300  2       WK-DL-FULL-NAME        X      18

LEGEND:
T (FIELD TYPE):  X=ALPHANUMERIC, N=NUMERIC, U=UNSIGNED, C=CONO.NAME, F=FLAG,
                 V=VARIABLE, D=DATE
I (INTERNAL NUMERIC TYPE):  BLANK OR P=PACKED, Z=ZONED, B=BINARY
CH/OG (CHARACTERS/DIGITS):  NUMBER OF CHARACTERS, INTEGERS, OR INTEGERS.DECIMALS
REDEF= REDEFINITION, DEP ON= DEPENDING ON
```

```
PGM  CCFA92CI    001   TEST    SYS:IO    MARCH 4, 1987        18:44:00

PROCEDURE:   CCFA92CI          VERSION:  001  STATUS:  TEST

    SEQ  STATEMENT
    ---- ------------------------------------------------------------------
     100 :
     200 <<MAIN>> PROCEDURE
     300    RESET FA9201I
     400    SET WK-HOLD-INPUT     = $SPACES
     500    SET WK-PNL-CTR        = 0
     600    SET WK-CTR            = 0
     700    SET WK-RESEND-CTR     = 0
     800
     900    LOOP
    1000       TRANSMIT FA9201I.REINPUT
    1100
    1200       DO EDIT-4-LINES-IN
    1300       IF WK-VALID-INPUT = 'T'
    1400          SET ATTRIBUTE 'P' ON FA9201I.MAJSYS
    1500          SET ATTRIBUTE 'P' ON FA9201I.ACTION
    1600          SET ATTRIBUTE 'P' ON FA9201I.OBJECT
    1700          SET ATTRIBUTE 'P' ON FA9201I.SEARCH
    1800          DO LOAD-COL-HEADS
    1900          MOVE '08=NEXT' TO PFK-MSG
    2000          MOVE $STRING('SELECT AN EMPLOYEE FROM FOLLOWING LIST BY ',
    2100                       'SPECIFYING THE LINE NUMBER') TO FA9201I.MSGLINE
    2200          SET ATTRIBUTE 'HP' TEMP ON FA9201I.MSGLINE
    2300
    2400          SELECT FIRST ACTION
    2500          WHEN $PF2
    2600             DO RESEND-PNL
    2700          WHEN $PF8
    2800             DO DISPLAY-INDEX
    2900          WHEN OTHER
    3000             SET WK-READ-KEY = FA9201I.USER-INPUT
    3100             DO DISPLAY-INDEX
    3200          ENDSEL
    3300       ELSE
    3400          DO RESEND-PNL
    3500       ENDIF
    3600
    3700       UNTIL $PF3
    3800
    3900    ENDLOOP
    4000
    4100 ENDPROC  :MAIN
    4200
    4300
    4400 <<EDIT-4-LINES-IN>> PROCEDURE
    4500    MOVE 'T' TO WK-VALID-INPUT
    4600
    4700    :VALIDATE USER 'SEARCH CRITERIA' INPUT
```

```
PGM  CCFA92CI   001   TEST    SYS:$IO    MARCH 4, 1987        18:44:00

PROCEDURE:    CCFA92CI          VERSION:  001  STATUS:  TEST

SEQ  STATEMENT
4800    IF FA92011.SEARCH = 'EI'
4900       SET FA92011.SEARCH-MSG = 'EMPLOYEE ID'
5000    ELSE
5100       SET FA92011.SEARCH-MSG = "INVALID SEARCH CRITERIA, USE 'EI'"
5200       SET ATTRIBUTE 'CH' TEMP ON FA92011.SEARCH
5300       MOVE "F" TO WK-VALID-INPUT
5400    ENDIF
5500
5600    :VALIDATE USER 'OBJECT' INPUT
5700    IF FA92011.OBJECT = 'EM'
5800       SET FA92011.OBJECT-MSG = 'EMPLOYEE'
5900    ELSE
6000       SET FA92011.OBJECT-MSG = "INVALID OBJECT, USE 'EM'"
6100       SET ATTRIBUTE 'CH' TEMP ON FA92011.OBJECT
6200       MOVE "F" TO WK-VALID-INPUT
6300    ENDIF
6400
6500    :VALIDATE USER 'ACTION' INPUT
6600    SELECT FIRST ACTION
6700    WHEN FA92011.ACTION = 'A'
6800       DO ADD-REC
6900    WHEN FA92011.ACTION = 'D'
7000       DO DISPLAY-INDEX
7100    WHEN FA92011.ACTION = 'E'
7200       DO ERASE-REC
7300    WHEN FA92011.ACTION = 'U'
7400       DO UPDATE-REC
7500    WHEN OTHER
7600       SET FA92011.MSGLINE = 'SELECT AN ACTION FROM FOLLOWING LIST'
7700       SET FA92011.POL-HEADER(1) = '   A     ADD     (INSERT)'
7800       SET FA92011.POL-HEADER(2) = '   D     DISPLAY (SHOW)  '
7900       SET FA92011.POL-HEADER(3) = '   E     ERASE   (REMOVE)'
8000       SET FA92011.POL-HEADER(4) = '   U     UPDATE  (CHANGE)'
8100       SET ATTRIBUTE 'CH' TEMP ON FA92011.ACTION
8200       SET WK-VALID-INPUT              = 'F'
8300    ENDSEL
8400
8500    :VALIDATE USER 'MAJOR SYSTEM' INPUT
8600    IF FA92011.MAJSYS = 'O'
8700       SET FA92011.MAJSYS-MSG = 'ORGANIZATION'
8800    ELSE
8900       SET FA92011.MAJSYS-MSG = "INVALID MAJOR SYSTEM, USE 'O'"
9000       SET ATTRIBUTE 'CH' TEMP ON FA92011.MAJSYS
9100       MOVE "F" TO WK-VALID-INPUT
9200    ENDIF
9300
9400 ENDPROC
```

```
PGM  CCFA92CI   001   TEST    SYS:ID    MARCH 4, 1987        18:44:00

PROCEDURE:    CCFA92CI          VERSION:  001  STATUS:  TEST

      SEQ  STATEMENT
     ------ ------------------------------------------------------------
      9500
      9600 :-------------------------------------------------------------
      9700 : ERASE DESIGNATED RECORD
      9800 :-------------------------------------------------------------
      9900 <<ERASE-REC>> PROCEDURE
     10000
     10100    FOR THE FIRST MS-EMPLOYEE-2-U
     10200        WHERE MS-EMPLOYEE-2-U.EMPNO = WK-READ-KEY
     10300      SET FA92011.POL-NEADER(1) =
     10400        $STRING('EMPLOYEE    ID          :  ',MS-EMPLOYEE-2-U.EMPNO)
     10500      SET FA92011.POL-HEADER(2) =
     10600        $STRING('            FIRST NAME  :  ',MS-EMPLOYEE-2-U.FIRSTNME)
     10700      SET FA92011.POL-NEADER(3) =
     10800        $STRING('            MIDDLE INIT :  ',MS-EMPLOYEE-2-U.MIDINIT)
     10900      SET FA92011.POL-HEADER(4) =
     11000        $STRING('            LAST NAME   :  ',MS-EMPLOYEE-2-U.LASTNAME)
     11100      SET FA92011.POL-HEADER(5) =
     11200        $STRING('            PHONE NUMBER:  ',MS-EMPLOYEE-2-U.PHONEND)
     11300      SET FA92011.POL-HEADER(6) =
     11400        $STRING('            WORK DEPT   :  ',MS-EMPLOYEE-2-U.WORKDEPT)
     11500      IF $PF9
     11600        DELETE MS-EMPLOYEE-2-U
     11700      ELSE
     11800        :NULL
     11900      ENDIF
     12000
     12100   WHEN NONE
     12200      SET FA92011.MSGLINE = 'EMPLOYEE ID NOT FOUND OF FILE'
     12300   ENDFOR   :MS-EMPLOYEE-2-U
     12400 ENDPROC   :ERASE-REC
     12500
     12600 :-------------------------------------------------------------
     12700 : ADD RECORD
     12800 :-------------------------------------------------------------
     12900 <<ADD-REC>> PROCEDURE
     12901
     13100   FOR NEW MS-EMPLOYEE-2-U
     13300      SET MS-EMPLOYEE-2-U.EMPNO = FA92011.POL-HEADER(1)
     13400      SET MS-EMPLOYEE-2-U.FIRSTNME = FA92011.POL-HEADER(2)
     13500      SET MS-EMPLOYEE-2-U.MIDINIT  = FA92011.POL-NEADER(3)
     13600      SET MS-EMPLOYEE-2-U.LASTNAME  = FA92011.POL-HEADER(4)
     13700      SET MS-EMPLOYEE-2-U.PHONEND   = FA92011.POL-HEADER(5)
     13800      SET MS-EMPLOYEE-2-U.WORKDEPT = FA92011.POL-HEADER(6)
     13900      DO EDIT-INPUT
     14000      IF WK-GOOD-EDIT = 'T'
     14100        :NULL, RECORD WILL BE ADDED
     14200      ELSE
```

```
PGM  CCFA92CI   001   TEST    SYS:$ID    MARCH 4, 1987        13:44:00

PROCEDURE:    CCFA92CI          VERSION:  001  STATUS:  TEST

 SEQ  STATEMENT
14300          SET NSGLINE = 'CORRECT HIGHLIGHTED FIELDS  '
14400        ENDIF
14500      WHEN DUPLICATE
14600        SET NSGLINE = 'EMPLOYEE ID ALREADY EXISTS ON FILE'
14700      ENDFOR
14800 ENDPROC    :ADD-REC
14900
15000 :-------------------------------------------------------------------
15100 : UPDATE RECORD
15200 :-------------------------------------------------------------------
15300 <<UPDATE-REC>> PROCEDURE
15400    IF WK-REC-DISP = 'F'
15500      FOR THE FIRST NS-EMPLOYEE-2-U NO_UPDATE
15600            WHERE NS-EMPLOYEE-2-U.EMPNO = WK-READ-KEY
15700
15800        SET FA92011.POL-HEADER(1) =
15900          $STRING('EMPLOYEE    ID          :  ',NS-EMPLOYEE-2-U.EMPNO)
16000        SET FA92011.POL-HEADER(2) =
16100          $STRING('                FIRST NAME   :  ',NS-EMPLOYEE-2-U.FIRSTNME)
16200        SET FA92011.POL-HEADER(3) =
16300          $STRING('                MIDDLE INIT  :  ',NS-EMPLOYEE-2-U.MIDINIT)
16400        SET FA92011.POL-HEADER(4) =
16500          $STRING('                LAST NAME    :  ',NS-EMPLOYEE-2-U.LASTNAME)
16600        SET FA92011.POL-HEADER(5) =
16700          $STRING('              PHONE NUMBER :  ',NS-EMPLOYEE-2-U.PHONENO)
16800        SET FA92011.POL-HEADER(6) =
16900          $STRING('              WORK DEPT    :  ',NS-EMPLOYEE-2-U.WORKDEPT)
17000        SET WK-SAVE-EMPNO     = NS-EMPLOYEE-2-U.EMPNO
17100        SET WK-SAVE-FNAME     = NS-EMPLOYEE-2-U.FIRSTNME
17200        SET WK-SAVE-MIDINIT   = NS-EMPLOYEE-2-U.MIDINIT
17300        SET WK-SAVE-LNAME     = NS-EMPLOYEE-2-U.LASTNAME
17400        SET WK-SAVE-PHONENO   = NS-EMPLOYEE-2-U.PHONENO
17500        SET WK-SAVE-WKDEPT    = NS-EMPLOYEE-2-U.WORKDEPT
17600      WHEN NONE
17700        SET FA92011.MSGLINE = 'EMPLOYEE ID NOT FOUND FOR UPDATE'
17800        SET WK-REC-DISP = 'F'
17900      ENDFOR
18000
18100    ELSE
18200      :RECORD FOUND AND DISPLAYED FOR USER
18300      SET WK-GOOD-EDIT = 'T'
18400      DO EDIT-INPUT
18500      IF WK-GOOD-EDIT = 'T'
18600         SET WK-GOOD-COMPARE = 'T'
18700         DO COMPARE-RECS
18800      ELSE
18900        SET FA92011.MSGLINE = $STRING('CORRECT HIGHLIGHTED FIELDS',
```

PROCEDURE: CCFA92CI VERSION: 001 STATUS: TEST

```
 SEQ  STATEMENT
19000                MSGLINE)
19100          ENDIF
19200       ENDIF
19300 ENDPROC   :UPDATE-REC
19400
19500 :-------------------------------------------------------------------
19600 : EDIT USER INPUT
19700 :-------------------------------------------------------------------
19800 <<EDIT-INPUT>> PROCEDURE
19900    IF FA92011.PDL-INPUT(4) = $SPACES
20000       SET MSGLINE = $SPACES
20100    ELSE
20200       SET MSGLINE = 'LAST NAME NOT SUPPLIED'
20300       SET WK-GOOD-EDIT = 'F'
20400    ENDIF
20500
20600    IF FA92011.PDL-INPUT(5)  > $SPACES
20700      IF $NUMERIC(FA92011.PDL-INPUT(5))
20800         SET MSGLINE = $SPACES
20900      ELSE
21000         SET WK-GOOD-EDIT = 'F'
21100         SET MSGLINE = 'PHONE NUMBER NOT NUMERIC'
21200      ENDIF
21300    ELSE
21400      :NULL
21500    ENDIF
21600 ENDPROC   :EDIT-INPUT
21700
21800 :-------------------------------------------------------------------
21900 : COMPARE COPIES OF RECORD TO SEE IF UPDATED SINCE SESSION BEGAN
22000 :-------------------------------------------------------------------
22100 <<COMPARE-RECS>> PROCEDURE
22200    SET WK-GOOD-COMPARE = 'T'
22300    <<COMPARE-LOOP>>
22400    FOR THE FIRST NS-EMPLOYEE-2-U
22500          WHERE NS-EMPLOYEE-2-U.EMPNO = WK-READ-KEY
22600
22700     IF WK-SAVE-ENPNO = FA92011.PDL-INPUT(1)
22800        :NULL
22900     ELSE
23000        SET WK-GOOD-COMPARE = 'F'
23100     ENDIF
23200     IF WK-SAVE-FNAME = FA92011.PDL-INPUT(2)
23300        :NULL
23400     ELSE
23500        SET WK-GOOD-COMPARE = 'F'
23600     ENDIF
```

```
PGM  CCFA92CI   001   TEST    SYS:$ID    MARCH 4, 1987        18:44:00

PROCEDURE:   CCFA92CI          VERSION:  001  STATUS:  TEST

      SEQ  STATEMENT
    23700     IF WK-SAVE-MIOINIT = FA92011.POL-INPUT(3)
    23800        :NULL
    23900     ELSE
    24000        SET WK-GOOD-COMPARE = 'F'
    24100     ENDIF
    24200     IF WK-SAVE-LNAME = FA92011.POL-INPUT(4)
    24300        :NULL
    24400     ELSE
    24500        SET WK-GOOD-COMPARE ='F'
    24600     ENDIF
    24700     IF WK-SAVE-PHONENO = FA92011.POL-INPUT(5)
    24800        :NULL
    24900     ELSE
    25000        SET WK-GOOD-COMPARE = 'F'
    25100     ENDIF
    25200     IF WK-SAVE-WKDEPT = FA92011.POL-INPUT(6)
    25300        :NULL
    25400     ELSE
    25500        SET WK-GOOD-COMPARE = 'F'
    25600     ENDIF
    25700     IF WK-GOOD-COMPARE = 'T'
    25800       :NULL
    25900     ELSE
    26000        SET ATTRIBUTE 'P' TEMP ON FA92011.POL-INPUT(1)
    26100        SET ATTRIBUTE 'P' TEMP ON FA92011.POL-INPUT(2)
    26200        SET ATTRIBUTE 'P' TEMP ON FA92011.POL-INPUT(3)
    26300        SET ATTRIBUTE 'P' TEMP ON FA92011.POL-INPUT(4)
    26400        SET ATTRIBUTE 'P' TEMP ON FA92011.POL-INPUT(5)
    26500        SET ATTRIBUTE 'P' TEMP ON FA92011.POL-INPUT(6)
    26600        SET FA92011.MSGLINE =
    26700          'NO UPDATE, DATA CHANGED SINCE UPDATE BEGAN, CONTACT OP'
    26800        QUIT COMPARE-LOOP
    26900     ENDIF
    27000
    27100     WHEN NONE
    27200        SET MSGLINE = 'EMPLOYEE ID NOT FOUND FOR UPDATE'
    27300     ENDFOR
    27400  ENDPROC    :COMPARE-RECS
    27500
    27600  :-----------------------------------------------------------------
    27700  : RESET THEN RESEND THE INITIAL SCREEN
    27800  :-----------------------------------------------------------------
    27900  <<RESEND-PNL>> PROCEDURE
    28000     RESET FA92011
    28100
    28200     SET ATTRIBUTE 'IP' TEMP ON FA92011.COLUMN-HEADS
    28300     SET ATTRIBUTE 'IP' TEMP ON FA92011.MSGLINE
```

```
PGM  CCFA92CI   001   TEST    SYS:$IO    MARCH 4, 1987        18:44:00

PROCEDURE:    CCFA92CI          VERSION:  001  STATUS:  TEST

  SEQ  STATEMENT
-------------------------------------------------------------------------------
28400     SET ATTRIBUTE 'IP' TEMP ON FA9201I.PFK-MSG
28500     SET ATTRIBUTE 'UAL' ON FA9201I.MAJSYS
28600     SET ATTRIBUTE 'UAL' ON FA9201I.ACTION
28700     SET ATTRIBUTE 'UAL' ON FA9201I.OBJECT
28800     SET ATTRIBUTE 'UAL' ON FA9201I.SEARCH
28900     MOVE $SPACES TO FA9201I.MAJSYS
29000     MOVE $SPACES TO FA9201I.ACTION
29100     MOVE $SPACES TO FA9201I.OBJECT
29200     MOVE $SPACES TO FA9201I.SEARCH
29300
29400     LOOP
29500         VARYING WK-RESEMO-CTR FROM 1 BY 1 UP THRU 12
29600       MOVE $SPACES TO FA9201I.POL-HEADER(WK-RESEMO-CTR)
29700     ENDLOOP
29800
29900  EMOPROC  :RESENO-PNL
30000
30100  :-------------------------------------------------------------------
30200  : DISPLAY THE EMPLOYEE/DEPARTMENT INDEX
30300  :-------------------------------------------------------------------
30400  <<DISPLAY-INDEX>> PROCEDURE
30500
30600     SET WK-PNL-CTR = 0
30700     SET WK-CTR     = 0
30800     FOR THE FIRST 12 MS-EMPLOYEE-2-U  NO UPDATE
30900         WHERE MS-EMPLOYEE-2-U.EMPNO    >= WK-READ-KEY
31000       ORDERED BY MS-EMPLOYEE-2-U.EMPNO
31100
31200       SET WK-CTR          = WK-CTR + 1       :VALUES 1 TO 12
31300       SET WK-PNL-CTR      = WK-PNL-CTR + 1   :VALUES 1 TO #EMP
31400
31500       SET WK-DL-NDX-ALPN  = WK-PNL-CTR
31600       SET WK-DL-NDX-ALPN  = $TRANSLATE(WK-DL-NDX-ALPN,FROM=' ',TO='0')
31700       SET WK-DL-WORKDEPT  = MS-EMPLOYEE-2-U.WORKDEPT
31800       SET WK-DL-EMPNO     = MS-EMPLOYEE-2-U.EMPNO
31900       SET WK-FNAME-BYTE-1 = $SUBSTR(MS-EMPLOYEE-2-U.LASTNAME,START=1,
32000                                     LENGTH=1)
32100       SET WK-DL-FULL-NAME = $STRING(WK-FNAME-BYTE-1,MIDINIT,' ',
32200                                     FIRSTNME)
32300
32400     FOR FIRST MS-DEPARTMENT-U NO UPDATE
32500         WHERE MS-DEPARTMENT-U.DEPTNO = MS-EMPLOYEE-2-U.WORKDEPT
32600       SET WK-DL-DEPTNAME = MS-DEPARTMENT-U.DEPTNAME
32700     WHEN NONE
32800       :NULL
32900     ENDFOR
33000
```

PROCEDURE: CCFA92CI VERSION: DDI STATUS: TEST

```
------ ---------------------------------------------------------------
  SEQ  STATEMENT
------ ---------------------------------------------------------------
33100          MOVE MS-EMPLOYEE-2-U.EMPNO TO WK-READ-KEY
33200          MOVE WK-DETAIL-LINE          TO FA92D1I.PDL-HEADER(WK-CTR)
33300
33400          SET WK-HOLD-NAME = WK-DL-FULL-NAME
33500      WHEN NONE
33600          SET FA92D1I.MSGLINE = 'END OF FILE'
33700      ENDFOR
33800 ENDPROC  :DISPLAY-INDEX
33900
34000 :-----------------------------------------------------------------
34100 :MOVE THE COLUMN HEADINGS TO THE REPORT
34200 :-----------------------------------------------------------------
34300 <<LOAD-COL-HEADS>> PROCEDURE
34400      MOVE $STRING('NO      ','D/ID   ','DEPARTMENT NAME                ',
34500                   '    E/ID      ','EMPLOYEE NAME')
34600        TO FA92D1I.COLUMN-HEADS
34700 ENDPROC
34800
34900 :<<ERROR>> PROCEDURE
35000 :    CALL CCFA99VI USING INPUT "ID"
35100 :    QUIT RUN
35200 :ENDPROC    :ERROR
```

```
PGM  CCFA92CI   001   TEST    SYS:$10    MARCH 4, 1987          18:44:00

ERRORS:        CCFA92CI          VERSION:  001  STATUS:  TEST

PROCEDURE:     CCFA92CI          VERSION:  001  STATUS:  TEST
31500 CMGENP24-I WK-PNL-CTR, A NUMERIC FIELD OR LITERAL, MAY BE  TRUNCATED IN
                 MOVE TO AN ALPHANUMERIC FIELD
```

```
PGM  CCFA92CI   001   TEST    SYS:&ID    MARCH 4, 1987        18:44:00


                                  COMPILER  DIAGNOSTICS  AND  SUMMARY

PROGRAM HAS BEEN SUCCESSFULLY COMPILED

NO ERROR MESSAGE(S) FLAGGED IN THIS COMPILATION
NO WARNING MESSAGE(S) FLAGGED IN THIS COMPILATION
    1 ADVISORY MESSAGE(S) FLAGGED IN THIS COMPILATION

COMPILED PROGRAM STATISTICS:

- NUMBER OF SHAREABLE     MEMBERS  OF THE PROGRAM =          001
- LENGTH OF SHAREABLE     PORTIONS OF THE PROGRAM =        8,996  BYTES
- NUMBER OF NON-SHAREABLE MEMBERS  OF THE PROGRAM =          001
- LENGTH OF NON-SHAREABLE PORTIONS OF THE PROGRAM =        2,540  BYTES
```

Appendix B

DB2 Programs and Output

TELEPHONE DIRECTORY

LAST NAME	FIRST NAME	INITIAL	PHONE NUMBER	EMPLOYEE NUMBER	WORK DEPT	WORK DEPT NAME
AAS	CHRISTINE	I	3978	000010	A00	SPIFFY COMPUTER SERVICE DIV.
UCCNESI	VINCENZO	G	3490	000110	A00	SPIFFY COMPUTER SERVICE DIV.
*CONNELL	SEAN		2167	000120	A00	SPIFFY COMPUTER SERVICE DIV.
MOMPSON	MICHAEL	L	3476	000020	B01	PLANNING
WAN	SALLY	A	4738	000030	C01	INFORMATION CENTER
UINTANA	DOLORES	M	4578	000130	C01	INFORMATION CENTER
ICNOLLS	HEATHER	A	1793	000140	C01	INFORMATION CENTER
TERN	IRVING	F	6423	000060	D11	MANUFACTURING SYSTEMS
OAMSON	BRUCE		4510	000150	D11	MANUFACTURING SYSTEMS
IANKA	ELIZABETH	R	3782	000160	D11	MANUFACTURING SYSTEMS
OSHIMURA	MASATOSHI	J	2890	000170	D11	MANUFACTURING SYSTEMS
COUTTEN	MARILYN	S	1682	000180	D11	MANUFACTURING SYSTEMS
ALKER	JAMES	H	2986	000190	D11	MANUFACTURING SYSTEMS
ROWN	DAVID		4501	000200	D11	MANUFACTURING SYSTEMS
ONES	WILLIAM	T	0942	000210	D11	MANUFACTURING SYSTEMS
UTZ	JENNIFER	K	0672	000220	D11	MANUFACTURING SYSTEMS
ULASKI	EVA	D	7831	000070	D21	ADMINISTRATION SYSTEMS
EFFERSON	JAMES	J	4265	000230	D21	ADMINISTRATION SYSTEMS
ARINO	SALVATORE	M	3780	000240	D21	ADMINISTRATION SYSTEMS
MITH	DANIEL	S	0961	000250	D21	ADMINISTRATION SYSTEMS
OHNSON	SYBIL	H	8953	000260	D21	ADMINISTRATION SYSTEMS
EREZ	MARIA	L	9001	000270	D21	ADMINISTRATION SYSTEMS
EYER	JOHN	B	6789	000050	E01	SUPPORT SERVICES
ENDERSON	EILEEN	W	5498	000090	E11	OPERATIONS
CHNEIDER	ETHEL	R	8997	000280	E11	OPERATIONS
ARKER	JOHN	R	4502	000290	E11	OPERATIONS
ITH	PHILIP	X	2095	000300	E11	OPERATIONS
ETRIGHT	MAUDE	F	3332	000310	E11	OPERATIONS
PENSER	THEODORE	Q	0972	000100	E21	SOFTWARE SUPPORT
ENTR	RAMLRL	V	9990	000320	E21	SOFTWARE SUPPORT

DB2 - Batch Program Output

EE	WING		2103	000330	E21	SOFTWARE SUPPORT
OUNOT	JASON	R	5698	000340	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300000	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300001	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300002	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300003	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300004	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300005	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300006	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300007	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300008	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300009	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300010	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300011	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300012	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300013	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300014	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300015	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300016	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300017	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300018	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300019	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300020	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300021	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300022	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300023	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300024	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300025	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300026	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300027	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300028	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300029	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300030	E21	SOFTWARE SUPPORT

0E	TEST	E	0000	300031	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300032	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300033	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300034	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300035	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300036	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300037	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300038	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300039	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300040	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300041	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300042	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300043	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300044	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300045	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300046	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300047	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300048	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300049	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300050	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300051	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300052	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300053	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300054	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300055	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300056	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300057	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300058	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300059	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300060	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300061	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300062	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300063	E21	SOFTWARE SUPPORT

0E	TEST	E	0000	300064	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300065	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300066	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300067	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300068	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300069	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300070	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300071	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300072	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300073	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300074	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300075	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300076	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300077	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300078	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300079	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300080	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300081	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300082	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300083	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300084	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300085	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300086	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300087	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300088	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300089	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300090	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300091	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300092	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300093	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300094	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300095	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300096	E21	SOFTWARE SUPPORT

0E	TEST	E	0000	300097	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300098	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300099	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300100	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300101	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300102	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300103	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300104	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300105	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300106	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300107	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300108	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300109	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300110	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300111	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300112	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300113	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300114	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300115	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300116	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300117	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300118	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300119	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300120	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300121	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300122	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300123	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300124	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300125	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300126	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300127	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300128	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300129	E21	SOFTWARE SUPPORT

0E	TEST	E	0000	300031	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300032	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300033	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300034	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300035	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300036	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300037	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300038	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300039	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300040	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300041	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300042	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300043	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300044	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300045	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300046	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300047	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300048	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300049	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300050	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300051	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300052	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300053	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300054	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300055	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300056	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300057	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300058	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300059	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300060	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300061	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300062	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300063	E21	SOFTWARE SUPPORT

0E	TEST	E	0000	300064	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300065	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300066	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300067	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300068	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300069	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300070	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300071	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300072	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300073	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300074	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300075	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300076	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300077	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300078	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300079	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300080	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300081	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300082	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300083	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300084	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300085	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300086	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300087	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300088	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300089	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300090	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300091	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300092	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300093	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300094	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300095	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300096	E21	SOFTWARE SUPPORT

OE	TEST	E	0000	300097	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300098	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300099	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300100	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300101	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300102	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300103	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300104	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300105	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300106	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300107	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300108	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300109	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300110	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300111	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300112	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300113	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300114	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300115	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300116	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300117	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300118	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300119	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300120	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300121	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300122	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300123	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300124	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300125	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300126	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300127	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300128	E21	SOFTWARE SUPPORT
OE	TEST	E	0000	300129	E21	SOFTWARE SUPPORT

0E	TEST	E	0000	300985	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300989	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300990	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300991	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300992	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300993	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300994	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300995	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300996	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300997	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300998	E21	SOFTWARE SUPPORT
0E	TEST	E	0000	300999	E21	SOFTWARE SUPPORT

TELEPHONE DIRECTORY

AST NAME	FIRST NAME	INITIAL	PHONE NUMBER	EMPLOYEE NUMBER	WORK DEPT	WORK DEPT NAME
ONES	WILLIAM	T	0942	000210	D11	MANUFACTURING SYSTEMS
OHNSON	SYBIL	V	3953	000260	D21	ADMINISTRATION SYSTEMS

TELEPHONE DIRECTORY

AST NAME	FIRST NAME	INITIAL	PHONE NUMBER	EMPLOYEE NUMBER	WORK DEPT	WORK DEPT NAME
NOMPSON	MICHAEL	L	3476	000020	B01	PLANNING
DAMSON	BRUCE		4510	000150	D11	MANUFACTURING SYSTEMS
EFFERSON	JAMES	J	4265	000230	D21	ADMINISTRATION SYSTEMS
ONNSON	SYBIL	V	8953	000260	D21	ADMINISTRATION SYSTEMS
ENDERSON	EILEEN	W	5498	000090	E11	OPERATIONS

TELEPHONE DIRECTORY

LAST NAME	FIRST NAME	INITIAL	PHONE NUMBER	EMPLOYEE NUMBER	WORK DEPT	WORK DEPT NAME
MITH	DANIEL	S	0961	000250	D21	ADMINISTRATION SYSTEMS
MITH	PHILIP	X	2095	000300	E11	OPERATIONS

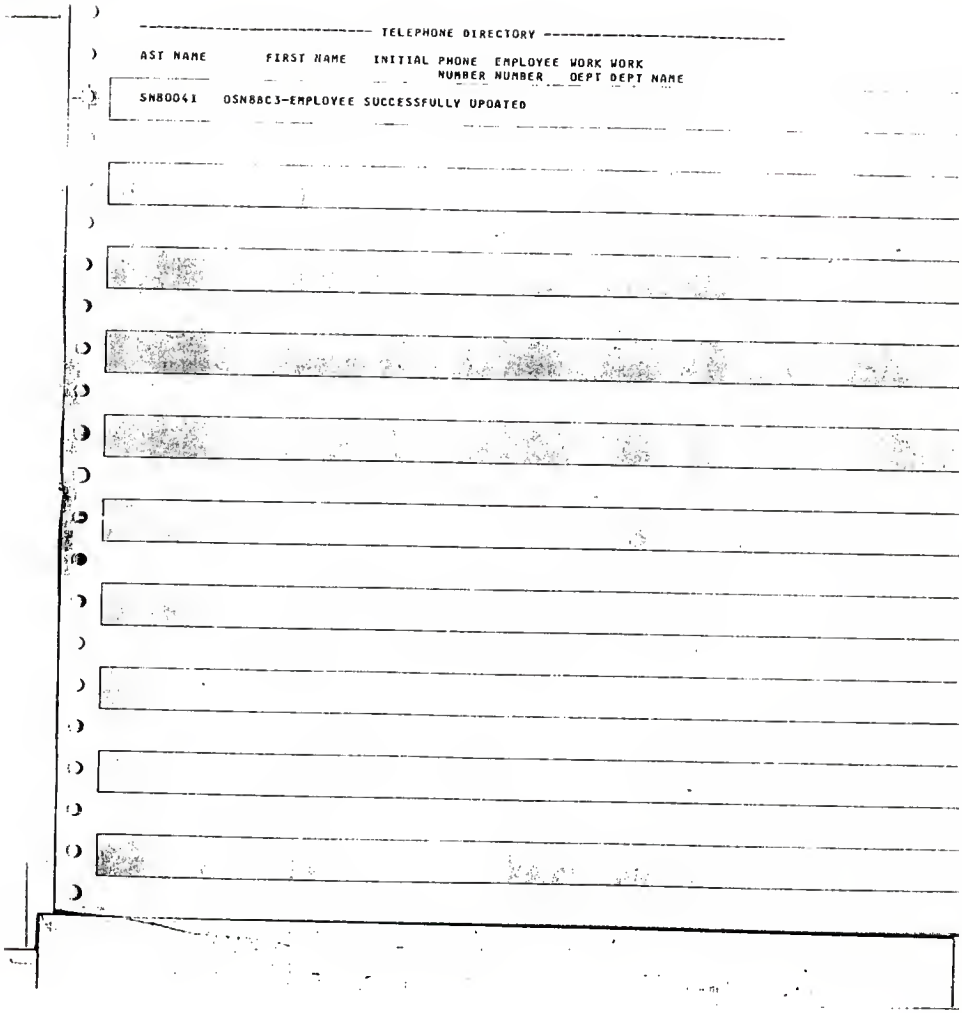
```
                    TELEPHONE DIRECTORY

LAST NAME       FIRST NAME   INITIAL PHONE  EMPLOYEE WORK WORK
                                     NUMBER NUMBER   DEPT DEPT NAME

SN80081   DSN83C3-NO EMPLOYEE FOUND IN TABLE
```

TELEPHONE DIRECTORY

LAST NAME	FIRST NAME	INITIAL	PHONE NUMBER	EMPLOYEE NUMBER	WORK DEPT	WORK DEPT NAME
BROWN	DAVID		4501	000200	D11	MANUFACTURING SYSTEMS


TELEPHONE DIRECTORY
AST NAME
FIRST NAME
INITIAL
PHONE NUMBER
EMPLOYEE NUMBER
WORK DEPT
WORK DEPT NAME
SN80041 OSN88C3-EMPLOYEE SUCCESSFULLY UPDATED

```
        DATASET: DSN120.DSNSAMP                                          DATE: 87/02/04
        MEMBER:  DSN8BC3                                                 TIME: 08:33
                                                                         PAGE: 1
START
 COL    ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   1  000100********* DSN8BC3 - DB2 SAMPLE PHONE APPLICATION - COBOL - BATCH *** 00010000
   1  000200*                                                                  * 00020000
   1  000300*    MODULE NAME = DSN8BC3                                         * 00030000
   1  000400*                                                                  * 00040000
   1  000500*    DESCRIPTIVE NAME = DB2  SAMPLE APPLICATION                    * 00050000
   1  000600*                       PHONE APPLICATION                          * 00060000
   1  000700*                       BATCH                                      * 00070000
   1  000800*                       COBOL                                      * 00080000
   1  000900*                                                                  * 00090000
   1  001000*    COPYRIGHT = 5740-XYR (C) COPYRIGHT IBM CORP 1982, 1985        * 00100000
   1  001000*     REFER TO COPYRIGHT INSTRUCTIONS FORM NUMBER G120-2083         * 00110000
   1  001000*                                                                  * 00120000
   1  001000*    STATUS = RELEASE 2, LEVEL 0                                   * 00130000
   1  001400*                                                                  * 00140000
   1  001500*    FUNCTION = THIS MODULE LISTS EMPLOYEE PHONE NUMBERS AND       * 00150000
   1  001600*               UPDATES THEM IF DESIRED.                           * 00160000
   1  001700*                                                                  * 00170000
   1  001800*    NOTES = NONE                                                  * 00180000
   1  001900*                                                                  * 00190000
   1  002000*    MODULE TYPE = COBOL PROGRAM                                   * 00200000
   1  002100*       PROCESSOR   = DB2  PRECOMPILER, VS COBOL                   * 00210000
   1  002200*       MODULE SIZE = SEE LINK EDIT                                * 00220000
   1  002300*       ATTRIBUTES  = NOT REENTRANT OR REUSABLE                    * 00230000
   1  002400*                                                                  * 00240000
   1  002500*                                                                  * 00250000
   1  002600*    ENTRY POINT = DSN8BC3                                         * 00260000
   1  002700*       PURPOSE = SEE FUNCTION                                     * 00270000
   1  002800*       LINKAGE = INVOKED FROM DSN RUN                             * 00280000
   1  002900*       INPUT   =                                                  * 00290000
   1  003000*                                                                  * 00300000
   1  003100*                 SYMBOLIC LABEL/NAME =  CARDIN                    * 00310000
   1  003200*                 DESCRIPTION         =  INPUT REQUEST FILE        * 00320000
   1  003300*                                                                  * 00330000
   1  003400*                 SYMBOLIC LABEL/NAME =  VPHONE                    * 00340000
   1  003500*                 DESCRIPTION         =  VIEW OF TELEPHONE         * 00350000
   1  003600*                                        INFORMATION               * 00360000
   1  003700*                                                                  * 00370000
   1  003800*                                                                  * 00380000
   1  003900*       OUTPUT  =                                                  * 00390000
   1  004000*                                                                  * 00400000
   1  004100*                 SYMBOLIC LABEL/NAME =  REPORT                    * 00410000
   1  004200*                 DESCRIPTION         =  REPORT OF EMPLOYEE        * 00420000
   1  004300*                                        PHONE NUMBERS             * 00430000
   1  004400*                                                                  * 00440000
   1  004500*                 SYMBOLIC LABEL/NAME =  VEMPLP                    * 00450000
   1  004600*                 DESCRIPTION         =  VIEW OF EMPLOYEE          * 00460000
   1  004700*                                        INFORMATION               * 00470000
   1  004800*                                                                  * 00480000
   1  004900*    EXIT-NORMAL = RETURN CODE 0 NORMAL COMPLETION                 * 00490000
   1  005000*                                                                  * 00500000
   1  005100*    EXIT-ERROR =                                                  * 00510000
   1  005200*                                                                  * 00520000
   1  005300*       RETURN CODE    =  NONE                                     * 00530000
   1  005400*                                                                  * 00540000
```

DB2 - Batch Program

