



JD Edwards World

Advanced Programming

Concepts & Skills Guide

Version A9.1

Revised - January 15, 2007

Copyright © 2006, Oracle. All rights reserved.

The Programs (which include both the software and documentation) contain proprietary information; they are provided under a license agreement containing restrictions on use and disclosure and are also protected by copyright, patent, and other intellectual and industrial property laws. Reverse engineering, disassembly, or decompilation of the Programs, except to the extent required to obtain interoperability with other independently created software or as specified by law, is prohibited.

The information contained in this document is subject to change without notice. If you find any problems in the documentation, please report them to us in writing. This document is not warranted to be error-free. Except as may be expressly permitted in your license agreement for these Programs, no part of these Programs may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose.

If the Programs are delivered to the United States Government or anyone licensing or using the Programs on behalf of the United States Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS

Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are “commercial computer software” or “commercial technical data” pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the Programs, including documentation and technical data, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement, and, to the extent applicable, the additional rights set forth in FAR 52.227-19, Commercial Computer Software--Restricted Rights (June 1987). Oracle Corporation, 500 Oracle Parkway, Redwood City, CA 94065.

The Programs are not intended for use in any nuclear, aviation, mass transit, medical, or other inherently dangerous applications. It shall be the licensee's responsibility to take all appropriate fail-safe, backup, redundancy and other measures to ensure the safe use of such applications if the Programs are used for such purposes, and we disclaim liability for any damages caused by such use of the Programs.

The Programs may provide links to Web sites and access to content, products, and services from third parties. Oracle is not responsible for the availability of, or any content provided on, third-party Web sites. You bear all risks associated with the use of such content. If you choose to purchase any products or services from a third party, the relationship is directly between you and the third party. Oracle is not responsible for: (a) the quality of third-party products or services; or (b) fulfilling any of the terms of the agreement with the third party, including delivery of products or services and warranty obligations related to purchased products or services. Oracle is not responsible for any loss or damage of any sort that you may incur from dealing with any third party.

Oracle, JD Edwards, PeopleSoft, and Retek are registered trademarks of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

Open Source Disclosure

Oracle takes no responsibility for its use or distribution of any open source or shareware software or documentation and disclaims any and all liability or damages resulting from use of said software or documentation. The following open source software may be used in Oracle's PeopleSoft products and the following disclaimers are provided.

This product includes software developed by the Apache Software Foundation (<http://www.apache.org/>). Copyright (c) 1999-2000 by The Apache Software Foundation. All rights reserved. THIS SOFTWARE IS PROVIDED “AS IS” AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Send Us Your Comments

JD Edwards World Release A9.1 Documentation, Revised - January 15, 2007

JD Edwards World welcomes your comments and suggestions on the quality and usefulness of this publication. Your input is an important part of the information used for revision.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where?
- Are the examples correct? Do you need more examples?
- What features did you like most about this manual?

If you find any errors or have any other suggestions for improvement, please indicate the title and part number of the documentation and the chapter, section, and page number (if available). You can send comments to us by e-mail at:

jde_world_doc_ww@oracle.com

If you would like a reply, please give your name, address, telephone number, and electronic mail address (optional).

Contact a JD Edwards World representative by calling Oracle Global Support Center at 1-800-289-2999 for current information or if you have any questions regarding this document.

Contents

1 Overview

Overview to JD Edwards World 1-1

Application Development Cycle..... 1-1

Universal Building Blocks of JD Edwards World Software..... 1-1

2 APCS System Overview

Overview to APCS System..... 2-1

Features 2-1

3 Version Control

Overview to Version Control 3-1

About Version Control..... 3-1

Version Control Process Flow 3-3

Version Control Menu Overview 3-4

Development Environment..... 3-5

About a Development Environment..... 3-5

JD Edwards World Libraries 3-6

Production and Development Examples..... 3-8

Creating Libraries 3-11

Creating a Development Source Library 3-13

About User Profiles 3-16

Defining Access for a User Profile using J98INITA 3-17

Copy Data to Your Development Environment 3-18

Project Management..... 3-25

About Project Management..... 3-25

Understanding Work Order Processing 3-25

Creating Work Orders..... 3-26

Accessing the Scheduling Workbench..... 3-33

Adding Record Types 3-36

Changing Record Types..... 3-37

Work with Software Action Requests	3-39
About SAR System Setup	3-39
Creating Record Type Codes	3-39
Defining Record Type Titles	3-41
Work with Software Versions Repository	3-43
Working with Software Versions Repository (SVR)	3-43
Accessing the Software Versions Repository	3-45
Member Identifiers	3-45
Naming Conventions	3-52
The JD Edwards World System Codes	3-56
Examples of Program and File Names	3-61
Optional Files Workbench	3-62
Navigation Functions	3-64
CASE Profiles	3-71
About CASE Profiles	3-71
Accessing CASE Profiles	3-71
Function Key Exits from the CASE Profiles Program	3-75
Summary of CASE Profiles	3-75
Working with SAR Log	3-77
About SAR Log	3-77
Setting Up User Input Options for SAR Logging	3-78
Selecting Types of SAR Information to Log	3-79
Accessing SAR Log Inquiry	3-80
Summary of the SAR Log Inquiry	3-83
Work with Promotion Paths and Projects	3-85
Working with Promotion Paths and Projects	3-85
Understanding Promotion Paths	3-85
Defining a Promotion Path	3-87
Defining a Project	3-92
Promote a Project	3-97
Promoting a Project	3-97
Update the SARs	3-97
Validating a Promotion Path	3-99
Promoting a Project	3-102
Promote Project Updates	3-105
Promoting Project Updates	3-105

4 Programming Tools

Overview to Programming Tools	4-1
About Programming Tools.....	4-1
Work with Data Modeling	4-3
Working with Data Modeling.....	4-3
Accessing Data Modeling.....	4-3
Work with the Object Cross Reference Repository	4-9
Working with the Object Cross Reference Repository	4-9
Work with Data Dictionary	4-13
About the Data Dictionary Repository	4-13
Understanding the Data Dictionary Structure.....	4-13
Locating A Data Item Name.....	4-15
Working with the Data Dictionary.....	4-16
Working with Data Item Alias Revisions.....	4-21
Working with the Data Dictionary Glossary	4-22
Working with User Defined Help Instructions	4-26
Working with Data Field Descriptions.....	4-27
Working with the Next Numbers Facility.....	4-27
About the Field Reference File.....	4-29
About the JD Edwards World Message File	4-30
Locating the Rebuild FRF and JD Edwards World Msg File Form.....	4-31
Work with Data File Design Aid	4-33
About the Data File Design Aid.....	4-33
About Assigning the File Prefix.....	4-34
Entering Data File Design Aid	4-35
Function Keys From File Design Aid.....	4-38
What are the Data File Design Aid Standards?	4-40
Data File Design Aid Summary	4-41
Work with Screen Design Aid	4-43
About Screen Design Aid	4-43
Editing Commands.....	4-44
Prefix Standards.....	4-45
Field Name Standards.....	4-46
Updating or Adding Fields through SDA.....	4-47
Working with Screen Design Aid.....	4-47
Function Key Exits.....	4-48
Updating an Existing Field.....	4-48
Accessing Fast Path Create for a New Screen.....	4-51

Adding Fields without Using a Pick List	4-54
Adding a Literal Field.....	4-56
Using the *BOTH and *ALL Features.....	4-56
Compiling Your Screen.....	4-60
Screen Design Standards and Tips.....	4-60
Adding Screen Fields Using Pick List.....	4-64
Selecting Database Fields	4-66
Placing Fields on a Screen Using a Pick List.....	4-70
Function Key Exits from Screen Design Aid	4-70
Changing Subfile Boundaries	4-77
Process Overview - Placing Selected Fields.....	4-78
Process Overview - Revising the Field Definition	4-80
Process Overview - Revising Vocabulary and Function Keys	4-81
Function Keys for Screen and Display Format Control	4-81
Summary of Screen Design Aid	4-82
Work with Report Design Aid.....	4-83
About Report Design Aid.....	4-83
Accessing Report Design Aid	4-90
Updating a Field in RDA.....	4-90
Compiling A Report.....	4-96
Changing the Compile Option Defaults for Reports.....	4-97
5 Programming Standards	
Overview to Programming Standards	5-1
Programming Standards	5-1
Program Specifications	5-3
About Program Specifications	5-3
What Are Header (Control) Specifications?.....	5-4
What Are File Description Specifications?.....	5-4
What Are Extension Specifications?	5-5
What Are Input Specifications?.....	5-5
What Are Calculation Specifications?.....	5-6
What Are Output Specifications?.....	5-7
6 Program Overview	
Program Overview	6-1
About the Program Overview	6-1
Subroutines.....	6-1
Error Handling.....	6-3
Indicator Usage.....	6-4

Documentation.....	6-5
Miscellaneous Items	6-7
Program Structure	6-11
About Program Structure	6-11
User Spaces.....	6-49
About User Spaces.....	6-49
What Is a User Space?	6-49
What Are the Advantages of Using a User Space?	6-49
How Does a User Space Function?.....	6-50
Creating a User Space.....	6-50
Writing to a User Space.....	6-53
Reading from a User Space.....	6-54
User Indices	6-57
About User Indices	6-57
What Are the Advantages of Using a User Index?	6-58
How Does a User Index Function?.....	6-58
Creating a User Index.....	6-59
Writing to a User Index.....	6-62
Appearance of Records	6-63
Retrieving Data from a User Index.....	6-64
File Servers	6-87
About File Servers.....	6-87
What is a File Server?	6-87
What are the Advantages of Using a File Server?	6-88
What are the Disadvantages of Using a File Server?	6-88
How Does a File Server Function?.....	6-89
What Are Control Parameters?	6-89
What Are Returned Parameters?	6-92
Implementing a File Server	6-92
Searching for Key Lists.....	6-93
Tips when Using File Servers.....	6-94
Commonly Used File Servers.....	6-99
Functional Servers.....	6-101
About Functional Servers	6-101
What Are Functional Servers?.....	6-101
What Are the Advantages of Using a Functional Server?.....	6-102
What Are the Disadvantages of Using a Functional Server?.....	6-103
Setting Up Business Rules for an Entry Program.....	6-103
How Does a Functional Server Function?	6-103

Functional Server Highlights.....	6-104
The Call Parameters for the Functional Server.....	6-113
Control Fields within the User Space	6-116
Error Message Index Line (C00RIX)	6-117
Example - Functional Server Program Sections	6-117
Available Functional Servers	6-121
Source Debugger	6-123
About Source Debugger	6-123
Using Debugger with an Interactive Program	6-123
Using Debugger with a Batch Program.....	6-126
Features of the JD Edwards World Source Debugger	6-129
Software Scan and Replace.....	6-133
About Software Scan and Replace	6-133
Report.....	6-134
Performance Issues	6-135
DREAM Writer	6-135
7 Group Jobs	
Overview to Group Jobs	7-1
About Group Jobs.....	7-1
Access the JD Edwards World Group Job window.....	7-3
About the JD Edwards World Group Job window.....	7-3
Accessing the JD Edwards World Group Job window	7-4
Creating New Group Jobs	7-4
Activating Suspended Group Jobs.....	7-5
Terminating Group Jobs.....	7-6
Changing to Non-Group Mode	7-6
Signing Off with Suspended Group Jobs	7-6
Work with Non-JD Edwards World Group Jobs	7-7
Advanced Functions of the JD Edwards World Group Job window	7-8
Work with the Attention MENU window.....	7-11
About the Attention MENU Window	7-11
Accessing the JD Edwards World Attention Menu window	7-11
Work with IBM Pass-Through	7-15
About Working with IBM Pass-Through.....	7-15
Setting Up Access to Remote Locations	7-15

Using IBM Pass-Through with Group Jobs.....	7-16
8 Universal File Converter	
Overview to Universal File Converter	8-1
About Universal File Converter.....	8-1
Set Up Universal File Converter	8-7
About Universal File Converter (UFC).....	8-7
Understanding the Universal File Converter Setup.....	8-7
Setting Up Universal File Converter	8-9
Work with Crossover Rules	8-15
Working with the Crossover Rules Screen.....	8-15
Displaying Field Descriptions.....	8-18
Adding Fields.....	8-19
Deleting Records.....	8-20
Keywords.....	8-21
About the Conversion Rule Program.....	8-22
Available Functions and Options	8-23
Work with File Conversion	8-25
Working with File Conversion.....	8-25
Print a Report	8-29
Printing a Report.....	8-29
Create Conversion Versions.....	8-31
Creating Conversion Versions	8-31
Creating Conversion Forms	8-31
Work with the Data Dictionary Glossary by File	8-33
About Working with the Data Dictionary Glossary by File.....	8-33
Accessing the Data Dictionary Glossary by File.....	8-33
Adding a File Specific Glossary Item.....	8-34
Printing the Data Dictionary Glossary Information.....	8-34
9 Appendices	
Appendix A – Common & Production Library Files	9-1
Appendix B – Upgrading Customized Source Code.....	9-3
S/Compare	9-3

Harmonizer	9-4
About Harmonizer Plus	9-5
Appendix C – CL Models	9-7
Appendix D – Universal File Converter	9-11
About Universal File Converter	9-11
Gathering Information.....	9-11
Appendix E – Dates	9-17
About Date Programs	9-17
X0027	9-18
X0028	9-19
X0035	9-21
X98DAY	9-22

1 Overview

Overview to JD Edwards World

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* document.

Level 2

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

Level 3

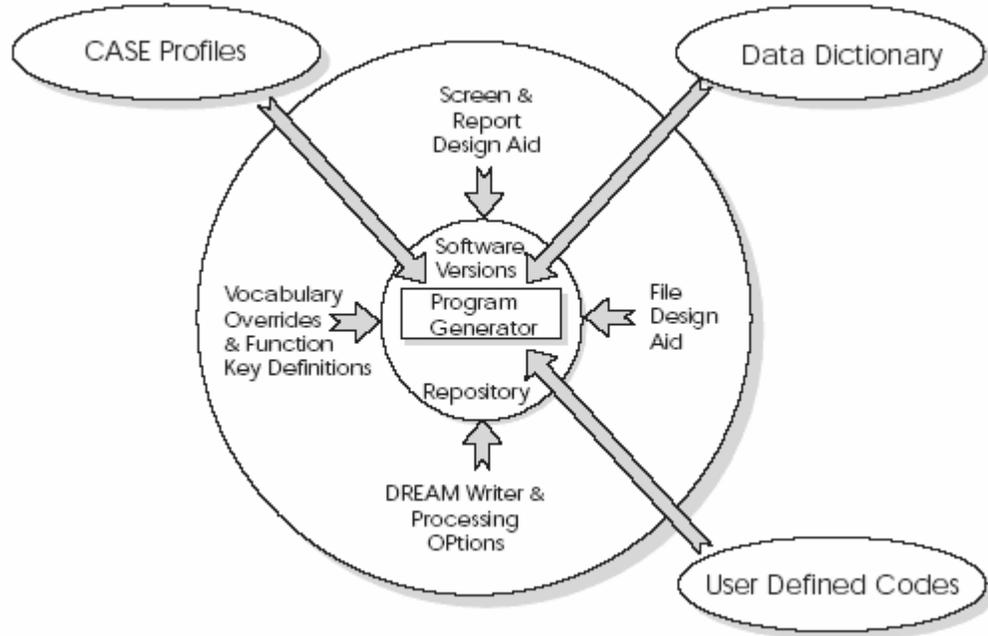
- The Development Platform, which is described in the *CASE* document.

Universal Building Blocks of JD Edwards World 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 JD Edwards World program.



2 APCS System Overview

Overview to APCS System

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
- JD Edwards World Programming Standards
- File Servers and Functional Servers
- User Spaces and User Indexes
- Group Jobs
- Programming Modifications
- Source Debugger
- Programming Impacts from Software Upgrades

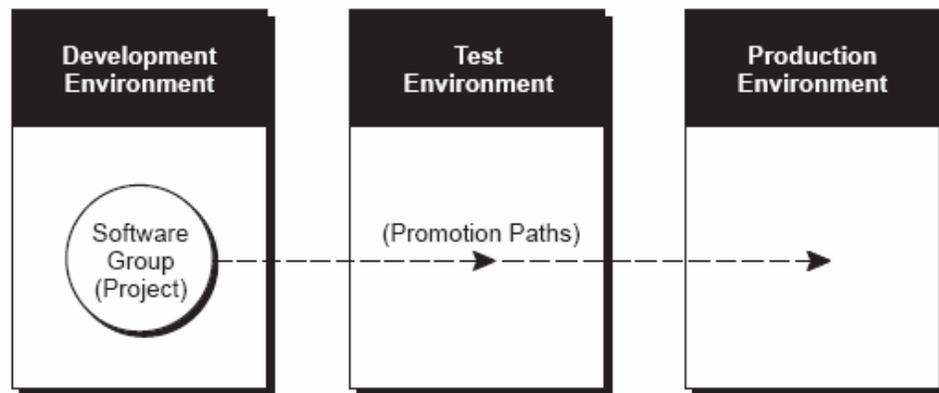
Note: Because this is a programming guide, and the majority of programmers do not use the Java interface, we are using non-Java (green screen) captures in this guide.

3 **Version Control**

Overview to Version Control

About Version Control

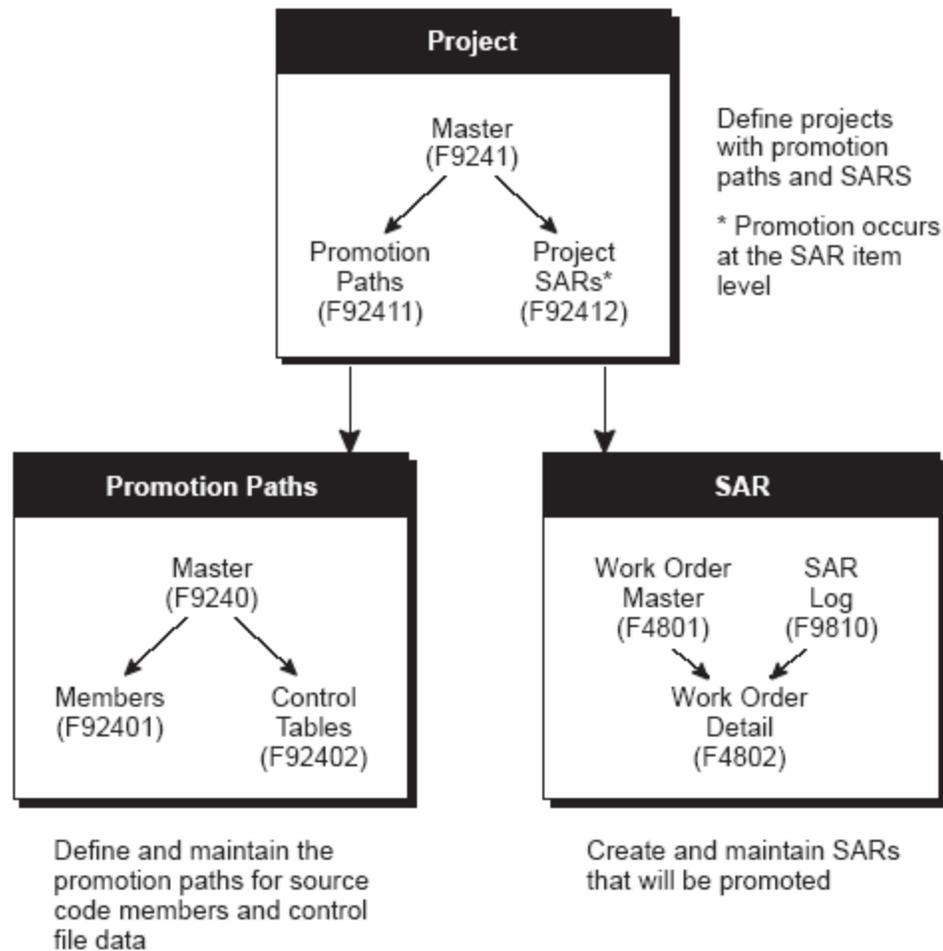
Use the JD Edwards World Version Control system to manage the movement of software between various environments, such as ones you have set up for software development, testing, and production.



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 files) 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.



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:

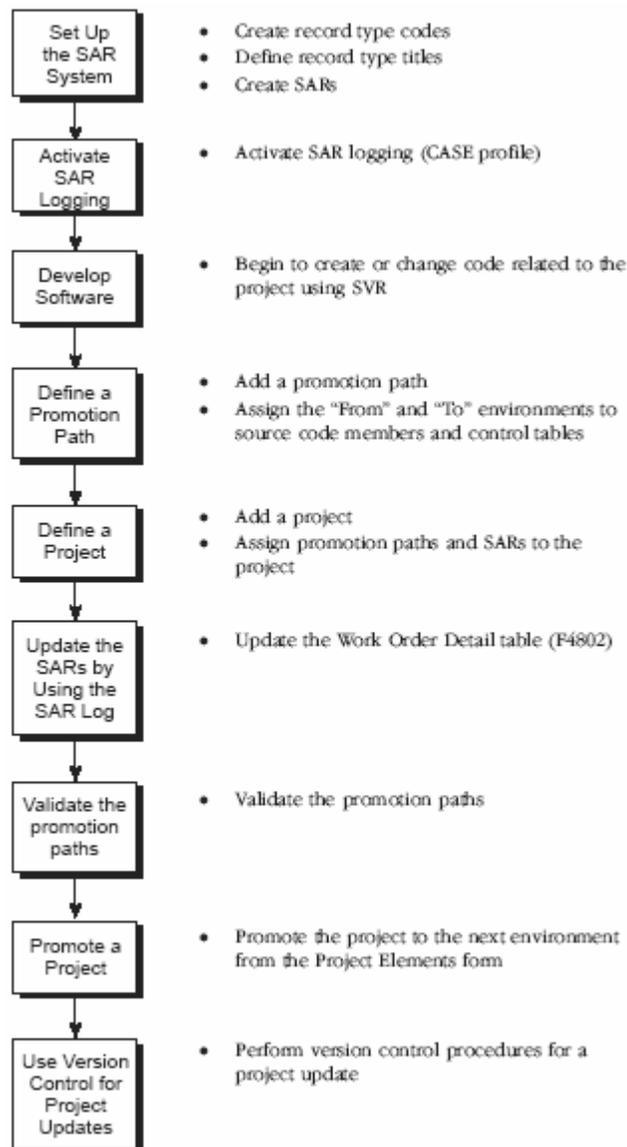
- Development Environment
- Project Management
- Work with Software Action Request
- Work with Software Versions Repository
- CASE Profiles

- Work with SAR Log
- Work with Promotion Paths and Projects
- Promote a Project
- Promote Project Updates

This section contains the following:

- [Version Control Process Flow](#)
- [Version Control Menu Overview](#)

Version Control Process Flow



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.

G9261 Daily Operation	J.D. Edwards & Company Version Control	JDED
... BASIC OPERATIONS	... SETUP	
2. Software Versions Repository	14. Record Type Codes	
3. Manage Promotion Paths	15. Record Type Titles	
4. Manage Projects	16. CASE Profiles	
... Double Byte Mandatory Options	... INQUIRIES	
7. Analysis Process	19. SAR Inquiry by Reference	
8. C9822 Conversion	20. Inquiry by SAR, Proj and Path	
... QA FUNCTIONS	... PURGE DATA FILES	
11. Edit and Promote	23. Purge SAR Log File	
12. Super SAR		
Selection or command ====>		
Thur, Apr 18, 1996 8:55:51am	A7.3 Development (C) J.D.Edwards & Co 1985,1996	LA5595234 QPADEV0014

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 JD Edwards World because of IBM standards and 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

This section contains the following:

- [JD Edwards World Libraries](#)
- [Production and Development Examples](#)
- [Creating Libraries](#)
- [Creating Common and Data Libraries](#)
- [Creating a Development Object Library](#)
- [Creating a Development Source Library](#)
- [Creating JDESRC for JD Edwards World Program Generator](#)
- [Creating JDESRC Without the Program Generator](#)
- [About User Profiles](#)

- [Defining Access for a User Profile using J98INITA](#)
- [Defining Access for a User Profile Using J98INIT](#)
- [Copy Data to Your Development Environment](#)
- [Copying a Library](#)
- [Copying a File](#)
- [Copying a Record](#)
- [Copying JD Edwards World Control Records](#)

JD Edwards World Libraries

The following libraries are delivered with JD Edwards World software:

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

Source Library (JDFSRC)

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

The file JDESRC contains the following source code:

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

The file JDECPY contains the source code for copy members.

The file F98CRTCMD contains pre-compiler commands.

- This is used to compile selected JD Edwards World programs.

Object Library (JDFOBJ)

The object library contains objects for your JD Edwards World software.

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

Data Library (JDFDATA)

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

Install Library (JDEINSTAL)

The install library is used to install programs and software that upgrade JD Edwards World software.

Plans Library (JDFINS)

The library is used to plan how to upgrade JD Edwards World software.

Security Library (CLTSEC)

You can create a security library which 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 may 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.

The World Technical Foundation manual states that a security library containing the F0092, F00921, F0093, F0094, and F0095 files can be used to minimize profile maintenance between JD Edwards World environments.

Note: There are many different security scenarios. For that reason, there is little documentation on the security library concept. This is just a discussion of how to use the Security library concept and is NOT a recommendation.

A security library makes most sense when J98INITA is the Initial Program on the IBM user profile. IBM object security may be necessary in addition to the JD Edwards World security options to complete the user security requirements.

1. The benefits of a security library are minimized if there is more than one. The maintenance and security tasks will have to be duplicated for each environment. If each environment has a different security scenario, a single security library should not be used. However, if separate security libraries are necessary, there will have to be a matching object library with the QJDF data area naming the security library in the User Profile Library field.
2. If the environments are not all at the same release (A73, A81, etc), the sign-on programs, the menu program, and the control files will all have to be at the highest release level.
3. If all the environments are at the same release level, other files besides the security files may be placed in the Security library for maintenance or control purposes (for example SVR, Function Key Security, Action code security). Any files in the Security library should apply to all environments and should not be found in any other user data library. The pristine JDFDATA library should still contain all the JD Edwards World files.
4. On an Upgrade Plan, the Security library should be specified as SEC type. The file duplication issues of paragraph 3 above must be emphasized. The Control File Sets in the Control File Dependencies appendix of the PTF Install Workbook should be maintained.

Production and Development Examples

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

Basic Production Environment

Library	Description
QTEMP	IBM Temporary data files
CLTOBJ	Client's objects
JDFOBJ	JD Edwards World objects
CLTCOM	Client's common files
CLTDTA	Client's data files
CLTSEC	Client's security files
QGPL	IBM general public library

Basic Development Environment

Library	Explanation
QTEMP	IBM Temporary data files

Library	Explanation
DEVOBJ	Development objects
CLTOBJ	Client's objects
JDFOBJ	JD Edwards World objects
DEVCOM	Development common files
DEVDTA	Development data files
CLTSEC	Client's security files
DEVSRC	Development source files
CLTSRC	Client's source files
JDFSRC	JD Edwards World source files
QGPL	IBM general public library

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

Library	Explanation
QTEMP	IBM Temporary data files
CLTOBJ	Client's objects
JDFOBJ	JD Edwards World objects
CLTCOM	Client's common files
COMMON	Common (shared) files
CLTDTA	Client's data files
CLTSEC	Client's security files
QGPL	IBM general public library

Basic Development Environment with a Shared Common

Library	Explanation
QTEMP	IBM Temporary data files
DEVOBJ	Development objects
CLTOBJ	Client's objects
JDFOBJ	JD Edwards World objects
DEVCOM	Development common files
COMMON	Common (shared) files
DEVDTA	Development data files
CLTSEC	Client's security files
DEVSRC	Development source files
CLTSRC	Client's source files
JDFSRC	JD Edwards World source files
QGPL	IBM general public library

No source libraries exist in the production environment because source code is not necessary to run JD Edwards World 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 may be changed during the testing process. For example, F98HELP may not be changed but the Vocabulary Overrides, Data Dictionary, User Defined Codes or other Control Files may be changed. 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

Library	Explanation
QTEMP	IBM Temporary data files
CLTMOD	Client's source and objects under modification
CLTOBJ	Client's objects
JDFOBJ	JD Edwards World objects
DEVCOM	Development common files
COMMON	Common unchanged files

Library	Explanation
DEVDTA	Development data files
CLTSEC	Client's security files
CLTSRC	Client's source files
JDFSRC	JD Edwards World source files
QGPL	IBM general public library

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 copied to CLTOBJ and source code is moved to CLTSRC. The purpose of having one object and source code library like CLTMOD is to aid in change management and simplify the development library list by having one place where all modifications and testing take 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 another development environment. If you do not specify the common library each time, the files will be created in your development data library.

Your common library should contain control files holding data that is used in several environments that is not application data. These control files may be changed during the development process.

See *Appendix A: Common and Production Library Files* for a list of common and production files. Common files will be marked with a "Y" in the Common File field on the SVR screen. Files marked with an "N" in this field are Production files. If the data is to be copied from JDFDATA into either of these files, the SVR Copy Data Y/N field will be marked 'Y'.

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 Computer Operations (G96), choose Data Base Management From Data Base Management (G9645), choose Data Libraries
---	---

On Create Production Environment

```

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 JDFDATA).                _____ JDFDATA
Enter the "TO" Production Library where
you are creating files (e.g. PRODLIB).           _____ DEVDTA
Enter the "TO" Common Library where you
are creating common files (e.g. COMMON)          _____ DEVCOM
If you do not enter a Common library
all common files will be created in the
Production Library.

                                                F5=Printer Overrides
  
```

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

Type the IBM command Create Library (CRTLIB) and press F4.

```

Create Library (CRTLIB)
Type choices, press Enter.
Library . . . . . DEVOBJ      Name
Library type . . . . . *TEST    *PROD, *TEST
Text 'description' . . . . . *BLANK

```

Bottom

F3=Exit F4=Prompt F5=Refresh F10=Additional parameters F12=Cancel
F13=How to use this display F24=More keys

Field	Explanation
Library	Your development object library name.
Library Type	*PROD or *TEST
Text 'description'	The description of your library

Creating a Development Source Library

To create the development source library (DEVSRCLIB), you create a source environment and a source physical file. The source physical file is the Program Source File (JDESRC). All JD Edwards World source members are located in the JDESRC file.

There are two possible methods to create the JDESRC file. You must determine if you have the JD Edwards World Program Generator (CASE) and then choose the appropriate method.

To create a development source library

Type the IBM command Create Library (CRTLIB) and press F4.

```

Create Library (CRTLIB)
Type choices, press Enter.
Library . . . . . DEVSRC      Name
Library type . . . . . *TEST   *PROD, *TEST
Text 'description' . . . . . *BLANK
    
```

```

F3=Exit   F4=Prompt   F5=Refresh   F10=Additional parameters   F12=Cancel
F13=How to use this display   F24=More keys
    
```

Field	Explanation
Library	Your development object library name.
Library Type	*PROD or *TEST
Text 'description'	The description of your library

Creating JDESRC for JD Edwards World Program Generator

When an RPGIII or RPGIV program is moved into production at JD Edwards World, 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 for JD Edwards World Program Generator

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

```

Copy File (CPYF)
Type choices, press Enter.
From file . . . . . F93002      Name
Library . . . . . *LIBL        Name, *LIBL, *CURLIB
To file . . . . . JDESRC        Name, *PRINT
Library . . . . . DEVSRC        Name, *LIBL, *CURLIB
From member . . . . . *FIRST     Name, generic*, *FIRST, *ALL
To member or label . . . . . *FIRST Name, *FIRST, *FROMMMR
Replace or add records . . . . . *NONE *NONE, *ADD, *REPLACE
Create file . . . . . *YES        *NO, *YES
Print format . . . . . *CHAR      *CHAR, *HEX
    
```

```

F3=Exit   F4=Prompt   F5=Refresh   F10=Additional parameters   F12=Cancel
F13=How to use this display   F24=More keys
    
```

Field	Explanation
From file	The file and library containing the data to be copied. The file is F93002 and the library can default to *LIBL.
To file	The name of the source file and your development source library. Generally, the file is JDESRC and the library is DEVSRC.
From member	The member name that will be the beginning of the copy process. Generally, this value is *FIRST.
To member or label	The member name that will be the beginning of the receiving process. Generally, this value is *FIRST.
Replace or add records	Specifies whether the records copied should replace or be added to the records in the <i>To</i> file. In this case since the <i>To</i> file does not exist, this value is *NONE.
Create file	Specifies whether the <i>To</i> file does not exist and needs to be created. This value is *YES.
Print format	Specifies whether the characters are printed in character or character and hexadecimal format. This option only applies if the <i>To</i> file is *PRINT.

2. To remove the empty member copied from JDESRC, type the IBM Remove Member command (RMVM) and press F4.

```

Remove Member (RMVM)
Type choices, press Enter.
Data base file . . . . . JDESRC      Name
Library . . . . . DEVSRC      Name, *LIBL, *CURLIB
Member . . . . . F93002      Name, generic*, *ALL

Bottom
F3=Exit  F4=Prompt  F5=Refresh  F12=Cancel  F13=How to use this display
F24=More keys

```

Field	Explanation
Database file	Type 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.
Member	Type the name of the member that is to be removed. This is F93002.

Creating JDESRC Without the Program Generator

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

To create JDESRC without the Program Generator

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

```

Create Source Physical File (CRTSRCPF)
Type choices, press Enter.
File . . . . . JDESRC      Name
Library . . . . . DEVSRC   Name, *CURLIB
Record length . . . . . 92   Number
Member, if desired . . . . . *NONE   Name, *NONE, *FILE
Text 'description' . . . . . *BLANK

Bottom
F3=Exit  F4=Prompt  F5=Refresh  F10=Additional parameters  F12=Cancel
F13=How to use this display  F24=More keys

```

Field	Explanation
File	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.
Record Length	The number of bytes in the length of the records to be stored in the source file. This value is 92.
Member, if desired	The member to be added to the source file. Generally, this member is left to *NONE.
Text Description	The description of your source file.

About User Profiles

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

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

Defining Access for a User Profile using J98INITA

	From Security Office (G94), choose Library List Control From Library List Control (G944), choose Library List Revisions
---	--

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. On Library List Revisions define the libraries that comprise the list.

```

0094                                Library List Revisions
Action Code. . . . . I
Library List Name. . . . . TEST
Description. . . . . Technical Training Example
Menu Program ID. . . . . P00MENU
Library List . . . . . QTEMP DEVORJ CLTOBJ JDFOBJ DEVDTA DEVCOM DEVSRC CLTSRC
JDFSRC QGPL
    
```

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).

```

0093                                User Signon List Revisions
Action Code. . . . . I
User ID. . . . . FRAZZINI
Seq      Library  Sign-on
Number  List      Menu      Description
-----  -----  -
5.00    PRISTINE  A92      MASTER PRISTINE DATA LIBL
10.00   A52DEV    A92      A5.2 Case Cert & G Development
11.00   PGMGEN    A92      Testing A52 Program Generator
20.00   TECPROG   A92      * List Name Not in Master File
30.00   TECOV     A92      Testing A52 Tech Foundations
55.00   KBGCASE   A92      * List Name Not in Master File
-----  -----  -
-----  -----  -
-----  -----  -
-----  -----  -
-----  -----  -
-----  -----  -
    
```

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

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

```

0092                User Information                Action Code. . . . . I
User ID. . . . . TEACH
Library List . . . . . QTEMP DEVOBJ CLTOBJ JDFOBJ DEVDTA DEVCOM
DEVSRV CLTSRC JDFSRC SECURITY QGPL
-
-
User Security:
  User Key . . . . . A J K DP F
  Initial Menu to Execute. . . . . A
  Initial Program to Execute . . . . .
  Allow Command Entry (Y/N) . . . . .
  Allow Menu Traveling (Y/N) . . . . .
  Allow Fast Path (Y/N) . . . . .
Menu Level . . . . .
User Type . . . . .
User Class/Group . . . . .
Batch Job Queue. . . . . QWATCH
Job Scheduling Priority. . . . . 5 5
Logging(level/severity/messages) . . . . . 4 00 *NOLIST
Output Queue . . . . . QPRINT
Optional Printer File Library. . . . .
Current Library. . . . .
Employee Address Number (PPAT) . . . . .
Set Attention Program. . . . .
F6=Display/Lang Pref  F9=Library Inquiry  F21=Print Lib List  F24=More

```

Note: Each user profile for the JD Edwards World software must have an IBM profile. To define an IBM profile, use the command, Create User Profile (CRTUSRPRF). When a JD Edwards World user profile is created or changed, the IBM JOB is created or changed in the library specified in the Processing Option for J00JOB. The default library is QGPL. If J98INITA is used as the initial program to execute, the JOB is copied into QTEMP and modified.

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
- JD Edwards World Record Types

Copying a Library

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

To copy a library

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

```

Copy Library (CPYLIB)
Type choices, press Enter.
Existing library . . . . . _____ Name
New library . . . . . _____ Name
Create library . . . . . *YES _____ *NO, *YES

```

Caution: If you use CPYLIB, any files that are in use are not copied.

Field	Explanation
Existing Library	The library to be copied in your Production Environment.
New Library	The new library that will be used in your Development Environment
Create Library	Specifies whether the New Library does not exist and needs to be created.

Copying a File

	From Computer Operations (G96), choose Data Base Management From Data Base Management (G9645), choose Data Files
---	---

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

To copy a file

1. On Data File Creation enter the following fields:
 - System Code
 - Create in Library
 - FROM Library
2. Do the following:
 - Enter a 1 next to the files you wish to create from source.
 - Enter a 2 next to the files to be duplicated without data. This is mainly Logical files but could be a Physical file also. Logical files will only be created if the based-on physical file is in the destination library.

- Enter a 3 next to the files to be duplicated with data. This applies only to Physical files.

3. Each selection will be submitted to batch.

```

98101                                Copy Data Files
Enter System Code. . . 01    Address Book
Library Name: From . . JDFDATA    To . . PROD

Sel File Name  File Type  Description
1 F0070        PHYSICAL  Country Constants Master File
1 F009101      PHYSICAL  Word Search Occurrence Master
1 F0101        PHYSICAL  Address Book Master
1 F0101A       PHYSICAL  Address Book Master File Audit Log
1 F0101XX      PHYSICAL  Address Book Master
1 F0101Z1      PHYSICAL  Address Book - Batch File
- F01090       PHYSICAL  Supplemental Data Base - CORE
- F01092       PHYSICAL  Supplemental Data Base - Code
- F01093       PHYSICAL  Supplemental Data Base - Narrative
- F01094       PHYSICAL  User Sequence Preference
- F0111        PHYSICAL  Address Book - Who's Who
- F0114        PHYSICAL  Address Book Memo/Text Information
- F0114W       PHYSICAL  WF - Memo Information Work File
- F0116        PHYSICAL  Address Book Locations
- F01800       PHYSICAL  Address Book Word Search Master

Opt: 1=Copy Data File
    
```

Note: When using this utility, be sure to copy all related files. All the physical files are listed first, followed by the logical files.

Copying a Record

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

To copy a record

1. Type the Copy File command (CPYF) and press F4.

```

Copy File (CPYF)
Type choices, press Enter.
From file . . . . . F0101    Name
Library . . . . . CLTDTA    Name, *LIBL, *CURLIB
To file . . . . . F0101    Name, *PRINT
Library . . . . . DEVDTA    Name, *LIBL, *CURLIB
From member . . . . . *FIRST Name, generic*, *FIRST, *ALL
To member or label . . . . . *FROMMMBR Name, *FIRST, *FROMMMBR
Replace or add records . . . . . *ADD    *NONE, *ADD, *REPLACE
Create file . . . . . *NO    *NO, *YES
Print format . . . . . *CHAR    *CHAR, *HEX

Additional Parameters

Which records to print . . . . . *NONE    *NONE, *EXCLD, *COPIED
Record format of logical file . . . *ONLY    Name, *ONLY, *ALL
Copy from record number . . . . . 365    Number, *START

F3=Exit  F4=Prompt  F5=Refresh  F10=Additional parameters  F12=Cancel  More...
F13=How to use this display  F24=More keys
    
```

2. Press F10 to display additional parameters.

Field	Explanation
From file	The file and library containing the data to be copied.

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

3. Page down 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.
Copy to record number . . . . . 365          Number, *END
Copy from record key:
Number of key fields . . . . . *NONE       Number, *NONE, *BLDKEY
Key value . . . . .
+ for more values

More...
F3=Exit  F4=Prompt  F5=Refresh  F12=Cancel  F13=How to use this display
F24=More keys

```

Field	Explanation
Copy to Record Number	Specifies the record number of the last record to be copied.
Copy from Record Key	Only applies when copying a file with keyed fields.

Copying JD Edwards World Control Records

You can copy any of the following control records:

- Vocabulary Overrides

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

To copy a JD Edwards World record type

	From Computer Assisted Programming (CAP) (G93), choose Developer's Workbench From Developer's Workbench(G9362), choose Copy DD,VO,DW,UDC,SVR,Menus
---	--

On Copy DD,VO,DW,UDC,SVR,Menus.

99630	Copy DD, VO, DW, UDC, SVR, Menus		
From Library	CLTCOM	To Library	DEVCOM
Dictionary Item.	AN8	Language	Appl Ovr. _____ Scrnr/Rpt. _____
Vocabulary Overrides	_____	Language	_____
DREAM Writer Form.	_____	Language	_____
User Def Codes Sys	_____	Language	_____
Type.	_____		
Software Versions Rep. _____			
Menu Identification.	_____	Language	_____
Generic Rate/Msg Sys _____			
Type _____			
F24=More			

Field	Explanation
From Library	The library containing the data to be copied.
To Library	The library in your Development Environment to receive the data.

Field	Explanation
Dictionary Item	<p>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.</p> <p>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 #, @, \$.</p> <p>You can create protected data names by using \$xxx and @xxx, where you define xxx.</p> <p>Messages can contain up to 10 characters. Types of messages are further defined by glossary group.</p>
Vocabulary Overrides	The name of the screen or report record to be copied. All records for soft coding will be copied.
DREAM Writer Form	The name of the DREAM Writer Form ID to be copied. All versions of the specified form will be copied.
User Def Codes Sys	The system code and type of the table to be copied. All values for the specified table will be copied.
Software Versions Rep	The record of the Software Versions Repository member to be copied.
Menu Identification	The menu ID and the display language of the record to be copied.

Note: 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. This section contains the following:

- [Understanding Work Order Processing](#)
- [Creating Work Orders](#)
- [Accessing the Scheduling Workbench](#)
- [Adding Record Types](#)
- [Changing 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 Processing system 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 replanned and thoroughly budgeted, software work orders are often completed without the prior knowledge of the accounting department. Work orders are frequently spontaneous and of short duration.

Note: 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).

Field	Explanation
Parent W.O. No	<p>This is the parent work order number. You can use this number to:</p> <ol style="list-style-type: none"> 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 <p><i>Form-specific information</i></p> <p>For Work Orders</p> <p>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:</p> <ul style="list-style-type: none"> ▪ Description ▪ Extended description ▪ Tax code ▪ Tax rate and area ▪ Date completed
Action Code	<p>A code that indicates the activity you want to perform. Valid codes are:</p> <p>A Add new record</p> <p>C Change existing record</p> <p>D Delete existing record</p> <p>I Inquire on existing record</p> <p>. End of program or function</p> <p>space Clear the form</p> <p>If you enter a code that is not active, the system highlights the code and no action occurs.</p> <p>Note: Depending on how your company has set up action code security, you might not be authorized to use all action codes.</p>
W.O. Number	<p>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.</p>
Description	<p>A brief description of an item, a remark, or an explanation.</p>
Status Comment	<p>A brief description to explain the status of the work order.</p>

Field	Explanation
Charge to BU	<p>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.</p> <p>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.</p> <p>Security for this field can prevent you from locating business units for which you have no authority.</p> <p>Note: The system uses this value for Journal Entries if you do not enter a value in the AAI table.</p>
Search X-Ref	<p>An alphanumeric value used as a cross-reference or secondary reference number. Typically, this is the customer number, supplier number, or job number.</p>
Cost Code	<p>A subdivision of an object account. Subsidiary accounts include more detailed records of the accounting activity for an object account.</p>
Est. Hours	<p>The estimated hours that are budgeted for this work order.</p>
Est. Amount	<p>The estimated dollar amount that is budgeted for this work order.</p>
Start Date	<p>This is a start date that you can enter, or an automatic start date which the planning system calculates using a back scheduling routine. The routine starts with the required date and offsets the total leadtime to calculate the appropriate start date.</p> <p>Will default from system date or you can enter a date.</p>
Planned Comp	<p>The date the work order is planned to be completed.</p>
Phase	<p>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.</p> <p>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.</p>
Completed	<p>The date the work order or engineering change order is completed or canceled.</p>

Field	Explanation
Type	<p>A user defined code (00/TY) that indicates the type classification of a work order or engineering change order.</p> <p>You can use work order type as a selection criteria for work order approvals.</p>
Priority	<p>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.</p> <p>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.</p>
Status	<p>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.</p>
Customer No	<p>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.</p>
Manager	<p>The address book number of a manager or planner.</p> <p>Note: 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.)</p>
Transaction	<p>The date that an order was entered into the system. This date determines which effective level that the system uses for inventory pricing.</p>
Date Assigned	<p>The date the person responsible for the work order receives the work order.</p>
Tax Expl Code	<p>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.</p> <p>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.</p>

Field	Explanation
Tax Rate/Area	<p>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).</p> <p>Typically, U.S. sales and use taxes require multiple tax authorities per tax rate/area, whereas VAT requires only one simple rate.</p> <p>The system uses this code to properly calculate the tax amount.</p>
Subledger Inact	<p>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.</p> <p>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.</p>

Processing Options

Processing options associated with the Single Task Details program 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.

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. <i>Form-specific information</i> 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). <i>Form-specific information</i> 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          Parent W.O. No
Description. . . . APCS Class  W.O.Number . . . 289
Status Comment . Student SAR   Charge to BU . . . 1001
Search X-Ref . . . .          Cost Code. . . .
Est. Hours . . . . 40         Start Date . . . 03.01.94
Est. Amount . . . . 1.500     Planned Comp . . 31.12.94
Phase . . . . . 55           Reserved for Clients
Type . . . . . 2             Priority . . . . . H   Status . . . . . 10
Tax Expl Code. . . .          4802T1 Work Order Search
Subledger Inact. . . .        Order Number 289      Type . . . . *
Customer No. . . . 1001      - 289 WO APCS Class
Manager . . . . . 6001      - 400 WO Rework Electrical
                               - 490 WO BACK, DRAWER, 12x30, DESK
                               - 511 WO AS/400 Chassis Frame Supports
                               - 641 WO AS/400 CRT Chassis Frame
                               - 764 WO Electrical
                               - 772 WO Electrical Phase II
                               - 781 WO Electrical Phase III
                               - 799 WO Other Electrical
                               - 801 WO Electrical
                               Opt: 4=Select      F3=Return      F24=More Keys
Description
SAR setup for work to be
Programming Concepts and
Engine REQ125-796
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

```

48201          Scheduling Workbench          Flr Rem Cat T P M
Action Code. . . . I          Flr Rem Cat T P M
Job or BU. . . . . 1001 Work Orders in Progress
Originator . . . .
Customer Number. . . .
Manager . . . . .
Supervisor . . . .          Parent W.O. No . . . Type . _ Model. _
Search X-Ref . . . .          Cost Code. . . . Prior. _
Category Codes . . Phs. ___ CC2. ___ CC3. ___ CC4. ___ CC5. ___
Sta. ___ Srv. ___ Ski. ___ Exp. ___ CC0. ___

O Number Description X-Ref No. St Status Comment T P
- 1347 Subcontractors 10
- 289 APCS Class 10 Student SAR 2 H

Opt: 1=W.O Entry 4=Return w/# F4=Detail F10=Eq. Workbench F24=More Keys

```

Field	Explanation
Category Codes	Any number or characters that will be used to cross-reference work orders. This value will default from the Single Task Defaults screen.
Job or BU	The business unit that is responsible for charges incurred. Must be a valid business unit setup in the Business Unit Master File (F0006).
Originator	The address number of the person who entered the work order. Must be a valid number in the Address Book Master file (F0101).
Customer	The Address Number of the customer. Must be a valid number in the Address Book Master file (F0101).
Manager	The Address Number of the manager in charge of 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).
Parent W.O. No	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. <i>Form-specific information</i> The parent work order number which groups work orders together in a "family".
Model	Determines whether model work orders will be displayed on the screen.
M	Determines whether model work orders will be displayed on the screen.
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.
Cost Code	The subsidiary account responsible for incurred charges.
Number	The work order identification number. This value defaults from the Single Task Details.

Field	Explanation
Description	Describes the function or option exit. Cannot exceed 40 characters. <i>Form-specific information</i> The name or a brief description of the work order.
X-Ref No	Any number or characters that will be used to cross-reference work orders. This value will default from the Single Task Defaults screen.
Status	A user defined code used to describe the current status of the work order; for example, planned, started, or completed.
Status Comment	This line allows status comments or further description of the work.
Type	User defined code describing the work order type.
Priority P	A user defined code used to assign the priority of the work order; for example, high, medium, or low.

F4 - More Detail

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

```

48201          Scheduling Workbench          Pha Cat Cat T P M
Action Code. . . . I
Job or BU. . . . 1001 Work Orders in Progress
Originator . . . .
Customer Number. . . .
Manager. . . .
Supervisor . . . . Parent W.O. No . . . .
Status . . . . Thru Search X-Ref . . . .
W.O. Date Range. . . . Thru.
Compl. Date Range. . . . Thru.
Supervisor . . . . Parent W.O. No . . . . Type . . . Model. .
Search X-Ref . . . . Cost Code. . . . Prior. .
Category Codes . . Phs. CC2. CC3. CC4. CC5.
Sta. Srv. Ski. Exp. CC0.

