World Software

Advanced Programming Concepts and Skills

Release A8.1
The information in this guide is confidential and a proprietary trade secret of J.D. Edwards & Company. It may not be copied, distributed, or disclosed without prior written permission. This guide is subject to change without notice and does not represent a commitment on the part of J.D. Edwards & Company. The software described in this guide is furnished under a license agreement and may be used or copied only in accordance with the terms of the agreement. J.D. Edwards & Company uses an automatic software disabling routine to monitor the license agreement. For more details about this routine, please refer to the Technical Foundation Guide.
Online Help
- Program
- Form
- Field

CD-ROM Guides

Technical Foundation
System Administration and Environment Fundamentals
- Understanding Your Environment
- Creating and Maintaining Environments
- Setting Up Security
- Upgrading Your System

Common Foundation
Prerequisite
J.D. Edwards Software Fundamentals
- Using Menus
- Getting Help
- Customizing Data
- Reporting
Important Note for Students in Training Classes

This guide is a source book for online helps, training classes, and user reference. Training classes may not cover all the topics contained here.
Welcome

About this Guide

This guide provides overviews, illustrations, procedures, and examples for release A7.3 of J.D. Edwards software. Forms (screens and windows) shown are only examples. If your company operates at a different software level, you might find discrepancies between what is shown in this guide and what you see on your screen.

This guide includes examples to help you understand how to use the system. You can access all of the information about a task using either the guide or the online help.

Before using this guide, you should have a fundamental understanding of the system, user defined codes, and category codes. You should also know how to:

- Use the menus
- Enter information in fields
- Add, change, and delete information
- Create and run report versions
- Access online documentation

Audience

This guide is intended primarily for the following audiences:

- Users
- Classroom instructors
- Client Services personnel
- Consultants and implementation team members

Organization

This guide is divided into sections for each major function. Sections contain chapters for each task or group of related tasks. Each chapter contains the information you need to accomplish the task, run the program, or print the
report. Chapters normally include an overview, form or report samples, and procedures.

When it is appropriate, chapters also might explain automatic accounting instructions, processing options, and warnings or error situations. Some chapters include self-tests for your use outside the classroom.

This guide has a detailed table of contents and an index to help you locate information quickly.

**Conventions Used in this Guide**

The following terms have specific meanings when used in this guide:

- *Form* refers to a screen or a window.
- *Table* generally means “file.”

We assume an “implied completion” at the end of a series of steps. That is, to complete the procedure described in the series of steps, either press Enter or click OK, except where noted.
## Table of Contents

J.D. Edwards Overview ........................................... 1-1  
Signing On and Off ............................................. 1-1  
Standard Screen Function Keys ................................. 1-3  
J.D. Edwards Product Line .................................... 1-6  
J.D. Edwards Regional Offices and Worldwide Offices ...... 1-9  
Application Development Cycle ................................. 1-11  
Universal Building Blocks of J.D. Edwards Software ...... 1-11  
J.D. Edwards Training Environment ............................ 1-13  
APCS System Overview .......................................... 1-17  
Features ......................................................... 1-17

## Version Control

Objectives ....................................................... 2-1  
About Version Control .......................................... 2-1  
Version Control Process Flow ................................ 2-4  
Version Control Menu Overview ............................... 2-5  
Development Environment ...................................... 2-7  
  About a Development Environment ......................... 2-7  
       Rules for Creating Development Environments ...... 2-7  
  J.D. Edwards Libraries ...................................... 2-8  
  Production and Development Examples .................. 2-10  
  Creating Libraries ........................................... 2-13  
  Creating Common and Data Libraries ................... 2-14  
  Creating a Development Object Library ................. 2-16  
  Creating a Development Source Library ............... 2-17  
  Creating JDESRC with J.D. Edwards Program Generator 2-18  
  Creating JDESRC Without the Program Generator ....... 2-20  
About User Profiles ............................................ 2-21  
Defining Access for a User Profile using J98INIT ......... 2-21  
Defining Access for a User Profile Using J98INIT ...... 2-23  
Copy Data to Your Development Environment ............... 2-23  
Copying a Library ............................................. 2-24  
Copying a File ................................................ 2-25  
Copying a Record .............................................. 2-26  
Copying J.D. Edwards Record Types ...................... 2-28  
Project Management ........................................... 2-31  
  About Project Management ................................ 2-31  
  Understanding Work Order Processing ................... 2-31  
  Creating Work Orders .................................... 2-32  
  Accessing the Scheduling Workbench ..................... 2-40  
  Adding Record Types ...................................... 2-44  
  Changing Record Types ..................................... 2-45
Advanced Programming Concepts and Skills

Work with Software Action Requests ........................................... 2–47
About SAR System Setup ............................................................ 2–47
Creating Record Type Codes ....................................................... 2–48
Defining Record Type Titles ....................................................... 2–50
Work with Software Versions Repository .................................... 2–53
Working with Software Versions Repository (SVR) ....................... 2–53
Accessing the Software Versions Repository ............................... 2–55
Member Identifiers ..................................................................... 2–56
Naming Conventions ................................................................. 2–65
The J.D. Edwards System Codes ................................................... 2–70
Examples of Program and File Names ......................................... 2–71
Optional Files Workbench .......................................................... 2–72
Navigation Functions ................................................................. 2–77
CASE Profiles .......................................................................... 2–83
About CASE Profiles .................................................................. 2–83
Accessing CASE Profiles ............................................................ 2–84
Function Key Exits From the CASE Profiles Program ................... 2–88
Summary of CASE Profiles ....................................................... 2–89
Working with SAR Log ............................................................... 2–91
About SAR Log ........................................................................ 2–91
Setting Up User Input Options for SAR Logging ......................... 2–92
Selecting Types of SAR Information to Log ................................. 2–94
Accessing SAR Log Inquiry ......................................................... 2–95
Summary of the SAR Log Inquiry ................................................. 2–98
Work with Promotion Paths and Projects ................................. 2–99
Working with Promotion Paths and Projects ............................... 2–99
Understanding Promotion Paths ................................................ 2–100
Defining a Promotion Path ........................................................ 2–102
Defining a Project ...................................................................... 2–107
Promote a Project ..................................................................... 2–115
Promoting a Project ................................................................... 2–115
Update the SARs ....................................................................... 2–116
Validating a Promotion Path ....................................................... 2–118
Promoting a Project ................................................................... 2–122
Promote Project Updates ............................................................ 2–125
Promoting Project Updates ......................................................... 2–125

Programming Tools

Objectives ................................................................................. 3–1
About Programming Tools .......................................................... 3–1
Work with Data Modeling .......................................................... 3–3
Working with Data Modeling ....................................................... 3–3
Accessing Data Modeling ........................................................... 3–4
Work with the Object Cross Reference Repository ..................... 3–11
Working with the Object Cross Reference Repository ................. 3–11
Example .................................................................................. 3–12
Work with Data Dictionary ......................................................... 3–15
About the Data Dictionary Repository ......................................... 3–15
Understanding the Data Dictionary Structure .............................. 3–16
Locating A Data Item Name ........................................... 3-18
Working with the Data Dictionary ................................. 3-19
  What You Should Know About ................................... 3-24
  The Function Keys for the Data Dictionary ................ 3-24
Working with Data Item Alias Revisions ......................... 3-25
Working with the Data Dictionary Glossary ..................... 3-26
  What are the Data Dictionary Glossary Groups? .......... 3-26
Working with User Defined Help Instructions .................. 3-30
Working with Data Field Descriptions ......................... 3-31
Working with the Next Numbers Facility ....................... 3-32
About the Field Reference File ................................. 3-34
  What Happens with the Rebuild? ............................ 3-35
About the J.D. Edwards Message File .......................... 3-35
  Rebuilding only the J.D. Edwards Message File? ........ 3-35
Locating the Rebuild FRF and JDE Msg File Form ............... 3-35
Work with Data File Design Aid ................................. 3-37
  About the Data File Design Aid ............................. 3-37
  About Assigning the File Prefix ............................ 3-39
Entering Data File Design Aid .................................. 3-40
Function Keys From File Design Aid ............................ 3-45
What Are the Data File Design Aid Standards? ................. 3-47
  Merge Functions for Program Temporary Fix (PTF) Installations and Reinstallations .................. 3-49
Data File Design Aid Summary .................................. 3-50
Work with Screen Design Aid .................................... 3-51
  About Screen Design Aid .................................... 3-51
Editing Commands ................................................ 3-52
Prefix Standards .................................................. 3-53
Field Name Standards ............................................ 3-54
  Updating or Adding Fields through SDA .................... 3-55
Working with Screen Design Aid ................................. 3-56
Function Key Exits ............................................... 3-56
Updating an Existing Field ...................................... 3-57
Accessing Fast Path Create for a New Form .................... 3-60
Adding Fields without Using a Pick List ....................... 3-63
Adding a Literal Field .......................................... 3-66
Using the *BOTH and *ALL Features ............................ 3-67
  Using *BOTH .................................................. 3-67
  Using *ALL ................................................. 3-69
  Understanding the SDA Exit/Save Function Key .......... 3-71
Compiling Your Form .............................................. 3-73
Screen Design Standards and Tips .............................. 3-74
Adding Video Fields Using Pick List ............................ 3-80
  About Record Formats ...................................... 3-81
Selecting Database Fields ...................................... 3-83
Placing Fields on a Form Using a Pick List ................... 3-88
Function Key Exits from Screen Design Aid .................... 3-89
  Hidden Fields ............................................. 3-94
Changing Subfile Boundaries ................................... 3-97
Process Overview - Placing Selected Fields ................... 3-98
Process Overview - Revising the Field Definition ............ 3-100
Process Overview - Revising Vocabulary and Function Keys 3–101
Function Keys for Form and Display Format Control 3–101
Summary of Screen Design Aid 3–102
Work with Report Design Aid 3–105
About Report Design Aid 3–105
Example - RDA and DREAM Writer 3–106
Comparing RDA and SDA – Field Definition Form 3–106
Cover Page Fields 3–107
Report Header Fields 3–108
What Are the Report Formats? 3–109
What Are the Report Design Standards? 3–110
About Designing the Report 3–112
Accessing Report Design Aid 3–113
Updating a Field in RDA 3–114
Understanding the Report Design Aid Function Keys 3–115
Compiling A Report 3–121
Changing the Compile Option Defaults for Reports 3–123

Programming Standards

Objectives 4–1
Programming Standards 4–1
Program Specifications 4–3
About Program Specifications 4–3
What Are Control Specifications? 4–4
What Are File Description Specifications? 4–5
What Are Extension Specifications? 4–6
What Are Input Specifications? 4–7
What Are Calculation Specifications? 4–8
What Are Output Specifications? 4–9
Program Overview 4–11
About the Program Overview 4–11
Subroutines 4–12
Error Handling 4–15
Indicator Usage 4–17
Documentation 4–19
Miscellaneous Items 4–21
Program Structure 4–25
About Program Structure 4–25
User Spaces 4–67
About User Spaces 4–67
What Is a User Space? 4–68
What Are the Advantages of Using a User Space? 4–68
How Does a User Space Function? 4–69
Creating a User Space 4–69
Writing to a User Space 4–72
Reading from a User Space 4–74
User Indices 4–75
About User Indices 4–75
What Are the Advantages of Using a User Index? 4–76
How Does a User Index Function? ........................................... 4–77
Creating a User Index ......................................................... 4–78
Writing to a User Index ....................................................... 4–81
Appearance of Records ...................................................... 4–83
Retrieving Data from a User Index ....................................... 4–84
File Servers ........................................................................ 4–107
About File Servers ............................................................. 4–107
What is a File Server? .......................................................... 4–108
What are the Advantages of Using a File Server? ................. 4–109
What are the Disadvantages of Using a File Server? ............. 4–109
How Does a File Server Function? ....................................... 4–110
What Are Control Parameters? .......................................... 4–111
What Are Returned Parameters? ........................................ 4–113
Implementing a File Server .................................................. 4–114
Searching for Key Lists ...................................................... 4–114
Tips when Using File Servers .............................................. 4–116
Commonly Used File Servers .............................................. 4–122
Functional Servers ............................................................. 4–123
About Functional Servers .................................................... 4–123
What Are Functional Servers? ............................................ 4–124
What are the Advantages of Using a Functional Server? ....... 4–125
What are the Disadvantages of Using a Functional Server? 4–126
Setting Up Business Rules for an Entry Program .................. 4–126
How Does a Functional Server Function? ......................... 4–126
Functional Server Highlights .............................................. 4–127
The Call Parameters for the Functional Server .................... 4–141
Control Fields within the User Space ................................. 4–144
Error Message Index Line (C00R1X) ...................................... 4–145
Example – Functional Server Program Sections ................. 4–146
Available Functional Servers ............................................. 4–151
Source Debugger ............................................................... 4–153
About Source Debugger ..................................................... 4–153
Using Debugger with an Interactive Program ...................... 4–154
Using Debugger with a Batch Program ............................... 4–157
Software Scan and Replace ............................................... 4–169
About Software Scan and Replace ...................................... 4–169
To Work with Software Scan and Replace ......................... 4–169
Report .............................................................................. 4–170
Guidelines ....................................................................... 4–170
Performance Issues ............................................................ 4–171

Group Jobs

Objectives ....................................................................... 5–1
About Group Jobs ............................................................. 5–1
Access the J.D. Edwards Group Job Form ......................... 5–3
About the J.D. Edwards Group Job Form ......................... 5–3
Accessing the J.D. Edwards Group Job Form .................... 5–5
Creating New Group Jobs .................................................. 5–6
Activating Suspended Group Jobs ..................................... 5–7
### Advanced Programming Concepts and Skills

- Terminating Group Jobs ........................................... 5–8
- Changing to Non-Group Mode .................................... 5–9
- Signing Off with Suspended Group Jobs ....................... 5–10
- Work with Non-J.D. Edwards Group Jobs ...................... 5–11
- Advanced Functions of the J.D. Edwards Group Job Form .... 5–13
- Work with the Attention MENU Form .......................... 5–15
- About the Attention MENU Form ............................... 5–15
- Accessing the J.D. Edwards Attention Menu Form .......... 5–17
- Work with IBM Pass-Through ................................... 5–19
- About Working with IBM Pass-Through ....................... 5–19
- Setting Up Access to Remote Locations ....................... 5–20
- Using IBM Pass-Through with Group Jobs ................... 5–22

### Universal File Converter

- Objectives .......................................................... 6–1
- About Universal File Converter ............................... 6–1
- Set Up Universal File Converter .............................. 6–7
- About Universal File Converter ............................... 6–7
- Understanding the Universal File Converter Setup .......... 6–8
- Setting Up Universal File Converter ......................... 6–8
- Work with Crossover Rules .................................... 6–13
  - Displaying Field Descriptions ............................ 6–19
  - Adding Fields .................................................. 6–21
  - Deleting Records ............................................. 6–22
  - Keywords ....................................................... 6–23
  - Available Functions and Options .......................... 6–26
- Work with File Conversion .................................... 6–29
  - Working with File Conversion ............................. 6–29
- Print a Report .................................................... 6–33
  - Printing a Report ............................................. 6–33
- Create Conversion Forms ...................................... 6–35
  - Creating Conversion Forms ............................... 6–35
  - Creating Conversion Forms ............................... 6–36
- Work with the Data Dictionary Glossary by File ......... 6–39
  - About Working with the Data Dictionary Glossary by File 6–39
  - Accessing the Data Dictionary Glossary by File ........ 6–40
  - Adding a File Specific Glossary Item .................... 6–41
  - Printing the Data Dictionary Glossary Information .... 6–41

### Appendices

- Appendix A - Common & Production Library Files ........ A–1
- Appendix B - Upgrading Customized Source Code .......... B–1
  - S/Compare .................................................... B–1
  - Harmonizer ................................................... B–3
  - About Harmonizer Plus .................................... B–4
J.D. Edwards Overview

Signing On and Off

To sign on

From the Sign On menu:

1. In the User field, type your User ID
2. In the Password field, type your Password
3. Press Enter

To sign off

On the Selection line:

1. Type a double period (..) or 90
2. Press Enter
## Standard Menu Function Keys

The following charts show the standard function keys on the AS/400 and their equivalents on the PC.

<table>
<thead>
<tr>
<th>AS/400 Keyboard</th>
<th>PC Keyboard</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>F4</td>
<td>F4</td>
<td>Command Entry Prompt</td>
</tr>
<tr>
<td>F8</td>
<td>F8</td>
<td>Access Menu Word Search</td>
</tr>
<tr>
<td>F9</td>
<td>F9</td>
<td>Retrieve previous command</td>
</tr>
<tr>
<td>F12</td>
<td>F12</td>
<td>Return to previous menu</td>
</tr>
<tr>
<td>F13</td>
<td>Shift + F1</td>
<td>Fast Path Commands</td>
</tr>
<tr>
<td>F14</td>
<td>Shift + F2</td>
<td>Menu Selection Detail</td>
</tr>
<tr>
<td>F16</td>
<td>Shift + F4</td>
<td>Display Menu List window</td>
</tr>
<tr>
<td>F18</td>
<td>Shift + F6</td>
<td>Access processing options</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Type desired menu selection and press F18</td>
</tr>
<tr>
<td>F24</td>
<td>Shift + F12</td>
<td>List available Function Keys</td>
</tr>
</tbody>
</table>
Standard Screen Function Keys

<table>
<thead>
<tr>
<th>AS/400 Keyboard</th>
<th>PC Keyboard</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>F1</td>
<td>F1</td>
<td>Display JDE field level help</td>
</tr>
<tr>
<td>F3</td>
<td>F3</td>
<td>Exit</td>
</tr>
<tr>
<td>F4</td>
<td>F4</td>
<td>Display Fold Area (more detailed information)</td>
</tr>
<tr>
<td>F7</td>
<td>F7</td>
<td>View error message text</td>
</tr>
<tr>
<td>F22</td>
<td>Shift F10</td>
<td>Clear screen</td>
</tr>
<tr>
<td>F24</td>
<td>Shift F12</td>
<td>Display available functions window</td>
</tr>
</tbody>
</table>

Additional Differences

The following table shows additional keys used on the AS/400 and their equivalents on the PC.

<table>
<thead>
<tr>
<th>AS/400</th>
<th>PC Keyboard</th>
</tr>
</thead>
<tbody>
<tr>
<td>Field Exit</td>
<td>Enter</td>
</tr>
<tr>
<td>Enter</td>
<td>Ctrl</td>
</tr>
<tr>
<td>Reset</td>
<td>Alt</td>
</tr>
<tr>
<td>Roll Up</td>
<td>Page Down</td>
</tr>
<tr>
<td>Roll Down</td>
<td>Page Up</td>
</tr>
<tr>
<td>Help</td>
<td>Scroll Lock</td>
</tr>
<tr>
<td>Attn</td>
<td>Esc</td>
</tr>
</tbody>
</table>
Frequently Used Hidden Selections

Every J.D. Edwards menu displays up to 24 menu selections typically unique to a system. Hidden Selections are menu selections that let you perform certain functions regardless of the current menu. Hidden Selections can display the menus for Advanced and Technical Operations for a particular application, perform special activities, access certain menus even if the system restricts direct menu traveling, and access certain IBM commands without allowing access to the Command Entry Line.

To access a Hidden Selection, from any J.D. Edwards menu, enter HS in the Selection line. The Hidden Selections form displays, listing the selection number for each function. Enter 4 in the field to the left of the Hidden Selection that you want. Alternatively, type the desired number (two digit code) on the Selection or Command line and press Enter.
### User Tools

<table>
<thead>
<tr>
<th>Selection</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>33</td>
<td>Display Submitted Jobs</td>
</tr>
<tr>
<td>34</td>
<td>Display User Messages</td>
</tr>
<tr>
<td>42</td>
<td>Display User Job Q</td>
</tr>
<tr>
<td>43</td>
<td>Display User Print Q</td>
</tr>
<tr>
<td>39</td>
<td>Change User Print Q</td>
</tr>
<tr>
<td>82</td>
<td>Hold Submitted Jobs</td>
</tr>
<tr>
<td>85</td>
<td>Display User Defaults</td>
</tr>
<tr>
<td>90</td>
<td>Sign Off</td>
</tr>
</tbody>
</table>

### Operator Tools

<table>
<thead>
<tr>
<th>Selection</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>27</td>
<td>Advanced Operations</td>
</tr>
<tr>
<td>29</td>
<td>Technical Operations</td>
</tr>
<tr>
<td>97</td>
<td>Install History Display</td>
</tr>
</tbody>
</table>

### Programming Tools

<table>
<thead>
<tr>
<th>Selection</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>25</td>
<td>Menu Specifications</td>
</tr>
<tr>
<td>40</td>
<td>File Field Description</td>
</tr>
</tbody>
</table>

Type HS on a *Selection or Command* line to display a list of available Hidden Selections.
J.D. Edwards Product Line

The following is a list of products available from J.D. Edwards:

**Financials**

- General Accounting
- Accounts Payable
- Accounts Receivable
- Fixed Assets
- Financial Modeling and Budgeting
- Multi-Currency, Multi-Language, Multi-National Processing
- Flexible Reporting Tools
- Address Book/Electronic Mail
- Human Resources
- Payroll
- Time Accounting

**Distribution/Logistics**

- Sales Order Management
- Configuration Management
- Advanced Pricing
- Forecasting
- Requirements Planning
- Enterprise Facility Planning
- Purchase Management
- Inventory Management
- Advanced Warehouse Management
- Transportation Management
- Data Collection
- EDI/Electronic Commerce
Manufacturing

- Product Data Management
- Configuration Management
- Plant and Equipment Maintenance
- Shop Floor Control
- Forecasting
- Requirements Planning
- Enterprise Facility Planning
- Capacity Requirements Planning
- Finite Scheduler
- Environmental Management System
- Data Collection

Energy and Chemical

- Process Manufacturing/Lube Oil Blending
- Equipment Management
- Inventory Management
- Bulk Stock Control
- Distribution Contracts
- Sales Order Management and Pricing
- Load and Delivery Management
- Forecasting
- Enterprise Facility Planning
- Purchase Management

Architecture, Engineering, Construction, and Real Estate

- Job/Project Cost Accounting
- Work Order Management
- Project Change Management
- Contract Management
- Contract Billing
- Engineering and Service Billing
- Equipment Management
- Homebuilder Management
- Real Estate Management

**Public Services: State and Local Governments, Education, and Utilities**

- Financial Administration and Reporting
- Budget Administration
- Fund and Encumbrance Accounting
- Grant and Endowment Management
- Purchasing and Material Management
- Warehousing and Central Stores Management
- Human Resources Management
- Service and Word Order Management
- Capital Project and Construction Management
- Contract Management
- Plant, Equipment, and Fleet Maintenance
- Customer Information and Billing Administration
- Assessment and Property Tax Administration

**Other Integrated Solutions**

- Bar Coding/Data Collection
- Connectivity/Network Solutions
- Development Tools
- Distributed Data Processing
- EDI/Electronic Commerce
- Enterprise Information Systems
- Facsimile Management
- PC Integration
J.D. Edwards Regional Offices and Worldwide Offices

The following is a list of all J.D. Edwards offices:

<table>
<thead>
<tr>
<th>Office</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Headquarters</td>
<td>Denver, Colorado</td>
</tr>
<tr>
<td>Regional U.S. Offices</td>
<td>East Rutherford, New Jersey</td>
</tr>
<tr>
<td></td>
<td>Herndon, Virginia</td>
</tr>
<tr>
<td></td>
<td>Atlanta, Georgia</td>
</tr>
<tr>
<td></td>
<td>Oak Brook, Illinois</td>
</tr>
<tr>
<td></td>
<td>Denver, Colorado</td>
</tr>
<tr>
<td></td>
<td>Costa Mesa, California</td>
</tr>
<tr>
<td></td>
<td>Foster City, California</td>
</tr>
<tr>
<td></td>
<td>Dallas, Texas</td>
</tr>
<tr>
<td></td>
<td>Houston, Texas</td>
</tr>
<tr>
<td></td>
<td>U.S. Satellite Offices</td>
</tr>
<tr>
<td></td>
<td>Waltham, Massachusetts</td>
</tr>
<tr>
<td></td>
<td>Beachwood, Ohio</td>
</tr>
<tr>
<td></td>
<td>Trumbull, Connecticut</td>
</tr>
<tr>
<td></td>
<td>Buffalo, New York</td>
</tr>
<tr>
<td></td>
<td>Melville, New York</td>
</tr>
<tr>
<td></td>
<td>New York, New York</td>
</tr>
<tr>
<td></td>
<td>Fair Oaks, California</td>
</tr>
<tr>
<td></td>
<td>Seattle, Washington</td>
</tr>
<tr>
<td></td>
<td>West Conshohocken, Pennsylvania</td>
</tr>
<tr>
<td></td>
<td>Bloomington, Minnesota</td>
</tr>
<tr>
<td></td>
<td>Milwaukee, Wisconsin</td>
</tr>
<tr>
<td></td>
<td>Lake Oswego, Oregon</td>
</tr>
<tr>
<td></td>
<td>St. Louis, Missouri</td>
</tr>
<tr>
<td></td>
<td>Tampa, Florida</td>
</tr>
<tr>
<td></td>
<td>Fort Lauderdale, Florida</td>
</tr>
<tr>
<td></td>
<td>Regional Canada</td>
</tr>
<tr>
<td></td>
<td>Willowdale, Ontario</td>
</tr>
<tr>
<td>North and South American</td>
<td>Canada</td>
</tr>
<tr>
<td>Affiliates</td>
<td>Mexico</td>
</tr>
<tr>
<td></td>
<td>Venezuela</td>
</tr>
<tr>
<td></td>
<td>Argentina</td>
</tr>
<tr>
<td>European Offices</td>
<td>Frankfurt, Germany</td>
</tr>
<tr>
<td></td>
<td>Bruxelles, Belgium</td>
</tr>
<tr>
<td></td>
<td>Paris, France</td>
</tr>
<tr>
<td></td>
<td>Milano, Italy</td>
</tr>
<tr>
<td></td>
<td>United Kingdom</td>
</tr>
<tr>
<td></td>
<td>Bourne End, U.K.</td>
</tr>
<tr>
<td>Office</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>----------------------------------------------------------</td>
</tr>
<tr>
<td>European Affiliates</td>
<td>United Kingdom</td>
</tr>
<tr>
<td></td>
<td>Ireland</td>
</tr>
<tr>
<td></td>
<td>Sweden</td>
</tr>
<tr>
<td></td>
<td>Germany</td>
</tr>
<tr>
<td></td>
<td>The Netherlands</td>
</tr>
<tr>
<td></td>
<td>Belgium</td>
</tr>
<tr>
<td></td>
<td>Austria</td>
</tr>
<tr>
<td></td>
<td>Switzerland</td>
</tr>
<tr>
<td></td>
<td>Spain</td>
</tr>
<tr>
<td></td>
<td>Portugal</td>
</tr>
<tr>
<td></td>
<td>Denmark</td>
</tr>
<tr>
<td>Australian Office</td>
<td>Chatswood, Australia</td>
</tr>
<tr>
<td>Middle East Affiliates</td>
<td>Israel</td>
</tr>
<tr>
<td></td>
<td>Jordan</td>
</tr>
<tr>
<td></td>
<td>Bahrain</td>
</tr>
<tr>
<td></td>
<td>Egypt</td>
</tr>
<tr>
<td>Asia/Pacific Rim Affiliates</td>
<td>Japan</td>
</tr>
<tr>
<td></td>
<td>China–Hong Kong City</td>
</tr>
<tr>
<td></td>
<td>Philippines</td>
</tr>
<tr>
<td></td>
<td>Malaysia</td>
</tr>
<tr>
<td></td>
<td>Singapore</td>
</tr>
<tr>
<td></td>
<td>Australia</td>
</tr>
<tr>
<td></td>
<td>New Zealand</td>
</tr>
</tbody>
</table>
Application Development Cycle

World Computer Aided Software Engineering (CASE) covers the entire spectrum of the application development life cycle, including design tools, code generation, automatic documentation generation, prototyping, repositories and other productivity improvement tools for the development, operation and maintenance of flexible, business application software.

You can describe the Application Development Cycle (A/D Cycle) in three levels, as follows:

**Level 1**

- The Application Platform, which is described in the *Technical Foundation* class.

**Level 2**

- The Design Platform, which is described in the *Advanced Programming Concepts and Skills* (APCS) class.

**Level 3**

- The Development Platform, which is described in the *CASE* class.

Universal Building Blocks of J.D. Edwards Software

World CASE covers the entire spectrum of the application development life cycle, including:

- Design tools
- Code generation
- Automatic documentation generation
- Prototyping
- Repositories
- Other productivity improvement tools

The following figure shows the separate modules that contribute to the functioning of a J.D. Edwards program.
J.D. Edwards Training Environment

A brief look at the Student Library setup will help you understand the training environment set up for your learning experience. The following is a description of signon naming conventions. There is also a description of library naming conventions.

- Signon Naming Conventions

Your signon depends on where you are located.

For example, in the Denver Headquarters Office, we have several classroom numbers, so the structure of signons is as follows:

- Library Naming Conventions

Your library names depends on where you are located.

For example, in the Denver Headquarters Office, we have several classroom numbers and libraries that are structured for that classroom. You will also have your own student library. That library will have the naming conventions of your student number. Other libraries contained in your library list are libraries that are standard to all J.D. Edwards class environments.
The library list at a J.D. Edwards facility appears as follows:

<table>
<thead>
<tr>
<th>Name</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>QTEMP</td>
<td>IBM temporary library</td>
</tr>
<tr>
<td>COMMON</td>
<td>Common library for training. Used for all J.D. Edwards Training Environments. It contains files that all training classes can share.</td>
</tr>
<tr>
<td></td>
<td>For example, Help Files, Message Files, and Field Reference Files.</td>
</tr>
<tr>
<td>STxxyyOBJ</td>
<td>Students object library. Used for the student to compile custom objects. It contains only programs that a student may modify in a class exercise.</td>
</tr>
<tr>
<td>(xx=classrm #) (yy=student #)</td>
<td></td>
</tr>
<tr>
<td>JDFOBJ</td>
<td>Common object library for training. Contains all of J.D. Edwards execution programs. All J.D. Edwards training environments use this library.</td>
</tr>
<tr>
<td>STxxyyDTA</td>
<td>Students data library. Used for the students custom data files. It contains only programs that a student may modify in a class exercise.</td>
</tr>
<tr>
<td>(xx=classrm #) (yy=student #)</td>
<td></td>
</tr>
<tr>
<td>xxSHARE</td>
<td>Classroom shared library. Is shared for that particular classroom environment. It contains files that the students all share. For example, the Data Dictionary file.</td>
</tr>
<tr>
<td>(xx=classrm #)</td>
<td></td>
</tr>
<tr>
<td>TRNSHARE</td>
<td>Shared library for all training. Used for all J.D. Edwards training environments. It contains files that all training classes can share. For example, word search files.</td>
</tr>
<tr>
<td>STxxyySRC</td>
<td>Students Source Library. Used for the student to write custom source programs. It contains only programs that a student may modify in a class exercise.</td>
</tr>
<tr>
<td>(xx=classrm #) (yy=student #)</td>
<td></td>
</tr>
<tr>
<td>JDFSRC</td>
<td>Common Source Library for Training. Contains all of J.D. Edwards source code programs. All J.D. Edwards training environments use this library.</td>
</tr>
<tr>
<td>QGPL</td>
<td>IBM general purpose library</td>
</tr>
</tbody>
</table>

1-14
The library list at an on-site location will appear as follows:

<table>
<thead>
<tr>
<th>Name</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>QTEMP</td>
<td>IBM temporary library</td>
</tr>
<tr>
<td>STUDSHARE</td>
<td>Contains files that will be shared for all students in class</td>
</tr>
<tr>
<td>STUDENTxoD</td>
<td>Contains files that will not be shared. Files are unique for each student.</td>
</tr>
<tr>
<td>STUDENTxoO</td>
<td>Contains any programs or objects that the student modifies in class (custom objects)</td>
</tr>
<tr>
<td>STUDENTxoS</td>
<td>Contains any source code that the student modifies in class (custom source)</td>
</tr>
<tr>
<td>JDETRAIN</td>
<td>Contains all J.D. Edwards execution programs</td>
</tr>
<tr>
<td>QGPL</td>
<td>IBM general purpose library</td>
</tr>
</tbody>
</table>

**Classes**

Classes consist of lectures and exercises. While each exercise is a separate task, they ultimately build upon each other to create a new program. It is imperative, therefore, that each student fully understand each exercise before continuing. At the end of the class, there are Case Studies which further enforce what you have learned by having you apply the information from this class to specific programming situations.
APCS System Overview

Features

*Advanced Programming Concepts and Skills* (APCS) focuses on the following World CASE features:

- Data Dictionary Repository
- Project Management (Software Action Request System)
- CASE Profiles
- SAR Log Inquiry
- Creating a Development Environment
- Software Versions Repository
- Data Modeling
- File Design Aid
- Screen Design Aid
- Report Design Aid
- J.D. Edwards Programming Standards
- File Servers and Functional Servers
- User Spaces and User Indexes
- Group Jobs
- Programming Modifications
- Source Debugger
- Programming Impacts from Software Upgrades
Version Control

Objectives

- To create a development environment
- To work with program management
- To create libraries
- To copy data files to the development environment

About Version Control

You use the J.D. Edwards Version Control system to manage the movement of software between various environments, such as ones you have set up for software development, testing, and production.

The Version Control system works with the Software Action Request (SAR) system and the SAR logging system. It performs three general functions:

- Groups source code members (such as RPG and CL programs, and physical and logical tables) and control file data (such as Data Dictionary and menus) together as a project
- Defines a promotion path, which specifies library information about the project’s current environment and the environment to which it will be moved
- Promotes the project from the current environment to the target environment as defined by the promotion path

The following diagram shows how the version control process divides the tasks.

![Diagram showing version control process]

To set up a software development project for development and promotion, you must:

- Create the SARs that you want to promote, and define promotion paths
- Link the project to the SARs that are associated with it, and assign a promotion path to it

All additions or changes you make to programs and control file data are logged in the SAR Log (F9810). Use this log to update the SARs, which are in the Work Order Detail table (F4802).
After you finish developing the software, you promote the software from the Project Elements form to the next environment.

You will work with the following areas:

- Development Environment
- Project Management
- Work with Software Action Request
- Work with Software Versions Repository
- CASE Profiles
- Work with SAR Log
- Work with Promotion Paths and Projects
- Promote a Project
- Promote Project Updates
Version Control Process Flow

- Set Up the SAR System
  - Create record type codes
  - Define record type titles
  - Create SARs

- Activate SAR Logging
  - Activate SAR logging (CASE profile)

- Develop Software
  - Begin to create or change code related to the project using SVR

- Define a Promotion Path
  - Add a promotion path
  - Assign the “From” and “To” environments to source code members and control tables

- Define a Project
  - Add a project
  - Assign promotion paths and SARs to the project

- Update the SARs by Using the SAR Log
  - Update the Work Order Detail table (F4802)

- Validate the promotion paths
  - Validate the promotion paths

- Promote a Project
  - Promote the project to the next environment from the Project Elements form

- Use Version Control for Project Updates
  - Perform version control procedures for a project update
Version Control Menu Overview

The figure below is the Version Control Menu. From this form, you will access the different features of the software development and promotion process.
Development Environment

About a Development Environment

A development environment contains objects and data being tested and edited. It is different from your production environment because it should not contain any live data files.

Rules for Creating Development Environments

You should be aware of the following rules when you create development libraries.

- Do not begin library names with Q, JDF, or JDE because of the upgrade process.
- Create custom libraries for custom modifications.
- Library names should be a maximum of 9 characters in length because of the upgrade process.
- Do not use JDFDATA for your own test data or live data because of the upgrade process.
- Do not include JDFDATA in a live user’s library list.

To create a Development Environment complete the following tasks:

- Create Libraries
- Define Access for a User Profile using J98INITA
- Define Access for a User Profile using J98INIT
- Copy Data to Your Development Environment
J.D. Edwards Libraries

The following libraries are delivered with J.D. Edwards software. They are:

- Source Library (JDFSRC)
- Object Library (JDFOBJ)
- Data Library (JDFDATA)
- Install Library (JDEINSTALL)
- Plans Library (JDFINS)
- Security Library (CLTSEC)

Source Library (JDFSRC)

This is the library that contains source code. Within the JDFSRC library, J.D. Edwards has three multi-member source files. The source files and their contents are illustrated below.

The file JDESRC contains the source code for:

- RPG Programs
- Printer files
- Display files
- CL Programs
- DDS for logical files
- DDS for physical files

The file JDECPY contains the source code for common subroutines

The file F98CRTCMD contains pre-compiler commands

- This is used to compile J.D. Edwards programs
**Object Library (JDFOBJ)**

The object library contains executable objects for your J.D. Edwards software.

- RPG programs
- CL programs
- Display files
- Report files

**Data Library (JDFDATA)**

The data library contains data files for your J.D. Edwards software (files in this library contain test data provided by J.D. Edwards).

**Install Library (JDEINSTAL)**

The install library is used to install programs and software that upgrade J.D. Edwards software.

**Plans Library (JDFINS)**

The library is used to plan how to upgrade J.D. Edwards software.

**Security Library (CLTSEC)**

You can create a security library that is shared across all environments. The benefit of having a security library is that you enter a user profile only once to have access to any environment. The following files must exist in the security library:

- User library list (F0092)
- Library list control (F0093)
- Library list master (F0094)
- User Preference (F00921)

In addition, all logical files associated with the above files must also exist in the security library.
Production and Development Examples

There are many ways to set up a production and development environment. The following are some examples.

Basic Production Environment

<table>
<thead>
<tr>
<th>Library</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>QTEMP</td>
<td>IBM Temporary data files</td>
</tr>
<tr>
<td>CLTOBJ</td>
<td>Client's objects</td>
</tr>
<tr>
<td>JDFOBJ</td>
<td>J.D. Edwards objects</td>
</tr>
<tr>
<td>CLTCOM</td>
<td>Client's common files</td>
</tr>
<tr>
<td>CLTDTA</td>
<td>Client's data files</td>
</tr>
<tr>
<td>CLTSEC</td>
<td>Client's security files</td>
</tr>
<tr>
<td>QGPL</td>
<td>IBM general public library</td>
</tr>
</tbody>
</table>

Basic Development Environment

<table>
<thead>
<tr>
<th>Library</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>QTEMP</td>
<td>IBM Temporary data files</td>
</tr>
<tr>
<td>DEVOBJ</td>
<td>Development objects</td>
</tr>
<tr>
<td>CLTOBJ</td>
<td>Client's objects</td>
</tr>
<tr>
<td>JDFOBJ</td>
<td>J.D. Edwards objects</td>
</tr>
<tr>
<td>DEVCOM</td>
<td>Development common files</td>
</tr>
<tr>
<td>DEVDTA</td>
<td>Development data files</td>
</tr>
<tr>
<td>CLTSEC</td>
<td>Client's security files</td>
</tr>
<tr>
<td>DEVSRC</td>
<td>Development source files</td>
</tr>
<tr>
<td>CLTSRC</td>
<td>Client's source files</td>
</tr>
<tr>
<td>JDFSRC</td>
<td>J.D. Edwards source files</td>
</tr>
<tr>
<td>QGPL</td>
<td>IBM general public library</td>
</tr>
</tbody>
</table>

All modifications and tests are performed in the development environment with the program's object and source residing in DEVOBJ and DEVSRC. After you complete the testing, the program's object is moved from DEVOBJ to CLTOBJ and the source is moved from DEVSRC to CLTSRC. You must create a separate data and common library (DEVDTA and DEVCOM) to ensure that any data changes during testing in the development environment do not affect live data in the production environment.
No Source in Production Environment and a Common Shared Library

<table>
<thead>
<tr>
<th>Library</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>QTEMP</td>
<td>IBM Temporary data files</td>
</tr>
<tr>
<td>CLTOBJ</td>
<td>Client's objects</td>
</tr>
<tr>
<td>JDFOBJ</td>
<td>J.D. Edwards objects</td>
</tr>
<tr>
<td>CLTCOM</td>
<td>Client's common files</td>
</tr>
<tr>
<td>COMMON</td>
<td>Common unchanged files</td>
</tr>
<tr>
<td>CLTDTA</td>
<td>Client's data files</td>
</tr>
<tr>
<td>CLTSEC</td>
<td>Client's security files</td>
</tr>
<tr>
<td>QGPL</td>
<td>IBM general public library</td>
</tr>
</tbody>
</table>

Basic Development Environment with a Shared Common

<table>
<thead>
<tr>
<th>Library</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>QTEMP</td>
<td>IBM Temporary data files</td>
</tr>
<tr>
<td>DEVOBJ</td>
<td>Development objects</td>
</tr>
<tr>
<td>CLTOBJ</td>
<td>Client's objects</td>
</tr>
<tr>
<td>JDFOBJ</td>
<td>J.D. Edwards objects</td>
</tr>
<tr>
<td>DEVCOM</td>
<td>Development common files</td>
</tr>
<tr>
<td>COMMON</td>
<td>Common unchanged files</td>
</tr>
<tr>
<td>DEVDTA</td>
<td>Development data files</td>
</tr>
<tr>
<td>CLTSEC</td>
<td>Client's security files</td>
</tr>
<tr>
<td>DEVSRC</td>
<td>Development source files</td>
</tr>
<tr>
<td>CLTSRC</td>
<td>Client's source files</td>
</tr>
<tr>
<td>JDFSRC</td>
<td>J.D. Edwards source files</td>
</tr>
<tr>
<td>QGPL</td>
<td>IBM general public library</td>
</tr>
</tbody>
</table>

No source libraries exist in the production environment because source code is not necessary to run J.D. Edwards programs. This makes the production environment easier to maintain. The only restriction is that users in the production environment cannot view source code. Another difference is that a third shared common library (COMMON) has been added to the environments. This library contains common files whose data is not changed during the testing process (For example, F98HELP). By having this type of common library not only are the environments easy to maintain, but you save considerable machine resources.
## One Development Source and Object Library

<table>
<thead>
<tr>
<th>Library</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>QTEMP</td>
<td>IBM Temporary data files</td>
</tr>
<tr>
<td>CLTMOD</td>
<td>Client’s source and objects under modification</td>
</tr>
<tr>
<td>CLTOBJ</td>
<td>Client’s objects</td>
</tr>
<tr>
<td>JDFOBJ</td>
<td>J.D. Edwards objects</td>
</tr>
<tr>
<td>DEVCOM</td>
<td>Development common files</td>
</tr>
<tr>
<td>COMMON</td>
<td>Common unchanged files</td>
</tr>
<tr>
<td>DEVDTA</td>
<td>Development data files</td>
</tr>
<tr>
<td>CLTSEC</td>
<td>Client’s security files</td>
</tr>
<tr>
<td>CLTSRC</td>
<td>Client’s source files</td>
</tr>
<tr>
<td>JDFSRC</td>
<td>J.D. Edwards source files</td>
</tr>
<tr>
<td>QGPL</td>
<td>IBM general public library</td>
</tr>
</tbody>
</table>

DEVOBJ and DEVSRC have been combined into one library called CLTMOD. This library contains both source code and compiled objects for programs while they are being modified and tested. After testing, the program objects are moved to CLTOBJ and source code is moved to CLTSRC. The purpose of having one object and source code library like CLTMOD is to simplify the development library list by having one place where all modifications and testing takes place.
Creating Libraries

Create the following libraries:

- Common and data libraries
- Development object library
- Development source library

If you create a common library (DEVCOM), be sure to specify it each time you create the other development libraries. If you do not specify the common library each time, the files are created in your development library.

Your common library should contain files with data that does not change because of development activities (For example, Help Instructions Master). If there is a possibility of the data changing, you should place the file in your test data library (DEVDTA). By doing this, you are insulating the end users from changes in the development environment.

See Appendix A: Common and Production Library Files, for a list of common and production files.
Creating Common and Data Libraries

You will create the libraries that contain common data files (DEVCOM) and test data files (DEVDTA).

To create common and data libraries

From the Data Base Management menu, select Data Libraries. This displays the following form.

```
98312  Create User Data Libraries   Form ID. . . . P98102
      Version. . . . ZJDE0001

Create Production Environment
This job has various options described below. Enter the desired values and press ENTER to continue.
Enter the "FROM" Library where data is to be copied from (e.g JDFFDATA).
Enter the "TO" Production Library where you are creating files (e.g. PRODLIB).
Enter the "TO" Common Library where you are creating common files (e.g. COMMON)
If you do not enter a Common library all common files will be created in the Production Library.
F5=Printer Overrides
```

<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Selection value</td>
<td>The library containing the data to be copied.</td>
</tr>
<tr>
<td></td>
<td>Because you are creating development libraries, type the development library name.</td>
</tr>
<tr>
<td></td>
<td>If you want to create a common library, you must specify the common library name. If you leave this field blank, the system creates the common files in the development Library you specified in the step above.</td>
</tr>
</tbody>
</table>

1. Complete the Create User Data Libraries form
   - Once you correctly complete the form and press Enter, the job (J98102) is submitted to batch.
2. Repeat the above step for each of the development data libraries you have.

The program automatically:

- Creates your libraries
- Creates the physical and logical files that should be maintained in your common library
- Creates the physical and logical files necessary for operations control in your development library
- Creates the physical and logical files for various applications in your development library
- Generates reports to identify all the physical, logical, and join files created and to identify where they were created
- Generates a report to identify all the optional files. The report explains why the files are optional so that you can determine if they should be deleted
Creating a Development Object Library

To create a development object library

From the Data Base Management menu, type the command Create Library (CRTLIB) and press F4.

<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Library Name</td>
<td>Your development object library name.</td>
</tr>
<tr>
<td>Type of Security</td>
<td>*PROD or *TEST</td>
</tr>
<tr>
<td>Text Number</td>
<td>The description of your library</td>
</tr>
</tbody>
</table>
Creating a Development Source Library

To create the development source library (DEVSRCS), you create a source environment and a source physical file. The source physical file is the Program Source File (JDESRC). All J.D. Edwards source programs are located in the JDESRC file.

There are two possible methods to create the JDESRC file. You must determine if you have the J.D. Edwards Program Generator and then choose the appropriate method.

▶ To create a development source library

From the Data Base Management menu, type the command Create Library (CRTLIB) and press F4.

Create Library (CRTLIB)

<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Library Name</td>
<td>Your development object library name.</td>
</tr>
<tr>
<td>Type of Security</td>
<td>*PROD or *TEST</td>
</tr>
<tr>
<td>Text Number</td>
<td>The description of your library</td>
</tr>
</tbody>
</table>
Creating JDESRC with J.D. Edwards Program Generator

When a program is moved into production at J.D. Edwards, the record length is 92 bytes. If you have J.D. Edward’s Program Generator product, the program source file format must be 142 bytes to allow for the Program Generator Serial Number and additional required data.

To create JDESRC with J.D. Edwards Program Generator

1. To copy an existing file with the correct format (F93002), type the Copy File command (CPYF) and press F4

<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>From file</td>
<td>The file and library containing the data to be copied. The file is F93002 and the library can default to *LIBL.</td>
</tr>
<tr>
<td>To file</td>
<td>The name of the source file and your development source library. Generally, the file is JDESRC and the library is DEVSRC.</td>
</tr>
<tr>
<td>From member</td>
<td>The member name that will be the beginning of the copy process. Generally, this value is *FIRST.</td>
</tr>
<tr>
<td>To member or label</td>
<td>The member name that will be the beginning of the receiving process. Generally, this value is *FIRST.</td>
</tr>
</tbody>
</table>
2. To remove the empty member copied from JDERS, type the Remove Member command (RMVM) and press F4.

<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Replace or add records</td>
<td>Specifies whether the records copied should replace or be added to the records in the To file. In this case since the To file does not exist, this value is *NONE.</td>
</tr>
<tr>
<td>Create file</td>
<td>Specifies whether the To file does not exist and needs to be created. This value is *YES.</td>
</tr>
<tr>
<td>Print format</td>
<td>Specifies whether the characters are printed in character or character and hexadecimal format. This option only applies if the To file is *PRINT.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data base file</td>
<td>Type the source file and your development source library that contains the record to be removed. Generally, this file is JDERS and the library is DEVSRC.</td>
</tr>
<tr>
<td>Member</td>
<td>Type the name of the record that is to be removed. This is F93002.</td>
</tr>
</tbody>
</table>
Creating JDESRC Without the Program Generator

If you do not have J.D. Edward’s Program Generator product, the program source file format can remain at 92 bytes, as it is when a program is moved into production at J.D. Edwards. To create the JDESRC file with a 92 byte record format, you can execute the Create Source Physical File command (CRTSRCPF).

To create JDESRC without the Program Generator

1. Type the Create Source Physical File command (CRTSRCPF) and press F4.

<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>File</td>
<td>The source file and your development source library that contains the record to be removed. Generally, this file is JDESRC and the library is DEVSRC.</td>
</tr>
<tr>
<td>Record Length</td>
<td>The number of bytes in the length of the records to be stored in the source file. This value is 92.</td>
</tr>
<tr>
<td>Member, if desired</td>
<td>The member to be added to the source file. Generally, this member is left to *NONE.</td>
</tr>
<tr>
<td>Text Description</td>
<td>The description of your source file.</td>
</tr>
</tbody>
</table>
About User Profiles

You must create profiles that allow users to have access to new environments.

There are two separate methods to defining access to an environment. The method you choose depends upon whether the User Profile accesses J.D. Edwards software using J98INITA or J98INIT.

Defining Access for a User Profile using J98INITA

If you are allowing access to your development environment for a user profile that is using J98INITA, you must define a development environment library list name. In addition, the User Signon List must contain the Development Environment Library List name.

To define access for a user profile using J98INITA

1. Select Library List Revisions, from the Library List Control menu (G944).

<table>
<thead>
<tr>
<th>0094</th>
<th>Library List Revisions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action Code.</td>
<td>. . .</td>
</tr>
<tr>
<td>Library List Name.</td>
<td>. . .</td>
</tr>
<tr>
<td>Description.</td>
<td>. . .</td>
</tr>
<tr>
<td>Menu Program ID.</td>
<td>. . .</td>
</tr>
<tr>
<td>Library List</td>
<td>. . .</td>
</tr>
</tbody>
</table>

F21=Print Library List  F9=Library Search
2. To assign the library list to each user, select User Signon List Revisions from the Library List Control menu (G944).

<table>
<thead>
<tr>
<th>Seq</th>
<th>Library</th>
<th>Sign-on</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.00</td>
<td>PRISTINE</td>
<td>A92</td>
<td>MASTER PRISTINE DATA LIBL</td>
</tr>
<tr>
<td>10.00</td>
<td>A52DEV</td>
<td>A92</td>
<td>A5.2 Case Cert &amp; G Development</td>
</tr>
<tr>
<td>11.00</td>
<td>PGMGEN</td>
<td>A92</td>
<td>Testing A52 Program Generator</td>
</tr>
<tr>
<td>20.00</td>
<td>TECPRG</td>
<td>A92</td>
<td>* List Name Not in Master File</td>
</tr>
<tr>
<td>30.00</td>
<td>TECOV</td>
<td>A92</td>
<td>Testing A52 Tech Foundations</td>
</tr>
<tr>
<td>55.00</td>
<td>KBGCASE</td>
<td>A92</td>
<td>* List Name Not in Master File</td>
</tr>
</tbody>
</table>
Defining Access for a User Profile Using J98INIT

If you are allowing access to your development environment for a user profile that is using J98INIT, you must define a new library list.

To define access for a User Profile using J98INIT

1. Select User Information (User Keys), from the Security Officer menu (G94).

Each user profile for the J.D. Edwards software must have an IBM profile. To define an IBM profile, use the command, Create User Profile (CRTUSRPRF).

Copy Data to Your Development Environment

You can use several methods to copy data to your development environment. The method you choose should depend upon how much data you need to copy to your development environment. You may copy the following:

- Libraries
● Files
● Records
● JDE Record Types

Copying a Library

If you need to duplicate several files in your development environment you can copy one or more libraries.

► To copy a library

1. To display the parameters, type the Copy Library command (CPYLIB) and press F4.

   ![Copy Library (CPYLIB)]

   Type choices, press Enter.
   
   **Copy Library (CPYLIB)**
   
<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Existing library</td>
<td>The library to be copied in your Production Environment.</td>
</tr>
<tr>
<td>New library</td>
<td>The new library that will be used in your Development Environment</td>
</tr>
<tr>
<td>Create library</td>
<td>Specifies whether the New Library does not exist and needs to be created.</td>
</tr>
</tbody>
</table>

If you use CPYLIB, you must rebuild your access paths. Any files that are in use are not copied.
Copying a File

If you need to copy specific files from a library in your production environment to a library in your development environment, you use the J.D. Edwards copy file utility.

To copy a file

1. From the Data Base Management menu (G9645), select Copy Data files.

   Enter the system code, the library to copy the data from, and the library to copy the data to.

2. Then type a 1 next to the files you wish to copy.

   All records in those specified files will be copied.

When using this utility, be sure to copy all related files.

NOTE: When using this utility, be sure to copy all related files.
Copying a Record

If you wish to copy a file with only selected records, use the Copy File command (CPYF).

To copy a record

1. Type the Copy File command (CPYF).

2. Press F10 to display additional parameters.

<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>From file</td>
<td>The file and library containing the data to be copied.</td>
</tr>
<tr>
<td>To file</td>
<td>The name of the file and your development library the data will be copied to.</td>
</tr>
<tr>
<td>From member</td>
<td>The member name that will be the beginning of the copy process.</td>
</tr>
<tr>
<td>To member or label</td>
<td>The member name that will be the beginning of the receiving process.</td>
</tr>
<tr>
<td>Replace or add records</td>
<td>Specifies whether the records copied should replace or be added to the records in the To file.</td>
</tr>
<tr>
<td>Create file</td>
<td>Specifies whether the To file does not exist and needs to be created.</td>
</tr>
</tbody>
</table>
### Field | Explanation
--- | ---
Print format | Specifies whether the characters are printed in character or character and hexadecimal format. This option only applies if the To file is *PRINT.
Copy from record number | Specifies the record number from which to start the copy.

3. Scroll up and enter the record number of the record to which you wish to copy.

- The Copy to record number is the field in which you specify the record number of the last record to be copied.

---

**Copy File (CPYF)**

Type choices, press Enter.

<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Copy to record number</td>
<td>Number, *END</td>
</tr>
<tr>
<td>Copy from record key: Number of key fields</td>
<td>Number, *NONE, *BLDKEY</td>
</tr>
<tr>
<td>Key value</td>
<td></td>
</tr>
</tbody>
</table>

+ for more values

More...

F3=Exit F4=Prompt F5=Refresh F12=Cancel F13=How to use this display F24=More keys
Copying J.D. Edwards Record Types

You can copy any of the following record types:

- Vocabulary Overrides
- Data Dictionary
- Software Inventory Revisions
- User Defined Code
- DREAM Writer
- Menu
- Generic Rate/Msg

To copy a J.D. Edwards record type

From the Developer’s Workbench menu (G9362) or Repository Services choose Copy DD,VO,DW,UDC,SVR,Menus.

<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>From Library</td>
<td>The library containing the data to be copied.</td>
</tr>
<tr>
<td>To Library</td>
<td>The library in your Development Environment to receive the data.</td>
</tr>
<tr>
<td>Field</td>
<td>Explanation</td>
</tr>
<tr>
<td>-----------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Dictionary Item</td>
<td>The RPG data name. This data field has been set up as a 10-byte field for future use. Currently, it is restricted to 4 bytes so that, when preceded by a 2-byte file prefix, the RPG data name does not exceed 6 bytes. Within the Data Dictionary, all data items are referenced by this 4-byte data name. As they are used in database tables, a 2-character prefix is added to create unique data names in each table specification (DDS). Special characters are not allowed as part of the data item name, with the exception of #, @, $. You can create protected data names by using $xxx and @xxx, where you define xxx. Messages can contain up to 10 characters. Types of messages are further defined by glossary group.</td>
</tr>
<tr>
<td>Vocabulary Overrides</td>
<td>The name of the screen or report record to be copied. All records for soft coding will be copied.</td>
</tr>
<tr>
<td>DREAM Writer Form</td>
<td>The name of the DREAM Writer Form ID to be copied. All versions of the specified form will be copied.</td>
</tr>
<tr>
<td>User Def Codes Sys</td>
<td>The system code and type of the table to be copied. All values for the specified table will be copied.</td>
</tr>
<tr>
<td>Software Versions Rep</td>
<td>The record of the Software Versions Repository member to be copied.</td>
</tr>
<tr>
<td>Menu Identification</td>
<td>The menu ID and the display language of the record to be copied.</td>
</tr>
</tbody>
</table>

You can enter and copy only one item at a time. If the item exists in the To Library, it is replaced.
Project Management

About Project Management

To manage projects you may use Work Order Processing. You will perform the following tasks:

- Understand Work Order Processing
- Create Work Orders
- Access the Scheduling Workbench
- Add or change record types

Understanding Work Order Processing

The Software Action Request System (SAR) is shipped to clients under the name of Work Order Processing.

The Work Order Processingsystem allows you to:

- Create and classify work orders with simple budgets or estimates
- Schedule and expedite work orders
- Perform cost accounting by specific work orders or family of work orders.

Unlike jobs that are often preplanned and thoroughly budgeted, work orders are often completed without the prior knowledge of the accounting department. Work orders are typically spontaneous and of short duration.

If you purchased system 48 (Work Order Processing), you have all of the programs associated with Work Orders (SARs). If you have not purchased the Work Order Processing system, you have only the programs from the Work Order Processing system that are defined as being part of the General Back Office System (00).
Creating Work Orders

There are only three required fields when creating a new work order:

- **Work Order Number** If you do not provide a work order number, the system assigns one automatically.
- **Description (short)**
- **Charge to Business Unit**

**To create work orders**

From the Simple Project Management menu (G4812), select Single Task Details.

```
48014  Single Task Details
Action Code... I  Parent W.O. No
Description... APCS Class  W.O.Number...     289
Status Comment.. Student SAR  Charge to BU...   1001
Search X-Ref...  Est. Hours... 40  Cost Code...
Est. Amount...  1,500  Start Date... 01/03/94
Phase... 55  Planned Comp... 12/31/94
Type... 2  Reserved for Clients
Priority... H  Completed...
Tax Expl Code... 1001  Status...
Tax Rate/Area... 10  10
Subledger Inact... 6001  Transaction... 11/12/93
Customer No... Edwards, J.D.  Date Assigned...
Manager... Allen, Ray

Description
SAR setup for work to be performed during the Advanced Programming Concepts and Skills class
Engine REQ125-796

Opt: 1=Insert 9=Del F5=More Desc F8=Cat Codes F21=Print F24=More Keys
```

**What You Should Know About**

**Accessing the W.O. Detail form**

To access the W.O. Detail form, choose More Description (F5).

**Searching for address numbers**

To search for address numbers for the Customer Number and Manager fields, choose More Keys (F24), then Exit to Name Search.
<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
</table>
| Parent W.O. No   | This is the parent work order number. You can use this number to:  
|                  | 1. Enter default values for newly added work orders, for example, Type, Priority, Status, or Manager.  
|                  | 2. Group work orders for project setup and reporting                                                                                                                                             |
|                  | **Form-specific information**                                                                                                                                                                          |
|                  | For Work Orders                                                                                                                                                                                          |
|                  | When you create a new work order using a parent work order, the system uses information from the parent work order as default values for the new work order. If you leave any of these information fields blank when you create the new work order, the system uses the values from the parent work order. The only information that the system does not use as default values from the parent work order includes:  
|                  |   • Description  
|                  |   • Extended description  
|                  |   • Tax code  
|                  |   • Tax rate and area  
|                  |   • Date completed                                                                                                                                                                                   |
| Action Code      | A code that indicates the activity you want to perform. Valid codes are:  
|                  | A   Add new record  
|                  | C   Change existing record  
|                  | D   Delete existing record  
|                  | I   Inquire on existing record  
|                  | .   End of program or function  
<p>|                  | space Clear the form                                                                                                                                                                                  |
|                  | If you enter a code that is not active, the system highlights the code and no action occurs.                                                                                                          |
|                  | <strong>NOTE:</strong> Depending on how your company has set up action code security, you might not be authorized to use all action codes.                                                                        |
| W.O.Number       | The number that identifies an original document. This can be a voucher, an order number, an invoice, unapplied cash, a journal entry number, and so on.                                                   |
| Description      | A brief description of an item, a remark, or an explanation.                                                                                                                                          |
| Status Comment   | A brief description to explain the status of the work order.                                                                                                                                           |</p>
<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Charge to BU</td>
<td>An alphanumeric field that identifies a separate entity within a business for which you want to track costs. For example, a business unit might be a warehouse location, job, project, work center, or branch/plant. You can assign a business unit to a voucher, invoice, fixed asset, and so on, for purposes of responsibility reporting. For example, the system provides reports of open accounts payable and accounts receivable by business units to track equipment by responsible department. Security for this field can prevent you from locating business units for which you have no authority. Note: The system uses this value for Journal Entries if you do not enter a value in the AAI table.</td>
</tr>
<tr>
<td>Search X-Ref</td>
<td>An alphanumeric value used as a cross-reference or secondary reference number. Typically, this is the customer number, supplier number, or job number.</td>
</tr>
<tr>
<td>Cost Code</td>
<td>A subdivision of an object account. Subsidiary accounts include more detailed records of the accounting activity for an object account.</td>
</tr>
<tr>
<td>Est. Hours</td>
<td>The estimated hours that are budgeted for this work order.</td>
</tr>
<tr>
<td>Est. Amount</td>
<td>The estimated dollar amount that is budgeted for this work order.</td>
</tr>
<tr>
<td>Start Date</td>
<td>This is a start date that you can enter, or an automatic start date which the planning system calculates using a backscheduling routine. The routine starts with the required date and offsets the total leadtime to calculate the appropriate start date. Will default from system date or you can enter a date.</td>
</tr>
<tr>
<td>Planned Comp</td>
<td>The date the work order is planned to be completed.</td>
</tr>
<tr>
<td>Phase</td>
<td>A user defined code (00/W1) that indicates the current stage or phase of development for a work order. You can assign a work order to only one phase code at a time. NOTE: Certain forms contain a processing option that allows you to enter a default value for this field. If you enter a default value on a form for which you have set this processing option, the system displays the value in the appropriate fields on any work orders that you create. The system also displays the value on the Project Setup form. You can either accept or override the default value.</td>
</tr>
<tr>
<td>Completed</td>
<td>The date the work order or engineering change order is completed or canceled.</td>
</tr>
<tr>
<td>Field</td>
<td>Explanation</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Type</td>
<td>A user defined code (00/TY) that indicates the type classification of a work order or engineering change order. You can use work order type as a selection criteria for work order approvals.</td>
</tr>
<tr>
<td>Priority</td>
<td>A user defined code (system 00, type PR) that indicates the relative priority of a work order or engineering change order in relation to other orders. A processing option for some forms lets you enter a default value for this field. The value then displays automatically in the appropriate fields on any work order you create on those forms and on the Project Setup form. You can either accept or override the default value.</td>
</tr>
<tr>
<td>Status</td>
<td>A user defined code (00/SS) that describes the status of a work order or engineering change order. Any status change from 90 thru 99 automatically updates the date completed.</td>
</tr>
<tr>
<td>Customer No</td>
<td>A number that identifies an entry in the Address Book system. Use this number to identify employees, applicants, participants, customers, suppliers, tenants, and any other Address Book members.</td>
</tr>
<tr>
<td>Manager</td>
<td>The address book number of a manager or planner. <strong>NOTE:</strong> A processing option for some forms lets you enter a default value for this field based on values for Category Codes 1 (Phase), 2, and 3. Set up the default values on the Default Managers and Supervisors form. After you set up the default values and the processing option, the information displays automatically on any work orders you create if the category code criterion is met. (You can either accept or override the default value.)</td>
</tr>
<tr>
<td>Transaction</td>
<td>The date that an order was entered into the system. This date determines which effective level that the system uses for inventory pricing.</td>
</tr>
<tr>
<td>Date Assigned</td>
<td>The date the person responsible for the work order receives the work order.</td>
</tr>
<tr>
<td>Tax Expl Code</td>
<td>A user defined code (00/EX) that controls how a tax is assessed and distributed to the general ledger revenue and expense accounts. You assign this code to a customer or supplier to set up a default code for their transactions. Do not confuse this with the taxable, non-taxable code. A single invoice can have both taxable and non-taxable items. The entire invoice, however, must have one tax explanation code.</td>
</tr>
</tbody>
</table>
### Field

<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tax Rate/Area</td>
<td>A code that identifies a tax or geographic area that has common tax rates and tax distribution. The tax rate/area must be defined to include the tax authorities (for example, state, county, city, rapid transit district, or province), and their rates. To be valid, a code must be set up in the Tax Rate/Area table (F4008). Typically, U.S. sales and use taxes require multiple tax authorities per tax rate/area, whereas VAT requires only one simple rate. The system uses this code to properly calculate the tax amount.</td>
</tr>
<tr>
<td>Subledger Inact</td>
<td>A code that indicates whether a specific subledger is active or inactive. Any value other than blank indicates that a subledger is inactive. Examples are jobs that are closed, employees that have been terminated, or assets that have been disposed. If a subledger becomes active again, set this field back to blank. If you want to use subledger information in the tables for reports but want to prevent transactions from posting to the master record, enter a value other than blank in this field.</td>
</tr>
</tbody>
</table>

### Processing Options

There are processing options associated with the Single Task Details program that allow you to default the value for the Type, Priority, Status, Phase, Category Code 2, Category Code 3, and Manager fields. To see the processing options, type the selection number for Single Task Details and press F18.

### Function Keys from Single Task Details

**F5 - Detailed Specifications**

F5 - Allows you to enter additional detailed information about your work order. Each detail screen is based on Record Type. Record Type A provides room for you to enter more description. You can customize other Record Types to fit your requirements. The steps to add and change Record Types are described later in this chapter.
F8 - Category Codes

F8 - Allows you to update other work order values.

4802
W.O. Detail Entry
Record Type. . .  A

Full Description of Request

Action Code. . .  C
Order Number :  289  Tech for Programmers Class

Description
Option
SAR setup for work to be performed during the Advanced
Programming Concepts and Skills class.

Opt: 1=Insert  9=Delete     F5=Re-Fresh  F8=Record Types  F24=More Keys

Option         Description
1 - Insert     Insert a blank line for additional text.
9 - Delete     Delete a line of text
### Field | Explanation
--- | ---
W.O. Number | The work order identification number. This value defaults from the Single Task Details.
W.O. Flash Message | A highlighted message that will be attached to the work order.

*Form-specific information*

The flash message appears as a highlighted message on Backlog Management, replacing the work order description.

Phase | A user defined code describing a stage or category in the development of a project. This value defaults from the Single Task Details.

Category 02 | Category Codes that are user defined values associated with the work order.

Originator | The address number of the person who entered the work order.

Must be a valid number in the Address Book Master file (F0101).

Supervisor | The address number of the work order supervisor.

Must be a valid number in the Address Book Master table (F0101).

Std. Desc | A user defined code describing instructional information.

Must be a valid number in the Address Book Master file (F0101).

*Form-specific information*

For Equipment/Plant Maintenance users:

You can use this code to assign narrative text for a standard procedure. The information appears on the Item PM schedule and the work order routing.

Search X–Ref | Any number or characters that will be used to cross–reference work orders. This value will default from the Single Task Defaults screen.

---

**F9 - Name Search**

F9 - Allows you to search for a specific address book number.

**F15 - Work Order Search Window**

F15 - Allows you to search for work order descriptions. It will only return the description.
48014 Single Task Details

Action Code . . . I
Description . . . APCS Class
Status Comment . . Student SAR
Search X-Ref . . . 40
Est. Hours . . . 40
Est. Amount . . 1,500
Phase . . . . . 55
Type . . . . . A
Tax Expl Code . . 4802T1
Subledger Inact. . . Order Number 289
Manager . . . . 6001

Description
SAR setup for work to be
Programming Concepts and
Engine REQ125–796

Opt: 4=Select  F3=Return   F24=More Keys
Opt:  1=Insert  9=Del  F5=More Desc  F8=Cat Codes  F21=Print  F24=More Keys

F21 - Print Work Order

F21 - Allows you to print the work order, including all of the associated record types.
Accessing the Scheduling Workbench

The Scheduling Workbench program allows you to review and update work orders. You can retrieve information about work orders in multiple ways. After retrieving the work orders that meet your search criteria, you can update selected fields in those work orders directly from the Scheduling Workbench form.

To access the Scheduling Workbench

From the Simple Project Management menu, select Scheduling Workbench

<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category Codes</td>
<td>Any number or characters that will be used to cross-reference work orders. This value will default from the Single Task Defaults screen.</td>
</tr>
<tr>
<td>Job or BU</td>
<td>The business unit that is responsible for charges incurred. Must be a valid business unit setup in the Business Unit Master File (F0006).</td>
</tr>
<tr>
<td>Originator</td>
<td>The address number of the person who entered the work order. Must be a valid number in the Address Book Master file (F0101).</td>
</tr>
<tr>
<td>Field</td>
<td>Explanation</td>
</tr>
<tr>
<td>-------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Customer</td>
<td>The Address Number of the customer. Must be a valid number in the Address Book Master file (F0101).</td>
</tr>
<tr>
<td>Manager</td>
<td>The Address Number of the manager in charge of the work order. Must be a valid number in the Address Book Master File (F0101).</td>
</tr>
<tr>
<td>Supervisor</td>
<td>The address number of the work order supervisor. Must be a valid number in the Address Book Master table (F0101).</td>
</tr>
<tr>
<td>Parent W.O. No</td>
<td>Through parent work order number, you can group work orders together based on one parent work order, such as the installation of a computer and its associated electrical wiring, which may involve more than one customer or manager.</td>
</tr>
<tr>
<td></td>
<td>Form-specific information</td>
</tr>
<tr>
<td></td>
<td>The parent work order number which groups work orders together in a &quot;family&quot;.</td>
</tr>
<tr>
<td>Model</td>
<td>Determines whether model work orders will be displayed on the screen.</td>
</tr>
<tr>
<td>M</td>
<td>Determines whether model work orders will be displayed on the screen.</td>
</tr>
<tr>
<td>Search X-Ref</td>
<td>Any number or characters that will be used to cross-reference work orders. This value will default from the Single Task Defaults screen.</td>
</tr>
<tr>
<td>Cost Code</td>
<td>The subsidiary account responsible for incurred charges.</td>
</tr>
<tr>
<td>Number</td>
<td>The work order identification number. This value defaults from the Single Task Details.</td>
</tr>
<tr>
<td>Description</td>
<td>Describes the function or option exit. Cannot exceed 40 characters.</td>
</tr>
<tr>
<td></td>
<td>Form-specific information</td>
</tr>
<tr>
<td></td>
<td>The name or a brief description of the work order.</td>
</tr>
<tr>
<td>X-Ref No</td>
<td>Any number or characters that will be used to cross-reference work orders. This value will default from the Single Task Defaults screen.</td>
</tr>
<tr>
<td>Status</td>
<td>A user defined code used to describe the current status of the work order; for example, planned, started, or completed.</td>
</tr>
<tr>
<td>Status Comment</td>
<td>This line allows status comments or further description of the work.</td>
</tr>
</tbody>
</table>
F4 - More Detail

F4 - Displays additional information concerning each work order that is hidden in the Fold Area.

<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>User defined code describing the work order type.</td>
</tr>
<tr>
<td>Priority P</td>
<td>A user defined code used to assign the priority of the work order; for example, high, medium, or low.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action Code</td>
<td>Subcontractors</td>
</tr>
<tr>
<td>Job or BU</td>
<td>Work Orders in Progress</td>
</tr>
<tr>
<td>Originator</td>
<td></td>
</tr>
<tr>
<td>Customer Number</td>
<td></td>
</tr>
<tr>
<td>Manager</td>
<td></td>
</tr>
<tr>
<td>Supervisor</td>
<td>Parent W.O. No</td>
</tr>
<tr>
<td>Status</td>
<td>Thru Search X-Ref</td>
</tr>
<tr>
<td>W.O. Date Range</td>
<td>Thru Search X-Ref</td>
</tr>
<tr>
<td>Compl. Date Range</td>
<td>Type Model.</td>
</tr>
<tr>
<td>Supervisor</td>
<td>Cost Code.</td>
</tr>
<tr>
<td>Search X-Ref</td>
<td>Prior.</td>
</tr>
<tr>
<td>Category Codes</td>
<td>Phs. CC2. CC3. CC4. CC5.</td>
</tr>
<tr>
<td>O Number</td>
<td>Description</td>
</tr>
<tr>
<td>Description</td>
<td>X-Ref No.</td>
</tr>
<tr>
<td>Start Date</td>
<td>Est. Hours.</td>
</tr>
<tr>
<td>Planned Comp</td>
<td>Hours Scheduled.</td>
</tr>
<tr>
<td>W.O. Flash Message</td>
<td>W.O. Date</td>
</tr>
<tr>
<td>W.O. Date</td>
<td></td>
</tr>
</tbody>
</table>

Field       | Explanation                                      |
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Planned Comp</td>
<td>The date the work is scheduled to be completed.</td>
</tr>
<tr>
<td>Hours Scheduled</td>
<td>The hours of work that has been scheduled.</td>
</tr>
<tr>
<td>Est. Hours</td>
<td>Total number of hours estimated for the work order.</td>
</tr>
<tr>
<td>Start Date Range</td>
<td>The initial date the work is scheduled to begin.</td>
</tr>
<tr>
<td>W.O. Flash Message</td>
<td>A highlighted message that will be attached to the work order.</td>
</tr>
<tr>
<td>W.O. Date</td>
<td>The date the work order was entered. Must be a valid number in the Address Book Master File (F0101).</td>
</tr>
</tbody>
</table>
Selection Exits from the Scheduling Workbench

Selection 1 - Work Order Entry

- Takes you to the Work Order Entry screen and automatically inquires on the selected work order

Processing Options

There are processing options associated with the Scheduling Workbench program that allow you to default a Work Order Status Range and a Work Order Type. In addition, you can call either Project Task Details (P48014) or the Equipment Work Orders (P48011) when the W.O. Entry option is selected. Be aware that Equipment Work Orders (P48011) is part of the Work Order Processing system (48). To see the processing options, type the selection number for Scheduling Workbench and press F18.
Adding Record Types

To add record types

1. From the Misc Additional Features menu (G4841), select Detail Spec. Types.

   00051  Detail Spec. Types
       Install System Code . . . 00
       User Defined Codes . . . RT
       Action Code . . . I
       Skip To Code . . .
       Work Order Detail Specs.

   01 Character Description
   A  Full Description of Request
   B  Final Disposition Remarks
   C  Tool and Equipment Instruct.
   D  Safety Provisions
   E  Plan and Drawing Reference
   F  Equipment Down Time

   F5=Code Types  F14=Memo  F15=Where Used  F18=Language  F21=Print  F24=More Keys

2. Add your specified record type and description to the table.
# Changing Record Types

You can change the format of your record type.

**To change record types**

1. From the Misc Additional Features menu (G4841), select Detail Spec. Over Titles.

<table>
<thead>
<tr>
<th>48002</th>
<th>Detail Spec. Over Titles</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action Code. . .</td>
<td>F</td>
</tr>
<tr>
<td>Record Type. . .</td>
<td>F</td>
</tr>
</tbody>
</table>

**Sub-Title 1**  
**Sub-Title 2**  
**Sub-Title 3**  
| Equipment Number | Production Time Out | Production Time In |

F24=More Keys

2. Enter the heading text of each column you wish to add to the format of your Record Type.
   - Work Order (SAR) file is F4801
   - Detail Record Type file is F4802
   - Method of tracking programming projects

**See Also**

- This is a brief overview of the Work Order Processing system. For more information, consult *Work Orders*.

**Exercises**

See the exercises for this chapter.
Work with Software Action Requests

About SAR System Setup

To set up a project, you must assign SARs and promotion paths to it. You create the SARs and define promotion paths first because the version control process uses the definitions.

After you set up your SAR system, you can develop the software. The SAR logging program keeps track of your changes as you have specified. While you develop the software, you can also define promotion paths and projects, and attach SARs to projects.

After you finish developing the software, you must update the SARs by using the SAR log before you promote the SAR.

Complete the following tasks:

- Create record type codes
- Define record type titles
Before You Begin

The SAR system uses the Work Order files (F4801 and F4802). If your production environment uses these files, and if the F4802 file has different record types than what version control needs, define a separate library that contains these files for version control purposes only.

See Also

- Defining a Promotion Path

Creating Record Type Codes

The Work Order Instructions table (F4802) has an essential role in the version control process. It identifies and captures, for promotion purposes, all the source code members and control table data associated with a SAR. The Version Control system assigns a record type code to each source code member or control table data item, which classifies it for promotion. You must create record type codes that your Work Order Instructions table does not have currently.

To create record type codes

1. From the Version Control menu (G9261), choose Record Type Codes.
2. On User Defined Code Revisions enter the following character codes and descriptions:

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Original Request</td>
</tr>
</tbody>
</table>

F5=Code Types  F14=Memo  F15=Where Used  F21=Print  F18=Translate  F24=More
<table>
<thead>
<tr>
<th>Character Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Original Request</td>
</tr>
<tr>
<td>C</td>
<td>Members Affected</td>
</tr>
<tr>
<td>D</td>
<td>Menu Modifications</td>
</tr>
<tr>
<td>E</td>
<td>Automatic Accounting Instructions</td>
</tr>
<tr>
<td>F</td>
<td>Software Inventory Record Updates</td>
</tr>
<tr>
<td>G</td>
<td>Processing Options/DREAM Writer</td>
</tr>
<tr>
<td>H</td>
<td>Vocabulary Override Changes</td>
</tr>
<tr>
<td>I</td>
<td>Database Changes</td>
</tr>
<tr>
<td>J</td>
<td>Constants Data File Changes</td>
</tr>
<tr>
<td>K</td>
<td>User Defined Code Changes</td>
</tr>
<tr>
<td>M</td>
<td>Connected SAR Numbers</td>
</tr>
<tr>
<td>N</td>
<td>Generic Rate/Message Type Changes</td>
</tr>
<tr>
<td>O</td>
<td>Connected SAR Numbers</td>
</tr>
<tr>
<td>Q</td>
<td>Generic Rate/Message Type Changes</td>
</tr>
<tr>
<td>S</td>
<td>Status History</td>
</tr>
<tr>
<td>U</td>
<td>Post-Installation Instructions</td>
</tr>
<tr>
<td>W</td>
<td>Pre-Compiler Commands</td>
</tr>
<tr>
<td>Z</td>
<td>First Included in PTF</td>
</tr>
<tr>
<td>3</td>
<td>Next Number Changes</td>
</tr>
</tbody>
</table>
Defining Record Type Titles

For each record type code you create, you must also define record type titles, which appear as column headings on the W.O. Detail Entry form.

Before You Begin

- Create record type codes before you define record type titles. See Creating Record Type Codes.

To define record type titles

From the Version Control menu (G9261), choose Record Type Titles.

On Record Type Titles

<table>
<thead>
<tr>
<th>Menu Modifications</th>
</tr>
</thead>
<tbody>
<tr>
<td>48002 Record Type Titles</td>
</tr>
<tr>
<td>Action Code. . . A</td>
</tr>
<tr>
<td>Record Type. . . B</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Sub-Title 1</th>
<th>Sub-Title 2</th>
<th>Sub-Title 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Option</td>
<td>Job To</td>
</tr>
<tr>
<td>Execute</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

F24=More Keys
For each record type you created, complete the following fields with the information in the chart that follows:

<table>
<thead>
<tr>
<th>TITLE</th>
<th>SUB-TITLE 1</th>
<th>SUB-TITLE 2</th>
<th>SUB-TITLE 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C</td>
<td><strong>Member</strong></td>
<td><strong>Name</strong></td>
<td><strong>Object</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Source</strong></td>
<td><strong>Library</strong></td>
<td><strong>Library</strong></td>
</tr>
<tr>
<td>D</td>
<td><strong>Menu</strong></td>
<td><strong>Option</strong></td>
<td><strong>Job_To</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Name</strong></td>
<td><strong>Number</strong></td>
<td><strong>Execute</strong></td>
</tr>
<tr>
<td>E</td>
<td><strong>AAI</strong></td>
<td><strong>Company</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>No</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>F</td>
<td>CL_Program</td>
<td><strong>Program</strong></td>
<td>Video/Rpt_</td>
</tr>
<tr>
<td>G</td>
<td><strong>Form</strong></td>
<td><strong>Version</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>ID</strong></td>
<td><strong>No</strong></td>
<td></td>
</tr>
<tr>
<td>H</td>
<td><strong>Scr/Rpt</strong></td>
<td><strong>Name</strong></td>
<td></td>
</tr>
<tr>
<td>I</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>J</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>K</td>
<td><strong>Help</strong></td>
<td><strong>Start</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Stop</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M</td>
<td><strong>Sys</strong></td>
<td><strong>DTAI</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Code</strong></td>
<td><strong>Name</strong></td>
<td></td>
</tr>
<tr>
<td>N</td>
<td><strong>Sys</strong></td>
<td><strong>Rec</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Code</strong></td>
<td><strong>Type</strong></td>
<td></td>
</tr>
<tr>
<td>O</td>
<td><strong>SAR</strong></td>
<td><strong>SAR</strong></td>
<td><strong>SAR</strong></td>
</tr>
<tr>
<td></td>
<td><strong>No</strong></td>
<td><strong>No</strong></td>
<td><strong>No</strong></td>
</tr>
<tr>
<td>Q</td>
<td><strong>Sys</strong></td>
<td><strong>Rec</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Code</strong></td>
<td><strong>Type</strong></td>
<td></td>
</tr>
<tr>
<td>S</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>U</td>
<td>Reference_</td>
<td>Attachment</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ID/Code__</td>
<td>Needed-Y/N</td>
<td></td>
</tr>
<tr>
<td>W</td>
<td><strong>Program</strong></td>
<td><strong>Name</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Z</td>
<td><strong>Release</strong></td>
<td><strong>PTF</strong></td>
<td><strong>Date</strong></td>
</tr>
<tr>
<td></td>
<td><strong>ID</strong></td>
<td><strong>Number</strong></td>
<td><strong>Included</strong></td>
</tr>
<tr>
<td>3</td>
<td>System_code</td>
<td>Line__</td>
<td>Action__</td>
</tr>
<tr>
<td></td>
<td>Code__</td>
<td>Number__</td>
<td>Code__</td>
</tr>
</tbody>
</table>
What You Should Know About

Verifying the record type titles

After you define the record type titles, you can view them to verify their accuracy. On Single Task Details, choose More Description. On W.O. Detail Entry, locate a record type you want to view by using the Record Type field.

To access Single Task Details, see *Creating SARs*. 
The Software Versions Repository (SVR) consists of the following master directories.

- **F9801 Software Versions Repository Master**
  - A master directory of all files, programs, screens, reports, and copy modules.

- **F9802 Software Versions Repository Master**
  - Stores the member locations for each member master record.

**Working with Software Versions Repository (SVR)**

The Software Versions Repository indicates in what environments a requested member is located and whether the environment is a production or development environment. The file is used extensively for documentation and plays an important role in J.D. Edwards Design and Development tools.

The Software Versions Repository is the natural starting point for all programming and software inquiry functions. It provides exits to the following features:

- Source Entry Utility (SEU)
- SAR Detail Entry
- Screen Design Aid (SDA)
- Report Design Aid (RDA)
- File Design Aid (FDA)
- The Program Generator
- Precompiler Commands
Repository Services
  - Data Dictionary
  - Menus
  - Vocabulary Overrides
  - Function Key Definitions
  - DREAM Writer Versions
  - Processing Options
  - User Defined Codes
  - Edit System Helps
  - CASE Profiles
  - SAR Log Inquiry
  - Copy DD, VO, DW, UDC, SVR, Menus

Optional Files Feature
- Programmer Checklists
- Where Used Facility
- Flowchart Programs/Illustrate File Models
- Source Modifications Editor

In addition, it provides access to the following functions:

- Copy Source
- Print Source
- Submit Creation of Object
- Generate Program Source and Help
- Edit Help Instructions
- Delete Source
- Print Help Instructions
Accessing the Software Versions Repository

The Software Versions Repository serves as the front-end for all J.D. Edwards design aids and programming utilities. You can also utilize this form as your own inventory file.

To access the Software Versions Repository

From the Computer Assisted Design menu, choose Software Versions Repository.
The top portion of the form identifies the member and its attributes. This information is stored in the Software Versions Repository master file (F9801).

**Member Identifiers**

The first two fields identify the member.

The Member ID and Description fields identify the SVR member.

<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Member ID</td>
<td>The record of the Software Versions Repository member to be copied.</td>
</tr>
<tr>
<td></td>
<td><strong>Form-specific information</strong></td>
</tr>
<tr>
<td></td>
<td>The source file containing the source member. At J. D. Edwards, three</td>
</tr>
<tr>
<td></td>
<td>source files reside inside of the JDFSRC library.</td>
</tr>
<tr>
<td></td>
<td>They are:</td>
</tr>
<tr>
<td></td>
<td>• JDECPY for copy modules</td>
</tr>
<tr>
<td></td>
<td>• JDESRC for other source code</td>
</tr>
<tr>
<td></td>
<td>• F98CRCTCMD for precompiler commands</td>
</tr>
<tr>
<td>Description</td>
<td>Identifying information of the member, such as Trial Balance by Business</td>
</tr>
<tr>
<td></td>
<td>Unit. Associated programs, screens, and reports should share the same</td>
</tr>
<tr>
<td></td>
<td>description.</td>
</tr>
<tr>
<td></td>
<td>The description associated with each member is used to further identify</td>
</tr>
<tr>
<td></td>
<td>the purpose of the member.</td>
</tr>
<tr>
<td></td>
<td>• Physical files should have a description that explains the purpose of the</td>
</tr>
<tr>
<td></td>
<td>file.</td>
</tr>
<tr>
<td></td>
<td>• Screens, reports, and CL programs should have the same description as the</td>
</tr>
<tr>
<td></td>
<td>associated RPG program.</td>
</tr>
<tr>
<td></td>
<td>• Logical files should be designated as follows: LF – filename, fname,</td>
</tr>
<tr>
<td></td>
<td>fname: where fname is a key field.</td>
</tr>
<tr>
<td></td>
<td>• Join files should be designated as follows: JF – filename/filename/filename</td>
</tr>
<tr>
<td></td>
<td>– fname, fname, fname, fname: where the filename is a file over which the</td>
</tr>
<tr>
<td></td>
<td>join is built and fname is the key field joining the files.</td>
</tr>
<tr>
<td></td>
<td>• Work files should be designated as follows: WF – filename; where filename</td>
</tr>
<tr>
<td></td>
<td>is the file that the work file accesses.</td>
</tr>
<tr>
<td></td>
<td>• Copy modules carry their own unique descriptions.</td>
</tr>
<tr>
<td></td>
<td>• File Server programs should be designated as follows: File Server –</td>
</tr>
<tr>
<td></td>
<td>filename; where filename is the file being served.</td>
</tr>
</tbody>
</table>
Type, Use, and Associated Systems

The following fields identify the associated systems, along with their type and use.