```
      DATASET: DSN120.DSNSAMP                                        DATE: 87/02/04
      MEMBER:  DSN8BC3                                               TIME: 08:33
                                                                     PAGE: 2
START
 COL   ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   1  005500*       ABEND CODES     =  NONE                                  * 00550000
   1  005600*                                                                * 00560000
   1  005700*       ERROR-MESSAGES =                                         * 00570000
   1  005800*             DSN8004I - EMPLOYEE SUCCESSFULLY UPDATED           * 00580000
   1  005900*             DSN8007E - EMPLOYEE DOES NOT EXIST, UPDATE NOT DONE* 00590000
   1  006000*             DSN8008I - NO EMPLOYEE FOUND IN TABLE              * 00600000
   1  006100*             DSN8053I - ROLLBACK SUCCESSFUL, ALL UPDATES REMOVED* 00610000
   1  006200*             DSN8060E - SQL ERROR, RETURN CODE IS:              * 00620000
   1  006300*             DSN8061E - ROLLBACK FAILED, RETURN CODE IS:        * 00630000
   1  006400*             DSN8068E - INVALID REQUEST, SHOULD BE 'L' OR 'U'   * 00640000
   1  006500*             DSN8075E - MESSAGE FORMAT ROUTINE ERROR,           * 00650000
   1  006600*                        RETURN CODE IS:                         * 00660000
   1  006700*                                                                * 00670000
   1  006800*  EXTERNAL REFERENCES =                                         * 00680000
   1  006900*     ROUTINES/SERVICES =                                        * 00690000
   1  007000*             DSNTIAR  -       TRANSLATE SQLCA INTO MESSAGES     * 00700000
   1  007100*             DSN8MCG  -       ERROR MESSAGE ROUTINE             * 00710000
   1  007200*                                                                * 00720000
   1  007300*       DATA-AREAS          =  NONE                              * 00730000
   1  007400*                                                                * 00740000
   1  007500*       CONTROL-BLOCKS      =                                    * 00750000
   1  007600*             SQLCA     -      SQL COMMUNICATION AREA            * 00760000
   1  007700*                                                                * 00770000
   1  007800*  TABLES =  NONE                                               * 00780000
   1  007900*                                                                * 00790000
   1  008000*                                                                * 00800000
   1  008100*  CHANGE-ACTIVITY = NONE                                        * 00810000
   1  008200*                                                                * 00820000
   1  008300*                                                                * 00830000
   1  008400*  *PSEUDOCODE*                                                  * 00840000
   1  008500*                                                                * 00850000
   1  008600*  PROCEDURE                                                     * 00860000
   1  008700*    GET FIRST INPUT                                             * 00870000
   1  008800*    DO WHILE NORE INPUT                                         * 00880000
   1  008900*      CREATE REPORT HEADING                                     * 00890000
   1  009000*                                                                * 00900000
   1  009100*      CASE (ACTION)                                             * 00910000
   1  009200*                                                                * 00920000
   1  009300*        SUBCASE ('L')                                           * 00930000
   1  009400*          IF LASTNAME IS '*' THEN                               * 00940000
   1  009500*            LIST ALL EMPLOYEES                                  * 00950000
   1  009600*          ELSE                                                  * 00960000
   1  009700*            IF LASTNAME CONTAINS '%' THEN                       * 00970000
   1  009800*              LIST EMPLOYEES GENERIC                            * 00980000
   1  009900*            ELSE                                                * 00990000
   1  010000*              LIST EMPLOYEES SPECIFIC                           * 01000000
   1  010100*        ENDSUB                                                  * 01010000
   1  010200*                                                                * 01020000
   1  010300*        SUBCASE ('U')                                           * 01030000
   1  010400*          UPDATE PHONENUMBER FOR EMPLOYEE                       * 01040000
   1  010500*          WRITE CONFIRMATION MESSAGE                            * 01050000
   1  010600*        OTHERWISE                                               * 01060000
   1  010700*          INVALID REQUEST                                       * 01070000
   1  010800*        ENDSUB                                                  * 01080000
```

```
        DATASET: DSN120.DSNSAMP                                  DATE: 87/02/04
        MEMBER:  DSN8BC3                                         TIME: 08:33
                                                                 PAGE: 3
START
 COL    ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   1  010900*                                                                  *
   1  011000*       ENDCASE                                                    *
   1  011100*       GET NEXT INPUT                                             *
   1  011200*     END                                                          *
   1  011300*                                                                  *
   1  011400*     IF SQL ERROR OCCURS THEN                                     *
   1  011500*        DO                                                        *
   1  011600*         FORMAT ERROR MESSAGE                                     *
   1  011700*         ROLLBACK                                                 *
   1  011800*        END                                                       *
   1  011900*   END.                                                           *
   1  012000*-----------------------------------------------------------------*
   1  012100
   1  012200
   1  012300/
   1  012400 IDENTIFICATION DIVISION.
   1  012500*-----------------------
   1  012600 PROGRAM-ID.    DSN8BC3.
   1  012700
   1  012800 ENVIRONMENT DIVISION.
   1  012900*--------------------
   1  013000 CONFIGURATION SECTION.
   1  013100 SPECIAL-NAMES.       C01 IS TO-TOP-OF-PAGE.
   1  013200 INPUT-OUTPUT SECTION.
   1  013300 FILE-CONTROL.
   1  013400     SELECT CARDIN
   1  013500            ASSIGN TO DA-S-CARDIN.
   1  013600     SELECT REPOUT
   1  013700            ASSIGN TO UT-S-REPORT.
   1  013800
   1  013900 DATA DIVISION.
   1  014000*-------------
   1  014100 FILE SECTION.
   1  014200 FD      CARDIN
   1  014300         RECORD CONTAINS 80 CHARACTERS
   1  014400         BLOCK CONTAINS 0 RECORDS
   1  014500         LABEL RECORDS ARE OMITTED.
   1  014600 01  CARDREC                    PIC X(80).
   1  014700
   1  014800 FD  REPOUT
   1  014900         RECORD CONTAINS 120 CHARACTERS
   1  015000         LABEL RECORDS ARE OMITTED
   1  015100         DATA RECORD IS REPREC.
   1  015200 01  REPREC                     PIC X(120).
   1  015300/
   1  015400 WORKING-STORAGE SECTION.
   1  015500
   1  015600*****************************************************
   1  015700* STRUCTURE FOR INPUT                               *
   1  015800*****************************************************
   1  015900 01  IOAREA.
   1  016000         02  ACTION              PIC X(01).
   1  016100         02  LNAME               PIC X(15).
   1  016200         02  FNAME               PIC X(12).
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   1  016300           02  ENO                  PIC X(06).                        01630000
   1  016400           02  NEWNO                PIC X(04).                        01640000
   1  016500           02  FILLER               PIC X(42).                        01650000
   1  016600                                                                      01660000
   1  016700*************************************************                    01670000
   1  016800* REPORT HEADER STRUCTURE                       *                    01680000
   1  016900*************************************************                    01690000
   1  017000 01  REPHDR1.                                                        01700000
   1  017100           02  FILLER PIC X(29)                                      01710000
   1  017200               VALUE '-----------------------------'.                01720000
   1  017300           02  FILLER PIC X(21)                                      01730000
   1  017400               VALUE ' TELEPHONE DIRECTORY '.                       01740000
   1  017500           02  FILLER PIC X(29)                                      01750000
   1  017600               VALUE '-----------------------------'.                01760000
   1  017700 01  REPHDR2.                                                        01770000
   1  017800           02  FILLER PIC X(09) VALUE 'LAST NAME'.                   01780000
   1  017900           02  FILLER PIC X(07) VALUE SPACES.                        01790000
   1  018000           02  FILLER PIC X(10) VALUE 'FIRST NAME'.                  01800000
   1  018100           02  FILLER PIC X(03) VALUE SPACES.                        01810000
   1  018200           02  FILLER PIC X(08) VALUE 'INITIAL'.                     01820000
   1  018300           02  FILLER PIC X(07) VALUE 'PHONE'.                       01830000
   1  018400           02  FILLER PIC X(09) VALUE 'EMPLOYEE'.                    01840000
   1  018500           02  FILLER PIC X(05) VALUE 'WORK'.                        01850000
   1  018600           02  FILLER PIC X(04) VALUE 'WORK'.                        01860000
   1  018700 01  REPHDR3.                                                        01870000
   1  018800           02  FILLER PIC X(37) VALUE SPACES.                        01880000
   1  018900           02  FILLER PIC X(07) VALUE 'NUMBER'.                      01890000
   1  019000           02  FILLER PIC X(09) VALUE 'NUMBER'.                      01900000
   1  019100           02  FILLER PIC X(05) VALUE 'DEPT'.                        01910000
   1  019200           02  FILLER PIC X(05) VALUE 'DEPT'.                        01920000
   1  019300           02  FILLER PIC X(04) VALUE 'NAME'.                        01930000
   1  019400                                                                      01940000
   1  019500*************************************************                    01950000
   1  019600* REPORT STRUCTURE                              *                    01960000
   1  019700*************************************************                    01970000
   1  019800 01  REPDATA.                                                        01980000
   1  019900           02  RLNAME     PIC X(15).                                01990000
   1  020000           02  FILLER     PIC X(01) VALUE SPACES.                   02000000
   1  020100           02  RFNAME     PIC X(12).                                02010000
   1  020200           02  FILLER     PIC X(04) VALUE SPACES.                   02020000
   1  020300           02  RMIDINIT   PIC X(01).                                02030000
   1  020400           02  FILLER     PIC X(04) VALUE SPACES.                   02040000
   1  020500           02  RPHONE     PIC X(04).                                02050000
   1  020600           02  FILLER     PIC X(03) VALUE SPACES.                   02060000
   1  020700           02  REMPNO     PIC X(06).                                02070000
   1  020800           02  FILLER     PIC X(03) VALUE SPACES.                   02080000
   1  020900           02  RDEPINO    PIC X(03).                                02090000
   1  021000           02  FILLER     PIC X(02) VALUE SPACES.                   02100000
   1  021100           02  RDEPTNAME PIC X(36).                                 02110000
   1  021200                                                                     02120000
   1  021300*************************************************                    02130000
   1  021400* WORKAREAS                                     *                    02140000
   1  021500*************************************************                    02150000
   1  021600 01  LNAME-WORK.                                                    02160000
```

```
DATASET: DSN120.DSNSAMP                                          DATE: 87/02/04
MEMBER:  DSN8BC3                                                 TIME: 08:33
                                                                 PAGE: 5
START
 COL    ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   1  021700       49 LNAME-WORKL      PIC S9(4)  COMP.                      02170000
   1  021800       49 LNAME-WORKC      PIC X(15).                            02180000
   1  021900 01  FNAME-WORK.                                                 02190000
   1  022000       49 FNAME-WORKL      PIC S9(4)  COMP.                      02200000
   1  022100       49 FNAME-WORKC      PIC X(12).                            02210000
   1  022200 77  INPUT-SWITCH         PIC X           VALUE  'Y'.           02220000
   1  022300         88  NOMORE-INPUT                 VALUE  'N'.           02230000
   1  022400 77  NOT-FOUND            PIC S9(9) COMP VALUE  +100.           02240000
   1  022500                                                                02250000
   1  022600*****************************************************           02260000
   1  022700* VARIABLES FOR ERROR-HANDLING                      *           02270000
   1  022800*****************************************************           02280000
   1  022900 01  ERROR-MESSAGE.                                             02290000
   1  023000         02  ERROR-LEN    PIC S9(4)  COMP VALUE +960.           02300000
   1  023100         02  ERROR-TEXT   PIC X(120) OCCURS 8 TIMES             02310000
   1  023200                                     INDEXED BY ERROR-INDEX.    02320000
   1  023300 77  ERROR-TEXT-LEN       PIC S9(9)  COMP VALUE +120.           02330000
   1  023400                                                                02340000
   1  023500*****************************************************           02350000
   1  023600* SQL INCLUDE FOR SQLCA                             *           02360000
   1  023700*****************************************************           02370000
   1  023800          EXEC SQL INCLUDE SQLCA  END-EXEC.                     02380000
   1  023900                                                                02390000
   1  024000*****************************************************           02400000
   1  024100* SQL DECLARATION FOR VIEW VPHONE                   *           02410000
   1  024200*****************************************************           02420000
   1  024300          EXEC SQL DECLARE VPHONE TABLE                         02430000
   1  024400                   (LASTNAME        VARCHAR(15)  NOT NULL,      02440000
   1  024500                    FIRSTNAME       VARCHAR(12)  NOT NULL,      02450000
   1  024600                    MIDDLEINITIAL     CHAR(01)   NOT NULL,      02460000
   1  024700                    PHONENUMBER       CHAR(04)               ,  02470000
   1  024800                    EMPLOYEENUMBER    CHAR(06)   NOT NULL,      02480000
   1  024900                    DEPTNUMBER        CHAR(03)   NOT NULL,      02490000
   1  025000                    DEPTNAME        VARCHAR(36)  NOT NULL)      02500000
   1  025100                  END-EXEC.                                     02510000
   1  025200                                                                02520000
   1  025300*****************************************************           02530000
   1  025400* STRUCTURE FOR PPHONE RECORD                       *           02540000
   1  025500*****************************************************           02550000
   1  025600 01  PPHONE.                                                    02560000
   1  025700         02  LASTNAME.                                          02570000
   1  025800               49  LASTNAMEL      PIC S9(4)  COMP.              02580000
   1  025900               49  LASTNAMEC      PIC X(15)  VALUE SPACES.      02590000
   1  026000         02  FIRSTNAME.                                         02600000
   1  026100               49  FIRSTNAMEL     PIC S9(4)  COMP.              02610000
   1  026200               49  FIRSTNAMEC     PIC X(12)  VALUE SPACES.      02620000
   1  026300         02  MIDDLEINITIAL       PIC X(01).                    02630000
   1  026400         02  PHONENUMBER         PIC X(04).                    02640000
   1  026500         02  EMPLOYEENUMBER      PIC X(06).                    02650000
   1  026600         02  DEPTNUMBER          PIC X(03).                    02660000
   1  026700         02  DEPTNAME.                                          02670000
   1  026800               49  DEPTNAMEL      PIC S9(4)  COMP.              02680000
   1  026900               49  DEPTNAMEC      PIC X(36)  VALUE SPACES.      02690000
   1  027000*                                                               02700000
```

```
        DATASET: DSN120.DSNSAMP                                    DATE: 87/02/04
        MEMBER:  DSN8BC3                                           TIME: 08:33
                                                                   PAGE: 6
START
 COL    ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   1  027100 77  PERCENT-COUNTER              PIC S9(4)  COMP.                  02710000
   1  027200                                                                    02720000
   1  027300*************************************************                  02730000
   1  027400* SQL DECLARATION FOR VIEW VENPLP                *                  02740000
   1  027500*************************************************                  02750000
   1  027600           EXEC SQL DECLARE VENPLP TABLE                            02760000
   1  027700                    (EMPLOYEENUMBER   CHAR(06)  NOT NULL,           02770000
   1  027800                     PHONENUMBER      CHAR(04)          )           02780000
   1  027900                END-EXEC.                                           02790000
   1  028000                                                                    02800000
   1  028100*************************************************                  02810000
   1  028200* SQL CURSORS                                    *                  02820000
   1  028300*************************************************                  02830000
   1  028400*** CURSOR LISTS ALL EMPLOYEE NAMES                                02840000
   1  028500                                                                    02850000
   1  028600           EXEC SQL DECLARE TELE1 CURSOR FOR                        02860000
   1  028700                    SELECT *                                        02870000
   1  028800                    FROM   VPHONE                                   02880000
   1  028900                END-EXEC.                                           02890000
   1  029000                                                                    02900000
   1  029100*** CURSOR LISTS ALL EMPLOYEE NAMES WITH A PATTERN (%) OR (_)        02910000
   1  029200*** FOR LAST NAME                                                   02920000
   1  029300                                                                    02930000
   1  029400           EXEC SQL DECLARE TELE2 CURSOR FOR                        02940000
   1  029500                    SELECT *                                        02950000
   1  029600                    FROM   VPHONE                                   02960000
   1  029700                    WHERE  LASTNAME  LIKE :LNAME-WORK               02970000
   1  029800                      AND  FIRSTNAME LIKE :FNAME-WORK               02980000
   1  029900                END-EXEC.                                           02990000
   1  030000                                                                    03000000
   1  030100*** CURSOR LISTS ALL EMPLOYEES WITH A SPECIFIC                       03010000
   1  030200*** LAST NAME                                                       03020000
   1  030300                                                                    03030000
   1  030400           EXEC SQL DECLARE TELE3 CURSOR FOR                        03040000
   1  030500                    SELECT *                                        03050000
   1  030600                    FROM   VPHONE                                   03060000
   1  030700                    WHERE  LASTNAME   =    :LNAME                   03070000
   1  030800                      AND  FIRSTNAME LIKE :FNAME-WORK               03080000
   1  030900                END-EXEC.                                           03090000
   1  031000/  ..                                                              03100000
   1  031100/************************************************                  03110000
   1  031200* FIELDS SENT TO MESSAGE ROUTINE                 *                  03120000
   1  031300************************************************                   03130000
   1  031400 01  MAJOR                    PIC X(07) VALUE 'DSN8BC3'.          03140000
   1  031500                                                                    03150000
   1  031600 01  MSGCODE                  PIC X(4).                           03160000
   1  031700                                                                    03170000
   1  031800 01  OUTMSG                   PIC X(69).                          03180000
   1  031900                                                                    03190000
   1  032000 01  MSG-REC1.                                                     03200000
   1  032100         02 OUTMSG1           PIC X(69).                          03210000
   1  032200         02 RETCODE           PIC S9(9).                          03220000
   1  032300                                                                    03230000
   1  032400 01  MSG-REC2.                                                    03240000
```

```
        DATASET: DSN120.DSNSAMP                                          DATE: 87/02/04
        MEMBER:  DSN8BC3                                                 TIME: 08:33
                                                                         PAGE: 7
START
 COL    ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   1  032500          02 OUTMSG2              PIC X(69).                          03250000
   1  032600                                                                      03260000
   1  032700                                                                      03270000
   1  032800 PROCEDURE DIVISION.                                                  03280000
   1  032900*-----------------                                                    03290000
   1  033000                                                                      03300000
   1  033100*************************************************                     03310000
   1  033200* SQL RETURN CODE HANDLING                      *                     03320000
   1  033300*************************************************                     03330000
   1  033400           EXEC SQL WHENEVER SQLERROR   GOTO OBERROR END-EXEC.        03340000
   1  033500           EXEC SQL WHENEVER SQLWARNING GOTO OBERROR END-EXEC.        03350000
   1  033600           EXEC SQL WHENEVER NOT FOUND  CONTINUE     END-EXEC.        03360000
   1  033700                                                                      03370000
   1  033800*************************************************                     03380000
   1  033900* MAIN PROGRAM ROUTINE                          *                     03390000
   1  034000*************************************************                     03400000
   1  034100 PROG-START.                                                          03410000
   1  034200*                                  **OPEN FILES                       03420000
   1  034300           OPEN INPUT  CARDIN                                         03430000
   1  034400                OUTPUT REPOUT.                                        03440000
   1  034500                                                                      03450000
   1  034600*                                  **GET FIRST INPUT                  03460000
   1  034700           READ CARDIN RECORD INTO IOAREA                             03470000
   1  034800              AT END MOVE 'N' TO INPUT-SWITCH.                        03480000
   1  034900                                                                      03490000
   1  035000*                                  **MAIN ROUTINE                     03500000
   1  035100           PERFORM PROCESS-INPUT                                      03510000
   1  035200              UNTIL NOMORE-INPUT.                                     03520000
   1  035300 PROG-END.                                                            03530000
   1  035400*                                  **CLOSE FILES                      03540000
   1  035500           CLOSE CARDIN                                               03550000
   1  035600                 REPOUT.                                              03560000
   1  035700           GOBACK.                                                    03570000
   1  035800                                                                      03580000
   1  035900*************************************************                     03590000
   1  036000* CREATE REPORT HEADING                         *                     03600000
   1  036100* SELECT ACTION                                 *                     03610000
   1  036200*************************************************                     03620000
   1  036300 PROCESS-INPUT.                                                       03630000
   1  036400*                                                 **PRINT HEADING    03640000
   1  036500           WRITE REPREC FROM REPHDR1                                  03650000
   1  036600              AFTER ADVANCING TO-TOP-OF-PAGE.                         03660000
   1  036700           WRITE REPREC FROM REPHDR2                                  03670000
   1  036800              AFTER ADVANCING 2 LINES.                                03680000
   1  036900           WRITE REPREC FROM REPHDR3.                                 03690000
   1  037000                                                                      03700000
   1  037100*                                                 **SELECT ACTION    03710000
   1  037200           IF ACTION = 'L'                                            03720000
   1  037300              PERFORM LIST-FUNCTION                                   03730000
   1  037400           ELSE                                                       03740000
   1  037500              IF ACTION = 'U'                                         03750000
   1  037600                 PERFORM TELEPHONE-UPDATE                             03760000
   1  037700                                                                      03770000
   1  037800              ELSE                                                    03780000
```

```
      DATASET: DSN120.DSNSAMP                                          DATE: 87/02/04
      MEMBER:  DSN8BC3                                                 TIME: 08:33
                                                                       PAGE: 8
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   1  037900*                                          **INVALID REQUEST      03790000
   1  038000*                                          **PRINT ERROR MESSAGE  03800000
   1  038100           MOVE '068E' TO MSGCODE                                  03810000
   1  038200           CALL 'DSN8MCG' USING MAJOR MSGCODE OUTMSG               03820000
   1  038300           MOVE OUTMSG TO OUTMSG2                                  03830000
   1  038400           WRITE REPREC FROM MSG-REC2                              03840000
   1  038500                AFTER ADVANCING 2 LINES.                           03850000
   1  038600           READ CARDIN RECORD INTO EOAREA                          03860000
   1  038700              AT END MOVE 'N' TO INPUT-SWITCH.                     03870000
   1  038800/                                                                 03880000
   1  038900*************************************************                 03890000
   1  039000* DETERMINE FORM OF NAME USED TO LIST EMPLOYEES     *              03900000
   1  039100*************************************************                 03910000
   1  039200 LIST-FUNCTION.                                                   03920000
   1  039300*                                          **NO LAST NAME GIVEN   03930000
   1  039400           IF LNAME = SPACES                                       03940000
   1  039500              MOVE '%' TO LNAME.                                   03950000
   1  039600*                                          **NO FIRST NAME GIVEN  03960000
   1  039700           IF FNAME = SPACES                                       03970000
   1  039800              MOVE '%' TO FNAME.                                   03980000
   1  039900*                                          **LIST ALL EMPLOYEES   03990000
   1  040000           IF LNAME = '%'                                          04000000
   1  040100              PERFORM LIST-ALL                                     04010000
   1  040200           ELSE                                                    04020000
   1  040300*                                          **UNSTRING LAST NAME   04030000
   1  040400              UNSTRING LNAME                                       04040000
   1  040500                 DELIMITED BY SPACE                                04050000
   1  040600                 INTO     LNAME-WORKC                              04060000
   1  040700                 COUNT IN LNAME-WORKL                              04070000
   1  040800*                                          **UNSTRING FIRST NAME  04080000
   1  040900              UNSTRING FNAME                                       04090000
   1  041000                 DELIMITED BY SPACE                                04100000
   1  041100                 INTO     FNAME-WORKC                              04110000
   1  041200                 COUNT IN FNAME-WORKL                              04120000
   1  041300*                                           **COUNT %'S           04130000
   1  041400              MOVE ZERO TO PERCENT-COUNTER                         04140000
   1  041500              INSPECT LNAME                                        04150000
   1  041600                 TALLYING PERCENT-COUNTER FOR ALL '%'              04160000
   1  041700              IF PERCENT-COUNTER > ZERO                            04170000
   1  041800*                                        **IF NO %'S THEN         04180000
   1  041900*                                        **LIST SPECIFIC NAME(S)  04190000
   1  042000*                                        **ELSE                   04200000
   1  042100*                                        **LIST GENERIC NAME(S)   04210000
   1  042200                 PERFORM LIST-GENERIC                              04220000
   1  042300              ELSE                                                 04230000
   1  042400                 PERFORM LIST-SPECIFIC.                            04240000
   1  042500/                                                                 04250000
   1  042600*************************************************                 04260000
   1  042700* LIST ALL EMPLOYEES                          *                   04270000
   1  042800*************************************************                 04280000
   1  042900 LIST-ALL.                                                        04290000
   1  043000*                                          **OPEN CURSOR          04300000
   1  043100           EXEC SQL OPEN TELE1 END-EXEC.                           04310000
   1  043200                                                                  04320000
```

```
        DATASET: DSN120.DSNSAMP                                          DATE: 87/02/04
        MEMBER:  DSN8BC3                                                 TIME: 08:33
                                                                         PAGE: 9
START
 COL    ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   1  043300*                                          **GET EMPLOYEES        04330000
   1  043400            EXEC SQL FETCH TELE1 INTO :PPHONE END-EXEC.            04340000
   1  043500                                                                   04350000
   1  043600            IF SQLCODE = NOT-FOUND                                 04360000
   1  043700*                                          **NO EMPLOYEE FOUND    04370000
   1  043800*                                          **PRINT ERROR MESSAGE  04380000
   1  043900              MOVE '0081' TO MSGCODE                               04390000
   1  044000              CALL 'DSN8MCG' USING MAJOR MSGCODE OUTMSG            04400000
   1  044100              MOVE OUTMSG TO OUTMSG2                               04410000
   1  044200              WRITE REPREC FROM MSG-REC2                           04420000
   1  044300                  AFTER ADVANCING 2 LINES                          04430000
   1  044400            ELSE                                                   04440000
   1  044500*                                          **LIST ALL EMPLOYEES   04450000
   1  044600              PERFORM PRINT-AND-GET1                               04460000
   1  044700                UNTIL SQLCODE IS NOT EQUAL TO ZERO.                04470000
   1  044800                                                                   04480000
   1  044900*                                          **CLOSE CURSOR         04490000
   1  045000            EXEC SQL CLOSE TELE1 END-EXEC.                         04500000
   1  045100                                                                   04510000
   1  045200 PRINT-AND-GET1.                                                   04520000
   1  045300            PERFORM PRINT-A-LINE.                                  04530000
   1  045400            EXEC SQL FETCH TELE1 INTO :PPHONE END-EXEC.            04540000
   1  045500/                                                                  04550000
   1  045600*****************************************************             04560000
   1  045700* LIST GENERIC EMPLOYEES                           *              04570000
   1  045800*****************************************************             04580000
   1  045900 LIST-GENERIC.                                                     04590000
   1  046000*                                          **OPEN CURSOR          04600000
   1  046100            EXEC SQL OPEN  TELE2 END-EXEC.                         04610000
   1  046200                                                                   04620000
   1  046300*                                          **GET EMPLOYEES        04630000
   1  046400            EXEC SQL FETCH TELE2 INTO :PPHONE END-EXEC.            04640000
   1  046500                                                                   04650000
   1  046600            IF SQLCODE = NOT-FOUND                                 04660000
   1  046700*                                          **NO EMPLOYEE FOUND    04670000
   1  046800*                                          **PRINT ERROR MESSAGE  04680000
   1  046900              MOVE '0081' TO MSGCODE                               04690000
   1  047000              CALL 'DSN8MCG' USING MAJOR MSGCODE OUTMSG            04700000
   1  047100              MOVE OUTMSG TO OUTMSG2                               04710000
   1  047200              WRITE REPREC FROM MSG-REC2                           04720000
   1  047300                  AFTER ADVANCING 2 LINES                          04730000
   1  047400            ELSE                                                   04740000
   1  047500*                                          **LIST GENERIC EMPLOYEE(S) 04750000
   1  047600              PERFORM PRINT-AND-GET2                               04760000
   1  047700                UNTIL SQLCODE IS NOT EQUAL TO ZERO.                04770000
   1  047800                                                                   04780000
   1  047900*                                          **CLOSE CURSOR         04790000
   1  048000            EXEC SQL CLOSE TELE2 END-EXEC.                         04800000
   1  048100                                                                   04810000
   1  048200 PRINT-AND-GET2.                                                   04820000
   1  048300            PERFORM PRINT-A-LINE.                                  04830000
   1  048400            EXEC SQL FETCH TELE2 INTO :PPHONE END-EXEC.            04840000
   1  048500/                                                                  04850000
   1  048600*****************************************************             04860000
```

```
      DATASET: OSN120.OSNSAMP                                    DATE: 87/02/04
      MEMBER:  OSN88C3                                           TIME: 08:33
                                                                 PAGE: 10
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
   1  048700* LIST SPECIFIC EMPLOYEES                          *          04870000
   1  048800*************************************************************04880000
   1  048900 LIST-SPECIFIC.                                               04890000
   1  049000*                                                **OPEN CURSOR04900000
   1  049100           EXEC SQL OPEN  TELE3 END-EXEC.                     04910000
   1  049200                                                              04920000
   1  049300*                                                **GET EMPLOYEES04930000
   1  049400           EXEC SQL FETCH TELE3 INTO :PPHONE END-EXEC.        04940000
   1  049500                                                              04950000
   1  049600           IF SQLCODE = NOT-FOUND                             04960000
   1  049700*                                                **NO EMPLOYEE FOUND04970000
   1  049800*                                                **PRINT ERROR MESSAGE04980000
   1  049900             MOVE '0081' TO MSGCODE                           04990000
   1  050000             CALL 'OSN88MCG' USING MAJOR MSGCODE OUTMSG       05000000
   1  050100             MOVE OUTMSG TO OUTMSG2                           05010000
   1  050200             WRITE REPREC FROM MSG-REC2                       05020000
   1  050300                 AFTER ADVANCING 2 LINES                      05030000
   1  050400           ELSE                                               05040000
   1  050500*                                          **LIST SPECIFIC EMPLOYEE(S)05050000
   1  050600             PERFORM PRINT-AND-GET3                           05060000
   1  050700                UNTIL SQLCODE IS NOT EQUAL TO ZERO.           05070000
   1  050800                                                              05080000
   1  050900*                                                **CLOSE CURSOR05090000
   1  051000           EXEC SQL CLOSE TELE3 END-EXEC.                     05100000
   1  051100                                                              05110000
   1  051200 PRINT-AND-GET3.                                              05120000
   1  051300           PERFORM PRINT-A-LINE.                              05130000
   1  051400           EXEC SQL FETCH TELE3 INTO :PPHONE END-EXEC.        05140000
   1  051500/                                                             05150000
   1  051600*************************************************************05160000
   1  051700* PRINT A LINE OF INFORMATION FROM DIRECTORY      *           05170000
   1  051800*************************************************************05180000
   1  051900 PRINT-A-LINE.                                                05190000
   1  052000*                                                **GET INFORMATION05200000
   1  052100           MOVE LASTNAMEC TO RLNAME.                          05210000
   1  052200           MOVE FIRSTNAMEC TO RFNAME.                         05220000
   1  052300           MOVE MIDDLEINITIAL TO RMIDINIT.                    05230000
   1  052400           MOVE PHONENUMBER OF PPHONE TO RPHONE.              05240000
   1  052500           MOVE EMPLOYEENUMBER OF PPHONE TO REMPNO.           05250000
   1  052600           MOVE DEPTNUMBER TO ROEPTNO.                        05260000
   1  052700           MOVE DEPTNAMEC TO ROEPTNAME.                       05270000
   1  052800*                                                **PRINT INFORMATION05280000
   1  052900           WRITE REPREC FROM REPDATA                          05290000
   1  053000              AFTER ADVANCING 2 LINES.                        05300000
   1  053100                                                              05310000
   1  053200           MOVE SPACES TO LASTNAMEC                           05320000
   1  053300                         FIRSTNAMEC                           05330000
   1  053400                         DEPTNAMEC.                           05340000
   1  053500/                                                             05350000
   1  053600*************************************************************05360000
   1  053700* UPDATES PHONE NUMBERS FOR EMPLOYEES             *           05370000
   1  053800*************************************************************05380000
   1  053900 TELEPHONE-UPDATE.                                            05390000
   1  054000           EXEC SQL UPDATE EMPLP                              05400000
```

```
      DATASET: DSN120.DSNSAMP                                                  DATE: 87/02/04
      MEMBER:  DSN8BC3                                                         TIME: 08:33
                                                                               PAGE: 11
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   1  054100                        SET   PHONENUMBER     = :NEWNO                 05410000
   1  054200                        WHERE EMPLOYEENUMBER = :ENO END-EXEC.          05420000
   1  054300           IF SQLCODE = ZERO                                           05430000
   1  054400*                                            **EMPLOYEE FOUND         05440000
   1  054500*                                            **UPDATE SUCCESSFUL      05450000
   1  054600*                                            **PRINT CONFIRMATION     05460000
   1  054700*                                            **MESSAGE                05470000
   1  054800              MOVE '0041' TO MSGCODE                                   05480000
   1  054900           ELSE                                                        05490000
   1  055000*                                             **NO EMPLOYEE FOUND     05500000
   1  055100*                                             **UPDATE FAILED         05510000
   1  055200*                                             **PRINT ERROR MESSAGE   05520000
   1  055300              MOVE '007E' TO MSGCODE.                                  05530000
   1  055400           CALL 'DSN8MCG' USING MAJOR MSGCODE OUTMSG.                  05540000
   1  055500           MOVE OUTMSG TO OUTMSG2.                                     05550000
   1  055600           WRITE REPREC FROM MSG-REC2                                  05560000
   1  055700                AFTER ADVANCING 2 LINES.                               05570000
   1  055800/                                                                     05580000
   1  055900************************************************                      05590000
   1  056000* SQL ERROR OCCURRED - GET ERROR MESSAGE           *                  05600000
   1  056100************************************************                      05610000
   1  056200 DBERROR.                                                           05620000
   1  056300*                                            **SQL ERROR              05630000
   1  056400*                                            **PRINT ERROR MESSAGE    05640000
   1  056500           MOVE '060E' TO MSGCODE                                      05650000
   1  056600           CALL 'DSN8MCG' USING MAJOR MSGCODE OUTMSG.                  05660000
   1  056700           MOVE OUTMSG TO OUTMSG1 OF  MSG-REC1.                        05670000
   1  056800           MOVE SQLCODE TO RETCODE OF MSG-REC1.                        05680000
   1  056900           WRITE REPREC FROM MSG-REC1                                  05690000
   1  057000              AFTER ADVANCING 2 LINES.                                 05700000
   1  057100           CALL 'DSNTIAR' USING SQLCA ERROR-MESSAGE ERROR-TEXT-LEN.    05710000
   1  057200           IF RETURN-CODE = ZERO                                       05720000
   1  057300              PERFORM ERROR-PRINT VARYING ERROR-INDEX                  05730000
   1  057400                 FROM 1 BY 1 UNTIL ERROR-INDEX GREATER THAN 8          05740000
   1  057500           ELSE                                                        05750000
   1  057600                                                                      05760000
   1  057700*                                            **MESSAGE FORMAT         05770000
   1  057800*                                            **ROUTINE ERROR          05780000
   1  057900*                                            **PRINT ERROR MESSAGE    05790000
   1  058000     **       MOVE '075E' TO MSGCODE                                   05800000
   1  058100              CALL 'DSN8MCG' USING MAJOR MSGCODE OUTMSG                05810000
   1  058200              MOVE OUTMSG TO OUTMSG1 OF MSG-REC1                       05820000
   1  058300              MOVE RETURN-CODE TO RETCODE OF MSG-REC1                  05830000
   1  058400              WRITE REPREC FROM MSG-REC1                               05840000
   1  058500                 AFTER ADVANCING 2 LINES.                              05850000
   1  058600                                                                      05860000
   1  058700*************************************************************         05870000
   1  058800* SQL RETURN CODE HANDLING WHEN PROCESSING CANNOT PROCEED *           05880000
   1  058900*************************************************************         05890000
   1  059000           EXEC SQL WHENEVER SQLERROR   CONTINUE     END-EXEC.       05900000
   1  059100           EXEC SQL WHENEVER SQLWARNING CONTINUE     END-EXEC.       05910000
   1  059200           EXEC SQL WHENEVER NOT FOUND  CONTINUE     END-EXEC.       05920000
   1  059300                                                                      05930000
   1  059400*                                            **PERFORM ROLLBACK       05940000
```

```
        DATASET: DSM120.DSMSAMP                                          DATE: 87/02/04
        MEMBER:  DSN8BC3                                                 TIME: 08:33
                                                                         PAGE: 12
 START
  COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

    1  059500              EXEC SQL ROLLBACK END-EXEC.                                 05950000
    1  059600                                                                          05960000
    1  059700              IF SQLCODE = ZERO                                           05970000
    1  059800                                                                          05980000
    1  059900*                                          **ROLLBACK SUCCESSFUL          05990000
    1  060000*                                          **PRINT CONFIRMATION           06000000
    1  060100*                                          **MESSAGE                      06010000
    1  060200                 MOVE '0531' TO MSGCODE                                   06020000
    1  060300              ELSE                                                        06030000
    1  060400                                                                          06040000
    1  060500*                                           **ROLLBACK FAILED             06050000
    1  060600*                                          **PRINT ERROR MESSAGE          06060000
    1  060700                 MOVE '061E' TO MSGCODE.                                  06070000
    1  060800              CALL 'DSN8MCG' USING MAJOR MSGCODE OUTMSG.                  06080000
    1  060900              MOVE OUTMSG TO OUTMSG1 OF MSG-REC1.                         06090000
    1  061000              MOVE SQLCODE TO RETCODE OF MSG-REC1.                        06100000
    1  061100              WRITE REPREC FROM MSG-REC1                                  06110000
    1  061200                    AFTER ADVANCING 2 LINES.                              06120000
    1  061300              GO TO PROG-END.                                             06130000
    1  061400                                                                          06140000
    1  061500*************************************************************          06150000
    1  061600*  PRINT MESSAGE TEXT                                       *          06160000
    1  061700*************************************************************          06170000
    1  061800 ERROR-PRINT.                                                          06180000
    1  061900              WRITE REPREC FROM ERROR-TEXT (ERROR-INDEX)                  06190000
    1  062000                AFTER ADVANCING 1 LINE.                                   06200000
```

```
                          SELECTING AN EMPLOYEE TO DISPLAY
MAJOR SYSTEM .....: O             ORGANIZATION
ACTION ...........: D             DISPLAY (SHOW)
OBJECT ...........: EM            EMEPLOYEE
SEARCH CRITERIA ..: EN            EMPLOYEE NAME
DATA .............: %

NO   D/ID   DEPARTMENT NAME                        E/ID     EMPLOYEE NAME

01   A00    SPIFFY COMPUTER SERVICE DIV.          000010   CI HASS
02   B01    PLANNING                              000020   ML THOMPSON
03   C01    INFORMATION CENTER                    000030   SA KWAN
04   E01    SUPPORT SERVICES                      000050   JB GEYER
05   D11    MANUFACTURING                         000060   IF STERN
06   D21    ADMINISTRATION SYSTEMS                000070   ED PULASKI
07   E11    OPERATIONS                            000090   EW HENDERSON
08   E21    SOFTWARE SUPPORT                      000100   TQ SPENSER
09   A00    SPIFFY COMPUTER SERVICE DIV.          000110   VG LUCCHESI
10   A00    SPIFFY COMPUTER SERVICE DIV.          000120   S  O'CONNELL
11   C01    INFORMATION CENTER                    000130   DM QUINTANA

PFK: 02=RESEND  03=END  08=NEXT
```

```
                              EMPLOYEE ADD
MAJOR SYSTEM . . .: O
ACTION . . . . . .: A
OBJECT . . . . . .: EM
SEARCH CRITERIA. .: EN
DATA . . . . . . .: 000030

  EMPLOYEE   ID              :  ______
             FIRST NAME      :  ____________
             MIDDLE INITIAL  :  _
             LAST NAME       :  _________
             WORK DEPT ID    :  ___
             PHONE NUMBER    :  ____


PFK: 02=RESEND 03=END
```

DB2 - Online program screens

```
                                    EMPLOYEE ERASE
MAJOR SYSTEM . . .: O
ACTION . . . . . .: E
OBJECT . . . . . .: EM
SEARCH CRITERIA. .: EN
DATA . . . . . .: 000030