O Number Description X-Ref No. St Status Comment T P
- 1347 Subcontractors 10
Planned Comp Hours Scheduled. Est. Hours .
Start Date . 05/26/92 W.O. Flash Message. W.O. Date . . 05/26/93
- 289 APCS Class 10 Student SAR 2 H
Planned Comp 12/31/94 Hours Scheduled. Est. Hours . 40
Start Date . 01/03/94 W.O. Flash Message. W.O. Date . . 11/12/93

Opt: 1=W.O Entry 4=Return w/# F4=Detail F10=Eq. Workbench F24=More Keys

```

Field	Explanation
Planned Comp	The date the work is scheduled to be completed.
Hours Scheduled	The hours of work that has been scheduled.
Est. Hours	Total number of hours estimated for the work order.
Start Date Range	The initial date the work is scheduled to begin.

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.

```

48002                               Detail Spec. Over Titles
Action Code. . . I
Record Type. . . E

Sub-Title 1      Sub-Title 2      Sub-Title 3
Equipment      Production      Production
Number         Time Out         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

- For more information, consult the *Work Orders Guide*.

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:

- [Creating Record Type Codes](#)
- [Defining 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.

Record Type	SUB-TITLE 1	SUB-TITLE 2	SUB-TITLE 3
F	CL_Program	_Program_	Video/Rpt_
G	_Form_ _ID_	_Version_ _No_	_____
H	_Scr/Rpt_ _Name_	_____	_____
I	_____	_____	_____
J	_____	_____	_____
K	_Help_ _Start_	_Help_ _Stop_	_____
M	_Sys_ _Code_	_DTAI_ _Name_	_____
N	_Sys_ _Code_	_Rec_ _Type_	_____
O	_SAR_ _No_	_SAR_ _No_	_SAR_ _No_
Q	_Sys_ _Code_	_Rec_ _Type_	_____
S	_____	_____	_____
U	Reference_ ID/Code_	Attachment Needed-Y/N	_____
W	_Program_ _Name_	_____	_____
Z	_Release_ _ID_	_PTF_ _Number_	_Date_ _Included_
3	System_ Code_	Line_ Number_	Action_ Code_

What You Should Know About

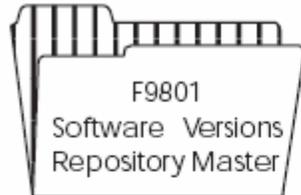
Verifying the record type titles

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.

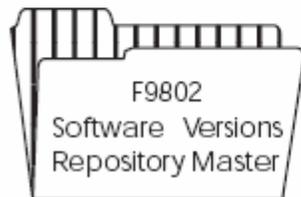
To access Single Task Details, see *Creating SARs*.

Work with Software Versions Repository

The Software Versions Repository (SVR) consists of the following master directories.



A master directory of all files, programs, screens, reports, and copy modules.



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 data is used extensively for documentation and plays an important role in JD Edwards World 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
 - Processing Options
 - User Defined Codes
 - 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
- Edit Helps

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

This section contains the following:

- [Accessing the Software Versions Repository](#)
- [Member Identifiers](#)
- [Naming Conventions](#)
- [The JD Edwards World System Codes](#)
- [Examples of Program and File Names](#)
- [Optional Files Workbench](#)
- [Navigation Functions](#)

Field	Explanation
Description	<p>Identifying information of the member, such as Trial Balance by Business Unit. Associated programs, screens, and reports should share the same description.</p> <p>The description associated with each member is used to further identify the purpose of the member.</p> <ul style="list-style-type: none"> ▪ Physical files should have a description that explains the purpose of the file. ▪ Screens, reports, and CL programs should have the same description as the associated RPG program. ▪ Logical files should be designated as follows: LF - fldname, fldname, fldname: where fldname is a key field. ▪ Join files should be designated as follows: JF - filename/filename/ filename - fldname,fldname,fldname; where the filename is a file over which the join is built and fldname is the key field joining the files. ▪ Work files should be designated as follows: WF - filename; where filename is the file that the work file accesses. ▪ Copy modules carry their own unique descriptions. ▪ File Server programs should be designated as follows: File Server - filename; where filename is the file being served.

Type, Use, and Associated Systems

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

Field	Explanation
Function Code	Designates the object type such as display file, physical and logical files. Use F1 in the field to view the available types. RPG IV programs will use RPGL. RPG IV copy members will use CPYL. UDC 98/F is used to control Member ID and Function Code combinations.
Function Use	Displays how the object is used in the system.
System Code/Product Code	<p>The system code the object is assigned to and delivered under.</p> <p style="text-align: center;"><i>Form-specific information</i></p> <p>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.</p>

Field	Explanation
Reporting System	Designates the system number for the using system. This may differ from the System/Product code. Exceptions occur for data files used by more than one system.

Member Relationship and Compiling Information

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

Field	Explanation
Base Member Name	<p>This field allows for the logical grouping of members.</p> <p><i>Form-specific information</i></p> <p>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.</p> <p>For logical files, this name is the physical file upon which it is based and is required.</p>
Omit Option	<p>Designates items in the Software Versions Repository file that would be bypassed for a new release. These codes are as follows:</p> <p>H Held from all releases X Omit from all releases S Omit Source from all releases O Omit Execution Object from all releases</p>
Generation Sev	<p>Allows the user to designate a severity level when compiling a member.</p> <p>Because some JD Edwards World 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:</p> <pre>CHGCMDDFT CMD(CRTRPGPGM) NEWDFT('GENLVL(20)')</pre> <p>RPG IV programs use CRTBNDRPG so the command to change that default is</p> <pre>CHGCMDDFT CMD(CRTBNDRPG) NEWDFT('GENLVL(20)')</pre> <p>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.</p>

Field	Explanation
Maint/RSTDSP	Designates the type of maintenance on a logical file, how a screen will be processed, or if the program contains embedded SQL statements.

Maintenance on a Logical File

Value	Description
0	No maintenance; or the logical is created dynamically.
1	Logical will be immediately updated when physical is updated.
2	Logical update will be delayed until the next time it is opened. - USE WITH CAUTION!

Processing a Screen

Value	Field Values	Description
1	RSTDSP = *NO (Restore Display) DFRWRT = *YES (Defer Write)	Use with OVERLAY. Do not use with PUTOVR/OVRDTA. All writes to the form field or file formats will be collected and written at one time.
A	RSTDSP = *NO DFRWRT = *NO	Overlay Each write statement is written to the screen.
B	RSTDST = *YES DFRWRT = *NO	Use with PUTOVER to clear and write screen at field level.
S		Used when compiling SQL, RPG, RPGL and PL1 programs. For example, if SQL statements exist within an RPG program, the compiler: <ol style="list-style-type: none"> 1) Executes a create SQL program statement 2) Executes the SQL statements (converts them to calls) 3) Comments them out 4) Executes a CRTSQLRPG program statement and continues as normal

File Information

The following fields identify the file information.

Field	Explanation
Source Library	<p>The library containing the data to be accessed.</p> <p><i>Form-specific information</i></p> <p>The source library where the source file for the object is maintained.</p> <p>This library is usually JDFSRC (for JD Edwards World) or CLTSRC (for the client) for production and DEVSRC for development.</p>
Object Library	<p>The library in your Development Environment to receive the compiled object.</p> <p><i>Form-specific information</i></p> <p>The destination library for the compiled object. This is for compile purposes only and no check is made to ensure that the object is in that library.</p> <p>Leave the object library name blank for copy modules since they are not compiled objects.</p>
Source File	<p>The source file containing the source member.</p> <p><i>Form-specific information</i></p> <p>At JD Edwards World, three source files reside inside of the JDFSRC library.</p> <p>They are</p> <ul style="list-style-type: none">▪ JDECPY for copy modules,▪ JDESRC for all other source code, and▪ F98CRTCMD for precompiler commands.

Field	Explanation
SAR Number	<p>An abbreviation for software action request (SAR).</p> <ul style="list-style-type: none"> ▪ If the CASE profile has *NONE for SAR Number, the SAR number will not be validated in any of the CAD/CAP programs and can be left blank. ▪ If a SAR number is entered, it will be validated against the Work Order file, F4801. <p style="text-align: center;"><i>Form-specific information</i></p> <p>The JD Edwards World Development process uses 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.</p> <p>If an Employee Address Number is specified on the User Information screen (F0092 file), that number will be used to list SARs currently assigned to that person when F1 is pressed on this field.</p> <p>The edit for this field is controlled by the SAR information entered in CASE Profiles.</p>
Version ID	<p>The software version number to be defaulted in the Software Versions Repository file.</p> <p style="text-align: center;"><i>Form-specific information</i></p> <p>Identifies the JD Edwards World release level of the member in the designated environment.</p> <p>Validated against User Defined Codes 98/RL.</p>
S C (Status Code)	Determines the status of the software in the development process.
Development	<p>Development Progress Code.</p> <p>Indicates the progress of modifications made to the member.</p>
User ID	<p>The IBM-defined user profile.</p> <p style="text-align: center;"><i>Form-specific information</i></p> <p>User ID that last modified the member (automatically updated).</p>
Date Modified	The date the member was last updated (automatically updated).

Note: Each subfile line represents a record in the Software Versions Repository detail file (F9802). The information can be entered manually or through the upgrade process. There is no presence check for the object so if the object was deleted or the information was entered in error, there may be invalid entries.

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

- 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.

O	Source	Object	Source	SAR	Version	S	D	User	Date
P	Library	Library	File	Number	ID	C	P	ID	Modified
—	JDFSRC	JDFOBJ	JDESRC	685935	A73	1	—	JDE	11/12/93

```

9801                               Software Versions Repository
Action Code. . . . I
Member ID. . . . R42565
Description. . . . Sales Order Invoices Print
Function Code. . . PRTF Printer Files
Function Use . . . 164 Special Forms
System Code. . . . 42 Sales Order Processing
Reporting System 42 Sales Order Processing
Base Member Name P42565 File Prefix. . .
Maint/RSTDSP . . . Omit Option. . . Generation Sev .
Copy Data (Y/N). N Optional File. . N Common File. . . N

O Source      Object      Source      SAR      Version    S D      User      Date
P Library    Library    File        Number   ID         C P      ID        Modified
-- JDFSRC    JDFOBJ    JDESRC     672721  A73       1 -      JDE       11/08/93

```

```

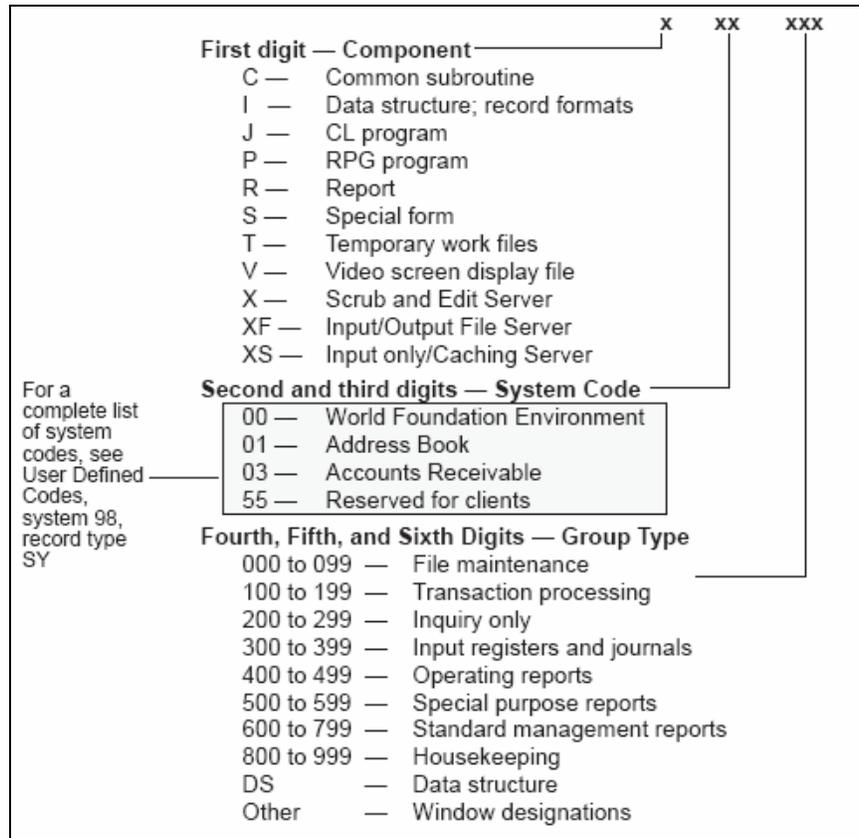
9801                               Software Versions Repository
Action Code. . . . I
Member ID. . . . J42565
Description. . . . Sales Order Invoices Print
Function Code. . . CLP CL Programs
Function Use . . . 164 Special Forms
System Code. . . . 42 Sales Order Processing
Reporting System 42 Sales Order Processing
Base Member Name P42550 File Prefix. . .
Maint/RSTDSP . . . Omit Option. . . Generation Sev .
Copy Data (Y/N). N Optional File. . N Common File. . . N

O Source      Object      Source      SAR      Version    S D      User      Date
P Library    Library    File        Number   ID         C P      ID        Modified
-- JDFSRC    JDFOBJ    JDESRC     644471  A73       1 -      JDE       08/09/93

```

The JD Edwards World naming structure identifies and describes major software components. 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.

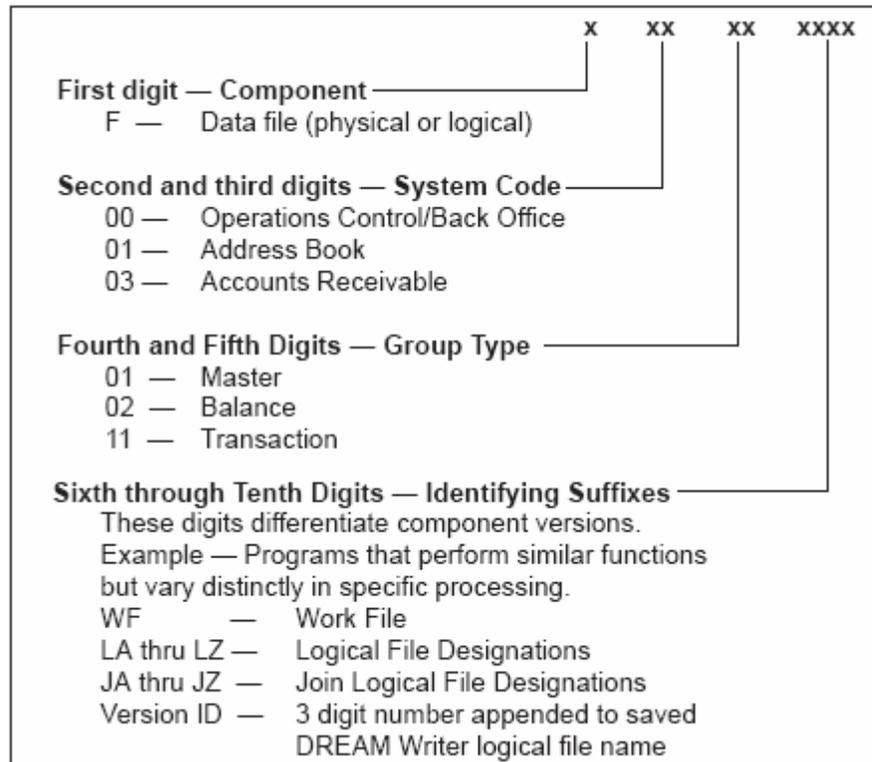
Use the following chart as your guide when naming objects.



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.



The following shows the names for different types of programs and files.

Item	Description
Maintenance program	The maintenance program for a file has the same name with a different prefix. For example, F9220 and P9220 or F9601 and 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 CRTDUPOBJ. The job then removes the object after completion. <ul style="list-style-type: none"> ▪ Usually Physical Files ▪ Begin with T ▪ Found in JDFOBJ

Item	Description
Dynamic work files	Dynamic work files are usually FASTR processing requirements. Dynamic work files create and delete after the job is complete. <ul style="list-style-type: none">▪ Usually logical files▪ Have same name as program

The JD Edwards World 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:

General System Codes

System Number	System
00	General Back Office
01	Address Book
02	Electronic Mail
03	Accounts Receivable
03B	Enhanced Accounts Receivable
03C	Customer Issue Tracking
04	Accounts Payable
05	Standalone Time Accounting
06	Old Payroll
07	New Payroll
08	Human Resources
08A	Application Tracking
08B	Benefits Administration
08C	Canadian Specific HR Functions
08H	Health and Safety
08P	Position Control
08R	Requisitions
08U	US Specific HR Functions

System Number	System
08W	Wage and Salary
09	General Accounting
09E	Expense Reimbursement Reporting System Only
10	Financial Reporting
10C	Multi-Site Consolidations
11	Foreign Currency/Cash Basis
11C	Cash Basis Accounting
12	Fixed Assets
13	Equipment/Plant Management
14	Modeling, Planning, and Budgeting
15	Commercial Property Management
16	Residential Property Management
17	Customer Service Management
17A	Ariba Integration Reporting Purposes Only
17C	Call Management Reporting Purposes Only
18	Resource Scheduling Reporting Purposes Only
19	Utility CIS
20	Energy Data Base
21	Lease Management
22	Production System
23	Revenue Distribution
24	Contracts
25	Joint Interest Billing
26	Gas Balancing
27	Investor Services
28	Projects on Hold – Energy
29	AFE Accounting
30	Product Data Management
30A	Product Costing Reporting Only

System Number	System
31	Shop Floor Control
31A	Manufacturing Accounting
3110	Process Control
32	Configuration Processing
32C	Custom Works
33	Capacity Requirements Planning
34	DRP/MPS/MRP
34A	Advanced Planning & Scheduling
35	Enterprise Facility Planning
36	Forecasting
37	Quality Management
38	Distribution Contracts
39	Bulk Stock Control
40	Inventory/OP Base
4010	Advanced Price Adjustments
41	Inventory Management
41B	Bulk Stock Control
42	Sales Order Processing
42A	Sales Force Automation Reporting Purposes Only
42B	Enterprise One Sales Order Entry
42E	ECS Sales Order Processing
43	Purchase Order Processing
44	Contract Administration
44H	Homebuilder Management
4401	Homebuilder Management
45	Advanced Price Analysis
46	Warehouse Management
47	Electronic Data Interchange
48	Work Order Processing

System Number	System
48S	Service Billing
49	Transportation
50	Job Cost Accounting
51	Job Cost Accounting
52	Job Cost Billing
53	Change Management
55-59	Reserved for Clients
60-69	Reserved for JD Edwards World Custom
70	Multi-National Products
71	Client Server Applications
73	M&D Complimentary Products
74	EMEA Localizations
74G	Greece
74H	Hungary
74I	Ireland
74L	Portugal
74N	Nordics
74P	Poland
74R	CIBS
74S	Spain
74T	Turkey
74Z	Czech Republic
75	Asia Pacific Localizations
75H	Thailand
75I	India
75K	South Korea
7T	Taiwan
76	Latin American Localization
76A	Argentinean Localization

System Number	System
76C	Columbia
76H	Chile
77	Canadian Payroll
77Y	Canadian Payroll Year End Programs Reporting System Only
78	OBSOLETE CS Travel Expense Management
79	OBSOLETE Foreign Translation

Technical Foundation Systems

System Number	System
00	General Back Office
80	Business Intelligence
81	DREAM Writer
82	World Writer
83	Finance Report Writer-FASTR
84	Distributed Data Processing
85	Custom Programming
86	Foreign Language Translation
87	JD Edwards World Internal
88	Cautious Purge System
89	Conversion Programs
91	Documentation
92	Computer Assisted Design
93	Computer Assisted Programming
94	Security Officer
95	Unattended Night Operations
96	Computer Operations
97	Software Load & Install
98	Technical Aids


```

9801                               Software Versions Repository
Action Code. . . . I
Member ID. . . . F0911IA
Description. . . . LF - Doc Type, Doc, Key Co, G/L Date(##YYMMDD), Line #, Ext
Function Code. . . LF Logical Files
Function Use . . . 230 Transaction Files
System Code. . . . 00 Technical Foundation
Reporting System 09 General Accounting
Base Member Name F0911 File Prefix. . . GL
Maint/RSTDSP . . 1 Omit Option. . . _ Generation Sev .
Copy Data (Y/N). N Optional File. . N Common File. . . N

O Source Object Source SAR Version S D User Date
P Library Library File Number ID C P ID Modified
-- JDFSRC JDFDTA JDESRC 591710 A61 1 _ JDE 03/22/93

```

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.
- If the Physical files exist in separate libraries, the CRTLF command must be used since the SVR process can only create a logical if the joined files are 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 .
Copy Data (Y/N). N Optional File. . N Common File. . . N

O Source Object Source SAR Version S D User Date
P Library Library File Number ID C P ID Modified
-- JDFSRC JDFDTA JDESRC 493167 A61 1 _ JDE 03/05/93

```

Copy Modules

- The Member ID begins with C, D, E, or I.
- The Source File is JDECPY.
- The Description describes the function of the module.
- The Function Code is COPY or CPYL (RPG IV code).

```

9801                Software Versions Repository                Release. . A61
Action Code. . . . I
Member ID. . . . . C00SC
Description. . . . Copy Module - Retrieve Soft Coding
Function Code. . . COPY   RPG Copy Module
Function Use . . . 194   Common Subroutine
System Code. . . . 98   Technical Tools
Reporting System 98   Technical Tools
Base Member Name C00SC
Maint/RSTDSP . . . Omit Option. . . . Generation Sev .
Copy Data (Y/N). N   Optional File. . N   Common File. . . N

O Source   Object   Source   SAR   Version   S D   User   Date
P Library  Library  File     Number  ID       C P   ID     Modified
-- JDFSRC   JDFOBJ   JDECPY   603784 A61     1 -   JDE   06/10/93
    
```

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.

```

9801                Software Versions Repository
Action Code. . . . I
Member ID. . . . . V09ACCT
Description. . . . Account Master Additions Window
Function Code. . . DSPF   Video Display Files
Function Use . . . 111   File Maintenance
System Code. . . . 09   General Accounting
Reporting System 09   General Accounting
Base Member Name V09ACCT
Maint/RSTDSP . . . Omit Option. . . . Generation Sev .
Copy Data (Y/N). N   Optional File. . N   Common File. . . N

O Source   Object   Source   SAR   Version   S D   User   Date
P Library  Library  File     Number  ID       C P   ID     Modified
-- JDFSRC   JDFOBJ   JDESRC   552868 A61     1 -   JDE   12/08/92
    
```

Navigation Functions

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

Function	Description
F6	Access Repository Services You can access the Repository Services form using F6. This form provides access to the other repository services within JD Edwards World.

Sample F6 - Repository Services

```

9801                Software Versions Repository
Action Code. . . . I
Member ID. . . . . P01051
Description. . . . Address Book Information
Function Code. . . RPG   RPG Programs
Function Use . . . 111   File Maintenance
System Code. . . . 01   Address
Reporting System 01   Address
Base Member Name P01051
Maint/RSTDSP . . .   Omit Opt
Copy Data (Y/N). N   Optional
                  DREAM Writ
O Source   Object   Sourc
P Library  Library  File
  JDFSRC61 JDFOBJ61 JDESR
  JDXSRC61 JDJOB61 JDESR
  PGXSRC61 PGJOB61 JDESR
                  Opt: 1=Select  F12=Previous
Opt: 1=Browse 2=Edit 3=Copy 5=SAR 8=Print 9=Dlt 10=Design 14=Crt  F24=More

```

Use selection 1 to select the available services.

Function	Description
F9	Automatic Reinquiry 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 When you inquire on a member, the system positions the cursor in the subfile for the screen. To reposition your cursor in the <i>Action Code</i> field, press F17.
F19	Previous Member When you inquire on a member, the system positions the cursor in the subfile for the screen. To reposition your cursor in the <i>Action Code</i> field, press F17.
F20	Next Member To access the member stored after the currently displayed member, press F20.

Other Function Keys

Function	Description
F2	<p>Automatic Reinquiry</p> <ul style="list-style-type: none"> To access a command line to enter a JD Edwards World or IBM command without having to exit to Command Entry or a menu. Calls a JD Edwards World program and not the IBM Command Entry. <p>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.</p>
F8	<p>Optional Files</p> <p>The system displays the optional files.</p>
F10	<p>Checklists</p> <p>Displays a user defined checklist. Opt 1 displays additional job information.</p>
F13	<p>Displays additional category code information for each member. You can cross-reference category code values to the Software Versions Search program (23/G91).</p>
F14	<p>Member Parameter/Key List</p> <p>Identifies the access path for keyed files.</p>
F15	<p>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.</p>

Below is an example form displaying every program that uses the Business Unit Master screen:

Use an * in Type on the Cross Reference screen to see the possible combinations.

```

980014                               Cross Reference
Object: Name . . . V0006             Business Unit Master Revisions - Single
      Type . . . E                   All programs using file
      To Display P
      Funct Cd . . .
O  Name          Description          Field Attr T Start Upd
P  P0006         Business Unit Master Revisions - Single  Len Dec Y Loc Y/N
-                                     Y

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

```

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.

This is 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.

Option	Description
1	Browse Displays the source member in SEU browse mode.
2	Edit Displays the source member in SEU update mode.
3	Copy the source member Copies the source member to another source file member or copies the member to a new member name in the same file. Adds SVR master and detail records if needed. 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 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.

Option	Description
10	<p>Exit to design aid</p> <p>Determines what type of member you are accessing and then exits to the correct JD Edwards World design tool; that is, SDA/RDA/FDA/Program Generator.</p>
11	<p>Precompiler commands for JD Edwards World compiler.</p> <p>Accesses the source code for the precompiler commands associated with a program.</p> <p>A highlighted message, <i>Precompiler Commands Exist</i>, indicates there is a member with the same name in the F98CRTCMD file.</p> <p>Contains information the compiler will use when compiling the program.</p> <p>Note: Only one person can view the same pre-compiler commands.</p>
14	<p>Submit object creation</p> <p>Compiles the member and generates an object. The type of object determines what library the compiled object will be placed into.</p>
15	<p>Generate program source and help</p> <p>Submits the member to the program generator in order to generate source and related helps.</p> <p>Only applicable to CASE users.</p>
17	<p>Edit help instructions</p> <p>Accesses the help instructions for a particular program in update mode utilizing SEU.</p>
18	<p>Generate & rebuild help instructions</p> <p>Submits the helps for generation and rebuilds them into their final form once they have been entered.</p>
20	<p>Browse SDA/RDA</p> <p>Accesses SDA or RDA in browse mode with Vocabulary Overrides.</p>
21	<p>Print help</p> <p>Prints the help instructions for the member.</p>
25	<p>Print illustration</p> <p>Prints an illustration of printer files, display files, or data base files.</p>

Option	Description
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.

CASE Profiles

About CASE Profiles

CASE profiles are user defined values that can pertain to individual users or to one *PUBLIC user profile. The profiles are stored in F98009. Parameters in these profiles are used when copying source members or when designing or compiling programs and files.

The following processing control parameters are defined by the user:

- Default development libraries
- Compile job queue
- Program Generator source generation job queue
- Compile print options
- SAR logging options

Note: You should immediately update the record for User ID *PUBLIC to reflect JOBQs, OUTQa and Libraries on your machine.

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.

This section contains the following:

- [Accessing CASE Profiles](#)
- [Function Key Exits from the CASE Profiles Program](#)
- [Summary of CASE Profiles](#)

Accessing CASE Profiles

To access CASE Profiles

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

1. From Computer Assisted Design (CAD) (G92), choose CASE Profiles.

Default Development Environment

Field	Explanation
Source File Name	The default source file name where source is to be stored within the source library. Must reside within the source library specified.
Source Library	The default library where source will be stored. The source file specified must reside within this library.
Object Library	The default library where compiled objects will be stored.
CL Source File	The default library where source for CL programs will be stored. The value specified must reside within the source library specified.
Data File Library	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.
SAR Number	An abbreviation for software action request (SAR). <ul style="list-style-type: none"> ▪ *NONE = the SAR number will not be validated in any of the CAD/CAP programs and can be left blank. ▪ If a SAR number is entered, it is used in conjunction with the SAR Delivery Type of *DFT (default).
Version ID	The software version number to be defaulted in the Software Versions Repository file.
Status Code	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.

Caution: 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 JD Edwards World 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.

Field	Explanation
Compile Job Queue	Specifies which job queue will be used when submitting programs to compile. This job queue is used for programs with function code of RPG, RPGL, CBL, PLI, C and SYSC.
Prog Gen Job Queue	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.
Compile Target Release	Used by various AS/400 compilers (RPG, RPGL, CLP, COBOL, C) to compile an object compatible with a specified target release. <ul style="list-style-type: none"> ▪ A value of *CURRENT compiles an object compatible to the release of the machine at compile time. ▪ A value of *PRV compiles an object compatible with both one release back and the current release.
Print Option	Used to designate whether or not a report will be generated when an object is compiled. <ul style="list-style-type: none"> 0 no print 1 print 2 print and hold spool file 3 print only, does not generate execution object (applies to COBOL and RPG only) 4 print when compile or creation fails
Cross-Reference Listing	Specifies whether a cross-reference listing will be generated for variables and fields in a program's compile listing.

Note: The default job queue for compiles is called COMPILE. If you do not have a COMPILE JOBQ, enter the JOBQ name for compiles on the *PUBLIC CASE profile.

SAR Options

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

Field	Explanation
SAR File Library	Specifies the library where the Software Action Request (SAR) file for software development exists. <ul style="list-style-type: none"> ▪ If left blank, the user's library list will be used. ▪ You can specify *NONE in the SAR number field if you do not want any SAR number editing.

Field	Explanation
SAR Delivery Type	<p>Associated with SAR logging. SAR logging is a feature which tracks all activities related to modifying JD Edwards World' software.</p> <ul style="list-style-type: none"> ▪ *NONE = no logging. ▪ *LOG = log to SAR number 00000000 (no SAR number is used for logging). ▪ *DFT = log to a default SAR number (specified in the SAR Number field). ▪ *PROMPT = log and prompt the user for the SAR number to be used and allow the user to enter the revision notes.

Miscellaneous

The following fields are reserved for future use.

Field	Explanation
Source Gen Opt (Future)	For future use.
Helps Maint Opt(Future)	For future use.

Function Key Exits from the CASE Profiles Program

Function Key	Description
F6	<p>Access Repository Services</p> <p>This key will display a form that provides access to the other repository services including CASE profiles.</p>
F9	<p>Previous Member</p> <p>Allows you to re-inquire on the last record viewed.</p>

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.

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:

- [Setting Up User Input Options for SAR Logging](#)
- [Selecting Types of SAR Information to Log](#)
- [Accessing SAR Log Inquiry](#)
- [Summary of the 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      Program Creation Options
Source File . . . . . JDESRC         Compile Job Queue . . . . . COMPILER
Source Library. . . . . PGFSRC61     Prog Gen Job Queue. . . . . CLONE
Object Library. . . . . PGFOBJ61     Compile Target Release. *CURRENT
CL Source File. . . . . JDECLSRC     Print Option . . . . . 1
Data File Library . . . . . PGPDTA61  Cross-Reference Listing N
SAR Number. . . . . 774487
Version ID. . . . . A61              A6.1 Base
Status Code . . . . . 2              Development

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

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

F24=More Keys
    
```

1. Complete the following fields:

- SAR File Library
- SAR Delivery Type

Caution: The SAR file library contains the Work Order system files (F4801 and F4802). If you use the Work Order application, or 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.

Note: If you set the SAR Delivery Type field to *PROMPT, the Maintain User Default SAR Information form displays whenever you change a source code member or control table.

```

9812  Maintain User Default SAR Info
Action. . . . . C
Repository Rec. P9242
SAR Number. . . . .          Transfer. . . . . 0
Revision Note . . . . .
Enter=Continue      F24=More
    
```

If you provided a SAR number on CASE Profiles, it displays 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, a record will be created for the Version Control system. If it is set to 0, there will not be a record for the Version Control System.

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 displays 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.

1. From the Version Control menu (G9261), access the processing options for Edit and Promote.
2. Make the following changes:

Item	Description
SAR Logging (1)	<p>Specify Y if you want to track SARs that are associated with JD Edwards World 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.</p> <p>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.</p> <p>In addition, the SAR logging program runs a double-byte analysis against your RPG programs if you set this processing option to Y.</p> <p>If you specify Y, you also must indicate the name of the library that contains your SAR files. The default library name is JDCOMDATA.</p>
DREAM Writer Copy (2)	<p>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.</p>

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 (F6 in SVR)

Field	Explanation
AC (Action)	The action that was taken on this record. The standard action code values apply.
Ty (Record Type)	The type of record that was updated. Use F1 to display all valid record types stored in User Defined Code 98/RT.
Item	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, and the like.
SAR Number	The SAR number under which this change was made. This field can be updated on this video.
Revision Note	A user defined description field to further clarify the change made. This field can be updated on this video.
Time	The time at which the change was made.
Date	The date on which the change was made.
User	The user who made the change.

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.

Exit	Explanation
2 - Edit	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.
5 - Work SAR	Exits to the SAR associated with the SAR log entry.
9 - Delete	Allows the user to delete entries from the SAR log.

Note: 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

Function Key	Description
F5	<p>ASI Entry</p> <p>Exits to Application Specific Instructions form for use during a software upgrade. You need the F0098 file to do this.</p>
F6	<p>Access Repository Services</p> <p>Pressing this key displays a form that provides access to the other repository services, except for SAR Log Inquiry.</p>
F21	<p>Print</p> <p>Allows you to print a SAR log report.</p> <p>Exits to a DREAM Writer versions list.</p>

Summary of the SAR Log Inquiry

The SAR Log Inquiry has the following features and restrictions:

- Uses the file F9810.
- If you do 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.

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

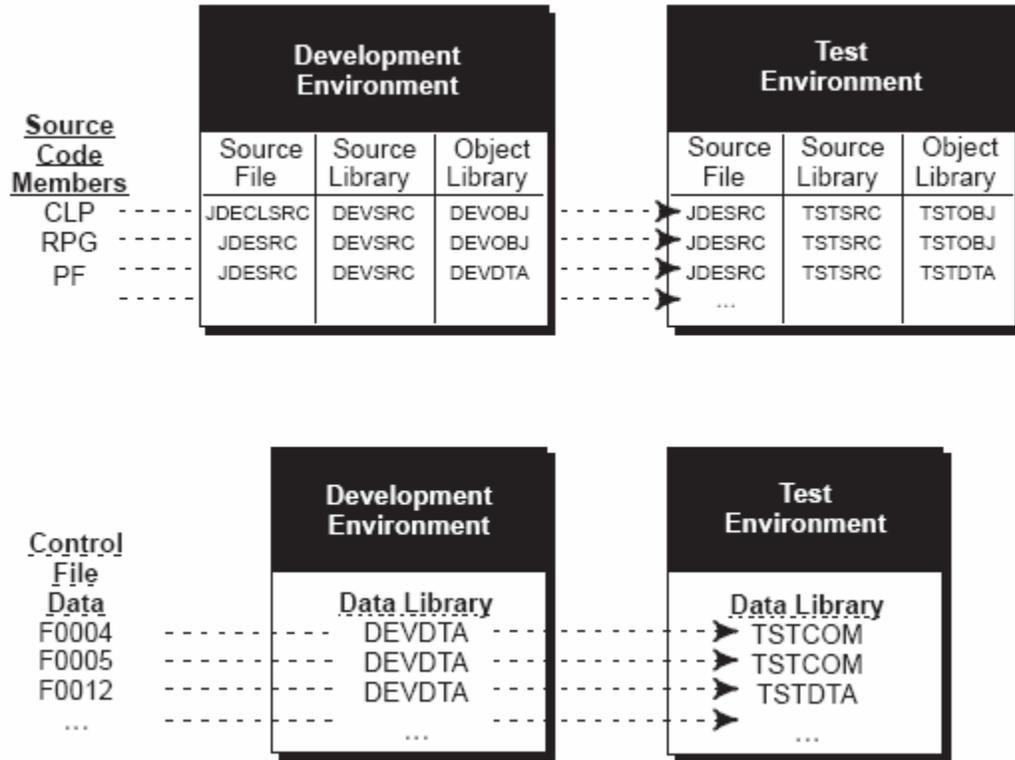
For control file data, the environment consists of a data library.

Perform the following tasks:

- [Understanding Promotion Paths](#)
- [Defining a Promotion Path](#)
- [Defining a Project](#)

Understanding Promotion Paths

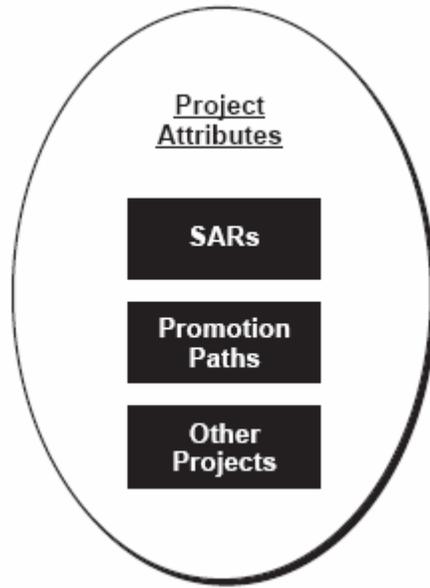
A promotion path specifies the current locations of source code members and control file data and where they will be moved. For example, promoting a project's source code members and control file data from a development environment to a test environment could look similar to the following illustrations.



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.

```

92403                                Manage Promotion Paths      Code 1 . . . . . ___
Promotion Path . JDF73                Code 2 . . . . . ___
                                        Code 3 . . . . . ___
                                        Code 4 . . . . . ___
                                        Code 5 . . . . . ___
O Promotion      Description          Release
P Path           Description          Number
- JDF73          Transfer to JDF73          A73
- JDF73T         'T' file transfer to JDF73 A73
- JDF73TEC       Transfer to JDF73 SECURE  A73
- JDU71          Transfer to JDU71         A71X
- JDX71          Transfer to JDX71         A71X
T130892PC2      Utility CIS - PCCPY       A71X
T130892PC3      Utility CIS - PCCPY       A71X
T130892PC4      Utility CIS - PCCPY       A71X
T130892PC5      Utility CIS - PCCPY       A71X
T130892PC6      Utility CIS - PCCPY       A71X
T130892PC7      Utility CIS - PCCPY       A71X
T130892PC8      Utility CIS - PCCPY       A71X
T130892PC9      Utility CIS - PCCPY       A71X
UQF62           UQF build for A6.2        A62
UQF71           UQF build                  A71
VCT             Version control training   A71
Opt: 1=Change  2=Members  3=Ctl Files  F5=Add Path  F24=More Keys
    
```

```

92403                                Manage Promotion Paths      Code 1 . . . . . ___
Promotion Path . JDF73                Code 2 . . . . . ___
                                        Code 3 . . . . . ___
                                        Code 4 . . . . . ___
                                        Code 5 . . . . . ___
O Promotion      Description          Release
P Path           Description          Number
- JDF73          Transfer to JDF73          A73
Code 1 . ___ Code 2 . ___ Code 3 . ___ Code 4 . CUR Code 5 . JDF
    
```

To locate a promotion path

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

1. On a blank Manage Promotion Paths form, press Enter.
The screen displays a complete list of promotion paths.
2. 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

1. On the Manage Promotion Paths form, press F5 (Add Path).

```

9240                                Promotion Path
Action Code . . . I
Promotion Path . _____
Description . . . _____
Release . . . . . _____
Code 1 . . . . . ___
Code 2 . . . . . ___
Code 3 . . . . . ___
Code 4 . . . . . ___
Code 5 . . . . . ___
F9=Redisplay  F10=Members  F11=Ctl Files  F24=More Keys
    
```


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

Key	Description
F9	Redisplay the record for the previously-changed path.
F11	Displays the Promotion Path Control Files form.
F13	Displays the CASE Profiles form.
F14	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.
F15	Duplicates the source file and library names from the first member type to the remaining member types.

- F14 Retrieves the data library from your CASE profile and fills in the first From Data Libr field. This overwrites any information currently in the field.
- F15 Duplicates the library names from the first record type to the remaining record types.

3. To copy library names from one record type to another, type 1 (Copy) in the OP (Option) field next to the record type you want to copy.
4. Type 2 (Target) in the OP fields next to the record types you want the information copied to, and press Enter. You can specify multiple targets.

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

To define 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.

```

92413                               Manage Projects
Project . . . . . _____          Code 1 . . . . . _____
Client . . . . . _____          Code 2 . . . . . TEC
Originator . . . . _____         Code 3 . . . . . _____
Assigned To. . . . _____         Code 4 . . . . . _____
                                     Code 5 . . . . . _____
O
P Project      Description          Client  Orig  Assigned
- REINSTALL    Simplified Reinstall Process
- TEC          Tech Foundation Corrections
- TECHENH      Tech Foundation Enhancements
- UBP          User Based Pricing
- UPGRADE      Upgrade Enhancements/fixes    256006  878411  878411
- VC           Version Control                875561
- 1055020     Merge PO Display Level
- 4 BYTE SYS  4 Byte System Code              878411  2211696
-
-
Opt: 1=Details 2=Paths 3=SARs  F5=Add Project  F24=More Keys
    
```

```

92413                               Manage Projects
Project . . . . . _____ Code 1 . . . . . ____
Client . . . . . _____ Code 2 . . . . . TEC
Originator . . . . . _____ Code 3 . . . . . ____
Assigned To. . . . . _____ Code 4 . . . . . ____
                                   Code 5 . . . . . ____

O
P Project _____ Description _____ Client _____ Orig _____ Assigned _____
- REINSTALL Simplified Reinstall Process
  Code 1 . 300 Code 2 . TEC Code 3 . ____ Code 4 . ____ Code 5 . ____
    
```

To locate a project

Locate a project using one of the following methods:

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

To add a project

1. On Manage Projects, choose Add Project.

```

9241                               Software Development Project
Action Code. . . I
Project. . . . . _____
Description. . . . . _____
Parent Project . . . . . _____
Edit File. . . . . -
Client . . . . . _____ Requested. . . . . _____
Originator . . . . . _____ Planned Comp . . . . . _____
Assigned To. . . . . _____ Date Assigned. . . . . _____
Auth List. . . . . _____
Design Doc . . . . . _____

Code 1 . . . . . ____
Code 2 . . . . . ____
Code 3 . . . . . ____
Code 4 . . . . . ____
Code 5 . . . . . ____

F9=Redisplay  F10=Promotion Paths  F11=Project SARs  F24=More Keys
    
```

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.

Key	Description
F9	Redisplays the record for the previously-changed project.
F10	Displays the Project Promotion Path form.
F11	Displays the Project Elements form.
F14	Displays the generic text associated with this project, and gives you access to text model selections.

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 selection (Option 4 on menu G9261), 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 (Manage projects, F5), press F10 (Promotion Paths).

```

92411                               Project Promotion Paths
Action Code. . . . I
Project. . . . . TEC Tech Foundation Corrections

O Promotion                               Release
P Path Description Number
- A62CUM Transfer to current A62 Cum A62
- A62PC000TI User based pricing A62
- A62PREV Transfer to previous A62 Cum A62
- A72CUM Transfer to current A71 Cum A71
- A71PREV Transfer to previous A71 Cum A71
- A72CUM Transfer to current A72 Cum A72
- JDF62 Transfer to JDF62 A62
- JDF62TEC Transfer to JDF62-SECURE A62
- JDF71 Transfer to JDF71 A71
- JDF71TEC Transfer to JDF71-SECURE A71
- JDF73 Transfer to JDF73 A73
- JDF73TEC Transfer to JDF73 SECURE A73
- VCTL Version control Update A73
-
-
-
Opt: 1=Details 2=Members 3=Ctrl Files F11=Project SARs F24=More Keys
    
```

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.

Some of the function keys available on this form include:

- F9 - Redisplays the record for the previously-changed project.
- F11 - Displays the Project Elements form.

Options available on this form include:

- 1 - Edit the promotion path details.

- 2 - Edit the promotion path members.
- 3 - Edit the promotion path control files.

To assign project SARs

SARs are one element of a project; 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 menu (G9261), 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).

Project Elements		St	Status
0	Project	T	
P	Element	Y	
-	00718047	S	% Menu Job Stream Int/Bth Mix 01 Complete - in next release.
-	00731073	S	JDEDBG - V2R3 Misc. Fixes 26 Test in Demo
-	00735672	S	Can't Chg Value--DD Item Array 01 Complete - in next release.
-	00736245	S	User Defined Code Security 26 Test in Demo
-	00863261	S	User Based Pricing 23 Manager Review
-	00907489	S	ASI Rpt and Screen 4 Digits 26 Test in Demo
-	00910451	S	DDP RJE Code Correction CL 7.1 01 Complete - in next release.
-	00913176	S	Version Control - Build PTF 23 Manager Review
-	00915577	S	Quick Start - Vocab Overrides 01 Complete - in next release.
-	00917732	S	Auto Build of JDE Msg File/JLF 25 Rework
-	00939827	S	Menu Integrity rpt/sys 55-59 01 Complete - in next release.
-	00945565	S	ASI Rpt Confusing When No ASIs 26 Test in Demo
-	00953602	S	F6 Copy from ASI -Prt Override 01 Complete - in next release.
-	00955229	S	Next Number Description-00 Sys 01 Complete - in next release.
-	00958278	S	Validation rpt-prt file names 01 Complete - in next release.
-	00966438	S	Action Code Security 01 Complete - in next release.
Opt: 1=SAR 2=SAR Log 3=Edit 4=Promote F10=Project Paths F24=More			

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). Some of the function keys available on this screen include:

Key	Description
F9	Redisplays the record for the previously-changed project.
F10	Displays the Project Promotion Paths screen.

Options available on this screen include:

- 1 - Displays or edits the SAR detail.
- 2 - Displays or edits the SAR log. The SAR Log Transfer screen displays, 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.

- 3 - Displays the Pre-Promotion Edit History form. For information about this function, see Promote a SAR in this publication.
- 4 - Promotes a project. For information about this function, see Promote a SAR in this publication.
- 5 - Displays the promotion history of a SAR ('Z' record).
- 6 - Displays or edits notes associated with a SAR (for JD Edwards World environments only; '*' record).

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, or if the F4802 file has different record types than is needed by version control, define a separate library that contains these files for version control purposes only.

Complete the following tasks:

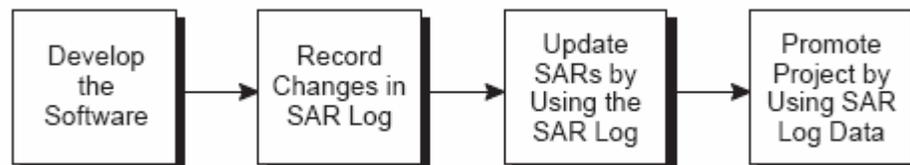
- [Update the SARs](#)
- [Validating a Promotion Path](#)
- [Promoting a Project](#)

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.



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.

```

92412                                Edit and Promote
Action Code. . . I                    SAR #. . . 1079777
Project. . . . TEC                    Project. . . .
Promotion Path . JDF73                Transfer to JDF73   Status . . . S/P _
O Project T
P Element Y Description St Status
- 01079777 Multiple Jobs Submitted 23 Manager Review
- 01081666 Release Specific Transfers 23 Manager Review
- 01083573 Help Window Mods A73 26 Test in Demo
- 01086299 Unable to use A Action Code 06 Returned - Already reported
- 01087558 Data Selection - HMCU 23 Manager Review
- 01088104 V3R1 CRTCPGM/CRTBND 28 A test complete
- 01088163 Handle special char for DBCS 01 Complete - in next release.
- 01089414 Localization Issues in A/B 01 Complete - in next release.
- 01093536 No previous item displayed 23 Manager Review
- 01099807 Finalize Version Control 23 Manager Review
- 01101364 Video Illustration 23 Manager Review
- 01102615 J97UPGRADE Command Validation 23 Manager Review
- 01104004 DW Merge Database-No Merge Opt 26 Test in Demo
- 01105226 WW-Reads all versions at once 28 A test complete
- 01107601 Don't delete SAR Log if Trf er 23 Manager Review
- 01113921 Variable Length Field Support 26 Test in Demo
Opt: 1=SAR 2=SAR Log 3=Edit 4=Promote 5='Z' F10=Proj Paths F24=More

```

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 displays, which lists all added or changed records logged in the SAR log (F9810) according to record type. The SAR Detail Status field shows whether the record has been updated in the F4802 file.

```

9242                                SAR Log Transfer
SAR Number . . . 1079777 Multiple Jobs Submitted
Record Type. . .
and/or Member. . .
O
P Record Type Primary Secondary Data File T A
Log Item Log Item Library R C SAR Detail Sts
- M - Data Dictionary 4888 JDFCTL73 1 A Updated
- M - Data Dictionary 4889 JDFCTL73 1 A Updated
- C - Modified Source J9242S JDFSRC73 1 A Updated
- C - Modified Source J98901B JDFSRC73 1 A Updated
- C - Modified Source J98901T JDFSRC73 1 A Updated
- C - Modified Source P924124 JDFSRC73 1 A Updated
- C - Modified Source P9242 JDFSRC73 1 A Updated
- C - Modified Source P9242D JDFSRC73 1 A Updated
- C - Modified Source P924210 JDFSRC73 1 A Updated
- C - Modified Source P98905 JDFSRC73 1 A Updated
- C - Modified Source P98907 JDFSRC73 1 A Updated
- F - Software Reposit J9242S JDFCTL73 1 A Updated
- F - Software Reposit P9242D JDFCTL73 1 A Updated

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

```

If this screen 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 displays, 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.

