INTEGRATED INFORMATION
SUPPORT SYSTEM (IISS)
Volume V - Common Data Model Subsystem
Part 15 - NDML Precompiler Decompose Conceptual Schema
NDML Request Product Specification

General Electric Company
Production Resources Consulting
One River Road
Schenectady, New York 12345


November 1985

Approved for public release; distribution is unlimited.

MATERIALS LABORATORY
AIR FORCE WRIGHT AERONAUTICAL LABORATORIES
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AFB, OH 45433-6533
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; and the fact that the government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

This report has been reviewed by the Office of Public Affairs (ASD/PA) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.

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

David Judson
PROJECT MANAGER
AFWAL/MLTC
WRIGHT PATTERSON AFB OH 45433

FOR THE COMMANDER

Gerald C. Shumaker
BRANCH CHIEF
AFWAL/MLTC
WRIGHT PATTERSON AFB OH 45433

"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 AFWAL/MLTC, W-PAFB, OH 45433 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.
This document is the product specification establishing the design implementation of the IISS Configuration Item PRE5 which will transform a neutral data request from the conceptual to many internal schemas.

(2 page vii)
11. Title

Integrated Information Support System (IISS)
Vol V - Common Data Model Subsystem
Part 15 - NDML Precompiler Decompose Conceptual Schema
NDML Request Product Specification

A S D 86 1467
17 Jul 1986
PREFACE

This product specification covers the work performed under Air Force Contract F33615-80-C-5155 (ICAM Project 6201). This contract is sponsored by the Materials Laboratory, Air Force Systems Command, Wright-Patterson Air Force Base, Ohio. It was administered under the technical direction of Mr. Gerald C. Shumaker, ICAM Program Manager, Manufacturing Technology Division, through Project Manager, Mr. David Judson. The Prime Contractor was Production Resources Consulting of the General Electric Company, Schenectady, New York, under the direction of Mr. Alan Rubenstein. The General Electric Project Manager was Mr. Myron Hurlbut of Industrial Automation Systems Department, Albany, New York.

Certain work aimed at improving Test Bed Technology has been performed by other contracts with Project 6201 performing integrating functions. This work consisted of enhancements to Test Bed software and establishment and operation of Test Bed hardware and communications for developers and other users. Documentation relating to the Test Bed from all of these contractors and projects have been integrated under Project 6201 for publication and treatment as an integrated set of documents. The particular contributors to each document are noted on the Report Documentation Page (DD1473). A listing and description of the entire project documentation system and how they are related is contained in document FTR620100001, Project Overview.

The subcontractors and their contributing activities were as follows:

**TASK 4.2**

<table>
<thead>
<tr>
<th>Subcontractors</th>
<th>Role</th>
</tr>
</thead>
<tbody>
<tr>
<td>Boeing Military Aircraft Company (BMAC)</td>
<td>Reviewer</td>
</tr>
<tr>
<td>D. Appleton Company (DACOM)</td>
<td>Responsible for IDEF support, state-of-the-art literature search</td>
</tr>
<tr>
<td>General Dynamics/ Ft. Worth</td>
<td>Responsible for factory view function and information models</td>
</tr>
</tbody>
</table>
## Subcontractors

<table>
<thead>
<tr>
<th>Subcontractor</th>
<th>Role</th>
</tr>
</thead>
<tbody>
<tr>
<td>Illinois Institute of Technology</td>
<td>Responsible for factory view function research (IITRI) and information models of small and medium-size business</td>
</tr>
<tr>
<td>North American Rockwell</td>
<td>Reviewer</td>
</tr>
<tr>
<td>Northrop Corporation</td>
<td>Responsible for factory view function and information models</td>
</tr>
<tr>
<td>Pritsker and Associates</td>
<td>Responsible for IDEF2 support</td>
</tr>
<tr>
<td>SofTech</td>
<td>Responsible for IDEF0 support</td>
</tr>
</tbody>
</table>

### TASKS 4.3 - 4.9 (TEST BED)

<table>
<thead>
<tr>
<th>Subcontractor</th>
<th>Role</th>
</tr>
</thead>
<tbody>
<tr>
<td>Boeing Military Aircraft Company (BMAC)</td>
<td>Responsible for consultation on applications of the technology and on IBM computer technology.</td>
</tr>
<tr>
<td>Computer Technology Associates (CTA)</td>
<td>Assisted in the areas of communications systems, system design and integration methodology, and design of the Network Transaction Manager.</td>
</tr>
<tr>
<td>Control Data Corporation (CDC)</td>
<td>Responsible for the Common Data Model (CDM) implementation and part of the CDM design (shared with DACOM).</td>
</tr>
<tr>
<td>D. Appleton Company (DACOM)</td>
<td>Responsible for the overall CDM Subsystem design integration and test plan, as well as part of the design of the CDM (shared with CDC). DACOM also developed the Integration Methodology and did the schema mappings for the Application Subsystems.</td>
</tr>
</tbody>
</table>
Subcontractors

Digital Equipment Corporation (DEC)  Consulting and support of the performance testing and on DEC software and computer systems operation.

McDonnell Douglas Automation Company (McAuto)  Responsible for the support and enhancements to the Network Transaction Manager Subsystem during 1984/1985 period.

On-Line Software International (OSI)  Responsible for programming the Communications Subsystem on the IBM and for consulting on the IBM.

Rath and Strong Systems Products (RSSP) (In 1985 became McCormack & Dodge)  Responsible for assistance in the implementation and use of the MRP II package (PIOS) that they supplied.

SofTech, Inc.  Responsible for the design and implementation of the Network Transaction Manager (NTM) in 1981/1984 period.

Software Performance Engineering (SPE)  Responsible for directing the work on performance evaluation and analysis.

Structural Dynamics Research Corporation (SDRC)  Responsible for the User Interface and Virtual Terminal Interface Subsystems.