  EMPLOYEE    ID                :  000030
              FIRST NAME        :  JANE_____________
              MIDDLE INITIAL    :  E
              LAST NAME         :  DOE_____________
              WORK DEPT ID      :  E21
              PHONE NUMBER      :  0000


PFK: 02=RESEND 03=END
```

```
                                    EMPLOYEE UPDATE
MAJOR SYSTEM . . .: O
ACTION . . . . . .: U
OBJECT . . . . . .: EM
SEARCH CRITERIA. .: EN
DATA . . . . . .: 000030

  EMPLOYEE    ID                :  000030
              FIRST NAME        :  JANE_____________
              MIDDLE INITIAL    :  E
              LAST NAME         :  DOE_____________
              WORK DEPT ID      :  E21
              PHONE NUMBER      :  0000


PFK: 02=RESEND 03=END
```

DB2 - Online program screens

```
DATASET: DSN120.DSN8M1P                                  DATE: 87/02/12
MEMBER:  DSN8MCG                                         TIME: 17:24
                                                         PAGE: 1
START
 COL    ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
      1 000100********************* DSN8MCG ******************************** 00010000
      1 000200*                                                            * 00020000
      1 000300*   MODULE NAME = DSN8MCG                                    * 00030000
      1 000400*                                                            * 00040000
      1 000500*   DESCRIPTIVE NAME = DB2  SAMPLE APPLICATION                * 00050000
      1 000600*                         MESSAGE ROUTINE                    * 00060000
      1 000700*                         COBOL                              * 00070000
      1 000800*                                                            * 00080000
      1 001200*                                                            * 00120000
      7       * COPYRIGHT = 5740-XYR (C) COPYRIGHT IBM CORP 1982, 1985      * 00122000
      7       * REFER TO COPYRIGHT INSTRUCTIONS FORM NUMBER G120-2083       * 00124000
      7       *                                                             * 00126000
      7       * STATUS = RELEASE 2, LEVEL 0                                * 00128000
      1 001300*   FUNCTION = THIS MODULE GIVEN A MESSAGE CODE, SENDS THE    * 00130000
      1 001400*              APPROPRIATE MESSAGE TO THE CALLING ROUTINE.    * 00140000
      1 001500*                                                            * 00150000
      1 001600*   NOTES = NONE                                             * 00160000
      1 001700*                                                            * 00170000
      1 001800*   MODULE TYPE = COBOL PROGRAM                              * 00180000
      1 001900*      PROCESSOR   = DB2 PRECOMPILER, COBOL COMPILER         * 00190000
      1 002000*      MODULE SIZE = SEE LINK EDIT                           * 00200000
      1 002100*      ATTRIBUTES  = NOT REENTRANT OR REUSABLE               * 00210000
      1 002200*                                                            * 00220000
      1 002300*                                                            * 00230000
      1 002400*   ENTRY POINT = DSN8MCG                                    * 00240000
      1 002500*      PURPOSE = SEE FUNCTION                                * 00250000
      1 002600*      LINKAGE = INVOKED FROM DSN RUN                        * 00260000
      1 002700*      INPUT   =                                             * 00270000
      1 002800*                                                            * 00280000
      1 002900*                SYMBOLIC LABEL/NAME =  MSGCODE              * 00290000
      1 003000*                DESCRIPTION         =  A MESSAGE CODE NO.   * 00300000
      1 003100*                                       PIC X(4)             * 00310000
      1 003200*                                                            * 00320000
      1 003300*                SYMBOLIC LABEL/NAME =  MAJOR                * 00330000
      1 003400*                DESCRIPTION         =  CALLING MODULE NAME  * 00340000
      1 003500*                                       PIC X(7)             * 00350000
      1 003600*      OUTPUT  =                                             * 00360000
      1 003700*                                                            * 00370000
      1 003800*                SYMBOLIC LABEL/NAME =  OUTMSG               * 00380000
      1 003900*                DESCRIPTION         =  A MESSAGE            * 00390000
      1 004000*                                       PIC X(69)            * 00400000
      1 004100*                                                            * 00410000
      1 004200*   EXIT-NORMAL = NONE                                       * 00420000
      1 004300*                                                            * 00430000
      1 004400*   EXIT-ERROR =                                             * 00440000
      1 004500*                                                            * 00450000
      1 004600*      RETURN CODE     =  NONE                               * 00460000
      1 004700*                                                            * 00470000
      1 004800*      ABEND CODES     =  NONE                               * 00480000
      1 004900*                                                            * 00490000
      1 005000/      ERROR-MESSAGES  =  ALL                                * 00500000
      1 005100*                                                            * 00510000
      1 005200*   EXTERNAL REFERENCES =                                     * 00520000
      1 005300*                                                            * 00530000
```

```
DATASET: DSN120.DSNSAMP                                        DATE: 87/02/12
MEMBER:  DSN8MCG                                               TIME: 17:24
                                                               PAGE: 2
START
COL    ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

       1  005400*       ROUTINES/SERVICES =  NONE                          * 00540000
       1  005500*                                                          * 00550000
       1  005600*       DATA-AREAS        =  NONE                          * 00560000
       1  005700*                                                          * 00570000
       1  005800*       CONTROL-BLOCKS    =  NONE                          * 00580000
       1  005900*                                                          * 00590000
       1  006000*   TABLES =  NONE                                         * 00600000
       1  006100*                                                          * 00610000
       1  006200*                                                          * 00620000
       1  006300*   CHANGE-ACTIVITY = NONE                                 * 00630000
       1  006400*                                                          * 00640000
       1  006500*                                                          * 00650000
       1  006600*  *PSEUDOCODE*                                           * 00660000
       1  006700*                                                          * 00670000
       1  006800*  PADCEDURE                                              * 00680000
       1  006900*    GET INPUT FROM CALLING ROUTINE                        * 00690000
       1  007000*    SEARCH CODE ARRAY                                     * 00700000
       1  007100*                                                          * 00710000
       1  007200*    IF CODES MATCH                                        * 00720000
       1  007300*       GET APPROPRIATE MESSAGE                            * 00730000
       1  007400*    ELSE                                                  * 00740000
       1  007500*       USE 'MESSAGE TEXT NOT FOUND' AS MESSAGE            * 00750000
       1  007600*                                                          * 00760000
       1  007700*    SEND MESSAGE TO CALLING ROUTINE                       * 00770000
       1  007800*  END.                                                    * 00780000
       1  007900*                                                          * 00790000
       1  008000*----------------------------------------------------------* 00800000
       1  008100/                                                           00810000
       1  004200 IDENTIFICATION DIVISION.                                   00820000
       1  008300*----------------------                                    00830000
       1  008400 PROGRAM-ID.  DSN8MCG                                       00840000
       1  008500                                                            00850000
       1  008600 ENVIRONMENT DIVISION.                                      00860000
       1  008700*----------------------                                    00870000
       1  008800                                                            00880000
       1  008900 DATA DIVISION.                                             00890000
       1  009000*-------------                                             00900000
       1  009100 WORKING-STORAGE SECTION.                                   00910000
       1  009200*                                  *OUTPUT MESSAGE          00920000
       1  009300    01 WORK-MSG.                                           00930000
       1  009400       02 HEAD-CODE          PIC X(04) VALUE 'DSN8'.       00940000
       1  009500       02 O-CODE             PIC X(04).                    00950000
       1  009600       02 FILLER1            PIC X(03) VALUE '   '.        00960000
       1  009700       02 O-MODULE           PIC X(07).                    00970000
       1  009800       02 DASH-SYM           PIC X(01) VALUE '-'.          00980000
       1  009900       02 O-MESSAGE          PIC X(50).                    00990000
       1  010000                                                           01000000
       1  010100    01 OMSG REDEFINES WORK-MSG PIC X(69).                  01010000
       1  010200                                                           01020000
       1  010300 LINKAGE SECTION.                                          01030000
       1  010400                                                           01040000
       1  010500*                                  *INPUT MESSAGE CODE      01050000
       1  010600    01 MSGCODE            PIC X(04).                      01060000
       1  010700                                                           01070000
```

```
START
 COL    ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

    I  010500*                                          *INPUT CALLING MODULE NAME  01080000
    1  010900    01 MAJOR                    PIC X(07).                             01090000
    1  011000                                                                       01100000
    1  011100*                                          *OUTPUT MESSAGE            01110000
    1  011200    01 OUTMSG                   PIC X(69).                             01120000
    1  011300                                                                       01130000
    1  011400*************************************************                     01140000
    1  011500*          MAIN PROGRAM ROUTINE                 *                     01150000
    1  011600*************************************************                     01160000
    1  011700                                                                       01170000
    1  011800 PROCEDURE DIVISION USING MAJOR MSGCODE OUTMSG.                        01180000
    1  011900*--------------------                                                 01190000
    1  012000 PROG-START.                                                          01200000
    1  012100*                                          **INITIALIZE OUTPUT        01210000
    1  012200       MOVE MSGCODE  TO O-CODE    OF WORK-MSG.                        01220000
    1  012300       MOVE MAJOR TO O-MODULE OF WORK-MSG.                            01230000
    1  012400       MOVE 'MESSAGE TEXT NOT FOUND                          '        01240000
    1  012500          TO O-MESSAGE OF WORK-MSG.                                   01250000
    1  012600                                                                       01260000
    1  012700                                                                       01270000
    1  012800*                                          **MESSAGE TEXT FOUND       01280000
    1  012900*                    * EMPLOYEE *                                      01290000
    1  013000*                                          *001I*                     01300000
    1  013100       IF MSGCODE EQUAL '001I' THEN                                   01310000
    1  013200       MOVE 'EMPLOYEE NOT FOUND                              ' TO     01320000
    1  013300       O-MESSAGE OF WORK-MSG.                                         01330000
    1  013400*                                          *002I*                     01340000
    1  013500       IF MSGCODE EQUAL '002I' THEN                                   01350000
    1  013600       MOVE 'EMPLOYEE SUCCESSFULLY ADDED                     ' TO     01360000
    1  013700       O-MESSAGE OF WORK-MSG.                                         01370000
    1  013800*                                          *003I*                     01380000
    1  013900       IF MSGCODE EQUAL '003I' THEN                                   01390000
    1  014000       MOVE 'EMPLOYEE SUCCESSFULLY ERASED                    ' TO     01400000
    1  014100       O-MESSAGE OF WORK-MSG.                                         01410000
    1  014200*                                          *004I*                     01420000
    1  014300       IF MSGCODE EQUAL '004I' THEN                                   01430000
    1  014400       MOVE 'EMPLOYEE SUCCESSFULLY UPDATED                   ' TO     01440000
    1  014500       O-MESSAGE OF WORK-MSG.                                         01450000
    1  014600*                                          *005E*                     01460000
    1  014700       IF MSGCODE EQUAL '005E' THEN                                   01470000
    1  014800       MOVE 'EMPLOYEE EXISTS ALREADY, ADD NOT DONE           ' TO     01480000
    1  014900       O-MESSAGE OF WORK-MSG.                                         01490000
    1  015000*                                          *006E*                     01500000
    1  015100       IF MSGCODE EQUAL '006E' THEN                                   01510000
    1  015200       MOVE 'EMPLOYEE DOES NOT EXIST, ERASE NOT DONE         ' TO     01520000
    1  015300       O-MESSAGE OF WORK-MSG.                                         01530000
    1  015400*                                          *007E*                     01540000
    1  015500       IF MSGCODE EQUAL '007E' THEN                                   01550000
    1  015600       MOVE 'EMPLOYEE DOES NOT EXIST, UPDATE NOT DONE        ' TO     01560000
    1  015700       O-MESSAGE OF WORK-MSG.                                         01570000
    1  015800*                                          *008I*                     01580000
    1  015900       IF MSGCODE EQUAL '008I' THEN                                   01590000
    1  016000       MOVE 'NO EMPLOYEE FOUND IN TABLE                      ' TO     01600000
    1  016100       O-MESSAGE OF WORK-MSG.                                         01610000
```

```
DATASET: DS4123.DSNSAMP                                        DATE: 87/02/12
MEMBER:  DSN3MC6                                               TIME: 17:24
                                                               PAGE: 4
START
 COL    ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
     1  016200                                                                  01620000
     1  016300                                                                  01630000
     1  016400*                    * DEPARTMENT *                               01640000
     1  016500*                                      *0111*                     01650000
     1  016600      IF MSGCODE EQUAL '011I' THEN                                01660000
     1  016700      MOVE 'DEPARTMENT NOT FOUND                          ' TO 01670000
     1  016800      O-MESSAGE OF WORK-MSG.                                      01680000
     1  016900*                                      *012I*                     01690000
     1  017000      IF MSGCODE EQUAL '012I' THEN                                01700000
     1  017100      MOVE 'DEPARTMENT SUCCESSFULLY ADDED                 ' TO 01710000
     1  017200      O-MESSAGE OF WORK-MSG.                                      01720000
     1  017300*                                      *0131*                     01730000
     1  017400      IF MSGCODE EQUAL '013I' THEN                                01740000
     1  017500      MOVE 'DEPARTMENT SUCCESSFULLY ERASED                ' TO 01750000
     1  017600      O-MESSAGE OF WORK-MSG.                                      01760000
     1  017700*                                      *0141*                     01770000
     1  017800      IF MSGCODE EQUAL '014I' THEN                                01780000
     1  017900      MOVE 'DEPARTMENT SUCCESSFULLY UPDATED               ' TO 01790000
     1  018000      O-MESSAGE OF WORK-MSG.                                      01800000
     1  018100*                                      *015E*                     01810000
     1  018200      IF MSGCODE EQUAL '015E' THEN                                01820000
     1  018300      MOVE 'DEPARTMENT EXISTS ALREADY, ADD NOT DONE       ' TO 01830000
     1  018400      O-MESSAGE OF WORK-MSG.                                      01840000
     1  018500*                                      *016E*                     01850000
     1  018600      IF MSGCODE EQUAL '016E' THEN                                01860000
     1  018700      MOVE 'DEPARTMENT DOES NOT EXIST, ERASE NOT DONE     ' TO 01870000
     1  018800      O-MESSAGE OF WORK-MSG.                                      01880000
     1  018900*                                      *017E*                     01890000
     1  019000      IF MSGCODE EQUAL '017E' THEN                                01900000
     1  019100      MOVE 'DEPARTMENT DOES NOT EXIST, UPDATE NOT DONE    ' TO 01910000
     1  019200      O-MESSAGE OF WORK-MSG.                                      01920000
     1  019300*                                      *018I*                     01930000
     1  019400      IF MSGCODE EQUAL '018I' THEN                                01940000
     1  019500      MOVE '''CURRENT'' DEPARTMENT NOT FOUND              ' TO 01950000
     1  019600      O-MESSAGE OF WORK-MSG.                                      01960000
     1  019700*                                      *019E*                     01970000
     1  019800      IF MSGCODE EQUAL '019E' THEN                                01980000
     1  019900      MOVE 'NO ''HIGHER'' DEPARTMENT EXISTS               ' TO 01990000
     1  020000      O-MESSAGE OF WORK-MSG.                                      02000000
     1  020100                                                                  02010000
     1  020200                                                                  02020000
     1  020300*                  * GENERAL INFO. MESSAGES *                     02030000
     1  020400*                                      *050I*                     02040000
     1  020500      IF MSGCODE EQUAL '050I' THEN                                02050000
     1  020600      MOVE 'PROGRAM STARTED                               ' TO 02060000
     1  020700      O-MESSAGE OF WORK-MSG.                                      02070000
     1  020800*                                      *051I*                     02080000
     1  020900      IF MSGCODE EQUAL '051I' THEN                                02090000
     1  021000      MOVE 'PROGRAM ENDED                                 ' TO 02100000
     1  021100      O-MESSAGE OF WORK-MSG.                                      02110000
     1  021200*                                      *052I*                     02120000
     1  021300      IF MSGCODE EQUAL '052I' THEN                                02130000
     1  021400      MOVE 'SQL WARNING, RETURN CODE IS:                  ' TO 02140000
     1  021500      O-MESSAGE OF WORK-MSG.                                      02150000
```

```
DATASET: DSH129.DSNSAMP                                        DATE: 87/02/12
MEMBER:  DSH8MCG                                               TIME: 17:24
                                                               PAGE: 5
STRRT
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
    1  021600*                                *0531*                              02160000
    1  021700      IF MSGCODE EQUAL '0531' THEN                                   02170000
    1  021800      MOVE 'ROLLBACK SUCCESSFUL, ALL UPDATES REMOVED          * TO   02180000
    1  021900      O-MESSAGE OF WORK-MSG.                                         02190000
    1  022000*                               *0561*                              02200000
    1  022100      IF MSGCODE EQUAL '0561' THEN                                   02210000
    1  022200      MOVE 'NO MORE DATA TO DISPLAY                           * TO   02220000
    1  022300      O-MESSAGE OF WORK-MSG.                                         02230000
    1  022400                                                                     02240000
    1  022500                                                                     02250000
    1  022600*           * GENERAL ERROR MESSAGES *                               02260000
    1  022700*                               *060E*                              02270000
    1  022800      IF MSGCODE EQUAL '060E' THEN                                   02280000
    1  022900      MOVE 'SQL ERROR, RETURN CODE IS:                        * TO   02290000
    1  023000      O-MESSAGE OF WORK-MSG.                                         02300000
    1  023100*                               *061E*                              02310000
    1  023200      IF MSGCODE EQUAL '061E' THEN                                   02320000
    1  023300      MOVE 'ROLLBACK FAILED, RETURN CODE IS:                  * TO   02330000
    1  023400      O-MESSAGE OF WORK-MSG.                                         02340000
    1  023500*                               *062E*                              02350000
    1  023600      IF MSGCODE EQUAL '062E' THEN                                   02360000
    1  023700      MOVE 'MISSING DETAIL MODULE                             * TO   02370000
    1  023800      O-MESSAGE OF WORK-MSG.                                         02380000
    1  023900*                               *063E*                              02390000
    1  024000      IF MSGCODE EQUAL '063E' THEN                                   02400000
    1  024100      MOVE 'MISSING SECONDARY SEL MODULE                      * TO   02410000
    1  024200      O-MESSAGE OF WORK-MSG.                                         02420000
    1  024300*                               *064E*                              02430000
    1  024400      IF MSGCODE EQUAL '064E' THEN                                   02440000
    1  024500      MOVE 'INVALID DL/I STC-CODE ON GU MSG                   * TO   02450000
    1  024600      O-MESSAGE OF WORK-MSG.                                         02460000
    1  024700*                               *065E*                              02470000
    1  024800      IF MSGCODE EQUAL '065E' THEN                                   02480000
    1  024900      MOVE 'INVALID DL/I STC-CODE ON ISRT MSG                 * TO   02490000
    1  025000      O-MESSAGE OF WORK-MSG.                                         02500000
    1  025100*                               *066E*                              02510000
    1  025200      IF MSGCODE EQUAL '066E' THEN                                   02520000
    1  025300      MOVE 'UNSUPPORTED PFK OR LOGIC ERROR                    * TO   02530000
    1  025400      O-MESSAGE OF WORK-MSG.                                         02540000
    1  025500*                               *067E*                              02550000
    1  025600      IF MSGCODE EQUAL '067E' THEN                                   02560000
    1  025700      MOVE 'UNSUPPORTED SEARCH CRITERIA FOR OBJECT            * TO   02570000
    1  025800      O-MESSAGE OF WORK-MSG.                                         02580000
    1  025900*                               *068E*                              02590000
    1  026000      IF MSGCODE EQUAL '068E' THEN                                   02600000
    1  026100      MOVE 'INVALID REQUEST, SHOULD BE **L** OR **U**         * TO   02610000
    1  026200      O-MESSAGE OF WORK-MSG.                                         02620000
    1  026300*                               *069E*                              02630000
    1  026400      IF MSGCODE EQUAL '069E' THEN                                   02640000
    1  026500      MOVE 'NO VALID SELECTIONS QUALIFY FOR THIS REQUEST      * TO   02650000
    1  026600      O-MESSAGE OF WORK-MSG.                                         02660000
    1  026700*                               *070E*                              02670000
    1  026800      IF MSGCODE EQUAL '070E' THEN                                   02680000
    1  026900      MOVE 'VITAL DATA IS MISSING IN TABLE **TOPTVAL**        * TO   02690000
```

```
START
 COL    ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
     1  027000      0-MESSAGE OF WORK-MSG.                                      02700000
     1  027100*                                  *072E*                         02710000
     1  027200      IF MSGCODE EQUAL '072E' THEN                                02720000
     1  027300      MOVE 'INVALID SELECTION ON SECONDARY SCREEN                 ' TO 02730000
     1  027400      0-MESSAGE OF WORK-MSG.                                      02740000
     1  027500*                                  *073E*                         02750000
     1  027600      IF MSGCODE EQUAL '073E' THEN                                02760000
     1  027700      MOVE 'SPECIFIED LINE-NUMBER NOT FOUND IN PREVIOUS SCREEN' TO 02770000
     1  027800      0-MESSAGE OF WORK-MSG.                                      02780000
     1  027900*                                  *074E*                         02790000
     1  028000      IF MSGCODE EQUAL '074E' THEN                                02800000
     1  028100      MOVE 'DATA IS TOO LONG FOR SEARCH CRITERIA                  ' TO 02810000
     1  028200      0-MESSAGE OF WORK-MSG.                                      02820000
     1  028300*                                  *075E*                         02830000
     1  028400      IF MSGCODE EQUAL '075E' THEN                                02840000
     1  028500      MOVE 'MESSAGE FORMAT ROUTINE ERROR, RETURN CODE IS:         ' TO 02850000
     1  028600      0-MESSAGE OF WORK-MSG.                                      02860000
     1  028700                                                                  02870000
     1  028800                                                                  02880000
     1  028900 PROG-END.                                                        02890000
     1  029000*                                  **RETURN TO CALLER             02900000
     1  029100*                                  **END OF PROGRAM               02910000
     1  029200      MOVE OMSG TO OUTMSG.                                        02920000
     1  029300         GOBACK.                                                  02930000
```

```
DATASET: OSN129.OSNSAMP                                   DATE: 87/02/12
MEMBER:  OSN8MCC2                                         TIME: 17:35
                                                          PAGE: 1
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
    7       * COMMAREA PART 2                                                00010000
                                                                             00020000
    8        1  WORK.                                                       00030000
    9         2 WRK          PIC X OCCURS 40.                              00040000
    8        1  LINE-SELECT.                                                00050000
    9         2 LINE-SELECT-C  PIC X(2).                                    00060000
    9         2 LINE-SELECT-P REDEFINES LINE-SELECT-C PIC 99.               00070000
    3        77 I              PIC 99 COMP.                                 00080000
    8        77 J              PIC 99 COMP.                                 00090000
    8        77 TOPLINE        PIC 99 COMP.                                 00100000
    8        77 TOPL-1         PIC 99 COMP.                                 00110000
    8        77 CUARLINE       PIC 99 COMP.                                 00120000
    8        77 CURRL-1        PIC 99 COMP.                                 00130000
    8        77 BOTLINE        PIC 99 COMP.                                 00140000
    8        77 CURSOR-VALUE PIC S9999 COMP.                                00150000
    8        77 MSG-INDEX      PIC X(4).                                    00160000
    8        77 SAVE-CONVID PIC X(16).                                      00170000
    8        77 HELPBIT         PIC X.                                     00180000
    8        77 SENDBIT         PIC X.                                     00190000
    8        77 ENDBIT          PIC X.                                     00200000
    8        77 NEXTBIT         PIC X.                                     00210000
    8        77 ON-I            PIC X.                                     00220000
    8        77 OFF-1           PIC X.                                     00230000
    8        77 PEACENT         PIC X  VALUE '%'.                          00240000
    8        77 OPTNF           PIC X(70)                                  00270000
   17                VALUE 'ENTRY MISSING IN TABLE TOPTVAL'.               00280000
    8        77 OSPNF           PIC X(70)                                  00290000
   17                VALUE 'ENTRY MISSING IN TABLE TOSPTXT'.               00300000
    8        77 DATA-LEN  PIC S9(9) COMP VALUE +79.                        00310000
    8        1  ERAGA-MESSAGE.                                             00320000
    9         2 RLEN     PIC S9(4) COMP VALUE +632.                       00330000
    9         2 ERROR-DATA.                                                00340000
   10          3 ERR-TEXT1  PIC X(79).                                     00350000
   10          3 ERR-TEXT2  PIC X(79).                                     00360000
   10          3 ERA-TEXT3  PIC X(79).                                     00370000
   10          3 ERA-TEXT4  PIC X(79).                                     00380000
   10          3 ERR-TEXT5  PIC X(79).                                     00390000
   10          3 ERR-TEXT6  PIC X(79).                                     00400000
   10          3 ERR-TEXT7  PIC X(79).                                     00410000
   10          3 ERR-TEXT8  PIC X(79).                                     00420000
    7       *                                                              00430000
    8        1  OSN8MCAE.                                                  00440000
    9         3 BLANK-DEPTNO       PIC X(3) VALUE SPACES.                 00450000
    9         3 BLANK-DEPTNAME.                                            00460000
   10          49 BLANK-DEPTNAMEL PIC S9(4) COMP-4 VALUE +36.              00470000
   10          49 BLANK-DEPTNAMEO PIC X(36) VALUE SPACES.                  00480000
    7       *                                                              00490000
    9         3 BLANK-FIRSTNME.                                            00500000
   10          49 BLANK-FIRSTNMEL PIC S9(4) COMP-4 VALUE +12.              00510000
   10          49 BLANK-FIRSTNMEO PIC X(12) VALUE SPACES.                  00520000
    9         3 BLANK-MIDINIT    PIC X(1) VALUE SPACES.                   00530000
    9         3 BLANK-LASTNAME.                                            00540000
   10          49 BLANK-LASTNAMEL PIC S9(4) COMP-4 VALUE +15.              00550000
   10          49 BLANK-LASTNAMEO PIC X(15) VALUE SPACES.                  00560000
```

```
START
COL ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
  7     ***** DSN8MC1 - SQL 1 COMMON MODULE FOR IMS AND CICS - COBOL **** 00010000
  7     *                                                               * 00020000
  7     * MODULE NAME = DSN8MC1                                         * 00030000
  7     *                                                               * 00040000
  7     * DESCRIPTIVE NAME = DB2 SAMPLE APPLICATION                     * 00050000
  7     *                    SQL 1 COMMON MODULE                        * 00060000
  7     *                    IMS & CICS                                 * 00070000
  7     *                    COBOL                                      * 00080000
  7     *                                                               * 00090000
  7     * COPYRIGHT = 5740-XYR (C) COPYRIGHT IBM CORP 1982, 1985        * 00100000
  7     * REFER TO COPYRIGHT INSTRUCTIONS FORM NUMBER G120-2083         * 00105000
  7     *                                                               * 00110000
  7     * STATUS = RELEASE 2, LEVEL 0                                   * 00120000
  7     *                                                               * 00140000
  7     * FUNCTION = RETRIEVES LAST CONVERSATION.                       * 00150000
  7     *            HANDLES 'RESEND' AND 'END' CASES.                  * 00160000
  7     *            CALLS VALIDATION ROUTINES DSN8MC3 THRU DSN8MC5.    * 00170000
  7     *            CALLS SQL2 ROUT (DSN8CC2 OR DSN8IC2).              * 00180000
  7     *                                                               * 00190000
  7     * NOTES = NONE                                                  * 00200000
  7     *                                                               * 00210000
  7     * MODULE TYPE =                                                 * 00220000
  7     *      PROCESSOR   = DB2 PRECOMPILER, VS COBOL                  * 00230000
  7     *      MODULE SIZE = SEE LINKEDIT                               * 00240000
  7     *      ATTRIBUTES  = REUSABLE                                   * 00250000
  7     *                                                               * 00260000
  7     * ENTRY POINT = DSN8MC1                                         * 00270000
  7     *      PURPOSE = SEE FUNCTION                                   * 00280000
  7     *      LINKAGE = INCLUDED BY MODULE DSN81C1 OR DSN8CC1          * 00290000
  7     *                                                               * 00300000
  7     *      INPUT = PARAMETERS EXPLICITLY PASSED TO THIS FUNCTION:   * 00310000
  7     *              SYMBOLIC LABEL/NAME =  NONE                      * 00320000
  7     *              DESCRIPTION = NONE                               * 00330000
  7     *                                                               * 00340000
  7     *      OUTPUT = PARAMETERS EXPLICITLY RETURNED:                 * 00350000
  7     *               SYMBOLIC LABEL/NAME =  NONE                     * 00360000
  7     *               DESCRIPTION = NONE                              * 00370000
  7     *                                                               * 00380000
  7     * EXIT-NORMAL = DROP THRU TO NEXT LINE OF CODE IN DSN8CP1/IP1    * 00390000
  7     *                                                               * 00400000
  7     * EXIT-ERROR = IF SQLERROR OR SQLWARNING, SQL WHENEVER          * 00410000
  7     *              CONDITION SPECIFIED IN DSN8CC1/IC1 WILL BE RAISED* 00420000
  7     *              AND PROGRAM WILL GO TO THE LABEL DB-ERROR.       * 00430000
  7     *                                                               * 00440000
  7     *                                                               * 00450000
  7     *      RETURN CODE = NONE                                       * 00460000
  7     *                                                               * 00470000
  7     *      ABEND CODES = NONE                                       * 00480000
  7     *                                                               * 00490000
  7     *      ERROR MESSAGES =                                         * 00500000
  7     *                       DSN8051I - PROGRAM ENDED                * 00510000
  7     *                                                               * 00520000
  7     * EXTERNAL REFERENCES = MOST VARIABLES ARE GLOBAL AND DEFINED    * 00530000
  7     *                       IN DSN8CC1/IC1.                         * 00540000
```

```
DATASET: DSN123.DSN5AMP                                        DATE: 87/02/12
MEMBER:  DSN8NC1                                               TIME: 17:36
                                                               PAGE: 2
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

       7       *     ROUTINES/SERVICES =                                    * 00550000
       7       *           INCLUDING DSN8NC3 THRU DSN8NC5.                  * 00560000
       7       *           DSN8NCG  - ERROR MESSAGE ROUTINE                 * 00570000
       7       *                                                            * 00580000
       7       *     DATA-AREAS =  NONE                                     * 00590000
       7       *                                                            * 00600000
       7       *     CONTROL-BLOCKS =                                       * 00610000
       7       *       SQLCA              - SQL COMMUNICATION AREA          * 00620000
       7       *                                                            * 00630000
       7       *     TABLES =  NONE                                         * 00640000
       7       *                                                            * 00650000
       7       *     CHANGE-ACTIVITY =  NONE                                * 00660000
       7       *                                                            * 00670000
       7       *                                                            * 00680000
       7       * *PSEUDOCODE*                                              * 00690000
       7       *                                                            * 00700000
       7       *  PROCEDURE                                                 * 00710000
       7       *   ELIMINATE LEADING BLANKS ON DATA LINE IF NOT ALL OF DATA  * 00720000
       7       *   LINE IS BLANK.                                           * 00730000
       7       *                                                            * 00740000
       7       *   SET UP CONTROL FLAGS FOR 'RESEND' 'END' 'NEXT'           * 00750000
       7       *       FIRST BY EXAMINING THE DATA LINE AND THEN COMPARING THE * 00760000
       7       *       PF KEYS (COMPARM.PFKIN)                              * 00770000
       7       *                                                            * 00780000
       7       *   RETRIEVE LAST CONVERSATION (FROM VCONA.)                 * 00790000
       7       *                                                            * 00800000
       7       *   IF LAST CONVERSATION IS NOT FOUND THEN DO.               * 00810000
       7       *       COMPARM.NEWCONV = 'Y'.                               * 00820000
       7       *       PCONVSTA        = '' .                               * 00830000
       7       *   END.                                                     * 00840000
       7       *                                                            * 00850000
       7       *   ELSE DO.                                                 * 00860000
       7       *       PCONVSTA = LAST CONVERSATION RETRIEVED.              * 00870000
       7       *       IF RESEND, BYPASS VALIDATION AND SAVE, JUST RESEND.  * 00880000
       7       *       IF END, DELETE CONVERSATION, SEND MESSAGE & GOTO CC1EXIT* 00890000
       7       *       IF NO SYSTEMS FIELD WAS CHANGED, BYPASS VALIDATION.  * 00900000
       7       *   END;                                                     * 00910000
       7       *                                                            * 00920000
       7       *   WHILE RETURN CODE IS 0 DO                                * 00930000
       7       *       CALL VALIDATION MODULES DSN8NC3 THRU DSN8NC5         * 00940000
       7       *   OTHERWISE                                                * 00950000
       7       *       GO TO NC1SAVE.                                       * 00960000
       7       *                                                            * 00970000
       7       *   GO TO CC1CALL IN DSN8CC1/IC1 TO CALL DSN8CC2/IC2.        * 00980000
       7       *                                                            * 00990000
       7       * NC1SAVE:                                                  * 01000000
       7       *   INSERT/UPDATE CURRENT CONVERSATION INTO VCONA.           * 01010000
       7       *                                                            * 01020000
       7       * END.                                                       * 01030000
       7       *                                                            * 01040000
       7       ****************************************************************** 01050000
       7       ****************************************************************** 01060000
       7       *                                                          *  01070000
       7       *          INITIAL EDITING FOR DATA INPUT                  *  01080000
```

```
START
 COL    ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7            *                                                                    01090000
   7            * 1. THE DATA LINE IS SHIFTED LEFT UNTIL ALL LEADING BLANKS         * 01100000
   7            *    HAVE BEEN ELIMINATED.                                         * 01110000
   7            * 2. THE APPROPRIATE BITS FOR 'RESEND','ENO' ETC. ARE THEN          * 01120000
   7            *    SET ACCORDING TO INPUT ON   DATA LINE.                        * 01130000
   7            * 3. IF PFKEYS 1,2, OR 8 HAS BEEN USED, THE APPROPRIATE BIT        * 01140000
   7            *     IS SET FOR 'RESEND' 'ENO' ETC..  THIS TAKES PRECEDENCE       * 01150000
   7            *     OVER THE SETTING OF THE SAME BITS IN STEP2.*                 * 01160000
   7            *                                                                  * 01170000
   7            *  *   I.E. IF SOMEONE TYPES IN 'RESEND' ON THE DATA LINE AND      * 01180000
   7            *      USES THE PF1 KEY AT THE SAME TIME, THE PF1 (END)            * 01190000
   7            *      FUNCTION IS ASSUMED TO BE THE ACTUAL REQUEST.               * 01200000
   7            ********************************************************************* 01210000
                                                                                     01220000
   8             OSNBMC1.                                                          01230000
                                                                                     01240000
   7            ********************************************************************* 01250000
   7            *                **INITIALIZE CONTROL FLAGS                         * 01260000
   7            ********************************************************************* 01270000
  12                 MOVE '0' TO OFF-1.                                            01280000
  12                 MOVE '1' TO ON-1.                                             01290000
  12                 MOVE '0' TO SENDBIT.                                          01300000
  12                 MOVE '0' TO ENDBIT.                                           01310000
  12                 MOVE '0' TO NEXTBIT.                                          01320000
                                                                                   01330000
  12                 MOVE 'OSNBMC1' TO MAJOR IN OSNB-MODULE-NAME.                  01340000
                                                                                   01350000
  12                 IF DATAIN IN INAREA = SPACE THEN GO TO MC1-18.                01360000
  12                 MOVE 0 TO I.                                                  01370000
   7            ********************************************************************* 01380000
   7            *                **GET RID OF LEADING BLANKS IN DATA                * 01390000
   7            ********************************************************************* 01400000
   8             MC1-10.                                                           01410000
   7            *                                          **SKIP LEADING BLANKS    01420000
  12                 ADD 1 TO I.                                                   01430000
   7            *                                          **IF ALL OF LINE         01440000
   7            *                                          **IS OF BLANKS           01450000
   7            *                                          **SEE IF A CONTROL       01460000
   7            *                                          **FLAG IS SET            01470000
  12                 IF I > 60 THEN GO TO MC1-18.                                  01480000
                                                                                   01490000
                                                                                   01500000
   7            *                                          **MC1-10 LOOP            01510000
   8             MC1-LOOP10.                                                       01520000
  12                 PERFORM MC1-10                                                01530000
  15                    UNTIL DATAIN(I) NOT = SPACE.                               01540000
                                                                                   01550000
                                                                                   01560000
   7            *                                          **IF FIRST CHARACTER IS  01570000
   7            *                                          **NON-BLANK, SEE IF A    01580000
   7            *                                          **CONTROL FLAG IS SET    01590000
  12                 IF I = 1 THEN                                                 01600000
  15                    GO TO MC1-18.                                              01610000
  12                 MOVE 1 TO J.                                                  01620000
```

```
DATASET: DSN!!0.DSNSAMP                                                  DATE: 87/02/12
MEMBER:  DSN8CC1                                                          TIME: 17:36
                                                                          PAGE: 4
START
 COL    ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

  8     MC1-12.
  7    *                                                 **GET NON-BLANK
  7    *                                                 **CHARACTERS
 12         MOVE DATAIN1(I) TO DATAIN1(J).
 12         ADD 1 TO I.
 12         ADD 1 TO J.

  7    *                                                 **MC1-12 LOOP
  8     MC1-LOOP12.
 12         PERFORM MC1-12
 15            UNTIL I > 60.

  8     MC1-14.
  7    *                                                 **PUT BLANKS AT END
  7    *                                                 **OF LINE
 12         MOVE SPACE TO DATAIN1(J).
 12         ADD 1 TO J.

  7    *                                                 **MC1-14 LOOP
  8     MC1-LOOP14.
 12         PERFORM MC1-14
 15            UNTIL J > 60.