The following SVR fields identify the associated systems, along with their type and use: Function Code, Function Use, System Code, and Reporting System.

<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Function Code</td>
<td>Designates the object type such as display file, physical and logical files. Use F1 in the field to view the available types.</td>
</tr>
<tr>
<td>Function Use</td>
<td>Displays the files that either match or have a function use less than the function use you specify. Form-specific information</td>
</tr>
<tr>
<td></td>
<td>Indicates how the member is being used.</td>
</tr>
<tr>
<td>System Code</td>
<td>The system code and type of the table to be copied. All values for the specified table will be copied. Form-specific information</td>
</tr>
<tr>
<td></td>
<td>Designates the system number associated with the member. The configuration of installation media and the install process itself are driven by this install system code. Use F1 in the field to view valid codes.</td>
</tr>
<tr>
<td>Reporting System</td>
<td>Designates the system number for reporting purposes. This rarely differs from the Install System. Exceptions occur for data files used by more than one system</td>
</tr>
</tbody>
</table>
Member Relationship and Compiling Information

The following fields identify the logical grouping of members and information used in the compile process.

The following SVR fields identify the logical grouping of members and information used in the compile process: Base Member Name, Omit Option, Generation Sev, and Maint/RSTDSP.

The Maint/RSTDSP field, in particular, designates the type of maintenance on a logical file, how a screen will be processed, or if the program contains embedded SQL statements.

<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Base Member Name</td>
<td>This field allows for the logical grouping of members. This field simply allows for logical grouping of members. For screens, reports, RPG programs and CL jobs, this name is usually the RPG program name associated with a particular member. For logical files, this name is the physical file upon which it is based and is required.</td>
</tr>
<tr>
<td>Omit Option</td>
<td>Designates items in the Software Versions Repository file that would be bypassed for a new release. These codes are as follows: H Held from all releases X Omit from all releases S Omit Source from all releases O Omit Execution Object from all releases</td>
</tr>
<tr>
<td>Generation Sev</td>
<td>Allows the user to designate a severity level when compiling a member. Because some J. D. Edwards programs contain messages that appear in the compile listing as a severity level 10 error, it is suggested that you override the IBM default of a severity level 9 to a level 20 for all programs. To do this, enter the following on any command line: CHGCMDDDFT CMD(DRTRPGPGM) NEW DFT(‘GENLVL(20)’) For those specific programs that must override the new default severity level of 20, you can enter the override value in the Generation Severity field.</td>
</tr>
<tr>
<td>Maint/RSTDSP</td>
<td>Designates the type of maintenance on a logical file, how a screen will be processed, or if the program contains embedded SQL statements.</td>
</tr>
</tbody>
</table>
Maintenance on a Logical File

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>No maintenance; or the logical is created dynamically</td>
</tr>
<tr>
<td>1</td>
<td>Logical will be immediately updated when physical is updated.</td>
</tr>
<tr>
<td>2</td>
<td>Logical update will be delayed until the next time it is opened. — USE WITH CAUTION</td>
</tr>
</tbody>
</table>

A value of 0 (zero) indicates no maintenance; or the logical is created dynamically.

A value of 1 (one) indicates the logical will be immediately updated when the physical is updated.

CAUTION: A value of 2 (two) indicates the logical update will be delayed until the next time it is opened. Use this value with caution.

Processing a Screen

<table>
<thead>
<tr>
<th>Value</th>
<th>Field Values</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>RSTDSP = *NO (Restore Display) DFRWRT = *YES (Defer Write)</td>
<td>Use with OVERLAY. Do not use with PUTOVR/OVRD1A All writes to the form field or file formats will be collected and written at one time</td>
</tr>
<tr>
<td>A</td>
<td>RSTDSP = *NO DFRWRT = *NO</td>
<td>Overlay Each write statement is written to the screen</td>
</tr>
<tr>
<td>Field</td>
<td>Explanation</td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>-------------</td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>RSTDST = *YES DFRWRT = *NO</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Use with PUTOVER to clear and write screen at field level</td>
<td></td>
</tr>
<tr>
<td>S</td>
<td>Used when compiling SQL, RPG, and PL1 programs. For example, if SQL statements exist within an RPG program, the compiler:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1) Executes a create SQL program statement</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2) Executes the SQL statements (converts them to calls)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3) Comments them out</td>
<td></td>
</tr>
<tr>
<td></td>
<td>4) Executes a create RPG program statement and continues as normal</td>
<td></td>
</tr>
</tbody>
</table>

**File Information**

The following fields identify the file information.

The following SVR fields identify the file information: File Prefix, Copy Data (Y/N), Optional File, and Common File.

<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>File Prefix</td>
<td>This field indicates the prefix associated with a file. Use F1 to display all file prefixes in use. Each physical file should have an unique file prefix.</td>
</tr>
<tr>
<td>Copy Data (Y/N)</td>
<td>Used to indicate when a database file must be copied with or without data.</td>
</tr>
<tr>
<td></td>
<td>The Create User Data Libraries (2/A9645) utility accesses this field to determine if the file copied will be copied with data.</td>
</tr>
<tr>
<td>Optional File</td>
<td>Indicates the file may be optional in your production environment. F8 provides a list of optional files.</td>
</tr>
<tr>
<td></td>
<td>................ Form-specific information ................</td>
</tr>
<tr>
<td></td>
<td>Designates if the file may not be needed at a client installation. The explanation of these situations can be found in the Generic Rate/Message information for that file for Generic Rate/Message Type 96/OF. All of these files that exist in a specified library can be listed in the Optional File Report on menu A9645.</td>
</tr>
<tr>
<td>Field</td>
<td>Explanation</td>
</tr>
<tr>
<td>--------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Common File</td>
<td>Indicates when a file should exist in the common library or user production library. The Create User Data Libraries (2/A9645) utility accesses this field to determine if the file should be placed in the specified common library or production library.</td>
</tr>
</tbody>
</table>
Where Are Members Maintained?

The bottom half of the Software Versions Repository form lists the libraries in which the member is maintained. This information is stored in the Software Versions Repository Detail file (F9802).

<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source Library</td>
<td>The library containing the data to be copied.</td>
</tr>
<tr>
<td>Object Library</td>
<td>The library where the source file for the object is maintained.</td>
</tr>
<tr>
<td>Member ID</td>
<td>The record of the Software Versions Repository member to be copied.</td>
</tr>
</tbody>
</table>

The source library where the source file for the object is maintained.

This library is usually JDFSRC (for J. D. Edwards) or CLTSRC (for the client) for production and DEVSRC for development.

The library where the compiled object resides.

Leave the object library name blank for copy modules since they are not compiled objects.

The source file containing the source member. At J. D. Edwards, three source files reside inside of the JDFSRC library.

They are:
- JDECPY for copy modules
- JDEESRC for other source code
- F98CRTCMD for precompiler commands
<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAR Number</td>
<td>An abbreviation for software action request (SAR).</td>
</tr>
<tr>
<td></td>
<td>• *NONE = the SAR number will not be validated in any of the CAD/CAP</td>
</tr>
<tr>
<td></td>
<td>programs and can be left blank.</td>
</tr>
<tr>
<td></td>
<td>• If a SAR number is entered, it is used in conjunction with the SAR</td>
</tr>
<tr>
<td></td>
<td>Delivery Type of *DFT (default).</td>
</tr>
</tbody>
</table>

**Form-specific information**

The most recent Software Action Request (SAR)/Work Order number associated with the member. This number must be valid, and if the status of the SAR number is complete, you should enter a new SAR to perform additional work on the member. A basic version of the Work Order system is sent to clients who have purchased the Computer Assisted Design (CAD) system and serves as a means for the client to keep track of their projects.

If a PPAT number is specified on the User Information screen (F0092 file), that number will show as the default for the window that comes up when F1 is pressed on this field.

The edit for this field is controlled by the SAR information entered in CASE Profiles.

| Version ID  | The software version number to be defaulted in the Software Versions       |
|-------------| Repository file.                                                          |

**Form-specific information**

Identifies the release level of the member in the designated environment.

Validated against User Defined Codes 98/RL/

| S C         | Determines the status of the software as well as where it resides in        |
|-------------| production.                                                               |
|             | It will specify that the software is in production, in development, or in |
|             | release.                                                                  |

**Form-specific information**

Status Code

Indicates the status of the software, that is, whether it is in production or development. These codes are as follows:

<table>
<thead>
<tr>
<th>Development</th>
<th>Development Progress Code</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Indicates the progress of modifications done to the member.</td>
</tr>
<tr>
<td>Field</td>
<td>Explanation</td>
</tr>
<tr>
<td>--------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>User ID</td>
<td>For World, The IBM-defined user profile.</td>
</tr>
<tr>
<td></td>
<td>For OneWorld, the creator of the version.</td>
</tr>
<tr>
<td></td>
<td>................................................................. Form-specific information ..........................................................</td>
</tr>
<tr>
<td></td>
<td>User ID that last modified the member (automatically updated).</td>
</tr>
<tr>
<td>Date Modified</td>
<td>The Date Modified field is simply the date that the DREAM Writer version,</td>
</tr>
<tr>
<td></td>
<td>Software Versions Repository Record, and so on, was last updated.</td>
</tr>
<tr>
<td></td>
<td>................................................................. Form-specific information ..........................................................</td>
</tr>
<tr>
<td></td>
<td>The date the member was last updated (automatically updated).</td>
</tr>
</tbody>
</table>

Each subfile line represents a record in the Software Versions Repository detail file (F9802).

Typing “D” in the Action Code deletes all the members and control data from:

- Software Versions Repository Master file (F9801)
- Software Versions Repository Detail file (F9802)
- Source and Object, if applicable
- Data Dictionary (F9200, F9203, F9816, F98163)
- Vocabulary Overrides (F9220)
- Function Key Definition (F9601,F9611)
- DREAM Writer forms (F98301, F9831, F98311, F98312)
- Cursor Sensitive Helps (F9620, F9621)
- Processing Options (F98302)
- Program Generator, if applicable
Naming Conventions

The following forms show how the report and CL program share the same description and base member as the program name. The same convention is true for the CL program and the special form.
A coded naming structure identifies and describes major components of J.D. Edwards software. The first character of the name indicates the type of component, such as program or data file. The second and third characters denote the system and are referred to extensively throughout the software. The fourth, fifth, and sixth characters represent the component group type, such as the function to be performed by the indicated component. The seventh through the tenth characters identify component versions. File names vary from four to eight characters in length, while all other component names are at least six characters long.
Use the following chart as your guide when naming objects.

<table>
<thead>
<tr>
<th>First digit — Component</th>
</tr>
</thead>
<tbody>
<tr>
<td>C — Common subroutine</td>
</tr>
<tr>
<td>I — Data structure; record formats</td>
</tr>
<tr>
<td>J — CL program</td>
</tr>
<tr>
<td>P — RPG program</td>
</tr>
<tr>
<td>R — Report</td>
</tr>
<tr>
<td>S — Special form</td>
</tr>
<tr>
<td>T — Temporary work files</td>
</tr>
<tr>
<td>V — Video screen display file</td>
</tr>
<tr>
<td>X — Scrub and Edit Server</td>
</tr>
<tr>
<td>XF — Input/Output File Server</td>
</tr>
<tr>
<td>XS — Input only/Caching Server</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Second and third digits — System Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>00 — World Foundation Environment</td>
</tr>
<tr>
<td>01 — Address Book</td>
</tr>
<tr>
<td>03 — Accounts Receivable</td>
</tr>
<tr>
<td>55 — Reserved for clients</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Fourth, Fifth, and Sixth Digits — Group Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>000 to 099 — File maintenance</td>
</tr>
<tr>
<td>100 to 199 — Transaction processing</td>
</tr>
<tr>
<td>200 to 299 — Inquiry only</td>
</tr>
<tr>
<td>300 to 399 — Input registers and journals</td>
</tr>
<tr>
<td>400 to 499 — Operating reports</td>
</tr>
<tr>
<td>500 to 599 — Special purpose reports</td>
</tr>
<tr>
<td>600 to 799 — Standard management reports</td>
</tr>
<tr>
<td>800 to 999 — Housekeeping</td>
</tr>
<tr>
<td>DS — Data structure</td>
</tr>
<tr>
<td>Other — Window designations</td>
</tr>
</tbody>
</table>

A CL program, RPG program and a Report file may have identical names with different prefixes.

For example: J01051, P01051, R01051 (Address Book Revisions).

Use the following as a guide when naming files.
First digit — Component
F — Data file (physical or logical)

Second and third digits — System Code
00 — Operations Control/Back Office
01 — Address Book
03 — Accounts Receivable

Fourth and Fifth Digits — Group Type
01 — Master
02 — Balance
11 — Transaction

Sixth through Tenth Digits — Identifying Suffixes
These digits differentiate component versions.
Example — Programs that perform similar functions
but vary distinctly in specific processing.
WF — Work File
LA thru LZ — Logical File Designations
JA thru JZ — Join Logical File Designations
Version ID — 3 digit number appended to saved
DREAM Writer logical file name
The following shows the names for different types of programs and files.

**Maintenance program**  The maintenance program for a file has the same name with a different prefix.

For example, F9220 is P9220 or F9601 is P9601.

**Logical files**  For logical files over one physical, the logical file has the same name as the physical followed by an L, followed by A thru Z.

For example, F0101 has logicals F0101LA, F0101LB, F0101LC, and F0101LD.

**Join logical files**  Join Logical files have the same name as the principal based-on file, a suffix of J followed by A thru Z.

For example, the system names the join of F0006 and F0911 as F0006JA

**Temporary files**  Batch jobs use T files doing a CRDTDUPOBJ. The job then removes the object after completion.

- Usually Physical Files
- Begin with T
- Found in JDFOBJ

**Dynamic work files**  Dynamic work files are usually FASTR processing requirements. Dynamic work files create and delete after the job is complete.

- Usually logical files
- Have same name as program
The J.D. Edwards System Codes

When used in menus, the system code follows the letter in the menu name. Shown below are the system codes for the standard AS/400 systems:

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>World Foundation Environment</td>
</tr>
<tr>
<td>01</td>
<td>Address Book</td>
</tr>
<tr>
<td>02</td>
<td>Electronic Mail</td>
</tr>
<tr>
<td>03</td>
<td>Accounts Receivable</td>
</tr>
<tr>
<td>04</td>
<td>Accounts Payable</td>
</tr>
<tr>
<td>05</td>
<td>Stand-Alone Time Accounting</td>
</tr>
<tr>
<td>07</td>
<td>Payroll “Enhanced”</td>
</tr>
<tr>
<td>08</td>
<td>Human Resources</td>
</tr>
<tr>
<td>09</td>
<td>General Accounting</td>
</tr>
<tr>
<td>10</td>
<td>Financial Reporting</td>
</tr>
<tr>
<td>11</td>
<td>Multi Currency/Cash Basis</td>
</tr>
<tr>
<td>12</td>
<td>Fixed Assets</td>
</tr>
<tr>
<td>13</td>
<td>Equipment/Plant Management</td>
</tr>
<tr>
<td>14</td>
<td>Modeling, Planning, &amp; Budgeting</td>
</tr>
<tr>
<td>15</td>
<td>Commercial Property Management</td>
</tr>
<tr>
<td>16</td>
<td>Resident Property Management</td>
</tr>
<tr>
<td>17</td>
<td>Property Management Base</td>
</tr>
<tr>
<td>18</td>
<td>Deal Management</td>
</tr>
<tr>
<td>20</td>
<td>Energy Base</td>
</tr>
<tr>
<td>30</td>
<td>Product Data Management</td>
</tr>
<tr>
<td>31</td>
<td>Shop Floor Control</td>
</tr>
<tr>
<td>32</td>
<td>Configuration Management</td>
</tr>
<tr>
<td>33</td>
<td>Capacity Requirements Planning</td>
</tr>
<tr>
<td>34</td>
<td>DRP/MRP/MPS</td>
</tr>
<tr>
<td>35</td>
<td>Enterprise Facility Planning</td>
</tr>
<tr>
<td>40</td>
<td>Inventory/OP Base</td>
</tr>
<tr>
<td>41</td>
<td>Inventory Management</td>
</tr>
<tr>
<td>42</td>
<td>Sales Order Processing</td>
</tr>
<tr>
<td>43</td>
<td>Purchasing Order Processing</td>
</tr>
<tr>
<td>44</td>
<td>Contract Management</td>
</tr>
<tr>
<td>45</td>
<td>Advanced Price Adjustments</td>
</tr>
<tr>
<td>46</td>
<td>Warehouse Management</td>
</tr>
<tr>
<td>47</td>
<td>Electronic Data Interchange</td>
</tr>
<tr>
<td>48</td>
<td>Work Order Processing</td>
</tr>
<tr>
<td>49</td>
<td>Load and Delivery</td>
</tr>
<tr>
<td>50</td>
<td>Job Cost Base</td>
</tr>
<tr>
<td>51</td>
<td>Job Cost Accounting</td>
</tr>
<tr>
<td>52</td>
<td>Job Cost Billing</td>
</tr>
<tr>
<td>53</td>
<td>Change Management</td>
</tr>
<tr>
<td>55</td>
<td>Client Use</td>
</tr>
<tr>
<td>60</td>
<td>JDE Internal Custom Programming</td>
</tr>
<tr>
<td>70</td>
<td>Multi-National Products</td>
</tr>
<tr>
<td>71</td>
<td>Client/Server Applications</td>
</tr>
<tr>
<td>72</td>
<td>World Vision</td>
</tr>
<tr>
<td>73</td>
<td>CS — A/P Entry</td>
</tr>
<tr>
<td>74</td>
<td>CS — Pay Time Entry</td>
</tr>
<tr>
<td>75</td>
<td>CS — Sales Order Entry</td>
</tr>
<tr>
<td>76</td>
<td>CS — Training and Development</td>
</tr>
<tr>
<td>77</td>
<td>Canadian Payroll</td>
</tr>
<tr>
<td>79</td>
<td>CS — Translation</td>
</tr>
<tr>
<td>80</td>
<td>COBOL Translator</td>
</tr>
<tr>
<td>81</td>
<td>DREAM Writer</td>
</tr>
<tr>
<td>82</td>
<td>World Writer</td>
</tr>
<tr>
<td>83</td>
<td>Management Reporting — FASTR</td>
</tr>
<tr>
<td>84</td>
<td>Distributive Data Processing</td>
</tr>
<tr>
<td>85</td>
<td>Custom Programming</td>
</tr>
<tr>
<td>86</td>
<td>Electronic Document Interchange</td>
</tr>
<tr>
<td>87</td>
<td>Miscellaneous Tech</td>
</tr>
</tbody>
</table>
Examples of Program and File Names

The following is a detailed breakdown of program and file names. All of the file types have the same system code and component group.

**Data Files**

Account Master File
- Component (File)
- System Code (General Accounting)
- Component Group Type (Master)

Account Master Alternate Logical
- Component (File)
- System Code (General Accounting)
- Component Group Type (Master)
- Version Identification (Logical)

**Videos (Screens)**

Component (Video)
- System Code (General Accounting)
- Component Group Type (File Maintenance)

**RPG Programs**

Component (RPG Program)
- System Code (General Accounting)
- Component Group Type (File Maintenance)

**CL Programs**

Component (CL Program)
- System Code (General Accounting)
- Component Group Type (File Maintenance)
Optional Files Workbench

The Optional Files Workbench provides better access to optional files. When you delete optional files, they are logged. If you reinstall, those files are not put back into the system. Each file has an explanation about the circumstances that makes it optional.

- If you need the deleted files, you can remove them from the logged optional files and copy them from JDFDATA.
Logical Files

- The Member ID for logical files ends with Lx, where x is the next available letter in alpha sequence.
- The Object Library is usually JDFDATA.
- The Description should list the key fields for the view.
- The Maint/RSTDSP is 1 for permanent system logicals.
- The Base Member Name is the physical file the logical view is over.
Join Logical Files

- The Description lists the files over which the join is built.
- The Base Member Name is the primary file in the join.
- Physical files must exist in the same library.

9801

Software Versions Repository

| Action Code | I
| Member ID   | F0006JA
| Description | JF – BILLING ONLY – F0006/F0911 – Business Unit
| Function Code | LF Logical Files
| Function Use  | 210 Master Files
| System Code   | 00 Technical Foundation
| Reporting System | 00 Technical Foundation
| Base Member Name | F0006
| File Prefix    | MC
| Maint/RSTDSP  | 2 Omit Option...
|  Generation Sev | 1
| Copy Data (Y/N) | N Optional File...
| Common File    | N

<table>
<thead>
<tr>
<th>Source</th>
<th>Object</th>
<th>Source</th>
<th>SAR</th>
<th>Version</th>
<th>SD</th>
<th>User</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>P Library</td>
<td>JDFSRC</td>
<td>Library</td>
<td>JDFDTA</td>
<td>File</td>
<td>JDESRC</td>
<td>Number</td>
<td>ID</td>
</tr>
<tr>
<td>JDE</td>
<td>493167</td>
<td>A61</td>
<td>I</td>
<td>JDE</td>
<td>03/05/93</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Copy Modules

- The Member ID begins with C, I, E, D, G.
- The Source File is JDECPY.
- The Description describes the function of the module.
- The Function Code is COPY.

<table>
<thead>
<tr>
<th>Action Code</th>
<th>Description</th>
<th>Function Code</th>
<th>Function Use</th>
<th>System Code</th>
<th>Reporting System</th>
<th>Base Member Name</th>
<th>Release</th>
<th>File Prefix</th>
<th>Omit Option</th>
<th>Generation Sev</th>
<th>Copy Data (Y/N)</th>
<th>Optional File</th>
<th>Common File</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>Copy Module – Retrieve Soft Coding</td>
<td>COPY</td>
<td>194</td>
<td>Technical Tools</td>
<td>Technical Tools</td>
<td>C00SC</td>
<td>A61</td>
<td></td>
<td></td>
<td></td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Library</th>
<th>Library</th>
<th>File</th>
<th>Number</th>
<th>ID</th>
<th>CP</th>
<th>ID</th>
<th>Modified</th>
</tr>
</thead>
<tbody>
<tr>
<td>JDFSRC</td>
<td>JDFOBJ</td>
<td>JDECPY</td>
<td>683784</td>
<td>A61</td>
<td>I</td>
<td>JDE</td>
<td>06/10/93</td>
</tr>
</tbody>
</table>
Windows

- The Member ID begins with V, the system number, then an alphabetic identifier as shown in the example below.
- The Description describes the function of the form.
- Maint/RSTDSP is left blank to allow the form to appear in front of text from the calling form.

<table>
<thead>
<tr>
<th>Action Code...</th>
<th>I</th>
</tr>
</thead>
<tbody>
<tr>
<td>Member ID......</td>
<td>V09ACCT</td>
</tr>
<tr>
<td>Description...</td>
<td>Account Master Additions Window</td>
</tr>
<tr>
<td>Function Code..</td>
<td>DSPF Video Display Files</td>
</tr>
<tr>
<td>Function Use...</td>
<td>111 File Maintenance</td>
</tr>
<tr>
<td>System Code....</td>
<td>09 General Accounting</td>
</tr>
<tr>
<td>Reporting System</td>
<td>09 General Accounting</td>
</tr>
<tr>
<td>Base Member Name</td>
<td>P09ACCT</td>
</tr>
<tr>
<td>Maint/RSTDSP...</td>
<td>Omit Option...</td>
</tr>
<tr>
<td>Copy Data (Y/N).</td>
<td>N</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>O</th>
<th>Source</th>
<th>Object</th>
<th>Source</th>
<th>SAR</th>
<th>Version</th>
<th>SD</th>
<th>User</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>P</td>
<td>Library</td>
<td>Library</td>
<td>File</td>
<td>Number</td>
<td>ID</td>
<td>ID</td>
<td>Modified</td>
<td></td>
</tr>
<tr>
<td></td>
<td>JDFSRC</td>
<td>JDFOBJ</td>
<td>JDESRC</td>
<td>552868</td>
<td>A61</td>
<td>1</td>
<td>JDE</td>
<td>12/08/92</td>
</tr>
</tbody>
</table>
Navigation Functions

The following Function keys facilitate navigating within the Software Versions Repository.

**F6 - Access Repository Services**

F6 – You can access the Repository Services form using F6. This form provides access to the other repository services within J.D. Edwards.

<table>
<thead>
<tr>
<th>9801</th>
<th>Software Versions Repository</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action Code...</td>
<td>I</td>
</tr>
<tr>
<td>Member ID...</td>
<td>P01051</td>
</tr>
<tr>
<td>Description...</td>
<td>Address Book Information</td>
</tr>
<tr>
<td>Function Code...</td>
<td>RPG RPG Programs</td>
</tr>
<tr>
<td>Function Use...</td>
<td>111 File Maintenance</td>
</tr>
<tr>
<td>System Code...</td>
<td>D1 Address 985001</td>
</tr>
<tr>
<td>Reporting System...</td>
<td>D1 Address</td>
</tr>
<tr>
<td>Base Member Name...</td>
<td>P01051</td>
</tr>
<tr>
<td>Maint/RSTDSP...</td>
<td>Omit Opt</td>
</tr>
<tr>
<td>Copy Data (Y/N)...</td>
<td>N Optional</td>
</tr>
<tr>
<td>O Source Library Library File</td>
<td></td>
</tr>
<tr>
<td>JDFSRC61 JDFOBJ61 JDESR</td>
<td></td>
</tr>
<tr>
<td>JDXSRC61 JDXOBJ61 JDESR</td>
<td></td>
</tr>
<tr>
<td>PGXSRC61 PGXOBJ61 JDESR</td>
<td></td>
</tr>
</tbody>
</table>

Available Services:

1. Data Dictionary
2. Menus
3. Vocabulary Overrides
4. Function Key Definitions
5. Processing Options
6. Edit System Helps
7. CASE Profiles
8. SAR Log Inquiry
9. Copy DD,VO,DW,UDC,SVR,Menus

Use selection 1 to select the available services.

**F9 - Automatic Reinquiry**

F9 – Once the system has accepted the changes you made to a member and cleared the screen, you can automatically inquire on that member by pressing F9.

**F17 - Position Cursor to Action Code**

F17 – When you inquire on a member, the system positions the cursor in the subfile for the screen. To reposition your cursor in the Action Code field, press F17.

**F19 - Previous Member**

F19 – To access the member stored before the current member, press F19.
F20 - Next Member

F20 – To access the member stored after the currently displayed member, press F20.

Other Function Keys

F2 - J.D. Edwards Command Line

F2 – To access a command line to enter a J.D. Edwards or IBM command without having to exit to Command Entry or a menu.

Calls a J.D. Edwards program and not the IBM Command Entry.

If you are secured out of Command Entry or Menu Traveling, you still receive this command line but you cannot execute commands or menu travel.

F8 - Optional Files

F8 – The system displays the optional files.

F10 - Checklists

F10 – Displays a user defined checklist. Opt 1 displays additional job information.

F13 - Member Category Codes

F13 – Displays additional category code information for each member. You can cross-reference category code values to the Software Versions Search program (23/G91).

F14 - Member Parameter/Key List

F14 – Identifies the access path for keyed files.

F15 – Where Used Facility

F15 – SVR Where Used Facility

F15 – You can access the Where Used facility using Function key 15 on Software Versions Repository. Use this facility to determine every location that a particular member is used.
Below is an example form displaying every program that uses the Business Unit Master form:

980014
Cross Reference

Object: Name . . . V0006 Business Unit Master Revisions – Single
Type . . . P All programs using file
To Display P
Funct Cd . ___

Opt: 1=SVR 2=Create Object 3=Field Explanation  F21=Print  F16=Regenerate

To use this facility, you must run the Cross Reference Rebuild.(6/G9642)

NOTE: To use this facility, you must run the Cross Reference Rebuild.(6/G9642)

**F23 - Flow Programs / Illustrate File Models**

F23 – To display a flowchart if the member is a program or a Data Model if the member is a file.

Only functional for programs and files.
Selection Exits from the Software Versions Repository

The following is a list of the options available from the Software Versions Repository. By referring to the form pictured in the beginning of the chapter, you can see that there are more options than can be displayed on the form.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
</table>
| 1      | Browse SEU member  
          Displays the SEU Member in browse mode. |
| 2      | Edit using SEU  
          Displays the SEU Member in update mode. |
| 3      | Copy/Add entry/source member  
          Copies the source member to another member.  
          Adds master and detail record for the member being copied to if they do not already exist.  
          Copies pre-compiler commands and Vocabulary Overrides. Copies program generator specifications if requested. |
| 5      | Work with SAR detail  
          Displays the SAR/Work Order Detail Entry screen, defaulting to the members affected portion of the SAR/Work Order. |
| 8      | Print source  
          Prints a spooled file of the member. |
| 9      | Delete/ remove source  
          Deletes the detail record and removes the source member from the source file.  
          The same IBM authority that applies to the command RMVM applies to this function.  
          Will delete the object if requested by the user. If you do not remove the source member, you will not be allowed to delete the object. |
| 10     | Exit to design aid  
          Determines what type of member you are accessing and then exits to the correct J.D. Edwards design tool; that is, SDA/RDA/FDA/Program Generator. |
<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
</table>
| 11     | Precompiler commands for J.D. Edwards compiler.  
          Accesses the source code for the precompiler commands associated with a program.  
          A highlighted message, *Precompiler Commands Exist*, indicates when they exist for the member.  
          Contains information for steps that need to be completed prior to compiling the program.  
          **NOTE:** Only one person can view the same pre–compiler commands. |
| 14     | Submit object creation  
          Compiles the member and generates an object. |
| 15     | Generate program source and help  
          Submits the member to the program generator in order to generate source and related helps.  
          Only applicable to CASE users. |
| 17     | Edit help instructions  
          Accesses the help instructions for a particular program in update mode utilizing SEU. |
| 18     | Generate & rebuild help instructions  
          Submits the helps for generation and rebuilds them into their final form once they have been entered. |
| 20     | Browse SDA/RDA  
          Accesses SDA or RDA in browse mode. |
| 21     | Print help  
          Prints the help instructions for the member. |
| 25     | Print illustration  
          Prints an illustration of printer files, display files, or data base files. |
| 30     | Source modifications editor  
          Allows you to view the source modifications made to the member through SEU after source was generated. Stored in the F93002 file.  
          Only applicable to CASE users. |

**Exercises**

See the exercises for this chapter.
**CASE Profiles**

**About CASE Profiles**

CASE profiles are user defined values that can pertain to individual users or to one *PUBLIC user profile.

![F98009](image)

Information is stored in the CASE profiles file (F98009).

These profiles are used to define the overall CASE operating environment.

Various processing control parameters are defined by the user including:

- Default development libraries
- Compile job queue
- Program Generator source generation job queue
- Compile print options
- SAR logging options
You should immediately update the record for User ID *PUBLIC.

When entering information for *PUBLIC, all fields are required.

Default CASE Profile values are maintained in a record with the User ID *PUBLIC. You should enter CASE Profile values for individual users only if you need overrides to the *PUBLIC values.

When entering values for individual users, you can leave all fields blank except for the specific values being overridden.

Accessing CASE Profiles

To access CASE Profiles

To access CASE Profiles, choose one of the following methods:

- From menu G92, choose CASE Profiles
From the Repository Services form, select CASE Profiles.

The new CASE Profiles form appears. The program attempts to automatically inquire on your User ID. If your ID is not set up, an error occurs. You can inquire on *PUBLIC.
## Default Development Environment

<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source File Name</td>
<td>The default source file name where source is to be stored within the source library. Must reside within the source library specified.</td>
</tr>
<tr>
<td>Source Library</td>
<td>The default library where source will be stored. The source file specified must reside within this library.</td>
</tr>
<tr>
<td>Object Library</td>
<td>The default library where compiled objects will be stored.</td>
</tr>
<tr>
<td>CL Source File</td>
<td>The default library where source for CL programs will be stored. The value specified must reside within the source library specified.</td>
</tr>
<tr>
<td>Data File Library</td>
<td>Used to specify the test (or development) library for physical and logical files. Used as the default object library for the Software Versions Repository when copying source code for physical or logical files.</td>
</tr>
<tr>
<td>SAR Number</td>
<td>An abbreviation for software action request (SAR).</td>
</tr>
<tr>
<td></td>
<td>• *NONE = the SAR number will not be validated in any of the CAD/CAP programs and can be left blank.</td>
</tr>
<tr>
<td></td>
<td>• If a SAR number is entered, it is used in conjunction with the SAR Delivery Type of *DFT (default).</td>
</tr>
<tr>
<td>Version ID</td>
<td>The software version number to be defaulted in the Software Versions Repository file.</td>
</tr>
<tr>
<td>Status Code</td>
<td>Determines the status of the software as well as where it resides in production. It will specify that the software is in production, in development, or in release.</td>
</tr>
</tbody>
</table>

If you create a custom environment, put 2, 3, or 4 in user defined codes. If you have a “1” (production) the system will think it is a J.D. Edwards file and write over it during the Software Version Repository Merge in an upgrade.
Program Creation Options

You have the following options when you create a program.

<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Compile Job Queue</td>
<td>Specifies which job queue will be used when submitting programs to compile. This job queue is used for programs with function code of RPG, CBL, PLI, C and SYSC.</td>
</tr>
<tr>
<td>Prog Gen Job Queue</td>
<td>Specifies which job queue will be used when submitting jobs from the Program Generator. These jobs include the source code generation and the source code monitor from SEU.</td>
</tr>
<tr>
<td>Compile Target Release</td>
<td>Used by various AS/400 compilers (RPG,CLP,COBOL,C) to compile an object compatible with a specified target release.</td>
</tr>
<tr>
<td></td>
<td>• A value of *CURRENT compiles an object compatible to the release of the machine at compile time.</td>
</tr>
<tr>
<td></td>
<td>• A value of *PRV compiles an object compatible with both one release back and the current release.</td>
</tr>
<tr>
<td>Print Option</td>
<td>Used to designate whether or not a report will be generated when an object is compiled.</td>
</tr>
<tr>
<td></td>
<td>• 0 = no print</td>
</tr>
<tr>
<td></td>
<td>• 1 = print</td>
</tr>
<tr>
<td></td>
<td>• 2 = print and hold spool file</td>
</tr>
<tr>
<td></td>
<td>• 3 = print only, does not generate execution object (applies to COBOL and RPG only)</td>
</tr>
<tr>
<td></td>
<td>• 4 = print when compile or creation fails</td>
</tr>
<tr>
<td>Cross–Reference Listing</td>
<td>Specifies whether a cross-reference listing will be generated for variables and fields in a program's compile listing.</td>
</tr>
</tbody>
</table>

You must have a job queue called COMPILE for COMPILE JOB QUEUE to compile programs or use a valid job queue.
**SAR Options**

The following fields provide you with options for the location of your SAR file and SAR logging.

<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAR File Library</td>
<td>Specifies the library where the Software Action Request (SAR) file for software development exists.</td>
</tr>
<tr>
<td></td>
<td>• If left blank, the user's library list will be used.</td>
</tr>
<tr>
<td></td>
<td>• You can specify *NONE in the SAR number field if you do not want any SAR number editing.</td>
</tr>
<tr>
<td>SAR Delivery Type</td>
<td>Associated with SAR logging. SAR logging is a feature which tracks all activities related to modifying J. D. Edwards' software.</td>
</tr>
<tr>
<td></td>
<td>• *NONE = no logging.</td>
</tr>
<tr>
<td></td>
<td>• *LOG = log to SAR number 00000000 (no SAR number is used for logging).</td>
</tr>
<tr>
<td></td>
<td>• *DFT = log to a default SAR number (specified in the SAR Number field).</td>
</tr>
<tr>
<td></td>
<td>• *PROMPT = log and prompt the user for the SAR number to be used and allow the user to enter the revision notes.</td>
</tr>
</tbody>
</table>

**Miscellaneous**

The following fields are reserved for future use.

<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source Gen Opt (Future)</td>
<td>For future use.</td>
</tr>
<tr>
<td>Helps Maint Opt(Future)</td>
<td>For future use.</td>
</tr>
</tbody>
</table>

**Function Key Exits From the CASE Profiles Program**

**F6 - Access Repository Services**

F6 – This key will display a form that provides access to the other repository services, except for CASE profiles.

**F9 - Previous Profile**

F9 – Allows you to re-inquire on the last record updated.
Summary of CASE Profiles

- The CASE Profiles file is F98009.
- You need to update the *PUBLIC record as well as add any additional individual records desired.
- You cannot delete the *PUBLIC record.
- When entering information for the *PUBLIC record, all fields are required.
- The record for User ID *PUBLIC contains the values that are used as the defaults for all users unless individual user profiles have been set up.
- When entering values for individual profiles, all fields are left blank EXCEPT for the specific values being overridden on the *PUBLIC profile.
- SAR Number and SAR Delivery type work together to determine what type of SAR logging should occur.
  - *NONE = no SAR logging at all.
  - *LOG = no SAR number is included as part of the SAR logging.
  - *DFT = the SAR number specified is used for the SAR logging.
  - *PROMPT = you are prompted for a SAR number and revision notes when an entry is about to be made to the SAR log.

Exercises

See the exercises for this chapter.
Working with SAR Log

About SAR Log

After you create the SARs, you must activate SAR logging, which automatically tracks the SARs as you develop the software.

The SAR Log Inquiry program allows you to review information in the SAR Log file (F9810).

You can also change the SAR Number and Revision Notes for individual log records.

Complete the following tasks:

- Set up user input options for SAR logging
- Select types of SAR information to log
- Access SAR Log Inquiry
Before You Begin

☐ Create SARs before you activate SAR logging.

From the Version Control menu (G9261), choose CASE Profiles.

Setting Up User Input Options for SAR Logging

To set up user input options for SAR logging

On CASE Profiles

```
98009                   CASE Profiles
Action Code. . . . . . .  I
User ID. . . . . . . . .  MORRIS

Default Development Environment
Source File . . . . . .  JDESRC
Source Library. . . .  PEGSRC561
Object Library. . . .  PGFOBJ561
CL Source File. . . .  JDECLSRC
Data File Library . .  PGFDTA61
SAR Number. . . . . .  774487
Version ID. . . . . .  A61
Status Code . . . . .  2

Program Creation Options
Compile Job Queue . . .  COMPILE
Prog Gen Job Queue . .  CLONE
Compile Target Release. *CURRENT
Print Option . . . . . .  1
Cross-Reference Listing N

SAR Options
SAR File Library . . .  DDPDATA
SAR Delivery Type . .  *DFT

Miscellaneous
Source Gen Opt (Future)  SEU
Helps Maint Opt (Future)  SEU

F24=More Keys
```

1. Complete the following fields:
   - SAR File Library
   - SAR Delivery Type

The SAR file library contains the Work Order system files (F4801 and F4802). If you currently use these files, and if the F4802 file has different record types than what version control needs, you must create a library that contains new F4801 and F4802 files for version control purposes only. Specify this new library name in the SAR File Library field.

If you set the SAR Delivery Type field to *PROMPT, the Maintain User Default SAR Information form appears whenever you change a source code member or control table.
What You Should Know About

- If you provided a SAR number on CASE Profiles, it appears on this form. If you did not provide a SAR number, provide one on this form.
- If the Transfer field on Maintain User Default SAR Information is set to 1, the Version Control system can promote the change. If it is set to 0, the system cannot promote the change.

2. Complete the following optional field:
   - SAR Number

What You Should Know About

**SAR number and delivery type combinations**

The information you provide for the SAR Number and SAR Delivery Type fields affects how the system handles SAR logging.

If you do not provide a SAR number, and set the SAR Delivery Type field to *PROMPT, the Maintain User Default SAR Information form prompts you for the SAR number whenever you change a source code member or control table.

If you provide a SAR number, and set the SAR Delivery Type field to *DFT, the system creates SAR log entries automatically without your input.

If you provide a SAR number, and set the SAR Delivery Type field to *PROMPT, the Maintain User Default SAR Information form prompts you to change the SAR number, if necessary, whenever you change a source code member or control table.

**Invalid SAR delivery types**

*LOG and *NONE are not valid for the SAR Delivery Type field when you use the Version Control system.

If you set the SAR Delivery Type field to *PROMPT, the Maintain User Default SAR Information form appears whenever you change a source code member or control table.
Selecting Types of SAR Information to Log

To select types of SAR information to log

In addition to setting up user input options for SAR logging, select the types of SAR information you want to log.

3. From the Version Control menu (G9261), access the processing options for Edit and Promote.

4. Make the following changes:

**SAR Logging (1)** Specify Y if you want to track SARs that are associated with J.D. Edwards source code and control file development only. Specify N if you want to track SARs that are associated with all software development. Leave this processing option blank to disable SAR logging and, therefore, version control.

If you specify Y, the SAR log keeps track of development automatically. It tracks changes to menus that start with “A” or “G” only. For DREAM Writer, it tracks changes to XJDE or ZJDE versions only. When you transfer these versions, the user ID associated with them changes to DEMO.

In addition, the SAR logging program runs a double-byte analysis against your RPG programs if you set this processing option to Y.

If you specify Y, you also must indicate the name of the library that contains your SAR files. The default library name is JDCOMDATA.

**DREAM Writer Copy (2)** Specify Y to track changes to DREAM Writer versions (XJDE and ZJDE versions only). Specify N to not track these changes. If you track changes, the user ID changes to DEMO automatically when you transfer the versions.
Accessing SAR Log Inquiry

The SAR Log Inquiry includes several functions:

- Inquiry by user ID or SAR number with date range
- Exit to a maintenance program for the record type
- Exit to SAR detail
- Print option that allows for DREAM Writer selection

There are two ways to access the SAR Log Inquiry.

To access the SAR Log Inquiry

1. To access the SAR Log Inquiry, select one of the following methods:
   - Choose SAR Inquiry from Menu G9362
   - Choose SAR Log Inquiry from the Repository Services form
The new SAR Log Inquiry form appears.

2. Complete one or more of the following fields:
   - User ID
   - SAR number
   - Date range

Records matching the search criteria are displayed.
<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>AC (Action)</td>
<td>The action that was taken on this record. The standard action code values apply.</td>
</tr>
<tr>
<td>Ty (Record Type)</td>
<td>The type of record that was updated. Use F1 to display all valid record types stored in User Defined Code 98/RT.</td>
</tr>
<tr>
<td>Item</td>
<td>The identification number (program number, file number, report number) assigned to any element of the software. These items are the members that reside in the Software Versions Repository or other repositories such as the Data Dictionary, Vocabulary Overrides etc.</td>
</tr>
<tr>
<td>SAR Number</td>
<td>The SAR number under which this change was made. This field can be updated on this video.</td>
</tr>
<tr>
<td>Revision Note</td>
<td>A user defined description field to further clarify the change made. This field can be updated on this video.</td>
</tr>
<tr>
<td>Time</td>
<td>The time at which the change was made.</td>
</tr>
<tr>
<td>Date</td>
<td>The date on which the change was made.</td>
</tr>
<tr>
<td>User</td>
<td>The user who made the change.</td>
</tr>
</tbody>
</table>

**Selection Exits from the SAR Log Inquiry**

The following is a list of selection exits from the SAR Log Inquiry form and an explanation of the effects of each selection.

<table>
<thead>
<tr>
<th>Exit</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 – Edit</td>
<td>Allows for maintenance of the record type. What program is accessed is based on the record type. For example, if the record type is ‘DD’, this exit will take the user to the Data Dictionary program.</td>
</tr>
<tr>
<td>5 – Work SAR</td>
<td>Exits to the SAR associated with the SAR log entry.</td>
</tr>
<tr>
<td>9 – Delete</td>
<td>Allows the user to delete entries from the SAR log.</td>
</tr>
</tbody>
</table>

If you entered this program from the Repository Services form from the Software Versions Repository program, selection exit 2 does not function with record types “SV” or “PG” as these record types attempt to call the Software Versions Repository, which causes a recursive call error.
Function Key Exits from the SAR Log Inquiry

**F5 - ASI Entry**
F5 – Exits to Application Specific Instructions form for use during a software upgrade. You need the F0098 file to do this.

**F6 - Access Repository Services**
F6 – Pressing this key displays a form that provides access to the other repository services, except for SAR Log Inquiry.

**F21 - Print**
F21 – Allows you to print a SAR log report.

Exits to a DREAM Writer versions list.

Summary of the SAR Log Inquiry

The SAR Log Inquiry has the following features and restrictions:

- Uses the file F9810.
- If you so not want to use the SAR Logging feature at all, specify *NONE in the SAR Delivery Type field for all CASE Profile records.
- To use the SAR Logging feature, you must specify a value of *LOG, *DFT, or *PROMPT in the SAR Delivery Type field for all CASE Profile records.
- The SAR Logging feature records any changes that you make to the Data Dictionary, Vocabulary Overrides, User Defined Codes, and so forth.
- The SAR Log Inquiry program allows you to see what changes you make to any of the above.
- The SAR Log Inquiry program has Function Keys and Selection Exits which allow you to change the SAR Log records in the SAR Log file (F9810) or to exit to the maintenance program for the change you made.

For example, exit to the Data Dictionary program if the record indicates a Data Dictionary item was added or updated.

Exercises

See the exercises for this chapter.
Work with Promotion Paths and Projects

Working with Promotion Paths and Projects

A promotion path defines how a project’s source code members and control file data will move from one environment to another. An environment consists of source code members and control file data. For source code members, the environment consists of:

- A source file
- A source library
- An object library

For control file data, the environment consists of a data library.

Perform the following tasks:

- Understand promotion paths
- Define a promotion path
- Define a project
Understanding Promotion Paths

A promotion path specifies the current locations of source code members and control file data and where they will be moved. For example, promoting a project’s source code members and control file data from a development environment to a test environment could look similar to the following illustrations.

Each move between two environments requires that you define a unique promotion path.
A project is a collection of software and data you want to group together for promotion. A project is defined by the following characteristics:

- SARs that are associated with the project
- Promotion paths that determine the movement of the project software and data between environments
- Other projects that are attached to the project

**Before You Begin Defining a Promotion Path**

- Verify that the SARs and promotion paths you want to associate with a project have been set up.

- The SAR system uses the Work Order files (F4801 and F4802). If your production environment uses these files, and if the F4802 file has different record types than what version control needs, define a separate library that contains these files for version control purposes only.
Defining a Promotion Path

Several steps are involved in defining promotion paths. Complete the following tasks:

- Locate a promotion path
- Add a promotion path
- Define a promotion path for source code members
- Define a promotion path for control tables

From the Version Control menu (G9261), select Manage Promotion Paths.

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
<th>Release</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>JDF73 Transfer to JDF73</td>
<td>A73</td>
</tr>
<tr>
<td>2</td>
<td>JDF73T 'T' file transfer to JDF73</td>
<td>A73</td>
</tr>
<tr>
<td>3</td>
<td>JDF73TEC Transfer to JDF73 SECURE</td>
<td>A73</td>
</tr>
<tr>
<td>4</td>
<td>JDU71 Transfer to JDU71</td>
<td>A71X</td>
</tr>
<tr>
<td>5</td>
<td>T130892PC2 Utility CIS - PCCPY</td>
<td>A71X</td>
</tr>
<tr>
<td>6</td>
<td>T130892PC3 Utility CIS - PCCPY</td>
<td>A71X</td>
</tr>
<tr>
<td>7</td>
<td>T130892PC4 Utility CIS - PCCPY</td>
<td>A71X</td>
</tr>
<tr>
<td>8</td>
<td>T130892PC5 Utility CIS - PCCPY</td>
<td>A71X</td>
</tr>
<tr>
<td>9</td>
<td>T130892PC6 Utility CIS - PCCPY</td>
<td>A71X</td>
</tr>
<tr>
<td>10</td>
<td>T130892PC7 Utility CIS - PCCPY</td>
<td>A71X</td>
</tr>
<tr>
<td>11</td>
<td>T130892PC8 Utility CIS - PCCPY</td>
<td>A71X</td>
</tr>
<tr>
<td>12</td>
<td>T130892PC9 Utility CIS - PCCPY</td>
<td>A71X</td>
</tr>
<tr>
<td>13</td>
<td>UQF62 UQF build for A6.2</td>
<td>A62</td>
</tr>
<tr>
<td>14</td>
<td>UQF71 UQF build</td>
<td>A71</td>
</tr>
<tr>
<td>15</td>
<td>VCT Version control training</td>
<td>A71</td>
</tr>
</tbody>
</table>

Opt: 1=Change  2=Members  3=Ctl Files  F5=Add Path  F24=More Keys
To locate a promotion path

Select one of the following methods to locate a promotion path:

- On a blank Manage Promotion Paths form, press Enter.

  The screen displays a complete list of promotion paths.

- On Manage Promotion Paths, enter the path name in the Promotion Path field.

  The screen displays the path name. If the promotion path does not exist, the screen displays the path name that is closest alphabetically.

To add a promotion path


2. Complete the Promotion Path form.

   - Add a new path name, a path description, and a release level.
   - Use the Code 1 through 5 fields for additional classifications.
     - Code 1 through 5 fields are user defined in system 92, types E1, E2, E3, E4, and E5.
     - Specify the status of the promotion path in the Code 4 field.
Field-sensitive help (function key F1) provides valid values for the Code 4 and 5 fields.

Specify the type of promotion environment in the Code 5 field.

To define a promotion path for source code members

1. Locate Promotion Path Members using one of the following methods:
   - On the Manage Promotion Paths form, locate the promotion path you want to define.
   - Enter 2 (Members) in the OP (Option) field next to the promotion path name.
   - On the Promotion Path form, press F10 (Members).

The From Environment area on the Promotion Path Members form shows the current locations of the source and object code. The To Environment area shows the locations to which the code will be moved.

2. Specify source files and library names for each member type you list on this screen.
   - To display valid member types and their descriptions, press F1 while your cursor is in a Mbr Type field. The member types are defined in the Function Codes user defined code table (98/FN).
   - To copy source file and library names from one member type to another, type 1 (Copy) in the OP (Option) field next to the member type you want to copy. Type 2 (Target) in the OP fields next to the
member types you want the information copied to, and press Enter. You can specify multiple targets.

The following chart shows some of the function keys available on this screen.

<table>
<thead>
<tr>
<th>KEY</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>F9</td>
<td>Redisplays the record for the previously-changed path.</td>
</tr>
<tr>
<td>F11</td>
<td>Displays the Promotion Path Control Files form.</td>
</tr>
<tr>
<td>F13</td>
<td>Displays the CASE Profiles form.</td>
</tr>
<tr>
<td>F14</td>
<td>Retrieves the source file, source library, and object library from your CASE profile and fills in the From environment. This overwrites any information currently in the fields.</td>
</tr>
<tr>
<td>F15</td>
<td>Duplicates the source file and library names from the first member type to the remaining member types.</td>
</tr>
</tbody>
</table>

What You Should Know About

**Copying an existing promotion path**  
If you copy an existing promotion path to create a new path, be sure that the source files and library names for the members are correct for the new path.

**Changing library names**  
To change library names, enter the new library names over the current ones.
To define a promotion path for control tables

1. Locate the Promotion Path Control Files using one of the following methods:
   - On Manage Promotion Paths, locate the promotion path you want to define and enter 3 (Ctl Files) in the OP (Option) field next to the promotion path name.
   - On the Promotion Path form, press F11 (Ctl Files).
   - On the Promotion Path Members form, press F11 (Ctrl Files).

The From Data Libr column on the Promotion Path Control Files form shows the current location of the data records. The To Control Lib column shows the location to which the data records will be moved.

2. Specify library names for each record type listed on this form.

The following chart shows some of the function keys available on this form.

<table>
<thead>
<tr>
<th>KEY</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>F9</td>
<td>Redisplays the record for the previously-changed project.</td>
</tr>
<tr>
<td>F10</td>
<td>Displays the Promotion Path Members form.</td>
</tr>
<tr>
<td>F13</td>
<td>Displays the CASE Profiles form.</td>
</tr>
</tbody>
</table>
### Guidelines

- If you copy an existing promotion path to create a new path, be sure the library names for the control files are correct for the new path.
- To change library names, enter the new library names over the current ones.

### Defining a Project

Complete the following tasks:

- Locate a project
- Add a project
- Assign promotion paths
- Assign project SARs

From the Version Control menu (G9261), choose Manage Projects.
To locate a project

Locate a project using one of the following methods:

- On a blank Manage Projects form, press Enter.
  A complete list of projects appears.
- On the Manage Projects form, enter the project name in the Project field.
  The project name appears on the form. If the project does not exist, the project name that is closest alphabetically appears on the form.
To add a project

1. On Manage Projects, choose Add Project.

2. On the Software Development Project form, do the following:
   - Enter a new project name.
   - Enter a project description.
   - Enter any other information you want to associate with the project.
   - Complete the Code 1 through 5 fields for additional classifications.
     - The Code 1 through 5 fields are user defined in system 92, types P1, P2, P3, P4, and P5.

3. If you want to attach this project to a parent project, specify the parent project name in the Parent Project field.
The following chart shows some of the function keys available on this form.

<table>
<thead>
<tr>
<th>KEY</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>F9</td>
<td>Redisplays the record for the previously-changed project.</td>
</tr>
<tr>
<td>F10</td>
<td>Displays the Project Promotion Paths form.</td>
</tr>
<tr>
<td>F11</td>
<td>Displays the Project Elements form.</td>
</tr>
<tr>
<td>F14</td>
<td>Displays the generic text associated with this project, and gives you access to text model selections.</td>
</tr>
</tbody>
</table>

You must assign promotion paths and SARs to the project you set up here. The following sections explain how to assign them.

### To assign promotion paths

1. Locate the project to which you want to assign promotion paths using one of the following methods.
   - On Manage Projects, locate the project to which you want to assign promotion paths.
   - In the OP (Option) field next to the project name, enter 2 (Paths)
   - On Software Development Project, press F10 (Promotion Paths).
2. Specify the promotion paths you want to assign to this project.
   - To display the available promotion paths, press F1 (Help) while the cursor is in a Promotion Path field.

The following chart shows some of the function keys available on this form.

<table>
<thead>
<tr>
<th>KEY</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>F9</td>
<td>Redisplays the record for the previously-changed project.</td>
</tr>
<tr>
<td>F11</td>
<td>Displays the Project Elements form.</td>
</tr>
</tbody>
</table>

The following chart shows options available on this form.

<table>
<thead>
<tr>
<th>OPTION</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Edit the promotion path details.</td>
</tr>
<tr>
<td>2</td>
<td>Edit the promotion path members.</td>
</tr>
<tr>
<td>3</td>
<td>Edit the promotion path control files.</td>
</tr>
</tbody>
</table>
To assign project SARs

SARs are elements of a project, however, other projects can also be elements of a project.

1. Access the Project Elements form using one of the following methods:
   - On the Version Control form, choose Edit and Promote.
   - On the Manage Projects form, locate the project to which you want to assign elements.
   - Enter 3 (SARs) in the OP (Option) field next to the project name.
   - On the Software Development Project form, press F11 (Project SARs).
   - On the Project Promotion Paths form, press F11 (Project SARs).

The Project Elements form displays the elements (usually SARs) assigned to the project.

2. Specify the elements (usually SARs) you want to assign to this project. You can also assign projects, which have SARs associated with them, as elements on this screen.
   - In the TY (Type) fields, specify the corresponding element types (S for SARs, and P for projects).
- In the TY (Type) fields, specify the corresponding element types (S for SARs, and P for projects). The following chart shows some of the function keys available on this screen.

<table>
<thead>
<tr>
<th>KEY</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>F9</td>
<td>Redisplays the record for the previously-changed project.</td>
</tr>
<tr>
<td>F10</td>
<td>Displays the Project Promotion Paths screen.</td>
</tr>
</tbody>
</table>

The following chart shows options available on this screen.

<table>
<thead>
<tr>
<th>OPTION</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Displays or edits the SAR detail.</td>
</tr>
<tr>
<td>2</td>
<td>Displays or edits the SAR log. The SAR Log Transfer screen appears, which lets you edit the SAR log and update the project SARs. For more information about updating the SARs by using this log, see Update the SARs in this publication. For information about the SAR log, refer to the Computer Assisted Design Reference Guide.</td>
</tr>
<tr>
<td>3</td>
<td>Displays the Pre-Promotion Edit History form. For information about this function, see Promote a SAR in this publication.</td>
</tr>
<tr>
<td>4</td>
<td>Promotes a project. For information about this function, see Promote a SAR in this publication.</td>
</tr>
<tr>
<td>5</td>
<td>Displays the promotion history of a SAR ('Z' record).</td>
</tr>
<tr>
<td>6</td>
<td>Displays or edits notes associated with a SAR (for J.D. Edwards environments only; 'w' record).</td>
</tr>
</tbody>
</table>
Promote a Project

Promoting a Project

After you create a project, link promotion paths and SARs to it, and complete project development, you are ready to begin the promotion process.

The SAR system uses the Work Order files (F4801 and F4802). If your production environment uses these files, and if the F4802 file has different record types than what version control needs, define a separate library that contains these files for version control purposes only.

Complete the following tasks:

☑ Update the SARs
☑ Validate a Promotion Path
☑ Promote a Project

See Also

- Defining a Promotion Path
### Update the SARs

The SARs, which are contained in the Work Order Header file (F4801), have detailed information in the Work Order Detail file (F4802). You must update the information in the SARs to reflect software developments that are recorded in the SAR log.

![Diagram](image-url)

When you developed the software, the changes were recorded in the SAR log automatically. You now must update the F4802 file.

From the Version Control menu (G9261), choose Edit and Promote.

![Edit and Promote](image-url)

**To update the SARs**

1. On the Edit and Promote form, inquire on the project you want to promote.
2. In the OP (Option) field next to the project SAR you want to update, enter 2 (SAR Log).
The SAR Log Transfer form appears, which lists all added or changed records logged in the SAR log (F9810) according to record type. The SAR Detail Sts field shows whether the record has been updated in the F4802 file.

### SAR Log Transfer

- **SAR Number**: 1079777
- **Multiple Jobs Submitted**: Yes
- **Record Type**:
  - M - Data Dictionary
  - G - Modified Source
  - C - Modified Source
  - F - Software Repository

<table>
<thead>
<tr>
<th>Record Type</th>
<th>Primary</th>
<th>Secondary</th>
<th>Data File</th>
<th>T</th>
<th>A</th>
<th>Library</th>
<th>R</th>
<th>C</th>
<th>SAR Detail Sts</th>
</tr>
</thead>
<tbody>
<tr>
<td>M - Data Dictionary 4888</td>
<td>JDFCTCL3</td>
<td>1</td>
<td>Updated</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M - Data Dictionary 4889</td>
<td>JDFCTCL3</td>
<td>1</td>
<td>Updated</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>G - Modified Source J9242S</td>
<td>JDFSRC73</td>
<td>1</td>
<td>Updated</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>G - Modified Source J98901B</td>
<td>JDFSRC73</td>
<td>1</td>
<td>Updated</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>G - Modified Source J98901T</td>
<td>JDFSRC73</td>
<td>1</td>
<td>Updated</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>G - Modified Source P924124</td>
<td>JDFSRC73</td>
<td>1</td>
<td>Updated</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>G - Modified Source P9242D</td>
<td>JDFSRC73</td>
<td>1</td>
<td>Updated</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>G - Modified Source P924210</td>
<td>JDFSRC73</td>
<td>1</td>
<td>Updated</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>G - Modified Source P98905</td>
<td>JDFSRC73</td>
<td>1</td>
<td>Updated</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>G - Modified Source P98907</td>
<td>JDFSRC73</td>
<td>1</td>
<td>Updated</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>F - Software Repository J9242S</td>
<td>JDFCTCL3</td>
<td>1</td>
<td>Updated</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>F - Software Repository P9242D</td>
<td>JDFCTCL3</td>
<td>1</td>
<td>Updated</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Opt: 1=Details  9=Delete  F5=Add  F10=Update SAR  F24=More Keys**

If this form lists many SAR log records, you can narrow your search by entering information in the Record Type and the and/or Member fields.

To view the details of a SAR log item, enter 1 (Details) in the OP (Option) field next to the item. On the form that appears, you can edit the SAR details. If the SAR logging system does not log an item that you want to include, press F5 (Add) from the SAR Log Transfer form to add it.

3. To update the F4802 file, press F10 (Update SAR).

Before you update a SAR, verify that each SAR log record should be transferred with the SAR. Change or delete those that are associated incorrectly with the SAR. To display all records with data that can be transferred (TR field value is 1) or with test data (TR field value is 0), press F16 (Display Update Capable/All Items). Update only those records that should be transferred with this SAR.

The system creates or updates the records in the SAR file that is located in the SAR library you indicated in the Edit and Promote processing options (not the SAR library appearing in your library list).
Validating a Promotion Path

Before you promote a SAR, you must perform a pre-promotion edit, or validation, against the promotion path that will be used for this SAR.

To validate a promotion path

1. From the Edit and Promote form, inquire on the project you want to promote.
2. In the Promotion Path field, type the name of the promotion path you want to use for your project.
3. In the OP (Option) field next to the project SAR you want to update, enter 3 (Edit).

If you did not choose a promotion path for the project, the Project Promotion Paths form lists all promotion paths defined for the project.

<table>
<thead>
<tr>
<th>Path Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>A62CUM</td>
<td>Transfer to current A62 Cum</td>
</tr>
<tr>
<td>A62PC00UTI</td>
<td>User based pricing</td>
</tr>
<tr>
<td>A71CUM</td>
<td>Transfer to current A71 Cum</td>
</tr>
<tr>
<td>A72CUM</td>
<td>Transfer to current A72 Cum</td>
</tr>
<tr>
<td>JDF62</td>
<td>Transfer to JDF62</td>
</tr>
<tr>
<td>JDF62TEC</td>
<td>Transfer to JDF62-SECURE</td>
</tr>
<tr>
<td>JDF71</td>
<td>Transfer to JDF71</td>
</tr>
<tr>
<td>JDF71TEC</td>
<td>Transfer to JDF71-SECURE</td>
</tr>
</tbody>
</table>

Opt: 4=Select  F24=More Keys

4. In the O (Option) field next to the promotion path you use to promote the project, enter 4 (Select). If you have run pre-promotion edits previously for this SAR, the Pre-Promotion Edit History form lists them. Otherwise, this form is blank.
5. To view the errors associated with a pre-promotion edit, enter 1 (Details) in the OP (Option) field next to the desired history record.

6. From the Pre-Promotion Edit History form, press F5 (Perform Edit) to perform the pre-promotion edit.

7. Correct any errors and perform the edit until no errors occur. You do not need to resolve warnings that can occur.

The following table shows a partial list of errors and how to resolve them.
<table>
<thead>
<tr>
<th>Error Code</th>
<th>Cause and Resolution</th>
</tr>
</thead>
</table>
| 0020       | **Cause:** You entered a “From” library that does not exist or you are not authorized to use.  
**Resolution:** Either correct the library name, create the library, or get authorization to use it. |
| 0092       | **Cause:** A database table or member could not be opened because it did not exist, a conflicting lock state held by another job exists, or you are not authorized to open it.  
**Resolution:** Check your job log messages. |
| 1046       | **Cause:** An XJDE or ZJDE version was expected but not found.  
**Resolution:** If an XJDE or ZJDE version should exist, create it. If not, then change the processing option for form ID P926304. |
| 1370       | **Cause:** You entered a “From” table that does not exist or you are not authorized to use.  
**Resolution:** Review the “From” library for the promotion path control table. Either correct the library name or create the table. |
| 1371       | **Cause:** You entered a “To” table that does not exist or you are not authorized to use.  
**Resolution:** Review the “To” library for the promotion path control table. Either correct the library name or create the table. |
| 1372       | **Cause:** A key you wanted to copy from the “From” library does not exist.  
**Resolution:** Review the “From” library for the promotion path control table. Either correct the library name or re-enter the data record. |
| 2892       | **Cause:** A “From” library name is the same as the corresponding “To” library name.  
**Resolution:** Review the “From” and “To” libraries for the promotion path control table. Make the appropriate changes. |
4395  
**Cause:** No records exist in the Promotion Path Members table (F92401) for the promotion path you specified.

**Resolution:** Complete the Promotion Path Members form for the promotion path.

4396  
**Cause:** No records exist in the Promotion Path Members table (F92401) for the promotion path you specified.

**Resolution:** Complete the Promotion Path Control Files form for the promotion path.

4397  
**Cause:** No records exist in the SAR Log table (F9810) for the project you specified.

**Resolution:** In the project master record, change the based-on table for the Pre-Promotion Edit to the SAR Detail table (F4802), then manually update the SAR Detail records for the members and control table records updated by this project.

4400  
**Cause:** No record exists in the Promotion Path Members table (F92401) for the function code of the member you want to promote.

**Resolution:** For the specified promotion path, enter the environment for the function code of the member.

4402  
**Cause:** No record exists in the Promotion Path Control Files table (F92402) for the control table you want to promote.

**Resolution:** For the specified promotion path, enter the environment for the control table of the record.

4439  
**Cause:** An error occurred while you attempted to copy a source code member.

**Resolution:** Check for valid library, table, and member names, as well as options in the CPYF command. Check the job log for the error message ID.
Promoting a Project

The promotion process involves transferring members and copying control file data.

Before You Begin

☐ Before you promote the project, be sure you have edited all items that appear on the SAR Log Transfer screen. Otherwise, the SAR Log Transfer screen appears when you attempt to promote the project.

☐ You must update all SAR log records associated with the SAR before you promote it.

☐ You also must resolve all errors (not warnings) before you promote the SAR.

To promote a project

1. On the Promote a Project form, inquire on the project you want to promote.
2. In the OP (Option) fields next to the project elements you want to promote, enter 4 (Promote).

    To select all project elements automatically for promotion, press F14.

3. On the Project Promotion Paths form, enter 4 (Select) in the O (Option) field.
Software Transfer

Project: TEC Tech Foundation Corrections
SAR Number: Multiple Jobs Submitted
Promotion Path: JDF73 Transfer to JDF73
Release: A73

Errors: 000  Warnings: 006

<table>
<thead>
<tr>
<th>O</th>
<th>Member ID</th>
<th>Src File</th>
<th>Src Libr</th>
<th>Obj Libr</th>
<th>Src File</th>
<th>Src Libr</th>
<th>Obj Libr</th>
</tr>
</thead>
<tbody>
<tr>
<td>J924147</td>
<td>JDESRC</td>
<td>PGFSRC73</td>
<td>PGFOBJ73</td>
<td>JDESRC</td>
<td>JDFSRC73</td>
<td>JDFOBJ73</td>
<td></td>
</tr>
<tr>
<td>P92402</td>
<td>JDESRC</td>
<td>PGFSRC73</td>
<td>PGFOBJ73</td>
<td>JDESRC</td>
<td>JDFSRC73</td>
<td>JDFOBJ73</td>
<td></td>
</tr>
<tr>
<td>P924124</td>
<td>JDESRC</td>
<td>PGFSRC73</td>
<td>PGFOBJ73</td>
<td>JDESRC</td>
<td>JDFSRC73</td>
<td>JDFOBJ73</td>
<td></td>
</tr>
<tr>
<td>P924127</td>
<td>JDESRC</td>
<td>PGFSRC73</td>
<td>PGFOBJ73</td>
<td>JDESRC</td>
<td>JDFSRC73</td>
<td>JDFOBJ73</td>
<td></td>
</tr>
<tr>
<td>V92402</td>
<td>JDESRC</td>
<td>PGFSRC73</td>
<td>PGFOBJ73</td>
<td>JDESRC</td>
<td>JDFSRC73</td>
<td>JDFOBJ73</td>
<td></td>
</tr>
</tbody>
</table>

Opt: 1=Src & Obj 2=Src 3=Obj F4=More F5=Ctl Files F6=Override F15=Edit Hist

4. In the OP (Option) fields next to the member IDs, specify whether to transfer:
   - Both source and object code (option 1)
   - Source code only (option 2)
   - Object code only (option 3)

To override the From Environment and To Environment object libraries before you transfer the members, press F6 before you enter options 1, 2, or 3. Enter the names of the object libraries to which you want the members transferred.

The system transfers the members you selected to the target environment.

You can review the batch job that was submitted by this transfer program from the J.D. Edwards command line. To display the command line, press F2.

If your promotion is successful, the system deletes all SAR log records for transferred items. It also creates a new SAR log record for each transferred item and associates it with the target library.

5. To copy control file data, press F5 (Control Files) from the Software Transfer form.
In the OP (Option) fields next to the items you want to copy, enter 1 (Copy to target library).

NOTE: Press F13 to select all items automatically for copying.

The system copies the items you selected to the target environment.
Promote Project Updates

Promoting Project Updates

The version control process for project updates includes the following general steps.

- Create the transfer library
- Prepare the SAR system
- Define promotion paths
- Define a project
- Update the SARs
- Validate the promotion path
- Promote a SAR
- Save the transfer library to tape
- Restore the transfer library from tape
- Print the transfer library report
- Load the transfer library
- Transfer individual control table records
To create the transfer library

1. From the Software Install menu (G9262), choose Build Transfer Library.

   WARNING!!!

   If you specify a library that already exists on your system to be used
   as a software transfer library it will be cleared prior to use.
   All data and objects in that library will be lost.

   If the library you specify does not exist it will be created for you.

   ( F6 - Execute )

2. After you read the warning message, press F6 (Execute).

3. In the processing option field, enter a name for the transfer library you
   want to create.
To prepare the SAR system

To prepare your SAR system, see *Prepare the SAR System* in this guide.

To define promotion paths

From the Version Control menu (G9261), choose Manage Promotion Paths. Use the project update library name as your promotion path name. For information about defining a promotion path, see *Define Promotion Paths* in this guide.

To define a project

To define a project, see *Define a Project* in this guide.

To update the SARs

From the Software Install menu (G9262), choose Edit and Promote. For information about updating a SAR, see *Update the SARs* in *Promote a SAR* in this guide.

To validate the promotion path

From the Software Install menu (G9262), choose Edit and Promote. For information about validating a promotion path, see *Validate the Promotion Path* in *Promote a SAR* in this guide.

To promote the project

From the Software Install menu (G9262), choose Edit and Promote. For information about promoting a SAR, see *Promote the Project* in *Promote a SAR* in this guide.

To save the transfer library to tape

1. From the Software Install menu (G9262), choose Save Library to Tape.
Save Library (SAVLIB)

Type choices, press Enter.

Library . . . . . . . . . . . . MYLIBRARY Name, *NONSYS, *ALLUSR, *IBM
+ for more values
Device . . . . . . . . . . . . TAP01 Name, *SAVF
+ for more values

2. In the Library field, type the name of your transfer library.
3. In the Device field, enter the name of your tape device.

To restore the transfer library from tape

1. From the Software Install menu (G9262), choose Restore Library from Tape.

Restore Library (RSTLIB)

Type choices, press Enter.

Saved library . . . . . . . . . . MYLIBRARY Name, *NONSYS, *ALLUSR, *IBM
 Device . . . . . . . . . . . . TAP01 Name, *SAVF
 + for more values

2. In the Saved Library field, type the name of your transfer library.
3. In the Device field, enter the name of your tape device.

**To print the transfer library report**

1. From the Software Install menu (G9262), choose Print Transfer Report.

   A processing options form appears. Use the cursor keys to display additional processing options.

   98312  Print Transfer report  Form ID. . . . P924143  
   Control File Changes to be Installed  
   Version. . . . ZJDE0001  
   Display Level. 4  
   This job has various options described below. Enter the desired values and press ENTER to continue.

   Enter name of Transfer Library. MYLIBRARY  
   Print UDCs 1=Yes, 0=No. 1  
   Print AAIs 1=Yes, 0=No. 1  
   Print Menus 1=Yes, 0=No. 1  
   Print Data Dictionary 1=Yes, 0=No. 1  
   Print Vocabulary/Exits 1=Yes, 0=No. 1  
   Print CASE specs 1=Yes, 0=No. 1  
   More... +  
   F5=Printer Overrides  

2. In the first processing option field, type the name of your transfer library.
This job has various options described below. Enter the desired values and press ENTER to continue.

Print Helps 1=Yes, 0=No.
Print SVR 1=Yes, 0=No.
Print DREAMWriter 1=Yes, 0=No.
Print Next Numbers 1=Yes, 0=No.

Bottom.

F5=Printer Overrides

3. In the remaining processing option fields, select the types of control files for which you want to print information.

4. To print the report, press Enter.
An example of the Print Install Records report (P924143) follows. It provides the total number of records for each type of control file. It also shows whether the transfer record already exists in your control file.

To load the transfer library

Before you load the transfer library, you must create new target libraries for the objects, source code, and data files you will transfer. In your target source library, you must create the following multi-member source files:

- JDESRC
- JDECPY
- F98CRTCMD

Load the contents of your transfer library into your target libraries. The process merges control file records into your library files. You also can transfer control file records individually. For more information, see Transfer Individual Control File Records following this procedure.

1. From the Software Install menu (G9262), choose Load Transferred Library.
WARNING!!!

This program will transfer source code, objects and new data files into the libraries you name in the processing options. It will also add to or replace data in the control files in your current library list.

It is recommended that you first run the 'Print Transfer Report' to view control file changes.

(F6 - Execute)

2. After you read the warning message, press F6 (Execute).

A processing options form appears. Use the cursor keys to display additional processing options.

98312 Load Transferred Library Form ID. . . . P924147
Load Transfer Software Version. . . . ZJDE0001
Display Level. 4
This job has various options described below. Enter the desired values and press ENTER to continue.

Enter name of Transfer Library or blank for no transfer. MYLIBRARY
Enter name of Target Object Library or blank for no transfer. MYOBJ
Enter Name of Target Source Library or blank for no transfer. MYSRC
Enter Name of Target New Files Library or blank for no transfer. MYDATA

More...
F5=Printer Overrides

3. In the first processing option field, specify the name of your transfer library.

4. In the next three fields, specify the libraries you created for the source code, objects, and data files you will transfer.
This job has various options described below. Enter the desired values and press ENTER to continue.

<table>
<thead>
<tr>
<th>Option</th>
<th>Yes/No</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transfer UDCs</td>
<td>1=Yes, 0=No</td>
<td>0</td>
</tr>
<tr>
<td>Transfer AAIs</td>
<td>1=Yes, 0=No</td>
<td>0</td>
</tr>
<tr>
<td>Transfer Menus</td>
<td>1=Yes, 0=No</td>
<td>0</td>
</tr>
<tr>
<td>Transfer Data Dictionary</td>
<td>1=Yes, 0=No</td>
<td>1</td>
</tr>
<tr>
<td>Transfer Vocabulary/Exits</td>
<td>1=Yes, 0=No</td>
<td>0</td>
</tr>
<tr>
<td>Transfer CASE specs</td>
<td>1=Yes, 0=No</td>
<td>0</td>
</tr>
<tr>
<td>Transfer Helps</td>
<td>1=Yes, 0=No</td>
<td>0</td>
</tr>
</tbody>
</table>

More...

F5=Printer Overrides

5. In the remaining fields, select the control files you want the system to transfer.

6. To begin the transfer, press Enter.
The system merges the control files into the target data library. For non-control files, the system adds the file if it currently does not exist in the target data library. If the file does exist in the target data library, the system does not transfer the file or any data. After the transfer process completes, you must change these files manually based on information in the Print Install Records report (P924145).

Even though you can include next numbers in the transfer library and display information about them in the Print Install Records report, the system will not transfer them automatically. This protects your next number tables. After the transfer process completes, you must change them manually based on information in the report.

**To transfer individual control table records**

1. From the Software Install menu (G9262), choose Copy DD,VO,DW,UDC,SVR,Menus.

2. In the From Library, type the name of your transfer library.

3. In the To Library, type the name of the target data file library.

4. In the appropriate fields, enter information that is specific to the control file record you want to transfer.
Programming Tools

Objectives

- To work with data modeling
- To understand the Software Versions Repository
- To set up user defined values
- To retrieve information
- To create data description specifications
- To design and maintain display forms
- To design reports

About Programming Tools

Perform the following tasks:

- Work with Data Modeling
- Work with the Object Cross Reference Repository
- Work with Data Dictionary
- Work with Data File Design Aid
- Work with Screen Design Aid
- Work with Report Design Aid
**Work with Data Modeling**

**Working with Data Modeling**

The Data Modeling feature provides graphic representation of the relationships of different files. The important aspects of the J.D. Edwards Data Modeling feature are:

- It is graphical in its presentation.
- It allows you to narrow the amount of information you view so you can better analyze the file and data relationships.
- It is integrated back to the Data Dictionary and other cross reference tools.

To create a data model, you must run the Data Model rebuild.
Accessing Data Modeling

There are two ways to access Data Modeling.

**To access data modeling**

1. Use one of the following two methods to access Data Modeling.
   - Inquire on the file through the Software Versions Repository and then press F23
   - From the Model Relations form, select Data Modeling

The Data Model Diagrammer displays models from Base Files stored in the Entity Relationship Tracking file (F9804). When using the Data Model Diagrammer for the first time, rebuild the Cross Reference Index of the menu G9642. This rebuild will create data in the Entity Relationship Tracking file and allow file relationships to be built.

( F6 – Execute )

A menu message form appears with sample data you can use to view a supplied data model.

2. Press F6 to continue.
The Data Modeling form appears with the cursor positioned in the field where you enter a file name.

<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max Levels</td>
<td>Determines what level of detail you want to view in terms of file relationships. Level 1 represents the highest level and level 10 represents the lowest level. The default value is level 08. Level 01 shows only those files that are directly related to the data model file.</td>
</tr>
<tr>
<td>Function Use</td>
<td>Displays the files that either match or have a function use less than the function use you specify.</td>
</tr>
<tr>
<td>Display Duplicate</td>
<td>Determines whether you want to display duplicate relationships or not. The valid values are:</td>
</tr>
<tr>
<td>Relationships</td>
<td>1 – no duplicates (default value)</td>
</tr>
<tr>
<td></td>
<td>2 – first logical only</td>
</tr>
<tr>
<td></td>
<td>3 – all files</td>
</tr>
<tr>
<td>In Sys</td>
<td>Limits your model to only those files from the specified install or reporting system codes. To toggle to reporting system codes, you press F11, Install/Reporting.</td>
</tr>
</tbody>
</table>

3. To view the Data Model, enter a file name and press Enter.
4. To narrow the amount of file information displayed, specify values in the four fields appearing in the upper right of the form.
**Detailed Explanation of a Line**

The following figure shows a portion of the Data Modeling form.

![Diagram of Base File and Business Unit Master](image)

Below is an explanation of the components displayed on the form.

- Business Unit Master is the primary file (F0006)
- Company Constants is the secondary file (F0010)
- <m:1> – There is a many to one, bi-directional relation between the files

<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quantifier</td>
<td>The quantifier notation indicates the following:</td>
</tr>
<tr>
<td></td>
<td>M:1    many to one</td>
</tr>
<tr>
<td></td>
<td>1:M    one to many</td>
</tr>
<tr>
<td></td>
<td>M:M    many to many</td>
</tr>
<tr>
<td></td>
<td>M:N    many to zero or many</td>
</tr>
<tr>
<td></td>
<td>N:M    zero or many to many</td>
</tr>
<tr>
<td></td>
<td>1:N    one to zero or many</td>
</tr>
<tr>
<td></td>
<td>1:1    one to one</td>
</tr>
<tr>
<td>Direction</td>
<td>The three direction notations are as follows:</td>
</tr>
<tr>
<td></td>
<td>-&gt;  refers to</td>
</tr>
<tr>
<td></td>
<td>&lt;-  referred to</td>
</tr>
<tr>
<td></td>
<td>&lt;-&gt;  2 way relation</td>
</tr>
<tr>
<td>Type</td>
<td>Used to distinguish between prototype and permanent files.</td>
</tr>
<tr>
<td>Subfile portion of screen</td>
<td>Displays the key fields that relate these two files together</td>
</tr>
</tbody>
</table>
**Function Key Exits from Data Modeling**

**Install/Reporting**

F11 – Allows you to toggle between displaying install or reporting system codes.

**Rebuild A File Relationship**

F16 – Rebuilds a data model.

Exits to a DREAM Writer versions list.

The rebuild is fundamentally based upon the program finding a connection between data items.

For example, if you create new data items in the Data Dictionary and use those data items when creating a new file, you do not get a graphic representation for that file because the data items do not exist in any other file. To create and present file relationships, there must be at least one data item in the primary file that also resides in some other file as well.

**Selection Exits from Data Modeling**

**Selection 1 - Move Top**

To select a file in the current data model and move it to the top to view its data model.

**Selection 5 - Display**

To view the file relationships. The Define a File Relationship form appears displaying the relationship detail for the two files.

**Selection 7 - Where Used**

Exits to the Object Cross Reference Repository and displays all the programs that access the particular file.
Selection 8 - Fields

To access the File Field Description form for any file displayed in the Data Model. The File Field Description form presents all the fields in a file, the field type, their size and their position in the file.
Work with the Object Cross Reference Repository

Working with the Object Cross Reference Repository

The Object Cross Reference Repository locates all the objects associated with a particular member or object. When you add a new member to the Software Versions Repository, run the Rebuild Cross Reference job to have the new member included in the display. You must have source code on your machine to run this rebuild and display this option.

Complete the following tasks:

☐ Access the Object Cross Reference Repository

☐ Conduct an Object Cross Reference Repository search

Accessing the Object Cross Reference Repository

▶ To access the Object Cross Reference Repository

Select one of the following methods.

- From the Master Directory, choose the Technical and Advanced Operations menu. From the Technical and Advanced Operations menu (G9), select Documentation Services. From the Documentation Services menu (G91), choose Object Cross Reference Repository.

- On Software Version Repository, press F15 to access the Object Cross Reference Repository.
Example

The following form displays all programs using the file F0006.

The first four fields on this form relate to the object being cross referenced. The remainder of the form lists the members found during the cross reference search.

```
980014
Object Cross Ref. Repository
Object: Name . . .  F0006    Business Unit Master
       Type . . .  F         All programs using file
       To Display  P
       Funct Cd .  ____

<table>
<thead>
<tr>
<th>O</th>
<th>Name</th>
<th>Description</th>
<th>Field Attr</th>
<th>T</th>
<th>Start Upd</th>
</tr>
</thead>
<tbody>
<tr>
<td>P</td>
<td>P000661</td>
<td>Business Unit Master Conversion</td>
<td>Y</td>
<td></td>
<td></td>
</tr>
<tr>
<td>P</td>
<td>P06238</td>
<td>Report - Payroll Check Register</td>
<td>N</td>
<td></td>
<td></td>
</tr>
<tr>
<td>P</td>
<td>P06371</td>
<td>Report - Certified Payroll Register</td>
<td>N</td>
<td></td>
<td></td>
</tr>
<tr>
<td>P</td>
<td>P06611I</td>
<td>Tip Credit Generation with Interim Check</td>
<td>N</td>
<td></td>
<td></td>
</tr>
<tr>
<td>P</td>
<td>P06638</td>
<td>Sales Allocation Report</td>
<td>N</td>
<td></td>
<td></td>
</tr>
<tr>
<td>P</td>
<td>P126410</td>
<td>STAR - Columnar Spreadsheet</td>
<td>N</td>
<td></td>
<td></td>
</tr>
<tr>
<td>P</td>
<td>P200</td>
<td>Submit Network Job</td>
<td>N</td>
<td></td>
<td></td>
</tr>
<tr>
<td>P</td>
<td>P233250</td>
<td>Texas 250 Report</td>
<td>N</td>
<td></td>
<td></td>
</tr>
<tr>
<td>P</td>
<td>P26011I</td>
<td>Gas Balancing - Entitlement Extract</td>
<td>N</td>
<td></td>
<td></td>
</tr>
<tr>
<td>P</td>
<td>P26112</td>
<td>Gas Balancing - Entitlement Explosion to</td>
<td>N</td>
<td></td>
<td></td>
</tr>
<tr>
<td>P</td>
<td>P26115</td>
<td>Gas Balancing Statement by Sales Point</td>
<td>N</td>
<td></td>
<td></td>
</tr>
<tr>
<td>P</td>
<td>P26116</td>
<td>Gas Balancing Statement by Owner</td>
<td>N</td>
<td></td>
<td></td>
</tr>
<tr>
<td>P</td>
<td>P26119</td>
<td>Gas Balancing Master Subfile Display</td>
<td>N</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Opt:  1=SVR  2=Create Object  3=Field Expl  F21=Print  F16=Regenerate
```

Conducting an Object Cross Reference Repository Search

All members of the Software Versions Repository are cross referenced, and you can search for these relationships in different ways.

**To conduct an Object Cross Reference Repository search**

From the Documentation Services menu (G91), choose Object Cross Ref. Repository.

The following form displays the statistics for program P0006.
1. To conduct an object cross reference repository search, enter an object Name, Type code and To Display code. To narrow the search, enter a Funct Cd.
   - If you are unfamiliar with the Cross Reference Relationships codes, type an asterisk (*) in the Type field, as shown below. Press Enter.

2. Enter 4 in the single character field to the left of the desired code. The Object Cross Ref. Repository form displays with the selected codes.
Work with Data Dictionary

About the Data Dictionary Repository

The Data Dictionary is the most powerful element in all of J.D. Edwards’ software offerings. We define all data items used by J.D. Edwards programs in the Data Dictionary. By requiring this up-front definition, the Data Dictionary enforces uniformity, consistency, and accuracy across all J.D. Edwards applications.

The Data Dictionary represents a centralized glossary of all:

- Field definitions
- Program error messages, both interactive and batch
- Menu messages
- Work fields
- User defined help instructions
- Program and field descriptions accessed by the Help facility

Complete the following tasks:

- [ ] Understand the Data Dictionary structure
- [ ] Locate a data item name
- [ ] Work with the Data Dictionary
- [ ] Work with data item alias revisions
- [ ] Work with Data Dictionary glossary
- [ ] Work with user defined help instructions
- [ ] Work with data field descriptions
- [ ] Work with the next numbers facility
- [ ] Locate the field reference rebuild
Understanding the Data Dictionary Structure

The following files comprise the Data Dictionary Repository.

The following diagram illustrates the relationships between these files.

Data Item Master (F9200)

This is the master file for the Data Dictionary. Every data item has a record in this file.