Other prime contractors under other projects who have contributed to Test Bed Technology, their contributing activities and responsible projects are as follows:

<table>
<thead>
<tr>
<th>Contractors</th>
<th>ICAM Project</th>
<th>Contributing Activities</th>
</tr>
</thead>
<tbody>
<tr>
<td>Boeing Military Aircraft Company (BMAC)</td>
<td>1701, 2201, 2202</td>
<td>Enhancements for IBM node use. Technology Transfer to Integrated Sheet Metal Center (ISMH)</td>
</tr>
<tr>
<td>Contractors</td>
<td>ICAM Project</td>
<td>Contributing Activities</td>
</tr>
<tr>
<td>--------------------------------------</td>
<td>--------------</td>
<td>-------------------------------------------------------------</td>
</tr>
<tr>
<td>Control Data Corporation (CDC)</td>
<td>1502, 1701</td>
<td>IISS enhancements to</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Common Data Model Processor (CDMP)</td>
</tr>
<tr>
<td>D. Appleton Company (DACON)</td>
<td>1502</td>
<td>IISS enhancements to</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Integration Methodology</td>
</tr>
<tr>
<td>General Electric</td>
<td>1502</td>
<td>Operation of the Test Bed and communications equipment.</td>
</tr>
<tr>
<td>Hughes Aircraft Company (HAC)</td>
<td>1701</td>
<td>Test Bed enhancements</td>
</tr>
<tr>
<td>Structural Dynamics Research Corporation (SDRC)</td>
<td>1502, 1701, 1703</td>
<td>IISS enhancements to</td>
</tr>
<tr>
<td></td>
<td></td>
<td>User Interface/Virtual Terminal Interface (UI/VTI)</td>
</tr>
<tr>
<td>Systran</td>
<td>1502</td>
<td>Test Bed enhancements.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Operation of Test Bed.</td>
</tr>
</tbody>
</table>
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0 Scope</td>
<td>SCOPE</td>
<td>1-1</td>
</tr>
<tr>
<td>1.1 Identification</td>
<td></td>
<td>1-1</td>
</tr>
<tr>
<td>1.2 Functional Summary</td>
<td></td>
<td>1-1</td>
</tr>
<tr>
<td>2.0 Documents</td>
<td>DOCUMENTS</td>
<td>2-1</td>
</tr>
<tr>
<td>2.1 Reference Documents</td>
<td></td>
<td>2-1</td>
</tr>
<tr>
<td>2.2 Terms and Abbreviations</td>
<td></td>
<td>2-1</td>
</tr>
<tr>
<td>3.0 Requirements</td>
<td>REQUIREMENTS</td>
<td>3-1</td>
</tr>
<tr>
<td>3.1 Structural Description</td>
<td></td>
<td>3-1</td>
</tr>
<tr>
<td>3.2 Functional Flow</td>
<td></td>
<td>3-1</td>
</tr>
<tr>
<td>3.3 Interfaces</td>
<td></td>
<td>3-2</td>
</tr>
<tr>
<td>3.3.1 Inputs/Outputs</td>
<td></td>
<td>3-3</td>
</tr>
<tr>
<td>3.4 Program Interrupts</td>
<td></td>
<td>3-2</td>
</tr>
<tr>
<td>3.5 Timing and Sequencing Description</td>
<td></td>
<td>3-3</td>
</tr>
<tr>
<td>3.6 Special Control Features</td>
<td></td>
<td>3-3</td>
</tr>
<tr>
<td>3.7 Storage Allocation</td>
<td></td>
<td>3-3</td>
</tr>
<tr>
<td>3.7.1 Database Definition</td>
<td></td>
<td>3-3</td>
</tr>
<tr>
<td>3.7.1.1 File Description</td>
<td></td>
<td>3-3</td>
</tr>
<tr>
<td>3.7.1.2 Table Description</td>
<td></td>
<td>3-3</td>
</tr>
<tr>
<td>3.7.1.3 Item Description</td>
<td></td>
<td>3-3</td>
</tr>
<tr>
<td>3.8 Object Code Creation</td>
<td></td>
<td>3-3</td>
</tr>
<tr>
<td>3.9 Adaptation Data</td>
<td></td>
<td>3-4</td>
</tr>
<tr>
<td>3.10 Detail Design Description</td>
<td></td>
<td>3-4</td>
</tr>
<tr>
<td>3.10.1 Main Program List</td>
<td></td>
<td>3-5</td>
</tr>
<tr>
<td>3.10.2 Module List</td>
<td></td>
<td>3-7</td>
</tr>
<tr>
<td>3.10.3 External Routines List</td>
<td></td>
<td>3-9</td>
</tr>
<tr>
<td>3.10.4 Include File List</td>
<td></td>
<td>3-11</td>
</tr>
<tr>
<td>3.10.5 Where Include File Used List</td>
<td></td>
<td>3-13</td>
</tr>
<tr>
<td>3.10.6 Where External Routine Used List</td>
<td></td>
<td>3-19</td>
</tr>
<tr>
<td>3.10.7 Main Program Parts List</td>
<td></td>
<td>3-22</td>
</tr>
<tr>
<td>3.10.8 Module Documentation</td>
<td></td>
<td>3-24</td>
</tr>
<tr>
<td>3.10.9 Include File Descriptions</td>
<td></td>
<td>3-29</td>
</tr>
<tr>
<td>3.10.10 Hierarchy Chart</td>
<td></td>
<td>3-57</td>
</tr>
<tr>
<td>3.11 Program Listings Comments</td>
<td></td>
<td>3-62</td>
</tr>
</tbody>
</table>
TABLE OF CONTENTS (Continued)

<table>
<thead>
<tr>
<th>SECTION 4.0</th>
<th>QUALITY ASSURANCE PROVISIONS</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1</td>
<td>Introduction and Definitions</td>
<td>4-1</td>
</tr>
<tr>
<td>4.2</td>
<td>Computer Programming Test</td>
<td>4-1</td>
</tr>
<tr>
<td></td>
<td>and Evaluation</td>
<td>4-1</td>
</tr>
</tbody>
</table>
SECTION 1

SCOPE

1.1 Identification

This specification establishes the design of Function PRES, "Decompose CS NDML", 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 break down a CS NDML transaction, or request, into its various CS NDML subtransactions.