  7    ******************************************************************
  7    *         **SET UP CONTROL FLAGS FOR 'RESEND' 'END' 'NEXT'
  7    ******************************************************************
  8     MC1-18.
 15            IF DATAIN = 'RESEND' OR
  7    *                                                   **RESEND COMMAND OR
  7    *                                                   **PF KEY 02
 18               PFKIN IN INAREA  = '02' THEN
 18               MOVE ON-1 TO SENDBIT
 15            ELSE
 18               IF DATAIN = 'END' OR
  7    *                                                   **END COMMAND OR
  7    *                                                   **PF KEY 03
 21                  PFKIN IN INAREA  = '03' THEN
 21                  MOVE ON-1 TO ENDBIT
 18               ELSE
 21                  IF DATAIN = 'NEXT' OR
  7    *                                                   **NEXT COMMAND OR
  7    *                                                   **PF KEY 08
 24                     PFKIN IN INAREA  = '08' THEN
 24                     MOVE ON-1 TO NEXTBIT.
  8     MC1-20.
  7    *******************************************************************
  7    *                                                                 *
  7    * RESTORE LAST MESSAGE AND DETERMINE IF VALIDATION IS NECESSARY *
  7    *                                                                 *
  7    * 1. ATTEMPT TO RETRIEVE LAST-MESSAGE STORED IN VCONA. IF        *
  7    *    NOT SUCCESSFUL, THEN CONVERSATION IS NEW.                    *
  7    * 2. IF RETRIEVAL IS SUCCESSFUL, THEN TRANSFER THE DATA          *
  7    *    INTO PCONVSTA.                                               *
```

```
DATASET: DSN120.DSNSAMP                                        DATE: 87/02/12
MEMBER:  DSNRNC1                                               TIME: 17:36
                                                               PAGE: 5
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        * 3. IF RESEND REQUEST,DON'T VALIDATE & DON'T SAVE,JUST RESEND  * 02170000
   7        * 4. IF END REQUEST,DELETE CONVERSATION,SEND END MESSAGE, EXIT   * 02180000
   7        * 5. IF ALL SYSTEM FIELDS HAVE NOT CHANGED SINCE THEY WERE      * 02190000
   7        *     LAST SAVED, BYPASS VALIDATION    ALSO.                    * 02200000
   7        * 6. OTHERWISE VALIDATE EACH OF THE SYSTEM FIELDS.              * 02210000
   7        ************************************************************** 02220000
                                                                            02230000
  12              MOVE 'N' TO NEWREQ IN COMPARM.                            02240000
  12              MOVE CONVID IN PCONVSTA TO SAVE-CONVID.                   02250000
  12              MOVE 'N' TO NEWCONV IN COMPARM.                           02260000
   7        *                                                               02270000
  12              EXEC SQL SELECT  *                                        02280000
  17                   INTO  :PCONA                                         02290000
  17                   FROM  VCONA                                          02300000
  17                   WHERE CONVID = :SAVE-CONVID END-EXEC.                02310000
                                                                            02320000
   7        *                                  **RETRIEVAL NOT SUCCESSFUL-  02330000
   7        *                                **INITIALIZE TO NEW CONVERSATION 02340000
                                                                            02350000
  12              IF SQLCODE = +100 THEN                                    02360000
  15                 MOVE 'Y' TO NEWCONV IN COMPARM                         02370000
  15                 MOVE SPACE TO PCONVSTA                                 02380000
  15                 MOVE SAVE-CONVID TO CONVID IN PCONVSTA                 02390000
  15                 MOVE SAVE-CONVID TO CONVID IN PCONA                    02400000
  15                 MOVE 'DSN8001 ' TO LASTSCR IN PCONVSTA                 02410000
  15                 GO TO NC1-VAL.                                         02420000
                                                                            02430000
   7        *                                        **RETRIEVAL SUCCESSFUL- 02440000
   7        *                                        **TRANSFER DATA TO PCONVSTA 02450000
                                                                            02460000
  12              MOVE LASTSCR  IN PCONA TO LASTSCR  IN PCONVSTA            02470000
  12              MOVE LASTPOS  IN PCONA TO LASTPOSO IN PCONVSTA            02480000
  12              MOVE LASTPOSC IN PCONA TO LASTPOSC IN PCONVSTA            02490000
  12              MOVE LASTMSG-TEXT  IN PCONA TO OUTAREA0.                  02500000
                                                                            02510000
   7        *                                    **IF CONVERSATION EXISTS BUT DATA 02520000
   7        *                                     **ENTERED FROM CLEARED SCREEN, 02530000
   7        *                                     **THEN TREAT LIKE RESEND   02540000
                                                                            02550000
  12              IF PFKIN IN INAREA = '00' OR                              02560000
  15                 SENDBIT = ON-1 THEN GO TO CC1-EXIT.                    02570000
                                                                            02580000
   7        *                                    **IF END REQUEST THEN DELETE CON- 02590000
   7        *                                     **VERSATION AND SEND END MESSAGE 02600000
                                                                            02610000
  12              IF ENDBIT NOT = ON-1 THEN GO TO NC1-30.                   02620000
  15                 MOVE 'DELETE  ' TO MINOR IN DSNS-MODULE-NAME           02630000
  15                 MOVE '1' TO EXITCODE                                   02640000
  15                 MOVE SPACE TO OUTAREA                                  02650000
  15                 EXEC SQL DELETE                                        02660000
  20                      FROM  VCONA                                       02670000
  20                      WHERE CONVID = :PCONA.CONVID END-EXEC              02680000
                                                                            02690000
   7        *                                          **PRINT MESSAGE:     02700000
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

  7        *                                                  **PROGRAM ENDED
 15                 MOVE 'DS1I' TO MSGCODE
 15                 CALL 'DSN8MCG' USING MAJOR MSGCODE OUTMSG
 15                 MOVE OUTMSG TO MSG IN OUTAREA.
 15                 MOVE MAJSYS IN INAREA  TO MAJSYS  IN OUTAREA.
 15                 GO TO CC1-EXIT.
  8         MC1-30.
  7        *
  7        *****************************************************************
  7        *                         **IF OLD CONVERSATION AND SYSTEM
  7        *                         **FIELDS HAVE NOT CHANGED THEN
  7        *                         **BYPASS VALIDATION
  7        *****************************************************************
  7        *                                                  ** NEW CONVERSATION
 12             IF ACTION IN INAREA NOT = ACTION IN OUTAREA OR
 15                 OBJECT IN INAREA NOT = OBJECT IN OUTAREA OR
 15                 SRCH   IN INAREA NOT = SRCH   IN OUTAREA THEN
 15                 GO TO MC1-VAL.

  7        *                                                  ** OLD CONVERSATION
 12             IF PREV IN PCONVSTA = 'D' AND
 15                 DATAIN IN INAREA NOT = DATAOUT IN OUTAREA THEN
 15                 MOVE 'Y' TO NEWREQ IN COMPARM.
 15                 GO TO MC1-BOTH.
  7        *****************************************************************
  7        *                         **VALIDATES FIELDS
  7        *****************************************************************
  8         MC1-VAL.
 12             MOVE 'Y' TO NEWREQ IN COMPARM.
 12             MOVE MAJSYS IN INAREA TO MAJSYS IN OUTAREA.
 12             MOVE OBJECT IN INAREA TO OBJECT IN OUTAREA.
 12             MOVE SRCH   IN INAREA TO SRCH   IN OUTAREA.
 12             MOVE SPACES           TO DESC3  IN OUTAREA.
 12             MOVE SPACES           TO DESC4  IN OUTAREA.

  7        *                                              **  VALIDATE ACTION
 12             PERFORM DSN8MC3 THRU END-DSN8MC3.
 12             IF RETCODE = '1' THEN GO TO MC1-SAVE.

  7        *                                              **  VALIDATE OBJECT
 12             PERFORM DSN8MC4 THRU END-DSN8MC4.
 12             IF RETCODE = '1' THEN GO TO MC1-SAVE.

  7        *                                              **  VALIDATE SEARCH
 12             PERFORM DSN8MC5 THRU END-DSN8MC5.
 12             IF RETCODE = '1' THEN GO TO MC1-SAVE.


  7        *****************************************************************
  7        *                 **IF ALL SYSTEM FIELDS ARE OK, CONTINUE
  7        *****************************************************************

  8         MC1-BOTH.

```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

  7        *                                           **NEW REQUEST
 15            IF ( PREV IN PCONVSTA = SPACE ) OR
 16               (( PREV IN PCONVSTA = 'S' ) AND
 15               ( DATA01 IN INAREA NOT = SPACE) AND
 16               ( NEXTBIT = OFF-1 )) THEN
 16               MOVE 'Y' TO NEWREQ IN COMPARM.

  7        *                                 **GO TO CC1-CALL WHERE A CALL
  7        *                                 **TO EITHER DSN8CC2 OR DSN8IC2
  7        *                                 **IS PERFORMED.

 13            GO TO CCI-CALL.
  7        ****************************************************************
  7        *                       **DSN8CC1 OR DSN8IC1 WILL
  7        *                       **BRANCH BACK TO MC1SAVE AFTER
  7        *                       **CALLING SQL2. AT MC1SAVE,
  7        *                       **THE DATA RETURNED BY SQL2 OR THE
  7        *                       **VALIDATION ROUTINES WILL BE
  7        *                       **SAVED IN VCONA
  7        ****************************************************************
  8         MC1-SAVE.
 13            MOVE DATAIN    IN INAREA     TO DATAOUT   IN OUTAREA.
 13            MOVE +1602                   TO LASTMSG-LEN.
 13            MOVE OUTAREAO                TO LASTMSG-TEXT.
 13            MOVE CONVID    IN PCONVSTA TO CONVID    IN PCONA.
 13            MOVE LASTSCR   IN PCONVSTA TO LASTSCR   IN PCONA.
 13            MOVE LASTPOSO  IN PCONVSTA TO LASTPOS   IN PCONA.
 13            MOVE LASTPOSC  IN PCONVSTA TO LASTPOSC  IN PCONA.
 13            MOVE 'MC1SAVE'               TO MINOR     IN DSN8-MODULE-NAME.

  7       *                                            **INSERT NEW VALUES
 13            IF NEWCONV = 'Y' THEN
 16               EXEC SQL INSERT
 21                    INTO   VCONA
 21                    VALUES (:PCONA) END-EXEC.

  7       *                                            **UPDATE OLD VALUES
 13            IF NEWCONV NOT = 'Y' THEN
 16               EXEC SQL UPDATE VCONA
 21                    SET    LASTSCR   = :PCONA.LASTSCR ,
 27                           LASTPOS   = :PCONA.LASTPOS ,
 27                           LASTPOSC  = :PCONA.LASTPOSC ,
 27                           LASTMSG   = :PCONA.LASTMSG
 21                    WHERE CONVID = :SAVE-CONVID END-EXEC.
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
       7  ***** DSN8MC1 - SQL 1 COMMON MODULE FOR IMS AND CICS - COBOL *** 00010000
       7  *                                                               * 00020000
       7  * MODULE NAME = DSN8MC1                                         * 00030000
       7  *                                                               * 00040000
       7  * DESCRIPTIVE NAME = DB2 SAMPLE APPLICATION                     * 00050000
       7  *                     SQL 1 COMMON MODULE                       * 00060000
       7  *                     IMS & CICS                                * 00070000
       7  *                     COBOL                                     * 00080000
       7  *                                                               * 00090000
       7  * COPYRIGHT = 5740-XYR (C) COPYRIGHT IBM CORP 1982, 1985        * 00100000
       7  * REFER TO COPYRIGHT INSTRUCTIONS FORM NUMBER G120-2083         * 00105000
       7  *                                                               * 00110000
       7  * STATUS = RELEASE 2, LEVEL 0                                   * 00120000
       7  *                                                               * 00140000
       7  * FUNCTION = RETRIEVES LAST CONVERSATION.                       * 00150000
       7  *            HANDLES 'RESEND' AND 'END' CASES.                  * 00160000
       7  *            CALLS VALIDATION ROUTINES DSN8MC3 THRU DSN8MC5.    * 00170000
       7  *            CALLS SQL2 ROOT (DSN3CC2 OR DSN8IC2).              * 00180000
       7  *                                                               * 00190000
       7  * NOTES =  NONE                                                 * 00200000
       7  *                                                               * 00210000
       7  * MODULE TYPE =                                                 * 00220000
       7  *      PROCESSOR    = DB2  PRECOMPILER, VS COBOL                * 00230000
       7  *      MODULE SIZE = SEE LINKEDIT                               * 00240000
       7  *      ATTRIBUTES  = REUSABLE                                   * 00250000
       7  *                                                               * 00260000
       7  * ENTRY POINT = DSN8MC1                                         * 00270000
       7  *      PURPOSE = SEE FUNCTION                                   * 00280000
       7  *      LINKAGE = INCLUDED BY MODULE DSN8IC1 OR DSN8CC1          * 00290000
       7  *                                                               * 00300000
       7  *      INPUT = PARAMETERS EXPLICITLY PASSED TO THIS FUNCTION:   * 00310000
       7  *              SYMBOLIC LABEL/NAME =  NONE                      * 00320000
       7  *              DESCRIPTION = NONE                               * 00330000
       7  *                                                               * 00340000
       7  *      OUTPUT = PARAMETERS EXPLICITLY RETURNED:                 * 00350000
       7  *               SYMBOLIC LABEL/NAME =  NONE                     * 00360000
       7  *               DESCRIPTION = NONE                              * 00370000
       7  *                                                               * 00380000
       7  * EXIT-NORMAL = DROP THRU TO NEXT LINE OF CODE IN DSN8CP1/IP1    * 00390000
       7  *                                                               * 00400000
       7  * EXIT-ERROR = IF SQLERROR OR SQLWARNING, SQL WHENEVER          * 00410000
       7  *              CONDITION SPECIFIED IN DSN8CC1/IC1 WILL BE RAISED* 00420000
       7  *              AND PROGRAM WILL GO TO THE LABEL D8-ERROR.       * 00430000
       7  *                                                               * 00440000
       7  *      RETURN CODE = NONE                                       * 00450000
       7  *                                                               * 00460000
       7  *      ABEND CODES = NONE                                       * 00470000
       7  *                                                               * 00480000
       7  *      ERROR MESSAGES =                                         * 00490000
       7  *                       DSN8051I -- PROGRAM ENDED               * 00500000
       7  *                                                               * 00510000
       7  * EXTERNAL REFERENCES = HOST VARIABLES ARE GLOBAL AND DEFINED    * 00520000
       7  *                       IN DSN8CC1/IC1.                         * 00530000
                                                                          * 00540000
```

```
CREATED: DSN120.DSN8SAMP                                  DATE: 87/02/12
MEMBER:  DSN8CC1                                          TIME: 17:18
                                                          PAGE: 1
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
   7      ********** DSN8CC1 - SQL 1 MAINLINE FOR CICS - COBOL **********  00010000
   7      *                                                                 00020000
   7      * MODULE NAME = DSN8CC1                                          00030000
   7      *                                                                 00040000
   7      * DESCRIPTIVE NAME = DB2  SAMPLE APPLICATION                     00050000
   7      *                    SQL 1 MAINLINE                               00060000
   7      *                    CICS                                         00070000
   7      *                    COBOL                                        00080000
   7      *                                                                 00090000
   7      * COPYRIGHT = 5740-XYR (C) COPYRIGHT IBM CORP 1982, 1985         00100000
   7      *  REFER TO COPYRIGHT INSTRUCTIONS FORM NUMBER G120-2083          00105000
   7      *                                                                 00110000
   7      * STATUS = RELEASE 2, LEVEL 0                                    00120000
   7      *                                                                 00130000
   7      * FUNCTION = THIS MODULE PERFORMS THE INCLUDES TO BRING IN THE    00140000
   7      *            SQL TABLE DCLS AND DCLGEN STRUCTURES AS WELL AS      00150000
   7      *            THE PARAMETER AREA.                                  00160000
   7      *                                                                 00170000
   7      *                                                                 00180000
   7      * NOTES =                                                        00190000
   7      *   DEPENDENCIES = CALLED BY DSN8CC0, CALLS DSN8CC2(CICS LINKS).  00200000
   7      *   RESTRICTIONS = NONE                                          00210000
   7      *                                                                 00220000
   7      *                                                                 00230000
   7      * MODULE TYPE =                                                  00240000
   7      *    PROCESSOR    = DB2 PRECOMPILER,CICS TRANSLATOR,COBOL COMPILER 00250000
   7      *    MODULE SIZE  = SEE LINK_EDIT                                 00260000
   7      *    ATTRIBUTES   = REUSABLE                                      00270000
   7      *                                                                 00280000
   7      * ENTRY POINT =  DSN8CC1                                         00290000
   7      *    PURPOSE = SEE FUNCTION                                       00300000
   7      *    LINKAGE = INCLUDED BY MODULE DSN8NC1                         00310000
   7      *                                                                 00320000
   7      *    INPUT  = PARAMETERS EXPLICITLY PASSED TO THIS FUNCTION:      00330000
   7      *                                                                 00340000
   7      *             SYMBOLIC LABEL/NAME = NONE                          00350000
   7      *             DESCRIPTION = NONE                                  00360000
   7      *                                                                 00370000
   7      *    OUTPUT  = PARAMETERS EXPLICITLY RETURNED:                    00380000
   7      *                                                                 00390000
   7      *              SYMBOLIC LABEL/NAME = NONE                         00400000
   7      *              DESCRIPTION = NONE                                 00410000
   7      *                                                                 00420000
   7      * EXIT-NORMAL = DSN8CC0                                          00430000
   7      *                                                                 00440000
   7      * EXIT-ERROR = DSN8CC0                                           00450000
   7      *                                                                 00460000
   7      *     RETURN CODE = NONE                                          00470000
   7      *                                                                 00480000
   7      *     ABEND CODES = NONE                                          00490000
   7      *                                                                 00500000
   7      *     ERROR-MESSAGES =  NONE                                      00510000
   7      *                                                                 00520000
   7      * EXTERNAL REFERENCES =                                          00530000
   7      *    ROUTINES/SERVICES =  DSN8CC2                                 00540000
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
      7  *                                                                  * 00550000
      7  *    DATA-AREAS =                                                  * 00560000
      7  *         DSN8MCCA          - COBOL STRUCTURE FOR DFHCOMMAREA      * 00570000
      7  *         DSN8MCCS          - VCONA TABLE DCL AND PCONA DCLGEN     * 00580000
      7  *         DSN8MCC2          - COMMON AREA PART 2                   * 00590000
      7  *         DSN8MCOV          - VOPTVAL TABLE DCL & POPTVAL DCLGEN   * 00600000
      7  *         DSN8MCVO          - FINDS VALID OPTIONS FOR ACTION,      * 00610000
      7  *                             OBJECT, SEARCH CRITERIA              * 00620000
      7  *         DSN8MC1           - SQL1 COMMON MODULE FOR IMS AND CICS  * 00630000
      7  *         DSN8MC3 - DSN8MC5 - VALIDATION MODULES CALLED BY DSN8MC0 * 00640000
      7  *         DSN8MCXX          - SQL ERROR HANDLER                    * 00650000
      7  *                                                                  * 00660000
      7  *    CONTROL-BLOCKS =                                              * 00670000
      7  *         SQLCA             - SQL COMMUNICATION AREA               * 00680000
      7  *                                                                  * 00690000
      7  * TABLES = NONE                                                    * 00700000
      7  *                                                                  * 00710000
      7  * CHANGE-ACTIVITY = NONE                                           * 00720000
      7  *                                                                  * 00730000
      7  *                                                                  * 00740000
      7  *  *PSEUDOCODE*                                                    * 00750000
      7  *                                                                  * 00760000
      7  *     PROCEDURE                                                    * 00770000
      7  *      INCLUDE DECLARATIONS.                                       * 00780000
      7  *      INCLUDE DSN8MC1.                                            * 00790000
      7  *      INCLUDE ERROR HANDLER.                                      * 00800000
      7  *                                                                  * 00810000
      7  *      CC1EXIT: ( REFERENCED BY DSN8MC1 )                          * 00820000
      7  *        EXEC CICS RETURN.                                         * 00830000
      7  *                                                                  * 00840000
      7  *      CC1CALL: ( REFERENCED BY DSN8MC1 )                          * 00850000
      7  *          EXEC CICS LINK PROGRAM('DSN8CC2')                       * 00860000
      7  *                         COMMAREA(DFHCOMMAREA).                   * 00870000
      7  *        GO TO MC1SAVE.  (LABEL IN DSN8MC1)                        * 00880000
      7  *                                                                  * 00890000
      7  *    INCLUDE VALIDATION MODULES.                                   * 00900000
      7  *                                                                  * 00910000
      7  *    END.                                                          * 00920000
      7  ****************************************************************** 00930000
      8   IDENTIFICATION DIVISION.                                         00940000
      7  *------------------------                                          00950000
      8   PROGRAM-ID. DSN8CC1.                                             00960000
                                                                           00970000
      8   ENVIRONMENT DIVISION.                                            00980000
      7  *---------------------                                           00990000
                                                                           01000000
      8   DATA DIVISION.                                                   01010000
      7  *-------------                                                   01020000
      8   WORKING-STORAGE SECTION.                                         01030000
      7  ****************************************************************** 01040000
      7  *      * DECLARE FIELD PASSED TO MESSAGE ROUTINE                 * 01050000
      7  *      * DECLARE CONVERSATION STATUS                             * 01060000
      7  *      * DECLARE MESSAGE TEXT                                    * 01070000
      7  *      * DECLARE OPTION VALIDATION                               * 01080000
```

```
START
COL ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
  7         *      * DECLARE COMMON AREA AND COMMON AREA PART 2              * 01090000
  7         ************************************************************** 01100000
  9          01 MSGCODE                  PIC X(04).                          01110000
                                                                             01120000
  9          01 OUTMSG                   PIC X(69).                          01130000
                                                                             01140000
 12              EXEC SQL INCLUDE DSN8MCCS END-EXEC.                         01150000
 12              EXEC SQL INCLUDE DSN8MCOV END-EXEC.                         01160000
 12              EXEC SQL INCLUDE SQLCA    END-EXEC.                         01170000
 12              EXEC SQL INCLUDE DSN8MCC2 END-EXEC.                         01180000
                                                                             01190000
  8         LINKAGE SECTION.                                                01200000
  8         01  DFHCOMMAREA.                                                01210000
 12              EXEC SQL INCLUDE DSN8MCCA END-EXEC.                         01220000
                                                                             01230000
  8         PROCEDURE DIVISION.                                             01240000
  7         *------------------------                                       01250000
                                                                             01260000
  7         ************************************************************* 01270000
  7         *  SQL RETURN CODE HANDLING                                   * 01280000
  7         ************************************************************* 01290000
 12              EXEC SQL WHENEVER SQLERROR GO TO DB-ERROR END-EXEC          01300000
 12              EXEC SQL WHENEVER SQLWARNING GO TO DB-ERROR END-EXEC.       01310000
                                                                             01320000
                                                                             01330000
 12              MOVE 'DSN8CC1' TO MAJOR IN DSN8-NODULE-NAME.                01340000
                                                                             01350000
  7         ************************************************************* 01360000
  7         *  FIND VALID OPTIONS FOR ACTION, OBJECT, SEARCH CRITERION*    01370000
  7         *  RETRIEVE CONVERSATION, VALIDATE, CALL SQL2                 * 01380000
  7         ************************************************************* 01390000
                                                                             01400000
 12              EXEC SQL INCLUDE DSN8MCVO END-EXEC.                         01410000
 12              EXEC SQL INCLUDE DSN8MC1 END-EXEC.                          01420000
                                                                             01430000
  7         *                                        **RETURN                01440000
  8         CC1-EXIT.                                                       01450000
 12              EXEC CICS RETURN END-EXEC.                                  01460000
                                                                             01470000
  7         ************************************************************* 01480000
  7         *  VALIDATE ACTION, OBJECT, SEARCH CRITERIA                   * 01490000
  7         *  HANDLE ERRORS                                              * 01500000
  7         ************************************************************* 01510000
                                                                             01520000
  8         CC1-CALL.                                                       01530000
 12              EXEC CICS LINK PROGRAM('DSN8CC2') COMMAREA(DFHCOMMAREA)     01540000
 21                    LENGTH(3000) END-EXEC.                                01550000
 12              GO TO MC1-SAVE.                                             01560000
 12              EXEC SQL INCLUDE DSN8MC3 END-EXEC.                          01570000
 12              EXEC SQL INCLUDE DSN8MC4 END-EXEC.                          01580000
 12              EXEC SQL INCLUDE DSN8MC5 END-EXEC.                          01590000
 12              EXEC SQL INCLUDE DSN8MCXX END-EXEC.                         01600000
 12              GOBACK.                                                     01610000
```

```
DATASET: DSN120.DSNSAMP                                   DATE: 87/02/12
MEMBER:  DSN8MC4                                          TIME: 17:37
                                                          PAGE: 1
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
   7  *******  DSN8MC4 - VALIDATION MODULE FOR OBJECT - COBOL *********  00010000
   7  *                                                                 *  00020000
   7  *      MODULE NAME = DSN8MC4                                      *  00030000
   7  *                                                                 *  00040000
   7  *      DESCRIPTIVE NAME = DB2 SAMPLE APPLICATION                  *  00050000
   7  *                         VALIDATION MODULE FOR OBJECT            *  00060000
   7  *                         COBOL                                   *  00070000
   7  *                                                                 *  00080000
   7  * COPYRIGHT = 5740-XYR (C) COPYRIGHT IBM CORP 1982, 1985          *  00090000
   7  * REFER TO COPYRIGHT INSTRUCTIONS FORM NUMBER G120-2083           *  00095000
   7  *                                                                 *  00100000
   7  * STATUS = RELEASE 2, LEVEL 0                                     *  00110000
   7  *                                                                 *  00130000
   7  *      FUNCTION = THIS MODULE VALIDATES SPECIFIC INPUT             *  00140000
   7  *                 AND MOVES IT TO THE OUTPUT MESSAGE              *  00150000
   7  *                 TOGETHER WITH A TEXT FIELD.                     *  00160000
   7  *                                                                 *  00170000
   7  *      NOTES = NONE                                               *  00180000
   7  *                                                                 *  00190000
   7  *                                                                 *  00200000
   7  *      MODULE TYPE =                                              *  00210000
   7  *          PROCESSOR    = DB2  PRECOMPILER, COBOL COMPILER        *  00220000
   7  *          MODULE SIZE = SEE LINKEDIT                             *  00230000
   7  *          ATTRIBUTES   = NONE                                    *  00240000
   7  *                                                                 *  00250000
   7  *      ENTRY POINT = DSN8MC4                                      *  00260000
   7  *          PURPOSE = SEE FUNCTION                                 *  00270000
   7  *          LINKAGE = INCLUDED BY MODULE DSN8CC1                   *  00280000
   7  *                                                                 *  00290000
   7  *          INPUT   = PARAMETERS EXPLICITLY PASSED                 *  00300000
   7  *                    TO THIS FUNCTION:                            *  00310000
   7  *                    SYMBOLIC LABEL/NAME =  NONE                  *  00320000
   7  *                    DESCRIPTION = NONE                           *  00330000
   7  *                                                                 *  00340000
   7  *          OUTPUT = PARAMETERS EXPLICITLY RETURNED:               *  00350000
   7  *                   SYMBOLIC LABEL/NAME =  NONE                   *  00360000
   7  *                   DESCRIPTION = NONE                            *  00370000
   7  *                                                                 *  00380000
   7  *      EXIT-NORMAL = THIS CODE IS "PERFORMED", SO IT EXITS TO     *  00390000
   7  *                    THE CODE FOLLOWING THE "PERFORM" STATEMENT    *  00400000
   7  *                                                                 *  00410000
   7  *      EXIT-ERROR = IF SQLERROR OR SQLWARNING, SQL WHENEVER       *  00420000
   7  *                   CONDITION SPECIFIED IN DSN3CC1/IC1 WILL BE    *  00430000
   7  *                   RAISED AND PROGRAM WILL GO TO THE LABEL       *  00440000
   7  *                   DB-ERROR.                                     *  00450000
   7  *                                                                 *  00460000
   7  *      RETURN CODE = NONE                                         *  00470000
   7  *                                                                 *  00480000
   7  *      ABEND CODES = NONE                                         *  00490000
   7  *                                                                 *  00500000
   7  *      ERROR MESSAGES =                                           *  00510000
   7  *           DSN8070E VITAL DATA MISSING IN TABLE 'TOPTVAL'        *  00520000
   7  *                                                                 *  00530000
   7  *      EXTERNAL REFERENCES = MOST VARIABLES ARE GLOBAL AND         *  00540000
```

```
START
 COL    ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

      7         *                                DEFINED IN DSN9CC1/IC1.          * 00550000
      7         *        ROUTINES/SERVICES =                                      * 00560000
      7         *          DSNAMCG           - ERROR MESSAGE ROUTINE              * 00570000
      7         *                                                                 * 00580000
      7         *        DATA-AREAS          =  NONE                            * 00590000
      7         *                                                                 * 00600000
      7         *        CONTROL-BLOCKS      =                                    * 00610000
      7         *          SQLCA             - SQL COMMUNICATION AREA             * 00620000
      7         *                                                                 * 00630000
      7         *      TABLES =  NONE                                             * 00640000
      7         *                                                                 * 00650000
      7         *      CHANGE-ACTIVITY =  NONE                                    * 00660000
      7         *                                                                 * 00670000
      7         *                                                                 * 00680000
      7         *     *PSEUDOCODE*                                                * 00690000
      7         *                                                                 * 00700000
      7         *      PROCEDURE                                                  * 00710000
      7         *      INITIALIZE RETURNCODE TO '0'.                              * 00720000
      7         *                                                                 * 00730000
      7         *      FILL IN THE DISPLAY AREA                                   * 00740000
      7         *      FROM VOPTVAL (OBJECT,SELTXT)                               * 00750000
      7         *           DEPENDING ON OBJECT REQUIRED                          * 00760000
      7         *           RETURN.                                               * 00770000
      7         *                                                                 * 00780000
      7         *      IF OBJECT NOT FOUND                                        * 00790000
      7         *          RETRIEVE A LIST OF OBJECTS WHICH EXISTS,               * 00800000
      7         *          HEADTXT, INFOTXT AND PFKTXT                            * 00810000
      7         *          FROM VOPTVAL                                           * 00820000
      7         *          DEPENDING ON MAJSYS = MAJSYS , ACTION = ACTION AND     * 00830000
      7         *                       OBJECT = BLANK                            * 00840000
      7         *          FILL IN DISPLAY AREA                                   * 00850000
      7         *          SET RETURNCODE TO '1'.                                 * 00860000
      7         *                                                                 * 00870000
      7         *  END.                                                           * 00880000
      7         *                                                                 * 00890000
      7         *-----------------------------------------------------------------* 00900000
      3          DSN8MC4.                                                         00910000
      7         ******************************************************************* 00920000
      7         *                                                                 * 00930000
      7         *                 *  INITIALIZE RETURN CODE                       * 00940000
      7         *                                                                 * 00950000
      7         ******************************************************************* 00960000
     12               MOVE '0' TO RETCODE.                                        00970000
     12               MOVE 'DSN8MC4' TO MAJOR IN DSN8-MODULE-NAME.                00980000
                                                                                  00990000
      7         ******************************************************************* 01000000
      7         *           ** LET'S SEE IF THE OBJECT SPECIFIED ON INPUT         * 01010000
      7         *           ** EXISTS BY TRYING TO RETRIEVE OBJECT AND TEXT       * 01020000
      7         ******************************************************************* 01030000
                                                                                  01040000
      7         *                                          **RETRIEVAL            * 01050000
     12               EXEC SQL SELECT SELTXT                                      01060000
     17                    INTO  :POPTVAL.SELTXT                                  01070000
     17                    FROM  VOPTVAL                                          01080000
```

```
DATASET: DSN120.DSNSAMP                                        DATE: 87/02/12
MEMBER:  DSNBMC4                                               TIME: 17:37
                                                               PAGE: 3
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

 17                 WHERE MAJSYS   = :INAREA.MAJSYS                          01090000
 17                 AND   ACTION   = :INAREA.ACTION                          01100000
 17                 AND   OBJECT   = :INAREA.OBJECT                          01110000
 17                 AND   OSJECT  <> '  '                                    01120000
 17                 AND   SRCNCRIT = '  '                                    01130000
 17                 END-EXEC.                                                01140000
                                                                             01150000
  7         *                                                                01160000
  7         *                                          **OBJECT EXISTS       01170000
 17                 MOVE OBJECT IN INAREA  TO OBJECT IN OUTAREA.             01180000
 17                 MOVE SELTXT IN POPTVAL TO DESC3  IN OUTAREA.             01190000
                                                                             01200000
  7         *                                          **RETURN              01210000
 12              IF   SQLCODE = +0  THEN                                     01220000
 17                   GO TO END-OSNBMC4.                                     01230000
                                                                             01240000
  7         ***************************************************************  01250000
  7         *              **      OBJECT NOT FOUND                        *  01260000
  7         *              **      PROVIDE A LIST OF OBJECTS WHICH EXIST   *  01270000
  7         ***************************************************************  01280000
 12              MOVE SPACE TO OBJECT IN OUTAREA.                            01290000
 12              MOVE SPACE TO DESC3  IN OUTAREA.                            01300000
                                                                             01310000
  7         *                                          **  OPEN CURSOR       01320000
 12              EXEC SQL OPEN V03 END-EXEC.                                 01330000
  7         *                                                                01340000
 12              MOVE +1 TO I.                                               01350000
                                                                             01360000
  8          MC4-10.                                                         01370000
  7         *                                          **RETRIEVE LIST       01380000
  7         *                                          **OF OBJECTS          01390000
 12              IF I NOT > 15                                               01400000
 15                THEN EXEC SQL FETCH V03 INTO :POPTVAL.OBJECT ,            01410000
 25                             :POPTVAL.SELTXT END-EXEC                     01420000
 20                     IF SQLCODE NOT EQUAL TO +100 THEN                    01430000
 28                          MOVE SPACES          TO FIELD-1(I)           01440000
 28                          MOVE OBJECT IN POPTVAL TO FIELD-2(I)           01450000
 28                          MOVE SELTXT IN POPTVAL TO FIELD-3(I)           01460000
 28                          ADD 1 TO I                                     01470000
 28                          GO TO MC4-10.                                  01480000
  8          MC4-20.                                                         01490000
  7         *                                          **CLOSE CURSOR        01500000
 12              EXEC SQL CLOSE V03 END-EXEC.                                01510000
  7         *                                                                01520000
 12              MOVE I TO J.                                                01530000
  7         *                                                                01540000
  7         *                                          **PUT BLANKS AT       01550000
  8          MC4-30.                                   **END OF LINE         01560000
 12              IF J NOT > 15 THEN MOVE SPACE TO LINEO(J)                   01570000
 17                   ADD 1 TO J                                             01580000
 17                   GO TO MC4-30.                                          01590000
  7         *                                                                01600000
  7         ***************************************************************  01610000
  7         *       **  CHECK FOR CONDITION WHERE THERE ARE NO VALID ENTRIES *  01620000
```

```
DATASET: DSN120.DSNSAMP                                    DATE: 87/02/12
MEMBER:  DSN8MC4                                           TIME: 17:37
                                                           PAGE: 4
START
 COL    ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
    7   ************************************************************** 01630000
                                                                       01640000
    7   *                                          **IF NO VALID ENTRY IN  01650000
    7   *                                          **OPTION VALIDATION     01660000
    7   *                                          **BASE TABLE (TOPTVAL)  01670000
    7   *                                          **TRY TO GET ERROR TEXT 01680000
   12       IF   I = 1 THEN MOVE '1' TO RETCODE                        01690000
    7   *                                              **ERROR TEXT FOUND  01700000
    7   *                                              **PRINT ERROR TEXT  01710000
   17            MOVE '070E' TO MSGCODE                                 01720000
   17            CALL 'DSN8MCG' USING MAJOR MSGCODE OUTMSG              01730000
   17            MOVE OUTMSG TO MSGTEXT IN MSG                          01740000
                                                                       01750000
    7   *                                                  **RETURN    01760000
   17            GO TO END-DSN8MC4.                                     01770000
    7   *                                                              01780000
    7   ************************************************************** 01790000
    7   *          ** IF ONLY ONE OBJECT EXISTS THEN USE IT AS DEFAULT  01800000
    7   *            ** SET UP OBJECT AND DESCRIPTION IN OUTPUT         01810000
    7   ************************************************************** 01820000
                                                                       01830000
   12       IF I = 2 AND OBJECT IN INAREA = ' ' THEN                   01840000
   15          MOVE '0' TO RETCODE                                     01850000
   15          MOVE FIELD-2(1) TO OBJECT IN INAREA                     01860000
   15          MOVE FIELD-2(1) TO OBJECT IN OUTAREA                    01870000
   15          MOVE FIELD-3(1) TO DESC3  IN OUTAREA                    01880000
   15          MOVE SPACE      TO LINE0 (1)                            01890000
    7   *                                                  **RETURN    01900000
   15            GO TO END-DSN8MC4.                                     01910000
                                                                       01920000
    7   ************************************************************** 01930000
    7   *                **   OBJECT WAS NOT FOUND                     01940000
    7   ************************************************************** 01950000
                                                                       01960000
   12       MOVE '1' TO RETCODE.                                       01970000
                                                                       01980000
   12       EXEC SQL SELECT *                                          01990000
   17            INTO  :POPTVAL                                        02000000
   17            FROM  VOPTVAL                                         02010000
   17            WHERE MAJSYS   = :INAREA.MAJSYS                       02020000
   17            AND   ACTION   = :INAREA.ACTION                       02030000
   17            AND   OBJECT   = '  '                                 02040000
   17            END-EXEC.                                             02050000
                                                                       02060000
    7   *                                          **FILL IN DISPLAY AREA  02070000
    7   *                                          **WITH HEADING, PFKEY   02080000
    7   *                                          **AND MESSAGE INFO.     02090000
                                                                       02100000
   12       MOVE HEADTXT IN POPTVAL TO HTITLE  IN OUTAREA.             02110000
   12       MOVE INFOTXT IN POPTVAL TO MSG     IN OUTAREA.             02120000
   12       MOVE PFKTXT  IN POPTVAL.TO PFKTEXT IN OUTAREA.             02130000
    7   *                                                              02140000
                                                                       02150000
    7   *                                              **RETURN  TO    02160000
```

DATASET: DSN120.SDSNSAMP
MEMBER: DSNXMC4

DATE: 87/02/12
TIME: 17:37
PAGE: 5

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
  7
  8          END-DSNXMC4.                    **DSNXMC1 MODULE      02170000
                                                                   02180000
```

```
START
COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
   7     *********DSN8MC3 - VALIDATION MODULE FOR ACTION - COBOL ********* 00010000
   7     *                                                                 * 00020000
   7     *       MODULE NAME = DSN8MC3                                     * 00030000
   7     *                                                                 * 00040000
   7     *       DESCRIPTIVE NAME = DB2 SAMPLE APPLICATION                 * 00050000
   7     *                          VALIDATION MODULE FOR ACTION           * 00060000
   7     *                          COBOL                                  * 00070000
   7     *                                                                 * 00080000
   7     * COPYRIGHT = 5740-XYR (C) COPYRIGHT IBM CORP 1982, 1985          * 00090000
   7     * REFER TO COPYRIGHT INSTRUCTIONS FORM NUMBER G120-2083           * 00095000
   7     *                                                                 * 00100000
   7     * STATUS = RELEASE 2, LEVEL 0                                     * 00110000
   7     *                                                                 * 00120000
   7     *       FUNCTION = THIS MODULE VALIDATES SPECIFIC INPUT           * 00130000
   7     *                  AND MOVES IT TO THE OUTPUT MESSAGE             * 00140000
   7     *                  TOGETHER WITH 4 TEXT FIELD.                    * 00150000
   7     *                                                                 * 00160000
   7     *       NOTES =   NONE                                            * 00170000
   7     *                                                                 * 00180000
   7     *                                                                 * 00190000
   7     *                                                                 * 00200000
   7     *       MODULE TYPE = BLOCK OF COBOL CODE                         * 00210000
   7     *       PROCESSOR   = DB2  PRECOMPILER, COBOL COMPILER            * 00220000
   7     *       MODULE SIZE = SEE LINKEDIT                                * 00230000
   7     *       ATTRIBUTES  = REUSABLE                                    * 00240000
   7     *                                                                 * 00250000
   7     *       ENTRY POINT = DSN8MC3                                     * 00260000
   7     *       PURPOSE = SEE FUNCTION                                    * 00270000
   7     *       LINKAGE = INCLUDED BY MODULE DSN8CC1                      * 00280000
   7     *                                                                 * 00290000
   7     *     INPUT = PARAMETERS EXPLICITLY PASSED TO THIS FUNCTION:      * 00300000
   7     *             SYMBOLIC LABEL/NAME =  NONE                         * 00310000
   7     *             DESCRIPTION = NONE                                  * 00320000
   7     *                                                                 * 00330000
   7     *       OUTPUT = PARAMETERS EXPLICITLY RETURNED:                  * 00340000
   7     *                SYMBOLIC LABEL/NAME =  NONE                      * 00350000
   7     *                DESCRIPTION = NONE                               * 00360000
   7     *                                                                 * 00370000
   7     *   EXIT-NORMAL = THIS CODE IS "PERFORMED", SO IT EXITS TO CODE   * 00380000
   7     *                 FOLLOWING THE "PERFORM" STATEMENT               * 00390000
   7     *                                                                 * 00400000
   7     *   EXIT-ERROR = IF SQLERROR OR SQLWARNING, SQL WHENEVER CON-     * 00410000
   7     *                DITION  SPECIFIED IN DSN8CC1/IC1 WILL BE RAISED  * 00420000
   7     *                AND PROGRAM WILL GO TO THE LABEL DB-ERROR.       * 00430000
   7     *                                                                 * 00440000
   7     *                                                                 * 00450000
   7     *       RETURN CODE = NONE                                        * 00460000
   7     *                                                                 * 00470000
   7     *       ABEND CODES = NONE                                        * 00480000
   7     *                                                                 * 00490000
   7     *       ERROR MESSAGES =                                          * 00500000
   7     *        DSN8070E - VITAL DATA IS MISSING IN TABLE 'TOPTVAL'      * 00510000
   7     *                                                                 * 00520000
   7     *   EXTERNAL REFERENCES = MOST VARIABLES ARE GLOBAL AND DEFINED   * 00530000
   7     *                         IN DSN8CC1/IC1.                         * 00540000
```