Data Field Specifications (F9201)

This file contains database fields, which is a glossary group of “D” or “S,” work fields, glossary group “U,” and categories, glossary group “C.” This file contains the base display and validation rules for all file and data items.
**Data Field Display Text (F9202)**

This file lets you define multiple row descriptions and column titles for each data item, based upon language or reporting system (application override). You can add a language value for each language translation required for the row description and column title. The reporting system code allows the entry of jargon or company terminology that overrides the generic text supplied with the application.

**Data Item Alpha Descriptions (F9203)**

This file contains the alpha and compressed descriptions for all data items. This allows you to perform a Data Dictionary search by description. You can also specify separate alpha descriptions by language preference and reporting system. Every data item has a record in this file.

**Data Item Aliases (F9204)**

This file only contains database fields, which is a glossary group of “D” or “S”. This file contains multiple aliases for both a COBOL alias and a C alias for each data item.

**Error Message Program ID (F9205)**

This file contains error messages that have a program, form, or report ID attached to them. You exit to this program, form, or report when you receive the error. For example, if you receive a user defined code error, you could exit to the User Defined Code Revisions program to modify a value.

**Glossary Text File (F9816)**

This file contains the glossary text for every data item. Each line of text in the glossary is one record.

**Key Index File (F98163)**

This file contains key information to link the data items to their glossary and to specific items.
Locating A Data Item Name

The system uses data items to define the parameters of a field or message. For example, AT1 defines the field Search Type. The system maintains each data item used in a file or retrieved for a form or report based on a data item name, such as AT1. To work with the Data Dictionary functions you need to know this name.

▶ To locate a data item name

The J.D. Edwards field-level help displays data item names.

Position the cursor on any field and press F1.

For example, position the cursor in the Search Type field on the Address Book Revisions form and press F1. The User Defined Codes form displays for the Search Type field. In the upper right corner of this form is the data item name for the Search Type field, which is AT1.

The data item name is always in the upper right corner of the help form, no matter which help form displays, such as the User Defined Codes form or the field explanation form.
Working with the Data Dictionary

The Data Dictionary provides many useful abilities. You can create data item aliases for other programming languages, work with the glossary, add or change user defined help instructions, and locate data field descriptions.

To work with the Data Dictionary

From menu G92, choose Data Dictionary. The Data Dictionary form is displayed.

You find the Data Dictionary selection on several J.D. Edwards menus and repository services.

You can also display the Data Dictionary form by entering the mnemonic DD in the Selection line of any J.D. Edwards menu.

Use the following fields where applicable:

<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rls Last Chg</td>
<td>The software version number to be defaulted in the Software Versions Repository file.</td>
</tr>
<tr>
<td>Item Parent</td>
<td>Display only. A data item which becomes the template from which other data items are created. For example, AC (Category Codes) is the parent to AC01.</td>
</tr>
<tr>
<td>Field</td>
<td>Explanation</td>
</tr>
<tr>
<td>-----------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Data Item</td>
<td>The RPG data name. This data field has been set up as a 10-byte field for future use. Currently, it is restricted to 4 bytes so that, when preceded by a 2-byte file prefix, the RPG data name does not exceed 6 bytes. Within the Data Dictionary, all data items are referenced by this 4-byte data name. As they are used in database tables, a 2-character prefix is added to create unique data names in each table specification (DDS). Special characters are not allowed as part of the data item name, with the exception of #, @, $. You can create protected data names by using $xxx and @xxx, where you define xxx. Messages can contain up to 10 characters. Types of messages are further defined by glossary group.</td>
</tr>
<tr>
<td>Glossary Group</td>
<td>Differentiates data items into types. These types include primary and secondary types, error messages, and help text. See UDC 98/GG for a complete listing of Glossary Groups. See also 'What Are the Data Dictionary Glossary Groups?' within this 'Data Dictionary Repository' chapter.</td>
</tr>
<tr>
<td>Alpha Desc</td>
<td>Database text string that names the data item. Enter text in upper and lower case. The system uses this field to search for similar data items. To enter an alpha description, follow these conventions: Dates – Begin all Date fields with Date Amounts – Begin all Amount fields with Amount Units – Begin all Unit, Quantity, and Volume fields with Units Name – Begin all 30-byte description fields with Name Prompt – Begin any Y/N prompting field with Prompt Address Number – Begin all address numbers (employee, customer, owner) with Address Number</td>
</tr>
<tr>
<td>Reporting System .ode</td>
<td>Designates the system number for reporting purposes. This rarely differs from the Install System. Exceptions occur for data files used by more than one system</td>
</tr>
<tr>
<td>System Code</td>
<td>The system code and type of the table to be copied. All values for the specified table will be copied.</td>
</tr>
<tr>
<td>Field</td>
<td>Explanation</td>
</tr>
<tr>
<td>-----------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Type</td>
<td>The Data Dictionary name of the field or the record format name.</td>
</tr>
<tr>
<td></td>
<td>• The file prefix is added to create unique data names for each file specification if a data item is entered in this field.</td>
</tr>
<tr>
<td></td>
<td>• The record format line is automatically defaulted in.</td>
</tr>
</tbody>
</table>

**Form-specific information**

Note: When using the “O” format, create the field as large as possible. This allows the use of ideographic languages such as Japanese.

| Size                  | The description of the data item entered in the previous field.                                                                                  |
|                       | • Comes from the Row Description field in the Data Dictionary.                                                                                   |

| Data File Decimals    | The number of positions to the right of the decimal of the data item.                                                                            |
| Data Item Class       | Defines the essential attributes and characteristics of a data item.                                                                            |

| Item Occurrences      | In setting up a data item in the data dictionary, you may specify a number of array elements. This will cause the automatic creation of one additional data item for each array element. |
|                       | The array data item names are restricted to certain lengths depending on the number of array elements: 3 bytes – 1 to 9 elements, 2 bytes – 10 to 99 elements, 1 byte – 100 to 999 elements |

| Display Decimals      | Use this parameter to designate the number of decimals in the currency, amount, or quantity fields the system displays. For example, U.S. Dollars would be 2 decimals, Japanese Yen would be no decimals, and Cameroon Francs would be 3 decimals. |

<p>| Row Description       | Creates the title on text and reports. It is used in a manner similar to the column description in the query facility. It should be less than 35 characters. Use abbreviations whenever possible. For example: |
|                       | U/M Units of measure                                                                                 |
|                       | YTD Year-to-date                                                                                     |
|                       | MTD Month-to-date                                                                                     |
|                       | PYE Prior year end                                                                                    |
|                       | QTY Quantity                                                                                         |
|                       | G/L General ledger                                                                                    |
|                       | A/P Accounts payable                                                                                  |
|                       | DEPR Depreciation                                                                                    |</p>
<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Column Title</td>
<td>The first line of description that will be used in column headings on a report or form. This description should be no larger than the data item size, if possible. If the column heading is only one line, it should be placed in this column. Use the second line of the Column Title when one is not clear.</td>
</tr>
<tr>
<td>Default Value</td>
<td>Used as the initial value on the data entry screen for the associated data item. The value entered must be the exact same length as the data item size. Place single quotes around the value if it contains any embedded blanks. The keywords *BLANKS and *ZEROS can be used as the default value. When entering a numeric data item with default values, the redisplay of the data item suppresses all leading zeros. CAUTION: If a blank entry is allowed, default values should not be used.</td>
</tr>
<tr>
<td>Data Display Rules</td>
<td>Keywords which describe an editing technique applied when data is displayed. Validation is applied to the data after Enter is pressed. The rule will be applied as specified in the F9207 table at the screen/report and/or the action code if desired. The developer can override these rules at the time of program creation. The current list of these rules is kept in the User Defined Codes at SYSTEM = 98 and RECORD TYPE = ER.</td>
</tr>
<tr>
<td>Data Edit Rules</td>
<td>Keywords which describe an editing technique applied when data is entered. Validation applied to the data after Enter is pressed. The rule will be applied as specified in the F9207 table at the screen/report and/or the action code as desired. The developer can override these rules at the time of program creation. The current list of these rules is kept in the User Defined Codes at SYSTEM = 98 and RECORD TYPE = ER.</td>
</tr>
<tr>
<td>Field</td>
<td>Explanation</td>
</tr>
<tr>
<td>---------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| Search Program| The Help Text Program field is used to call a program when the function key - F1 is pressed on its Data Item. When F1 is pressed, the program entered in this field will be executed. If this field is left blank, the glossary will be used. If you wish the User Defined Code window to appear when F1 is pressed, enter "UCD" in this field (this is the default when 'UCD' is entered in the Data Edit Rules field). If you do not want the UDC window to appear and you have 'UCD' in the Data Edit Rules field, change this field to be blank. Program Requirements: For your text program to work correctly, you must allow it to accept three standard parameters:  
  - PARM 1  
    Field Name, size 10, type alpha  
  - PARM 2  
    Return Value, size 30, type alpha  
  - PARM 3  
    Return Description, size 30, type alpha |
| Next Nbr System | Designates the system number for the Next Number retrieval. See User Defined Codes, system code '98', record type 'SY'. |
| Next Number Index | The array element number retrieved in the Next Number Revisions program. For example, the next voucher number is array element '02' of system '04'. |
What You Should Know About

Data Dictionary Security Once a system is operational, you must be particularly careful to secure the integrity of the Data Dictionary. Two facilities are provided to aid you with the security:

- Operational systems coding - System numbers and names are defined in User Defined Codes, system code 98, record type SY. If you place an X in the second line of description for a particular system, it will be designated as operational. Once a system has been set up as operational, all data fields coded to this system are protected from modifications. This control, however, can be violated by removing the X in User Defined Codes.
- Action Code Security - A more prudent form of control is to assign change/delete authority to only one individual, the database administrator. If you choose to use this control, you should restrict access to the Data Dictionary program (P9201) in Action Code Security. See Working with Action Code Security. All users must be set up with add authority only. The database administrator would be set up with add/change/delete authority.

The Function Keys for the Data Dictionary

The following function keys are available from the Data Dictionary form.

- F4. A data item search facility. If you are a double-byte user, you must provide a search description for each data item you create or change in order for the search facility to function properly. Enter the search text in the Search Description field on the Data Dictionary screen.
- F6. Repository Services
- F8. User Defined Code Tables
- F9. Automatic Requery
- F15. A data item cross reference
Working with Data Item Alias Revisions

Use the Data Item Alias form to assign alias names to a data item that other programming languages use. When adding a data item of glossary group “D” or “S”, you must enter an alias for that field. This form automatically appears on an Add function when the alias is not unique. The alias defaults from the alpha description.

To work with data item alias revisions

On Data Dictionary Repository form

1. Press F5. The Data Field Alias form displays.

2. Enter an alias type and name.

An alias name must be unique to the system or the system does not let you exit from the Data Field Alias form.

Current alias types required:

- 1 = PL1 or COBOL
- 2 = C language

An alias must adhere to J.D. Edwards’ syntax rules of the “C” language.
Working with the Data Dictionary Glossary

What are the Data Dictionary Glossary Groups?

The Data Dictionary consists of several glossary groupings that define the data item in the J.D. Edwards software. All glossary groups typically have associated text. The glossary stores this text. The major glossary groups follow:

The Data Dictionary consists of several glossary groupings that define the data item in the J.D. Edwards software. All glossary groups typically have associated text. The glossary stores this text.

E

J.D. Edwards interactive error messages

- J.D. Edwards defines interactive error messages with numbers less than 5000 and with numbers from 000A to 999Z. For example, 0001
- Client defines interactive error messages with numbers from 5001 to 9999

M

Menu Messages

- J.D. Edwards defines menu message data items as MENUMSGxxx, where xxx represents a number. For example, MENUMSG044
- Client defines menu message data items as MENUCLTxxx, where xxx represents a number

J

J.D. Edwards batch error messages

- J.D. Edwards defines batch error messages with JDExxxx, where xxxx represents a number less than 7000. For example, JDE0001
- Client defines batch error messages with JDExxxx, where xxxx represents a number greater than 7000 and less than 9000
- The QJDEMSG message file contains batch error messages
- A J.D. Edwards program found on Rebuilds and Global Updates (G9642) must build the batch error messages files QJDEMSG

C

Data Item Functions Categories

- Groups common data elements
- For example, CURRENCY
D or S
Primary or Secondary Data Items
- Used for validations
- Text on Videos
- Text on Reports
- Field Reference Files - F98RFA-Z $ and @
- For example, AC for a D data item; AC01 for an S data item

F
Files

G
General Narrative. Used to add information about a specific data item

H
User Defined program Helps
- Client use only for adding custom helps for J.D. Edwards programs
- For example, U00MENU

L
Report Messages. Messages or warnings for certain procedures, or letters written and produced through DREAM Writer

N
Program Notes
- Used by programmers to type notes about a program in the system
- Add the notes to the glossary in the Data Dictionary
- Create notes for a program, add a data item with an “N” as a prefix in front of the program name. For example, N00HELP

P
Program Purposes
- Used in the general summary help instructions
- Used for the Program Generator Product
- For example, P01051

R
Report Data Elements - the majority of these data items are letters produced through DREAM Writer

T
Terms
- These data items are definitions of commonly used terms
- The prefix of the data item name is “TERM.” For example, the AAI definition is in the glossary under the data item TERMAAI.
For work fields that a program utilizes

- Begin with #
- For example, #AA

To work with the glossary


   If your glossary group is E, H, J, or M, this form automatically displays when you press Enter on the main Data Dictionary form.

   A user defined code (system 01, type ST) that identifies the kind of Address Book record you want the system to select when you do name or message searches. Examples:

   - E - Employees
   - X - Ex-Employees
   - V - Vendors
   - C - Customers
   - P - Prospects
   - M - Mail Distribution Lists

2. Do the following that applies:

   - Use the Language, Applic Override, and Scrn/Rpt fields for jargon. See About Language and Jargon for details.
   - Use cursor keys to see additional text lines.
   - When entering an “E” glossary group item, which is an interactive error message, use F5 to define a program, form, or report to reference when the system displays the error message.
   - On double-byte machines, this form displays the Search Desc field. To ensure the data item search facility functions properly, you must enter a search description for each data item you create or change. You can enter it on this form or on the Data Dictionary form.

3. Always leave the last two character positions of each text line blank.
<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
</table>
| Data Item             | The RPG data name. This data field has been set up as a 10-byte field for future use. Currently, it is restricted to 4 bytes so that, when preceded by a 2-byte file prefix, the RPG data name does not exceed 6 bytes. Within the Data Dictionary, all data items are referenced by this 4-byte data name. As they are used in database tables, a 2-character prefix is added to create unique data names in each table specification (DDS). Special characters are not allowed as part of the data item name, with the exception of #, @, $.

You can create protected data names by using $xxx and @xxx, where you define xxx.

Messages can contain up to 10 characters. Types of messages are further defined by glossary group.                                                                                                                                   |
|                       | **Form-specific information**                                                                                                                                                                                                                                                                                                                |
|                       | If you are adding an error message, this field must be left blank. The system assigns the error message number using next numbers. The name appears on a successful add. You should assign interactive error message numbers greater than 5000.                                                                                                                                         |
| Glossary Group        | Differentiates data items into types. These types include primary and secondary types, error messages, and help text. See UDC 98/GG for a complete listing of Glossary Groups.                                                                                                            |
|                       | See also 'What Are the Data Dictionary Glossary Groups?' within this 'Data Dictionary Repository' chapter.                                                                                                                                                                                                                               |
|                       | **Form-specific information**                                                                                                                                                                                                                                                                                                                |
|                       | NOTE: If you need to assign your own error message numbers, use 4 digit numbers greater than '5000'.                                                                                                                                                                                                                                      |
|                       | For help text (glossary group H), the data dictionary “Inquiry/Revision Program” field may be used to specify the name of a follow-on item.                                                                                                                                                                                                 |
|                       | To create your own messages for the IBM message file (glossary group J), begin the data item name with your own three characters (e.g., CLT0001).                                                                                                                                                                                                 |
Working with User Defined Help Instructions

The easiest way to modify help instructions is to utilize the User Defined Instructions in Data Dictionary.

To work with user defined help instructions

On the Data Item Glossary Revisions form

<table>
<thead>
<tr>
<th>92001</th>
<th>Data Item Glossary Revisions</th>
<th>Language . . . . . __</th>
<th>Applic Override __</th>
<th>Scrn/Rpt . __</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action Code. . . . I</td>
<td>Install System Code. 00</td>
<td>Desc Help - User Defined Instructions</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data Item. . . . U00MENU</td>
<td>Reporting System Code. 00</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Glossary Group . . . H</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

This is a sample of user defined instructions that may be entered by users for any given program in the system. If you wish to provide highlighted, or underlined, or both highlighted and underlined text refer to the special attributes section of help instructions. All user defined instructions may be entered directly into the data dictionary.

F4=Search  F9=Redisplay Prev  F19/F20=Prev/Next Item  F24=More

J.D. Edwards provides an example record (U00MENU) in your system.

1. Enter a program name in the Data Item field, replacing the “P” with “U”. For example, for program P01051, create a data item U01051.

2. Enter H in the Glossary Group field. The H Glossary Group defines user defined help. J.D. Edwards does not replace H Glossary Group data items during an upgrade.

3. Perform an add or change.

On the Help Task List form, “F5=User Inst” is displayed if you wrote your own User Defined Help instructions.
Working with Data Field Descriptions

To work with data field descriptions

1. From Data Dictionary, press F11.

<table>
<thead>
<tr>
<th>Action Code</th>
<th>Data Item</th>
<th>Row Description</th>
<th>Column Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>AN8</td>
<td>Address Number</td>
<td>Address</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Number</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>O Lan Appl</th>
</tr>
</thead>
<tbody>
<tr>
<td>P</td>
</tr>
<tr>
<td>_ _ 44</td>
</tr>
<tr>
<td>Row</td>
</tr>
<tr>
<td>Vendor Number</td>
</tr>
<tr>
<td>Customer Number</td>
</tr>
<tr>
<td>Customer Number</td>
</tr>
<tr>
<td>Customer Number</td>
</tr>
</tbody>
</table>

2. On the Data Field Descriptions form, enter specific jargon or language descriptions for each data item. See About Language and Jargon in Technical Foundation for details.
Working with the Next Numbers Facility

The Next Number facility controls the automatic numbering for such items as new G/L account numbers, voucher numbers, address numbers. It allows you to specify what numbering system you want to use and gives you a method of incrementing numbers to reduce transpositions and keying errors.

Complete the following tasks:

- Locate the Next Numbers facility
- Work with Next Numbers by company and fiscal year

► To locate the Next Numbers facility

From menu G00, choose Next Numbers.

<table>
<thead>
<tr>
<th>Action Code</th>
<th>System Code</th>
<th>Next Numbers</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>09</td>
<td>General Accounting</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Use</th>
<th>Next Number</th>
<th>Check Digit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Next Account ID</td>
<td>21811</td>
<td></td>
</tr>
<tr>
<td>Journal Entries</td>
<td>1966</td>
<td></td>
</tr>
<tr>
<td>Consol Accounts</td>
<td>90000214</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

CAUTION: Changing the data on this screen may make it impossible to retrieve previously added addresses and may result in attempts to assign duplicate numbers.

F8=Next Numbers by Co/FY   F24=More Keys
What You Should Know About

What You Should Know About Next Numbers

**Next Numbers**

The next numbers file is F0002

- 10 element array
- 1 record per system
- Modulus 11 check optional

Once set, do not change:

- Has an impact on system performance
- Will not duplicate numbers. When it reaches max, starts over
- Cannot change position of user or add new entry without programming modifications

Ties with the Data Dictionary:

- Data Item in Data Dictionary points to the Next Number System. For example, System Code 09 AID Data Item

---

The next numbers file is F0002:

- 10 element array
- 1 record per system
- Modulus 11 check optional

Once set, do not change:

- Has an impact on system performance
- Will not duplicate numbers; when it reaches max, starts over
- Cannot change position of user or add new entry without programming modifications

Ties with the Data Dictionary:

- Data Item in Data Dictionary points to the Next Number System (for example, System Code 09 AID Data Item)

---

To work with Next Numbers by company and fiscal year

1. From Next Numbers, press F8.
2. Set the Next Number constant field to maintain next numbers by
   - Company
   - Company and Fiscal Year

Use Next Number by Company for these original documents:
   - Journal Entries
   - Accounts Payable Vouchers
   - Accounts Receivable Invoices
   - Sales Orders
   - Purchase Orders

**About the Field Reference File**

The Field Reference File contains the specifications for each data item in the J.D. Edwards Data Dictionary. Because the J.D. Edwards Data Dictionary is different from the standard IBM Data Dictionary, each data item record needs to be translated from the J.D. Edwards standard to the IBM standard.

When building the Field Reference File, J.D. Edwards groups the data items. Items that begin with “A” are translated into the IBM-readable format and stored in file F98FRFA. Data items that begin with “B” are in F98FRFB. Each letter of the alphabet has a corresponding F98FRF file. Client data items are stored in F98FRFS and F98FRFB. You can rebuild one file at a time. You can also build the message file in alternative languages.
What Happens with the Rebuild?

The system does the following:

- Rebuilds F98FRFA-Z, $, and @
- Picks up Data Dictionary data item glossary groups D and S
- Rebuilds the message file (QJDEMSG) in QGPL. Uses a processing option, Form ID J98DDMSGF, to determine which library to build the QJDEMSG file. The default is QGPL
- Does not rebuild the J.D. Edwards message file if entering a single field reference file to be built
- Builds a separate message file for each language installed. Enter "@@" for all languages installed on the system.

Always rebuild the files in the same library as previously built.

About the J.D. Edwards Message File

The J.D. Edwards Message (QJDEMSG) file contains all the messages that are coded Glossary Group J. The programs access the messages from this file. If a client adds messages with Glossary Group J, a rebuild is necessary to correctly add the new messages to the J.D. Edwards Message (QJDEMSG) file.

Rebuilding only the J.D. Edwards Message File?

The system does the following:

- Rebuilds the message file (QJDEMSG) in QGPL. Uses a processing option, Form ID J98DDMSGF, to determine which library to build the QJDEMSG file. The default is QGPL
- Picks up Data Dictionary data item glossary group J

Enter a value from UDC table 01/LP to generate a message file for a single language. Enter "@@" for all languages installed on the system.

Locating the Rebuild FRF and JDE Msg File Form

To locate the Rebuild FRF and JDE Msg File form

From menu G9642, choose Rebuild FRF & JDE Msg File
The Field Reference Files are facsimiles of the J. D. Edwards Data Dictionary and are vital for the creation of all data base files. The version of the Data Dictionary upon which they are based determines the type and characteristics of all application data elements. This procedure will recreate these files based upon the Data Dictionary files found in the library specified, placing the DDS source in the JDESRC source file the Source Library selected, with the Field Reference Files being created in the Data Library selected.

Base Field Ref Files on Data Dictionary in Library ________
Create Field Ref source in Source Library ________
Create Field Ref Files in Data Library ________
Single field ref($) @, A-Z or blank=all) _______
Language for message file (** for all) _______

NOTE: Generation of Field Reference and Message File is submitted to batch. No data files may be created during this generation process.

Press Enter to Rebuild Field Reference Files F3=Exit without Rebuild

Exercises
See the exercises for this chapter.
Work with Data File Design Aid

About the Data File Design Aid

J.D. Edwards Data File Design Aid provides a simple mechanism for creating Data Description Specifications (DDS) for physical and logical files.

- J.D. Edwards does not allow any file changes through the Source Entry Utility (SEU) to enforce standards. Changes must be done through File Design Aid.

What You Should Know About

Enforced Prefixes

Throughout the Data Dictionary, J.D. Edwards makes extensive use of the data item name. Within files, these data item names are qualified with a prefix to make them unique. Every data file in J.D. Edwards software is assigned a two-character prefix. For example:

- Business Unit Master file is MC
- Address Book Master is AB
- The data name MCU in the Business Unit Master file is MCMCU
- The data name in the Address Book file is ABMCU
- Use of prefixes ensures that data item names are both consistent and unique.

Enforced naming conventions

At J.D. Edwards, file names begin with an F prefix and the format within that file begins with an I prefix.

Data Dictionary validation

- All data fields defined in files are verified against the Data Dictionary.
- Programmers cannot enter data names without first creating and documenting them in the Data Dictionary.
- Prefixes of $ and @ are reserved for client use.
Automatic reference to Field Reference Files

- J.D. Edwards uses IBM's Field Reference File (FRF) technology for all files. When creating the DDS for a file, you need to enter the Data Dictionary data item name. Data File Design Aid automatically enters the correct keywords for referring to the FRFs.
- If data items are added to the Data Dictionary, the user needs to run the rebuild for the Field Reference Files before using Data File Design Aid.

Resequencing

- A sequence number allows you to rearrange data items within a file while you are designing.
About Assigning the File Prefix

File prefixes are assigned through the Software Versions Repository.

9801
Software Versions Repository

- The Q series is reserved for clients.
- If creating a new logical file, the prefix defaults from the based on physical file.
- To view all file prefixes currently in use, press F1 on the File Prefix field. A file prefix can display in this list more than once if it is attached to more than one file.
  - Pressing F10 from this form displays all file prefixes that you should not use.
F10 - User Defined Code Form

F10 – User Defined Code Form

F10. Displays the User Defined Code form to see which prefixes you should not use.

<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>File Description</td>
<td>The description of a record in the Software Versions Repository file. The member description is consistent with the base member description.</td>
</tr>
</tbody>
</table>

- The information in this form comes from a logical file built over the Software Versions Repository.
- The information in this form is updated automatically whenever the user adds, updates, or deletes software version repository record(s) for files.
- Programmers are responsible for not assigning the same prefix to different files used in the same program.

Entering Data File Design Aid

You must have access to the source file to enter FDA.

To enter Data File Design Aid

1. Inquire on a physical file.
2. Copy the production source down to a development environment.
3. Choose Option 10 to take you to the appropriate Design Aid form based on the members Function Code value.

- A PF or LF value takes you to File Design Aid.
## Data File Design Aid

### File Description
- **Unique Keys (Y/N)**: Y
- **File Prefix**: QX
- **File Description**: SDM Item Master File
- **Src Library**: PGFSRC
- **Source File Name**: JDESRC

#### Data Item Specifications

<table>
<thead>
<tr>
<th>Data Item</th>
<th>Data Field Desc.</th>
<th>K/S</th>
<th>Function Specifications</th>
<th>Seg No</th>
</tr>
</thead>
<tbody>
<tr>
<td>92801</td>
<td>Item ID</td>
<td>K</td>
<td>REFFLD(XIT) F98FRFX</td>
<td>1.00</td>
</tr>
<tr>
<td>XDS</td>
<td>Description</td>
<td></td>
<td>REFFLD(XDS) F98FRFX</td>
<td>2.00</td>
</tr>
<tr>
<td>XTY</td>
<td>Item Type</td>
<td></td>
<td>REFFLD(XTY) F98FRFX</td>
<td>3.00</td>
</tr>
<tr>
<td>XDT</td>
<td>Date Last Ship</td>
<td></td>
<td>REFFLD(XDT) F98FRFX</td>
<td>4.00</td>
</tr>
<tr>
<td>XBU</td>
<td>Business Unit</td>
<td></td>
<td>REFFLD(XBU) F98FRFX</td>
<td>5.00</td>
</tr>
<tr>
<td>XIT</td>
<td>Quantity – On Hand</td>
<td></td>
<td>REFFLD(XIT) F98FRFX</td>
<td>6.00</td>
</tr>
<tr>
<td>XUM</td>
<td>Item Unit of Measur</td>
<td></td>
<td>REFFLD(XUM) F98FRFX</td>
<td>7.00</td>
</tr>
<tr>
<td>X001</td>
<td>Item Category Code</td>
<td></td>
<td>REFFLD(X001) F98FRFX</td>
<td>8.00</td>
</tr>
<tr>
<td>X002</td>
<td>Item Category Code</td>
<td></td>
<td>REFFLD(X002) F98FRFX</td>
<td>9.00</td>
</tr>
<tr>
<td>X003</td>
<td>Item Category Code</td>
<td></td>
<td>REFFLD(X003) F98FRFX</td>
<td>10.00</td>
</tr>
<tr>
<td>X004</td>
<td>Item Category Code</td>
<td></td>
<td>REFFLD(X004) F98FRFX</td>
<td>11.00</td>
</tr>
<tr>
<td>X005</td>
<td>Item Category Code</td>
<td></td>
<td>REFFLD(X005) F98FRFX</td>
<td>12.00</td>
</tr>
<tr>
<td>XIT</td>
<td>Item ID</td>
<td>K</td>
<td>REFFLD(XIT) F98FRFX</td>
<td>13.00</td>
</tr>
</tbody>
</table>

### Field Explanation

**File Description**
Database text string that names the data item. Enter text in upper and lower case. The system uses this field to search for similar data items. To enter an alpha description, follow these conventions:
- Dates – Begin all Date fields with Date
- Amounts – Begin all Amount fields with Amount
- Units – Begin all Unit, Quantity, and Volume fields with Units
- Name – Begin all 30-byte description fields with Name
- Prompt – Begin any Y/N prompting field with Prompt
- Address Number – Begin all address numbers (employee, customer, owner) with Address Number

---

A8.1(8/97) 3-41
<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
</table>
| Unique Keys(Y/N)    | Specifies if the data file contains unique keys.  
|                     | - If you say yes, Data File Design Aid puts the unique keyword in the DDS. As a result, no two records may have duplicate keys.  
|                     | - If you say no, Data File Design Aid leaves the UNIQUE keyword out of the file DDS.                                                        |
|                     | **Form-specific information**                                                                                                                |
|                     | If a file can be organized so the key will uniquely identify only one specific record, define the Unique Keys field. Uniqueuness can be specified for physical and logical files.  
|                     | Most J. D. Edwards physical files in the past have been defined as sequential and logicals were used for creating keyed sequences. More recently, however, physical files have been keyed. |
| Member ID           | The record of the Software Versions Repository member to be copied.  
|                     | **Form-specific information**                                                                                                                |
|                     | The name assigned to the file. Defaults in from the Software Versions Repository.                                                            |
| File Prefix         | This field indicates the prefix associated with a file. Use F1 to display all file prefixes in use. Each physical file should have an unique file prefix. |
| Src Library         | The library containing the data to be copied.  
|                     | **Form-specific information**                                                                                                                |
|                     | The library where the source for the data file resides. Defaults in from the Software Versions Repository.                                   |
| Source File Name    | The name of the file within the source library that contains the source member. Defaults in form the Software Versions Repository.         |
| Based on File       | Designates the physical file on which a logical file is based.  
<p>|                     | - Defaults in from the Software Versions Repository and only displays for logical files.                                                    |
|                     | <strong>Form-specific information</strong>                                                                                                                |
|                     | For physical and logical files, the Based On File is the same as the physical file.                                                        |
|                     | For join files, the Based On File is the name of the first physical file that the join is built over.                                            |</p>
<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
</table>
| Data Item Type        | The Data Dictionary name of the field or the record format name.  
  - The file prefix is added to create unique data names for each file specification if a data item is entered in this field.  
  - The record format line is automatically defaulted in.                                           |
| Data Item Size        | The description of the data item entered in the previous field.  
  - Comes from the Row Description field in the Data Dictionary.                                      |
| K/S                   | Identifies the DDS Type indicating whether the field is a format name, key field, select logic field or omit logic field. It may be used in conjunction with information that appears in the Function Specifications field. |
| Function Specifications| Used with the DDS Type specified in the K/S column.  
  - If it is a record format name:  
    It is blank for physical files  
    Contains the PFILE (Filename) statement for a logical file and you enter: JFILE (Filename Filename) statement for join files listing all the files involved in the join. Right below the JFILE statement, you use the JFLD (Field Field) statement to list the fields that are used to construct the join.  
  - If you are defining a normal data item and you want the FRF field designation pulled in, you leave it blank.  
  - If you are defining Select/Omit logic on a field, you enter the logic itself.  
  - If you are defining a key data item, you may leave the Function Specifications field blank, or you may enter any valid DDS function keyword (DESCEND, RENAME, SIGNED, ZONE, and so forth). |
| Seq No                | Determines the order of the fields in the file. **Form-specific information**  
  When designing a physical, list the component fields in descending order of their importance to the file. Keyed items must always be last in sequence number within the Data File Design Aid program itself. |

There is a fold area which includes additional information: data item type, data item size, and number of display decimals.
### Sample — Logical File

J.D. Edwards logical files contain all fields from the PF, only keys are specified.

<table>
<thead>
<tr>
<th>Data Item</th>
<th>Data Field Desc.</th>
<th>K/S</th>
<th>Function Specifications</th>
<th>Seq No</th>
</tr>
</thead>
<tbody>
<tr>
<td>I92801</td>
<td>Business Unit</td>
<td>K</td>
<td>PFILE(F92801)</td>
<td>1.00</td>
</tr>
<tr>
<td>XIT</td>
<td>Item ID</td>
<td>K</td>
<td></td>
<td>2.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>4.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>5.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>6.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>7.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>8.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>9.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>10.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>11.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>12.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>13.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>14.00</td>
</tr>
</tbody>
</table>

F3=Exit/Save  F16=Search by File  F1=Search by Name  F4=Field Attributes

### Sample — Logical File with Selects

This example represents an AND condition for the selects.

<table>
<thead>
<tr>
<th>Data Item</th>
<th>Data Field Desc.</th>
<th>K/S</th>
<th>Function Specifications</th>
<th>Seq No</th>
</tr>
</thead>
<tbody>
<tr>
<td>I0911</td>
<td>Account ID</td>
<td>K</td>
<td>PFILE(F0911)</td>
<td>1.00</td>
</tr>
<tr>
<td>LT</td>
<td>Ledger Type</td>
<td>K</td>
<td></td>
<td>2.00</td>
</tr>
<tr>
<td>DOI</td>
<td>DOI Sub</td>
<td>K</td>
<td></td>
<td>3.00</td>
</tr>
<tr>
<td>SBL</td>
<td>Subledger</td>
<td>K</td>
<td></td>
<td>4.00</td>
</tr>
<tr>
<td>DSV</td>
<td>Date - Service/Tax</td>
<td></td>
<td></td>
<td>5.00</td>
</tr>
<tr>
<td>DSVM</td>
<td>Date - Service/Tax</td>
<td></td>
<td></td>
<td>6.00</td>
</tr>
<tr>
<td>DSVD</td>
<td>Date - Service/Tax</td>
<td></td>
<td></td>
<td>7.00</td>
</tr>
<tr>
<td>DCT</td>
<td>Document Type</td>
<td>K</td>
<td></td>
<td>8.00</td>
</tr>
<tr>
<td>DOC</td>
<td>Document (Voucher, Document Company)</td>
<td></td>
<td></td>
<td>9.00</td>
</tr>
<tr>
<td>POST</td>
<td>G/L Posted Code</td>
<td>S</td>
<td>CMP(EQ 'P')</td>
<td>10.00</td>
</tr>
<tr>
<td>BC</td>
<td>Bill Code</td>
<td></td>
<td>CMP(NE 'H')</td>
<td>11.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>12.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>13.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>14.00</td>
</tr>
</tbody>
</table>

F3=Exit/Save  F16=Search by File  F1=Search by Name  F4=Field Attributes
Sample — Logical File with Omits

This example represents an AND condition for the omissions.

Creating Join Files and Work Files

To create a join file or a work file, you should use the Source Edit Utility.

Function Keys From File Design Aid

F1 - Field Help on Data Item

F1. Using F1 in the Data Item field takes you to the Data Item Search form.

F2 - J.D. Edwards Command Line

F2. Access the command line to enter a J.D. Edwards or IBM command without having to exit to Command Entry or a menu. If you are secured out of Command Entry or Menu Traveling, you can still get to this command line but you cannot execute commands or menu travel.
**F3 - Exiting Data File Design Aid**

F3. When you press F3 to exit Data File Design Aid, the following form appears.

```
Data File Design Aid

Update Source Changes (Y/N) . . N
Member ID. . . . . . . . . . . F92801
File ID. . . . . . . . . . . . JDESRC
Src Library. . . . . . . . . . PGFSRC
Description. . . . . . . . . . SDM Item Master File
Function Code. . . . . . . . . PF
Return to Design (Y/N) . . . . N
```

On this form, you can choose to:

- Exit without saving the changes made.
- Exit and save the changes made.
- Save the changes made and return to the Design Aid form.

**F6 - Access Repository Services**

F6. This form provides access to other repository services within J.D. Edwards.

**F16 - Search by File**

F16. Accesses the File Field Description form to view file formats and field descriptions for any file on the system.
**What Are the Data File Design Aid Standards?**

<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unique Keys</td>
<td>Specifies if the data file contains unique keys. If Yes, FDA puts the unique keyword in the DDS. No two records can have duplicate data in the key field. If No, FDA leaves the keyword out of the file DDS. Records can share data in those key fields.</td>
</tr>
<tr>
<td></td>
<td><strong>Form-specific information</strong></td>
</tr>
<tr>
<td></td>
<td>If a file can be organized so the key will uniquely identify only one specific record, define the Unique Keys field. Uniqueness can be specified for physical and logical files.</td>
</tr>
<tr>
<td></td>
<td>Most J. D. Edwards physical files in the past have been defined as sequential and logicals were used for creating keyed sequences. More recently, however, physical files have been keyed.</td>
</tr>
<tr>
<td>File Description</td>
<td>The description of a record in the Software Versions Repository file. The member description is consistent with the base member description.</td>
</tr>
<tr>
<td></td>
<td><strong>Form-specific information</strong></td>
</tr>
<tr>
<td></td>
<td>The description associated with each file is used to further identify the relation of the file and its purpose.</td>
</tr>
<tr>
<td></td>
<td>- Physical files should have a description that explains the purpose of the file.</td>
</tr>
<tr>
<td></td>
<td>- Logical files should be designated as follows: LF – filename, filename, filename: where filename is a key field.</td>
</tr>
<tr>
<td></td>
<td>- Join files should be designated as follows: JF – filename, filename, filename: filename, filename, filename: where the filename is a file over which the join is built and filename is the key field joining the files.</td>
</tr>
<tr>
<td></td>
<td>- Work files should be designated as follows: WF – filename; where filename is the file that the work file accesses.</td>
</tr>
<tr>
<td>Based On File</td>
<td>Designates the physical file on which a logical file is based.</td>
</tr>
<tr>
<td></td>
<td>- Defaults in from the Software Versions Repository and only displays for logical files.</td>
</tr>
<tr>
<td></td>
<td><strong>Form-specific information</strong></td>
</tr>
<tr>
<td></td>
<td>For physical and logical files, the Based On File is the same as the physical file.</td>
</tr>
<tr>
<td></td>
<td>For join files, the Based On File is the name of the first physical file that the join is built over.</td>
</tr>
<tr>
<td>Field</td>
<td>Explanation</td>
</tr>
<tr>
<td>------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Ordering of Fields</td>
<td>Determines the order of the fields in the file.</td>
</tr>
<tr>
<td></td>
<td><em>Form-specific information</em></td>
</tr>
<tr>
<td></td>
<td>When designing a physical, list the component fields in descending order of their importance to the file. Keyed items must always be last in sequence number within the Data File Design Aid program itself.</td>
</tr>
<tr>
<td>Logical Files</td>
<td>Logical files include all fields; we do not define specific fields.</td>
</tr>
<tr>
<td>Recompiling</td>
<td>When recompiling a physical, you need to delete any logicals or joins from the data file library and then recompile them after the physical has been recompiled.</td>
</tr>
<tr>
<td>Record Format</td>
<td>It is a J.D. Edwards standard that only one record format is defined for each physical and logical file. Joins may contain more. Record format names begin with I followed by the physical file number.</td>
</tr>
<tr>
<td>Field Reference Files</td>
<td>Used in all file creations to retrieve field descriptions.</td>
</tr>
</tbody>
</table>
Merge Functions for Program Temporary Fix (PTF) Installations and Reinstallations

The PTF installation or reinstallation does the following:

- A PTF installation prints a report that identifies all files that are in the PTF library but were not installed in the client’s production libraries. You must add the new files manually into the appropriate libraries.
- A reinstallation prints a report to add new files into appropriate libraries.
- Updates JDFDATA in a PTF installation; replaces JDFDATA in a reinstallation.
- Adds new keys to both logical and physical files.
- Changes the file formats of logical and physical files.

The Data Models display relational models of the major files within each J.D. Edwards product.
Data File Design Aid Summary

In summary, the Data File Design Aid has the following features or restrictions:

- It has direct ties to the Data Dictionary and the Field Reference Files.
- It attaches a two-character prefix to each data item to create a unique field within the file.
- A record format must be defined for all files with a K/S value of R. This is the default record format.
- The PFILE keyword is automatically pulled in for logical files.
- Logical files must have a Based on File designated in the Software Versions Repository, which carries over to the design form.
- You must enter the data item names from the Data Dictionary.
- Perform these steps for creating a new file:
  - Data items must reside in the Data Dictionary.
  - You must rebuild the FRF files if new data items were added (from the Rebuilds menu, G9642).
  - A new file must have a file prefix specified on the Software Versions Repository record.
- Field Reference Files are characterized by the following:
  - They contain all the definitions for creating fields.
  - There are 28 in all (F98FRFA–F98FRFZ, F98FRF$, and F98FRF@).
  - Each field reference file contains all the data items beginning with the same character as the field reference file.

For example: F98FRFA contains all Data Dictionary data items beginning with the letter A.

Exercises

See the exercises for this chapter.
Work with Screen Design Aid

About Screen Design Aid

Screen Design Aid (SDA) is an interactive feature you use to design and maintain forms. This full-screen editor validates your work against the Data Dictionary and adds records to vocabulary overrides. You can work with multiple record formats simultaneously and you can move fields from one format to another.

Below is a list of some features of the Screen Design Aid:

- Design is conducted in a safe work environment. If you make a mistake you can exit without changing a form’s Data Description Specifications (DDS).

- Form specifications are stored in data structures in the QRECOVERY library. This is similar to the IBM recovery of SEU.

- You can create a form in normal mode (80 columns by 24 rows) or wide mode (132 columns by 27 rows). You can also design wide forms on 80 column devices using a windowing facility.

- Answering initial yes/no options allows you to create a basic form skeleton for a subfile, non-subfile or window-style form.

- SDA is fully integrated with the Data Dictionary and vocabulary override files. You can place fields on the form by referring to a Data Dictionary name and override default attributes, if necessary. You can place vocabulary override fields on the form and, if desired, modify their contents through the full form.

- SDA is fully integrated with the system database. You can select fields from the system database, create a pick list and then reorder fields in the pick list. You can place fields on the form individually or all at once by pinpointing locations on the full form with an ampersand (&) or asterisk (*).

- SDA has full screen capability. You can add, change, move, or delete fields by entering control characters directly on the form.

- Unlike the IBM SDA, the JDE SDA allows you to work with multiple record formats at one time. You can display and change any combination of formats simultaneously (as long as they do not overlap). You can also move fields from one format to another.
- SDA allows you to simulate a form at program execution time. You can run the simulation for any set of conditioning indicators to represent a particular error condition or other program functions.

**Editing Commands**

Below is a list of editing commands available in the SDA:

<table>
<thead>
<tr>
<th>Command</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>*DEL</td>
<td>Delete field(s) (used in Field Definition window)</td>
</tr>
<tr>
<td>d</td>
<td>Delete field(s) (used in Field Definition window)</td>
</tr>
<tr>
<td>(cannot be uppercase D)</td>
<td></td>
</tr>
<tr>
<td>&lt;&lt;, &gt;&gt;</td>
<td>Shift field(s) to the left or right</td>
</tr>
<tr>
<td>(xx...xx) ’xx...xx’</td>
<td>Literals (use apostrophes)</td>
</tr>
<tr>
<td>-</td>
<td>Move from position.</td>
</tr>
<tr>
<td>=</td>
<td>Move to position.</td>
</tr>
<tr>
<td>=</td>
<td>Move block from position</td>
</tr>
<tr>
<td>F7</td>
<td>Move block to position.</td>
</tr>
<tr>
<td></td>
<td>Restore the form if you accidently press Field Exit or a power failure knocks you off.</td>
</tr>
</tbody>
</table>

The following is a list of precautions and automatic features of the SDA:

- Do not use the INSERT or DELETE keys while in the actual design portion of SDA.
- Do not use the INSERT or DELETE keys while in the actual design portion of SDA.
- SDA automatically assigns editing indicators.
  - Indicators 40 to 79 are reserved for editing.
  - Indicator 40 is reserved for the Action Code field.
  - Indicator 41 is reserved for the key fields.
  - If you use all available indicators, you will get an error message.
- Indicator 37 is used in subfile forms to highlight all fields on the last line of the subfile to indicate that no more records exist.
Prefix Standards

Below is a list of prefix standards for use in the SDA:

<table>
<thead>
<tr>
<th>Prefix</th>
<th>Explanation</th>
</tr>
</thead>
</table>
| VD     | Video display fields.  
  - VD fields display database information from the file being used for the form and you can use them to enter database information.  
  - Default size is the size specified in the Data Dictionary for the data item being displayed.  
  - Reside in the based on file and can be input/output. |
| SF     | Subfile fields.  
  - Same as VD fields, but they are in a subfile.  
  - Default size is the size specified in the Data Dictionary for the data item being displayed plus editing characters. |
| SH     | Subfile Hidden fields.  
  - SH fields store data that is not displayed on a form. |
# Field Name Standards

Below is a list of field name standards for use in the SDA:

<table>
<thead>
<tr>
<th>Field</th>
<th>Standard</th>
</tr>
</thead>
</table>
| VC0 – Video constants | VC0 fields display definitions or descriptions for a single piece of data or for a group of data.  
VC0 fields are always output fields and the description that is loaded into the VC0 field is obtained from a separate file  
- For example, if creating a form using the Item Master file (F92801), you need to take the Item Master Business Unit field and chain out to the Business Unit Master file (F0006) to get the description for that Business Unit.  
- You enter *VC0 for the Field Name field in the Field Definition form when adding a new constant or description field.  
- The default size for VC0 fields is 30. |
| VTX – Video text | These fields display the row description or column headings from the Data Dictionary.  
- The text that displays in the VTX fields is stored in the Vocabulary Overrides file (F9220).  
- You can type directly over Vocabulary Override fields in SDA.  
You enter *VTX for the Field Name field in the Field Definition form when adding a new text field.  
- The default size for VTX fields is 16. |
| Line 24 is always VDL24 | You cannot change the text for Line 24 by using the Field Definition form because it is too large.  
Type over the text in Line 24 to change it. |
| TTL@       | Uses the default title from Vocabulary Overrides if the form is called from another form.  
Uses the menu selection text if the form is called from a menu. |
| ACTION     | Action Code field.  
- The name assigned by SDA.  
- The default cursor keyword is assigned to the action code field. |
| *LITER – Literal fields | Literals are added by placing apostrophes around the text on the screen and pressing Enter. (For example, “V928011”). |
## Updating or Adding Fields through SDA

<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>* – Field Definition Window</td>
<td>Allows you to update existing fields and add new fields without using the Pick List feature. Place the * one space to the left of the first character of the requested field to display the Field Definition form.</td>
</tr>
<tr>
<td></td>
<td>• To add a field, place an asterisk (*) on the SDA design area where you want to add the field.</td>
</tr>
<tr>
<td></td>
<td>• To update a field, place an asterisk in the attribute character of the field you want to update.</td>
</tr>
<tr>
<td></td>
<td>You can pull in the form field, the Row Description/Column Headings (VTX), and a 30 character description field (VC0) all at the same time by making special entries in the field definition form (*BOTH and *ALL).</td>
</tr>
<tr>
<td>&amp; – Field Selection Window</td>
<td>Allows you to add new fields using the Pick List feature Causes the Field Selection form to display.</td>
</tr>
<tr>
<td></td>
<td>To place a field on the screen from your Pick List, place an ampersand (&amp;) on the SDA design area where you want to place the first character of the field.</td>
</tr>
<tr>
<td></td>
<td>Allows you to pull in one or all of the following at the same time:</td>
</tr>
<tr>
<td></td>
<td>• The Row Description/Column Headings (VTX)</td>
</tr>
<tr>
<td></td>
<td>• The form field</td>
</tr>
<tr>
<td></td>
<td>• A description field (VC0)</td>
</tr>
</tbody>
</table>
Working with Screen Design Aid

To work with Screen Design Aid you must have access to the source file.

To work with Screen Design Aid

1. Inquire on a form in SVR
2. Copy the production source code down to a development environment using selection 3.
3. Choose option 10 to access the appropriate Design Aid form based on the member's Function Code value.

Function Key Exits

F12 - Return to Previous Panel

F12. Exits you out of the current form or utility and returns to the form you were on previously.

- Use F12 instead of F3; however, if you are calling another program outside of SDA (for example: F13, F24), you must use F3 to return to SDA.
Updating an Existing Field

To update an existing field

Place an asterisk (*) in front of the field (in the attribute character).

<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dict Name</td>
<td>Identifies the four-byte data item name from the Data Dictionary.</td>
</tr>
<tr>
<td></td>
<td>This is the only required field for most data items, the rest will default.</td>
</tr>
<tr>
<td>Text</td>
<td>Describes the Dictionary Name.</td>
</tr>
<tr>
<td></td>
<td>On VTX fields contains soft coded description that updates F9220.</td>
</tr>
<tr>
<td>Data Type</td>
<td>S Numeric data items.</td>
</tr>
<tr>
<td></td>
<td>A Alphanumeric.</td>
</tr>
<tr>
<td></td>
<td>Blank w/decimal position blank defaults to A.</td>
</tr>
<tr>
<td></td>
<td>Blank w/decimal position defined defaults to an S.</td>
</tr>
<tr>
<td></td>
<td>All J.D. Edwards fields are defined as A.</td>
</tr>
</tbody>
</table>
### Field Explanation

**Field Name** identifies a screen field name.
- *VTX (VTX001–VTX200) automatically assigns next available.
- *VC0 (VC0001–VC0200) automatically assigns next available.
- *LITER literal fields.
- *BOTH or *ALL to bring in video (VD), VC0, and VTX fields.

**Row/Column**
- Two 3-digit fields that define the row and column location of field.

**Field Use**
- How the data is to be used on the screen.
  - I input only.
  - O output only.
  - B Both input and output.
  - H Hidden field.
  - M IBM Message field.

**Size**
- Two fields identify the length of the data item and for numeric fields, the decimal places.
  - If left blank, automatically fills.

**Text Form**
- For VTX fields, identifies the field from the Data Dictionary that is used for headings.
  - R Row Description.
  - C Column Heading 1.
  - D Column Heading 2.

**Dft Cursor**
- Starting cursor position on a data entry screen, Y or N.

**Edited**
- Should the field be checked for error conditions, Y or N.
  - Will assign an indicator for error handling and default Condition Indicator information.
  - Assigns error indicators 40–79.
  - Key fields, K. Assigns indicator 41.

**Lower Case**
- To allow lowercase, Y or N.

**Change**
- CHANGE keyword is in effect, Y or N. The indicator will be seton whenever the value in this field is changed.

**OVERDTA**
- OVRDTA keyword is in effect, Y or N. Used with PUTOVR to override data that is in a field already on the video.

---

You should edit all input capable fields. (There will be a "Y" or "K" in the "Edited" field).

NOTE: All input capable fields should be edited ('Y' or 'K' in Edited field).
<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Duplicate</td>
<td>Duplicate the data. Only valid for an SFL format. Puts the DUP keyword in the video/report DDS but the Program Generator does <strong>not</strong> generate any code to enable this.</td>
</tr>
<tr>
<td>OVRATR</td>
<td>OVRATR keyword is in effect, Y or N. Used with PUTOVR to override display attributes of a field on the video.</td>
</tr>
<tr>
<td>Field Cond</td>
<td>Field Conditioning Indicators. Determines if the user can see the field or not.</td>
</tr>
</tbody>
</table>
| Condition Indicators | To set a condition indicator on a field, enter a Y in the first blank to the right of the desired condition. You have the option of entering up to 3 indicators to be associated with the condition. Three spaces are provided to allow an N prior to the two digit indicator to create a negative condition. The allowed conditions are:  
  - RI  Reverse Image  
  - HI  Highlight  
  - UL  Underline  
  - ND  Nondisplay  
  - BL  Blink  
  - PR  Protect  
  - PC  Place Cursor  
  A blank or N will deactivate the condition. |
| Color        | F8 toggles to display the color attributes for the field. The first blank to the right of each color controls the order that multiple colors will appear in the DDS (1–7). If multiple colors are defined, the first enabled color appears and the remaining colors are ignored. A blank or N disables the color. The color values default based on whether you selected JDE or SAA colors in QJDF. |
Accessing Fast Path Create for a New Form

When you design the format for a new form, you have the option to use Fast Path Create.

To access Fast Path Create for a new form

1. Locate your form and enter selection 10
   - If SDA cannot find the existing DDS for your form, the following form will appear:
   - If SDA cannot find the existing DDS for your form, the Create New Screen (V927400) form will appear.

<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Describes the function or option exit.</td>
</tr>
<tr>
<td></td>
<td>Cannot exceed 40 characters.</td>
</tr>
<tr>
<td>Fast Path Create</td>
<td>Automatically create record formats, fields, file, and record level parameters.</td>
</tr>
<tr>
<td>Window</td>
<td>Video is a window.</td>
</tr>
<tr>
<td>Wide Screen</td>
<td>Video is in wide format (132 columns by 27 rows) or normal format (80 columns by 24 rows).</td>
</tr>
</tbody>
</table>
### Field | Explanation
--- | ---
Subfile | Create subfile format.
Subfile Fold | Create a fold area in the subfile using SFLDROP and SFLFOLD keywords.
Subfile Clear | Use SFLCLR (Y) OR FSLINZ (N).
Selection Exits | Create selection exits to allow the user to exit the program using selection codes.
PUTOVVR | The video record format used the PUTOVVR keyword. Causes the video to be erased and redisplayed when a window is displayed.
OVERLAY | The video record format uses the OVERLAY keyword. Will not erase and redisplay video when a window is displayed. Most J. D. Edwards videos use OVERLAY.

2. Press Enter and SDA begins the creation of your form based on what you specified.

**Example - Form with Action Code and No Subfile**

```
92700                            Item Maintenance
Action Code... B

F24=More Keys
```
Example - Form with Action Code and Subfile

92700                            Item Maintenance
Action Code. . . B

SFLCTL
DELETE THIS FIELD
DELETE THIS FIELD
DELETE THIS FIELD
DELETE THIS FIELD
DELETE THIS FIELD
DELETE THIS FIELD
DELETE THIS FIELD
DELETE THIS FIELD
DELETE THIS FIELD
DELETE THIS FIELD
DELETE THIS FIELD
DELETE THIS FIELD
DELETE THIS FIELD
DELETE THIS FIELD
DELETE THIS FIELD
DELETE THIS FIELD
DELETE THIS FIELD
DELETE THIS FIELD
DELETE THIS FIELD
DELETE THIS FIELD
DELETE THIS FIELD
DELETE THIS FIELD
DELETE THIS FIELD
DELETE THIS FIELD
DELETE THIS FIELD
DELETE THIS FIELD
DELETE THIS FIELD
DELETE THIS FIELD
DELETE THIS FIELD

F24=More Keys

Example - Form with Action Code, Subfile and Selection Exits

92700                            Item Maintenance
Action Code. . . B

O
P
B
B
B
B
B
B
B
B
B
B
F24=More Keys
Adding Fields without Using a Pick List

To add a Video Text Field (VTX)

1. Place an asterisk (*) on the SDA design area where you want to place the video text field.

When the field definition form appears:

2. In the Dict Name field, enter the Data Dictionary item name.
3. In the Field Name field, specify *VTX.
   - The system assigns the next available VTX number.
4. Enter a value in the Text Form field to indicate whether the row description or a column heading from the Data Dictionary should be used as the text.
   - R – Row Description.
   - C – Column Heading 1.
   - D – Column Heading 2.

   Default is R for non-subfile formats

   - Text defaults from the Data Dictionary based upon the Text Form value.
5. Enter a value in the Size field only if you want to override the default length of 16 for the Row Description that will be brought in.
You should start your fields in column two (unless selection exits exist). This allows you to place an asterisk to the left of the first field in column one.

NOTE: You should start your fields in column two (unless selection exits exist). This allows you to place an asterisk to the left of the first field in column one.

To add a Database Video Field (VD)

1. Place an * on the SDA design area where you want the field to be placed.

2. On the field definition form, enter the Data Dictionary item name in the Dict Name field.

3. Specify a field use.
   - The default for field use is O for output.
   - Editing indicators are not assigned for output fields.

4. Enter the Data Type, Size, and Text defaults from the Data Dictionary.
To add a Video Constant Field (VC0)

1. Place an * on the SDA design area where you want to place the description or constant field.

The Field Definition form is displayed.

2. On the field definition form, specify *VC0 in the Field Name field.
   - The system assigns the next available VC0 number.

3. If you want to override the default length of 30, enter a value in the size field.
Adding a Literal Field

To add a literal field

Enter the literal text in the SDA Design area, enclose the text within single quotes, and press Enter.

'928011'

Item Master Information

J.D. Edwards standard is that the only literal on a form is the program ID in the top left corner.

J.D. Edwards standard is that the only literal on a video is the program ID in the top left corner.
Using the \*BOTH and \*ALL Features

The field definition form allows you to enter some special keywords in the Field Name field. Two of these special keywords are \*BOTH and \*ALL.

This feature provides for placement of multiple fields with a single entry.

Using \*BOTH

If you use the keyword \*BOTH with a valid data dictionary item, screen design will place a VTX field and a video (VD) field on the screen.

To use \*BOTH

On Field Definition type "\*BOTH" in the Field Name
When you enter *BOTH, the following form appears in the SDA design area for the Unit of Measure field:
Using *ALL

If you use the keyword *ALL with a valid Data Dictionary item, screen design places a VTX field, a video (VD) field, and a VC0 field on the form.

To use *ALL

On the Field Definition form, type "*ALL" in the Field Name.
When you enter *ALL, the following form appears in the SDA design area for the Unit of Measure field:

![Form Diagram]

**Field Defaults**

The following are field defaults in the SDA:

**VD - Video Display field**

- Output only

You can enter a B in the Field Use field to override the default and change it to both input and output.

- No Editing

If you enter B in the Field Use field, the Edited field defaults to Y. The Condition Indicators default to Y and the next available editing indicator is assigned to that field.

**VTX - Video Text field**

- 16 bytes long
- Defaults to Row description rather than column description

**VC0 — Video Constant field**
Understand the SDA Exit/Save Function Key

F3 - Design Aid Exit/Save

F3. Saves or exits or does both from Screen Design Aid.

<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Save DDS (Y/N)</td>
<td>Whether or not to include the function or option key on the screen.</td>
</tr>
<tr>
<td></td>
<td><strong>Form-specific information</strong></td>
</tr>
<tr>
<td></td>
<td>Saves the DDS and updates or creates Vocabulary Overrides and Function Key definitions.</td>
</tr>
<tr>
<td>Member ID</td>
<td>The record of the Software Versions Repository member to be copied.</td>
</tr>
<tr>
<td></td>
<td><strong>Form-specific information</strong></td>
</tr>
<tr>
<td></td>
<td>Name of the screen.</td>
</tr>
<tr>
<td>File ID</td>
<td>The name of the file within the source library that contains the source member. Defaults in form the Software Versions Repository.</td>
</tr>
<tr>
<td></td>
<td><strong>Form-specific information</strong></td>
</tr>
<tr>
<td></td>
<td>Identifies the file that will contain the source code.</td>
</tr>
<tr>
<td>Field</td>
<td>Explanation</td>
</tr>
<tr>
<td>---------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Src Library</td>
<td>The library containing the data to be copied.</td>
</tr>
<tr>
<td></td>
<td>....................  <em>Form-specific information</em>  ....................  Identifies the library where the source code resides.</td>
</tr>
<tr>
<td>Description</td>
<td>The description of a record in the Software Versions Repository file. The member description is consistent with the base member description.</td>
</tr>
<tr>
<td></td>
<td>....................  <em>Form-specific information</em>  ....................  Description of the Member ID.</td>
</tr>
<tr>
<td></td>
<td>Should be the same as in F9801.</td>
</tr>
<tr>
<td>Function Code</td>
<td>Designates the object type such as display file, physical and logical files. Use F1 in the field to view the available types.</td>
</tr>
<tr>
<td></td>
<td>....................  <em>Form-specific information</em>  ....................  Identifies the Member ID.</td>
</tr>
<tr>
<td>Return to Edit (Y/N)</td>
<td>Logical files include all fields; we do not define specific fields.</td>
</tr>
<tr>
<td></td>
<td>....................  <em>Form-specific information</em>  ....................  EOJ or return to SDA.</td>
</tr>
</tbody>
</table>
Compiling Your Form

To compile your form

From the Software Versions Repository form, enter 14 next to the member in the subfile that you want to create and press Enter.
Screen Design Standards and Tips

Title

A screen title is limited to 30 characters and should match the Software Versions Repository (F9801). The title you enter in SDA updates the vocabulary overrides record for the form. If you access the form using a menu selection, the menu selection name overrides the form title. If you access the form using a selection option or function key, the vocabulary overrides title is used.

Line 24

You should document all function keys on the right side of line 24 and you should document options on the left side. The following guidelines should also be followed:

- List both the options and function keys in numeric order.
- F24 should always appear and should say MORE KEYS or MORE.
- F4 should always read MORE DETAIL or DETAIL.
- Do not include standard exits of F3, F7, F22, Help, Rollup, Rolldown.
- Line 24 should be in reverse image during an error condition except on forms. Line 24 is conditioned to appear in reverse image on forms based on indicator 93.
- If you specify *SAME for the field “Error Text” for Line 24 in vocabulary overrides, then the text displayed is the same as the text specified for the normal Line 24.

Forms

Within a form, line 24 should include F3 and F24 when the form is initially displayed. When designing forms in SDA, fill in unused line space with literal fields to prevent data on the calling form from showing through on the form. You can add the literal fields as blanks with a single quote on each end or through the Field Definition form.
Default Cursor

You set the default cursor attribute to Action Code for the input field closest to the upper-left corner of the form.

Fold Area

Keep the number of Fold Area lines to no more than two lines to avoid excessive use of the cursor keys when the Fold Area is open.

Description Fields

Define all description input fields to allow for uppercase and lowercase letters. Use VC0 descriptions when a field's value has no obvious meaning and you can retrieve a description from a master file or user defined codes.

Alpha Fields

Because of the dynamic nature of international currency, you must define every field as alpha. The only exception is that you can define hidden fields as numeric. J.D. Edwards scrubbing routines handle the two-way conversion between numeric file data and alpha form fields.
General Aesthetics

The following are things you might want to consider when designing forms. They are guidelines that will give your forms a more professional look.

Alignment

Line up fields vertically. This includes row descriptions, input fields, and description fields. Fields on the left side of the form should be in column space 2 (column 1 is needed for the attribute byte).

Use periods to equalize length of row descriptions
Line up input fields
Line up VC0 fields of row descriptions

08332                                      Single D/B Relation Entry
Action Code. . . . . . . . I
Employee Number. . . . . . 6001 Allen, Raymond
Plan ID. . . . . . . . . . DEPCARE
Dependent/Beneficiary No. 4036 Name. Allen, Cindy
Effective From . . . . . . 01/01/90 Thru .

Relationship Data:
Dependent or Beneficiary . D
Relationship . . . . . . . C Child
Dep/Ben Type . . . . . . . Primary Beneficiary
Percent Allocated. . . .

Dependent/Beneficiary Data:
Social Security Number . . . 524-58-5113
Date Of Birth. . . . . . . 04/01/72
Dep/Ben Status . . . . . .

Memo/Address Info. . . . 2525 E. 11th Avenue
Denver, Colorado
80206

F5=D/B Relationships         F21=Print         F24=More Keys
**Grouping Fields**

When entering a description heading to group related fields, use up to 40 characters for the description (or as long as space permits). Highlight the heading and end it with a colon. Underneath the heading, indent the group of fields one space to the right.

---

**Spacing**

Use the following as your standards when spacing different form elements.

- Separate column headings with one space.
- End row descriptions with at least one period followed by a single space before you begin associated input fields.

- Indent Fold Area fields one or more spaces to offset them from regular subfile.

- Use two or more spaces to separate Fold Area data fields from row descriptions that follow on the same line. End Fold Area row descriptions with a colon instead of periods to aid legibility.
- Insert a blank line between header and subfile information.

<table>
<thead>
<tr>
<th>Benefit Grp.</th>
<th>Business Unit</th>
<th>Started.</th>
<th>Terminated</th>
</tr>
</thead>
<tbody>
<tr>
<td>Benefit Name</td>
<td>Business Unit</td>
<td>Effective</td>
<td>Contributions</td>
</tr>
</tbody>
</table>

- When possible, insert a blank line between the title and first field. Begin fields on line 3 unless you need to use the upper right corner of line 1 and 2.

069116       Pay Type Specifications

Action Code. . . . .

Exercises

See the exercises for this chapter.
Adding Video Fields Using Pick List

To add video fields

From the SDA form:

1. Access the Records Formats List using the F10 key
2. Complete the Record Formats List form

F10 - Record Formats List

F10. Displays the Record Formats List.

<table>
<thead>
<tr>
<th>Opt</th>
<th>Format Name</th>
<th>Type</th>
<th>Fast Path</th>
<th>Start / End</th>
<th>Related</th>
<th># Fields</th>
<th>Fld Pfx</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>V928200C</td>
<td>SFLCTL</td>
<td>F92801</td>
<td>001 006</td>
<td>V928200S</td>
<td>000</td>
<td>VD</td>
</tr>
<tr>
<td>2</td>
<td>V928200S</td>
<td>SFL</td>
<td></td>
<td>007 022</td>
<td></td>
<td>000</td>
<td>SF</td>
</tr>
<tr>
<td>3</td>
<td>V9282001</td>
<td>RECORD</td>
<td>024 024</td>
<td></td>
<td></td>
<td>000</td>
<td>VD</td>
</tr>
</tbody>
</table>

This form is used to select database fields and maintain record formats, record types, fast path files, and record format keywords.

Field | Explanation
-----|--------------------------------------------------
Opt   | Enter the appropriate number to indicate you want to select one of the following values:
      | File/field pick list of ampersand functions.
      | File/field pick list for fast path functions.
      | List of defined fields in the format.
      | Delete format.
      | Record format keywords.
<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Format Name</td>
<td>Screen record format. The format name will be the video ID followed by a specific format suffix value. Typically, the suffix values are: subfile control format subfile format record format If additional formats are required, each format name must be unique so new format suffix values must be assigned.</td>
</tr>
<tr>
<td>Type</td>
<td>Record format type. See types listed below.</td>
</tr>
<tr>
<td>Fast Path File</td>
<td>The database file you want to select fields from.</td>
</tr>
<tr>
<td>Start/End Lines</td>
<td>Specifies the line number range of the format.</td>
</tr>
<tr>
<td>Related Record</td>
<td>Field that ties a subfile to a control record format. Required in all SFLCTL record formats.</td>
</tr>
<tr>
<td># Fields Selected</td>
<td>The number of database fields that have been selected for use on the format.</td>
</tr>
<tr>
<td>Fld Pfx</td>
<td>Form field prefix to be used for the video fields: VD, SF.</td>
</tr>
</tbody>
</table>

**About Record Formats**

Several Record Format Types are valid for forms. Currently, they are

- **SFLCTL - Subfile control**

  Present in all subfile forms. Contains all of the fields in the header or top portion of the form, including the subfile column headings.

V928200C (SFLCTL)
• SFL – Subfile

Contains all of the fields in the subfile portion of the form, including the fold area, if applicable.

V928200S (SFL)

• RECORD

Present in all forms. In subfile forms, contains VDL24 (line 24 text). In non-subfile forms, can contain all fields on the form, including VDL24.

V9282001 (RECORD)

• SFLMSG – Subfile Message

Displays error message text. J.D. Edwards does not use this format because errors are handled through RPG programs.
Selecting Database Fields

There are two methods of selecting database fields for placement on the form:

- With Fast Path
- With the File Selection List

<table>
<thead>
<tr>
<th>Method</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fast Path</td>
<td>Type 1 next to the format on which you want to place the fields and enter a file name under the Fast Path File column.</td>
</tr>
<tr>
<td>File Selection List</td>
<td>Type 1 next to the format on which you want to place the fields but do not enter a file name. Accesses a file selection form where you can specify multiple files and libraries from which to select database fields.</td>
</tr>
</tbody>
</table>

To select a database field using Fast Path

1. On the Record Formats List form, enter a Fast Path File for the specified format.

2. For database field selection, choose option 1 f

<table>
<thead>
<tr>
<th>Opt</th>
<th>Format Name</th>
<th>Type</th>
<th>Fast Path File</th>
<th>Start / End Lines</th>
<th>Related Record</th>
<th># Fields Selected</th>
<th>Fld Pfx</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>V928200C</td>
<td>SFLCTL</td>
<td>F92801</td>
<td>001 006</td>
<td>V928200S</td>
<td>000</td>
<td>VD</td>
</tr>
<tr>
<td>2</td>
<td>V928200S</td>
<td>SFL</td>
<td></td>
<td>007 022</td>
<td></td>
<td>000</td>
<td>SP</td>
</tr>
<tr>
<td>3</td>
<td>V9282001</td>
<td>RECORD</td>
<td>024 024</td>
<td>024</td>
<td>000</td>
<td>000</td>
<td>VD</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Opt: 1=DB Field Selection  3=Field List  4=Delete  5=Format Keywords
The Field Selection List appears.

<table>
<thead>
<tr>
<th>Seq No</th>
<th>Field Name</th>
<th>Description</th>
<th>DT</th>
<th>Size</th>
<th>HDG</th>
<th>D</th>
<th>Use</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>QXXIT</td>
<td>Item ID.</td>
<td>S</td>
<td>8</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>QXXDS</td>
<td>Description</td>
<td>A</td>
<td>30</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>QXTY</td>
<td>Item Type.</td>
<td>A</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>QXXDT</td>
<td>Date Last Ship</td>
<td>S</td>
<td>6</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>QXXCC</td>
<td>Business Unit.</td>
<td>A</td>
<td>12</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>QXXQT</td>
<td>Quantity On Hand</td>
<td>S</td>
<td>15</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>QXXUM</td>
<td>Unit of Measure.</td>
<td>A</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>QXX001</td>
<td>Item Code 001.</td>
<td>A</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>QXX002</td>
<td>Item Code 002.</td>
<td>A</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>QXX003</td>
<td>Item Code 003.</td>
<td>A</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>QXX004</td>
<td>Item Code 004.</td>
<td>A</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>QXX005</td>
<td>Item Code 005.</td>
<td>A</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

F3=Exit  F12=Prev Screen  F21=Select All

<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Seq No</td>
<td>Sequence Number to indicate which data items you want on the video you are creating and what order you want them to be displayed in the Pick List window accessed from SDA.</td>
</tr>
<tr>
<td>Field Name</td>
<td>The name given to a record format for a form, report, or database table.</td>
</tr>
<tr>
<td>Description</td>
<td>The Data Dictionary row description.</td>
</tr>
<tr>
<td>Data Item Type</td>
<td>The type of data. The data item types are defined in User Defined Codes, system code '98', record type 'DT'. Note: All amount fields should be entered as 15 bytes, 0 decimals, and data item type should be P (packed).</td>
</tr>
<tr>
<td>Data Item Size</td>
<td>The field size of the data item. NOTE: All amount fields should be entered as 15 bytes, 0 decimals, and the data item type should be P (packed).</td>
</tr>
<tr>
<td>HDG</td>
<td>Which heading to use from the Data Dictionary.</td>
</tr>
<tr>
<td></td>
<td>Row Description</td>
</tr>
<tr>
<td></td>
<td>Column 1 heading</td>
</tr>
<tr>
<td></td>
<td>Column 1 and 2 heading</td>
</tr>
<tr>
<td>D</td>
<td>Used to indicate whether a 30 character VC field should be brought for constant information to be loaded into.</td>
</tr>
<tr>
<td>Field</td>
<td>Explanation</td>
</tr>
<tr>
<td>-------</td>
<td>-------------</td>
</tr>
</tbody>
</table>
| Use   | Specifies how the data field is to be used on the video:  
|       | Input only.  
|       | Output only (default).  
|       | Both input and output.  
|       | IBM Message field. |

**F21 - Select All**

F21. To select all the fields for the file instead of selecting them individually, press F21 from this form.

Based on the record format for which you are using the Field Selection List, the following information is the default:

- For a subfile control record format, the HDG field will default to R for the type of heading and the Use field will default to B for input/output.
- For a subfile record format, the HDG field will default to D for the type of heading and the Use field will default to B for input/output.
- For a non-subfile form, the HDG field will default to R for the type of heading and the Use field will default to B for input/output.
- For a report, the HDG field will default to D for the type of heading and the Use field will default to O for output.
To select database fields using the File Selection List

From the Record Formats List form:

1. Choose option “1” but do not specify a file.

The File Selection List appears.
2. Enter the files from which you want to select fields.

Fields for files requested will be displayed through the Field Selection List form.

3. Select fields using the same techniques as in the Fast Path method.
   
   • If you select a key field, that field is edited as the key of the form. An edit indicator of 41 is assigned.
Placing Fields on a Form Using a Pick List

To place fields on a form using a Pick List:

On the Item Master Information form:

928011                       Item Master Information

Action Code. . . . B

Screen: V928200            Field Selection List           Format: V928200C
Seq   Fields to select                               Row  Desc Length. .  10
No  Field Name               Description                DT  Size HDG  D  Use
1  Qx$XIT 01          Item ID. . . . . . . . . . . . . A  8  O  R  D  B
2  Qx$XCC          Business Unit. . . . . . . . . . . . . A  12  R  D  B

F3=Exit  F10=Formats  F12=Prev Screen  F16=Field List

1. Type either one or more ampersands (&) on the form where you want to place the fields from the pick list you created.
   - If you place more than one &, make sure that you allow room for all of the fields that are returned to the form, so that you do not overlap fields.
2. On the Field Selection form, verify the information that is on the form (VTX field - HDG, 30-character description - D, and field Use - USE), as well as the order that they will be brought back (the sequence number), and row description length.

Adding a Fold Area to a Subfile

To add a Fold Area, place an asterisk (*) or ampersand (&) on the second line in the subfile format of your form. If you need a second line in the Fold Area, you can place an asterisk (*) or ampersand (&) on the third line of the subfile format. HDG should be “R” when adding to the fold.

Exercises

See the exercises for this chapter.
Function Key Exits from Screen Design Aid

**F2 - J.D. Edwards Command Line**

F2. Access a command line to enter a J.D. Edwards or IBM command without having to exit to Command Entry or a menu.

If you are secured out of Command Entry or Menu Traveling, you still get this command line, but you cannot execute commands or menu travel.

**F5 - Format Display Control Window**

F5. Shows the Format Display Control form.

<table>
<thead>
<tr>
<th>928200</th>
<th>Item Search</th>
</tr>
</thead>
</table>
| Business Unit: BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Selection</td>
<td>Controls the display of record formats.</td>
</tr>
<tr>
<td></td>
<td>1 Format is active.</td>
</tr>
<tr>
<td></td>
<td>Blank Not to display.</td>
</tr>
<tr>
<td>Format</td>
<td>Lists the DDS format names for the video screen.</td>
</tr>
<tr>
<td></td>
<td>All names begin with Video name</td>
</tr>
<tr>
<td></td>
<td>• Subfile control formats end with C.</td>
</tr>
<tr>
<td></td>
<td>• Subfile formats end with S.</td>
</tr>
<tr>
<td></td>
<td>• Record (non-subfile) formats end with 1.</td>
</tr>
<tr>
<td>Type</td>
<td>Describes the DDS format name.</td>
</tr>
<tr>
<td>Boundaries</td>
<td>Two 3-digit numbers that define the range (rows) for the DDS.</td>
</tr>
<tr>
<td>Window</td>
<td>Allows access to fields outside the boundaries.</td>
</tr>
<tr>
<td>Browse (Y/N)</td>
<td>Allows user to enable/disable the browse mode and view the screen as it would appear when executed.</td>
</tr>
<tr>
<td></td>
<td>• Cannot change or access any item while in browse mode.</td>
</tr>
</tbody>
</table>

**F4 - Subfile Drop (while in browse mode)**

F4. Provides the following:

- Toggle between displaying the Fold Area or not for a subfile form
- Must set Browse in Format Display Control Window (F5)

**F6 - Access Repository Services**

F6. This form provides access to other repository services within J.D. Edwards.

**F8 - Toggle Monochrome/Color Display**

F8. Provides the following:

- Will display your form in monochrome or color
- If accessing the Field Definition form, will toggle between Condition Indicators and Color Attributes

**F10 - Option 5 — Format Keyword Maintenance**

F10 – Displays the Format Keyword Maintenance form.
### Field | Explanation
--- | ---
PUTOVR (Y/N) | The video record format used the PUTOVR keyword. Causes the video to be erased and redisplayed when a window is displayed.
OVERLAY (Y/N) | The video record format uses the OVERLAY keyword. Will not erase and redisplay video when a window is displayed. Most J. D. Edwards videos use OVERLAY.
Subfile Fold | Create a fold area in the subfile using SFLDROP and SFLFOLD keywords.
Type (A/F) | Further identifies subfile fold area:
- A Will lose modified data in the subfile when you press F4.
- F Data is retained.
Subfile Clear | This option specifies if you want to use SFLCLR or SFLINZ. The default is SFLCLR. This option is ignored when designing non-subfile screens.
- Y means you want SFLCLR
- N will give you SFLINZ
Subfile Next Change | Whether or not to use SFLNXTCHG (Y/N). Will require the user to correct any errors in the subfile before further execution of the program.
Subfile Page | Identifies the number of records on one subfile page, with the fold area open, if applicable.
- 1 to 27 inclusive
Field | Explanation
--- | ---
Subfile Size | Identifies the total number of records in the subfile that will be loaded in one program cycle.
- 1 to 9999 inclusive

**F13 - Function Key/Opt Definition**

F13. Displays the Function Key/Opt Definition form.

- Used to define the function keys for the form
- Function Key Definition files (F9601 and F9611)

<table>
<thead>
<tr>
<th>Include</th>
<th>Description</th>
<th>Key/Opt</th>
<th>Field</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>Exit Program</td>
<td>03</td>
<td>#FEOJ</td>
</tr>
<tr>
<td>Y</td>
<td>Clear Screen</td>
<td>22</td>
<td>#FCLR</td>
</tr>
<tr>
<td></td>
<td>Help Instructions</td>
<td>HL</td>
<td>#FHELP</td>
</tr>
<tr>
<td></td>
<td>Roll Up/Next Record</td>
<td>RU</td>
<td>#FROLU</td>
</tr>
<tr>
<td></td>
<td>Roll Down/Previous Record</td>
<td>RD</td>
<td>#FROLD</td>
</tr>
<tr>
<td></td>
<td>Field Sensitive Help</td>
<td>01</td>
<td>#FQMRK</td>
</tr>
<tr>
<td></td>
<td>Display Error Message(s)</td>
<td>07</td>
<td>#FERRD</td>
</tr>
<tr>
<td></td>
<td>Display All Function Keys</td>
<td>24</td>
<td>#FKEYS</td>
</tr>
<tr>
<td></td>
<td>Item Maintenance</td>
<td>05</td>
<td>#F01</td>
</tr>
<tr>
<td></td>
<td>Item Master Information</td>
<td>01</td>
<td>#S01</td>
</tr>
</tbody>
</table>

Include: Y/N  F16=Display All

<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action Code</td>
<td>One character field used to indicate the action that the user wants to take on the record requested. Inquire on a record before you attempt to change it.</td>
</tr>
<tr>
<td>Video Screen</td>
<td>The name of the screen or report record to be copied. All records for soft coding will be copied.</td>
</tr>
<tr>
<td>Video Title</td>
<td>The vocabulary overrides title used on forms and reports. On forms, the title is retrieved from the Menu table. If a record is not found, then the title is retrieved from the Vocabulary Overrides table. Report titles are retrieved from the DREAM Writer Version ID (F98301).</td>
</tr>
</tbody>
</table>
**F14 - Indicator Control**

F14. Displays the Indicator Control portion of a simulated program execution form.

- Used with the Browse mode to simulate a form at program execution.
**F16 - List of Defined Fields**

F16. Displays the List of Defined Fields form.

- Used to maintain the defined fields and add hidden fields.
- Only shows fields for the formats that are active.

![List of Defined Fields](image_url)

**Hidden Fields**

Used to store hidden field information

**To add a hidden field to a form**

1. Roll to the bottom blank line of the format that contains the field.
2. Choose option 5, Display/Update.
3. Enter the field with a prefix of SH, description, type, size, and press Enter.
   - This information should be the same as the displayed database field that is affected.
Option 5. Select Field Definition

F17 – Define Soft Coding (Vocabulary Override) Fields

F17. Used to define soft coding fields.

- To define VTX fields other than row and column headings on the form. Row and column headings are protected here. Specify whether you want to use the Data Dictionary row description, column heading 1 or column heading 2.
- You can specify the literal text that will be loaded into a *VC0 field.
- You must save your form at least once in order to be able to update vocabulary overrides and/or function key definitions by this method. This is because when you are first defining a form, the vocabulary override record and function key definition record are not created until you save the form.

After changing the size of a VTX field, you should execute the Rebuild Vocabulary Override Field Lengths program (11/G9642). This displays the correct VTX field lengths in the Field Size field in Vocabulary Overrides.

An example of the Define Soft Coding Fields form follows:
**F19 - Window left**

F19. Window left

- This function key is applicable only when designing wide forms (132 by 27 rows) on 80 column terminal.

**F20 - Window right**

F20. Window right

- This function key is applicable only when designing wide forms (132 by 27 rows) on 80 column terminal.
Changing Subfile Boundaries

Be careful when changing the size of a subfile through SDA. Consider using these processes to make such changes easier and less confusing.

To make a subfile smaller

1. To access the Record Formats List form, press F10.
2. Change the starting line number for the subfile format (VxxxxxS).
3. To return to SDA, press Enter.
4. To access the Record Formats List form again, press F10.
5. Change the ending line number for the control format (VxxxxxC).
6. To return to SDA, press Enter.
7. Move or add headings.

To make a subfile larger

You can reverse the above steps if you want to make the subfile larger. You must move the control format fields up before changing the starting line number for the subfile format.

1. Move headings.
2. To access the Record Formats List form, press F10.
3. Change the ending line number for the control format (VxxxxxC).
4. To return to SDA, press Enter.
5. To access the Record Formats List form again, press F10.
6. Change the starting line number for the subfile format (VxxxxxS).
7. To return to SDA, press Enter.
8. In SDA, press F10 to alter the format.
9. Enter 5 on the control format field.
   - Change subfile page size if desired.

When you change the subfile, you must change the subfile page and subfile size to correctly reflect the size of the new subfile.
Process Overview - Placing Selected Fields

Once you have established your field pick list, use the ampersand (&) to specify where you want to locate the field.

The ampersand (&) calls up the pick list in the Field Selection form where you can order the fields and further define their specifications.

Options

The following options are available. You can:

- Override Row Description length
- Resequence fields in list
- Select headings (Row, Column headings) *VTX
- Description Field (*VC0)
- Usage (O=Output, B=Both Input and Output)
Once you have sequenced the fields, they are retrieved from the file and placed on the design area.
Process Overview - Revising the Field Definition

SDA Design Area
Place asterisk in the field’s attribute byte to revise

OR

F16

OR

F10

List of Defined Fields

Use the Field Definition information to display or create attributes for the data item.

Option 5 next to displayed

Field Definition

From the Field Definition Information to display color attributes for the data item.
Process Overview - Revising Vocabulary and Function Keys

Define Soft Coding Fields

SDA Design Area

Function Key/Opt Definitions

F17
Use the Define Soft Coding Fields form to define VTX fields other than row and column headings.

F13
Use the Function Key/Opt Definition screen to define the function keys for the form.

Function Keys for Form and Display Format Control

Indicator Control

User can turn indicators on and off to see how the form will look.

Format Display Control

Browse (Y/N)

The form will show a subfile form that has a fold area in its folded & un-folded formats. The form must be in browse mode for this to function.

Use the format display control form to:
1) Put the form in browse mode
2) Activate or deactivate formats
3) Enable the F4 key to see a subfile form in the folded or unfolded format.
Summary of Screen Design Aid

- Editing options
  - d, *DEL
  - <<, >>
  - ‘xx...xx’
  - -, =
  - – –, =
  - * and &

- You should not use the INSERT and DELETE keys while in SDA.

- F7 restores a form if you accidentally press Field Exit.

- Standard prefixes
  - VD, SF, SH

- Special Fields
  - ACTION
  - VDL24
  - TTL@

- Error indicators 40 to 79 are automatically assigned to VD and SF fields that are defined as input or input/output

- Update fields by using *

- You have two methods of adding fields to a form
  - * (non-pick list method)
  - & (pick list method)

  You can pull in VTX, VC, and the form database fields all at the same time for one database field

- You have two methods of selecting database fields
  - Fast Path
  - Non-Fast Path - Accesses File Selection form
• If you are changing subfile boundaries, you should use the outlined processes to make this process easier

• You must save a form at least once before updating vocabulary overrides or Function Key Definitions because the exit from SDA creates these records

• You add hidden fields from the List of Defined Fields form, which you access by pressing F16 from SDA
  • You add hidden fields one at a time
  • You must enter a selection exit 5 to actually add the field

Exercises

See the exercises for this chapter.
Work with Report Design Aid

About Report Design Aid

The Report Design Aid (RDA) is a powerful and versatile tool for designing reports.

It uses the same process as the Screen Design Aid (SDA), except:

- It extends to column 227
- It has windowing capability

You need to identify only field names, field lengths, and field positions on the report.

J.D. Edwards reports are externally defined, which means that all the Data Description Specifications are created and compiled as a printer file, separate from the program object. Report Design Aid automatically generates the DDS. It also incorporates the report information into the documentation and adds it to the cross reference facilities. You can print illustrations of each report.

RDA differs from SDA in that its parameters are targeted for print-based output, which includes page skipping, line skipping, and relative positioning.