The following functions will be performed by this CPCI:

1. Map all CS attribute use classes, entity classes and relation classes.

2. Identify all unions, intradatabase joins, inter database joins and unions and Not In Set operators.

3. Reformat the original request into single database subtransactions.

4. Control further code generation of request processors.
SECTION 2

DOCUMENTS

2.1 Reference Documents


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 sections 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.

All functionality of PRE5 is contained in a single COBOL module called CDPRES. A separate module CDDBTP is used to look up certain information about a database.

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.

This 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 DBMS installed on a DEC VAX computer.
3.3 Interfaces

The following diagram depicts the interface of PRE5 with other CPCI's in the system.

+--------------+  
|   PRE4       |  
|   PS41321    |  
+--------------+
     |             |
     +--------------+
     |   PRE5       |
     +--------------+
     |             |
     +--------------+
     |   CDP13      |
     |   PS413210   |
     +--------------+

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: PRE5**

<table>
<thead>
<tr>
<th>INPUT</th>
<th>OUTPUT</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS Action List</td>
<td>Module Status</td>
</tr>
<tr>
<td>CS Qualify List</td>
<td></td>
</tr>
<tr>
<td>ES Action List</td>
<td></td>
</tr>
<tr>
<td>ES Qualify List</td>
<td></td>
</tr>
<tr>
<td>User View Abbreviation List</td>
<td></td>
</tr>
<tr>
<td>Code Generator Table</td>
<td></td>
</tr>
<tr>
<td>I/O Status indicator</td>
<td></td>
</tr>
<tr>
<td>Parcel 1 File Name</td>
<td></td>
</tr>
<tr>
<td>Parcel 2 File Name</td>
<td></td>
</tr>
<tr>
<td>Parcel 3 File Name</td>
<td></td>
</tr>
<tr>
<td>Parcel 4 File Name</td>
<td></td>
</tr>
<tr>
<td>Host of the precompiler</td>
<td></td>
</tr>
<tr>
<td>Host the AP is to run on</td>
<td></td>
</tr>
<tr>
<td>Source Language</td>
<td></td>
</tr>
<tr>
<td>CE-Work-List</td>
<td></td>
</tr>
<tr>
<td>Error File Name</td>
<td></td>
</tr>
<tr>
<td>Oracle Logon Data Area</td>
<td></td>
</tr>
</tbody>
</table>

3-2
3.4 Program Interrupts

Not applicable to this CPCI.

3.5 Timing and Sequencing Description

This module, CDPRE6, is called from CPCI PRE4 for each conceptual schema NDNL request. A conceptual schema request may be the user's original request or a referential integrity test which is handled as a search request. Control logic CDPRE6 in turn translates each request into many subtransactions. When complete, CDP13 is called by the CDPRE6. The CDP13 is the module which controls code generation for each subtransaction.

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 CDMI, the IDEF-1 model of the CDM, Reference Document Number 3.

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 language compiler.

3.9 Adaptation Data

This CPCI has been coded using ANSI COBOL language. The intent was to provide a transportable system. Any system environment supporting this language, a virtual memory management scheme, the COMM and MTM subsystem 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 Main Program List

The following is a list of all "Main Programs" which are modules that are not called by any other module being documented here. These modules are either program entry points or, if they are hooked into another set of programs via subroutine calls, they are the points the external programs can call and therefore enter through. To differentiate between the two types of entry points, look at the individual Module Documentation (section 3.10.8) and look at Module Type for each of the Main Program modules listed. Note whether the routine is a Program, Subroutine, or Function. If it is a Program, it is truly a main program entry point. If not, then it is merely called by other programs not being documented here.
DECOMPOSE CS REQUEST Main Program List

<table>
<thead>
<tr>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>CDPRE6</td>
<td>DECOMPOSE CONCEPTUAL SCHEMA NDML REQUEST</td>
</tr>
</tbody>
</table>
3.10.2 Module List

The following is a list of all the modules being documented here along with their purpose. Each module has a unique name, no matter what language it was written in.
DECOMPOSE CS REQUEST Module List

<table>
<thead>
<tr>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>CDPRES</td>
<td>DECOMPOSE CONCEPTUAL SCHEMA NDML REQUEST</td>
</tr>
</tbody>
</table>
3.10.3 External Routines List

The following is a list of all routines or functions not documented here that are called by modules that are documented here. The first caller, in alphabetical order, is listed as well. The specification in which any module is documented may be found in the Module Documentation Index (Document Number CM 620100001). See section 3.10.6 for a list of the modules that call each of these external routines.
DECOMPOSE CS REQUEST External Routines List

<table>
<thead>
<tr>
<th>Module Name</th>
<th>First User</th>
</tr>
</thead>
<tbody>
<tr>
<td>CDBBTP</td>
<td>CDPRE6</td>
</tr>
<tr>
<td>CDP13</td>
<td>CDPRE6</td>
</tr>
<tr>
<td>ERRPRO</td>
<td>CDPRE6</td>
</tr>
<tr>
<td>OBINDN</td>
<td>CDPRE6</td>
</tr>
<tr>
<td>OCLOSE</td>
<td>CDPRE6</td>
</tr>
<tr>
<td>ODFINN</td>
<td>CDPRE6</td>
</tr>
<tr>
<td>OEXEC</td>
<td>CDPRE6</td>
</tr>
<tr>
<td>OFETCH</td>
<td>CDPRE6</td>
</tr>
<tr>
<td>OOPEN</td>
<td>CDPRE6</td>
</tr>
<tr>
<td>OSQL3</td>
<td>CDPRE6</td>
</tr>
<tr>
<td>RPTERR</td>
<td>CDPRE6</td>
</tr>
</tbody>
</table>
3.10.4 Include File List

The following is a list of all include files called in by modules being documented here. Each include file has a unique name regardless of the language being used. The purpose of each include file is listed as well. A more complete description of each include file is given in section 3.10.9. The purpose listed is the one that is in the source code of the include file.