92411W Project Promotion Paths	
Project . . TEC Tech Foundation Corre	
O Path Name	Description
— A62CUM	Transfer to current A62 Cum
— A62PC000TI	User based pricing
— A71CUM	Transfer to current A71 Cum
— A72CUM	Transfer to current A72 Cum
— JDF62	Transfer to JDF62
— JDF62TEC	Transfer to JDF62-SECURE
— JDF71	Transfer to JDF71
— JDF71TEC	Transfer to JDF71-SECURE
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.

```

9243                               Pre Promotion Edit History

Project . . . . . TEC              Tech Foundation Corrections
SAR . . . . . 1079777            Multiple Jobs Submitted
Promotion Path . JDF73          Transfer to JDF73

O                               Hrd No.
P  Date      Time      User ID  Err  Err
- 01/22/96  13:18:24  TFRCTL6
- 01/22/96   9:17:28  CHAN           8
- 01/16/96  12:39:59  CHAN           4

Opt: 1=Details  F5=Perform Edit  F24=More Keys
    
```

- To view the errors associated with a pre-promotion edit, enter 1 (Details) in the OP (Option) field next to the desired history record.

```

92431                               Pre Promotion Edit Details

Project . . . . . TEC              Tech Foundation Corrections
SAR Number . . . 1079777          Multiple Jobs Submitted
Promotion Path . JDF73          Transfer to JDF73
Date of Edit . . 01/22/96
Time of Edit . . 9:17:28

O R                               Member      Err      E
P T  Description      Name      2nd Item  Key      Description  T
- C Members Affe J9242S      1946 Object Not Found  W
- C Members Affe J98901B     1946 Object Not Found  W
- C Members Affe J98901T     1946 Object Not Found  W
- C Members Affe P924124     1946 Object Not Found  W
- C Members Affe P9242       1946 Object Not Found  W
- C Members Affe P9242D     1946 Object Not Found  W
- C Members Affe P98905     1946 Object Not Found  W
- C Members Affe P98907     1946 Object Not Found  W

Opt: 1=Error Details      F24=More Keys
    
```

- From the Pre-Promotion Edit History form, press F5 (Perform Edit) to perform the pre-promotion edit.
- Correct any errors and perform the edit until no errors occur. You do not need to resolve warnings that can occur.

Error Codes

The following table shows a partial list of errors and how to resolve them.

Error Code	Cause and Resolution
0020	<p>Cause: You entered a "From" library that does not exist or you are not authorized to use.</p> <p>Resolution: Correct the library name, create the library, or get authorization to use it.</p>

Error Code	Cause and Resolution
0092	<p>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.</p> <p>Resolution: Check your job log messages.</p>
1046	<p>Cause: An XJDE or ZJDE version was expected but not found.</p> <p>Resolution: If an XJDE or ZJDE version should exist, create it. If not, then change the processing option for form ID P926304.</p>
1370	<p>Cause: You entered a "From" table that does not exist or you are not authorized to use.</p> <p>Resolution: Review the "From" library for the promotion path control table. Either correct the library name or create the table.</p>
1371	<p>Cause: You entered a "To" table that does not exist or you are not authorized to use.</p> <p>Resolution: Review the "To" library for the promotion path control table. Either correct the library name or create the table.</p>
1372	<p>Cause: A key you wanted to copy from the "From" library does not exist.</p> <p>Resolution: Review the "From" library for the promotion path control table. Either correct the library name or re-enter the data record.</p>
2892	<p>Cause: A "From" library name is the same as the corresponding "To" library name.</p> <p>Resolution: Review the "From" and "To" libraries for the promotion path control table. Make the appropriate changes.</p>
4395	<p>Cause: No records exist in the Promotion Path Members file (F92401) for the promotion path you specified.</p> <p>Resolution: Complete the Promotion Path Members form for the promotion path.</p>
4396	<p>Cause: No records exist in the Promotion Path Control file Members file (F92402) for the promotion path you specified.</p> <p>Resolution: Complete the Promotion Path Control Files form for the promotion path.</p>
4397	<p>Cause: No records exist in the SAR Log table (F9810) for the project you specified.</p> <p>Resolution: In the project master record, change the based-on file for the Pre-Promotion Edit to the SAR Detail table (F4802), then manually update the SAR Detail records for the members and control file records updated by this project.</p>

Error Code	Cause and Resolution
4400	<p>Cause: No record exists in the Promotion Path Members file (F92401) for the function code of the member you want to promote.</p> <p>Resolution: For the specified promotion path, enter the environment for the function code of the member.</p>
4402	<p>Cause: No record exists in the Promotion Path Control Files file (F92402) for the control table you want to promote.</p> <p>Resolution: For the specified promotion path, enter the environment for the control file of the record.</p>
4439	<p>Cause: An error occurred while you attempted to copy a source code member.</p> <p>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.</p>

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 displays 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.

```

924124                               Software Transfer
Project . . . . . TEC                Tech Foundation Corrections
SAR Number . . . . 1079777          Multiple Jobs Submitted
Promotion Path . JDF73              Transfer to JDF73
Release . . . . . A73               Errors . . 000 Warnings . . 006

O Member                               From Environment                To Environment
P ID      Src File   Src Libr   Obj Libr   Src File   Src Libr   Obj Libr
- J924147 JDECLSRC  PGFSRC73  PGFOBJ73  JDESRC    JDFSRC73  JDPOBJ73
- P92402  JDESRC    PGFSRC73  PGFOBJ73  JDESRC    JDFSRC73  JDPOBJ73
- P924124 JDESRC    PGFSRC73  PGFOBJ73  JDESRC    JDFSRC73  JDPOBJ73
- P924127 JDESRC    PGFSRC73  PGFOBJ73  JDESRC    JDFSRC73  JDPOBJ73
- P924147 JDESRC    PGFSRC73  PGFOBJ73  JDESRC    JDFSRC73  JDPOBJ73
- V92402  JDESRC    PGRSRC73  PGFOBJ73  JDESRC    JDFSRC73  JDPOBJ73

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 JD Edwards World 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.

```

924127                               Control Files Copy
Project . . . . . TEC                Tech Foundation Corrections
SAR Number . . . . 1079777          Multiple Jobs Submitted
Promotion Path . JDF73              Transfer to JDF73
Release . . . . . A73               Errors . . 000 Warnings . . 000

O Record Type  Member Name  Secondary Name  Data Libr From  Data Libr To  Copy Status
- M - Data Dictio 4888                JDFCTL73      JDFTEM71
- M - Data Dictio 4889                JDFCTL73      JDFTEM71

- F - Software Re J9242S                JDFCTL73      JDFTEM71
- F - Software Re P9242D                JDFCTL73      JDFTEM71

Opt: 1=Copy to target library  F13=Copy All  F15=Edit History

```

6. 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 steps:

- [To create the transfer library](#)
- [To prepare the SAR system](#)
- [To define promotion paths](#)
- [To define a project](#)
- [To update the SARs](#)
- [To validate the promotion path](#)
- [To promote the project](#)
- [To save the transfer library to tape](#)
- [To restore the transfer library from tape](#)
- [To print the transfer library report](#)
- [To load the transfer library](#)
- [To 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 you 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 exit it will be created for you.  
  
( F6 - Execute )
```

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

```

98312                Build Transfer Library      Form ID. . . . P92414
Build Skeleton Transfer Library                Version. . . . ZJDE0001
                                                Display Level. 4
This job has various options described below. Enter the desired values and
press ENTER to continue.

Enter the name of the transfer library
to be created. If the library already
exists it will be CLEARED before use.
                MYLIBRARY

```

F5=Printer Overrides

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

                                Bottom
F3=Exit  F4=Prompt  F5=Refresh  F10=Additional parameters  F12=Cancel
F13=How to use this display  F24=More keys

```

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

                                Bottom
F3=Exit  F4=Prompt  F5=Refresh  F10=Additional parameters  F12=Cancel
F13=How to use this display  F24=More keys

```

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 displays. 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.

```

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

924143		JD Edwards World		Page	-	2
		Print Install Records		Date	-	2/05/16
Record Type	Primary Item	Secondary Item	Description	New/	Change	
Menus	G9261	Version Control			Changed	
Menus	G9262	Software Install			Changed	
Total Number of Records for Menus				00002		
.						
.						
DREAMwriter / PO	P00PURGE	ZJDE0024	Payee Control File Purge		Changed	
DREAMwriter / PO	P92412	ZJDE0001	Promote a Project		Changed	
DREAMwriter / PO	P92413	ZJDE0001	Manage Projects		Changed	
DREAMwriter / PO	P92414	ZJDE0001	Build Skeleton Transfer Librar		Changed	
DREAMwriter / PO	P924143	ZJDE0001	Control File Changes to be Ins		Changed	
DREAMwriter / PO	P924147	ZJDE0001	Load Transfer Software		Changed	
DREAMwriter / PO	P924801	ZJDE0001	SAR Inquiry by Reference		Changed	
Total Number of Records for DREAMwriter / PO				00007		
.						
.						
Vocabulary /Exits	R924143		Print Install Records		Changed	
Vocabulary /Exits	V9240		Promotion Path		Changed	
Vocabulary /Exits	V9240W		Promotion Path Master Window		Changed	
Total Number of Records for Vocabulary /Exits				00030		

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 displays. 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          MYLIBRARY
blank for no transfer.

Enter name of Target Object Library or     MYOBJ
blank for no transfer.

Enter Name of Target Source Library or     MYSRC
blank for no transfer.

EnterName of Target New Files Library or  MYDATA
blank for no transfer.

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.

```

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.

Transfer UDCs                1=Yes, 0=No.         0
Transfer AAI's              1=Yes, 0=No.         0
Transfer Menus              1=Yes, 0=No.         0
Transfer Data Dictionary    1=Yes, 0=No.         1
Transfer Vocabulary/Exits  1=Yes, 0=No.         0
Transfer CASE specs        1=Yes, 0=No.         0
Transfer Helps             1=Yes, 0=No.         0

More... +
F5=Printer Overrides
    
```

```

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.

Transfer SVR                1=Yes, 0=No         0
Transfer DREAMWriter       1=Yes, 0=No         0

Bottom.

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 (P924143).

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.

99630		Copy DD, VO, DW, UDC, SVR, Menus	
From Library	<u>MYLIBRARY</u>	To Library	<u>MYDATA</u>
Dictionary Item.	_____	Language	App1 Ovr Scrn/Rpt
Vocabulary Overrides	_____	Language	App1 Ovr
DREAM Writer Form.	_____	Language	
User Def Codes Sys	_____	Language	
Type.	__		
Software Versions Rep.	_____		
Menu Identification.	_____		
Generic Rate/Msg Sys	_____		
Type.	__		
F24=More			

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.

4 Programming Tools

Overview to Programming Tools

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 JD Edwards World 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.

Note: To create a data model, you must run the Data Model rebuild, P9804DM from the DREAM Writer version list.

Accessing Data Modeling

To access data modeling

1. Use one of the following methods to access Data Modeling.
 - Inquire on the file through Software Versions Repository (SVR) and press F23.
 - Type the menu selection on the command line and press F23.
 - From the Model Relations form (G9241), 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 )
```

2. Press F6 to continue.

The Data Modeling form displays with the cursor positioned in the field where you enter a file name.

98042	Data Modeling	Max Levels . 08
<u>Base File</u>		Function Use 230
		Display Dupl 1
		In Sys 0009 0304
Opt: 1=Move Top 5=Display 7=Where Used 8=Fields F11=Install/Reporting		

3. To view the Data Model, enter a file name and press Enter.

Field	Explanation
Max Levels	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.
Function Use	Displays the files that either match or have a function use less than the function use you specify.
Display Duplicate Relationships	Determines whether you want to display duplicate relationships or not. The valid values are: <ul style="list-style-type: none"> 1 No duplicates (default value) 2 First logical only 3 All files
In Sys	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.

4. To narrow the amount of file information displayed, specify values in the four fields appearing in the upper right of the form.

```

98042                               Data Modeling                               Max Levels . 08
                                     Function Use 230
                                     Display Dupl 1
                                     In Sys 00 09 03 04
Base File
F0006 Business Unit Master
- <M:1> F0010 F0010 Company Constants
-   |
-   | <1:M> F0901 F0901LE Account Master
-   |   |
-   |   | -M:M> F4801 F4801LB Work Order Master File
-   |   | -M:1> F0902 F0902LA Account Balances
-   |   |   |
-   |   |   | -M:M> F0311 F0311LG Accounts Receivable Ledger
-   |   |   | -M:M> F0411 F0411LK Accounts Payable Ledger
-   |   |   | -M:M> F0911 F0911LD Account Ledger
-   |   |   |
-   |   |   |
-   |   |   |

Opt: 1=Move Top 5=Display 7=Where Used 8=Fields F11=Install/Reporting
    
```

Detailed Explanation of a Line

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

```

Base File
F0006 Business Unit Master
|<M:1> F0010 F0010 Company Constants
    
```

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

Field	Explanation
Quantifier	The quantifier notation indicates the following: M:1 many to one 1:M one to many M:M many to many M:N many to zero or many N:M zero or many to many 1:N one to zero or many 1:1 one to one
Direction	The three direction notations are as follows: -> refers to <- referred to <-> way relation

Field	Explanation
Type	Used to distinguish between prototype and permanent files.
Subfile portion of screen	Displays the key fields that relate these two files together.

Function Key Exits from Data Modeling

Function Key	Description
F11	Install/Reporting Allows you to toggle between displaying install or reporting system codes.
F16	Rebuild A File Relationship 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. Print the Data Model
F21	Print the Data Model Prints the current data model

Selection Exits from Data Modeling

Option	Description
1	Move Top To select a file in the current data model and move it to the top to view its data model.
5	Display To view the file relationships. The Define a File Relationship form displays the relationship detail for the two files.
7	Where Used Exits to the Object Cross Reference Repository and displays all the programs that access the particular 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. The cross reference files are shipped empty. You must first run the Cross-Ref Index rebuild on menu G9642. If you add a new member to the Software Versions Repository, the Rebuild Cross Reference job must be run to ensure the new member is included in the display. You must have source code on your machine to run this rebuild.

Accessing the Object Cross Reference Repository

To access the Object Cross Reference Repository

Select one of the following methods:

- From the Master Directory (G), choose Hidden Selection 27. From Advanced and Technical Operations (G9), choose Documentation Services. From Documentation Services (G91), choose Object Cross Reference Repository.
- From Software Version Repository, press F15 to access the Object Cross Reference Repository.
- The Fast Path 'XREF' may also have been set up.

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 . _____

O  Name          Description          Field Attr T Start Upd
P  _____    _____          _____
   P000661      Business Unit Master Conversion      Y
   P06238       Report - Payroll Check Register      N
   P06371       Report - Certified Payroll Register    N
   P06611I      Tip Credit Generation with Interim Check N
   P06638       Sales Allocation Report              N
   P126410      STAR - Columnar Spreadsheet              N
   P200         Submit Network Job                          N
   P23250       Texas 250 Report                          N
   P26011      Gas Balancing - Entitlement Extract              N
   P26112      Gas Balancing - Entitlement Explosion to          N
   P26115      Gas Balancing Statement by Sales Point      N
   P26116      Gas Balancing Statement by Owner                  N
   P26119      Gas Balancing Master Subfile Display          N

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

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

The following form displays the statistics for program P0006.

```

980014          Object Cross Ref. Repository
Object: Name . . . P0006      Business Unit Master Revisions - Single
       Type . . . P          Statistics for program
       To Display #
       Funct Cd . _____

O  Name          Description          Field Attr T Start Upd
P  _____    _____          _____
   -           Total Statements in RPG II
   -           Total Statements as Comments
   -           Total Statements in RPG III
   -           Total Statements in Program
    
```

2. 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.

Note: If you are unfamiliar with the Cross Reference Relationships codes, type an asterisk (*) in the Type field, as shown below.

The Cross Reference Relationships codes appear in a new form.

```

Object: Name . . . F0006 Business Unit Master
Type . . . *
To Display _
Funct Cd . . .

O Name Description Field Attr T Start Upd
P Len Dec Y Loc Y/N

```

3. Press Enter.

```

81QM User Defined Codes Window
98 XR Cross-Reference Relationships
Skip To Code . . .
_ /D All data fields in /COPY
_ /F All files in /COPY
_ /I Program invocations from /COPY
_ /P Programs containing /COPY
_ CP All Programs using command
_ DF All files using data field
_ DP All programs using data field
_ EP Error messages in a program
_ F/ All /COPY members using file
4 FD All data fields in file
Opt: 4=Select F9=Glossary F14=Memo

```

4. Enter a 4 to select the desired code. The Object Cross Ref. Repository form displays with the selected codes. The F/D combination shown is very valuable to document the fields in a file.

```

980014 Object Cross Ref. Repository
Object: Name . . . F0006 Business Unit Master
Type . . . F All data fields in file
To Display D
Funct Cd . . .

O Name Description Field Attr T Start Upd
P Len Dec Y Loc Y/N
_ MCMCU Business Unit . . . . . 12 A 1
_ MCSTYL Type Business Unit . . . . . 2 A 13
_ MCDC Description - Compressed . . . . 40 A 15
_ MCLDM Level of Detail . . . . . 1 A 55
_ MCCO Company . . . . . 5 A 56
_ MCAN8 Address Number . . . . . 8 0 S 61
_ MCAN80 Owner/Receivable Address . . . . 8 0 S 69
_ MCNTY County . . . . . 3 A 77
_ MCADDS State . . . . . 3 A 80
_ MCDL01 Description . . . . . 30 A 83
_ MCDL02 Description 02 . . . . . 30 A 113
_ MCDL03 Description 03 . . . . . 30 A 143
_ MCDL04 Description 04 . . . . . 30 A 173
Opt: 1=SVR 2=Create Object 3=Field Expl F21=Print F16=Regenerate

```

Work with Data Dictionary

About the Data Dictionary Repository

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

The Data Dictionary represents a centralized repository 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

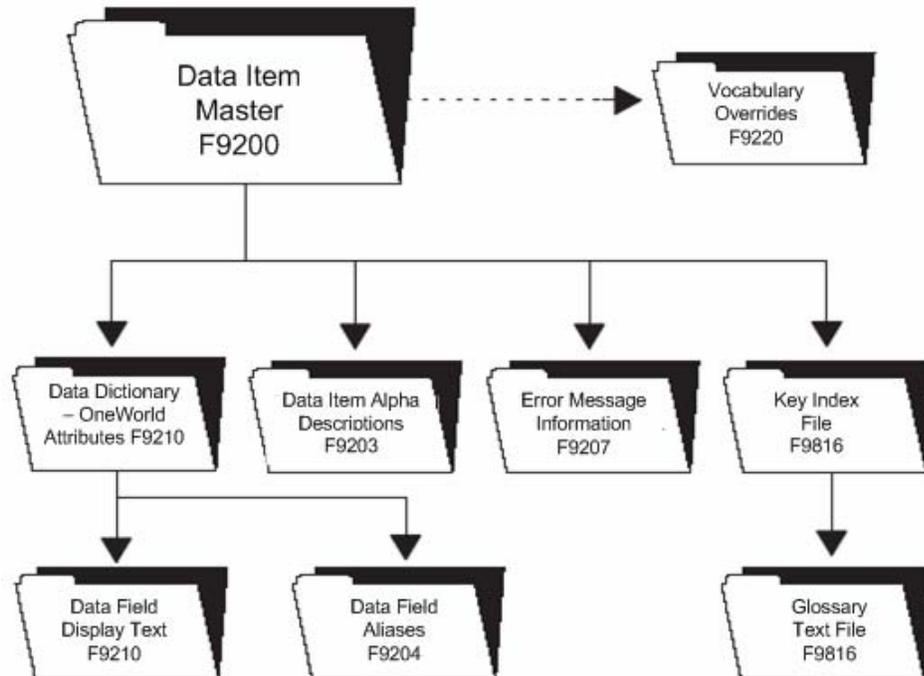
This section has the following topics:

- [Understanding the Data Dictionary Structure](#)
- [Locating A Data Item Name](#)
- [Working with the Data Dictionary](#)
- [Working with Data Item Alias Revisions](#)
- [Working with the Data Dictionary Glossary](#)
- [Working with User Defined Help Instructions](#)
- [Working with Data Field Descriptions](#)
- [Working with the Next Numbers Facility](#)
- [About the Field Reference File](#)
- [About the JD Edwards World Message File](#)
- [Locating the Rebuild FRF and JD Edwards World Msg File Form](#)

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 Dictionary - OneWorld Attributes (F9210)

This file contains the base display and validation rules for all file and data items. It is also used in Enterprise 1 environments.

It also contains database fields (glossary group of D or S), categories (glossary group C), and work fields (glossary group U). In addition, the C alias for each data item is stored in this file.

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 contains database fields (glossary group of D or S). It also contains COBOL aliases for each data item.

Error Message Information (F9207)

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 JD Edwards World 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 (option 3 from menu G01) 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.

```

01051                               Address Book Revisions
                               Long Addr No. _____
                               Resp. Bus. Unit . . . _____

Action Code. . . _
Address Number . _____

Alpha Name . . . _____

81QM  User Defined Codes Window  AT1
01  ST  Search Type
Skip To Code . . . .
- A      Applicants
- C      Customers
- E      Employees
- F      Facilities
- I      Investors
- J      Jobs
- M      Mail Distribution List
- O      Company
- P      Prospects
- Q      Participants
Opt:  4=Select  F9=Glossary  F14=Memo

Search Type. . . _
Payables Y/N/M . _
Receivable Y/N . _
Employee Y/N . . _
User Code. . . . _
Subldgr Inact. . _

13=Add'l Info F24=More

```

The data item name is often in the upper right corner of the help screen or the User Defined Codes screen. It can also be seen by using F9 for the Glossary on other Help screens.

Working with the Data Dictionary

The Data Dictionary provides many useful features. 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.

```

9201                               Data Dictionary                               Rls Last Chg _____
Action Code. . . . _                                                         Item Parent.
Data Item. . . . _
Glossary Group . . _
-----
Alpha Desc . . . . _
Reporting System . . . . _
System Code. . . . _   Type . _ Size . _____ Data File Decimals ___
Data Item Class. . . . _   Item Occurrences ___   Display Decimals . _
-----
Row Description. . . . _
Column Title . . . . _
-----
Default Value. . . . _
Data Display Rules _____ Justify. _
Data Edit Rules. . . . _
-----
Search Program . . . . _
Next Nbr System. . . . _   Next Number Index . . _
-----
F4=Search  F8=UDC  F9=Prev  F10=Glossary  F11=Descriptions  F15=Where Used

```

You find the Data Dictionary selection on several JD Edwards World menus and repository services.

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

Use the following fields where applicable:

Field	Explanation
Rls Last Chg	The software version number to be defaulted in the Software Versions Repository file.
Item Parent	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.
Data Item	<p>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.</p> <p>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 #, @, \$.</p> <p>You can create protected data names by using \$xxx and @xxx, where you define xxx.</p> <p>Messages can contain up to 10 characters. Types of messages are further defined by glossary group.</p> <p style="text-align: center;"><i>Form-specific information</i></p> <p>Messages can contain up to 10 characters. Types of messages are further defined by glossary group.</p>
Glossary Group	<p>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.</p> <p>See also 'What Are the Data Dictionary Glossary Groups?' within this 'Data Dictionary Repository' chapter.</p>

Field	Explanation
Alpha Desc	<p>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 (F4). To enter an alpha description, follow these conventions:</p> <p>Dates - Begin all Date fields with Date</p> <p>Amounts - Begin all Amount fields with Amount</p> <p>Units - Begin all Unit, Quantity, and Volume fields with Units</p> <p>Name - Begin all 30-byte description fields with Name</p> <p>Prompt - Begin any Y/N prompting field with Prompt</p> <p>Address Number - Begin all address numbers (employee, customer, owner) with Address Number</p>
Reporting System Code	Designates the system number for reporting purposes. This rarely differs from the Install System. Exceptions occur for data items used by more than one system.
Product Code	The application system code where the item is used.
Type	<p>The RPG data type. Refer to UDC 98/DT.</p> <p><i>Form-specific information</i></p> <p>Note: When using the 'O' type, create the field as large as possible. This allows the use of ideographic (Double byte) languages such as Japanese.</p>
Size	<ul style="list-style-type: none"> The length of the data item.
Data File Decimals	The number of positions to the right of the decimal of the data item. This is usually zero. See Display decimals.
Data Item Class	Defines the essential attributes and characteristics of a data item. There will be a data item for each of the Data Item Class entries. Data items are grouped into these Classes and the Data Item attributes are the same as the Class. F1 on this field will show all the defined Classes.
Item Occurrences	<p>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.</p> <p>The array data item names are restricted to certain lengths depending on the number of array elements:</p> <p>3 bytes - 1 to 9 elements</p> <p>2 bytes - 10 to 99 elements</p> <p>1 byte - 100 to 999 elements</p>

Field	Explanation
Display Decimals	<p>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. Data Item Class groups will have the same display decimals as the Class definition. Most decimal data fields are stored in the data files as whole numbers and the Display Decimals value is used to display or print the data correctly.</p>
Row Description	<p>Stores 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:</p> <p>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>
Column Title	<p>The first line of description that will be used in column headings on a report or screen. 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 two are needed.</p>
Default Value	<p>Used as the default value on the data entry screen for the associated data item. This value will be entered into the field upon exit from the screen if no other value is entered. 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.</p> <p>Caution: If a blank entry is allowed, default values should not be used.</p>

Field	Explanation
Data Display Rules	<p>Keywords which describe an editing technique applied when data is displayed. Validation is applied to the data after Enter is pressed. The rules will be incorporated into the source code at program generation time. This is usually numeric editing or masking.</p> <p>The developer can override these rules at the time of program creation.</p> <p>The current list of these rules is kept in the User Defined Codes at SYSTEM = 98 and RECORD TYPE = DR.</p>
Data Edit Rules	<p>Keywords which control allowed values when data is entered. Validation is applied to the data after Enter is pressed. These controls will be incorporated into the program at generation time. The data may be required to be in a range, in a UDC table, or in a file.</p> <p>The developer can override these rules at the time of program creation.</p> <p>The current list of these rules is kept in the User Defined Codes at SYSTEM = 98 and RECORD TYPE = ER.</p>
Search Program	<p>The Help Text Program 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. There are some fields where the Help Program is hard coded in the Help system such as AN8 in the Name Search program (P01200). 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 '*UDC' in this field (this is the default when 'UDC' is entered in the Data Edit Rules field). If you do not want the UDC window to appear and you have 'UDC' in the Data Edit Rules field, change this field to be blank.</p> <p>Program Requirements: For your text program to work correctly, you must allow it to accept three standard parameters:</p> <ul style="list-style-type: none"> ▪ 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	<p>Designates the system number for the Next Number retrieval. See User Defined Codes, system code '98', record type 'SY'.</p>
Next Number Index	<p>The array element number retrieved in the Next Number Revisions program. There are ten NN array elements for each System Code. For example, the next Address Book number (AN8) is array element '01' of system '01'.</p>

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 add/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 should be set up with Inquiry 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.

Function Key	Description
F4	A data item search facility. Enter the search text in the Alpha Name field on the Data Item Search screen. 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.
F6	Repository Services
F8	User Defined Code Tables
F9	Automatic Reinquiry
F10	Glossary
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 displays 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 screen

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

```

9201          Data Dictionary Repository  Rls Last Chg
                                           Item Parent .
Action Code. . . . I
Data Item. . . . AT1
Glossary Group . . D
-----
Alpha Desc . . . Search
Reporting System . 01
System Code. . . .
Data Item Class. .
-----
Row Description. . Search
Column Title . . . S
                    T
                    -
                    Defa
Default Value. . .
Data Display Rules
Data Edit Rules. . UDC 01
                    ST
Search Program . .
Next Nbr System. . Next Number Index . .
-----
F4=Search F8=UDC F9=Prev F10=Glossary F11=Descriptions F15=Where Used
    
```

```

9204          Data Field Alias
Action Code. . . I
Data Item. . . . AT1
Search Type
Alias
Type          Alias
1 ADDRESS TYPE 1
2 Address Typel
-----
F24=More
    
```

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 JD Edwards World' 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 JD Edwards World software. All glossary groups typically have associated text. The glossary stores this text. The major glossary groups follow:

Group	Explanation
E	JD Edwards World Interactive error messages <ul style="list-style-type: none"> ▪ JD Edwards World 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

Group	Explanation
M	<p>Menu Messages</p> <ul style="list-style-type: none"> ▪ JD Edwards World 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	<p>JD Edwards World Batch error messages</p> <ul style="list-style-type: none"> ▪ JD Edwards World 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 ▪ The JD Edwards World program found on Rebuilds and Global Updates (G9642) builds the batch error messages file QJDEMSG.
C	<p>Data Field Function Categories (Data Item Class)</p> <ul style="list-style-type: none"> ▪ Groups common data elements ▪ For example, CURRENCY, QTYINV.
D or S	<p>Primary or Secondary Data Items</p> <ul style="list-style-type: none"> ▪ Used for validations ▪ Text on Videos ▪ Text on Reports ▪ Field Reference Files - F98FRFA-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	<p>User Defined program Helps</p> <ul style="list-style-type: none"> ▪ Client use only for adding custom helps for JD Edwards World programs ▪ For example, U00MENU, U01051. ▪ When HELP is keyed on command line, 'F5' on the help window indicates that there are customer helps available. Use F5 to view.
L	Report Messages. Messages or warnings for certain procedures, or letters written and produced through DREAM Writer

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.

Note: Always leave the last two character positions of each text line blank.

Field	Explanation
Data Item	<p>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.</p> <p>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 #, @, \$.</p> <p>You can create protected data names by using \$xxx and @xxx, where you define xxx.</p> <p>Messages can contain up to 10 characters. Types of messages are further defined by glossary group.</p> <p style="text-align: center;"><i>Form-specific information</i></p> <p>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 displays on a successful add. You should assign interactive error message numbers greater than 5000.</p>

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 the Data Dictionary screen, press F11.

O	Lan	Appl		Column
P	Over		Description	Titles
---	44	Row	Vendor Number	Vendor Number
---	48	Row	Customer Number	Address Number
---		Row		
---		Row		

Opt: 5=Glossary F9=Redisplay Prev F19/20=Prev/Next Item F24=More

2. On the Data Field Descriptions screen, 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.

0002		Next Numbers	
Action Code.	I	General Accounting	
System Code.	09		
<u>Use</u>	<u>Next Number</u>	<u>Check Digit</u>	
<u>Next Account ID</u>	<u>21831</u>	-	
<u>Journal Entries</u>	<u>1946</u>	-	
<u>Consol Accounts</u>	<u>90000214</u>	-	
		-	
		-	
		-	
		-	
		-	
		-	
		-	
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

Next Numbers

The next numbers file is F0002

- 10 element array
- 1 record per system
- Modulus 11 check digit 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, Data Item AID has System Code 09/01.

What Happens with the Rebuild?

The system does the following:

- Rebuilds F98FRFA-Z, \$, and @ using Data Dictionary data item glossary groups D and S
- Rebuilds the message file (QJDEMSG) in QGPL
- Uses a processing option located on Form ID J98DDMSGF, to determine which library to build the QJDEMSG file. The default is QGPL
- Does not rebuild the JD Edwards World 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 JD Edwards World Message File

The JD Edwards World 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 JD Edwards World Message (QJDEMSG) file.

Rebuilding only the JD Edwards World 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. Only those translated error messages which can be seen on the Glossary screen with a language code will be included. Enter ‘***’ for all languages installed on the system.

Locating the Rebuild FRF and JD Edwards World Msg File Form

To locate the Rebuild FRF and JD Edwards World Msg File form

From menu G9642, choose FRF & JD Edwards World Msg File. The first and third fields are self-explanatory. The source referred to on the second parameter is not useful so enter QTEMP in that field.

```
98FRF                               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
```

Work with Data File Design Aid

About the Data File Design Aid

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

To enforce standards, JD Edwards World recommends against file changes made through the Source Entry Utility (SEU). Changes should be done through File Design Aid. Non-JD Edwards World changes to a JD Edwards World file make support difficult if not impossible.

What You Should Know About

Enforced Prefixes

Data items are defined in the JD Edwards World Data Dictionary. Within files, these data item names are qualified with a prefix to make them unique. Every data file in JD Edwards World 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

JD Edwards World 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

- JD Edwards World uses IBM's Field Reference File (FRF) facility when creating files. When creating the DDS for a file, you need only enter the Data Dictionary data item name. The 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.

Pressing F10 from this form displays all file prefixes that you should not use.

Entering Data File Design Aid

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 Item	Data Field Desc.	K/S	Function Specifications	Seq No
I92801		R		1.00
XIT	Item ID	-	REFFLD(XIT F98FRFX)	2.00
XDS	Description	-	REFFLD(XDS F98FRFX)	3.00
XTY	Item Type	-	REFFLD(XTY F98FRFX)	4.00
XDT	Date Last Ship	-	REFFLD(XDT F98FRFX)	5.00
XBU	Business Unit	-	REFFLD(XCC F98FRFX)	6.00
XQT	Quantity - On Hand	-	REFFLD(XQT F98FRFX)	7.00
XUM	Item Unit of Measur	-	REFFLD(XUM F98FRFX)	8.00
X001	Item Category Code	-	REFFLD(X001 F98FRFX)	9.00
X002	Item Category Code	-	REFFLD(X002 F98FRFX)	10.00
X003	Item Category Code	-	REFFLD(X003 F98FRFX)	11.00
X004	Item Category Code	-	REFFLD(X004 F98FRFX)	12.00
X005	Item Category Code	-	REFFLD(X005 F98FRFX)	13.00
XIT	Item ID	K		14.00

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

Field	Explanation
File Description	SVR member description for the file.
Unique Keys(Y/N)	Specifies if the data file contains unique keys. <ul style="list-style-type: none"> ▪ 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. <p style="text-align: center;"><i>Form-specific information</i></p> <p>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.</p> <p>Most JD Edwards World 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.</p>

Field	Explanation
Member ID	The name assigned to the file. Defaults in from the Software Versions Repository.
File Prefix	This is the SVR field which indicates the prefix associated with a file.
Src Library	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 from the Software Versions Repository.
Based on File	<p>Designates the physical file on which a logical file is based.</p> <ul style="list-style-type: none"> ▪ Defaults in from the Software Versions Repository and only displays for logical files. <p style="text-align: center;"><i>Form-specific information</i></p> <p>For physical and logical files, the Based On File is the same as the physical file.</p> <p>For join files, the Based On File is the name of the first physical file that the join is built over.</p>
Data Item	<p>The Data Dictionary name of the field or the record format name.</p> <ul style="list-style-type: none"> ▪ The file prefix is added to create a unique data name for each field specified in this field. ▪ The record format line is automatically defaulted in.
K/S	Identifies the DDS Type indicating whether the field is a format name (R), key field (K), select logic field (S) or omit logic field (O). It may be used in conjunction with information that appears in the Function Specifications field.
Function Specifications	<p>Used with the DDS Type specified in the K/S column.</p> <ul style="list-style-type: none"> ▪ If it is a record format name this field will be blank. ▪ 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).

Field	Explanation
Seq No	Determines the order of the fields in the file. <i>Form-specific information</i> When designing a physical, list the component fields in descending order of their importance to the file. The K entries for keyed fields must always be last in sequence number within the Data File Design Aid program itself.
Data Item Type	A is for Alpha, S is for simple numeric, P is for Packed numeric, and O is for Open (any character can be entered).
Item Size	Length of field taken from the Data Dictionary.
Display Decimals	Usually Packed fields, the number of decimal places on a display. The data will actually be stored in the field in the file as a whole number.

Note: The detail area includes additional information: data item type, data item size, and number of display decimals. This information will be available after the enter key is pressed, at which time the information will be retrieved from the Data Dictionary along with the REFFLD keyword which indicates which F98FRF file was used to retrieve the field information.

Sample - Logical File

JD Edwards World logical files contain all fields from the PF, only keys are specified.

```

92102                               Data File Design Aid
File Description . . . LF - Business Unit, Item ID
Unique Keys (Y/N) . . . Y           Member ID. . . . . F92801LA
File Prefix. . . . . QX           Src Library. . . . . PGFSRC
Based on File. . . . . F92801     Source File Name . . . JDESRC

Data Item. Data Field Desc.  K/S Function Specifications      Seq No
I92801     Business Unit      R  PFILE(F92801)                1.00
XCC        Business Unit      K  _____                  2.00
XIT        Item ID            K  _____                  3.00
_____    _____          _____                  4.00
_____    _____          _____                  5.00
_____    _____          _____                  6.00
_____    _____          _____                  7.00
_____    _____          _____                  8.00
_____    _____          _____                  9.00
_____    _____          _____                 10.00
_____    _____          _____                 11.00
_____    _____          _____                 12.00
_____    _____          _____                 13.00
_____    _____          _____                 14.00

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.

92102		Data File Design Aid		
File Description . . .	LF - Acct ID, LT, DOI, Sub LT, Serv Date, Doc TyN#			
Unique Keys(Y/N) . . .	-	Member ID.	F0911LH	
File Prefix.	GL	Src Library.	PGFSRC	
Based on File.	F0911	Source File Name . . .	JDESRC	
Data Item.	Data Field Desc.	K/S	Function Specifications	Seq No
I0911		R	PFILE(F0911)	1.00
AID	Account ID	K		2.00
LT	Ledger Type	K		3.00
DOI	DOI Sub	K		4.00
SBL	Subledger	K		5.00
DSV	Date - Service/Tax	K		6.00
DSVY	Date - Service/Tax	K		7.00
DSVM	Date - Service/Tax	K		8.00
DSVD	Date - Service/Tax	K		9.00
DCT	Document Type	K		10.00
DOC	Document (Voucher,	K		11.00
KCO	Document Company	K		12.00
POST	G/L Posted Code	S	CMP(EQ 'P')	13.00
BC	Bill Code	-	CMP(NE 'H')	14.00
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 omits.

92102		Data File Design Aid		
File Description . . .	LF - Report Code 01			
Unique Keys(Y/N) . . .	-	Member ID.	F0101LH	
File Prefix.	AB	Src Library.	PGFSRC	
Based on File.	F0101	Source File Name . . .	JDESRC	
Data Item.	Data Field Desc.	K/S	Function Specifications	Seq No
I0101		R	PFILE(F0101)	1.00
AC01	Category Code - Add	K		2.00
ALPH	Name - Alpha	K		3.00
AN8	Address Number	K		4.00
DFI	Date - First Invoic	O	CMP(EQ 000000)	5.00
DLI	Date - Last Invoice	-	CMP(EQ 000000)	6.00
				7.00
				8.00
				9.00
				10.00
				11.00
				12.00
				13.00
				14.00
F3=Exit/Save F16=Search by File F1=Search by Name F4=Field Attributes				

Creating Join Files and Work Files

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

Function Keys From File Design Aid

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

Function Key	Definition
F2	Access the command line to enter a JD Edwards World 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	Press to exit Data File Design Aid, the following form appears.

Example F3 - Data File Design Aid

```

                                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.

Function Key	Definition
F6	This form provides access to other repository services within JD Edwards World.
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?

Field	Explanation
Unique Keys	<p>Specifies if the data file contains unique keys. If Yes, FDA puts the UNIQUE keyword in the DDS. No two records in the data file can have duplicate data in the key fields. If No, FDA leaves the UNIQUE keyword out of the file DDS. Records with duplicate keys will be allowed in the data file.</p> <p style="text-align: center;"><i>Form-specific information</i></p> <p>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.</p> <p>Most JD Edwards World physical files in the past have been defined as arrival sequence and logicals were used for creating keyed sequences. More recently, however, physical files have been keyed.</p>
File Description	<p>The description of a record in the Software Versions Repository file. The member description is often consistent with the base member description.</p> <p style="text-align: center;"><i>Form-specific information</i></p> <p>The description associated with each file is used to further identify the relation of the file and its purpose.</p> <ul style="list-style-type: none"> ▪ Physical files should have a description that explains the purpose of the file. ▪ Logical files should be designated as follows: LF - fldname, fldname, fldname: where fldname is a key field. ▪ Join files should be designated as follows: JF - filename/filename/filename - fldname,fldname,fldname; where the filename is a file over which the join is built and fldname is the key field joining the files. ▪ Work files should be designated as follows: WF - filename; where filename is the file that the work file accesses.
Based On File	<p>Designates the physical file on which a logical file is based.</p> <ul style="list-style-type: none"> ▪ Defaults in from the Software Versions Repository and only displays for logical files. <p style="text-align: center;"><i>Form-specific information</i></p> <p>For physical and logical files, the Based On File is the same as the physical file.</p> <p>For join files, the Based On File is the name of the first physical file that the join is built over.</p>

Field	Explanation
Ordering of Fields	Determines the order of the fields in the file. <i>Form-specific information</i> 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.
Logical Files	Logical files may include all fields; we do not define specific fields.
Recompiling	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.
Record Format	It is a JD Edwards World 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.
Field Reference Files	Used in all file creations to retrieve field descriptions.

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.
- May add new keys to both logical and physical files.
- May change the file format of logical or physical files.

Note: The Data Model displays relational models of the major files within the JD Edwards World environment.

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, F98FRFS, and F98FRF@).
 - Each field reference file points to all the data items beginning with the same character as the field reference file.

For example: F98FRFA is a logical file which connects the F98FRFA1, F98FRFA2, and so on. physical files which contain all the Data Dictionary data items beginning with the letter A.

Work with Screen Design Aid

About Screen Design Aid

Screen Design Aid (SDA) is an interactive feature you use to design and maintain screens. 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 SDA:

- Design is conducted in a safe work environment. If you make a mistake you can exit without changing a screen's Data Description Specifications (DDS).
- Screen specifications are stored in data structures in the QRECOVERY library. This is similar to the IBM recovery of SEU.
- You can create a screen in normal mode (80 columns by 24 rows) or wide mode (132 columns by 27 rows). You can also design wide screens on 80 column devices using a windowing facility.
- Answering initial yes/no options allows you to create a basic screen skeleton for a subfile, non-subfile or window-style screen.
- SDA is fully integrated with the Data Dictionary and vocabulary override files. You can place fields on the screen by referring to a Data Dictionary name and override default attributes, if necessary. You can place vocabulary override fields on the screen and, if desired, modify their contents through the full screen.
- 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 screen individually or all at once by pinpointing locations on the full screen 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 screen.
- Unlike the IBM SDA, JD Edwards World 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 screen 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.

This section contains the following:

- [Editing Commands](#)
- [Prefix Standards](#)
- [Field Name Standards](#)
- [Working with Screen Design Aid](#)
- [Function Key Exits](#)
- [Updating an Existing Field](#)
- [Accessing Fast Path Create for a New Screen](#)
- [Adding Fields without Using a Pick List](#)
- [Adding a Literal Field](#)
- [Using the *BOTH and *ALL Features](#)
- [Compiling Your Screen](#)
- [Screen Design Standards and Tips](#)
- [Adding Screen Fields Using Pick List](#)
- [Selecting Database Fields](#)
- [Placing Fields on a Screen Using a Pick List](#)
- [Function Key Exits from Screen Design Aid](#)
- [Changing Subfile Boundaries](#)
- [Process Overview - Placing Selected Fields](#)
- [Process Overview - Revising the Field Definition](#)
- [Process Overview - Revising Vocabulary and Function Keys](#)
- [Function Keys for Screen and Display Format Control](#)
- [Summary of Screen Design Aid](#)

Editing Commands

Standard editing commands available in SDA include:

Command	Description
*DEL	Delete fields (used in Field Definition window)
d (cannot be uppercase D)	Delete fields (used in Field Definition window)
<<, >>	Shift fields to the left or right
(xx...xx) 'xx...xx'	Literals (use apostrophes)
-	Move from position.
=	Move to position.

Command	Description
- -	Move block from position
=	Move block to position.
F7	Restore the screen if you accidentally press Field Exit or a power failure knocks you off.

Precautions and automatic features of the SDA include:

- 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 screens to highlight all fields on the last line of the subfile to indicate that no more records exist.

Prefix Standards

Prefix standards for use in the SDA include:

Prefix	Description
VD	Screen display fields. <ul style="list-style-type: none"> ▪ VD fields display database information from the file being used for the screen 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. <ul style="list-style-type: none"> ▪ 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 screen.

Field Name Standards

Field name standards for use in SDA include:

Field	Standard
VC0 - Screen constants	<p>VC0 (zero) fields display definitions or descriptions for a single piece of data or for a group of data.</p> <p>VC0 fields are always output fields and the description that is loaded into the VC0 field is obtained from a separate file</p> <ul style="list-style-type: none"> ▪ For example, if creating a screen 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 window when adding a new constant or description field. ▪ The default size for VC0 fields is 30.
VTX - Screen text	<p>These fields display the row description or column headings from the Data Dictionary.</p> <ul style="list-style-type: none"> ▪ 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. <p>You enter *VTX in the Field Name field in the Field Definition window for the next sequential text field name.</p> <ul style="list-style-type: none"> ▪ The default size for VTX fields is 16.
Line 24 is always VDL24	<p>You cannot change the text for Line 24 by using the Field Definition window because it is too large.</p> <p>Type over the text in Line 24 to change it.</p>
TTL@	<p>Uses the default title from Vocabulary Overrides if the screen program is called from another program.</p> <p>Uses the menu selection text if the screen is called from a menu.</p>
ACTION	<p>Action Code field.</p> <ul style="list-style-type: none"> ▪ The name assigned by SDA. ▪ The default cursor keyword is assigned to the action code field.
*LITER - Literal fields	<p>Literals are added by placing apostrophes around the text on the screen and pressing Enter. (For example, 'V928011').</p>

Updating or Adding Fields through SDA

Field	Explanation
* - Field Definition Window	<p>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 window.</p> <ul style="list-style-type: none"> ▪ To add a field, place an asterisk (*) on the SDA design area where you want to add the field. ▪ To update a field, place an asterisk in the attribute character (first position to the left) of the field you want to update. <p>You can pull in the screen 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 Name field on the window (*BOTH or *ALL).</p>
& - Field Selection Window	<p>Allows you to add new fields using the Pick List feature Causes the Field Selection window to display.</p> <p>To place a field on the screen from your Pick List, place an ampersand (&) on the SDA design area where you want to place the first character of the field.</p> <p>Allows you to pull in one or all of the following at the same time:</p> <ul style="list-style-type: none"> ▪ The Row Description/Column Headings (VTX) ▪ The screen field ▪ A description field (VC0)

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 screen 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 screen based on the member's Function Code value.

Field	Explanation
Dict Name	Identifies the four-byte data item name from the Data Dictionary. This is the only required field for most data items, the rest will default.
Text	Describes the Dictionary Name. VTX fields contain the soft coded description from the Data Dictionary that updates F9220 (Vocabulary Overrides).
Data Type	S Numeric data items. A Alphanumeric. Blank (w/decimal position blank in Size field)defaults to A. Blank (w/decimal position defined in Size field) defaults to an S. All JD Edwards World fields are defined as A.
Field Name	Identifies a screen field name. <ul style="list-style-type: none"> ▪ *VTX (VTX001-VTX200) automatically assigns next available. ▪ *VC0 (VC0001-VC0200) automatically assigns next available. ▪ *LITER literal fields. ▪ *BOTH or *ALL to bring in screen (VD), VC0, and/or 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. <ul style="list-style-type: none"> 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. <ul style="list-style-type: none"> ▪ If left blank, automatically fills.
Text Form	For VTX fields, identifies the field from the Data Dictionary that is used for headings. <ul style="list-style-type: none"> R Row Description. C Column Heading 1. D Column Heading 2.

Field	Explanation
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. <ul style="list-style-type: none"> ▪ 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 set on whenever the value in this field is changed.
OVERDTA	OVERDTA keyword is in effect, Y or N. Used with PUTOVR to override data that is in a field already on the screen.

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

Field	Explanation
Duplicate	Duplicate the data. Only valid for a SFL format. Puts the DUP keyword in the screen/report DDS but the Program Generator does <i>not</i> generate any code to enable this.
OVRATR	OVRATR keyword is in effect, Y or N. Used with PUTOVR to override display attributes of a field on the screen.
Field Cond	Field Conditioning Indicators. Determines if the user can see the field or not.
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: <ul style="list-style-type: none"> RI Reverse Image HI Highlight UL Underline ND Nondisplay BL Blink PR Protect PC Place Cursor A blank or N will deactivate the condition.

Field	Explanation
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 JD Edwards World or SAA colors in QJDF.

Accessing Fast Path Create for a New Screen

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

To access Fast Path Create for a new screen

1. Locate your screen and enter selection 10.

If SDA cannot find the existing DDS for your screen, the Create New Screen screen displays:

```

92510                               Create New Screen
Screen: V927400
Text Description. . . Item Search
                               (Y/N)
Fast Path Create                 Y
Screen Type                      Y
Action Code                      Y
Window                          N
Wide Screen (Y/N). .           N
Subfile Creation
Subfile                          N
Subfile Fold                    N
Subfile Clear                   Y
Selection Exits                 N
Record Format Level
PUTOVR                          N
OVERLAY                         Y
                               F3=Exit   F12=Previous

```

Field	Explanation
Screen and Text Description	Taken from the SVR entry for this member.
Fast Path Create	Automatically create record formats, fields, file, and record level parameters.
Action Code	Automatically create an Action Code field.
Window	Screen is a window.
Wide Screen	Screen is in wide format (132 columns by 27 rows) or normal format (80 columns by 24 rows).
Subfile	Create subfile format.

Field	Explanation
Subfile Fold	Create a fold area in the subfile using SFLDROP and SFLFOLD keywords.
Subfile Clear	Use SFLCLR (Y) OR SFLINZ (N).
Selection Exits	Create selection exits to allow the user to exit the program using selection codes.
PUTOVR	The screen record format uses the PUTOVR keyword. Causes the screen to be erased and redisplayed when a window is displayed.
OVERLAY	The screen record format uses the OVERLAY keyword. Will not erase and redisplay screen when a window is displayed. Most JD Edwards World screens use OVERLAY.

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

Example - Screen with Action Code and No Subfile

```

92700                               Item Maintenance
Action Code. . . B

F24=More Keys

```


Adding Fields without Using a Pick List

To add a Screen Text Field (VTX)

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

```

928011                               Item Master Information
Action Code. . . E
*

Screen: V928011-----Field Definition-----Format: V9280111-
Dict Name  XIT      Text
Data Type  -       Field Name *VTX      Cond Ind
Row/Column - -     Field Use  -
Size       - -     Text Form  R        RI - - - - -
Dft Cursor - -     Edited      -        HI - - - - -
Lower Case - -     Change     - -     UL - - - - -
OVRDTA    -       Duplicate  - -     ND - - - - -
OVRATR    -       Field Cond - - - - - BL - - - - -
                                           PR - - - - -
                                           PC - - - - -
--F3=Exit  F12=Prev Screen F17=Dictionary-----

```

2. When the Field Definition window displays, do the following:
 - In the Dict Name field, enter the Data Dictionary item name.
 - In the Field Name field, specify *VTX.
The system assigns the next available VTX number.
3. 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.

Note: Text defaults from the Data Dictionary based upon the Text Form value. The default value is R for non-subfile formats.

4. 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.

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 Screen Field (VD)

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

```

928011                Item Master Information
Action Code. . . E
                    *

Screen: V928011-----Field Definition-----Format: V9280111-
Dict Name  XIT      Text
Data Type  -       Field Name
Row/Column -       Field Use  B          RI - - - - -
Size       -       Text Form  -          HI - - - - -
Dft Cursor -       Edited     -          UL - - - - -
Lower Case -       Change     -          ND - - - - -
OVRDTA    -       Duplicate   -          BL - - - - -
OVRATR    -       Field Cond  -          PR - - - - -
                    -          PC - - - - -
--F3=Exit  F12=Prev Screen  F17=Dictionary-----

```

2. On the Field Definition window, 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. The Data Type, Size, and Text default from the Data Dictionary.

To add a Screen Constant Field (VC0)

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

```

928011                Item Master Information
Action Code. . . E
                    *

Screen: V928011-----Field Definition-----Format: V9280111-
Dict Name  -       Text
Data Type  -       Field Name  *VC0
Row/Column -       Field Use  -          RI - - - - -
Size       -       Text Form  -          HI - - - - -
Dft Cursor -       Edited     -          UL - - - - -
Lower Case -       Change     -          ND - - - - -
OVRDTA    -       Duplicate   -          BL - - - - -
OVRATR    -       Field Cond  -          PR - - - - -
                    -          PC - - - - -
--F3=Exit  F12=Prev Screen  F17=Dictionary-----

```

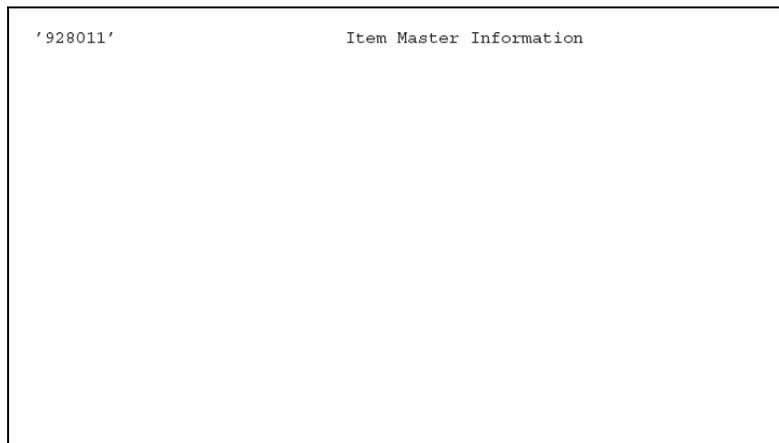
The Field Definition window displays .

2. On the Field Definition window, specify *VC0 in the Field Name field.
The system assigns the next available VC0 number.
3. To override the default length of 30, enter a value in the size field.

Adding a Literal Field

JD Edwards World standard is that the only literal on a screen is the program ID in the top left corner.

To add a literal field



1. Enter the literal text in the SDA Design area.
2. Enclose the text within single quotes.
3. Press Enter.

Using the *BOTH and *ALL Features

The Field Definition window 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 screen (VD) field on the screen.

To use *ALL

On the Field Definition window

```

928011                      Item Master Information
Action Code. . . . B
Item ID. . . . . BBBBBBBB      Item Desc. . . . BBBBBBBBBBBBBBBBBBBBBBBBBB
Business Unit. . . BBBBBBBBBB  00000000000000000000000000000000
Item Type. . . . . BB          00000000000000000000000000000000
Date Last Ship . BBBBBBBB
Qty On Hand. . . . BBBBBBBBBB

*
Item Code 001. . . . . BBB      00000000000000000000000000000000
Screen: V928011-----Field Definition-----Format: V9280111-
Dict Name  XUM      Text
Data Type  _____  Field Name  *ALL      Cond Ind
Row/Column _____  Field Use  _____  RI  _____
Size       _____  Text Form  _____  HI  _____
Dft Cursor _____  Edited    _____  UL  _____
Lower Case _____  Change   _____  ND  _____
OVRDTA    _____  Duplicate _____  BL  _____
OVRATR    _____  Field Cond _____  PR  _____
PC  _____

--F3=Exit  F12=Prev Screen  F17=Dictionary-----
    
```

Type “*ALL” in the Field Name.

When you enter *ALL, the following screen displays in the SDA design area for the Unit of Measure field:

```

928011                      Item Master Information
Action Code. . . . . _
Item ID. . . . . _____  Item Desc. . . . _____
Business Unit. . . _____  00000000000000000000000000000000
Item Type. . . . . _          00000000000000000000000000000000
Date Last Ship . . _____
Quantity On Hand . _____

Unit of Measure. _____  00000000000000000000000000000000
Item Code 001. . . . . _____  00000000000000000000000000000000
Item Code 002. . . . . _____  00000000000000000000000000000000
Item Code 003. . . . . _____  00000000000000000000000000000000
Item Code 004. . . . . _____  00000000000000000000000000000000
Item Code 005. . . . . _____  00000000000000000000000000000000

F24=More Keys
    
```

Field Default Values

The following are field defaults in SDA:

Field	Explanation
VD - Screen Display field	<ul style="list-style-type: none"> 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.

Field	Explanation
VTX - Screen Text field	<ul style="list-style-type: none"> 16 bytes long Defaults to Row description rather than column description
VC0 - Screen Constant field	<ul style="list-style-type: none"> 30 bytes long

Understanding the SDA Exit/Save Function Key

F3 - Design Aid Exit/Save

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

```

92590                                Design Aid Exit/Save

Save DDS (Y/N) . . . N
Member ID. . . . . V928200
File ID. . . . . JDESRC
Src Library. . . . . STB301SRC

Description. . . . . Item Search
Function Code. . . . . DSPF

Return to Edit (Y/N) N

F12=Previous

```

Field	Explanation
Save DDS (Y/N)	Saves the DDS and updates or creates Vocabulary Overrides and Function Key definitions.
Member ID	The record of the Software Versions Repository member to be copied. <i>Screen-specific information</i> Name of the screen.
File ID	Identifies the file that will contain the source code.
Src Library	Identifies the library where the source code resides.
Description	The description of a record in the Software Versions Repository file. The member description is consistent with the base member description.
Function Code	Should be DSPF for a Screen.
Return to Edit (Y/N)	EOJ or allows return to SDA.

Alignment

Line up fields vertically. This includes row descriptions, input fields, and description fields. Fields on the left side of the screen 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 Care Reimb. Account
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 descriptive 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.

```

08332                               Single D/B Relation Entry

Action Code. . . . . _____
Employee Number. . . . . _____
Plan ID. . . . . _____
Dependent/Beneficiary No.. _____ Name . _____
Effective From . . . . . _____ Thru . _____

Relationship Data:
Dependent or Beneficiary . _____
Relationship . . . . . _____
Dep/Ben Type . . . . . _____
Percent Allocated. . . . . _____

Dependent/Beneficiary Data:
Social Security Number . . _____
Date Of Birth. . . . . _____
Dep/Ben Status . . . . . _____

Memo/Address Info. . . . . _____
                               _____
                               _____

F5=D/B Relationships           F21=Print           F24=More Keys
    
```

Spacing

Use the following as your standards when spacing different screen elements:

```

O . . . . Dependent / Beneficiary . . . . Plan D DB % S Effect.
P Number Name ID B Rel Ty Alloc. T From
- - - - -
- - - - -

```

- Separate column headings with one space.

```

Dependent or Beneficiary . -
Relationship . . . . . -
Dep/Ben Type . . . . . -
Percent Allocated. . . . . _____

```

- End row descriptions with at least one period followed by a single space before you begin associated input fields.

```

08335 Benefits by Employee Year . . . . 90
Type of Year C
Dates:
Employee . . . . 6001 Allen, Raymond Birth. . . . 10/20/58
Soc Sec No . 798-52-5841 Orig. Hire . 12/15/88
Benefit Grp. Started. . . 12/15/88
Business Unit 9 An Energy Deleted Interes Terminated .

O . . . . . Effective. . . Contributions .
P Plan Name From Through Employee Employer
- Dependent Care Reimb. Account
- Plan ID: DEPCARE Provider/Trustee: Edwards, J. D.
- Life Insurance 01/01/90 12/31/90
- Plan ID: LIFE Provider/Trustee: State Mutual Insurance Company

```

- Indent Fold Area fields one or more spaces to offset them from regular subfile.

```

O . . . . . Effective. . . Contributions .
P Plan Name From Through Employee Employer
- Dependent Care Reimb. Account
- Plan ID: DEPCARE Provider/Trustee: Edwards, J. D.
- Life Insurance 01/01/90 12/31/90
- Plan ID: LIFE Provider/Trustee: State Mutual Insurance Company

```

- 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.

```

Benefit Grp. Started. . .
Business Unit Terminated .

O . . . . . Effective. . . Contributions .
P Plan Name From Through Employee Employer

```

- Insert a blank line between header and subfile information.

Field	Explanation
Format Name	Screen record format. The format name will be the screen ID followed by a specific suffix value. Typically, the suffix values are: <ul style="list-style-type: none"> ▪ 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 database 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.
# Fields Selected	The number of database fields that have been selected for use on the format.
Fld Pfx	Screen field prefix to be used for the screen fields: VD, SF.

About Record Formats

Several Record Format Types are valid for screens. Currently, they include:

- SFLCTL - Subfile control

```

928200                               Item Search
Business Unit. BBBBBBBBBBB 00000000000000000000000000000000
O Item                               Ship
P Number Description Date Quantity On Hand UM

```

Present in all subfile screens. Contains all of the fields in the header or top portion of the screen, including the subfile column headings.

- SFL - Subfile

```

B 00000000 00000000000000000000000000000000 00000000 0000000000000000 00
Code 1 . 000 Code 2 . 000 Code 3 . 000 Code 4 . 000 Code 5 . 000
B 00000000 00000000000000000000000000000000 00000000 0000000000000000 00
Code 1 . 000 Code 2 . 000 Code 3 . 000 Code 4 . 000 Code 5 . 000
B 00000000 00000000000000000000000000000000 00000000 0000000000000000 00
Code 1 . 000 Code 2 . 000 Code 3 . 000 Code 4 . 000 Code 5 . 000
B 00000000 00000000000000000000000000000000 00000000 0000000000000000 00
Code 1 . 000 Code 2 . 000 Code 3 . 000 Code 4 . 000 Code 5 . 000
B 00000000 00000000000000000000000000000000 00000000 0000000000000000 00
Code 1 . 000 Code 2 . 000 Code 3 . 000 Code 4 . 000 Code 5 . 000
B 00000000 00000000000000000000000000000000 00000000 0000000000000000 00
Code 1 . 000 Code 2 . 000 Code 3 . 000 Code 4 . 000 Code 5 . 000

```

Contains all the fields in the subfile portion of the screen, including the fold area (if applicable).

- RECORD

Opt:1=Item Master Information	F5=Item Maintenance	F24=More Keys
-------------------------------	---------------------	---------------

Present in all screens. In subfile screens, contains VDL24 (line 24 text). In non-subfile screens, can contain all fields on the screen, including VDL24.

- SFLMSG - Subfile Message

Displays error message text. JD Edwards World 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 screen:

- With Fast Path
- With the File Selection List

Method	Description
Fast Path	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.
File Selection List	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 screen where you can specify multiple files and libraries from which to select database fields.

To select a database field using Fast Path

1. On the Record Formats List screen, enter a Fast Path File for the specified format.
2. For database field selection, choose option 1.

Function	Description
F6	This screen provides access to other repository services within JD Edwards World.
F8	Provides the following: <ul style="list-style-type: none"> ▪ Will display your screen in monochrome or color. ▪ If accessing the Field Definition window, will toggle between Condition Indicators and Color Attributes.
F10	Displays the Format Keyword Maintenance screen.

Example F10 - Format Keyword Maintenance screen

```

92537          Format Keyword Maint          Format: V928200C
Screen: V928200

      General Keywords
PUTOVR (Y/N) . . . . . N
OVERLAY (Y/N) . . . . . Y
-----
      Subfile Keywords
Subfile Fold . . . . . Y
Type (A/F) . . . . . F
Subfile Clear. . . . . Y
Subfile Next Change. . . . . Y
Subfile Page . . . . . 8
Subfile Size . . . . . 30
    
```

Field	Explanation
PUTOVR (Y/N)	The screen record format uses the PUTOVR keyword. Causes the screen to be erased and redisplayed when a window is displayed.
OVERLAY (Y/N)	The screen record format uses the OVERLAY keyword. Will not erase and redisplay screen when a window is displayed. Most JD Edwards World screens use OVERLAY.
Subfile Fold	Create a fold area in the subfile using SFLDROP and SFLFOLD keywords.
Type (A/F)	Further identifies subfile fold area: <ul style="list-style-type: none"> 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. <ul style="list-style-type: none"> Y means you want SFLCLR N will give you SFLINZ

Field	Explanation
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. <ul style="list-style-type: none"> 1 to 27 inclusive (Number of lines in SFL area divided by number of lines in fold.)
Subfile Size	Identifies the total number of records in the subfile that will be loaded in one program cycle. <ul style="list-style-type: none"> 1 to 9999 inclusive

Function	Description
F13	Displays the Function Key/Opt Definition screen. <ul style="list-style-type: none"> Used to define the function keys for the screen. Function Key Definition files (F9601 and F9611).

Example F13 Function Key/Opt Definition screen

9601	Function Key/Opt Definition		
Action Code. . . I	Video Screen . . .	V928200	
	Video Title. . . .	Item Search	
Line 24	Opt:l=Item Master Information F5=Item Maintenance F24=More Keys		
<u>Include</u>	<u>Description</u>	<u>Key/Opt</u>	<u>Field</u>
Y	Exit Program	03	#FEOJ
Y	Clear Screen	22	#FCLR
Y	Help Instructions	HL	#FHELP
Y	Roll Up/Next Record	RU	#FROLU
Y	Roll Down/Previous Record	RD	#FROLD
Y	Field Sensitive Help	01	#FQMRK
Y	Display Error Message(s)	07	#FERRD
Y	Display All Function Keys	24	#FKEYS
Y	Item Maintenance	05	#F01
Y	Item Master Information	01	#S01
Include: Y/N		F16=Display All	

Field	Explanation
Action Code	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.
Screen Screen	The name of the screen or report record to be copied. All records for soft coding will be displayed.

Function	Description
F16	<p>Displays the List of Defined Fields screen.</p> <ul style="list-style-type: none"> Used to maintain the defined fields and add hidden fields. Only shows fields for the formats that are active.

Example F16 - List of Defined Fields screen.

```

92540                               List of Defined Fields
Screen: V92801

Opt  Fmt/Field  Description                               Row/Col Typ  Size  Use
--  -
VTX007          Item                               006 013  A   30   O
VTX009          Quantity                               006 044  A   21   O
VTX011          Ship                               006 066  A    8   O
VTX003          P                               007 002  A    1   O
VTX006          Number                               007 004  A    8   O
VTX008          Description                               007 013  A   30   O
VTX010          On Hand                               007 044  A   21   O
VTX012          Date                               007 066  A    8   O
V92801S        Record Format                               SFL
SHXIT          Item ID - Hidden Field                   000 000  A    8   H
SFSELC        Selection Exits                           008 002  A    1   B
SFXIT         Item ID. . . . .                       008 004  A    8   B
SFXDS         Description . . . . .                   008 013  A   30   B
SFXQT         Quantity On Hand . . . . .             008 044  A   21   B

Opt: 4=Delete    5=Display/Update    F3=Exit    F12=Prev Screen

```

Hidden Fields

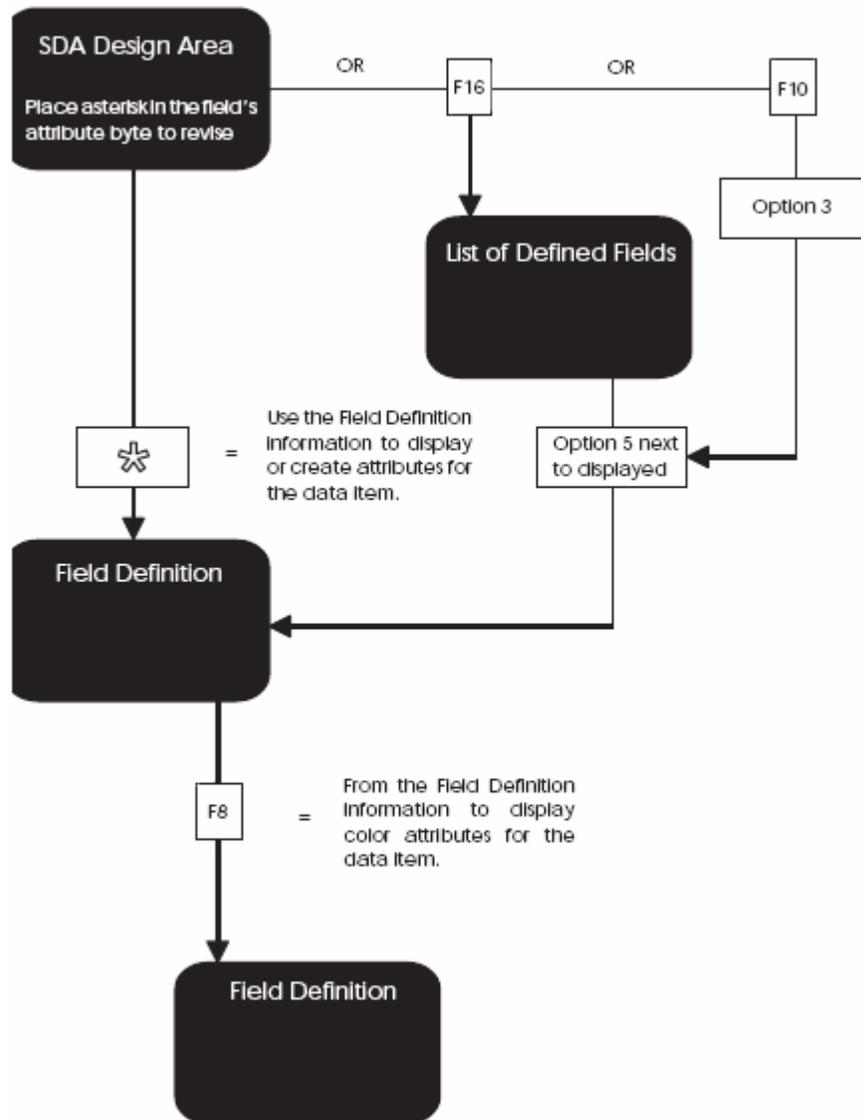
Used to store hidden field information.

To add a hidden field to a screen

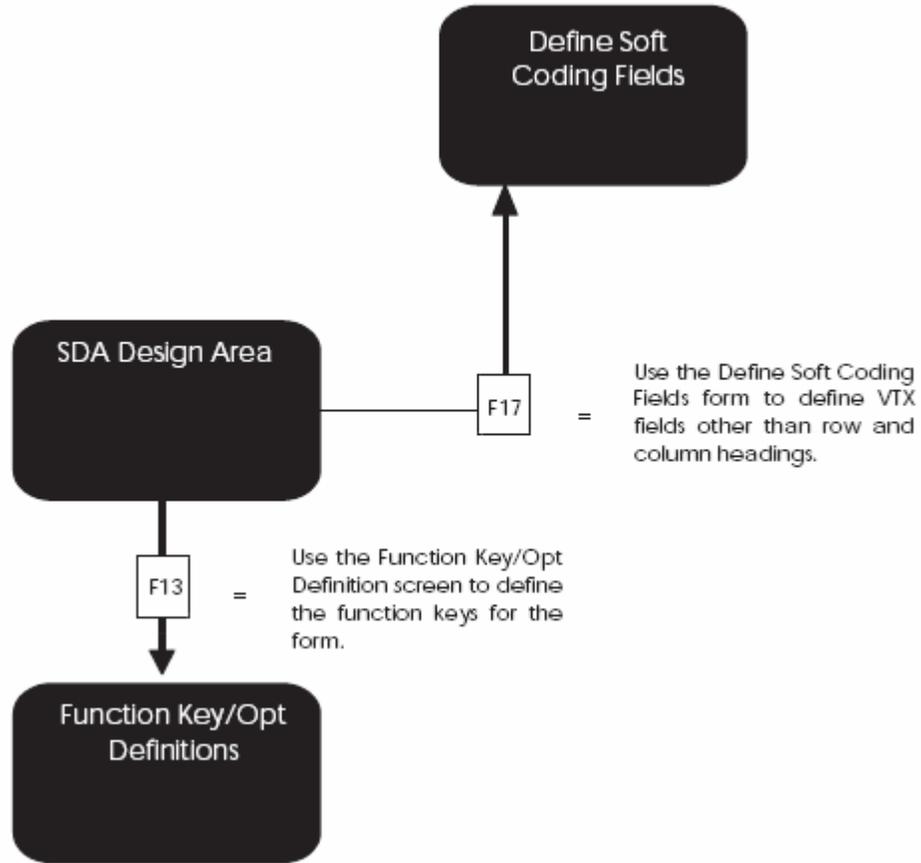
- Roll to the bottom blank line of the format that contains the field.
- Choose option 5, Display/Update.
- Enter the field with a prefix of SH, description, type, size, and press Enter.

This information should be the same as the associated database field.

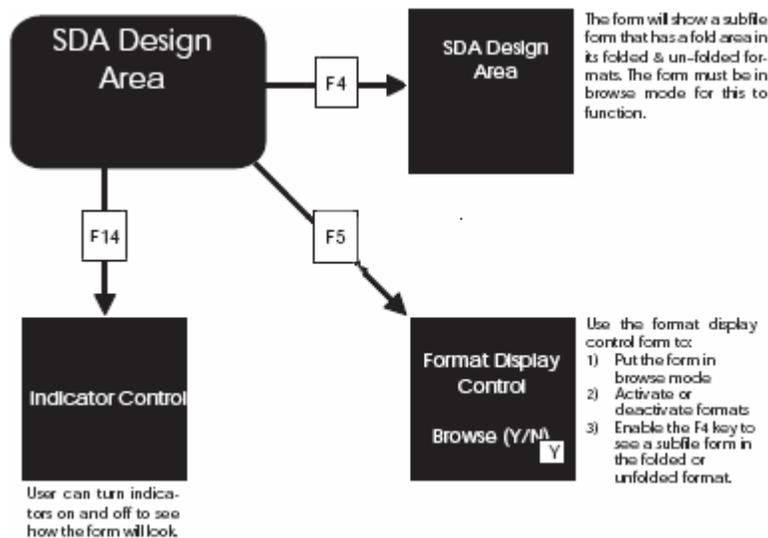
Process Overview - Revising the Field Definition



Process Overview - Revising Vocabulary and Function Keys



Function Keys for Screen and Display Format Control



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 screen if you accidentally press Field Exit.

Standard prefixes:

- VD, SF, SH

Special Fields:

- *VTX, *VC0, *LITER, *DATE, *TIME
- 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 screen:

- * (non-pick list method)
- & (pick list method)

You can pull in VTX, VC, and the screen 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 screen

If you are changing subfile boundaries, you should use the outlined processes to make this process easier.

You must save a screen 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 screen, 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.

Work with Report Design Aid

About Report Design Aid

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.

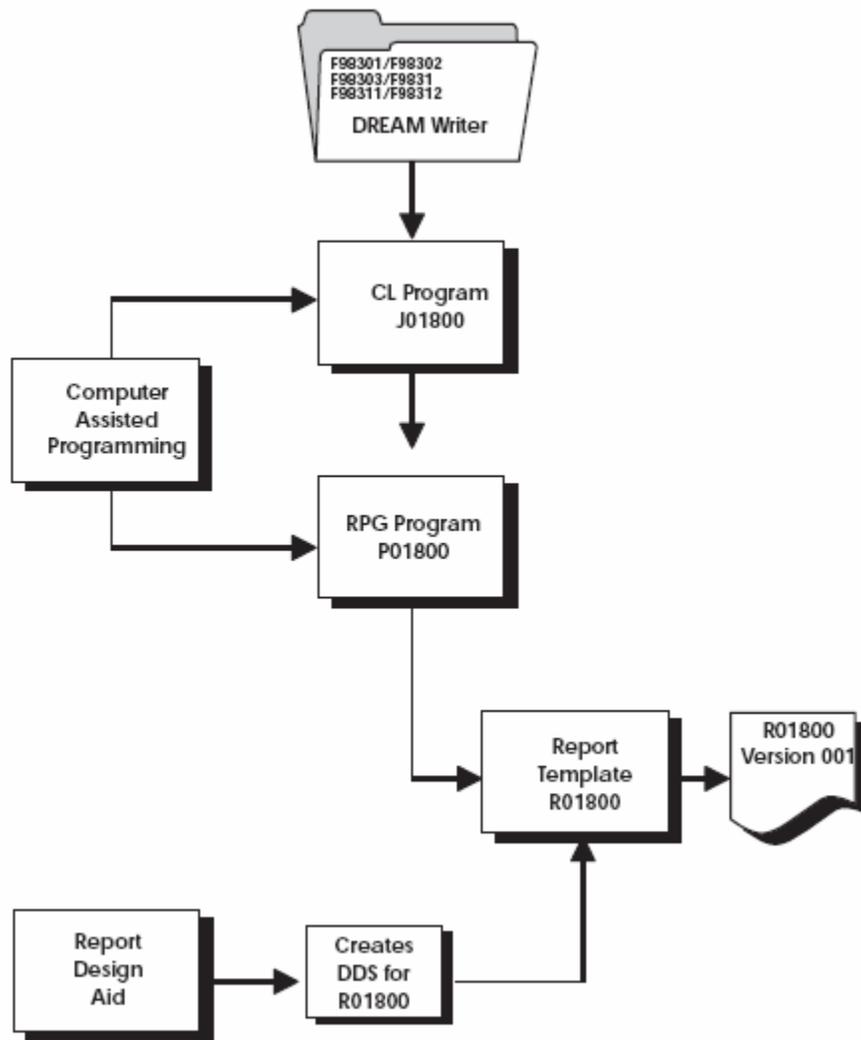
JD Edwards World reports are externally defined, which means that all the Data Description Specifications are created and compiled as a printer file, separate from the program object. RDA automatically generates the DDS. It also incorporates the report information into the documentation and adds it to 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.

This section includes the following tasks:

- [Accessing Report Design Aid](#)
- [Updating a Field in RDA](#)
- [Compiling A Report](#)
- [Changing the Compile Option Defaults for Reports](#)

Example - RDA and DREAM Writer



Comparing RDA and SDA - Field Definition Form

```

Screen: V5501Z-----Field Definition-----Format: V5501ZS
Dict Name  SXDS      Text      Description
Data Type  A          Field Name SF$XDS
Row/Column 8 13      Field Use  B          RI  Y 44
Size       12          Text Form  -          HI  Y 44
Dft Cursor -          Edited    Y 44      UL  Y N44
Lower Case Y          Change    -          ND  -
OVRDTA    -          Duplicate -          BL  -
OVRATR    -          Field Cond -        PR  -
                                           PC  -
--F3=Exit  F12=Prev Screen  F17=Dictionary--
  
```

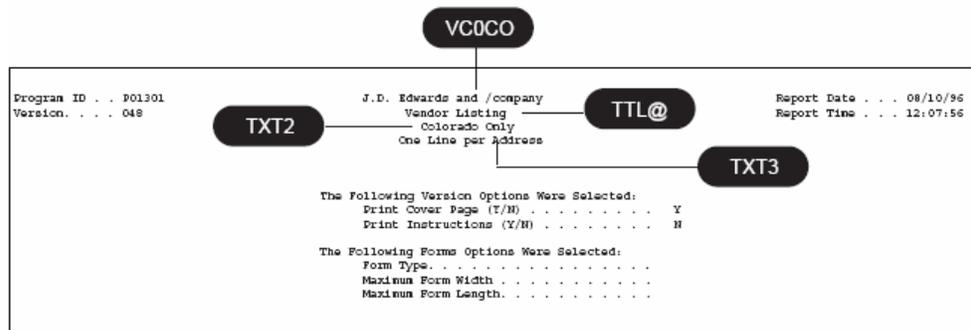
```

Report:R55400X-----Field-Definition-----Format:-DETAIL1--
Dict Name  SXDS      Text      Description
Data Type  A          Field Name RR$XDS
Row/Column 9 89      Field Use  0          Highlight -
Size       12          Text Form  -          Underline -
                                           Field Cond -
Space Before -          Lines  Cond Ind -        Char per Inch -
Space After -          -          -          Edit Code -
Skip Before -          -          -          Asterisk Fill -
Skip After  -          -          -          Float Symbol -
--F3=Exit  F12=Prev-Screen  F17=Dictionary--
  
```

Item	FIELD POSITIONING	FIELD CONDITIONING
RDA	Row positions are relative to the other rows, not fixed. The location on the report is determined by Space and Skip designations. Column positions are fixed.	A field can optionally appear bold, underlined, and so forth. JD Edwards World does not typically use these features because they impact printer performance.
SDA	Both row and column positions are fixed. A field displays on the screen exactly where the Row and Column indicators specify.	A field can appear highlighted, underlined, in reverse image, and so forth JD Edwards World 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.

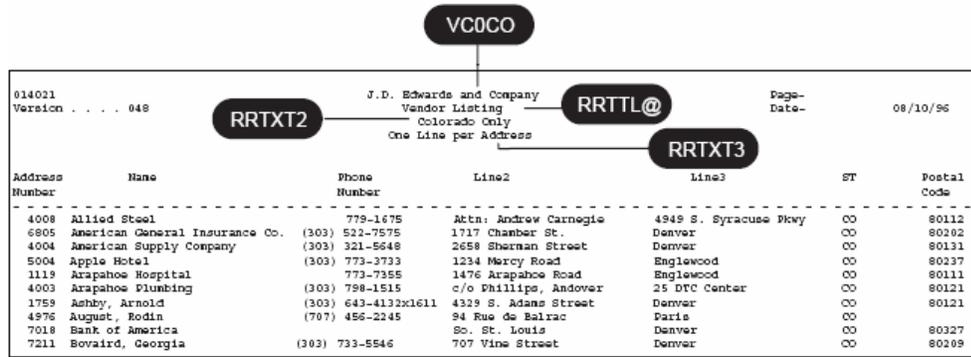


The following table provides the field names and a description of each.

Field	Explanation
VC0CO	Name of company 00000
TTL@	Line 1 of DREAM Writer Version ID if it exists, otherwise it is blank
TXT2	Line 2 of DREAM Writer Version ID, or blank
TXT3	Line 3 of DREAM Writer Version ID, or blank

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.

Field	Explanation
VCOCO	Name of company 00000
RRTTL@	Default Title from Vocabulary Override
RRTXT2	Line 2 of DREAM Writer Version ID, or blank
RRTXT3	Line 3 of DREAM Writer Version ID, or blank

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.

FORMAT	FIELD	DESCRIPTION
Any format	*VTX	Assigns the first available VTX name to the field and gets a description from the Data Dictionary that you can change.
	*VC0	Assigns the first available VC0 field and assigns a default size of thirty.

FORMAT	FIELD	DESCRIPTION
HEADING1 - contains the standard fields to be printed on the top of every page	VTX001	The default VTX field which prints the row description, "Page -."
	*PAGE	The default special field that inserts the DDS keyword PAGNBR in the source and retrieves the current page number on the report.
	VTX002	The default VTX field which prints the row description, "Date -".
	*DATE	Special field that retrieves today's date.
	VC0CO	The name of the default company 000, it displays on the first line of each page.
	RRTTL@	Default Title from Vocabulary Overrides.
HEADING2 - contains the subheading fields used to describe the level break detail that is to follow	RRTXT2 & RRTXT3	DREAM Writer overrides that correspond to the second and third header lines of the report.
	VC0ROW	Data Dictionary row description of the level break field.
	VC0KEY	The value of the level break field.
DETAIL1 - contains the data line fields	VC0DSC	The description of the value of the level break field.
	RRxxxx	The value of the data for these fields
TOTAL1 - contains the total line fields	VC1ROW	Data Dictionary row description of the level break field.
	VC1KEY	The value of the level break field
	VC1DSC	The description of the level break field
	\$\$XXX	Value on total line.

Note: 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.

	VC0KEY	VC0DSC	
	Item Description	Quantity	HEADING1
VC0ROW RR Fields	Business Unit . . .	4	Denver
	1 Bolt	300	
	2 Nut	400	
	3 Nail	150	
\$\$ Fields		850	
VC1ROW	Business Unit . . .	4	Denver
	Business Unit . . .	9	Boulder
			TOTAL1
	VC1KEY	VC1DSC	

What Are the Report Design Standards?

The following is a list of report design standards. Using these standards will give your reports a uniform appearance.

RDA Features

Some 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

JD Edwards World 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

Some of the following recommendations can be ignored since most printers in use are not impact printers.

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 statements 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.

Field	Explanation
Boundaries	Two 3-digit numbers that define the range (rows) for the DDS. <ul style="list-style-type: none"> ▪ HEADING1 is usually rows 1 to 8 ▪ DETAIL1 is row 9 (Only one detail line is defined.) ▪ TOTAL1 is rows 10 to 11 (Leave one line for the dashes above the Total field.)
Window	Allows you to access fields outside the boundaries.
Browse (Y/N)	Indicator that allows you to enable/disable the browse mode.
Form Width	Width of the form in print positions.

Caution: RDA might automatically adjust displayed formats with those formats that are not displayed.

Function Key	Description
F6	Shows the Repository Services portion of a form.

Example F6 - the Repository Services portion

```

928400                                00000000000000000000000000000000
                                      00000000000000000000000000000000
                                      00000000000000000000000000000000
                                      00000000000000000000000000000000

Business                               It
Unit                                  Ty   Description
-----
000000000000 00000000000000000000 985001----Repository-Services-----
"1" Available Services
- Data Dictionary
- Menus
- Vocabulary Overrides
- Function Key Definitions
- Processing Options
- User Defined Codes
- Edit System Helps
- CASE Profiles
- SAR Log Inquiry
- Copy DD,VO,DW,UDC,SI,Menus
-Sel:--"1"=Select----F12=Previous-----

```

Function Key	Description
F10	Displays the Record Formats List form.

1. Enter 14 next to the member in the subfile that you want to create and press Enter.

A form of printer file parameters displays.

Printer File Parameters	
Member ID.	R928400
Forms Length	068
Forms Width.	132
Lines/Inch (4/6/8/9) .	8
Char./Inch (10/15) . .	15
Overflow Line.	062
Align Forms.	N
Form Type.	*STD
Copies	001
Separator Pages. . . .	1

2. Do one of the following:
 - Accept the defaults.
 - Change the defaults, as necessary.

Note: Make sure Copies is non-zero.

Changing the Compile Option Defaults for Reports

You must compile reports through the JD Edwards World compiler by this method so that R98COVER and R98RPTH are pulled in for the cover page and help instructions when the Function Code is PRTF. Only one PRTF can be included in an RPG program. Second print files or Special form print files must have a Function Code of PRTS in SVR. Compiling PRTF items through the Production Development Manager (PDM) or some other method will not bring the additional formats in automatically.

To change compile option defaults for reports

The Data Dictionary default values were set for 8 ½ by 14 printer paper.

Change the Data Dictionary defaults for the following data items for your purposes:

Item	Description
#FLN	Forms Length
WDTH	Forms Width
LPI	Lines Per Inch
#CPI	Characters Per Inch
#OVF	Overflow Line Number (Usually forms length minus one inch.)

Item	Description
#ALN	Alignment (Y/N)
#FTY	Form Type
#CPY	Number of Copies
#SPG	Number of Separator Pages

Note: 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.

5 Programming Standards

Overview to Programming Standards

Programming Standards

The Program Generator serves as the primary enforcer of JD Edwards World 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
- JD Edwards World Source Debugger

Program Specifications

About Program Specifications

There are several kinds of RPG/400 specifications as described in IBM's *Languages: RPG/400 User's Guide*. When your source program is compiled, these specifications are arranged in the following sequence:

- [What Are Header \(Control\) Specifications?](#)
- [What Are File Description Specifications?](#)
- [What Are Extension Specifications?](#)
- [What Are Input Specifications?](#)
- [What Are Calculation Specifications?](#)
- [What Are Output Specifications?](#)

An RPG/400 program does not have to use all specifications. A typical JD Edwards World program contains Header, File Description, Extension, Input, and Calculation specifications. Few programs have Output specifications.

Note: There are a few programs which have been converted to, or written in, RPGIV (ILE). RPGIV has been available since 1994. RPGIV programs have Data specification entries and no Extension specification entries. Consult IBM reference manuals for details.

What Are Header (Control) Specifications?

The Header (control) specifications (H specs) 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 JD Edwards World programs for copyright purposes and reproduction restrictions.

```

Columns . . . :   1 71          Browse          DEVSRC/JDESRC

SEU==>> _____ P55011X
FMT **  ...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7
***** Beginning of data *****
0001.00 H/TITLE P55011X - Item Information Update
0002.00 H*
0003.00 H*
0004.00 H* Copyright (c) 1993
0005.00 H* J. D. Edwards & Company
0006.00 H*
0007.00 H* This unpublished material is proprietary to
0008.00 H* J. D. Edwards & Company. All rights reserved.
0009.00 H* The methods and techniques described herein are
0010.00 H* considered trade secrets and/or confidential.
0011.00 H* Reproduction or distribution, in whole or in part,
0012.00 H* is forbidden except by express written permission
0013.00 H* of J. D. Edwards & Company.
0014.00 H*
0015.00 H*
0016.00 F*

F3=Exit F5=Refresh F9=Retrieve F10=Cursor F12=Cancel
F16=Repeat find F24=More keys

```

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 if the file has an external definition
- Whether or not the file is keyed
- Input or output device used for the file
- If the file will have records added to it

```

Columns . . . :   1 71          Browse          DEVSRC/JDESRC

SEU==>> _____ P55011X
FMT **  ...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7
0027.00 F*
0028.00 FP0001 IP E K DISK
0029.00 FP5501X UF E K DISK A
0030.00 FP55011X CP E WORKSTN KINPDS SBVFDS
0031.00 F*****
0032.00 F*
0033.00 F* Copy Member for Composite Common Subroutine - C0001
0034.00 F*
0035.00 F/COPY JDECPY,D0001
0036.00 F*****
0037.00 E*****
0038.00 E* PROGRAM TABLES AND ARRAYS
0039.00 E*
0040.00 E*

F3=Exit F5=Refresh F9=Retrieve F10=Cursor F12=Cancel
F16=Repeat find F24=More keys

```

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. Some general rules, in regard to files, are the following:

- 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 copy names 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

```

Columns . . . : 1 71          Browse          DEVSRC/JDESRC
SEU==>>      . . .+ . 1 . . .+ . 2 . . .+ . 3 . . .+ . 4 . . .+ . 5 . . .+ . 6 . . .+ . 7
FMT **
0040.00      E*****
0041.00      E*   PROGRAM TABLES AND ARRAYS
0042.00      E*   -----
0043.00      E*
0044.00      E           EMK           64 4           Error Msg
0045.00      E           @MK           64 1           Error Msg
0046.00      E           @ER           64 4           Error Msg
0047.00      E           @DV           40 1           Dflt Wrk
0048.00      E*
0049.00      E*
0050.00      E*   Copy Member for Composite Common Subroutine - C0001
0051.00      E*
0052.00      E/COPY JDECPY,E0001
0053.00      E*****

F3=Exit  F5=Refresh  F9=Retrieve  F10=Cursor  F12=Cancel
F16=Repeat find  F24=More keys

```

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

```

Columns . . . : 1 71          Browse          DEVSRC/JDESRC
SEU==>>
FMT **      ...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7
0067.00      I*****
0068.00      I*  PROGRAM INPUT SPECIFICATIONS AND DATA STRUCTURES
0069.00      I*  -----
0070.00      I*
0071.00      I*  Data Structure to Load Video Screen Text
0072.00      I*
0073.00      IDSTXT      DS          520
0074.00      I          1 18 VTX001
0075.00      I          41 58 VTX002
0076.00      I          81 92 VTX003
0077.00      I          121 138 VTX004
0078.00      I          161 178 VTX005
0079.00      I          201 218 VTX006
0080.00      I          241 258 VTX007
0081.00      I          281 298 VTX008
0082.00      I          321 338 VTX009
0083.00      I          361 378 VTX010

F3=Exit  F5=Refresh  F9=Retrieve  F10=Cursor  F12=Cancel
F16=Repeat find  F24=More keys
There are no commands to retrieve.
    
```

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.
- In JD Edwards World RPGIV (RPGL) programs, there may be D specs and no I or E specs.

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 JD Edwards World 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

```

Columns . . . : 1 71          Browse          DEVSRC/JDESRC
SEU=>>>          P55011X
FMT **  ...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7
0098.00 C*      MAINLINE PROGRAM
0099.00 C*      -----
0100.00 C*
0101.00 C*      Process housekeeping.
0102.00 C*
0103.00 C*      EXSR S999
0104.00 C*
0105.00 C*
0106.00 C*      If LR on, end program.
0107.00 C*
0108.00 C*      *INLR CABEQ'1'   EQI
0109.00 C*      -----
0110.00 C*
0111.00 C*
0112.00 C*      If automatic inquiry set, process inquiry.
0113.00 C*
0114.00 C*      $AUTO CASEQ'1'   S003   24
                                -----
F3=Exit  F5=Refresh  F9=Retrieve  F10=Cursor  F12=Cancel
F16=Repeat find  F24=More keys

```

The C Specs are the heart of the processing of a program. JD Edwards World 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

```

Columns . . . : 1 71          Browse          DEVSRC/JDESRC
SEU=>>>          P55011X
FMT **  ...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7
2334.00 CSR      MOVE '0026'   EMK,08   Inv MCU
2335.00 CSR      MOVE '0027'   EMK,09   Inv Desc Ttl
2336.00 C*      -----
2337.00 C*
2338.00 C*      Load invalid action code array.
2339.00 C*
2340.00 CSR      MOVEA'   '   @NAC
2341.00 C*      -----
2342.00 C*
2343.00 C*      Load system date.
2344.00 C*
2345.00 CSR      TIME           $WRK12 120
2346.00 CSR      MOVE $WRK12   $$EDT   60
2347.00 C*      -----
2348.00 CSR      END999      ENDSR
2349.00 C*      *****
2350.00 OI5501X  E           UNLOCK
F3=Exit  F5=Refresh  F9=Retrieve  F10=Cursor  F12=Cancel
F16=Repeat find  F24=More keys

```

JD Edwards World uses the RPG EXCPT operation to release locks on data records. The 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.

Note: Typically, JD Edwards World does not perform reporting functions using O Specs.

You can use the Opcode "UNLCK" instead of EXCPT/O-SPECS.

6 Program Overview

Program Overview

About the Program Overview

The program overview provides a basic overview of the standards used in a program. It includes the following:

- [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              END001      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.

```

1874.00      C*-----
1875.00      CSR              END001      ENDSR
1876.00      C*****

```

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. JD Edwards World stores all common subroutines in the file JDECPY.

At compile time, the compiler copies in source code for all instances of the COPY statement. The included code appears only once and 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.

```

0731.00      C*****
0732.00      C*
0733.00      C*      Copy Common Subroutine - Right Justify Numeric Fields
0734.00      C*
0735.00      C/COPY JDECPY,C0012
0736.00      C*****

```

This example shows how the COPY statement in the source (above) brings in additional code to the compiled source.

```

73400 C*
73500 C/COPY JDECPY,C0012
Q000000+ MEMBER C0012 IN FILE JDECPY LIBRARY JDFSRC OPENED FOR /COPY.
Q000100+ C*****
Q000200+ C* This is part of a composite common subroutine. In
Q000300+ C* order for the subroutine to work correctly, the
Q000400+ C* RPG program must /COPY in the following members:
Q000500+ C* E0012, C0012
Q000600+ C*****
Q000700+ C* MAINLINE PROGRAM
Q000800+ C* -----
Q000900+ C*
Q001000+ C*
Q001100+ C* SUBROUTINE C0012 - Right Justify Numeric Fields
Q001200+ C* -----
Q001300+ C*
Q001400+ C* PURPOSE
Q001500+ C* -----
Q001600+ C* To provide a subroutine common to all programs which
Q001700+ C* right justifies numeric fields and places the sign over
Q001800+ C* the low order byte of the fields, designated by either a
Q001900+ C* leading or trailing minus sign. This routine also ignores
Q002000+ C* all non-numeric characters in the input field, and
Q002100+ C* determines the placement of the decimal point.
Q002200+ C*
Q002300+ C* REMARKS
Q002400+ C* -----
Q002500+ C*
Q002600+ C* Prior to executing this subroutine data from an
Q002700+ C* alphanumeric input field should be placed in the array
Q002800+ C* named 'ANM' with a 'MOVEA' command. The right justified
Q002900+ C* number is available from the subroutine field named
Q003000+ C* '#NUMR', which is a 15 digit 6 decimal field.
Q003100+ C* CAUTION: The largest number that can be handled
Q003200+ C* by this subroutine is 999,999,999.999999.
Q003300+ C* However, the input field may contain only 15
Q003400+ C* numbers.
Q003500+ C*
Q003600+ CSR C0012 BRGSR
Q003700+ C* -----
Q003800+ C*
Q003900+ CSR Z-ADD0 #NUMR 299
Q004000+ CSR Z-ADD0 #NUMR2 152 Compile only
Q004100+ CSR Z-ADD0 #NUMR9 159 Compile only
Q004200+ C*
Q004300+ CSR MOVEA@NM #ALNUM
Q004400+ CSR CABEQ*BLANKS EN0012
Q004500+ CSR MOVE *ALL'0' #ALNUM 22

```

The following user defined code contains an online listing and specifications:

- Install System Code: 93
- User Defined Code: /C

Error Handling

JD Edwards World has devised an efficient means of handling errors by way of arrays.

```

Columns . . . : 1 71 Browse DEVSRC/JDESRC
SEU=>>> ----- P55011X
0040.00 E*****
0041.00 E* PROGRAM TABLES AND ARRAYS
0042.00 E*
0043.00 E*
0044.00 E EMK 64 4 Error Msg
0045.00 E @MK 64 1 Error Msg
0046.00 E @RR 64 4 Error Msg
0047.00 E @DV 40 1 Dflt Wrk
0048.00 E @AV 10 10 Allowed Values
0049.00 E @40 40 1 Allowed Values
0050.00 E @10 10 1 Allowed Values
0051.00 E*
0052.00 E*
0053.00 E* Copy Member for Composite Common Subroutine - C0001
0054.00 E*
0055.00 E/COPY JDECPY,E0001
0056.00 E*****
0057.00 E*
0058.00 E* Copy Member for Composite Common Subroutine - C0012
0059.00 E*
0060.00 E/COPY JDECPY,E0012

```

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.

```

Columns . . . : 1 71          Browse          DEVSRC/JDESRC
SEU==>>          P55011X
0278.00          C*
0279.00          C*          If Display errors pressed, exit to error messages.
0280.00          C*          -----
0281.00          C*
0282.00          CSR          @@AID          IFEQ #FERRD
0283.00          CSR          Z-ADD1          #G
0284.00          CSR          Z-ADD1          #H
0285.00          CSR          #G          DOWLB64
0286.00          CSR          @MK, #G          IFEQ '1'
0287.00          CSR          MOVE EMK, #G          @ER, #H
0288.00          CSR          ADD 1          #H
0289.00          CSR          END
0290.00          CSR          ADD 1          #G
0291.00          CSR          END
0292.00          CSR          CALL' P0000E'          98
0293.00          C*          -----
0294.00          CSR          PARM          @ER
0295.00          CSR          GOTO ENDEXE
0296.00          C*
0297.00          CSR          END
0298.00          C*

```

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.