Perform the following tasks:

- Access Report Design Aid
- Update a Field in RDA
- Compile a Report
- Change the Compile Option Defaults for Reports
Example - RDA and DREAM Writer

Comparing RDA and SDA - Field Definition Form

<p>| Screen: V5501Z---Field Definition---Format: V5501ZS |</p>
<table>
<thead>
<tr>
<th>Dict Name</th>
<th>$XDS</th>
<th>Text</th>
<th>Description</th>
<th>Data Type</th>
<th>A</th>
<th>Field Name</th>
<th>SP$XDS</th>
<th>Size</th>
<th>Text Form</th>
<th>Cond Ind</th>
</tr>
</thead>
<tbody>
<tr>
<td>Row/Column</td>
<td>8 13</td>
<td>Field Use</td>
<td>B</td>
<td>RI</td>
<td>Y</td>
<td>44</td>
<td>HI</td>
<td>Y</td>
<td>44</td>
<td>--</td>
</tr>
<tr>
<td>Dft Cursor</td>
<td>44</td>
<td>Edited</td>
<td>Y</td>
<td>ND</td>
<td>--</td>
<td>BL</td>
<td>--</td>
<td>ND</td>
<td>--</td>
<td></td>
</tr>
<tr>
<td>Lower Case</td>
<td>Change</td>
<td>Duplicate</td>
<td>--</td>
<td>BL</td>
<td>--</td>
<td>PR</td>
<td>--</td>
<td>BL</td>
<td>--</td>
<td></td>
</tr>
<tr>
<td>OVRDTA</td>
<td>Field Cond</td>
<td>--</td>
<td>PC</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>PC</td>
<td>--</td>
<td></td>
</tr>
</tbody>
</table>

--F3=Exit  F12=Prev Screen  F17=Dictionary
**FIELD POSITIONING**

**RDA**
Row positions are relative to the other field, not fixed. The location on the report is determined by Space and Skip designations. Column positions are fixed.

**SDA**
Both row and column positions are fixed. A field appears on the screen exactly where the Row and Column indicators specify.

**FIELD CONDITIONING**

A field can optionally appear bold, underlined, and so forth. J.D. Edwards does not typically use these features because they impact printer performance.

A field can appear highlighted, underlined, in reverse image, and so forth. J.D. Edwards makes use of these attributes for marking fields in error.

**Cover Page Fields**

The figure below shows the fields used on the cover page of a report. These fields would indicate your company in a production environment.

---

Program ID . . P01301
Version . . . 048
J.D. Edwards and /company Vendor Listing
Colorado Only
One Line per Address

The Following Version Options Were Selected:
Print Cover Page (Y/N) . . . . . . . . . . Y
Print Instructions (Y/N) . . . . . . . . . . N

The Following Forms Options Were Selected:
Form Type . . . . . . . . . . . . . . . . . . .
Maximum Form Width . . . . . . . . . . .
Maximum Form Length . . . . . . . . . . .

---

Report Date . . . 08/10/96
Report Time . . . 12:07:56
The following table provides the field names and a description of each.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>VC0CO</td>
<td>Name of company 00000</td>
</tr>
<tr>
<td>TTL@</td>
<td>Line 1 of DREAM Writer Version ID if it exists, otherwise it is blank</td>
</tr>
<tr>
<td>TXT2</td>
<td>Line 2 of DREAM Writer Version ID, or blank</td>
</tr>
<tr>
<td>TXT3</td>
<td>Line 3 of DREAM Writer Version ID, or blank</td>
</tr>
</tbody>
</table>

Report Header Fields

The figure below shows the fields used on the report header. These fields would indicate your company in a production environment.

The following table provides the field names and a description of each.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>VC0CO</td>
<td>Name of company 00000</td>
</tr>
<tr>
<td>RRTTL@</td>
<td>Line 1 of DREAM Writer Version ID if it exists, otherwise it is blank</td>
</tr>
<tr>
<td>RRTXT2</td>
<td>Line 2 of DREAM Writer Version ID, or blank</td>
</tr>
<tr>
<td>RRTXT3</td>
<td>Line 3 of DREAM Writer Version ID, or blank</td>
</tr>
</tbody>
</table>
What Are the Report Formats?

The first step in designing a new report is determining the format of the report. You should account for all lines of information on the report to correctly define the formats needed and their size.

<table>
<thead>
<tr>
<th>FORMAT</th>
<th>FIELD</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Any format</td>
<td>*VTX</td>
<td>Assigns the first available VTX name to the field and gets a description from the Data Dictionary that you can change.</td>
</tr>
<tr>
<td></td>
<td>*VC0</td>
<td>Assigns the first available VC0 field and assigns a default size of thirty.</td>
</tr>
<tr>
<td>HEADING1 – contains the standard fields to be printed on the top of every page</td>
<td>VTX001</td>
<td>The default VTX field which prints the row description, Page –.</td>
</tr>
<tr>
<td></td>
<td>*PAGE</td>
<td>The default special field that inserts the DDS keyword PAGNBR in the source and retrieves the current page number on the report.</td>
</tr>
<tr>
<td></td>
<td>VTX002</td>
<td>The default VTX field which prints the row description, Date –.</td>
</tr>
<tr>
<td></td>
<td>*DATE</td>
<td>Special field that retrieves today's date.</td>
</tr>
<tr>
<td></td>
<td>VCOCO</td>
<td>The name of the default company 000, it appears on the first line of each page.</td>
</tr>
<tr>
<td></td>
<td>RRTTL@</td>
<td>Line 1 of DREAM Writer Version ID if it exists, otherwise it is blank.</td>
</tr>
<tr>
<td></td>
<td>RRTXT2 &amp; RRTXT3</td>
<td>DREAM Writer overrides that correspond to the second and third header lines of the report.</td>
</tr>
<tr>
<td>HEADING2 – contains the subheading fields used to describe the level break detail that is to follow</td>
<td>VC0ROW</td>
<td>Data Dictionary row description of the level break field.</td>
</tr>
<tr>
<td></td>
<td>VC0KEY</td>
<td>The value of the level break field.</td>
</tr>
<tr>
<td></td>
<td>VC0DSC</td>
<td>The description of the value of the level break field.</td>
</tr>
<tr>
<td>DETAIL1 – contains the data line fields</td>
<td>RRxxxx</td>
<td>The value of the data for this field.</td>
</tr>
</tbody>
</table>
What Are the Report Design Standards?

The following is a list of report design standards. Using these standards will give your reports an uniform appearance.

<table>
<thead>
<tr>
<th>FORMAT</th>
<th>FIELD</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>TOTAL1 – contains the total line fields</td>
<td>VC1ROW</td>
<td>Data Dictionary row description of the level break field</td>
</tr>
<tr>
<td></td>
<td>VC1KEY</td>
<td>The value of the level break field</td>
</tr>
<tr>
<td></td>
<td>VC1DSC</td>
<td>The description of the level break field</td>
</tr>
<tr>
<td></td>
<td>$$$XXX</td>
<td>Value on total line.</td>
</tr>
</tbody>
</table>

You can have as many formats as you can fit on one RDA form. Just remember to increment the suffix number for each format added as well as any VC fields you may be using.

Certain fields are used in RDA when generating reports that contain subheadings or dynamic (hierarchical) totaling. The following illustrates how these fields are used within a report.

---

**RR Fields**

<table>
<thead>
<tr>
<th>VC0ROW</th>
<th>VC0DSC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Item</td>
<td>Description</td>
</tr>
<tr>
<td>Business Unit</td>
<td>...</td>
</tr>
<tr>
<td>1 Bolt</td>
<td>300</td>
</tr>
<tr>
<td>2 Nut</td>
<td>400</td>
</tr>
<tr>
<td>3 Nail</td>
<td>150</td>
</tr>
<tr>
<td>Total</td>
<td>850</td>
</tr>
</tbody>
</table>

**$$ Fields**

<table>
<thead>
<tr>
<th>VC1ROW</th>
<th>VC1DSC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Business Unit</td>
<td>...</td>
</tr>
</tbody>
</table>

**TOTAL1**

<table>
<thead>
<tr>
<th>VC1KEY</th>
<th>VC1DSC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Business Unit</td>
<td>...</td>
</tr>
</tbody>
</table>
RDA Features

Some of the features of RDA are:

- Normal design range of 132 – 198 character reports
- Validates against the Data Dictionary
- Automatically adds records to the vocabulary overrides file

J.D. Edwards Standards for Record Formats

Prefix standards

- RR for output fields
- $$ for total fields

General Aesthetics

When possible, design your reports using the following set of rules:

Column Headings

Column headings should not be wider than the length of the data that appear below them.

Alignment

Begin fields in column space 2 and do not extend fields beyond column 132 unless necessary.

Spacing

Use the following as your guides when spacing different report elements:

- Separate column headings by one space
- Use both column headings when one heading is not clear enough

Special Effects

You should always use dashes below column headings instead of underlines. Underlines can impact the performance of printers. You enter dashes as literal fields.

Do not use highlight as it prints a line three times to achieve the highlighted (or boldface) effect, again impacting performance.
Format

To avoid overflow, limit the number of lines in any detail or total format to six or less.

Line and Page Skipping

To be consistent with other report programs, use SPACEB and SKIPB instead of SPACEA and SKIPA.

About Designing the Report

- DDS are being created as you design the report
  - SPACEB and SPACEA are entered and removed as you add and move fields around.
  - Multiple formats are relative to each other.

<table>
<thead>
<tr>
<th>Function</th>
<th>What to use</th>
</tr>
</thead>
<tbody>
<tr>
<td>Changing the Report Title</td>
<td>TTL®</td>
</tr>
<tr>
<td>Adding a New Field</td>
<td>*, &amp;</td>
</tr>
<tr>
<td>Updating Existing Fields</td>
<td>*</td>
</tr>
<tr>
<td>Deleting an Existing Field</td>
<td>*DEL on field definition form</td>
</tr>
<tr>
<td>Format Name</td>
<td>Displayed in upper right hand corner of form.</td>
</tr>
<tr>
<td>Field positions</td>
<td>Represent starting positions.</td>
</tr>
</tbody>
</table>
Accessing Report Design Aid

You must have access to the source file to enter RDA.

To access Report Design Aid

From the the Software Versions Repository

<table>
<thead>
<tr>
<th>Source</th>
<th>Object</th>
<th>Source</th>
<th>SAR</th>
<th>Version</th>
<th>SD</th>
<th>User</th>
<th>Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>P Library</td>
<td>Library</td>
<td>File</td>
<td>Number</td>
<td>ID</td>
<td>C P</td>
<td>ID</td>
<td>Modified</td>
</tr>
<tr>
<td>JDFSRC71</td>
<td>JDFOBJ71</td>
<td>JDESRC</td>
<td>834451</td>
<td>A71</td>
<td>I</td>
<td>QUARLES</td>
<td>10/26/94</td>
</tr>
<tr>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

Opt: 1=Browse 2=Edit 3=Copy 5=SAR 8=Print 9=Dlt 10=Design 14=Crt  F24=More

1. Inquire on a report.
2. Copy the production source code down to a development environment.
3. Choose option 10 on the Software Versions Repository form to go to the appropriate Design Aid form based on the members Function Code value.
   - To go to Report Design Aid, enter “PRTF” or “PRTS” in the Function Code field
Updating a Field in RDA

The field definition form in RDA is slightly different from SDA.

To update a field in RDA

From Software Versions Repository, choose the design option.

1. Enter “*” in the field you wish to update.

<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Space Before</td>
<td>Specifies the number of lines a printer device is to space before printing the next line(s)</td>
</tr>
<tr>
<td>Space After</td>
<td>Specifies the number of lines a printer device is to space after printing the next line(s)</td>
</tr>
<tr>
<td>Skip Before</td>
<td>Specifies that the printer device is to skip to a specific line number before it prints the next line(s)</td>
</tr>
<tr>
<td>Skip After</td>
<td>Specifies that the printer device is to skip to a specific line after it prints the next line(s)</td>
</tr>
<tr>
<td>Field Cond</td>
<td>Indicates whether the field conditioning (to print this field or not) is in effect</td>
</tr>
</tbody>
</table>
### Field | Explanation
--- | ---
Char per Inch | Specifies the horizontal printing density. J.D. Edwards specifies this at the report level and this field is not used.
Edit Code | Used to specify output formatting of numeric data. Used in conjunction with *DATE, *TIME, *PAGE.
Asterisk Fill | Optionally specify asterisk fill for edit codes 1–4, A–D, and J–M. An asterisk will print for each zero suppressed in the edited field.
Float Symbol | Specify a currency symbol (corresponding to the system value QCURSYM) that will be printed immediately to the left of the left-most digit of an edited field. Valid for a numeric field that has an edit code of 1–4, A–D, or J–M.

---

**Understanding the Report Design Aid Function Keys**

**F5 - Format Display Control**

F5. Shows the Format Display Control portion of a form.
<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sel</td>
<td>Selection.</td>
</tr>
<tr>
<td></td>
<td>Controls the display of record formats.</td>
</tr>
<tr>
<td>Format</td>
<td>Lists the DDS format names.</td>
</tr>
<tr>
<td></td>
<td>Valid format names are:</td>
</tr>
<tr>
<td></td>
<td>• HEADING1</td>
</tr>
<tr>
<td></td>
<td>• HEADING2</td>
</tr>
<tr>
<td></td>
<td>• DETAIL1</td>
</tr>
<tr>
<td></td>
<td>• TOTAL1</td>
</tr>
<tr>
<td>Type</td>
<td>Describes the DDS format type.</td>
</tr>
<tr>
<td></td>
<td>Always REPORT or SFORMS in RDA.</td>
</tr>
<tr>
<td>Boundaries</td>
<td>Two 3-digit numbers that define the range (rows) for the DDS.</td>
</tr>
<tr>
<td></td>
<td>• HEADING1 is rows 1 to 8</td>
</tr>
<tr>
<td></td>
<td>• DETAIL1 is row 9</td>
</tr>
<tr>
<td></td>
<td>• TOTAL1 is rows 10 to 11</td>
</tr>
<tr>
<td>Window</td>
<td>Allows you to access fields outside the boundaries.</td>
</tr>
<tr>
<td>Browse (Y/N)</td>
<td>Indicator that allows you to enable/disable the browse mode.</td>
</tr>
</tbody>
</table>

RDA might automatically adjust displayed formats with those formats that are not displayed.

**F6 - Repository Services**

F6. Shows the Repository Services portion of a form.
F10. Displays the Record Formats List form.

The Record Formats establish the arrangement of fields on your report and in what segment of the page they are to print.
**Field** | **Explanation**
---|---
Opt | Enter the appropriate number to indicate you want to select one of the following values:
  - File/field pick list of ampersand functions.
  - File/field pick list for fast path functions.
  - List of defined fields in the format.
  - Delete format.
  - Record format keywords.
Format Name | Screen record format.
  The format name will be the video ID followed by a specific format suffix value. Typically, the suffix values are:
  - subfile control format
  - subfile format
  - record format
If additional formats are required, each format name must be unique so new format suffix values must be assigned.
Type | Record format type. See types listed below.
Fast Path File | The data base file you want to select fields from.
Start/End Lines | Specifies the line number range of the format.
Related Record | Field that ties a subfile to a control record format. Required in all SFLCTL record formats.
Fld Pfx | Screen field prefix to be used for the video fields: VD, SF.

There should be no gaps between the end line of one format and the start line of the next format. If you make changes to the positioning of a format and leave a gap between formats, RDA will automatically adjust the end lines for you.
F14 – Indicator Control form

F14. Displays the Indicator Control form.

F16 – Display All Defined Fields

F16. Displays the List of Defined Fields form.
F17 - Maintain Vocabulary Override Fields

F17. Used to maintain vocabulary override fields.

You must save your report at least once to update vocabulary overrides by this method. This is because when you are first defining a report, the vocabulary override record is not created until you save the report.

F19 - Window Left

F19 – Window Left

F20 - Window Right

F20 – Window Right
Compiling A Report

To compile a report

From the Software Versions Repository form

Enter 14 next to the member in the subfile that you want to create and press Enter.
A form of printer file parameters displays.

2. You can either accept the defaults or change them as necessary.
Changing the Compile Option Defaults for Reports

You must compile reports through the J.D. Edwards compiler by this method so that R98COVER and R98RPTH are pulled in for the cover page and help instructions. Compiling through the Production Development Manager (PDM) or some other method will not bring this information in automatically.

To change compile option defaults for reports

Change the Data Dictionary defaults for the following data items:

- #FLN – Forms Length
- WDTH – Forms Width
- LPI – Lines Per Inch
- #CPI – Characters Per Inch
- #OVF – Overflow Line Number
- #ALN – Alignment (Y/N)
- #FTY – Form Type
- #CPY – Number of Copies
- #SPG – Number of Separator Pages

Some severity level 10 errors can occur when your report compiles because of R98COVER (DREAM Writer cover page) and R98RPTH (DREAM Writer help instructions). These are only warning errors.

Exercises

See the exercises for this chapter.
Programming Standards

Objectives

- To understand and use J.D. Edwards programming standards

Programming Standards

The Program Generator serves as the primary enforcer of J.D. Edwards programming standards. These standards include subroutines and consistent formats that ease the maintenance process. The following areas are covered in the programming standards.

- Program Specifications
- Program Overview
- Program Structure
- Performance Issues
- User Spaces
- User Indices
- File Servers
- Functional Servers
- Group Jobs
- J.D. Edwards Source Debugger
Program Specifications

About Program Specifications

As described in IBM's *Languages: RPG/400 User's Guide*, there are several kinds of RPG/400 specifications. When your source program is compiled, these specifications are arranged in the following sequence:

- Control specifications (H Specs)
- File description specifications (F Specs)
- Extension specifications (E Specs)
- Input specifications (I Specs)
- Calculation specifications (C Specs)
- Output specifications (O Specs)

An RPG/400 program does not have to use all specifications. A typical J.D. Edwards program contains control, file description, extension, input, calculation, and output specifications. The following descriptions are from the *Languages: RPG/400 User's Guide*, and are repeated here for your convenience.
What Are Control Specifications?

The control specifications include the name of the program.

- The first line identifies the program, P55011X, including its description, Item Information Update.
- The next fourteen lines are comments that are included in J.D. Edwards programs for copyright purposes and reproduction restrictions.
What Are File Description Specifications?

File description specifications (F Specs) describe all the files that your program uses. The information for each file includes:

- The name of the file
- How the file is used (for example, input)
- The size of records in the file for internal files or an external designation
- Whether or not the file is keyed
- Input or output device used for the file
- If the file will have records added to it

When the Program Generator generates a program, it arranges the included files in alphabetical order within the F Specs.

When a program runs, it opens the files in bottom-to-top order. As a general rule:

- Place the files that have the most I/Os at the bottom of the F specs.
- Place any small usage files or files that are closed after first use at the top of the F specs.
- Place the display or print files at the bottom of the list.
Line 35 shows a request for the compiler to copy in F specs from JDECPY. All F spec copies begin with D.

**What Are Extension Specifications?**

Extension specifications describe all tables and arrays used in the program. The information includes:

- Name of the file, table, or array
- Number of entries in a table or array input record
- Length of the table or array entry
- Optional comment text

<table>
<thead>
<tr>
<th>Columns</th>
<th>PROCEDURE TABLES AND ARRAYS</th>
</tr>
</thead>
<tbody>
<tr>
<td>0041.00</td>
<td>E* PROGRAM TABLES AND ARRAYS</td>
</tr>
<tr>
<td>0042.00</td>
<td>E* ................................</td>
</tr>
<tr>
<td>0043.00</td>
<td>E* ................................</td>
</tr>
<tr>
<td>0044.00</td>
<td>E EMK 64 4 Error Msg</td>
</tr>
<tr>
<td>0045.00</td>
<td>E @MK 64 1 Error Msg</td>
</tr>
<tr>
<td>0046.00</td>
<td>E @ER 64 4 Error Msg</td>
</tr>
<tr>
<td>0047.00</td>
<td>E @DV 40 1 Dflt Wrk</td>
</tr>
</tbody>
</table>

Lines 44 through 47 are used in this program to facilitate error handling and field editing.

- The first line defines an array called EMK which has a maximum of 64 entries, each with a length of 4 characters.

Line 52 requests that the compiler program copy in a specific set of E Specs.

- The E Specs, E0001, are used in any program that executes the common subroutine, C0001.
What Are Input Specifications?

Input specifications describe the records, fields, data structures, and named constants used by the program. The information in the input specifications includes:

- The name of the file
- The sequence of record types
- Whether record-identifying indicators, control-level indicators, field-record relation indicators, or field indicators are used
- Whether data structures, look-ahead fields, record identification codes, or match fields are used
- The type of each file (alphanumeric or numeric; packed-decimal, zoned decimal, or binary format)
- The location of each field in the record
- The name of each field in the record
- All named constants

Lines 73 through 83 are used to define some of the vocabulary overrides that appear on this screen.

- The ending lengths change from program to program, and the program retrieves the values for each field at the time it executes the housekeeping subroutine, S999.
What Are Calculation Specifications?

Calculation specifications describe the calculations to be done on the data and the order of the calculations. Calculation specifications can also be used to control certain input and output operations. The information includes:

- Control-level and conditioning indicators for the operation specified (generally not used in J.D. Edwards software)
- Fields or constants to be used in the operation
- The operation to be processed
- Whether resulting indicators are set after the operation is processed

The C Specs are the heart of the processing of a program. J.D. Edwards programs are designed with a MAINLINE portion which is a select set of C Specs that call other subroutines.
What Are Output Specifications?

Output specifications describe the records and fields in the output files and the conditions under which output operations are processed. They include information such as:

- Name of the file
- Type of record to be written
- Spacing and skipping instructions of Printer files
- Output indicators that condition when the record is to be written
- Name of each field in the output record
- Location of each field in the output record
- Edit codes and edit words
- Constants to be written
- Format name for a workstation file

J.D. Edwards utilizes the RPG EXCPT operation to release locks on data records. This O Specs inform the program which record format is to be released when the EXCPT UNLOCK calculation is performed. Additional formats can be identified with a name such as UNLCKA or UNLCKB.

- Typically, J.D. Edwards does not perform reporting functions using O Specs.
- You can use the Opcode “UNLCK” instead of EXCPT/O–SPECs.
Program Overview

About the Program Overview

The program overview provides a basic overview of the standards used in a program, including:

- Subroutines
- Error Handling
- Indicator Usage
- Documentation
- Miscellaneous Items
Subroutines

The Program Generator uses two categories of subroutines:

- Standard Subroutines
- Common Subroutines

Standard Subroutines

The Program Generator includes the required standard routines in the Calculation Specifications at the time it generates a program. It arranges them in alphanumeric order.

If you must enter your own standard subroutine, name it in such a way that it will be executed in the necessary order. For example, if you need your subroutine to be executed after the scrub and edit subroutine (S005) but before the update files subroutine (S010), begin the name with an S and then use a three to four character suffix that fits in logically, such as S005A or S006.

Standard subroutine code lines are identified in positions 7 and 8 with SR. Their names always begin with an S. Subroutines are separated by a single line of asterisks. Major blocks of code within a subroutine are separated by a single line of dashes.

```
1870.00  CSR         MOVE *BLANK    HRJBCD
1871.00  CSR         MOVE *BLANK    HRJBST
1872.00  CSR         MOVE *BLANK    HRRVW
1873.00  CSR         END

1874.00  C*––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––
1875.00  CSR         ENDSR
1876.00  C****************************************************************
1877.00  C*
1878.00  C*    SUBROUTINE S003 – Edit Key
1879.00  C*                  ————————————————————
1880.00  C*
```

Place an END tag on the ENDSR statement. The TAG name should start with END. The subroutine name is added as a suffix. For example, END001 would be the used for subroutine S001. Do not use the end tag for anything else. Use a T tag if the code needs to be executed prior to the ENDSR statement. For example, T001 would be used for subroutine S001 if the tag is used in the middle of the subroutine.
Common Subroutines

Common subroutines are maintained outside the program and are included at the appropriate times using the COPY statement. Common subroutines are also referred to as copy modules for that reason. J.D. Edwards stores all common subroutines in the file JDECOPY.

At compile time, the compiler copies in code for all instances of the COPY statement. The included code appears only once and then can be called from anywhere within the program.

The statement that instructs the compiler to copy in the source code is shown below. Single lines of asterisks separate common subroutines.

This example shows how the COPY statement in the source (above) brings in additional code to the compiled source (below).
The following user defined code contains an online listing and specifications.

Install System Code: 93

User Defined Code: /C
Error Handling

J.D. Edwards has devised an efficient means of handling errors by way of arrays.

- The EMK array holds the four byte data dictionary name of every error that could occur in this program. The array is loaded in Housekeeping (S999).
- The @MK array maintains a flag setting for each error identified in EMK. If one of the errors occurs, the flag is set on.
- The @ER array loads the related error messages when the user presses F7 to view the errors that actually occurred.
- A program may have up to 64 errors.
The call to the error message handling program is shown below.

If any error flag is set to one, then the program moves the corresponding data item from the array of all possible errors (EMK) into the array of the errors that have actually occurred (@ER). P0000E is called to display the errors when the function key is pressed.

The next example of code shows how a flag is set in the @MK array.

If indicator 82 is on, the standard indicator for an error (93) is set on and indicator 41 is set on to highlight the field in error.
The next example of code shows the loading of the array that contains every possible error for this program. This loading takes place only once (in S999).

Columns . . . :    1  71           Browse                       DEVSRC/JDESRC
SEU=>>>                                                                 P55011X
2324.00  C*   Load error messages array.
2325.00  C*       CSR                   MOVE '0001'    EMK,01           Inv Action
2326.00  C*       CSR                   MOVE '0002'    EMK,02           Inv Key
2327.00  C*       CSR                   MOVE '0003'    EMK,03           Inv Blanks
2328.00  C*       CSR                   MOVE '0004'    EMK,04           Inv Date
2329.00  C*       CSR                   MOVE '0005'    EMK,05           Inv Next Nbr
2330.00  C*       CSR                   MOVE '0007'    EMK,06           In Use
2331.00  C*       CSR                   MOVE '0025'    EMK,07           Inv Values
2332.00  C*       CSR                   MOVE '0026'    EMK,08           Inv MCU
2333.00  C*       CSR                   MOVE '0027'    EMK,09           Inv Desc Ttl
2334.00  C*                                                                             
2335.00  C*       CSR                   MOVE '0001'    EMK,01           Inv Action
2336.00  C*                                                                             
2337.00  C*   Load invalid action code array.
2338.00  C*       CSR                   MOVEA'     '   @NAC
2339.00  C*                                                                             
2340.00  C*       CSR                   MOVEA'     '   @NAC
2341.00  C*                                                                             
2342.00  C*   Load system date.
2343.00  C*                                                                             
2344.00  C*   

**Indicator Usage**

There are 99 indicators available for use. They are grouped by purpose. The chart on the next page lists the available indicators and their description.
<table>
<thead>
<tr>
<th>Indicator</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>Causes the Invalid Function Key Pressed message to appear</td>
</tr>
<tr>
<td>02</td>
<td>Dictates the color palette to be used</td>
</tr>
<tr>
<td>15</td>
<td>Indicates a function key was pressed.</td>
</tr>
<tr>
<td>20</td>
<td>Handles the clear screen action code</td>
</tr>
<tr>
<td>21</td>
<td>Handles the add action code</td>
</tr>
<tr>
<td>22</td>
<td>Handles the change action code</td>
</tr>
<tr>
<td>23</td>
<td>Handles the delete action code</td>
</tr>
<tr>
<td>24</td>
<td>Handles the inquire action code</td>
</tr>
<tr>
<td>25</td>
<td>Handles the inquire action code 'P' for print (payroll)</td>
</tr>
<tr>
<td>31</td>
<td>Used in conjunction with subfile processing to initiate the INVITE or SFLCLR keyword. Using INVITE will slow processing</td>
</tr>
<tr>
<td>32</td>
<td>Used in conjunction with subfile processing initiating the keyword SFLNXTCHG</td>
</tr>
<tr>
<td>37</td>
<td>Used in conjunction with subfile processing to avoid display of an empty subfile (used only with inquiry subfiles)</td>
</tr>
<tr>
<td>38</td>
<td>Used in conjunction with subfile processing to highlight the last record in the display (keyword SFLDSP) and avoid display of an empty subfile</td>
</tr>
<tr>
<td>40–79</td>
<td>Used for error processing to indicate which fields are in error and need to be highlighted</td>
</tr>
<tr>
<td>40</td>
<td>Reserved for errors in the Action Code field</td>
</tr>
<tr>
<td>41</td>
<td>Reserved for errors in the key fields</td>
</tr>
<tr>
<td>80–89</td>
<td>General reusable one–time indicators. Use them as needed.</td>
</tr>
<tr>
<td>93</td>
<td>Global error indicator that highlights line 24</td>
</tr>
<tr>
<td>98</td>
<td>Indicates a chain or read failure</td>
</tr>
<tr>
<td>99</td>
<td>Indicates a record is in use or file error</td>
</tr>
<tr>
<td>OF</td>
<td>Indicates overflow for report processing</td>
</tr>
<tr>
<td>LR</td>
<td>Indicates that the last record has been read and the program should end normally</td>
</tr>
<tr>
<td>RT</td>
<td>Indicates that a temporary or final halt in the program should take place. Returns to calling program leaving files open.</td>
</tr>
</tbody>
</table>
Documentation

In the F specifications the program contains several comment lines that are to serve as the program revisions log. The log should list all programmers who have revised the program, the date the revisions were made and the SAR outlining the change that was made.

When entering comment lines, use the following conventions.

- An asterisk in column seven specifies that the line is a comment line only.
- The asterisk should be followed by four blank spaces before the comment begins.
- Precede and follow the comment lines with a blank line.

The example below shows how these conventions are observed.

<table>
<thead>
<tr>
<th>Line</th>
<th>Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>0016.00</td>
<td>F* PROGRAM REVISION LOG</td>
</tr>
<tr>
<td>0017.00</td>
<td>F* -----------</td>
</tr>
<tr>
<td>0018.00</td>
<td>F* Date Programmer Nature of Revision</td>
</tr>
<tr>
<td>0020.00</td>
<td>F* ---------------</td>
</tr>
<tr>
<td>0021.00</td>
<td>F* AUTRF* 03/18/93 MARTIN SAR # 00000005 (AS/400 A/G)</td>
</tr>
<tr>
<td>0023.00</td>
<td>F* 05/01/93 RIPPEY SAR # 00167542</td>
</tr>
<tr>
<td>0034.00</td>
<td>F************</td>
</tr>
<tr>
<td>0035.00</td>
<td>F* Copy Member for Composite Common Subroutine -- C0001</td>
</tr>
<tr>
<td>0037.00</td>
<td>F*</td>
</tr>
</tbody>
</table>
**Guidelines**

Common sense should be your guide when documenting your programs. Be thorough and descriptive. Put yourself in the place of the next programmer who will inherit your work. Use English and not “programmerese” to specify the action occurring. For example, for the code shown below:

```
0130.00  C*  998  CASEQ‘  S998
0131.00  C    ----  ----
0132.00  C    END
```

- **DON’T WRITE:** If $998 is blank, execute S998.
- **INSTEAD WRITE:** Load data field dictionary parameters (one cycle only).

The following example gives more detail than can be inferred from the actual code.

Include a line of dashes beneath any line of code that branches to another line of code (CASxx, CABxx, GOTO, EXSR, CALL, BEGSR). The receiving tag statement should also be followed by a line of dashes as shown in the example below.

```
0275.00  C*  EXSR S999
0276.00  C    ----  ----
0277.00  C*    ----
```
**Miscellaneous Items**

The following represent miscellaneous items of note that you should keep in mind when writing your own code.

**Naming Conventions**

Use the following first character to distinguish different item names:

- @ Array names
- $ Program created field names (flags and work fields)
- # Fields defined in common subroutines

**Key List (KLIST)**

Key lists should all be defined in the housekeeping subroutine.

Begin the key list name with the data file prefix. For example, the Address Book Master file prefix is AB, so the key list would be ABKY01.

The Program Generator creates key lists using the following naming conventions:

- XXKY01 for physical files where XX = the file prefix. For example, ABKY01
- When a physical file needs to have more than one key list in a program, the successive files are noted in the last character space. For example, for three key lists for the physical file F0101, the key lists would be: ABKY01, ABKY02, and ABKY03.
- XXKY0x for logical files where XX is equal to the file prefix and x is equal to the last letter of the logical file name. For example: ABKY0A for F0101LA, ABKY0B for F0101LB.
- When a logical file needs to have more than one key list in a program, the successive files are noted in the second to last character space. For example for three key lists for the logical file F0101LA, the key lists would be: ABKY0A, ABKY1A, and ABKY2A.
Work Fields

Define work fields only once within a program. The use of the *LIKE DEFN command is highly recommended for defining work fields when their attributes are directly tied to those of database fields.

For example, if the work field needs to have the same attributes as a field that exists in a file:

```
MOVE ABANS $$ANS,
```

then define $$ANS as follows:

```
*LIKE DEFN ABANS $$ANS
```

The advantage of this method is that the work field and database field will retain the same attributes even if the database field changes.

When using work fields as a flag, you should assign them the prefix $ and have the remainder of the name be descriptive. In the example below, the work field name is $GLOBL. This name is more descriptive than a field name such as $G.

```
0831.00   C*  If F6 pressed, Global Update by Percent or Amount.
0832.00   C*  -----------------------------------------------
0833.00   C*  IF RDN #F03
0834.00   CSR  @RDN IF RDN #F03
0835.00   CSR  MOVE '1' $GLOBL 1
```

Optional Files

If a program uses files which are dependent upon your particular setup, you should designate those files as user control open (UC) in the file specifications and then write the program such that they are opened, if needed, in the Housekeeping subroutine. This eliminates the need to open files unnecessarily and conserves resources.

```
FF085201 UF E   K DISK UC
FF08501LA1F UF E   K DISK UC
```
The lines that perform the open are shown below.

<table>
<thead>
<tr>
<th>Columns . . :</th>
<th>1 71</th>
<th>Browse</th>
<th>JDFSRC/JDESRCC</th>
<th>P08320</th>
</tr>
</thead>
<tbody>
<tr>
<td>3825.00</td>
<td>C*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3826.00</td>
<td>C*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3827.00</td>
<td>C*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3828.00</td>
<td>CSR</td>
<td>OPEN</td>
<td>F085201</td>
<td>99</td>
</tr>
<tr>
<td>3829.00</td>
<td>CSR</td>
<td>*IN99</td>
<td>IFEQ '0'</td>
<td></td>
</tr>
<tr>
<td>3830.00</td>
<td>CSR</td>
<td>MOVE '1'</td>
<td>$PENS 1</td>
<td></td>
</tr>
<tr>
<td>3831.00</td>
<td>CSR</td>
<td>END</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3832.00</td>
<td>C*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3833.00</td>
<td>CSR</td>
<td>OPEN</td>
<td>F08501LA</td>
<td>99</td>
</tr>
<tr>
<td>3834.00</td>
<td>CSR</td>
<td>*IN99</td>
<td>IFEQ '0'</td>
<td></td>
</tr>
<tr>
<td>3835.00</td>
<td>CSR</td>
<td>MOVE '1'</td>
<td>$PENS2 1</td>
<td></td>
</tr>
<tr>
<td>3836.00</td>
<td>CSR</td>
<td>END</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

If you are doing a user-controlled open for a file that is part of another system, you will also need to provide pre-compiler commands in the event the user hasn’t purchased that system. The example below illustrates the necessary pre-compiler commands designed to address just such a situation.

In the example, if a Payroll client has not purchased Human Resources, the code specifies a file override and then substitutes an empty file (identified with the suffix E) which all Payroll clients receive.

```
****** Begin of data ******************************************************
0001.00      OVRRDBF  FILE(F082001B) TOFILE(F082001E)
0002.00      OVRRDBF  FILE(F08001) TOFILE(F08001E)
0003.00      OVRRDBF  FILE(F08005B) TOFILE(F08005E)
****** End of data ******************************************************
```

The user-controlled opens in the program allow the program to run in the absence of certain files, whereas the precompiler commands allow the program to be compiled in the absence of those files.
Program Structure

About Program Structure

There are several types of subroutines used in the J.D. Edwards program structure, including the following:

- Internal RPG Subroutines within J.D. Edwards programs
- Subfile program with selection exits
- Interactive non-subfile program
- Report program without subheadings
- Report program with subheadings
- Maintenance program without a subfile
Internal RPG Subroutines Within J.D. Edwards Programs

- Standard names make program maintenance easier.
- Called primarily from Mainline.

The table below describes internal RPG subroutines within J.D. Edwards programs:

<table>
<thead>
<tr>
<th>Name</th>
<th>Explanation</th>
</tr>
</thead>
</table>
| S00EX  | Processes all function key exits.  
|        | - Calls P9601H if F24 was pressed  
|        | - Calls X96CCX if F1 was pressed  
|        | - Calls subroutine S00VL if F1 was pressed after X96CCX was called  
|        | - Calls P0000E if F7 was pressed  
|        | - Calls P00HELP if the HELP key was pressed  
|        | - Calls subroutine S001 if F22 was pressed  
|        | - Calls all programs to process all user defined function keys  |
| S00VL  | Values returned with Cursor Sensitive Help.  
|        | Is called from the subroutine S00EX after the program X96CCX is called  |
| S00OP  | Subfile Selection Exits (Options).  |
| S001   | Clears all database and form fields.  
|        | - Usually only clears key fields and VC0 fields if F22 (Clear) is pressed  |
| S002   | Checks for level breaks for reports.  
|        | - Turns on level break flags.  
|        | - Retrieves total line description  |
| S003   | Validates the key fields.  
|        | Calls S998 subroutine if auto inquire was invoked  
|        | Sets the file pointer.  
|        | - Performs a SETLL or CHAIN if a single record maintenance program  
|        | - Performs a SETLL for subfile programs  
|        | Calls a subroutine S004 to load form or report fields  
|        | Monitors for no subfile records loaded if a subfile  
<p>|        | Loads unused subfile records with blanks  |
| S004   | Display or load form or report fields.  |</p>
<table>
<thead>
<tr>
<th>Name</th>
<th>Explanation</th>
</tr>
</thead>
</table>
| S005  | Scrubs and edits form fields.  
- Moves form data to database fields  
- Turns on error indicators if a field is in error  
- Updates or writes records to the database file if a subfile  
- Updates the subfile |
| S010  | For reports with level breaks it:  
- Prints the total  
- Clears the level break totals  
- Prints the grand total (if it has reached the end of the file)  
- Prints the detail  
- Adds to the new level break totals  
Calls subroutine S020 if it is a report with subheadings  
If it is *not* a report, it updates, adds, or deletes records from the database file  
- Turns on F22 (Clear) to force S001 to be executed to clear the buffer before reading another record. |
| S020  | Print Report Subheadings. |
| S998  | Loads Data Dictionary values. (One time only)  
- Retrieves row description for level breaks and subheadings, if applicable |
| S999  | Housekeeping. (One time only)  
- Sets auto inquiry  
- Defines key lists  
- Retrieves processing options and level breaks, if applicable  
- Retrieves vocabulary overrides  
- Loads error messages  
- Performs file opens  
- Current date retrieval  
- Work fields defined using *LIKE  
- Prints cover page and Helps in a report |
Interactive Non-Subfile Program

Mainline

- Set auto inquiry
- Key lists
- Retrieve processing options
- Retrieve vocabulary overrides
- Load error messages
- File opens
- Date retrieval

If auto inquiry

Write Forms

- One-time only - load Data Dictionary editing information
- Update or add records to file if no error

If auto inquiry

Read Forms

- Process Function keys
- Return F1 values to form fields

Action Code

Load form fields

Validate key fields set file pointer

Edit form data and move to file fields, turn on error indicators

Clears fields

Load error messages

If auto inquiry
Subfile Program with Selection Exits

Mainline

S001

Write Forms

S005

S003

Read Forms

S00EX

S00P

S00VL

S001

S004

C0001

S998

S999

Simulates the 'Clear Screen' function key to clear fields

Start at beginning of subfile and read each record. Edit the data. Turn on error indicator Update/write records to file If no errors. Update subfile

Validate key fields(s). Set file pointer Monitor for no subfile records loaded. Load remaining subfile records with blanks

Clear fields If 'Clear Screen' function key is pressed

Edit the action code

Process selection exits

Return F1 values to form fields

Set auto inquiry
Key lists
Retrieve processing options
Retrieve vocabulary overrides
Load error messages
File opens
Date retrieval

If auto inquiry

Load subfile records

Clear fields

One-time only - load Data Dictionary editing information

Clear fields

Load remaining subfile records with blanks

Process function keys

One-time only - load Data Dictionary editing information

Start at beginning of subfile and read each record. Edit the data. Turn on error indicator Update/write records to file If no errors. Update subfile

Validate key fields(s). Set file pointer Monitor for no subfile records loaded. Load remaining subfile records with blanks

Clear fields If 'Clear Screen' function key is pressed

Edit the action code

Process selection exits

Return F1 values to form fields
Report Program without Subheadings

**Mainline**

- Key lists
- Load vocabulary overrides
- File opens
- Print cover page and helps
- Retrieve processing options and level breaks
- Retrieve Data Dictionary editing information
- Retrieve row description for subheadings

**S999**

- If level break, print totals
- If level break, clear totals
- If end of file, print grand total
- Print detail
- Add to totals

**S998**

- Load report fields

**C0000**

- Check cost center security

**S010**

- Check for level breaks
- Set level break flag(s)
- Retrieve total line description

**S004**

- Read a Record

**S002**
Report Program with Subheadings

Mainline

S999
Key lists
Load vocabulary overrides
File opens
Print cover page and helps
Retrieve processing options and
level breaks

S998
Retrieve Data Dictionary
editing information
Retrieve row description
for subheadings

S010
Read a Record
If level break, print totals
If level break, clear totals
If end of file, print grand total
Print detail
Add to totals

C0000
Check cost center security

S004
Load report fields

S002
Check for level breaks
Set level break flag(s)
Retrieve total line description

S020
Print subheadings
if overflow

S020
Print subheadings
Review an RPG Program’s Source

The following pages illustrate a maintenance program without a subfile.

Some of the more important areas and commonly used fields are highlighted and explained.
Item Master Information

Copyright (c) 1993 J. D. Edwards & Company

This unpublished material is proprietary to J. D. Edwards & Company. All rights reserved.

The methods and techniques described herein are considered trade secrets and/or confidential.

Reproduction or distribution, in whole or in part, is forbidden except by express written permission of J. D. Edwards & Company.

PROGRAM REVISION LOG

Date       Programmer    Nature of Revision
----------- ----------- ----------------
12/07/93    QUARLES     SAR # 241883 (AS/400 A/G)

PROGRAM TABLES AND ARRAYS

EMK  64  4 Error Msg
@MK  64  1 Error Msg
@ER  64  4 Error Msg
@DV  40  1 Dflt Wrk
@C  256  1 Literal Work

Copy Member for Composite Common Subroutine - C0001

COPY MEMBER FOR COMPOSITE COMMON SUBROUTINE - C0001

Will copy in additional specifications for copy module C0001

Arrays that handle error messages

Shows all SARs used to make changes to the program

The Program Generator puts in numeric order. RPG opens from bottom to top so JDE puts more heavily used files at the bottom.

Informational data structure for the video
<table>
<thead>
<tr>
<th>Data Structure to Load Video Screen Text</th>
</tr>
</thead>
<tbody>
<tr>
<td>IDSTXT       DS 1000</td>
</tr>
<tr>
<td>I 1 18 VTX001</td>
</tr>
<tr>
<td>I 41 58 VTX002</td>
</tr>
<tr>
<td>I 81 92 VTX003</td>
</tr>
<tr>
<td>I 121 138 VTX004</td>
</tr>
<tr>
<td>I 161 178 VTX005</td>
</tr>
<tr>
<td>I 201 218 VTX006</td>
</tr>
<tr>
<td>I 241 258 VTX007</td>
</tr>
<tr>
<td>I 281 298 VTX008</td>
</tr>
<tr>
<td>I 321 338 VTX009</td>
</tr>
<tr>
<td>I 361 378 VTX010</td>
</tr>
<tr>
<td>I 401 418 VTX011</td>
</tr>
<tr>
<td>I 441 458 VTX012</td>
</tr>
<tr>
<td>I 481 498 VTX013</td>
</tr>
<tr>
<td>I 521 536 VTX014</td>
</tr>
<tr>
<td>I 561 576 VTX015</td>
</tr>
<tr>
<td>I 601 616 VTX016</td>
</tr>
<tr>
<td>I 641 656 VTX017</td>
</tr>
<tr>
<td>I 681 696 VTX018</td>
</tr>
<tr>
<td>I 721 736 VTX019</td>
</tr>
<tr>
<td>I 761 776 VTX020</td>
</tr>
<tr>
<td>I 801 816 VTX021</td>
</tr>
<tr>
<td>I 841 856 VTX022</td>
</tr>
<tr>
<td>I 881 896 VTX023</td>
</tr>
<tr>
<td>I 921 936 VTX024</td>
</tr>
<tr>
<td>I 961 976 VTX025</td>
</tr>
</tbody>
</table>

Data structure for commonly used indexes

Data structure used with file servers

Program status data structure

Copy Member for Composite Common Subroutine - C00SC

Copy Member for Server - X0005

Copy Member for Server - X0006

Copy Member for Server - X9800E

C* MAINLINE PROGRAM

EXSR S999 One time only functions

If information is passed to this program, it will automatically inquire on the record

If LR on, end program.

If automatic inquiry set, process inquiry.

Write video screen.
150.00 C WRITEV9280111
151.00 C  MOVE '1' @@AID  
152.00 C CXSR S001  
153.00 C* –––– ––––  
154.00 C*      Load data field dictionary parameters (one cycle only).  
155.00 C*  
156.00 C* $998 CASEQ' ' $998  
157.00 C* –––––––– ––––  
158.00 C*      Begin video screen read processing.  
159.00 C*  
160.00 C* SETOF 999301  
161.00 C* READ V92801   9998  
162.00 C* Z–ADD0 ##RROW  
163.00 C* Z–ADD0 ##RCOL  
164.00 C*  
165.00 C*       If video read timed out, end program.  
166.00 C*  
167.00 C* *IN99 CABEQ'1' EOJ LR  
168.00 C*  
169.00 C*       If valid function key pressed, process and return.  
170.00 C*  
171.00 C* $998 CASEQ' ' $998  
172.00 C* –––––––– ––––  
173.00 C*  
174.00 C*       Edit the action code.  
175.00 C*  
176.00 C* EXSR C0001  
177.00 C*  
178.00 C*       If end of job requested, end program.  
179.00 C*  
180.00 C* @&AID CABEQ#FEOJ EOJ LR  
181.00 C*  
182.00 C*       If clear screen requested, process and return.  
183.00 C*  
184.00 C*  
185.00 C*  
186.00 C*  
187.00 C*  
188.00 C*  
189.00 C*  
190.00 C*  
191.00 C*  
192.00 C*  
193.00 C*  
194.00 C*  
195.00 C*  
196.00 C*  
197.00 C*  
198.00 C*  
199.00 C*  
200.00 C*  
201.00 C*  
202.00 C*  
203.00 C*  
204.00 C*  
205.00 C*  
206.00 C*  
207.00 C*  
208.00 C*  
209.00 C*  
210.00 C*  
211.00 C*  
212.00 C*  
213.00 C*  
214.00 C*  
215.00 C*  
216.00 C*  
217.00 C*  
218.00 C*  
219.00 C*  
220.00 C*  
221.00 C*  
222.00 C*  
223.00 C*  
224.00 C*  
225.00 C*  
226.00 C*  
227.00 C*  
228.00 C*  

---

End of program.
Set correct message in line 24.

Sets the message for Line 24

Contains what function key was pressed by the user

Values assigned in the Function Key Definitions program

External programs start with an X. This is the cursor sensitive help program

Parameters passed identifying where the cursor was when F1 was pressed
A8.1 (8/97)  
4–37

If Display errors press, exit to error messages.

CSR @@AID IFEQ #FERRD
CSR Z–ADD1 #G
CSR Z–ADD1 #H
CSR #G DOWLE64
CSR 0MK,#G IFEQ ‘1’
CSR MOVE EMK, #G @ER, #H
CSR END
CSR ADD 1 #H
CSR CALL ‘P0000E’ 98
CSR PARM @ER
CSR GOTO ENDEXE
CSR END

If HELP key pressed, exit to help facility and return.

CSR @@AID IFEQ #FHELP
CSR CALL ‘P00HELP’ 99
CSR PARM HS@@
CSR PARM HE@@
CSR PARM I00SC
CSR PARM SRVIDS
CSR GOTO ENDEXE
CSR END
CSR END

If Clear screen pressed, clear screen and return.

CSR @@AID IFEQ #FROLU
CSR @@AID OREQ #FROLD
CSR $SECUR DOUEQ’ ‘
CSR MOVE ‘ ‘ $SECUR 1
CSR $RUKEY SETLLI92801
CSR SETOF 8299
CAR READ192801
CSR $IN81 IFEQ ‘1’
CSR $RUKEY SETLLI92801
CSR SETOF 8299
CAR READ192801
CSR END
CSR END

If ROLL UP key pressed, process read next.

If Clear screen pressed, clear screen and return.

If error on read, set error.

CSR MOVEA$RESET *IN, 41
CSR MOVE ‘0’ *IN, 40
CSR SETOF 818299
CSR READ 192801 9981
CSR $IN81 IFEQ ‘1’
CSR $RUKEY SETLLI92801
CSR SETOF 8299
CAR READ192801 9982
CSR END
CSR END

Reset error indicators if roll

If error on read, set error.

CSR *IN82 IFEQ ‘1’
CSR SETON 9341
CSR MOVE ‘1’ @MK,2
CSR GOTO ENDEXE
CSR END
CSR END
If ROLL DOWN key pressed, process read prior.

Reset error indicators if roll

Load video screen data on roll keys.

Release record lock or report record in use.

Cost Center security edit.

Program that will display a record lock window when a record in use error is encountered

Could not find a match in the Function Key Definitions for the function key pressed, so program displays Invalid Function Key message.
SUBROUTINE SG OVL – Cursor Control Return Values

By format, find the field to update and move in the returned value. If the format is a subfile, the record to change is found in @@RRN.

RETURN values for fields in format V9280111

##RFMT IFEQ 'V9280111'
##FLDN IFEQ 'ACTION        '
MOVEL##RVAL ACTION
GOTO ENDOVL

##FLDN IFEQ 'VDXIT        '
MOVEL##RVAL VDXIT
GOTO ENDOVL

##FLDN IFEQ 'VDXDS        '
MOVEL##RVAL VDXDS
GOTO ENDOVL

##FLDN IFEQ 'VDXCC        '
MOVEL##RVAL VDXCC
GOTO ENDOVL

##FLDN IFEQ 'VDXTY        '
MOVEL##RVAL VDXTY
GOTO ENDOVL

##FLDN IFEQ 'VDXDT        '
MOVEL##RVAL VDXDT
GOTO ENDOVL

##FLDN IFEQ 'VDXQT        '
MOVEL##RVAL VDXQT
GOTO ENDOVL

##FLDN IFEQ 'VDXUM        '
MOVEL##RVAL VDXUM
GOTO ENDOVL

##FLDN IFEQ 'VDX001       '
MOVEL##RVAL VDX001
GOTO ENDOVL
SUBROUTINE S001 – Clear Fields

Processing: 1. Reset all video screen and data file fields for next transaction.
2. Clear action code only if requested.

Reset fields for next transaction.

Clears all the fields in the record format for F92801

C* Clear action code only if clear screen action.

C* - Clear the video fields

These fields will only be cleared if the user presses the function key to clear the screen. We want to save certain information like key fields and descriptions so they don’t get cleared every time S001 is executed.
SUBROUTINE S003 – Edit Key

Sets the file pointer and edits the key.

Processing:
1. Clear error indicators and arrays.
2. Load input keys.
3. Validate master file key.
4. Release master file record lock.
5. Load video screen output on inquiry.

                   2. Load input keys.
                   3. Validate master file key.
                   4. Release master file record lock.
                   5. Load video screen output on inquiry.

CSR S003 BEGSR

Load data field dictionary parameters (one cycle only).

CSR $998 CASEQ ‘’ S998

CSR END

Reset error indicators and arrays.

CSR MOVE *ALL’0’ $RESET 39
CSR MOVE *BLANK $RESTl 63
CSR MOVEA$RESET *IN,41
CSR MOVEA$RESTl @MK,2
CSR CLEAR@ER

Load video input field for – Item ID

CSR MOVEAVDXIT @NM
CSR EXSR C0012

Automatic Next Number for – Item ID

CSR *IN21 IFEQ ‘1’
CSR VDXIT ANDEQ*BLANK
CSR SETON 81
CSR *IN81 DOWEQ‘1’
CSR MOVE NIDXIT PSIDX 2
CSR CALL ‘X0010’ 82
CSR END

Automatic Next Number for – Item ID

CSR QXKY01 CHAINI92801 9899
CSR QXXIT SETLLF92801 8281
CSR END

Cost Center security edit.

CSR QXKY01 CHAINI92801 9899
CSR QXXIT SETLLF92801 8281
CSR END

Checks cost center security

CSR MOVEL’P92801 ‘#FILE
CSR MOVELQXXCC #MCU
CSR PARM *ZERO #NXTNO 80
CSR PARM #NXTNO VDXIT
CSR CALL ‘X0010’ 82
CSR END

If security violation, set error condition.

CSR $$SECR IFEQ ‘1’
CSR MOVE ‘1’ @MK,8
CSR SETON 9341

CSR END

CSR $$SECR IFEQ ‘1’
CSR MOVE ‘1’ @MK,8
CSR SETON 9341
701.00  CSR  MOVE ' ' $$SEC R 1  
702.00  CSR  GOTO END003  
703.00  C*  ------ ------  
704.00  CSR  END  
705.00  C*  
706.00  C*  Edit result of read and action code.  
707.00  C*  
708.00  CSR  *IN98  IFEQ '1'  
709.00  CSR  *IN21  COMP '0'  ------  41 "error"  
710.00  CSR  ELSE  
711.00  CSR  *IN21  COMP '1'  41 "error"  
712.00  CSR  END  
713.00  C*  
714.00  C*  If indicator 41 on, invalid key for action code.  
715.00  C*  
716.00  CSR  *IN41  IFEQ '1'  
717.00  CSR  MOVE '1'  @MK,2  
718.00  CSR  SETON 93  
719.00  CSR  END  
720.00  C*  
721.00  C*  If indicator 99 on, record in use.  
722.00  C*  
723.00  CSR  *IN99  IFEQ '1'  
724.00  CSR  CALL '98RLCK' 81  
725.00  C*  ------ ------  
726.00  CSR  PARM ##PSDS  
727.00  CSR  MOVE '1'  @MK,6  
728.00  CSR  SETON 9341  
729.00  CSR  END  
730.00  C*  
731.00  C*  If not inquiry, skip remainder of subroutine.  
732.00  C*  
733.00  CSR  *IN24  CABEQ'0'  END003  
734.00  C*  ------ ------  
735.00  C*  
736.00  C*  
737.00  C*  Release record lock on master file.  
738.00  C*  
739.00  CSR  *IN98  IFEQ '0'  
740.00  CSR  *IN99  ANDEQ'0'  
741.00  CSR  EXCPTUNLOCK  
742.00  CSR  END  
743.00  C*  
744.00  C*  If errors, skip remainder of subroutine.  
745.00  C*  
746.00  C*  
747.00  CSR  *IN93  CABEQ'1'  END003  
748.00  C*  ------ ------  
749.00  C*  
750.00  C*  
751.00  C*  Move data base information to video screen.  
752.00  C*  
753.00  CSR  --------  
754.00  C*  
755.00  C*  
756.00  CSR  END003  ENDSR  
757.00  C*  
758.00  C*  Copy Common Subroutine - Right Justify Numeric Fields  
759.00  C*  
760.00  C*  
761.00  C/COPY JDECPY,C0012  
762.00  C*  
763.00  C*  SUBROUTINE S004 - Load Video Screen Data  
764.00  C*  
765.00  C*  
766.00  C*  
767.00  C*  Processing: 1. Move data base information to video screen.  
768.00  C*  
769.00  C*  All video screen fields are alpha and therefore numeric information must be  
770.00  C*  processed through subroutine C0014 to set  
771.00  C*  proper decimals and provide editing for  
772.00  C*  display on screen  
773.00  C*  
774.00  C*  Date fields must be converted from their  
775.00  C*  internal format of month, day and year or  
776.00  C*  julian to the system format using program X0028  
777.00  C*  

---

JDE uses this or SETLL to release record locks  
Moves information to the video/report fields
778.00 C*  
779.00 CSR S004 BEGSR  
780.00 C*  
781.00 C*  
782.00 C*  
783.00 C* Move to output - Description for Cost Center  
784.00 C*  
785.00 CSR CALL 'X0006' 81  
786.00 C*  
787.00 CSR PARM *BLANKS PSOMOD 1  
788.00 CSR PARM '1' PSIMOD 1  
789.00 CSR PARM QXXCC PSMCU 12  
790.00 CSR PARM *BLANKS PSERRM 4  
791.00 CSR PARM I0006  
792.00 C*  
793.00 CSR PSERRM MOVE *BLANK VC0001  
794.00 CSR MOVELMCDL01 VC0001  
795.00 CSR END  
796.00 CSR C*  
797.00 C*  
798.00 C*  
799.00 C* Description display for - Item Type  
800.00 C*  
801.00 CSR CLEARI0005U  
802.00 CSR MOVELS$XTY #USY  
803.00 CSR MOVE R8XTY #URT  
804.00 CSR MOVE QXXTY #UKY  
805.00 CSR CALL 'X0005' 81  
806.00 C*  
807.00 CSR PARM I0005U  
808.00 CSR MOVE *BLANK VC0002  
809.00 CSR IFEQ '0'  
810.00 CSR MOVEL#UDL01 VC0002  
811.00 CSR END  
812.00 C*  
813.00 C*  
814.00 C* Description display for - Item Unit of Measure  
815.00 C*  
816.00 CSR CLEARI0005U  
817.00 CSR MOVELS$XUM #USY  
818.00 CSR MOVE R8XUM #URT  
819.00 CSR MOVE QXXUM #UKY  
820.00 CSR CALL 'X0005' 81  
821.00 C*  
822.00 CSR PARM I0005U  
823.00 CSR MOVE *BLANK VC0003  
824.00 CSR IFEQ '0'  
825.00 CSR MOVEL#UDL01 VC0003  
826.00 CSR END  
827.00 C*  
828.00 C*  
829.00 C* Description display for - Item Category Code 001  
830.00 C*  
831.00 CSR CLEARI0005U  
832.00 CSR MOVELS$X0001 #USY  
833.00 CSR MOVE R8X001 #URT  
834.00 CSR MOVE QXX001 #UKY  
835.00 CSR CALL 'X0005' 81  
836.00 C*  
837.00 CSR PARM I0005U  
838.00 CSR MOVE *BLANK VC0004  
839.00 CSR IFEQ '0'  
840.00 CSR MOVEL#UDL01 VC0004  
841.00 CSR END  
842.00 C*  
843.00 C*  
844.00 C* Description display for - Item Category Code 002  
845.00 C*  
846.00 CSR CLEARI0005U  
847.00 CSR MOVELS$X0002 #USY  
848.00 CSR MOVE R8X002 #URT  
849.00 CSR MOVE QXX002 #UKY  
850.00 CSR CALL 'X0005' 81  
851.00 C*  
852.00 CSR PARM I0005U  
853.00 CSR MOVE *BLANK VC0005  
854.00 CSR IFEQ '0'  
855.00 CSR END
855.00  CSR               MOVEL#UDL01 VC0005
856.00  CSR               END
857.00  C*-----------------------------------------------------
858.00  C* Description display for - Item Category Code 003
859.00  C*-----------------------------------------------------
860.00  C*-----------------------------------------------------
861.00  CSR               CLEARI0005U
862.00  CSR               MOVE#S@X003 USY  
863.00  CSR               MOVE R@X003 URT  
864.00  CSR               MOVE QXX003 UKY  
865.00  CSR               CALL 'X0005' 81
866.00  C*-----------------------------------------------------
867.00  CSR               PARM I0050U
868.00  CSR               MOVE 'BLANK VC0006
869.00  CSR               #UERR IFEQ '0'
870.00  CSR               MOVEL#UDL01 VC0006
871.00  CSR               END
872.00  C*-----------------------------------------------------
873.00  C* Description display for - Item Category Code 004
874.00  C*-----------------------------------------------------
875.00  C*-----------------------------------------------------
876.00  CSR               CLEARI0005U
877.00  CSR               MOVE#S@X004 USY  
878.00  CSR               MOVE R@X004 URT  
879.00  CSR               MOVE QXX004 UKY  
880.00  CSR               CALL 'X0005' 81
881.00  C*-----------------------------------------------------
882.00  CSR               PARM I0050U
883.00  CSR               MOVE 'BLANK VC0007
884.00  CSR               #UERR IFEQ '0'
885.00  CSR               MOVEL#UDL01 VC0007
886.00  CSR               END
887.00  C*-----------------------------------------------------
888.00  C* Description display for - Item Category Code 005
889.00  C*-----------------------------------------------------
890.00  C*-----------------------------------------------------
891.00  CSR               CLEARI0005U
892.00  CSR               MOVE#S@X005 USY  
893.00  CSR               MOVE R@X005 URT  
894.00  CSR               MOVE QXX005 UKY  
895.00  CSR               CALL 'X0005' 81
896.00  C*-----------------------------------------------------
897.00  CSR               PARM I0050U
898.00  CSR               MOVE 'BLANK VC0008
899.00  CSR               #UERR IFEQ '0'
900.00  CSR               MOVEL#UDL01 VC0008
901.00  CSR               END
902.00  C*-----------------------------------------------------
903.00  C* Move to output - Cost Center
904.00  C*-----------------------------------------------------
905.00  C*-----------------------------------------------------
906.00  CSR               MOVE *BLANK #SINBR
907.00  CSR               MOVEL#XXCC #SINBR
908.00  CSR               MOVE#XXCC #UDF1
909.00  CSR               MOVE#XXCC #UDMSD  
910.00  CSR               MOVE@XCC #EC  
911.00  CSR               MOVE F@XCC #DSPD
912.00  CSR               MOVE G@XCC #DATD
913.00  CSR               MOVE J@XCC #ALR
914.00  CSR               MOVE ' ' #ECOR
915.00  CSR               MOVE ' ' #DCOR
916.00  CSR               EXSR C00161
917.00  C*-----------------------------------------------------
918.00  CSR               #ALR IFEQ 'L'
919.00  CSR               MOVEL#SINBR VDXCC
920.00  CSR               ELSE
921.00  CSR               MOVE #SINBR VDXCC
922.00  CSR               END
923.00  C*-----------------------------------------------------
924.00  C* Move to output - Description
925.00  C*-----------------------------------------------------
926.00  C*-----------------------------------------------------
927.00  CSR               MOVEL#XXDS VDXDS
928.00  C*-----------------------------------------------------
929.00  C*-----------------------------------------------------
930.00  C* Move to Output - Date Last Ship
931.00  C*-----------------------------------------------------

Editing information retrieved in S998
Copy module to edit field for use on screen/report
A8.1 (8/97)

4–45

932.00 CSR MOVE QXXDT #SIDAT 6
933.00 CSR MOVE *BLANK #EDAT 8
934.00 CSR MOVEQ XJUL #PPMT 7
935.00 CSR MOVEQ SYSVAL #TPMT 7
936.00 CSR MOVEQ SYSVAL #SEP 7
937.00 CSR MOVE ' ' SERITST 1
938.00 CSR CALL 'XQ028' 81
939.00 C*---- ------
940.00 CSR PARM #SIDAT
941.00 CSR PARM #EDAT
942.00 CSR PARM #FFMT
943.00 CSR PARM #TFMT
944.00 CSR PARM #SEP
945.00 CSR PARM SERITST
946.00 CSR MOVEL #EDAT VDXDT
947.00 C*---------------------------------------
948.00 C*
949.00 C* Move to output - Item ID
950.00 C*---------------------------------------
951.00 CSR MOVE *BLANK #SINBR
952.00 CSR MOVEQ XQXIT #SINBR
953.00 CSR MOVE W0XIT #EWRD
954.00 CSR MOVE 0XQX #DTYP
955.00 CSR MOVE 00XQX #DATD
956.00 CSR MOVE 0IXQX #ALR
957.00 CSR MOVE ' ' #ECOR
958.00 CSR MOVE ' ' #DCOR
959.00 CSR EXSR C00161
960.00 C*---- ------
961.00 CSR #ALR IFEQ 'L'
962.00 CSR MOVEL #SINBR VDXIT
963.00 CSR ELSE
964.00 CSR MOVEL #SINBR VDXIT
965.00 CSR END
966.00 C*---------------------------------------
967.00 C*
968.00 C* Move to output - Quantity - On Hand
969.00 C*---------------------------------------
970.00 CSR MOVE *BLANK #SINBR
971.00 CSR MOVEQ XQXQT #SINBR
972.00 CSR MOVE T0XQT #DTYP
973.00 CSR MOVE W0XQT #EWRD
974.00 CSR MOVE 0XQX #DTYP
975.00 CSR MOVE 00XQX #DATD
976.00 CSR MOVE 0IXQX #ALR
977.00 CSR MOVE ' ' #ECOR
978.00 CSR MOVE ' ' #DCOR
979.00 CSR EXSR C00161
980.00 C*---- ------
981.00 CSR #ALR IFEQ 'L'
982.00 CSR MOVEL #SINBR VDXIT
983.00 CSR ELSE
984.00 CSR MOVEL #SINBR VDXIT
985.00 CSR END
986.00 C*---------------------------------------
987.00 C*
988.00 C* Move to output - Item type
989.00 C*---------------------------------------
990.00 CSR MOVEQ XQXTY VDXTY
991.00 CSR MOVEQ XQXUM VDXUM
992.00 CSR MOVEQ XQX001 VDX001
993.00 CSR MOVEQ XQX001 VDX001
994.00 C*---------------------------------------
995.00 C* Move to output - Item Unit of Measure
996.00 C*---------------------------------------
997.00 CSR MOVELQXUM VDXUM
998.00 CSR MOVELQX001 VDX001
999.00 C*---------------------------------------
1000.00 C* Move to output - Item Category Code 001
1001.00 C*---------------------------------------
1002.00 C*
1003.00 C* Move to output - Item Category Code 001
1004.00 C*---------------------------------------
1005.00 C*
1006.00 C*
1007.00 C*
1008.00 C*---------------------------------------
1009.00 C*---------------------------------------

External program used to edit dates
1010.00 CSR MOVE J@X001 #ALR
1011.00 CSR MOVE ' ' #ECOR
1012.00 CSR MOVE ' ' #DCOR
1013.00 CSR EXSR C00161
1014.00 C* ---- ------
1015.00 CSR #ALR IFEQ 'L'
1016.00 CSR MOVE@#SINBR VDX001
1017.00 CSR ELSE
1018.00 CSR MOVE #SINBR VDX001
1019.00 CSR END
1020.00 C*-----------------------------
1021.00 C* Move to output – Item Category Code 002
1022.00 C* 1023.00 C*
1024.00 CSR MOVE *BLANK #SINBR
1025.00 CSR MOVE@#SINBR VDX002
1026.00 CSR MOVE T@X002 #DTYP
1027.00 CSR MOVE W@X002 #ENRD
1028.00 CSR MOVE E@X002 #EC
1029.00 CSR MOVE F@X002 #DSPD
1030.00 CSR MOVE G@X002 #DATD
1031.00 CSR MOVE J@X002 #ALR
1032.00 CSR MOVE ' ' #ECOR
1033.00 CSR MOVE ' ' #DCOR
1034.00 CSR EXSR C00161
1035.00 C* ---- ------
1036.00 CSR #ALR IFEQ 'L'
1037.00 CSR MOVE@#SINBR VDX002
1038.00 CSR ELSE
1039.00 CSR MOVE #SINBR VDX002
1040.00 CSR END
1041.00 C*-----------------------------
1042.00 C* 1043.00 C* Move to output – Item Category Code 003
1044.00 C* 1045.00 CSR MOVE *BLANK #SINBR
1046.00 CSR MOVE@#SINBR VDX003
1047.00 CSR MOVE T@X003 #DTYP
1048.00 CSR MOVE W@X003 #ENRD
1049.00 CSR MOVE E@X003 #EC
1050.00 CSR MOVE F@X003 #DSPD
1051.00 CSR MOVE G@X003 #DATD
1052.00 CSR MOVE J@X003 #ALR
1053.00 CSR MOVE ' ' #ECOR
1054.00 CSR MOVE ' ' #DCOR
1055.00 CSR EXSR C00161
1056.00 C*-----------------------------
1057.00 CSR #ALR IFEQ 'L'
1058.00 CSR MOVE@#SINBR VDX003
1059.00 CSR ELSE
1060.00 CSR MOVE #SINBR VDX003
1061.00 CSR END
1062.00 C*-----------------------------
1063.00 C* 1064.00 C* Move to output – Item Category Code 004
1065.00 C* 1066.00 CSR MOVE *BLANK #SINBR
1067.00 CSR MOVE@#SINBR VDX004
1068.00 CSR MOVE T@X004 #DTYP
1069.00 CSR MOVE W@X004 #ENRD
1070.00 CSR MOVE E@X004 #EC
1071.00 CSR MOVE F@X004 #DSPD
1072.00 CSR MOVE G@X004 #DATD
1073.00 CSR MOVE J@X004 #ALR
1074.00 CSR MOVE ' ' #ECOR
1075.00 CSR MOVE ' ' #DCOR
1076.00 CSR EXSR C00161
1077.00 C* ---- ------
1078.00 CSR #ALR IFEQ 'L'
1079.00 CSR MOVE@#SINBR VDX004
1080.00 CSR ELSE
1081.00 CSR MOVE #SINBR VDX004
1082.00 CSR END
1083.00 C*-----------------------------
1084.00 C* 1085.00 C* Move to output – Item Category Code 005
1086.00 C*
1087.00 CSR MOVE *BLANK #$SINBR
1088.00 CSR MOVE LQXX005 #$SINBR
1089.00 CSR MOVE TX005 #DTYP
1090.00 CSR MOVE WX005 #$EMRD
1091.00 CSR MOVE EX005 #EC
1092.00 CSR MOVE FX005 #$DSPD
1093.00 CSR MOVE G8X005 #$DATD
1094.00 CSR MOVE J8X005 #ALR
1095.00 CSR MOVE ' ' #ECOR
1096.00 CSR MOVE ' ' #DCOR
1097.00 CSR EXSR C00161
1098.00 CSR #ALR IFEQ 'L'
1100.00 CSR #SINBR VOX005
1102.00 CSR MOVE #SINBR VDX005
1103.00 CSR END
1104.00 CSR END004 ENDSR
1106.00 C***************************************************************
1107.00 C* Copy Common Subroutine – Format Numeric Fields for Output with Override
1109.00 C* C/COPY JDECOPY,C00161
1110.00 C***************************************************************
1112.00 C* SUBROUTINE S005 – Scrub Input
1113.00 C* Validates and edits data entered by the user
1114.00 C* -------------------------------------------------------------------
1115.00 C* If not addition or change, bypass subroutine. Only performs this
1116.00 C* Processing: 1. Validate all video input. subroutine if action code is
1117.00 C* All numeric fields must be processed  add or change
1118.00 C* thru subroutines C0012 and C0015 in order
1119.00 C* to scrub the alpha input field and convert
1120.00 C* 15 digits and 0 decimals.
1121.00 C* Date fields must be converted from system
1122.00 C* format to their internal format of month,
1123.00 C* day and year or julian using program X0028.
1124.00 C* 2. Update data record fields from video.
1125.00 C* Scrub and edit – Cost Center
1126.00 C* Scrub and edit – Description
1127.00 CSR S005 BEGSR
1128.00 CSR #ALR IFEQ 'L'
1129.00 CSR SETON 4393
1131.00 CSR MOVE *BLANK EMK,10
1132.00 CSR MOVE ' ' @MK,10
1133.00 CSR CALL 'X0006' 99
1134.00 CSR PARM '1' PSOMOD 1
1135.00 CSR PARM ' ' PSIMOD 1
1136.00 CSR PARM VDXCC PSMCU 12
1137.00 CSR PARM *BLANKS PSERRM 4
1138.00 CSR PARM 10006
1139.00 CSR PSERRM IFNE *BLANK
1140.00 CSR SETON 4393
1141.00 CSR MOVE #SINBR QXXCC
1142.00 CSR MOVE LQXX005 #SINBR
1143.00 CSR MOVE TX005 #DTYP
1144.00 CSR MOVE WX005 #$EMRD
1145.00 CSR MOVE EX005 #EC
1146.00 CSR MOVE FX005 #$DSPD
1147.00 CSR MOVE G8X005 #$DATD
1148.00 CSR MOVE J8X005 #ALR
1149.00 CSR MOVE ' ' #ECOR
1150.00 CSR MOVE ' ' #DCOR
1151.00 CSR CSR PSERRM IFNE *BLANK
1152.00 CSR SETON 4393
1153.00 CSR MOVE #SINBR QXXCC
1154.00 CSR MOVE ' ' QXXDS
1155.00 CSR END
1156.00 CSR MOVE PSMCU #SINBR
1157.00 CSR END004 ENDSR
1158.00 CSR #ALR IFEQ 'L'
1159.00 CSR GOTO END005
1160.00 CSR SETON 4393
1161.00 CSR MOVE #SINBR QXXCC
1162.00 CSR MOVE LQXX005 #SINBR
1163.00 CSR MOVE TX005 #DTYP
1164.00 CSR MOVE WX005 #$EMRD
1166.00 CSR END004 ENDSR
1199.00 CSR SETON 4393
1200.00 CSR MOVE #SINBR VDX005
1201.00 CSR END
1202.00 CSR END004 ENDSR
```
1165.00  CSR  QXXDS  IFEQ *BLANK
1166.00  CSR  D@XDS  IFNE *BLANK
1167.00  CSR  MOVEA@XDS @DV
1168.00  CSR  MOVEA@DV QXXDS
1169.00  CSR  @DV,1  IFEQ
1170.00  CSR  MOVE ' ' @DV,1
1171.00  CSR  Z-ADD2
1172.00  CSR  #M  DOWLE40
1173.00  CSR  @DV,#M  IFEQ ' '
1174.00  CSR  MOVE ' ' @DV,#M
1175.00  CSR  END
1176.00  CSR  ADD 1 #M
1177.00  CSR  END
1178.00  CSR  MOVEA@DV,2 QXXDS
1179.00  CSR  END
1180.00  CSR  END
1181.00  CSR  END
1182.00  C*  Edit allowed values - Description
1183.00  C*  Edit allowed values - Description
1184.00  CSR  A@XDS IFEQ '*NB'
1185.00  CSR  QXXDS ANDEQ*BLANK
1186.00  CSR  MOVE '1' @MK,03
1187.00  CSR  SETON 4293
1188.00  CSR  END
1189.00  CSR  END

1190.00  C*
1191.00  C*  Scrub and edit - Date Last Ship
1192.00  C*  Scrub and edit - Date Last Ship
1193.00  CSR  MOVEAVDXDT @NM
1194.00  CSR  EXSR C0012
1195.00  C* –––– –––––
1196.00  CSR  Z-ADD#NUMR $NBR6 60
1197.00  CSR  MOVE $NBR6 QXXDT
1198.00  C* –––– –––––
1199.00  CSR  PARM #SIDAT
1200.00  CSR  PARM #EDAT
1201.00  CSR  PARM #FFMT
1202.00  CSR  PARM #TFMT
1203.00  CSR  PARM #SEP
1204.00  CSR  PARM $ERTST
1205.00  CSR  MOVEL#SIDAT QXXDT
1206.00  CSR  $ERTST IFEQ '1'
1207.00  CSR  MOVE '1' @MK,04
1208.00  CSR  SETON 4593

1209.00  C*  Set default value - Item ID
1210.00  C*  Set default value - Item ID
1211.00  CSR  VDXIT IFNE *BLANK
1212.00  CSR  F@XIT #DSPD
1213.00  CSR  PARM #SIDAT
1214.00  CSR  PARM #EDAT
1215.00  CSR  PARM #FFMT
1216.00  CSR  PARM #SEP
1217.00  CSR  PARM $ERTST
1218.00  CSR  MOVEF@XIT #FFMT 7
1219.00  CSR  MOVEL*JUL '#TFMT 7
1220.00  CSR  MOVEL*NONE '#SEP 7
1221.00  CSR  CALL 'X0028 99
1222.00  C* –––– –––––
1223.00  CSR  PARM #SIDAT
1224.00  C*  Scrub and edit - Item ID
1225.00  C*  Scrub and edit - Item ID
1226.00  CSR  MOVEA@XIT @NM
1227.00  CSR  EXSR C0012
1228.00  C*  --- -----
1229.00  CSR  MOVEL$ERTST QXXIT
1230.00  CSR  MOVE $ERTST IFEQ '1'
1231.00  CSR  MOVEF@XIT #DSPD
1232.00  CSR  EXSR C00151
1233.00  C*  --- -----
1234.00  CSR  MOVE #NUMBR QXXIT
1235.00  C*  --- -----
1236.00  CSR  MOVE #NUMBR QXXIT
1237.00  C*  Set default value - Item ID
1238.00  CSR  VDXIT IFNE *BLANK
1239.00  CSR  D@XIT ANDNE*BLANK
1240.00  CSR  MOVEA@XIT @NM
1241.00  CSR  EXSR C0012
```

**Advanced Programming Concepts and Skills**

**Work fields used in the RPG program begin with $**

**Work fields used in a copy module begin with #**

**Common subroutine to convert screen fields to numeric data**

**Convert to numeric**

**Adjust for display decimals**

---

4-48

A8.1 (8/97)
C*       ----- ----- 
1242.00 CSR  MOVE F@XIT  #DSPD 
1244.00 CSR  MOVE G@XIT  #DCTD 
1245.00 CSR  EXSR C00151 
1246.00 C*       ----- C* 
1247.00 CSR  MOVE #NUMBR QXXIT 
1248.00 CSR  END 
1249.00 C*       Edit upper and lower range – Item ID 
1250.00 C* 
1251.00 CSR  L@XIT  IFNE *BLANK 
1252.00 CSR  MOVE *BLANK  X@XIT  15 
1254.00 CSR  MOVE '1'  $ERTST 
1255.00 CSR  MOVELQXXIT X@XIT 
1256.00 CSR  X@XIT  IFGE L@XIT 
1257.00 CSR  ANDLEU@XIT 
1258.00 CSR  MOVE ' '  $ERTST 
1259.00 CSR  END 
1260.00 CSR  $ERTST  IFEQ '1' 
1261.00 CSR  MOVE '1' @MK,07 
1262.00 CSR  SETON  4193 
1270.00 CSR  END 
1271.00 C*       Scrub and edit – Quantity – on Hand 
1276.00 CSR  MOVEAVDXQT @NM 
1277.00 CSR  EXSR C0012 
1278.00 CSR  VXQT  #DSPD 
1279.00 CSR  VXQT  #DATD 
1284.00 CSR  EXSR C00151 
1287.00 CSR  END 
1291.00 CSR  END 
1292.00 C*       Edit upper and lower range – Quantity – on Hand 
1293.00 C* 
1294.00 CSR  VXQT  #DSPD 
1295.00 CSR  VXQT  #DATD 
1297.00 CSR  EXSR C00151 
1298.00 CSR  VXQT  #DATD 
1300.00 CSR  VXQT  #DATD 
1301.00 CSR  END 
1302.00 CSR  $ERTST  IFEQ '1' 
1303.00 CSR  MOVE '1' @MK,07 
1304.00 CSR  SETON  4693 
1305.00 CSR  END 
1306.00 CSR  END 
1307.00 CSR  END 
1309.00 CSR  END 
1311.00 CSR  MOVELVDXTY QXXY 
1312.00 CSR  END 
1313.00 CSR  END 
1314.00 CSR  END 
1315.00 CSR  QXXY  #DSPD 
1316.00 CSR  QXXY  #DATD 
1317.00 CSR  QXXY  #DATD 
1318.00 CSR  QXXY  #DATD 

Default value from Data Dictionary

Upper and lower ranges from Data Dictionary
1319.00 CSR @40,1 IFEQ '1'  
1320.00 CSR MOVE ' ' @40,1 
1321.00 CSR Z-ADD2 #M 
1322.00 CSR #M DOWLE40 
1323.00 CSR @40,#M IFEQ '1'  
1324.00 CSR MOVE CSR @40,#M IFEQ '1'  
1325.00 CSR END 
1326.00 CSR ADD 1 #M 
1327.00 CSR END CSR 
1328.00 CSR MOVEA@40,2 QXXTY 
1329.00 CSR END 
1330.00 CSR END 
1331.00 CSR END 
1332.00 CSR END 
1333.00 CSR END 
1334.00 CSR END 
1335.00 CSR END 
1336.00 CSR END 
1337.00 CSR END 
1338.00 CSR END 
1339.00 CSR END 
1340.00 CSR END 
1341.00 CSR END 
1342.00 CSR END 
1343.00 CSR END 
1344.00 CSR END 
1345.00 CSR END 
1346.00 CSR END 
1347.00 CSR END 
1348.00 CSR END 
1349.00 CSR END 
1350.00 CSR END 
1351.00 CSR END 
1352.00 CSR END 
1353.00 CSR END 
1354.00 CSR END 
1355.00 CSR END 
1356.00 CSR END 
1357.00 CSR END 
1358.00 CSR END 
1359.00 CSR END 
1360.00 CSR END 
1361.00 CSR END 
1362.00 CSR END 
1363.00 CSR END 
1364.00 CSR END 
1365.00 CSR END 
1366.00 CSR END 
1367.00 CSR END 
1368.00 CSR END 
1369.00 CSR END 
1370.00 CSR END 
1371.00 CSR END 
1372.00 CSR END 
1373.00 CSR END 
1374.00 CSR END 
1375.00 CSR END 
1376.00 CSR END 
1377.00 CSR END 
1378.00 CSR END 
1379.00 CSR END 
1380.00 CSR END 
1381.00 CSR END 
1382.00 CSR END 
1383.00 CSR END 
1384.00 CSR END 
1385.00 CSR END 
1386.00 CSR END 
1387.00 CSR END 
1388.00 CSR END 
1389.00 CSR END 
1390.00 CSR END 
1391.00 CSR END 
1392.00 CSR END 
1393.00 CSR END 
1394.00 CSR END 
1395.00 CSR END
1396.00 C* Set default value – Item Unit of Measure
1397.00 C*
1398.00 CSR QXXUM IFEQ *BLANK
1399.00 CSR D@XUM IFNE *BLANK
1400.00 CSR MOVED@XUM 040
1401.00 CSR MOVEA@40 QXXUM
1402.00 CSR 040,1 IFEQ''''
1403.00 CSR MOVE ' ' 040,1
1404.00 CSR Z-ADD2 @M
1405.00 CSR DOWLE40
1406.00 CSR 040,@M IFEQ''''
1407.00 CSR MOVE ' ' 040,@M
1408.00 CSR END
1409.00 CSR ADD 1 #M
1410.00 CSR END
1411.00 CSR MOVEA@40,2 QXXUM
1412.00 CSR END
1413.00 CSR END
1414.00 CSR END
1415.00 C*
1416.00 C* Edit allowed values – Item Unit of Measure
1417.00 C*
1418.00 CSR A@XUM IFNE *BLANK
1419.00 CSR A@XUM IFEQ '*NB'
1420.00 CSR QXXuM ANDEQ*BLANK
1421.00 CSR MOVE '1' 0MK, 03
1422.00 CSR SETON 4793
1423.00 CSR ELSE
1424.00 CSR MOVEAA@XUM 040
1425.00 CSR MOVE *HIVAL 0AV
1426.00 CSR EXSR C997
1427.00 C* ---- ----
1428.00 CSR MOVE ' ' $ERTST 1
1429.00 CSR MOVE *BLANK $WRK10 10
1430.00 CSR MOVEQXXUM $WRK10
1431.00 CSR 0AV,1 IFNE *HIVAL
1432.00 CSR $WRK10 LOKUP@AV 81
1433.00 CSR *IN81 IFEQ '0'
1434.00 CSR MOVE '1' $ERTST
1435.00 CSR END
1436.00 CSR $ERTST IFEQ '1'
1437.00 CSR MOVE '1' 0MK, 07
1438.00 CSR SETON 4793
1439.00 CSR END
1440.00 CSR END
1441.00 CSR END
1442.00 CSR END
1443.00 C*
1444.00 C* Edit upper and lower range – Item Unit of Measure
1445.00 C*
1446.00 CSR L@XUM IFNE *BLANK
1447.00 CSR MOVE '1' $ERTST
1448.00 CSR QXXUM IFGE L@XUM
1449.00 CSR QXXUM ANDEU@XUM
1450.00 CSR MOVE ' ' $ERTST
1451.00 CSR END
1452.00 CSR $ERTST IFEQ '1'
1453.00 CSR MOVE '1' 0MK, 07
1454.00 CSR SETON 4793
1455.00 CSR END
1456.00 CSR END
1457.00 C*
1458.00 C* Edit from User Defined Codes – Item Unit of Measure
1459.00 C*
1460.00 CSR R@XUM IFNE *BLANK
1461.00 CSR CLEAR10005U
1462.00 CSR MOVELS@XUM #USY
1463.00 CSR MOVE R@XUM #URT
1464.00 CSR MOVE QXXUM #URY
1465.00 CSR CALL 'X0005' 81
1466.00 C* ---- ------
1467.00 CSR PARAM IO005U
1468.00 CSR #UERR IFEQ '1'
1469.00 CSR MOVE '1' 0MK, 09
1470.00 CSR SETON 4793
1471.00 CSR END
1472.00 CSR END
1473.00 C*--------------------------------------------
1474.00 C*
1475.00 C* Scrub and edit - Item Category Code 001
1476.00 C*
1477.00 CSR MOVEVDX001 QXX001
1478.00 C* Set default value - Item Category Code 001
1479.00 C*
1480.00 C*
1481.00 CSR QXX001 IFEQ *BLANK
1482.00 CSR @#X001 IFNE *BLANK
1483.00 CSR MOVEAD@X001 @00
1484.00 CSR MOVEAA@40 QXX001
1485.00 CSR @00,1 IFEQ ""
1486.00 CSR MOVE ' ' @00,1
1487.00 CSR Z-ADD2 @M
1488.00 CSR @M DOWLE40
1489.00 CSR @00,#M IFEQ ""
1490.00 CSR MOVE ' ' @00,#M
1491.00 CSR END
1492.00 CSR ADD 1 @M
1493.00 CSR END
1494.00 CSR MOVEA@40,2 QXX001
1495.00 CSR END
1496.00 CSR END
1497.00 CSR END
1498.00 C*
1499.00 C* Edit allowed values - Item Category Code 001
1500.00 C*
1501.00 CSR ASX001 IFNE *BLANK
1502.00 CSR ASX001 IFEQ 'NB'
1503.00 CSR QXX001 ANDEQ*BLANK
1504.00 CSR MOVE '1' @M,03
1505.00 CSR SETON 4893
1506.00 CSR ELSE
1507.00 CSR MOVEAA@X001 @00
1508.00 CSR MOVE *HIVAL @AV
1509.00 CSR EXSR C997
1510.00 C* ---- ----
1511.00 CSR MOVE '1' $ERTST 1
1512.00 CSR MOVE *BLANK $WRK10 10
1513.00 CSR MOVEQXX001 $WRK10
1514.00 CSR @AV,1 IFNE *HIVAL
1515.00 CSR $WRK10 LOKUP@AV 81
1516.00 CSR IN@0 IFEQ '0'
1517.00 CSR MOVE '1' $ERTST
1518.00 CSR END
1519.00 CSR $ERTST IFEQ '1'
1520.00 CSR MOVE '1' @M,07
1521.00 CSR SETON 4893
1522.00 CSR END
1523.00 CSR END
1524.00 CSR END
1525.00 CSR END
1526.00 C*
1527.00 C* Edit upper and lower range - Item Category Code 001
1528.00 C*
1529.00 CSR L#X001 IFNE *BLANK
1530.00 CSR MOVE '1' $ERTST
1531.00 CSR QXX001 IFEQ L#X001
1532.00 CSR QXX001 ANDLEU@X001
1533.00 CSR MOVE ' ' $ERTST
1534.00 CSR END
1535.00 CSR $ERTST IFEQ '1'
1536.00 CSR MOVE '1' @M,07
1537.00 CSR SETON 4893
1538.00 CSR END
1539.00 CSR END
1040.00 C* Edit from User Defined Codes - Item Category Code 001
1541.00 C* 
1542.00 C* 
1543.00 CSR R@X001 IFNE *BLANK
1544.00 CSR CLEARI0005U
1545.00 CSR MOVELS@X001 #USY
1546.00 CSR MOVE PX001 #URT
1547.00 CSR MOVE QXX001 #UKY
1548.00 CSR CALL 'X0005' 81
1549.00 C* ---- -------
A8.1 (8/97)  4–53