A purpose of "***** PURPOSE NOT FOUND BY STRIPPER *****" indicates that a purpose statement was not written into the include file itself. The most common reason for this is that the include file comes from system libraries that were not developed by the project, such as 'C' libraries that are provided with the 'C' compiler.

See section 3.10.6 for a set of lists which show all the modules which call in each of these include files.
DECOMPOSE CS REQUEST Include File List

<table>
<thead>
<tr>
<th>File Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>APL</td>
<td>JOIN QUERY ATTRIBUTE PAIR LIST</td>
</tr>
<tr>
<td>AUCDAT</td>
<td>ORACLE CURSOR DEFINITION</td>
</tr>
<tr>
<td>CEWORK</td>
<td>CS TO ES WORK LIST INFORMATION</td>
</tr>
<tr>
<td>CTABLE</td>
<td>CODE GENERATING TABLE- TRACKS ALL GENERATED SOFTWARE</td>
</tr>
<tr>
<td>CSAL</td>
<td>CONCEPTUAL SCHEMA ACTION LIST</td>
</tr>
<tr>
<td>CSQUAL</td>
<td>CONCEPTUAL SCHEMA QUALIFY LIST</td>
</tr>
<tr>
<td>DMPCSRL</td>
<td>DISPLAYS THE CONTENTS OF THE CS ACTION LIST</td>
</tr>
<tr>
<td>DMPREPL</td>
<td>DISPLAYS THE CONTENTS OF THE REPL JOIN TABLE</td>
</tr>
<tr>
<td>DMPSETT</td>
<td>DISPLAYS THE CONTENTS OF THE SET TABLE</td>
</tr>
<tr>
<td>DUMPAPL</td>
<td>DISPLAYS THE CONTENTS OF THE ATTRIBUTE PAIR LIST</td>
</tr>
<tr>
<td>DUMPCSQ</td>
<td>DISPLAY THE CONTENTS OF THE CS QUALIFY LIST</td>
</tr>
<tr>
<td>DUMPGRP</td>
<td>DISPLAY THE CONTENTS OF THE GROUP TABLE</td>
</tr>
<tr>
<td>DUMPJQG</td>
<td>DISPLAY THE CONTENTS OF THE JQG TABLE</td>
</tr>
<tr>
<td>ERRCDM</td>
<td>IISS ERROR STATUS CODES FOR CDMP MODULES</td>
</tr>
<tr>
<td>ERRPRO</td>
<td>PROCESS ERROR INCLUDE FILE</td>
</tr>
<tr>
<td>ESAL</td>
<td>EXTERNAL SCHEMA ACTION LIST</td>
</tr>
<tr>
<td>ESQUAL</td>
<td>EXTERNAL SCHEMA QUALIFY LIST</td>
</tr>
<tr>
<td>ISAL</td>
<td>INTERNAL SCHEMA ACTION LIST</td>
</tr>
<tr>
<td>ISQUAL</td>
<td>INTERNAL SCHEMA QUALIFY LIST</td>
</tr>
<tr>
<td>JQGTBL</td>
<td>JOIN QUERY GRAPH TELLS HOW TO CONNECT SUBTRANSACTIONS</td>
</tr>
<tr>
<td>LNKEDGE</td>
<td>**** PURPOSE NOT FOUND BY STRIPPER ****</td>
</tr>
<tr>
<td>ORCLEDA</td>
<td>WS DEFINITION FOR THE ORACLE LOGIN AREA</td>
</tr>
<tr>
<td>REPLJN</td>
<td>REPL-JOIN-LIST SUBTRANSACTION JOINS</td>
</tr>
<tr>
<td>RFTABLE</td>
<td>THE RESULT FIELD TABLE</td>
</tr>
<tr>
<td>SETTAB</td>
<td>LIST OF SETS OWNER-MEMBER RELATIONSHIPS</td>
</tr>
<tr>
<td>SUBPROC</td>
<td>SUBTRANSACTION PROCESSES ID TABLE</td>
</tr>
<tr>
<td>UVABBQ</td>
<td>USER VIEW ABBREVIATION LIST</td>
</tr>
</tbody>
</table>

3-12
3.10.5 Where Include File Used List

The following lists each include file from 3.10.4 and all the modules documented in this specification which include them. The purpose of each module is listed as well.
DECOMPOSE CS REQUEST Where-include-file-used List

<table>
<thead>
<tr>
<th>Include File</th>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>APL</td>
<td>CDPRE6</td>
<td>DECOMPOSE CONCEPTUAL SCHEMA NDML REQUEST</td>
</tr>
<tr>
<td>AUCDAT</td>
<td>CDPRE6</td>
<td>DECOMPOSE CONCEPTUAL SCHEMA NDML REQUEST</td>
</tr>
<tr>
<td>CEWORK</td>
<td>CDPRE6</td>
<td>DECOMPOSE CONCEPTUAL SCHEMA NDML REQUEST</td>
</tr>
<tr>
<td>CGTABLE</td>
<td>CDPRE6</td>
<td>DECOMPOSE CONCEPTUAL SCHEMA NDML REQUEST</td>
</tr>
<tr>
<td>CSAL</td>
<td>CDPRE6</td>
<td>DECOMPOSE CONCEPTUAL SCHEMA NDML REQUEST</td>
</tr>
<tr>
<td>CSQUAL</td>
<td>CDPRE6</td>
<td>DECOMPOSE CONCEPTUAL SCHEMA NDML REQUEST</td>
</tr>
<tr>
<td>DMPGSRL</td>
<td>CDPRE6</td>
<td>DECOMPOSE CONCEPTUAL SCHEMA NDML REQUEST</td>
</tr>
</tbody>
</table>