```

Columns . . . : 1 71          Browse          DEVSRC/JDESRC
SEU==>>          P55011X
0347.00          C*
0348.00          C*          If error on read, set error.
0349.00          C*
0350.00          CSR          *IN82          IFEQ '1'
0351.00          CSR          SETON          9341
0352.00          CSR          MOVE '1'          @MK, 2
0353.00          CSR          GOTO ENDEXE
0354.00          C*          -----
0355.00          CSR          END
0356.00          CSR          END
0357.00          CSR          END
0358.00          C*
0359.00          C*          If ROLL DOWN key pressed, process read prior.
0360.00          C*          -----
0361.00          C*
0362.00          CSR          @@AID          IFEQ #PROLD
0363.00          C*
0364.00          C*          Reset error indicators if roll
0365.00          C*
0366.00          CSR          MOVE$RESET          *IN, 41
0367.00          CSR          MOVE '0'          *IN, 40

```

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.

Indicator	Explanation
01	Causes the Invalid Function Key Pressed message to appear
02	Dictates the color palette to be used

Indicator	Explanation
15	Indicates a function key was pressed
20	Handles the clear screen action code
21	Handles the add action code
22	Handles the change action code
23	Handles the delete action code
24	Handles the inquire action code
25	Handles the inquire action code 'P' for print (payroll)
31	Used in conjunction with subfile processing to initiate the INVITE or SFLCLR keyword. Using INVITE will slow processing
32	Used in conjunction with subfile processing initiating the keyword SFLNXTCHG
37	Used in conjunction with subfile processing to avoid display of an empty subfile (used only with inquiry subfiles)
38	Used in conjunction with subfile processing to highlight the last record in the display (keyword SFLDSP) and avoid display of an empty subfile
40-79	Used for error processing to indicate which fields are in error and need to be highlighted
40	Reserved for errors in the Action Code field
41	Reserved for errors in the key fields
80-89	General reusable one-time indicators. Use them as needed
93	Global error indicator that highlights line 24
98	Indicates a chain or read failure
99	Indicates a record is in use or file error
OF	Indicates overflow for report processing
LR	Indicates that the last record has been read and the program should end normally
RT	Indicates that a temporary or final halt in the program should take place. Returns to calling program leaving files open

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.

```

0016.00 F*
0017.00 F* PROGRAM REVISION LOG
0018.00 F* -----
0019.00 F*
0020.00 F* Date Programmer Nature of Revision
0021.00 F* -----
0022.00 AUTHRF* 03/18/93 MARTIN SAR # 00000005 (AS/400 A/G)
0023.00 F* 05/01/93 RIPPEY SAR # 00167542
    
```

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.

```

0034.00 F*****
0035.00 F*
0036.00 F* Copy Member for Composite Common Subroutine - C0001
0037.00 F*
    
```

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*
0131.00 C $998 CASEQ' ' S998
0132.00 C* -----
0133.00 C END
    
```

DON'T WRITE: If \$998 is blank, execute S998.

INSTEAD WRITE: Load data field dictionary parameters (one cycle only).

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*
0276.00 C EXSR S999
0277.00 C* -----
    
```

Miscellaneous Items

The following represent miscellaneous items 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.

If the work field needs to have the same attributes as a field that exists in a file, for example:

```
MOVE ABANS $$ANS
```

Then define \$\$ANS:

```
*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*
0832.00 C*   If F6 pressed, Global Update by Percent or Amount.
0833.00 C*
0834.00 C*
0835.00 CSR   @@AID   IPEQ #F03
0836.00 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
FF08501LAIF E      K      DISK      UC

```

The lines that perform the open are shown below.

```

Columns . . . : 1 71      Browse      JDFSRC/JDESRC
SEU==>>>
3825.00 C*
3826.00 C*   Check for existence of pension files.
3827.00 C*
3828.00 CSR
3829.00 CSR   *IN99   OPEN F085201      99
3830.00 CSR           IFBQ '0'
3831.00 CSR           MOVE '1'   $PENS 1
3832.00 CSR           END
3833.00 C*
3834.00 CSR   *IN99   OPEN F08501LA      99
3835.00 CSR           IFBQ '0'
3836.00 CSR           MOVE '1'   $PENS2 1

```

If your program performs 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 has not purchased that system. The example below illustrates the necessary pre-compiler commands designed to address this 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.

```

***** Beginning 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

The JD Edwards World program generator produces several types of 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 JD Edwards World Programs

Standard subfile names make program maintenance easier. The system calls these subfiles primarily from the Mainline.

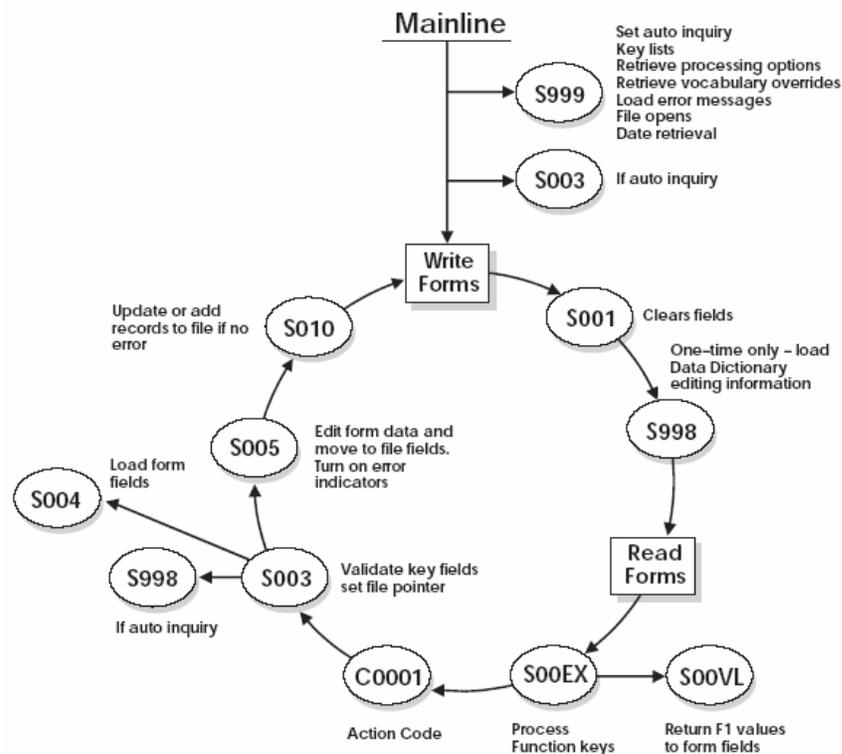
JD Edwards World program structure uses several types of subroutines, including:

Name	Description
S00EX	Processes all function key exits. <ul style="list-style-type: none">▪ Calls J96012 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. <ul style="list-style-type: none">▪ Usually only clears key fields and VC0 fields if F22 (Clear) is pressed

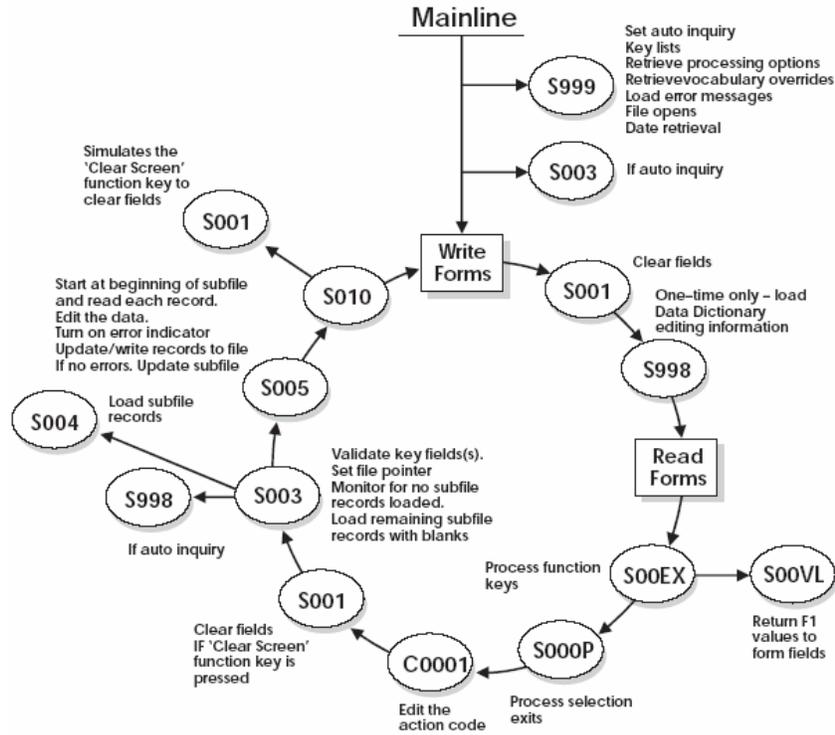
Name	Description
S002	Checks for level breaks for reports. <ul style="list-style-type: none">▪ Turns on level break flags.▪ Retrieves total line description
S003	Validates the key fields. Calls S998 subroutine if auto inquire was invoked Sets the file pointer. <ul style="list-style-type: none">▪ Performs 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 Loads unused subfile records with blanks
S004	Display or load form or report fields.
S005	Scrubs and edits form fields. <ul style="list-style-type: none">▪ Moves screen 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: <ul style="list-style-type: none">▪ 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 <i>not</i> a report, it updates, adds, or deletes records from the database file <ul style="list-style-type: none">▪ 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

Name	Description
S999	Housekeeping. (One time only) <ul style="list-style-type: none"> ▪ 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

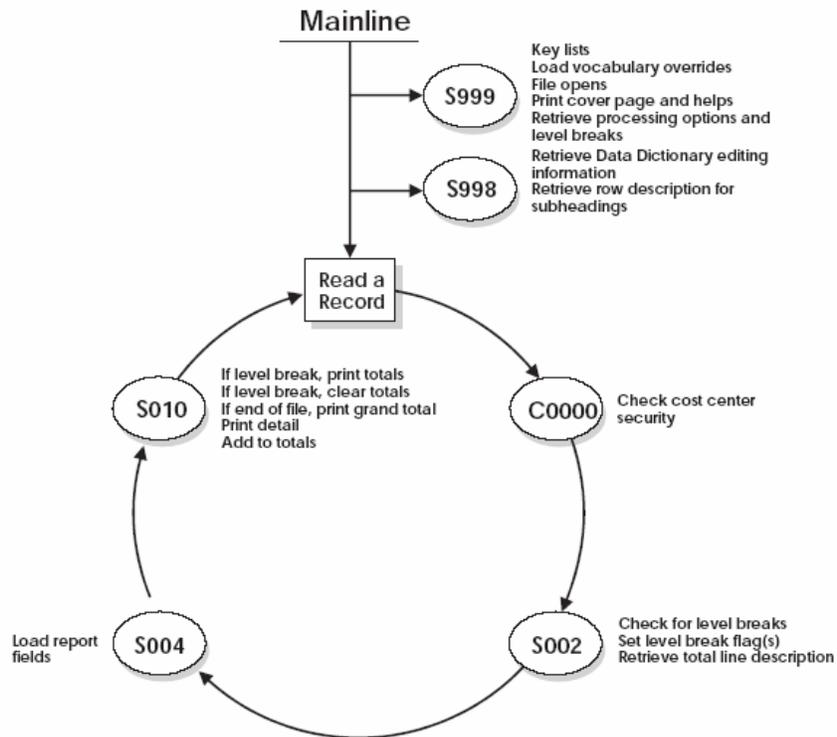
Interactive Non-Subfile Program



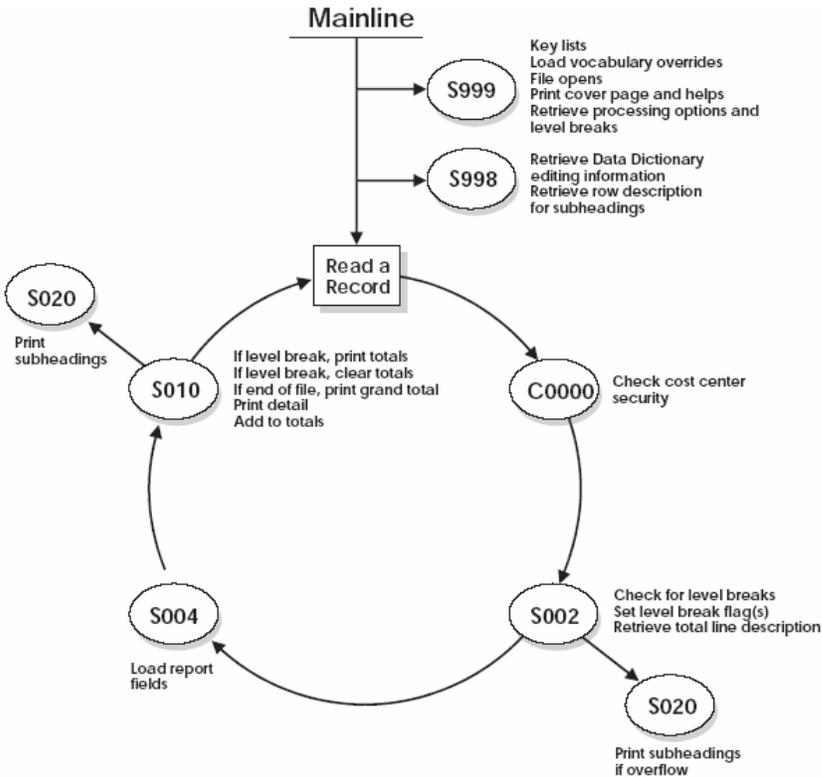
Subfile Program with Selection Exits



Report Program without Subheadings



Report Program with 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.

```

1.00 H/TITLE P928011 Item Master Information
2.00 H* -----
3.00 H*
4.00 H* Copyright (c) 1993
5.00 H* J. D. Edwards & Company
6.00 H*
7.00 H* This unpublished material is proprietary to
8.00 H* J. D. Edwards & Company. All rights reserved.
9.00 H* The methods and techniques described herein are
10.00 H* considered trade secrets and/or confidential.
11.00 H* Reproduction or distribution, in whole or in part,
12.00 H* is forbidden except by express written permission
13.00 H* of J. D. Edwards & Company.
14.00 H*
15.00 H* -----
16.00 F*
17.00 F* PROGRAM REVISION LOG
18.00 F* -----
19.00 F*
20.00 F* Date Programmer Nature of Revision
21.00 F* 12/07/93 QUARLES SAR # 241883 (AS/400 A/G)
22.00 AUTHRP*
23.00 F*
24.00 F* B0010 - Standard Maintenance Program Type
25.00 F* This program provides the standard single cycle
26.00 F* processing for adding, changing, deleting and
27.00 F* inquiring into data records as requested.
28.00 F*
29.00 F*
30.00 F*
31.00 F* FF0001 IF E K DISK
32.00 F* FF92801 UF E K DISK
33.00 F* FV928011 CF E WORKSTN KINFDS SKVFDSD A
34.00 F*
35.00 F*
36.00 F* Copy Member for Composite Common Subroutine - C0001
37.00 F*
38.00 F* F/COPY JDECPY_D0001
39.00 F*
40.00 E*
41.00 E* PROGRAM TABLES AND ARRAYS
42.00 E*
43.00 E*
44.00 E* @MK 64 4 Error Msg
45.00 E* @MK 64 1 Error Msg
46.00 E* @ER 64 4 Error Msg
47.00 E* @DV 40 1 Dflt Wrk
48.00 E* @C 256 1 Literal Work
49.00 E*
50.00 E*
51.00 E* Copy Member for Composite Common Subroutine - C0001
52.00 E*
53.00 E* E/COPY JDECPY_E0001
54.00 E*
55.00 E*
56.00 E* Copy Member for Composite Common Subroutine C0012
57.00 E*
58.00 E* E/COPY JDECPY_E0012
59.00 E*
60.00 E*
61.00 E* Copy Member for Composite Common Subroutine - C997
62.00 E*
63.00 E* E/COPY JDECPY_E997
64.00 E*
65.00 I*
66.00 I* PROGRAM INPUT SPECIFICATIONS AND DATA STRUCTURES
67.00 I* -----
68.00 I*

```

Copyright statement can be changed through the Program Generator

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

Arrays that handle error messages

Will copy in additional specifications for copy module C0001

```

69.00 I*      Data Structure to Load Video Screen Text
70.00 I*
71.00 I*      IDGNTX      Dg              1000
72.00 I          I          1  19 VTX001
73.00 I          I          41  59 VTX002
74.00 I          I          91  92 VTX003
75.00 I          I         121 139 VTX004
76.00 I          I         161 179 VTX005
77.00 I          I         201 219 VTX006
78.00 I          I         241 259 VTX007
79.00 I          I         281 299 VTX008
80.00 I          I         321 339 VTX009
81.00 I          I         361 379 VTX010
82.00 I          I         401 419 VTX011
83.00 I          I         441 459 VTX012
84.00 I          I         481 499 VTX013
85.00 I          I         521 539 VTX014
86.00 I          I         561 576 VTX015
87.00 I          I         601 616 VTX016
88.00 I          I         641 656 VTX017
89.00 I          I         681 696 VTX018
90.00 I          I         721 736 VTX019
91.00 I          I         761 776 VTX020
92.00 I          I         801 816 VTX021
93.00 I          I         841 856 VTX022
94.00 I          I         881 896 VTX023
95.00 I          I         921 936 VTX024
96.00 I          I         961 976 VTX025
97.00 I*
98.00 I/COPY JDRCPY, I0005IX  Data structure for commonly used indexes
99.00 I/COPY JDRCPY, I0005S  Data structure used with file servers
100.00 I/COPY JDRCPY, I0005P  Program status data structure
101.00 I*
102.00 I*
103.00 I*      Copy Member for Composite Common Subroutine - C005C
104.00 I*
105.00 I*
106.00 I/COPY JDRCPY, I0005  Data structure for vocabulary overrides and function keys
107.00 I*
108.00 I*
109.00 I*      Copy Member for Server - X0005
110.00 I*
111.00 I/COPY JDRCPY, I0005R  Data structure for file server X0005
112.00 I*
113.00 I*
114.00 I*      Copy Member for Server - X0006
115.00 I*
116.00 I/COPY JDRCPY, I0005E1
117.00 I*
118.00 I*
119.00 I*      Copy Member for Server - X9900E
120.00 I*
121.00 I/COPY JDRCPY, I9900E
122.00 I*
123.00 C*
124.00 C*      MAINLINE PROGRAM
125.00 C*      -----
126.00 C*
127.00 C*      Process housekeeping.
128.00 C*
129.00 C*      E00R 9999          One time only
130.00 C*      -----          functions
131.00 C*
132.00 C*      IF LR on, end program.
133.00 C*
134.00 C          *INLR      CASEQ'1'          903
135.00 C          -----          ---
136.00 C*
137.00 C*      If automatic inquiry set, process inquiry.
138.00 C*
139.00 C          SAUTO      CASEQ'1'          903
140.00 C          -----          ---
141.00 C          END
142.00 C*
143.00 C*      Begin normal program processing.
144.00 C*      -----
145.00 C*
146.00 C          *INLR      DOWRQ'0'
147.00 C*
148.00 C*      Write video screen.
149.00 C*

```

Each VTX field is 40 long but may not use all 40. Pulls in text from Vocabulary Overrides.

One time only functions

If information is passed to this program, it will automatically inquire on the record


```

229.00 C*      Set correct message in line 24.
230.00 C*
231.00 C      *IN93      IFRQ '1'
232.00 C      NOVELGVL24E      VDL24
233.00 C      ELER
234.00 C      NOVELGVL24M      VDL24
235.00 C      END
236.00 C*
237.00 C      END
238.00 C*
239.00 C      ROJ      TAG
240.00 C*      ---
241.00 C*
242.00 C*      END MAINLINE PROGRAM
243.00 C*
244.00 C*-----
245.00 C*
246.00 C*      Copy Common Subroutine - Edit Action Code
247.00 C*
248.00 C/COPY JDRCPY,C0001
249.00 C*-----
250.00 C*
251.00 C*      SUBROUTINE S00EX - Process Function Keys
252.00 C*-----
253.00 C*
254.00 C*      Processing:  1. Determine function key pressed.
255.00 C*                 2. Process function key request.
256.00 C*
257.00 C*      S00EX      S00ER
258.00 C*      -----
259.00 C*      TO0EXA      TAG
260.00 C*      -----
261.00 C*
262.00 C*      IF ROJ requested, exit subroutine.
263.00 C*
264.00 C*      S00AID      CARRQ#PROJ      ENDEXE      LR
265.00 C*
266.00 C*      IF Display Keys pressed, exit to help facility and return.
267.00 C*
268.00 C*-----
269.00 C*
270.00 C*      S00AID      IFRQ #FNYS
271.00 C*      CALL 'P960LR'      99
272.00 C*      -----
273.00 C*      FARM      FARM      S00GC
274.00 C*      FARM      FARM      S00VDS
275.00 C*      FARM      FARM      S00CNR
276.00 C*
277.00 C*      S00AID      CARRQ#FNYS      TO0EXA
278.00 C*      -----
279.00 C*      GOTO ENDEXE
280.00 C*
281.00 C*      END
282.00 C*
283.00 C*      IF Cursor Sensitive Help Pressed, exit to CR Help.
284.00 C*
285.00 C*-----
286.00 C*      S00AID      IFRQ #P0MX
287.00 C*      NOVEL*IR      S00M
288.00 C*      CALL 'X960CX'      99
289.00 C*      -----
290.00 C*      FARM      FARM      S00GC
291.00 C*      FARM      FARM      S00VDS
292.00 C*      FARM      FARM      S00CNR
293.00 C*      FARM      FARM      S00CFF 2
294.00 C*      FARM      FARM      S00MDS
295.00 C*
296.00 C*      S00AID      IFRQ *SLANR
297.00 C*      EXER S00VL
298.00 C*      -----
299.00 C*      NOVEL*IR      *IN,1
300.00 C*      END
301.00 C*      NOVEL*SLANR      S00TAI
302.00 C*      GOTO ENDEXE
303.00 C*
304.00 C*      END
305.00 C*

```

Get correct message in line 24.

```

*IN93      IFRQ '1'
NOVELGVL24E      VDL24
ELER
NOVELGVL24M      VDL24
END

```

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

```

306.00 C*      If Display errors pressed, exit to error messages.
307.00 C*      -----
308.00 C*
309.00 CRN      @SAID      IPBQ #PRND
310.00 CRN      Z-ADD1      WG
311.00 CRN      Z-ADD1      WH
312.00 CRN      #G      DOWLR#4
313.00 CRN      @MK,#G      IPBQ '1'
314.00 CRN      MOVE @MK, #G      @GR, #H
315.00 CRN      ADD 1      WH
316.00 CRN      END
317.00 CRN      ADD 1      WG
318.00 CRN      END
319.00 CRN      CALL 'P00002'      PP
320.00 C*      -----
321.00 CRN      PARM      @GR
322.00 CRN      GOTO @ENDXK
323.00 C*      -----
324.00 CRN      END
325.00 C*
326.00 C*      If HELP key pressed, exit to help facility and return.
327.00 C*      -----
328.00 C*
329.00 CRN      @SAID      IPBQ #PHLP
330.00 CRN      CALL 'P00002'      PP      Access JDE program level
331.00 C*      -----      Help information
332.00 CRN      PARM      HNSO
333.00 CRN      PARM      HNSO
334.00 CRN      PARM      IOGOC
335.00 CRN      PARM      @RVIDG
336.00 CRN      GOTO @ENDXK
337.00 C*      -----
338.00 C*      END
339.00 C*
340.00 C*      If Clear screen pressed, clear screen and return.
341.00 C*      -----
342.00 C*
343.00 C*
344.00 CRN      @SAID      IPBQ #PCLR
345.00 CRN      EXGR @001
346.00 C*      -----
347.00 CRN      GOTO @ENDXK
348.00 C*      -----
349.00 CRN      END
350.00 C*
351.00 C*      Process roll up and down keys.
352.00 C*      -----
353.00 C*
354.00 CRN      @SAID      IPBQ #PROLU
355.00 CRN      @SAID      ORBQ #PROLD
356.00 CRN      @SRUCR      DOUNQ' '
357.00 CRN      MOVE ' '      @SRUCR 1
358.00 C*
359.00 C*      If ROLL UP key pressed, process read next.
360.00 C*      -----
361.00 C*
362.00 C*
363.00 C*      @SAID      IPBQ #PROLU
364.00 C*
365.00 C*      Reset error indicators if roll
366.00 CRN      MOVE@SRUCR      *IN, 41
367.00 CRN      MOVE '0'      *IN, 40
368.00 CRN      @STOP
369.00 CRN      READ I@2@01      @1@2@00
370.00 CRN      *IN@1      IPBQ '1'      @@1
371.00 CRN      @SRUCR      @RTLLI@2@01
372.00 CRN      @STOP      @2@00
373.00 CRN      READI@2@01      @@2
374.00 C*
375.00 C*      If error on read, set error.
376.00 C*
377.00 CRN      *IN@2      IPBQ '1'      @@41
378.00 CRN      @STOP
379.00 CRN      MOVE '1'      @MK, 1
380.00 CRN      GOTO @ENDXK
381.00 C*      -----
382.00 CRN      END
383.00 CRN      END

```

```

385.00 C*
386.00 C* IF ROLL DOWN key pressed, process read prior.
387.00 C* -----
388.00 C*
389.00 CGR      @SAID      IFRQ #FROLD
390.00 C*
391.00 C* Reset error indicators if roll
392.00 C*
393.00 CGR      NOVEL#RSEXY      *IN,41
394.00 CGR      MOVE '0'      *IN,40
395.00 CGR      SETOP      010299
396.00 CGR      READ#F92901      9901
397.00 CGR      *IN91      IFRQ '1'
398.00 CGR      $RSEXY      SETTLI92901
399.00 CGR      SETOP      0299
400.00 CGR      READ#F92901      9902
401.00 C*
402.00 C* If error on read, set error.
403.00 C*
404.00 CGR      *IN92      IFRQ '1'
405.00 CGR      SETOP      9741
406.00 CGR      MOVE '1'      BNX,2
407.00 CGR      GOTO ENDEXE
408.00 C* -----
409.00 CGR      END
410.00 CGR      END
411.00 CGR      END
412.00 C*
413.00 C* Load video screen data on roll keys.
414.00 C* -----
415.00 C*
416.00 CGR      @SAID      IFRQ #FROLD
417.00 CGR      @SAID      ORGQ #FROLD
418.00 C*
419.00 C* Release record lock or report record in use.
420.00 C*
421.00 CGR      *IN99      IFRQ '0'
422.00 CGR      RCTCTUNLOCK
423.00 CGR      ELSE
424.00 CGR      CALL 'F99ELCK'      01
425.00 C* -----
426.00 CGR      PARM      #F99DC
427.00 CGR      SETOP      9241
428.00 CGR      MOVE '1'      BNX,6
429.00 CGR      GOTO ENDEXE
430.00 C* -----
431.00 CGR      END
432.00 C*
433.00 C*
434.00 C* Cost Center security edit.
435.00 C*
436.00 CGR      NOVEL#F92901      #FILE
437.00 CGR      NOVEL#QX0C      #MCU
438.00 CGR      #AUT      IFRQ '1'
439.00 CGR      #PAUT      ANSWR'1'
440.00 CGR      EDGE C0000
441.00 C* -----
442.00 CGR      END
443.00 CGR      #AUT      IFRQ '1'
444.00 CGR      #PAUT      ANSWR'1'
445.00 CGR      #GAUT      ANSWR'1'
446.00 CGR      MOVE '1'      $SECUR
447.00 CGR      END
448.00 CGR      $SECUR      CARRQ' '      0004
449.00 C* -----
450.00 CGR      END
451.00 C*
452.00 CGR      END
453.00 C*
454.00 CGR      END
455.00 CGR      GOTO ENDEXE
456.00 C* -----
457.00 CGR      END
458.00 C*
459.00 CGR      @SAID      IFRQ '1'
460.00 CGR      SETOP      0192
461.00 CGR      GOTO ENDEXE
462.00 C* -----
463.00 CGR      END
464.00 C*
465.00 CGR      @SAID      IFRQ '1'

```

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.

```

466.00 C*.....
467.00 C*
468.00 C*      Copy Common Subroutine - Cost Center Security Check
469.00 C*
470.00 C/CPY JDECPY,C0000
471.00 C*.....
472.00 C*
473.00 C*      SUBROUTINE RSOVL - Cursor Control Return Values
474.00 C*
475.00 C*
476.00 C*      By format, find the field to update and move in the
477.00 C*      returned value. If the format is a subfile, the record
478.00 C*      to change is found in @@R333.
479.00 C*
480.00 CBR          SOOVL          BRGGR
481.00 C*
482.00 C*
483.00 CBR          ##RVAL          IPRQ 'BLANK
484.00 CBR                          MOVE 'BLANK          ##RVAL
485.00 CBR                          END
486.00 C*
487.00 C*      Return values for fields in format V9200111
488.00 C*
489.00 CBR          ##RWT          IPRQ 'V9200111'
490.00 C*
491.00 CBR          ##PLEM          IPRQ 'ACTION          '
492.00 CBR                          MOVE##RVAL          ACTION
493.00 CBR                          GOTO ENDOVL
494.00 C*
495.00 CBR                          END
496.00 C*
497.00 CBR          ##PLEM          IPRQ 'VDXIT          '
498.00 CBR                          MOVE##RVAL          VDXIT
499.00 CBR                          GOTO ENDOVL
500.00 C*
501.00 CBR                          END
502.00 C*
503.00 CBR          ##PLEM          IPRQ 'VDXDS          '
504.00 CBR                          MOVE##RVAL          VDXDS
505.00 CBR                          GOTO ENDOVL
506.00 C*
507.00 CBR                          END
508.00 C*
509.00 CBR          ##PLEM          IPRQ 'VDXCC          '
510.00 CBR                          MOVE##RVAL          VDXCC
511.00 CBR                          GOTO ENDOVL
512.00 C*
513.00 CBR                          END
514.00 C*
515.00 CBR          ##PLEM          IPRQ 'VDXTY          '
516.00 CBR                          MOVE##RVAL          VDXTY
517.00 CBR                          GOTO ENDOVL
518.00 C*
519.00 CBR                          END
520.00 C*
521.00 CBR          ##PLEM          IPRQ 'VDXDT          '
522.00 CBR                          MOVE##RVAL          VDXDT
523.00 CBR                          GOTO ENDOVL
524.00 C*
525.00 CBR                          END
526.00 C*
527.00 CBR          ##PLEM          IPRQ 'VDXQT          '
528.00 CBR                          MOVE##RVAL          VDXQT
529.00 CBR                          GOTO ENDOVL
530.00 C*
531.00 CBR                          END
532.00 C*
533.00 CBR          ##PLEM          IPRQ 'VDXDM          '
534.00 CBR                          MOVE##RVAL          VDXDM
535.00 CBR                          GOTO ENDOVL
536.00 C*
537.00 CBR                          END
538.00 C*
539.00 CBR          ##PLEM          IPRQ 'VDX001          '
540.00 CBR                          MOVE##RVAL          VDX001
541.00 CBR                          GOTO ENDOVL
542.00 C*

```

For cursor sensitive help. Information was retrieved in program X96CCX. The retrieved information is returned to the video fields in this subroutine.

```

543.00 CWR          END
544.00 C*
545.00 CWR          ##FLM  IPRQ 'VDX002  '
546.00 CWR          MOVBL#RVAL
547.00 CWR          GOTO ENDOWL
548.00 C*
549.00 CWR          END
550.00 C*
551.00 CWR          ##FLM  IPRQ 'VDX003  '
552.00 CWR          MOVBL#RVAL
553.00 CWR          GOTO ENDOWL
554.00 C*
555.00 CWR          END
556.00 C*
557.00 CWR          ##FLM  IPRQ 'VDX004  '
558.00 CWR          MOVBL#RVAL
559.00 CWR          GOTO ENDOWL
560.00 CWR          END
561.00 C*
562.00 CWR          ##FLM  IPRQ 'VDX005  '
563.00 CWR          MOVBL#RVAL
564.00 CWR          GOTO ENDOWL
565.00 CWR          END
566.00 C*
567.00 CWR          END
568.00 CWR          END
569.00 C*
570.00 CWR          ENDOWL  ENDWR
571.00 C*
572.00 C*
573.00 C*
574.00 C*
575.00 C*
576.00 C*
577.00 C*
578.00 C*
579.00 C*
580.00 CWR          S001  BRGR
581.00 C*
582.00 C*
583.00 C*
584.00 C*
585.00 C*
586.00 CWR          *NOKEY  CTR#P2201
587.00 CWR          MOVE *BLANK  ##CPL
588.00 CWR          MOVE *BLANK  ##CNC
589.00 CWR          Z-ADD*ZERO  ##CCL
590.00 CWR          Z-ADD*ZERO  ##RRCM
591.00 CWR          MOVE *BLANK  VDXCC
592.00 CWR          MOVE *BLANK  VDXDS
593.00 CWR          MOVE *BLANK  VDXDT
594.00 CWR          MOVE *BLANK  VDXIT
595.00 CWR          MOVE *BLANK  VDXQT
596.00 CWR          MOVE *BLANK  VDXTY
597.00 CWR          MOVE *BLANK  VDXCM
598.00 CWR          MOVE *BLANK  VDX001
599.00 CWR          MOVE *BLANK  VDX002
600.00 CWR          MOVE *BLANK  VDX003
601.00 CWR          MOVE *BLANK  VDX004
602.00 CWR          MOVE *BLANK  VDX005
603.00 CWR          MOVBL#VL24M  VDL24
604.00 CWR          MOVE ' '  @IN27  1
605.00 C*
606.00 C*
607.00 C*
608.00 CWR          @S0AID  IPRQ #PCL#
609.00 CWR          MOVE *ALL'0'  $$$SET
610.00 CWR          MOVBRSET  *IN,41
611.00 CWR          MOVE ' '  ACTION  1
612.00 CWR          Z-ADD*ZERO  QXKIT
613.00 CWR          MOVE *BLANK  VC001
614.00 CWR          MOVE *BLANK  VC002
615.00 CWR          MOVE *BLANK  VC003
616.00 CWR          MOVE *BLANK  VC004
617.00 CWR          MOVE *BLANK  VC005
618.00 CWR          MOVE *BLANK  VC006
619.00 CWR          MOVE *BLANK  VC007
620.00 CWR          MOVE *BLANK  VC008
621.00 CWR          Z-ADD*ZERO  $$$ZDT  60
622.00 C*
623.00 CWR          END001  ENDWR

```

Clears all the fields in the record format for P92801

Clears 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 everytime S001 is executed.

```

624.00 C*-----
625.00 C*
626.00 C* SUBROUTINE 8003 - Edit Key ----- Sets the file pointer and edits the key
627.00 C*-----
628.00 C*
629.00 C* Processing: 1. Clear error indicators and arrays.
630.00 C*              2. Load input keys.
631.00 C*              3. Validate master file key.
632.00 C*              4. Release master file record lock.
633.00 C*              5. Load video screen output on inquiry.
634.00 C*
635.00 CER      8003      8003
636.00 C*      ----      ----
637.00 C*
638.00 C* Load data field dictionary parameters (one cycle only).
639.00 C*
640.00 CER      9999      CARRQ' '      9999
641.00 C*
642.00 CER      END
643.00 C*
644.00 C* Reset error indicators and arrays.
645.00 C*
646.00 CER      MOVE 'ALL'0'      SRRSET 29
647.00 CER      MOVE 'BLANK'      SRRSTL 63
648.00 CER      MOVE#SRRST      'DI,41
649.00 CER      MOVE#SRRSTL      &MX,2
650.00 CER      CLEARERR
651.00 C*-----
652.00 C*
653.00 C* Load video input field for - Item ID
654.00 C*
655.00 CER      MOVE#VIXIT      &NM
656.00 CER      EXER C0012
657.00 C*      ----
658.00 CER      1-ADD#NUMB      SRRR02 80
659.00 CER      MOVE #SRRR02      QXXIT
660.00 C*
661.00 C* Automatic Next Number for - Item ID
662.00 C*
663.00 CER      *IN21      IFRQ '1'
664.00 CER      VIXIT      ANDRQ'BLANK
665.00 CER      SRTON
666.00 CER      *IN91      DOWRQ'1'
667.00 CER      MOVE #VIXIT      PDIXM 2
668.00 CER      CALL 'X0010'
669.00 C*      ----
670.00 CER      SRRM #VIXIT      SRRY 4
671.00 CER      SRRM
672.00 CER      SRRM *SRR0      SRRITNO 80
673.00 CER      MOVE #SRRTNO      QXXIT
674.00 CER      MOVE #SRRTNO      VIXIT
675.00 CER      QXXIT      SRTLLP2201
676.00 CER      END
677.00 CER      END
678.00 C*-----
679.00 CER      QXXY01      CHAINI92001      9999
680.00 C*
681.00 C* Cost Center security edit.
682.00 C*
683.00 CER      MOVE#F92001      'SFILE
684.00 CER      MOVE#QXXCC      &MCU
685.00 CER      #AUT      IFRQ '1'
686.00 CER      #FAUT      ANDNR'1'
687.00 CER      EXER C0000 ----- Checks cost center security
688.00 C*
689.00 CER      END
690.00 CER      #AUT      IFRQ '1'
691.00 CER      #FAUT      ANDNR'1'
692.00 CER      #MAUT      ANDNR'1'
693.00 CER      MOVE '1'      $$$SRR 1
694.00 CER      END
695.00 C*
696.00 C* If security violation, set error condition.
697.00 C*
698.00 CER      $$$SRR      IFRQ '1'
699.00 CER      MOVE '1'      &MX,2
700.00 CER      SRTON

```

```

701.00 CSM      MOVE ' '          $$$RCH 1
702.00 CSM      GOTO END003
703.00 C*      -----
704.00 CSM      END
705.00 C*
706.00 C*      Edit result of read and action code.
707.00 C*
708.00 CSM      *IN99      IPRQ '1'
709.00 CSM      *IN21      COMP '0'          41 "error"
710.00 CSM      ELSE
711.00 CSM      *IN21      COMP '1'          41 "error"
712.00 CSM      END
713.00 C*
714.00 C*      If indicator 41 on, invalid key for action code.
715.00 C*
716.00 CSM      *IN41      IPRQ '1'
717.00 CSM      MOVE '1'          @MX,2      92
718.00 CSM      SETON
719.00 CSM      END
720.00 C*
721.00 C*      If indicator 99 on, record in use.
722.00 C*
723.00 CSM      *IN99      IPRQ '1'
724.00 CSM      CALL '999BLCK'          91
725.00 C*      -----
726.00 CSM      PARM          ##PDCG
727.00 CSM      MOVE '1'          @MX,6      9241
728.00 CSM      SETON
729.00 CSM      END
730.00 C*      -----
731.00 C*
732.00 C*      If not inquiry, skip remainder of subroutine.
733.00 C*
734.00 CSM      *IN24      CANRQ'0'          END003
735.00 C*      -----
736.00 C*      -----
737.00 C*
738.00 C*      Release record lock on master file.
739.00 C*
740.00 CSM      *IN99      IPRQ '0'
741.00 CSM      *IN99      ANDRQ'0'
742.00 CSM      EXECUTUNLOCK
743.00 CSM      END
744.00 C*
745.00 C*      If errors, skip remainder of subroutine.
746.00 C*
747.00 CSM      *IN93      CANRQ'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 CSM      EXEC 8004
754.00 C*      -----
755.00 C*
756.00 CSM      END003      ENDGR
757.00 C*      -----
758.00 C*
759.00 C*      Copy Common Subroutine - Right Justify Numeric Fields
760.00 C*
761.00 C/COPY JNRCPY,C0012
762.00 C*      -----
763.00 C*
764.00 C*      SUBROUTINE 8004 - Load Video Screen Data
765.00 C*      -----
766.00 C*
767.00 C*      Processing: 1. Move data base information to video screen.
768.00 C*      All video screen fields are alpha and
769.00 C*      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
777.00 C*      X0019

```

```

740.00 CSM      *IN99      IPRQ '0'
741.00 CSM      *IN99      ANDRQ'0'
742.00 CSM      EXECUTUNLOCK
743.00 CSM      END

```

JDE uses this or SETLL
to release record locks

```

753.00 CSM      EXEC 8004

```

Moves information to the video/report fields

```

778.00 C*
779.00 CRR      8004      8888
780.00 C*
781.00 C*
782.00 C*
783.00 C*      Move to output - Description for Cost Center
784.00 C*
785.00 CRR      CALL 'X0005'          01
786.00 C*
787.00 CRR      FARM *BLANK          PGM000 1
788.00 CRR      FARM '1'          PGM000 1
789.00 CRR      FARM QXX00          PGM000 12
790.00 CRR      FARM *BLANK          PRR000 4
791.00 CRR      FARM          I0005
792.00 C*
793.00 CRR      MOVE *BLANK          VCO001
794.00 CRR      PRR000          IPREQ *BLANK
795.00 CRR      MOVEBLMCDL01          VCO001
796.00 CRR
797.00 C*
798.00 C*
799.00 C*      Description display for - Item Type
800.00 C*
801.00 CRR      CLEAR I0005U
802.00 CRR      MOVEBLMXXTY          WUNY
803.00 CRR      MOVE BXXTY          WUNT
804.00 CRR      MOVE QXXTY          WUNY
805.00 CRR      CALL 'X0005'          01
806.00 C*
807.00 CRR      FARM          I0005U
808.00 CRR      MOVE *BLANK          VCO002
809.00 CRR      #0000          IPREQ '0'
810.00 CRR      MOVEBLMUDL01          VCO002
811.00 CRR
812.00 C*
813.00 C*
814.00 C*      Description display for - Item Unit of Measure
815.00 C*
816.00 CRR      CLEAR I0005U
817.00 CRR      MOVEBLMXXUM          WUNY
818.00 CRR      MOVE BXXUM          WUNT
819.00 CRR      MOVE QXXUM          WUNY
820.00 CRR      CALL 'X0005'          01
821.00 C*
822.00 CRR      FARM          I0005U
823.00 CRR      MOVE *BLANK          VCO003
824.00 CRR      #0000          IPREQ '0'
825.00 CRR      MOVEBLMUDL01          VCO003
826.00 CRR
827.00 C*
828.00 C*
829.00 C*      Description display for - Item Category Code 001
830.00 C*
831.00 CRR      CLEAR I0005U
832.00 CRR      MOVEBLMXX001          WUNY
833.00 CRR      MOVE BXX001          WUNT
834.00 CRR      MOVE QXX001          WUNY
835.00 CRR      CALL 'X0005'          01
836.00 C*
837.00 CRR      FARM          I0005U
838.00 CRR      MOVE *BLANK          VCO004
839.00 CRR      #0000          IPREQ '0'
840.00 CRR      MOVEBLMUDL01          VCO004
841.00 CRR
842.00 C*
843.00 C*
844.00 C*      Description display for - Item Category Code 002
845.00 C*
846.00 CRR      CLEAR I0005U
847.00 CRR      MOVEBLMXX002          WUNY
848.00 CRR      MOVE BXX002          WUNT
849.00 CRR      MOVE QXX002          WUNY
850.00 CRR      CALL 'X0005'          01
851.00 C*
852.00 CRR      FARM          I0005U
853.00 CRR      MOVE *BLANK          VCO005
854.00 CRR      #0000          IPREQ '0'

```

Server for Bus. Unit

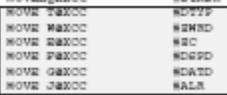
Description loaded to the *VCO field

File server for user defined codes

```

955.00 CGR          NOVEL#UIDL01      VC0005
956.00 CGR          END
957.00 C*
958.00 C*
959.00 C*   Description display for - Item Category Code 003
960.00 C*
961.00 CGR          CLEARI0005U
962.00 CGR          NOVELG0002      WURY
963.00 CGR          MOVE 00X002      WURT
964.00 CGR          MOVE 00X002      WURY
965.00 CGR          CALL '00005'      91
966.00 C*
967.00 CGR          PARM              I0005U
968.00 CGR          MOVE *BLANK      VC0006
969.00 CGR          #UERR          IPRQ '0'
970.00 CGR          NOVEL#UIDL01      VC0006
971.00 CGR          END
972.00 C*
973.00 C*
974.00 C*   Description display for - Item Category Code 004
975.00 C*
976.00 CGR          CLEARI0005U
977.00 CGR          NOVELG0004      WURY
978.00 CGR          MOVE 00X004      WURT
979.00 CGR          MOVE 00X004      WURY
980.00 CGR          CALL '00005'      91
981.00 C*
982.00 CGR          PARM              I0005U
983.00 CGR          MOVE *BLANK      VC0007
984.00 CGR          #UERR          IPRQ '0'
985.00 CGR          NOVEL#UIDL01      VC0007
986.00 CGR          END
987.00 C*
988.00 C*
989.00 C*   Description display for - Item Category Code 005
990.00 C*
991.00 CGR          CLEARI0005U
992.00 CGR          NOVELG0005      WURY
993.00 CGR          MOVE 00X005      WURT
994.00 CGR          MOVE 00X005      WURY
995.00 CGR          CALL '00005'      91
996.00 C*
997.00 CGR          PARM              I0005U
998.00 CGR          MOVE *BLANK      VC0008
999.00 CGR          #UERR          IPRQ '0'
1000.00 CGR          NOVEL#UIDL01      VC0008
1001.00 CGR          END
1002.00 C*
1003.00 C*
1004.00 C*   Move to output - Cost Center
1005.00 C*
1006.00 CGR          MOVE *BLANK      #SINR
1007.00 CGR          NOVELG000C      #SINR
1008.00 CGR          MOVE *BLANK      #UTFP
1009.00 CGR          MOVE 00X00C      #ENRD
1010.00 CGR          MOVE 00X00C      #EC
1011.00 CGR          MOVE 00X00C      #C0SD
1012.00 CGR          MOVE 00X00C      #C0TD
1013.00 CGR          MOVE 00X00C      #ALS
1014.00 CGR          MOVE ' '      #E0CR
1015.00 CGR          MOVE ' '      #D0CR
1016.00 CGR          #XGR C0016L
1017.00 C*
1018.00 CGR          #ALS          IPRQ 'L'
1019.00 CGR          NOVEL#SINR      VENCC
1020.00 CGR          ELGR
1021.00 CGR          MOVE #SINR      VENCC
1022.00 CGR          END
1023.00 C*
1024.00 C*
1025.00 C*   Move to output - Description
1026.00 C*
1027.00 CGR          NOVELG000G      VENDR
1028.00 C*
1029.00 C*
1030.00 C*   Move to Output - Date Last Ship
1031.00 C*

```



Editing information
retrieved in S998



Copy module to edit
field for use on
screen/report

```

932.00 CWR      MOVE GXXDT      #GIDAT 6
933.00 CWR      MOVE *BLANK      #SDAT 9
934.00 CWR      MOVEL' 'DUL      'SFFMT 7
935.00 CWR      MOVEL' 'SUSVAL  'SFFMT 7
936.00 CWR      MOVEL' 'SUSVAL  'SFFMT 7
937.00 CWR      MOVE ' '          SERTST 1
938.00 CWR      CALL 'X0000  '  01
939.00 C*
940.00 CWR      FARM          #GIDAT
941.00 CWR      FARM          #SDAT
942.00 CWR      FARM          SFFMT
943.00 CWR      FARM          SFFMT
944.00 CWR      FARM          SFFMT
945.00 CWR      FARM          SERTST
946.00 CWR      FARM          SERTST
947.00 CWR      MOVEL#SDAT  VEXDT
948.00 C*-----
949.00 C*
949.00 C*      Move to output - Item ID
950.00 C*
951.00 CWR      MOVE *BLANK      #GIDNR
952.00 CWR      MOVEL#GXXDT  #GIDNR
953.00 CWR      MOVE TXKIT      #DTYP
954.00 CWR      MOVE #XKIT      #ENRD
955.00 CWR      MOVE #XKIT      #EC
956.00 CWR      MOVE #XKIT      #DFFD
957.00 CWR      MOVE GXXIT      #DSTD
958.00 CWR      MOVE #XKIT      #SAL
959.00 CWR      MOVE ' '          #SECR
960.00 CWR      MOVE ' '          #DCCR
961.00 CWR      EXER C00101
962.00 C*-----
963.00 CWR      #SAL      IFRQ 'L'
964.00 CWR      MOVEL#GIDNR  VEXIT
965.00 CWR      ELGR
966.00 CWR      MOVE #GIDNR  VEXIT
967.00 CWR      END
968.00 C*-----
969.00 C*
970.00 C*      Move to output - Quantity - On Hand
971.00 C*
972.00 CWR      MOVE *BLANK      #GIDNR
973.00 CWR      MOVEL#GXXQT  #GIDNR
974.00 CWR      MOVE TXKIT      #DTYP
975.00 CWR      MOVE #XKIT      #ENRD
976.00 CWR      MOVE #XKIT      #EC
977.00 CWR      MOVE #XKIT      #DFFD
978.00 CWR      MOVE GXXQT      #DSTD
979.00 CWR      MOVE #XKIT      #SAL
980.00 CWR      MOVE ' '          #SECR
981.00 CWR      MOVE ' '          #DCCR
982.00 CWR      EXER C00101
983.00 C*-----
984.00 CWR      #SAL      IFRQ 'L'
985.00 CWR      MOVEL#GIDNR  VEXQT
986.00 CWR      ELGR
987.00 CWR      MOVE #GIDNR  VEXQT
988.00 CWR      END
989.00 C*-----
990.00 C*
991.00 C*      Move to output - Item type
992.00 C*
993.00 CWR      MOVEL#GXTY  VEXTY
994.00 C*-----
995.00 C*
996.00 C*      Move to output - Item Unit of Measure
997.00 C*
998.00 CWR      MOVEL#XUM  VEXUM
999.00 C*-----
1000.00 C*
1001.00 C*      Move to output - Item Category Code 001
1002.00 C*
1003.00 CWR      MOVE *BLANK      #GIDNR
1004.00 CWR      MOVEL#X0001  #GIDNR
1005.00 CWR      MOVE TX0001  #DTYP
1006.00 CWR      MOVE #X0001  #ENRD
1007.00 CWR      MOVE #X0001  #EC
1008.00 CWR      MOVE GXX001  #DSTD
1009.00 CWR      MOVE #X0001  #DFFD

