

WRDC-TR-90-8007
Volume V
Part 25



AD-A250 462



INTEGRATED INFORMATION SUPPORT SYSTEM (IISS)
Volume V - Common Data Model Subsystem
Part 25 - Neutral Data Manipulation Language (NDML) Precompiler
Generator Request Processor Driver Product Specification

M. Apicella, J. Slaton, B. Levi

Control Data Corporation
Integration Technology Services
2970 Presidential Drive
Fairborn, OH 45324-6209



September 1990

Final Report for Period 1 April 1987 - 31 December 1990

Approved for Public Release; Distribution is Unlimited

MANUFACTURING TECHNOLOGY DIRECTORATE
WRIGHT RESEARCH AND DEVELOPMENT CENTER
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO 45433-6533

92-13576

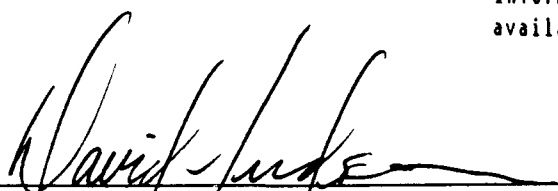


NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever, regardless whether or not the government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data. It should not, therefore, be construed or implied by any person, persons, or organization that the Government is licensing or conveying any rights or permission to manufacture, use, or market any patented invention that may in any way be related thereto.

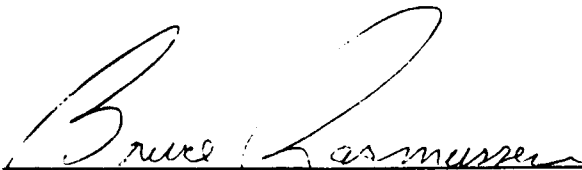
This technical report has been reviewed and is approved for publication.

This report is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations


DAVID L. JUDSON, Project Manager
WRDC/MTI
Wright-Patterson AFB, OH 45433-6533

25 July 91
DATE

FOR THE COMMANDER:


BRUCE A. RASMUSSEN, Chief
WRDC/MTI
Wright-Patterson AFB, OH 45433-6533

25 July 91
DATE

If your address has changed, if you wish to be removed from our mailing list, or if the addressee is no longer employed by your organization please notify WRDC/MTI, Wright-Patterson Air Force Base, OH 45433-6533 to help us maintain a current mailing list.

Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document.

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE				
1a. REPORT SECURITY CLASSIFICATION Unclassified		1b. RESTRICTIVE MARKINGS		
2a. SECURITY CLASSIFICATION AUTHORITY		3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for Public Release; Distribution is Unlimited.		
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE				
4. PERFORMING ORGANIZATION REPORT NUMBER(S) PS 620341261		5. MONITORING ORGANIZATION REPORT NUMBER(S) WRDC-TR-90-8007 Vol. V, Part 25		
6a. NAME OF PERFORMING ORGANIZATION Control Data Corporation; Integration Technology Services	6b. OFFICE SYMBOL (if applicable)	7a. NAME OF MONITORING ORGANIZATION WRDC/MTI		
6c. ADDRESS (City, State, and ZIP Code) 2970 Presidential Drive Fairborn, OH 45324-6209		7b. ADDRESS (City, State, and ZIP Code) WPAFB, OH 45433-6533		
8a. NAME OF FUNDING/SPONSORING ORGANIZATION Wright Research and Development Center, Air Force Systems Command, USAF	8b. OFFICE SYMBOL (if applicable) WRDC/MTI	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUM. F33600-87-C-0464		
8c. ADDRESS (City, State, and ZIP Code) Wright-Patterson AFB, Ohio 45433-6533		10. SOURCE OF FUNDING NOS.		
11. TITLE (Include Security Classification) NI See block 19		PROGRAM ELEMENT NO. 78011F	PROJECT NO. 595600	TASK NO. F95600 WORK UNIT NO. 20950607
12. PERSONAL AUTHOR(S) Control Data Corporation: Apicella, M. L., Slaton, J., Levi, B., Pashak, A.				
13a. TYPE OF REPORT Final Report	13b. TIME COVERED 4 / 1 / 87 - 12 / 31 / 90	14. DATE OF REPORT (Yr., Mo., Day) 1990 September 30		15. PAGE COUNT 33
16. SUPPLEMENTARY NOTES WRDC/MTI Project Priority 6203				
17. COSATI CODES		18. SUBJECT TERMS (Continue on reverse if necessary and identify block no.)		
FIELD	GROUP	SUB GR.		
1308	0905			
19. ABSTRACT (Continue on reverse if necessary and identify block number) This document establishes the design of Function PRE14, "Generate Request Process Driver", one of the major functions of the Configuration Item "Precompiler" to be built and formally accepted by the ICAM program office. BLOCK 11: INTEGRATED INFORMATION SUPPORT SYSTEM Vol V - Common Data Model Subsystem Part 25 - Neutral Data Manipulation Language (NDML) Precompiler Generator Request Processor Driver Product Specification				
20. DISTRIBUTION/AVAILABILITY OF ABSTRACT UNCLASSIFIED/UNLIMITED x SAME AS RPT. DTIC USERS		21. ABSTRACT SECURITY CLASSIFICATION Unclassified		
22a. NAME OF RESPONSIBLE INDIVIDUAL David L. Judson		22b. TELEPHONE NO. (Include Area Code) (513) 255-7371	22c. OFFICE SYMBOL WRDC/MTI	

FOREWORD

This technical report covers work performed under Air Force Contract F33600-87-C-0464, DAPro Project. This contract is sponsored by the Manufacturing Technology Directorate, Air Force Systems Command, Wright-Patterson Air Force Base, Ohio. It was administered under the technical direction of Mr. Bruce A. Rasmussen, Branch Chief, Integration Technology Division, Manufacturing Technology Directorate, through Mr. David L. Judson, Project Manager. The Prime Contractor was Integration Technology Services, Software Programs Division, of the Control Data Corporation, Dayton, Ohio, under the direction of Mr. W. A. Osborne. The DAPro Project Manager for Control Data Corporation was Mr. Jimmy P. Maxwell.

The DAPro project was created to continue the development, test, and demonstration of the Integrated Information Support System (IISS). The IISS technology work comprises enhancements to IISS software and the establishment and operation of IISS test bed hardware and communications for developers and users.