3-14
DECOMPOSE CS REQUEST Where-include-file-used List

<table>
<thead>
<tr>
<th>Include File</th>
<th>Module Name</th>
<th>Module Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>DMPREPL</td>
<td>CDPRE6</td>
<td>DECOMPOSE CONCEPTUAL SCHEMA NDML REQUEST</td>
</tr>
<tr>
<td>DMPSETT</td>
<td>CDPRE6</td>
<td>DECOMPOSE CONCEPTUAL SCHEMA NDML REQUEST</td>
</tr>
<tr>
<td>DUMPAPL</td>
<td>CDPRE6</td>
<td>DECOMPOSE CONCEPTUAL SCHEMA NDML REQUEST</td>
</tr>
<tr>
<td>DUMPCSQ</td>
<td>CDPRE6</td>
<td>DECOMPOSE CONCEPTUAL SCHEMA NDML REQUEST</td>
</tr>
<tr>
<td>DUMPGRP</td>
<td>CDPRE6</td>
<td>DECOMPOSE CONCEPTUAL SCHEMA NDML REQUEST</td>
</tr>
<tr>
<td>DUMPJQG</td>
<td>CDPRE6</td>
<td>DECOMPOSE CONCEPTUAL SCHEMA NDML REQUEST</td>
</tr>
</tbody>
</table>
**DECOMPOSE CS REQUEST Where-include-file-used List**

<table>
<thead>
<tr>
<th>Include File</th>
<th>Module Name</th>
<th>Module Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>ERRCDM</td>
<td>CDPRE6</td>
<td>DECOMPOSE CONCEPTUAL SCHEMA NDML REQUEST</td>
</tr>
<tr>
<td>ERRPRO</td>
<td>CDPRE6</td>
<td>DECOMPOSE CONCEPTUAL SCHEMA NDML REQUEST</td>
</tr>
<tr>
<td>ESAL</td>
<td>CDPRE6</td>
<td>DECOMPOSE CONCEPTUAL SCHEMA NDML REQUEST</td>
</tr>
<tr>
<td>ESQUAL</td>
<td>CDPRE6</td>
<td>DECOMPOSE CONCEPTUAL SCHEMA NDML REQUEST</td>
</tr>
<tr>
<td>ISAL</td>
<td>CDPRE6</td>
<td>DECOMPOSE CONCEPTUAL SCHEMA NDML REQUEST</td>
</tr>
<tr>
<td>ISQUAL</td>
<td>CDPRE6</td>
<td>DECOMPOSE CONCEPTUAL SCHEMA NDML REQUEST</td>
</tr>
<tr>
<td>JQGTBL</td>
<td>CDPRE6</td>
<td>DECOMPOSE CONCEPTUAL SCHEMA NDML REQUEST</td>
</tr>
</tbody>
</table>

3-16
DECOMPOSE CS REQUEST Where-include-file-used List

<table>
<thead>
<tr>
<th>Include File</th>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>LNKEDGE</td>
<td>CDPRE6</td>
<td>DECOMPOSE CONCEPTUAL SCHEMA NDML REQUEST</td>
</tr>
<tr>
<td>ORCLEDA</td>
<td>CDPRE6</td>
<td>DECOMPOSE CONCEPTUAL SCHEMA NDML REQUEST</td>
</tr>
<tr>
<td>REPLJN</td>
<td>CDPRE6</td>
<td>DECOMPOSE CONCEPTUAL SCHEMA NDML REQUEST</td>
</tr>
<tr>
<td>RTABLE</td>
<td>CDPRE6</td>
<td>DECOMPOSE CONCEPTUAL SCHEMA NDML REQUEST</td>
</tr>
<tr>
<td>SETTAB</td>
<td>CDPRE6</td>
<td>DECOMPOSE CONCEPTUAL SCHEMA NDML REQUEST</td>
</tr>
<tr>
<td>SUBPROC</td>
<td>CDPRE6</td>
<td>DECOMPOSE CONCEPTUAL SCHEMA NDML REQUEST</td>
</tr>
</tbody>
</table>
DECOMPOSE CS REQUEST Where-inclde-file-used List

<table>
<thead>
<tr>
<th>Include File</th>
<th>Module Name</th>
<th>Module Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>UVABBR</td>
<td>CDPRES6</td>
<td>DECOMPOSE CONCEPTUAL SCHEMA NDML REQUEST</td>
</tr>
</tbody>
</table>
3.10.6 Where External Routine Used List

The following lists each external function or routine listed in 3.10.3 and all the documented modules which call it. The purpose of each module is listed as well.
DECOMPOSE CS REQUEST Where-external-routine-used List

<table>
<thead>
<tr>
<th>System Module</th>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>CDDBTP</td>
<td>CDPRE6</td>
<td>DECOMPOSE CONCEPTUAL SCHEMA NDML REQUEST</td>
</tr>
<tr>
<td>CDP13</td>
<td>CDPRE6</td>
<td>DECOMPOSE CONCEPTUAL SCHEMA NDML REQUEST</td>
</tr>
<tr>
<td>ERRPRO</td>
<td>CDPRE6</td>
<td>DECOMPOSE CONCEPTUAL SCHEMA NDML REQUEST</td>
</tr>
<tr>
<td>OBINDN</td>
<td>CDPRE6</td>
<td>DECOMPOSE CONCEPTUAL SCHEMA NDML REQUEST</td>
</tr>
<tr>
<td>OCLOSE</td>
<td>CDPRE6</td>
<td>DECOMPOSE CONCEPTUAL SCHEMA NDML REQUEST</td>
</tr>
<tr>
<td>ODFINN</td>
<td>CDPRE6</td>
<td>DECOMPOSE CONCEPTUAL SCHEMA NDML REQUEST</td>
</tr>
<tr>
<td>OEXEC</td>
<td>CDPRE6</td>
<td>DECOMPOSE CONCEPTUAL SCHEMA NDML REQUEST</td>
</tr>
<tr>
<td>OFETCH</td>
<td>CDPRE6</td>
<td>DECOMPOSE CONCEPTUAL SCHEMA NDML REQUEST</td>
</tr>
</tbody>
</table>
DECOMPOSE CS REQUEST Where-external-routine-used List