1550.00 CSR    PARAM  I0005U
1551.00 CSR  #UERR  IFEQ '1'
1552.00 CSR  MOVE '1' @MK, 09
1553.00 CSR  SETON  4893
1554.00 CSR  END
1555.00 CSR  END
1556.00 C*     Scrub and edit - Item Category Code 002
1557.00 C*     Set default value - Item Category Code 002
1558.00 CSR  MOVELVDX002  QXX002
1559.00 CSR  VELVDX002  QXX002
1560.00 CSR  QXX002 IFEQ *BLANK
1561.00 CSR  D@X002 IFNE *BLANK
1562.00 CSR  MOVEA@X002  @40
1563.00 CSR  MOVEA@40  QXX002
1564.00 CSR  MOVEA@X002  @40
1565.00 CSR  QXX002 IFEQ *BLANK
1566.00 CSR  MOVEA@X002  QXX002
1567.00 CSR  QXX002 IFEQ *BLANK
1568.00 CSR  QXX002 IFEQ *NB'
1569.00 CSR  QYY002 ANDLEU@X002
1570.00 CSR  Z-ADD2  #M
1571.00 CSR  DOWLE40
1572.00 CSR  $ERTST IFEQ *BLANK
1573.00 CSR  $ERTST IFEQ '1'
1574.00 CSR  END
1575.00 CSR  END
1576.00 CSR  END
1577.00 CSR  END
1578.00 CSR  END
1579.00 CSR  END
1580.00 CSR  END
1581.00 CSR  END
1582.00 CSR  END
1583.00 CSR  END
1584.00 CSR  END
1585.00 CSR  END
1586.00 CSR  END
1587.00 CSR  END
1588.00 CSR  END
1589.00 CSR  END
1590.00 CSR  END
1591.00 CSR  END
1592.00 CSR  END
1593.00 CSR  END
1594.00 CSR  END
1595.00 CSR  END
1596.00 CSR  END
1597.00 CSR  END
1598.00 CSR  END
1599.00 CSR  END
1600.00 CSR  END
1601.00 CSR  END
1602.00 CSR  END
1603.00 CSR  END
1604.00 CSR  END
1605.00 CSR  END
1606.00 CSR  END
1607.00 CSR  END
1608.00 CSR  END
1609.00 CSR  END
1610.00 CSR  END
1611.00 CSR  END
1612.00 CSR  END
1613.00 CSR  END
1614.00 CSR  END
1615.00 CSR  END
1616.00 CSR  END
1617.00 CSR  END
1618.00 CSR  END
1619.00 CSR  END
1620.00 CSR  END
1621.00 CSR  END
1622.00 CSR  END
1623.00 CSR  END
1624.00 CSR  END
1625.00 CSR  END
1626.00 CSR  END

1560.00 CSR  MOVELVDX002  QXX002
1561.00 CSR  D@X002 IFNE *BLANK
1562.00 CSR  MOVEA@X002  @40
1563.00 CSR  MOVEA@40  QXX002
1564.00 CSR  QXX002 IFEQ *BLANK
1565.00 CSR  D@X002 IFNE *BLANK
1566.00 CSR  MOVEA@X002  QXX002
1567.00 CSR  QXX002 IFEQ *BLANK
1568.00 CSR  QXX002 IFEQ *NB'
1569.00 CSR  QYY002 ANDLEU@X002
1570.00 CSR  Z-ADD2  #M
1571.00 CSR  DOWLE40
1572.00 CSR  $ERTST IFEQ *BLANK
1573.00 CSR  $ERTST IFEQ '1'
1574.00 CSR  END
1575.00 CSR  END
1576.00 CSR  END
1577.00 CSR  END
1578.00 CSR  END
1579.00 CSR  END
1580.00 CSR  END
1581.00 CSR  END
1582.00 CSR  END
1583.00 CSR  END
1584.00 CSR  END
1585.00 CSR  END
1586.00 CSR  END
1587.00 CSR  END
1588.00 CSR  END
1589.00 CSR  END
1590.00 CSR  END
1591.00 CSR  END
1592.00 CSR  END
1593.00 CSR  END
1594.00 CSR  END
1595.00 CSR  END
1596.00 CSR  END
1597.00 CSR  END
1598.00 CSR  END
1599.00 CSR  END
1600.00 CSR  END
1601.00 CSR  END
1602.00 CSR  END
1603.00 CSR  END
1604.00 CSR  END
1605.00 CSR  END
1606.00 CSR  END
1607.00 CSR  END
1608.00 CSR  END
1609.00 CSR  END
1610.00 CSR  END
1611.00 CSR  END
1612.00 CSR  END
1613.00 CSR  END
1614.00 CSR  END
1615.00 CSR  END
1616.00 CSR  END
1617.00 CSR  END
1618.00 CSR  END
1619.00 CSR  END
1620.00 CSR  END
1621.00 CSR  END
1622.00 CSR  END
1623.00 CSR  END
1624.00 CSR  END
1625.00 CSR  END
1626.00 CSR  END

A8.1 (8/97)  4–53
1627.00 CSR CLEARI0005U
1628.00 CSR MOVELSQX002 #USY
1629.00 CSR MOVE R@X002 #URT
1630.00 CSR MOVE QXX002 #UKY
1631.00 CSR CALL 'X0005' C* 81
1632.00 C* ---- ----
1633.00 CSR #UERR IFEQ '1'
1634.00 CSR MOVE '1' @MK,09
1635.00 CSR MOVE R@X002 END
1636.00 CSR SETON 4993
1637.00 CSR END
1638.00 CSR END
1639.00 C*------------------------
1640.00 C* Scrub and edit – Item Category Code 003
1641.00 C*
1642.00 C*
1643.00 CSR MOVELVQX03 QxX003
1644.00 C* Set default value – Item Category Code 003
1645.00 C*
1646.00 C*
1647.00 CSR QXX003 IFEQ 'BLANK'
1648.00 CSR D@X003 IFNE 'BLANK'
1649.00 CSR MOVEA@X003 040
1650.00 CSR MOVEA@40 QXX003
1651.00 CSR @40,1 IFEQ 40,1
1652.00 CSR MOVE ' ' @MK,09
1653.00 CSR &ADD2 #M
1654.00 CSR #40,DOWLE40
1655.00 CSR &40,#M IFEQ '####'
1656.00 CSR MOVE ' ' @MK,09
1657.00 CSR END
1658.00 CSR ADD 1 #M
1659.00 CSR END
1660.00 CSR MOVEA@40,2 QXX003
1661.00 CSR END
1662.00 CSR END
1663.00 CSR END
1664.00 C*
1665.00 C* Edit allowed values – Item Category Code 003
1666.00 C*
1667.00 CSR A@X003 IFNE 'BLANK'
1668.00 CSR A@X003 'NB'
1669.00 CSR QXX003 ANDEQ 'BLANK'
1670.00 CSR MOVE '1' @MK,03
1671.00 CSR SETON 5093
1672.00 CSR ELSE
1673.00 CSR MOVEA@X003 040
1674.00 CSR MOVE 'HIVAL' @AV
1675.00 CSR EXSR C997
1676.00 CSR &AV,1 IFEQ HIVAL
1677.00 CSR MOVE ' ' $ERTST
1678.00 CSR MOVE 'BLANK' $WRK10 10
1679.00 CSR MOVEQX003 $WRK10
1680.00 CSR @AV,1 IFNE 'HIVAL'
1681.00 CSR $WRK10 LOKUP@AV 81
1682.00 CSR *IN81 IFEQ '0'
1683.00 CSR MOVE '1' $ERTST
1684.00 CSR END
1685.00 CSR $ERTST IFEQ '1'
1686.00 CSR MOVE '1' @MK,07
1687.00 CSR SETON 5093
1688.00 CSR END
1689.00 CSR END
1690.00 CSR END
1691.00 CSR END
1692.00 C* Edit upper and lower range – Item Category Code 003
1693.00 C*
1694.00 C*
1695.00 CSR L@X003 IFNE 'BLANK'
1696.00 CSR MOVE '1' $ERTST
1697.00 CSR QXX003 IFGE L@X003
1698.00 CSR QXX003 ANDLEU@X003 $ERTST
1699.00 CSR MOVE ' ' @MK,07
1700.00 CSR END
1701.00 CSR $ERTST IFEQ '1'
1702.00 CSR MOVE '1' @MK,07
1703.00 CSR SETON 5093
1704.00 CSR END
1705.00 CSR END
1706.00 C*
1707.00 C* Edit from User Defined Codes - Item Category Code 003
1708.00 C*
1709.00 CSR R8X003 IFNE *BLANK
1710.00 CSR CLEAR10005U
1711.00 CSR MOVELS8X003 #$USY
1712.00 CSR MOVE PSX003 #$UKT
1713.00 CSR MOVE QXX003 #$UKY
1714.00 CSR CALL 'X0005' 81
1715.00 C* ----- -----
1716.00 CSR PARM I0005U
1717.00 CSR #UERR IFEQ '1'
1718.00 CSR MOVE '1' $MK,09
1719.00 CSR SETON 5093
1720.00 CSR END
1721.00 CSR END
1722.00 C*------------------------------------------
1723.00 Ct
1724.00 C* Scrub and edit - Item Category Code 004
1725.00 C*
1726.00 CSR MOVEVL0X004 QXX004
1727.00 C*
1728.00 Ct Set default value - Item Category Code 004
1729.00 C*
1730.00 CSR QXX004 IFEQ *BLANK
1731.00 CSR D8X004 IFNE *BLANK
1732.00 CSR MOVEA8X004 $40
1733.00 CSR MOVEA40 QXX004
1734.00 CSR $40,1 IFEQ **
1735.00 CSR MOVE ' ' $40,1
1736.00 CSR Z-ADD2 $M
1737.00 CSR $M DOWLE40
1738.00 CSR $M IFEQ **
1739.00 CSR MOVE ' ' $M
1740.00 CSR END
1741.00 CSR END
1742.00 CSR END
1743.00 CSR MOVEA40,2 QXX004
1744.00 CSR END
1745.00 CSR END
1746.00 CSR END
1747.00 C*
1748.00 C* Edit allowed values - Item Category Code 004
1749.00 C*
1750.00 CSR A8X004 IFNE *BLANK
1751.00 CSR A8X004 IFEQ **NB'
1752.00 CSR QXX004 ANDEQ*BLANK
1753.00 CSR MOVE '1' $MK,03
1754.00 CSR SETON 5193
1755.00 CSR ELSE
1756.00 CSR MOVEA8X004 $40
1757.00 CSR MOVE *HIVAL $AV
1758.00 CSR EXSR C997
1759.00 C* ----- -----
1760.00 CSR MOVE ' ' SERTST 1
1761.00 CSR MOVE *BLANK $WRK10 10
1762.00 CSR MOVEXQXX004 $WRK10
1763.00 CSR $AV,1 IFEQ *HIVAL
1764.00 CSR $WRK10 LOKUP $AV 81
1765.00 CSR *IN81 IFEQ '0'
1766.00 CSR MOVE '1' SERTST
1767.00 CSR END
1768.00 CSR SERTST IFEQ '1'
1769.00 CSR MOVE '1' $MK,07
1770.00 CSR SETON 5193
1771.00 CSR END
1772.00 CSR END
1773.00 CSR END
1774.00 CSR END
1775.00 C* Edit upper and lower range - Item Category Code 004
1776.00 C*
1777.00 C*
1778.00 CSR L8X004 IFNE *BLANK
1779.00 CSR MOVE '1' SERTST
1780.00 CSR QXX004 IFEQ L8X004
<table>
<thead>
<tr>
<th>Line</th>
<th>Instruction</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>1781</td>
<td>CSR QXX004 ANDLEU@X004</td>
<td></td>
</tr>
<tr>
<td>1782</td>
<td>CSR MOVE ‘ ’ SERTST</td>
<td></td>
</tr>
<tr>
<td>1783</td>
<td>CSR END</td>
<td></td>
</tr>
<tr>
<td>1784</td>
<td>CSR $ERTST IFEQ ‘1’</td>
<td></td>
</tr>
<tr>
<td>1785</td>
<td>CSR MOVE ‘1’ @MK,07</td>
<td></td>
</tr>
<tr>
<td>1786</td>
<td>CSR SETON 5193</td>
<td></td>
</tr>
<tr>
<td>1787</td>
<td>CSR END</td>
<td></td>
</tr>
<tr>
<td>1788</td>
<td>CSR END</td>
<td></td>
</tr>
<tr>
<td>1789</td>
<td>CSR</td>
<td>Edit from User Defined Codes - Item Category Code 004</td>
</tr>
<tr>
<td>1790</td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>1791</td>
<td>CSR R@X004 IFNE *BLANK</td>
<td></td>
</tr>
<tr>
<td>1792</td>
<td>CSR CLEAR@0005U</td>
<td></td>
</tr>
<tr>
<td>1793</td>
<td>CSR MOVE@R@X004 #USY</td>
<td></td>
</tr>
<tr>
<td>1794</td>
<td>CSR MOVE R@X004 #URT</td>
<td></td>
</tr>
<tr>
<td>1795</td>
<td>CSR MOVE QXX004 #UKY</td>
<td></td>
</tr>
<tr>
<td>1796</td>
<td>CSR CALL ‘X0005’ #1</td>
<td></td>
</tr>
<tr>
<td>1797</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1798</td>
<td>* END</td>
<td></td>
</tr>
<tr>
<td>1799</td>
<td>CSR</td>
<td>Scrub and edit - Item Category Code 005</td>
</tr>
<tr>
<td>1800</td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>1801</td>
<td>CSR</td>
<td>Set default value - Item Category Code 005</td>
</tr>
<tr>
<td>1802</td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>1803</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1804</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1805</td>
<td>* END</td>
<td></td>
</tr>
<tr>
<td>1806</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1807</td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>1808</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1809</td>
<td>CSR QXX005 IFEQ *BLANK</td>
<td></td>
</tr>
<tr>
<td>1810</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1811</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1812</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1813</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1814</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1815</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1816</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1817</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1818</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1819</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1820</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1821</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1822</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1823</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1824</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1825</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1826</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1827</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1828</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1829</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1830</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1831</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1832</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1833</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1834</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1835</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1836</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1837</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1838</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1839</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1840</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1841</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1842</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1843</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1844</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1845</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1846</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1847</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1848</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1849</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1850</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1851</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1852</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1853</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1854</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1855</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1856</td>
<td>CSR</td>
<td></td>
</tr>
<tr>
<td>1857</td>
<td>CSR</td>
<td></td>
</tr>
</tbody>
</table>
1858.00 C* Edit upper and lower range - Item Category Code 005
1860.00 C*
1861.00 CSR L@X005 IFNE 'BLANK
1862.00 CSR MOVE 'I' $ERTST
1863.00 CSR QXX005 IFGE L@X005
1864.00 CSR QXX005 ANDLEU@X005
1865.00 CSR MOVE ' ' SERTST
1866.00 CSR END
1867.00 CSR SERTST IFEQ '1'
1868.00 CSR MOVE '1' @MK,07
1869.00 CSR SETON 5293
1870.00 CSR END
1871.00 CSR END
1872.00 C* Edit from User Defined Codes - Item Category Code 005
1874.00 C*
1875.00 CSR R@X005 IFNE 'BLANK
1876.00 CSR CLEARI0005U
1877.00 CSR MOVEI@X005 #USY
1878.00 CSR MOVEI@X005 #URT
1879.00 CSR MOVEI@X005 #UKY
1880.00 CSR CALL 'X0005' 81
1881.00 C*
1882.00 CSR PARM I0005U
1883.00 CSR #UERR IFEQ '1'
1884.00 CSR MOVE '1' @MK,09
1885.00 CSR SETON 5293
1886.00 CSR END
1887.00 CSR END
1888.00 C* Copy Common Subroutine – Currency – Translate Video Fields to Data Base
1893.00 C*
1894.00 C/COPY JDECPY,C00151
1895.00 C**************************************************************
1896.00 C*
1897.00 C* Copy Common Subroutine – Build Allowed Values Work Array
1898.00 C**************************************************************
1899.00 C/COPY JDECPY,C997
1900.00 C**************************************************************
1901.00 C*
1902.00 C* SUBROUTINE S010 – Update Data Base
1903.00 C* --------------------------
1904.00 C* Processing: 1. Update data base file based upon valid
1905.00 C* action codes.
1906.00 C*
1907.00 C* If add action, add record.
1908.00 C* If change action, update record.
1909.00 C* If delete action, delete record.
1910.00 C* Indicator value for action code is
1911.00 C* assigned in copy module C0001.
1912.00 C*
1913.00 CSR *IN21 IFEQ '1'
1914.00 CSR WRITEI92801 99
1915.00 CSR END
1916.00 C*
1917.00 C* If change action, update record.
1918.00 C*
1919.00 CSR *IN22 IFEQ '1'
1920.00 CSR UDATI92801 99
1921.00 CSR END
1922.00 C*
1923.00 C* If delete action, delete record.
1924.00 C*
1925.00 CSR *IN23 IFEQ '1'
1926.00 CSR DELETI92801 99
1927.00 CSR END
1928.00 C*
1929.00 C* Clear data field for next transaction
1930.00 C* MOVE ‘#FCLR’ @AID
1931.00 C* EXSR S001
1933.00 C* ----
1934.00 CSR END010 ENDSR
1935.00 C*****************************************************
1936.00 C*
1937.00 C* SUBROUTINE S998 - Load dictionary parameters.
1938.00 C*------------------------------------------------------
1939.00 C*
1940.00 CSR S998 BEGSR
1942.00 C* ---- ----
1943.00 C* Dictionary parameters for - Cost Center
1944.00 C*
1946.00 CSR MOVE ‘BLANK’ FRDTAI
1948.00 CSR CALL ‘X9800E’ 81
1950.00 CSR PARM 19800E
1951.00 CSR FRERR IFEQ ‘0’
1953.00 CSR MOVE FRDTAT T@XCC 1
1954.00 CSR MOVE FREC E@XCC 1
1955.00 CSR MOVE FRDTAS C@XCC 50
1956.00 CSR MOVE FRDTAD G@XCC 20
1957.00 CSR MOVE FRCDEC F@XCC 1
1958.00 CSR MOVELFRSY S@XCC 4
1959.00 CSR MOVE FRRT R@XCC 2
1960.00 CSR MOVE FRDVAL D@XCC 40
1961.00 CSR MOVE FRVAL A@EDS 40
1962.00 CSR MOVE FRNNIX N@XCC 20
1963.00 CSR Z–ADD1 #@XCC 110
1964.00 CSR MOVE F@XCC #A
1965.00 CSR DO   #A
1966.00 CSR MULT 10 #@XCC
1967.00 CSR END
1969.00 CSR END
1970.00 CSR END
1971.00 CSR END
1972.00 CSR END
1973.00 C*------------------------------------------------------
1974.00 C* Dictionary parameters for - Description
1975.00 C*
1976.00 C*
1977.00 CSR MOVE ‘BLANK’ FRDTAI
1978.00 CSR MOVEL’XDS’ FRDTAI
1979.00 CSR CALL ‘X9800E’ 81
1980.00 C* ---- ----
1981.00 CSR PARM 19800E
1982.00 CSR FRERR IFEQ ‘0’
1984.00 CSR MOVE FRDTAT T@XDS 1
1985.00 CSR MOVE FREC E@XDS 1
1986.00 CSR MOVE FRDTAS C@XDS 50
1987.00 CSR MOVE FRDTAD G@XDS 20
1988.00 CSR MOVE FRCDEC F@XDS 1
1989.00 CSR MOVELFRSY S@XDS 4
1990.00 CSR MOVE FRRT R@XDS 2
1991.00 CSR MOVE FRDVAL D@XDS 40
1992.00 CSR MOVE FRVAL A@EDS 40
1993.00 CSR MOVE FRNNIX N@XCC 20
1994.00 CSR Z–ADD1 #@XCC 110
1995.00 CSR MOVE F@XCC #A
2000.00 CSR DO   #A
2001.00 CSR MULT 10 #@XCC
2002.00 CSR END
2003.00 CSR END
2004.00 C*------------------------------------------------------
2005.00 C*
2006.00  C*  Dictionary parameters for – Date Last Ship
2007.00  C*
2008.00  CSR  MOVE 'BLANK' FRDTAI
2009.00  CSR  MOVEL' XDT' FRDTAI
2010.00  CSR  CALL 'X9800E'  81
2011.00  C*  ----  ----
2012.00  CSR  PARM  I9800E
2013.00  CSR  FRERR  IFEQ '0'
2015.00  CSR  MOVE FRDTAT T@XDT  1
2016.00  CSR  MOVE FREC E@XDT  1
2017.00  CSR  MOVE FRDTAS C@EDT  50
2018.00  CSR  MOVE FRDTAD G@XDT  20
2019.00  CSR  MOVE FRDCDEC F@XDT  1
2020.00  CSR  MOVEFRSY S@XDT  4
2021.00  CSR  MOVE FRRT R@XDT  2
2022.00  CSR  MOVE FRDVAL D@XDT  40
2023.00  CSR  MOVE FRVAL A@XDT  40
2024.00  CSR  MOVE FRLVAL L@XDT  40
2025.00  CSR  MOVE FRUVAL U@XDT  40
2026.00  CSR  MOVE FREDWR M@XDT  30
2027.00  CSR  MOVE FRRA J@XDT  1
2029.00  CSR  MOVE FRNNIX N@XDT  20
2029.00  CSR  Z–ADD1 @XDT  110
2030.00  CSR  MOVE F@XDT #A
2031.00  CSR  DO  #A
2032.00  CSR  MULT 10 @XDT
2033.00  CSR  END
2034.00  CSR  END
2035.00  C*  ----
2036.00  C*  Dictionary parameters for – Item ID
2037.00  C*
2039.00  CSR  MOVE 'BLANK' FRDTAI
2040.00  CSR  MOVEL'XIT' FRDTAI
2041.00  CSR  CALL 'X9800E'  81
2042.00  C*  ----  ----
2043.00  CSR  PARM  I9800E
2044.00  CSR  FRERR  IFEQ '0'
2046.00  CSR  MOVE FRDTAT T@XIT  1
2047.00  CSR  MOVE FREC E@XIT  1
2048.00  CSR  MOVE FRDTAS C@XIT  50
2049.00  CSR  MOVE FRDTAD G@XIT  20
2050.00  CSR  MOVE FRDCDEC F@XIT  1
2051.00  CSR  MOVEFRSY S@XIT  4
2052.00  CSR  MOVE FRRT R@XIT  2
2053.00  CSR  MOVE FRDVAL D@XIT  40
2054.00  CSR  MOVE FRVAL A@XIT  40
2055.00  CSR  MOVE FRLVAL L@XIT  40
2056.00  CSR  MOVE FRUVAL U@XIT  40
2057.00  CSR  MOVE FREDWR M@XIT  30
2058.00  CSR  MOVE FRRA J@XIT  1
2059.00  CSR  MOVE FRNNIX N@XIT  20
2060.00  CSR  Z–ADD1 @XIT  110
2061.00  CSR  MOVE F@XIT #A
2062.00  CSR  DO  #A
2063.00  CSR  MULT 10 @XIT
2064.00  CSR  END
2065.00  CSR  END
2066.00  C*  ----
2067.00  C*  Dictionary parameters for – Quantity – On Hand
2068.00  C*
2069.00  CSR  MOVE 'BLANK' FRDTAI
2071.00  CSR  MOVEL'XQT' FRDTAI
2072.00  CSR  CALL 'X9800E'  81
2073.00  C*  ----  ----
2074.00  CSR  PARM  I9800E
2075.00  CSR  FRERR  IFEQ '0'
2077.00  CSR  MOVE FRDTAT T@XQT  1
2078.00  CSR  MOVE FREC E@XQT  1
2079.00  CSR  MOVE FRDTAS C@XQT  50
2080.00  CSR  MOVE FRDTAD G@XQT  20
2081.00  CSR  MOVE FRDCDEC F@XQT  1
2082.00  CSR  MOVEFRSY S@XQT  4
2083.00  CSR  MOVE FDRT  R@XQT  2
2084.00  CSR  MOVE FRDVAL D@XQT  40
2085.00  CSR  MOVE FRVAL A@XQT  40
2086.00  CSR  MOVE FRLVAL L@XQT  40
2087.00  CSR  MOVE FRUVAL U@XQT  40
2088.00  CSR  MOVE FREDWR N@XQT  30
2089.00  CSR  MOVE FRLR J@XQT  1
2090.00  CSR  MOVE FRNIX N@XQT  20
2091.00  CSR  Z–ADD1 #@XQT  110
2092.00  CSR  MOVE FRXQT #A
2093.00  CSR  DO  #A
2094.00  CSR  MULT 10 #@XQT
2095.00  CSR  END
2096.00  CSR  END

2097.00  C*-----------------------------------------------
2098.00  C*
2099.00  C*     Dictionary parameters for – Item Type
2100.00  C*
2101.00  CSR  MOVE 'BLANK' FRDTAI
2102.00  CSR  MOVE 'XTY' FRDTAI
2103.00  CSR  CALL 'X9800E'  81
2104.00  C*     -----       
2105.00  CSR  PARM  I9800E
2106.00  CSR  FRERR  IFRQ '0'
2108.00  CSR  MOVE FDRTAT T@XTY  1
2109.00  CSR  MOVE FREC E@XTY  1
2110.00  CSR  MOVE FRDTAS C@XTY  50
2111.00  CSR  MOVE FRDDEC G@XTY  20
2112.00  CSR  MOVE FRDDEC F@XTY  1
2113.00  CSR  MOVEFRSY S@XTY  4
2114.00  CSR  MOVE FDRT  P@XTY  2
2115.00  CSR  MOVE FRDVAL D@XTY  40
2116.00  CSR  MOVE FRVAL A@XTY  40
2117.00  CSR  MOVE FRLVAL L@XTY  40
2118.00  CSR  MOVE FRUVAL U@XTY  40
2119.00  CSR  MOVE FREDWR N@XTY  30
2120.00  CSR  MOVE FRLR J@XTY  1
2121.00  CSR  MOVE FRNIX N@XTY  20
2122.00  CSR  Z–ADD1 #@XTY  110
2123.00  CSR  MOVE F@XTY #A
2124.00  CSR  DO  #A
2125.00  CSR  MULT 10 #@XTY
2126.00  CSR  END
2127.00  CSR  END

2128.00  C*-----------------------------------------------
2129.00  C*
2130.00  C*     Dictionary parameters for – Item Unit of Measure
2131.00  C*
2132.00  CSR  MOVE 'BLANK' FRDTAI
2133.00  CSR  MOVE 'XUM' FRDTAI
2134.00  CSR  CALL 'X9800E'  81
2135.00  C*     -----       
2136.00  CSR  PARM  I9800E
2137.00  CSR  FRERR  IFRQ '0'
2139.00  CSR  MOVE FDRTAT T@XUM  1
2140.00  CSR  MOVE FREC E@XUM  1
2141.00  CSR  MOVE FRDTAS C@XUM  50
2142.00  CSR  MOVE FRDDEC G@XUM  20
2143.00  CSR  MOVE FRDDEC F@XUM  1
2144.00  CSR  MOVEFRSY S@XUM  4
2145.00  CSR  MOVE FDRT  P@XUM  2
2146.00  CSR  MOVE FRDVAL D@XUM  40
2147.00  CSR  MOVE FRVAL A@XUM  40
2148.00  CSR  MOVE FRLVAL L@XUM  40
2149.00  CSR  MOVE FRUVAL U@XUM  40
2150.00  CSR  MOVE FREDWR N@XUM  30
2151.00  CSR  MOVE FRLR J@XUM  1
2152.00  CSR  MOVE FRNIX N@XUM  20
2153.00  CSR  Z–ADD1 #@XUM  110
2154.00  CSR  MOVE F@XUM #A
2155.00  CSR  DO  #A
2156.00  CSR  MULT 10 #@XUM
2157.00  CSR  END
2158.00  CSR  END
2159.00  C*
Dictionary parameters for – Item Category Code 001

2160.00 C*
2161.00 C*  Dictionary parameters for – Item Category Code 001
2162.00 C*
2163.00 CSR  MOVE 'BLANK'  FRDTAI
2164.00 CSR  MOVEL'X001'  FRDTAI
2165.00 CSR  CALL 'X9800E'  81
2166.00 C*  ----- -------
2167.00 CSR  PARM  19800E
2168.00 CSR  FRERR  IFREQ '0'
2169.00 CSR  MOVE FRDTAT T@X001  1
2170.00 CSR  MOVE FREC E@X001  1
2171.00 CSR  MOVE FRDTAS C@X001  50
2172.00 CSR  MOVE FRDTAD G@X001  20
2173.00 CSR  MOVE FRCDEC F5X001  1
2174.00 CSR  MOVE FRVAL A@X001  40
2175.00 CSR  MOVE FRRT R@X001  2
2176.00 CSR  MOVE FREDWR W@X001  30
2177.00 CSR  MOVE FRNNIX N@X001  20
2178.00 CSR  MOVE FRLVAL L@X001  40
2179.00 CSR  MOVE FRLR J@X001  1
2180.00 CSR  MOVE FRLRSY S@X001  4
2181.00 CSR  MOVE FRLRSY S@X001  4
2182.00 CSR  Z–ADD1 #@X001  110
2183.00 CSR  MOVE F5X001 #A
2184.00 CSR  DO  #A
2185.00 CSR  MULTI @X001
2186.00 CSR  END
2187.00 CSR  END

Dictionary parameters for – Item Category Code 002

2190.00 C*  Dictionary parameters for – Item Category Code 002
2191.00 C*
2192.00 CSR  MOVE 'BLANK'  FRDTAI
2193.00 CSR  MOVEL'X002'  FRDTAI
2194.00 CSR  CALL 'X9800E'  81
2195.00 CSR  ----- -------
2196.00 CSR  PARM  19800E
2197.00 CSR  FRERR  IFREQ '0'
2198.00 CSR  MOVE FRDTAT T@X002  1
2199.00 CSR  MOVE FREC E@X002  1
2200.00 CSR  MOVE FRDTAS C@X002  50
2201.00 CSR  MOVE FRDTAD G@X002  20
2202.00 CSR  MOVE FRCDEC F5X002  1
2203.00 CSR  MOVE FRVAL A@X002  40
2204.00 CSR  MOVE FRRT R@X002  2
2205.00 CSR  MOVE FREDWR W@X002  30
2206.00 CSR  MOVE FRNNIX N@X002  20
2207.00 CSR  Z–ADD1 #@X002  110
2208.00 CSR  MOVE F5X002 #A
2209.00 CSR  DO  #A
2210.00 CSR  MULTI @X002
2211.00 CSR  END
2212.00 CSR  END

Dictionary parameters for – Item Category Code 003

2220.00 CSR  END
2221.00 CSR  END
2222.00 C*
2223.00 C*  Dictionary parameters for – Item Category Code 003
2224.00 C*
2225.00 CSR  MOVE 'BLANK'  FRDTAI
2226.00 CSR  MOVEL'X003'  FRDTAI
2227.00 CSR  CALL 'X9800E'  81
2228.00 C*  ----- -------
2229.00 CSR  PARM  19800E
2230.00 CSR  FRERR  IFREQ '0'
2231.00 CSR  MOVE FRDTAT T@X003  1
2232.00 CSR  MOVE FREC E@X003  1
2233.00 CSR  MOVE FRDTAS C@X003  50
2234.00 CSR  MOVE FRDTAD G@X003  20
2235.00 CSR  MOVE FREDWR W@X003  30
2236.00 CSR  MOVE FRCDEC F@X003  1
2237.00 CSR MOVELFRSY @X003 4
2238.00 CSR MOVE FRST @X003 2
2239.00 CSR MOVE FRDVAL D@X003 40
2240.00 CSR MOVE FRLVAL A@X003 40
2241.00 CSR MOVE FRLR J@X003 1
2242.00 CSR MOVE FREC E@X003 450
2243.00 CSR MOVE FREDWR W@X003 30
2244.00 CSR MOVE FRNNIX N@X003 20
2245.00 CSR MOVE FRDVAL D@X003 40
2246.00 CSR MOVE FRRT R@X003 2
2247.00 CSR MOVE FRDVAL D@X003 40
2248.00 CSR MOVE FREDWR W@X003 30
2249.00 CSR MOVE FRNNIX N@X003 20
2250.00 CSR MOVE FRDVAL D@X003 40
2251.00 CSR MOVE FRRT R@X003 2
2252.00 CSR MOVE FRDVAL D@X003 40
2253.00 CSR MOVE FREDWR W@X003 30
2254.00 CSR MOVE FRNNIX N@X003 20
2255.00 CSR MOVE FRDVAL D@X003 40
2256.00 CSR MOVE FREDWR W@X003 30
2257.00 CSR MOVE FRNNIX N@X003 20
2258.00 CSR MOVE FREDWR W@X003 30
2259.00 CSR MOVE FRNNIX N@X003 20
2260.00 CSR MOVE FREDWR W@X003 30
2261.00 CSR MOVE FRNNIX N@X003 20
2262.00 CSR MOVE FRRT R@X003 2
2263.00 CSR MOVE FREDWR W@X003 30
2264.00 CSR MOVE FRNNIX N@X003 20
2265.00 CSR MOVE FRRT R@X003 2
2266.00 CSR MOVE FREDWR W@X003 30
2267.00 CSR MOVE FRNNIX N@X003 20
2268.00 CSR MOVE FRRT R@X003 2
2269.00 CSR MOVE FREDWR W@X003 30
2270.00 CSR MOVE FRNNIX N@X003 20
2271.00 CSR MOVE FRRT R@X003 2
2272.00 CSR MOVE FREDWR W@X003 30
2273.00 CSR MOVE FRNNIX N@X003 20
2274.00 CSR MOVE FRRT R@X003 2
2275.00 CSR MOVE FREDWR W@X003 30
2276.00 CSR MOVE FRNNIX N@X003 20
2277.00 CSR MOVE FREDWR W@X003 30
2278.00 CSR MOVE FRNNIX N@X003 20
2279.00 CSR MOVE FREDWR W@X003 30
2280.00 CSR MOVE FREDWR W@X003 30
2281.00 CSR MOVE FREDWR W@X003 30
2282.00 CSR MOVE FREDWR W@X003 30
2283.00 CSR MOVE FREDWR W@X003 30
2284.00 CSR MOVE FREDWR W@X003 30
2285.00 CSR MOVE FREDWR W@X003 30
2286.00 CSR MOVE FREDWR W@X003 30
2287.00 CSR MOVE FREDWR W@X003 30
2288.00 CSR MOVE FREDWR W@X003 30
2289.00 CSR MOVE FREDWR W@X003 30
2290.00 CSR MOVE FREDWR W@X003 30
2291.00 CSR MOVE FREDWR W@X003 30
2292.00 CSR MOVE FREDWR W@X003 30
2293.00 CSR MOVE FREDWR W@X003 30
2294.00 CSR MOVE FREDWR W@X003 30
2295.00 CSR MOVE FREDWR W@X003 30
2296.00 CSR MOVE FREDWR W@X003 30
2297.00 CSR MOVE FREDWR W@X003 30
2298.00 CSR MOVE FREDWR W@X003 30
2299.00 CSR MOVE FREDWR W@X003 30
2300.00 CSR MOVE FREDWR W@X003 30
2301.00 CSR MOVE FREDWR W@X003 30
2302.00 CSR MOVE FREDWR W@X003 30
2303.00 CSR MOVE FREDWR W@X003 30
2304.00 CSR MOVE FREDWR W@X003 30
2305.00 CSR MOVE FREDWR W@X003 30
2306.00 CSR MOVE FREDWR W@X003 30
2307.00 CSR MOVE FREDWR W@X003 30
2308.00 CSR MOVE FREDWR W@X003 30
2309.00 CSR MOVE FREDWR W@X003 30
2310.00 CSR MOVE FREDWR W@X003 30
2311.00 CSR MOVE FREDWR W@X003 30
2312.00 CSR MOVE FREDWR W@X003 30
2313.00 CSR MOVE FREDWR W@X003 30
A8.1 (8/97)

4–63

2314.00 C*-----------------------------------------------
2315.00 C* Set subroutine execution flag.
2316.00 C*-----------------------------------------------
2317.00 C* CSR MOVE '1' $998 1
2318.00 C*-----------------------------------------------
2319.00 C* SUBROUTINE S999 - Housekeeping
2320.00 C*-----------------------------------------------
2321.00 C* Processing: 1. Load video screen text.
2322.00 C* 2. Retrieve screen title data area, test
2323.00 C* for unauthorized access, center video
2324.00 C* title and move to video screen.
2325.00 C* 3. Initialize key list.
2326.00 C* 4. Load roll keys.
2327.00 C* 5. Passed parameters.
2328.00 C* 6. Load error message array.
2329.00 C*-----------------------------------------------
2330.00 CSR S999 BEGSR
2331.00 C* Required program parameters.
2332.00 C*-----------------------------------------------
2333.00 CSR *ENTRY PLIST
2334.00 C* Passed Parameter - Item ID
2335.00 CSR PARM ##XIT 8
2336.00 C*-----------------------------------------------
2337.00 CSR MOVEL@@FILE PSKEY 10
2338.00 CSR Z–ADD025 PSVTX# 30
2339.00 C/COPY JDECPY,C00SC
2340.00 C*-----------------------------------------------
2341.00 CSR QXKY01 KLIST
2342.00 CSR KFLD QXXIT
2343.00 C*-----------------------------------------------
2344.00 CSR *LIKE DEFN QXXIT $RUKEY
2345.00 CSR *LIKE DEFN $RUKEY $RDKEY
2346.00 CSR MOVE *LOVAL $RUKEY
2347.00 CSR MOVE *ALL'9' $RDKEY
2348.00 C*-----------------------------------------------
2349.00 C*-----------------------------------------------
2350.00 CSR S999 BEGSR
2351.00 C* Parameters passed to program
2352.00 C*-----------------------------------------------
2353.00 CSR VDXIT IFNE *BLANK
2354.00 CSR MOVE '1' $AUTO 1
2355.00 CSR END
2356.00 C*-----------------------------------------------
2357.00 C* Load video screen text.
2358.00 C*-----------------------------------------------
2359.00 CSR MOVE4@FILE PSKEY 10
2360.00 CSR I–ADD025 PSVTX# 30
2361.00 C/COPY JDECPY,C00SC
2362.00 C*-----------------------------------------------
2363.00 CSR MSKY01 KLIST
2364.00 CSR KFLD MSUSER
2365.00 CSR KFLD MSFILE
2366.00 CSR KFLD MSMCUT
2367.00 C*-----------------------------------------------
2368.00 CSR QXKY01 KLIST
2369.00 CSR KFLD QXXIT
2370.00 C*-----------------------------------------------
2371.00 C* Load roll key upper and lower key values.
2372.00 C*-----------------------------------------------
2373.00 CSR *LIKE DEFN QXXIT $RUKEY
2374.00 CSR *LIKE DEFN $RUKEY $RDKEY
2375.00 CSR MOVE *LOVAL $RUKEY
2376.00 CSR MOVE *ALL'9' $RDKEY
2377.00 C*-----------------------------------------------
Advanced Programming Concepts and Skills

2386.00 C* Load error messages array.

2387.00 C*  

2388.00 CSR 

2389.00 CSR 

2390.00 CSR 

2391.00 CSR 

2392.00 CSR 

2393.00 CSR 

2394.00 CSR 

2395.00 CSR 

2396.00 CSR 

2397.00 CSR 

2398.00 C*  

2399.00 C*  

2400.00 C*  

2401.00 C*  

2402.00 CSR 

2403.00 C*  

2404.00 C*  

2405.00 C*  

2406.00 C*  

2407.00 CSR 

2408.00 CSR 

2409.00 CSR 

2410.00 CSR 

2411.00 CSR 

2412.00 CSR 

2413.00 CSR 

2414.00 CSR 

2415.00 CSR 

2416.00 C*  

2417.00 CSR 

2418.00 CSR 

2419.00 CSR 

2420.00 CSR 

2421.00 CSR 

2422.00 CSR 

2423.00 CSR 

2424.00 C*  

2425.00 CSR 

2426.00 C*  

2427.00 C*  

2428.00 G192801 E  

---

Error message numbers from Data Dictionary

Lockout action code function used with the Program Generator

Use the TIME feature to allow for all date formats

Method of releasing master file record locks
Exercises

See the exercises for this chapter.
User Spaces

About User Spaces

User spaces are objects managed by Application Program Interfaces (APIs) to store data. User object APIs create, manipulate, and delete user spaces and indexes. An API provides you with:

- A faster method of retrieving information
- A means of dynamically modifying sizes
- A means of manipulating user objects

You should place your user spaces in library QTEMP so that it is deleted automatically when you sign off. In this chapter you will learn the following about user spaces.

To work with user spaces, perform the following tasks:

- Create a user space
- Write to a user space
- Read from a user space
What Is a User Space?

A user space is an object made up of a collection of bytes that are used for storing any user defined information.

When you use a user space, there is no key to retrieve the information placed in the space. Therefore, the information in the user space is in the order that it was entered. A user space can store up to 16 megabytes of information.

To see the contents of a User Space, enter the command DMPOBJ (Dump Object) from any command line after the space has been loaded.

What Are the Advantages of Using a User Space?

The main advantage of using a user space is its speed. Because a user space consists of bytes instead of elements like an array, you can write and retrieve records faster using a user space than an array.

In addition to speed, a user space provides you with more flexibility. A user space does not have a fixed record length. When you write a record to a user space, you define the length of that record. Therefore, each record you write to your user space can be a different size. In addition, it is possible to dynamically increase the size of your user space by calling the Enter User Space program (X00SPC) after creating the user space.

For example: @EX 999 30

The array @EX has a fixed record length of 30, therefore no record smaller or larger than 30 bytes can be written to this array.

User spaces are also used when communicating between two programs. The space can carry information loaded in one program to another program for retrieval.

For example: Program A creates the user space and loads information into a user space. Then Program A calls Program B and passes the name of the user space to it. Program B can retrieve information from the user space that was loaded by Program A.
**How Does a User Space Function?**

Remember that a user space is nothing more than a collection of bytes used to store information:

---

You write information to a user space, as well as retrieve information from it. Since there is no key associated with a user space, the information contained in a user space is in a user-defined order. The order is based on program controlled offset and length values.

---

**Creating a User Space**

▶ **To create a User Space**

1. Determine if a user space already exists by using the J.D. Edwards program J98CKOBJ.

For example:

<table>
<thead>
<tr>
<th>CALL</th>
<th>'J98CKOBJ'</th>
<th>81</th>
</tr>
</thead>
<tbody>
<tr>
<td>PARM</td>
<td>PSOBJ</td>
<td></td>
</tr>
<tr>
<td>PARM</td>
<td>PSLIB</td>
<td></td>
</tr>
<tr>
<td>PARM</td>
<td>PSTYPE</td>
<td></td>
</tr>
<tr>
<td>PARM</td>
<td>PSMID</td>
<td></td>
</tr>
<tr>
<td>PARM</td>
<td>PSAUT</td>
<td></td>
</tr>
<tr>
<td>PARM</td>
<td>PSERR</td>
<td></td>
</tr>
</tbody>
</table>

**PARM (Length)**

<table>
<thead>
<tr>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PSOBJ (10)</td>
</tr>
<tr>
<td>The name of your user space.</td>
</tr>
<tr>
<td>PSLIB (10)</td>
</tr>
<tr>
<td>The name of the library in which you wish to check for the existence of the</td>
</tr>
<tr>
<td>user space. Generally, this is *LIBL to check all of the libraries in the</td>
</tr>
<tr>
<td>library list.</td>
</tr>
</tbody>
</table>
### PARM (Length) | Description
--- | ---
PSTYPE (8) | The type of object you are checking for. Generally, this is *USRSPC for a user space.
PSMID (10) | The member ID if you are checking for a database file. Generally, this is *NONE.
PSAUT (10) | The authority or authorization list to be checked for the user. Generally, this is *NONE.
PSERR (1) | The error parameter that will indicate an error while checking your object. Generally, this is *BLANK.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>No authority</td>
</tr>
<tr>
<td>1</td>
<td>Not found</td>
</tr>
<tr>
<td>3</td>
<td>No library</td>
</tr>
<tr>
<td>4</td>
<td>Member not found</td>
</tr>
<tr>
<td>5</td>
<td>No authority to library</td>
</tr>
<tr>
<td>6</td>
<td>Cannot assign library</td>
</tr>
</tbody>
</table>

2. If a user space does exist you should clear it and write your new information over the old.

3. If the user space does not exist and no errors occurred, you can create your user space. To create a user space, use the QUICRTUS (Create User Space) command.

For example:

```
CALL 'QUICRTUS' 81
```

<table>
<thead>
<tr>
<th>PARM</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>PARM #SPNAM</td>
<td>The first 10 characters contain your user space name, and the second 10 characters contain the name of the library where your user space is located. Remember, place your user space in library QTEMP to automatically delete your space when you sign off.</td>
</tr>
<tr>
<td>PARM #SPATT</td>
<td>The extended attribute of your user space. You may use this field to classify your user space. For example, JDE uses this field to label all of the user spaces with JDE.</td>
</tr>
</tbody>
</table>
**PARM (Length) | Explanation**
---|---
#SPSIZ  (4 binary) | The initial size of your user space. Any value from 1 byte to 16 megabytes.
#SPVAL  (1) | The initial value of all bytes in the user space. Generally, this is *BLANK.
#SPAUT  (10) | The authority you give users to your user space. Generally, this is *ALL.
#SPTXT  (50) | The text description of your user space.

4. To dynamically increase the size of your user space when maximum allocation is reached, call the Enlarge User Space program (X00SPC).

For example: CALL ‘X00SPC’ 81

---

**PARM (Length) | Explanation**
---|---
#XSPCN (20) | The first 10 characters contain your user space name, and the second 10 characters contain the name of the library where your user space is located. Remember to place your user space in library QTEMP to automatically delete your space when you sign off.
#XRQSZ (15,0) | The requested size to increase your space.
#XERR  (1) | An error flag:
1 – Space not found
2 – Not authorized
3 – Error
## Writing to a User Space

### To write to a User Space

Use either the QUSCHGUS or the X98CHGUS (Change User Space) command.

For example:

```
CALL 'QUSCHGUS' 81
   PARM #SPNAM
   PARM #SPPOS
   PARM #SPLGH
   PARM #SPVAL
   PARM #SPAUX
```

<table>
<thead>
<tr>
<th>PARM (Length)</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>#SPNAM (20)</td>
<td>The first 10 characters contain your user space name, and the second 10 characters contain the name of the library where your user space is located. Remember to place your user space in library QTEMP to automatically delete your space when you sign off.</td>
</tr>
<tr>
<td>#SPPOS (4 binary)</td>
<td>The starting position in your user space where the information will begin. It must be the first byte and must have a value greater than 0.</td>
</tr>
<tr>
<td>#SPLGH (4 binary)</td>
<td>The length of the information that is being written to your user space. This field is user-defined, but it must be greater than 0.</td>
</tr>
<tr>
<td>#SPVAL (* user defined)</td>
<td>The actual information to be written to your user space. The field must be at least as long as the length parameter.</td>
</tr>
</tbody>
</table>
| #SPAUX (1)     | Used to force changes made to your user space to auxiliary storage, such as a disk. The valid values are:  
  0 – do not force changes  
  1 – write changes  
  2 – write changes immediately |

The X98CHGUS program, JDE's version of the IBM command QUSCHGUS, will perform a transfer control to QUSCHGUS.
Tracking Information if Writing Variable Length Records

Method 1

During the process of writing information to your user space, you should keep track of a pointer. This will ensure that you will not overwrite information or retrieve incorrect information.

One way to do this is to initialize your pointer to 1 and after you write information to your user space, add the length of the information to your pointer. The pointer is now set at the next starting point and ready for you to enter new information.

If the information you are writing to your user space contains various lengths, you should maintain the length of each piece of information in save fields. You can use the save fields when you wish to retrieve the information from your user space.

<table>
<thead>
<tr>
<th>Initialize pointer to 1</th>
<th>Add 30 bytes to pointer and to save field</th>
<th>Add 41 bytes to pointer and to another save field</th>
</tr>
</thead>
<tbody>
<tr>
<td>▼</td>
<td>▼</td>
<td>▼</td>
</tr>
<tr>
<td>◄ 30 bytes ►</td>
<td>◄ 41 bytes ►</td>
<td>▼</td>
</tr>
</tbody>
</table>

Method 2

You can also reserve the first 2 or 3 bytes of every record for the size of that record. Then you would only have to load that part of the record with its length. When you read the record from the user space, the first 2 or 3 bytes will tell you how long the record is.
Reading from a User Space

Once you have loaded information into your user space, you are ready to retrieve it. Do not forget that your pointer must be set to the proper starting position to ensure the correct information is retrieved.

To read from a User Space

Use the QUSRTVUS (Retrieve User Space) command.

For example:  CALL ‘QUSRTVUS’ 81

          |  PARM  |  PARM  |  PARM  |  PARM  |
          |  #SPNAM|  #SPPOS|  #SPLGH|  #SPREC|

<table>
<thead>
<tr>
<th>PARM (Length)</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>#SPNAM (20)</td>
<td>The first 10 characters contain your user space name, and the second 10 characters contain the name of the library where your user space is located. Remember to place your user space in library QTEMP to automatically delete your space when you sign off.</td>
</tr>
<tr>
<td>#SPPOS (4 binary)</td>
<td>The starting position in your user space where the information will begin. It must be the first byte and must have a value greater than 0.</td>
</tr>
<tr>
<td>#SPLGH (4 binary)</td>
<td>The length of the information that is being retrieved to your user space. This field is user-defined, it must not be larger than the variable that will receive the information, and it must be greater than 0.</td>
</tr>
<tr>
<td>#SPREC (* user defined)</td>
<td>The variable that will receive the information from your user space.</td>
</tr>
</tbody>
</table>
User Indices

About User Indices

A user index is an object that will:

- Store data
- Allow search functions
- Automatically sort data based on its value

When you use a user index you must have a key to retrieve the information placed in the index. The key must be unique and you can only retrieve data using the key in ascending or descending order.

When you enter data into a user index, it is placed in order according to its value.

A user index can store up to 4 gigabytes of information. Each key and record within a user index can be 1 to 999 bytes long.

To see the contents of a user index, enter the command DMPOBJ (Dump Object) from any command line after the index has been loaded.

You should place your user indices in library QTEMP so that it will be deleted automatically when you sign off.

To work with user indices, perform the following tasks:

- Create a User Index
- Write to a User Index
- Retrieve from a User Index
What Are the Advantages of Using a User Index?

When you load data into your user index, it is automatically sorted for you. Based on your key for the index, the information is arranged according to its value. This will help streamline table searches, cross referencing, and the ordering of data.

The size flexibility of a user index is much better than an array because arrays have a fixed size. A user index is only as big as the information it contains at one time. User indices expand as you add data to them.

For example: @EX 999 30

The array @EX has a fixed size of approximately 3 kilobytes. Each record must be 30 bytes long and up to 999 records can be loaded. If you have 300 records loaded into @EX, you will waste approximately 2 kilobytes. On the other hand, if you have 1500 records to load, the program will error when record number 1000 is loaded. A user index would be able to accommodate both situations.

- A user index is able to retrieve records faster than an array.
- Although a user index may expand to hold more records, it will not contract when records are removed. If you load 100 records into a user index and then remove 50 of them, the user index will remain at the 100 record level size.

You may retrieve data from a user index in ascending order or descending order. When data is loaded into a user index, it is loaded in ascending order. This does not restrict you to retrieving it in this order.
How Does a User Index Function?

A user index stores data and allows you to retrieve it by a key, which must be unique. The data it stores is made up of a data structure that consists of several fields that you wish to store. A user index is capable of expanding when you add data to it.

J.D. Edwards leaves the first byte in the user index blank for clearing purposes.

When using a user index you can create it, add data to it, remove data from it, and delete it.

User indices, like user spaces, should be created in your QTEMP library so you do not have to worry about deleting them.
Creating a User Index

Before you actually create a user index, check to see if one already exists using the JDE program J98CKOBJ.

For example:

```
CALL 'J98CKOBJ' 81
```

- **PSOBJ** (10)
  The name of your user index.

- **PSLIB** (10)
  The name of the library in which you wish to check for the existence of the user index. Generally, this is *LIBL to check all of the libraries in the library list.

- **PSTYPE** (8)
  The type of object you are checking for. Generally, this is *USRIDX for a user index.

- **PSMID** (10)
  The member if you are checking for a database file. Generally, this is *NONE.

- **PSAUT** (10)
  The authority or authorization list to be checked for the user. Generally, this is *NONE.

- **PSERR** (1)
  The error parameter that will indicate an error while checking your object. Generally, this is *BLANK.
  - 0 – No authority
  - 1 – Not found
  - 3 – No library
  - 4 – Member not found
  - 5 – No authority to library
  - 6 – Cannot assign library

---

**PARM (Length) | Explanation**

| **PSOBJ** (10) | The name of your user index. |
| **PSLIB** (10) | The name of the library in which you wish to check for the existence of the user index. Generally, this is *LIBL to check all of the libraries in the library list. |
| **PSTYPE** (8) | The type of object you are checking for. Generally, this is *USRIDX for a user index. |
| **PSMID** (10) | The member if you are checking for a database file. Generally, this is *NONE. |
| **PSAUT** (10) | The authority or authorization list to be checked for the user. Generally, this is *NONE. |
| **PSERR** (1) | The error parameter that will indicate an error while checking your object. Generally, this is *BLANK. |
  - 0 – No authority
  - 1 – Not found
  - 3 – No library
  - 4 – Member not found
  - 5 – No authority to library
  - 6 – Cannot assign library **|**
If a user index exists, clear it and write your new information over the old.

For example:  

```
CALL 'X00IDX' 81
--- --- ---
PARM #0XNAM
PARM 'D' #0XACT
PARM 'EQ' #0XRUL
PARM '1' #0XKLN
PARM *BLANK #0XKEY
PARM #0XRLN
PARM #0XREC
PARM #0XSTA
```

If the user index did not exist, you can now create your user index.

▶ **To create a User Index**

Use the QUSCRTUI (Create User Index) command.

For example:  

```
CALL 'QUSCRTUI' 81
--- --- ---
PARM #IDNAM
PARM #IDA TT
PARM #IDENT
PARM #IDLEN
PARM #IDINS
PARM #IDKEY
PARM #IDUPD
PARM #IDOPT
PARM #IDAUT
PARM #IDTXT
```

### PARM (Length)

<table>
<thead>
<tr>
<th>PARM (Length)</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>#IDNAM (20)</td>
<td>The first 10 characters contain your user index name, and the second 10 characters contain the name of the library where your user index is located. Remember to place your user index in library QTEMP to automatically delete your index when you sign off.</td>
</tr>
<tr>
<td>PARM (Length)</td>
<td>Explanation</td>
</tr>
<tr>
<td>---------------</td>
<td>-------------</td>
</tr>
<tr>
<td>#IDATT (10)</td>
<td>The extended attribute of your user index. You may use this field to classify your user index. For example, JDE uses this field to label all of the user indexes with JDE.</td>
</tr>
<tr>
<td>#IDENT (1)</td>
<td>Whether the records you are loading into your user index are Fixed–length (F) or Variable–length (V). Generally, this is set to ‘F’.</td>
</tr>
<tr>
<td>#IDLEN (4 binary)</td>
<td>The length of the records to be entered into your user index. For fixed–length records valid values are 1 to 999. For variable–length records, enter 0 for a key length of 1 to 120, or 1 for a key length of 1 to 999.</td>
</tr>
<tr>
<td>#IDINS (1)</td>
<td>Whether you are loading your user index by a key or not. Generally, this is set to 1 to load your index by a key. A value of 0 means you are not loading your index by a key.</td>
</tr>
<tr>
<td>#IDKEY (4 binary)</td>
<td>The length of your key. The first byte in your record must be the beginning of your key. The values are 1 to 999 or 0 for no key.</td>
</tr>
<tr>
<td>#IDUPD (1)</td>
<td>Whether or not the data in your user index will be immediately updated. Each data change to your index is written to auxiliary storage. The values are 0 for no immediate update or 1 for immediate update. Generally, this is 0.</td>
</tr>
<tr>
<td>#IDOPT (1)</td>
<td>The type of access in which to optimize your index. The values are 0 to optimize for random references or 1 to optimize for sequential references. Generally, this is 1.</td>
</tr>
<tr>
<td>#SPAUT (10)</td>
<td>The authority you give users to your user index. Generally, this is *ALL.</td>
</tr>
<tr>
<td>#SPTXT (50)</td>
<td>The text description of your user index.</td>
</tr>
</tbody>
</table>

You may want to define data structures containing some of the information required for the parameters to avoid having to enter values. The user index name, record length, key length, and user index text are good examples.
Writing to a User Index

To write to a User Index

J.D. Edwards provides an external program called User Index Server (X00IDX) to manipulate data for user index entries.

For example:

```
CALL 'X00IDX' 81
PARM #0XNAM
PARM #0XACT
PARM #0XRUL
PARM #0XKLN
PARM #0XKEY
PARM #0XRLN
PARM #0XREC
PARM #0XSTA
```

<table>
<thead>
<tr>
<th>PARM (Length)</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>#0XNAM (20)</td>
<td>The first 10 characters contain your user index name, and the second 10 characters contain the name of the library where your user index is located. Remember to place your user index in library QTEMP to automatically delete you index when you sign off.</td>
</tr>
<tr>
<td>#0XACT (1)</td>
<td>The action you want to perform on your user index. The valid values are:</td>
</tr>
<tr>
<td></td>
<td>I – Inquire</td>
</tr>
<tr>
<td></td>
<td>A – Add</td>
</tr>
<tr>
<td></td>
<td>C – Change</td>
</tr>
<tr>
<td></td>
<td>D – Delete</td>
</tr>
<tr>
<td>#0XRUL (2)</td>
<td>The rule used to search your user index using the record. The valid values are:</td>
</tr>
<tr>
<td></td>
<td>EQ – Equal to</td>
</tr>
<tr>
<td></td>
<td>GT – Greater than</td>
</tr>
<tr>
<td></td>
<td>LT – Less than</td>
</tr>
<tr>
<td></td>
<td>GE – Greater than or Equal to</td>
</tr>
<tr>
<td></td>
<td>LE – Less than or Equal to</td>
</tr>
<tr>
<td>PARM (Length)</td>
<td>Explanation</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------</td>
</tr>
<tr>
<td>#0XKLN (3,0)</td>
<td>The length of your key. The first byte in your record must be the beginning of your key. The values are 1 to 999 or 0 for no key.</td>
</tr>
<tr>
<td>#0XKEY (120)</td>
<td>The fields that make up the key to your user index. *FIRST (first record) and *LAST (last record) are allowed.</td>
</tr>
<tr>
<td>#0XRLN (3,0)</td>
<td>The length of your record. The values are 1 to 999.</td>
</tr>
<tr>
<td>#0XREC (120)</td>
<td>The record you are entering or deleting from your user index. This parameter will also receive the record when you inquire on your user index.</td>
</tr>
</tbody>
</table>
| #0XSTA (1)  | The error status of the manipulation. The possible values are:  
  0 – Record found  
  1 – Record not found, not authorized  
  8 – Rule invalid  
  9 – Error on action |
**Appearance of Records**

The records added to your user index will appear in ascending order.

For example: You created a user index to keep track of your ice cream sales. Each record within your user index contains the total sales amount, item, item description, and cost center. The key for your user index consists of total sales amount and item (remember the key must be unique).

The following records are to be loaded into your user index:

<table>
<thead>
<tr>
<th>Total</th>
<th>Sales Item</th>
<th>Description</th>
<th>Cost Center</th>
</tr>
</thead>
<tbody>
<tr>
<td>$ 500.00</td>
<td>CHO</td>
<td>Chocolate</td>
<td>Denver</td>
</tr>
<tr>
<td>$ 250.00</td>
<td>STR</td>
<td>Strawberry</td>
<td>Denver</td>
</tr>
<tr>
<td>$ 750.00</td>
<td>C&amp;C</td>
<td>Cookies &amp; Cream</td>
<td>Denver</td>
</tr>
<tr>
<td>$1200.00</td>
<td>VAN</td>
<td>Vanilla</td>
<td>Denver</td>
</tr>
<tr>
<td>$ 400.00</td>
<td>ROC</td>
<td>Rocky Road</td>
<td>Denver</td>
</tr>
</tbody>
</table>

Because the key to your user index is total sales amount and item, the records will be entered into your index in ascending order by total sales amount first, then item. So your user index will look like this:

**Ice Cream Sales Index**

<table>
<thead>
<tr>
<th>Total</th>
<th>Sales Item</th>
<th>Description</th>
<th>Cost Center</th>
</tr>
</thead>
<tbody>
<tr>
<td>$ 250.00</td>
<td>STR</td>
<td>Strawberry</td>
<td>Denver</td>
</tr>
<tr>
<td>$ 400.00</td>
<td>ROC</td>
<td>Rocky Road</td>
<td>Denver</td>
</tr>
<tr>
<td>$ 500.00</td>
<td>CHO</td>
<td>Chocolate</td>
<td>Denver</td>
</tr>
<tr>
<td>$ 750.00</td>
<td>C&amp;C</td>
<td>Cookies &amp; Cream</td>
<td>Denver</td>
</tr>
<tr>
<td>$1200.00</td>
<td>VAN</td>
<td>Vanilla</td>
<td>Denver</td>
</tr>
</tbody>
</table>
Retrieving Data from a User Index

You can retrieve data in ascending or descending order.

▶ To retrieve data in Ascending Order

1. Use the User Index Server (X00IDX).
2. Set the Action parm to inquire (I)
3. Set Rule to Equal to (EQ)
4. Set the Key to the first record (*FIRST)
   For example:  CALL  ‘X00IDX’
                  ------------------
                  PARM  #0XNAM 20
                  PARM  ‘I’    #0XACT 1
                  PARM  ‘EQ’   #0XRUL 2
                  PARM  #0XKLN 30
                  PARM  “*FIRST”  #0XKEY120
                  PARM  #0XRLN 30
                  PARM  #0XREC120
                  PARM  #0XSTA 1

5. To retrieve the next record, load the key with the current record’s values and change your rule to “GT”.
   For example:  CALL  ‘X00IDX’
                  ------------------
                  PARM  #0XNAM 20
                  PARM  ‘I’    #0XACT 1
                  PARM  ‘GT’   #0XRUL 2
                  PARM  #0XKLN 30
                  PARM  #0XKEY120
                  PARM  #0XRLN 30
                  PARM  #0XREC120
                  PARM  #0XSTA 1
To retrieve data in descending order

1. Use the User Index Server (X00IDX)
2. Set the Action parm to inquire (I)
3. Set Rule to Equal to (EQ)
4. Set the Key to the first record (*LAST)
   For example: CALL 'X00IDX'
   
   PARM #0XNAM 20
   PARM 'I' #0XACT 1
   PARM 'EQ' #0XRUL 2
   PARM #0XKLN 30
   PARM '*LAST' #0XKEY120
   PARM #0XRLN 30
   PARM #0XREC120
   PARM #0XSTA 1

5. To retrieve the next record, load the key with the current record’s values and change your rule to “LT”.
   For example: CALL 'X00IDX'
   
   PARM #0XNAM 20
   PARM 'I' #0XACT 1
   PARM 'LT' #0XRUL 2
   PARM #0XKLN 30
   PARM #0XKEY120
   PARM #0XRLN 30
   PARM #0XREC120
   PARM #0XSTA 1
# User Index Example Program

<table>
<thead>
<tr>
<th>Date</th>
<th>Programmer</th>
<th>Nature of Revision</th>
</tr>
</thead>
<tbody>
<tr>
<td>12/02/93</td>
<td>FRAZZINI</td>
<td>SAR # 289 (AS/400 A/G)</td>
</tr>
</tbody>
</table>

| Data Structure to Load Video Screen Text
<table>
<thead>
<tr>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>IDSTXT DS 240</td>
</tr>
<tr>
<td>1 16 VTX001</td>
</tr>
<tr>
<td>41 56 VTX002</td>
</tr>
<tr>
<td>81 92 VTX003</td>
</tr>
<tr>
<td>121 150 VTX004</td>
</tr>
<tr>
<td>161 163 VTX005</td>
</tr>
<tr>
<td>201 203 VTX006</td>
</tr>
</tbody>
</table>
83.00  I*  
84.00  I*  Copy Member for Composite Common Subroutine – C00SC  
85.00  I*  
86.00  I*  
87.00  I/COPY JDECPY, I00SC  
88.00  I*  
89.00  I*  Data Structures for user index.  
90.00  I*  ===============================================================  
91.00  I*  
92.00  I*  
93.00  I*  Entry Record  
94.00  I*  
95.00  I*  
96.00  I  1  1 $1BLK  
97.00  I  2  6 $1CO  
98.00  I  7  18 $1MCU  
99.00  I  19  48 $1DL01  
100.00  I  49  51 $1RP01  
101.00  I  52  54 $1RP02  
102.00  I*  
103.00  I*  Entry Length, Name/Library, Text  
104.00  I*  
105.00  I*  
106.00  I*  
107.00  I I  54                        B  1  40$1RECL  
108.00  I I  'PINDEX QTEMP            5  24 $1IDX  
109.00  I I  'Demonstration Index      25  44 $1TEXT  
110.00  I*  
111.00  I*  Partial keys 1 & 2, full unique key KEYL.  
112.00  I*  
113.00  I*  
114.00  I I  1                        B  1  40$1KEY1  
115.00  I I  6                        B  5  80$1KEY2  
116.00  I I  18                       B  9  120$1KEYL  
117.00  I*  
118.00  I*  
119.00  I*  ****************************  
120.00  I*  Data Structure for File Servers  
121.00  I*  
122.00  I*  MAINLINE PROGRAM  
123.00  I*  
124.00  I*  Process housekeeping.  
125.00  I*  
126.00  I*  EXSR S999  
127.00  I*  
128.00  I*  End  
129.00  I*  
130.00  I*  
131.00  I C  "INLR CBEQ’1’ EOJ  
132.00  I C  "------- "  
133.00  I C  
134.00  I C  
135.00  I C  
136.00  I C  "$AUTO CASEQ’1’ S003 24  
137.00  I C  "------- "  
138.00  I C  END  
139.00  I C  
140.00  I C  Begin normal program processing.  
141.00  I C  
142.00  I C  "INLR DWEQ’0’  
143.00  I C  
144.00  I C  
145.00  I C  If subfile page display not set, set subfile page display.  
146.00  I C  
147.00  I C  #$SFRNO IFEQ 0  
148.00  I C  Z-ADD1 #$SFRNO  
149.00  I C  END  
150.00  I C  
151.00  I C  
152.00  I C  
153.00  I C  I1 IFLE 0  
154.00  I C  SETOF 38  
155.00  I C  ELSE  
156.00  I C  SETON 38  
157.00  I C  END  
158.00  I C  
159.00  I C  Write video screen.
160.00 C* WRITEVINDEX1
161.00 C WRITEVINDEXC
162.00 C MOVE ‘1’ @AID
164.00 C EXSR S001
165.00 C* ---- ----
166.00 C* Load data field dictionary parameters (one cycle only).
167.00 C* $998 CASEQ’ ’ S998
170.00 C* ------ ----
171.00 C END
172.00 C* Begin video screen read processing
173.00 C* LOAD VINDEX
174.00 C* SETOF 999301
175.00 C READ VINDEX 9998
176.00 C Z–ADD0 ##RROW
177.00 C Z–ADD0 #RCOL
178.00 C END
179.00 C* If video read timed out, end program.
180.00 C* *IN99 CABEQ’1’ EOJ LR
181.00 C* @@AID CABEQ#FEOJ EOJ LR
182.00 C* –––– –––
183.00 C* If valid function key pressed, process and return.
184.00 C* $998 CASEQ’ ‘ S998
185.00 C* ------- ----
186.00 C* If end of job requested, end program.
187.00 C* @AID IFEQ #FCLR
188.00 C* –––– –––
189.00 C* Edit the action code.
190.00 C EXSR C0001
191.00 C* ---- ----
192.00 C* If add or change, validate all video input.
193.00 C* $998 CASEQ’ ‘ S998
194.00 C* ------- ----
195.00 C END
196.00 C* If clear screen requested, process and return.
197.00 C* *IN93 CASEQ’0’ S005
198.00 C* –––– –––
199.00 C* If no errors and not inquiry, update file.
200.00 C EXSR S003
201.00 C* ---- ----
202.00 C* Return for next input.
203.00 C* *IN93 IFEQ ‘1’
COPY JDECPY,C0001

SUBROUTINE S00EX – Process Function Keys

2. Process special function key exits.

Retain current page of subfile.

If EOJ requested, exit subroutine.

If Display Keys pressed, exit to help facility and return.

If Cursor Sensitive Help Pressed, exit to CS Help.

If Display errors pressed, exit to error messages.

Copy Common Subroutine – Edit Action Code

END MAINLINE PROGRAM

***************************************************************
COPY JDECPY,C0001
***************************************************************
If HELP key pressed, exit to help facility and return.

If ROLL UP key pressed, load next page of subfile.

If ROLL DOWN key pressed, reset subfile page display.

If Clear screen pressed, clear screen and return.
SUBROUTINE S00VL – Cursor Control Return Values

By format, find the field to update and move in the returned value. If the format is a subfile, the record to change is found in @@RRN.

CSR S00VL BEGSR

CSR ##RVAL IFEQ '*BLANK'
MOVE *BLANK ##RVAL
END

Return values for fields in format VINDEXC

CSR ##RFMT IFEQ 'VINDEXC'
CSR ##FLDN IFEQ 'ACTION'
MOVEL##RVAL ACTION
GOTO ENDOVL

CSR ##FLDN IFEQ 'VDCO'
MOVEL##RVAL VDCO
GOTO ENDOVL

Return values for fields in format VINDEXS

CSR ##RFMT IFEQ 'VINDEXS'
CSR @@RRN ANDGTO
CSR MOVEL##IN SHIN
CSR @@RRN CHAINVINDEXS 81
CSR *IN81 IFEQ '0'
MOVEASHIN *IN,1

CSR ##FLDN IFEQ 'SFMCU'
MOVEL##RVAL SFMCU
GOTO ENDOVL

CSR ##FLDN IFEQ 'SFDL01'
MOVEL##RVAL SFDL01
GOTO ENDOVL

CSR ##FLDN IFEQ 'SFPR01'
MOVEL##RVAL SFPR01
GOTO ENDOVL

Return values for fields in format VINDEX1

CSR ##RFMT IFEQ 'VINDEX1'
CSR MOVEL##RVAL SFPR02
GOTO T00VLA

CSR T00VLA TAG

CSR ---- ----
CSR SETON 32
CSR MOVEA*IN SHIN
CSR UPDATVINDEXS 81
CSR END
CSR END
CSR END
CSR END
CSR END
CSR C* SUBROUTINE S001 – Clear Fields
495.00  C*  ------------------------------------------
496.00  C*  Processing:  1.  Reset all video screen and data file fields
497.00  C*  for next transaction.
498.00  C*  2.  Clear action code only if requested.
499.00  C*
500.00  C*  501.00  CSR  S001  BEGSR
502.00  C*  ------  ------
503.00  CSR  MOVE *BLANK  $1DL01
504.00  CSR  MOVE *BLANK  $1RP01
505.00  CSR  MOVE *BLANK  $1RP02
506.00  CSR  Z–ADD*ZERO  #$RCOL
507.00  CSR  Z–ADD*ZERO  #$RROW
508.00  CSR  Z–ADD*ZERO  #$SRNO
509.00  CSR  MOVE *BLANK  SFDL01
510.00  CSR  MOVE *BLANK  SFMCU
511.00  CSR  MOVE *BLANK  SFPR01
512.00  CSR  MOVE *BLANK  SFPR02
513.00  CSR  MOVE *BLANK  SHMCU
514.00  CSR  MOVE *BLANK  VDCO
515.00  CSR  MOVESVL24M  VDL24
516.00  CSR  MOVE '0'  SHIN37
517.00  C*
518.00  C*  519.00  C*  Clear action code only if clear screen action.
520.00  C*
521.00  CSR  @@AID  IFEQ  #FCLR
522.00  CSR  MOVE *ALL'0'  $RESET
523.00  CSR  MOVES$RESET  *IN,41
524.00  CSR  MOVE *'0'  ACTION  1
525.00  CSR  Z–ADD00000  #$SRNO
526.00  CSR  SETON  31
527.00  CSR  WRITEINDEXC  99
528.00  CSR  SETOF  203193
529.00  CSR  Z–ADD  11
530.00  CSR  DO  $PGSZ
531.00  CSR  ADD  1  11
532.00  CSR  MOVEA*IN  SHIN
533.00  CRS  WRITEINDEXS  81
534.00  CSR  END
535.00  CSR  Z–ADDI1  $SVI1
536.00  CSR  MOVE *BLANK  $1CO
537.00  CSR  MOVE *BLANK  $1MCU
538.00  CSR  MOVE *BLANK  VC0001
539.00  CSR  END
540.00  C*
541.00  C*  542.00  C*  SUBROUTINE S003 – Edit Key
543.00  C*
544.00  C*  545.00  C*  -------------------
546.00  C*  547.00  C*  Processing:  1.  Initialize error arrays and subfile.
548.00  C*  2.  Load inquiry selection.
549.00  C*  3.  Load subfile information.
550.00  C*  3.  Monitor for empty subfile.
551.00  C*
552.00  CSR  S003  BEGSR
553.00  C*  ------  ------
554.00  CSR  END001  ENDSR
555.00  C*  556.00  C*  Reset error indicators and arrays.
557.00  CSR  MOVE *ALL'0'  $RESET 39
558.00  CSR  MOVE *BLANK  $REST1 63
559.00  CSR  MOVES$RESET  *IN,41
560.00  CSR  MOVES$RESET  @MK,2
561.00  CSR  CLEAR@ER
562.00  C*  563.00  C*  Clear the user index to begin with; set flag.
564.00  C*  565.00  C*  -----------------------------------------------------------------
566.00  CSR  MOVE 'Y'  $START  1
567.00  C*  568.00  C*  Load video input field for - Company
569.00  C*  570.00  C*  571.00  CSR  MOVEAVDCO  @NM
572.00  CSR  EXSR  C0012
573.00  C*  ------  ------
574.00  CSR  Z–ADD#NUMR  $WK5  50
575.00  CSR  MOVE $WK5  $1CO
576.00  CSR  MOVE $WK5  VDCO
577.00  C*  -----------------------------------------------------------------
578.00 C*  Determine if any entries exist for that company.
579.00 C*
580.00 C*
581.00 CSR Z-ADD$1KEY2 PSKEYL
582.00 CSR Z-ADD$1KEYCL PSKEYL
583.00 CSR MOVELOGIDX1 PSKY
584.00 C*
585.00 CSR CALL 'X00IDX'
586.00 C*
587.00 CSR PARM $1IDX  Idx Name Lib
588.00 CSR PARM 'Y' PSACTN Action Code
589.00 CSR PARM 'EQ' PSRULE Action Rule
590.00 CSR PARM PSKEYL Key Length
591.00 CSR PARM PSKY Key Fields
592.00 CSR PARM PSRECL Entry Length
593.00 CSR PARM PSREC Entry
594.00 CSR PARM PSSTS Error Status
595.00 C*
596.00 CSR *IN23 COMP '1'             41 *ERROR*
597.00 C*
598.00 CSR *IN99 IFEQ '1'
599.00 CSR MOVE '1' @MK,6
600.00 CSR SETON             4193
601.00 CSR END
602.00 C*  If indicator 41 on, invalid key for action code.
603.00 C*  If indicator 99 on, record in use.
604.00 CSR *IN41 IFEQ '1'
605.00 CSR MOVE '1' @MK,2
606.00 CSR SETON             93
607.00 CSR END
608.00 C*  If not inquiry, skip remainder of subroutine.
609.00 C*  If errors, skip remainder of subroutine.
610.00 C*  Initialize subfile indexes.
611.00 CSR *IN93 CABEQ'0' END003
612.00 C*
613.00 CSR SETON             31
614.00 CSR WRITEVINDEXC  99
615.00 CSR SETOF             31
616.00 C*  Reinitialize subfile display
617.00 CSR EXSR S004
618.00 CSR Z-ADD0 $SVI1 50
619.00 CSR Z-ADD0 $SVI1 50
620.00 CSR Z-ADD0 $SPARO
621.00 CSR MOVE 'O' $SEND 1
622.00 C*
623.00 CSR *IN93 CABEQ'1' END003
624.00 C*  ----- ------
625.00 C*  ----- ------
626.00 C*  ----- ------
627.00 CSR
628.00 CSR
629.00 CSR
630.00 CSR
631.00 CSR
632.00 C*
633.00 C*
634.00 C*  ----- ------
635.00 CSR SETON             31
636.00 CSR WRITEVINDEXC  99
637.00 CSR SETOF             31
638.00 C*  Load subfile records.
639.00 C*  ----- ------
640.00 CSR
641.00 CSR
642.00 C*  ----- ------
643.00 C*  ----- ------
644.00 CSR
645.00 CSR
646.00 C*
647.00 CSR
648.00 CSR
649.00 CSR
650.00 CSR
651.00 CSR
652.00 CSR
653.00 CSR
654.00 CSR
655.00 CSR
656.00 CSR
657.00 CSR
658.00 CSR
659.00 CSR

Load key length, record length, and key with values

Call to User Index to inquire on an existing record

Check error status parameter to see if a record was found
**Advanced Programming Concepts and Skills**

660.00 C***************************************************************************
661.00 C* Copy Common Subroutine – Right Justify Numeric Fields
662.00 C***************************************************************************
663.00 C* SUBROUTINE S004 – Load Video Screen Data
664.00 C***************************************************************************
665.00 C***************************************************************************
666.00 C***************************************************************************
667.00 C* Processing: 1. Move data base information to video screen.
668.00 C* All video screen fields are alpha and
669.00 C* therefore numeric information must be
670.00 C* processed through subroutine C0014 to set
671.00 C* proper decimals and provide editing for
672.00 C* display on screen.
673.00 C* Date fields must be converted from their
674.00 C* internal format of month, day and year or
675.00 C* julian to the system format using program
676.00 C* X0028.
677.00 C***************************************************************************
678.00 CSR S004 BEGSR
679.00 C* ––––– –––––
680.00 CSR $SEND IFEQ '1'
681.00 CSR Z–ADD0 #SFRNO
682.00 CSR GOTO END004
683.00 CSR END
684.00 C*––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––
685.00 CSR MOVE $1CO $$CO    5
686.00 C*––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––
687.00 CSR $SEND IFEQ '1'
688.00 CSR Z–ADD0 #SFRNO
689.00 CSR Z–ADD$SVI1 I1
690.00 CSR END
691.00 C*––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––
692.00 CSR MOVE *BLANKS PS@@
693.00 CSR MOVEL$1CO KY@@
694.00 CSR CALL 'XS0010'              81
695.00 CSR PARM PS@@
696.00 CSR PARM DS0010
697.00 CSR MOVELCCNAME VC0001
698.00 CSR END
699.00 C*––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––
700.00 CSR Z–ADD0 $PG     30
701.00 CSR Z–ADD0 #SFRNO
702.00 CSR Z–ADD$SVI1 I1
703.00 CSR END
704.00 C* Move to output – company description.
705.00 C* Move to output – company description.
706.00 CSR MOVE *BLANKS PS@@
707.00 CSR MOVEL$1CO KY@@
708.00 CSR CALL 'XS0010'              81
709.00 CSR MOVELCCNAME VC0001
710.00 CSR END
711.00 CSR Z–ADD0 $PG     30
712.00 CSR Z–ADD0 #SFRNO
713.00 CSR Z–ADD$SVI1 I1
714.00 C* Initialize subfile page control and index.
715.00 C* Initialize subfile page control and index.
716.00 CSR Z–ADD0 $PG     30
717.00 CSR Z–ADD0 #SFRNO
718.00 CSR Z–ADD$SVI1 I1
719.00 CSR END
720.00 C* Read user index until end or subfile page filled.
721.00 C* Read user index until end or subfile page filled.
722.00 CSR SETOF 96
723.00 CSR *IN96 DOWEQ’0’
724.00 CSR *IN96 DOWEQ’0’
725.00 CSR SETOF 96
726.00 CSR SETOF 96
727.00 CSR SETOF 96
728.00 CSR SETOF 96
729.00 CSR END
730.00 CSR $START IFEQ 'Y'
731.00 CSR MOVE ‘ ’ $START
732.00 CSR ELSE
Successive times through, read next "greater" entry.

Load key length, record length, and key with values

CALL 'X00IDX'

Retrieve entry to load data structure.

Check error status parameter to see if a record was found.

At end of index, set subfile completion flag and set high intensity attribute on last subfile record.

Reset record selection flag ($SEL).

Update subfile for selected records.

Move to output – Description 01

Move to output – Cost Center
816.00 CSR END
817.00 C***********************************************************************
818.00 CSR Move to output – Category Code – Cost Center 01
819.00 C*
820.00 CSR MOVE *BLANK #SINBR
821.00 CSR MOVEL1RP01 #SINBR
822.00 CSR MOVE T@RP01 #DTYP
823.00 CSR MOVE W@RP01 #EMRD
824.00 CSR MOVE E@RP01 #EC
825.00 CSR MOVE F@RP01 #DSPD
826.00 CSR MOVE G@RP01 #DATD
827.00 CSR MOVE J@RP01 #ALR
828.00 CSR MOVE ' ' #ECOR
829.00 CSR MOVE ' ' #DCOR
830.00 CSR EXSR C00161
831.00 CSR
832.00 CSR MOVE #SINBR SFRP01
833.00 CSR END
834.00 CSR
835.00 CSR ELSE
836.00 CSR MOVE #SINBR SFRP01
837.00 CSR END
838.00 C***********************************************************************
839.00 C*
840.00 CSR Move to output – Category Code – Cost Center 02
841.00 C*
842.00 CSR MOVE *BLANK #SINBR
843.00 CSR MOVEL1RP02 #SINBR
844.00 CSR MOVE T@RP02 #DTYP
845.00 CSR MOVE W@RP02 #EMRD
846.00 CSR MOVE E@RP02 #EC
847.00 CSR MOVE F@RP02 #DSPD
848.00 CSR MOVE G@RP02 #DATD
849.00 CSR MOVE J@RP01 #ALR
850.00 CSR MOVE ' ' #ECOR
851.00 CSR MOVE ' ' #DCOR
852.00 CSR EXSR C00161
853.00 CSR
854.00 CSR #ALR IFEQ 'L'
855.00 CSR MOVEL#SINBR SFRP02
856.00 CSR ELSE
857.00 CSR MOVE #SINBR SFRP02
858.00 CSR END
859.00 C***********************************************************************
860.00 C*
861.00 CSR Move to output – Cost Center
862.00 C*
863.00 CSR MOVE *BLANK #SINBR
864.00 CSR MOVEL1MCU #SINBR
865.00 CSR MOVE T@MCU #DTYP
866.00 CSR MOVE W@MCU #EMRD
867.00 CSR MOVE E@MCU #EC
868.00 CSR MOVE F@MCU #DSPD
869.00 CSR MOVE G@MCU #DATD
870.00 CSR MOVE J@MCU #ALR
871.00 CSR MOVE ' ' #ECOR
872.00 CSR MOVE ' ' #DCOR
873.00 CSR EXSR C00161
874.00 CSR
875.00 CSR #ALR IFEQ 'L'
876.00 CSR MOVEL#SINBR SHMCU
877.00 CSR ELSE
878.00 CSR MOVE #SINBR SHMCU
879.00 CSR CSR END
880.00 C***********************************************************************
881.00 C* Increment subfile page control and index.
882.00 C*
883.00 CSR ADD  1 $PG
884.00 CSR ADD  1 I1
885.00 CSR ADD  1 $PG
886.00 CSR ADD  1 I1
887.00 CSR
888.00 C* If subfile page display not set, set subfile page display.
889.00 C*
890.00 CSR #SFRNO IFEQ 0
891.00 CSR Z-ADD1 #SFRNO
892.00 CSR CSR END
893.00 C*
894.00 CSR Write subfile record and save current subfile index.
895.00 C*
896.00 CSR MOVEA*IN SHIN
897.00 CSR WRITEVINDEXS 99
898.00  CSR  Z-ADD11  $SVI1
899.00  C*  If subfile page loaded, drop out of subroutine.
900.00  C*
901.00  CSR  $PG  CABEQ$PGSZ  END004
902.00  C*  ------  ------
904.00  CSR  END
905.00  CSR  END
906.00  C*  ---------------------------------------------
907.00  CSR  END004  ENDSR
908.00  C*  Copy Common Subroutine - Format Numeric Fields for Output with Override
909.00  C*
910.00  C/COPY JDECPY,C00161
911.00  C************************************************************************
912.00  C*
913.00  C*       SUBROUTINE S005 - Validate and update input data.
914.00  C************************************************************************
915.00  C*
916.00  CSR  S005  BEGSR
917.00  C*   ---   ---
918.00  C*  If not addition or change, bypass subroutine
919.00  C*
920.00  CSR  *IN21  IFEQ '0'
921.00  CSR  *IN22  ANDEQ '0'
922.00  CSR  GOTO  END005
923.00  C*   ---   ---
924.00  CSR  END
925.00  C*
926.00  CSR  MOVE  ' '  $WRT    1
927.00  CSR  Z–ADD1  $$IX    70
928.00  CSR  SETOF  9699
929.00  CSR  *IN96  DOWEQ '0'
930.00  CSR  *IN99  ANDEQ '0'
931.00  CSR  $$IX  ANDLE$SVI1
932.00  CSR  MOVEA$RESET  *IN,41
933.00  CSR  $$IX  CHAINVINDEXS  9699
934.00  CSR  *IN96  IFEQ '0'
936.00  CSR  *IN99  ANDEQ '0'
937.00  CSR  END
938.00  CSR  S005  BEGSR
939.00  C*   ---   ---
940.00  C*  Process all subfile transactions.
941.00  C*
942.00  CSR  MOVE  ' '  $WRT    1
943.00  CSR  Z-ADD1  $$IX    70
944.00  CSR  SETOF  9699
945.00  CSR  *IN96  DOWEQ '0'
946.00  CSR  *IN99  ANDEQ '0'
947.00  CSR  $$IX  ANDEG$SVI1
948.00  CSR  MOVEAS$RESET  *IN,41
949.00  CSR  $$IX  CHAINVINDEXS  9699
950.00  CSR  *IN96  IFEQ '0'
951.00  CSR  *IN99  ANDEQ '0'
952.00  CSR  $WRT  1
953.00  CSR  Z-ADD1  $$IX    70
954.00  CSR  SETOF  9699
955.00  CSR  *IN96  DOWEQ '0'
956.00  CSR  *IN99  ANDEQ '0'
957.00  CSR  $$IX  ANDEG$SVI1
958.00  CSR  MOVEAS$RESET  *IN,41
959.00  CSR  $$IX  CHAINVINDEXS  9699
960.00  CSR  *IN96  IFEQ '0'
961.00  CSR  *IN99  ANDEQ '0'
962.00  CSR  $WRT  1
963.00  CSR  Z-ADD1  $$IX    70
964.00  CSR  SETOF  9699
965.00  CSR  *IN96  DOWEQ '0'
966.00  CSR  *IN99  ANDEQ '0'
967.00  CSR  $$IX  ANDEG$SVI1
968.00  CSR  MOVEAS$RESET  *IN,41
969.00  CSR  $$IX  CHAINVINDEXS  9699
970.00  CSR  *IN96  IFEQ '0'
971.00  CSR  *IN99  ANDEQ '0'
972.00  CSR  $WRT  1
973.00  CSR  Z-ADD1  $$IX    70
974.00  CSR  SETOF  9699
975.00  CSR  *IN96  DOWEQ '0'
976.00  CSR  *IN99  ANDEQ '0'
977.00  CSR  $$IX  ANDEG$SVI1
978.00  CSR  MOVEAS$RESET  *IN,41
979.00  CSR  $$IX  CHAINVINDEXS  9699

---

Loading of parameters and call to User Index to see if a record exists.