The following list names the Control Data Corporation subcontractors and their contributing activities:

SUBCONTRACTOR

ROLE

Control Data Corporation	Responsible for the overall Common Data Model design development and implementation, IISS integration and test, and technology transfer of IISS.
D. Appleton Company	Responsible for providing software information services for the Common Data Model and IDEF1X integration methodology.
ONTEK	Responsible for defining and testing a representative integrated system base in Artificial Intelligence techniques to establish fitness for use.
Simpact Corporation	Responsible for Communication development.
Structural Dynamics Research Corporation	Responsible for User Interfaces, Virtual Terminal Interface, and Network Transaction Manager design, development, implementation, and support.
Arizona State University	Responsible for test bed operations and support.

TABLE OF CONTENTS

		<u>Page</u>
SECTION 1.0	SCOPE	1-1
1.1	Identification	1-1
1.2	Functional Summary	1-1
SECTION 2.0	DOCUMENTS	2-1
2.1	Reference Documents	2-1
2.2	Terms and Abbreviations	2-1
SECTION 3.0	REQUIREMENTS	3-1
3.1	Structural Description	3-1
3.2	Functional Flow	3-1
3.3	Interfaces	3-1
3.3.1	Inputs/Outputs	3-1
3.4	Program Interrupts	3-2
3.5	Timing and Sequencing Description	3-2
3.6	Special Control Features	3-2
3.7	Storage Allocation	3-2
3.7.1	Database Definition	3-2
3.7.1.1	File Description	3-2
3.7.1.2	Table Description	3-2
3.7.1.3	Item Description	3-2
3.8	Object Code Creation	3-2
3.9	Adaptation Data	3-2
3.10	Detail Design Description	3-3
3.10.1	Where Include File Used List	3-3
3.10.2	Where External Routine Used List ...	3-4
3.10.3	Main Program Parts List	3-7
3.10.4	Module Documentation	3-10
3.10.5	Include File Description	3-17
3.10.6	Hierarchy Chart	3-19
3.11	Program Listings Comments	3-24
SECTION 4.0	QUALITY ASSURANCE PROVISIONS	4-1
4.1	Introduction and Definitions	4-1
4.2	Computer Programming and Test Evaluation	4-1

SECTION 1

SCOPE

1.1 Identification

This specification establishes the design of Function PRE14, "Generate Request Processor Driver", one of the major functions of the Configuration Item "Precompiler" to be built and formally accepted by the ICAM Program Office. This CI constitutes one of the subsystems of the Common Data Model Processor (CDMP).

1.2 Functional Summary

The purpose of this Computer Program Configuration Item (CPCI) is to generate a main program for each Request Processor needed to handle all of the NDML requests found in a user's logical unit of work grouping of user software modules.

The following functions will be performed by this CPCI:

1. Sort the table listing all generated Request Processors on database identifier.
2. For each database, generate a COBOL, FORTRAN, or C Request Processor driver module:
 - a) Use a macro to generate the Identification, Environment and Data Divisions of the program.
 - b) Use a macro to generate the case test and the call syntax in the Procedure Division of the program.
 - c) Use a macro for the error handling and termination of the program.

Accession For	
NTIS GRA&I	☒
DTIC TAB	☐
Unannounced	☐
Justification	
By	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A-1	

SECTION 2

DOCUMENTS

2.1 Reference Documents

1. ICAM Documentation Standards: IDS15012000A, 28 December 1981.
2. D. Appleton Co., CDM Administrators Manual; UM620141000, March 1984.
3. D. Appleton Co., CDM1-IDEF Model of the Common Data Model; CCS620141000, 15 May 1985.
4. D. Appleton Co., Computer Program Development Specification (DS) for ICAM Integrated Support System (IISS) Configuration Item: NDML Precompiler; DS620141200, October 1984.
5. D. Appleton Co., Embedded NDML Programmer's Reference Manual; PRM620141200, March 1985.
6. Softech, Inc., NTM Programmers Guide; UM620140001, July 1985.
7. Control Data Corp., Computer Program Development Specification (DS) for ICAM Integrated Support System (IISS) Configuration Item: NDDL Command Processor; DS620141100, June 1985.

2.2 Terms and Abbreviations

Attribute Use Class: (AUC)

Conceptual Schema: (CS)

Common Data Model Processor: (CDMP)

Common Data Model: (CDM) Describes common data application process formats, form definitions, etc, of the IISS and includes conceptual schema, external, internal schemas, and schema transformation operators.

Data Field: (DF) An element of data in the external schema. It is by this name that an NDML programmer references data.

Database Management System: (DBMS)

Distributed Request Supervisor: (DRS) This IISS CDM subsystem configuration item controls the execution of distributed NDML queries and non distributed updates.

Domain: A logical definition of legal attribute class values.

Domain Constraint: Predicate that applies to a single domain.

External Schema: (ES)

Forms: Structured views which may be imposed on windows or other forms. A form is composed of fields where each field is a form, item, or window.

Forms Processor: (FP) A set of callable execution time routines available to an application program for form processing.

Internal Schema: (IS)

Integrated Information Support System: (IISS) A test computing environment used to investigate, demonstrate and test the concepts of information management and information integration in the context of Aerospace Manufacturing. The IISS addresses the problems of integration of data resident on heterogeneous databases supported by heterogeneous computers interconnected via a local Area Network.

Mapping: The correspondence of independent objects in two schemas: ES to CS or CS to IS.

Network Transaction Manager: (NTM) Performs the coordination, communication and housekeeping functions required to integrate the application processes and system services resident on the various hosts into a cohesive system.

Neutral Data Manipulation Language: (NDML) A language developed by the IISS project to provide uniform access to common data, regardless of database manager or distribution criteria. It provides distributed retrieved and single node updates.

ORACLE: Relational DBMS based on the SQL (Structured Query Language, a product of ORACLE Corp, Menlo Park, CA). The CDM is an ORACLE database.

Parcel: A sequential file containing section source code of the input application program.

Request Processor: (RP) A COBOL program that will satisfy a retrieval or update NDML subtransaction against a particular Database Management System.

User Interface: (UI) Controls the user's terminal and interfaces with the rest of the system.

Virtual Terminal Interface: (VTI) Performs the interfacing between different terminals and the UI. This is done by defining a specific set of terminal features and protocols which

must be supported by UI software which constitutes the Virtual Terminal Definition. Specific terminals are then mapped against the Virtual Terminal software by specific software modules written for each type of real terminal supported.

SECTION 3

REQUIREMENTS

3.1 Structural Description

A graphic portrayal of this CPCI is included in Section 3.10. This chart shows the hierarchical relationship of each module making up this CPCI.

This CPCI uses a lower level module to write macros with the proper substitution parameters to the output file representing the generated Request Processor.