<table>
<thead>
<tr>
<th>System</th>
<th>Module</th>
<th>Module Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>OOPEN</td>
<td>CDPRE6</td>
<td>DECOMPOSE CONCEPTUAL SCHEMA NDML REQUEST</td>
<td></td>
</tr>
<tr>
<td>OSQL3</td>
<td>CDPRE6</td>
<td>DECOMPOSE CONCEPTUAL SCHEMA NDML REQUEST</td>
<td></td>
</tr>
<tr>
<td>RPTERR</td>
<td>CDPRE6</td>
<td>DECOMPOSE CONCEPTUAL SCHEMA NDML REQUEST</td>
<td></td>
</tr>
</tbody>
</table>
3.10.7 Main Program Parts List

The following lists each Main Program listed in 3.10.1 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.
<table>
<thead>
<tr>
<th>Main Pgm Name</th>
<th>Module Name</th>
<th>Module Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>CDPRES</td>
<td></td>
<td>Purpose—DECOMPOSE CONCEPTUAL SCHEMA NDML REQUEST</td>
</tr>
<tr>
<td>CDDBTP</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>CDP13</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>ERRPRO</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>OBINDN</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>OCLOSE</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>ODFINN</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>OEXEC</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>OFETCH</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>OOPEN</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>OSQL3</td>
<td>External routine</td>
<td></td>
</tr>
<tr>
<td>RPTERR</td>
<td>External routine</td>
<td></td>
</tr>
</tbody>
</table>
3.10.8 Module Documentation

The following documentation describes information which is specific to each individual module being documented in this specification as listed in section 3.10.2. 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 3.10.7.

The Module Documentation is arranged alphabetically according to Module Name.
DECOMPOSE CS REQUEST Module Documentation

NAME: CDPRE6
PURPOSE: DECOMPOSE CONCEPTUAL SCHEMA NDML REQUEST
LANGUAGE: VAX-11 COBOL
MODULE TYPE: SUBROUTINE
SOURCE FILE: CDPRE6
SOURCE FILE TYPE: .COB
HOST: SUBSYSTEM: CDM
SUBDIRECTORY: DOCUMENTATION GROUP: PS41232

DESCRIPTION:

THE CS NDML DECOMPOSER IS A COMPILe-TIME MODULE WHOSE PURPOSE IS TO BREAK DOWN A CS NDML TRANSACTION INTO VARIOUS IS NDML SUBTRANSACTIONS. EACH SUBTRANSACTION ACCSESSES ONLY ONE DATABASE, MANAGED BY ONE DBMS, AT ONE COMPUTER; ITS RESULT IS A SINGLE RELATION.

THE DECOMPOSER MAPS FROM CS ATTRIBUTE USE CLASSES TO IS COUNTERPARTS, AND PASSES CS/IS TRANSFORM INFORMATION TO THE QUERY PROCESSOR GENERATOR.

THE DECOMPOSER ALSO GENERATES A JOIN QUERY GRAPH (JQG) AND RESULT FIELD TABLE (RFT) WHICH WILL BE INPUT TO THE STAGER/SCHEDULER CONFIGURATION ITEM TO DETERMINE THE BEST SEQUENCE OF INTERDATABASE JOINS AND UNIONS TO COMBINE THE RESULTS OF THE SUBTRANSACTIONS.

THE CS NDML DECOMPOSER PERFORMS THE FOLLOWING:

1. IT MAPS ALL CS ATTRIBUTE USE CLASSES TO IS COUNTERPARTS, DETERMINING THE LOCATIONS OF ALL ATTRIBUTE USE CLASSES INVOLVED.

2. IT IDENTIFIES ALL UNIONS, INTRADATABASE JOINS, AND INTERDATABASE JOINS AND UNIONS IN THE TRANSACTION.

3. IT REFORMATS THE ORIGINAL TRANSACTION INTO
SINGLE DATABASE SUBTRANSACTIONS, EACH OF WHICH RESULTS IN A SINGLE RELATION. IT PROVIDES A UNIQUE NAME FOR EACH RESULT RELATION.

4. IT CREATES A JOIN QUERY GRAPH (JQG) TO RECORD THE INTERDATABASE JOINS AND UNIONS NECESSARY TO COMPLETE THE TRANSACTION.

5. IT CREATES A RESULT FIELD TABLE (RFT) TO RECORD THE FIELDS THAT WILL COMPRISRE THE ANSWER TO THE TRANSACTION. WITH EACH ATTRIBUTE IS RECORDED ITS SUBTRANSACTION-ID.

ARGUMENTS:

CS-ACTION-LIST = RECRD
CS-QUALIFY-LIST = RECRD
ES-ACTION-LIST = RECRD
ES-QUALIFY-LIST = RECRD
UV-ABBR-LIST = RECRD
CODE-GENERATOR-TABLE = RECRD
IOS-IND = DSPLY [9]
PARCL1 = DSPLY [X(30)]
PARCL2 = DSPLY [X(30)]
PARCL3 = DSPLY [X(30)]
PARCL4 = DSPLY [X(30)]
MY-HOST = DSPLY [X(3)]
TARGET-HOST = DSPLY [X(3)]
SOURCE-LANGUAGE = DSPLY [X(10)]
CE-WORK-LIST = RECRD
ERROR-FILE = DSPLY [X(30)]
ORACLE-LDA = RECRD
RETURN-STATUS = DSPLY [X(5)]

INCLUDE FILES:

ERRCDM - IISS ERROR STATUS CODES FOR CDMP MODULES
REPLJN - REPL-JOIN-LIST SUBTRANSACTION JOINS
ISAL - INTERNAL SCHEMA ACTION LIST
ISQUAL - INTERNAL SCHEMA QUALIFY LIST
RFTABLE - THE RESULT FIELD TABLE
JQGTBL - JOIN QUERY GRAPH TELLS HOW TO CONNECT
          SUBTRANSACTIONS
APL - JOIN QUERY ATTRIBUTE PAIR LIST
SETTAB - LIST OF SETS OWNER-MEMBER RELATIONSHIPS
SUBPROC - SUBTRANSACTION PROCESSES ID TABLE
AUCDAT - ORACLE CURSOR DEFINITION
CSAL - CONCEPTUAL SCHEMA ACTION LIST
CSQUAL - CONCEPTUAL SCHEMA QUALIFY LIST
ESAL - EXTERNAL SCHEMA ACTION LIST
ESQUAL - EXTERNAL SCHEMA QUALIFY LIST
UVABBR - USER VIEW ABBREVIATION LIST
CGTABLE - CODE GENERATING TABLE- TRACKS ALL GENERATED SOFTWARE
CEWORK - CS TO ES WORK LIST INFORMATION
ORCLEDA - WS DEFINITION FOR THE ORACLE LOGIN AREA
LNKEDGE - **** PURPOSE NOT FOUND BY STRIPPER ****
DMPCSRL - DISPLAYS THE CONTENTS OF THE CS ACTION LIST
DUMPCSQ - DISPLAY THE CONTENTS OF THE CS QUALIFY LIST
DUMPJQG - DISPLAY THE CONTENTS OF THE JQG TABLE
DUMPAPL - DISPLAYS THE CONTENTS OF THE ATTRIBUTE PAIR LIST
DUMPSETT - DISPLAYS THE CONTENTS OF THE SET TABLE
DMPREPL - DISPLAYS THE CONTENTS OF THE REPL JOIN TABLE
DUMPGRP - DISPLAY THE CONTENTS OF THE GROUP TABLE
ERRPRO - PROCESS ERROR INCLUDE FILE

ROUTINES CALLED:
-----------------
RPTERR
CDP13
CDDBTP
OOPEN
OSQL3
ODFINN
OBINDN
OEXEC
OFETCH
OCLOSE
ERRPRO
3.10.9 Include File Descriptions

The following list contains a purpose and description of each include file listed in 3.10.4 as specified in the source code. The language it is written in is also given.
DECOMPOSE CS REQUEST Include File Description

FILE NAME: APL
PURPOSE: JOIN QUERY ATTRIBUTE PAIR LIST
LANGUAGE: VAX-11 COBOL

DESCRIPTION:

CONTAINS INFORMATION ABOUT THE JOIN
DECOMPOSE CS REQUEST Include File Description

FILE NAME: AUCDAT
PURPOSE: ORACLE CURSOR DEFINITION
LANGUAGE: VAX-11 COBOL

DESCRIPTION:
-----------

AUC QUERY
DECOMPOSE CS REQUEST Include File Description

FILE NAME: BEING
PURPOSE: DETERMINE DUPLICATE EDGES IN THE JQG
LANGUAGE: VAX-11 COBOL

DESCRIPTION:

DURING JQG COLLAPSING, DUPLICATE JQG ENTRIES MAY RESULT WITH DIFFERENT APL’S. THIS WILL BE EXECUTED AT THE END OF SENDS FOR A STAGE AND WILL FIND THE DUPLICATE EDGES AND HOOK THE APL’S TOGETHER BEFORE THE CIT IS REBUILT AT THE BEGINNING OF THE NEXT STAGE.
DECOMPOSE CS REQUEST Include File Description

FILE NAME: CEWORK
PURPOSE: CS TO ES WORK LIST INFORMATION
LANGUAGE: VAX-11 COBOL

DESCRIPTION:
DECOMPOSE CS REQUEST Include File Description

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

DESCRIPTION:

CODE GENERATOR TABLE, HOLDS PERTINENT RESULTS
ABOUT ALL CODE GENERATED OR MODIFIED BY THE
PRECOMPILER
NOTE: ROW 100 RESERVED FOR SWAPPING DURING SOR'ING.
DECOMPOSE CS REQUEST Include File Description

FILE NAME: CSAL
PURPOSE: CONCEPTUAL SCHEMA ACTION LIST
LANGUAGE: VAX-11 COBOL

DESCRIPTION:

TABLE TO HOLD CONCEPTUAL DATA ABOUT THE REQUEST

***** THE CONCEPTUAL SCHEMA ACTION LIST
DECOMPOSE CS REQUEST Include File Description

FILE NAME: CSQUAL
PURPOSE: CONCEPTUAL SCHEMA QUALIFY LIST
LANGUAGE: VAX-11 COBOL

DESCRIPTION:

CONTAINS CONCEPTUAL SCHEMA INFORMATION FOR THE REQUESTS QUALIFICATION

THE CONCEPTUAL SCHEMA QUALIFY LIST
DECOMPOSE CS REQUEST Include File Description

FILE NAME: DMPCSRL
PURPOSE: DISPLAYS THE CONTENTS OF THE CS ACTION LIST
LANGUAGE: VAX-11 COBOL

DESCRIPTION:

-------------
DECOMPOSE CS REQUEST Include File Description

FILE NAME: DMPREPL
PURPOSE: DISPLAYS THE CONTENTS OF THE REPL JOIN TABLE
LANGUAGE: VAX-11 COBOL

DESCRIPTION:  
----------------
DECOMPOSE CS REQUEST Include File Description

FILE NAME: DMPSETT
PURPOSE: DISPLAYS THE CONTENTS OF THE SET TABLE
LANGUAGE: VAX-11 COBOL

DESCRIPTION:
---------
DECOMPOSE CS REQUEST Include File Description

FILE NAME: DUMPAPL
PURPOSE: DISPLAYS THE CONTENTS OF THE ATTRIBUTE PAIR LIST
LANGUAGE: VAX-11 COBOL

DESCRIPTION:
-------------
DECOMPOSE CS REQUEST Include File Description

FILE NAME: DUMPCSQ
PURPOSE: DISPLAY THE CONTENTS OF THE CS QUALIFY LIST
LANGUAGE: VAX-11 COBOL

DESCRIPTION:

-------------
DECOMPOSE CS REQUEST Include File Description

FILE NAME: DUMPGRP
PURPOSE: DISPLAY THE CONTENTS OF THE GROUP TABLE
LANGUAGE: VAX-11 COBOL

DESCRIPTION:

-------------
DECOMPOSE CS REQUEST Include File Description

FILE NAME: DUMPJQG
PURPOSE: DISPLAY THE CONTENTS OF THE JQG TABLE
LANGUAGE: VAX-11 COBOL

DESCRIPTION:

-------------

3-43
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
DECOMPOSE CS REQUEST Include File Description

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

DESCRIPTION:
----------
DECOMPOSE CS REQUEST Include File Description

FILE NAME: ESAL
PURPOSE: EXTERNAL SCHEMA ACTION LIST
LANGUAGE: VAX-11 COBOL

DESCRIPTION:

CONTAINS THE EXTERNAL SCHEMA INFORMATION FOR AN NDML REQUEST

THE EXTERNAL SCHEMA ACTION LIST
DECOMPOSE CS REQUEST Include File Description

FILE NAME: ESQUAL
PURPOSE: EXTERNAL SCHEMA QUALIFY LIST
LANGUAGE: VAX-11 COBOL

DESCRIPTION:

CONTAINS EXTERNAL SCHEMA INFORMATION FOR THE NDML QUALIFICATION

THE EXTERNAL SCHEMA QUALIFY LIST
DECOMPOSE CS REQUEST Include File Description

FILE NAME: ISAL
PURPOSE: INTERNAL SCHEMA ACTION LIST
LANGUAGE: VAX-11 COBOL

DESCRIPTION:

CONTAINS INTERNAL SCHEMA INFORMATION ABOUT AN NDML REQUEST

THE INTERNAL SCHEMA ACTION LIST
DECOMPOSE CS REQUEST Include File Description

FILE NAME: ISQUAL
PURPOSE: INTERNAL SCHEMA QUALIFY LIST
LANGUAGE: VAX-11 COBOL

DESCRIPTION:
------------
CONTAINS INTERNAL SCHEMA INFORMATION FOR AN NDML QUALIFICATION

THE INTERNAL SCHEMA QUALIFY LIST
DECOMPOSE CS REQUEST Include File Description

FILE NAME: JQGTL
PURPOSE: JOIN QUERY GRAPH TELLS HOW TO CONNECT SUBTRANSACTIONS
LANGUAGE: VAX-11 COBOL

DESCRIPTION:
-------------
DECOMPOSE CS REQUEST Include File Description

FILE NAME: ORCLEDA
PURPOSE: WS DEFINITION FOR THE ORACLE LOGIN AREA
LANGUAGE: VAX-11 COBOL

DESCRIPTION:

-------------

THE ORACLE LOGON DATA AREA
DECOMPOSE CS REQUEST Include File Description

FILE NAME: REPLJN
PURPOSE: REPL-JOIN-LIST SUBTRANSACTION JOINS
LANGUAGE: VAX-11 COBOL

DESCRIPTION:

POTENTIAL INTER- AND INTRA- SUBTRANSACTION JOINS
DECOMPOSE CS REQUEST Include File Description

FILE NAME: RFTABLE
PURPOSE: THE RESULT FIELD TABLE
LANGUAGE: VAX-11 COBOL

DESCRIPTION:

---------------

CONTAINS CONCEPTUAL SCHEMA INFORMATION ABOUT
THE RESULTS OF AN NDML REQUEST
FILE NAME: SETTAB
PURPOSE: LIST OF SETS OWNER-MEMBER RELATIONSHIPS
LANGUAGE: VAX-11 COBOL

DESCRIPTION:

SET TABLE TO KEEP TRACK OF CODASYL NDML REQUESTS IN TERMS OF OWNER AND MEMBER RELATIONSHIPS
DECOMPOSE CS REQUEST Include File Description

FILE NAME: SUBPROC
PURPOSE: SUBTRANSACTION PROCESSES ID TABLE
LANGUAGE: VAX-11 COBOL

DESCRIPTION:

-----------

THIS TABLE MUST HAVE THE SAME NUMBER OF OCCURS
DECOMPOSE CS REQUEST Include File Description

FILE NAME: UVABB
PURPOSE: USER VIEW ABBREVIATION LIST
LANGUAGE: VAX-11 COBOL

DESCRIPTION:
-----------------
CONTAINS THE ABBREVIATIONS FOR ALL USER
VIEW REFERENCED IN THE NDML REQUEST
3.10.10 Hierarchy Chart

The following hierarchy charts show the relationships between all of the modules mentioned in the above documentation. A module may call a subroutine several times within its code, but the call will only be shown once as a single relationship on this hierarchy chart. All modules shown at the top of the first page are considered Main Programs as described in section 3.10.1 above.

There is an internal paging scheme as marked by the numbers in the upper right corner of each page. An index after the last page of the chart shows where a routine and its calls are first defined. If a routine has no page reference, it either makes no calls or is an external routine. A continuation box on the end of a tree limb shows where that the tree continues on the page numbered mentioned. A number in a box with a routine name points to the page where the routine is further defined within the hierarchy tree. If there is no number in a box, the routine either makes no calls or is an external routine.
PS 620141232
1 November 1985

---

I (CONT) I

I ODFINN I

I OBINDN I

I OEXEC I

I OFETCH I

I (CONT) I

---1---

---3---

---59---
PS 620141232
1 November 1985

CDDBTP
CDP13
CDPRES...
ERRPRO
OBINDN
OCLOSE
ODFILLN
OEXEC
OFETCH
OPEN
OSQL3
RPTERR
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 consist 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."
END

7-87

DTIC