---

If no data and prior record existed, delete old record.

---

A8.1 (8/97)  4-97
Delete error status parameter to see if record has found

Deletion of record from User Index

980.00 CSR [FPEST IFNE '0']
981.00 CSR SFMCU ANDEQ*BLANK
982.00 C* ** Process only non-blank records.**
983.00 CSR SFMCU IFNE *BLANK
984.00 C*
985.00 CSR CAMT 'X00IDX'
986.00 CSR PARM $1IDX Idx Name/Lib
987.00 CSR PARM 'D' PSACTN Action
988.00 CSR PARM 'EQ' PSRULE Action Rule
989.00 CSR PARM PSKEYL Key Length
990.00 CSR PARM PSKY Key Fields
991.00 CSR PARM PSRECL Entry Length
992.00 CSR PARM PSREC Entry
993.00 CSR PARM PSSTS Error Status
994.00 C*
995.00 C*
996.00 C* scrubs and edits Description 01
997.00 C*
998.00 CSR MOVELSFDL01 $1DL01
999.00 C*
1000.00 C* scrubs and edits Cost Center
1001.00 C*
1002.00 CSR MOVEASFMCU @FI A
1003.00 CSR EXSR C0042
1004.00 C* set default value – Description 01
1005.00 C*
1006.00 CSR MOVELSFRP01 $1RP01
1007.00 C*
1008.00 CSR $1RP01 IFEQ *BLANK
1009.00 CSR MOVE '1' $MK,03
1010.00 CSR SETON 4293
1011.00 CSR END
1012.00 C*
1013.00 C* scrubs and edits – Category Code – Cost Center 01
1014.00 C*
1015.00 C* set default value – Category Code – Cost Center 01
1016.00 C*
1017.00 CSR MOVEASFMCU 0F1 A
1018.00 CSR EXSR C0042
1019.00 C* set default value – Category Code – Cost Center 01
1020.00 CSR MOVE #RADJ $1MCU
1021.00 C*
1022.00 C* scrubs and edits – Category Code – Cost Center 01
1023.00 C*
1024.00 CSR MOVELSFRP01 $1RP01
1025.00 CSR END
1026.00 C* scrubs and edits – Category Code – Cost Center 01
1027.00 C* set default value – Category Code – Cost Center 01
1028.00 C*
1029.00 CSR $1RP01 IFEQ *BLANK
1030.00 CSR D$RP01 IFNE *BLANK
1031.00 CSR MOVEAD$RP01 040
1032.00 CSR MOVEA@40 $1RP01
1033.00 CSR @40,1 IFEQ ‘‘
1034.00 CSR MOVE ‘‘ 040,1
1035.00 CSR Z-ADD2 #M
1036.00 CSR #M DOWLE40
1037.00 CSR IF$EQ ‘‘
1038.00 CSR MOVE ‘‘ 040,#M
1039.00 CSR END
1040.00 CSR ADD 1 #M
1041.00 CSR END
1042.00 CSR MOVEA@40,2 $1RP01
1043.00 CSR END
1044.00 CSR END
1045.00 CSR END
1046.00 C* edit allowed values – Category Code – Cost Center 01
1047.00 C*
1048.00 C*
1049.00 CSR A$RP01 IFNE *BLANK
1050.00 CSR MOVEA@RP01 040
1051.00 CSR MOVE *HIVAL @AV
1052.00 CSR EXSR C997
1053.00 C* ** Set default value – Category Code – Cost Center 01**
1054.00 CSR MOVE ‘’ $ERTST
1055.00 CSR MOVE *BLANK $WRK10 10
1056.00 CSR MOVE$1RP01 $WRK10
1057.00 CSR @AV,1 IFEQ *HIVAL
1058.00 CSR $WRK10 LOKUP$AV $1
1059.00 CSR *IN8l IFEQ ‘0’
1060.00 CSR MOVE ‘’ $ERTST
1061.00 CSR END
1062.00 CSR $ERTST IFEQ ‘1’
1063.00 CSR MOVE ‘’ $MK,07
1064.00 CSR  SETON             4393
1065.00 CSR  END
1066.00 CSR  END
1067.00 CSR  END
1068.00 CSR
1069.00 CSR  ** Edit upper and lower range – Category Code – Cost Center 01**
1070.00 CSR
1071.00 CSR  L@RP01  IFNE *BLANK  $ERTST
1072.00 CSR  MOVE ‘1’ $ERTST
1073.00 CSR  $1RP01  IFGE L@RP01
1074.00 CSR  $1RP01  ANDLEU@RP01
1075.00 CSR  MOVE ‘’ $ERTST
1076.00 CSR  END
1077.00 CSR  $ERTST  IFGE L@RP01
1078.00 CSR  MOVE ‘1’ $MK,07
1079.00 CSR  SETON             4393
1080.00 CSR  END
1081.00 CSR  END
1082.00 CSR
1083.00 CSR  ** Edit from descriptive titles – Category Code – Cost Center 01**
1084.00 CSR
1085.00 CSR  R@RP01  IFNE *BLANK
1086.00 CSR  CLEAR100050
1087.00 CSR  MOVE ‘’ $ERTST
1088.00 CSR  MOVEL@RP01  #USY
1089.00 CSR  MOVE R@RP01  #URT
1090.00 CSR  MOVE $1RP01  #UKY
1091.00 CSR  CALL ‘X0005 ’               81
1092.00 CSR
1093.00 CSR  PARM  10005U
1094.00 CSR  #UERR  IFEQ ‘1’
1095.00 CSR  MOVE ‘1’ $MK,09
1096.00 CSR  SETON             4393
1097.00 CSR  END
1098.00 CSR  END
1099.00 CSR
1100.00 CSR  ** Scrub and edit – Category Code – Cost Center 02**
1101.00 CSR
1102.00 CSR
1103.00 CSR  MOVELSFRP02 $1RP02
1104.00 CSR
1105.00 CSR  ** Set default value – Category Code – Cost Center 02**
1106.00 CSR
1107.00 CSR  $1RP02  IFEQ *BLANK
1108.00 CSR  D@RP02  IFNE *BLANK
1109.00 CSR  MOVEAD@RP02  @040
1110.00 CSR  MOVEA@40  $1RP02
1111.00 CSR  $@40,1  IFEQ ‘’ $M
1112.00 CSR  MOVE ‘’ $@40,1
1113.00 CSR  $2-ADD2  #M
1114.00 CSR  DOWLE@40
1115.00 CSR  $@40,#M  IFEQ ‘’ $M
1116.00 CSR  MOVE ‘’ $@40,#M
1117.00 CSR  END
1118.00 CSR  ADD 1 #M
1119.00 CSR  END
1120.00 CSR  MOVEA@40,2  $1RP02
1121.00 CSR  END
1122.00 CSR  END
1123.00 CSR  END
1124.00 CSR
1125.00 CSR  ** Edit allowed values – Category Code – Cost Center 02**
1126.00 CSR
1127.00 CSR  A@RP02  IFNE *BLANK
1128.00 CSR  MOVEA@RP02  @040
1129.00 CSR  MOVE *HIVAL  $AV
1130.00 CSR  EASR  C997
1131.00 CSR  ---- ----
1132.00 CSR  MOVE ‘’ $ERTST
1133.00 CSR  MOVE *BLANK  SWRK10 10
1134.00 CSR  MOVEL@RP02  SWRK10
1135.00 CSR  @AV,1  IFNE *HIVAL
1136.00 CSR  SWRK10  LOKUP@AV
1137.00 CSR  *IN01  IFEQ ‘0’
1138.00 CSR  MOVE ‘1’ $ERTST
1139.00 CSR  END
1140.00 CSR  $ERTST  IFEQ ‘1’
1141.00 CSR  MOVE ‘1’ $MK,07
1142.00 CSR  SETON             4493
1143.00 CSR  END
1144.00 CSR  END
1145.00 CSR  END
Advanced Programming Concepts and Skills

1146.00 C* Edit upper and lower range – Category Code – Cost Center 02
1147.00 C* 1148.00 C* 1149.00 CSR L8RP02 IFNE 'BLANK
1150.00 CSR MOVE '1' $ERTST
1151.00 CSR $1RP02 IFGE $ERTST
1152.00 CSR $1RP02 ANDLUE@RP02
1153.00 CSR MOVE $ERTST
1154.00 CSR END
1155.00 CSR $ERTST IFEQ '1'
1156.00 CSR MOVE '1'
1157.00 CSR SETON 4493
1158.00 CSR END
1159.00 cSR END
1160.00 C* Edit from descriptive titles – Category Code – Cost Center 02
1161.00 C* 1162.00 C* Edit upper and lower range – Category Code – Cost Center 02
1163.00 CSR R8RP02 IFNE * BLANK
1164.00 CSR CLEARI005U
1165.00 CSR MOVE ' ' $ERTST
1166.00 CSR MOVE @USY
1167.00 CSR MOVE #URY
1168.00 CSR CALL 'X0005 81
1169.00 C* 1170.00 CSR PARM I0005U
1171.00 CSR #UERR IFEQ '1'
1172.00 CSR SETON 4493
1173.00 CSR END
1174.00 CSR END
1175.00 CSR END
1176.00 CSR END
1177.00 CSR END
1178.00 C* 1179.00 C* If no errors, update user index.
1180.00 C* 1181.00 C* 1182.00 CSR *IN93 IFEQ '0'
1183.00 C* 1184.00 CSR $1MCU ANDEQSHMCU and same CC
1185.00 CSR CALL 'X00IDX'
1186.00 CSR PARM $1IDX Index Name
1187.00 CSR PARM 'C' PSACTN Action Code
1188.00 CSR PARM PSRULE Function Rule
1189.00 CSR PARM PSKEYL Key Length
1190.00 CSR PARM PSKY Key
1191.00 CSR PARM PSRECL Recd Length
1192.00 CSR PARM PSREC Record
1193.00 CSR PARM PSSTS Status
1194.00 CSR ELSE
1195.00 CSR CALL 'X00IDX'
1196.00 CSR PARM $1IDX Idx Name/Lib
1197.00 CSR PARM 'I' PSACTN Action Code
1198.00 CSR PARM PSRULE Action Rule
1199.00 CSR PARM PSKEYL Key Length
1200.00 CSR PARM PSKY Key Fields
1201.00 CSR PARM PSRECL Entry Length
1202.00 CSR PARM PSREC Entry
1203.00 CSR PARM PSSTS Error Status
1204.00 CSR ELSE
1205.00 CSR CALL 'X00IDX'
1206.00 CSR PARM $1IDX Idx Name/Lib
1207.00 CSR PARM 'A' PSACTN Action Code
1208.00 CSR PARM PSRULE Action Rule
1209.00 CSR PARM PSKEYL Key Length
1210.00 CSR PARM PSKY Key Fields
1211.00 CSR PARM PSRECL Entry Length
1212.00 CSR PARM PSREC Entry
1213.00 CSR PARM PSSTS Error Status
1214.00 CSR ELSE
1215.00 CSR PARM $1IDX Idx Name/Lib
1216.00 CSR PARM 'E' PSACTN Action Code
1217.00 CSR PARM PSRULE Action Rule
1218.00 CSR PARM PSKEYL Key Length
1219.00 CSR PARM PSKY Key Fields
1220.00 CSR PARM PSRECL Entry Length
1221.00 CSR PARM PSREC Entry
1222.00 CSR PARM PSSTS Error Status

Loading key length, record length, key and record for a change or addition
Check if record exists
Call to User Index to change a record
Inquire on a record
Check if record exists
Add a record to the User Index
1231.00  C*   MOVE '1'  $WRT
1232.00  CSR   END
1233.00  CSR   END
1234.00  CSR   END
1235.00  CSR   END
1236.00  C*   *IN93  IFEQ '1'
1237.00  CSR   $FRNO  ANDEQ*ZERO
1238.00  CSR   Z-ADD1 $FRNO
1239.00  CSR   END
1240.00  C*   END
1241.00  CSR   END
1242.00  CSR   END
1243.00  C*   If errors, set subfile next change flag.
1244.00  C*   *IN93  IFEQ '1'
1245.00  CSR   SETON 32
1246.00  CSR   END
1247.00  C*   Update all subfile records read.
1248.00  C*   *IN93  IFEQ '1'
1249.00  CSR   SETON 32
1250.00  C*   Read next subfile record.
1251.00  C*   ADD 1 $IX
1252.00  CSR   END
1253.00  CSR   END
1254.00  CSR   END
1255.00  C*   If error detected on a add, Change Action Code to 'C'
1256.00  C*   *IN93  IFEQ '1'
1257.00  CSR   $WRT  ANDEQ'1'
1258.00  CSR   MOVE 'C' ACTION
1259.00  CSR   END005 ENDSR
1260.00  C***************************************************************
1261.00  C***************************************************************
1262.00  C***************************************************************
1263.00  C***************************************************************
1264.00  C***************************************************************
1265.00  C***************************************************************
1266.00  C***************************************************************
1267.00  C***************************************************************
1268.00  C***************************************************************
1269.00  CSR   END005 ENDSR
1270.00  C***************************************************************
1271.00  C***************************************************************
1272.00  C***************************************************************
1273.00  C***************************************************************
1274.00  C***************************************************************
1275.00  C***************************************************************
1276.00  C***************************************************************
1277.00  C***************************************************************
1278.00  C***************************************************************
1279.00  C***************************************************************
1280.00  C***************************************************************
1281.00  C***************************************************************
1282.00  C***************************************************************
1283.00  C***************************************************************
1284.00  C***************************************************************
1285.00  C***************************************************************
1286.00  C***************************************************************
1287.00  C***************************************************************
1288.00  C***************************************************************
1289.00  C***************************************************************
1290.00  C***************************************************************
1291.00  C***************************************************************
1292.00  CSR   *IN23  IFEQ '1'
1293.00  CSR   _ADD$1KEY2  PSKEYL
1294.00  CSR   _ADD$1RECL  PSRECL
1295.00  CSR   CALL 'X00IDX'
1296.00  CSR   ---- ----
1297.00  CSR   ---- ----
1298.00  CSR   FARM  $1IDX  Idx Name/Lib
1299.00  CSR   FARM  'D'  PSACTN  Action
1300.00  CSR   FARM  'EQ'  PSRULE  Action Rule
1301.00  CSR   FARM  PSKEYL  Key Length
1302.00  CSR   FARM  PSKY  Key Fields
1303.00  CSR   FARM  PSRECL  Entry Length
1304.00  CSR   FARM  PSREC  Entry
1305.00  CSR   FARM  PSSTS  Error Status
1306.00  CSR   END
1307.00  C*   Clear data field for next transaction
1308.00  C*   MOVE #FCLR @@AID
1309.00  CSR   EXSR S001
1310.00  CSR   ---- ----
1311.00  CSR   EXSR S001
1312.00  C*   ---- ----
1313.00  CSR   END010 ENDSR
A8.1 (8/97)

1314.00 C****************************************************************
1315.00 C SUBROUTINE S998 – Load dictionary parameters.
1316.00 C****************************************************************
1317.00 CSR S998 BEGSR
1318.00 C* Dictionary parameters for – Description Ol
1319.00 CSR MOVE *BLANK FRDTAI
1320.00 CSR MOVEL'DL01' FRDTAI
1321.00 CSR CALL 'X9800E' 81
1322.00 CSR PARM 19800E
1323.00 CSR FRERR IFEQ '0'
1324.00 CSR MOVE FRDSCR B@DL01 40
1325.00 CSR MOVE FRDTAT T@DL01 1
1326.00 CSR MOVE FREC E@DL01 1
1327.00 CSR MOVE ERDTAS C@DL01 40
1328.00 CSR MOVE FRDTAD G@DL01 10
1329.00 CSR MOVE MOVELFRSY S@DL01 4
1330.00 CSR MOVE FRRT R@DL01 2
1331.00 CSR MOVE FRDVAL D@DL01 40
1332.00 CSR MOVE FRVAL A@DL01 40
1333.00 CSR MOVE FRLVAL L@DL01 40
1334.00 CSR MOVE FRDTAD G@DL01 10
1335.00 CSR MOVE MOVELFRSY S@DL01 4
1336.00 CSR MOVE MOVELFRSY S@DL01 4
1337.00 CSR MOVE MOVELFRSY S@DL01 4
1338.00 CSR MOVE MOVELFRSY S@DL01 4
1339.00 CSR MOVE MOVELFRSY S@DL01 4
1340.00 CSR MOVE MOVELFRSY S@DL01 4
1341.00 CSR MOVE MOVELFRSY S@DL01 4
1342.00 CSR MOVE MOVELFRSY S@DL01 4
1343.00 CSR MOVE MOVELFRSY S@DL01 4
1344.00 CSR MOVE MOVELFRSY S@DL01 4
1345.00 CSR MOVE MOVELFRSY S@DL01 4
1346.00 CSR MOVE MOVELFRSY S@DL01 4
1347.00 CSR MOVE MOVELFRSY S@DL01 4
1348.00 CSR DO #A
1349.00 CSR MULT 10 @#DL01
1350.00 CSR END
1351.00 CSR END
1352.00 C* Dictionary parameters for – Cost Center
1353.00 C* Dictionary parameters for – Category Code – Cost Center 01
1354.00 C* Dictionary parameters for – Cost Center
1355.00 C* Dictionary parameters for – Category Code – Cost Center 01
1356.00 CSR MOVE *BLANK FRDTAI
1357.00 CSR MOVE 'MCU' FRDTAI
1358.00 CSR CALL 'X9800E' 81
1359.00 CSR PARM 19800E
1360.00 CSR FRERR IFEQ '0'
1361.00 CSR MOVE FRDSCR B@MCU 40
1362.00 CSR MOVE FRDTAT T@MCU 1
1363.00 CSR MOVE FREC E@MCU 1
1364.00 CSR MOVE ERDTAS C@MCU 40
1365.00 CSR MOVE FRDTAD G@MCU 10
1366.00 CSR MOVE MOVELFRSY S@MCU 4
1367.00 CSR MOVE MOVELFRSY S@MCU 4
1368.00 CSR MOVE MOVELFRSY S@MCU 4
1369.00 CSR MOVE MOVELFRSY S@MCU 4
1370.00 CSR MOVE MOVELFRSY S@MCU 4
1371.00 CSR MOVE MOVELFRSY S@MCU 4
1372.00 CSR MOVE MOVELFRSY S@MCU 4
1373.00 CSR MOVE MOVELFRSY S@MCU 4
1374.00 CSR MOVE MOVELFRSY S@MCU 4
1375.00 CSR MOVE MOVELFRSY S@MCU 4
1376.00 CSR MOVE MOVELFRSY S@MCU 4
1377.00 CSR Z-ADDI #@MCU 110
1378.00 CSR MOVE F@MCU #A
1379.00 CSR DO #A
1380.00 CSR MULT 10 @#MCU
1381.00 CSR END
1382.00 CSR END
1383.00 C* Dictionary parameters for – Category Code – Cost Center 01
1384.00 C* Dictionary parameters for – Category Code – Cost Center 01
1385.00 CSR MOVE *BLANK FRDTAI
1386.00 CSR MOVE 'RP01' FRDTAI
1387.00 CSR CALL 'X9800E' 81
1388.00 CSR PARM 19800E
1389.00 CSR FRERR IFEQ '0'
1390.00 CSR MOVE FRDSCR B@RP01 40
1391.00 CSR MOVE FRDTAT T@RP01 1
1392.00 CSR MOVE FREC E@RP01 1
1393.00 CSR MOVE ERDTAS C@RP01 40
1394.00 CSR MOVE FRDTAD G@RP01 10
1395.00 CSR MOVE MOVELFRSY S@RP01 4
<table>
<thead>
<tr>
<th>Line</th>
<th>Description</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>1396</td>
<td>CSR MOVE FRDTAS</td>
<td>C@RP01 40</td>
</tr>
<tr>
<td>1397</td>
<td>CSR MOVE FRDTAD</td>
<td>G@RP01 10</td>
</tr>
<tr>
<td>1398</td>
<td>CSR MOVE FRCDEC</td>
<td>A@RP01 1</td>
</tr>
<tr>
<td>1399</td>
<td>CSR MOVELFRSY</td>
<td>S@RP01 4</td>
</tr>
<tr>
<td>1400</td>
<td>CSR MOVE FRRT</td>
<td>R@RP01 2</td>
</tr>
<tr>
<td>1401</td>
<td>CSR MOVE FRVAL</td>
<td>D@RP01 40</td>
</tr>
<tr>
<td>1402</td>
<td>CSR MOVE FRVAL</td>
<td>A@RP01 40</td>
</tr>
<tr>
<td>1403</td>
<td>CSR MOVE FRLVAL</td>
<td>L@RP01 40</td>
</tr>
<tr>
<td>1404</td>
<td>CSR MOVE FRLVAL</td>
<td>U@RP01 40</td>
</tr>
<tr>
<td>1405</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP01 30</td>
</tr>
<tr>
<td>1406</td>
<td>CSR MOVE FRNL</td>
<td>J@RP01 1</td>
</tr>
<tr>
<td>1407</td>
<td>CSR MOVE FRNL</td>
<td>N@RP01 20</td>
</tr>
<tr>
<td>1408</td>
<td>CSR Z-ADD1</td>
<td>$@RP01 110</td>
</tr>
<tr>
<td>1409</td>
<td>CSR MOVE F@RP01</td>
<td>#A</td>
</tr>
<tr>
<td>1410</td>
<td>CSR DO</td>
<td>#A</td>
</tr>
<tr>
<td>1411</td>
<td>CSR CALL</td>
<td>'X9800E' 81</td>
</tr>
<tr>
<td>1412</td>
<td>CSR PARM</td>
<td>I9800E</td>
</tr>
<tr>
<td>1413</td>
<td>CSR Calls</td>
<td>'X9800E'</td>
</tr>
<tr>
<td>1414</td>
<td>CSR PARM</td>
<td>I9800E</td>
</tr>
<tr>
<td>1415</td>
<td>CSR FRERR</td>
<td>IFEQ '0'</td>
</tr>
<tr>
<td>1416</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1417</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1418</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1419</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1420</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1421</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1422</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1423</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1424</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1425</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1426</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1427</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1428</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1429</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1430</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1431</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1432</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1433</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1434</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1435</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1436</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1437</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1438</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1439</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1440</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1441</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1442</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1443</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1444</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1445</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1446</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1447</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1448</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1449</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1450</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1451</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1452</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1453</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1454</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1455</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1456</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1457</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1458</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1459</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1460</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1461</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1462</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1463</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1464</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1465</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1466</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1467</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1468</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1469</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1470</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1471</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1472</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1473</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
<tr>
<td>1474</td>
<td>CSR MOVE FREDWR</td>
<td>W@RP02 30</td>
</tr>
</tbody>
</table>
Create or clear the Demonstration User Index

Check to see if User Index already exists

Create User Index if User Index did not already exist

Check error status parameter to see if User Index exists

Delete all records from User Index

Load key length, record length, and key to clear User Index if it already exists
1557.00 C*   Test for auto inquiry function.
1558.00 C*
1559.00 CSR SAUTO IFNE *BLANK
1560.00 CSR MOVE '1' $AUTO
1561.00 CSR END
1562.00 C*---------------------------------------------
1563.00 C*
1564.00 C*   Load video screen text.
1565.00 C*
1566.00 CSR MOVEL@FILE PSKEY       10
1567.00 CSR Z-ADD006 PSVTX#  30
1568.00 C/COPY JDECPY,C00SC
1569.00 C*---------------------------------------------
1570.00 C*
1571.00 C*   Load error messages array.
1572.00 C*
1573.00 CSR MOVE '0001' EMK,01 Inv Action
1574.00 CSR MOVE '0002' EMK,02 Inv Key
1575.00 CSR MOVE '0003' EMK,03 Inv Blanks
1576.00 CSR MOVE '0004' EMK,04 Inv Date
1577.00 CSR MOVE '0005' EMK,05 Inv Next Nbr
1578.00 CSR MOVE '0007' EMK,06 In Use
1579.00 CSR MOVE '0025' EMK,07 Inv Values
1580.00 CSR MOVE '0026' EMK,08 Inv MCU
1581.00 CSR MOVE '0027' EMK,09 Inv Desc Ttl
1582.00 C*---------------------------------------------
1583.00 C*
1584.00 C*   Load invalid action code array.
1585.00 C*
1586.00 CSR MOVEA'     ' @NAC
1587.00 C*---------------------------------------------
1588.00 C*
1589.00 C*   Initialize subfile display.
1590.00 C*
1591.00 CSR Z-ADD0   I1
1592.00 CSR Z-ADD15 $PGSZ  30
1593.00 CSR DO $PGSZ
1594.00 CSR ADD  1   I1
1595.00 CSR MOVEA'IN SHIN
1596.00 CSR WRITEINDEXS  99
1597.00 CSR END
1598.00 CSR Z-ADD1   $SVI1
1599.00 C*---------------------------------------------
1600.00 C*
1601.00 C*   Load system date.
1602.00 C*
1603.00 CSR TIME $WRK12  120
1604.00 CSR MOVE $WRK12 $$EDT  60
1605.00 C*---------------------------------------------
1606.00 CSR ENDSR
Exercises
See the exercises for this chapter.
File Servers

About File Servers

File servers (sometimes called I/O servers) allow you to enhance the processing time of your program. In addition, they ease the maintenance of your programs by making your system more modular. There should be no reason to bypass the use of a server. Eventually, every program should perform database functions using either a file server or a functional server. Note that all logical files are accessed through servers by their based-on file. Embedded in one server, there may be many access paths available.
What is a File Server?

A file server, or I/O server, is a server that performs all RPG database operation codes.

This type of server has no effect on program logic, but it isolates the actual database from the application program. Once you implement a file server into a program, the file specification is no longer required.

Types of File Servers

There are three types of file servers you can use:

- **XS** Input–Only/Caching Servers

  They should be used when you would otherwise use a simple CHAIN operation for input only. You may request descriptions only, or the entire record. They provide caching logic to decrease physical I/O for duplicate requests.

- **XF** Input/Output File Servers

  They will allow you to replace all RPG database operation codes for a given file with program calls. They can read, chain, setll, and so forth, to a file.

- **X** Special Scrub & Edit Servers

  They can accept the cost center or account numbers in any valid data entry or file format, convert them to any format, validate the existence of the master record, and optionally pass the master record back to the calling program.
What are the Advantages of Using a File Server?

The advantages of using a file server are:

- Minimizes maintenance of your software
- The ability to change a physical file without having to make changes to application programs that use the file, or even having to recompile them
- Using versions in future releases to allow programs from a previous release to run against a changed database
- The transition from an old database to a new database will be smoother. Instead of applying all new programs, you will only have to apply a new set of file servers.
- Ability to implement one file server at a time without affecting the rest of your system

What are the Disadvantages of Using a File Server?

The disadvantages of using a file server are:

- A file server is minutely slower because you are performing an external call to the server from your program.
- File server programs tend to be large.

File servers are designed to perform all database functions that can be performed directly.
How Does a File Server Function?

A file server performs all the interfaces between a program and file. First you will load the control parameters, which contain information about the record you are retrieving. The file server converts the control parameters and retrieves a record back to the program.

If you plan on using any of the file server programs and you are asking them to return the database record, you must use the record image /COPY member that the corresponding I/O server uses. The /COPY member has the following naming convention:

I(file name) (release level).

For example: The copy member for the F0101 record image should appear as:

I/COPY JDECPY, I010171

Some technical file servers (X9800E, X0005) have a /COPY member with the naming convention:

I(file name)(special character)

I/COPY JDECPY, I0005U

A file server is called with two parameters:

For example: CALL ‘XF0101’ 81

- - - - - - - -

PARM PS@@1
PARM I0101

<table>
<thead>
<tr>
<th>PARM</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>PS@@1</td>
<td>Contains all of the control parameters. It is contained in copy module I00XFSRV, and it is common to all file servers.</td>
</tr>
<tr>
<td>I(file name)</td>
<td>Contains the record image for updates and writes specific for each I/O server. It is an exact duplicate of the record image. It is contained in the copy module I(file name) (release level).</td>
</tr>
</tbody>
</table>
What Are Control Parameters?

The parameter PS@@1 contains all the control parameters for the file server. All control parameters, except the format name, are cleared every time the server returns control to the calling program. You must set the parameter values every time the server is called unless you are satisfied with the default values.

<table>
<thead>
<tr>
<th>PARM (Length)</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>@@ACCS (1)</td>
<td>The type of access to the file. The valid values are K for Keyed access (default), R for relative record number access and S for sequential access (DREAM Writer).</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>@@OPER (10)</th>
<th>The operation to be performed to the file. The valid values are presented below:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operation</td>
<td>Description</td>
</tr>
<tr>
<td>CHAIN</td>
<td>Chain by key list or RRN</td>
</tr>
<tr>
<td>CLOSE</td>
<td>Close the access path</td>
</tr>
<tr>
<td>DELET</td>
<td>Delete current record or by key or RRN</td>
</tr>
<tr>
<td>EXIST</td>
<td>Test existence of record by key</td>
</tr>
<tr>
<td>OPEN</td>
<td>Open access path (optional)</td>
</tr>
<tr>
<td>READ</td>
<td>Read next record</td>
</tr>
<tr>
<td>READE</td>
<td>Read next equal key</td>
</tr>
<tr>
<td>READP</td>
<td>Read previous record</td>
</tr>
<tr>
<td>REDPE</td>
<td>Read previous equal key</td>
</tr>
<tr>
<td>SETGT</td>
<td>Set greater than key</td>
</tr>
<tr>
<td>SETHV</td>
<td>Set greater than with *HIVAL</td>
</tr>
<tr>
<td>SETLL</td>
<td>Set lower limit by key</td>
</tr>
<tr>
<td>SETLV</td>
<td>Set lower limit with *LOVAL</td>
</tr>
<tr>
<td>UPDAT</td>
<td>Update locked record</td>
</tr>
<tr>
<td>UPDATC</td>
<td>Update current record</td>
</tr>
<tr>
<td>WRITE</td>
<td>Write new record</td>
</tr>
<tr>
<td>UNLCK</td>
<td>Unlock current record</td>
</tr>
</tbody>
</table>

<p>| @@LOCK (1) | Whether you do or do not want to lock the record. The valid values are Y to lock the record (default) or N to not lock the record. |
| Note:      | This parameter is only valid for chain and read operations, and is ignored for all other operations. You should set it to N when reading records not to be updated. |</p>
<table>
<thead>
<tr>
<th>PARM (Length)</th>
<th>Explanation</th>
</tr>
</thead>
</table>
| @@CHGR (1)   | Servers allow records to be read without lock and then be updated (UPDATC). In this situation, the record will be re-read before it is updated and if it has changed since it was last read, action will have to be taken. This parameter determines what that action will be. The valid values are:  
  N– Do not update the record. A return code (RC) is returned and it comes up to the program to determine what action to take. (default)  
  O– Overlay the changed record with the values you are currently updating. This will cause the changes made by the other user to be lost.  
  W– Call the Changed Record Window (P0000U) that will prompt you for what action to take. Use this option with interactive programs only. |
| @@KLST (10)  | The key list that will be used for access. The calling program does not specify a logical file so that the application program is isolated from any database changes. A value must be specified unless you are accessing the file by relative record number or sequentially ( @@ACC = R or S).  
  Note: The server maintains status information for each access path, so multiple paths can be accessed through the server in one program. The @@KLST parameter determines which access path is affected by the current call to the server. |
| @@KNUM (5,0) | Specifies how many key fields in the list will be used for the current operation. This allows you to perform a read equal by a partial key. The valid values are 1 through the number of fields in the key, and blank for operations not requiring a key. |
| @@FMT (10)   | Specifies the release level the program is expecting. This field does not get cleared upon returning from the server, so it can be set once in S999. |
| @@#RRN (9,0) | The relative record number for RRN access. |
| I (file name) | Record image for updates and writes. This parameter is optional for OPEN, CLOSE, DELET, SETHV, SETLV, and UNLCK operations. |

- Access paths are opened automatically when the first operation is performed. Therefore, it is not necessary to call the server with the OPEN operation.  
- A server normally remains active as long as the calling program is active. If you know you will need a server for only a limited period of time and do not want it taking up space in the PAG, you can call the server the @@OPER parameter blank, this causes the server to return and end.
What Are Returned Parameters?

When the file server returns the record to the program, there are several parameters associated with it.

<table>
<thead>
<tr>
<th>PARM (Length)</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>@@IOR(3)</td>
<td>The I/O return code. The possible values are: blank – No errors NF – Record not found NE – Not equal for a READE operation EOF – End of file EQ – Equal for a SETLL operation BOF – Beginning of file RL – Record Locked, could not read RC – Record changed YES – Record found NO – Record not found ERR – Error, check error fields for explanation</td>
</tr>
<tr>
<td>@@ERR (10)</td>
<td>Short description of the cause of the problem (invalid, reclock, error, required, deleted, chgrec).</td>
</tr>
<tr>
<td>@@ERRS (10)</td>
<td>The subject causing the error. The value could be a parameter (KLST), an operation (OPEN), or a file name (Fxxxx). Used in combination with @@ERR gives a good idea of what happened. The application program will generally only use @@IOR. @@ERR and @@ERRS are most useful for debugging purposes.</td>
</tr>
<tr>
<td>@@#RRN (9,0)</td>
<td>Returns the relative record number of the record just read (both input and output).</td>
</tr>
<tr>
<td>I (filename)</td>
<td>Returns an exact duplicate of the record image (both input and output).</td>
</tr>
</tbody>
</table>
Implementing a File Server

To implement a file server

The following are generally the steps needed to set up a file server in a program. Some programs may differ.

1. Remove F-spec line for file being accessed through the server, and replace it with a comment mentioning access through the server.
2. Add clear statement in S999 (CLEAR PS@@1). You can optionally set @@FMT to “A71” so it does not have to be set on every call.
3. Copy in I-spec copy module I00XFSRV.
4. Copy in I-spec copy module for the required server, following the naming convention: I(file name) (release level). For example, IO10171.
5. Code call to server for each database access. Naming convention for server is X(file name). For example, XF0101 for F0101 and any of its logicals.
   - Load control parameters
   - Load record image if a write or update
   - Call the server
   - Check the return code
6. Remove any open statements and key lists for this file from S999.
7. Remove any output specifications dealing with EXCPT unlock statements at the bottom of the program. The server will handle all of the unlock and lock operations.

When reading sequentially (@@ACCS = S) through the physical file or through a DREAM Writer based-on file that is overridden to the physical, some operations are not available. Do not use: CHAIN, EXIST, READE, REDPE, UPDATC, SETLL, SETGT, SETHV, SETLV. Since UPDATC is not available and you are going to update a record, you need to read with lock.

If the file you are accessing though the server is the DREAM Writer based on file, the Open Query Options on the DREAM Writer Additional Parameters screen need to be changed. Change all of the “Open for xxxxx” parameters to “Y”.

Searching for Key Lists

When converting programs to use the file servers, make note of what logical files are being accessed, and what mode (update or input) and what each of the defined key lists for those access paths represent.
**To search for Key Lists**

1. Look up the corresponding server key list name using P93KL (fast path, KL).
2. Search for the format name for files that are accessed in the program.
3. Replace each instance of file access code with a call to the server with the correct parameters.

<table>
<thead>
<tr>
<th>Key</th>
<th>Access</th>
<th>S Key</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABKY01</td>
<td>F0101LA</td>
<td>AN8</td>
<td>Address Number</td>
</tr>
<tr>
<td>ABKY02</td>
<td>F0101LB</td>
<td>DC</td>
<td>Description – Compressed</td>
</tr>
<tr>
<td>ABKY03</td>
<td>F0101LC</td>
<td>PH1</td>
<td>Phone Number</td>
</tr>
<tr>
<td>ABKY04</td>
<td>F0101LD</td>
<td>PA8</td>
<td>Parent Number</td>
</tr>
<tr>
<td>ABKY05</td>
<td>F0101LE</td>
<td>* AN8</td>
<td>Address Number</td>
</tr>
</tbody>
</table>

**Selection or command**

```plaintext
=> KL
```

**Thur, Jan 6, 1994**

---

G92                                      J.D. Edwards & Company
Computer Assisted Design (CAD)

... SYSTEM DESIGN TOOLS ... PROGRAM DESIGN TOOLS
2. Software Versions Repository   14. Processing Options
3. Menus                           15. Help Instructions
4. Data Dictionary                 93KL   File Server Key Lists
5. Model Relations                 Skip To Key Name...
6. CASE Profiles                   Server Name...
7. Function Key D                  **KL**
8. Vocabulary Overview

Selection or command

```plaintext
=> KL
```

**Thur, Jan 6, 1994**

**ABKY05 F0101LE** * AN8 Address Number...
Tips when Using File Servers

The following tips can help when using file servers:

- When converting a program to use the file servers, always set the @@LOCK parameter to “N” when reading records through an access path that the program uses to open for input only.

  The reason for this is that all access paths are open for update in the server. This can cause record lock problems when a program opens multiple paths into the same file. Correct use of the @@LOCK parameter solves these problems.

- Some programs may be doing a CHAIN or EXCPT to unlock a record. Instead of replacing it with a CHAIN through the server, take advantage of the UNLCK operation. Performing an UNLCK on a file that does not have a record locked does not produce an error.

- Some programs perform a SETLL to validate that a record exists. The new operation EXIST is provided to handle this function. It returns a YES or NO in return code (@@IOR).

- There is only one instance in which a particular file server is active in your job at one time, so if one program calls another program that accesses the file through the same access path, they are actually sharing the same open data path. If it is possible that a call to another program could relocate a file pointer that could mess up the program, it would be a good idea to save the keys and reset the pointer (CHAIN or SETLL) upon returning.
### File Server Examples

<table>
<thead>
<tr>
<th>Seq No.</th>
<th>C*</th>
<th>Mod DC*e</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.00</td>
<td>C*</td>
<td>09.11.92</td>
</tr>
<tr>
<td>2.00</td>
<td>C*</td>
<td>10.11.92</td>
</tr>
<tr>
<td>3.00</td>
<td>C*</td>
<td>10.11.92</td>
</tr>
<tr>
<td>4.00</td>
<td>C*</td>
<td>10.11.92</td>
</tr>
<tr>
<td>5.00</td>
<td>C*</td>
<td>10.11.92</td>
</tr>
<tr>
<td>6.00</td>
<td>C*</td>
<td>10.11.92</td>
</tr>
<tr>
<td>7.00</td>
<td>C*</td>
<td>10.11.92</td>
</tr>
<tr>
<td>8.00</td>
<td>C*</td>
<td>1.1 File Server Calls: 10.11.92</td>
</tr>
<tr>
<td>9.00</td>
<td>C*</td>
<td>1.1.1 Chain: 10.11.92</td>
</tr>
<tr>
<td>10.00</td>
<td>C*</td>
<td>1.1.2 Close: 10.11.92</td>
</tr>
<tr>
<td>11.00</td>
<td>C*</td>
<td>1.1.3 Delete: 10.11.92</td>
</tr>
<tr>
<td>12.00</td>
<td>C*</td>
<td>1.1.4 Existence Test: 10.11.92</td>
</tr>
<tr>
<td>13.00</td>
<td>C*</td>
<td>1.1.5 Open: 10.11.92</td>
</tr>
<tr>
<td>14.00</td>
<td>C*</td>
<td>1.1.6 Read: 10.11.92</td>
</tr>
<tr>
<td>15.00</td>
<td>C*</td>
<td>1.1.7 Read Equal: 10.11.92</td>
</tr>
<tr>
<td>16.00</td>
<td>C*</td>
<td>1.1.8 Read Previous: 10.11.92</td>
</tr>
<tr>
<td>17.00</td>
<td>C*</td>
<td>1.1.9 Read Previous Equal: 10.11.92</td>
</tr>
<tr>
<td>18.00</td>
<td>C*</td>
<td>1.1.10 Set Greater Than: 10.11.92</td>
</tr>
<tr>
<td>19.00</td>
<td>C*</td>
<td>1.1.11 Set Lover Level: 10.11.92</td>
</tr>
<tr>
<td>20.00</td>
<td>C*</td>
<td>1.1.12 Update: 10.11.92</td>
</tr>
<tr>
<td>21.00</td>
<td>C*</td>
<td>1.1.13 Write: 10.11.92</td>
</tr>
<tr>
<td>22.00</td>
<td>C*</td>
<td>1.1.14 Unlock: 10.11.92</td>
</tr>
<tr>
<td>23.00</td>
<td>C*</td>
<td>1.1.15 XREF: 10.11.92</td>
</tr>
<tr>
<td>24.00</td>
<td>C*</td>
<td>2.1.1 XREF: 10.11.92</td>
</tr>
</tbody>
</table>

1. **Determine from P93KL what the key list name is for the access path being used; this name is moved to the @@KLST.**

2. **If the operation uses a key list, determine how many keys the key list represents; this number is Z-ADDed to @@KNUM.**

3. **The format is the release level (A61) and can be moved to @@FMT in S999 once for the rest of the calls.**
80.00 CSR CALL 'XF0101'  
81.00 C*  
82.00 CSR PARM PS001 10.11.92  
83.00 CSR PARM I0101 10.11.92  
84.00 CSR @ IOR COMP 'RL' 99 10.11.92  
85.00 C* 10.11.92  
86.00 C* Old Code:  
87.00 CSR ABKYOB DELET10101B 8399 10.11.92  
88.00 C* 10.11.92  
89.00 C* New Code:  
90.00 CSR MOVEL'A61' @MFMT 10.11.92  
91.00 CSR MOVEL'ABY02' @KLST 10.11.92  
92.00 CSR MOVEL'DELET' @OPER 10.11.92  
93.00 CSR Z-ADD3 @KNUM 10.11.92  
94.00 CSR CALL 'XF0101'  
95.00 C* 10.11.92  
96.00 CSR PARM PS001 10.11.92  
97.00 CSR PARM I0101 10.11.92  
98.00 CSR @ IOR COMP 'RL' 99 10.11.92  
99.00 CSR @ IOR COMP 'NF' 83 10.11.92  
100.00 C* 10.11.92  
101.00 C* 1.1.4 Existence Test:  
102.00 C* Old Code:  
103.00 CSR ABKY02 SETLLI0101D 9982 10.11.92  
104.00 C* 1.1.5 Open:  
105.00 CSR OPEN F0006 10.11.92  
106.00 C* 1.1.6 Read:  
107.00 C* Old Code:  
108.00 CSR READ I0901A 9982 10.11.92  
109.00 C* New Code:  
110.00 CSR MOVEL'A61' @MFMT 10.11.92  
111.00 CSR MOVEL'MCY01' @KLST 10.11.92  
112.00 CSR MOVEL'OPEN' @OPER 10.11.92  
113.00 CSR CALL 'XF0901'  
114.00 C* 10.11.92  
115.00 CSR PARM PS001 10.11.92  
116.00 CSR @ IOR COMP 'YES' 82 10.11.92  
117.00 CSR @ IOR COMP 'ERR' 99 11.12.92  
118.00 C* 10.11.92  
119.00 C* 1.1.7 Read Equal:  
120.00 C* Old Code:  
121.00 CSR READ I0901A 9982 10.11.92  
122.00 C* New Code:  
123.00 CSR MOVEL'A61' @MFMT 10.11.92  
124.00 CSR MOVEL'GMKY01' @KLST 10.11.92  
125.00 CSR MOVEL'READ' @OPER 10.11.92  
126.00 CSR MOVE 'N' @LOCK 05.12.92  
127.00 CSR CALL 'XF0901'  
128.00 C* 10.11.92  
129.00 CSR PARM PS001 10.11.92  
130.00 CSR PARM I0901 10.11.92  
131.00 C* 10.11.92  
132.00 C* 10.11.92  
133.00 C* 10.11.92  
134.00 C* 10.11.92  
135.00 C* 10.11.92  
136.00 C* 10.11.92  
137.00 C* 10.11.92  
138.00 CSR READ I0901A 9982 10.11.92  
139.00 C* New Code:  
140.00 CSR MOVEL'A61' @MFMT 10.11.92  
141.00 CSR MOVEL'GMKY01' @KLST 10.11.92  
142.00 CSR MOVEL'READ' @OPER 10.11.92  
143.00 CSR MOVE 'N' @LOCK 05.12.92  
144.00 CSR CALL 'XF0901'  
145.00 C* 10.11.92  
146.00 C* 10.11.92  
147.00 CSR PARM PS001 10.11.92  
148.00 CSR PARM I0901 10.11.92  
149.00 CSR @ IOR COMP 'EOF' 82 10.11.92  
150.00 CSR @ IOR COMP 'RL' 99 10.11.92  
151.00 C* 10.11.92  
152.00 C* 10.11.92  
153.00 C* 1.1.7 Read Equal:  
154.00 C* 10.11.92  
155.00 C* 10.11.92  
156.00 CSR ABKY03 READ10101C 9987 10.11.92  
157.00 C* 10.11.92  
158.00 C* New Code:  
159.00 CSR MOVEL'A61' @MFMT 10.11.92  
160.00 CSR MOVEL'ABKY03' @KLST 10.11.92  
161.00 CSR MOVEL'MEAD' @OPER 10.11.92  
162.00 CSR MDVE 'N' @LOCK 05.12.92
<table>
<thead>
<tr>
<th>Line</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>163.00</td>
<td>CSR CALL 'XF0101'</td>
</tr>
<tr>
<td>164.00</td>
<td>CSR PARM PS001</td>
</tr>
<tr>
<td>165.00</td>
<td>CSR PARM 10101</td>
</tr>
<tr>
<td>166.00</td>
<td>CSR @81OR COMP 'NE' 87</td>
</tr>
<tr>
<td>167.00</td>
<td>CSR @81OR COMP 'RL' 99</td>
</tr>
<tr>
<td>168.00</td>
<td>CSR CALL 'XF0101' 10.11.92</td>
</tr>
<tr>
<td>169.00</td>
<td>CSR PARM PS001 10.11.92</td>
</tr>
<tr>
<td>170.00</td>
<td>CSR PARM I0101 10.11.92</td>
</tr>
<tr>
<td>171.00</td>
<td>CSR CALL 'XF0901' 10.11.92</td>
</tr>
<tr>
<td>172.00</td>
<td>CSR CALL 'XF0902' 10.11.92</td>
</tr>
<tr>
<td>173.00</td>
<td>CSR CALL 'XF0902' 10.11.92</td>
</tr>
<tr>
<td>174.00</td>
<td>CSR READPI0901B 9982 10.11.92</td>
</tr>
<tr>
<td>175.00</td>
<td>CSR MOVEL'A61' @@FMT 10.11.92</td>
</tr>
<tr>
<td>176.00</td>
<td>CSR MOVEL'GMKY02' @@KLST 10.11.92</td>
</tr>
<tr>
<td>177.00</td>
<td>CSR MOVEL'READP' @@OPER 10.11.92</td>
</tr>
<tr>
<td>178.00</td>
<td>CSR MOVEL 'N' @@LOCK 05.12.92</td>
</tr>
<tr>
<td>179.00</td>
<td>CSR CALL 'XF0901' 10.11.92</td>
</tr>
<tr>
<td>180.00</td>
<td>CSR CALL 'XF0902' 10.11.92</td>
</tr>
<tr>
<td>181.00</td>
<td>CSR CALL 'XF0902' 10.11.92</td>
</tr>
<tr>
<td>182.00</td>
<td>CSR CALL 'XF0902' 10.11.92</td>
</tr>
<tr>
<td>183.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>184.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>185.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>186.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>187.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>188.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>189.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>190.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>191.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>192.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>193.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>194.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>195.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>196.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>197.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>198.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>199.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>200.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>201.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>202.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>203.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>204.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>205.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>206.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>207.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>208.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>209.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>210.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>211.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>212.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>213.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>214.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>215.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>216.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>217.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>218.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>219.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>220.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>221.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>222.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>223.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>224.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>225.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>226.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>227.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>228.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>229.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>230.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>231.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>232.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>233.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>234.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>235.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>236.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>237.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>238.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>239.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>240.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>241.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>242.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>243.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>244.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
<tr>
<td>245.00</td>
<td>CSR CALL 'XF0903' 10.11.92</td>
</tr>
</tbody>
</table>
329.00  C'  
330.00  C*  New Code:  
331.00  CSR  MOVE L'AGL'  @FMT  
332.00  CSR  MOVE L'ABKLYO1'  @KLST  
333.00  CSR  MOVE L'UNLCK'  @KPER  
334.00  CSR  CALL 'XFO101'  
335.00  C*  ----  
336.00  CSR  PARM  PS01  
337.00  CSR  PARM  10D1  
338.00  CSR  COMP 'ERR'  @IQR  99  
339.00  C*  
340.00  C*  
341.00  C*  2.1.1  X09031:  
342.00  C*  
343.00  CSR  CALL 'X09031'  
344.00  C*  
345.00  CSR  PARM '2'  #CALC 1  
346.00  CSR  PARM  #CO  5  
347.00  CSR  PARM  #DG  60  
348.00  CSR  PARM  #PN  20  
349.00  CSR  PARM  #FY  20  
350.00  CSR  PARM  #CTY  20  
351.00  CSR  PRAM  #EDT  1  
352.00  CSR  PARM 'I'  #DDSY  1  
353.00  C*  
354.00  C*  
355.00  C*  
356.00  C*  2.2.1  X0901:  
357.00  C*  
358.00  CSR  CALL 'X0901'  
359.00  C*  
360.00  CSR  PARM 'I'  PSSYM  1  
361.00  CSR  PARM  RFAM  PSOMOD  1  
362.00  CSR  PARM 'I'  PSIMOD  1  
363.00  CSR  PARM RPOLBA  PSANI  29  
364.00  CSR  PARM *BLANK  PSMCU  12  
365.00  CSR  PARM *BLANK  PSBBJ  6  
366.00  CSR  PARM *BLANK  PSSUB  8  
367.00  CSR  PARM  PSERRM  4  
368.00  C*  
369.00  C*  
370.00  C*  
371.00  C*  2.3.1  X0006:  
372.00  C*  
373.00  CSR  CALL 'X0006'  
374.00  C*  
375.00  CSR  PARM 'I'  PSOMOD  1  output mode  
376.00  CSR  PARM  PSIMOD  1  input mode  
377.00  CSR  PARM SPFCU  PSMCU  12  cost center  
378.00  CSR  PARM  PSERRM  4  error flag  
379.00  CSR  PARM  100D6  F0006 record  
380.00  C*  
381.00  C*
## Commonly Used File Servers

The following is a list of commonly used file servers:

<table>
<thead>
<tr>
<th>File Server</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>X0005</td>
<td>User Defined Codes Server</td>
<td>Retrieve Only</td>
</tr>
<tr>
<td>X0006</td>
<td>Retrieve Cost Center Master</td>
<td>Retrieve &amp; Scrub</td>
</tr>
<tr>
<td>XF0006</td>
<td>Cost Center I/O</td>
<td>Add/Change/Delete</td>
</tr>
<tr>
<td>X0010</td>
<td>Automatic Next Numbering</td>
<td>Retrieve &amp; Increment</td>
</tr>
<tr>
<td>X9203</td>
<td>DD Alpha Description</td>
<td>Retrieve Only</td>
</tr>
<tr>
<td>X9800E</td>
<td>Data Dictionary Info</td>
<td>Editing Info</td>
</tr>
<tr>
<td>XF0101</td>
<td>Address Book I/O</td>
<td>Add/Change/Delete</td>
</tr>
<tr>
<td>XS0101LA</td>
<td>Address Book</td>
<td>Retrieve Only</td>
</tr>
<tr>
<td>X0901</td>
<td>Account Master</td>
<td>Retrieve &amp; Formats</td>
</tr>
<tr>
<td>XF0901</td>
<td>Account Master I/O</td>
<td>Add/Change/Delete</td>
</tr>
<tr>
<td>X41LOCN</td>
<td>Location Format</td>
<td></td>
</tr>
<tr>
<td>X41LOT</td>
<td>Lot Number Assignment</td>
<td></td>
</tr>
<tr>
<td>X41DUP</td>
<td>Lot Master Duplicate</td>
<td>Edits</td>
</tr>
<tr>
<td>X4101</td>
<td>Item Master</td>
<td>Retrieve &amp; Edit</td>
</tr>
<tr>
<td>X4108</td>
<td>Lot Master Update</td>
<td>Creates &amp; Updates</td>
</tr>
<tr>
<td>X4111</td>
<td>Write to Item Ledger</td>
<td>Writes Only</td>
</tr>
<tr>
<td>XF4111</td>
<td>CARDEX I/O</td>
<td>Retrieve Only</td>
</tr>
<tr>
<td>XF42119</td>
<td>Sales History I/O</td>
<td>Add/Change/Delete</td>
</tr>
<tr>
<td>XF42199</td>
<td>Sales Detail Ledger I/O</td>
<td>Add/Change/Delete</td>
</tr>
<tr>
<td>XF43199</td>
<td>Purchasing History I/O</td>
<td>Add/Change/Delete</td>
</tr>
</tbody>
</table>
Functional Servers

About Functional Servers

A functional server allows you to enhance the processing and maintenance of your application programs. Functional servers provide a central location for standard business rules about entering documents, such as vouchers, invoices, and journal entries. These business rules establish the following:

- Data Dictionary default values
- Field edits and valid values
- Error processing
- Relationships between fields or applications

To work with functional servers you should understand:

- What functional servers are
- What the advantages of using them are
- How to set up the business rules
- How they function
What Are Functional Servers?

A functional server is a server that performs all transaction validation and database updates.

This type of server is designed to relieve application programs from the burden of performing edit and update operations. This functionality is removed from the application program and placed into a server.

A functional server is a called program. The application program calling the server must tell the server what action is to be performed for every transaction. In turn, the functional server will return error messages, record error flags, and record update flags to the application program to use when determining the result of a call to a server.

Functional Servers have the following naming convention:

   XT (file name) (server version)

For example: The function server for the F0411 file should appear as:

   XT0411Z1
The following diagram depicts the flow of a typical program using a functional server:

![Diagram showing the flow of a typical program using a functional server]

**What Are the Advantages of Using a Functional Server?**

The following is a list of advantages in using a functional server:

- Minimizes maintenance and versioning of your software.
- Data editing routines and actual file updates can be isolated.
- Provide greater flexibility. Multiple programs can use the same functional server.
- The transition from an old database to a new database will be smoother. Instead of applying all new programs, you will only have to apply a new set of functional servers.
- Ability to implement one functional server at a time without affecting the rest of your system.
What Are the Disadvantages of Using a Functional Server?

The following is a list of disadvantages in using a functional server:

- A functional server is minutely slower because you are performing an external call to the server from your program.
- Functional server programs tend to be large.

Setting Up Business Rules for an Entry Program

To set up business rules for an entry program

1. Create a DREAM Writer version for a specific functional server program (for example, XT0411Z1 for voucher entry).
2. Set the processing options within the version according to your company requirements.
3. Specify the version you want the entry program to use in the processing options for that entry program.

You can have all your entry programs use the same DREAM Writer version (and thus, use the same rules) or you can set up different DREAM Writer versions. J.D. Edwards provides DREAM Writer version ZJDE0001 as the default functional server version for your entry programs.

Only the person responsible for system-wide setup should make changes to the functional server version. For more information about how to set up DREAM Writer versions, see the Technical Foundation Guide.

How Does a Functional Server Function?

When a functional server is called, an entire transaction is processed.

- Generally, once a functional server is called, it will receive the data entered by you and load it into a user space.
- It will then perform its functionality on the data.
- Finally, it will return the requested data back to the calling program via the user space. If any errors occur, they will be loaded into a user index.

Three interfaces are used to communicate with the functional server. They are:

- The call parameters
- The control fields within each user space line
- The error index

Functional Server Highlights

The following is a list of highlights of a functional server:

- Provides all editing for a transaction
- Provides field default values
- Provides all database updates
- Performs inquiry for an entire transaction
- Runs interactively or in batch
- Supports a multitude of user interfaces

Basic Accounting Transactions

In the Financial System there are five basic transactions:

- Journal Entries
- A/P Voucher Entry
- A/P Checks
- A/R Invoice Entry
- A/R Cash Receipts

J.D. Edwards uses one program for each part or transaction of the system.
Example: Voucher Processing Functional Server

The following graphic shows the programs that use the voucher processing functional server. J.D. Edwards provides two demo versions of the functional server, ZJDE0001 and ZJDE0002.
Program Example - Traditional Architecture

Each program contains both the User Interface Logic and the Data Integrity Logic. You would access this one program to interface with the database.
**User Interface Logic**

The following are aspects of the user interface logic:

- Screen format
- Skip to and section
- Fill screen
- Field formatting
- Help functions
- Error message display
- Touch and feel

**Data Integrity Logic**

The following are aspects of the data integrity logic:

- Field editing
- Multi-field editing
- Transaction editing
- Default logic
- Error message selection
- Tax processing
- Currency processing
- Database update
Example - Traditional Architecture

... Alternative Method of Entry

If a user wanted the screen to look different, the User Interface Logic would have to change. The Data Integrity Logic remained the same as it was duplicated.
Example - Traditional Architecture . . .
. . . Various Entry Methods

Several users each wanted their own User Integrity logic. The Data Integrity Logic remained the same and was duplicated too many times.
Example - JDE Open Application Architecture . . .
. . . Various Entry Methods

The creation of a Functional Server allows you to maintain the Data Integrity Logic in one common program. The Functional Server becomes separated from each User Integrity Logic program. All of the User Integrity Logic programs access one Functional Server to interface with the database. This concept is called an Open Application Architecture.
Open Application Architecture

In the Open Application Architecture, the database is separated from each User Integrity Logic program by the Functional Server. The advantages of the Open Application Architecture are:

- Automatic Consistency
- Reduced Maintenance Burden
- Stability of Custom Code
- Separation of Development Efforts
- Performance Enhancements
**Functional Server Interface**

A Functional Server must handle two basic components:

- Data
- Error messages
**Functional Server Transaction Data**

Arithmetic:

1) Full transaction passed to server at one time

2) A single transaction can have more than 1,000 lines

3) Each line from 500 to 1,000 characters long

= A lot of space

Story Problem:

How can program A pass program B a one thousand line transaction without using a 1-meg parameter?

**Functional Server Error Messages**

Arithmetic:

1) Each field can have an error

2) Each line can have 150 or more fields

3) Each transaction can have hundreds of lines

= A lot of space

Story Problem:

How can program A pass program B a one thousand line transaction without using a 1-meg parameter?

**Answers**

#1. User Space

#2 User Index

**Functional Server Interface**

A Functional Server can interact with a User Space and a User Index by passing and receiving parameters.
**Functional Server Parameters**

Single data structure defined in /COPY module

Two sections: fixed and application specific

- Fixed parameters
  - Action code (edit, update, inquire)
  - Number of lines in transactions
  - DREAM Writer version of Functional Server

- Application specific parameters
  - Contains header information for a transaction
  - Document number of transaction
  - Total amount of transaction
  - Batch number of transaction

**Functional Server User Space**

Single data structure defined in /COPY module

- One big data area
- Maximum of 16 meg
- Beginning 100 bytes of user space reserved
- Data portion of user space contains formatted lines
  - User space lines defined by /COPY module
  - Each line contains three sections
    - Control section
    - Application specific section
    - Record format section
**Functional Server User Index**

Single data structure defined in /COPY module

- One big keyed data area
- Used to pass error messages back to application
- User index entry defined using a /COPY module
- Each user index entry contains two sections
  - Key
    - Application ID
    - Line number (assigned by application program)
    - Data item in error
    - Error code
  - Data - value of erroneous data

**Functional Server /COPY Modules**

Repository for all user space and user index formats

- All User Space and User Index formats contained in /COPY modules
- All database record formats contained in /COPY modules
- /COPY module I00FS@@ contains generic data structures and constants
- Each Functional Server has its own I00FSxx /COPY module to define application specific data structures
Creating User Space and User Index

When you create user space and user index formats, use the following tools:

- OS/400 APIs
- X00991
  - Called once for each Functional Server an application program intends to use
  - Creates user space and user index for each Functional Server
  - Returns name and library where user space exists
  - Returns the length each user space line should be

Accessing the User Space

- Writing to the user space X98CHGUS
  - J.D. Edwards version of QUSCHGUS API
  - Updates a user space beginning at offset x for length
  - Similar to CHGDTAARA command
- Reading from the user space QUSRIVUS
  - API
  - Retrieves data from a user space beginning at offset x for length
  - Similar to RTVDTAARA command
- Application responsibilities
  - Remember number of lines written to user space
  - Increment user space offset


**Accessing the User Index**

- User Index written to by Functional Server
- Reading from the User Index
  - C00RIX/COPY module reads the User Index
  - C00RIX returns formatted error message defined by /COPY module
  - First execution of C00RIX reads first entry in User Index
  - Subsequent executions of C00RIX do read nexts
  - Uses X00IDX under the covers
- Application responsibilities
  - Remember the value of your Application ID (typically program name)
  - Set flag for initial read of User Index by C00RIX
  - Use the data item name and line number in error to set on screen indicators

**Interactive Program Cycle Using a Functional Server**

- Mainline - no change
- S001 - no change
- S003
  - No change for add, change, or delete
  - Call Functional Server to perform an inquiry
- S004 – Retrieve records from User Space for display on screen
- S005:
  - Application program performs “scrubs” only
  - Write data records to User Space
  - Call Functional Server to perform edits
  - Read each line from User Space to redisplay defaulted information
  - Execute C00RIX to determine each data item in error so that screen error indicators may be set ON
- S010 – call Functional Server to perform an update
# The Call Parameters for the Functional Server

The call parameters provide commands to the functional server which apply to all transaction lines in the input user space.

<table>
<thead>
<tr>
<th>PARM (Length)</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>#PFUNC (1)</td>
<td>Specifies a function code. The valid values are:</td>
</tr>
<tr>
<td></td>
<td>0 Edit and Update</td>
</tr>
<tr>
<td></td>
<td>1 Edit only</td>
</tr>
<tr>
<td></td>
<td>2 Update only</td>
</tr>
<tr>
<td></td>
<td>1 Inquire</td>
</tr>
<tr>
<td>#PVERS (3)</td>
<td>The DREAM Writer version number you are executing. This parameter uses the version number to retrieve processing options for the server. The default version number will be 001. This allows global processing options to be set at the server level, instead of for each program. (10 as of A6)</td>
</tr>
<tr>
<td>#PSPCN (20)</td>
<td>The name of the user space which the program has used. The user space contains the modified database records. Characters 1–10 contain the space name, and characters 11–20 contain the library name.</td>
</tr>
<tr>
<td>#PSPCB (9,0)</td>
<td>The byte position within the user space where the application data begins. Characters in the space prior to this position contain header information used by the functional server.</td>
</tr>
<tr>
<td>#PNBRL (5,0)</td>
<td>The number of lines in the input user space which the application program has loaded. When inquiring, this contains the number of lines output to the user space.</td>
</tr>
<tr>
<td>#PWARN (1)</td>
<td>This parameter contains a code explaining how you want warnings to be handled. The valid values are:</td>
</tr>
<tr>
<td></td>
<td>0 Normal warning processing</td>
</tr>
<tr>
<td></td>
<td>1 Treat warnings as errors</td>
</tr>
<tr>
<td></td>
<td>2 Ignore warnings</td>
</tr>
<tr>
<td>#PCYCL (1)</td>
<td>This parameter is only used if the #PWARN parameter specifies normal warning processing. The valid values are:</td>
</tr>
<tr>
<td></td>
<td>0 No cycle, all cycle processing ignored</td>
</tr>
<tr>
<td></td>
<td>1 First cycle, all warning messages are sent to the program</td>
</tr>
<tr>
<td></td>
<td>2 Second cycle, only warning messages not previously sent are sent to the program</td>
</tr>
<tr>
<td>#PDFTC (1)</td>
<td>Specifies how you want field values to be defaulted. 0 will default field values for add lines only and 1 will default field values for change or add lines.</td>
</tr>
<tr>
<td>#PXATP (3)</td>
<td>The application specific transaction type.</td>
</tr>
<tr>
<td>PARM (Length)</td>
<td>Explanation</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------</td>
</tr>
<tr>
<td>#PLVL (1)</td>
<td>The transaction level. 0 implies that each detail record to be updated or added has been sent in the input user space. 1 applies only to changes or deletions because only one record is sent in the input user space and the server will change or delete all other records for that transaction.</td>
</tr>
<tr>
<td>#PPROG (10)</td>
<td>The name of the calling program. This is used by the server to update the program name field in the updated database records.</td>
</tr>
<tr>
<td>#PAPPL (10)</td>
<td>The application ID value used for writing entries to the error index. Generally, this may be the same value as the calling program.</td>
</tr>
<tr>
<td>#PFLDS (4,0)</td>
<td>The number of fields which have been loaded to the Field Names Array parameter.</td>
</tr>
<tr>
<td>#PFMT (10)</td>
<td>The record format identifier the application program has used. This is used for versioning, allowing the database to change without the need for recompiling the application program.</td>
</tr>
<tr>
<td>#PEDIT (1)</td>
<td>Indicates the overall result of edits performed against all transaction lines. 0 implies that the edits went OK, 1 means there were some warnings, 2 is errors occurred.</td>
</tr>
<tr>
<td>#PUPDT (5,0)</td>
<td>The number of database updates which occurred. This will allow the program to know whether any updates actually occurred.</td>
</tr>
<tr>
<td>#PERR (4)</td>
<td>Specifies any errors that occurred within the server. A non–blank value indicates a fatal error occurred.</td>
</tr>
<tr>
<td>#PFERR (4)</td>
<td>Contains the first error message found during editing.</td>
</tr>
<tr>
<td>#PFDTA (4)</td>
<td>Contains the data item of the first field which had an error during editing.</td>
</tr>
<tr>
<td>#P#MDE (1)</td>
<td>For currency translations, this contains the mode of entry. If this value is passed as blank, the server will output the default mode of entry.</td>
</tr>
<tr>
<td>#PCRCD (3)</td>
<td>For currency translations, this contains the currency code of entry. If this value is passed as blank, the server will output the default currency code.</td>
</tr>
<tr>
<td>#PCRR (15,7)</td>
<td>For currency translations, this contains the currency exchange rate of entry. If this value is passed as zero, the server will output the default currency rate.</td>
</tr>
<tr>
<td>#PIDXN (20)</td>
<td>The name of the user index which the functional server will use to return error messages to the program. Characters 1–10 contain the index name, and characters 11–20 contain the library name.</td>
</tr>
<tr>
<td>#PSPCL (5,0)</td>
<td>The total length of each user space record. This includes both the user space control fields and the database record format.</td>
</tr>
<tr>
<td>PARM (Length)</td>
<td>Explanation</td>
</tr>
<tr>
<td>--------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>#0SPSPEC (100)</td>
<td>This is a data structure which is redefined by each server. Generally, this will contain the key fields which a specific server uses.</td>
</tr>
<tr>
<td>VariableVary</td>
<td>An array of field names which the program has used. Only fields in this array will be updated in the database. If the first element contains *ALL, then all fields will be used. The number of field names parameter should contain the number of entries loaded into this array.</td>
</tr>
</tbody>
</table>
Control Fields within the User Space

The input user space can contain multiple lines for each control field.

<table>
<thead>
<tr>
<th>PARM (Length)</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>#SPCAC (1)</td>
<td>The line action code. The valid values are:</td>
</tr>
<tr>
<td></td>
<td>A</td>
</tr>
<tr>
<td></td>
<td>D</td>
</tr>
<tr>
<td></td>
<td>C</td>
</tr>
<tr>
<td></td>
<td>U</td>
</tr>
<tr>
<td></td>
<td>V</td>
</tr>
<tr>
<td>#SPCID (15,0)</td>
<td>Used by the program to uniquely identify each line in the user space. (optional)</td>
</tr>
<tr>
<td>#SPCER (1)</td>
<td>The line error code.</td>
</tr>
<tr>
<td></td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>2</td>
</tr>
<tr>
<td>#SPCUP (1)</td>
<td>The line update code.</td>
</tr>
<tr>
<td></td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>1</td>
</tr>
<tr>
<td>#SPCRR (9,0)</td>
<td>Contains the database relative record number which corresponds to this user space record. For adds, this is only loaded following an update operation. For changes and deletes, this is updated following an edit operation.</td>
</tr>
<tr>
<td>#SPCMN (2,0)</td>
<td>Contains the database physical file member number which corresponds to this user space record. For adds, this is only loaded following an update operation. For changes and deletes, this is updated following an edit operation.</td>
</tr>
<tr>
<td>#SPCPG (12)</td>
<td>Allows the program to store up to 12 bytes of information with each user space record.</td>
</tr>
<tr>
<td>#SPCAP (100)</td>
<td>Any application specific information which must be passed to the server for each transaction line, but is not contained within the transaction record format.</td>
</tr>
<tr>
<td>VariableVary</td>
<td>Externally described record format for the transaction record.</td>
</tr>
</tbody>
</table>
**Error Message Index Line (C00RIX)**

The output error message index contains warning and error messages issued for each line in the user space. The structure of the message index line is as follows:

<table>
<thead>
<tr>
<th>Field (Length)</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>#IDXAP (10)</td>
<td>The application identifier from the input parameter. Allows a program to access only its error messages.</td>
</tr>
<tr>
<td>#IDXID (15,0)</td>
<td>The line identifier from the input user space.</td>
</tr>
<tr>
<td>#IDXFN (10)</td>
<td>The data item portion of the field name.</td>
</tr>
<tr>
<td>#IDXER (4)</td>
<td>Contains the data dictionary error message code.</td>
</tr>
<tr>
<td>#IDXMD (88)</td>
<td>Contains the error message substitution data. Generally, this is the value of the field which caused the error.</td>
</tr>
</tbody>
</table>

Interactive programs using a functional server must include a call to P0000EX (in addition to P0000E) in S00EX when the F7 (Display Errors) key is pressed. P0000EX will retrieve and display the error messages contained in the Error Message Index (C00RIX).
Example - Functional Server Program Sections

Copy module containing generic data structures for functional server.

Contains control parameter list for file servers.

Contains record image of F0101 version A6.1 for file servers.

Call to file server XS0010 to retrieve company currency code.

Call to file server XS0013 to retrieve display decimals.

Call to file server XF0101 to retrieve record.
Call to file server XF0101 to update record

Load AR Specific Parameters

MOVE #GLDCT #ARDCT
MOVE $SVKCO #ARKCO
Z–ADD #GLDOC #ARDOC
Z–ADD #GLICU #ARICU
MOVE #GLICT #ARICT
MOVE *BLANK #ARSPL

Load functional server parms for edit/update.

MOVE #XIDXN #PIDXN index name
MOVE #SPAR #PSPEC application
MOVE #EDUP #PFUNC function
MOVE #311 #PVERS DW version
Z–ADD #ARBG #PSPCB space offset
MOVE #IGNW #PWARN warning handler
MOVE #OFF #PDFTC default on chg
MOVE #PROG #PPROG program name
MOVE 'INV' #PXATP type
MOVE #ARSN #PSFCN space name
Z–ADD #ARSL #PSFCL space length
Z–ADD #AR #PFMT number of field
MOVE *BLANKS #PFMT
MOVE #AR1 #PFMT format name

CALL 'XT0311Z1' 81

----- -------
PARM #PPARM
PARM @ARN

Call functional server XT0311Z1
User space description

MOVESUGL SPCD

Current user space offset

ZADD$GLBG SPCOF

Set update flag

MOVE #OFF SPCUP

General Ledger record

MOVEAGL01 SSPC

Application specific line data

MOVELSSGL SPCAP

Write record to user space

CALL 'X98CHGUS'PCHUS S1

ENDIF

##edit
Load G/L Functional Server Specific Parameters

MOVE *ZERO #GLDOC One-to-One Rel
MOVE '8' #GLDCT Document Type
MOVE $SVKCO #GLKCO Document Co.
Z-ADD$GLDG #GLDG G/L Date
Z-ADD$GLDG# #GLDG# G/L Date
$PICU
Z-ADD*ZERO #GLICU Batch Number
ELSE
Z-ADD$ICU #GLICU Batch Number
ENDIF

MOVE 'I' #GLICT Batch Type
MOVE $SVCO #GLCO Company
MOVE *BLANKS #GLMOD Add a Model
MOVE *BLANKS #GLMOD Change a Model
MOVE #ARSN #GLCSN A/R Spc Name
MOVE #AR1 #GLCFM A/R Spc Fmt
MOVE #ARSL #GLCLN A/R Spc Length
MOVE #OFF #GLCDG
MOVE *ZERO #GLCD
MOVE #ON #GLONE One-to-One Rel

Call functional server - XT0911Z1 – Edit and Update

Load functional server parms for edit and update

MOVE $GACTN #PFUNC Action Code
MOVE$#911 #PVERS DW version
MOVE #GLSN #PSPCN DW version
Z-ADD#GLBG #PSPCB DW version
Z-ADD1 #FMBRL DW version
MOVE ##IGNW #PWARN warning handler
Z-ADD*ZERO #PCRR Exchange Rate
MOVE #OFF #PCYCL cycle
MOVE #OFF #PDFTC default on chg
MOVE 'INV' #PXATP type
MOVE #OFF #PLVL detail level
MOVE #PROG #PPROG program name
Z-ADD$GL #PFHDS number of field
MOVE *BLANKS #PFMT format name
MOVE $GL #FPMT mode of entry
MOVE *BLANKS #P#MDE currency code
MOVE *BLANKS #PCR exchange rate
MOVE #XIDXN #PIDXN index name
Z-ADD$GL #PSPCL space length
MOVEL#GSL #PSPE application par

CALL 'XT0911Z1' 81

PARM #PPARM
PARM #GLN

User space description

MOVEL#SUAR #SSPCD

Current user space offset

Z-ADD$ARRBG #SPCOF

Read record from user space

CALL 'QUSRTVUS' #PRTUS 81

Retrieved record from user space.
Create Functional Server Objects for XT0311Z1

```
CREATE #PCRT
MOVE #AD #PCRTF
MOVE *BLANK #PCRTN
MOVEL'XT0311Z1' #PCRTN
CALL 'X00991' S1
- PARM #PCRT
```

Create user space and user index for XT0311Z1.

Create Functional Server Objects for XT0911Z1

```
CREATE #PCRT
MOVE #AD #PCRTF
MOVE *BLANK #PCRTN
MOVEL'XT0911Z1' #PCRTN
CALL 'X00991' S1
- PARM #PCRT
```

Create user space and use index for XT0911Z1.
### Available Functional Servers

<table>
<thead>
<tr>
<th>Funct. Server</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>XT006Z1</td>
<td>Cost Center Master</td>
<td></td>
</tr>
<tr>
<td>XT0101Z1</td>
<td>Address Book</td>
<td></td>
</tr>
<tr>
<td>XT0311Z1</td>
<td>Accounts Receivable</td>
<td></td>
</tr>
<tr>
<td>XT0311Z1E</td>
<td>Accounts Receivable</td>
<td></td>
</tr>
<tr>
<td>XT0411Z1</td>
<td>Accounts Payable</td>
<td></td>
</tr>
<tr>
<td>XT0411Z1E</td>
<td>Accounts Payable</td>
<td></td>
</tr>
<tr>
<td>XT0411Z2</td>
<td>Accounts Payable Check</td>
<td></td>
</tr>
<tr>
<td>XT06116Z1</td>
<td>Payroll Time Entry</td>
<td></td>
</tr>
<tr>
<td>XT0901Z1</td>
<td>Account Master</td>
<td></td>
</tr>
<tr>
<td>XT0911Z1</td>
<td>Journal Entry</td>
<td></td>
</tr>
<tr>
<td>XT0911Z1E</td>
<td>Journal Entry</td>
<td></td>
</tr>
<tr>
<td>XT4102Z1</td>
<td>Item Balance</td>
<td></td>
</tr>
</tbody>
</table>

**Exercises**

See the exercises for this chapter.
Source Debugger

About Source Debugger

There are two types of programs that can be executed under the J.D. Edwards Source Debugger - interactive and batch. The only difference when running the Source Debugger on an interactive program compared to a batch program is the initial execution statements. Once the Source Debugger has begun, all of the features are the same for both interactive and batch programs.

The J.D. Edwards Source Debugger is a tool designed to help you determine where a bug exists in your program. You can apply the Source Debugger to any program, whether it is in production or development. Since the Source Debugger displays source code, you must have the source code installed on your machine.

The source code you see while running the Source Debugger is displayed in SEU Browse mode, so you can not change a line within the program. However, you may display or change the value of any field, variable, or indicator within the program. In addition, you can add or remove a breakpoint anywhere in the program.

Before You Begin

- If you are not accessing the J.D. Edwards training machine, you must recompile programs into your student object library or your client object library, CLTOBJ or DEVOBJ before executing JDEDBG.
  
  - This ensures that the program is observable and therefore, accessible to the Source Debugger.

- If you are accessing the J.D. Edwards training machine, you may execute the JDEDBG command on any of the following programs: P92801, J928401, and P928401. You can also recompile any desired program in JDOBJ to run in the Source Debugger.
Using Debugger with an Interactive Program

The program can exist in your production environment, your development environment, or both. To use Debugger complete the following tasks:

- Determine the program environment
- Initiate the J.D. Edwards Source Debugger
- Execute the program being debugged

To determine the program environment

1. From the Computer Assisted Design menu (G92), select Software Versions Repository.

```
9801 Software Versions Repository
Action Code. . . . I
Member ID. . . . P01051
Description. . . Address Book Information
Function Code. . RPG RPG Programs
Function Use . . File Maintenance
System Code. . . 91 Address Book
Reporting System 91 Address Book
Base Member Name P01051
Maint/RTSPSFP . . File Prefix. .
DREAM Writer Form Exists
Opt: 1=Browse 2=Edit 3=Copy 5=SAR 8=Print 9=Dlt 10=Design 14=Crt F24=More
```

2. Locate the program on which you want to run the Source Debugger, to determine in what environments the program exists.

If the program exists in several environments (production and development), you must determine against which program environment to run the Source Debugger.
To initiate the J.D. Edwards Source Debugger

1. Type the J.D. Edwards debug command (JDEDBG) and press F4.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Program Name</td>
<td>Type your program name</td>
</tr>
<tr>
<td>Source File</td>
<td>Type the file name that contains the source code of your program. Generally, this is JDESRC.</td>
</tr>
<tr>
<td>Library</td>
<td>Type the name of the library that contains the source file. Generally, this is JDFSRC for your production environment or DEVSRC for your development environment.</td>
</tr>
</tbody>
</table>

2. Enter the correct values in the proper fields and press Enter to start the Source Debugger.
   - Now, any time the program being debugged is executed, the source code will display in debug mode, until you end the Source Debugger.
To execute the program being debugged

Because it is an interactive program, you can either call the program from a command line or select the menu option that will execute the program.

- Call program name ("parameters")
- Selection/Menu

After you have executed the program, the first thing you will see is the program source code.

---

The source code is displayed in browse mode, so you cannot edit or change any code.
Using Debugger with a Batch Program

The program may exist in your production environment, your development environment, or both.

To use Debugger with a batch program you should complete the following tasks:

- Determine the program environment
- Initiate the J.D. Edwards Source Debugger
- Execute the program
- Set the break point
- Continue execution

To determine the program environment

This step is the same as the first one for an interactive program.

1. Go to the Software Versions Repository and inquire on your program, to determine against which environment the Source Debugger will be run.

To initiate the J.D. Edwards Source Debugger

This step is similar to debugging an interactive program. The difference is that you must enter the debug command twice.

The first time you initiate J.D. Edwards Source Debugger (JDEDBG - F4), the Program Name will be the CL Program.

1. Enter the correct values in the proper fields on the Debug Program form and press Enter.
2. Enter the J.D. Edwards Source Debugger command (JDEDBG - F4) again, but this time change the Program Name to the RPG Program Name.

   - The reason for this is, you cannot run the Source Debugger on a program that is submitted and executed in a subsystem. You must "trick" the Source Debugger into thinking that your batch program is actually an interactive program.
To execute the program

Since you are executing a batch program interactively, you must call the CL Program from a command line.

call CL program ("program name" "version")

The CL Program source code appears.

The source code is displayed in browse mode, so you cannot edit or change any code.
To set the break point

Set a break point on the line testing the job type in order to change a variable in the CL. The variable &JOBTYPY normally edits against a batch program being executed by calling it from a command line.

1. Find the line of code that contains the variable &JOBTYPY.

2. Press F5 anywhere on the line containing &JOBTYPY to set the breakpoint.
   - The line is highlighted, indicating that a breakpoint has been set on that line.

To continue execution

1. Allow your program to continue executing. Press F3 to continue to a breakpoint.
   - The line on which you set the breakpoint will display in reverse image. This indicates that the program has reached this point in the CL program and is ready to execute this line.
   - You must change the value of &JOBTYPY to something other than 1, and other than the value specified in the CL program.

2. To change the value of &JOBTYPY, press F8 to access the Change Program Variable form.
Change Program Variable (CHGPGMVAR)

Type choices, press Enter.

Program variables:
  Program variable . . . . . . .   &JOBTYPE'
  -
  - Basing pointer variable . .
  -
  - + for more values

New value . . . . . . . . . . .   '12'
Program . . . . . . . . . . . > J928401   Name, *DFTPGM

3. Complete the Change Program Variable form and press enter.
   - The value of &JOBTYPE is now changed to your specified value.

4. Press F3 to allow the CL program to continue processing.
   - The RPG program source is displayed next.
Features of the J.D. Edwards Source Debugger

F2 - J.D. Edwards Command Line Window

To display a J.D. Edwards command line window, press F2.

F3 - Continue processing

Once the program hits a breakpoint or when you first enter the source, F3 will allow the program to continue processing.

F5 - Add breakpoint

Position the cursor on an executable line and press F5 to add a breakpoint. You cannot add breakpoints to a comment line, only to executable lines. Once the breakpoint is set, the line will be highlighted. If the program executes a line with a breakpoint set on it, the line will appear in reverse image and the program will pause before executing the line.
**F6 - Add breakpoint with prompt**

Position the cursor on an executable line and press F6 to add a breakpoint with a prompt. You cannot add breakpoints to a comment line, only to executable lines. Once the breakpoint is set, the line will highlight. If the program executes a line with a breakpoint set on it, the line will reverse image and the program will pause before executing the line.

Use the prompt, after pressing F10, to assign a skip value or breakpoint conditions.
F7 - Display Program Variable

Position the cursor on an executable line and press F7 to display the values of all of the variables on that line. Breakpoints within copy modules will stop at the correct source sequence number.

Display Program Variables

Program . . . . . . . . . . . . . . . : P01051
Recursion level . . . . . . . . . . . : 1
Start position . . . . . . . . . . . : 1
Format . . . . . . . . . . . . . . . . : *CHAR
Length . . . . . . . . . . . . . . . . : *DCL

Variable . . . . . . . . . . . . . . . : *IN99
Type . . . . . . . . . . . . . . . . . : CHARACTER
Length . . . . . . . . . . . . . . . . : 1
'0'

Variable . . . . . . . . . . . . . . . : *IN93
Type . . . . . . . . . . . . . . . . . : CHARACTER
Length . . . . . . . . . . . . . . . . : 1
'0'

Press Enter to continue.