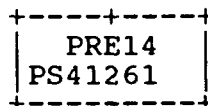
3.2 Functional Flow

This CPCI implements the logic defined in the Development Specification for this CPCI. Details of inputs/outputs and relationships between modules are to be found in Section 3.10.

The CPCI has been designated to operate in a batch or interactive mode. It must operate in the system environment established for IISS; that is, use of the Network Transaction Manager. It must use the ORACLE Database Management System installed on a DEC VAX computer.

3.3 Interface

The following diagram depicts the interface of PRE14 with other CPCI's in the system.



3.3.1 Inputs/Outputs

The following table depicts the inputs and outputs of this CPCI. A detail description for each item can be found in the DS for this CPCI.

Function: PRE14

<u>INPUT</u>	<u>OUTPUT</u>
None	

3.4 Program Interrupts

Not applicable to this CPCI.

3.5 Timing and Sequencing Description

This CPCI is a main module that calls several other modules for processing to generate a request processor driver program. It is assumed that all modules making up a logical unit of work are compiled as a single group.

3.6 Special Control Features

Not applicable to this CPCI.

3.7 Storage Allocation

3.7.1 Database Definition

The database used by this CPCI is the Common Data Model (CDM) database. This model is defined by the CDM1, the IDEF-1 model of the CDM, Reference Number 31.

3.7.1.1 File Description

No permanent files have been defined for this CPCI. It may use temporary scratch files for such things as generated program source code or temporary query results.

3.7.1.2 Table Description

All tables used by this CPCI have been defined by the Development Specification for this CPCI.

3.7.1.3 Item Description

Not applicable to this CPCI.

3.8 Object Code Creation

The object code for this CPCI will be created by the system integration test team by using defined IISS Software Configuration Management procedures. This CPCI will use the COBOL "C" language compilers.

3.9 Adaptation Data

This CPCI has been coded using ANSI COBOL. The intent was to provide a transportable system. Any system environment

supporting this language, a virtual memory management scheme, the COMM and NTM subsystems of IISS and the ORACLE Database Management System should be able to support this CPCI. Every possible attempt has been made to localize and identify any machine or environment dependent modules through the original design of the IISS and application of Configuration Management Procedures.

3.10 Detail Design Description

The following sections have been computer generated for this CPCI.

3.10.1 Where Include File Used List

The following lists each include file in the documentation group and all the modules documented in this specification which include them. The purpose of each module is listed as well.

DOCGROUP PS41261 Where-include-file-used List

Include File -----	Module Name -----
CHKCDM	CDCKNM CDELRPM CDP14 CDP14A CDRPSM
ERRCDM	CDCKNM CDELRPM CDP14 CDP14A CDRPSM
EOD	CDCKNM CDP14 CDP14A CDRPSM
ERRPRO	CDCKNM CDELRPM CDP14 CDP14A CDRPSM
ERRFS	CDP14
SBSTLST	CDP14 CDP14A
CGTABLE	CDP14

3.10.2 Where External Routine Used List

The following lists each external function or routine in the documentation group and all the documented modules which call it. The purpose of each module is listed as well.

DOCGROUP PS41261 Where-external-routine-used List

System Module -----	Module Name -----
SQLSCA	CDCKNM CDEL RPM CDP14 CDP14A CDRPSM
SQLBS1	CDCKNM CDEL RPM CDP14 CDP14A CDRPSM
SQLSCH	CDCKNM CDEL RPM CDP14 CDP14A CDRPSM
SQLSCC	CDCKNM CDEL RPM CDP14 CDP14A CDRPSM
SQLTFL	CDCKNM CDEL RPM CDRPSM
SQLOPN	CDCKNM CDEL RPM

DOCGROUP PS41261 Where-external-routine-used List

System Module -----	Module Name -----
SQLSQ	CDRPSM
	CDCKNM
	CDEL RPM
	CDP14
	CDP14A
	CDRPSM
SQLADR	
	CDCKNM
	CDEL RPM
	CDP14
	CDP14A
	CDRPSM
SQLAB1	
	CDCKNM
	CDEL RPM
	CDP14
	CDP14A
	CDRPSM
SQL EXE	
	CDCKNM
	CDEL RPM
	CDP14
	CDP14A
	CDRPSM
SQLAD1	
	CDCKNM
	CDP14
	CDP14A
SQLFCH	
	CDCKNM

DOCGROUP PS41261 Where-external-routine-used List

System Module -----	Module Name -----
ERRPRO	CDP14
	CDP14A
SQLWNR	CDCKNM
	CDEL RPM
	CDP14
	CDP14A
SQLLO1	CDRPSM
	CDEL RPM
CDDBMSS	CDRPSM
	CDP14
CDLKLUW	CDP14
	CDP14
SQLTOC	CDP14
	CDP14A
SQLCLS	CDP14
	CDP14A
OPNFIL	CDP14
OUTFIL	CDP14
CLSFIL	CDP14
CDDBTP	CDP14

DOCGROUP PS41261 Where-external-routine-used List

System Module -----	Module Name -----
CDMACR	CDP14 CDP14A

3.10.3 Main Program Parts List

The following lists each Main Program in the documentation group and all the modules which are called either by that module itself or by any of the documented modules which it calls. It is possible for a non-main module to be listed more than once if it is called by multiple modules. The called modules, in this case known as program parts, are marked as to whether they are documented here. If so, the phrase "well-defined module" appears by the module name, if not it is an "external routine". The Purpose of the Main Program module is listed as well.

DOCGROUP PS41261 Main Program Parts List

Main Pgm Name -----	Module Name -----	Module Type -----
CDCKNM	SQLSCA	External routine
	SQLBS1	External routine
	SQLSCH	External routine
	SQLSCC	External routine
	SQLTFL	External routine
	SQLOPN	External routine
	SQLOSQ	External routine
	SQLADR	External routine
	SQLAB1	External routine
	SQLEXE	External routine
	SQLAD1	External routine
	SQLFCH	External routine
	ERRPRO	External routine
CDEL RPM	SQLSCA	External routine
	SQLBS1	External routine
	SQLSCH	External routine
	SQLSCC	External routine
	SQLTFL	External routine
	SQLOPN	External routine
	SQLOSQ	External routine
	SQLADR	External routine
	SQLAB1	External routine
	SQLEXE	External routine
	ERRPRO	External routine
	SQLWNR	External routine
CDP14	SQLSCA	External routine
	SQLBS1	External routine
	SQLSCH	External routine