```

External program
used to edit dates

```

1010.00 CGR          MOVE JAX001      WALS
1011.00 CGR          MOVE ' '          WDCOR
1012.00 CGR          MOVE ' '          WDCOR
1013.00 CGR          EDGE C00161
1014.00 C*          -----
1015.00 CGR          WALS      IFRQ 'L'
1016.00 CGR          NOVEL#SINR      VEX001
1017.00 CGR          ELSE
1018.00 CGR          MOVE #SINR      VEX001
1019.00 CGR          END
1020.00 C*          -----
1021.00 C*
1022.00 C*      Move to output - Item Category Code 002
1023.00 C*
1024.00 CGR          MOVE *BLANK      #SINR
1025.00 CGR          NOVELQXX002      #SINR
1026.00 CGR          MOVE TAX002      #DTYP
1027.00 CGR          MOVE MAX002      #WTRD
1028.00 CGR          MOVE SXX002      #RC
1029.00 CGR          MOVE PXX002      #GSPD
1030.00 CGR          MOVE GXX002      #DSTD
1031.00 CGR          MOVE JAX002      WALS
1032.00 CGR          MOVE ' '          WDCOR
1033.00 CGR          MOVE ' '          WDCOR
1034.00 CGR          EDGE C00161
1035.00 C*          -----
1036.00 CGR          WALS      IFRQ 'L'
1037.00 CGR          NOVEL#SINR      VEX002
1038.00 CGR          ELSE
1039.00 CGR          MOVE #SINR      VEX002
1040.00 CGR          END
1041.00 C*          -----
1042.00 C*
1043.00 C*      Move to output - Item Category Code 003
1044.00 C*
1045.00 CGR          MOVE *BLANK      #SINR
1046.00 CGR          NOVELQXX003      #SINR
1047.00 CGR          MOVE TAX003      #DTYP
1048.00 CGR          MOVE MAX003      #WTRD
1049.00 CGR          MOVE SXX003      #RC
1050.00 CGR          MOVE PXX003      #GSPD
1051.00 CGR          MOVE GXX003      #DSTD
1052.00 CGR          MOVE JAX003      WALS
1053.00 CGR          MOVE ' '          WDCOR
1054.00 CGR          MOVE ' '          WDCOR
1055.00 CGR          EDGE C00161
1056.00 C*          -----
1057.00 CGR          WALS      IFRQ 'L'
1058.00 CGR          NOVEL#SINR      VEX003
1059.00 CGR          ELSE
1060.00 CGR          MOVE #SINR      VEX003
1061.00 CGR          END
1062.00 C*          -----
1063.00 C*
1064.00 C*      Move to output - Item Category Code 004
1065.00 C*
1066.00 CGR          MOVE *BLANK      #SINR
1067.00 CGR          NOVELQXX004      #SINR
1068.00 CGR          MOVE TAX004      #DTYP
1069.00 CGR          MOVE MAX004      #WTRD
1070.00 CGR          MOVE SXX004      #RC
1071.00 CGR          MOVE PXX004      #GSPD
1072.00 CGR          MOVE GXX004      #DSTD
1073.00 CGR          MOVE JAX004      WALS
1074.00 CGR          MOVE ' '          WDCOR
1075.00 CGR          MOVE ' '          WDCOR
1076.00 CGR          EDGE C00161
1077.00 C*          -----
1078.00 CGR          WALS      IFRQ 'L'
1079.00 CGR          NOVEL#SINR      VEX004
1080.00 CGR          ELSE
1081.00 CGR          MOVE #SINR      VEX004
1082.00 CGR          END
1083.00 C*          -----
1084.00 C*
1085.00 C*      Move to output - Item Category Code 005
1086.00 C*

```

```

1097.00 CSR          MOVE *BLANK          #SINR
1098.00 CSR          MOVE#QX005          #SINR
1099.00 CSR          MOVE TX005          #DTYP
1100.00 CSR          MOVE MXX005          #MWD
1101.00 CSR          MOVE #X005          #EC
1102.00 CSR          MOVE #X005          #DOPD
1103.00 CSR          MOVE #X005          #DWD
1104.00 CSR          MOVE #X005          #ALR
1105.00 CSR          MOVE ' '          #ECC
1106.00 CSR          MOVE ' '          #ECC
1107.00 CSR          #X005 C00161
1108.00 C*          -----
1109.00 CSR          #ALR          IF#Q 'L'
1110.00 CSR          MOVE#SINR          #VMOO5
1111.00 CSR          #R          #R
1112.00 CSR          MOVE #SINR          #VMOO5
1113.00 CSR          #R          #R
1114.00 C*          -----
1115.00 CSR          #R          #R
1116.00 C*          Copy Common Subroutine - Format Numeric Fields for Output with Override
1117.00 C*          C/COPY JDS005,C00161
1118.00 C*          -----
1119.00 C*          SUBROUTINE #005 - Scrub Input          Validates and edits data
1120.00 C*          -----          entered by the user
1121.00 C*          Processing: 1. Validate all video input.
1122.00 C*          All numeric fields must be processed
1123.00 C*          thru subroutines C0012 and C0015 in order
1124.00 C*          to scrub the alpha input field and convert
1125.00 C*          15 digits and 0 decimals.
1126.00 C*          Date fields must be converted from system
1127.00 C*          format to their internal format of month,
1128.00 C*          day and year or julian using program X0028.
1129.00 C*          2. Update data record fields from video.
1130.00 CSR          #005          #RGR
1131.00 C*          ----          ----
1132.00 C*          If not addition or change, bypass subroutine.
1133.00 C*          *IN21          IF#Q '0'
1134.00 C*          *IN22          AND#Q'0'
1135.00 C*          GOTO #R005
1136.00 C*          -----
1137.00 CSR          #R          #R
1138.00 C*          #R          #R
1139.00 C*          #R          #R
1140.00 C*          #R          #R
1141.00 C*          Scrub and edit - Cost Center
1142.00 C*          #R          #R
1143.00 CSR          CALL 'X0006'          #R
1144.00 C*          -----
1145.00 CSR          #R          #R          #R          #R
1146.00 CSR          #R          #R          #R          #R
1147.00 CSR          #R          #R          #R          #R
1148.00 CSR          #R          #R          #R          #R
1149.00 CSR          #R          #R          #R          #R
1150.00 C*          #R          #R          #R          #R
1151.00 CSR          #R          #R          #R          #R
1152.00 CSR          #R          #R          #R          #R
1153.00 CSR          #R          #R          #R          #R
1154.00 CSR          #R          #R          #R          #R
1155.00 CSR          #R          #R          #R          #R
1156.00 CSR          #R          #R          #R          #R
1157.00 C*          -----
1158.00 C*          Scrub and edit - Description
1159.00 C*          #R          #R          #R          #R
1160.00 CSR          #R          #R          #R          #R
1161.00 CSR          #R          #R          #R          #R
1162.00 C*          Set default value - Description
1163.00 C*          #R          #R          #R          #R
1164.00 C*          #R          #R          #R          #R

```

SUBROUTINE #005 - Scrub Input

Validates and edits data entered by the user

If not addition or change, bypass subroutine.
 *IN21 IF#Q '0'
 *IN22 AND#Q'0'
 GOTO #R005

 #R #R

Only performs this subroutine if action code is add or change

```

1165.00 CGR      QXXDS      IPFRQ *BLANK
1166.00 CGR      DAXDS      IPFRQ *BLANK
1167.00 CGR      NOVSRDAXDS  NOVSRDAXDS      @DV
1168.00 CGR      NOVSRD@DV  QXXDS
1169.00 CGR      @DV,1      IPFRQ
1170.00 CGR      MOVE ' '      @DV,1
1171.00 CGR      I-ADD2
1172.00 CGR      @M      DOWLR40
1173.00 CGR      @DV,@M      IPFRQ '''
1174.00 CGR      MOVE ' '      @DV,@M
1175.00 CGR      END
1176.00 CGR      ADD 1      @M
1177.00 CGR      END
1178.00 CGR      NOVSRD@DV,2  QXXDS
1179.00 CGR      END
1180.00 CGR      END
1181.00 CGR      END
1182.00 C*
1183.00 C*      Edit allowed values - Description
1184.00 C*
1185.00 CGR      AXDS      IPFRQ *NR
1186.00 CGR      QXXDS      ANDRQ*BLANK
1187.00 CGR      MOVE '1'      @MX,02
1188.00 CGR      SETON      4293
1189.00 CGR      END
1190.00 C*
-----
1191.00 C*
1192.00 C*      Scrub and edit - Date Last Ship
1193.00 C*
1194.00 CGR      NOVSRDAXIT  @M
1195.00 CGR      @M      A
1196.00 C*      Work fields used in the RPG
1197.00 CGR      I-ADD@M      @M      50      program begin with $
1198.00 CGR      MOVE @M      QXXDT
1199.00 C*
1200.00 C*      Edit julian date - Date Last Ship
1201.00 C*
1202.00 CGR      V@M      IPFRQ *BLANK
1203.00 CGR      MOVE QXXDT  @MIDAT 6
1204.00 CGR      MOVE *BLANK  @MIDAT 8
1205.00 CGR      MOVE *@MVAL  @MPPMT 7
1206.00 CGR      MOVE *@MUL  @MPPMT 7
1207.00 CGR      MOVE *@MONE  @MSEP 7
1208.00 CGR      MOVE ' '      @MSEPT 1
1209.00 CGR      CALL 'X0028  '      99
1210.00 C*
1211.00 CGR      PARM      @MIDAT
1212.00 CGR      PARM      @MIDAT
1213.00 CGR      PARM      @MPPMT
1214.00 CGR      PARM      @MPPMT
1215.00 CGR      PARM      @MSEP
1216.00 CGR      PARM      @MSEPT
1217.00 CGR      @MSEPT  @MSEPT  @MSEPT
1218.00 CGR      @MSEPT  IPFRQ '1'
1219.00 CGR      MOVE '1'      @MX,04
1220.00 CGR      SETON      4592
1221.00 CGR      END
1222.00 CGR      END
1223.00 C*
-----
1224.00 C*
1225.00 C*      Scrub and edit - Item ID
1226.00 C*
1227.00 CGR      NOVSRDAXIT  @M
1228.00 CGR      @M      A
1229.00 C*      Convert to numeric
1230.00 CGR      MOVE @M      @M@PD
1231.00 CGR      MOVE @M      @M@TD
1232.00 CGR      @M      A
1233.00 C*      Adjust for display
1234.00 CGR      MOVE @M@M  QXXIT
1235.00 C*
1236.00 C*      Set default value - Item ID
1237.00 C*
1238.00 CGR      V@M      IPFRQ *BLANK
1239.00 CGR      D@M      ANDRQ*BLANK
1240.00 CGR      NOVSRD@M      @M
1241.00 CGR      @M      C0012

```

```

1242.00 C*          -----
1243.00 CSR          MOVE P&XIT          W&GPD
1244.00 CSR          MOVE Q&XIT          W&CTD
1245.00 CSR          EXGR C00151
1246.00 C*          -----
1247.00 CSR          MOVE #NUMBER          Q&XIT
1248.00 CSR          END
1249.00 C*
1250.00 C*      Edit upper and lower range - Item ID
1251.00 C*
1252.00 CSR          L&XIT          IPSE *BLANK
1253.00 CSR          MOVE *BLANK          X&XIT 15
1254.00 CSR          MOVE '1'          S&RTST
1255.00 CSR          MOVRLQ&XIT
1256.00 CSR          X&XIT          IPGR L&XIT
1257.00 CSR          X&XIT          ANGLRUM&XIT
1258.00 CSR          MOVE ' '          S&RTST
1259.00 CSR          END
1260.00 CSR          S&RTST          IPRQ '1'
1261.00 CSR          MOVE '1'          @MK,07          4193
1262.00 CSR          SRTON
1263.00 CSR          END
1264.00 CSR          END
1265.00 C*-----
1266.00 C*
1267.00 C*      Scrub and edit - Quantity - on Hand
1268.00 C*
1269.00 CSR          MOVEAVD&QT          @HM
1270.00 CSR          EXGR C0012
1271.00 C*          -----
1272.00 CSR          MOVE P&XQT          W&GPD
1273.00 CSR          MOVE Q&XQT          W&C&D
1274.00 CSR          EXGR C00151
1275.00 C*          -----
1276.00 CSR          MOVE #NUMBER          Q&XQT
1277.00 C*
1278.00 C*      Set default value - Quantity - On Hand
1279.00 C*
1280.00 CSR          V&XQT          IPRQ *BLANK
1281.00 CSR          D&XQT          ANGR*BLANK
1282.00 CSR          MOVRAV&QT          @HM
1283.00 CSR          EXGR C0012
1284.00 C*          -----
1285.00 CSR          MOVE P&XQT          W&GPD
1286.00 CSR          MOVE Q&XQT          W&C&D
1287.00 CSR          EXGR C00151
1288.00 C*          -----
1289.00 CSR          MOVE #NUMBER          Q&XQT
1290.00 CSR          END
1291.00 C*
1292.00 C*      Edit upper and lower range - Quantity - On Hand
1293.00 C*
1294.00 CSR          L&XQT          IPSE *BLANK
1295.00 CSR          MOVE *BLANK          X&XQT 15
1296.00 CSR          MOVE '1'          S&RTST 1
1297.00 CSR          MOVRLQ&XQT
1298.00 CSR          X&XQT          IPGR L&XQT
1299.00 CSR          X&XQT          ANGLRUM&XQT
1300.00 CSR          MOVE ' '          S&RTST
1301.00 CSR          END
1302.00 CSR          S&RTST          IPRQ '1'
1303.00 CSR          MOVE '1'          @MK,07          4693
1304.00 CSR          SRTON
1305.00 CSR          END
1306.00 CSR          END
1307.00 C*-----
1308.00 C*
1309.00 C*      Scrub and edit - Item Type
1310.00 C*
1311.00 CSR          MOVRLV&XTY          Q&XTY
1312.00 C*
1313.00 C*      Set default value - Item Type
1314.00 C*
1315.00 CSR          Q&XTY          IPRQ *BLANK
1316.00 CSR          D&XTY          IPSE *BLANK
1317.00 CSR          MOVRAV&XTY          @40
1318.00 CSR          MOVRA#40          Q&XTY

```

Default value from Data Dictionary

Upper and lower ranges from Data Dictionary

```

1319.00 CBR      @40,1  IPFQ ' ' ' '
1320.00 CBR      MOVE ' ' '
1321.00 CBR      I-ADD2      @40,1
1322.00 CBR      @M      DCMLR40      MM
1323.00 CBR      @40,MM  IPFQ ' ' ' '
1324.00 CBR      MOVE ' ' '
1325.00 CBR      END      @40,MM
1326.00 CBR      ADD 1      MM
1327.00 CBR      END
1328.00 CBR      NOVBAR@40,2  QXXTY
1329.00 CBR      END
1330.00 CBR      END
1331.00 CBR      END
1332.00 C*
1333.00 C*      Edit allowed values - Item Type
1334.00 C*
1335.00 CBR      @XNTY  IPFR *BLANK
1336.00 CBR      @XNTY  IPFQ *'SR'
1337.00 CBR      QXXTY  ANDRQ*BLANK
1338.00 CBR      MOVE '1'      @MK,02  4493
1339.00 CBR      SETON
1340.00 CBR      ELSE
1341.00 CBR      NOVBAR@XNTY      @40
1342.00 CBR      MOVE *RIVAL      @AV
1343.00 CBR      EGGR C99T
1344.00 C*
1345.00 CBR      MOVE ' '      @ERTGT 1
1346.00 CBR      MOVE *BLANK      @WRK10 10
1347.00 CBR      NOVELQXXTY      @WRK10
1348.00 CBR      @AV,1      IPFR *RIVAL
1349.00 CBR      @WRK10  LCKUP@AV      91
1350.00 CBR      *IN91  IPFQ '0'
1351.00 CBR      MOVE '1'      @ERTGT
1352.00 CBR      END
1353.00 CBR      @ERTGT  IPFQ '1'
1354.00 CBR      MOVE '1'      @MK,07  4493
1355.00 CBR      SETON
1356.00 CBR      END
1357.00 CBR      END
1358.00 CBR      END
1359.00 CBR      END
1360.00 C*
1361.00 C*      Edit upper and lower range - Item Type
1362.00 C*
1363.00 CBR      L@NTY  IPFR *BLANK
1364.00 CBR      MOVE '1'      @ERTGT
1365.00 CBR      QXXTY  IPFR L@NTY
1366.00 CBR      QXXTY  ANDRQL@NTY
1367.00 CBR      MOVE ' '      @ERTGT
1368.00 CBR      END
1369.00 CBR      @ERTGT  IPFQ '1'
1370.00 CBR      MOVE '1'      @MK,07  4493
1371.00 CBR      SETON
1372.00 CBR      END
1373.00 CBR      END
1374.00 C*
1375.00 C*      Edit from User Defined Codes - Item Type
1376.00 C*
1377.00 CBR      @XNTY  IPFR *BLANK
1378.00 CBR      CLEAR1000SU
1379.00 CBR      NOVEL@XNTY      WUGY
1380.00 CBR      MOVE @XNTY      WURT
1381.00 CBR      MOVE QXXTY      WUXY
1382.00 CBR      CALL 'X0005'      91
1383.00 C*
1384.00 CBR      @XNTY  IPFR *BLANK
1385.00 CBR      MOVE '1'      @MK,09  4493
1386.00 CBR      MOVE '1'
1387.00 CBR      SETON
1388.00 CBR      END
1389.00 CBR      END
1390.00 C*
1391.00 C*
1392.00 C*      Scrub and edit - Item Unit of Measure
1393.00 C*
1394.00 CBR      NOVELVDXIM      QXXUM
1395.00 C*

```

1396.00	C*	Set default value - Item Unit of Measure			
1397.00	C*				
1398.00	CSR	QXXUM	IPRQ 'BLANK		
1399.00	CSR	MAXUM	IPRQ 'BLANK		
1400.00	CSR		MOVEMAXUM	840	
1401.00	CSR		MOVEM840	QXXUM	
1402.00	CSR	840,1	IPRQ '1'		
1403.00	CSR		MOVE '1'	840,1	
1404.00	CSR		Z-ADD2	8M	
1405.00	CSR	8M	DOMLR840		
1406.00	CSR	840,8M	IPRQ '1'		
1407.00	CSR		MOVE '1'	840,8M	
1408.00	CSR		END		
1409.00	CSR		ADD 1	8M	
1410.00	CSR		END		
1411.00	CSR		MOVEM840,2	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	MAXUM	IPRQ 'BLANK		
1419.00	CSR	MAXUM	IPRQ 'NR'		
1420.00	CSR	QXXUM	ANDRQ'BLANK		
1421.00	CSR		MOVE '1'	8MX,03	
1422.00	CSR		ENTON	4792	
1423.00	CSR		ELGE		
1424.00	CSR		MOVEMMAXUM	840	
1425.00	CSR		MOVE 'RIVAL	8AV	
1426.00	CSR		EXRQ C997		
1427.00	C*		-----		
1428.00	CSR		MOVE '1'	8ERTST 1	
1429.00	CSR		MOVE 'BLANK	8NRK10 10	
1430.00	CSR		MOVEMQXXUM	8NRK10	
1431.00	CSR	8AV,1	IPRQ 'RIVAL		
1432.00	CSR	8NRK10	LOXUP8AV	81	
1433.00	CSR	'IN91	IPRQ '0'		
1434.00	CSR		MOVE '1'	8R-TST	
1435.00	CSR		END		
1436.00	CSR	8ERTST	IPRQ '1'		
1437.00	CSR		MOVE '1'	8MX,07	
1438.00	CSR		ENTON	4792	
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	LAXUM	IPRQ 'BLANK		
1447.00	CSR		MOVE '1'	8ERTST	
1448.00	CSR	QXXUM	IPRQ LAXUM		
1449.00	CSR	QXXUM	ANDLMAXUM		
1450.00	CSR		MOVE '1'	8ERTST	
1451.00	CSR		END		
1452.00	CSR	8ERTST	IPRQ '1'		
1453.00	CSR		MOVE '1'	8MX,07	
1454.00	CSR		ENTON	4792	
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	MAXUM	IPRQ 'BLANK		
1461.00	CSR		CLGRN100050		
1462.00	CSR		MOVEMMAXUM	8URY	
1463.00	CSR		MOVE MAXUM	8URT	
1464.00	CSR		MOVE QXXUM	8URY	
1465.00	CSR		CALL 'X0005'	81	
1466.00	C*		-----		
1467.00	CSR		PARM	100050	
1468.00	CSR	8URT	IPRQ '1'		
1469.00	CSR		MOVE '1'	8MX,09	
1470.00	CSR		ENTON	4792	
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 CBR          MOVELV0X001          QXX001
1478.00 C*
1479.00 C*      Set default value - Item Category Code 001
1480.00 C*
1481.00 CBR          QXX001  IPFK *BLANK
1482.00 CBR          DXX001  IPFK *BLANK
1483.00 CBR          MOVELV0X001          @40
1484.00 CBR          MOVEW@40          QXX001
1485.00 CBR          @40,1  IPFK '''
1486.00 CBR          MOVE ' '          @40,1
1487.00 CBR          I-ADD2          NM
1488.00 CBR          NM          DCMLE@40
1489.00 CBR          @40,NM  IPFK '''
1490.00 CBR          MOVE ' '          @40,NM
1491.00 CBR          END
1492.00 CBR          ADD 1          NM
1493.00 CBR          END
1494.00 CBR          MOVEW@40,2          QXX001
1495.00 CBR          END
1496.00 CBR          END
1497.00 CBR          END
1498.00 C*
1499.00 C*      Edit allowed values - Item Category Code 001
1500.00 C*
1501.00 CBR          AX001  IPFK *BLANK
1502.00 CBR          AX001  IPFK *NR'
1503.00 CBR          QXX001  ANDRQ*BLANK
1504.00 CBR          MOVE '1'          @MK,02
1505.00 CBR          STON          4893
1506.00 CBR          ELSE
1507.00 CBR          MOVEWAX001          @40
1508.00 CBR          MOVE *RIVAL          @AV
1509.00 CBR          RGR C997
1510.00 C*      ----
1511.00 CBR          MOVE ' '          $ENTST 1
1512.00 CBR          MOVE *BLANK          $NRK10 10
1513.00 CBR          MOVELVQXX001          $NRK10
1514.00 CBR          @AV,1  IPFK *RIVAL
1515.00 CBR          $NRK10  LCHT@AV          91
1516.00 CBR          *IN91  IPFK '0'
1517.00 CBR          MOVE '1'          $ENTST
1518.00 CBR          END
1519.00 CBR          $ENTST  IPFK '1'
1520.00 CBR          MOVE '1'          @MK,07
1521.00 CBR          STON          4893
1522.00 CBR          END
1523.00 CBR          END
1524.00 CBR          END
1525.00 CBR          END
1526.00 C*
1527.00 C*      Edit upper and lower range - Item Category Code 001
1528.00 C*
1529.00 CBR          LXX001  IPFK *BLANK
1530.00 CBR          MOVE '1'          $ENTST
1531.00 CBR          QXX001  IPFK LXX001
1532.00 CBR          QXX001  ANDLRUX001
1533.00 CBR          MOVE ' '          $ENTST
1534.00 CBR          END
1535.00 CBR          $ENTST  IPFK '1'
1536.00 CBR          MOVE '1'          @MK,07
1537.00 CBR          STON          4893
1538.00 CBR          END
1539.00 CBR          END
1540.00 C*
1541.00 C*      Edit from User Defined Codes - Item Category Code 001
1542.00 C*
1543.00 CBR          BXX001  IPFK *BLANK
1544.00 CBR          CLEAR10005U
1545.00 CBR          MOVELVBXX001          #UEY
1546.00 CBR          MOVE PXX001          #URT
1547.00 CBR          MOVE QXX001          #UEY
1548.00 CBR          CALL 'X0005'          91
1549.00 C*      ----

```

```

1550.00 CGR      FARM      100050
1551.00 CGR      #USER  IPRQ '1'
1552.00 CGR      MOVE '1'      @MK,09
1553.00 CGR      SRTON
1554.00 CGR      END      4993
1555.00 CGR      END
1556.00 C*-----
1557.00 C*
1558.00 C*      Scrub and edit - Item Category Code 002
1559.00 C*
1560.00 CGR      MOVELVX002      QXX002
1561.00 C*
1562.00 C*      Set default value - Item Category Code 002
1563.00 C*
1564.00 CGR      QXX002      IPRQ *BLANK
1565.00 CGR      @MK002      IPNE *BLANK
1566.00 CGR      MOVEVA@X002      @40
1567.00 CGR      MOVEVA@40      QXX002
1568.00 CGR      @40,1      IPRQ ' '
1569.00 CGR      MOVE ' '      @40,1
1570.00 CGR      X-ADD2      @M
1571.00 CGR      @M      DOWRL@40
1572.00 CGR      @40,@M      IPRQ ' '
1573.00 CGR      MOVE ' '      @40,@M
1574.00 CGR      END
1575.00 CGR      ADD 1      @M
1576.00 CGR      END
1577.00 CGR      MOVEVA@40,2      QXX002
1578.00 CGR      END
1579.00 CGR      END
1580.00 CGR      END
1581.00 C*
1582.00 C*      Edit allowed values - Item Category Code 002
1583.00 C*
1584.00 CGR      @MK002      IPNE *BLANK
1585.00 CGR      @MK002      IPRQ *SE'
1586.00 CGR      QXX002      ANDRQ*BLANK
1587.00 CGR      MOVE '1'      @MK,02
1588.00 CGR      SRTON      4993
1589.00 CGR      ELSE
1590.00 CGR      MOVEVA@X002      @40
1591.00 CGR      MOVE *RIVAL      @AV
1592.00 CGR      ELSE C997
1593.00 C*-----
1594.00 CGR      MOVE ' '      SRTST
1595.00 CGR      MOVE *BLANK      @MK10 10
1596.00 CGR      MOVELVQXX002      @MK10
1597.00 CGR      @AV,1      IPNE *RIVAL
1598.00 CGR      @MK10      LORUP@AV      91
1599.00 CGR      *IN@1      IPRQ '0'
1600.00 CGR      MOVE '1'      SRTST
1601.00 CGR      END
1602.00 CGR      SRTST      IPRQ '1'
1603.00 CGR      MOVE '1'      @MK,07
1604.00 CGR      SRTON      4993
1605.00 CGR      END
1606.00 CGR      END
1607.00 CGR      END
1608.00 CGR      END
1609.00 C*
1610.00 C*      Edit upper and lower range - Item Category Code 002
1611.00 C*
1612.00 CGR      L@X002      IPNE *BLANK
1613.00 CGR      MOVE '1'      SRTST
1614.00 CGR      QXX002      IPRQ L@X002
1615.00 CGR      QXX002      ANDL@X002
1616.00 CGR      MOVE ' '      SRTST
1617.00 CGR      END
1618.00 CGR      SRTST      IPRQ '1'
1619.00 CGR      MOVE '1'      @MK,07
1620.00 CGR      SRTON      4993
1621.00 CGR      END
1622.00 CGR      END
1623.00 C*
1624.00 C*      Edit from User Defined Codes - Item Category Code 002
1625.00 C*
1626.00 CGR      @MK002      IPNE *BLANK

```

```

1627.00 CDR          CLEAR10005U
1628.00 CDR          MOVELWGX002          WURY
1629.00 CDR          MOVE WGX002          WURY
1630.00 CDR          MOVE QXX002          WURY
1631.00 CDR          CALL 'X0005'          01
1632.00 C*          -----
1633.00 CDR          PARM          10005U
1634.00 CDR          #WERR          IFBQ '1'
1635.00 CDR          MOVE '1'          WGX,09
1636.00 CDR          GETON          4992
1637.00 CDR          END
1638.00 CDR          END
1639.00 C*          -----
1640.00 C*
1641.00 C*          Scrub and edit - Item Category Code 002
1642.00 C*
1643.00 CDR          MOVELVGX002          QXX002
1644.00 C*
1645.00 C*          Set default value - Item Category Code 002
1646.00 C*
1647.00 CDR          QXX002          IFBQ 'BLANK'
1648.00 CDR          WGX002          IFBQ 'BLANK'
1649.00 CDR          MOVEWGX002          @40
1650.00 CDR          MOVEW@40          QXX002
1651.00 CDR          @40,1          IFBQ
1652.00 CDR          MOVE ' '          @40,1
1653.00 CDR          2-ADD2          WM
1654.00 CDR          WM          DCMLE@40
1655.00 CDR          @40,WM          IFBQ ' '
1656.00 CDR          MOVE ' '          @40,WM
1657.00 CDR          END
1658.00 CDR          ADD 1          WM
1659.00 CDR          END
1660.00 CDR          MOVEW@40,2          QXX002
1661.00 CDR          END
1662.00 CDR          END
1663.00 CDR          END
1664.00 C*
1665.00 C*          Edit allowed values - Item Category Code 002
1666.00 C*
1667.00 CDR          WGX002          IFBQ 'BLANK'
1668.00 CDR          WGX002          IFBQ 'SR'
1669.00 CDR          QXX002          ANDRQ'BLANK'
1670.00 CDR          MOVE '1'          WGX,02
1671.00 CDR          GETON          5092
1672.00 CDR          ELGR
1673.00 CDR          MOVEW@WGX002          @40
1674.00 CDR          MOVE 'HIVAL'          @AV
1675.00 CDR          EXOR C997
1676.00 C*          -----
1677.00 CDR          MOVE ' '          $ENTST
1678.00 CDR          MOVE 'BLANK'          $WFK10 10
1679.00 CDR          MOVELVGX002          $WFK10
1680.00 CDR          @AV,1          IFBQ 'HIVAL'
1681.00 CDR          $WFK10          LCHTOP@AV          01
1682.00 CDR          *IN@1          IFBQ '0'
1683.00 CDR          MOVE '1'          $ENTST
1684.00 CDR          END
1685.00 CDR          $ENTST          IFBQ '1'
1686.00 CDR          MOVE '1'          WGX,07
1687.00 CDR          GETON          5092
1688.00 CDR          END
1689.00 CDR          END
1690.00 CDR          END
1691.00 CDR          END
1692.00 C*
1693.00 C*          Edit upper and lower range - Item Category Code 002
1694.00 C*
1695.00 CDR          L@X002          IFBQ 'BLANK'
1696.00 CDR          MOVE '1'          $ENTST
1697.00 CDR          QXX002          IFBQ L@X002
1698.00 CDR          QXX002          ANDL@WGX002          $ENTST
1699.00 CDR          MOVE ' '
1700.00 CDR          END
1701.00 CDR          $ENTST          IFBQ '1'
1702.00 CDR          MOVE '1'          WGX,07
1703.00 CDR          GETON          5092

```

```

1704.00 CGR          END
1705.00 CGR          END
1706.00 C*
1707.00 C*      Edit from User Defined Codes - Item Category Code 003
1708.00 C*
1709.00 CGR          BAX003  IPNE 'BLANK
1710.00 CGR          CLEAR100050
1711.00 CGR          MOVEL@BAX003          WURY
1712.00 CGR          MOVE BAX003          WURT
1713.00 CGR          MOVE QXX003          WURY
1714.00 CGR          CALL 'X0005'          91
1715.00 C*          -----
1716.00 CGR          BAX003  IPNE '1'          100050
1717.00 CGR          #WERR  IPFQ '1'
1718.00 CGR          MOVE '1'          @MX,09
1719.00 CGR          SNTON
1720.00 CGR          END          5093
1721.00 CGR          END
1722.00 C*-----
1723.00 Ct
1724.00 C*      Scrub and edit - Item Category Code 004
1725.00 C*
1726.00 CGR          MOVELVX004          QXX004
1727.00 C*
1728.00 Ct      Set default value - Item Category Code 004
1729.00 C*
1730.00 CGR          QXX004  IPFQ 'BLANK
1731.00 CGR          BAX004  IPNE 'BLANK
1732.00 CGR          MOVEV@BAX004          @40
1733.00 CGR          MOVEV@40          QXX004
1734.00 CGR          @40,1  IPFQ ' '
1735.00 CGR          MOVE ' '          @40,1
1736.00 CGR          Z-ADD2          @M
1737.00 CGR          @M      D@ML@40
1738.00 CGR          @40,@M  IPFQ ' '
1739.00 CGR          MOVE ' '          @40,@M
1740.00 CGR          END
1741.00 CGR          ADD 1          @M
1742.00 CGR          END
1743.00 CGR          MOVEV@40,2          QXX004
1744.00 CGR          END
1745.00 CGR          END
1746.00 CGR          END
1747.00 C*
1748.00 C*      Edit allowed values - Item Category Code 004
1749.00 C*
1750.00 CGR          BAX004  IPNE 'BLANK
1751.00 CGR          BAX004  IPFQ 'SR'
1752.00 CGR          QXX004  AND@Q'BLANK
1753.00 CGR          MOVE '1'          @MX,03
1754.00 CGR          SNTON          5193
1755.00 CGR          ELSE
1756.00 CGR          MOVEV@BAX004          @40
1757.00 CGR          MOVE 'RIVAL          @AV
1758.00 CGR          EXAR C997
1759.00 C*          -----
1760.00 CGR          MOVE ' '          @ERTWT 1
1761.00 CGR          MOVE 'BLANK          @WRK10 10
1762.00 CGR          MOVEL@QXX004          @WRK10
1763.00 CGR          @AV,1  IPNE 'RIVAL
1764.00 CGR          @WRK10  L@KUP@AV          91
1765.00 CGR          *IN@1  IPFQ '0'
1766.00 CGR          MOVE '1'          @ERTWT
1767.00 CGR          END
1768.00 CGR          @ERTWT  IPFQ '1'
1769.00 CGR          MOVE '1'          @MX,07
1770.00 CGR          SNTON          5193
1771.00 CGR          END
1772.00 CGR          END
1773.00 CGR          END
1774.00 CGR          END
1775.00 C*
1776.00 C*      Edit upper and lower range - Item Category Code 004
1777.00 C*
1778.00 CGR          L@004  IPNE 'BLANK          @ERTWT
1779.00 CGR          MOVE '1'
1780.00 CGR          QXX004  IPGE L@X004

```

```

1781.00 CGM      QXX004  AMDLRUX004
1782.00 CGM      QXX004  ENDD
1783.00 CGM      QXX004  ENDD
1784.00 CGM      QXX004  IPFRQ '1'
1785.00 CGM      QXX004  MOVE '1'          QMX,07
1786.00 CGM      QXX004  SETON
1787.00 CGM      QXX004  ENDD
1788.00 CGM      QXX004  ENDD
1789.00 C*
1790.00 C*      Edit from User Defined Codes - Item Category Code 004
1791.00 C*
1792.00 CGM      RAX004  IPFRQ 'BLANK'
1793.00 CGM      RAX004  CLEAR I0005U
1794.00 CGM      RAX004  MOVE RAX004      WUY
1795.00 CGM      RAX004  MOVE QXX004      WUY
1796.00 CGM      RAX004  MOVE QXX004      WUY
1797.00 CGM      RAX004  CALL 'X0005'      Q1
1798.00 C*
1799.00 CGM      RAX004  PARM          I0005U
1800.00 CGM      RAX004  IPFRQ '1'
1801.00 CGM      RAX004  MOVE '1'          QMX,09
1802.00 CGM      RAX004  SETON
1803.00 CGM      RAX004  ENDD
1804.00 CGM      RAX004  ENDD
1805.00 C*
-----
1806.00 C*
1807.00 C*      Scrub and edit - Item Category Code 005
1808.00 C*
1809.00 CGM      NOVELVIX005  QXX005
1810.00 C*
1811.00 C*      Set default value - Item Category Code 005
1812.00 C*
1813.00 CGM      QXX005  IPFRQ 'BLANK'
1814.00 CGM      QXX005  IPFRQ 'BLANK'
1815.00 CGM      QXX005  MOVE RAX005      Q40
1816.00 CGM      QXX005  MOVE RAX005      QXX005
1817.00 CGM      Q40,1  IPFRQ ''
1818.00 CGM      Q40,1  MOVE ' '          Q40,1
1819.00 CGM      Q40,1  I-ADD2
1820.00 CGM      Q40,1  DOWLEQ40      QM
1821.00 CGM      Q40,1  IPFRQ
1822.00 CGM      Q40,1  MOVE ' '          Q40,1
1823.00 CGM      Q40,1  ENDD
1824.00 CGM      Q40,1  ADD 1          QM
1825.00 CGM      Q40,1  ENDD
1826.00 CGM      Q40,1  MOVE RAX005      QXX005
1827.00 CGM      Q40,1  ENDD
1828.00 CGM      Q40,1  ENDD
1829.00 CGM      Q40,1  ENDD
1830.00 C*
1831.00 C*      Edit allowed values - Item Category Code 005
1832.00 C*
1833.00 CGM      QXX005  IPFRQ 'BLANK'
1834.00 CGM      QXX005  IPFRQ 'NR'
1835.00 CGM      QXX005  AMDRQ 'BLANK'
1836.00 CGM      QXX005  MOVE '1'          QMX,02
1837.00 CGM      QXX005  SETON
1838.00 CGM      QXX005  ENDD
1839.00 CGM      QXX005  MOVE RAX005      Q40
1840.00 CGM      QXX005  MOVE 'RIVAL'      QAV
1841.00 CGM      QXX005  ENDD
1842.00 C*
1843.00 CGM      QXX005  MOVE ' '          QXX005
1844.00 CGM      QXX005  MOVE 'BLANK'      QNRK10 10
1845.00 CGM      QXX005  MOVE QXX005      QNRK10
1846.00 CGM      QAV,1  IPFRQ 'RIVAL'
1847.00 CGM      QNRK10  LOWTPARAV
1848.00 CGM      QNRK10  IPFRQ '0'
1849.00 CGM      QNRK10  MOVE '1'          QNRK10
1850.00 CGM      QNRK10  ENDD
1851.00 CGM      QNRK10  IPFRQ '1'
1852.00 CGM      QNRK10  MOVE '1'          QMX,07
1853.00 CGM      QNRK10  SETON
1854.00 CGM      QNRK10  ENDD
1855.00 CGM      QNRK10  ENDD
1856.00 CGM      QNRK10  ENDD
1857.00 CGM      QNRK10  ENDD

```

```

1958.00 C*
1959.00 C* Edit upper and lower range - Item Category Code 005
1960.00 C*
1961.00 CGR L&X005 IPFK 'BLANK
1962.00 CGR MOVE '1' SERTST
1963.00 CGR QXX005 IPFG L&X005
1964.00 CGR QXX005 ANDLR&X005
1965.00 CGR MOVE ' ' SERTST
1966.00 CGR END
1967.00 CGR SERTST IPFQ '1'
1968.00 CGR MOVE '1' SMX,07
1969.00 CGR SRTOM 5293
1970.00 CGR END
1971.00 CGR END
1972.00 C*
1973.00 C* Edit from User Defined Codes - Item Category Code 005
1974.00 C*
1975.00 CGR R&X005 IPFK 'BLANK
1976.00 CGR CL&R&10005U
1977.00 CGR MOV&L&R&X005 WUY
1978.00 CGR MOVE R&X005 WUNT
1979.00 CGR MOVE QXX005 WUXY
1980.00 CGR CALL 'X0005' 91
1981.00 C*
1982.00 CGR F&R& 10005U
1983.00 CGR W&R&R IPFQ '1'
1984.00 CGR MOVE '1' SMX,09
1985.00 CGR SRTOM 5293
1986.00 CGR END
1987.00 CGR END
1988.00 C*-----
1989.00 CGR END005 ENDR
1990.00 C*-----
1991.00 C*
1992.00 C* Copy Common Subroutine - Currency - Translate Video Fields to Data Base
1993.00 C*
1994.00 C/COPY JDRCPY,C00151
1995.00 C*-----
1996.00 C*
1997.00 C* Copy Common Subroutine - Build Allowed Values Work Array
1998.00 C*-----
1999.00 C/COPY JDRCPY,C997
2000.00 C*-----
2001.00 C*
2002.00 C* SUBROUTINE #010 - Update Data Base
2003.00 C*-----
2004.00 C*
2005.00 C* Processing: 1. Update data base file based upon valid
2006.00 C* action codes.
2007.00 C*
2008.00 CGR #010 ENDR
2009.00 C* ----
2010.00 C*
2011.00 C* If add action, add record. Indicator value for action code is
2012.00 C* assigned in copy module C0001.
2013.00 CGR *IN21 IPFQ '1'
2014.00 CGR W&ITR192801 99
2015.00 CGR END
2016.00 C*
2017.00 C* If change action, update record.
2018.00 C*
2019.00 CGR *IN22 IPFQ '1'
2020.00 CGR UD&T192801 99
2021.00 CGR END
2022.00 C*
2023.00 C* If delete action, delete record.
2024.00 C*
2025.00 CGR *IN23 IPFQ '1'
2026.00 CGR D&L&T192801 99
2027.00 CGR END
2028.00 C*

```

```

1929.00 C*      Clear data field for next transaction
1930.00 C*
1931.00 CSM      MOVE #PCLA      @@AID
1932.00 CSM      EXEC S001
1933.00 C*
1934.00 CSM      END010      ENDR
1935.00 C*-----
1936.00 C*
1937.00 C*      SUBROUTINE S999 - Load dictionary parameters.
1938.00 C*
1939.00 C*
1940.00 CSM      S999      P999R
1941.00 C*      -----
1942.00 C*
1943.00 C*
1944.00 C*      Dictionary parameters for - Cost Center
1945.00 C*
1946.00 CSM      MOVE *BLANK      PRDTAI
1947.00 CSM      MOVEL'XCC'      PRDTAI
1948.00 CSM      CALL 'X9900R'
1949.00 C*
1950.00 CSM      PARM      19900R
1951.00 CSM      P999R      IPBQ '0'
1952.00 CSM      MOVE PRDTAT      TWXCC      1
1953.00 CSM      MOVE PRDCC      @XWCC      1
1954.00 CSM      MOVE PRDTAG      @XWCC      50
1955.00 CSM      MOVE PRDTAD      @XWCC      20
1956.00 CSM      MOVE PRDCCC      @XWCC      1
1957.00 CSM      MOVE PRDCCY      @XWCC      4
1958.00 CSM      MOVE PRDCT      @XWCC      2
1959.00 CSM      MOVE PRDVAL      @XWCC      40
1960.00 CSM      MOVE PRVAL      @XWCC      40
1961.00 CSM      MOVE PRVAL      @XWCC      40
1962.00 CSM      MOVE PRVAL      @XWCC      40
1963.00 CSM      MOVE PRVAL      @XWCC      40
1964.00 CSM      MOVE PRDWRM      @XWCC      20
1965.00 CSM      MOVE PRLE      @XWCC      1
1966.00 CSM      MOVE PRNNIX      @XWCC      20
1967.00 CSM      I-ADDL      @XWCC      110
1968.00 CSM      MOVE PRWCC      @A
1969.00 CSM      DO      @A
1970.00 CSM      MULT 10      @XWCC
1971.00 CSM      END
1972.00 CSM      END
1973.00 C*-----
1974.00 C*
1975.00 C*      Dictionary parameters for - Description
1976.00 C*
1977.00 CSM      MOVE *BLANK      PRDTAI
1978.00 CSM      MOVEL'XDS'      PRDTAI
1979.00 CSM      CALL 'X9900R'
1980.00 C*
1981.00 CSM      PARM      19900R
1982.00 CSM      P999R      IPBQ '0'
1983.00 CSM      MOVE PRDTAT      TWXDS      1
1984.00 CSM      MOVE PRDCC      @XWDS      1
1985.00 CSM      MOVE PRDTAG      @XWDS      50
1986.00 CSM      MOVE PRDTAD      @XWDS      20
1987.00 CSM      MOVE PRDCCC      @XWDS      1
1988.00 CSM      MOVE PRDCCY      @XWDS      4
1989.00 CSM      MOVE PRDCT      @XWDS      2
1990.00 CSM      MOVE PRDVAL      @XWDS      40
1991.00 CSM      MOVE PRVAL      @XWDS      40
1992.00 CSM      MOVE PRVAL      @XWDS      40
1993.00 CSM      MOVE PRVAL      @XWDS      40
1994.00 CSM      MOVE PRVAL      @XWDS      40
1995.00 CSM      MOVE PRDWRM      @XWDS      20
1996.00 CSM      MOVE PRLE      @XWDS      1
1997.00 CSM      MOVE PRNNIX      @XWDS      20
1998.00 CSM      I-ADDL      @XWDS      110
1999.00 CSM      MOVE PRWDS      @A
2000.00 CSM      DO      @A
2001.00 CSM      MULT 10      @XWDS
2002.00 CSM      END
2003.00 CSM      END
2004.00 C*-----
2005.00 C*

```

Forces clear of everything before processing next record. Simulates user pressing the Clear Screen function key.

Retrieves all of the Data Dictionary editing parameters for necessary data items used in the program and moves the information into constant fields

Data Dictionary file server

2006.00	C*	Dictionary parameters for - Date Last Ship		
2007.00	C*			
2008.00	CRB	MOVE *BLANK	PRDTAI	
2009.00	CRB	MOVE *XIT'	PRDTAI	
2010.00	CRB	CALL 'XPR00E'		01
2011.00	C*	-----		
2012.00	CRB	FARM	IG000E	
2013.00	CRB	FRMR	IFRQ '0'	
2015.00	CRB	MOVE PRDTAT	TENDT	1
2016.00	CRB	MOVE PRRC	RENDT	1
2017.00	CRB	MOVE PRDTAR	CRNDT	50
2018.00	CRB	MOVE PRDTAD	CRNDT	20
2019.00	CRB	MOVE PRDRC	PRNDT	1
2020.00	CRB	MOVE PRST	RENIT	4
2021.00	CRB	MOVE PRVAL	RENIT	2
2022.00	CRB	MOVE PRVAL	RENIT	40
2023.00	CRB	MOVE PRVAL	RENIT	40
2024.00	CRB	MOVE PRVAL	LENIT	40
2025.00	CRB	MOVE PRVAL	UNENIT	40
2026.00	CRB	MOVE PRDWR	RENIT	20
2027.00	CRB	MOVE PRLA	JENIT	1
2029.00	CRB	MOVE PRNIX	RENIT	20
2029.00	CRB	Z-ADD1	RENIT	110
2030.00	CRB	MOVE PRXIT	NA	
2031.00	CRB	DO NA		
2032.00	CRB	MULT 10	RENIT	
2033.00	CRB	END		
2034.00	CRB	END		
2035.00	C*	-----		
2036.00	C*	Dictionary parameters for - Item ID		
2037.00	C*			
2039.00	CRB	MOVE *BLANK	PRDTAI	
2040.00	CRB	MOVE *XIT'	PRDTAI	
2041.00	CRB	CALL 'XPR00E'		01
2042.00	C*	-----		
2043.00	CRB	FARM	IG000E	
2044.00	CRB	FRMR	IFRQ '0'	
2046.00	CRB	MOVE PRDTAT	TENIT	1
2047.00	CRB	MOVE PRRC	RENIT	1
2048.00	CRB	MOVE PRDTAR	CRNIT	50
2049.00	CRB	MOVE PRDTAD	CRNIT	20
2050.00	CRB	MOVE PRDRC	PRNIT	1
2051.00	CRB	MOVE PRST	RENIT	4
2052.00	CRB	MOVE PRVAL	RENIT	2
2053.00	CRB	MOVE PRVAL	RENIT	40
2054.00	CRB	MOVE PRVAL	RENIT	40
2055.00	CRB	MOVE PRVAL	LENIT	40
2056.00	CRB	MOVE PRVAL	UNENIT	40
2057.00	CRB	MOVE PRDWR	RENIT	20
2058.00	CRB	MOVE PRLA	JENIT	1
2059.00	CRB	MOVE PRNIX	RENIT	20
2060.00	CRB	Z-ADD1	RENIT	110
2061.00	CRB	MOVE PRXIT	NA	
2062.00	CRB	DO NA		
2063.00	CRB	MULT 10	RENIT	
2064.00	CRB	END		
2065.00	CRB	END		
2066.00	C*	-----		
2067.00	C*	Dictionary parameters for - Quantity - On Hand		
2069.00	C*			
2070.00	CRB	MOVE *BLANK	PRDTAI	
2071.00	CRB	MOVE *XIT'	PRDTAI	
2072.00	CRB	CALL 'XPR00E'		01
2073.00	C*	-----		
2074.00	CRB	FARM	IG000E	
2075.00	CRB	FRMR	IFRQ '0'	
2077.00	CRB	MOVE PRDTAT	TENQT	1
2078.00	CRB	MOVE PRRC	RENQT	1
2079.00	CRB	MOVE PRDTAR	CRNQT	50
2080.00	CRB	MOVE PRDTAD	CRNQT	20
2081.00	CRB	MOVE PRDRC	PRNQT	1
2082.00	CRB	MOVE PRST	RENQT	4