```
START
 CDL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
       7       *       ROUTINES/SERVICES =                                      * 00550000
       7       *          DSN#MCG              - ERROR MESSAGE ROUTINE          * 00560000
       7       *                                                                * 00570000
       7       *       DATA-AREAS          = NONE                               * 00580000
       7       *                                                                * 00590000
       7       *       CONTROL-BLOCKS      =                                    * 00600000
       7       *          SQLCA                - SQL COMMUNICATION AREA         * 00610000
       7       *                                                                * 00620000
       7       *    TABLES = NONE                                               * 00630000
       7       *                                                                * 00640000
       7       *    CHANGE-ACTIVITY = NONE                                      * 00650000
       7       *                                                                * 00660000
       7       *                                                                * 00670000
       7       *  *PSEUDOCODE*                                                  * 00680000
       7       *                                                                * 00690000
       7       *    PROCEDURE                                                   * 00700000
       7       *      INITIALIZE RETURNCODE TO '0'.                             * 00710000
       7       *                                                                * 00720000
       7       *      FILL IN THE DISPLAY AREA                                  * 00730000
       7       *          FROM VOPTVAL (ACTION,SELTXT)                          * 00740000
       7       *               DEPENDING ON REQUIRED MAJSYS AND ACTION          * 00750000
       7       *          RETURN.                                               * 00760000
       7       *                                                                * 00770000
       7       *      IF ACTION NOT FOUND                                       * 00780000
       7       *          RETRIEVE A LIST OF ACTION WHICH EXISTS,               * 00790000
       7       *          HEADTXT, INFOTXT AND PFKTXT FILL IN DISPLAY AREA      * 00800000
       7       *          FROM VOPTVAL                                          * 00810000
       7       *          DEPENDING ON MAJSYS = MAJSYS AND ACTION = BLANK       * 00820000
       7       *          SET RETURNCODE TO '1'.                                * 00830000
       7       *                                                                * 00840000
       7       * END.                                                           * 00850000
       7       *                                                                * 00860000
       7       *--------------------------------------------------------------* 00870000
       8        DSN8MC3.                                                        00880000
                                                                                00890000
                                                                                00900000
       7       ****************************************************************  00910000
       7       *                  ** INITIALIZE RETURN CODE                   *  00920000
       7       ****************************************************************  00930000
      12           MOVE '0' TO RETCODE.                                         00940000
      12           MOVE 'DSN8MC3' TO MAJOR IN DSN8-MODULE-NAME.                 00950000
                                                                                00960000
       7       ****************************************************************  00970000
       7       *       ** LET'S SEE IF THE ACTION SPECIFIED ON INPUT EXISTS     00980000
       7       *       **BY TRYING TO RETRIEVE ACTION AND TEXT.                 00990000
       7       ****************************************************************  01000000
                                                                                01010000
       7       *                                          **RETRIEVAL           01020000
      12           EXEC SQL SELECT SELTXT                                       01030000
      17                INTO  :PDPTVAL.SELTXT                                   01040000
      17                FROM  VOPTVAL                                           01050000
      17                WHERE MAJSYS   = :INAREA.MAJSYS                         01060000
      17                AND   ACTION   = :INAREA.ACTION                         01070000
      17                AND   ACTION  ¬= ' '                                    01080000
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

  17                 AND   OBJECT   = ' '                                        01090000
  17                 END-EXEC.                                                  01100000
                                                                                01110000
   7     *                                              **ACTION EXISTS         01120000
   7     *                                              **FILL IN DISPLAY AREA  01130000
  16                 MOVE ACTION IN INAREA  TO ACTION IN OUTAREA.               01140000
  16                 MOVE SELTXT IN POPTVAL TO DESC2  IN OUTAREA.               01150000
                                                                                01160000
   7     *                                              **RETURN                01170000
  12             IF   SQLCODE = +0  THEN                                        01180000
  16                 GO TO END-DSN8MC3.                                         01190000
   7     ***************************************************************       01200000
   7     *              ** ACTION NOT FOUND                                      01210000
   7     *              ** PROVIDE A LIST OF ACTIONS WHICH EXIST                 01220000
   7     ***************************************************************       01230000
  12             MOVE SPACE TO ACTION IN OUTAREA.                               01240000
  12             MOVE SPACE TO DESC2  IN OUTAREA.                               01250000
                                                                                01260000
   7     *                                          **   OPEN CURSOR            01270000
  12             EXEC SQL OPEN VO2 END-EXEC.                                    01280000
   7     *                                                                      01290000
  12             MOVE +1 TO I.                                                  01300000
   7     *                                          **RETRIEVE LIST OF ACTIONS  01310000
   8     MC3-10.                                                                01320000
  12             IF I NOT > 15                                                  01330000
  15                THEN EXEC SQL FETCH VO2 INTO :POPTVAL.ACTION ,              01340000
  25                          :POPTVAL.SELTXT END-EXEC                          01350000
                                                                                01360000
  20                      IF SQLCODE NOT = +100 THEN                            01370000
  26                         MOVE SPACES           TO FIELD-1(I)              01380000
  26                         MOVE ACTION IN POPTVAL TO FIELD-2(I)              01390000
  26                         MOVE SELTXT IN POPTVAL TO FIELD-3(I)              01400000
  26                         ADD 1 TO I                                        01410000
  26                         GO TO MC3-10.                                     01420000
   8     MC3-20.                                                                01430000
   7     *                                              **CLOSE CURSOR          01440000
  12             EXEC SQL CLOSE VO2 END-EXEC.                                   01450000
   7     *                                                                      01460000
  12             MOVE I TO J.                                                   01470000
                                                                                01480000
   7     *                                          **PUT BLANKS AT             01490000
   7     *                                          **END OF LINE               01500000
   8     MC3-30.                                                                01510000
  12             IF J NOT > 15 THEN MOVE SPACE TO LINEO(J)                      01520000
  17                ADD 1 TO J                                                  01530000
  17                GO TO MC3-30.                                               01540000
   7     *                                                                      01550000
   7     ***************************************************************       01560000
   7     *      **  CHECK FOR CONDITION WHERE THERE ARE NO VALID ENTRIES */     01570000
   7     ***************************************************************       01580000
                                                                                01590000
   7     *                                          **NO VALID ENTRY IN         01600000
   7     *                                          **OPTION VALIDATION         01610000
   7     *                                          **BASE TABLE (TOPTVAL)      01620000
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
  7       *                                          **TRY TO GET ERROR TEXT 01630000
 12            IF   I = 1 THEN MOVE '1' TO RETCODE                           01640000
                                                                             01650000
 22                  MOVE '070E' TO MSGCODE                                  01660000
 22                  CALL 'DSN8MC6' USING MAJOR MSGCODE OUTMSG               01670000
 22                  MOVE OUTMSG TO MSGTEXT IN MSG                           01680000
                                                                             01690000
  7       *                                          **RETURN               01700000
 22                  GO TO END-DSN8MC3.                                      01710000
  7       *                                                                  01720000
  7       ************************************************************       01730000
  7       *          **  IF ONLY THE ACTION EXISTS THEN USE AS DEFAULT       01740000
  7       *          **  SET UP ACTION AND DESCRIPTION IN OUTPUT             01750000
  7       ************************************************************       01760000
                                                                             01770000
 12            IF I = 2 AND ACTION IN INAREA = ' ' THEN                      01780000
 15               MOVE '0' TO RETCODE                                        01790000
 15               MOVE FIELD-2(1) TO ACTION IN INAREA                        01800000
 15               MOVE FIELD-2(1) TO ACTION IN OUTAREA                       01810000
 15               MOVE FIELD-3(1) TO DESC2  IN OUTAREA                       01820000
 15               MOVE SPACE      TO LINE0 (1)                               01830000
  7       *                                          **RETURN               01840000
 15               GO TO END-DSN8MC3.                                         01850000
                                                                             01860000
  7       ************************************************************       01870000
  7       *          **  ACTION WAS NOT FOUND                               01880000
  7       ************************************************************       01890000
                                                                             01900000
 12            MOVE '1' TO RETCODE.                                          01910000
                                                                             01920000
 12            EXEC SQL SELECT *                                             01930000
 17                 INTO  :POPTVAL                                          01940000
 17                 FROM  VOPTVAL                                           01950000
 17                 WHERE MAJSYS   = :INAREA.MAJSYS                         01960000
 17                 AND   ACTION   = ' '                                    01970000
 17                 END-EXEC.                                               01980000
                                                                             01990000
  7       *                                          **FILL DISPLAY AREA    02000000
  7       *                                          **WITH HEADING, PFKEY, 02010000
  7       *                                          **& MESSAGE INFO.      02020000
                                                                             02030000
 15               MOVE HEADTXT IN POPTVAL TO HTITLE  IN OUTAREA.             02040000
 15               MOVE INFOTXT IN POPTVAL TO MSG     IN OUTAREA.             02050000
 15               MOVE PFKTXT  IN POPTVAL TO PFKTEXT IN OUTAREA.             02060000
  7       *                                                                  02070000
                                                                             02080000
  7       *                                          **RETURN TO            02090000
  7       *                                          **DSN8MC1 MODULE       02100000
  8        END-DSN8MC3.                                                      02110000
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
    7 **** DSN8CCO - SUBSYSTEM INTERFACE MODULE FOR CICS/VS - COBOL *** 00010000
    7 *                                                                00020000
    7 *   MODULE NAME = DSN8CCO                                        00030000
    7 *                                                                00040000
    7 *   DESCRIPTIVE NAME = DB2  SAMPLE APPLICATION                   00050000
    7 *                      SUBSYSTEM INTERFACE MODULE                00060000
    7 *                      CICS                                      00070000
    7 *                      COBOL                                     00080000
    7 *                                                                00090000
    7 *                                                                00100000
    7 * COPYRIGHT = 5740-XYR (C) COPYRIGHT IBM CORP 1982, 1985         00110000
    7 * REFER TO COPYRIGHT INSTRUCTIONS FORM NUMBER G120-2083          00115000
    7 *                                                                00120000
    7 * STATUS = RELEASE 2, LEVEL 0                                   00130000
    7 *                                                                00150000
    7 *   FUNCTION =  THIS MODULE ISSUES CICS RECEIVE MAP TO RETRIEVE  00160000
    7 *               INPUT, CALLS DSN8CC1, AND ISSUE CICS SEND MAP    00170000
    7 *               AFTER RETURNING.                                 00180000
    7 *                                                                00190000
    7 *   NOTES =                                                      00200000
    7 *      1. THIS IS A CICS PSEUDO CONVERSATION PROGRAM WHICH       00210000
    7 *         INITIALIZES ITSELF WHEN A TERMINAL OPERATOR ENTERS     00220000
    7 *         INPUT AFTER VIEWING THE SCREEN SENT BY PREVIOUS        00230000
    7 *         ITERATIONS OF THE PROGRAM.                             00240000
    7 *                                                                00250000
    7 *      DEPENDENCIES = TWO CICS MAPS(OBJECTS) ARE REQUIRED:       00260000
    7 *                                 DSN8MCNG AND DSN8MCNO          00270000
    7 *                     MODULE  DSN8CC1 IS REQUIRED.               00280000
    7 *                     DCLGEN STRUCTURE DSN8MCCS IS REQUIRED      00290000
    7 *                     INCLUDED COBOL STRUCTURE DSN8MCCA IS       00300000
    7 *                     REQUIRED.                                  00310000
    7 *                                                                00320000
    7 *      RESTRICTIONS = NONE                                       00330000
    7 *                                                                00340000
    7 *   MODULE TYPE =                                                00350000
    7 *      PROCESSOR    = DB2 PRECOMPILER,CICS TRANSLATOR,COBOL COMPIL 00360000
    7 *      MODULE SIZE = SEE LINK-EDIT                               00370000
    7 *      ATTRIBUTES  = REUSABLE                                    00380000
    7 *                                                                00390000
    7 *   ENTRY POINT = DSN8CCO                                        00400000
    7 *      PURPOSE = SEE FUNCTION                                    00410000
    7 *      LINKAGE = CICS/OS/VS ENTRY                                00420000
    7 *                                                                00430000
    7 *      INPUT = PARAMETERS EXPLICITLY PASSED TO THIS FUNCTION:    00440000
    7 *                                                                00450000
    7 *            SYMBOLIC LABEL/NAME = DSN8CCGI                      00460000
    7 *            DESCRIPTION = CICS/OS/VS BMS MAP FOR GENERAL INPUT  00470000
    7 *                                                                00480000
    7 *            SYMBOLIC LABEL/NAME = DSN8CCOI                      00490000
    7 *            DESCRIPTION = CICS/OS/VS BMS MAP FOR DETAIL INPUT   00500000
    7 *                                                                00510000
    7 *      OUTPUT = PARAMETERS EXPLICITLY RETURNED:                  00520000
    7 *                                                                00530000
                                                                       00540000
```

```
DATASET: DSN120.DSNSAMP                                        DATE: 87/02/12
MEMBER:  DSN8CCO                                               TIME: 17:17
                                                               PAGE: 2
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7      *              SYMBOLIC LABEL/NAME = DSN8CCGO                          00550000
   7      *              DESCRIPTION = CICS/OS/VS BMS MAP FOR GENERAL OUTPUT     00560000
   7      *                                                                      00570000
   7      *              SYMBOLIC LABEL/NAME = DSN8CCDO                          00580000
   7      *              DESCRIPTION = CICS/OS/VS BMS MAP FOR DETAIL OUTPUT      00590000
   7      *                                                                      00600000
   7      *                                                                      00610000
   7      *   EXIT-NORMAL = CICS RETURN TRANSID                                  00620000
   7      *                                                                      00630000
   7      *   EXIT-ERROR = SQL ERROR FOR SQL ERRORS                              00640000
   7      *                CICS ABEND FOR CICS PROBLEMS                          00650000
   7      *                                                                      00660000
   7      *     RETURN CODE = NONE                                               00670000
   7      *                                                                      00680000
   7      *     ABEND CODES = LSCR - LOGICAL SCREEN SET INCORRECTLY              00690000
   7      *                                                                      00700000
   7      *     ERROR-MESSAGES = NONE                                            00710000
   7      *                                                                      00720000
   7      *                                                                      00730000
   7      *   EXTERNAL REFERENCES = COMMON CICS REQUIREMENTS                     00740000
   7      *     ROUTINES/SERVICES =                                              00750000
   7      *         CICS/VS SERVICES                                             00760000
   7      *         DSN8CC1             - SQL I MAINLINE CODE                    00770000
   7      *                                                                      00780000
   7      *     DATA-AREAS  =                                                    00790000
   7      *         DSN8MCCA            - PARAMETER TO BE PASSED TO DSN8CC1      00800000
   7      *                               COMMON AREA     -                      00810000
   7      *         DSN8MCC1            - DECLARE CONVERSATION STATUS            00820000
   7      *         DSN8MCC2            - COMMON AREA PART 2                     00830000
   7      *         DSN8MCMD            - CICS/OS/VS COBOL MAP, ORGANIZATION     00840000
   7      *         DSN8MCMG            - CICS/OS/VS COBOL MAP, ORGANIZATION     00850000
   7      *                                                                      00860000
   7      *     CONTROL-BLOCKS =                                                 00870000
   7      *         SQLCA                - SQL COMMUNICATION AREA                00880000
   7      *                                                                      00890000
   7      *   TABLES =  NONE                                                     00900000
   7      *                                                                      00910000
   7      *   CHANGE-ACTIVITY =  NONE                                            00920000
   7      *                                                                      00930000
   7      *                                                                      00940000
   7      *  *PSEUDOCODE*                                                        00950000
   7      *                                                                      00960000
   7      *    PROCEDURE                                                         00970000
   7      *    DECLARATIONS.                                                     00980000
   7      *       ALLOCATE COBOL WORK AREA FOR COMMAREA.                         00990000
   7      *       PUT MODULE NAME 'DSN8CCO' IN AREA USED BY ERROR-HANDLER.       01000000
   7      *       PUT CICS EIBTRMID IN PCONVSTA.CONVID TO BE PASSED TO           01010000
   7      *       DSN8CC1.                                                       01020000
   7      *       RETRIEVE LASTSCR FROM VCONA USING THE CONVID TO DETERMINE      01030000
   7      *       WHICH OF THE TWO BMS MAPS SHOULD BE USED TO MAP IN DATA.       01040000
   7      *                                                                      01050000
   7      *                                                                      01060000
   7      *    IF RETRIEVAL OF MAPS IS SUCCESSFUL, THEN DO;                      01070000
   7      *       EXEC CICS RECEIVE MAP ACCORDING TO SPECIFIED LASTSCR           01080000
```

```
START
COL ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

 7      *                                                                01090000
 7      *       IF MAPFAIL CONDITION IS RAISED, THEN DO:                 01100000
 7      *               COMPARM.PFKIN = '00'                             01110000
 7      *               GO TO CCOSEND                                    01120000
 7      *       END                                                      01130000
 7      *                                                                01140000
 7      *       ELSE                                                     01150000
 7      *               PUT DATA FROM MAP INTO COMPARM **                01160000
 7      *                                                                01170000
 7      *   ELSE                                                         01180000
 7      *               IT IS A NEW CONVERSATION                         01190000
 7      *               AND NO EXEC CICS RECEIVE MAP IS ISSUED.          01200000
 7      *                                                                01210000
 7      *   CCOSEND:                                                     01220000
 7      *   EXEC CICS LINK PROGRAM('DSN8CC1') COMMAREA(CONNAREA).        01230000
 7      *   UPON RETURN FROM DSN8CC1, EXEC CICS SEND MAP ACCORDING TO    01240000
 7      *        THE TYPE SPECIFIED IN PCONVSTA.LASTSCR.                 01250000
 7      *   EXEC CICS RETURN TRANSID(D8CS).                              01260000
 7      *                                                                01270000
 7      *  END.                                                          01280000
 7      *                                                                01290000
 7      *    *    I.E. LAST CONVERSATION EXISTS, BUT OPERATOR HAD ENTERED 01300000
 7      *         DATA FROM A CLEARED SCREEN OR HAD ERASED ALL DATA ON A  01310000
 7      *         FORMATTED SCREEN AND PRESSED ENTER                      01320000
 7      *                                                                01330000
 7      *    **   COMPARM.PFKIN = PF KEY ACTUALLY USED I.E. '01' FOR      01340000
 7      *           PF1 **                                               01350000
 7      *-------------------------------------------------------------**01360000
                                                                         01370000
                                                                         01380000
 3       IDENTIFICATION DIVISION.                                        01390000
 7      *----------------------                                          01400000
 8       PROGRAM-ID. DSN8CCO.                                            01410000
                                                                         01420000
 8       ENVIRONMENT DIVISION.                                           01430000
 7      *--------------------                                            01440000
                                                                         01450000
 8       DATA DIVISION.                                                  01460000
 7      *-------------                                                   01470000
 8       WORKING-STORAGE SECTION.                                        01480000
 8       77   FOUND              PIC  S99.                               01490000
13            EXEC SQL INCLUDE SQLCA     END-EXEC.                      01500000
13            EXEC SQL INCLUDE DSN8MCC2 END-EXEC.                       01510000
 8       01   COMMAREA.                                                  01520000
15            EXEC SQL INCLUDE DSN8MCCA END-EXEC.                       01530000
13            EXEC SQL INCLUDE DSN8MCCS END-EXEC.                       01540000
13            EXEC SQL INCLUDE DSN8MCNG END-EXEC.                       01550000
13            EXEC SQL INCLUDE DSN8MCND END-EXEC.                       01560000
 7      ***************************************************************-01570000
 7      *    MAPD REDEFINES THE COBOL STRUCTURE ASSOCIATED WITH THE     01580000
 7      *    CICS MAP DSN8CCD.                                          01590000
 7      ***************************************************************-01600000
 8       01   MAPD REDEFINES DSN8CCDI.                                   01610000
13            02  FILLER       PIC X(387).                              01620000
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   15             02  SUB8MAP      OCCURS 15 TIMES.                          01630000
   15               03 COL1LEN     PIC S9(4) COMP.                           01640000
   15               03 COL1ATTR    PIC X(1).                                 01650000
   15               03 COL1DATA    PIC X(57).                                01660000
   15               03 COL2LEN     PIC S9(4) COMP.                           01670000
   15               03 COL2ATTR    PIC X(1).                                 01680000
   15               03 COL2DATA    PIC X(60).                                01690000
    7       *-------------------------------------------------------------*-01700000
    7       *     PFKEYS IS AN ARRAY OF 24 ELEMENTS REPRESENTING THE DIFFERENT01710000
    7       *     PFKEYS AS THEY WOULD BE REPRESENTED IN EIBAID.              01720000
    7       *-------------------------------------------------------------*-01730000
    8        01   PFKEYS-DUMB.                                               01740000
   10          02 PFKEYS-ALL      PIC  X(24) VALUE '123456789:#@ABCDEFGHI.<(+.01750000
   10          02 PFKEYS  REDEFINES PFKEYS-ALL PIC X(1)  OCCURS 24 TIMES.   01760000
    7       *-------------------------------------------------------------*-01770000
    7       *     PFK IS AN ARRAY OF 12 TWO-BYTE CHARS REPRESENTING THE PFKEYS01780000
    7       *     ALLOWED AS INPUT TO DSN8CC1 AND DSN8CC2 ETC.                01790000
    7       *-------------------------------------------------------------*-01800000
    8        01   PFK-DUMB.                                                  01810000
   10          02 PFK-ALL         PIC  X(24) VALUE '0102030405060708091011l2'.01820000
   10          02 PFK      REDEFINES PFK-ALL    PIC X(2)  OCCURS 12 TIMES.  01830000
                                                                             01840000
                                                                             01850000
    8        PROCEDURE DIVISION.                                            01860000
    7       *-----------------                                             01870000
    7       *-------------------------------------------------------------*-01880000
    7       *     SQL RETURN CODE HANDLING                                  01890000
    7       *-------------------------------------------------------------*-01900000
   12             EXEC SQL WHENEVER SQLERROR GO TO DB-ERROR END-EXEC        01910000
   12             EXEC SQL WHENEVER SQLWARNING GO TO DB-ERROR END-EXEC.     01920000
                                                                             01930000
    7       *-------------------------------------------------------------*-01940000
    7       *     ALLOCATE COBOL WORK AREA / INITIALIZE VARIABLES           01950000
    7       *-------------------------------------------------------------*-01960000
   12             MOVE SPACES TO COMMAREA.                                  01970000
   12             MOVE 'DSN8CC0' TO MAJOR IN DSN8-MODULE-NAME.              01980000
   12             MOVE '0'       TO MAJSYS IN OUTAREA.                      01990000
   12             MOVE '0'       TO EXITCODE.                               02000000
   12             MOVE EIBTRMID TO CICSID OF PCONVSTA.                      02010000
   12             MOVE CONVID OF PCONVSTA TO SAVE-CONVID.                   02020000
                                                                             02030000
    7       *-------------------------------------------------------------*-02040000
    7       *     TRY TO RETRIEVE LAST CONVERSATION. IF SUCCESSFUL, USE THE 02050000
    7       *     LAST SCREEN SPECIFIED TO RECEIVE INPUT FROM TERMINAL.     02060000
    7       *-------------------------------------------------------------*-02070000
                                                                             02080000
   12             EXEC SQL SELECT LASTSCR                                   02090000
   21                      INTO    :PCONA.LASTSCR                           02100000
   21                      FROM    VCONA                                    02110000
   21                      WHERE   CONVID = :SAVE-CONVID END-EXEC.          02120000
                                                                             02130000
    7       *-------------------------------------------------------------*-02140000
    7       *     IF LAST CONVERSATION DOES NOT EXIST, THEN DO NOT ATTEMPT TO 02150000
    7       *     RECEIVE INPUT MAP. GO DIRECTLY TO VALIDATION MODULES      02160000
```

```
        DATASET: DSN120.DSNSAMP                                  DATE: 87/02/12
        MEMBER:  DSN8CCO                                         TIME: 17:17
                                                                 PAGE: 5
 START
  COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
     7         *    TO GET TITLE ETC. FOR OUTPUT MAP.                              02170000
     7         ********************************************************************02180000
                                                                                   02190000
    12               IF SQLCODE = +100 THEN                                        02200000
    21                     GO TO CCOSEND.                                         02210000
                                                                                   02220000
     7         ********************************************************************02230000
     7         *    IF LAST CONVERSATION EXISTS, BUT OPERATOR HAS ENTERED DATA    M02240000
     7         *    FROM A CLEARED SCREEN OR HAD ERASED ALL DATA ON A FORMATTED   M02250000
     7         *    SCREEN AND PRESSED ENTER THEN .......                         M02260000
     7         *    MOVE DATA INTO CORRESPONDING FIELDS IN INAREA AND GO TO       02270000
     7         *    VALIDATION MODULES.                                           02280000
     7         ********************************************************************02290000
    12               EXEC CICS HANDLE CONDITION MAPFAIL (CCOSEND) END-EXEC.        02300000
                                                                                   02310000
    12               IF LASTSCR OF PCONA  NOT = 'DSN8002 ' THEN                    02320000
    14                 GO TO CCO-LABEL1.                                           02325000
                                                                                   02330000
     7         *                                              **DSN8002            02333000
     7         *                                              **DETAIL MAP         02336000
     7         *                                              **MOVE DATA INTO     02340000
     7         *                                              **INPUT FIELDS       02350000
                                                                                   02360000
    14               EXEC CICS RECEIVE MAP ('DSN8CCO') MAPSET ('DSN8CCO')          02370000
    19                    END-EXEC.                                              02380000
    14               IF BMAJSYSL NOT = 0 THEN MOVE BMAJSYSI TO MAJSYS OF INAREA    02390000
    32                                   ELSE MOVE '0'      TO MAJSYS OF INAREA.  02400000
    14               IF BACTIONL NOT = 0 THEN MOVE BACTIONI TO ACTION OF INAREA    02410000
    32                                   ELSE MOVE SPACES   TO ACTION OF INAREA.  02420000
    14               IF BOBJECTL NOT = 0 THEN MOVE BOBJECTI TO OBJECT OF INAREA    02430000
    32                                   ELSE MOVE SPACES   TO OBJECT OF INAREA.  02440000
    14               IF BSEARCHL NOT = 0 THEN MOVE BSEARCHI TO SRCH   OF INAREA    02450000
    32                                   ELSE MOVE SPACES   TO SRCH   OF INAREA.  02460000
    14               IF BDATAL   NOT = 0 THEN MOVE BDATAI   TO DATAIN OF INAREA    02470000
    32                                   ELSE MOVE SPACES   TO DATAIN OF INAREA.  02480000
    14               MOVE 1 TO L.                                                  02490000
                                                                                   02500000
     7         *                                              **GO TO VALIDATION   02510000
                                                                                   02520000
    14               GO TO CCO-LABELX.                                             02530000
                                                                                   02540000
     7         *                                              **ERROR ON LASTSCREEN?02550000
                                                                                   02560000
     6         CCO-LABEL1.                                                         02565000
    12               IF LASTSCR OF PCONA NOT = 'DSN8001 ' THEN                     02570000
    14                 EXEC CICS ABEND ABCODE ('LSCR') END-EXEC                    02580000
    14                 GOBACK.                                                     02590000
                                                                                   02600000
     7         *                                              **USING LAST SCREEN  02610000
     7         *                                              **SPECIFIED TO RECEIVE02620000
     7         *                                              **INPUT FROM TERMINAL02630000
                                                                                   02640000
    12               EXEC CICS RECEIVE MAP ('DSN8CCG') MAPSET('DSN8CCG') END-EXEC.02650000
                                                                                   02660000
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

  7       *-------------------------------------------------------------*-02670000
  7       *    IF DATA IS RECEIVED FOR A FIELD, THEN MOVE THE DATA INTO THE 02680000
  7       *    CORRESPONDING FIELD IN INAREA, OTHERWISE MOVE BLANKS.      02690000
  7       *-------------------------------------------------------------*-02700000
                                                                          02710000
 12           IF AMAJSYSL NOT = 0 THEN MOVE AMAJSYSI TO MAJSYS OF INAREA  02720000
 32                               ELSE MOVE '0'      TO MAJSYS OF INAREA. 02730000
 12           IF AACTIONL NOT = 0 THEN MOVE AACTIONI TO ACTION OF INAREA  02740000
 32                               ELSE MOVE SPACES   TO ACTION OF INAREA. 02750000
 12           IF AOBJECTL NOT = 0 THEN MOVE AOBJECTI TO OBJECT OF INAREA  02760000
 32                               ELSE MOVE SPACES   TO OBJECT OF INAREA. 02770000
 12           IF ASEARCHL NOT = 0 THEN MOVE ASEARCHI TO SRCH   OF INAREA  02780000
 32                               ELSE MOVE SPACES   TO SRCH   OF INAREA. 02790000
 12           IF ADATAL   NOT = 0 THEN MOVE ADATAI   TO DATAIN OF INAREA  02800000
 32                               ELSE MOVE SPACES   TO DATAIN OF INAREA. 02810000
 12           GO TO CC0-LABEL3.                                          02820000
                                                                         02830000
  8       CC0-LABEL4.                                                    02840000
 14             IF COL2LEN(I) NOT = 0 THEN MOVE COL2DATA(I) TO TRANDATA(I) 02850000
 34                                   ELSE MOVE SPACES      TO TRANDATA(I). 02860000
 14             ADD 1 TO I.                                              02870000
                                                                         02880000
  7       *                                        ** CC0-LABEL4 LOOP     02890000
  8        CC0-LOOP4.                                                    02900000
 14             PERFORM CC0-LABEL4 UNTIL I > 15.                         02910000
                                                                         02920000
  8       CC0-LABEL3.                                                    02930000
 14             MOVE 1 TO I.                                             02940000
 14             MOVE 0 TO FOUND.                                         02950000
                                                                         02960000
  7       *-------------------------------------------------------------*-02970000
  7       *    CONVERT THE PFKEY INFO IN EIBAID TO THE FORM ACCEPTED      02980000
  7       *    BY DSN8CC1 AND DSN8CC2 EG. PF1 = '01' AND PF13 = '0I'.     02990000
  7       *-------------------------------------------------------------*-03000000
  8        CC0-LABEL4.                                                   03010000
                                                                         03020000
  7       *                                        **PF KEYS 1-12         03030000
                                                                         03040000
 12           IF PFKEYS(I) = EIBAID THEN MOVE 1 TO FOUND                 03050000
 16             ELSE ADD 1 TO I.                                         03060000
                                                                         03070000
  7       *                                        ** CC0-LABEL4 LOOP     03080000
  8        CC0-LOOP4.                                                    03090000
 12            PERFORM CC0-LABEL4 UNTIL                                  03100000
 14              I > 24 OR FOUND = 1.                                    03110000
                                                                         03120000
  7       *                                        **PF KEYS > 12         03130000
  8        CC0-LABEL5.                                                   03140000
 12            IF I >  12 THEN SUBTRACT 12 FROM I.                       03150000
 12            IF FOUND = 1 THEN                                         03160000
 14              MOVE PFK(I) TO PFKIN OF INAREA                          03170000
 12            ELSE MOVE SPACES TO PFKIN OF INAREA.                      03180000
 12            GO TO CC0-LABEL6.                                         03190000
                                                                         03200000
```

```
DATASET: DSN120.DSN3ANP                                        DATE: 87/02/12
MEMBER:  DSN8CC0                                               TIME: 17:17
                                                               PAGE: 7
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
   7      ************************************************************ -03210000
   7      *    GO TO DSN8CC1, GET DCLGEN STRUCTURES AND TABLE DCL      03220000
   7      ************************************************************-03230000
                                                                       03240000
   8      CC0SEND.                                                     03250000
  12          MOVE SPACES TO INAREA.                                   03260000
  12          MOVE '00' TO PFKIN OF INAREA.                            03270000
                                                                       03280000
   8      CC0-LABEL6.                                                  03290000
  12          MOVE '0' TO NAJSYS IN INAREA.                            03300000
  12          EXEC CICS LINK PROGRAM ('DSN8CC1') COMMAREA(CONNAREA)    03310000
  16              LENGTH(3000) END-EXEC.                               03320000
  12          GO TO CC0-NORMAL.                                        03330000
  12          EXEC SQL INCLUDE DSN8CXX END-EXEC.                       03340000
                                                                       03350000
   7      ************************************************************ -03360000
   7      *    AFTER RETURN FROM DSN8CC1, MOVE DATA TO OUTPUT MAP AREA AND 03370000
   7      *    SEND MAP ACCORDING TO MAP SPECIFIED IN LASTSCR OF PCONVSTA. 03380000
   7      ************************************************************-03390000
   8      CC0-NORMAL.                                                  03400000
  12          IF LASTSCR OF PCONVSTA = 'DSN8002 ' THEN GO TO CC0-LABEL9.  03410000
                                                                       03420000
   7      *                                        **MOVE DATA INTO     03430000
   7      *                                        **OUTPUT FIELDS      03440000
  12          MOVE HTITLE OF OUTAREA  TO ATITLEO.                      03450000
  12          MOVE MAJSYS OF OUTAREA TO ANAJSYSO.                      03460000
  12          MOVE ACTION OF OUTAREA TO AACTIONO.                      03470000
  12          MOVE OBJECT OF OUTAREA TO AOBJECTO.                      03480000
  12          MOVE SRCH   OF OUTAREA TO ASEARCHO.                      03490000
  12          MOVE DATAOUT           TO ADATAO.                        03500000
  12          MOVE MSG    OF OUTAREA TO AMSGO.                         03510000
  12          MOVE DESC2  OF OUTAREA TO ADESCL2O.                      03520000
  12          MOVE DESC3  OF OUTAREA TO ADESCL3O.                      03530000
  12          MOVE DESC4  OF OUTAREA TO ADESCL4O.                      03540000
  12          MOVE PFKTEXT OF OUTAREA TO APFKEYO.                      03550000
  12          MOVE 1 TO I.                                             03560000
                                                                       03570000
   7      *                                        **SEND MAP ACCORDING TO 03580000
   7      *                                        **PREVIOUS SCREEN    03590000
   8      CC0-LABEL7.                                                  03600000
  12          MOVE LINE0(I) TO ALINE0(I).                              03610000
  12          ADD 1 TO I.                                              03620000
                                                                       03630000
   7      *                                        **CC0-LABEL7 LOOP    03640000
   8      CC0-LOOP7.                                                   03650000
  12          PERFORM CC0-LABEL7 UNTIL                                 03660000
  16              I > 15.                                              03670000
                                                                       03680000
   7      ************************************************************-03690000
   7      *    CREATES A DYNAMIC CURSOR                                03700000
   7      ************************************************************-03710000
                                                                       03720000
   7      *                                        **SET CURSOR POSITION 03730000
   8      CC0-LABEL8.                                                  03740000
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   12          MOVE ZEROES TO CURSOR-VALUE.                                  03750000
   12          IF AACTIONO = SPACES THEN MOVE +179 TO CURSOR-VALUE            03760000
   12          ELSE IF AOBJECTO = SPACES THEN MOVE +259 TO CURSOR-VALUE       03770000
   12          ELSE IF ASEARCHO = SPACES THEN MOVE +339 TO CURSOR-VALUE       03780000
   12          ELSE IF ADATAO = SPACES OR AACTIONO = '0' OR 'E' THEN          03790000
   12          MOVE +419 TO CURSOR-VALUE.                                     03800000
                                                                              03810000
    7      *                                              **SEND OUTPUT MAP  03820000
                                                                              03830000
   12          IF CURSOR-VALUE = ZEROES THEN                                  03840000
   14            EXEC CICS SEND MAP('DSN8CCG') MAPSET('DSN8CCG') END-EXEC     03850000
   12          ELSE                                                           03860000
   14            EXEC CICS SEND MAP('DSN8CCG') MAPSET('DSN8CCG') ERASE        03870000
   14            CURSOR(CURSOR-VALUE) END-EXEC.                               03880000
                                                                              03890000
    7      *                                                  **FINISHED?    03900000
                                                                              03910000
   12          IF EXITCODE = '1' THEN GO TO CCO-LABEL12.                      03920000
   12          EXEC CICS RETURN TRANSID('D8CS') END-EXEC.                     03930000
                                                                              03940000
    7      ***************************************************************-03950000
    7      *    MOVES DATA FROM OUTPUT MAP AREA TO                          03960000
    7      *    RECEIVE MAP ACCORDING TO MAP SPECIFIED IN LASTSCR OF PCONVST 03970000
    7      ***************************************************************-03980000
                                                                              03990000
    7      *                                          **MOVE DATA           04000000
    7      *                                          **FROM OUTPUT FIELDS  04010000
    8      CCO-LABEL9.                                                       04020000
   12          MOVE HTITLE OF OUTAREA TO BTITLEO.                            04030000
   12          MOVE MAJSYS OF OUTAREA TO BMAJSYSO.                           04040000
   12          MOVE ACTION OF OUTAREA TO BACTIONO.                           04050000
   12          MOVE OBJECT OF OUTAREA TO BOBJECTO.                           04060000
   12          MOVE SRCH   OF OUTAREA TO BSEARCHO.                           04070000
   12          MOVE DATAOUT           TO BDATAO.                             04080000
   12          MOVE MSG    OF OUTAREA TO BMSGO.                              04090000
   12          MOVE DESC2  OF OUTAREA TO BDESCL2O.                           04100000
   12          MOVE DESC3  OF OUTAREA TO BDESCL3O.                           04110000
   12          MOVE DESC4  OF OUTAREA TO BDESCL4O.                           04120000
   12          MOVE PFKTEXT OF OUTAREA TO BPFKEYO.                           04130000
   12          MOVE 1 TO I.                                                  04140000
    7      *                                          **RECEIVE MAP ACCORDING 04150000
    7      *                                            **TO PREVIOUS SCREEN  04160000
                                                                              04170000
    8      CCO-LABEL10.                                                      04180000
   12          MOVE FIELD1(I)  TO COL1DATA(I) .                              04190000
    7      *                              ** CHECK FOR ATTRIBUTE OF X'COC1'   04200000
   12          IF ATTR(1) = -16191 THEN MOVE -1 TO COL2LEN(I).               04210000
   12          MOVE ATTR2(I)   TO COL2ATTR(I) .                              04220000
   12          MOVE FIELD2(I)  TO COL2DATA(I) .                              04230000
   12          ADD 1 TO 1.                                                   04240000
                                                                              04250000
    7      *                                  ** CCO-LABEL10 LOOP            04260000
    8      CCO-LOOP10.                                                       04270000
   12          PERFORM CCO-LABEL10 UNTIL                                     04280000
```

```
START
 COL   ----+----1----+----2----+----3----+----4----+----5----+----6----+----7--+----8
 15                   I > I5.