DOCGROUP PS41261 Main Program Parts List

Main Pgm Name -----	Module Name -----	Module Type -----
	SQLSCC	External routine
	SQLSQ	External routine
	SQLADR	External routine
	SQLAB1	External routine
	SQLXEX	External routine
	SQLAD1	External routine
	SQLFCH	External routine
	ERRPRO	External routine
	SQLLO1	External routine
	CDDBMSS	External routine
	CDLKLW	External routine
	CDCKNM	External routine
	SQLTOC	External routine
	SQLCLS	External routine
	OPNFIL	External routine
	OUTFIL	External routine
	CLSFIL	External routine
	CDDBTP	External routine
	CDMACR	External routine
	CDP14A	Well-defined module
	CDEL RPM	External routine
	CDRPSM	Well-defined module
CDP14A	SQLSCA	External routine
	SQLBS1	External routine
	SQLSCH	External routine
	SQLSCC	External routine
	SQLSQ	External routine
	SQLADR	External routine
	SQLAB1	External routine
	SQLXEX	External routine

DOCGROUP PS41261 Main Program Parts List

Main Pgm Name -----	Module Name -----	Module Type -----
	SQLAD1	External routine
	SQLFCH	External routine
	ERRPRO	External routine
	SQLTOC	External routine
	SQLCLS	External routine
	CDMACR	External routine
CDRPSM	SQLSCA	External routine
	SQLBS1	External routine
	SQLSCH	External routine
	SQLSCC	External routine
	SQLTFL	External routine
	SQLOPN	External routine
	SQLSQ	External routine
	SQLADR	External routine
	SQLAB1	External routine
	SQLXEX	External routine
	ERRPRO	External routine
	SQLWNR	External routine

3.10.4 Module Documentation

The following documentation describes information which is specific to each individual module in the documentation group being documented in this specification. It provides a compact way of getting information that would be otherwise buried within each module's source code.

The specific items in this module documentation have the following meanings:

NAME: Name of program Module.

PURPOSE: Purpose of Module as detailed in the source code.

LANGUAGE: Programming language source code is written in.
The choices are:
VAX-11 FORTRAN
C (I/S-1 Workbench 'C')
VAX-11 COBOL

MODULE TYPE: Whether a Program, Subroutine, or Function.

SOURCE FILE: Name of Source File from file specification.

SOURCE FILE TYPE: Source File Extension from file specification.

HOST: Whether this is a host-dependent routine (VAX or IBM) or blank if host-independent.

SUBSYSTEM: IISS sub-system this file resides in.

SUBDIRECTORY: Sub-directory of that subsystem in which this file resides.

DOCUMENTATION GROUP: Name of documentation group of which this source file is a member.

DESCRIPTION: A description of the module as obtained from the source code.

ARGUMENTS: The arguments with which this routine is called if it is a Subroutine or a Function.

INCLUDE FILES: A list of all the files that are included into this module as well as their purposes.

ROUTINES CALLED: Subroutines or Functions, either documented or external, called by this module, if any.

CALLED DIRECTLY BY: The documented routines which call this module, if any.

USED IN MAIN PROGRAM(S): The documented Main Programs which contain this module in their parts list according to the list in section 3.10.3.

The Module Documentation is arranged alphabetically according to Module Name.

DOCGROUP PS41261 Module Documentation

NAME: CDCKNM
PURPOSE: GET A COUNT OF UNSUCCESSFULLY PRECOMPILED NDML
MODULES
LANGUAGE: VAX-11 COBOL
SOURCE FILE: CDCKNM
SOURCE FILE TYPE: PCO
HOST:
SUBSYSTEM: CDM
SUBDIRECTORY: NDML

DESCRIPTION:

GIVEN A LOGICAL UNIT OF WORK NAME, DETERMINE HOW
MANY NDML MODULES HAVE BEEN RECORDED IN THE CDM
WHICH HAVE BEEN EITHER UNSUCCESSFULLY PRECOMPILED
OR NOT PRECOMPILED AT ALL YET. IF ANY ARE FOUND,
THE CALLER WILL NOT BE ABLE TO GENERATE ANY
REQUEST PROCESSOR DRIVERS.

ARGUMENTS:

LUW-NAME DSPLY[X(30)]
BAD-COUNT DSPLY[S9(9)]
RET-STATUS DSPLY[X(5)]

INCLUDE FILES:

CHKCDM
ERRCDM
EOD
ERRPRO

ROUTINES CALLED:

SQLSCA
SQLBS1
SQLSCH
SQLSCC
SQLTFL
SQLOPN
SQLOSQ
SQLADR
SQLAB1
SQLEXE
SQLAD1
SQLFCH
ERRPRO

DOCGROUP PS41261 Module Documentation

NAME: CDELRPM
PURPOSE: DELETE AN OBSOLETE RP-MAIN REFERENCE.
LANGUAGE: VAX-11 COBOL
SOURCE FILE: CDELRPM
SOURCE FILE TYPE: PCO
HOST:
SUBSYSTEM: CDM
SUBDIRECTORY: NDML

DESCRIPTION:

-

GIVEN THE NAME OF AN RP MAIN MODULE
DISCOVERED TO BE OBSOLETE, SINCE IT HAS
NO RP-SUB'S ASSOCIATED WITH IT, THIS
ROUTINE WILL DELETE THE REFERENCE TO IT FROM
THE CDM.

ARGUMENTS:

MOD-ID	DSPLY[X(30)]
RET-STATUS	DSPLY[X(5)]

INCLUDE FILES:

CHKCDM
ERRCDM
ERRPRO

ROUTINES CALLED:

SQLSCA
SQLBS1
SQLSCH
SQLSCC
SQLTFL
SQLOPN
SQLOSQ
SQLADR
SQLAB1
SQLEXE
SQLWNR
ERRPRO

DOCGROUP PS41261 Module Documentation

NAME: CDP14
PURPOSE: GENERATE REQUEST PROCESSOR DRIVERS FOR A LUW
LANGUAGE: VAX-11 COBOL
SOURCE FILE: CDP14
SOURCE FILE TYPE: PCO
HOST:
SUBSYSTEM: CDM
SUBDIRECTORY: NDML

DESCRIPTION:

UPDATED 1/31/89:
WILL ALWAYS CREATE FILES RPMIN(F/C) TO CDMTEMPS
(WILL BE STUBS IF THAT LANGUAGE NOT USED).
UPDATED 12/12/89:
RPMIN.C WILL GET CREATED AS A STUB IF LOCAL RP-MAIN
HAS BECOME OBSOLETE.