```

2093.00 CSR      MOVE PRRT      NEXQT      2
2094.00 CSR      MOVE PRDVAL     DEXQT      40
2095.00 CSR      MOVE PRVAL     LEXQT      40
2096.00 CSR      MOVE PRVAL     LEXQT      40
2097.00 CSR      MOVE PRVAL     UEXQT      40
2098.00 CSR      MOVE PRDMM     WEXQT      20
2099.00 CSR      MOVE PRLE     JEXQT      1
2100.00 CSR      MOVE PRMIX     MEXQT      20
2101.00 CSR      I-ADDL     NEXQT      110
2102.00 CSR      MOVE PRXQT     NA
2103.00 CSR      DO NA
2104.00 CSR      MULT 10      NEXQT
2105.00 CSR      END
2106.00 CSR      END
2107.00 C*-----
2108.00 C*
2109.00 C*      Dictionary parameters for - Item Type
2110.00 C*
2111.00 CSR      MOVE *BLANK     PRDTAI
2112.00 CSR      MOVE *XTY'     PRDTAI
2113.00 CSR      CALL 'X9900E'
2114.00 C*-----
2115.00 CSR      PARM          I9900E
2116.00 CSR      PRERR      IPRTQ '0'
2117.00 CSR      MOVE PRDTAT     TEXTY 1
2118.00 CSR      MOVE PRDC      REXTY 1
2119.00 CSR      MOVE PRDTAG     CEXTY 50
2120.00 CSR      MOVE PRDTAD     GEXTY 20
2121.00 CSR      MOVE PRDCRC     PEXTY 1
2122.00 CSR      MOVE PRDPRY     SEXTY 4
2123.00 CSR      MOVE PRRT      REXTY 2
2124.00 CSR      MOVE PRDVAL     DEXTY 40
2125.00 CSR      MOVE PRVAL     AEXTY 40
2126.00 CSR      MOVE PRVAL     LEXTY 40
2127.00 CSR      MOVE PRVAL     UEXTY 40
2128.00 CSR      MOVE PRDMM     WEXTY 20
2129.00 CSR      MOVE PRLE     JEXTY 1
2130.00 CSR      MOVE PRMIX     MEXTY 20
2131.00 CSR      I-ADDL     NEXTY 110
2132.00 CSR      MOVE PRXTY     NA
2133.00 CSR      DO NA
2134.00 CSR      MULT 10      WEXTY
2135.00 CSR      END
2136.00 C*-----
2137.00 C*
2138.00 C*      Dictionary parameters for - Item Unit of Measure
2139.00 C*
2140.00 CSR      MOVE *BLANK     PRDTAI
2141.00 CSR      MOVE *XUM'     PRDTAI
2142.00 CSR      CALL 'X9900E'
2143.00 C*-----
2144.00 CSR      PARM          I9900E
2145.00 CSR      PRERR      IPRTQ '0'
2146.00 CSR      MOVE PRDTAT     TEXUM 1
2147.00 CSR      MOVE PRDC      REXUM 1
2148.00 CSR      MOVE PRDTAG     CEXUM 50
2149.00 CSR      MOVE PRDTAD     GEXUM 20
2150.00 CSR      MOVE PRDCRC     PEXUM 1
2151.00 CSR      MOVE PRDPRY     SEXUM 4
2152.00 CSR      MOVE PRRT      REXUM 2
2153.00 CSR      MOVE PRDVAL     DEXUM 40
2154.00 CSR      MOVE PRVAL     AEXUM 40
2155.00 CSR      MOVE PRVAL     LEXUM 40
2156.00 CSR      MOVE PRVAL     UEXUM 40
2157.00 CSR      MOVE PRDMM     WEXUM 20
2158.00 CSR      MOVE PRLE     JEXUM 1
2159.00 CSR      MOVE PRMIX     MEXUM 20
2160.00 CSR      I-ADDL     NEXUM 110
2161.00 CSR      MOVE PRXUM     NA
2162.00 CSR      DO NA
2163.00 CSR      MULT 10      WEXUM
2164.00 CSR      END
2165.00 C*

```

```

2160.00 C*
2161.00 C* Dictionary parameters for - Item Category Code 001
2162.00 C*
2163.00 CSR MOVE *BLANK PRDTAI
2164.00 CSR MOVEL 'X001' PRDTAI
2165.00 CSR CALL 'X9900E' 91
2166.00 C* -----
2167.00 CSR FARM 19900E
2168.00 CSR PRRR IPRQ '0'
2170.00 CSR MOVE PRDTAT T99001 1
2171.00 CSR MOVE PRRC R99001 1
2172.00 CSR MOVE PRDTAR C99001 50
2173.00 CSR MOVE PRDTAD G99001 20
2174.00 CSR MOVE PRCDRC F99001 1
2175.00 CSR MOVELPRSY G99001 4
2176.00 CSR MOVE PRAT R99001 2
2177.00 CSR MOVE PRVAL L99001 40
2178.00 CSR MOVE PRVAL L99001 40
2179.00 CSR MOVE PRVAL L99001 40
2180.00 CSR MOVE PRVAL U99001 40
2181.00 CSR MOVE PRDWR H99001 30
2182.00 CSR MOVE PRLA J99001 1
2183.00 CSR MOVE PRSNIX H99001 20
2184.00 CSR S-ADD1 H99001 110
2185.00 CSR MOVE PRX001 NA
2186.00 CSR DO NA
2187.00 CSR MULT 10 H99001
2188.00 CSR END
2189.00 CSR END
2190.00 C* -----
2191.00 C*
2192.00 C* Dictionary parameters for - Item Category Code 002
2193.00 C*
2194.00 CSR MOVE *BLANK PRDTAI
2195.00 CSR MOVEL 'X002' PRDTAI
2196.00 CSR CALL 'X9900E' 91
2197.00 C* -----
2198.00 CSR FARM 19900E
2199.00 CSR PRRR IPRQ '0'
2201.00 CSR MOVE PRDTAT T99002 1
2202.00 CSR MOVE PRRC R99002 1
2203.00 CSR MOVE PRDTAR C99002 50
2204.00 CSR MOVE PRDTAD G99002 20
2205.00 CSR MOVE PRCDRC F99002 1
2206.00 CSR MOVELPRSY G99002 4
2207.00 CSR MOVE PRAT R99002 2
2208.00 CSR MOVE PRVAL L99002 40
2209.00 CSR MOVE PRVAL L99002 40
2210.00 CSR MOVE PRVAL L99002 40
2211.00 CSR MOVE PRVAL U99002 40
2212.00 CSR MOVE PRDWR H99002 30
2213.00 CSR MOVE PRLA J99002 1
2214.00 CSR MOVE PRSNIX H99002 20
2215.00 CSR S-ADD1 H99002 110
2216.00 CSR MOVE PRX002 NA
2217.00 CSR DO NA
2218.00 CSR MULT 10 H99002
2219.00 CSR END
2220.00 CSR END
2221.00 C*
2222.00 C*
2223.00 C* Dictionary parameters for - Item Category Code 003
2224.00 C*
2225.00 CSR MOVE *BLANK PRDTAI
2226.00 CSR MOVEL 'X003' PRDTAI
2227.00 CSR CALL 'X9900E' 91
2228.00 C* -----
2229.00 CSR FARM 19900E
2230.00 CSR PRRR IPRQ '0'
2232.00 CSR MOVE PRDTAT T99003 1
2233.00 CSR MOVE PRRC R99003 1
2234.00 CSR MOVE PRDTAR C99003 50
2235.00 CSR MOVE PRDTAD G99003 20
2236.00 CSR MOVE PRCDRC F99003 1

```

```

2237.00 CBR          MOVE PRST          BEX002  4
2238.00 CBR          MOVE PRST          BEX002  2
2239.00 CBR          MOVE PRVAL        DEX002 40
2240.00 CBR          MOVE PRVAL        AEX002 40
2241.00 CBR          MOVE PRVAL        LEX002 40
2242.00 CBR          MOVE PRVAL        UXEX02 40
2243.00 CBR          MOVE PRDMM        WEX002 30
2244.00 CBR          MOVE PRLE         JEX002  1
2245.00 CBR          MOVE PRSNIX       MEX002 20
2246.00 CBR          I-ADD1            WEX002 110
2247.00 CBR          MOVE PRX002       NA
2248.00 CBR          DO NA
2249.00 CBR          MULT 10           WEX002
2250.00 CBR          END
2251.00 CBR          END
2252.00 C*-----
2253.00 C*
2254.00 C*      Dictionary parameters for - Item Category Code 004
2255.00 C*
2256.00 CBR          MOVE *BLANK        PRDTA1
2257.00 CBR          MOVE 'X004'        PRDTA1
2258.00 CBR          CALL 'X09000E'
2259.00 C*-----
2260.00 CBR          PARM                I9900E
2261.00 CBR          PRERR             IFRQ '0'
2262.00 CBR          MOVE PRDTAT        TEX004  1
2263.00 CBR          MOVE PRFC         BEX004  1
2264.00 CBR          MOVE PRDTAG        OEX004 50
2265.00 CBR          MOVE PRDTAD        GEX004 20
2266.00 CBR          MOVE PRCDRC        FEX004  1
2267.00 CBR          MOVE PRCDRC        FEX004  1
2268.00 CBR          MOVE PRST          BEX004  4
2269.00 CBR          MOVE PRST          BEX004  2
2270.00 CBR          MOVE PRVAL        DEX004 40
2271.00 CBR          MOVE PRVAL        AEX004 40
2272.00 CBR          MOVE PRVAL        LEX004 40
2273.00 CBR          MOVE PRVAL        UXEX04 40
2274.00 CBR          MOVE PRDMM        WEX004 30
2275.00 CBR          MOVE PRLE         JEX004  1
2276.00 CBR          MOVE PRSNIX       MEX004 20
2277.00 CBR          I-ADD1            WEX004 110
2278.00 CBR          MOVE PRX004       NA
2279.00 CBR          DO NA
2280.00 CBR          MULT 10           WEX004
2281.00 CBR          END
2282.00 CBR          END
2283.00 C*-----
2284.00 C*
2285.00 C*      Dictionary parameters for - Item Category Code 005
2286.00 C*
2287.00 CBR          MOVE *BLANK        PRDTA1
2288.00 CBR          MOVE 'X005'        PRDTA1
2289.00 CBR          CALL 'X09000E'
2290.00 C*-----
2291.00 CBR          PARM                I9900E
2292.00 CBR          PRERR             IFRQ '0'
2293.00 CBR          MOVE PRDTAT        TEX005  1
2294.00 CBR          MOVE PRFC         BEX005  1
2295.00 CBR          MOVE PRDTAG        OEX005 50
2296.00 CBR          MOVE PRDTAD        GEX005 20
2297.00 CBR          MOVE PRCDRC        FEX005  1
2298.00 CBR          MOVE PRCDRC        FEX005  1
2299.00 CBR          MOVE PRST          BEX005  4
2300.00 CBR          MOVE PRST          BEX005  2
2301.00 CBR          MOVE PRVAL        DEX005 40
2302.00 CBR          MOVE PRVAL        AEX005 40
2303.00 CBR          MOVE PRVAL        LEX005 40
2304.00 CBR          MOVE PRVAL        UXEX05 40
2305.00 CBR          MOVE PRDMM        WEX005 30
2306.00 CBR          MOVE PRLE         JEX005  1
2307.00 CBR          MOVE PRSNIX       MEX005 20
2308.00 CBR          I-ADD1            WEX005 110
2309.00 CBR          MOVE PRX005       NA
2310.00 CBR          DO NA
2311.00 CBR          MULT 10           WEX005
2312.00 CBR          END
2313.00 CBR          END

```

```

2214.00 C* -----
2215.00 C*
2216.00 C*   Set subroutine execution flag.
2217.00 C*
2218.00 CBR      MOVE '1'          S998      1
2219.00 C*
2220.00 CBR      END999      BRGR
2221.00 C* -----
2222.00 C*
2223.00 C*   SUBROUTINE 999 - Housekeeping
2224.00 C* -----
2225.00 C*
2226.00 C*   Processing:  1. Load video screen text.
2227.00 C*                2. Retrieve screen title data area, test
2228.00 C*                   for unauthorized access, center video
2229.00 C*                   title and move to video screen.
2230.00 C*                3. Initialize key list.
2231.00 C*                4. Load roll keys.
2232.00 C*                5. Passed parameters.
2233.00 C*                6. Load error message array.
2234.00 C*
2235.00 CBR      999      BRGR
2236.00 C*   ----
2237.00 C*
2238.00 C*   Required program parameters.
2239.00 C*
2240.00 CBR      *ENTRY  VLIST
2241.00 C*
2242.00 C*   Passed Parameter - Item ID
2243.00 C*
2244.00 CBR      PARM      #XIT      9
2245.00 C*
2246.00 C*   Move to internal reference - Item ID
2247.00 C*
2248.00 CBR      MOVE #XIT      VDXIT
2249.00 C*
2250.00 C*   Test for auto inquiry function.
2251.00 C*
2252.00 CBR      VDXIT  IPSE *BLANK
2253.00 CBR      MOVE '1'          SAUTO      1
2254.00 CBR      BND
2255.00 C* -----
2256.00 C*
2257.00 C*   Load video screen text.
2258.00 C*
2259.00 CBR      MOVE @@FILE      PKEY      10
2260.00 CBR      E-ADD@IS      PVTX      20
2261.00 C/COPIY JDCOPY,CO@C
2262.00 C* -----
2263.00 C*
2264.00 C*   Key list for - Cost Center Security
2265.00 C*
2266.00 CBR      MKY01  VLIST
2267.00 CBR      KPLD  KPLD  MNUER
2268.00 CBR      KPLD  KPLD  M@FILE
2269.00 CBR      KPLD  KPLD  M@CUT
2270.00 C* -----
2271.00 C*
2272.00 C*   Key list for - SUM Item Master File
2273.00 C*
2274.00 CBR      QXKY01  VLIST
2275.00 CBR      KPLD  KPLD  QXKIT
2276.00 C* -----
2277.00 C*
2278.00 C*   Load roll key upper and lower key values.
2279.00 C*
2280.00 CBR      *LIKE  DRPN QXKIT  S@KEY
2281.00 CBR      *LIKE  DRPN @UNRY  S@KEY
2282.00 CBR      MOVE  *LOWAL  S@KEY
2283.00 CBR      MOVE  *ALL'@'  S@KEY
2284.00 C* -----
2285.00 C*

```

Assures S998 will only be executed once

Parameters passed to program

Set auto-inquiry if information is passed

Retrieves vocabulary overrides
Only loads these VTX fields displayed on the video instead of all 144.

Composite keys are defined here

Using *LIKE more and more, especially for work fields.

```

2386.00 C*          Load error messages array.
2387.00 C*
2388.00 CBR          MOVE '0001'          SMK_01      Inv Action      Error message
2389.00 CBR          MOVE '0002'          SMK_02      Inv Key         numbers from
2390.00 CBR          MOVE '0002'          SMK_02      Inv Blanks     Data Dictionary
2391.00 CBR          MOVE '0004'          SMK_04      Inv Date
2392.00 CBR          MOVE '0005'          SMK_05      Inv Next Mbr
2393.00 CBR          MOVE '0007'          SMK_06      In Use
2394.00 CBR          MOVE '0025'          SMK_07      Inv Values
2395.00 CBR          MOVE '0026'          SMK_08      Inv MCU
2396.00 CBR          MOVE '0027'          SMK_09      Inv Desc Ttl
2397.00 CBR          MOVE '0052'          SMK_10
2398.00 C*
-----
2400.00 C*          Load invalid action code array.          Lockout action code function used
2401.00 C*
2402.00 CBR          MOVEA' '          SHAC          with the Program Generator
2403.00 C*
-----
2404.00 C*
2405.00 C*          Load system data.
2406.00 C*
2407.00 CBR          TIME          $WRM12 120          Use the TIME
2408.00 CBR          MOVE $WRM12          $SERST 50          feature to allow for
2409.00 CBR          MOVE $SERST          $SIDAT 5          all date formats
2410.00 CBR          NOVEL*$SERVAL '          $PFMT 7
2411.00 CBR          NOVEL*$LAWKE          $WDAT 9
2412.00 CBR          NOVEL*$JUL '          $TFMT 7
2413.00 CBR          NOVEL*$MON2 '          $GEP 7
2414.00 CBR          MOVE ' '          $SERST 1
2415.00 CBR          CALL 'X002R '
2416.00 C*
-----
2417.00 CBR          PARM          $SIDAT
2418.00 CBR          PARM          $WDAT
2419.00 CBR          PARM          $PFMT
2420.00 CBR          PARM          $TFMT
2421.00 CBR          PARM          $GEP
2422.00 CBR          PARM          $SERST
2423.00 CBR          MOVE $SIDAT          $SUFM 50
2424.00 C*
-----
2425.00 CBR          END999          ENDCR
2426.00 C*
-----
2427.00 C*
2428.00 CBR          OI92R01 R          UNLOCK          Method of releasing
2429.00

```

User Spaces

About User Spaces

User spaces are IBM Operating System 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 the user signs off.

In this chapter you will learn the following about user spaces.

- [What Is a User Space?](#)
- [What Are the Advantages of Using a User Space?](#)
- [How Does a User Space Function?](#)
- [Creating a User Space](#)
- [Writing to a User Space](#)
- [Reading 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.

Note: 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 a string 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 JD Edwards World program J98CKOBJ.

```
For example:  CALL      'J98CKOBJ'      81
              ----      -
              PARM      PSOBJ
              PARM      PSLIB
              PARM      PSTYPE
              PARM      PSMID
              PARM      PSAUT
              PARM      PSERR
```

PARAM (Length)	Description
PSOBJ (10)	The name of your user space.
PSLIB (10)	The name of the library in which you wish to check for the existence of the user space. 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 *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 initialized with *BLANK. 0 No authority 1 Not found 3 No library 4 Member not found 5 No authority to library 6 Cannot assign library

2. Do one of the following:

- If a user space does exist you should clear it and write your new information over the old.
- If the user space does not exist and no errors occurred, you can create your user space. To create a user space, use the QUSCRTUS (Create User Space) command.

```

For example:  CALL      'QUSCRTUS'      81
              ----      -
              PARM      #SPNAM
              PARM      #SPATT
              PARM      #SPSIZ
              PARM      #SPVAL
              PARM      #SPAUT
              PARM      #SPTXT

```

PARAM (Length)	Description
#SPNAM (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; place your user space in library QTEMP to automatically delete your space when you sign off.
#SPATT (10)	The extended attribute of your user space. You may use this field to classify your user space. For example, JD Edwards World uses this field to label all of the user spaces with JD Edwards World.
#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.

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      #XSPCN
              PARM      #XRQSZ
              PARM      #XERR

```

PARAM (Length)	Description
#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.
#XRQSZ (15,0)	The requested size to increase your space.
#XERR (1)	An error flag: <ul style="list-style-type: none"> 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
```

PARAM (Length)	Description
#SPNAM (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.
#SPPOS (4 binary)	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.
#SPLGH (4 binary)	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.
#SPVAL (* user defined)	The actual information to be written to your user space. The field must be at least as long as the length parameter.
#SPAUX (1)	Used to force changes made to your user space to auxiliary storage, such as a disk. The valid values are: <ul style="list-style-type: none"> 0 do not force changes 1 write changes 2 write changes immediately

Note: The X98CHGUS program, JD Edwards World'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 keep track of the pointer is to initialize it to 1. 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.

Initialize pointer to 1	Add 30 bytes to pointer and to save field	Add 41 bytes to pointer and to another save field
B A- 30 bytes -"	B "	B A- 41 bytes -"

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      #SPNAM
              PARM      #SPPOS
              PARM      #SPLGH
              PARM      #SPREC
```

PARM (Length)	Description
#SPNAM (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.
#SPPOS (4 binary)	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.
#SPLGH (4 binary)	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.
#SPREC (* user defined)	The variable that will receive the information from your user space.

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 key 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.

Note: 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

This section contains the following:

- [What Are the Advantages of Using a User Index?](#)
- [How Does a User Index Function?](#)
- [Creating a User Index](#)
- [Writing to a User Index](#)
- [Appearance of Records](#)
- [Retrieving Data 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.

JD Edwards World leaves the first byte in the user index blank for clearing purposes.

A ----- Key ----- "	A ----- Data ----- "

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

To check for user index

Before you actually create a user index, check to see if one already exists using the JD Edwards World program J98CKOBJ.

```
For example:  CALL      'J98CKOBJ'      81
              -----
              PARM      PSOBJ
              PARM      PSLIB
              PARM      PSTYPE
              PARM      PSMID
              PARM      PSAUT
              PARM      PSERR
```

PARM (Length)	Description
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.

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
```

Note: The above code deletes (clears) every record with a key of *Blank in position one.

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                    #IDATT
              PARM                    #IDENT
              PARM                    #IDLEN
              PARM                    #IDINS
              PARM                    #IDKEY
              PARM                    #IDUPD
              PARM                    #IDOPT
              PARM                    #IDAUT
              PARM                    #IDTXT
```

PARAM (Length)	Description
#IDNAM (20)	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.
#IDATT (10)	The extended attribute of your user index. You may use this field to classify your user index. For example, JD Edwards World uses this field to label all of the user indexes with JD Edwards World.
#IDENT (1)	Whether the records you are loading into your user index are Fixed-length (F) or Variable-length (V). Generally, this is set to 'F'.
#IDLEN (4 binary)	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.
#IDINS (1)	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.
#IDKEY (4 binary)	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.
#IDUPD (1)	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.
#IDOPT (1)	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.
#SPAUT (10)	The authority you give users to your user index. Generally, this is *ALL.
#SPTXT (50)	The text description of your user index.

Note: 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

JD Edwards World 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
  
```

PARAM (Length)	Description
#0XNAM (20)	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.
#0XACT (1)	The action you want to perform on your user index. The valid values are: I Inquire A Add C Change D Delete
#0XRUL (2)	The rule used to search your user index depending on the action to be performed. The valid values are: EQ Equal to GT Greater than LT Less than GE Greater than or Equal to LE Less than or Equal to

PARAM (Length)	Description
#0XKLN (3,0)	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.
#0XKEY (120)	The fields that make up the key to your user index. *FIRST (first record) and *LAST (last record) are allowed.
#0XRLN (3,0)	The length of your record. The values are 1 to 999.
#0XREC (120)	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.
#0XSTA (1)	The error status of the manipulation. The possible values are: <ul style="list-style-type: none"> 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:

Total	Sales Item	Description	Cost Center
\$ 500.00	CHO	Chocolate	Denver
\$ 250.00	STR	Strawberry	Denver
\$ 750.00	C&C	Cookies & Cream	Denver
\$1200.00	VAN	Vanilla	Denver
\$ 400.00	ROC	Rocky Road	Denver

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

Total	Sales Item	Description	Cost Center
\$ 250.00	STR	Strawberry	Denver
\$ 400.00	ROC	Rocky Road	Denver
\$ 500.00	CHO	Chocolate	Denver
\$ 750.00	C&C	Cookies & Cream	Denver
\$1200.00	VAN	Vanilla	Denver

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          #OXNAM 20
              PARM  'I'      #OXACT 1
              PARM  'EQ'     #OXRUL 2
              PARM          #OXKLN 30
              PARM  '*FIRST' #OXKEY120
              PARM          #OXRLN 30
              PARM          #OXREC120
              PARM          #OXSTA 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          #OXNAM 20
              PARM      'I'    #OXACT 1
              PARM      'GT'   #OXRUL 2
              PARM          #OXKLN 30
              PARM          #OXKEY120
              PARM          #OXRLN 30
              PARM          #OXREC120
              PARM          #OXSTA 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          #OXNAM 20
              PARM      'I'    #OXACT 1
              PARM      'GT'   #OXRUL 2
              PARM          #OXKLN 30
              PARM          #OXKEY120
              PARM          #OXRLN 30
              PARM          #OXREC120
              PARM          #OXSTA 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		#OXNAM 20
	PARM	'I'	#OXACT 1
	PARM	'LT'	#XRUL 2
	PARM		#XKLN 30
	PARM		#XKEY120
	PARM		#XRLN 30
	PARM		#XREC120
	PARM		#XSTA 1

```

1.00 H/TITLE INDEX - User Index Demonstration
2.00 H*
3.00 H*
4.00 H* copyright (c) 1993
5.00 H* J. D. Edwards & company
6.00 H* this unpublished material is proprietary to
7.00 H* J. D. Edwards & company. All rights reserved.
8.00 H* the methods and techniques described herein are
9.00 H* considered trade secrets and/or confidential.
10.00 H* reproduction or distribution, in whole or in part,
11.00 H* is forbidden except by express written permission
12.00 H* of J. D. Edwards & company.
13.00 H*
14.00 H*
15.00 H*
16.00 F*
17.00 F* PROGRAM REVISION LOG
18.00 F*
19.00 F*
20.00 F*
21.00 F*
22.00A0THHF*
23.00 F*
24.00 F*
25.00 F*
26.00 F*
27.00 F*INDEX CP E WORKSTN KINPDS SHVFD
28.00 F* II SHFILE VINDEX
29.00 F*
30.00 F*
31.00 F* copy member for composite common subroutine - c0001
32.00 F*
33.00 F/COPY JDECFY,C0001
34.00 F*
35.00 E*
36.00 E* PROGRAM TABLES AND ARRAYS
37.00 E*
38.00 E*
39.00 E*
40.00 E* SNK 64 4 ERROR MSG
41.00 E* SNK 64 1 ERROR MSG
42.00 E* SER 64 4 ERROR MSG
43.00 E* SDV 40 1 dflt wrk
44.00 E* SI# 99 1 save indicator
45.00 E* SC 256 1 Literal work
46.00 E*
47.00 E* copy composite member for common subroutine c0001
48.00 E*
49.00 E/COPY JDECFY,C0001
50.00 E*
51.00 E*
52.00 E* copy member for composite common subroutine c0012
53.00 E*
54.00 E/COPY JDECFY,C0012
55.00 E*
56.00 E*
57.00 E* copy member for composite common subroutine c0042
58.00 E*
59.00 E/COPY JDECFY,C0042
60.00 E*
61.00 E*
62.00 E* copy member for composite common subroutine c0097
63.00 E*
64.00 E/COPY JDECFY,C0097
65.00 E*
66.00 I*
67.00 I* PROGRAM INPUT SPECIFICATIONS AND DATA STRUCTURES
68.00 I*
69.00 I*
70.00 I* data structure to load video screen text
71.00 I*
72.00 IDSTXT DS 240
73.00 I 1 16 VTX001
74.00 I 41 56 VTX002
75.00 I 81 92 VTX003
76.00 I 121 150 VTX004
77.00 I 161 163 VTX005
78.00 I 201 203 VTX006
79.00 I*
80.00 I/COPY JDECFY,I00DSINX
81.00 I/COPY JDECFY,I00DS00
82.00 I/COPY JDECFY,I00DSPROG

```

```

83.00 I*
84.00 I*
85.00 I*      copy number for composite common subroutine - cooscc
86.00 I*
87.00 I/COPY JDECPY, roasc
88.00 I*
89.00 I*
90.00 I*      data structures for user index.
91.00 I*      -----
92.00 I*
93.00 I*      * entry record
94.00 I*
95.00 I      DS
96.00 I          1 1 $IRLK
97.00 I          2 6 $IRCO
98.00 I          7 18 $IRMCU
99.00 I          19 48 $IRDLO1
100.00 I          49 81 $IRKPO1
101.00 I          82 84 $IRKPO2
102.00 I*
103.00 I*      * entry length, name/library, text
104.00 I*
105.00 I      DS
106.00 I I      54          B 1 40$IRECL
107.00 I I      'PINDEX  QTEMP  5 24 $IIRK
108.00 I I      'REGENERATION INDEX 25 44 $ITEXT
109.00 I*
110.00 I*      * partial keys 1 & 2, full unique key KEV1.
111.00 I*
112.00 I      DS
113.00 I I      1          B 1 40$IKEY1
114.00 I I      6          B 5 80$IKEY2
115.00 I I      18         B 9 120$IKEYL
116.01 I*
117.02 I*      data structure for file servers
118.03 I*
119.04 I      DS          E DSFO010
120.05 I*
121.00 I/COPY JDECPY, rds008
122.00 I/COPY JDECPY, rds009
123.00 I/COPY JDECPY, rds009
124.00 I*
125.00 I*
126.00 I*
127.00 I*
128.00 I*
129.00 I*
130.00 I*
131.00 I      * INLR      CAREQ'1'      803
132.00 I*
133.00 I*
134.00 I*      if automatic inquiry set, process inquiry.
135.00 I      $AUTO      CAREQ'1'      8003      24
136.00 I*
137.00 I*
138.00 I*      END
139.00 I*
140.00 I*      begin normal program processing.
141.00 I*      -----
142.00 I*
143.00 I      * INLR      DOWEQ'0'
144.00 I*
145.00 I*      if subfile page display not set, set subfile page display.
146.00 I*
147.00 I      #SPRNO      IPKQ 0
148.00 I      E-ADD1
149.00 I      END
150.00 I*
151.00 I*      if subfile page empty, don't display srl page.
152.00 I*
153.00 I      I1      IPLE 0
154.00 I      SETOF
155.00 I      ELSE
156.00 I      SETCN
157.00 I      END
158.00 I*
159.00 I*      write video screen.

```

Record format to be used with User Index defined as a Data Structure

Data Structure containing the record length, User Index name, and User Index description text

Data structure defining three possible key lengths. \$IKEYL is the full key length. Refer to DSIDX1 to see which fields are key fields when \$IKEY1 (1 byte), \$IKEY2 (1-6 bytes), or \$\$IKEYL (1-18 bytes) are being used.

```

160.00 C*
161.00 C      WRITEVINDEK1
162.00 C      WRITEVINDEK2
163.00 C      MOVE '1'          @@AID
164.00 C      EXER @001
165.00 C*
166.00 C*
167.00 C*      Load data field dictionary parameters (one cycle only).
168.00 C*
169.00 C      @998      CASEQ' '          @998
170.00 C*      -----      ----
171.00 C      END
172.00 C*
173.00 C*      begin video screen read processing
174.00 C*
175.00 C      SETOF                                999301
176.00 C      READ VINDEK                          9998
177.00 C      E-ADDO                                @@RROW
178.00 C      E-ADDO                                @@RCOL
179.00 C*
180.00 C*      if video read timed out, end program.
181.00 C*
182.00 C      *IN99      CASEQ'1'          E0J      LE
183.00 C*      -----      ----
184.00 C      @@AID      CASEQ#@R0J          E0J      LE
185.00 C*      -----      ----
186.00 C*
187.00 C*      if valid function key pressed, process and return.
188.00 C*
189.00 C      *IN15      IFMQ '1'
190.00 C      EXER @008K
191.00 C*      -----      ----
192.00 C      *IN18      CASEQ'1'          E0J
193.00 C*      -----      ----
194.00 C      *IN15      CASEQ'1'          END
195.00 C*      -----      ----
196.00 C      END
197.00 C*
198.00 C*      edit the action code.
199.00 C*
200.00 C      EXER @001
201.00 C*      -----      ----
202.00 C*
203.00 C*      if end of job requested, end program.
204.00 C*
205.00 C      @@AID      CASEQ#@R0J          E0J
206.00 C*      -----      ----
207.00 C*
208.00 C*      if clear screen requested, process and return.
209.00 C*
210.00 C      @@AID      IFMQ @FCLE
211.00 C      EXER @001
212.00 C*      -----      ----
213.00 C      GOTO END
214.00 C*      -----      ----
215.00 C      END
216.00 C*
217.00 C*      Load subfile records.
218.00 C*
219.00 C      EXER @003
220.00 C*
221.00 C*      if add or change, validate all video input.
222.00 C*
223.00 C*
224.00 C*
225.00 C      *IN93      CASEQ'0'          @005
226.00 C*      -----      ----
227.00 C      END
228.00 C*
229.00 C*
230.00 C*      if no errors and not inquiry, update file.
231.00 C*
232.00 C      *IN93      IFMQ '0'
233.00 C*      *IN24      CASEQ'0'          @010
234.00 C*      -----      ----
235.00 C      END
236.00 C      END
237.00 C*      return for next input.
238.00 C*
239.00 C      END      TAG
240.00 C*      -----      ----
241.00 C*
242.00 C*      set correct message in line 24.

```

```

244.00
245.00 C          NOVEL/VL24E          VDL24
246.00 C          ELSE
247.00 C          NOVEL/VL24M          VDL24
248.00 C          END
249.00 C*
250.00 C          END
251.00 C*
252.00 C          EOL          TAG
253.00 C*          ---
254.00 C*
255.00 C*          END MAINLINE PROGRAM
256.00 C*
257.00 C*-----
258.00 C*
259.00 C*          copy common subroutine - edit action code
260.00 C*
261.00 C/COPY JDECFY,CO001
262.00 C*-----
263.00 C*
264.00 C*          SUBROUTINE S00EX - process function keys
265.00 C*
266.00 C*
267.00 C*          Processing:  1. process standard function keys.
268.00 C*                    2. process special function key exits.
269.00 C*
270.00 CSE          S00EX          BEGIN
271.00 C*          -----
272.00 C*
273.00 C*          retain current page of subfile.
274.00 C*
275.00 C*                    Z-ADDRESSCH          #GFPRD
276.00 C*
277.00 CSE          T00EX          TAG
278.00 C*          -----
279.00 C*
280.00 C*          if EOL requested, exit subroutine.
281.00 C*
282.00 CSE          @@AID          CARBQ#PMDJ          ENDEXE          LE
283.00 C*          -----
284.00 C*
285.00 C*          if display keys pressed, exit to help facility and return.
286.00 C*
287.00 C*
288.00 CSE          @@AID          IFEQ #FKEYS
289.00 CSE          CALL 'P9601H'
290.00 C*          -----
291.00 CSE          PARM          IO00C
292.00 CSE          PARM          SEVFDG
293.00 CSE          PARM          IO00CE
294.00 C*
295.00 CSE          @@AID          CARNE#FKEYS          T00EXA
296.00 C*          -----
297.00 CSE          GOTO ENDEXE
298.00 C*          -----
299.00 CSE          END
300.00 C*
301.00 C*          if cursor sensitive help pressed, exit to cs help.
302.00 C*
303.00 C*
304.00 CSE          @@AID          IFEQ #FCMEX
305.00 CSE          MOVE#IN
306.00 CSE          CALL 'X960CF'          ##IN          98
307.00 C*          -----
308.00 CSE          PARM          IO00C
309.00 CSE          PARM          SEVFDG
310.00 CSE          PARM          IO00CE
311.00 CSE          PARM ' '          ##CCFF 2
312.00 C*
313.00 CSE          ##FLDN          IFHE *BLANKS
314.00 CSE          EXSE #00VL
315.00 C*          -----
316.00 CSE          MOVE##IN          *IN,1
317.00 CSE          END
318.00 CSE          NOVEL*BLANKS          ##DTAI
319.00 CSE          GOTO ENDEXE
320.00 C*          -----
321.00 CSE          END
322.00 C*
323.00 C*          if display errors pressed, exit to error messages.
324.00 C*
325.00 C*
326.00 CSE          @@AID          IFEQ #FERND
327.00 CSE          Z-ADD1          #G

```

```

327.01 CSR      E-ADD1      #H
328.00 CSR      DOWLES4
329.00 CSR      #G
330.00 CSR      ENK,#G    IREQ '1'
331.00 CSR      MOVE ENK, #G  WER, #H
332.00 CSR      ADD 1        #H
333.00 CSR      END
334.00 CSR      ADD 1        #G
335.00 CSR      END
336.00 CSR      CALL 'POOOOE' 98
337.00 C*
338.00 CSR      PARM
339.00 CSR      GOTO ENDEXE
340.00 C*
341.00 CSR      END
342.00 C*
343.00 C*      if HELP key pressed, exit to help facility and return.
344.00 C*
345.00 CSR      @SAID      IREQ #FHELP
346.00 CSR      CALL 'POOIKLP' 99
347.00 C*
348.00 CSR      PARM
349.00 CSR      PARM      HSEB
350.00 CSR      PARM      HSEB
351.00 CSR      PARM      IOOC
352.00 CSR      PARM      SEVPE
353.00 CSR      PARM      IOOCSE
354.00 CSR      GOTO ENDEXE
355.00 C*
356.00 CSR      END
357.00 C*
358.00 C*      if ROLL UP key pressed, load next page of subfile.
359.00 C*
360.00 CSR      @SAID      IREQ #IRCLU
361.00 CSR      @GEND      IREQ '1'
362.00 CSR      MOVE ' '      VSELCL 1
363.00 CSR      ERSE 2004
364.00 C*
365.00 CSR      ELSE
366.00 CSR      E-ADD@SV11    I1
367.00 CSR      MOVE 'BLANK'  @FDL01
368.00 CSR      MOVE 'BLANK'  @FMCU
369.00 CSR      MOVE 'BLANK'  @FPC01
370.00 CSR      MOVE 'BLANK'  @FPC02
371.00 CSR      MOVE 'BLANK'  @FMCU
372.00 CSR      I1      ADD 1      @SPRNO
373.00 CSR      DO @PAGE
374.00 CSR      ADD 1
375.00 CSR      MOVE@IH      I1
376.00 CSR      WRITEINDEXS  SWH
377.00 CSR      END
378.00 CSR      E-ADDI1      @SV11
379.00 CSR      END
380.00 CSR      GOTO ENDEXE
381.00 C*
382.00 CSR      END
383.00 C*
384.00 C*      if ROLL DOWN key pressed, reset subfile page display.
385.00 C*
386.00 C*
387.00 CSR      @SAID      IREQ #FOLD
388.00 CSR      MOVE @SV11    @SPRNO
389.00 CSR      GOTO ENDEXE
390.00 C*
391.00 CSR      END
392.00 C*
393.00 C*      if clear screen pressed, clear screen and return.
394.00 C*
395.00 C*
396.00 CSR      @SAID      IREQ #FCLR
397.00 CSR      ERSE 2001
398.00 C*
399.00 CSR      GOTO ENDEXE
400.00 C*
401.00 CSR      END
402.00 CSR      @AID      IREQ '1'
403.00 CSR      SETON      0193
404.00 CSR      GOTO ENDEXE
405.00 C*
406.00 CSR      END
407.00 CSR      ENDEXE      ENDCR
408.00 C*
409.00 C*
410.00 C*
411.00 C*

```

```

412.00 C*      SUBROUTINE 200VL - cursor control return values
413.00 C*      -----
414.00 C*
415.00 C*      by format, find the field to update and move in the
416.00 C*      returned value. if the format is a subfile, the record
417.00 C*      to change is found in @SERH.
418.00 C*
419.00 CSE      200VL      REGR
420.00 C*      -----
421.00 C*
422.00 CSE      **EVAL      IFEQ 'BLANK'
423.00 CSE      MOVE 'BLANK'          **EVAL
424.00 CSE      END
425.00 C*
426.00 C*      return values for fields in format VINDEXC
427.00 C*
428.00 CSE      **RPMT      IFEQ 'VINDEXC '
429.00 C*
430.00 CSE      **FLEN      IFEQ 'ACTION '
431.00 CSE      MOVE**EVAL      ACTION
432.00 CSE      GOTO ENDOVL
433.00 C*      -----
434.00 C*      END
435.00 C*
436.00 CSE      **FLEN      IFEQ 'VDCO '
437.00 CSE      MOVE**EVAL      VDCO
438.00 CSE      MOVE**EVAL      VCO001
439.00 CSE      GOTO ENDOVL
440.00 C*      -----
441.00 CSE      END
442.00 CSE      END
443.00 C*
444.00 C*      return values for fields in format VINDEXS
445.00 C*
446.00 CSE      **RPMT      IFEQ 'VINDEXS '
447.00 CSE      @SERH      ANDGTO
448.00 C*
449.00 CSE      @SERH      MOVE**IN      SHIN      81
450.00 CSE      CHAINVINDEXS
451.00 CSE      'IN,1      IFEQ 'O'
452.00 CSE      MOVE@SHIN      *IN,1
453.00 C*
454.00 C*
455.00 CSE      **FLEN      IFEQ 'SFMCU '
456.00 CSE      MOVE**EVAL      SFMCU
457.00 CSE      GOTO TOOVLA
458.00 C*      -----
459.00 CSE      END
460.00 C*
461.00 CSE      **FLEN      IFEQ 'SFDLO1 '
462.00 CSE      MOVE**EVAL      SFDLO1
463.00 CSE      GOTO TOOVLA
464.00 C*      -----
465.00 CSE      END
466.00 C*
467.00 CSE      **FLEN      IFEQ 'SFRP01 '
468.00 CSE      MOVE**EVAL      SFRP01
469.00 CSE      GOTO TOOVLA
470.00 C*      -----
471.00 CSE      END
472.00 C*
473.00 CSE      **FLEN      IFEQ 'SFRP02 '
474.00 CSE      MOVE**EVAL      SFRP02
475.00 CSE      GOTO TOOVLA
476.00 C*      -----
477.00 CSE      END
478.00 CSE      TAG
479.00 C*      ----
480.00 CSE      SETON
481.00 CSE      MOVE@IN      SHIN      32
482.00 CSE      UPDATVINDEXS
483.00 CSE      END
484.00 CSE      END
485.00 C*
486.00 C*      return values for fields in format VINDEXI
487.00 C*
488.00 CSE      **RPMT      IFEQ 'VINDEXI '
489.00 CSE      END
490.00 C*
491.00 CSE      ENDOVL      ENDS
492.00 C*      *****
493.00 C*
494.00 C*      SUBROUTINE 2001 - clear fields

```

```

495.00 C* -----
496.00 C*
497.00 C* processing: 1. reset all video screen and data file fields
498.00 C* for next transaction.
499.00 C* 2. clear action code only if requested.
500.00 C*
501.00 CSE      2001      BEGIN
502.00 C* -----
503.00 CSE      MOVE *BLANK      $DLO1
504.00 CSE      MOVE *BLANK      $DPO1
505.00 CSE      MOVE *BLANK      $DPO2
506.00 CSE      Z-ADD*ENR0      ##R0L
507.00 CSE      Z-ADD*ENR0      ##R0W
508.00 CSE      Z-ADD*ENR0      ##R00
509.00 CSE      MOVE *BLANK      $DLO1
510.00 CSE      MOVE *BLANK      $DNCU
511.00 CSE      MOVE *BLANK      $DPO1
512.00 CSE      MOVE *BLANK      $DPO2
513.00 CSE      MOVE *BLANK      $DNCU
514.00 CSE      MOVE *BLANK      VD00
515.00 CSE      MOVE$VUL24M      VDL24
516.00 CSE      MOVE '0'          $HIN37
517.00 C* -----
518.00 C*
519.00 C* clear action code only if clear screen action.
520.00 C*
521.00 CSE      @BARD      IFNE #PCLN
522.00 CSE      MOVE *ALL'0'      $RESET
523.00 CSE      MOVE$RESET      *IN,41
524.00 CSE      MOVE ' '          ACTION 1
525.00 CSE      Z-ADD00000      ##R00
526.00 CSE      SETCN                      31
527.00 CSE      WRITEVINDEXC                      99
528.00 CSE      SETCF                      203193
529.00 CSE      Z-ADD0          IL
530.00 CSE      DO $PGSE
531.00 CSE      ADD 1          IL
532.00 CSE      MOVE$IN      $HIN
533.00 CSE      WRITEVINDEXS                      81
534.00 CSE      END
535.00 CSE      Z-ADD11          $V11
536.00 CSE      MOVE *BLANK      $100
537.00 CSE      MOVE *BLANK      $JNCU
538.00 CSE      MOVE *BLANK      VC0001
539.00 CSE      END
540.00 C* -----
541.00 CSE      END001      ENDR
542.00 C* -----
543.00 C*
544.00 C* SUBROUTINE 2003 - edit key
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 CSE      2003      BEGIN
553.00 C* -----
554.00 C*
555.00 C* reset error indicators and arrays.
556.00 C*
557.00 CSE      MOVE *ALL'0'      $RESET 39
558.00 CSE      MOVE *BLANK      $RESET 63
559.00 CSE      MOVE$RESET      *IN,41
560.00 CSE      MOVE$RESET      $MK,2
561.00 CSE      CLEARERR
562.00 C*
563.00 C* clear the user index to begin with; set flag.
564.00 C* -----
565.00 CSE      CLEAR$INDEX
566.00 CSE      MOVE 'Y'          $START 1
567.00 C*
568.00 C* Load video input field for - company
569.00 C*
570.00 CSE      MOVE$VD00      $EN
571.00 CSE      ERER C0012
572.00 C* -----
573.00 C*
574.00 CSE      Z-ADD#NUMR      $WKS 50
575.00 CSE      MOVE $WKS      $100
576.00 CSE      MOVE $WKS      VD00
577.00 C* -----

```

Clear Data
Structure
containing record
format for User
Index

```

578.00 C*
579.00 C*
580.00 C*
581.00 CSE
582.00 CSE
583.00 CSE
584.00 C*
585.00 CSE
586.00 C*
587.00 CSE
588.00 CSE
589.00 CSE
590.00 CSE
591.00 CSE
592.00 CSE
593.00 CSE
594.00 CSE
595.00 C*
596.00 C*
597.00 C*
598.00 CSE
599.00 CSE
600.00 CSE
601.00 C*
602.00 C*
603.00 C*
604.00 CSE
605.00 CSE
606.00 CSE
607.00 CSE
608.00 C*
609.00 C*
610.00 C*
611.00 CSE
612.00 CSE
613.00 CSE
614.00 CSE
615.00 C*
616.00 C*
617.00 C*
618.00 CSE
619.00 C*
620.00 C*
621.00 C*
622.00 C*
623.00 CSE
624.00 C*
625.00 C*
626.00 C*
627.00 C*
628.00 CSE
629.00 CSE
630.00 CSE
631.00 CSE
632.00 C*
633.00 C*
634.00 C*
635.00 CSE
636.00 CSE
637.00 CSE
638.00 C*
639.00 C*
640.00 C*
641.00 CSE
642.00 C*
643.00 C*
644.00 CSE
645.00 CSE
646.00 C*
647.00 CSE
648.00 CSE
649.00 CSE
650.00 CSE
651.00 CSE
652.00 CSE
653.00 CSE
654.00 CSE
655.00 CSE
656.00 CSE
657.00 CSE
658.00 CSE
659.00 CSE

```

determine if any entries exist for that company.

Z-ADD@1REV2	PSKEYL
Z-ADD@1RECL	PSRECL
MOVE@ASIDX1	PSKEY

Load key length, record length, and key with values

CALL 'X00IDX'

PARM 'I'	%IDX	idx name lib
PARM 'EQ'	PSACTM	action code
	PSRULE	action rule
	PSKEYL	key length
	PSKEY	key fields
	PSRECL	entry length
	PSRECL	entry
	PSSTS	error status

Call to User Index to inquire on an existing record

```

error of trying to delete but not found.
      PSSTS  IPHE '0'          NOT FOUND
*IN23      COMP '1'          41 *ERROR*
          END

```

if indicator 41 on, invalid key for action code.

```

*IN41      IREQ '1'
          MOVE '1'          @NK, 2
          SETCN
          END

```

if indicator 99 on, record in use.

```

*IN99      IREQ '1'
          MOVE '1'          @NK, 6
          SETCN
          END

```

if not inquiry, skip remainder of subroutine.

```

*IN24      CABEQ '0'        @ND003
          -----

```

if errors, skip remainder of subroutine.

```

*IN93      CABEQ '1'        @ND003
          -----

```

initialize subfile indexes.

```

          Z-ADD@          I1          50
          Z-ADD@          @SV11      50
          Z-ADD@          @SPRNO
          MOVE '0'        @GEND      1

```

reinitialize subfile display

```

          SETCN
          WRITEVINDEEC
          SETCF

```

Load subfile records.