F3=Exit   F12=Cancel
**F8 - Change Program Variable**

To change the value of a variable, press F8 and type the correct values in the prompt screen.

```
Change Program Variable (CHGPGMVAR)

Type choices, press Enter.

Program variables:
- Program variable . . . . . .
- Basing pointer variable . . .
- + for more values
- New value . . . . . . . . . . .

Program . . . . . . . . . . . . > P01051 Name, *DFTPGM
```

F3=Exit  F4=Prompt  F5=Refresh  F10=Additional parameters  F12=Cancel
F13=How to use this display  F24=More keys

**F10 - Move Line to Top of Page**

**F12 - Remove Current Breakpoint**

From anywhere on the screen, press F12 to remove the current breakpoint. The line is no longer highlighted, indicating the line is no longer set as a breakpoint. The program will immediately continue processing.
**F13 - Display Indicator Values**

To display the current values of all indicators, press F13.

```
Display Program Variables

Program . . . . . . . . . . . . . . . . : P01051
Recursion level . . . . . . . . . . . . : 1
Start position . . . . . . . . . . . . : 1
Format . . . . . . . . . . . . . . . . : *CHAR
Length . . . . . . . . . . . . . . . . : *DCL

Variable . . . . . . . . . . . . . . . : *IN
Lower/upper bounds . . . . . . . . . : (1:99)
Type . . . . . . . . . . . . . . . . : CHARACTER
Length . . . . . . . . . . . . . . . . : 1

Element | Values
--- | ---
1 | '0' '0' '0' '0' '0' '0' '0' '0' '0' '0'
11 | '0' '0' '0' '0' '0' '0' '0' '0' '0' '0'
21 | '0' '0' '0' '0' '0' '0' '0' '0' '0' '0'
31 | '0' '0' '0' '0' '0' '0' '0' '0' '0' '0'
41 | '0' '0' '0' '0' '0' '0' '0' '0' '0' '0'
51 | '0' '0' '0' '0' '0' '0' '0' '0' '0' '0'

Press Enter to continue.
```

F3=Exit  F12=Cancel

**F15 - Scan Backward**

Type in a value on the Scan Line at the top of the screen and press F15 to scan backward from the point you are at to the end of the source code. If a match is found, the line containing the matching value will be displayed. To continue scanning backward, press F15 again.

**F16 - Scan Forward**

Type in a value on the Scan Line at the top of the screen and press F16 to scan forward from the point you are at to the beginning of the source code. If a match is found, the line containing the matching value will be displayed. To continue scanning forward, press F16 again.
**F21 - Command Line Window**

To display a command line, press F21.

**ENDBBG  End Debug**

To stop the J.D. Edwards Source Debugger, enter ENDBBG from a command line. You can not enter ENDBBG while displaying the source code of a program in debug. This command will end debug mode for all programs in the Debugger at that point.

You can remove a single program from debug mode by using the RMVPGM (remove program) command.

**Exercises**

See the exercises for this chapter.
Software Scan and Replace

About Software Scan and Replace

The Software Scan and Replace feature lets you scan source members to accomplish the following:

- Scan for a particular item and replace it with a new item
- Produce a list of all members that meet the search criteria
- Scan for a particular item and insert a source file after each occurrence

Because you can potentially replace source code across all systems, this job is submitted to batch and held in the job queue until you release it.

To Work with Software Scan and Replace

1. From the Computer Assisted Programming menu, select Developer’s Workbench. From the Developer’s Workbench menu, select Software Scan and Replace.

The previous screen illustrates how you replace the copy module I00SC with the copy module I00RSC for all RPG members coded to install system code 55.
2. Complete the form and press Enter.
   • The job submits to batch and a message displays. The job is held on the job queue.

3. When you are ready to process the job, go to the Work with Submitted Jobs form (hidden selection 35) and release the job.

Report

When the job completes, it produces a report that indicates those objects where the scan and replace occurred.

```
98810                                J. D. Edwards & Company 1
Scan Software Source                4/01/91

System 55
Function: RPG
Object:
   File: JDESRC
Source Lib: DEVSRC
   Argument: "I00SC"
Replace By: "I00RSC"
Column End: 000
Column End: 000
Allow Ovrf:
   Insert Frm- File:            Libr:            Memb:            Action    : Replacement    Scan/Replace Characters= 05/06
5501G       - Item Maintenance - Gregg          1st Occurrence at 010200
P5501X      - Item Maintenance                1st Occurrence at 010200
P55011X     - Item Information Update         1st Occurrence at 005200
```

Guidelines

If you leave the Replacement argument field blank, the utility produces a listing of all source members that meet the search criteria.

Because this job could be used to update all code across systems and could severely impact processing, it is automatically held.

Use this job to replace a copy module across systems or determine a listing of members that meet certain criteria. Use with caution.
Performance Issues

About Performance Issues

Following are some performance issues you should consider when executing J.D. Edwards software, changing current J.D. Edwards programs or writing new programs:

- Purge your files on a regular basis to avoid excess, unnecessary records existing in files.
- Minimize the number of open files in a program. If a file may not be used, define it as a User Controlled Open file.
- Use User Spaces and User Indexes wherever possible.
- Use File Servers and Functional Servers wherever possible.
- Minimize the number of subroutine calls within your program.
- Weigh the advantages of inter-program calls. Although this method is very modular in design, you should consider the effect on performance.
- Substitute the comparison of a literal with the comparison of a variable.

For example: Use *ON and *OFF to set an indicator on and off rather that a 1 and 0.

- Consider flexibility versus performance when using User Defined Codes, Vocabulary Overrides, and loading Data Dictionary values extensively.
Group Jobs

Objectives

- Work with the J.D. Edwards Group Job Form
- Work with J.D. Edwards group jobs
- Work with non-J.D. Edwards group jobs
- Work with the J.D. Edwards Attention MENU Form
- Use IBM Pass-Through with group jobs

About Group Jobs

The Group Jobs form allows you to perform a number of tasks from a single form, saving you both time and effort. You can perform the following functions from this form:

- Run up to 16 jobs under a single signon
- Execute (or run) CL and fast path commands from a single command line
- Execute (or run) J.D. Edwards Hidden Selections

In addition to the added convenience, the Group Jobs function keeps the files for each of the jobs selected opened, whether they are currently active or not.

Perform the following tasks:

- Access the J.D. Edwards Group Job Form
- Create New Group Jobs
- Activate Suspended Group Jobs
- Terminate Job Groups
- Change to Non-Group Mode
- Sign Off with Suspended Group Jobs
Access the J.D. Edwards Group Job Form

About the J.D. Edwards Group Job Form

You can perform several operations using the J.D. Edwards Group Jobs Form, including:

- Create new group jobs
- Activate suspended group jobs
- Terminate group jobs
- Change to non-group mode
- Sign off with suspended group jobs
- Work with non-J.D. Edwards group jobs
Advanced Programming Concepts and Skills

Before You Begin

For a user to access the J.D. Edwards Group Job Form at any time, the ATTN key program should be set to call the J. D. Edwards Group Job Form program (P98GRP).

To set the ATTN key program

1. From the Security Officer Menu, select User Information.

```
0092 User Information Action Code. . . . . I

User ID. . . . . . . . . . . . . . . TEACH
Library List . . . . . . . . . . . . QTEMP JDFOBJ COMMON PRODDATA JDFSRC QGPL

User Security:
User Key . . . . . . . . . . . . . . A J K DP F
Initial Menu to Execute . . . . A
Initial Program to Execute . . . .
Menu Level. . . . . . . . . . . . . .
User Type. . . . . . . . . . . . . . TEACHER
User Class/Group . . . . . . . . QBATCH
Init Job Scheduling Priority. . . 5 5
Logging(level/severity/messages) . . 4 00 *Nolist
Output Queue . . . . . . . . . . . . P4B
Optional Printer File Library. . . .
Current Library. . . . . . . . . . .
Employee Address Number (PPAT) . . . . . . . . . . . .
Set Attention Program . . . . . . . P98GRP

F6=Display/Lang Pref  F9=Library Inquiry  F21=Print Lib List  F24=More Keys
```

2. Enter the J.D. Edwards Group Job Form program ID (P98GRP) in the Set Attention Program field.
Accessing the J.D. Edwards Group Job Form

After the ATTN Key program has been set up in the J.D. Edwards software, you can access the Group Job Form.

To access the J.D. Edwards Group Job Form

1. Sign off and sign back on to reset the ATTN key program within the J.D. Edwards Menu Driver.
2. Press the ATTN key and the following is displayed.
Creating New Group Jobs

To create new group jobs

2. When the J.D. Edwards Menu Driver is displayed, press the ATTN key and the following is displayed.

The new group job GROUP02 is now in process. The group job GROUP01 was suspended when the function key F5 was pressed.

If you are set up to access J.D. Edwards software by J98INITA, your library list selection list will appear. Select an environment and then you will be able to display the J.D. Edwards Group Job Form.
Activating Suspended Group Jobs

To activate suspended group jobs

Press the ATTN key to display the J.D. Edwards Group Job Form and enter option 4 next the the job you want to activate.

- All suspended group jobs is displayed in the form.
- Any suspended group job can be activated, as illustrated below.

<table>
<thead>
<tr>
<th>Description</th>
<th>Group Job</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>GROUP02</td>
<td>Active</td>
<td></td>
</tr>
<tr>
<td>GROUP01</td>
<td>Suspended</td>
<td></td>
</tr>
</tbody>
</table>

Selection or command  
===>>

Cmd/HS: Opt: 4=Sel 9=End F3=Exit F4=Prompt F5=New Job
# Terminating Group Jobs

Any group job, active or suspended, may be terminated from the J.D. Edwards Group Job Form.

**To terminate group jobs**

Enter option 9 next to the group job you want to terminate.

```
<table>
<thead>
<tr>
<th>Description</th>
<th>Group Job</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>98GRP----------</td>
<td>GROUP01</td>
<td>Active</td>
</tr>
<tr>
<td>GROUP02</td>
<td>Suspended</td>
<td></td>
</tr>
</tbody>
</table>
```

Selection or command

```
===>>
```

Cmd/HS: Opt: 4=Sel 9=End F3=Exit F4=Prompt F5=New Job
# Changing to Non-Group Mode

To change to non-group mode

Enter option 9 beside all active and suspended group jobs.

<table>
<thead>
<tr>
<th>Selection or command</th>
<th>Opt: 4-Sel 9-End F3-Exit F4-Prompt F5-New Job</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>98GRP------------------Group Jobs--------------E-</td>
<td></td>
</tr>
<tr>
<td>9 Description</td>
<td>Group Job</td>
</tr>
<tr>
<td>2</td>
<td>GROUP01</td>
</tr>
<tr>
<td>9</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Signing Off with Suspended Group Jobs

You can use two different methods to sign off with suspended group jobs.

To sign off with suspended group jobs

Select one of the following methods:

- Press F18 within the J.D. Edwards Group Job Form.
- Enter SIGNOFF, 90, or “...” on any J.D. Edwards Menu.

Because group jobs are created under one signon, all group jobs are terminated when the signoff command is executed.
Work with Non-J.D. Edwards Group Jobs

To work with non-J.D. Edwards group jobs

To create group jobs that call a program outside the J.D. Edwards software, the J.D. Edwards Group Job Form allows an external program to be executed. In addition, the ATTN Key can be pressed within the external program and still allow access to the J.D. Edwards group jobs.

1. To call an external program, press F11 within the J.D. Edwards Group Job Form.

The following illustrates what will be displayed when F11 is pressed.

2. Complete the Change Library List form.

You can enter libraries related to the external program. Libraries currently in the library list can be removed if desired. However, the following libraries must be left in the library list to retain the link to the J.D. Edwards group jobs:

- QTEMP
- Library containing F9220 (J.D. Edwards Group Job Form Vocabulary Overrides)
- Library containing F0090 (J.D. Edwards Hidden Selections)
- Library containing F0092 (J.D. Edwards User Information)
- Library containing J.D. Edwards Objects (For example, RPG, CL, DSPF)

After the CHGLIBL command has been executed, the CALL command prompt is displayed.

3. Enter the external program.

The following illustrates the CALL command prompt.

When the CALL command is executed, the external program will be executed.

- To work with a J.D. Edwards group job, the ATTN Key can be pressed to display the J.D. Edwards Group Job Form.
- Any suspended group job can be activated from the J.D. Edwards Group Job Form.
Advanced Functions of the J.D. Edwards Group Job Form

**J.D. Edwards Hidden Selections**

Hidden Selections are commands and features of the J.D. Edwards products that are not available through a menu selection.

- Most J.D. Edwards Hidden Selections (31+) can be executed from the command line at the bottom of the J.D. Edwards Group Job Form.
- The J.D. Edwards Hidden Selection Form (HS) can be used to display and execute hidden selections.
- J.D. Edwards Hidden Selection Security is used when users execute hidden selections.
- No J.D. Edwards Menus or J.D. Edwards Hidden Selection related to menus are allowed.

**Entering Commands**

You can enter any command on the command line at the bottom of the J.D. Edwards Group Job Form.

- You can press F4 to prompt for a command
- You can place a “?” in front of a command to prompt
- You can press F9 to retrieve previous commands
- Any parameters you enter while in prompt mode are not retrieved
  - The last 10 previous commands are saved.
  - Only successfully executed commands are saved.
  - When you exit by pressing F3, previous commands are lost
- J.D. Edwards Fast Path Commands from User Defined Code 00/FP can be executed. F13 to display all Fast Path Commands.

- To retain all commands entered and retrieve parameters entered in prompt mode, access the IBM Command Entry Form from the J.D. Edwards Group Job Form (For example, J.D. Edwards Hidden Selection 36) and enter commands.
- Commands can only be executed if there is a value of “Y” or “ ” in the Allow Command Entry (Y/N) field defined in the J.D. Edwards User Information option found on A94.
**J.D. Edwards Group Job Form Summary**

The program allows you to:

- Create up to 16 jobs per signon
- Execute commands, J.D. Edwards hidden selections, J.D. Edwards Fast Path Command, and J.D. Edwards Fast Path Menu Execution

**Available Function Keys**

- F3 = Exit the J.D. Edwards Group Job Form
- F4 = Prompt a command
- F5 = Create a new J.D. Edwards group job
- F6 = Submit job to batch
- F8 = J.D. Edwards Menu Word Search
- F9 = Retrieve previous command
- F11 = Create a new Non-J.D. Edwards group job
- F13 = Display all fast path commands
- F18 = SIGOFF all group jobs

**Available Selection Exits**

- 4 = Activate a suspended group job
- 9 = End a group job

J.D. Edwards Group Job Form is not accessible when using

- SysReq (Source Machine Only)
- A program that has reset the ATTN Key program (For example, OFFICE/400)
Work with the Attention MENU Form

About the Attention MENU Form

The J.D. Edwards Attention Menu Form program is a generic program that allows you to access up to 15 predefined programs via the ATTN Key. The 15 predefined programs are associated with options on a J.D. Edwards Menu.

- Each user can be assigned a different J.D. Edwards Menu
- The program was available in Release A4.1 PTF00 – – – – – –1

Before You Begin

To access the J.D. Edwards Attention Menu Form at any time, the ATTN Key program should be set to call some other J.D. Edwards Menu. For example G92.
The following illustrates how the ATTN Key program is set in the J.D. Edwards software. The User Information screen can be accessed from the Security Officers Menu.

<table>
<thead>
<tr>
<th>User ID</th>
<th>TEACH</th>
</tr>
</thead>
<tbody>
<tr>
<td>Library List</td>
<td>QTEMP JDFOBJ COMMON PRODDATA JDFSRC QGPL</td>
</tr>
</tbody>
</table>

**User Security:**
- User Key: A J K DE F
- Initial Program to Execute: A
- Initial Menu to Execute: Allow Command Entry (Y/N) Y
- Allow Menu Traveling (Y/N): Y
- Allow Fast Path (Y/N): Y

<table>
<thead>
<tr>
<th>User Type</th>
<th>TEACHER</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Batch Job Queue</th>
<th>QBATCH</th>
</tr>
</thead>
<tbody>
<tr>
<td>Job Scheduling Priority</td>
<td>5</td>
</tr>
<tr>
<td>Logging (level/severity/messages)</td>
<td>4 00 *NOLIST</td>
</tr>
<tr>
<td>Output Queue</td>
<td>P4B</td>
</tr>
<tr>
<td>Optional Printer File Library</td>
<td></td>
</tr>
<tr>
<td>Current Library</td>
<td></td>
</tr>
<tr>
<td>Employee Address Number (PPAT)</td>
<td></td>
</tr>
<tr>
<td>Set Attention Program</td>
<td></td>
</tr>
</tbody>
</table>

An *(asterisk)* must precede the menu name.

F6=Display/Lang Pref   F9=Library Inquiry   F21=Print Lib List   F24=More Keys
Accessing the J.D. Edwards Attention Menu Form

After the ATTN Key program has been setup for you the J.D. Edwards software you can access the J.D. Edwards attention menu form.

To access the J.D. Edwards attention menu form

1. Sign off and sign back on to reset the ATTN key program within the J.D. Edwards Menu Driver.
   - SETATNPGM PGM(P00AMNU) SET(*ON).
2. Press the ATTN key and the menu options for the menu will be displayed as follows.

**Original Job** refers to the current job that has been converted to a group job. The remaining jobs refer to the first 15 interactive programs on the menu which the user is authorized to.
Summary of J.D. Edwards Attention Menu Form Functions

The program allows you to:

- Access 15 predefined programs via the ATTN Key
- Execute commands, J.D. Edwards Hidden Selections, J.D. Edwards Fast Path Commands, and J.D. Edwards Fast Path Menu Executions

Available Function Keys

- F3 = Exit the J.D. Edwards Attention Menu Form
- F4 = Prompt a command
- F6 = Submit a job to batch
- F8 = J.D. Edwards Menu Word Search
- F9 = Retrieve previous command
- F13 = Display all fast path commands
- F18 = SIGNOFF all group jobs

Available Selection Exits

- 4 = Activate a group job
- 9 = End a group job

J.D. Edwards Attention Menu Form is not accessible while using

- SysReq (Source Machine Only)
- a program that has reset the ATTN Key program (i.e. OFFICE/400)
Work with IBM Pass-Through

About Working with IBM Pass-Through

To create group jobs on remote locations and still retain a link to the group jobs created on the source machine, use IBM Pass-Through. Perform the following tasks:

- Set up access to remote locations
- Use IBM Pass-Through with Group Jobs
Setting Up Access to Remote Locations

To setup access to remote locations

1. To setup access to remote locations, go to the DREAM Writer versions list for Form ID P98GRP5.

<table>
<thead>
<tr>
<th>Version</th>
<th>Description</th>
<th>User</th>
<th>Chg Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>XJDE0001</td>
<td>Denver A</td>
<td>DEMO</td>
<td>08/23/93</td>
</tr>
<tr>
<td>XJDE0002</td>
<td>Denver C</td>
<td>DEMO</td>
<td>08/23/93</td>
</tr>
<tr>
<td>XJDE0003</td>
<td>Denver D</td>
<td>DEMO</td>
<td>08/23/93</td>
</tr>
<tr>
<td>XJDE0004</td>
<td>Denver E</td>
<td>DEMO</td>
<td>08/23/93</td>
</tr>
<tr>
<td>XJDE0005</td>
<td>Denver I</td>
<td>DEMO</td>
<td>08/23/93</td>
</tr>
<tr>
<td>XJDE0006</td>
<td>Atlanta</td>
<td>DEMO</td>
<td>11/13/91</td>
</tr>
<tr>
<td>XJDE0007</td>
<td>Chicago</td>
<td>DEMO</td>
<td>11/13/91</td>
</tr>
<tr>
<td>XJDE0008</td>
<td>New York</td>
<td>DEMO</td>
<td>11/13/91</td>
</tr>
<tr>
<td>XJDE0009</td>
<td>Dallas</td>
<td>DEMO</td>
<td>11/13/91</td>
</tr>
<tr>
<td>XJDE0010</td>
<td>Houston</td>
<td>DEMO</td>
<td>11/13/91</td>
</tr>
<tr>
<td>XJDE0011</td>
<td>San Francisco</td>
<td>DEMO</td>
<td>11/13/91</td>
</tr>
<tr>
<td>XJDE0012</td>
<td>Washington DC</td>
<td>DEMO</td>
<td>11/13/91</td>
</tr>
</tbody>
</table>

Opt: 1=Run 2=Chg 3=Add 4=Rpt Dist 5=Cover 6=Prt Ovr 8=Repair 9=Dlt F13=Form
The processing options for each version provides setup on exactly how to access the remote location. The following illustrates the processing options.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Destination Virtual Control Unit</strong></td>
<td>This is the control unit that the user will connect to at the remote location. The first available device on the control unit will be selected.</td>
</tr>
<tr>
<td><strong>Destination Location</strong></td>
<td>This is the APPN network name for the remote location.</td>
</tr>
<tr>
<td>(Used in AS/400 Environment)</td>
<td></td>
</tr>
<tr>
<td><strong>APPC Device(s)</strong></td>
<td>These are the APPC devices that identify the route to the remote location.</td>
</tr>
<tr>
<td>(Used in S/38 Environment)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Only one intermediate node is supported.</td>
</tr>
</tbody>
</table>
Using IBM Pass-Through with Group Jobs

To use IBM Pass-Through with Group Jobs

1. Use the J.D. Edwards menu B98P to start an IBM Pass-Through session to a remote machine.

2. Use the J.D. Edwards Menu Design Aid (G92) to attach your user defined DREAM Writer Form ID P98GRP5 versions to menu B98P.

When an option is selected on the menu, the IBM Start Pass-Through command will be executed to the remote machine, and still retain a link to the source machine group jobs.

The mechanism used to attach remote locations to the J.D. Edwards Group Job Form on the source machine is a parameter on the STRPASTHR (Start Pass-Through) command. The following illustrates the link to the source machine.
The SRQ10PGM (SysReq 10) parameter allows a program to be called on the source machine from the remote location. By entering the J.D. Edwards Group Job Form program (P98GRP) in this parameter, the J.D. Edwards Group Job Form can be displayed on the remote location by pressing SysReq 10, NOT the ATTN Key. This allows access to all suspended group jobs on the source machine and other remote locations.
**Universal File Converter**

**Objectives**

- Initially convert existing client files to J.D. Edwards data files
- Create recurring interfaces or bridges between J.D. Edwards and non-J.D. Edwards application systems

**About Universal File Converter**

There is constant change in data processing. For example, when you upgrade your J.D. Edwards software, you are changing several pieces of the software. Your data files may be greatly impacted when you upgrade. J.D. Edwards Universal File Converter will assist you in converting your data files.

Universal File Converter allows you to store conversion information for future conversions. It automatically matches data fields to be converted together.

J.D. Edwards Universal File Converter accesses standing instruction files and transfers data in fields:

- From one file to another file
- From one file to multiple files
- From multiple files to a single file
The instruction file defines the association between two files and includes data field information.
Step 1
You specify From files and To files through DREAM Writer processing options. You can specify up to four To files. If you require multiple From files, specify a join logical as the From file in the DREAM Writer “based on” file. The system returns file field information and pre-loads the Cross-Over Rules file with field name, length, size, type and reference (data dictionary name). The system pre-loads information in the Cross-Over Rules file for all fields that have the same reference (data dictionary field name) as the From file.

Step 2
You must manually associate the fields that were not automatically loaded in the Cross-Over Rules file. If you need special calculations for a field, you can specify special processing key words in the Conversion Rule field. You can also add the calculations into an external program that can be called from the converter program. The external program needs several parameters that are sent and passed back to the converter program. These parameters are: data, error, From field name, To field name, and number of To file records. You must specify the external program in the Conversion Rule field in the Cross-Over Rules file.

Step 3
In this step you specify the form ID and the version you selected in the first step. The From and To files should be the same (or exact equivalent) as the files specified in Step 1. The converter program accesses the cross-over instructions for the “From/To” combination and loads the information to arrays. The system then processes the arrays for each field that has an association. Finally, the system transfers the value in the From file to the To file.
Special Processing

Special processing procedures are available to help you in the conversion of one field to another.

To execute any of the special processing procedures listed below, you must type the appropriate key word into the From or To Conversion Rule field. This is explained in Detail Cross Over Rules, later in this guide. There are special keywords for the following.

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dates</td>
<td>The converter uses a keyword to decide what date translation is necessary.</td>
</tr>
<tr>
<td>Numeric Fields</td>
<td>The converter translates non-packed numeric data to packed data or vice versa, depending on your need. It also maintains decimal alignment, performing rounding or zero padding if required. Alphanumeric representations of numeric fields can be translated to numeric fields. Numeric fields can be translated into alphanumeric fields.</td>
</tr>
<tr>
<td>Business Unit</td>
<td>The converter processes the field through the Business Unit scrub routine. This routine right adjusts and fills the field with blanks.</td>
</tr>
<tr>
<td>Data Dictionary Default</td>
<td>The converter uses the reference field in the To file to access the data dictionary and retrieve the default value for the field.</td>
</tr>
<tr>
<td>Initialization</td>
<td>Fields in the To file are initialized to blanks for alphanumerics and zeros for numerics if no fields are defined to map to them.</td>
</tr>
<tr>
<td>Next Number</td>
<td>You can specify to have a next number value assigned to a field.</td>
</tr>
<tr>
<td>Check Data Dictionary</td>
<td>You can specify to have the value of the field validated against the data dictionary values, ranges, and user defined codes.</td>
</tr>
<tr>
<td>User Defined Code Lookup</td>
<td>Use the fields in the From file to look up a user defined code (UDC) and return the associated value in the Description 1 field as the To field value.</td>
</tr>
</tbody>
</table>
Default Constant

Specify constant value, up to six characters, for the To field value.

Database Considerations

The system creates new records in the Cross-Over Rules file for each version of cross-over rules you specify. This file contains information explaining the fields in the From file and the To file and how the two files are associated.

If the field lengths or characteristics of the files that the cross-over rules have been built upon change, you must redefine the cross-over rules. Otherwise, the rules are based on the erroneous descriptions.

The system handles extra calculations through called programs specified in the Cross-Over Rules file for each field.

User Responsibilities

You are responsible for developing and maintaining the cross-over instruction rules. If the From file or To file definition of the cross-over instructions changes, you must revise the Cross-Over Rules.

Perform the following tasks:

- Set Up Universal File Converter
- Work with Crossover Rules
- Work with File Conversion
- Print a Report
- Create Conversion Forms
- Work with the Data Dictionary Glossary by File
Set Up Universal File Converter

About Universal File Converter

If you have more than one file to convert, you can set up a separate version for each type of conversion required. The Universal File Conversion Setup program loads information to the Crossover Rules file (F0031) about the fields in the files you are converting.

The system uses the information in the Crossover Rules file to transfer the data from a field in one file to a field in another file, or to a field in multiple files.

This program also has processing options that let you convert data from both J.D. Edwards and non-J.D. Edwards files.

Before You Begin

☐ Before you run the setup procedure make sure the To files exist.

⚠️ Do not attempt to use the Universal File Converter on a file that contains “double byte” data. The converter program may corrupt the integrity of the bracketing “shift in” and “shift out” characters that are automatically inserted by double byte terminals.
Understanding the Universal File Converter Setup

The setup program is the first part of a three-part conversion process. Specify a From file and a To file through the DREAM Writer processing options. You can specify up to four To files. If you require multiple From files, specify a join logical as the From file. This join logical is over all the files you select for the From file. Use the name of the join logical in the first processing option.

The program retrieves field information for all fields in the From file and loads this information to the Crossover Rules file.

The program then retrieves field information for the To files. If the Reference (data dictionary) field in the To file matches the From file Reference field, the program makes an association between the two fields. The system writes information for the To file to the record in the Crossover Rules file associated with the From file field.

**FILLER** conversions are automatically generated for From file fields with no corresponding To file fields and for To file fields with no corresponding From file fields. You can override a **FILLER** entry with the appropriate field name, position, and characteristics if the field exists in the file but has a different field name.

If there are any other associations you need, do them manually using the Crossover Rules selection on the menu.

Setting Up Universal File Converter
To access Versions Setup

1. From the Universal File Converter menu (G9841), choose Versions Setup.

<table>
<thead>
<tr>
<th>Version</th>
<th>Description</th>
<th>User</th>
<th>Chg Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>XJDE0001</td>
<td>Generate Crossover Instruction - Sample</td>
<td>DEMO</td>
<td>07/23/93</td>
</tr>
<tr>
<td>XJDE0002</td>
<td>F0101 - Data Requirements - A7.1</td>
<td>DEMO</td>
<td>07/20/93</td>
</tr>
</tbody>
</table>

Opt: 1=Run 2=Chg 3=Add 4=Rpt Dist 5=Cover 6=Prt Ovr 8=Repair 9=Dlt

The Versions Setup form appears. The examples shown are for illustrative purposes only.

- This program loads information into the Crossover Rules File (F0031) about the fields in the files you are converting. The system uses the information in the Crossover Rules File to transfer the data from a field in one file to a field in another file or to a field in multiple files.

2. Add your own version from a Demo version and go to the processing options of your new version.

3. Once you have displayed the processing options, you must specify a From file and a To file. You can specify up to four To files. If you require multiple From files, specify a join logical as the based on file for your version. The join logical will encompass all the files you wish to use for the From file.
Generate Cross Over Instructions
This job has various options described below. Enter the desired values and press ENTER to continue.

FILE SPECIFICATION:
1. Enter the name of the file to convert the data from.
   JDE File? Y
2. Enter the name of the file OR files to convert the data to.
   File 1
   JDE File?
   File 2
   JDE File?
   File 3
   JDE File?
   File 4
   JDE File?

F5=Printer Overrides

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enter the name of the file</td>
<td>The name of the <em>From</em> file to convert the data from.</td>
</tr>
<tr>
<td>JDE File?</td>
<td>Y if the <em>From</em> file is a JDE file, or N if it is not</td>
</tr>
<tr>
<td>Enter the name of the file OR files to convert the data to.</td>
<td>The name(s) of the <em>To</em> file(s) in the spaces provided to.</td>
</tr>
<tr>
<td>JDE File?</td>
<td>Y if the <em>To</em> file is a JDE file, or N if it is not</td>
</tr>
</tbody>
</table>
Generate Cross Over Instructions
This job has various options described below. Enter the desired values and press ENTER to continue.

3. Enter the library containing the "from" file. If left blank the library list will be searched for the "from" files.

4. Enter the library containing the "To" file. If left blank the library list will be searched for the "To" file.

F5=Printer Overrides

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enter the library containing the from file.</td>
<td>The name of the From file library, or leave blank to search your library list</td>
</tr>
<tr>
<td>Enter the library containing the to file.</td>
<td>The name of the To file library, or leave blank to search your library list</td>
</tr>
</tbody>
</table>
Work with Crossover Rules

Working with the Crossover Rules Form

The Crossover Rules form lets you add, change, and delete crossover rules used in the Universal File Converter process. Use this form to set up or maintain associations between fields in the From file and the To file.

Using filler fields, you can view From file fields with no corresponding To file fields. You can also view To file fields with no corresponding From file fields.

To work with the Crossover Rules form

1. From the Universal File converter menu, choose Crossover Rules.

The form above displays illustrative data only. The From files appear on the left. The To files display on the right.

2. Complete the Crossover Rules form.
   - F8 and F9 are toggles. Press them to suppress or activate the display of the **FILLER fields in the From and To files.
- F14 is cursor-sensitive. If you are on a From file field, press F14 to enter text for that field. When the cursor is on a To file field name, press F14 and the Generic Text Form opens for that To file field name. You can also enter text for the From file and To file by placing the cursor on the appropriate field. The field name is highlighted on V0031 if generic text exists. For additional information refer to the Advanced Functions Reference Guide.

- Press F4 to display detail information in the fold area.

<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Form Id</td>
<td>For World, the RPG program name defined in the Software Versions Repository Master table. See also J.D. Edwards Standards.</td>
</tr>
<tr>
<td></td>
<td><strong>T SS XXX</strong></td>
</tr>
<tr>
<td></td>
<td>T Specific member ID number</td>
</tr>
<tr>
<td></td>
<td>SS System number (for example, 01 for Address Book)</td>
</tr>
<tr>
<td></td>
<td>XXX Member type (for example, P for Program, R for Report, and so on)</td>
</tr>
<tr>
<td></td>
<td>For OneWorld, the name of the OneWorld batch or interactive application (APPL or UBE object).</td>
</tr>
<tr>
<td>To File Name</td>
<td>The file that data is being transferred “to” in the file conversion process.</td>
</tr>
<tr>
<td>Field Name</td>
<td>The field that data is being transferred “from” in the file conversion process.</td>
</tr>
<tr>
<td>Field Name</td>
<td>The field that data is being transferred “to” in the file conversion process.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Program ID</td>
<td>For World, the RPG program name defined in the Software Versions Repository Master table. See also J.D. Edwards Standards.</td>
</tr>
<tr>
<td></td>
<td><strong>T SS XXX</strong></td>
</tr>
<tr>
<td></td>
<td>T Specific member ID number</td>
</tr>
<tr>
<td></td>
<td>SS System number, for example, 01 for Address Book</td>
</tr>
<tr>
<td></td>
<td>XXX Member type, for example, P for Program, R for Report, and so on</td>
</tr>
<tr>
<td></td>
<td>For OneWorld, the name of the OneWorld batch or interactive application (APPL or UBE object)</td>
</tr>
<tr>
<td>Foreign File Name</td>
<td>The file that data is being transferred “to” in the file conversion process.</td>
</tr>
</tbody>
</table>
### Field | Explanation
--- | ---
Domestic Field Name | The field that data is being transferred “from” in the file conversion process.
Foreign Field Name | The field that data is being transferred “to” in the file conversion process.

**Field**: The set of characters used to identify the file.  
**Bytes**: The number of bytes in the field.  
**Array N**: The number of repeating units in the array.  
**Desc Item ID**: The identifier for the description item.  
**Desc Date Last**: The date the description was last updated.  
**Desc Description**: The description of the field.  
**Desc Bus Unit**: The bus unit associated with the field.  
**Desc Item ID**: The identifier for the description item.  
**Desc Date Last**: The date the description was last updated.  
**Desc Description**: The description of the field.  
**Desc Bus Unit**: The bus unit associated with the field.  
**OPT**: Options for the field.  

#### Crossover Rules

<table>
<thead>
<tr>
<th>Field</th>
<th>T Begin ... Field ...</th>
<th>Field</th>
<th>T Begin ... Field ...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Pos</td>
<td>Bytes</td>
<td>Name</td>
</tr>
<tr>
<td>XXCC</td>
<td>A</td>
<td>47</td>
<td>12</td>
</tr>
<tr>
<td>Desc Bus Unit</td>
<td>Conv Rule</td>
<td>Desc Bus Unit</td>
<td>Conv Rule</td>
</tr>
<tr>
<td>Key Pos</td>
<td>Ref XCC</td>
<td>Array N</td>
<td>Key Pos</td>
</tr>
<tr>
<td>XXCD</td>
<td>A</td>
<td>10</td>
<td>00</td>
</tr>
<tr>
<td>Desc Bus Unit</td>
<td>Conv Rule</td>
<td>Desc Bus Unit</td>
<td>Conv Rule</td>
</tr>
<tr>
<td>Key Pos</td>
<td>Ref XCC</td>
<td>Array N</td>
<td>Key Pos</td>
</tr>
<tr>
<td>XXDT</td>
<td>S</td>
<td>41</td>
<td>04</td>
</tr>
<tr>
<td>Desc Date Last</td>
<td>Conv Rule</td>
<td>Desc Date Last</td>
<td>Conv Rule</td>
</tr>
<tr>
<td>Key Pos</td>
<td>Ref XCC</td>
<td>Array N</td>
<td>Key Pos</td>
</tr>
<tr>
<td>OPT: 9=Del</td>
<td>F4=Detail</td>
<td>F6=Add</td>
<td>F8=From Fill</td>
</tr>
</tbody>
</table>

### Field

<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>From File</td>
<td>Allows you to skip to a field in either the From file to the To file. Pressing F1 in one of these two fields will display the File Field Descriptions Window.</td>
</tr>
<tr>
<td>Version</td>
<td>For World, identifies a group of items that the system can process together, such as reports, business units, or subledgers.</td>
</tr>
<tr>
<td>Type</td>
<td>The data dictionary data type.</td>
</tr>
<tr>
<td>Begin Pos</td>
<td>The number of the beginning position of the field.</td>
</tr>
<tr>
<td>Bytes</td>
<td>The number of bytes in the field.</td>
</tr>
<tr>
<td>Dig</td>
<td>Actual number of digits in the field. In a non-packed field, this number is the same as the number of bytes.</td>
</tr>
<tr>
<td>Field</td>
<td>Explanation</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Dec</td>
<td>The number of decimal positions in the field. (Future Use).</td>
</tr>
<tr>
<td>Type</td>
<td>The type of data of the field in the “to” file. The data item types are defined in User Defined Codes, system code ‘98’, record type ‘DT’.</td>
</tr>
<tr>
<td>Begin Pos</td>
<td>The beginning position of the field in the “to” file.</td>
</tr>
<tr>
<td>Bytes</td>
<td>The number of file bytes for the field in the “to” file.</td>
</tr>
<tr>
<td>Dig</td>
<td>The actual number of digits in the “to” file field. In a non-packed field this is the same as the number of bytes.</td>
</tr>
<tr>
<td>Dec</td>
<td>The number of decimal positions in the “to” file field.</td>
</tr>
<tr>
<td>O</td>
<td>Option 9 = Delete the line.</td>
</tr>
<tr>
<td>Description</td>
<td>The description of the file field.</td>
</tr>
<tr>
<td>Domestic File Name</td>
<td>The file that data is being transferred “from” in the file conversion process.</td>
</tr>
<tr>
<td>Version</td>
<td>For World, identifies a group of items that the system can process together, such as reports, business units, or subledgers.</td>
</tr>
<tr>
<td></td>
<td>For OneWorld, the name of the version. It is created when the version is added.</td>
</tr>
<tr>
<td></td>
<td>Inquire on either a version number or the from/to tables.</td>
</tr>
<tr>
<td>Domestic Field Data Type</td>
<td>The type of data of the field in the “from” file. The data item types are defined in User Defined Codes, system code ‘98’, record type ‘DT’.</td>
</tr>
<tr>
<td>Domestic Field Begin Pos</td>
<td>The beginning position of the field in the “from” file.</td>
</tr>
<tr>
<td>Domestic Field Size In Bytes</td>
<td>The number of file bytes for the field in the “from” file.</td>
</tr>
<tr>
<td>Domestic Number Of Digits</td>
<td>The actual number of digits in the “from” file field. In a non-packed field this is the same as the number of bytes.</td>
</tr>
<tr>
<td>Domestic Field Decimal Positions</td>
<td>The number of decimal positions in the “from” file field.</td>
</tr>
<tr>
<td>Foreign Field Data Type</td>
<td>The type of data of the field in the “to” file. The data item types are defined in User Defined Codes, system code ‘98’, record type ‘DT’.</td>
</tr>
<tr>
<td>Foreign Field Begin Pos</td>
<td>The beginning position of the field in the “to” file.</td>
</tr>
<tr>
<td>Foreign Field Size In Bytes</td>
<td>The number of file bytes for the field in the “to” file.</td>
</tr>
<tr>
<td>Foreign Number of Digits</td>
<td>The actual number of digits in the “to” file field. In a non-packed field this is the same as the number of bytes.</td>
</tr>
<tr>
<td>--------------------------</td>
<td>----------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Foreign Field Decimal Positions</td>
<td>The number of decimal positions in the “to” file field.</td>
</tr>
<tr>
<td>Selection Exits</td>
<td>Selection exit codes are options and function keys that are used to perform a specific function for a selected line or form of data. The most commonly used selection exits for each program are displayed in highlighted text at the bottom of the form. To display all available selection exits, press F24. Press F1 in the Option field to display all available Options for the program.</td>
</tr>
<tr>
<td>Domestic Field Description</td>
<td>The description of the “from” file field.</td>
</tr>
<tr>
<td>Domestic Conversion Rule</td>
<td>Specifies a keyword or external program that is used for special calculations to the “from” file field before transfer of data to the “to” file. Valid keywords are listed below: Date Conversion *MDY, *DMY, *YMD, *JUL, *SYSVAL Initialization *BLANKS, *ZEROES Business Unit *RAB, right adjust blank fill Default from Dictionary *DEF User Defined Code Lookup *UDCssssrr, where ssss is System, rr is Code Type Default Constant *DFTcccccc, where ccccc is the constant. Next Number *NNssssxx, where ssss is System, xx is Number Check Data Dictionary *CHK, edits field for DD values and ranges. Alpha translation *TRANxx, where xx is language to translate to In addition to the above keywords, user developed external programs can be specified. These external programs must begin with the letter X. Refer to the Universal File Convertor Reference Manual for more information.</td>
</tr>
<tr>
<td>Foreign Field Description</td>
<td>The description of the “to” file field.</td>
</tr>
</tbody>
</table>
**Foreign Conversion Rule**

Specifies a keyword or external program that is used for special calculations to the “from” file field before transfer of data to the “to” file.

Valid keywords are listed below:

- Initialization – *BLANKS, *ZEROES
- Business Unit – *RAB, right adjust blank fill
- Default from Dictionary – *DEF
- User Defined Code Lookup – *UDCssssss, where ssss is System, ss is Code Type
- Default Constant – *DFTccccccc, where ccccc is the constant.
- Next Number – *NNssssxxx, where ssss is System, xx is Number
- Check Data Dictionary – *CHK, edits field for DD values and ranges.
- Alpha translation – *TRANxxx where xx is language to translate to

In addition to the above keywords, user developed external programs can be specified. These external programs must begin with the letter X. Refer to the Universal File Convertor Reference Manual for more information.

**Domestic Key Position**

Specifies the position in the key list for the field in the “from” file. **For future use***

**Domestic Reference Field**

The field name in the “from” file with the prefix removed. This is used for automatically pre-loading the associations between the “from” and “to” file fields.

**Domestic Field Type Array**

Designates the field as part of an array. **This field is for future use**

**Foreign Key Position**

Specifies the position in the key list for the field in the “to” file. **For future use***

**Foreign Reference Field**

The field name in the “to” file with the prefix removed. This is used for automatically pre-loading the associations between the “from” and “to” file fields.

**Foreign Field Type Array**

Designates the field as part of an array. **This field is for future use**
The left side of the form contains information about the From file.

The right side of the form contains information about the To file fields. If the setup program made associations with the To file fields, they display in the right columns when you inquire on a Form ID. Otherwise, these columns contain *FILLER information.

**What You Should Know About**

You should be aware of the following rules when you work with crossover rules.

- To review a specific set of crossover rules, enter the DREAM Writer version you used to create the rules.

- To update information on Crossover Rules form, enter the To file field, type, beginning position, number of bytes, and number of digits and decimals, if applicable. Required information is name, type, beginning position and number of bytes.

- Two “skip to” capabilities are available on this form. You can skip to a field in either the From file or the To file.

**Displaying Field Descriptions**

**To display field descriptions**

1. Press F13 in the Field Name column for the From or To file.
   - The File Field Descriptions form appears, as shown below.

   ![Field Descriptions Form](image)

   - When you use option 4 to select a field from the form, the program returns the name, type, number of bytes, number of decimals, number of
digits, description, reference, and key position to the appropriate fields on the form. For Crossovers on the File Field Descriptions window, refer to the *Computed Assisted Design Reference Guide*.

- For details on the Data Dictionary Repository form, the Glossary form, and the Cross Reference options on the File Field Description form, refer to the *Advanced Functions Reference Guide*.

2. Enter 4 in the option field. The program returns the field description to the associated field as shown in this example.

2. Enter 4 in the option field. The program returns the field description to the associated field.

---

0031  Crossover Rules

<table>
<thead>
<tr>
<th>Field</th>
<th>T Begin</th>
<th>...</th>
<th>Field</th>
<th>T Begin</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Pos</td>
<td>Bytes</td>
<td>Dig</td>
<td>Dec</td>
<td>Name</td>
</tr>
<tr>
<td><strong>FILLER</strong></td>
<td>A</td>
<td>1</td>
<td>00</td>
<td>00</td>
<td><strong>SZACOM</strong></td>
</tr>
<tr>
<td><strong>FILLER</strong></td>
<td>A</td>
<td>1</td>
<td>00</td>
<td>00</td>
<td><strong>SZAEXP</strong></td>
</tr>
<tr>
<td><strong>FILLER</strong></td>
<td>A</td>
<td>1</td>
<td>00</td>
<td>00</td>
<td><strong>SZAID</strong></td>
</tr>
<tr>
<td><strong>FILLER</strong></td>
<td>A</td>
<td>1</td>
<td>00</td>
<td>00</td>
<td><strong>SZAIL</strong></td>
</tr>
<tr>
<td><strong>FILLER</strong></td>
<td>A</td>
<td>1</td>
<td>00</td>
<td>00</td>
<td><strong>SZAITEM</strong></td>
</tr>
<tr>
<td><strong>FILLER</strong></td>
<td>A</td>
<td>1</td>
<td>00</td>
<td>00</td>
<td><strong>SZAITM</strong></td>
</tr>
<tr>
<td><strong>FILLER</strong></td>
<td>A</td>
<td>1</td>
<td>00</td>
<td>00</td>
<td><strong>SZAOPT</strong></td>
</tr>
<tr>
<td><strong>FILLER</strong></td>
<td>A</td>
<td>1</td>
<td>00</td>
<td>00</td>
<td><strong>SZAPTS</strong></td>
</tr>
<tr>
<td><strong>FILLER</strong></td>
<td>A</td>
<td>1</td>
<td>00</td>
<td>00</td>
<td><strong>SZAATM</strong></td>
</tr>
<tr>
<td><strong>FILLER</strong></td>
<td>A</td>
<td>1</td>
<td>00</td>
<td>00</td>
<td><strong>SZACDC</strong></td>
</tr>
<tr>
<td><strong>FILLER</strong></td>
<td>A</td>
<td>1</td>
<td>00</td>
<td>00</td>
<td><strong>SZCDCD</strong></td>
</tr>
</tbody>
</table>

Opt:  9=Del  F4=Detl  F6=Add  F8=From Fill  F9=To Fill  F13=File  F14=Text
Adding Fields

To add a field

1. Press F6 to open the Add Crossover Instructions form.

```
    00312   Add Crossover Instructions
    From File... F4001Z
    Field Name   
    Field Data Type
    Field Begin Pos.
    Number of Bytes
    Number of Digits
    Field Dec Pos.
    Field Description
    Conversion Rule
    To File...   F4011Z
    Field Name   
    Field Data Type
    Field Begin Pos.
    Number of Bytes
    Number of Digits
    Field Dec Pos.
    Field Description
    Conversion Rule

F3=Exit
```

2. With the cursor in the Field Name field, press F13 to open the File Field Descriptions form.
   - After you press Enter, the program returns field information to the Field Name when you exit the form.
   - The required fields for adding a field are:
     - From field name, type, number of bytes, and beginning position
     - To field name, type, number of bytes, and beginning position

The add function is available to associate a single field in the From file with multiple fields in the To file and to break apart a From field into multiple fields.

A field can exist in the To file and have nothing associated with it in the From file. In this case, the To file field is initialized as described in the section Special Processing in the Introduction of this guide.
Deleting Records

To delete a record

Choose option 9 to delete records from the Crossover Rules file.

- This cancels the From-To relationship so that no conversion takes place.
- If you blank out the To file field name, the program does not delete the record from the Crossover Rules file, but only clears the To file field information. The converter program looks only at records that have both a From and To file field name.

NOTE: You do not need to delete lines with blank (**FILLER) To file field names, they are automatically omitted.
Keywords

Keywords in the Conversion Rule field (in the fold area) trigger special processing for a field before the data is transferred. Following are the keywords that are available and a brief explanation of what processing they trigger.

With the exception of the date keywords listed below, specify conversion rules for either From field or To field, never for both.


These keywords activate a date conversion between the From file field and the To file field. You must type keywords into both the From file Conversion Rule field and the To file Conversion Rule field. Each keyword on the From field specifies how the field is stored in the From file. The keyword on the To field conveys the output format on the To field. NOTE: This does not work on packed fields.

**Business Unit – **RAB.

This keyword activates the business unit scrub of right adjust and blank fill to the From file field before moving it to the To field.

**Initialize – **ZEROES, BLANKS

These keywords move either zeroes or blanks to the From file field before it is transferred. With the initialization rules, these keywords are not required unless you want to initialize an alphanumeric field to zeroes.

**Data Dictionary Default – **DEF

This keyword retrieves the Data Dictionary default for the To file field, using the Reference field in the Data Dictionary, and loads it to the From file field before it is transferred.

**User Defined Code Lookup – **UDC<ssssrr>

This keyword retrieves the definition of the user defined code used in a specific system and loads it to the To field. When typing your request, <ssss> is the system and <rr> is the user defined code.

**Default Constant – **DFT<cccccc>

This keyword loads a default constant to the To field. When typing your request, <cccccc> is the default constant.

**Terminal ID – **TID

This keyword loads the terminal ID to the To field.
**Next Number - **

*NNssssxx.*

This keyword computes a next number and loads it to the To field. When typing your request, *ssss* is the system and *xx* is the number.

**Check Data Dictionary - **

*CHK.*

This keyword lets you edit a field against Data Dictionary values and ranges. The results of the edit print on the File Conversion report whenever any errors are detected.
About the Conversion Rule Program

Besides specifying the use of keywords in the conversion rule, you can specify an external program that runs before the data is transferred to the To file field. You must name the external program beginning with an X. For example, use an “X” program to determine a range of valid values in a From file field, excluding records based on a given field. Other examples include writing multiple To file records based on a single From file record, or manipulating the data before it is transferred.

The external program requires five parameters:

- **First parameter**: Must be 50 bytes and contains the value of the field being processed. Use it to pass back the value to the converter program when the “X” program is done with it.

- **Second parameter**: One-byte error flag. If the error flag returns blank, the data in parameter 1 from the “X” program is placed in the To file.
  - If the error flag returns with 2, the data in parameter 1 is not transferred to the To file. Use this error if you are writing multiple To file records and different From file fields are used for a single field in the To file.
  - If the error flag returns with 3, a record will not be written to the To file. Use this error if you do not want to write a record when the value of a certain field in parameter 1 is blank, zero, or not valid for your purposes.

- **Third parameter**: Four-byte alphanumeric field for the number of the To file records. The field always has numeric characters and is zero-filled. This lets your “X” program know which record the converter program will write when you are writing multiple To file records.

- **Fourth parameter**: Ten-byte field for the From file field name. This lets your “X” program know which field you are processing if multiple fields in the From file are updating a single To file field.
**Fifth parameter**

Ten-byte field for the To file field name. This lets your “X” program know which field you are processing if multiple fields in the From file are updating a single To file field.

## Available Functions and Options

### F6 – Add Instructions

F6. To add fields to be converted, press F6 to access the Add Cross Over Instructions Form. The required fields for adding a field are Field Name, Field Data Type, Field Beginning Position, and Number of Bytes.

<table>
<thead>
<tr>
<th>Crossover Rules</th>
</tr>
</thead>
<tbody>
<tr>
<td>0031</td>
</tr>
<tr>
<td>Action Code.</td>
</tr>
<tr>
<td>Version.</td>
</tr>
<tr>
<td>To File Name</td>
</tr>
<tr>
<td>Skip to.</td>
</tr>
<tr>
<td>From File.</td>
</tr>
<tr>
<td>Field Name</td>
</tr>
<tr>
<td>Field Begin Pos.</td>
</tr>
<tr>
<td>Number of Digits</td>
</tr>
<tr>
<td>Field Description</td>
</tr>
<tr>
<td>To File.</td>
</tr>
<tr>
<td>Field Name</td>
</tr>
<tr>
<td>Field Begin Pos.</td>
</tr>
<tr>
<td>Number of Bytes.</td>
</tr>
<tr>
<td>Field Description</td>
</tr>
<tr>
<td>Conversion Rule.</td>
</tr>
</tbody>
</table>

### F8 – Suppress From **FILLER** Fields

F8. Will not display those lines with **FILLER** values in the From field

### F9 – Suppress To **FILLER** Fields

F9. Will not display those lines with **FILLER** values in the To field
F13 – File Field Description

F13. Place cursor on any Field Name field and press F13 to display the File Field Description form.

F14 – User Defined Text

F14. This allows text to be entered about information on this form. The field will highlight to indicate that there is generic text associated with this field.

- Press F14 in the top area of the form to enter text about the conversion.
- Press F14 in the From Field area (left side of the form) to enter text describing the From Field.
- Press F14 in the To Field area (right side of the form) to obtain text describing the To Field.
  - The field will highlight to indicate that there is generic text associated with this field.

Option 9 – Delete Records

To delete records so that no conversion takes place, enter Option 9. If you blank out the To File Field Name, the program does not delete the record from the Cross Over Rules file (F0031), but only clears the To File Field information. The converter program will only look at records that have both a from and to file field name.
Work with File Conversion

Working with File Conversion

The File Conversion program accesses the Crossover Rules file (F0031) and transfers data fields from one file to another, from one file to multiple files, or from multiple files to one file.

► To run File Conversion

1. From the Universal File Converter menu (G9841), choose File Conversion.

When creating an execution form, be sure the Based on File and the Format Name fields contain your From File name. In addition, the Data Selection and Data Sequence forms should display fields from your From File.

2. Add your own version from a Demo version and go to the processing options of your new version.
This job has various options described below. Enter the desired values and press ENTER to continue.

FILE SPECIFICATION:
1. Enter the name of the Form ID and version containing the conversion specifications.
   Form ID                                      P00120
   Version                                      APCS

2. Enter the name and library of the "from" file, if different than the Form ID and version containing the conversion specifications.
   From File name                               
   From File library                            
   ** Caution - file must be the same field format as file used to generate rules. +

F5=Printer Overrides

<table>
<thead>
<tr>
<th>Option</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enter the name of the Form ID and version containing the Initial Setup step.</td>
<td>Type your Form ID and version from the conversion specifications.</td>
</tr>
<tr>
<td>Enter the name and library of “from” file, if different than the Form ID and version specified.</td>
<td>Type the name of the From file and library, if it is different than the From file and library in the Form ID and version specified above.</td>
</tr>
</tbody>
</table>
This job has various options described below. Enter the desired values and press ENTER to continue.

3. Enter the name of the file OR files to convert the data to. Leave blank to convert all files in setup specifications.

   File 1
   File 2
   File 3
   File 4

4. Enter the library the "to" files are in. If left blank, the library list will be searched for the "to" files.

F5=Printer Overrides

---

<table>
<thead>
<tr>
<th>Option</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enter the name of the file OR files to convert the data to.</td>
<td>Type the name(s) of the To file(s). Up to four files can be specified. If these fields are left blank, all files entered in the setup version are converted.</td>
</tr>
<tr>
<td>Enter the library the to files are in.</td>
<td>Type the name of the library containing the To file(s), or leave blank to have the library list searched.</td>
</tr>
</tbody>
</table>

---

5. Enter a '1' to clear the file data is being transferred to.

6. Enter the number of "to" file records to be created for each "from" file record. If left blank, a single "to" file record will be created for each "from" file record.

   File 1
   File 2
   File 3
   File 4

Bottom

F5=Printer Overrides
Enter a “1” to clear the file data is being transferred to. Enter “1” to clear the To file. The To file will be filled only with converted records. If this field is left blank, the converted data records are added to the To file.

Enter the number of to file records to be created for each from file record. Enter the number of To file records you want to create for each From file record. If this field is left blank, only a single To file record will be created for each From file record.

- If you are using multiple From file(s), remember to create a join logical over all the From files you wish to use.
- When adding a new version, you should check to see that the format name under Additional Parameters is correct for the based on file.

3. Enter the correct values on Processing Options and submit your version to complete the conversion process.

What You Should Know About

- If you are using multiple From files, remember to create a join logical over all the From files you want to use.
- When adding a new version, check to see that the format name for the based-on file is correct for the file. The default is lxxxx and may not be appropriate.
- The From file name and the To file names should be the same as used to set up the conversion rules in Step 1.
- You can use DREAM Writer data selection to specify which records in the From file are to be converted. For example, convert one branch or one company only.
- A printed report lists error conditions detected by *CHK keyword and lists the total number of records read and number of records converted. The report lists the description of the errors. Depending on the error condition, you may need to correct the values in the incoming data and rerun the conversion.
Print a Report

Printing a Report

You can print a report that displays the Cross Over Rules and any associated generic text.

To print a report

1. From Universal File Converter, choose Report

The form can list different versions of the File Converter Report. The example shown is for illustrative purposes only.

2. On the Data Selection form, specify your Form ID and version.
This job has various options described below. Enter the desired values and press ENTER to continue.

1) Enter a "1" to print Data Dictionary Glossary for each item. Leave blank to not print the Data Dict. Glossary. (Prints for "TO" fields only)

2) Enter a "1" to print File Specific Glossary for each data item. Leave blank to not print. (Prints for "TO" fields only)

3) Enter a "1" to print the Generic Text Instructions for each data item. Leave blank to not print the Generic Text. (Prints for both "FROM" and "TO" fields)

3. Select one of the following print options:

<table>
<thead>
<tr>
<th>Field</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enter a “1” to print Data Dictionary Glossary for each item.</td>
<td>Prints Data Dictionary Glossary for each To field.</td>
</tr>
<tr>
<td>Enter “1” to print File Specific Glossary for each item.</td>
<td>Prints file specific glossary from Generic Text file (F00163) for each To file.</td>
</tr>
<tr>
<td>Enter “1” to print generic text instructions for each item</td>
<td>Prints any generic text associated with either To fields or From fields.</td>
</tr>
</tbody>
</table>
Create Conversion Forms

Creating Conversion Forms

The Universal File Converter helps you create conversion forms that you may want to use for planning purposes when you convert your non-JDE files into JDE files.

- Start by creating a form that specifies the major file in the “Convert to” file. The name of the file you convert from is intentionally left blank. This lets you create a blank set of conversion rules which you can print using the Report selection.

- JDE supplies a special Data Dictionary glossary relating to specific fields in specific files in your JDE Data Dictionary text. You can also create new field descriptions that better correspond to your system by pressing F14 for generic text in the crossover rules revisions.

- If you decide to use the blank version (described above) for actual file conversion, type the From file specifications corresponding to the appropriate To field using the Crossover Rules. Be sure to override the From file before you execute the conversion program.
Creating Conversion Forms

To create a conversion form

1. From the Universal File Converter menu, select either Versions Setup or Report.

2. Complete the Processing Options Revisions form
   - If you selected Versions Setup, be sure to leave the first processing option blank under File Conversion.
   - In the second option, type the name of the files you want to convert, and then Y if they are JDE files or N if they are not.
This job has various options described below. Enter the desired values and press ENTER to continue.

1) Enter a "1" to print Data Dictionary Glossary for each item. Leave blank to not print the Data Dict. Glossary. (Prints for "TO" fields only)

2) Enter a "1" to print File Specific Glossary for each data item. Leave blank to not print. (Prints for "TO" fields only)

3) Enter a "1" to print the Generic Text Instructions for each data item. Leave blank to not print the Generic Text. (Prints for both "FROM" and "TO" fields)

If you select Report, type 1 next to all three options as shown above.
Work with the Data Dictionary Glossary by File

About Working with the Data Dictionary Glossary by File

When using the Universal File Converter, small details often differ for each file. Keeping these details clear, especially when the conversion form might be used by another department, is a potential problem. To remedy this, J.D. Edwards has made it possible to attach Data Dictionary glossary text to each data item that explains the details particular to that specific file.

To work with the Data Dictionary Glossary by file perform the following tasks:

- Access the Data Dictionary Glossary by file
- Add a file specific glossary item
- Print the Data Dictionary Glossary information
Accessing the Data Dictionary Glossary by File

To access the Data Dictionary Glossary by file

1. From the Universal File Converter menu, type DD and press Enter.
   - The Data Dictionary Repository screen appears.

2. Press F10 to display the glossary definition of the data item you selected.

Use the Data Item Glossary Revisions form to change the glossary text for a Data Dictionary item or to add a File-Specific glossary item.
Adding a File Specific Glossary Item

To add a File Specific Glossary item

1. Type A in the Action Code field.
2. Type the file name in the Scrn/Rpt field.
3. Type the new text and press Enter.

Printing the Data Dictionary Glossary Information

To print the Data Dictionary Glossary information

2. Complete the Processing Options Revisions form.
   - Type 1 next to all three options to print the Data Dictionary glossary.
   - Option 2 prints the File-Specific glossary text.
Appendix A - Common & Production Library Files

This appendix lists the files that are automatically created in the common and production libraries during the installation process.

Chart A - Common Library Files Automatically Created by J.D. Edwards Build Programs

The following chart contains files automatically generated as a result of a build program that J.D. Edwards offers from a menu. It is recommended that these files be maintained in your common library.

<table>
<thead>
<tr>
<th>File Name</th>
<th>File Description</th>
<th>System Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>F98FRF@</td>
<td>Field Reference - “@” Data Items</td>
<td>98</td>
</tr>
<tr>
<td>F98FRFS</td>
<td>Field Reference - “$” Data Items</td>
<td>98</td>
</tr>
<tr>
<td>F98FRFA thru</td>
<td>Field Reference - “A” Data Items through</td>
<td>98</td>
</tr>
<tr>
<td>F98FRFZ</td>
<td>Field Reference - “Z” Data Items</td>
<td>98</td>
</tr>
</tbody>
</table>
Chart B - Physical and Logical Files Created in a Common Library

The following chart shows the physical and the logical files that were created in a Common Library if one was specified for the Create User Data Libraries selection on menu A9645. Logical Files contain no data. Therefore, data copied is N.

<table>
<thead>
<tr>
<th>File Name</th>
<th>File Description</th>
<th>Copy Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>F0002</td>
<td>Next Numbers - Automatic</td>
<td>Y</td>
</tr>
<tr>
<td>F0004</td>
<td>User Defined Code Types</td>
<td>Y</td>
</tr>
<tr>
<td>F0004D</td>
<td>User Defined Codes - Alternate Language Desc</td>
<td>Y</td>
</tr>
<tr>
<td>F0005</td>
<td>User Defined Codes</td>
<td>Y</td>
</tr>
<tr>
<td>F0005D</td>
<td>User Defined Codes - Alternate Language Desc</td>
<td>Y</td>
</tr>
<tr>
<td>F0005LA</td>
<td>LF - System Code, Desc Title Type, Desc., Desc Title</td>
<td>N</td>
</tr>
<tr>
<td>F0016</td>
<td>Generic Text File</td>
<td>N</td>
</tr>
<tr>
<td>F00161</td>
<td>Generic Text Window Definition File</td>
<td>Y</td>
</tr>
<tr>
<td>F00162</td>
<td>Generic Text Key Definition File</td>
<td>Y</td>
</tr>
<tr>
<td>F00163</td>
<td>Generic Text Key Index File</td>
<td>N</td>
</tr>
<tr>
<td>F00163LA</td>
<td>Generic Text Key Index File - LF By Key Serial Number</td>
<td>N</td>
</tr>
<tr>
<td>F00164</td>
<td>Generic Text Key Index File (120 character key)</td>
<td>N</td>
</tr>
<tr>
<td>F00164LA</td>
<td>Generic Text Key Index File - LF by Key Serial Number</td>
<td>N</td>
</tr>
<tr>
<td>F0082</td>
<td>Menu Master</td>
<td>Y</td>
</tr>
<tr>
<td>F00821</td>
<td>Menu Selection Detail</td>
<td>Y</td>
</tr>
<tr>
<td>F0083</td>
<td>Menu Selection Text</td>
<td>Y</td>
</tr>
<tr>
<td>F0082H</td>
<td>Menu Selection History</td>
<td>N</td>
</tr>
<tr>
<td>F0090HL@</td>
<td>LF - Combined Sequences</td>
<td>N</td>
</tr>
<tr>
<td>F0090L@</td>
<td>LF - Job To Execute</td>
<td>N</td>
</tr>
<tr>
<td>F009141</td>
<td>Word Search Occurrences Master</td>
<td>Y</td>
</tr>
<tr>
<td>F009141S</td>
<td>Word Search Occurrences Master - Dist Supplemental</td>
<td>N</td>
</tr>
<tr>
<td>F009190</td>
<td>Word Search Occurrences Master</td>
<td>Y</td>
</tr>
<tr>
<td>F009191</td>
<td>Question &amp; Answer Search Occurrence Master</td>
<td>Y</td>
</tr>
<tr>
<td>F009198</td>
<td>Question &amp; Answer Search Occurrence Master</td>
<td>Y</td>
</tr>
<tr>
<td>F0095</td>
<td>Open File Directory</td>
<td>Y</td>
</tr>
</tbody>
</table>
### Appendix A - Common & Production Library Files

<table>
<thead>
<tr>
<th>File Name</th>
<th>File Description</th>
<th>Copy Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>F009690</td>
<td>Menu Word Search Master</td>
<td>Y</td>
</tr>
<tr>
<td>F009690LA</td>
<td>LF - By Key and Search Word</td>
<td>N</td>
</tr>
<tr>
<td>F009691</td>
<td>Question &amp; Answer Word Search Master</td>
<td>Y</td>
</tr>
<tr>
<td>F009691LA</td>
<td>LF - By Key and Search Word</td>
<td>N</td>
</tr>
<tr>
<td>F009698</td>
<td>Word Search Master - Question and Answer Data Base</td>
<td>N</td>
</tr>
<tr>
<td>F009698LA</td>
<td>LF - By Key and Search Word</td>
<td>N</td>
</tr>
<tr>
<td>F009790</td>
<td>Word Search Verbs</td>
<td>Y</td>
</tr>
<tr>
<td>F0098</td>
<td>ASI Master File</td>
<td>Y</td>
</tr>
<tr>
<td>F0098LA</td>
<td>LF - System Code, Job to Execute</td>
<td>N</td>
</tr>
<tr>
<td>F0098LB</td>
<td>LF - Release, Type, System Code</td>
<td>N</td>
</tr>
<tr>
<td>F12601</td>
<td>WF - STAR</td>
<td>Y</td>
</tr>
<tr>
<td>F12601LA</td>
<td>LF - SK01 through SK09</td>
<td>N</td>
</tr>
<tr>
<td>F12601LB</td>
<td>LF - STAR Logical Over Workfile</td>
<td>N</td>
</tr>
<tr>
<td>F12603</td>
<td>STAR General Specifications Master File</td>
<td>Y</td>
</tr>
<tr>
<td>F12603LA</td>
<td>LF - STAR General Specifications Master File</td>
<td>N</td>
</tr>
<tr>
<td>F12604</td>
<td>STAR - Column Specification Master File</td>
<td>Y</td>
</tr>
<tr>
<td>F12605</td>
<td>STAR - Row Specifications Master File</td>
<td>Y</td>
</tr>
<tr>
<td>F12606</td>
<td>STAR - Cell Specifications Master File</td>
<td>Y</td>
</tr>
<tr>
<td>F12607</td>
<td>STAR - Row Creation File</td>
<td>Y</td>
</tr>
<tr>
<td>F12608</td>
<td>WF - STAR - Balance Auditor</td>
<td>Y</td>
</tr>
<tr>
<td>F12609</td>
<td>STAR - Print Image File</td>
<td>Y</td>
</tr>
<tr>
<td>F81900</td>
<td>DREAM Writer - Performance Statistics Master</td>
<td>Y</td>
</tr>
<tr>
<td>F81900LA</td>
<td>DREAM Writer - Performance Statistics</td>
<td>N</td>
</tr>
<tr>
<td>F81901</td>
<td>DREAM Writer Statistics Detail</td>
<td>Y</td>
</tr>
<tr>
<td>F81901LA</td>
<td>LF - File and Keys</td>
<td>N</td>
</tr>
<tr>
<td>F81902</td>
<td>DREAM Writer - Statistics Detail Accumulator</td>
<td>Y</td>
</tr>
<tr>
<td>F83JOIN</td>
<td>FASTR - Format File for Open Query Dynamic Join</td>
<td>Y</td>
</tr>
<tr>
<td>F83JOINA</td>
<td>FASTR - Format File for Open Query Dynamic Join</td>
<td>Y</td>
</tr>
<tr>
<td>F83JOINB</td>
<td>FASTR - Format File for Open Query Dynamic Join</td>
<td>Y</td>
</tr>
<tr>
<td>F83WORK</td>
<td>FASTR - Work File Save Data</td>
<td>Y</td>
</tr>
<tr>
<td>File Name</td>
<td>File Description</td>
<td>Copy Data</td>
</tr>
<tr>
<td>------------</td>
<td>------------------------------------------------------</td>
<td>-----------</td>
</tr>
<tr>
<td>F83WORKB</td>
<td>FASTR - Work File Save Data</td>
<td>Y</td>
</tr>
<tr>
<td>F8301</td>
<td>WF - FASTR</td>
<td>Y</td>
</tr>
<tr>
<td>F8302</td>
<td>WF - Level of Detail</td>
<td>Y</td>
</tr>
<tr>
<td>F8303</td>
<td>FASTR General Specifications Master File</td>
<td>Y</td>
</tr>
<tr>
<td>F8303LA</td>
<td>LF - FASTR General Specifications Master File</td>
<td>N</td>
</tr>
<tr>
<td>F8304</td>
<td>FASTR - Column Specifications Master File</td>
<td>Y</td>
</tr>
<tr>
<td>F8305</td>
<td>FASTR - Row Specifications Master File</td>
<td>Y</td>
</tr>
<tr>
<td>F8306</td>
<td>FASTR - Cell Specifications Master File</td>
<td>Y</td>
</tr>
<tr>
<td>F8307</td>
<td>FASTR - Row Creation File</td>
<td>Y</td>
</tr>
<tr>
<td>F8308</td>
<td>WF - FASTR - Balance Auditor</td>
<td>Y</td>
</tr>
<tr>
<td>F8309</td>
<td>FASTR - Print Image File</td>
<td>Y</td>
</tr>
<tr>
<td>F8310</td>
<td>WF - FASTR - Balance Auditor</td>
<td>Y</td>
</tr>
<tr>
<td>F8350</td>
<td>FASTR - Cost Center Organizational Chart</td>
<td>Y</td>
</tr>
<tr>
<td>F8410</td>
<td>DDP Routing Master</td>
<td>Y</td>
</tr>
<tr>
<td>F8415</td>
<td>DDP Transfer File Setup</td>
<td>Y</td>
</tr>
<tr>
<td>F9200</td>
<td>Data Item Master</td>
<td>Y</td>
</tr>
<tr>
<td>F9200JA</td>
<td>JF - Data Item (F9203 F9200)</td>
<td>N</td>
</tr>
<tr>
<td>F9200JB</td>
<td>JF - Data Item (F9200 F9205) Error Messages Only</td>
<td>N</td>
</tr>
<tr>
<td>F9200JC</td>
<td>JF - Data Item (F9203 F9200)</td>
<td>N</td>
</tr>
<tr>
<td>F9200JD</td>
<td>JF - Data Item (F9201 F9200)</td>
<td>N</td>
</tr>
<tr>
<td>F9200LA</td>
<td>LF - Glossary Group, Data Item</td>
<td>N</td>
</tr>
<tr>
<td>F9200LB</td>
<td>LF - System Code, Data Item</td>
<td>N</td>
</tr>
<tr>
<td>F9201</td>
<td>Data Field Specifications</td>
<td>Y</td>
</tr>
<tr>
<td>F9201JA</td>
<td>JF - Data Item (F9202 F9201)</td>
<td>N</td>
</tr>
<tr>
<td>F9201LA</td>
<td>LF - Data Edit Rule, ER Spec 1, ER Spec 2</td>
<td>N</td>
</tr>
<tr>
<td>F9201LB</td>
<td>LF - Data Item Class, Data Item</td>
<td>N</td>
</tr>
<tr>
<td>F9202</td>
<td>Data Field Display Text</td>
<td>Y</td>
</tr>
<tr>
<td>F9203</td>
<td>Data Item Alpha Descriptions</td>
<td>Y</td>
</tr>
<tr>
<td>F9204</td>
<td>Data Item Aliases</td>
<td>Y</td>
</tr>
<tr>
<td>F9204LA</td>
<td>LF - Alias Type, Alias, Data Item</td>
<td>Y</td>
</tr>
<tr>
<td>File Name</td>
<td>File Description</td>
<td>Copy Data</td>
</tr>
<tr>
<td>-----------</td>
<td>-----------------</td>
<td>-----------</td>
</tr>
<tr>
<td>F9205</td>
<td>Data Dictionary - Error Message Program ID</td>
<td>Y</td>
</tr>
<tr>
<td>F9220</td>
<td>Screen/Report Text Master</td>
<td>Y</td>
</tr>
<tr>
<td>T9220</td>
<td>Screen/Report Text Master</td>
<td>Y</td>
</tr>
<tr>
<td>F92710</td>
<td>Action Diagramming Translation Master</td>
<td>Y</td>
</tr>
<tr>
<td>F93000</td>
<td>Model Program Definition Master</td>
<td>Y</td>
</tr>
<tr>
<td>F93000LA</td>
<td>LF - Model Program Definition - X-Ref</td>
<td>N</td>
</tr>
<tr>
<td>F93001</td>
<td>Source Code Inventory Master</td>
<td>Y</td>
</tr>
<tr>
<td>F93001LA</td>
<td>LF - Primary Source Key</td>
<td>N</td>
</tr>
<tr>
<td>F93002</td>
<td>Additional Help/Modifications Master</td>
<td>N</td>
</tr>
<tr>
<td>F93002LA</td>
<td>LF - Primary, Secondary and Serial Number</td>
<td>N</td>
</tr>
<tr>
<td>F93003</td>
<td>WF - Source Merge Monitor</td>
<td>N</td>
</tr>
<tr>
<td>F93004</td>
<td>User Defined Entry Point Source Code Master</td>
<td>Y</td>
</tr>
<tr>
<td>F93101</td>
<td>General Purpose/Type Parameters</td>
<td>N</td>
</tr>
<tr>
<td>F93101LA</td>
<td>LF - Program ID by Program Type</td>
<td>N</td>
</tr>
<tr>
<td>F93102</td>
<td>File Specifications</td>
<td>N</td>
</tr>
<tr>
<td>F93103</td>
<td>Data Base Format Parameters</td>
<td>N</td>
</tr>
<tr>
<td>F93103LA</td>
<td>LF - Program ID, Format Name, File Name</td>
<td>N</td>
</tr>
<tr>
<td>F93104</td>
<td>Program Exit Parameters</td>
<td>N</td>
</tr>
<tr>
<td>F93105</td>
<td>Detail Program Logic Parameters</td>
<td>N</td>
</tr>
<tr>
<td>F93105LA</td>
<td>LF - Program ID, Data Field Name</td>
<td>N</td>
</tr>
<tr>
<td>F93105LB</td>
<td>LF - Program ID, File Name, Key Position</td>
<td>N</td>
</tr>
<tr>
<td>F93105LC</td>
<td>LF - Program ID, Clear After, Field Name</td>
<td>N</td>
</tr>
<tr>
<td>F93105LD</td>
<td>LF - Program ID, Field Type, Field Name</td>
<td>N</td>
</tr>
<tr>
<td>F93105LE</td>
<td>LF - Program ID, Data Field Name</td>
<td>N</td>
</tr>
<tr>
<td>F93105LF</td>
<td>LF - Program ID, Data Field Name</td>
<td>N</td>
</tr>
<tr>
<td>F93105LG</td>
<td>LF - Program ID, Data Field Name</td>
<td>N</td>
</tr>
<tr>
<td>F93105LH</td>
<td>LF - Data Field Parameters LF - #DICT, #DDFTY</td>
<td>N</td>
</tr>
<tr>
<td>F93105LI</td>
<td>LF - Program ID, Parameter Sequence</td>
<td>N</td>
</tr>
<tr>
<td>F93105LJ</td>
<td>LF - Program ID, Field Name</td>
<td>N</td>
</tr>
<tr>
<td>F93105LK</td>
<td>LF - Program ID, Field Name, File Name</td>
<td>N</td>
</tr>
<tr>
<td>File Name</td>
<td>File Description</td>
<td>Copy Data</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------------------------------------------------</td>
<td>-----------</td>
</tr>
<tr>
<td>F93105LL</td>
<td>LF - Program ID, Data Item</td>
<td>N</td>
</tr>
<tr>
<td>F93106</td>
<td>Automatic Accounting Instruction Parameters</td>
<td>N</td>
</tr>
<tr>
<td>F93107</td>
<td>Print Control Parameters</td>
<td>N</td>
</tr>
<tr>
<td>F93108</td>
<td>Operation Code to Logic Module X-Ref</td>
<td>Y</td>
</tr>
<tr>
<td>F93109</td>
<td>User Defined Procedures</td>
<td>N</td>
</tr>
<tr>
<td>F93110</td>
<td>User Defined Procedures Detail</td>
<td>N</td>
</tr>
<tr>
<td>F93111</td>
<td>User Defined Procedures Work Field Definition</td>
<td>N</td>
</tr>
<tr>
<td>F93112</td>
<td>User Defined Entry Point Definition</td>
<td>N</td>
</tr>
<tr>
<td>F93201</td>
<td>Key List File Maintenance</td>
<td>Y</td>
</tr>
<tr>
<td>F9501</td>
<td>Unattended Operations Scheduling Master</td>
<td>N</td>
</tr>
<tr>
<td>F9501LA</td>
<td>LF - System, Pgm ID, Library, User</td>
<td>N</td>
</tr>
<tr>
<td>F9501LB</td>
<td>LF - Pgm ID, Library, User</td>
<td>N</td>
</tr>
<tr>
<td>F9501LC</td>
<td>LF - Library, Pgm ID, User</td>
<td>N</td>
</tr>
<tr>
<td>F9501LD</td>
<td>LF - User, Pgm ID, Library</td>
<td>N</td>
</tr>
<tr>
<td>F9501LE</td>
<td>LF - Execution Date, Execution Time</td>
<td>N</td>
</tr>
<tr>
<td>F9601</td>
<td>Function Key Translation Master</td>
<td>Y</td>
</tr>
<tr>
<td>F9601D</td>
<td>Function Key Definitions - Alternate Language Desc</td>
<td>Y</td>
</tr>
<tr>
<td>F9611</td>
<td>Function Key Translation Detail</td>
<td>Y</td>
</tr>
<tr>
<td>F9611LA</td>
<td>LF - Function Key Field Name, Screen Name</td>
<td>N</td>
</tr>
<tr>
<td>F9612</td>
<td>Function Key Security</td>
<td>N</td>
</tr>
<tr>
<td>F9612LA</td>
<td>LF - Function Key Security</td>
<td>N</td>
</tr>
<tr>
<td>F9620</td>
<td>Cursor Sensitive Control Master</td>
<td>Y</td>
</tr>
<tr>
<td>F9620LA</td>
<td>LF - File, Field, and Format</td>
<td>N</td>
</tr>
<tr>
<td>F9620LB</td>
<td>LF - File, Format, and Field</td>
<td>N</td>
</tr>
<tr>
<td>F9621</td>
<td>Cursor Control Format Master Maintenance</td>
<td>Y</td>
</tr>
<tr>
<td>F9621LA</td>
<td>LF - By Formats</td>
<td>N</td>
</tr>
<tr>
<td>F9701</td>
<td>ASI SAR Information Master File</td>
<td>Y</td>
</tr>
<tr>
<td>F98HELP</td>
<td>Help Instructions Master File</td>
<td>Y</td>
</tr>
<tr>
<td>F98HEPLA</td>
<td>LF - Help Instructions Master File</td>
<td>N</td>
</tr>
<tr>
<td>F9800Y</td>
<td>Data Dictionary (Field Reference)</td>
<td>Y</td>
</tr>
<tr>
<td>File Name</td>
<td>File Description</td>
<td>Copy Data</td>
</tr>
<tr>
<td>------------</td>
<td>-------------------------------------------------------</td>
<td>-----------</td>
</tr>
<tr>
<td>F98001</td>
<td>Cross-Reference Relationships</td>
<td>N</td>
</tr>
<tr>
<td>F98001LA</td>
<td>LF - Cross-Reference Relationships</td>
<td>N</td>
</tr>
<tr>
<td>F98001LB</td>
<td>LF - Cross-Reference Relationships</td>
<td>N</td>
</tr>
<tr>
<td>F98001LC</td>
<td>LF - Cross-Reference Relationships</td>
<td>N</td>
</tr>
<tr>
<td>F98001LD</td>
<td>LF - Cross-Reference Relationships</td>
<td>N</td>
</tr>
<tr>
<td>F98002</td>
<td>Cross-Reference File Information</td>
<td>N</td>
</tr>
<tr>
<td>F98002LA</td>
<td>LF - Cross-Reference Relationships</td>
<td>N</td>
</tr>
<tr>
<td>F98002LB</td>
<td>LF - Cross-Reference Relationships</td>
<td>N</td>
</tr>
<tr>
<td>F98002LC</td>
<td>LF - Cross-Reference Relationships</td>
<td>N</td>
</tr>
<tr>
<td>F98003</td>
<td>Cross-Program Field Information</td>
<td>N</td>
</tr>
<tr>
<td>F98003LA</td>
<td>LF - Cross-Reference Program Field Information</td>
<td>N</td>
</tr>
<tr>
<td>F98003LB</td>
<td>LF - Cross-Reference Program Field Information</td>
<td>N</td>
</tr>
<tr>
<td>F98003LC</td>
<td>LF - Cross-Reference Program Field Information</td>
<td>N</td>
</tr>
<tr>
<td>F98009</td>
<td>CASE Profiles File</td>
<td>N</td>
</tr>
<tr>
<td>F9801</td>
<td>Software Versions Repository Master</td>
<td>Y</td>
</tr>
<tr>
<td>F9801JA</td>
<td>JF - Member ID (F9801, F9802)</td>
<td>N</td>
</tr>
<tr>
<td>F9801L@</td>
<td>LF - Functional Usage/System/Function/Member ID</td>
<td>N</td>
</tr>
<tr>
<td>F9801LA</td>
<td>LF - Future Planning - Software Inventory Master</td>
<td>N</td>
</tr>
<tr>
<td>F9801LB</td>
<td>LF - Functional Usage/System/Function/Member ID</td>
<td>N</td>
</tr>
<tr>
<td>F9801LC</td>
<td>Software Inventory Master Logical-Sys, Base, MID</td>
<td>N</td>
</tr>
<tr>
<td>F9801LD</td>
<td>LF - Function Code, Member ID</td>
<td>N</td>
</tr>
<tr>
<td>F9801LE</td>
<td>LF - File Prefix, Member ID</td>
<td>N</td>
</tr>
<tr>
<td>F9801LF</td>
<td>LF - Function Code, System Code, Member ID</td>
<td>N</td>
</tr>
<tr>
<td>F9801LG</td>
<td>LF - Member ID</td>
<td>N</td>
</tr>
<tr>
<td>F9801LH</td>
<td>LF - Member Suffix, Member ID</td>
<td>N</td>
</tr>
<tr>
<td>F9801LI</td>
<td>LF - Reporting System, Member Suffix, Member ID</td>
<td>N</td>
</tr>
<tr>
<td>F9801LJ</td>
<td>LF - Member ID (System Code=2 bytes)</td>
<td>N</td>
</tr>
<tr>
<td>F98012</td>
<td>SVR Member Category Codes</td>
<td>N</td>
</tr>
<tr>
<td>F98013</td>
<td>SVR Member Parm/Key List</td>
<td>N</td>
</tr>
<tr>
<td>F9802</td>
<td>Software Versions Repository Detail</td>
<td>Y</td>
</tr>
<tr>
<td>File Name</td>
<td>File Description</td>
<td>Copy Data</td>
</tr>
<tr>
<td>---------------</td>
<td>-------------------------------------------------------</td>
<td>-----------</td>
</tr>
<tr>
<td>F9802LA</td>
<td>LF - SAR/MID</td>
<td>N</td>
</tr>
<tr>
<td>F9802LB</td>
<td>LF - Version/Type</td>
<td>N</td>
</tr>
<tr>
<td>F9805</td>
<td>Printer File Creation Parameters</td>
<td>Y</td>
</tr>
<tr>
<td>F98100</td>
<td>Report Writer Combined Versions List</td>
<td>Y</td>
</tr>
<tr>
<td>F9816</td>
<td>Data Dictionary Generic Text File</td>
<td>N</td>
</tr>
<tr>
<td>F98163</td>
<td>Data Dictionary Generic Text Key Index File</td>
<td>N</td>
</tr>
<tr>
<td>F98163LA</td>
<td>DD Generic Text Key Index File - LF by Key Serial Num</td>
<td>N</td>
</tr>
<tr>
<td>F98200</td>
<td>Report Writer Version Selection Definition</td>
<td>Y</td>
</tr>
<tr>
<td>F98301</td>
<td>DREAM Writer Master Parameter</td>
<td>Y</td>
</tr>
<tr>
<td>F98301LA</td>
<td>LF - Record Type, Program, Version and Sequence No</td>
<td>N</td>
</tr>
<tr>
<td>F98301LB</td>
<td>LF - Program ID, Version, Type, Prompt Line - Window</td>
<td>N</td>
</tr>
<tr>
<td>F98301LC</td>
<td>LF - Program ID, Version, Option #</td>
<td>N</td>
</tr>
<tr>
<td>F98301LD</td>
<td>LF - Key on Form Id &amp; FldName</td>
<td>N</td>
</tr>
<tr>
<td>F98302</td>
<td>DREAM Writer - Processing Options (Language Pref)</td>
<td>Y</td>
</tr>
<tr>
<td>F98302LA</td>
<td>DREAM Writer - Processing Options (Language Pref)</td>
<td>Y</td>
</tr>
<tr>
<td>F98303</td>
<td>DREAM Writer - Version Headings (Language Pref)</td>
<td>Y</td>
</tr>
<tr>
<td>F9831</td>
<td>DREAM Writer Values Parameter</td>
<td>Y</td>
</tr>
<tr>
<td>F98311</td>
<td>DREAM Writer - Headings File</td>
<td>Y</td>
</tr>
<tr>
<td>F98312</td>
<td>DREAM Writer - Printer Overrides</td>
<td>Y</td>
</tr>
<tr>
<td>F98501</td>
<td>Dialogue Description Master</td>
<td>Y</td>
</tr>
<tr>
<td>F98501LA</td>
<td>LF - Dialogue Type, Member, Data Item</td>
<td>N</td>
</tr>
<tr>
<td>F98501LB</td>
<td>LF - Data Item, Member ID</td>
<td>N</td>
</tr>
<tr>
<td>F98510</td>
<td>Dialogue Question Master</td>
<td>Y</td>
</tr>
<tr>
<td>F98511</td>
<td>Dialogue Question Responses</td>
<td>Y</td>
</tr>
<tr>
<td>F98511LA</td>
<td>LF - Keys: Mid, Dtai, Nxts</td>
<td>N</td>
</tr>
<tr>
<td>F98519</td>
<td>Dialogue *LIST Responses</td>
<td>Y</td>
</tr>
<tr>
<td>F98520</td>
<td>Quiz History Master</td>
<td>N</td>
</tr>
<tr>
<td>F98521</td>
<td>Quiz History Detail</td>
<td>N</td>
</tr>
</tbody>
</table>
Chart C - Physical and Logical Files Created in Production Library with Data

The following chart shows the physical and the logical files that were created in Production Library with data. Logical files contain no data, therefore data copied is N.