ARGUMENTS:

LUW-NAME DSPLY[X(30)]
CDM-USER-NAME DSPLY[X(30)]
CODE-GENERATOR-TABLE RECRD
MY-HOST DSPLY[XXX]
RET-STATUS DSPLY[X(5)]

INCLUDE FILES:

CHKCDM
ERRCDM
ERRFS
SBSTLST
EOD
CGTABLE
ERRPRO

ROUTINES CALLED:

SQLSCA
SQLADR
SQLLO1
CDDBMSS
CDLKLW
CDCKNM
SQLBS1
SQLSCH
SQLSCC
SQLTOC
SQLOSQ
SQLAB1
SQLEXE
SQLAD1
SQLFCH
SQLCLS

OPNFIL
OUTFIL
CLSFIL
ERRPRO
CDDBT
CDMACR
CDP14A
CDEL RPM
CDRPSM

DOCGROUP PS41261 Module Documentation

NAME: CDP14A
PURPOSE: GENERATE CASE CALL CODE INTO THE RP DRIVER
LANGUAGE: VAX-11 COBOL
SOURCE FILE: CDP14A
SOURCE FILE TYPE: PCO
HOST:
SUBSYSTEM: CDM
SUBDIRECTORY: NDML

DESCRIPTION:

SEARCH THE CDM FOR ALL RP-SUB'S FOR A
SINGLE REQUEST PROCESSOR DRIVER (
ONE DB_ID AND ONE LUW). FOR EACH
RP-SUB, GENERATE THE PROPER CODE INTO
THE RPD WITH A CALL TO CDMACR.
MODIFIED 10/30/87
REMOVAL OF DYNAMIC CALL CAPABILITY ON THE IBM.

ARGUMENTS:

DB-ID DSPLY[9(5)]
LUW-NAME DSPLY[X(30)]
FCB-O DSPLY[S9(9)]
LIBRARY-NAME DSPLY[X(30)]
HOST-ID RECD
LANG-NAME DSPLY[X(10)]
LOCAL-REMOTE DSPLY[X]
RET-STATUS DSPLY[X(5)]

INCLUDE FILES:

CHKCDM
ERRCDM
SBSTLST
EOD
ERRPRO

ROUTINES CALLED:

SQLSCA
SQLBS1

SQLSCH
SQLSCC
SQLTOC
SQLOSQ
SQLADR
SQLAB1
SQLEXE
SQLAD1
SQLFCH
SQLCLS
CDMACR
ERRPRO

DOCGROUP PS41261 Module Documentation

NAME: CDRPSM
PURPOSE: DROP THE SOFTWARE MODULE ROW FOR A GIVEN MODULE
LANGUAGE: VAX-11 COBOL
SOURCE FILE: CDRPSM
SOURCE FILE TYPE: PCO
HOST:
SUBSYSTEM: CDM
SUBDIRECTORY: SHARE

DESCRIPTION:

USE SQL TO DELETE ONE ROW FROM THE SOFTWARE MODULE
CDM TABLE.

ARGUMENTS:

```
MOD-ID          DSPLY[X(10)]
RET-STATUS      DSPLY[X(5)]
```

INCLUDE FILES:

CHKCDM
ERRCDM
EOD
ERRPRO

ROUTINES CALLED:

SQLSCA
SQLBS1
SQLSCH
SQLSCC
SQLTFL
SQLOPN
SQLOSQ
SQLADR
SQLAB1
SQLEXE
SQLWNR
ERRPRO

3.10.5 Include File Descriptions

The following list contains a purpose and description of each include file in the documentation group as specified in the source code. The language it is written in is also given.

DOCGROUP PS41261 Include File Description

FILE NAME: CGTABLE
PURPOSE: CODE GENERATING TABLE- TRACKS ALL GENERATED SOFTWARE
LANGUAGE: VAX-11 COBOL

DESCRIPTION:

HOLDS PERTINENT RESULTS
ABOUT ALL CODE GENERATED OR MODIFIED BY THE
PRECOMPILER

DOCGROUP PS41261 Include File Description

FILE NAME: CHKCDM
PURPOSE: IISS CDMP CHECK STATUS CODES
LANGUAGE: VAX-11 COBOL

DESCRIPTION:

CONTAINS ALL STATUS CODES FOR THE *
CDMP MODULES *

DOCGROUP PS41261 Include File Description

FILE NAME: EOD
PURPOSE: SQL END OF DATA DEFINITION
LANGUAGE: VAX-11 COBOL

DESCRIPTION:

DOCGROUP PS41261 Include File Description

FILE NAME: ERRCDM
PURPOSE: IISS ERROR STATUS CODES FOR CDMP MODULES
LANGUAGE: VAX-11 COBOL

DESCRIPTION:

CONTAINS ALL ERROR CODES USED BY CDMP *
MODULES FOR ERROR HANDLING *

DOCGROUP PS41261 Include File Description

FILE NAME: ERRFS
PURPOSE: ERRFS.INC - FILE I/O PRIMITIVES (FILE SERVICES)
LANGUAGE: VAX-11 COBOL

DESCRIPTION:

IISS ERROR CODES

THIS FILE DEFINES THE FS STATUS
CODES IN COBOL FORMAT

DOCGROUP PS41261 Include File Description

FILE NAME: ERRPRO
PURPOSE: PROCESS ERROR INCLUDE FILE
LANGUAGE: VAX-11 COBOL

DESCRIPTION:

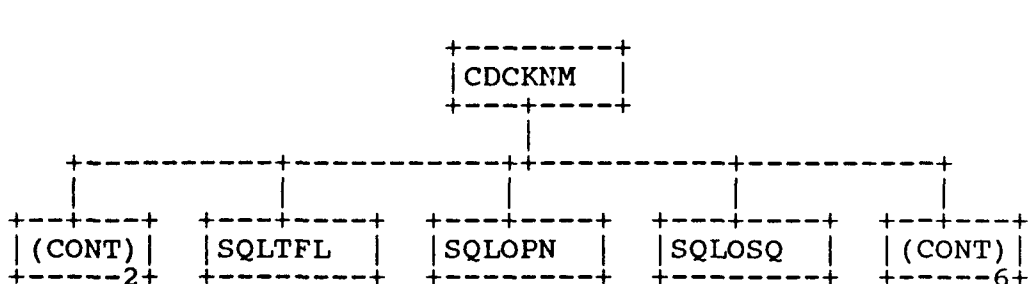
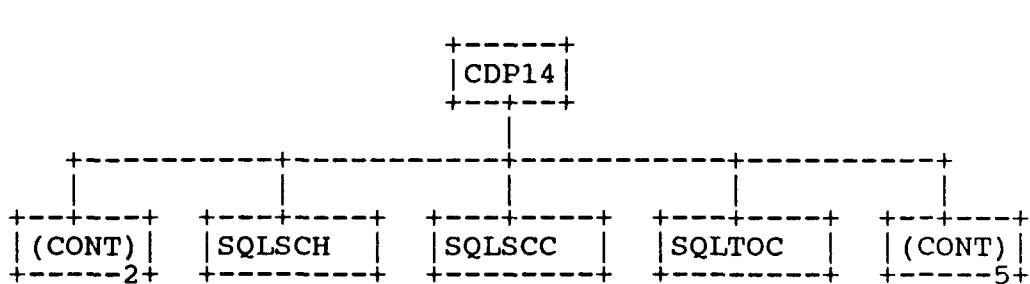
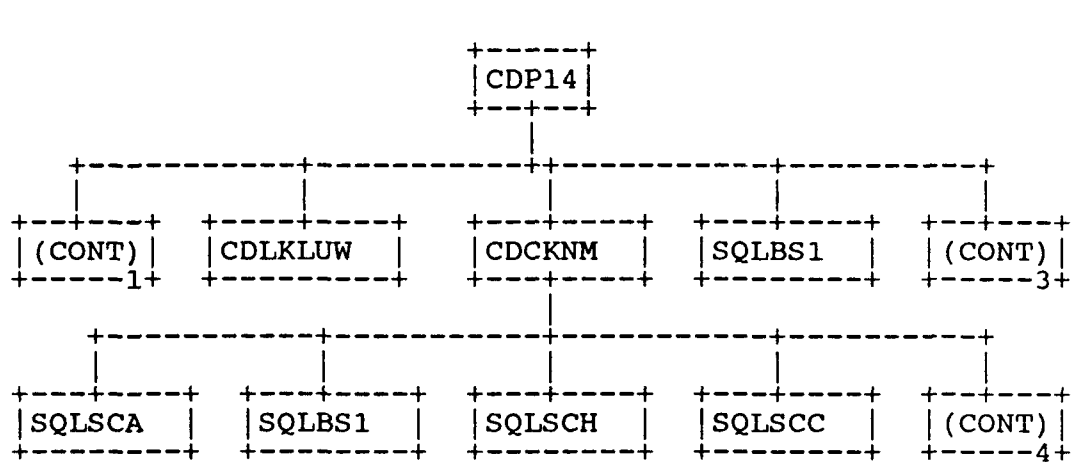
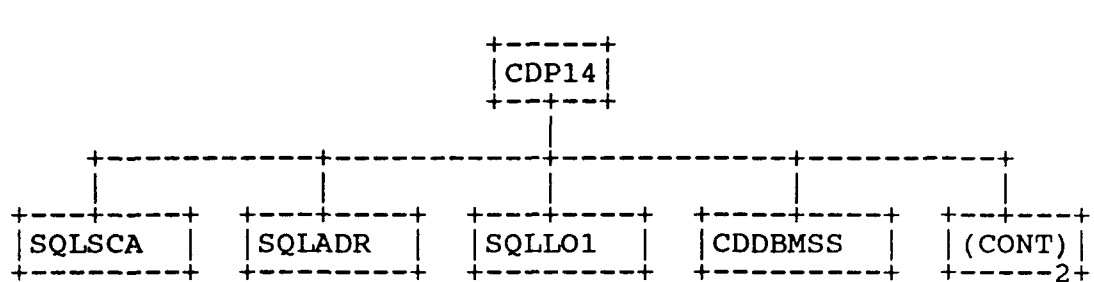
DOCGROUP PS41261 Include File Description

FILE NAME: SBSTLST
PURPOSE: WS DEFINITION FOR THE SUBSTITUTION LIST TABLE
LANGUAGE: VAX-11 COBOL

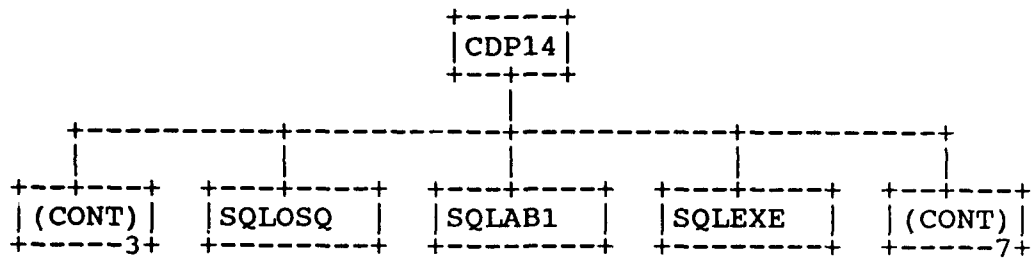
DESCRIPTION:

SUBSTITUTION-LIST REPRESENTS THE INPUT TABLE
OF SUBSTITUTION PARAMETERS FOR THE CDMACR
MACRO EXPANSION SUBROUTINE

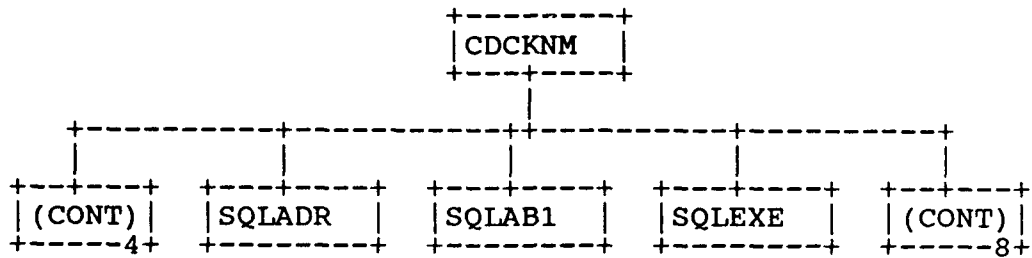
3.10.6 Hierarchy Chart



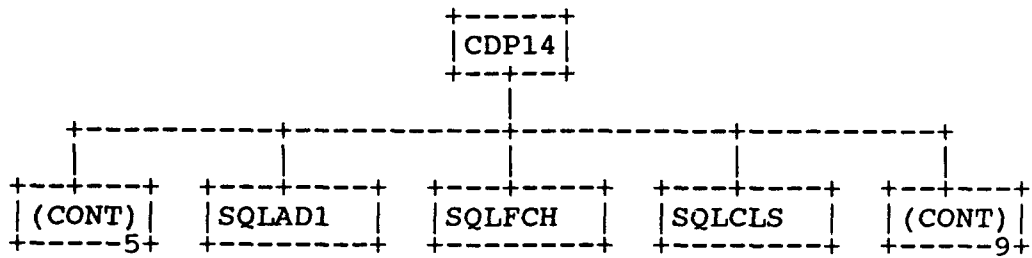
5



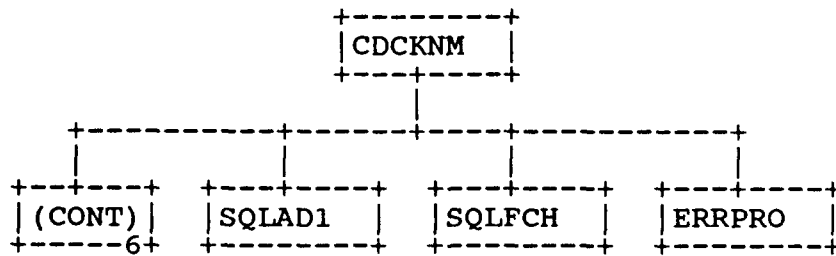
6



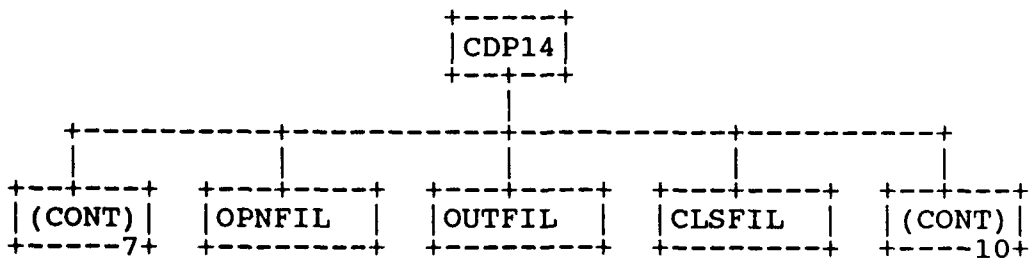
7



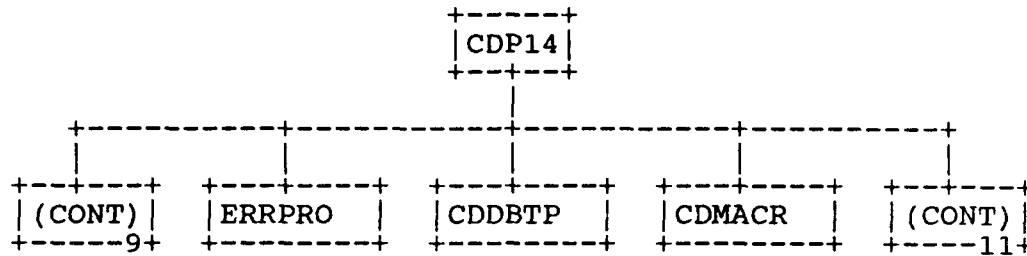
8



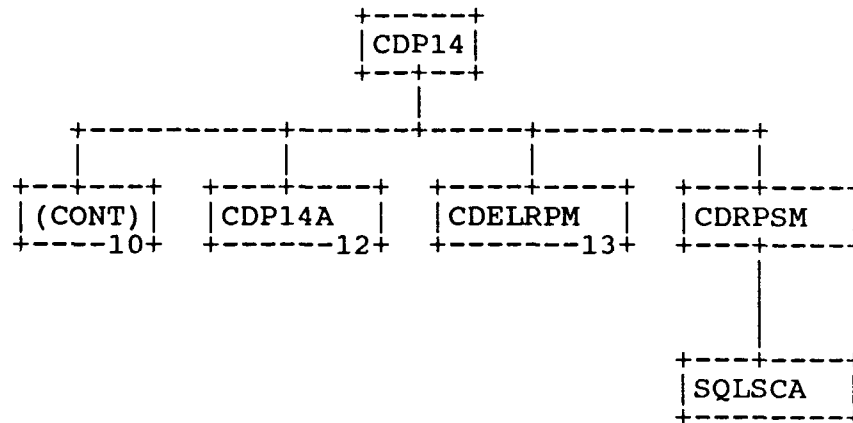
9



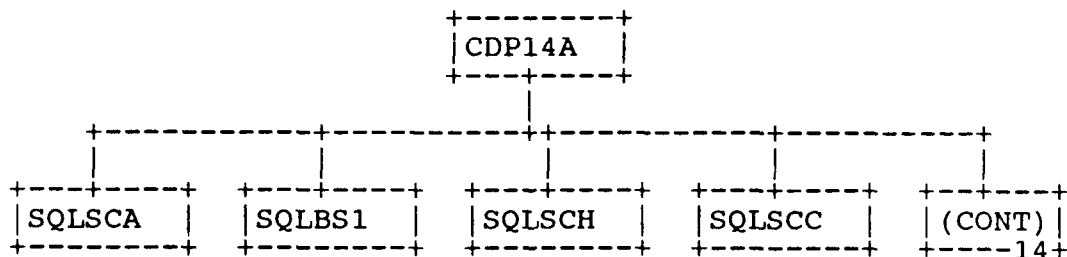
10



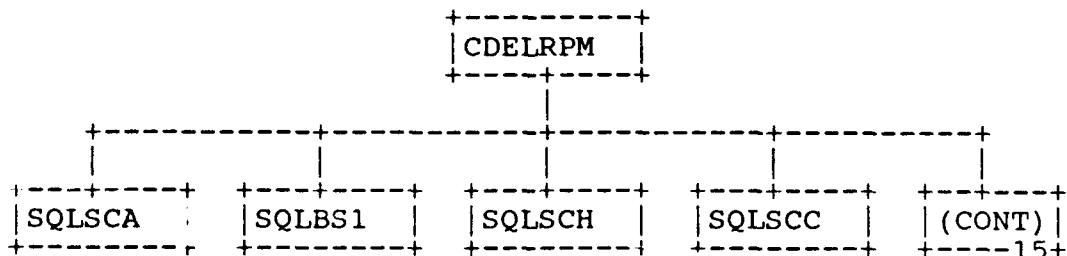
11



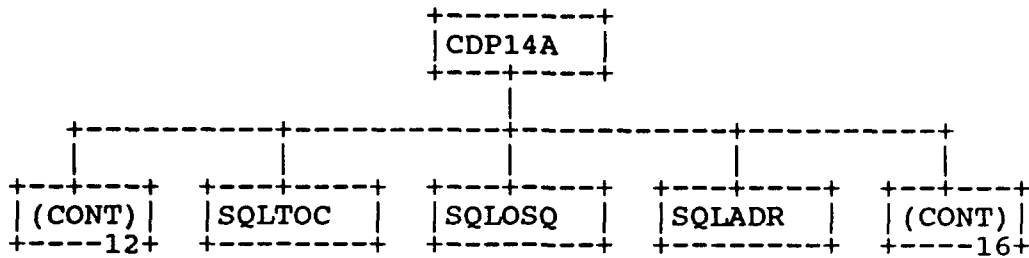
12



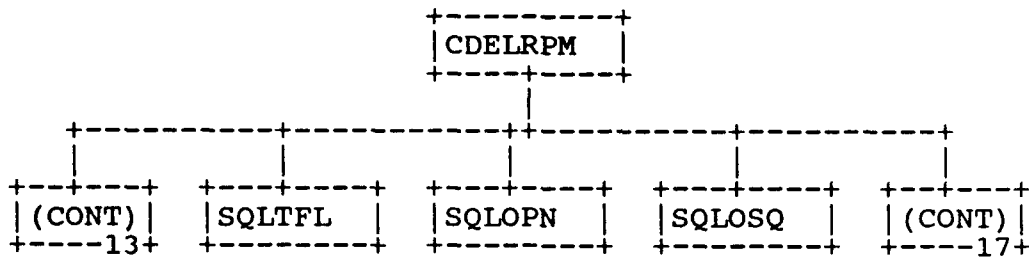
13



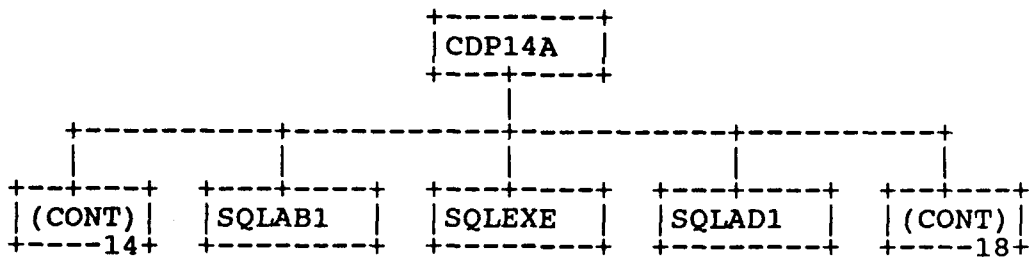
14



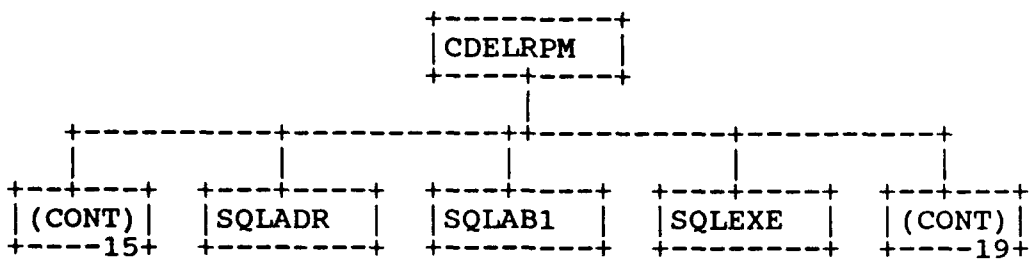
15



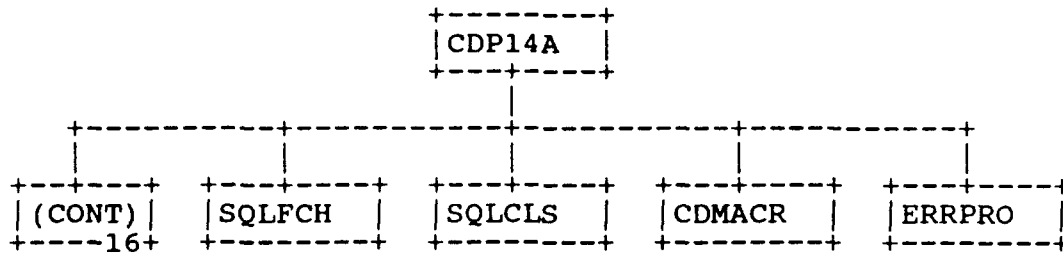
16



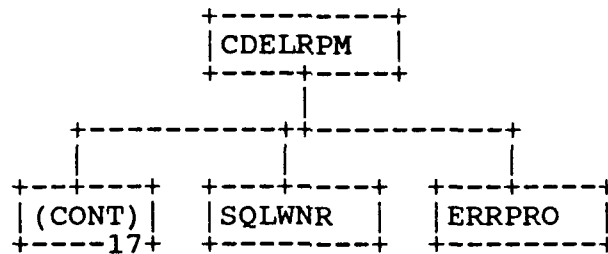
17



18



19



CDCKNM2
 CDDBMSS
 CDDBTP
 CDEL RPM ...13
 CDLKLUW
 CDMACR
 CDP141
 CDP14A12
 CDRPSM11
 CLSFIL
 ERRPRO
 OPNFIL
 OUTFIL
 SQLAB1
 SQLAD1
 SQLADR
 SQLBS1
 SQLCLS
 SQLEXE
 SQLFCH
 SQLLO1
 SQLOPN
 SQLOSQ
 SQLSCA
 SQLSCC
 SQLSCH
 SQLTFL
 SQLTOC
 SQLWNR

3.11 Program Listings Comments

This information is contained in the Module Descriptions in section 3.10.

SECTION 4

QUALITY ASSURANCE PROVISIONS

4.1 Introduction and Definitions

"Testing" is a systematic process that may be preplanned and explicitly stated. Test techniques and procedures may be defined in advance, and a sequence of test steps may be specified. "Debugging" is the process of isolation and correction of the cause of an error.

"Antibugging" is defined as the philosophy of writing programs in such a way as to make bugs less likely to occur and when they do occur, to make them more noticeable to the programmer and the user. In other words, as much error checking as is practical and possible in each routine should be performed.

4.2 Computer Programming Test and Evaluation

The quality assurance provisions for test consists of the normal testing techniques that are accomplished during the construction process. They consist of design and code walk-throughs, unit testing, and integration testing. These tests are performed by the design team. Structured design, design walk-through and the incorporation of "antibugging" facilitate this testing by exposing and addressing problem areas before they become coded "bugs."