  8           CCO-LABEL11.
  7          *****************************************************************
  7          *     CREATES A DYNAMIC CURSOR
  7          *****************************************************************

  7          *                                              **SET CURSOR POSITION
 12               MOVE ZEROES TO CURSOR-VALUE.
 12               IF BACTIONO = SPACES THEN MOVE +179 TO CURSOR-VALUE
 12               ELSE IF BOBJECTO = SPACES THEN MOVE +259 TO CURSOR-VALUE
 12               ELSE IF BSEARCHO = SPACES THEN MOVE +339 TO CURSOR-VALUE
 12               ELSE IF BDATAO = SPACES OR BACTIONO = 'D' OR 'E' THEN
 12               MOVE +419 TO CURSOR-VALUE.

  7          *                                                  **SEND INPUT MAP

 12               IF CURSOR-VALUE = ZEROES THEN
 16                   EXEC CICS SEND MAP('DSN8CCD') MAPSET('DSN8CCD') END-EXEC
 12               ELSE
 16                   EXEC CICS SEND MAP('DSN8CCD') MAPSET('DSN8CCD') ERASE
 16                   CURSOR(CURSOR-VALUE) END-EXEC.

  7          *                                                    **FINISHED?
 12               IF EXITCODE = '1' THEN GO TO CCO-LABEL12.
 12               EXEC CICS RETURN TRANSID('D8CS') END-EXEC.
 12               GOBACK.


  7          *                                                   **RETURN
  8           CCO-LABEL12.
 12               EXEC CICS RETURN END-EXEC.
 12               GOBACK.
```

```
        DATASET: DSN120.DSNSAMP                                  DATE: 87/02/12
        MEMBER:  DSN34CV0                                        TIME: 17:35
                                                                 PAGE: 1
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

    7       *                                                              00010000
    7       * FIND VALID OPTIONS FOR - ACTION                             00020000
    7       *                                                              00030000
   15             EXEC SQL DECLARE V02 CURSOR FOR                          00040000
   20                  SELECT                                              00050000
   27                       ACTION, SELTXT                                 00060000
   23                    FROM VOPTVAL                                      00070000
   23                    WHERE MAJSYS    = :INAREA.MAJSYS                  00080000
   29                      AND ACTION   ¬= ' '                             00090000
   29                      AND OBJECT    = ' '                             00100000
   23                    ORDER BY ACTION ASC END-EXEC.                     00110000
    7       *                                                              00120000
    7       * FIND VALID OPTIONS FOR - OBJECT                             00130000
    7       *                                                              00140000
   15             EXEC SQL DECLARE V03 CURSOR FOR                          00150000
   20                  SELECT                                              00160000
   27                       OBJECT, SELTXT                                 00170000
   23                    FROM VOPTVAL                                      00180000
   23                    WHERE MAJSYS    = :INAREA.MAJSYS                  00190000
   29                      AND ACTION    = :INAREA.ACTION                  00200000
   29                      AND OBJECT   ¬= ' '                             00210000
   29                      AND SRCHCRIT  = ' '                             00220000
   23                    ORDER BY OBJECT ASC END-EXEC.                     00230000
    7       *                                                              00240000
    7       * FIND VALID OPTIONS FOR - SEARCH CRITERION                    00250000
    7       *                                                              00260000
   15             EXEC SQL DECLARE V04 CURSOR FOR                          00270000
   20                  SELECT                                              00280000
   27                       SRCHCRIT, SELTXT                               00290000
   23                    FROM VOPTVAL                                      00300000
   23                    WHERE MAJSYS    = :INAREA.MAJSYS                  00310000
   29                      AND ACTION    = :INAREA.ACTION                  00320000
   29                      AND OBJECT    = :INAREA.OBJECT                  00330000
   29                      AND SRCHCRIT ¬= ' '                             00340000
   29                      AND (SCRTYPE  = ' ' OR SCRTYPE = 'S')           00350000
   23                    ORDER BY SRCHCRIT ASC END-EXEC.                   00360000
```

```
        DATASET: DSN120.DSNSAMP                                  DATE: 87/02/12
        MEMBER:  DSN8MCCA                                        TIME: 17:34
                                                                 PAGE: 1
 START
  COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

    7        ***** THE COMMAREA STRUCTURE DECLARED BELOW IS USED TO PASS INPUT 00010000
    7        ***** DATA BETWEEN THE SUBSYSTEM DEPENDENT MODULES (CICS,INS,TSO) 00020000
    7        ***** AND SQL1 AND SQL2                                           00030000
    9         2  PCONVSTA.                                                     00040000
   10          3  CONVIN.                                                     00050000
   11           4  TRMID.                                                    00060000
   12            5  CICSID    PIC X(4).                                      00070000
   12            5  CICSSK    PIC X(4).                                      00080000
   11           4  USERID     PIC X(8).                                      00090000
   10          3  LASTSCR     PIC X(3).                                      00100000
   10          3  LASTPOS.                                                   00110000
   11           4  PREV       PIC X.                                         00120000
   11           4  MAXSEL     PIC 99.                                        00130000
   11           4  POSREST.                                                  00140000
   12            5 NPASAVE.                                                  00150000
   13             6 LDEPTNO.                                                 00160000
   14              49 LDEPTNOL  PIC S9(4) COMP-4.                            00170000
   14              49 LDEPTNOD  PIC X(3).                                    00180000
   13             6 LDEPTNAN.                                                00190000
   14              49 LDEPTNANL PIC S9(4) COMP-4.                            00200000
   14              49 LDEPTNAND PIC X(36).                                   00210000
   13             6 LMGRNO.                                                  00220000
   14              49 LMGRNOL   PIC S9(4) COMP-4.                            00230000
   14              49 LMGRNOD   PIC X(6).                                    00240000
   13             6 LMGRNAME.                                                00250000
   14              49 LMGRNAMEL PIC S9(4) COMP-4.                            00260000
   14              49 LMGRNAMEO PIC X(15).                                   00270000
   13             6 LEMPNO.                                                  00280000
   14              49 LEMPNOL   PIC S9(4) COMP-4.                            00290000
   14              49 LEMPNOD   PIC X(6).                                    00300000
   13             6 LEMPNAME.                                                00310000
   14              49 LEMPNAMEL PIC S9(4) COMP-4.                            00320000
   14              49 LEMPNAMED PIC X(15).                                   00330000
   13             6 OININ      PIC X(3).                                     00340000
   13             6 EININ      PIC X(6).                                     00350000
   12            5 NPDSAVE.                                                  00360000
   13             6 T2NIN    PIC X.                                         00370000
   13             6 D2NIN    PIC X(3).                                      00380000
   13             6 E2NIN    PIC X(6).                                      00390000
   12            5  DSN8-MODULE-NAME.                                       00400000
   13             6  MAJOR  PIC X(8).                                       00410000
   13             6  MINOR  PIC X(8).                                       00420000
   12            5 FILLER    PIC X(123).                                    00430000
   10          3  LASTPOSO REDEFINES LASTPOS PIC X(254).                    00440000
   10          3  LASTPOSC   PIC X(254).                                    00450000
    9         2  PCONVSTAO REDEFINES PCONVSTA PIC X(532).                   00460000
    9         2  OUTAREA.                                                   00470000
   10          3  MAJSYS     PIC X.                                         00480000
   10          3  ACTION     PIC X.                                         00490000
   10          3  OBJECT     PIC XX.                                        00500000
   10          3  SRCH       PIC XX.                                        00510000
   10          3  DATAOUT    PIC X(60).                                     00520000
   10          3  DATAOU2    REDEFINES DATAOUT.                             00530000
   11           4 DATO2      PIC X(2).                                      00540000
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

  11          4 FILLER      PIC X(58).                                    00550000
  10          3 ATITLE      PIC X(50).                                    00560000
  10          3 DESC2       PIC X(50).                                    00570000
  10          3 DESC3       PIC X(50).                                    00580000
  10          3 DESC4       PIC X(50).                                    00590000
  10          3 MSG.                                                      00600000
  11            4 MSGTXT.                                                 00640000
  12              5 FILLER      PIC X(29).                                00650000
  12              5 MSGM30      PIC X(4).                                 00660000
  12              5 MSGM002     PIC X(11).                                00670000
  12              5 FILLER      PIC X(27).                                00680000
  11            4 MSGTEXT REDEFINES MSGTXT PIC X(70).                     00690000
  11            4 FILLER      PIC X(9).                                   00695000
  10          3 PFKTEXT       PIC X(79).                                  00700000
  10          3 OUTPUTO.                                                  00710000
  11            4 LINE15   OCCURS 15.                                     00720000
  12              5 LINEO       PIC X(79).                                00730000
  12              5 LINE-1  REDEFINES LINEO.                              00740000
  13                6 FIELD-1   PIC X(3).                                 00750000
  13                6 FIELD-2   PIC X(6).                                 00760000
  13                6 FIELD-3   PIC X(70).                                00770000
  12              5 LINE-2  REDEFINES LINEO.                              00780000
  13                6 FIELD1    PIC X(37).                                00790000
  13                6 ATTR      PIC S9(4) COMP.                           00800000
  13                6 ATTRO   REDEFINES ATTR.                             00810000
  14                  7 ATTR1     PIC X.                                  00820000
  14                  7 ATTR2     PIC X.                                  00830000
  13                6 FIELD2    PIC X(40).                                00840000
  12              5 LEFT-OPT  REDEFINES LINEO.                            00850000
  13                6 O1NO      PIC X(3).                                 00860000
  13                6 X11       PIC X(2).                                 00870000
  13                6 O1NA      PIC X(34).                                00880000
  13                6 FILLER    PIC X(40).                                00890000
  12              5 LEFT-MGR  REDEFINES LINEO.                            00900000
  13                6 X21       PIC X.                                    00910000
  13                6 M1NO      PIC X(6).                                 00920000
  13                6 X22       PIC X(2).                                 00930000
  13                6 M1NA      PIC X(30).                                00940000
  13                6 FILLER    PIC X(40).                                00950000
  12              5 RIGHT-OPT REDEFINES LINEO.                            00960000
  13                6 X12       PIC X(40).                                00970000
  13                6 O2NO      PIC X(3).                                 00980000
  13                6 X13       PIC X(2).                                 00990000
  13                6 O2NA      PIC X(34).                                01000000
  12              5 RIGHT-MGR REDEFINES LINEO.                            01010000
  13                6 X23       PIC X(40).                                01020000
  13                6 X24       PIC X.                                    01030000
  13                6 M2NO      PIC X(6).                                 01040000
  13                6 X25       PIC X(2).                                 01050000
  13                6 M2NA      PIC X(30).                                01060000
  12              5 RIGHT-EMP REDEFINES LINEO.                            01070000
  13                6 X14       PIC X(40).                                01080000
  13                6 E2NO      PIC X(6).                                 01090000
  13                6 X15       PIC X(2).                                 01100000
```

```
START
COL    ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
```

```
 13          6 E2NA      PIC X(31).                                    01110000
 10        3 OUTPUT1  REDEFINES OUTPUTO.                               01120000
 11          4 LINE2     PIC X(79) OCCURS 2.                           01130000
 11          4  BGMC10.                                                01140000
 12           5 BGMC1 OCCURS 13.                                       01150000
 13             6 LINENO    PIC 99.                                   01160000
 13             6 FILLER    PIC X(2).                                 01170000
 13             6 DEPTNUM   PIC X(3).                                 01180000
 13             6 FILLER    PIC X(3).                                 01190000
 13             6 DEPTNA    PIC X(36).                                01200000
 13             6 FILLER    PIC X(3).                                 01210000
 13             6 MGRNUM    PIC X(6).                                 01220000
 13             6 FILLER    PIC X(3).                                 01230000
 13             6 MGRFIN    PIC X.                                    01240000
 13             6 MGRSIN    PIC X.                                    01250000
 13             6 FILLER    PIC X.                                    01260000
 13             6 MGRLNAM   PIC X(15).                                01270000
 13             6 FILLER    PIC X(3).                                 01280000
  9        2 OUTAREAO REDEFINES OUTAREA PIC X(1609).                  01290000
  9        2 COMPARM.                                                 01300000
 10          3 NEWCONV     PIC X.                                    01310000
 10          3 NEWREQ      PIC X.                                    01320000
 10          3 RETCODE     PIC X.                                    01330000
 10          3 EXITCODE    PIC X.                                    01340000
  9        2 INAREA.                                                  01350000
 10          3 MAJSYS      PIC X.                                    01360000
 10          3 ACTION      PIC X.                                    01370000
 10          3 OBJECT      PIC XX.                                   01380000
 10          3 SRCH        PIC XX.                                   01390000
 10          3 PFKIN       PIC XX.                                   01400000
 10          3 DATA1N      PIC X(60).                                01410000
 10          3 DATA1N1  REDEFINES DATAIN PIC X OCCURS 60.            01420000
 10          3 DATAINO  REDEFINES DATAIN.                            01430000
 11            4 LINE-SEL  PIC X(2).                                 01440000
 11            4 DATAO1    PIC X(58).                                01450000
 10          3 DATA36   REDEFINES DATAIN PIC X(36).                  01460000
 10          3 DATA24   REDEFINES DATAIN PIC X(24).                  01470000
 10          3 DATA15   REDEFINES DATAIN PIC X(15).                  01480000
 10          3 DATA6    REDEFINES DATAIN PIC X(6).                   01490000
 10          3 DATA3    REDEFINES DATAIN PIC X(3).                   01500000
 10          3 DATA2    REDEFINES DATAIN PIC 99.                     01510000
 10          3 DATA19V1 REDEFINES DATAIN.                            01520000
 14            10 DAT1   PIC X.                                      01530000
 14            10 DAT2   PIC X.                                      01540000
 10          3 TRANDATA  PIC X(40) OCCURS 15.                        01550000
  9        2 INAREAO REDEFINES INAREA PIC X(668).                    01560000
                                                                      01570000
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7        ********** DSN8CC2 - COMMON MODULE FOR CICS - COBOL********** 00010000
   7        *                                                             * 00020000
   7        *  MODULE NAME = DSN8CC2                                      * 00030000
   7        *                                                             * 00040000
   7        *  DESCRIPTIVE NAME = DB2  SAMPLE APPLICATION                 * 00050000
   7        *                     SQL 2 COMMON MODULE                     * 00060000
   7        *                     CICS                                    * 00070000
   7        *                     COBOL                                   * 00080000
   7        *                                                             * 00090000
   7        *                                                             * 00100000
   7        * COPYRIGHT = 5740-XYR (C) COPYRIGHT IBM CORP 1982, 1985      * 00110000
   7        * REFER TO COPYRIGHT INSTRUCTIONS FORM NUMBER G120-2083       * 00115000
   7        *                                                             * 00120000
   7        * STATUS = RELEASE 2, LEVEL 0                                * 00130000
   7        *                                                             * 00150000
   7        *  FUNCTION = ROUTER FOR SECONDARY SELECTION AND/OR           * 00160000
   7        *                        DETAIL PROCESSING                    * 00170000
   7        *             CALLS SECONDARY SELECTION MODULES               * 00180000
   7        *                     DSN8MCA DSN8MCM                         * 00190000
   7        *             CALLS DETAIL MODULES                            * 00200000
   7        *                     DSN8MCD DSN8MCE DSN8MCF                 * 00210000
   7        *                     DSN8MCT DSN8MCV DSN8MCW DSN8MCX DSN8MCZ  * 00220000
   7        *             CALLED BY DSN8MC1 (SQL1)                        * 00230000
   7        *                                                             * 00240000
   7        *  NOTES =  NONE                                              * 00250000
   7        *                                                             * 00260000
   7        *                                                             * 00270000
   7        *  MODULE TYPE =                                              * 00280000
   7        *     PROCESSOR   = DB2 PRECOMPILER, CICS TRANSLATOR,         * 00290000
   7        *                   VS COBOL                                  * 00300000
   7        *     MODULE SIZE = SEE LINKEDIT                              * 00310000
   7        *     ATTRIBUTES  = REUSABLE                                  * 00320000
   7        *                                                             * 00330000
   7        *  ENTRY POINT = DSN8CC2                                      * 00340000
   7        *     PURPOSE = SEE FUNCTION                                  * 00350000
   7        *     LINKAGE = NONE                                          * 00360000
   7        *     INPUT   =                                               * 00370000
   7        *                                                             * 00380000
   7        *               SYMBOLIC LABEL/NAME = COMMPTR                 * 00390000
   7        *               DESCRIPTION         = POINTER TO COMMAREA     * 00400000
   7        *                                     (COMMUNICATION AREA)    * 00410000
   7        *                                                             * 00420000
   7        *     OUTPUT  =                                               * 00430000
   7        *                                                             * 00440000
   7        *               SYMBOLIC LABEL/NAME = COMMPTR                 * 00450000
   7        *               DESCRIPTION         = POINTER TO COMMAREA     * 00460000
   7        *                                     (COMMUNICATION AREA)    * 00470000
   7        *                                                             * 00480000
   7        *  EXIT-NORMAL =  RETURN CODE 0 NORMAL COMPLETION             * 00490000
   7        *                                                             * 00500000
   7        *  EXIT-ERROR =                                               * 00510000
   7        *           IF SQLERROR OR SQLWARNING, SQL WHENEVER CONDITION * 00520000
   7        *           SPECIFIED IN DSN8CC2 WILL BE RAISED AND PROGRAM   * 00530000
   7        *           WILL GO TO THE LABEL DB-ERROR.                    * 00540000
```

DATASET: DSN120.DSNSAMP
MEMBER: DSN8CC2

DATE: 87/02/12
TIME: 17:18
PAGE: 2

```
START
 COL   ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
  7      *
  7      *                                                                      * 00550000
  7      *         RETURN CODE = NONE                                           * 00560000
  7      *                                                                      * 00570000
  7      *                                                                      * 00580000
  7      *         ABEND CODES = NONE                                           * 00590000
  7      *                                                                      * 00600000
  7      *         ERROR-MESSAGES =                                             * 00610000
  7      *             DSN8062E-MISSING DETAIL MODULE                           * 00620000
  7      *             DSN8063E-MISSING SECONDARY SEL MODULE                    * 00630000
  7      *             DSN8066E-UNSUPPORTED PFK OR LOGIC ERROR                  * 00640000
  7      *             DSN8072E-INVALID SELECTION ON SECONDARY SCREEN           * 00650000
  7      *                                                                      * 00660000
  7      *     EXTERNAL REFERENCES = NONE                                       * 00670000
  7      *         ROUTINES/SERVICES = 10 MODULES LISTED ABOVE                  * 00680000
  7      *             DSN8MCG     -   ERROR MESSAGE ROUTINE                    * 00690000
  7      *                                                                      * 00700000
  7      *         DATA-AREAS =                                                 * 00710000
  7      *             DSN8MCA     -   SECONDARY SELECTION FOR ORGANIZATION     * 00720000
  7      *             DSN8MCAD    -   DECLARE ADMINISTRATION DETAIL            * 00730000
  7      *             DSN8MCAE    -   CURSOR EMPLOYEE LIST                     * 00740000
  7      *             DSN8MCAL    -   CURSOR ADMINISTRATION LIST               * 00750000
  7      *             DSN8MCA2    -   DECLARE ADMINISTRATION DETAIL            * 00760000
  7      *             DSN8MCCA    -   COMMON AREA                              * 00770000
  7      *             DSN8MCC2    -   COMMON AREA PART 2                       * 00780000
  7      *             DSN8MCD     -   DEPARTMENT STRUCTURE DETAIL              * 00790000
  7      *             DSN8MCDA    -   CURSOR ADMINISTRATION DETAIL             * 00800000
  7      *             DSN8MCDN    -   CURSOR FOR DISPLAY TEXT FROM             * 00810000
  7      *                             TDSPTXT TABLE                            * 00820000
  7      *             DSN8MCDM    -   DECLARE DEPARTMENT MANAGER               * 00830000
  7      *             DSN8MCDP    -   DECLARE DEPARTMENT                       * 00840000
  7      *             DSN8MCDT    -   DECLARE DISPLAY TEXT                     * 00850000
  7      *             DSN8MCE     -   DEPARTMENT DETAIL                        * 00860000
  7      *             DSN8MCEM    -   DECLARE EMPLOYEE                         * 00870000
  7      *             DSN8MCF     -   EMPLOYEE   DETAIL                        * 00880000
  7      *             DSN8MCOV    -   DECLARE OPTION VALIDATION                * 00890000
  7      *             DSN8MCXX    -   ERROR HANDLER                            * 00900000
  7      *                                                                      * 00910000
  7      *         CONTROL-BLOCKS =                                             * 00920000
  7      *             SQLCA       -   SQL COMMUNICATION AREA                   * 00930000
  7      *                                                                      * 00940000
  7      *    TABLES = NONE                                                     * 00950000
  7      *                                                                      * 00960000
  7      *    CHANGE-ACTIVITY = NONE                                            * 00970000
  7      *                                                                      * 00980000
  7      *                                                                      * 00990000
  7      *  *PSEUDOCODE*                                                        * 01000000
  7      *                                                                      * 01010000
  7      *                                                                      * 01020000
  7      * THIS MODULE DETERMINES WHICH SECONDARY SELECTION AND/OR              * 01030000
  7      * DETAIL MODULE(S) ARE TO BE CALLED IN THE CICS/COBOL                  * 01040000
  7      * ENVIRONMENT.                                                         * 01050000
  7      *                                                                      * 01060000
  7      * WHAT HAS HAPPENED SO FAR?.............THE SUBSYSTEM                   * 01070000
  7      * DEPENDENT MODULE (IMS,CICS,TSO) OR (SQL O) HAS                       * 01080000
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
   7        * READ THE INPUT SCREEN, FORMATTED THE INPUT AND PASSED CONTROL * 01090000
   7        * TO SQL 1. SQL 1 PERFORMS VALIDATION ON THE SYSTEM DEPENDENT   * 01100000
   7        * FIELDS (MAJOR SYSTEM, ACTION, OBJECT, SEARCH CRITERIA). IF    * 01110000
   7        * ALL SYSTEM FIELDS ARE VALID SQL 1 PASSED CONTROL TO THIS      * 01120000
   7        * MODULE. PASSED PARAMETERS CONSIST ONLY OF A POINTER WHICH     * 01130000
   7        * POINTS TO A COMMUNICATION CONTROL AREA USED TO COMMUNICATE    * 01140000
   7        * BETWEEN SQL 0 , SQL 1, SQL 2 AND THE SECONDARY SELECTION      * 01150000
   7        * AND DETAIL MODULES.                                           * 01160000
   7        *                                                               * 01170000
   7        *WHAT IS INCLUDED IN THIS MODULE?.............                  * 01180000
   7        * ALL SECONDARY SELECTION AND DETAIL MODULES ARE 'INCLUDED'.     * 01190000
   7        * ALL VARIABLES KNOWN IN THIS PROCEDURE ARE KNOWN IN THE         * 01200000
   7        * SUB PROCEDURES. ALL SQL CURSOR DEFINITIONS AND                 * 01210000
   7        * SQL 'INCLUDES' ARE DONE IN THIS PROCEDURE. BECAUSE OF THE      * 01220000
   7        * RESTRICTION THAT CURSOR HOST VARIABLES MUST BE DECLARED BEFORE* 01230000
   7        * THE CURSOR DEFINITION ALL CURSOR HOST VARIABLES ARE DECLARED   * 01240000
   7        * IN THIS PROCEDURE.                                            * 01250000
   7        *                                                               * 01260000
   7        * PROCEDURE                                                     * 01270000
   7        *    IF ANSWER TO DETAIL SCREEN AND DETAIL PROCESSOR            * 01280000
   7        *    IS NOT WILLING TO ACCEPT AN ANSWER THEN                    * 01290000
   7        *                NEW REQUEST*                                   * 01300000
   7        *                                                               * 01310000
   7        *    ELSE                                                       * 01320000
   7        *          IF ANSWER TO A SECONDARY SELECTION THEN              * 01330000
   7        *                DETERMINE IF NEW REQUEST.                      * 01340000
   7        *                                                               * 01350000
   7        *                                                               * 01360000
   7        *                                                               * 01370000
   7        *    CASE (NEW REQUEST)                                         * 01380000
   7        *                                                               * 01390000
   7        *          SUBCASE ('ADD')                                      * 01400000
   7        *                DETAIL PROCESSOR                               * 01410000
   7        *                RETURN TO SQL 1                                * 01420000
   7        *          ENDSUB                                               * 01430000
   7        *                                                               * 01440000
   7        *          SUBCASE ('ERASE','DISPLAY','UPDATE')                 * 01450000
   7        *                CALL SECONDARY SELECTION                       * 01460000
   7        *                IF # OF POSSIBLE CHOICES IS ¬= 1 THEN          * 01470000
   7        *                      RETURN TO SQL 1                          * 01480000
   7        *                ELSE                                           * 01490000
   7        *                      CALL THE DETAIL PROCESSOR                * 01500000
   7        *                      RETURN TO SQL 1                          * 01510000
   7        *          ENDSUB                                               * 01520000
   7        *                                                               * 01530000
   7        *    ENDCASE                                                    * 01540000
   7        *                                                               * 01550000
   7        *    IF ANSWER TO SECONDARY SELECTION AND A SELECTION HAS       * 01560000
   7        *    ACTUALLY BEEN MADE THEN                                    * 01570000
   7        *          IF IT IS A VALID SELECTION NUMBER THEN               * 01580000
   7        *                CALL DETAIL PROCESSOR                          * 01590000
   7        *                RETURN TO SQL 1                                * 01600000
   7        *          END                                                  * 01610000
   7        *                                                               * 01620000
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7  *                ELSE                                          * 01630000
   7  *                     PRINT ERROR MSG                          * 01640000
   7  *                     RETURN TO SQL 1                          * 01650000
   7  *              END.                                            * 01660000
   7  *                                                              * 01670000
   7  *    IF ANSWER TO SECONDARY SELECTION THEN                     * 01680000
   7  *              CALL SECONDARY SELECTION                        * 01690000
   7  *              RETURN TO SQL 1                                 * 01700000
   7  *    END.                                                      * 01710000
   7  *                                                              * 01720000
   7  *    IF ANSWER TO DETAIL THEN                                  * 01730000
   7  *              CALL DETAIL PROCESSOR                           * 01740000
   7  *              RETURN TO SQL 1                                 * 01750000
   7  *    END.                                                      * 01760000
   7  *                                                              * 01770000
   7  *    RETURN TO SQL 1.                                          * 01780000
   7  *                                                              * 01790000
   7  * END.                                                         * 01800000
   7  *                                                              * 01810000
   7  *   *EXAMPLE- A ROW IS SUCCESSFULLY ADDED, THE OPERATOR RECEIVES* 01820000
   7  *    THE SUCCESSFULLY ADDED MESSAGE AND JUST HITS ENTER.       * 01830000
   7  *--------------------------------------------------------------* 01840000
                                                                      01850000
                                                                      01860000
   8   IDENTIFICATION DIVISION.                                       01870000
   7  *-----------------------                                       01880000
   8   PROGRAM-ID. DSN8CC2.                                           01890000
                                                                      01900000
   8   ENVIRONMENT DIVISION.                                          01910000
   7  *--------------------                                          01920000
                                                                      01930000
   8   DATA DIVISION.                                                 01940000
   7  *-------------                                                 01950000
   8   WORKING-STORAGE SECTION.                                       01960000
                                                                      01970000
   7  ****************************************                        01980000
   7  *  FIELDS SENT TO MESSAGE ROUTINE      *                       01990000
   7  ****************************************                        02000000
   9    01 MSGCODE                  PIC X(04).                       02010000
                                                                      02020000
   9    01 OUTMSG                   PIC X(69).                       02030000
                                                                      02040000
  12       EXEC SQL INCLUDE SQLCA END-EXEC.                           02050000
  12       EXEC SQL INCLUDE DSN8MCC2 END-EXEC.                        02060000
  12       EXEC SQL INCLUDE DSN8MCDP END-EXEC.                        02070000
  12       EXEC SQL INCLUDE DSN8MCEM END-EXEC.                        02080000
  12       EXEC SQL INCLUDE DSN8MCOM END-EXEC.                        02090000
  12       EXEC SQL INCLUDE DSN8MCAD END-EXEC.                        02100000
  12       EXEC SQL INCLUDE DSN8MCA2 END-EXEC.                        02110000
  12       EXEC SQL INCLUDE DSN8MCOV END-EXEC.                        02120000
  12       EXEC SQL INCLUDE DSN8MCOT END-EXEC.                        02130000
                                                                      02140000
   8   LINKAGE SECTION.                                               02150000
   8   01  DFHCOMMAREA.                                              02160000
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
  12            EXEC SQL INCLUDE DSN8MCCA END-EXEC.                          02170000
                                                                             02180000
                                                                             02190000
   8       PROCEDURE DIVISION.                                               02200000
   7       *------------------                                               02210000
  12            EXEC SQL INCLUDE DSN8MCAE END-EXEC.                          02220000
  12            EXEC SQL INCLUDE DSN8MCAL END-EXEC.                          02230000
  12            EXEC SQL INCLUDE DSN8MCCH END-EXEC.                          02240000
  12            EXEC SQL INCLUDE DSN8MCDA END-EXEC.                          02250000
                                                                             02260000
   7       ************************************************************      02270000
   7       *  SQL RETURN CODE HANDLING                                *      02280000
   7       ************************************************************      02290000
  12            EXEC SQL WHENEVER SQLERROR   GO TO D8-ERROR END-EXEC         02300000
  12            EXEC SQL WHENEVER SQLWARNING GO TO D8-ERROR END-EXEC.        02310000
                                                                             02320000
   7       ************************************************************      02330000
   7       *  INITIALIZATIONS                                         *      02340000
   7       ************************************************************      02350000
  12            MOVE 'DSN8CC2' TO MAJOR.                                     02360000
  12            MOVE  SPACES   TO MINOR.                                     02370000
                                                                             02380000
   7       ************************************************************      02390000
   7       *  DETERMINES WHETHER NEW REQUEST OR NOT                   *      02400000
   7       ************************************************************      02410000
   8        IC200B.                                                          02420000
                                                                             02430000
  12            IF PREV OF PCONVSTA = ' ' THEN                               02440000
  15               MOVE 'Y' TO NEWREQ OF CONPARM.                            02450000
                                                                             02460000
  12            IF NEWREQ OF COMPARM = 'N' AND PREV OF PCONVSTA = 'S'        02470000
  15               AND DATA01 NOT = ' '                                      02480000
  15               AND DATA1N NOT = 'NEXT'                                   02490000
  15               THEN MOVE 'Y' TO NEWREQ OF CONPARM.                       02500000
                                                                             02510000
  12            IF NEWREQ OF CONPARM NOT = 'Y' THEN                          02520000
  15               GO TO IC2010.                                             02530000
                                                                             02540000
   7       ************************************************************      02550000
   7       *  IF NEW REQUEST AND ACTION IS 'ADD' THEN                 *      02560000
   7       *       CALL DETAIL PROCESSOR                              *      02570000
   7       *  ELSE CALL SECONDARY SELECTION                           *      02580000
   7       ************************************************************      02590000
  12            IF ACTION OF INAREA = 'A' THEN                               02600000
   7       *                                        **DETAIL PROCESSOR        02610000
  15               GO TO DETAIL0.                                            02620000
   7       *                                     **SECONDARY SELECTION        02630000
  12            PERFORM SECSEL THRU END-SECSEL.                              02640000
   7       *                                  **IF NO. OF CHOICES = 1         02650000
   7       *                                  **GO TO DETAIL PROCESSOR        02660000
  12            IF MAXSEL = 1 THEN                                           02670000
  15               GO TO DETAIL0.                                            02680000
  12            GO TO EXIT0.                                                 02690000
   7       ************************************************************      02700000
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
   7          *  DETERMINES IF VALID SELECTION NUMBER                 *      02710000
   7          ***************************************************************02720000
   8           IC201D.                                                       02730000
   7          *                                        **VALID SELECTION NO. GIVEN 02740000
  12               IF PREV OF PCONVSTA NOT = 'S' OR                          02750000
  15                  MAXSEL < 1 OR                                          02760000
  15                  DATA1N = 'NEXT' OR                                     02770000
  15                  DATA2 = DATO2 THEN                                     02780000
  15                  GO TO IC201.                                           02790000
                                                                             02800000
   7          *                                          **DETAIL SELECTION GIVEN 02810000
  12               IF DAT1 NUMERIC AND DAT2 = ' ' THEN                       02820000
  15                  MOVE DAT1 TO DAT2                                      02830000
  15                  MOVE '0' TO DAT1.                                      02840000
  12               IF DATA2 NUMERIC                                          02850000
  15                  AND DATA2 > '00' AND DATA2 NOT > MAXSEL THEN           02860000
  15                  MOVE 'Y' TO NEWREQ OF COMPARM                          02870000
  15                  GO TO DETAIL0.                                         02880000
   7          *                                        **INVALID SELECTION NO. 02890000
   7          *                                        **PRINT ERROR MESSAGE   02900000
  12               MOVE '072E' TO MSGCODE.                                   02910000
  12               CALL 'DSN8MCG' USING MAJOR MSGCODE OUTMSG.                02920000
  12               MOVE OUTMSG TO MSG OF OUTAREA.                            02930000
  12               GO TO EXITO.                                              02940000
                                                                             02950000
   7          ***************************************************************02960000
   7          *  DETERMINES WHETHER SECONDARY SELECTION OR DETAIL     *      02970000
   7          ***************************************************************02980000
   8           IC201.                                                        02990000
   7          *                                          **SECONDARY SELECTION 03000000
  12               IF PREV OF PCONVSTA = 'S' THEN                            03010000
  15                  PERFORM SECSEL THRU END-SECSEL                         03020000
  15                  GO TO EXITO                                            03030000
  12               ELSE                                                      03040000
                                                                             03050000
   7          *                                          **DETAIL PROCESSOR    03060000
  15                  IF PREV OF PCONVSTA = 'D' THEN GO TO DETAIL0.          03070000
                                                                             03080000
   7          *                                          **LOGIC ERROR         03090000
   7          *                                          **PRINT ERROR MESSAGE 03100000
  12               MOVE '066E' TO MSGCODE.                                   03110000
  12               CALL 'DSN8MCG' USING MAJOR MSGCODE OUTMSG.                03120000
  12               MOVE OUTMSG TO MSG OF OUTAREA.                            03130000
  12               GO TO EXITO.                                              03140000
                                                                             03150000
   7          *                                          **HANDLES ERRORS      03160000
  12               EXEC SQL INCLUDE DSN8MCXX END-EXEC.                       03170000
  12               GO TO EXITO.                                              03180000
                                                                             03190000
   7          ***************************************************************03200000
   7          *  CALLS SECONDARY SELECTION AND RETURNS TO SQL 1       *      03210000
   7          ***************************************************************03220000
   8           SECSEL.                                                       03230000
  12               MOVE 'DSN8001' TO LASTSCR IN PCONVSTA.                    03240000
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