<table>
<thead>
<tr>
<th>File Name</th>
<th>File Description</th>
<th>Copy Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>F0009</td>
<td>General Constants</td>
<td>Y</td>
</tr>
<tr>
<td>F0010</td>
<td>Company Constants</td>
<td>Y</td>
</tr>
<tr>
<td>F0012</td>
<td>Automatic Accounting Instructions Master</td>
<td>Y</td>
</tr>
<tr>
<td>F0012LA</td>
<td>LF - Sequence No., Item No., Company</td>
<td>N</td>
</tr>
<tr>
<td>F0012LB</td>
<td>LF - System, Sequence No., Item No., Com</td>
<td>N</td>
</tr>
<tr>
<td>F06211</td>
<td>Payroll Cycle Version File</td>
<td>Y</td>
</tr>
<tr>
<td>F063920</td>
<td>Payroll Archive Version File</td>
<td>Y</td>
</tr>
<tr>
<td>F06723</td>
<td>W-2 Audit Report File</td>
<td>Y</td>
</tr>
<tr>
<td>F06723LA</td>
<td>W-2 Audit Report File (vers)</td>
<td>Y</td>
</tr>
<tr>
<td>F069016</td>
<td>Tax Area Constant</td>
<td>Y</td>
</tr>
<tr>
<td>F069016A</td>
<td>LF - Tax Area Code</td>
<td>N</td>
</tr>
<tr>
<td>F069016B</td>
<td>LF - Tax Area Code</td>
<td>N</td>
</tr>
<tr>
<td>F069016C</td>
<td>LF - Statutory Code, Tax Type</td>
<td>N</td>
</tr>
<tr>
<td>F069027</td>
<td>Table Unit of Measure File</td>
<td>Y</td>
</tr>
<tr>
<td>F06917</td>
<td>Tax Payment Schedule File</td>
<td>Y</td>
</tr>
<tr>
<td>F126JOIN</td>
<td>STAR - Join format file for F1201 &amp; F1202</td>
<td>Y</td>
</tr>
<tr>
<td>F1510</td>
<td>Property Management Constants</td>
<td>Y</td>
</tr>
<tr>
<td>F200001</td>
<td>Energy Constants Revisions</td>
<td>Y</td>
</tr>
<tr>
<td>F200002</td>
<td>Interest Type Constants</td>
<td>Y</td>
</tr>
<tr>
<td>F200003</td>
<td>Product Codes Constants</td>
<td>Y</td>
</tr>
<tr>
<td>F200004</td>
<td>Revenue/Prod Trans Typ Constants</td>
<td>Y</td>
</tr>
<tr>
<td>F230001</td>
<td>WPT Inflation Factor Constants</td>
<td>Y</td>
</tr>
<tr>
<td>F230002</td>
<td>Tax and Deduction Profile Constants</td>
<td>Y</td>
</tr>
<tr>
<td>F230003</td>
<td>Tax Rates Constants</td>
<td>Y</td>
</tr>
<tr>
<td>F230003A</td>
<td>LF - Tax Code, Effective Date</td>
<td>N</td>
</tr>
<tr>
<td>File Name</td>
<td>File Description</td>
<td>Copy Data</td>
</tr>
<tr>
<td>-----------</td>
<td>-------------------------------------------------------</td>
<td>-----------</td>
</tr>
<tr>
<td>F230004</td>
<td>Sliding Scale Tax Rates Constants</td>
<td>Y</td>
</tr>
<tr>
<td>F40096</td>
<td>Default Print Queues</td>
<td>Y</td>
</tr>
<tr>
<td>F40105</td>
<td>Subsystem Control Parameters</td>
<td>Y</td>
</tr>
<tr>
<td>F4090</td>
<td>Distribution/Manufacturing - AAI Master File</td>
<td>Y</td>
</tr>
<tr>
<td>F4091</td>
<td>Category Code Key Position File</td>
<td>Y</td>
</tr>
<tr>
<td>F48090</td>
<td>Work Order Supplemental Data Types</td>
<td>Y</td>
</tr>
<tr>
<td>F4849</td>
<td>Available Data Items</td>
<td>Y</td>
</tr>
<tr>
<td>F4849LA</td>
<td>Available Data Items</td>
<td>Y</td>
</tr>
<tr>
<td>F4849LB</td>
<td>Available Data Items</td>
<td>Y</td>
</tr>
<tr>
<td>F4857</td>
<td>Retrieval Code Definition</td>
<td>Y</td>
</tr>
<tr>
<td>F5192</td>
<td>Inquiry Columns</td>
<td>Y</td>
</tr>
<tr>
<td>F5193</td>
<td>Inquiry Formats</td>
<td>Y</td>
</tr>
<tr>
<td>F5194</td>
<td>Inquiry Paths</td>
<td>Y</td>
</tr>
<tr>
<td>F82013</td>
<td>World Writer Multi-Currency File</td>
<td>Y</td>
</tr>
<tr>
<td>F82100</td>
<td>Query Header File</td>
<td>Y</td>
</tr>
<tr>
<td>F82101</td>
<td>Query Data File Selections</td>
<td>Y</td>
</tr>
<tr>
<td>F82102</td>
<td>Query Data Join Fields</td>
<td>Y</td>
</tr>
<tr>
<td>F82103</td>
<td>Query Output Print Fields</td>
<td>Y</td>
</tr>
<tr>
<td>F82104</td>
<td>Query Output Print Field Calculations</td>
<td>Y</td>
</tr>
<tr>
<td>F82105</td>
<td>Query Data Selection Fields</td>
<td>Y</td>
</tr>
<tr>
<td>F82106</td>
<td>Query Data Selection Values</td>
<td>Y</td>
</tr>
<tr>
<td>F82107</td>
<td>Query Sort Fields</td>
<td>Y</td>
</tr>
<tr>
<td>F82108</td>
<td>Query Field Summary Functions</td>
<td>Y</td>
</tr>
<tr>
<td>F82109</td>
<td>Query File Update Specifications</td>
<td>Y</td>
</tr>
</tbody>
</table>
Appendix B - Upgrading Customized Source Code

J.D. Edwards provides you access to several complementary products. If you have customized J.D. Edwards source code, the following products will help you upgrade your source code.

**S/Compare**

Overall, S/Compare is a valuable aid used to:

- Identify differences between any two programs
- Simplify the task of documenting program changes
- Simplify the task of consolidating your custom changes into new releases of programs
- Identify differences between the names of the programs in two different files to quickly locate added or deleted programs in the new release

The S/Compare utility is specifically designed to compare two versions of source code. It will locate inserted, deleted, changed, or moved records in a source program. Processing options are provided to include or exclude comment lines, blank lines, and formatting differences. S/Compare’s output clearly identifies differences between two source members on a composite list of both programs. An option allows the records that are the same in the programs to be omitted from the listing to produce a report of only the differences between the files. This option also allows a given number of matching records on each side of a mismatch to be listed to help in identifying the section of source code.
Features of S/Compare

Some of the features and capabilities of S/Compare are:

- Flags are used in the composite listing to clearly mark statements or blocks of statements that have been inserted, deleted, or moved.
- Records that are moved from one location in the original file to another in the new program are indicated by source and target locations.
- Printing large blocks of identical code can be eliminated by a processing option. Only the differences will be printed and you can control the number of matching lines that are listed before and after each block of mismatched code.
- Differences between your program and the new program can be listed in an edit program.
- There is a processing option that can eliminate mismatches being printed because of spacing between words.
Harmonizer

Harmonizer adds to the capabilities of S/Compare by allowing the comparison of 3 to 16 program versions. Like S/Compare, the comparison results are written in a format that clearly depicts the differences between source members. In addition, Harmonizer has the capability of merging program versions to generate a composite source member. You can control what is written to the composite source member when potential conflicts are found.

Features and Capabilities of Harmonizer

Some of the features and capabilities of Harmonizer are:

- The comparison of 3 to 16 versions of a program.
- Two report formats are available. The MULTI-Compare report compares 3 to 16 programs. The TRI-Compare report is specifically designed for 3 programs.
- Statements from the original file that have been replaced, inserted, or deleted are noted on the comparison reports.
- All of the features of S/Compare are supported by Harmonizer when 3 programs are being compared, except the creation of an edit program which has been replaced by the creation a composite output program.
- The composite program may be compiled immediately or it may be edited. The ScmpEdit utility can be used to remove specified code in the composite program.
- The HARMONIZER command can be used to execute S/Compare and Harmonizer making the utilities easier to use.

Harmonizer Added to S/Compare

- You can incorporate your program changes into new releases easier. Harmonizer can compare the J.D. Edwards original program, the J.D. Edwards new release, and your customized program to produce a composite source file and a composite report. The composite report notifies you of discrepancies in the replacement, insertion, or deletion of code.
- The Source File Synopsis report produces a comparison of the program names in the J.D. Edwards original source file, the J.D. Edwards new source file, and your source file to determine any additions or deletions of programs.
- You can merge the development work of several programmers working on the same program.
About Harmonizer Plus

Harmonizer Plus adds to the capabilities of S/Compare and Harmonizer by helping you manage the ENTIRE process of building a new software release.

About the Project Manager Feature

The Project Manager feature will display an up-to-the-minute status of every program in your upgrade project. It shows:

- Which merged objects need a programmer review due to conflicts between local changes and vendor changes.
- Modified objects that are already created and ones that need to be created.
- Objects that are ready for production.
- Unmodified objects that must be recreated because they are dependent on modified objects.
- Objects that must be present before the object you are working with can be created.

Additional Functions

Harmonizer Plus provides a workbench for programmers to perform a variety of functions. Given the proper authority, a programmer can:

- Directly access SEU for editing programs.
- Mass compile entire groups of programs.
- Selectively compile individual programs.
- Selectively create all objects dependent on a modified object.
- Add or delete programs from the new production version.

Harmonizer Plus identifies unchanged modules that must be recompiled due to changes in prerequisite objects. For example, if you have modified DDS, Harmonizer Plus can identify programs that reference the related files. It can then recompile those programs. All you need to do is test and move the new libraries into production.
Appendix C - CL Models

J98MODEL1 - Interactive Video

9801 Software Versions Repository

Action Code... I
Member ID... J98MODEL1
Description... Model CL Program - Interactive Video
Function Code... CLP CL Programs
Function Use... 198 Model Source Member
System Code... 98 Technical Tools
Reporting System 98 Technical Tools
Base Member Name J98MODEL1 File Prefix...
Maint/RSTDSP... Omit Option... O Generation Sev...
Copy Data (Y/N)... N Optional File... N Common File...

Opt: 1=Browse 2=Edit 3=Copy 5=SAR 8=Print 9=Dlt 10=Design 14=Crt
J 98MODEL2 - Batch DREAM Writer without Printer File

9801 Software Versions Repository

Action Code. . .  I
Member ID. . .  J98MODEL2
Description. . . Model CL Program - Batch DREAM Writer without Printer File
Function Code. . CLP  CL Programs
Function Use . .  198  Model Source Member
System Code. .  98  Technical Tools
Reporting System  98  Technical Tools
Base Member Name  J98MODEL2  File Prefix. . .

<table>
<thead>
<tr>
<th>O Source</th>
<th>Object</th>
<th>Source</th>
<th>SAR</th>
<th>Version</th>
<th>S D</th>
<th>User</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Library</td>
<td>Library</td>
<td>File</td>
<td>Number</td>
<td>ID</td>
<td>CF</td>
<td>ID</td>
<td>Modified</td>
</tr>
<tr>
<td>JDFSRC73</td>
<td>JDFOBJ73</td>
<td>JDESRC</td>
<td>867923</td>
<td>AJ3</td>
<td>1</td>
<td>BECK</td>
<td>07/07/95</td>
</tr>
</tbody>
</table>

Opt: 1=Browse  2=Edit  3=Copy  5=SAR  8=Print  9=Dlt  10=Design  14=Crt

J 98MODEL3 - Interactive Video Prompt

9801 Software Versions Repository

Action Code. . .  I
Member ID. . .  J98MODEL3
Description. . . Model CL Program - Interactive Video Prompt
Function Code. . CLP  CL Programs
Function Use . .  198  Model Source Member
System Code. .  98  Technical Tools
Reporting System  98  Technical Tools
Base Member Name  J98MODEL3  File Prefix. . .

<table>
<thead>
<tr>
<th>O Source</th>
<th>Object</th>
<th>Source</th>
<th>SAR</th>
<th>Version</th>
<th>S D</th>
<th>User</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Library</td>
<td>Library</td>
<td>File</td>
<td>Number</td>
<td>ID</td>
<td>CF</td>
<td>ID</td>
<td>Modified</td>
</tr>
<tr>
<td>JDFSRC73</td>
<td>JDFOBJ73</td>
<td>JDESRC</td>
<td>867923</td>
<td>AJ3</td>
<td>1</td>
<td>BECK</td>
<td>07/07/95</td>
</tr>
</tbody>
</table>

Opt: 1=Browse  2=Edit  3=Copy  5=SAR  8=Print  9=Dlt  10=Design  14=Crt
### J98MODEL4 - Interactive/Batch with Processing Options

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action Code</td>
<td>I</td>
</tr>
<tr>
<td>Member ID</td>
<td>J98MODEL4</td>
</tr>
<tr>
<td>Description</td>
<td>Model CL Program - Interactive/Batch with Processing Options</td>
</tr>
<tr>
<td>Function Code</td>
<td>CLP</td>
</tr>
<tr>
<td>Function Use</td>
<td>CL Programs</td>
</tr>
<tr>
<td>System Code</td>
<td>98</td>
</tr>
<tr>
<td>Reporting System</td>
<td>Technical Tools</td>
</tr>
<tr>
<td>Base Member Name</td>
<td>J98MODEL4</td>
</tr>
<tr>
<td>Maint/RSTDSP</td>
<td>Omit Option: O</td>
</tr>
<tr>
<td>Copy Data (Y/N)</td>
<td>N</td>
</tr>
<tr>
<td>Library</td>
<td>JDFSRC73</td>
</tr>
<tr>
<td>Object</td>
<td>JDFOBJ73</td>
</tr>
<tr>
<td>Source</td>
<td>JDESRC</td>
</tr>
<tr>
<td>SAR</td>
<td>867923</td>
</tr>
<tr>
<td>Version</td>
<td>A73</td>
</tr>
<tr>
<td>User</td>
<td>BECK</td>
</tr>
<tr>
<td>Date</td>
<td>07/07/95</td>
</tr>
</tbody>
</table>

Opt: 1=Browse  2=Edit  3=Copy  5=SAR  8=Print  9=Dlt  10=Design  14=Crt

### J98MODEL5 - Batch Report Writer - No DDS File

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action Code</td>
<td>I</td>
</tr>
<tr>
<td>Member ID</td>
<td>J98MODEL5</td>
</tr>
<tr>
<td>Description</td>
<td>Model CL Program - Batch Report Writer - No DDS File</td>
</tr>
<tr>
<td>Function Code</td>
<td>CLP</td>
</tr>
<tr>
<td>Function Use</td>
<td>CL Programs</td>
</tr>
<tr>
<td>System Code</td>
<td>98</td>
</tr>
<tr>
<td>Reporting System</td>
<td>Technical Tools</td>
</tr>
<tr>
<td>Base Member Name</td>
<td>J98MODEL5</td>
</tr>
<tr>
<td>Maint/RSTDSP</td>
<td>Omit Option: O</td>
</tr>
<tr>
<td>Copy Data (Y/N)</td>
<td>N</td>
</tr>
<tr>
<td>Library</td>
<td>JDFSRC73</td>
</tr>
<tr>
<td>Object</td>
<td>JDFOBJ73</td>
</tr>
<tr>
<td>Source</td>
<td>JDESRC</td>
</tr>
<tr>
<td>SAR</td>
<td>867923</td>
</tr>
<tr>
<td>Version</td>
<td>A73</td>
</tr>
<tr>
<td>User</td>
<td>BECK</td>
</tr>
<tr>
<td>Date</td>
<td>07/07/95</td>
</tr>
</tbody>
</table>

Opt: 1=Browse  2=Edit  3=Copy  5=SAR  8=Print  9=Dlt  10=Design  14=Crt
J 98MODEL6 - Batch Report Writer OPNQRYF

9801 Software Versions Repository

Action Code. . . 1
Member ID. . .  J98MODEL6
Description. . . Model CL Program - Batch Report Writer OPNQRYF
Function Code. . CLP CL Programs
Function Use . . 198 Model Source Member
System Code. . . 98 Technical Tools
Reporting System 98 Technical Tools
Base Member Name  J98MODEL6
Maint/RSTDSP . . Omit Option. . 0 Generation Sel

<table>
<thead>
<tr>
<th>Source</th>
<th>Object</th>
<th>Source</th>
<th>SAR</th>
<th>Version</th>
<th>SD</th>
<th>User</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Library</td>
<td>Library</td>
<td>File</td>
<td>Number</td>
<td>ID</td>
<td>CP</td>
<td>ID</td>
<td>Modified</td>
</tr>
<tr>
<td>JDFSRC73</td>
<td>JDFOBJ73</td>
<td>JDESRC</td>
<td>867923</td>
<td>A73</td>
<td>1</td>
<td>BECK</td>
<td>07/07/95</td>
</tr>
</tbody>
</table>

Opt: 1=Browse 2=Edit 3=Copy 5=SAR 8=Print 9=Dlt 10=Design 14=Crt

J 98MODEL7 - Batch Report Writer OPNQRYF w/OQF Reset

9801 Software Versions Repository

Action Code. . . 1
Member ID. . .  J98MODEL7
Description. . . Model CL Program - Batch Report Writer OPNQRYF w/OQF Reset
Function Code. . CLP CL Programs
Function Use . . 198 Model Source Member
System Code. . . 98 Technical Tools
Reporting System 98 Technical Tools
Base Member Name  J98MODEL7
Maint/RSTDSP . . Omit Option. . 0 Generation Sel

<table>
<thead>
<tr>
<th>Source</th>
<th>Object</th>
<th>Source</th>
<th>SAR</th>
<th>Version</th>
<th>SD</th>
<th>User</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Library</td>
<td>Library</td>
<td>File</td>
<td>Number</td>
<td>ID</td>
<td>CP</td>
<td>ID</td>
<td>Modified</td>
</tr>
<tr>
<td>JDFSRC73</td>
<td>JDFOBJ73</td>
<td>JDESRC</td>
<td>867923</td>
<td>A73</td>
<td>1</td>
<td>BECK</td>
<td>07/07/95</td>
</tr>
</tbody>
</table>

Opt: 1=Browse 2=Edit 3=Copy 5=SAR 8=Print 9=Dlt 10=Design 14=Crt
## J98MODEL8 - Control File Driven Batch Process

<table>
<thead>
<tr>
<th>Action Code</th>
<th>Member ID</th>
<th>Description</th>
<th>Function Code</th>
<th>Function Use</th>
<th>System Code</th>
<th>Reporting System</th>
<th>Base Member Name</th>
<th>File Prefix</th>
<th>Maint/RSTDSP</th>
<th>Omit Option</th>
<th>Generation Sev</th>
<th>Copy Data (Y/N)</th>
<th>Optional File</th>
<th>Common File</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>J98MODEL8</td>
<td>Model CL Program - Control File Driven Batch Process</td>
<td>CLP</td>
<td>198</td>
<td>98</td>
<td>Technical Tools</td>
<td>J98MODEL8</td>
<td>__</td>
<td>O</td>
<td>__</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Source</th>
<th>Object</th>
<th>Source</th>
<th>SAR</th>
<th>Version</th>
<th>SD</th>
<th>User</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>JDFSRC73</td>
<td>JDFOBJ73</td>
<td>JDESRC</td>
<td>867923</td>
<td>A13</td>
<td>1</td>
<td>BECK</td>
<td>07/07/95</td>
</tr>
</tbody>
</table>

Opt: 1=Browse 2=Edit 3=Copy 5=SAR 8=Print 9=Dlt 10=Design 14=Crt
Glossary
Glossary

This glossary defines terms in the context of your use of J.D. Edwards systems and the accompanying user guide.

AAI. See Automatic Accounting Instructions.

access. To get to the information or functions provided by the system through menus, screens, and reports.

activity levels. The activity level of a storage pool is the number of jobs that can run at the same time in a storage pool. The machine manages the control of this level. Often during processing in a job, a program waits for a system resource or a response from a work station user. During such waits, a job gives up its use of the storage pools in order that another job that is ready to be processed can take its place.


advanced operating system. A single integrated operating system which contains: relational database, display manager, storage manager, communication manager, work manager, security manager and other managers.

AEC. Architectural, Engineering and Construction group.

allocating pools. If the system cannot allocate all the requested storage, it allocates as much storage as is available and allocates all the other as storage becomes available.

alphabetic character. Represents data by using letters and other symbols from the keyboard (such as *%&amp;#). Contrast with numeric character.

alphanumeric character. Represents data in a combination of letters, numbers, and other symbols (such as *%&amp;#).

ANSI. American National Standards Institute.

answers. Remember the online education system on the AS/400. All you need to remember is the command, GO SUPPORT.

AP. Accounts Payable.

APD. Application Program Driver.

API. An application programming interface describes the means by which a programmer can access the features provided by the interfaced object.

APPC. Advanced Program to Program Communications.

application. A collection of computer programs that allows you to perform specific business tasks. Some examples of applications are accounts payable, inventory, and order processing. Synonymous with system.

APPN. Advanced Peer-to-Peer Networking.

AS/400. Application System/400.

AS/400 Office. An IBM word processing program.

ASCII. American Standard Code for Information Interchange.

ASPs. Auxiliary Storage Pools.

attributes. To regard as belonging.

attribute byte. First character on a display field. This character controls how the field is displayed.

audit trail. The detailed, verifiable history of a processed transaction. The history consists of the original documents, transaction entries, and posting of records, and usually concludes with a report.

authority. The right to do something on the system or to use an object in the system, such as a file or a program.
**Automatic accounting instruction (AAI).** A code that points to an account in the chart of accounts. AAIIs define rules for programs that automatically generate journal entries. This includes interfaces between Accounts Payable, Accounts Receivable, and Financial Reporting and the General Accounting system. Each system that interfaces with the General Accounting system has AAIIs. For example, AAIIs can direct the Post to General Ledger program to post a debit to a certain expense account and an automatic credit to a certain accounts payable account.

**Autostart job entry.** A job is automatically started each time the subsystem is started.

**ATC.** Area Training Coordinator.

**AR.** Accounts Receivable.

**Backup copy.** A copy of original data preserved on a magnetic tape or diskette as protection against destruction or loss.

**BAPR.** Approved Budget Field Description.

**BASIC.** Beginners Application Software Introduction Class.

**Batch.** A group of like records or transactions that the computer treats as a single unit during processing. For identification purposes, the system usually assigns each batch a unique identifier, known as a “batch number.”

**Batch header.** Information the computer uses as identification and control for a group of transactions or records in a batch.

**Batch job.** A task or group of tasks you submit for processing that the system treats as a single unit during processing, for example, printing reports and purging files. The computer performs these tasks with little or no user interaction.

**Batch processing.** A method by which the computer selects jobs from the job queue, processes them, and writes output to the output queue. Contrast with *interactive processing.*

**Batch type.** A code that designates which J.D. Edwards system the associated transactions pertain to, thus controlling what records are selected for processing. For example, in the Post General Journal process, only unposted transaction batches with a batch type of G for General Accounting are selected for posting.

**Bit.** Binary digit. Either a zero or a one at the MI level.

**Bomb.** Fail.

**Boolean logic operand.** In J.D. Edwards DREAM Writer, the parameter of the Relationship field. The Boolean logic operand tells the system to perform a mathematical calculation on certain records or parameters. Available operands are:

- **EQ** = Equal To
- **LT** = Less Than
- **LE** = Less Than or Equal To
- **GT** = Greater Than
- **GE** = Greater Than or Equal To
- **NE** = Not Equal To
- **NL** = Not Less Than
- **NG** = Not Greater Than

**BORG.** Original/Beginning Budget Field BPC v. Budget Pattern Code.

**BREQ.** Requested Budget Field Description.

**B/S.** Balance Sheet.

**Buffer.** A reserved memory area used for performing input/output operations.

**Business unit.** Formerly cost center.

**Caching.** Refers to the use of a technique to locally store the results of input and output operations to minimize the use of slower accesses to disk drives and other storage devices.
**CAD/CAP.** Computer Assisted Design/Computer Assisted Programming. A set of automated programming tools for designing and developing applications. These tools automate system design, generate source code and documentation, enforce design standards, and help to ensure consistency throughout all J.D. Edwards systems.

**category code.** In user defined codes, a temporary title for an undefined category. For example, if you are adding a code that designates different sales regions, you could change *category code 4 to Sales Region,* and define E (East), W (West), N (North), and S (South) as the valid codes. Category codes were formerly known as *reporting codes.*

**CC.** Cost center. Now known as Business Unit.


**character.** Any letter, number, or other symbol that a computer can read, write, and store.

**character, special.** Representation of data in symbols that are neither letters nor numbers. Some examples are: *&@/#. 

**CLONE.** Crazy Logic Only Nerds Enjoy. (Old term for the Program Generator.)

**COBOL.** Common Business Oriented Language.

**Column.** See *field.*

**command.** A character, word, phrase, or combination of keys you use to tell the computer to perform a defined activity.

**compile.** To change source code into computer readable code.

**constants.** Parameters or codes that rarely change. The computer uses constants to standardize information processing by an associated system. Some examples of constants are allowing or disallowing out-of-balance postings and having the system perform currency conversions on all amounts. Once you set constants such as these, the system follows these rules until you change the constants.

**Core.** The central and foundational systems of J.D. Edwards software, including General Accounting, Accounts Payable, Accounts Receivable, Address Book, Financial Reporting, Financial Modeling and Allocations, and Back Office. Now called Financials.

**CPG.** Complementary Products Group.

**CRP.** Capacity Requirements Planning.

**CRP.** Conference Room Pilot. A simulation of the client’s business in a conference room environment.

**CUA.** Common User Access. IBM’s specification of a user interface definition across applications.

**CUM.** A representation of changes to J.D. Edwards software, which your organization receives on magnetic tapes or diskettes.

**current library.** Specifies a single library that is searched before any other user libraries in the library list. A current library is optional and can be different for each user or job. On displays, the current library is represented by the value "CURLIB."

**cursor.** The blinking underscore or rectangle on your screen that indicates where the next keystroke appears.

**cursor sensitive help.** See *field help.*

**data.** Numbers, letters, or symbols that represent facts, definitions, conditions, and situations, that a computer can read, write, and store.

**data item.** A code which represents a field, file, program, menu message, error message or help text stored in the data dictionary. Each piece of information within the database is defined by a data item. Data item name definition is limited to four characters in the J.D. Edwards systems to allow for program manipulation of the item.
**database.** A continuously updated collection of all information a system uses and stores. Databases make it possible to create, store, index, and cross-reference information online.

**data character.** A pattern of 8 bits.

**data dictionary.** A database file consisting of the definitions, structures, and guidelines for the usage of fields, messages, and help text. The data dictionary file does not contain the actual data itself.

**data field.** A collection of data characters.

**data Integrity.** Refers to checking the relationships between data items (fields) and being sure that values correlate correctly.

**data validation.** Determining if data is correct when compared to a set of conditions.

**DDE.** Dynamic Data Exchange.

**DDM.** Distributed Data Management.

**DDP.** Distributed Data Processing.

**DDS.** Data Description Specifications.

**default.** A code, number, or parameter the system supplies when you do not enter one. For example, if an input field's default is N and the you do not enter something in that field, the system supplies an N.

**descriptive title.** See user defined code.

**detail.** The individual pieces of information and data that make up a record or transaction. Contrast with summary.

**DFU.** Data File Utility. An IBM product.

**DIF.** Data Interchange Format.

**display.** (1) To cause the computer to show information on a terminal's screen. (2) A specific set of fields and information that a J.D. Edwards system might show on a screen. Some screens can show more than one display when you press a specified function key.

**display field.** A field of information on a screen that contains a system-provided code or parameter that you cannot change. Contrast with input field.

**DMA.** Direct Memory Access.

**DNS.** Do Not Spread.

**DOS.** Disk Operating System.

**DREAM Writer.** Data Record Extraction And Management Writer. A flexible data manipulator and cataloging tool. You use this tool to select and sequence the data that is to appear on a programmed report.

**DRP.** Distribution Requirements Planning.

**Dynamic.** Is constantly changing.

**DASD.** Data Auxiliary Storage Device.

**ECS.** Electronic Customer Support.

**edit.** (1) To make changes to a file by adding, changing, or removing information. (2) The program function of highlighting fields into which you have entered inadequate or incorrect data.

**EDI.** Electronic Data Interchange. The transmission of business documents among computers of independent organizations.

**EFT.** Electronic Fund Transfer.

**EIS.** Executive Information System.

**Engagement letter.** A letter identifying the mutual understandings and initial expectation of the client and J.D. Edwards.

**environment.** The list of files required by a user to perform certain tasks. For example, a programmer has access to a test environment and an environment which includes live data. Each environment utilizes a different set of files.

**execute.** See run.

**exit.** (1) To interrupt or leave a computer program by pressing a specific key or a sequence of keys. (2) An option or function key displayed on a screen that allows you to access another screen.
facility. A collection of computer
language statements or programs that
provides a specialized function throughout
a system or throughout all integrated
systems. Some examples DREAM Writer and
FASTR.

Fast Path Mnemonics. A method of
using a UDC to define execution to a J.D.
Edwards program.

FASTR. Financial Analysis Spreadsheet
Tool and Report Writer. A report writer that
lets you design your own report
specifications using the general ledger
database.

FDA. File Design Aid. A J.D. Edwards
design tool.

field. (1) An area on a screen where you
type in data, values, or characters. (2) A
defined area, usually within a record, which
contains a specific piece of information
such as name, document type or amount.
For example, a vendor record consists of
the fields Vendor Name, Vendor Address
and Telephone Number. The field Vendor
Name contains only the name of the
vendor. See input field and display field.
Also known as column.

field help. J.D. Edwards online Help
function, which lets you view a description
of a field, its purpose and, when applicable,
a list of the valid codes that you can enter.
You access this information by pressing F1
with the cursor positioned in the field.

file. A collection of related data records
organized for a specific use and
electronically stored by the computer. Also
known as table.

financial systems. The central and
foundational systems of J.D. Edwards
software, including General Accounting,
Accounts Payable, Accounts Receivable,
Address Book, Financial Reporting,
Financial Modeling and Allocations, and
Back Office. Previously known as core.

fold area. An area of a screen, accessed
by pressing F4, that displays additional
information associated with the records or
data items displayed on the screen.

function. A separate feature within a
facility that allows you to perform a specific
task, for example, the field help function.

function key. A key you press to perform
a system operation or action. For example,
you press F4 to have the system display the
fold area of a screen.

Form. One World term for video.

glossary. The collection of text related to
specific data items. The glossary contains
help text and message text.

GL. General Ledger.

GA. General Accounting.

GST. Goods & Service Tax.

GUI. Graphical User Interface.

hard code. Program instructions which
can only be altered by a programmer. The
altered instructions must then recompiled so
the computer can understand them.

hard copy. A presentation of computer
information printed on paper. Synonymous
with printout.

header. Information at the beginning of a
file. This information is used to identify or
provide control information for the group of
records that follows.

help instructions. Online documentation
or explanations of fields that you access by
pressing the Help key or by pressing F1
with your cursor in a particular field.

helps. See help instructions.

hidden selections. Menu selections you
cannot see until you enter HS in a menu’s
Selection field. Although you cannot see
these selections, they are available from any
menu. They include such items as Display
Submitted Jobs (33), Display User Job
Queue (42), and Display User Print Queue.
Advanced Programming Concepts and Skills

(43). The Hidden Selections window displays three categories of selections: user tools, operator tools, and programmer tools.

**HMC.** Horizontal Microcode.

**HS.** J.D. Edwards Hidden Selections.

**ICCC.** InterCompany Cost Center. *Now known as business unit.*

**ICF.** Intersystem Communication Function.

**ICH.** InterCompany Hub.

**IDDU.** Interactive Data Definition Utility – IBM Product.

**IMP.** Internal Microprogram Load.

**IMPI.** Internal Microprogramming Interface.

**Implementation Methodology.** Nine steps to provide J.D. Edwards consulting staff with a guide for implementing the software in a thorough and consistent manner.

**input.** Information you enter in the input fields on a screen or that the computer enters from other programs, then edits and stores in files.

**input field.** An area on a screen, distinguished by underscores (_ _), where you type data, values, or characters. A field represents a specific type of information such as name, document type, or amount. Contrast with *display field.*

**install system code.** The four-character identifier of a J.D. Edwards system. For example, 01 for the Address Book system, 04 for the Accounts Payable system, and 09 for the General Accounting system. *Now known as system code.*

**integrity.** Soundness, completeness.

**interactive job.** An interactive job starts when a user signs on a display station and ends when the user signs off. During the job, the user interacts with the system.

**interactive processing.** A job the computer performs in response to commands you enter from a terminal.

During interactive processing, you are in direct communication with the computer, and it might prompt you for additional information during the processing of your request. See *online.* Contrast with *batch processing.*

**interface.** A link between two or more J.D. Edwards systems that allows these systems to send information to and receive information from one another.

**I/O.** Input/Output.

**IPL.** Initial Program Load.

**ITF.** Interactive Terminal Facility.

**JDE.** Jack, Dan and Ed. Founders of JD Edwards & Co.

**jargon.** A J.D. Edwards term for system-specific text. You base your jargon help text on a specific reporting code you designate in the Data Dictionary Glossary. You can display this text as part of online help. You create your jargon text descriptions and titles for data items through the Data Dictionary, menu and vocabulary overrides record using a reporting system code. Jargon text descriptions and titles for data items display on screens as field names.

**job.** A single identifiable set of processing actions you tell the computer to perform. You start jobs by choosing menu selections, entering commands, or pressing designated function keys. An example of a computer job is check printing in the Accounts Payable system.

**job description.** An object consisting of a set of specifications about a computer job and its executing environment.

**job log.** A job log is a record of requests (such as commands) submitted by the system by a job, the messages related to the requirements and the actions performed by the system on the job.

**job queue.** A group of jobs waiting to enter a subsystem.
Join logical file. Presents composite records consisting of fields extracted from two or more physical records from two or more physical files.

justify. To shift information you enter in an input field to the right or left side of the field. Many of the facilities within J.D. Edwards systems justify information. The system does this only after you press Enter.

KBG. Knowledge-Based Generator. See program generator.

key field. A series of identifying or controlling characters a computer uses to retrieve related information tied to the key. An employee number, for example, is a key field consisting of references to other files in the system that contain information about the given employee.

Key General Ledger Account (Key G/L). See automatic accounting instructions.

LAN. Local Area Network.

leading zeros. A series of zeros that certain facilities in J.D. Edwards systems place in front of a value you enter. This normally occurs when you enter a value that is smaller than the specified length of the field. For example, if you enter 4567 in a field that accommodates eight numbers, the facility places four zeros in front of the four numbers you enter. The result would look like this: 00004567.

level check. A mechanism of the OS/400 that assures that a file version and program using that file are in sync with one another.

level of detail. (1) The degree of difficulty of a menu in J.D. Edwards software. The levels of detail for menus are as follows:

A=Major Product Directories
B=Product Groups
1=Daily Operations
2=Periodic Operations
3=Adv/Tech Operations
4=Computer Operations

5=Programmers
6=Advanced Programmers

Also known as menu levels. (2) The degree to which account information in the General Accounting system is summarized. The highest level of detail is 1 (least detailed) and the lowest level of detail is 9 (most detailed).

library. A library groups objects. A library is an object itself. Similar to directory on a PC.

library list. An ordered list of libraries used for locating objects. Similar to path on a PC.

LIOM. Line Input/Output Manager.

LOD. Level of Detail.

logical file. Contains no data, but provides a view of one or more physical files upon which it is based.

master file. A computer file that a system uses to store data and information which is permanent and necessary to the system’s operation. Master files might contain data or information such as paid tax amounts and vendor names and addresses.

MDA. Menu Design Aid. A J.D. Edwards design tool.

menu. A screen that displays numbered selections. Each of these selections represents a program. To access a selection from a menu, type the selection number and then press Enter.

menu levels. See level of detail.

menu masking. A security feature of J.D. Edwards systems that allows you to prevent individual users from accessing specified menus or menu selections. When this security is in effect for a user, the selections that have been secured do not appear on the screen.

menu message. Text that appears on a screen after you make a menu selection. It displays a warning, caution, or information about the requested selection.
menu traveling.  A method of moving between menus by typing the menu identifier in the selection field of the screen.

MI.  Machine Interface.

MRP.  Manufacturing Resource Planning.

MRP.x.  J.D. Edwards Manufacturing Software.

MVS.  Multiple Virtual Storage.

next number facility.  A J.D. Edwards software facility you use to control the automatic numbering of such items as new G/L accounts, vouchers, and addresses. It lets you specify your desired numbering system and provides a method to increment numbers to reduce transposition and typing errors.

non-join logical file.  Presents records that are composed of fields extracted from just one physical record, but can effectively merge two or more physical files.

numeric character.  Represents data using the numbers 0 through 9. Contrast with alphabetic character and alphanumeric character.

object.  A discrete entity.

object existence.  The right to delete an object from the system.

object management.  The right to change the name or library of an object, for physical files, the right to create a logical file over it.

object operational.  The right to display the description of an object and the right to the general use of that object.

object orientation.  Everything on the AS/400 system that can be stored or retrieved is contained in an object.

offline.  Computer functions that are not under the continuous control of the system. For example, if you were to run a certain job on a personal computer and then transfer the results to a host computer, that job would be considered an offline function. Contrast with online.

One Step Install.  A method developed to make our software easier to install.

online.  Computer functions over which the system has continuous control. Each time you work with a J.D. Edwards system-provided screen, you are online with the system. Contrast with offline. See interactive processing.

online information.  Information the system retrieves, usually at your request, and immediately displays on the screen. This information includes items such as database information, documentation, and messages.

Open Application Architecture.  An architectures that uses a functional server to allow the various blocks of user interface logic to access the same block of data integrity logic.

operand.  See Boolean logic operand.

option.  A numbered selection from a J.D. Edwards screen that performs a particular function or task. To select an option, you enter its number in the Option field next to the item you want the function performed on. When available, for example, option 4 lets you return to a prior screen with a value from the current screen.

OS/400.  Operating system for the AS/400.

OS/2.  Operating system for the IBM personal computer.

OSI.  Open Systems Interconnection.

output.  Information the computer transfers from internal storage to an external device, such as a printer or a computer screen.

output queue.  A group of spool files waiting to be attached to a writer.
override. The process of entering a code or parameter other than the one provided by the system. Many J.D. Edwards systems offer screens that provide default field values when they appear. By typing a new value over the default code, you can override the default. See default.

PACO. Posted After Cutoff.

parameter. A number, code, or character string you specify in association with a command or program. The computer uses parameters as additional input or to control the actions of the command or program.

password. A unique group of characters that you enter when you sign on to the system that the computer uses to identify you as a valid user.

PBCO. Posted Before Cutoff.

PC. Personal computer.

PDM. Program Development Manager. IBM design tool.

PDM. Product Data Management—a module of J.D. Edwards software.

physical file. A file that contains actual data records. Max a maximum record length of 32K, maximum fields per record is 8000.

Plug-&-Go. A 2/18/92 announcement where J.D. Edwards selects PROGRESS to develop client applications for the AS/400. The plug–&–go format offers clients the J.D. Edwards Core financial solutions on the IBM AS/400 E series model.

PPAT. People, Places and Things.

printout. A presentation of computer information printed on paper. Synonymous with hard copy.

print queue. A group of items waiting to be printed. See output queue.

processing options. A feature of the J.D. Edwards DREAM Writer that lets you supply parameters to direct the functions of a program. For example, processing options allow you to specify defaults for certain screen displays, control the format in which information gets printed on reports, change the way a screen displays information, and enter “as of” dates.

product library. A library containing programs and related data needed for IBM licensed programs that are installed on your system.

production library. A production library is a library you create to contain your live J.D. Edwards data files.

production environment. A list of libraries that contains “live” programs and data.

program. A collection of computer statements that tells the computer to perform a specific task or group of tasks.

Progress. A software corporation that is a partner with J.D. Edwards. They are a leading supplier of 4th generation application development systems.

program generator. The World CASE system of programs which create a new program based upon user specifications.

program help. J.D. Edwards online facility which displays information about a program's use and functionality.

program-specific help text. Glossary text written to describe the function of a field within the context of the program.

prompt. (1) A reminder or request for information displayed by the system. When a prompt appears, you must respond in order to proceed. (2) A list of codes or parameters or a request for information provided by the system as a reminder of the type of information you should enter or action you should take.

PTF. See CUM.

purge. The process of removing records or data from a file.

PYEB. Post Year End Balance.

P&L. Profit and Loss Statements.

PG. Program Generator.
QA.  Quality Assurance.

QJDF data area.  A space within the system to hold the system values information for the J.D. Edwards software.  This area is referenced at sign-on and during installs and re-installs for critical system information, such as security codes and initial libraries.

QSECOFR.  The security officer of the AS/400.

query.  A fast means to select and display (or print) information from a database. An IBM utility for databases.

queue.  A list of things to be used in an order.  See job queue, output queue, and print queue.

RAID.  Redundant Array of inexpensive disks.

RAM.  Random Access Memory.


read only.  A type of access to data that allows it to be read but not copied, printed or modified.

rebuild.  The process of sequencing files, integrating recently added data.

record.  A collection of related, consecutive fields of data the system treats as a single unit of information.  For example, a vendor record consists of information such as the vendor’s name, address, and telephone number. Also known as row.

record format.  The definition of how data is structured in the records contained in a file.

record level locking.  Prevents two people from simultaneously updating the same data base information.

REP.  Rapidly, Economically and Predictably.

reply list.  A system wide automatic message handler for the system.

recursive.  In DREAM Writer, the ability to create a unique version from the original, process the new version and delete it, leaving the original intact.

re-engineering modules.  Programs written for the purpose of changing many existing programs in mass.

reporting system code.  The four-character identifier of a J.D. Edwards system that uses an object for reporting.

REQIO.  Request Input/Output.

reverse image.  Screen text that displays in the opposite color combination of characters and background from what the screen typically displays (for example, black on green instead of green on black).

RIBA.  Ricevuta Bancaria Elettronica — common way for vendors to receive payments from their customers in Italy.

ROM.  Read Only Memory.

ROW.  See record.

RPG.  Report Program Generator. A programming language developed by IBM.

Rumba.  A PC Emulator for the AS/400.

run.  To cause the computer to perform a routine, process a batch of transactions, or carry out computer program instructions.

SAA.  Systems Application Architecture.

SAR.  See Software Action Request.

server.  A program that speeds the flow of data between screens, reports and the data files. These programs can also be used to edit data fields.

scroll.  To use the roll keys to move screen information up or down a screen at a time. When you press the Rollup key, for instance, the system replaces the currently displayed text with the next screen of text if more text is available.

SDA.  Screen Design Aid Utility.  An IBM product.
selection. Found on J.D. Edwards menus, selections represent functions that you can access from a given menu. To make a selection, you type its associated number in the Selection field and press Enter.

SEU. Source Entry Utility.

SIC. Standard Industry Code.

SIOM. Station Input/Output Manager.

Ski Slope. Reflects the analogy between the diverse nature of a ski slope and the diverse nature of our software. S levels: Basic, Intermediate, Advanced, Computer Operations and Program Modifications.

SNA. Systems Network Architecture.

SNADS. Systems Network Architecture Distribution Services.

Sleeper. A subsystem which activates jobs set to run during off-peak hours.

softcoding. A J.D. Edwards term that describes an entire family of features that lets you customize and adapt J.D. Edwards software to your business environment. These features lessen the need for you to use computer programmers when your data processing needs change.

software. The operating system and application programs that tell the computer how and what tasks to perform.

Software Action Request. A record which identifies an activity, such as the development of a new program or maintenance of an existing program.

Software Security Code. A code that restricts user access to software.

special character. Representation of data in symbols that are neither letters nor numbers. Some examples are * & # /.

spool. Simultaneous Peripheral Operations On Line. The function by which the system puts generated output into a storage area to await printing or processing.

spooled file. A holding file for output data waiting to be printed or input data waiting to be processed.

SQL. Structure Query Language.

STAR. Spreadsheet Tool for Asset Reporting.

subfile. An area on the screen where the system displays detailed information related to the header information at the top of the screen. Subfiles might contain more information than the screen can display in the subfile area. If so, use the roll keys to display the next screen of information. See scroll.

submit. See run.

subsystem. An operating environment where jobs are run.

summary. The presentation of data or information in a cumulative or totaled manner in which most of the details have been removed. Many of the J.D. Edwards systems offer screens and reports that are summaries of the information stored in certain files.

SVR. Software Versions Repository.

system. A collection of computer programs that lets you perform a specific business function, such as Accounts Payable, Inventory, or Order Processing. Synonymous with application.

system library. Lists libraries containing objects, such as user profiles, that are used by the system. This part of a library list is defined by the system value QSYSLIBL and is usually the same for all jobs.

Simplified Install. J.D. Edwards new way to install J.D. Edwards software. Also called one step Install.

SME. Subject Matter Expert.

T/B. Trial Balance.

Table. One World term for a file.

UNIX. A multi-user, multi-tasking operating system.
**Unscheduled PTF.** A form of PTF that includes fixed for a particular system.

**UPS.** Uninterruptible power source.

**user class/group.** Place to enter group profiles associated with J.D. Edwards Users.

**user defined code.** The individual codes you create and define within a user defined code type. Code types are used by programs to edit data and allow only defined codes. These codes might consist of a single character or a set of characters that represents a word, phrase, or definition. These characters can be alphabetic, alphanumeric, or numeric. For example, in the user defined code type table ST (Search Type), a few codes are C for Customers, E for Employees, and V for Vendors.

**user defined code (type).** The identifier for a table of codes with a meaning you define for the system (for example, ST for the Search Type codes table in Address Book). J.D. Edwards systems provide a number of these tables and allow you to create and define tables of your own. User defined codes were formerly known as descriptive titles.

**user index.** An object that stores data, allows search functions, and automatically sorts data based upon a key value.

**user identification (user ID).** The unique name you enter when you sign on to a J.D. Edwards system to identify yourself to the system. This ID can be up to 10 characters long and can consist of alphabetic, alphanumeric, and numeric characters.

**user library.** A libraries that contains objects, such as files and programs used by the user.

**user profile.** A file of information which identifies the user to the J.D. Edwards system. This file is used to validate the user authority within the system.

**user space.** An object made up of a collection of bytes used for storing user-defined information.

**user type.** A code which identifies a list of files which remain open while the user is signed on to the system.

**valid codes.** The allowed codes, amounts, or types of data that you can enter in a specific input field. The system checks, or edits, user defined code fields for accuracy against the list of valid codes.

**version.** A specific release of software. Usually numbered in ascending order.

**VCS.** Version Control System.

**Vertex.** Callable routines and tables that calculate US PIR taxes.

**video.** The display of information on your monitor screen. Normally referred to as the screen.

**VM.** Virtual Machine.

**VMC.** Vertical Microcode.

**vocabulary overrides.** A J.D. Edwards facility that lets you override field, row, or column title text on a screen-by-screen or report- by-report basis.

**WACO.** Way After Cutoff.

**WAN.** Wide Area Network.

**window.** A software feature that allows a part of your screen to function as if it were a screen in itself. Windows serve a dedicated purpose within a facility, such as searching for a specific valid code for a field.

**writer.** A J.D. Edwards printer attached to an outqueue.

**World Vision.** A complementary product that converts graphical user interfaces to J.D. Edwards business applications for the AS400.

**World VISTA.** A windows–based direct access to J.D. Edwards data on the AS/400.


**XREF.** Cross reference tool for J.D. Edwards software.
**YTD.** Year to Date.
Index
Symbols

*ALL, using, 3–67
*BOTH, using, 3–65
*BOTH and *ALL features, using, 3–65

A

About attention MENU window, 5–15
About CASE profiles, 2–75
About common subroutines, 4–13
About creating libraries, 2–8
About data dictionary repository, 3–15
About designing the report, 3–109
About development environments, 2–7
About file servers, 4–107
About functional servers, 4–123
About group jobs, 5–1
About performance issues, 4–171
About program specifications, 4–3
About programming tools, 3–1
About record formats, 3–78
About report design aid, 3–101
About SAR log, 2–83
About SAR system setup, 2–41
    See also Defining a promotion path
About screen design aid, 3–49
About setting up universal file converter, 6–7
About software scan and replace, 4–169
About source debugger, 4–153
About standard subroutines, 4–12
About subroutines, 4–25
About the conversion rule program, 6–22
About the data file design aid, 3–37
    automatic reference to field reference files, 3–38
    data dictionary validation, 3–37
    enforced naming conventions, 3–37
    enforced prefixes, 3–37
    resequencing, 3–38
About the field reference file, 3–33
About the J. D. Edwards message file, 3–34
About the program overview, 4–11
About universal file converter, 6–1
About user indexes, 4–75
About user spaces, 4–67
About using IBM pass-through with group jobs, 5–19
About version control, 2–1
About working with the J. D. Edwards group job window, 5–3
Accessing, data modeling, 3–4
Accessing CASE profiles, 2–76
Accessing report design aid, 3–110
Accessing SAR log inquiry, 2–87
Accessing the J.D. Edwards attention MENU window, 5–17
Accessing the J.D. Edwards group job window, 5–5
Accessing the software versions repository, 2–49
Accessing the user index, 4–140
Accessing the user space, 4–139
Accessing versions setup, 6–9
Activating suspended group jobs, 5–7
Add
    a project, 2–101
    promotion path, 2–95
Add Cross Over Instructions form (00312), 6–18
Adding a literal video field, adding, 3–64
Adding fields, 6–18
    See also Special processing
Adding fields without using a pick list, 3–61
Adding record types, 2–39
Address numbers, searching for, 2–30
Advanced functions of the J.D. Edwards group job window, 5–13
    entering commands, 5–13
    hidden selections, 5–13
Application development cycle, 1–10
Architecture, engineering, construction, and real estate, J.D. Edwards product line, 1–6
Assign
    project SABs, 2–104
    promotion paths, 2–102
Assigning the file prefix, 3–39
Advanced Programming Concepts and Skills

Attention MENU window, 5–15
   accessing, 5–17
functions summary, 5–18
ATTN key program, setting, 5–4
Available functional servers, 4–151
Available functions and options, 6–23

B

Basic accounting transactions, graphic, 4–127
Build Transfer Library form (P98312), 2–118
Building blocks, J.D. Edwards software, 1–10

C

CAD. See Computer Assisted Design
Calculation specifications, 4–8
Call Program form (CALL), 5–12
CASE profiles, 2–75
   accessing, 2–76
   summary, 2–81
CASE Profiles form (P98009), 2–77, 2–84
CASE profiles program, function key exits, 2–80
Change Library List form (CHGLIBL), 5–11
Changing compile option defaults for reports, 3–120
Changing record types, 2–40
Changing subfile boundaries, 3–93
   to make larger, 3–93
   to make smaller, 3–93
Changing to non-group mode, 5–9
Column Title 1 – XREF build, defined, 3–21
Commonly used file servers, 4–122
Compile option defaults for reports, changing, 3–120
Compiling a report, 3–118
Compiling your video, 3–70
Computer Assisted Design. See CAD
Computer Assisted Design (CAD) form (G92), 2–76
Conducting a search, 3–12
Continue execution, source debugger, 4–160
Control fields, user space, 4–144
Control fields within the user space, 4–144
Control Files Copy form (P924127), 2–116
Control parameters, 4–111
Control specifications, 4–4
Control table records, transferring individual, 2–127
Conventions, naming, 2–57
Conversion Forms, Printing, 6–33
Conversion forms, creating, 6–31
Conversion rule program, 6–22
   fifth parameter, 6–23
   first parameter, 6–22
   fourth parameter, 6–22
   second parameter, 6–22
   third parameter, 6–22
Copy Data Files form (P98101), 2–24
Copy DD, VO, DW, UDC, SVR, Menus form (P99630), 2–27, 2–127
Copy File form (CPYF), 2–17, 2–25
COPY modules, functional servers, 4–138
Copy modules, 2–67
Copying a file, 2–24
Copying a library, 2–23
Copying a record, 2–25
Copying data to your development environment, 2–23
Copying J.D. Edwards record types, 2–27
Create Library form (CRTLIB), 2–16
Create New Screen form (P92510), 3–58
Create Source Physical File form (CRTRSCPF), 2–19
Create User Data Libraries form (P98312), 2–13
Creating a development environment, 2–7
Creating a development source library, 2–16
Creating a user index, 4–78
Creating a user space, 4–69
Creating common libraries, 2–13
Creating conversion forms, 6–31, 6–32
Creating development object libraries, 2–14
Creating JDESRC with J.D. Edwards
   program generator, 2–17
Creating JDESRC without the program
   generator, 2–19
Creating libraries, 2–12
Creating new job groups, 5–6
Creating record type codes, 2–42
Creating the transfer library, 2–118
See also Define a project; Define promotion paths; Prepare the SAR system; Promote the project in Promote a SAR; Update the SARs in Promote a SAR; Validate the promotion path in Promote a SAR
Creating user space and user index, 4–139
Cross Reference form (P980014), 2–71, 3–9
Crossover rules, working with, 6–15
Crossover Rules form (0031), 6–13, 6–14, 6–16, 6–17, 6–23

D

Data base video field (VD), adding, 3–62
Data dictionary
security, 3–23
user defined help instructions, 3–29
working with, 3–19
Data Dictionary form (P9201), 3–19, 6–36
Data dictionary glossary
groups, 3–25
working with, 3–25
working with by file, 6–36
Data dictionary repository, 3–15
Data dictionary structure
data field display text, 3–17
data field specifications, 3–16
data item aliases, 3–17
data item alpha description, 3–17
data item master, 3–16
error message program ID, 3–17
glossary text file, 3–17
key index file, 3–17
understanding, 3–16
Data Display Decimals, defined, 3–21
Data Display Rules, defined, 3–22
Data Edit Rules, defined, 3–22
Data Field – Parent, defined, 3–19
Data Field Alias form (P9204), 3–24
Data field descriptions, working with, 3–30
Data Field Descriptions form (P9202), 3–30
Data File Decimals, defined, 3–21
Data file design aid, 3–37
entering, 3–40
Data File Design Aid form (P92102), 3–40, 3–43, 3–44
Data file design aid standards, 3–46
Data file design aid summary, 3–48
Data files relationships, graphic, 3–16
Data integrity logic, 4–130
Data Item, defined, 3–20, 3–28
Data item alias, revisions, 3–24
Data Item Class, defined, 3–21
Data Item Glossary Revisions form (P92001), 3–27, 3–29, 6–36
Data item name, locating, 3–18
Data Item Size, defined, 3–21
Data Item Type, defined, 3–20
Data modeling
accessing, 3–4
function key exits, 3–8
selection exits, 3–9
working with, 3–3
Data Modeling form (P98042), 3–5, 3–6, 3–10
Database considerations, 6–5
Database fields, selecting, 3–80
Debugger
using with interactive program, 4–154
with batch program, 4–157
Define
project, 2–119
promotion path, 2–119
promotion path for control tables, 2–98
promotion path for source code members, 2–96
Define Soft Coding Fields form (P928400), 3–117
Defining a project, 2–99
Defining a promotion path, 2–94
Defining access for a user profile using J98INIT, 2–22
Defining access for a user profile using J98INITA, 2–20
Defining record type titles, 2–44
See also Creating record type codes
Deleting records, 6–19
Description, defined, 3–21
Description–Alpha, defined, 3–20
Design Aid Exit/Save form (P92590), 3–69
Designing the report, 3–109
Detail Spec. Over Titles form (P48002), 2–40

Release A7.3 (June 1996)
Advanced Programming Concepts and Skills

Detail Spec. Types form (P00051), 2–39
Detailed explanation of a line, 3–7
Determining, program environment, 4–154, 4–157
Developer’s Workbench form (G9362), 2–87
Development cycle, 1–10
Development environment, copying data to, 2–23
Development environments, 2–7
rules for creating, 2–7
Displaying field descriptions, 6–16
Distribution/logistics, J.D. Edwards product line, 1–5
documentation, 4–19

E

Edit and Promote form (P92412), 2–108
Editing commands, 3–50
End debug (ENDDBG), 4–167
ENDDBG (end debug), 4–167
Energy and chemical, J.D. Edwards product line, 1–6
Entering Data file design aid, 3–40
Entry program, setting up business rules, 4–126
Error handling, 4–15
Error message, index line (COORIX), 4–145
Error message index line (COORIX), 4–145
Examples, production and development, 2–10
Executing
  program being debugged, 4–156
  source debugger, 4–159
Existing field, updating, 3–55
Exits, function key, 3–54
Extension specifications, 4–6

F

Fast path create, accessing for a new form, 3–58
Features, system integration, 1–15
Field definition, revising, 3–96
Field Definition report (P928400), 3–111, 3–113, 3–114
Field definition window, report design aid vs. screen design aid, 3–103
Field descriptions, displaying, 6–16
Field in RDA, updating, 3–111
Field name standards, 3–52
  updating/adding fields through SDA, 3–53
Field reference file, 3–33
  rebuild, 3–33
Field Selection List form (P92524), 3–81, 3–83
Fields
  add hidden, 3–90
  adding, 6–18
  adding without using pick list, 3–61
  Column Title 1 – XREF build, 3–21
  cover page, graphic, 3–104
  Data Display Decimals, 3–21
  Data Display Rules, 3–22
  Data Edit Rules, 3–22
  Data Field – Parent, 3–19
  Data File Decimals, 3–21
  Data Item, 3–20, 3–28
  Data Item Class, 3–21
  Data Item Size, 3–21
  Data Item Type, 3–20
  Description, 3–21
  Description–Alpha, 3–20
displaying. See Advanced Functions
  Reference Guide; Computer Assisted
  Design Reference guide
Glossary Group, 3–20, 3–28
Help Text Program, 3–22
Next Numbering Index Number, 3–23
Number of Array Elements, 3–21
Release Number, 3–19
report header, graphic, 3–105
System Code, 3–20
System Code – Next Numbers, 3–23
System Code/Reporting, 3–20
Value for Entry – Default, 3–22
work, 4–22
File conversion, working with, 6–25
File description specifications, 4–5
File design aid, function keys from, 3–44
File prefix, 3–39
File Selection List form (P92522), 3–83
File servers, 4–107
  advantages, 4–109
  commonly used, 4–122
disadvantages, 4–109
how does it function, 4–110
implementing, 4–114
tips when using, 4–116
types, 4–108
what is a, 4–108
Files, copying, 2–24
Financials, J.D. Edwards product line, 1–5
Flow of typical program, graphic, 4–125
Format Keyword Maintenance form
(P92537), 3–86
Forms
   Add Cross Over Instructions, 6–18
   Build Transfer Library, 2–118
   Call Program, 5–12
   CASE Profiles, 2–77, 2–84
   Change Library List, 5–11
   Computer Assisted Design (CAD), 2–76
   Control Files Copy, 2–116
   Copy Data Files, 2–24
   Copy DD, VO, DW, UDC, SVR, Menus, 2–27, 2–127
   Copy File, 2–25
   Copy File (CPYF), 2–17
   Create Library (CRTLIB), 2–16
   Create New Screen, 3–58
   Create Source Physical File (CRTSRCPF), 2–19
   Create User Data Libraries, 2–13
   Cross Reference, 3–9
   Cross reference, 2–71
   Crossover Rules, 6–13, 6–14, 6–16, 6–17, 6–23
   Data Dictionary, 3–19, 6–36
   Data Field Alias, 3–24
   Data Field Descriptions, 3–30
   Data file design Aid, 3–40
   Data File Unit, 3–43, 3–44
   Data Item Glossary Revisions, 3–27, 3–29
   Data Item Glossary Revisions, 6–36
   Data Modeling, 3–5, 3–6, 3–10
   Design Aid Exit/Save, 3–69
   Detail Spec. Over Titles, 2–40
   Detail Spec. Types, 2–39
   Developer’s Workbench, 2–87
   Edit and Promote, 2–108
   field Selection List, 3–81, 3–83
   File Selection List, 3–83
   Format Keyword Maintenance, 3–86
   FRF & JDE Msg File, 3–34
   Function Key/Opt Definition, 3–88
   Indicator Control, 3–116
   Item Maintenance, 3–59, 3–60
   Item Search, 3–54, 3–55, 3–78
   JDE Pass thru Network, 5–22
   JDE Visual Debug, 4–156, 4–159, 4–160, 4–162
   Library List Revisions, 2–20
   List of Defined Fields, 3–90, 3–116
   Load Transferred Library, 2–125
   Maintain User Default SAR Info, 2–85
   maintain vocabulary override fields, 3–117
   Manage Program Paths, 2–94
   Manage Projects, 2–100
   Next Numbers, 3–31
   Next Numbers by Company/Fiscal, 3–32
   Object Cross Reference Repository, 3–12, 3–14
   Optional Files Workbench, 2–64
   Pre Promotion Edit Details, 2–111
   Pre Promotion Edit History, 2–111
   Print Transfer Report, 2–122
   Processing Options Revisions, 5–21, 6–9, 6–25, 6–30, 6–32, 6–37
   Project Elements, 2–104
   Project Promotion Paths, 2–102, 2–110
   Promotion Path, 2–95
   Promotion Path Control Files, 2–98
   Promotion Path Members, 2–96
   Record Formats List, 3–77, 3–80, 3–82
   Record Type Titles, 2–44
   Remove Member (RVM), 2–18
   Restore Library, 2–121
   SAR Log Inquiry, 2–88
   SAR Log Transfer, 2–109
   Save Library, 2–120
   Single D/B Relation Entry, 3–73, 3–74
   Software Development Project, 2–101
   Software Scan and Replace, 4–169
   Software Transfer, 2–115
   Start Pass-Through, 5–22
   Universal File Converter, 6–8, 6–25
   User Defined Code Revisions, 2–42
User Defined Codes Window, 3–14, 3–18
User Information, 2–22, 5–4, 5–16
User Signon List Revisions, 2–21
Version Control, 2–5
Versions List, 6–9
Versions Setup, 5–20
Frequently used hidden selections, 1–4
FRF & JDE Msg File form (P98FRF), 3–34
Function key
  exits from screen design aid, 3–85
  indicator control, 3–89
Function key exits, 3–8
  install/reporting, 3–8
  list of defined fields, 3–90
  rebuild a file relationship, 3–8
  return to previous panel, 3–54
Function key exits from screen design aid, 3–85
Function key exits from the CASE profiles program, 2–80
Function key exits from the SAR log inquiry, 2–90
Function Key/Opt Definition form (P9601), 3–88
Function keys, 2–70
  access repository services, 3–86
  automatic reinquiry, 3–23
  checklists, 2–70
  data dictionary, 3–23
  data item cross reference, 3–23
  data item search, 3–23
  define soft coding (vocabulary overrides) fields, 3–91
  flow program/illustrate file models, 2–71
  format display control window, 3–85
  format keyword maintenance, 3–87
  function key/opt definition, 3–88
  J.D. Edwards command line, 2–70, 3–85
  member parameter/key list, 2–70
  optional files, 2–70
  repository services, 3–23
  screen and display format control, graphic, 3–97
  subfile drop, 3–86
  toggle monochrome/color display, 3–86
  user defined code tables, 3–23
  where used facility, 2–70
  window left, 3–92
  window right, 3–92
Function keys from file design aid
  access repository services, 3–45
  exiting data file design aid, 3–45
  field help on data item, 3–44
  J.D. Edwards command line, 3–44
  search by file, 3–45
Functional server, interactive program cycle, 4–140
Functional server error messages, graphic, 4–136
Functional server highlights, 4–127
  basic accounting transactions, 4–127
Functional server interface, 4–136
  graphic, 4–135
Functional server parameters, 4–137
Functional server program sections, graphic, 4–146
Functional server transaction data, graphic, 4–136
Functional server user index, 4–138
Functional server user space, 4–137
Functional server/COPY modules, 4–138
Functional servers, 4–123
  advantages, 4–125
  available, 4–151
  call parameters, 4–141
  COPY modules, 4–138
  disadvantages, 4–126
  highlights, 4–127
  how they function, 4–126
  parameters, 4–137
  user indexes, 4–138
  user space, 4–137
  what are they, 4–124
Functions, navigation, 2–69
Functions and options, 6–23
  add instructions, 6–23
  delete records, 6–24
  file field description, 6–24
  suppress from, 6–23
  suppress to, 6–23
  user defined text, 6–24

G

General aesthetics, 3–73
  alignment, 3–73
  grouping fields, 3–74
spacing, 3–74
Glossary Group, defined, 3–20, 3–28
Graphics
   library naming conventions, 1–12
   project attributes, 2–93
   SAR log, 2–108
   version control, 2–1, 2–2
   version control menu overview, 2–5
   version control process flow, 2–4
Group job window
   accessing, 5–5
   advanced functions, 5–13
   entering commands, 5–13
   hidden selections, 5–13
   working with, 5–3
Group job window summary, 5–14
Group jobs, 5–1
   activating suspended, 5–7
   signing off with suspended, 5–10
   terminating, 5–8
   working with non-J.D. Edwards, 5–11
Guidelines, 2–99, 4–20
   scan and replace, 4–170

H
Help Text Program, defined, 3–22
Hidden fields, 3–90
Hidden selections, 1–4
How does a file server function?, 4–110
How does a functional server function?, 4–126
How does a user index function?, 4–77
How does a user space function?, 4–69

I
IBM pass-through
   setting up access to remote locations, 5–20
   using with group jobs, 5–22
   working with, 5–19
Identifiers
   file information, 2–53
   maintenance on a logical file, 2–52
   member, 2–50
   member relationship and compiling information, 2–52
   processing a screen, 2–53
   type, use, and associated systems, 2–51
   where members are maintained, 2–54
Implementing a file server, 4–114
Index line (COORIX), error message, 4–145
Indicator Control form (P928400), 3–116
Indicator usage, 4–17
Initiating, source debugger, 4–155, 4–157
Input specifications, 4–7
Install/reporting, function key exits, 3–8
Interactive non-subfile program, graphic, 4–28
Interactive program cycle using a functional server, 4–140
Internal RPG subroutines within J.D. Edwards programs, 4–26
Item description, graphic, 3–107
Item Maintenance form (P92700), 3–59, 3–60
Item Search form (P928200), 3–54, 3–55, 3–78

J
J.D. Edwards group job window summary, 5–14
J.D. Edwards product line, 1–5
   architecture, engineering, construction, and real estate, 1–6
   distribution/logistics, 1–5
   energy and chemical, 1–6
   financials, 1–5
   manufacturing, 1–6
   other integrated solutions, 1–7
   public services: state and local governments, education, and utilities, 1–7
J.D. Edwards regional offices and worldwide offices, 1–8
J.D. Edwards training environment, 1–12
J98INIT, defining access for a user profile, 2–22
J98INITA, defining access for a user profile, 2–20
JDE open application architecture, graphic, 4–133
JDE Passthru Network form (B98P), 5–22
JDE Visual Debug form (P3701), 4–156
JDE Visual Debug form (P93701), 4–159, 4–160, 4–162
Job groups, creating new, 5–6
Join logical files, 2–66

K

Key lists, searching for, 4–115
Keywords, 6–20
business unit, 6–20
check data dictionary, 6–21
data dictionary default, 6–20
dates, 6–20
default constant, 6–20
initialize, 6–20
next number, 6–21
terminal ID, 6–20
user defined code lookup, 6–20

L

Libraries
CLITSEC, 2–9
copying, 2–23
creating, 2–8, 2–12
creating a development source, 2–16
creating common, 2–13
creating development object, 2–14
data, 2–9
install, 2–9
J.D. Edwards, 2–8
JDEINSTAL, 2–9
JDFDATA, 2–9
JDFINS, 2–9
JDFOBJ, 2–8
JDFSRC, 2–8
object, 2–8
plans, 2–9
security, 2–9
source, 2–8
understanding development source, 2–15
Library List Revisions form (P0094), 2–20
List of Defined Fields form (P92540), 3–90, 3–116
Literal field, adding, 3–64
Load Transferred Library form (P98312), 2–125
Loading the transfer library, 2–124
Locate
a project, 2–100
promotion path, 2–95
Locating a data item name, 3–18
Locating the next numbers facility, 3–31
Locating the rebuild FRF and JDE Msg file form, 3–34
Logic
data integrity, 4–130
user interface, 4–130
Logical file, sample, 3–43
Logical files, 2–65
join, 2–66
Logical files with omits, sample, 3–44
Logical files with selects, sample, 3–43

M

Maintain User Default SAR Info form
(P9812), 2–85
Manage Projects form (P92413), 2–100
Manage Promotion Paths form (P92403), 2–94
Manufacturing, J.D. Edwards product line, 1–6
Member identifiers, 2–50
Menu flow, 3–39
Merge functions for PTFs and reinstalls, 3–47
Message file
J.D. Edwards, 3–34
locating rebuild FRF & JDE Msg, 3–34
rebuilding, 3–34
Miscellaneous items, 4–21
key list (KLIST), 4–21
naming conventions, 4–21 Modules, copy, 2–67
N

Naming conventions, 2–57
Naming conventions for objects, 2–59
Navigation functions, 2–69
  access repository services, 2–69
  automatic reinquiry, 2–69
  next member, 2–70
  position cursor to action code, 2–69
  previous member, 2–69
Next Numbering Index Number, defined, 3–23
Next Numbers by Company/Fiscal form (P00021), 3–32
Next numbers facility
  locating, 3–31
  working with, 3–31
  working with by company & fiscal year, 3–32
Next Numbers form (P0002), 3–31
Non-group mode, changing to, 5–9
Number of Array Elements, defined, 3–21

O

Object cross reference repository, working with, 3–11
Object Cross Reference Repository form (P980014), 3–12, 3–14
Open application architecture, graphic, 4–134
Optional files workbench, 2–64
Optional Files Workbench form (P98290), 2–64
Other integrated solutions, J.D. Edwards product line, 1–7
Output specifications, 4–9

P

Parameters
  call, for the functional server, 4–141
  control, 4–111
  functional servers, 4–137
  returned, 4–113
Performance issues, 4–171
Pick list, placing fields on a form using, 3–84
Placing fields on a form using a pick list, 3–84
Pre Promotion Edit Details form (P92431), 2–111
Pre Promotion Edit History form (P9243), 2–111
Prefix standards, 3–51
Prepare, SAR system, 2–119
Print Transfer Report form (P98312), 2–122
Printing a report, 6–29
Printing the transfer library report, 2–122
Process overview, revising vocabulary and function keys, graphic, 3–97
Process overview – placing selected fields, 3–94
Process overview – revising the field definition, 3–95
Processing Options Revisions form (P98312), 5–21, 6–9, 6–25, 6–30, 6–32, 6–37
Production and development examples, 2–10
  basic development environment, 2–10
  basic production environment, 2–10
  common shared library, 2–10
  no source in production environment, 2–10
  one source and object library, 2–11
Profiles
  defining access for using J98INIT, 2–22
  defining access for using J98INITA, 2–20
  understanding user, 2–20
Program and file names, 2–63
Program being debugged, executing, 4–156
Program environment, determining, 4–154, 4–157
Program generator
  creating JDESRC, 2–17
  creating JDESRC without, 2–19
Program overview, 4–11
Program specifications, 4–3
Program Structure, 4–25
Programming standards, 4–1
Programming tools, 3–1
Programs and IDs
  0031 (crossover rules), 6–13, 6–14, 6–16, 6–17, 6–23
  00312 (add cross over instructions), 6–18
  B98P (JDE passthru network), 5–22
CALL (call program), 5–12
CHGLIBL (change library list), 5–11
CPYF (copy file), 2–17, 2–25
CRTLIB (create library), 2–16
CRTSRCPF (create source physical file), 2–19
G92 (computer assisted design), 2–76
G9261 (version control menu), 2–5
G9362 (developer's workbench), 2–87
G9841 (universal file converter), 6–8, 6–25
P0002 (next numbers), 3–31
P00021 (next numbers by company/fiscal), 3–32
P00051 (detail spec. types), 2–39
P00051 (user defined code revisions), 2–42
P0092 (user information), 2–22, 5–4, 5–16
P0093 (user signon list revisions), 2–21
P0094 (library list revisions), 2–20
P08332 (single d/b relation entry), 3–73, 3–74
P3701 (JDE visual debug), 4–156
P48002 (detail spec. over titles), 2–40
P48002 (record type titles), 2–44
P81QM (user defined codes window), 3–14, 3–18
P90630 (Copy DD, VO, DW, UDC, SVR, Menus), 2–127
P92001 (data item glossary revisions), 3–27, 3–29, 6–36
P9201 (data dictionary), 3–19, 6–36
P9202 (data field descriptions), 3–30
P9204 (data field alias), 3–24
P92102 (data file design aid), 3–40, 3–43, 3–44
P9240 (promotion path), 2–95
P92401 (promotion path members), 2–96
P92402 (promotion path control files), 2–98
P92403 (manage program paths), 2–94
P9241 (software development project), 2–101
P92411 (project promotion paths), 2–102
P92411W (project promotion paths), 2–110
P92412 (project elements), 2–104
P92412 (edit and promote), 2–108
P924124 (software transfer), 2–115
P924127 (control files copy), 2–116
P92413 (manage projects), 2–100
P9242 (SAR log transfer), 2–109
P9243 (pre promotion edit history), 2–111
P92431 (pre promotion edit details), 2–111
P92510 (create new screen), 3–58
P92520 (record formats list), 3–77, 3–80, 3–82, 3–114
P92522 (file selection list), 3–83
P92524 (field selection list), 3–81, 3–83
P92537 (format keyword maintenance), 3–86
P92540 (list of defined fields), 3–90, 3–116
P92590 (design aid exit/save), 3–69
P92700 (item maintenance), 3–59, 3–60
P928200 (item search), 3–54, 3–55, 3–78
P928400 (field definition report), 3–111, 3–113, 3–114
P928400 (indicator control), 3–116
P928400 (maintain vocabulary override fields), 3–117
P93701 (JDE visual debug), 4–159, 4–160, 4–162
P9601 (function key/opt definition), 3–88
P980014 (cross reference), 2–71, 3–9
P980014 (Object Cross Reference Repository), 3–12, 3–14
P98009 (CASE Profiles), 2–77
P98009 (CASE profiles), 2–84
P98042 (data modeling), 3–5, 3–6, 3–10
P9810 (SAR log inquiry), 2–88
P98101 (copy data files), 2–24
P9812 (maintain user default SAR info), 2–85
P98290 (optional files workbench), 2–64
P98300 (versions list), 6–9
P98300 (versions setup), 5–20
P98312 (build transfer library), 2–118
P98312 (create user data libraries), 2–13
P98312 (load transferred library), 2–125
P98312 (print tranfer report), 2–122
P98312 (processing options revisions), 5–21, 6–9, 6–25, 6–30, 6–32, 6–37
P98810 (software scan and replace), 4–169
P98FRF (FRF & JDE Msg file), 3–34
P99630 (copy DD, VO, DW, UDC, SVR, Menus), 2–27
RMVM (remove member), 2–18
RSTLIB (restore library), 2–121
SAVLIB (save library), 2–120
STRPASTHR (start pass-through), 5–22
Project
  define, 2–119
defining, 2–99
  promote, 2–119
promoting, 2–107, 2–114
to add, 2–101
to locate, 2–100
to promote, 2–114
Project Elements form (P92412), 2–104
Project Promotion Paths form (P92411), 2–102
Project Promotion Paths form (P92411W), 2–110
Project SABs, to assign, 2–104
Project updates, promoting, 2–117
Promote, project, 2–114, 2–119
Promoting a project, 2–107, 2–114
  See also Defining a promotion path
Promoting project updates, 2–117
Promotion path
  define, 2–119
  validate, 2–119
validating, 2–110
Promotion Path Control Files form (P92402), 2–98
Promotion Path form (P9240), 2–95
Promotion Path Members form (P92401), 2–96
Promotion paths
defining, 2–94
to add, 2–95
to assign, 2–102
to define for control tables, 2–98
to define for source code members, 2–96
to locate, 2–95
to validate, 2–110
understanding, 2–92
Promotion paths and projects, working with, 2–91
PTFs, merge functions, 3–47
Public services: state and local
governments, education, and utilities, J.D.
Edwards product line, 1–7

R

RDA and DREAM Writer, graphic, 3–102
Reading from a user space, 4–74
Rebuild a file relationship, function key
exists, 3–8
Record formats, 3–78
Record Formats List (P92520), 3–114
Record Formats List form (P92520), 3–77,
3–80, 3–82
Record type codes, creating, 2–42
Record type titles
  defining, 2–44
  verifying, 2–46
  See also Creating SARs
Record Type Titles form (P48002), 2–44
Record types
  adding, 2–39
  chaining, 2–40
  copying, 2–27
Records
  copying, 2–25
deleting, 6–19
Reinstalls, merge functions, 3–47
Release Number, defined, 3–19
Remove Member form (RMVM), 2–18
Report
  compiling, 3–118
  printing, 6–29
  scan and replace, 4–170
Report design aid, 3–101
accessing, 3–110
Report design aid function keys
display all defined fields, 3–116
format display control, 3–113
indicator control window, 3–116
maintain vocabulary override fields,
3–117
record formats list, 3–114
repository services, 3–114
understanding, 3–113
window left, 3–117
window right, 3–117
Report design standards, 3–108
  general aesthetics, 3–108
  J.D. Edwards standards/record formats, 3–108
  RDA features, 3–108
Report formats, 3–106
Report program without subheadings, graphic, 4–30
Reports
  Field Definition, 3–111, 3–113, 3–114
  Print Install Records, 2–124
  Record Formats List, 3–114
  Software Scan and Replace, 4–170
  Universal File Converter, 6–29
Restore Library form (RSTLIB), 2–121
Restoring the transfer library from tape, 2–121
Retrieving data from a user index, 4–84
Returned parameters, 4–113
Review an RPG program's source, 4–32
RPG program's source, review, 4–32
Rules for creating development environments, 2–7

S

SAR information, selecting types to log, 2–86
SAR log, 2–83
SAR log inquiry
  accessing, 2–87
  function key exits, 2–90
  selection exits, 2–89
  summary, 2–90
SAR Log Inquiry form (P9810), 2–88
SAR Log Transfer form (P9242), 2–109
SAR logging, setting up user input options, 2–84
SAR system
  setup, 2–41
  to prepare, 2–119
SARs, updating, 2–108
Save Library form (SAVLIB), 2–120
Saving the transfer library to tape, 2–120
Scan and replace, 4–169
  guidelines, 4–170
  report, 4–170
  working with, 4–169
Screen design aid, 3–49
  summary, 3–98
  working with, 3–54
Screen design standards and tips, 3–71
  alpha fields, 3–72
  default cursor, 3–72
  description fields, 3–72
  fold area, 3–72
  line 24, 3–71
  title, 3–71
  window, 3–71
SDDExit/save function key, 3–69
Search, conducting, 3–11, 3–12
Searching for key lists, 4–115
Select all function key, understanding, 3–82
Selected fields, placing, 3–94
Selecting database fields, 3–80
Selecting types of SAR information to log, 2–86
Selection exits
  data modeling, 3–9
  display, 3–9
  fields, 3–10
  move top, 3–9
  software versions repository, 2–72
  where used, 3–9
Selection exits from the scheduling workbench, 2–38
  processing options, 2–39
Setting the break point, source debugger, 4–160
Setting up universal file converter, 6–8
Setting up user input options for SAR logging, 2–84
Signing off with suspended group jobs, 5–10
Signing on and off, 1–1
Single D/B Relation Entry form (P08332), 3–73, 3–74
Software Versions Repository form
  (P9801), 2–88
Software Development Project form
  (P9241), 2–101
Software Scan and Replace form (P98810), 4–169
Software Scan and Replace report (P98810), 4–170
Software Transfer form (P924124), 2–115
Software Versions Repository
  See also SVR
working with, 2–47
Software versions repository
accessing, 2–49
selection exits, 2–72
Software Versions Repository form (P9801),
Source debugger, 4–153
add breakpoint, 4–162
add breakpoint with prompt, 4–163
change program variable, 4–165
command line window, 4–162, 4–167
continue execution, 4–160
continue processing, 4–162
display indicator values, 4–166
display program variable, 4–164
execute the program, 4–159
features, 4–162
initiating, 4–155, 4–157
move line to top of page, 4–165
remove current breakpoint, 4–165
scan backward, 4–166
scan forward, 4–166
set the break point, 4–160
Standard screen function keys, 1–3
Standards
field name, 3–52
prefixes, 3–51
screen design, 3–71
Start Pass-Through form (STRPASTHR),
5–22
Subfile boundaries, changing, 3–93
Subfile program with selection exits,
graphic, 4–29
Subroutines, 4–12
common, 4–13
internal RPG, within J.D. Edwards
programs, 4–26
standard, 4–12
Summary of CASE profiles, 2–81
Summary of J.D. Edwards attention MENU
window functions, 5–18
Summary of screen design aid, 3–98
Summary of the SAR log inquiry, 2–90
Suspended group jobs, signing off with,
5–10
SVR. See Software Versions Repository
System Code, defined, 3–20
System Code – Next Numbers, defined,
3–23
System Code/Reporting, defined, 3–20
System integration, 1–15
features, 1–15

T

Terminating group jobs, 5–8
The call parameters for the functional
server, 4–141
The function keys for the data dictionary,
3–23
Tips when using file servers, 4–116
To work with software scan and replace,
4–169
Tracking information if writing variable
length records, 4–73
Traditional architecture, graphic, 4–129,
4–132
Training environment, 1–12
classes, 1–14
library naming conventions, 1–12
signon naming conventions, 1–12
student library setup, 1–12
Transfer library
creating, 2–118
loading, 2–124
restoring from tape, 2–121
saving to tape, 2–120
Transfer library report, printing, 2–122
Transferring individual control table
records, 2–127
Types of file servers, 4–108

U

Understanding development source
libraries, 2–15
Understanding promotion paths, 2–92
Understanding the data dictionary
structure, 3–16
Understanding the report design aid
function keys, 3–113
Understanding the SDA exit/save function
key, 3–69
Understanding the select all function key,
3–82
Understanding the universal file converter setup, 6–8
Understanding user profiles, 2–20
Universal building blocks of J.D. Edwards software, 1–10
Universal file converter, 6–1
  business unit, 6–4
  check data dictionary, 6–4
  data dictionary default, 6–4
  database considerations, 6–5
dates, 6–4
default constant, 6–5
graphic, 6–2, 6–3
initialization, 6–4
next number, 6–4
numeric fields, 6–4
setting up, 6–7, 6–8
special processing, 6–4
  understanding set up, 6–8
  user defined code lookup, 6–4
  user responsibilities, 6–5
Universal File Converter form (G9841), 6–8, 6–25
Universal File Converter report (P98300), 6–29
Updating a field in RDA, 3–111
Updating an existing field, 3–55
Updating the SARs, 2–108
Updating/adding fields through SDA, 3–53
Usage, indicator, 4–17
User Defined Code Revisions form (P00051), 2–42
User defined code window, 3–40
User Defined Codes Window form (P81QM), 3–14, 3–18
User defined help instructions, data dictionary, 3–29
User indexes, 4–75
  accessing, 4–140
  advantages, 4–76
  creating, 4–78, 4–79, 4–139
  functional servers, 4–138
  how does it function, 4–77
  retrieving data from, 4–84
  writing to, 4–81
User Information form (P0092), 2–22, 5–4, 5–16
User interface logic, 4–130
User responsibilities, 6–5
User Signon List Revisions form (P0094), 2–21
User space, 4–67, 4–68
  accessing, 4–139
  advantages, 4–68
  control fields, 4–144
  creating, 4–69, 4–139
  functional servers, 4–137
  how it functions, 4–69
  reading from, 4–74
  writing to, 4–72
Using *ALL, 3–67
Using *BOTH, 3–65
Using debugger with a batch program, 4–157
Using debugger with an interactive program, 4–154
Using IBM pass-through with group jobs, 5–20

V

Validate, promotion path, 2–110, 2–119
Validating a promotion path, 2–110
Value for Entry – Default, defined, 3–22
Variable length records, tracking information, 4–73
Version control, 2–1
Version control menu, 2–5
Version Control Menu form (G9261), 2–5
Version control menu overview, 2–5
Versions List form (P98300), 6–9
Versions Setup form (P98300), 5–20
Video, compiling, 3–70
Video constant field (VCO), adding, 3–63
  Video fields
    adding, 3–77
    record formats list, 3–77

W

W.O. Detail form, accessing, 2–30
What are calculation specifications?, 4–8
What are control parameters?, 4–111
What are control specifications?, 4–4
  What are extension specifications?, 4–6
  What are file description specifications?, 4–5
What are functional servers?, 4–124
What are input specifications?, 4–7
What are output specifications?, 4–9
What are returned parameters?, 4–113
What are the advantages of using a file server?, 4–109
What are the advantages of using a functional server?, 4–125
What are the advantages of using a user index?, 4–76
What are the advantages of using a user space?, 4–68
What are the data file design aid standards, 3–45
What are the disadvantages of using a file server?, 4–109
What are the disadvantages of using a functional server?, 4–126
What are the report design standards, 3–108
What are the report formats?, 3–106
What is a file server?, 4–108
What is a user space?, 4–68
Windows, 2–68
Work fields, 4–22
option files, 4–22
Work with non-J.D. Edwards group jobs, 5–11
Workbench, optional files, 2–64
Working with data field descriptions, 3–30
Working with data item alias revisions, 3–24
Working with data modeling, 3–3
Working with file conversion, 6–25
Working with next number by company and fiscal year, 3–32
Working with object cross reference repository, 3–11
Working with promotion paths and projects, 2–91
Working with screen design aid, 3–54
Working with Software Versions Repository, 2–47
Working with the crossover rules form, 6–13
Working with the data dictionary, 3–19
Working with the data dictionary glossary, 3–25
Working with the data dictionary glossary by file, 6–36
Working with the next numbers facility, 3–31

Working with user defined help instructions, 3–29
Writing to a user indexes, 4–81
Writing to a user space, 4–72
Exercises