      7       *                                                **ADMINISTRATIVE          03250000
      7       *                                                **DEPARTMENT STRUCTURE    03260000
     12           IF OBJECT OF INAREA = 'DS' THEN                                       03270000
     15              PERFORM DSN8MCA THRU END-DSN8MCA                                   03280000
     12           ELSE                                                                  03290000
      7       *                                                **INDIVIDUAL DEPARTMENT   03300000
      7       *                                                **PROCESSING              03310000
     15              IF OBJECT OF INAREA = 'DE' THEN                                    03320000
     18                 PERFORM DSN8MCA THRU END-DSN8MCA                                03330000
     15              ELSE                                                               03340000
      7       *                                                **INDIVIDUAL EMPLOYEE     03350000
      7       *                                                **PROCESSING              03360000
     18                 IF OBJECT OF INAREA = 'EM' THEN                                 03370000
     21                    PERFORM DSN8MCA THRU END-DSN8MCA                             03380000
     18                 ELSE                                                            03390000
                                                                                        03400000
      7       *                                                **ERROR MESSAGE           03410000
      7       *                                                **MISSING SECONDARY SEL   03420000
     21                    MOVE '063E' TO MSGCODE                                      03430000
     21                    CALL 'DSN8MCG' USING MAJOR MSGCODE OUTMSG                   03440000
     21                    MOVE OUTMSG TO MSG OF OUTAREA                               03450000
     21                    GO TO EXITO.                                                03460000
      8       END-SECSEL.                                                            03470000
                                                                                        03480000
      7       ************************************************************            03490000
      7       *  CALLS DETAIL PROCESSOR AND RETURNS TO SQL 1               *           03500000
      7       ************************************************************            03510000
      8       DETAILO.                                                                03520000
     12           MOVE 'DSN8002' TO LASTSCR IN PCONVSTA.                              03530000
      7       *                                                **ADMINISTRATIVE          03540000
      7       *                                                **DEPARTMENT STRUCTURE    03550000
     12           IF OBJECT OF INAREA = 'DS' THEN                                       03560000
     15              PERFORM DSN8MCD THRU END-DSN8MCD                                   03570000
     12           ELSE                                                                  03580000
      7       *                                                **INDIVIDUAL DEPARTMENT   03590000
      7       *                                                **PROCESSING              03600000
     15              IF OBJECT OF INAREA = 'DE' THEN                                    03610000
     18                 PERFORM DSN8MCE THRU END-DSN8MCE                                03620000
     15              ELSE                                                               03630000
      7       *                                                **INDIVIDUAL EMPLOYEE     03640000
      7       *                                                **PROCESSING              03650000
     18                 IF OBJECT OF INAREA = 'EM' THEN                                 03660000
     21                    PERFORM DSN8MCF THRU END-DSN8MCF                             03670000
     18                 ELSE                                                            03680000
                                                                                        03690000
      7       *                                                **ERROR MESSAGE           03700000
      7       *                                                **MISSING DETAIL MODULE   03710000
     21                    MOVE '062E' TO MSGCODE                                      03720000
     21                    CALL 'DSN8MCG' USING MAJOR MSGCODE OUTMSG                   03730000
     21                    MOVE OUTMSG TO MSG OF OUTAREA.                              03740000
     12              GO TO EXITO.                                                     03750000
                                                                                        03760000
      7       *                                                **RETURNS TO SQL 1        03770000
      8       EXITO.                                                                  03780000
```

DATASET: DSN122.DSN3MP
MEMBER: DSN3CC2

DATE: 87/02/12
TIME: 17:13
PAGE: 3

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
   12             EXEC CICS RETURN END-EXEC.                              03790000
                                                                          03800000
   12             EXEC SQL INCLUDE DSN8MCA END-EXEC.                      03810000
   12             EXEC SQL INCLUDE DSN8MCD END-EXEC.                      03820000
   12             EXEC SQL INCLUDE DSN8MCE END-EXEC.                      03830000
   12             EXEC SQL INCLUDE DSN8MCF END-EXEC.                      03840000
   12             GOBACK.                                                 03850000
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7      ****DSN8MCS - VALIDATION MODULE FOR SEARCH CRITERIA - COBOL*****  00010000
   7      *                                                                 * 00020000
   7      *       MODULE NAME = DSN8MCS                                     * 00030000
   7      *                                                                 * 00040000
   7      *       DESCRIPTIVE NAME = DB2 SAMPLE APPLICATION                 * 00050000
   7      *                          VALIDATION MODULE FOR SEARCH CRITERIA  * 00060000
   7      *                          COBOL                                  * 00070000
   7      *                                                                 * 00080000
   7      *                                                                 * 00090000
   7      * COPYRIGHT = 5740-XYR (C) COPYRIGHT IBM CORP 1982, 1985          * 00100000
   7      * REFER TO COPYRIGHT INSTRUCTIONS FORM NUMBER G120-2083           * 00105000
   7      *                                                                 * 00110000
   7      * STATUS = RELEASE 2, LEVEL 0                                     * 00120000
   7      *                                                                 * 00140000
   7      *       FUNCTION = THIS MODULE VALIDATES SPECIFIC INPUT           * 00150000
   7      *                  AND MOVES IT TO THE OUTPUT MESSAGE             * 00160000
   7      *                  TOGETHER WITH A TEXT FIELD.                    * 00170000
   7      *                                                                 * 00130000
   7      *       NOTES = NONE                                              * 00190000
   7      *                                                                 * 00200000
   7      *                                                                 * 00210000
   7      *       MODULE TYPE =                                             * 00220000
   7      *           PROCESSOR    = DB2  PRECOMPILER, COBOL COMPILER       * 00230000
   7      *           MODULE SIZE = SEE LINKEDIT                            * 00240000
   7      *           ATTRIBUTES   =  NONE                                  * 00250000
   7      *                                                                 * 00260000
   7      *       ENTRY POINT = DSN8MCS                                     * 00270000
   7      *           PURPOSE = SEE FUNCTION                                * 00280000
   7      *           LINKAGE = INCLUDED BY MODULE DSN8CC1                  * 00290000
   7      *                                                                 * 00300000
   7      *           INPUT    = PARAMETERS EXPLICITLY PASSED TO THIS       * 00310000
   7      *                      FUNCTION:                                  * 00320000
   7      *                     SYMBOLIC LABEL/NAME =  NONE                 * 00330000
   7      *                     DESCRIPTION = NONE                          * 00340000
   7      *                                                                 * 00350000
   7      *           OUTPUT = PARAMETERS EXPLICITLY RETURNED:              * 00360000
   7      *                     SYMBOLIC LABEL/NAME =  NONE                 * 00370000
   7      *                     DESCRIPTION = NONE                          * 00380000
   7      *                                                                 * 00390000
   7      *       EXIT-NORMAL = THIS CODE IS "PERFORMED", SO IT EXITS TO    * 00400000
   7      *                     THE CODE FOLLOWING THE "PERFORM" STATEMENT  * 00410000
   7      *                                                                 * 00420000
   7      *       EXIT-ERROR = IF SQLERROR OR SQLWARNING, SQL WHENEVER      * 00430000
   7      *                    CONDITION SPECIFIED IN DSN8CC1/IC1 WILL BE   * 00440000
   7      *                    RAISED AND PROGRAM WILL GO TO THE LABEL      * 00450000
   7      *                    DB-ERROR.                                    * 00460000
   7      *                                                                 * 00470000
   7      *       RETURN CODE = NONE                                        * 00480000
   7      *                                                                 * 00490000
   7      *       ABEND CODES = NONE                                        * 00500000
   7      *                                                                 * 00510000
   7      *       ERROR MESSAGES =                                          * 00520000
   7      *             DSN8070E - VITAL DATA MISSING IN TABLE 'TOPTVAL'    * 00530000
   7      *                                                                 * 00540000
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
    7   *                                                                          * 00550000
    7   *      EXTERNAL REFERENCES = HOST VARIABLES ARE GLOBAL AND                  * 00560000
    7   *                            DEFINED IN DSN8CC1/IC1.                       * 00570000
    7   *       ROUTINES/SERVICES =                                                * 00580000
    7   *        DSN8MCG                - ERROR MESSAGE ROUTINE                    * 00590000
    7   *                                                                          * 00600000
    7   *       DATA-AREAS          =  NONE                                        * 00610000
    7   *                                                                          * 00620000
    7   *       CONTROL-BLOCKS      =                                              * 00630000
    7   *        SQLCA                  - SQL COMMUNICATION AREA                   * 00640000
    7   *                                                                          * 00650000
    7   *      TABLES = NONE                                                       * 00660000
    7   *                                                                          * 00670000
    7   *      CHANGE-ACTIVITY =  NONE                                             * 00680000
    7   *                                                                          * 00690000
    7   *                                                                          * 00700000
    7   *     *PSEUDOCODE*                                                         * 00710000
    7   *                                                                          * 00720000
    7   *      PROCEDURE                                                           * 00730000
    7   *      INITIALIZE RETURNCODE TO '0'.                                       * 00740000
    7   *                                                                          * 00750000
    7   *      FILL IN THE DISPLAY AREA                                            * 00760000
    7   *      FROM VOPTVAL (SEARCH,SELTXT)                                        * 00770000
    7   *          DEPENDING ON SEARCH REQUIRED                                    * 00780000
    7   *          RETURN.                                                         * 00790000
    7   *                                                                          * 00800000
    7   *      IF SEARCH CRITERIA NOT FOUND                                        * 00810000
    7   *          RETRIEVE A LIST OF SEARCH CRITERIA WHICH EXISTS,                * 00820000
    7   *          HEADTXT, INFOTXT AND PFKTXT                                     * 00830000
    7   *          FROM VOPTVAL                                                    * 00840000
    7   *          DEPENDING ON MAJSYS = MAJSYS , ACTION = ACTION ,                * 00850000
    7   *          OBJECT = OBJECT  AND SEARCH = BLANK                             * 00860000
    7   *          FILL IN DISPLAY AREA                                            * 00870000
    7   *          SET RETURNCODE TO '1'.                                          * 00880000
    7   *                                                                          * 00890000
    7   *   END.                                                                   * 00900000
    7   *                                                                          * 00910000
    7   *--------------------------------------------------------------------------* 00920000
                                                                                    00930000
    8    DSN8MCS.                                                                   00940000
                                                                                    00950000
    7   ****************************************************************         00960000
    7   *                  **  INITIALIZE RETURN CODE                  *         00970000
    7   ****************************************************************         00980000
   12        MOVE '0' TO RETCODE.                                                  00990000
   12        MOVE 'DSN8MCS' TO MAJOR IN DSN8-MODULE-NAME.                          01000000
                                                                                    01010000
    7   ****************************************************************         01020000
    7   *        **   LET'S SEE IF THE SEARCH CRITERIA SPECIFIED ON     T01030000
    7   *        **   INPUT EXISTS BY TRYING TO RETRIEVE SEARCH CRITERIAT01040000
    7   *        **   AND TEXT                                          T01050000
    7   ****************************************************************         01060000
                                                                                    01070000
    7   *                                    **RETRIEVAL                          01080000
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

  12              EXEC SQL SELECT SELTXT                                          01090000
  17                   INTO  :POPTVAL.SELTXT                                      01100000
  17                   FROM  VOPTVAL                                              01110000
  17                   WHERE MAJSYS    = :INAREA.MAJSYS                           01120000
  17                   AND   ACTION    = :INAREA.ACTION                           01130000
  17                   AND   OBJECT    = :INAREA.OBJECT                           01140000
  17                   AND   SRCHCRIT  = :INAREA.SRCH                             01150000
  17                   AND   SRCHCRIT <> ' '                                      01160000
  17                   AND  (SCRTYPE   = ' ' OR  SCRTYPE   = 'S'I                 01170000
  17                   END-EXEC.                                                  01180000
   7          *                                          **SEARCH CRITERIA EXIST 01190000
   7          *                                          **FILL IN DISPLAY AREA  01200000
                                                                                  01210000
  12              MOVE SELTXT    IN POPTVAL TO DESC4 IN OUTAREA.                 01220000
  12              MOVE SRCH      IN INAREA TO SRCH   IN OUTAREA.                 01230000
                                                                                  01240000
   7          *                                                  **RETURN        01250000
  12              IF   SQLCODE = +0  THEN                                         01260000
  12              GO TO END-DSN8MC5.                                              01270000
                                                                                  01280000
   7          ****************************************************************** 01290000
   7          *          **   SEARCH CRITERIA NOT FOUND                         01300000
   7          *          **   PROVIDE A LIST OF SEARCH CRITERIA WHICH EXIST     01310000
   7          ****************************************************************** 01320000
  12              MOVE SPACE TO SRCH  IN OUTAREA.                                 01330000
  12              MOVE SPACE TO DESC4 IN OUTAREA.                                 01340000
   7          *                                          **  OPEN CURSOR         01350000
  12              EXEC SQL OPEN V04 END-EXEC.                                     01360000
   7          *                                                                  01370000
  12              MOVE +1 TO I.                                                   01380000
                                                                                  01390000
   7          *                                          **RETRIEVE LIST OF      01400000
   7          *                                          **SEARCH CRITERIA       01410000
   8          MC5-10.                                                            01420000
  12              IF I NOT > 15 THEN                                              01430000
  16                  EXEC SQL FETCH V04 INTO :POPTVAL.SRCHCRIT ,                 01440000
  25                                  :POPTVAL.SELTXT END-EXEC                    01450000
  16                  IF SQLCODE IS NOT EQUAL TO +100 THEN                        01460000
  23                      MOVE SPACES                TO FIELD-1(I)            01470000
  23                      MOVE SRCHCRIT IN POPTVAL TO FIELD-2(I)                  01480000
  23                      MOVE SELTXT   IN POPTVAL TO FIELD-3(I)                  01490000
  23                      ADD 1 TO I                                              01500000
  23                      GO TO MC5-10.                                           01510000
   8          MC5-20.                                                            01520000
   7          *                                                  **CLOSE CURSOR  01530000
  12              EXEC SQL CLOSE V04 END-EXEC.                                    01540000
   7          *                                                                  01550000
  12              MOVE I TO J.                                                    01560000
                                                                                  01570000
   7          *                                                  **PUT BLANKS AT 01580000
   7          *                                                  **END OF LINE   01590000
   8          MC5-30.                                                            01600000
  12              IF J NOT > 15 THEN                                              01610000
  17                  MOVE SPACE TO LINE01(J)                                     01620000
```

```
 START
  COL   ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

  17                   ADD 1 TO J                                                01630000
  17                   GO TO MC5-30.                                             01640000
   7          *                                                                  01650000
   7          ****************************************************************** 01660000
   7          *     **  CHECK FOR CONDITION WHERE THERE ARE NO VALID ENTRIES */01670000
   7          ****************************************************************** 01680000
                                                                                 01690000
   7          *                                        **IF NO VALID ENTRY IN    01700000
   7          *                                        **OPTION VALIDATION       01710000
   7          *                                        **BASE TABLE (TOPTVAL)    01720000
   7          *                                        **TRY TO SET ERROR TEXT   01730000
  12               IF   I = 1 THEN MOVE '1' TO RETCODE                           01740000
                                                                                 01750000
  20                      MOVE '070E' TO MSGCODE                                 01760000
  20                      CALL 'DSN8MCG' USING MAJOR MSGCODE OUTMSG              01770000
  20                      MOVE OUTMSG TO MSGTEXT IN MSG                          01780000
                                                                                 01790000
   7          *                                            **RETURN              01800000
  20                      GO TO END-DSN8MC5.                                     01810000
   7          *                                                                  01820000
   7          ****************************************************************** 01830000
   7          *              ** IF ONLY THE SEARCH CRITERIA EXISTS THEN USE THE 01840000
   7          *              ** DEFAULT TO SET UP SEARCH CRITERIA AND            01850000
   7          *              ** DESCRIPTION IN OUTPUT                            01860000
   7          ****************************************************************** 01870000
                                                                                 01880000
  12               IF I = 2 AND SRCH IN INAREA = ' ' THEN                        01890000
  15                    MOVE '0' TO RETCODE                                      01900000
  15                    MOVE FIELD-2(I) TO SRCH    IN INAREA                     01910000
  15                    MOVE FIELD-2(I) TO SRCH    IN OUTAREA                    01920000
  15                    MOVE FIELD-3(I) TO DESC4   IN OUTAREA                    01930000
  15                    MOVE SPACE         TO LINEO (1)                          01940000
   7          *                                                **RETURN          01950000
  15                    GO TO END-DSN8MC5.                                       01960000
                                                                                 01970000
   7          ****************************************************************** 01980000
   7          *              **  SEARCH CRITERIA WAS NOT FOUND                   01990000
   7          ****************************************************************** 02000000
                                                                                 02010000
  12               MOVE 'I' TO RETCODE.                                          02020000
                                                                                 02030000
  12               EXEC SQL SELECT *                                             02040000
  17                   INTO  :POPTVAL                                            02050000
  17                   FROM  VOPTVAL                                             02060000
  17                   WHERE MAJSYS    = :INAREA.MAJSYS                          02070000
  17                   AND   ACTION    = :INAREA.ACTION                          02080000
  17                   AND   OBJECT    = :INAREA.OBJECT                          02090000
  17                   AND   SRCHCRIT =  ' '  '.                               02100000
  17                   END-EXEC.                                                 02110000
   7          *                                          **FILL IN DISPLAY AREA  02120000
   7          *                                          **WITH HEADING, PFKEY,  02130000
   7          *                                          **& MESSAGE INFO.       02140000
                                                                                 02150000
  12               MOVE HEADTXT IN POPTVAL TO HTITLE  IN OUTAREA.                02160000
```

```
DATASET: DSN120.DSNSAMP                                  DATE: 87/02/12
MEMBER:  DSN8MC5                                         TIME: 17:37
                                                         PAGE: 5
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

 12             MOVE INFOTXT IN POPTVAL TO MSG     IN OUTAREA.          02170000
 12             MOVE PFKTXT  IN POPTVAL TO PFKTEXT IN OUTAREA.          02180000
  7         *                                                          02190000
                                                                       02200000
  7         *                                  **RETURN TO             02210000
  7         *                                  **DSN8MC1 MODULE        02220000
  8       END-DSN8MC5.                                                 02230000
```

```
START
COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

 7   *************************************************************** 00010000
 7   *                                                             * 00020000
 7   *       MODULE NAME = DSNSMCXX                                * 00030000
 7   *                                                             * 00040000
 7   *       DESCRIPTIVE NAME = SQLERROR HANDLING MODULE            * 00050000
 7   *                          (COBOL VERSION                     * 00060000
 7   *                                                             * 00070000
 7   * COPYRIGHT = 5740-XYR (C) COPYRIGHT IBM CORP 1982, 1985      * 00080000
 7   * REFER TO COPYRIGHT INSTRUCTIONS FORM NUMBER G120-2083       * 00085000
 7   *                                                             * 00090000
 7   * STATUS = RELEASE 2, LEVEL 0                                 * 00100000
 7   *                                                             * 00120000
 7   *       FUNCTION = THIS MODULE IS ENTERED AS STANDARD ACTION  * 00130000
 7   *                  WHEN A 'SQLERROR' OR 'SQLWARNING' OCCURS   * 00140000
 7   *                                                             * 00150000
 7   *       NOTES =                                               * 00160000
 7   *                                                             * 00170000
 7   *       MODULE TYPE = BLOCK OF COBOL CODE                     * 00180000
 7   *                                                             * 00190000
 7   *       PROCESSOR = DB2  PRECOMPILER, COBOL COMPILER          * 00200000
 7   *                                                             * 00210000
 7   *       MODULE SIZE = SEE LINKEDIT                            * 00220000
 7   *                                                             * 00230000
 7   *       ATTRIBUTES = REUSEABLE                                * 00240000
 7   *                                                             * 00250000
 7   *       ENTRY POINT =                                         * 00260000
 7   *                                                             * 00270000
 7   *       PURPOSE = SEE FUNCTION                                * 00280000
 7   *                                                             * 00290000
 7   *       LINKAGE = *                                           * 00300000
 7   *                                                             * 00310000
 7   *       INPUT = PARAMETERS EXPLICITLY PASSED TO THIS FUNCTION:* 00320000
 7   *          SYMBOLIC LABEL/NAME =  N/A                         * 00330000
 7   *          DESCRIPTION = DDNAME                               * 00340000
 7   *                                                             * 00350000
 7   *       OUTPUT = PARAMETERS EXPLICITLY RETURNED:              * 00360000
 7   *                PCONVSTA.OUTPUT.LINE(*)                      * 00370000
 7   *          SYMBOLIC LABEL/NAME =  N/A                         * 00380000
 7   *          DESCRIPTION = N/A                                  * 00390000
 7   *                                                             * 00400000
 7   *       EXIT-NORMAL =                                         * 00410000
 7   *                                                             * 00420000
 7   *       EXIT-ERROR =                                          * 00430000
 7   *                                                             * 00440000
 7   *       RETURN CODE = N/A                                     * 00450000
 7   *          REASON CODE = N/A                                  * 00460000
 7   *          MESSAGE ID  = N/A                                  * 00470000
 7   *                                                             * 00480000
 7   *       ABEND CODES = N/A                                     * 00490000
 7   *                                                             * 00500000
 7   *       ERROR-MESSAGES =                                      * 00510000
 7   *                                                             * 00520000
 7   *       EXTERNAL REFERENCES =                                 * 00530000
 7   *       ROUTINES/SERVICES = MODULE DSNTIAR                    * 00540000
```

```
DATASET: DSN120.DSNSAMP                                        DATE: 87/02/12
MEMBER:  DSN8MCAX                                              TIME: 17:37
                                                               PAGE: 2
START
CHG  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
     7       *                                                          * 00550000
     7       *      DATA-AREAS =                                        * 00560000
     7       *      CONTROL-BLOCKS =                                    * 00570000
     7       *                                                          * 00580000
     7       *      TABLES =                                            * 00590000
     7       *                                                          * 00600000
     7       *      CHANGE-ACTIVITY =                                   * 00610000
     7       *                                                          * 00620000
     7       *      PSEUDOCODE  =                                       * 00630000
     7       *                                                          * 00640000
     7       * THIS CODE IS ENTERED AS STANDARD ACTION WHEN AN 'SQLEAADR' OA * 00650000
     7       * 'SQLWARNING' OCCURS.                                     * 00660000
     7       *                                                          * 00670000
     7       * INFOAMATION DESCAIBING IHE EAROR WILL BE PLACED IN THE DISPLAY* 00680000
     7       * AAEA OF THE OUTPUT MESSAGE / PCONVSTA.OUTPUT.LINE(*) /   * 00690000
     7       * IN THE FOLLOWING WAY:                                    * 00700000
     7       *                                                          * 00710000
     7       *   LINE  4  WILL BE BLANK                                 * 00720000
     7       *   LINE  5  CONTAINS A MESSAGE INCLUDING NAME (MAJOR AND MINOR)* 00730000
     7       *            OF THE MODULE WHERE THE ERROR OCCURRED        * 00740000
     7       *   LINE  6  WILL BE BLANK                                 * 00750000
     7       *   LINES 7-14 CONTAIN THE CONTENTS OF 'SQL COMMUNICATION AREA' * 00760000
     7       *                                                          * 00770000
     7       ************************************************************ 00780000
     8        DB-EAADA.                                                  00790000
    12            MOVE 'DSN8001' TO LASTSCR IN PCONVSTA.                 00800000
    12            MOVE '1'       TO EXITCODE.                            00810000
    12            MOVE  SPACES   TO LINE0(4), LINE0(6).                  00820000
    12            STRING '<==========    A SQLEAADR HAS OCCURAED IN MODULE: ',  00830000
    20                   MAJOA, SPACE, MINOR, '   ==========>'          00840000
    20                   DELIMITED BY SIZE                              00850000
    20                   INTO LINE0(5).                                 00860000
    12            CALL 'DSNIIAA' USING SQLCA ERROR-MESSAGE DATA-LEM.     00870000
    12            MOVE EAR-IEXT1 TO LINE0(7).                            00880000
    12            MOVE ERR-TEXT2 TO LINE0(8).                            00890000
    12            MOVE ERR-TEXT3 TO LINE0(9).                            00900000
    12            MOVE ERR-IEXT4 TO LINE0(10).                           00910000
    12            MOVE ERA-TEXT5 TO LINE0(11).                           00920000
    12            MOVE EAA-TEXT6 TO LINE0(12).                           00930000
    12            MOVE EAA-TEXT7 TO LINE0(13).                           00940000
    12            MOVE ERA-TEXT8 TO LINE0(14).                           00950000
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
   7  *********** DSN8MCF - DETAIL EMPLOYEE MODULE - COBOL **********  00010000
   7  *                                                                 00020000
   7  *  MODULE NAME      = DSN8MCF                                     00030000
   7  *                                                                 00040000
   7  *  DESCRIPTIVE NAME = DB2 SAMPLE APPLICATION                      00050000
   7  *                     DETAIL EMPLOYEE MODULE                      00060000
   7  *                     COBOL                                       00070000
   7  *                     ORGANIZATION                                00080000
   7  *                                                                 00090000
   7  * COPYRIGHT = 5740-XYR (C) COPYRIGHT IBM CORP 1982, 1985          00100000
   7  * REFER TO COPYRIGHT INSTRUCTIONS FORM NUMBER G120-2083           00105000
   7  *                                                                 00110000
   7  * STATUS = RELEASE 2, LEVEL 0                                     00120000
   7  *                                                                 00140000
   7  *    FUNCTION = THIS MODULE HANDLES THE DETAIL OPERATIONS          00150000
   7  *               FOR AN EMPLOYEE SUCH AS DISPLAY, ADD(INSERT),     00160000
   7  *               UPDATE, AND ERASE(DELETE) IN THE MAJOR            00170000
   7  *               SYSTEM ORGANIZATION.                              00180000
   7  *                                                                 00190000
   7  *    NOTES =                                                      00200000
   7  *       DEPENDENCIES = NONE                                       00210000
   7  *       RESTRICTIONS = THE VALID OPTIONS ARE:                     00220000
   7  *         .O-D-EM-EI,EN,DI,DN                                     00230000
   7  *         .O-A-EM-EI,EN,DI                                        00240000
   7  *         .O-U-EM-EI,EN,DI,DN                                     00250000
   7  *         .O-E-EM-EI,EN,DI,DN                                     00260000
   7  *                                                                 00270000
   7  *    MODULE TYPE =                                                00280000
   7  *       PROCESSOR    = DB2  PRECOMPILER, COBOL COMPILER          00290000
   7  *       MODULE SIZE = SEE LINK-EDIT                              00300000
   7  *       ATTRIBUTES  =  REUSABLE                                  00310000
   7  *                                                                 00320000
   7  *    ENTRY POINT =                                                00330000
   7  *       PURPOSE =  SEE FUNCTION                                   00340000
   7  *       LINKAGE =  MODULE CALLED BY                               00350000
   7  *         .DSNRMCA  FOR DISPLAY, AND FIRST STEP UPDATE OR ERASE   00360000
   7  *         .DSN8IC2  FOR FIRST STEP ADD, AND ALL SECOND STEPS.     00370000
   7  *                                                                 00380000
   7  *       INPUT = PARAMETERS EXPLICITLY PASSED TO THIS FUNCTION:    00390000
   7  *       COMMON AREA.                                              00400000
   7  *                                                                 00410000
   7  *                 SYMBOLIC LABEL/NAME = PCONVSTA.PREV             00420000
   7  *                 DESCRIPTION         = 'D' OR ' ' PREVIOUS REQUEST 00430000
   7  *                                                                 00440000
   7  *                 SYMBOLIC LABEL/NAME = .MAXSEL                   00450000
   7  *                 DESCRIPTION         = 1-13 NUMBER OF SELECTIONS 00460000
   7  *                                                                 00470000
   7  *                 SYMBOLIC LABEL/NAME = OUTAREA.OUTPUTO           00480000
   7  *                 DESCRIPTION         = SECONDARY SELECTION OUTPUT 00490000
   7  *                                                                 00500000
   7  *                 SYMBOLIC LABEL/NAME = COMPARM .NEWREQ           00510000
   7  *                 DESCRIPTION         = 'Y' OR 'N' NEW REQUEST    00520000
   7  *                                                                 00530000
   7  *                 SYMBOLIC LABEL/NAME = INAREA                    00540000
```

```
DATA SET: DSN120.DSNSAMP                                        DATE: 87/02/17
MEMBER:   DSN8MCF                                               TIME: 17:39
                                                                PAGE: 2
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

       /  *              DESCRIPTION          = USER INPUT                        00550000
       7  *                                                                      00560000
       7  *                                                                      00570000
       7  *     OUTPUT = PARAMETERS EXPLICITLY RETURNED:                         00580000
       7  *       COMMON AREA.                                                   00590000
       7  *                                                                      00600000
       7  *              SYMBOLIC LABEL/NAME = OUTAREA.OUTPUTO                   00610000
       7  *              DESCRIPTION         = SCREEN DETAIL OUTPUT              00620000
       7  *                                                                      00630000
       7  *              SYMBOLIC LABEL/NAME = PCONVSTA.PREV                     00640000
       7  *              DESCRIPTION         = '0' OR ' ' DEPENDING ON           00650000
       7  *                                    STEP NUMBER                       00660000
       7  *                                                                      00670000
       7  *                                                                      00680000
       7  *  EXIT-NORMAL =                                                       00690000
       7  *                                                                      00700000
       7  *  EXIT-ERROR =                                                        00710000
       7  *                                                                      00720000
       7  *     RETURN CODE =    NONE                                            00730000
       7  *                                                                      00740000
       7  *     ABEND CODES =    NONE                                            00750000
       7  *                                                                      00760000
       7  *     ERROR-MESSAGES =                                                 00770000
       7  *      DSN8001I EMPLOYEE NOT FOUND                                     00780000
       7  *      DSN8002I EMPLOYEE SUCCESSFULLY ADDED                            00790000
       7  *      DSN8003I EMPLOYEE SUCCESSFULLY ERASED                           00800000
       7  *      DSN8004I EMPLOYEE SUCCESSFULLY UPDATED                          00810000
       7  *      DSN8005E EMPLOYEE EXISTS ALREADY, ADD NOT DONE                  00820000
       7  *      DSN8006E EMPLOYEE DOES NOT EXIST, ERASE NOT DONE                00830000
       7  *      DSN8007E EMPLOYEE DOES NOT EXIST, UPDATE NOT DONE               00840000
       7  *      DSN8069E NO VALID SELECTIONS QUALIFY FOR THIS REQUEST           00850000
       7  *                                                                      00860000
       7  *                                                                      00870000
       7  *  EXTERNAL REFERENCES =                                               00880000
       7  *     ROUTINES/SERVICES =                                              00890000
       7  *        DSN8MCG              - ERROR MESSAGE ROUTINE                  00900000
       7  *                                                                      00910000
       7  *     DATA-AREAS =                                                     00920000
       7  *        DSN8MCCA             - SAMPLE COMMON AREA                     00930000
       7  *                                                                      00940000
       7  *     CONTROL-BLOCKS =                                                 00950000
       7  *        SQLCA                - SQL COMMUNICATION AREA                 00960000
       7  *                                                                      00970000
       7  *  TABLES =                                                            00980000
       7  *           VDEPT   = DEPARTMENT     TABLE VIEW                       00990000
       7  *           VEMPL   = EMPLOYEE       TABLE VIEW                       01000000
       7  *           VOPTVAL = VALID OPTIONS TABLE VIEW                        01010000
       7  *           VDSPTXT = DISPLAY TEXTS TABLE VIEW                        01020000
       7  *                                                                     01030000
       7  *  CHANGE-ACTIVITY = NONE                                             01040000
       7  *                                                                     01050000
       7  *                                                                     01060000
       7  *PSEUDOCODE*                                                         01070000
       7  *                                                                     01080000
                                                                               01090000
```

```
DATASET: DSN.V3.DSNSAMP                                  DATE: 87/02/12
MEMBER:  DSN    CF                                       TIME: 17:39
                                                         PAGE: 3
START
 COL    ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

       7     *    PROCEDURE                                                  01100000
       7     *     DECLARATIONS.                                             01110000
       7     *                                                               01120000
       7     *    INITIALIZATION.                                            01130000
       7     *     .CHECK IF OPTION IS VALID FOR THIS MODULE                 01140000
       7     *       MAJOR SYSTEM = 'D' AND OBJECT = 'EMP'                   01150000
       7     *     .IF NOT, RETURN WITH ERROR MSG DB9E INVALID REQUEST.      01160000
       7     *                                                               01170000
       7     *    STEP-1.                                                    01180000
       7     *     .FILL IN TEXT LINES (HEADER,INFORMATION AND PFK)          01190000
       7     *       FROM VOPTVAL DEPENDING ON ACTION REQUIRED.              01200000
       7     *     .IF NOT ADD, SAVE EMPLOYEE ID, DEPENDING ON MAXSEL.       01210000
       7     *       IF MAXSEL=1 EMPL-ID IS ON THE FIRST DETAIL LINE,        01220000
       7     *       IF MAXSEL>1 THE INPUT DATA CONTAINS THE DETAIL LINE     01230000
       7     *       NUMBER.                                                 01240000
       7     *     .GET DEPARTMENT AND EMPLOYEE FIELD NAMES,                 01250000
       7     *       FROM VDSPTXT.                                           01260000
       7     *     .IF DISPLAY OR DELETE ACTION,                             01270000
       7     *       PROTECT EVERY DETAIL INPUT FIELD.                       01280000
       7     *     .IF ADD OR UPDATE ACTION,                                 01290000
       7     *       PROTECT EMPLOYEE-ID AND ALL DEPARTMENT FIELDS,          01300000
       7     *       POSITION THE SCREEN CURSOR TO EMPLOYEE NAME FIELD.      01310000
       7     *     .IF ADD, UNPROTECT EMPLOYEE-ID FIELD,                     01320000
       7     *       MOVE USER INPUT TO CORRESPONDING OUTPUT DATA FIELD,      01330000
       7     *       PREV='D' AND RETURN.                                    01340000
       7     *     .AND FOR DISPLAY, UPDATE AND ERASE,                       01350000
       7     *       FETCH EMPLOYEE AND DEPARTMENT CURRENT VALUES,           01360000
       7     *       PREV='D' AND RETURN.                                    01370000
       7     *       OR MSG 'EMPLOYEE NOT FOUND' AND RETURN.                 01380000
       7     *                                                               01390000
       7     *    STEP-2.                                                    01400000
       7     *     .IF ADD, DO IT AND MSG                                    01410000
       7     *       EITHER 'EMPLOYEE ADDED SUCCESSFULLY'                    01420000
       7     *           OR 'EMPLOYEE EXISTS ALREADY, ADD NOT DONE'          01430000
       7     *       PREV=' ' AND RETURN.                                    01440000
       7     *     .IF UPDATE, DO IT AND MSG                                 01450000
       7     *       EITHER 'EMPLOYEE UPDATED SUCCESSFULLY'                  01460000
       7     *           OR 'EMPLOYEE DOES NOT EXIST, UPDATE NOT DONE'       01470000
       7     *       RETURN.                                                 01480000
       7     *     .IF ERASE, DO IT AND MSG                                  01490000
       7     *       EITHER 'EMPLOYEE ERASED SUCCESSFULLY'                   01500000
       7     *           OR 'EMPLOYEE DOES NOT EXIST, ERASE NOT DONE'        01510000
       7     *       PREV=' ' AND RETURN.                                    01520000
       7     *     .OR MSG DB9E INVALID REQUEST.                             01530000
       7     *       RETURN.                                                 01540000
       7     *    END.                                                       01550000
       7     *                                                               01560000
       7     ****************************************************************01570000
                                                                             01580000
       8      DSN8MCF.                                                       01590000
                                                                             01600000
       7     ****************************************************************01610000
       7     *   CHECKS IF OPTION IS VALID                                   01620000
       7     ****************************************************************01630000
```

```
DATASET: DSN120.DSNSAMP                                  DATE: 87/02/12
MEMBER:  DSN8MCF                                         TIME: 17:39
                                                         PAGE: 4
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
   7      *                                  **INITIALIZE VARIABLES                  01640000
  12          MOVE 'DSN8MCF' TO MAJOR.                                               01650000
  12          MOVE SPACES    TO MINOR.                                               01660000
   7      *                                  **IS OPTION VALID?                      01670000
   7      *                                  **MAJOR SYSTEM=D                        01680000
   7      *                                  **OBJECT=EM                             01690000
  12          IF MAJSYS OF INAREA NOT = 'D' OR                                       01700000
  15             OBJECT OF INAREA NOT = 'EM' THEN                                    01710000
  15             MOVE 1 TO I                                                         01720000
  15             GO TO MCFNSUP.                                                      01730000
  12          IF ACTION OF INAREA = 'D' THEN                                         01740000
  15             GO TO MCF1-STEP.                                                    01750000
  12          IF NEWREQ = 'N' THEN                                                   01760000
  15             GO TO MCF2-STEP.                                                    01770000
  12          IF NEWREQ NOT = 'Y' THEN                                               01780000
  15             MOVE 2 TO I                                                         01790000
  15             GO TO MCFNSUP.                                                      01800000
                                                                                     01810000
   7      ********************************************************************** 01820000
   7      *  FETCHES AND PROTECTS FIELDS FOR A CERTAIN COMMAND                     01830000
   7      ********************************************************************** 01840000
   8      MCF1-STEP.                                                             01850000
  12          MOVE 'STEP-1'  TO MINOR.                                           01860000
   7      *                                  **FETCH FIELDS FOR                      01870000
   7      *                                  ** A CERTAIN REQUEST                    01880000
  12          EXEC SQL SELECT *                                                  01890000
  13           INTO :POPTVAL  FROM VOPTVAL                                       01900000
  13           WHERE MAJSYS='D'                                                  01910000
  15             AND ACTION=:INAREA.ACTION                                       01920000
  15             AND OBJECT='EM'                                                 01930000
  15             AND SCRTYPE='D'                                                 01940000
  15             AND SRCHCRIT='EI'                                               01950000
  12          END-EXEC.                                                          01960000
   7      *                                  **ERROR?                                01970000
  12          IF SQLCODE = +100 THEN                                             01980000
  15             MOVE OPTNF TO MSG OF OUTAREA                                    01990000
  15             GO TO END-DSN8MCF.                                              02000000
                                                                                 02010000
   7      *                                  **FILL IN TEXT LINES                    02020000
   7      *                                  **(HEADING,MESSAGE,PFKEYS)              02030000
  12          MOVE HEADTXT OF POPTVAL TO HTITLE.                                 02040000
  12          MOVE INFOTXT OF POPTVAL TO MSG OF OUTAREA.                         02050000
  12          MOVE PFKTXT  OF POPTVAL TO PFKTEXT OF OUTAREA.                     02060000
                                                                                 02070000
   7      *                                  **SAVE EMPLOYEE ID                      02080000
   7      *                                  **ON FIRST DETAIL LINE                  02090000
  12          IF ACTION OF INAREA = 'A' THEN                                     02093000
  15             GO TO MCF010.                                                   02096000
  12          IF MAXSEL = 1 THEN                                                 02100000
  15             MOVE MGRNUM(1) TO EMPNO OF EMPL                                 02110000
  15             GO TO MCF010.                                                   02130000
  12          IF MAXSEL < 1 THEN                                                 02140000
  15             MOVE 3 TO I                                                     02150000
  15             GO TO MCFNSUP.                                                  02160000
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

          IF DAT1 NOT NUMERIC THEN                                      02170000
             MOVE 4 TO I                                                02180000
             GO TO MCFNSUP.                                             02190000
          IF DAT2 NOT NUMERIC THEN                                      02200000
             MOVE DAT1 TO DAT2                                          02210000
             MOVE '0' TO DAT1.                                          02220000
                                                                        02230000
     *                                          **INPUT DATA CONTAINS  02240000
     *                                          **THE DETAIL LINE NO.  02250000
      MCF005.                                                           02260000
          MOVE DATA2 TO I.                                              02270000
          IF I > MAXSEL THEN                                            02280000
             MOVE 5 TO I                                                02290000
             GO TO MCFNSUP.                                             02300000
     *                                          **SAVE EMPLOYEE ID     02310000
          MOVE HORNUM(I) TO EMPNO OF PEMPL.                             02320000
                                                                        02330000
     *                                        **CLEAR FIELD WITH BLANKS 02340000
      MCF010.                                                           02350000
          MOVE 0 TO I.                                                  02360000
      MCF012.                                                           02370000
          ADD 1 TO I.                                                   02380000
          MOVE SPACES TO LINEO(I).                                      02390000
     *                                        **MCF012 LOOP            02400000
      MCF-LOOP12.                                                       02410000
          PERFORM MCF012                                                02420000
             UNTIL I > 14.                                              02430000
     *                                          **OPEN DH CURSOR       02440000
          EXEC SQL OPEN DH END-EXEC.                                    02450000
          MOVE 0 TO I.                                                  02460000
                                                                        02470000
     *                                          **GET DEPARTMENT &     02480000
     *                                          **EMPLOYEE FIELD NAMES 02490000
     *                                          **FROM DISPLAY LINE    02500000
      MCF014.                                                           02510000
          ADD 1 TO I.                                                   02520000
          EXEC SQL FETCH DH                                             02530000
                   INTO :PDSPTXT.DSPLINE, :PDSPTXT.LINENO               02540000
          END-EXEC.                                                     02550000
                                                                        02560000
          IF SQLCODE NOT = +100 THEN                                    02570000
             MOVE DSPLINE TO FIELD1(I)                                  02580000
             IF I < 10 THEN                                             02590000
                GO TO MCF014.                                           02600000
                                                                        02610000
      MCF015.                                                           02620000
     *                                          **CLOSE DH CURSOR      02630000
          EXEC SQL CLOSE DH END-EXEC.                                   02640000
                                                                        02650000
          IF I = 1 THEN                                                 02660000
            MOVE DSPMF TO MSG OF OUTAREA                                02670000
            GO TO END-DSN8MCF.                                          02680000
     *                                        **PROTECT THE MODIFIABLE 02690000
     *                                        **ATTRIBUTE FIELDS       02700000
```

```
DATASET: DSN120.DSNSAMP                                        DATE: 87/02/12
MEMBER:  DSN4MCF                                               TIME: 17:39
                                                               PAGE: 6
START
COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
                                                                          02710000
      7       *                                **REPLACE PROTECTED PRE-MODIFIED   02720000
      7       *                                ***225 = X'00E1'                   02730000
     12           MOVE 0 TO I.                                                   02740000
                                                                                 02750000
      8       MCF016.                                                           02760000
     12           ADD 1 TO I.                                                    02770000
     12           MOVE +225 TO ATTRI(I).                                         02780000
                                                                                 02790000
      7       *                                     **MCF016 LOOP                02800000
      8       MCF-LOOP16.                                                       02810000
     12           PERFORM MCF016                                                 02820000
     15             UNTIL I > 14.                                               02830000
      7       *                                **IF DISPLAY OR ERASE ACTION       02840000
      7       *                                **PROTECT EVERY DETAIL             02850000
      7       *                                **INPUT FIELD                      02860000
     12           IF ACTION OF INAREA = 'D' OR                                   02870000
     15             ACTION OF INAREA = 'E' THEN                                  02880000
     15             GO TO MCF030.                                               02890000
                                                                                 02900000
      7       *                                  **IF UPDATE OR ADD ACTION        02910000
      7       *                                  **PROTECT EMPLOYEE-ID            02920000
      7       *                                  **AND DEPARTMENT FIELDS          02930000
     12           IF ACTION OF INAREA = 'U' THEN                                 02940000
     15             GO TO MCF022.                                               02950000
     12           IF ACTION OF INAREA NOT = 'A' THEN                             02960000
     15             MOVE 6 TO I                                                 02970000
     15             GO TO MCFNSUP.                                              02980000
      7       *                                **IF ADD                          02990000
      7       *                                **UNPROTECT EMPLOYEE ID FIELD     03000000
     12           IF SRCH OF INAREA = 'EI' THEN                                  03010000
     15             MOVE DATA6 TO FIELD2(6)                                     03020000
     15             EXEC SQL SELECT EMPNO INTO :PEMPL.EMPNO                     03030000
     21                  FROM VEMPL WHERE EMPNO=:DATA6                          03040000
     15           END-EXEC                                                      03050000
                                                                                 03060000
      7       *                                     **DOES EMPLOYEE             03070000
      7       *                                     **EXIST ALREADY?            03080000
     15           IF SQLCODE = 0 THEN                                           03090000
     18             GO TO MCF038                                                03100000
     15           ELSE                                                          03110000
     18             GO TO MCF020.                                               03120000
                                                                                 03130000
      7       *                                        **EMPLOYEE NAME          03140000
     12           IF SRCH OF INAREA = 'EN' THEN                                 03150000
     15             MOVE DATA15 TO FIELD2(9)                                    03160000
     15             GO TO MCF020.                                               03170000
      7       *                                     **DEPARTMENT ID             03180000
     12           IF SRCH OF INAREA NOT = 'DI' THEN                             03190000
     15             MOVE 7 TO I                                                 03200000
     15             GO TO MCFNSUP.                                              03210000
     12           MOVE DATA3 TO FIELD2(10).                                     03220000
                                                                                 03230000
      7       *                          ** REPLACE UNPROTECTED PRE-MODIFIED    03240000
```

```
DATASET: DSN120.DSNSAM                                         DATE: 87/02/12
MEMBER:  DSN8MCF                                               TIME: 17:39
                                                               PAGE: 7
 START
 COL   ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

  7        *                            ** +193 = X'00C1'
  8         MCF02[illegible].
 12             MOVE +193 TO ATTR(6).
  8         MCF022.
 12             MOVE 6 TO I.
  8         MCF024.
 12             ADD 1 TO I.
 12             MOVE +193 TO ATTR(I).
  7        *                                          **MCF024 LOOP
  8         MCF-LOOP24.
 12             PERFORM MCF024
 15                UNTIL I > 9.

  7        *                                       **CURSOR POSITION
  7        *                                       ** -16191 = X'COC1'
 12             MOVE -16191 TO ATTR(7).
 12             IF ACTION OF INAREA = 'A' THEN
 15                GO TO MCFRET1.
  7        ****************************************************************
  7        *  ADDS, UPDATES, OR ERASES AND PRINTS A MESSAGE
  7        ****************************************************************
  8         MCF030.
 12             MOVE EMPNO OF PEMPL TO FIELD2(6).
 12             MOVE SPACES TO FIRSTNME-TEXT OF PEMPL,
 27                            LASTNAME-TEXT OF PEMPL.

  7        *                                                 **FETCH EMPLOYEE
 12             EXEC SQL SELECT *
 15                INTO :PEMPL FROM VEMPL
 15                WHERE EMPNO=:PEMPL.EMPNO
 12             END-EXEC.

  7        *                                            **EMPLOYEE NOT FOUND
 12             IF SQLCODE = +100 THEN
 15                MOVE '001I' TO MSGCODE
 15                GO TO MCFMSG.

 12             MOVE WORKDEPT OF PEMPL TO FIELD2(1), FIELD2(10).
 12             MOVE FIRSTNME-TEXT OF PEMPL TO FIELD2(7).
 12             MOVE MIDINIT  OF PEMPL TO FIELD2(8).
 12             MOVE LASTNAME-TEXT OF PEMPL TO FIELD2(9).
 12             MOVE SPACES TO DEPTNAME-TEXT OF PDEPT.

  7        *                                              **FETCH DEPARTMENT
 12             EXEC SQL SELECT *
 15                INTO :PDEPT FROM VDEPT
 15                WHERE DEPTNO=:PEMPL.WORKDEPT
 12             END-EXEC.

  7        *                                              **DEPARTMENT FOUND
 12             IF SQLCODE NOT = +100 THEN
 15                MOVE DEPTNAME-TEXT OF PDEPT TO FIELD2(2)
 15                MOVE MGRNO     OF PDEPT TO FIELD2(3)
 15                MOVE ADMRDEPT OF PDEPT TO FIELD2(4).
```

```
START
 COL   ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

  8        MCFRET1.
  7        *                                                      **RETURN
 12            MOVE '0' TO PREV.
 12            GO TO END-DSN8MCF.

  8        MCF2-STEP.
 12            MOVE 'STEP-2'  TO MINOR.
 12            MOVE 0 TO I.
  8        MCF032.
 12            ADD 1 TO I.
 12            MOVE +225 TO ATTR(I).
 12            MOVE TRANDATA(I) TO FIELD2(I).
  7        *                                              **MCF032 LOOP
  8        MCF-LOOP32.
 12            PERFORM MCF032
 15               UNTIL I > 14.

 12            MOVE TRANDATA(6) TO EMPNO OF PEMPL.
 12            IF ACTION OF INAREA = 'E' THEN
 15               GO TO MCF050.
 12            MOVE TRANDATA(7) TO FIRSTNME-TEXT OF PEMPL, WORK.
 12            MOVE 12  TO I.

  7        *                                             **CALCULATE FIRST NAME
  7        *                                             **LENGTH
  8        MCF034.
 12            IF WRK(I) = ' ' THEN
 15               SUBTRACT 1 FROM I
 15               IF I > 1 THEN
 18                  GO TO MCF034.

  8        MCF035.
 12            MOVE I TO FIRSTNME-LEN OF PEMPL.
 12            MOVE TRANDATA(8) TO MIDINIT OF PEMPL.
 12            MOVE TRANDATA(9) TO LASTNAME-TEXT OF PEMPL, WORK.
 12            MOVE 15  TO I.

  7        *                                             **CALCULATE LAST NAME
  7        *                                             **LENGTH
  8        MCF036.
 12            IF WRK(I) = ' ' THEN
 15               SUBTRACT 1 FROM I
 15               IF I > 1 THEN
 18                  GO TO MCF036.

  8        MCF037.
 12            MOVE I TO LASTNAME-LEN OF PEMPL.
 12            MOVE TRANDATA(10) TO WORKDEPT OF PEMPL.
 12            IF ACTION OF INAREA NOT = 'A' THEN GO TO MCF040.

  7        ***********************************************************************
  7        *                            ** INSERT
  7        ***********************************************************************
```

```
START
 COL   ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

 12           EXEC SQL WHENEVER SQLERROR CONTINUE END-EXEC.

  7     *                                          **PERFORM INSERT
 12           EXEC SQL INSERT INTO VEMPL
 19                  (EMPNO,FIRSTNME,MIDINIT,LASTNAME,WORKDEPT)
 15            VALUES(:PEMPL.EMPNO,:PEMPL.FIRSTNME,:PEMPL.MIDINIT,
 20                  :PEMPL.LASTNAME,:PEMPL.WORKDEPT)
 12           END-EXEC.

 12           IF SQLCODE = 0 THEN

  7     *                                   ** EMPLOYEE SUCCESSFULLY ADDED

 15              MOVE ' ' TO PREV
 15              MOVE '002I' TO MSGCODE
 15              GO TO MCF041.

 12           IF SQLCODE NOT = -803 THEN GO TO DB-ERROR.

  7     *                                         ** EMPLOYEE EXISTS ALREADY,
  7     *                                         ** ADD NOT DONE

  8        MCF038.
 12           EXEC SQL WHENEVER SQLERROR GO TO DB-ERROR END-EXEC.
 12           MOVE '005E' TO MSGCODE
 12           GO TO MCFMSG.
  8        MCF040.
  7     ***************************************************************
  7     *                     ** UPDATE
  7     ***************************************************************
 12           IF ACTION OF INAREA NOT = 'U' THEN
 15              MOVE 8 TO I
 15              GO TO MCFNSUP.
  7     *                                            **PERFORM UPDATE
 12           EXEC SQL UPDATE VEMPL
 15              SET FIRSTNME=:PEMPL.FIRSTNME,MIDINIT=:PEMPL.MIDINIT,
 19                  LASTNAME=:PEMPL.LASTNAME,WORKDEPT=:PEMPL.WORKDEPT
 15              WHERE EMPNO=:PEMPL.EMPNO
 12           END-EXEC.

 12           IF SQLCODE = +100 THEN

  7     *                                          ** EMPLOYEE DOES NOT EXIST,
  7     *                                          ** UPDATE NOT DONE

 15              MOVE '007E' TO MSGCODE
 15              GO TO MCFMSG.

  7     *                                 ** EMPLOYEE SUCCESSFULLY UPDATED

 12           MOVE '004I' TO MSGCODE.
  8        MCF041.
 12           MOVE WORKDEPT OF PEMPL TO FIELD2(1).
```

```
START
 CBL ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
  12             MOVE SPACES TO DEPTNAME-TEXT OF PDEPT.                    04870000
  12             EXEC SQL SELECT *                                         04880000
  15                INTO :PDEPT FROM VDEPT                                 04890000
  15                WHERE DEPTNO=:WORKDEPT                                 04900000
  12             END-EXEC.                                                 04910000
                                                                           04920000
  12             IF SQLCODE = 0 THEN                                       04930000
  15                MOVE DEPTNAME-TEXT OF PDEPT TO FIELD2(2).              04940000
  15                MOVE MGRNO    OF PDEPT TO FIELD2(3)                    04950000
  15                MOVE ADMRDEPT OF PDEPT TO FIELD2(4)                    04960000
  15                GO TO MCFMSG.                                          04970000
  12             MOVE 1 TO I.                                              04980000
   7          *                                    **PUT SPACES AT END OF FIELD 04990000
   8           MCF042.                                                     05000000
  12             ADD 1 TO I.                                               05010000
  12             MOVE SPACES TO FIELD2(I).                                 05020000
   7          *                                          **MCF042 LOOP     05030000
   8           MCF-LOOP42.                                                 05040000
  12             PERFORM MCF042                                            05050000
  15                UNTIL I > 3.                                           05060000
  12             GO TO MCFMSG.                                             05070000
   8           MCF050.                                                     05080000
   7          ************************************************************ 05090000
   7          *                    ** ERASE                                05100000
   7          ************************************************************ 05110000
   7          *                                        **PERFORM ERASE     05120000
  12             EXEC SQL DELETE FROM VEMPL                                05130000
  15                WHERE EMPNO=:PEMPL.EMPNO                               05140000
  12             END-EXEC.                                                 05150000
                                                                           05160000
   7          *                                  ** EMPLOYEE SUCCESSFULLY ERASED 05170000
  12             IF SQLCODE = 0 THEN                                       05180000
  15                MOVE ' ' TO PREV                                       05190000
  15                MOVE '0031' TO MSGCODE.                                05200000
                                                                           05210000
   7          *                                      ** EMPLOYEE DOES NOT EXIST, 05220000
   7          *                                      ** ERASE NOT DONE     05230000
  12             IF SQLCODE = +100 THEN                                    05240000
  14                MOVE '0068' TO MSGCODE.                                05250000
                                                                           05260000
  12             IF SQLCODE = 0 OR +100 THEN                               05270000
  15                GO TO MCFMSG.                                          05280000
                                                                           05290000
   7          *                                **ERROR - INVALID REQUEST     05300000
                                                                           05310000
   8           MCFNSUP.                                                    05320000
  12             MOVE '069E' TO MSGCODE.                                   05330000
   7          ************************************************************ 05340000
   7          *                    ** PRINT MESSAGE                        05350000
   7          ************************************************************ 05360000
   8           MCFMSG.                                                     05370000
  12             CALL 'DSN8MCG' USING MAJOR MSGCODE OUTMSG.                05380000
  12             MOVE OUTMSG TO MSGTXT OF MSG.                             05390000
                                                                           05400000
```

```
START
COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
  8          END-DSNSMCF.                                              05410000
```

```
START
COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
```

```
      7  * DSN8MCA - SQL 2 SECONDARY SELECTION FOR MAJOR SYSTEM 0 - COBOL *00010000
      7  *                                                                 00020000
      7  *   MODULE NAME = DSN8MCA                                         00030000
      7  *                                                                 00040000
      7  *   DESCRIPTIVE NAME = DB2  SAMPLE APPLICATION                    00050000
      7  *                      SQL 2 SECONDARY SELECTION                  00060000
      7  *                      CICS                                       00070000
      7  *                      COBOL                                      00080000
      7  *                      ORGANIZATION                               00090000
      7  *                                                                 00100000
      7  * COPYRIGHT = 5740-XYR (C) COPYRIGHT IBM CORP 1982, 1985          00110000
      7  * REFER TO COPYRIGHT INSTRUCTIONS FORM NUMBER G120-2083           00115000
      7  *                                                                 00120000
      7  * STATUS = RELEASE 2, LEVEL 0                                     00130000
      7  *                                                                 00150000
      7  *   FUNCTION = THIS MODULE PRODUCES A SECONDARY SELECTION SCREEN   00160000
      7  *              FOR OBJECTS IN MAJOR SYSTEM '0' (ORGANIZATION)     00170000
      7  *              CALLED BY DSN8IC2 (SQL2 MAINLINE)                  00180000
      7  *                                                                 00190000
      7  *   NOTES =  NONE                                                 00200000
      7  *                                                                 00210000
      7  *                                                                 00220000
      7  *   MODULE TYPE =  BLOCK OF COBOL CODE                            00230000
      7  *      PROCESSOR   = DB2  PRECOMPILER, COBOL COMPILER             00240000
      7  *      MODULE SIZE = SEE LINKEDIT                                 00250000
      7  *      ATTRIBUTES  = REUSABLE                                     00260000
      7  *                                                                 00270000
      7  *   ENTRY POINT = DSN8MCA                                         00280000
      7  *      PURPOSE = SEE FUNCTION                                     00290000
      7  *      LINKAGE = NONE                                             00300000
      7  *      INPUT  =                                                   00310000
      7  *               SYMBOLIC LABEL/NAME = COMMPTR                     00320000
      7  *               DESCRIPTION         = POINTER TO COMMAREA         00330000
      7  *                                     (COMMUNICATION AREA)        00340000
      7  *                                                                 00350000
      7  *      OUTPUT =                                                   00360000
      7  *               SYMBOLIC LABEL/NAME = COMMPTR                     00370000
      7  *               DESCRIPTION         = POINTER TO COMMAREA         00380000
      7  *                                     (COMMUNICATION AREA)        00390000
      7  *                                                                 00400000
      7  *   EXIT-NORMAL = END OF CODE BLOCK                               00410000
      7  *                                                                 00420000
      7  *   EXIT-ERROR = IF SQL ERROR OR SQL WARNING,                     00430000
      7  *                SQL WHENEVER CONDITION                           00440000
      7  *                SPECIFIED IN DSN8IC2 WILL BE RAISED AND PROGRAM  00450000
      7  *                WILL GO TO THE LABEL DB_ERROR.                   00460000
      7  *                                                                 00470000
      7  *                                                                 00480000
      7  *      RETURN CODE = NONE                                         00490000
      7  *                                                                 00500000
      7  *      ABEND CODES = NONE                                         00510000
      7  *                                                                 00520000
      7  *      ERROR MESSAGES =                                           00530000
      7  *        DSN8067E - UNSUPPORTED SEARCH CRITERIA FOR OBJECT        00540000
```

```
DATASET: DSN120.DSN5AMP                                  DATE: 87/02/12
MEMBER:  DSN8MCA                                         TIME: 17:24
                                                         PAGE: 2
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7      *          DSM8009E - NO VALID SELECTIONS QUALIFY FOR THIS REQUEST 00550000
   7      *          DSN8074E - DATA IS TOO LONG FOR SEARCH CRITERIA         00560000
   7      *                                                                  00570000
   7      *   EXTERNAL REFERENCES =                                          00580000
   7      *      ROUTINES/SERVICES =                                         00590000
   7      *        DSN8MCG                 - ERROR MESSAGE ROUTINE           00600000
   7      *                                                                  00610000
   7      *      DATA-AREAS =                                                00620000
   7      *        COMMAREA                - PGM COMMUNICATION AREA          00630000
   7      *                                                                  00640000
   7      *      CONTROL-BLOCKS =                                            00650000
   7      *        SQLCA                   - SQL COMMUNICATION AREA          00660000
   7      *                                                                  00670000
   7      *   TABLES = NONE                                                  00680000
   7      *                                                                  00690000
   7      *   CHANGE-ACTIVITY = NONE                                         00700000
   7      *                                                                  00710000
   7      *                                                                  00720000
   7      *  *PSEUDOCODE*                                                    00730000
   7      *        /* SECONDARY SELECTION FOR MAJSYS 'D' - OBJECTS           00740000
   7      *  1. DS - ADMINISTRATIVE LISTING                                  00750000
   7      *  2. DE - INDIVIDUAL DEPARTMENTS                                  00760000
   7      *  3. EM - INDIVIDUAL EMPLOYEES                                    00770000
   7      *  DS AND DE USE THE SAME CURSOR WHICH SELECTS DEPARTMENTS AND      00780000
   7      *  MANAGERS. EM USES ANOTHER CURSOR WHICH SELECTS DEPARTMENTS       00790000
   7      *  AND EMPLOYEES. THE FIELDS SELECTED BY THE TWO DIFFERENT          00800000
   7      *  CURSORS ARE THE SAME IN NUMBER AND HAVE MATCHING                 00810000
   7      *  CHARACTERISTICS. THEREFORE IT IS POSSIBLE TO USE THE SAME        00820000
   7      *  CODE FOR BOTH SITUATIONS MOST OF THE TIME.                       00830000
   7      *                                                                  00840000
   7      *  THERE ARE TWO SITUATIONS UNDER WHICH THIS MODULE CAN BE CALLED   00850000
   7      *    1. THE SYSTEM FIELDS HAVE CHANGED - NEW REQUEST               00860000
   7      *    2. AM ANSWER TO A PREVIOUS REQUEST                            00870000
   7      *  IF COMPARM.NEWREQ='Y' THEN SYSTEM FIELDS CHANGED AND            00880000
   7      *  THIS IS A NEW REQUEST                                           00890000
   7      *                                                                  00900000
   7      *  THIS MODULE SHOULD SET THE FOLLOWING TWO FIELDS BEFORE EXITING  00910000
   7      *    1. PCONVSTA.PREV='S'   (FOR NEXT TIME AROUND)                 00920000
   7      *    2. PCONVSTA.MAXSEL= NO. OF ENTRIES ON SEC SEL SCREEN BUILT    00930000
   7      *                                                                  00940000
   7      *   PROCEDURE                                                      00950000
   7      *     INITIALIZE TWO CONTROL FIELDS                                00960000
   7      *                                                                  00970000
   7      *     CASE(NEW REQUEST)                                            00980000
   7      *           INITIALIZE MINIMUM VALUES                              00990000
   7      *           ASSIGN FIELD VALUES FOR 'LIKE' IN SQL SELECT           01000000
   7      *           RETRIEVE HEADING LINE,PFK DESC,AND INFO MESSAGE        01010000
   7      *           RETRIEVE TEXT DESCRIPTION LINES                        01020000
   7      *     ENDCASE                                                      01030000
   7      *                                                                  01040000
   7      *     ASSIGN DATA VALUE FROM SCREEN FOR 'LIKE' PROCESSING          01050000
   7      *                                                                  01060000
   7      *     IF 'EM' SEARCH CRITERIA THEN                                 01070000
   7      *           OPEN EMPLOYEE CURSOR ASCENDING                         01080000
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
   7        *    ELSE                                                          01090000
   7        *         OPEN ADMIN ST CURSOR ASCENDING                           01100000
   7        *                                                                  01110000
   7        *    SET UP 'DO LOOP' VALUES                                       01120000
   7        *                                                                  01130000
   7        *    'FETCH' FROM THE APPROPRIATE CURSOR UP TO MAX OF 13 TIMES     01140000
   7        *                                                                  01150000
   7        *    IF NO VALID ENTRIES THEN                                      01160000
   7        *         SEND MESSAGE                                             01170000
   7        *                                                                  01180000
   7        *    SAVE MIN VALUE FOR POSSIBLE SCROLLING REQUEST                 01190000
   7        *                                                                  01200000
   7        *    RETURN                                                        01210000
   7        *                                                                  01220000
   7        * END.                                                            01230000
   7        ------------------------------------------------------------------01240000
                                                                               01250000
   8         DSN8MCA.                                                         01260000
                                                                               01270000
  12             MOVE 'DSN8MCA' TO MAJOR.                                     01280000
  12             MOVE  SPACES   TO MINOR.                                     01290000
                                                                               01300000
   7        ***************************************************************** 01310000
   7        *     **INITIALIZE CONTROL FIELDS                                 01320000
   7        ***************************************************************** 01330000
  12             MOVE 'S' TO PREV OF LASTPOS.                                 01340000
  12             MOVE  0  TO MAXSEL OF LASTPOS.                               01350000
  12             MOVE 0 TO I.                                                 01360000
                                                                               01370000
   7        *                                          **BLANK OUT LINE         01380000
   8         MCA010.                                                          01390000
  12             ADD 1 TO I.                                                  01400000
  12             MOVE SPACES TO LINE01(I).                                    01410000
                                                                               01420000
   7        *                                           **MCA010 LOOP          01430000
   8         MCA-LOOP10.                                                      01440000
  12             PERFORM MCA010                                               01450000
  16                 UNTIL I > 14.                                            01460000
                                                                               01470000
   7        ***************************************************************** 01480000
   7        *     **DETERMINE IF NEW REQUEST                                  01490000
   7        ***************************************************************** 01500000
                                                                               01510000
   7        *                                              **NEW REQUEST       01520000
  12             IF NEWREQ OF COMPARM = 'Y' THEN                              01530000
   7        *                                              ** INITIALIZE MINIMUM 01540000
   7        *                                              ** VALUES           01550000
  15                 MOVE LOW-VALUES  TO DIMIN, EIMIN                         01560000
   7        *                                              ** ASSIGN FIELD VALUES 01570000
   7        *                                              ** FOR LIKE SQL SELECT 01580000
  15                 MOVE 1          TO LDEPTNOL, LDEPTNAML, LMGRNOL, LMGRNAML, 01590000
  28                                    LEMPNOL,  LEMPNAML                     01600000
  15                 MOVE PERCENT TO LDEPTNOD, LDEPTNAMD, LMGRNOD, LMGRNAMED, 01610000
  28                                    LEMPNOD,  LEMPNAMED.                   01620000
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

                                                                          01630000
      7  ************************************************************** 01640000
      7  *   **RETRIEVES HEADING LINE, PFKEY DESCRIPTION, INFO MESSAGE,  01650000
      7  *    **& POINTER INTO TABLE OF DETAIL HEADING TEXT              01660000
      7  ************************************************************** 01670000
      8   MCA020.                                                        01680000
      7  *                                          **RETRIEVE INFORMATION 01690000
     12       EXEC SQL SELECT HEADTXT, INFOTXT, PFKTXT, DSPINDEX         01700000
     24                INTO :POPTVAL.HEADTXT, :POPTVAL.INFOTXT,         01710000
     29                     :POPTVAL.PFKTXT,  :POPTVAL.DSPINDEX         01720000
     24                FROM VOPTVAL                                     01730000
     24                WHERE     MAJSYS   = :INAREA.MAJSYS              01740000
     30                      AND ACTION   = :INAREA.ACTION              01750000
     30                      AND OBJECT   = :INAREA.OBJECT              01760000
     30                      AND SRCHCRIT = :INAREA.SRCH                01770000
     30                      AND SCRTYPE  = 'S'                         01780000
                                                                        01790000
      7  *                                          **ERROR?            01800000
     12       END-EXEC.                                                 01810000
     12       IF SQLCODE = +100 THEN                                    01820000
     15          MOVE ' ' TO PREV OF LASTPOS                            01830000
     15          MOVE OPTNF TO MSG OF OUTAREA                           01840000
     15          STRING MAJSYS OF INAREA SPACE, ACTION OF INAREA SPACE, 01850000
     22                 OBJECT OF INAREA SPACE, SRCH OF INAREA SPACE,   01860000
     22                 'S' DELIMITED BY SIZE                          01870000
     18             INTO MSGMOD2                                       01880000
     15          GO TO END-DSN8MCA.                                    01890000
                                                                        01900000
      7  *                                          **OBTAIN INFORMATION 01910000
     12       MOVE HEADTXT OF POPTVAL TO HTITLE.                        01920000
     12       MOVE INFOTXT OF POPTVAL TO MSG OF OUTAREA.                01930000
     12       MOVE PFKTXT  OF POPTVAL TO PFKTEXT OF OUTAREA.            01940000
                                                                        01950000
      7  ************************************************************** 01960000
      7  *  **RETRIEVES TEXT DESCRIPTION LINES                          01970000
      7  ************************************************************** 01980000
      7  *                                          **TRY TO            01990000
      7  *                                          **RETRIEVE INFORMATION 02000000
     12       EXEC SQL SELECT DSPLINE                                   02010000
     24                INTO :PDSPTXT.DSPLINE                           02020000
     24                FROM VDSPTXT                                    02030000
     24                WHERE DSPINDEX = :POPTVAL.DSPINDEX              02040000
     30                      AND LINENO ='01'                          02050000
     12       END-EXEC.                                                 02060000
                                                                        02070000
      7  *                                          **ERROR?            02080000
     12       IF SQLCODE = +100 THEN                                    02090000
     15          MOVE ' ' TO PREV OF LASTPOS                            02100000
     15          MOVE DSPNF TO MSG OF OUTAREA                           02110000
     15          STRING 'INDX ', DSPINDEX OF POPTVAL, ' L01',           02120000
     23                 DELIMITED BY SIZE                              02130000
     18             INTO MSGMOD2                                       02140000
     15          GO TO END-DSN8MCA.                                    02150000
                                                                        02160000
```

```
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

   7  *                                                **OBTAIN INFORMATION 02170000
  12       MOVE DSPLINE IO LINE0(1).                                        02180000
                                                                            02190000
   7  ************************************************************************02200000
   7  *  ** ASSIGN DATA VALUE FROM SCREEN FOR 'LIKE' PROCESSING              02210000
   7  *  **NOTE THAT ALL THE FOLLOWING SEARCH CRITERIA MAY NOT BE            02220000
   7  *  **SUPPORTED IN ALL SITUATIONS - HOWEVER SQL 1 WILL ONLY             02230000
   7  *  **PERMIT VALID ENTRIES TO BE PASSED.                                02240000
   7  ************************************************************************02250000
                                                                            02260000
  12       MOVE 60 TO I.                                                    02270000
   7  *                                                **SKIPS END BLANKS   02280000
   8  MCA022.                                                               02290000
  12       IF DATAIN1(I) = SPACE THEN                                       02300000
  15           SUBTRACT 1 FROM I                                            02310000
  15           IF I > 0 THEN                                                02320000
  19               GO TO MCA022.                                            02330000
                                                                            02340000
   8  MCA024.                                                               02350000
  12       IF SRCH OF INAREA = 'DI' THEN                                    02360000
   7  *                                                **DEPARTMENT IO      02370000
  15           MOVE I       TO LDEPTNOL                                     02380000
  15           MOVE DATAIN TO LDEPTNOD                                      02390000
  15           IF I > 3 THEN GO TO MCA025                                   02400000
  15           ELSE GO TO MCA001.                                           02410000
  12       IF SRCH OF INAREA = 'DN' THEN                                    02420000
   7  *                                                **DEPARTMENT NAME    02430000
  15           MOVE I       TO LDEPTNAL                                     02440000
  15           MOVE DATAIN TO LDEPTNAMD                                     02450000
  15           IF I > 36 THEN GO TO MCA025                                  02460000
  15           ELSE GO TO MCA001.                                           02470000
  12       IF SRCH OF INAREA = 'MI' THEN                                    02480000
   7  *                                                **MANAGER IO         02490000
  15           MOVE I       TO LMGRNOL                                      02500000
  15           MOVE DATAIN TO LMGRNOD                                       02510000
  15           IF I > 6 THEN GO TO MCA025                                   02520000
  15           ELSE GO TO MCA001.                                           02530000
  12       IF SRCH OF INAREA = 'MN' THEN                                    02540000
   7  *                                                **MANAGER NAME       02550000
  15           MOVE I       TO LMGRNAEL                                     02560000
  15           MOVE DATAIN TO LMGRNAED                                      02570000
  15           IF I > 15 THEN GO TO MCA025                                  02580000
  15           ELSE GO TO MCA001.                                           02590000
  12       IF SRCH OF INAREA = 'EI' THEN                                    02600000
   7  *                                                **EMPLOYEE IO        02610000
  15           MOVE I       TO LEMPNOL                                      02620000
  15           MOVE DATAIN TO LEMPNOD                                       02630000
  15           IF I > 6 THEN GO TO MCA025                                   02640000
  15           ELSE GO TO MCA001.                                           02650000
  12       IF SRCH OF INAREA = 'EN' THEN                                    02660000
   7  *                                                **EMPLOYEE NAME      02670000
  15           MOVE I       TO LEMPNAEL                                     02680000
  15           MOVE DATAIN TO LEMPNAED                                      02690000
  15           IF I > 15 THEN GO TO MCA025                                  02700000
```

```
DATASET: DSN120.QUYSA4P                                   DATE: 87/02/12
MEMBER:  DSN8MCA                                          TIME: 17:24
                                                          PAGE: 6
START
 COL  ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

  15                ELSE GO TO MCA031.                                        02710000
                                                                              02720000
   7        *                                          **UNSUPPORTED SEARCH   02730000
   7        *                                          **CRITERIA FOR OBJECT  02740000
   7        *                                          **PRINT ERROR MESSAGE  02750000
  12             MOVE '067E' TO MSGCODE.                                      02760000
  12             CALL 'DSN8MCG' USING MAJOR MSGCODE OUTMSG.                   02770000
  12             MOVE OUTMSG TO MSGTEXT OF MSG.                               02780000
  12             GO TO END-DSN8MCA.                                           02790000
                                                                              02800000
   7        *                                          **DATA TOO LONG        02810000
   7        *                                          **PRINT ERROR MESSAGE  02820000
   8         MCA025.                                                          02830000
  12             MOVE '074E' TO MSGCODE.                                      02840000
  12             CALL 'DSN8MCG' USING MAJOR MSGCODE OUTMSG.                   02850000
  12             MOVE OUTMSG TO MSGTEXT OF MSG.                               02860000
  12             MOVE ' ' TO PREV OF LASTPOS.                                 02865000
  12             GO TO END-DSN8MCA.                                           02870000
                                                                              02880000
   7        ***************************************************************02890000
   7        *  ** OPEN CURSORS                                              02900000
   7        ***************************************************************02910000
   8         MCA001.                                                          02920000
   7        *                                          **OPEN EMPLOYEE        02930000
   7        *                                          **CURSOR               02940000
  15                IF OBJECT OF INAREA = 'EM' THEN                           02950000
  18                  EXEC SQL OPEN EMA END-EXEC                              02960000
   7        *                                          **OPEN DEPARTMENT      02970000
   7        *                                          **CURSOR               02980000
  15                ELSE                                                      02990000
  18                   IF OBJECT OF INAREA = 'DE' THEN                        03000000
  21                     EXEC SQL OPEN DEA END-EXEC                           03010000
   7        *                                          **OPEN ALA             03020000
   7        *                                          **CURSOR               03030000
  13                   ELSE                                                   03040000
  21                     EXEC SQL OPEN ALA END-EXEC.                          03050000
                                                                              03060000
   8         MCA030.                                                          03070000
  12             MOVE 1 TO I.                                                 03080000
                                                                              03090000
   7        ***************************************************************03100000
   7        *  ** FETCH FROM THE APPROPRIATE CURSOR                         03110000
   7        ***************************************************************03120000
   8         MCA031.                                                          03130000
   7        *                                                 **EMPLOYEE     03140000
  15                IF OBJECT OF INAREA = 'EM' THEN                           03150000
  19                  MOVE SPACES TO DEPTNAME-TEXT IN PDEPT,                  03160000
  34                                 FIRSTNME-TEXT IN PEMPL,                  03170000
  34                                 LASTNAME-TEXT IN PEMPL                   03180000
  19                  EXEC SQL FETCH EMA                                      03190000
  20                    INTO :PDEPT.DEPTNO, :PDEPT.DEPTNAME, :PDEPT.MGRNO,    03200000
  22                      :PEMPL.FIRSTNME, :PEMPL.MIDINIT,:PEMPL.LASTNAME     03210000
  19                  END-EXEC                                                03220000
                                                                              03230000
```

```
START
 COL   ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8

  7          *                                                    **DEPARTMENT
 15                  ELSE
 19                      IF OBJECT OF INAREA = 'DE' THEN
 23                          MOVE SPACES TO DEPTNAME-TEXT IN PDEPT,
 38                                         FIRSTNME-TEXT IN PEMPL,
 38                                         LASTNAME-TEXT IN PEMPL
 23                          EXEC SQL FETCH DEA
 22                         INTO :PDEPT.DEPTNO, :PDEPT.DEPTNAME, :PDEPT.MGRNO,
 24                           :PEMPL.FIRSTNME, :PEMPL.MIDINIT,:PEMPL.LASTNAME
 23                          END-EXEC

  7          *                                                **NOT DEPARTMENT
  7          *                                                **OR EMPLOYEE
 19                      ELSE
 23                          MOVE SPACES TO DEPTNAME-TEXT IN PDEPT,
 38                                         FIRSTNME-TEXT IN PEMPL,
 38                                         LASTNAME-TEXT IN PEMPL
 23                          EXEC SQL FETCH ALA
 22                         INTO :PDEPT.DEPTNO, :PDEPT.DEPTNAME, :PDEPT.MGRNO,
 24                           :PEMPL.FIRSTNME, :PEMPL.MIDINIT,:PEMPL.LASTNAME
 23                          END-EXEC.

  7          *                                              ** GET INFORMATION
  8           MCA032.
 12               IF SQLCODE = +100 THEN GO TO MCA004.
 12               MOVE DEPTNO    OF PDEPT TO DEPTNUM OF BGMC1(I).
 12               MOVE DEPTNAME-TEXT OF PDEPT TO DEPTNA OF BGMC1(I).
 12               MOVE MGRNO          OF PDEPT TO MGRNUM  OF BGMC1(I).
 12               MOVE FIRSTNME-TEXT OF PEMPL TO MGRFIN  OF BGMC1(I).
 12               MOVE MIDINIT  OF PEMPL TO MGRSIN  OF BGMC1(I).
 12               MOVE LASTNAME-TEXT OF PEMPL TO MGRLNAN OF BGMC1(I).
 12               MOVE I                   TO LINENO  OF BGMC1(I).
 12               ADD 1 TO MAXSEL.
 12               ADD 1 TO I.
 12               IF I NOT = 13 THEN
 17                    GO TO MCA031.

  7          *                                       **SAVE MINIMUM
  7          *                                       **EMPLOYEE NO.
 15                  IF OBJECT OF INAREA = 'EM' THEN
 18                     MOVE MGRNO  OF PDEPT TO EININ
 15                  ELSE
  7          *                                       **SAVE MINIMUM
  7          *                                       **DEPARTMENT NO.
 18                     MOVE DEPTNO OF PDEPT TO DININ.

  8           MCA004.

  7          *                                       **NO SELECTIONS QUALIFY
  7          *                                       **FOR THIS REQUEST
  7          *                                       **PRINT ERROR MESSAGE

 12               IF SQLCODE NOT = +100 OR MAXSEL > 0 THEN GO TO MCA090.
 12               MOVE '069E' TO MSGCODE.
```

```
START
 COL    ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
   12             CALL 'DSN8MCU' USING MAJOR *SQCODE OUTMSG.                    03780000
   12             MOVE OUTMSG TO MSGTEXT OF OUTAREA.                            03790000
   12             MOVE ' ' TO PREV OF LASTPOS.                                  03795000
                                                                                03800000
                                                                                03810000
    7       ************************************************************      03820000
    7       *  **CLOSE CURSORS AND RETURN                                     03830000
    7       ************************************************************      03840000
    8        MCA090.                                                            03850000
    7       *                                          **CLOSE EMPLOYEE      03860000
    7       *                                          **CURSOR              03870000
   15                IF OBJECT OF INAREA = 'EM' THEN                           03880000
   18                  EXEC SQL CLOSE EMA END-EXEC                             03890000
    7       *                                          **CLOSE DEPARTMENT    03900000
    7       *                                          **CURSOR              03910000
   15                ELSE                                                      03920000
   18                  IF OBJECT OF INAREA = 'DE' THEN                         03930000
   21                    EXEC SQL CLOSE DEA END-EXEC                           03940000
    7       *                                          **CLOSE ALA           03950000
    7       *                                          **CURSOR              03960000
   18                ELSE                                                      03970000
   21                  EXEC SQL CLOSE ALA END-EXEC.                            03980000
                                                                                03990000
    7       *                                          **RETURN              04000000
    8        END-DSN8MC0.                                                       04010000
```

Appendix C

Formal/DB2/ADR Database Entity Comparison

Formal Relational	DB2	ADR
Database	Database	Database
Relation	Table	File
View	View	Dataview
Tuple	Row	Record
Attribute	Column	Field
Domain	**	**

** The system has no term for domain. Domains are implicitly defined when the attributes (columns or fields) are declared.

Formal Relational/DB2/ADR Entity Comparison

TWO RELATIONAL DBMS: A COMPARISON

by

GARY F. GARTEN

B.S., Kansas State University, 1978

B.A., Kansas State University, 1979

B.S., Kansas State University, 1979

AN ABSTRACT OF A MASTER'S DISSERTATION

submitted in partial fulfillment of
of the requirements for the degree of

MASTER OF SCIENCE, COMPUTER SCIENCE

College of Arts and Science

KANSAS STATE UNIVERSITY

Manhattan, Kansas

1987

Abstract

Data Base Management Systems (DBMS) are playing an increasingly important role in the development of computer systems. Few studies comparing the relational systems available for mainframe computers exist. This work compares two commercially available systems, IBM's DB2 (Data Base 2) and Applied Data Research's Datacom/DB.

After a brief introduction to the subject of relational data base management, DB2 and Datacom/DB product families are compared based on product descriptions, current product prices, and system overviews. Further comparisons of the two DBMS product families are made by comparing a miniature application system implemented. Difficulties encountered in reproducing the DB2 mini-system at the ADR site are described in detail.