```

          ERSE S004
          -----
          I1          IPLT @PGSE
          @PGSE      SUB I1          @G
          MOVE *BLANK  SFDLO1
          MOVE *BLANK  SFPNCU
          MOVE *BLANK  SFRPO1
          MOVE *BLANK  SFRPO2
          MOVE *BLANK  SFPNCU
          DO @G
          ADD 1          I1
          MOVE@*IN      SHIN
          WRITEVINDEEC
          END
          Z-ADDI1
          END
          @SV11
          END
          @ND003      ENDRN

```

Check error status parameter to see if a record was found

```

660.00 C*****
661.00 C*
662.00 C*      copy common subroutine - right justify numeric fields
663.00 C*
664.00 C/COPY JMSOPY, C0012
665.00 C*****
666.00 C*
667.00 C*      SUBROUTINE S004 - Load Video screen Data
668.00 C*      -----
669.00 C*
670.00 C*      Processing: 1. Move data base information to video screen.
671.00 C*                    All video screen fields are alpha and
672.00 C*                    therefore numeric information must be
673.00 C*                    processed through subroutine C0014 to set
674.00 C*                    proper decimals and provide editing for
675.00 C*                    display on screen.
676.00 C*
677.00 C*                    data fields must be converted from their
678.00 C*                    internal format of month, day and year or
679.00 C*                    Julian to the system format using program
680.00 C*                    X0028.
681.00 C*
682.00 CSE      S004      BEGIN
683.00 C*
684.00 C*
685.00 C*      Load data field dictionary parameters (one cycle only).
686.00 C*
687.00 CSE      $$$$      CASHQ' '      $$$$
688.00 C*
689.00 CSE      END
690.00 C*
691.00 C*      if subfile load completed, skip subroutine.
692.00 C*
693.00 CSE      $END      IFMQ '1'
694.00 CSE      Z-ADDO      #SFRMO
695.00 CSE      GOVD ENDD004
696.00 CSE      END
697.00 C*-----
698.00 C*
699.00 C*      save company number for comparison later.
700.00 C*      -----
701.00 C*
702.00 CSE      MOVE $100      $$$00 5
703.00 C*-----
704.00 C*
705.00 C*      Move to output - company description.
706.00 C*
707.00 CSE      MOVE *BLANKS      P000
708.00 CSE      MOVE$100      KY00
709.00 CSE      CALL 'XS0010'
710.00 C*
711.00 CSE      PARM
712.00 CSE      PARM      P000
713.00 C*
714.00 CSE      MOVE$CNAME      VC0001
715.00 C*-----
716.00 C*
717.00 C*      initialize subfile page control and index.
718.00 C*
719.00 CSE      Z-ADDO      $PG 30
720.00 CSE      Z-ADDO      $SFRMO
721.00 CSE      Z-ADD$SV11      11
722.00 C*-----
723.00 C*
724.00 C*      read user index until end or subfile page filled.
725.00 C*
726.00 CSE      SETOF      $6
727.00 CSE      *IN96      DOWEQ'0'
728.00 C*
729.00 C*      first time through, have already read first record, so skip
730.00 C*      the index logic. (first time through if $START = 'Y'
731.00 C*      -----
732.00 C*
733.00 CSE      $START      IFMQ 'Y'
734.00 CSE      MOVE ' '
735.00 CSE      ELSE

```

```

733.00 C*
734.00 C* successive times through, read next "greater" entry.
735.00 C*
736.00 C*
737.00 CWR
738.00 CWR
739.00 CWR
740.00 C*
741.00 CWR
742.00 C*
743.00 CWR
744.00 CWR Call to User
745.00 CWR Index to
746.00 CWR retrieve next
747.00 CWR record that is
748.00 CWR greater than
749.00 CWR current key
750.00 CWR value
751.00 C*
752.00 CWR
753.00 C*
754.00 C* if status is '0' than assume not found.
755.00 C*
756.00 C*
757.00 CWR
758.00 CWR
759.00 C*
760.00 C* retrieve entry to load data structure.
761.00 C*
762.00 C*
763.00 CWR
764.00 C* compare new company to inquiry : if changed, and.
765.00 C*
766.00 C*
767.00 C*
768.00 CWR
769.00 CWR
770.00 CWR
771.00 C*
772.00 C* At end of index, set subfile completion flag and set high
773.00 C* intensity attribute on last subfile record.
774.00 C*
775.00 CWR
776.00 CWR
777.00 CWR
778.00 CWR
779.00 C*
780.00 CWR
781.00 C*
782.00 C*
783.00 C* reset record selection flag ($SEL).
784.00 C*
785.00 CWR
786.00 C*
787.00 C*
788.00 C* update subfile for selected records.
789.00 C*
790.00 CWR
791.00 C*
792.00 C*
793.00 C* move to output - description 01
794.00 C*
795.00 CWR
796.00 C*
797.00 C*
798.00 C* move to output - cost center
799.00 C*
800.00 CWR
801.00 CWR
802.00 CWR
803.00 CWR
804.00 CWR
805.00 CWR
806.00 CWR
807.00 CWR
808.00 CWR
809.00 CWR
810.00 CWR
811.00 C*
812.00 CWR
813.00 CWR
814.00 CWR
815.00 CWR

```

Z-ADD@1KEYL	P@KEYL	
Z-ADD@1RECL	P@RECL	
NOVEL@SIDX1	P@KEY	Load key length, record length, and key with values

CALL 'X00IDX'			
PARAM	@1IDX		index name
PARAM 'I'	P@ACTN 1		action code
PARAM 'GT'	P@RULE		action rule
PARAM	P@KEYL		key length
PARAM	P@KEY		key fields
PARAM	P@RECL		entry length
PARAM	P@RSC		entry
PARAM	P@STS		error status
END	@START		

	96	
P@STS	@CORP '0'	96 IF 'GT' '0'

	96	
@100	IPNE @@00	@END
	@END	96

```

816.00 CSE          END
817.00 C**
818.00 C*
819.00 C*      MOVE TO OUTPUT - CATEGORY CODE - COST CENTER 01
820.00 C*
821.00 CSE          MOVE *BLANK          #GINKR
822.00 CSE          MOVE @IRP01          #GINKR
823.00 CSE          MOVE TRRP01          #DTYP
824.00 CSE          MOVE WRRP01          #ENRD
825.00 CSE          MOVE EWRP01          #EC
826.00 CSE          MOVE FWRP01          #GSPD
827.00 CSE          MOVE GWRP01          #DRTD
828.00 CSE          MOVE JWRP01          #ALR
829.00 CSE          MOVE ' '          #ECCR
830.00 CSE          MOVE ' '          #DCCR
831.00 CSE          EXER C00161
832.00 C*          ----
833.00 CSE          #ALR      IFEQ 'L'
834.00 CSE          MOVE @SINKR          @FRP01
835.00 CSE          ELSE
836.00 CSE          MOVE #GINKR          @FRP01
837.00 CSE          END
838.00 C**
839.00 C*
840.00 C*      MOVE TO OUTPUT - CATEGORY CODE - COST CENTER 02
841.00 C*
842.00 CSE          MOVE *BLANK          #GINKR
843.00 CSE          MOVE @IRP02          #GINKR
844.00 CSE          MOVE TRRP02          #DTYP
845.00 CSE          MOVE WRRP02          #ENRD
846.00 CSE          MOVE EWRP02          #EC
847.00 CSE          MOVE FWRP02          #GSPD
848.00 CSE          MOVE GWRP02          #DRTD
849.00 CSE          MOVE JWRP01          #ALR
850.00 CSE          MOVE ' '          #ECCR
851.00 CSE          MOVE ' '          #DCCR
852.00 CSE          EXER C00161
853.00 C*          ----
854.00 CSE          #ALR      IFEQ 'L'
855.00 CSE          MOVE @SINKR          @FRP02
856.00 CSE          ELSE
857.00 CSE          MOVE #GINKR          @FRP02
858.00 CSE          END
859.00 C**
860.00 C*
861.00 C*      MOVE TO OUTPUT - COST CENTER
862.00 C*
863.00 CSE          MOVE *BLANK          #GINKR
864.00 CSE          MOVE @IRPCU          #GINKR
865.00 CSE          MOVE TRNCU          #DTYP
866.00 CSE          MOVE WANCU          #ENRD
867.00 CSE          MOVE EANCU          #EC
868.00 CSE          MOVE FANCU          #GSPD
869.00 CSE          MOVE GANCU          #DRTD
870.00 CSE          MOVE JANCU          #ALR
871.00 CSE          MOVE ' '          #ECCR
872.00 CSE          MOVE ' '          #DCCR
873.00 CSE          EXER C00161
874.00 C*          ----
875.00 CSE          #ALR      IFEQ 'L'
876.00 CSE          MOVE @SINKR          @GNCU
877.00 CSE          ELSE
878.00 CSE          MOVE #GINKR          @GNCU
879.00 CSE          END
880.00 C**
881.00 C**
882.00 C*
883.00 C*      INCREMENT SUBFILE PAGE CONTROL AND INDEX.
884.00 C*
885.00 CSE          ADD 1          @PG
886.00 CSE          ADD 1          I1
887.00 C*
888.00 C*      IF SUBFILE PAGE DISPLAY NOT SET, SET SUBFILE PAGE DISPLAY.
889.00 C*
890.00 CSE          #GFRNO      IFEQ 0
891.00 CSE          E-ADD1          #GFRNO
892.00 CSE          END
893.00 C*
894.00 C*      WRITE SUBFILE RECORD AND SAVE CURRENT SUBFILE INDEX.
895.00 C*
896.00 CSE          MOVER*IN          @WIN
897.00 CSE          WRITEINDEXS

```

```

896.00 CSE          E-ADD11          @SV11
899.00 C*
900.00 C*      if subfile page loaded, drop out of subroutine.
901.00 C*
902.00 CSE          @PG          CARR@PGSZ          ENDD004
903.00 C*
904.00 CSE          END
905.00 CSE          END
906.00 C*-----
907.00 CSE          ENDD004          ENDR
908.00 C*-----
909.00 C*
910.00 C*      copy common subroutine - format numeric fields for output with override
911.00 C*
912.00 C/COPY JMSOPY,COO161
913.00 C*-----
914.00 C*
915.00 C*      SUBROUTINE 2005 - validate and update input data.
916.00 C*-----
917.00 C*
918.00 C*      processing:  1.  validate all video input.  numeric data
919.00 C*                   must be processed thru subroutines 2002 &
920.00 C*                   2003 to be converted to internal numeric
921.00 C*                   representation (18 digits & decimals).
922.00 C*                   data fields must be converted from system
923.00 C*                   format to their internal format of month,
924.00 C*                   day and year or julian using program 2002a.
925.00 C*                   2.  update data fields from input and process
926.00 C*                   subfile transaction.
927.00 C*
928.00 CSE          2005          BEGR
929.00 C*-----
930.00 C*
931.00 C*      if not addition or change, bypass subroutine
932.00 C*
933.00 CSE          *IN21          IFEQ '0'
934.00 CSE          *IN22          ANDEQ '0'
935.00 CSE          GOTO ENDD005
936.00 C*-----
937.00 CSE          END
938.00 C*
939.00 C*      process all subfile transactions.
940.00 C*
941.00 CSE          MOVE ' '          @WRT          1
942.00 CSE          E-ADD1          @IX          70
943.00 CSE          SETCF
944.00 CSE          *IN96          DOWEQ '0'
945.00 CSE          *IN99          ANDEQ '0'
946.00 CSE          @IX          ANDLE@SV11
947.00 CSE          MOVE@RHSSET          *IN,41
948.00 CSE          @IX          CHAINVINDEXS
949.00 CSE          *IN96          IFEQ '0'
950.00 CSE          *IN99          ANDEQ '0'
951.00 C*
952.00 C*      Load video input field for - cost center
953.00 C*
954.00 CSE          MOVE@RHSVC          @FI
955.00 CSE          ENDR C0042
956.00 C*-----
957.00 CSE          MOVE @RADJ          @INCV
958.00 C*
959.00 C*      determine if prior record existed in user index.
960.00 C*-----
961.00 C*
962.00 CSE          E-ADD@INREYL          @KEYL
963.00 CSE          E-ADD@INRECL          @RECL
964.00 CSE          MOVE@LSIDX1          @KEY
965.00 C*
966.00 CSE          CALL 'X00IDX'
967.00 C*
968.00 C*      Loading of
969.00 C*      parameters
970.00 C*      and call to
971.00 C*      User Index to
972.00 C*      see if a record
973.00 C*      exists
974.00 CSE          PARM          @INDEX          idx name/lib
975.00 CSE          PARM '1'          @ACTN          action code
976.00 CSE          PARM 'EQ'          @RULE          action rule
977.00 CSE          PARM          @KEYL          key length
978.00 CSE          PARM          @KEY          key fields
979.00 CSE          PARM          @RECL          entry length
980.00 CSE          PARM          @REC          entry
981.00 CSE          PARM          @STS          error status
982.00 C*
983.00 C*      if no data and prior record existed, delete old record.
984.00 C*-----
985.00 C*
986.00 C*
987.00 C*
988.00 C*
989.00 C*

```

```

980.00 CSR          IFNE '0'          Check error status parameter
981.00 CSR          SFNCU          ANDEQ *BLANK          to see if record has found
982.00 C*
983.00 CSR
984.00 C*
985.00 CSR          CALL 'X00IDX'
986.00 CSR          PARM          $IIDX          idx name/lib
987.00 CSR          PARM          'D'          DEACTH          action
988.00 CSR          PARM          'EQ'          PERHLE          action rule
989.00 CSR          PARM          PKEYL          key length
990.00 CSR          PARM          PKEY          key fields
991.00 CSR          PARM          PSECL          entry length
992.00 CSR          PARM          PSECL          entry
993.00 C*
994.00 CSR          PARM          PASTE          error status
995.00 C*
996.00 C*
997.00 C*          process only non-blank records.
998.00 C*
999.00 CSR          SFNCU          IFNE *BLANK
1000.00 C*
1001.00 C*
1002.00 C*          scrub and edit - description 01
1003.00 C*
1004.00 CSR          NOVEL$PDL01          $IIDL01
1005.00 C*
1006.00 C*          edit allowed values - description 01
1007.00 C*
1008.00 CSR          ADEL01          IFNE 'NR'
1009.00 CSR          $IIDL01          ANDEQ *BLANK
1010.00 CSR          MOVE '1'          SMK,03          4293
1011.00 CSR          SETCM
1012.00 CSR          END
1013.00 C*
1014.00 C*
1015.00 C*          scrub and edit - cost center
1016.00 C*
1017.00 CSR          NOVEL$FNCU          $FI
1018.00 CSR          EXER C0042
1019.00 C*
1020.00 CSR          MOVE $RADJ          $IMCU
1021.00 C*
1022.00 C*
1023.00 C*          scrub and edit - category code - cost center 01
1024.00 C*
1025.00 CSR          NOVEL$PRP01          $IRP01
1026.00 C*
1027.00 C*          set default value - category code - cost center 01
1028.00 C*
1029.00 CSR          $IRP01          IFNE *BLANK
1030.00 CSR          DWRP01          IFNE *BLANK
1031.00 CSR          NOVEL$DWRP01          $40
1032.00 CSR          MOVE$40          $IRP01
1033.00 CSR          $40,1          IFNE ' '
1034.00 CSR          MOVE ' '          $40,1
1035.00 CSR          Z-ADD2          $M
1036.00 CSR          $M          DOWLE$40
1037.00 CSR          $40,$M          IFNE ' '
1038.00 CSR          MOVE ' '          $40,$M
1039.00 CSR          END
1040.00 CSR          ADD 1          $M
1041.00 CSR          END
1042.00 CSR          NOVEL$40,2          $IRP01
1043.00 CSR          END
1044.00 C*
1045.00 C*
1046.00 C*          edit allowed values - category code - cost center 01
1047.00 C*
1048.00 C*
1049.00 CSR          AWRP01          IFNE *BLANK
1050.00 CSR          NOVEL$AWRP01          $40
1051.00 CSR          MOVE *HIVAL          $AV
1052.00 CSR          EXER C997
1053.00 C*
1054.00 CSR          MOVE ' '          $ERTGT
1055.00 CSR          MOVE *BLANK          $WRKLO 10
1056.00 CSR          NOVEL$IRP01          $WRKLO
1057.00 CSR          $AV,1          IFNE *HIVAL
1058.00 CSR          $WRKLO          LOKUP$AV          01
1059.00 CSR          *IN61          IFNE '0'
1060.00 CSR          MOVE '1'          $ERTGT
1061.00 CSR          END
1062.00 CSR          $ERTGT          IFNE '1'
1063.00 CSR          MOVE '1'          SMK,07

```

```

1146.00 C*
1147.00 C* edit upper and lower range - category code - cost center 02
1148.00 C*
1149.00 CWR      L&RPO2      IFNE 'BLANK
1150.00 CWR      MOVE '1'          $SERST
1151.00 CWR      $L&RPO2      IFGE L&RPO2
1152.00 CWR      $L&RPO2      ANDL&R&RPO2
1153.00 CWR      MOVE          $SERST
1154.00 CWR      END
1155.00 CWR      $SERST      IFEQ '1'
1156.00 CWR      MOVE '1'
1157.00 CWR      SETON          4493
1158.00 CWR      END
1159.00 CWR      END
1160.00 C*
1161.00 C* edit from descriptive titles - category code - cost center 02
1162.00 C*
1163.00 CWR      R&RPO2      IFNE 'BLANK
1164.00 CWR      CLEAR100050
1165.00 CWR      MOVE ' '
1166.00 CWR      MOVE&R&RPO2 $SERST
1167.00 CWR      MOVE R&RPO2 $OUT
1168.00 CWR      MOVE $L&RPO2 $KEY
1169.00 CWR      CALL 'X0005          81
1170.00 C*
1171.00 CWR      P&RPN      P&RPN          100050
1172.00 CWR      #USER      IFEQ '1'
1173.00 CWR      MOVE '1'          @MK,09
1174.00 CWR      SETON          4493
1175.00 CWR      END
1176.00 CWR      END
-----
1177.00 C*
1178.00 C*
1179.00 C* if no errors, update user index.
1180.00 C* =====
1181.00 C*
1182.00 CWR      *INS3      IFEQ '0'
1183.00 C*
1184.00 CWR      Z-ADD@1KEYL  PKKEYL
1185.00 CWR      Z-ADD@1RECL  PKRECL
1186.00 CWR      MOVE&G1DXL  PKKEY
1187.00 CWR      MOVE&G1DXL  PKREC
1188.00 C*
1189.00 CWR      $SERST      IFEQ '0'
1190.00 CWR      $L&RPO2      ANDG&R&RPO2
1191.00 C*
1192.00 CWR      CALL 'X0010X'
1193.00 C*
1194.00 CWR      P&RPN      $LIDX      index name
1195.00 CWR      P&RPN 'C'      P&RACTN  action code
1196.00 CWR      P&RPN      P&RULS   function rule
1197.00 CWR      P&RPN      P&RKEYL  key length
1198.00 CWR      P&RPN      P&RKEY   key
1199.00 CWR      P&RPN      P&RRECL  rec'd length
1200.00 CWR      P&RPN      P&RREC  record
1201.00 CWR      P&RPN      P&RSTS  status
1202.00 C*
1203.00 CWR      ELSE
1204.00 C*
1205.00 CWR      CALL 'X0010X'
1206.00 C*
1207.00 CWR      P&RPN      $LIDX      idx name/lib
1208.00 CWR      P&RPN 'I'      P&RACTN  action code
1209.00 CWR      P&RPN 'EQ'     P&RULS   action rule
1210.00 CWR      P&RPN      P&RKEYL  key length
1211.00 CWR      P&RPN      P&RKEY   key fields
1212.00 CWR      P&RPN      P&RRECL  entry length
1213.00 CWR      P&RPN      P&RREC  entry
1214.00 CWR      P&RPN      P&RSTS  error status
1215.00 C*
1216.00 CWR      $SERST      IFEQ '0'
1217.00 CWR      MOVE '1'          @MK,2
1218.00 CWR      SETON          4193
1219.00 CWR      ELSE
1220.00 C*
1221.00 CWR      CALL 'X0010X'
1222.00 C*
1223.00 CWR      P&RPN      $LIDX      idx name/lib
1224.00 CWR      P&RPN 'A'      P&RACTN  action code
1225.00 CWR      P&RPN      P&RULS   action rule
1226.00 CWR      P&RPN      P&RKEYL  key length
1227.00 CWR      P&RPN      P&RKEY   key fields
1228.00 CWR      P&RPN      P&RRECL  entry length
1229.00 CWR      P&RPN      P&RREC  entry
1230.00 CWR      P&RPN      P&RSTS  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*
1232.00 CGR          MOVE '1'          $MKT
1233.00 CGR          END
1234.00 CGR          END
1235.00 CGR          END
1236.00 C*
1237.00 CGR          *IN93      IFEQ '1'
1238.00 CGR          $GFRNO    ANDEQ $ERR0
1239.00 CGR          E-ADD11    $GFRNO
1240.00 CGR          END
1241.00 C*
1242.00 CGR          END
1243.00 C*
1244.00 C*      if errors, set subfile next change flag.
1245.00 C*
1246.00 CGR          *IN93      IFEQ '1'
1247.00 CGR          SETCN          32
1248.00 CGR          END
1249.00 C*
1250.00 C*      update all subfile records read.
1251.00 C*
1252.00 CGR          MOVE $IN          $MIN
1253.00 CGR          UPDATVINDEXS      81
1254.00 CGR          SETOF          32
1255.00 C*
1256.00 C*      read next subfile record.
1257.00 C*
1258.00 CGR          ADD 1          $QIX
1259.00 CGR          END
1260.00 CGR          END
1261.00 C*
1262.00 C*      if error detected on a add, change action code to 'c'
1263.00 C*
1264.00 CGR          *IN93      IFEQ '1'
1265.00 CGR          $MKT      ANDEQ '1'
1266.00 CGR          MOVE 'C'          ACTION
1267.00 CGR          END
1268.00 C*
1269.00 CGR          END005      ENDGR
1270.00 C*
1271.00 C*
1272.00 C*      copy common subroutine - right adjust alphanumeric field
1273.00 C*
1274.00 C/COPY JDBCPY,CO042
1275.00 C*
1276.00 C*
1277.00 C*      copy common subroutine - build Allowed values work Array
1278.00 C*
1279.00 C/COPY JDBCPY,C097
1280.00 C*
1281.00 C*
1282.00 C*      SUBROUTINE S010 - update data base
1283.00 C*
1284.00 C*
1285.00 C*      processing: 1. update data base file for delete action.
1286.00 C*
1287.00 CGR          S010      BEGR
1288.00 C*
1289.00 C*
1290.00 C*      if delete action, delete all records by primary partial key.
1291.00 C*
1292.00 CGR          *IN23      IFEQ '1'
1293.00 CGR          E-ADD$1KEY2  $KEYL
1294.00 CGR          E-ADD$1RECL  $RECL
1295.00 C*
1296.00 CGR          CALL 'X00IDX'
1297.00 C*
1298.00 CGR          PARM          $1IDX      idx name/lib
1299.00 CGR          PARM 'D'      $ACTN      action
1300.00 CGR          PARM 'EQ'     $RULE      action rule
1301.00 CGR          PARM          $KEYL      key length
1302.00 CGR          PARM          $KEY      key fields
1303.00 CGR          PARM          $RECL      entry length
1304.00 CGR          PARM          $RECL      entry
1305.00 CGR          PARM          $STS      error status
1306.00 CGR          END
1307.00 C*
1308.00 C*      clear data field for next transaction
1309.00 C*
1310.00 CGR          MOVE $FCLX      @BAID
1311.00 CGR          ERSE S001
1312.00 C*
1313.00 CGR          END010      ENDGR

```

```

1314.00 C*-----
1315.00 C*
1316.00 C*      SUBROUTINE S998 - Load dictionary parameters.
1317.00 C*      -----
1318.00 C*
1319.00 CSE          S998      BEGIN
1320.00 C*          ----      ----
1321.00 C*
1322.00 C*
1323.00 C*      dictionary parameters for - description of
1324.00 C*
1325.00 CSE          MOVE 'BLANK'      FEDTRAI
1326.00 CSE          MOVEV 'DL01'      FEDTRAI
1327.00 CSE          CALL 'X9800E'      51
1328.00 C*          -----
1329.00 CSE          PARM              19800E
1330.00 CSE          FREER      IFREQ 'O'
1331.00 CSE          MOVE FEDGCR      @D0L01      40
1332.00 CSE          MOVE FEDTAT      T@D0L01      1
1333.00 CSE          MOVE FREC      @D0L01      1
1334.00 CSE          MOVE FEDTAG      @D0L01      40
1335.00 CSE          MOVE FEDTAD      @D0L01      10
1336.00 CSE          MOVE FECEEC      @D0L01      1
1337.00 CSE          MOVEV FFEY      @D0L01      4
1338.00 CSE          MOVE FFEY      @D0L01      2
1339.00 CSE          MOVE FEVVAL      @D0L01      40
1340.00 CSE          MOVE FEVVAL      @D0L01      40
1341.00 CSE          MOVE FEVVAL      @D0L01      40
1342.00 CSE          MOVE FEVVAL      @D0L01      40
1343.00 CSE          MOVE FEKWR      @D0L01      30
1344.00 CSE          MOVE FELE      J@D0L01      1
1345.00 CSE          MOVE FEHIX      @D0L01      20
1346.00 CSE          Z-ADD1      @D0L01      110
1347.00 CSE          MOVE @D0L01      @1
1348.00 CSE          DO      @A
1349.00 CSE          MULT 10      @D0L01
1350.00 CSE          END
1351.00 CSE          END
1352.00 C*-----
1353.00 C*
1354.00 C*      dictionary parameters for - cost center
1355.00 C*
1356.00 CSE          MOVE 'BLANK'      FEDTRAI
1357.00 CSE          MOVEV 'MCU'      FEDTRAI
1358.00 CSE          CALL 'X9800E'      51
1359.00 C*          -----
1360.00 CSE          PARM              19800E
1361.00 CSE          FREER      IFREQ 'O'
1362.00 CSE          MOVE FEDGCR      @MNCU      40
1363.00 CSE          MOVE FEDTAT      T@MNCU      1
1364.00 CSE          MOVE FREC      @MNCU      1
1365.00 CSE          MOVE FEDTAG      @MNCU      40
1366.00 CSE          MOVE FEDTAD      @MNCU      10
1367.00 CSE          MOVE FECEEC      @MNCU      1
1368.00 CSE          MOVEV FFEY      @MNCU      4
1369.00 CSE          MOVE FFEY      @MNCU      2
1370.00 CSE          MOVE FEVVAL      @MNCU      40
1371.00 CSE          MOVE FEVVAL      @MNCU      40
1372.00 CSE          MOVE FEVVAL      @MNCU      40
1373.00 CSE          MOVE FEVVAL      @MNCU      40
1374.00 CSE          MOVE FEKWR      @MNCU      30
1375.00 CSE          MOVE FELE      J@MNCU      1
1376.00 CSE          MOVE FEHIX      @MNCU      20
1377.00 CSE          Z-ADD1      @MNCU      110
1378.00 CSE          MOVE @MNCU      @A
1379.00 CSE          DO      @A
1380.00 CSE          MULT 10      @MNCU
1381.00 CSE          END
1382.00 CSE          END
1383.00 C*-----
1384.00 C*
1385.00 C*      dictionary parameters for - category code - cost center of
1386.00 C*
1387.00 CSE          MOVE 'BLANK'      FEDTRAI
1388.00 CSE          MOVEV 'EP01'      FEDTRAI
1389.00 CSE          CALL 'X9800E'      51
1390.00 C*          -----
1391.00 CSE          PARM              19800E
1392.00 CSE          FREER      IFREQ 'O'
1393.00 CSE          MOVE FEDGCR      @E0P01      40
1394.00 CSE          MOVE FEDTAT      T@E0P01      1
1395.00 CSE          MOVE FREC      @E0P01      1

```

1396.00	CSE	MOVE FEDTRG	CRFP01	40
1397.00	CSE	MOVE FEDTRD	GRFP01	10
1398.00	CSE	MOVE FECDSC	FRFP01	1
1399.00	CSE	NOVELFRGY	GRFP01	4
1400.00	CSE	MOVE FERT	RRFP01	2
1401.00	CSE	MOVE FEDVAL	DRFP01	40
1402.00	CSE	MOVE FEVAL	ARFP01	40
1403.00	CSE	MOVE FELVAL	LRFP01	40
1404.00	CSE	MOVE FEUVAL	URFP01	40
1405.00	CSE	MOVE FEDWRK	WRFP01	30
1406.00	CSE	MOVE FELR	JRFP01	1
1407.00	CSE	MOVE FEHNIX	HRFP01	20
1408.00	CSE	Z-ADD1	#RFP01	110
1409.00	CSE	MOVE FRFP01	#A	
1410.00	CSE	DO #A		
1411.00	CSE	MULT 10	#RFP01	
1412.00	CSE	END		
1413.00	CSE	END		
1414.00	C*	-----		
1415.00	C*			
1416.00	C*	dictionary parameters for - category code - cost center 02		
1417.00	C*			
1418.00	CSE	MOVE *BLANK	FDTAI	
1419.00	CSE	NOVEL'RP02'	FDTAI	
1420.00	CSE	CALL 'X9800E'		81
1421.00	C*	-----		
1422.00	CSE	PARM	I9800E	
1423.00	CSE	FRERR IFEQ '0'		
1424.00	CSE	MOVE FEDSCR	DRFP02	40
1425.00	CSE	MOVE FEDTAT	TRFP02	1
1426.00	CSE	MOVE FEFC	DRFP02	1
1427.00	CSE	MOVE FEDTRG	CRFP02	40
1428.00	CSE	MOVE FEDTRD	GRFP02	10
1429.00	CSE	MOVE FECDSC	FRFP02	1
1430.00	CSE	NOVELFRGY	GRFP02	4
1431.00	CSE	MOVE FERT	RRFP02	2
1432.00	CSE	MOVE FEDVAL	DRFP02	40
1433.00	CSE	MOVE FEVAL	ARFP02	40
1434.00	CSE	MOVE FELVAL	LRFP02	40
1435.00	CSE	MOVE FEUVAL	URFP02	40
1436.00	CSE	MOVE FEDWRK	WRFP02	30
1437.00	CSE	MOVE FELR	JRFP02	1
1438.00	CSE	MOVE FEHNIX	HRFP02	20
1439.00	CSE	Z-ADD1	#RFP02	110
1440.00	CSE	MOVE FRFP02	#A	
1441.00	CSE	DO #A		
1442.00	CSE	MULT 10	#RFP02	
1443.00	CSE	END		
1444.00	CSE	END		
1445.00	C*	-----		
1446.00	C*			
1447.00	C*	dictionary parameters for - company		
1448.00	C*			
1449.00	CSE	MOVE *BLANK	FDTAI	
1450.00	CSE	NOVEL'CO'	FDTAI	
1451.00	CSE	CALL 'X9800E'		81
1452.00	C*	-----		
1453.00	CSE	PARM	I9800E	
1454.00	CSE	FRERR IFEQ '0'		
1455.00	CSE	MOVE FEDSCR	DRCO	40
1456.00	CSE	MOVE FEDTAT	TRCO	1
1457.00	CSE	MOVE FEFC	DRCO	1
1458.00	CSE	MOVE FEDTRG	CRCO	40
1459.00	CSE	MOVE FEDTRD	GRCO	10
1460.00	CSE	MOVE FECDSC	FRCO	1
1461.00	CSE	NOVELFRGY	GRCO	4
1462.00	CSE	MOVE FERT	RRCO	2
1463.00	CSE	MOVE FEDVAL	DRCO	40
1464.00	CSE	MOVE FEVAL	ARCO	40
1465.00	CSE	MOVE FELVAL	LRCO	40
1466.00	CSE	MOVE FEUVAL	URCO	40
1467.00	CSE	MOVE FEDWRK	WRCO	30
1468.00	CSE	MOVE FELR	JRCO	1
1469.00	CSE	MOVE FEHNIX	HRCO	20
1470.00	CSE	Z-ADD1	#RCO	110
1471.00	CSE	MOVE FRCO	#A	
1472.00	CSE	DO #A		
1473.00	CSE	MULT 10	#RCO	
1474.00	CSE	END		

```

1475.00 CBR          END
1476.00 C*
1477.00 C*
1478.00 C*   create or clear the demonstration user index
1479.00 C*   =====
1480.00 C*
1481.00 CBR          MOVE 'BLANKS'      PERRR
1482.00 CBR          CALL 'J98CICORJ'          99
1483.00 C*
1484.00 CBR          PARM 'FINDEX'      'PSCOBJT 10
1485.00 CBR          PARM 'QTEMP'      'PSLIB 10
1486.00 CBR          PARM '*USRIDX'    'PSTTPE 7
1487.00 CBR          PARM '*HCHK'     'PSPEX 10
1488.00 CBR          PARM '*HCHK'     'PSAUT 10
1489.00 CBR          PARM          'PSENR 1
1490.00 C*
1491.00 C*   if it doesn't exist, create it.
1492.00 C*
1493.00 CBR          PERRR      IFEQ '1'
1494.00 C*
1495.00 CBR          CALL 'QUSCRTUI'          99
1496.00 CBR
1497.00 CBR          PARM          $IIX      idx name/lib
1498.00 CBR          PARM 'BLANKS'  PSACT 10  extend Atcb
1499.00 CBR          PARM 'F'      PSXFL 1   length Atcb
1500.00 CBR          PARM          $IRECL  entry length
1501.00 CBR          PARM '1'      PSKIND 1   key insert
1502.00 CBR          PARM          $IKL  key length
1503.00 CBR          PARM 'O'      PSIMUP 1   delay update
1504.00 CBR          PARM 'O'      PSOPTH 1   opts = randn
1505.00 CBR          PARM '*ALL'    PSIDAU 10  public auth
1506.00 CBR          PARM $ITEXT    PSTEXT 50  index descr
1507.00 C*
1508.00 C*   if it does axis, clear it.
1509.00 C*
1510.00 CBR          ELSE
1511.00 CBR
1512.00 CBR          E-ADD@IBRT1  PERRYL
1513.00 CBR          E-ADD@IRECL PERRCL
1514.00 CBR          MOVE 'BLANK'  PKEY
1515.00 C*
1516.00 CBR          CALL 'XOCIDX
1517.00 C*
1518.00 CBR          PARM          $IIX      idx name/lib
1519.00 CBR          PARM 'D'      PSACTN 1   Action
1520.00 CBR          PARM 'EQ'     PSRULE 2   Action rule
1521.00 CBR          PARM          PKEYL 30   key length
1522.00 CBR          PARM          PKEY 120  key fields
1523.00 CBR          PARM          PERECL 30  entry length
1524.00 CBR          PARM          PEREC 120  entry
1525.00 CBR          PARM          PSEST 1   error status
1526.00 C*
1527.00 CBR          END
1528.00 C*
1529.00 C*
1530.00 C*   set subroutine execution flag.
1531.00 C*
1532.00 CBR          MOVE '1'          $998 1
1533.00 C*
1534.00 CBR          END998  ENDGR
1535.00 C*
1536.00 C*
1537.00 C*   SUBROUTINE 9999 - housekeeping
1538.00 C*   -----
1539.00 C*
1540.00 C*   Processing:  1. Load video screen text.
1541.00 C*                2. Retrieve screen title data area, test
1542.00 C*                   for unauthorized access, center video
1543.00 C*                   title and move to video screen.
1544.00 C*                3. Initialize key list.
1545.00 C*                4. Load roll keys.
1546.00 C*                5. Pass parameters.
1547.00 C*                6. Load error message array.
1548.00 C*                7. Initialize subfile display.
1549.00 C*
1550.00 CBR          9999  BEGIN
1551.00 C*
1552.00 C*
1553.00 C*   Required program parameters.
1554.00 C*
1555.00 C*   . . . No parameters passed
1556.00 C*

```

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 CSE      $AUTO      IFNE 'BLANK      $AUTO      1
1560.00 CSE      MOVE '1'
1561.00 CSE      END
1562.00 C*-----
1563.00 C*
1564.00 C*      Load video screen text.
1565.00 C*
1566.00 CSE      MOVE$PFILE      PKEY      10
1567.00 CSE      E-ADD006      PVTX#      30
1568.00 C/COPY JDECPY,COOAC
1569.00 C*-----
1570.00 C*
1571.00 C*      Load error messages array.
1572.00 C*
1573.00 CSE      MOVE '0001'      ENK,01      INV ACTION
1574.00 CSE      MOVE '0002'      ENK,02      INV Key
1575.00 CSE      MOVE '0003'      ENK,03      INV blanks
1576.00 CSE      MOVE '0004'      ENK,04      INV Data
1577.00 CSE      MOVE '0005'      ENK,05      INV Next Mbr
1578.00 CSE      MOVE '0007'      ENK,06      IN USS
1579.00 CSE      MOVE '0025'      ENK,07      INV Values
1580.00 CSE      MOVE '0026'      ENK,08      INV MCU
1581.00 CSE      MOVE '0027'      ENK,09      INV Desc rtl
1582.00 C*-----
1583.00 C*
1584.00 C*      Load invalid action code array.
1585.00 C*
1586.00 CSE      MOVE'      '      ENAC
1587.00 C*-----
1588.00 C*
1589.00 C*      initialize subfile display.
1590.00 C*
1591.00 CSE      E-ADD0      I1
1592.00 CSE      E-ADD15      $PAGE      30
1593.00 CSE      DO $PAGE
1594.00 CSE      ADD 1      I1
1595.00 CSE      MOVE'IN      SWIN
1596.00 CSE      WRITEINDEX2
1597.00 CSE      END
1598.00 CSE      E-ADD11      $SV11
1599.00 C*-----
1600.00 C*
1601.00 C*      Load system data.
1602.00 C*
1603.00 CSE      TIME      $WRK12      120
1604.00 CSE      MOVE $WRK12      $QEDT      60
1605.00 C*-----
1606.00 CSE      ENDP99      ENDR

```

File Servers

About File Servers

- Sometimes called I/O servers.
- Allow you to enhance the processing time of your program.
- Ease the maintenance of your programs by making your system more modular.

Eventually, every program should perform database functions using either a file server or a functional server.

Note: all logical files are accessed through servers by their based-on file. Embedded in one server, there may be many access paths available.

This section contains the following:

- [What is a File Server?](#)
- [What are the Advantages of Using a File Server?](#)
- [What are the Disadvantages of Using a File Server?](#)
- [How Does a File Server Function?](#)
- [What Are Control Parameters?](#)
- [What Are Returned Parameters?](#)
- [Implementing a File Server](#)
- [Searching for Key Lists](#)
- [Tips when Using File Servers](#)
- [Commonly Used File Servers](#)

What is a File Server?

A file server, or I/O server, is a server that performs RPG database operations.

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:

File Server	Description
XS	Input-Only/Caching Servers They can be used in place of 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 allow you to replace all RPG database operation codes for a given file with program calls. They can perform such functions as READ, CHAIN, and SETLL to a file.
X	Special Scrub & Edit Servers They can accept cost center, account numbers, numeric fields, or a data string 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 or the scrubbed data back to the calling program.

What are the Advantages of Using a File Server?

The advantages of using a file server are

- Minimized maintenance of your software
- Ability to change a physical file without having to make changes to application programs that use the file, or even having to recompile them
- Use of versions in future releases to allow programs from a previous release to run against a changed database
- Smoother transition from an old database to a new database. 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 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. After you load the control parameters, which contain information about the record you are retrieving, the file server converts the control parameters and returns a record to the program.

If you plan to use 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

Note: 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 may be called with two parameters:

```
For example:  CALL      'XF0101'      81
              - - - - - - - - - -
              PARM      PS@@@1
              PARM      I0101
```

PARAM	Description
PS@@@1	Contains all of the control parameters. It is contained in copy module I00XFSRV, and it is common to all file servers.
I(file name)	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).

What Are Control Parameters?

The parameter PS@@@1 is a Data Structure which 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.

PARM (Length)	Description
@@ACCS (1)	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).
@@OPER (10)	The operation to be performed to the file. The valid values are presented below: CHAIN Chain by key list or RRN CLOSE Close the access path DELET Delete current record or by key or RRN EXIST Test existence of record by key OPEN Open access path (optional) READ Read next record READE Read next equal key READP Read previous record REDPE Read previous equal key SETGT Set greater than key SETHV Set greater than with *HIVAL SETLL Set lower limit by key SETLV Set lower limit with *LOVAL UPDAT Update locked record UPDATC Update current record WRITE Write new record UNLCK Unlock current record
@@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.

PARM (Length)	Description
@@CHGR (1)	<p>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:</p> <p>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)</p> <p>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.</p> <p>W Call the Changed Record Window (P0000U) that will prompt you for what action to take. Use this option with interactive programs only.</p>
@@KLST (10)	<p>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 (@@ACCS = R or S).</p> <p>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.</p>
@@KNUM (5,0)	<p>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.</p>
@@FMT (10)	<p>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.</p>
@@#RRN (9,0)	<p>The relative record number for RRN access.</p>
I (file name)	<p>Record image for updates and writes. This parameter is optional for OPEN, CLOSE, DELET, SETHV, SETLV, and UNLCK operations.</p>

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.

PARAM (Length)	Description
@@IOR(3)	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
@@ERR (10)	Short description of the cause of the problem (invalid, relock, error, required, deleted, chgrec).
@@ERRS (10)	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.
@@#RRN (9,0)	Returns the relative record number of the record just read (both input and output).
I (filename)	Returns an exact duplicate of the record image (both input and output).

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 the JD Edwards base release value (“A73” or “A81”) 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, I010171.
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 in the calling program.
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.

Note: 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” on the DW Additional Parameters screen.

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. There are some servers with information in the F93201 which is accessed by the following process. Otherwise, using the Where Used function on a file entry in SVR may show which “X” (server) programs access a particular file.

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.

```

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  Server Name . . . . . XF0101
6. CASE Profiles    Skip To Key Name . . .
7. Function Key D
8. Vocabulary Ove
          Key      Access  S Key
          List     Path    L Item  Description
ABKY01 F0101LA    AN8  Address Number . . . . .
ABKY02 F0101LB    DC   Description - Compressed
ABKY03 F0101LC    PH1  Phone Number . . . . .
ABKY04 F0101LD    PA8  Parent Number . . . . .
          AN8  Address Number . . . . .
ABKY05 F0101LE    * AN8 Address Number . . . . .

Selection or command
====> KL

Thur, Jan 6, 1994

```

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, 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 error out the program, it would be a good idea to save the keys and reset the pointer (CHAIN or SETLL) upon returning.

File Server Examples

```

Seq No. U C*----- Mod DC*e
1.00 C* 09.11.92
2.00 C* 10.11.92
3.00 C* Index of Examples: 10.11.92
4.00 C* ----- 10.11.92
5.00 C* 10.11.92
6.00 C* Scan For: To Find: 10.11.92
7.00 C* ----- 10.11.92
8.00 C* 1.1 File Server Calls 10.11.92
9.00 C* 1.1.1 Chain 10.11.92
10.00 C* 1.1.2 Close 10.11.92
11.00 C* 1.1.3 Delete 10.11.92
12.00 C* 1.1.4 Existence Test 10.11.92
13.00 C* 1.1.5 Open 10.11.92
14.00 C* 1.1.6 Read 10.11.92
15.00 C* 1.1.7 Read Equal 10.11.92
16.00 C* 1.1.8 Read Previous 10.11.92
17.00 C* 1.1.9 Read Previous Equal 10.11.92
18.00 C* 1.1.10 Set Greater Than 10.11.92
19.00 C* 1.1.11 Set Lower Level 10.11.92
20.00 C* 1.1.12 Update 10.11.92
21.00 C* 1.1.13 Write 10.11.92
22.00 C* 1.1.14 Unlock 10.11.92
23.00 C* 2.1.1 X09021 19.01.92
24.00 C* ----- 10.11.92
25.00 C* 10.11.92
26.00 C* *1.1 File Server Calls: 10.11.92
27.00 C* 09.11.92
28.00 C* *Determine from P92KL what the key list name is for the 09.11.92
29.00 C* access path being used; this name is moved to the @RELST. 09.11.92
30.00 C* If the operation uses a key list, determine how many keys 09.11.92
31.00 C* the key list represents; this number is E-ADDED to @SENUM. 09.11.92
32.00 C* the format is the release level (A61) and can be moved to 10.11.92
33.00 C* @SPMT in @999 once for the rest of the calls. 10.11.92
34.00 C* ----- 09.11.92
35.00 C* 09.11.92
36.00 C* 1.1.1 Chain: 10.11.92
37.00 C* 10.11.92
38.00 C* Old Code: 10.11.92
39.00 CSE ABEY02 CHAINI0101C 8199 10.11.92
40.00 C* 10.11.92
41.00 C* New Code: 10.11.92
42.00 CSE NOVEL'A61' @SPMT 10.11.92
43.00 CSE NOVEL'ABEY04' @RELST 10.11.92
44.00 CSE NOVEL'CHAIN' @SOPER 10.11.92
45.00 CSE MOVE 'H' @SELCK 05.12.92
46.00 CSE E-ADD2 @SENUM 10.11.92
47.00 CSE CALL 'XFO101' 10.11.92
48.00 C* ----- 10.11.92
49.00 CSE PARM P9981 10.11.92
50.00 CSE PARM PARM I0101 10.11.92
51.00 CSE @SECR COMP 'EL' 99 10.11.92
52.00 CSE @SECR COMP 'NP' 81 10.11.92
53.00 C* ----- 10.11.92
54.00 C* 10.11.92
55.00 C* 1.1.2 Close: 10.11.92
56.00 C* 10.11.92
57.00 C* Old Code: 10.11.92
58.00 CSE CLOSEF09021D 99 10.11.92
59.00 C* 10.11.92
60.00 C* New Code: 10.11.92
61.00 CSE NOVEL'A61' @SPMT 10.11.92
62.00 CSE NOVEL'GLEY04' @RELST 10.11.92
63.00 CSE NOVEL'CLOSE' @SOPER 10.11.92
64.00 CSE CALL 'XF0902' 10.11.92
65.00 C* ----- 10.11.92
66.00 CSE PARM P9981 10.11.92
67.00 CSE PARM PARM I0902 10.11.92
68.00 CSE @SECR COMP 'EUN' 99 10.11.92
69.00 C* ----- 10.11.92
70.00 C* 10.11.92
71.00 C* 1.1.3 Delete: 10.11.92
72.00 C* 10.11.92
73.00 C* Old Code: 10.11.92
74.00 CSE DELETI0101B 99 10.11.92
75.00 C* 10.11.92
76.00 C* New Code: 10.11.92
77.00 CSE NOVEL'A61' @SPMT 10.11.92
78.00 CSE NOVEL'ABEY02' @RELST 10.11.92
79.00 CSE NOVEL'DELET' @SOPER 10.11.92

```

80.00	CSE		CALL 'XF0101'			1-00:1-1:-2-
81.00	C*					
82.00	CSE		PARM	P0001		10.11.92
83.00	CSE		PARM	I0101		10.11.92
84.00	CSE		SBICR	COMP 'RL'	99	10.11.92
85.00	C*					10.11.92
86.00	C*	Old Code:				10.11.92
87.00	CSE	ASCT08	DELETI0101B		8199	10.11.92
88.00	C*					10.11.92
89.00	C*	New Code:				10.11.92
90.00	CSE		NOVEL'AG1'	GBPWT		10.11.92
91.00	CSE		NOVEL'AMV02'	GBCLGT		10.11.92
92.00	CSE		NOVEL'DELET'	GBOPER		10.11.92
93.00	CSE		S-ADD3	GBNFRM		10.11.92
94.00	CSE		CALL 'XF0101'			10.11.92
95.00	C*					10.11.92
96.00	CSE		PARM	P0001		10.11.92
97.00	CSE		PARM	I0101		10.11.92
98.00	CSE		SBICR	COMP 'RL'	99	10.11.92
99.00	CSE		SBICR	COMP 'RF'	82	10.11.92
100.00	C*					10.11.92
101.00	C*					10.11.92
102.00	C*	1.1.4 Existence Test:				10.11.92
103.00	C*					09.11.92
104.00	C*	Old Code:				09.11.92
105.00	CSE	ASCT02	SETLLI0101D		9982	10.11.92
106.00	C*					09.11.92
107.00	C*	New Code:				09.11.92
108.00	CSE		NOVEL'AG1'	GBPWT		10.11.92
109.00	CSE		NOVEL'EPK01'	GBCLGT		10.11.92
110.00	CSE		NOVEL'EKIST'	GBOPER		10.11.92
111.00	CSE		S-ADD3	GBNFRM		09.11.92
112.00	CSE		CALL 'XF0101'			10.11.92
113.00	C*					09.11.92
114.00	CSE		PARM	P0001		09.11.92
115.00	CSE					10.11.92
116.00	CSE		SBICR	COMP 'YES'	82	10.11.92
117.00	CSE		SBICR	COMP 'ERR'	99	10.11.92
118.00	C*					09.11.92
119.00	C*					10.11.92
120.00	C*	1.1.5 Open:				10.11.92
121.00	C*					10.11.92
122.00	C*	Old Code:				10.11.92
123.00	CSE		OPEN F0006			10.11.92
124.00	C*					10.11.92
125.00	C*	New Code:				10.11.92
126.00	CSE		NOVEL'AG1'	GBPWT		10.11.92
127.00	CSE		NOVEL'HCY01'	GBCLGT		10.11.92
128.00	CSE		NOVEL'OPEN'	GBOPER		10.11.92
129.00	CSE		CALL 'YF0006'			10.11.92
130.00	C*					10.11.92
131.00	CSE		PARM	P0001		10.11.92
132.00	CSE		PARM	I0006		10.11.92
133.00	C*					10.11.92
134.00	C*					10.11.92
135.00	C*	1.1.6 Read:				10.11.92
136.00	C*					10.11.92
137.00	C*	Old Code:				10.11.92
138.00	CSE		READ I0901A		9982	10.11.92
139.00	C*					10.11.92
140.00	C*	New Code:				10.11.92
141.00	CSE		NOVEL'AG1'	GBPWT		10.11.92
142.00	CSE		NOVEL'GMK01'	GBCLGT		10.11.92
143.00	CSE		NOVEL'READ'	GBOPER		10.11.92
144.00	CSE		NOVEL'H'	GBLOCK		05.12.92
145.00	CSE		CALL 'XF0901'			10.11.92
146.00	C*					10.11.92
147.00	CSE		PARM	P0001		10.11.92
148.00	CSE		PARM	I0901		10.11.92
149.00	CSE		SBICR	COMP 'EOP'	82	10.11.92
150.00	CSE		SBICR	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:				17.11.92
154.00	C*					10.11.92
155.00	C*	Old Code:				10.11.92
156.00	CSE	ASCT02	HEADRI0101C		9987	10.11.92
157.00	C*					10.11.92
158.00	C*	New Code:				10.11.92
159.00	CSE		NOVEL'AG1'	GBPWT		10.11.92
160.00	CSE		NOVEL'ASCT01'	GBCLGT		10.11.92
161.00	CSE		NOVEL'HEAD'	GBOPER		10.11.92
162.00	CSE		NOVEL'H'	GBLOCK		05.12.92

163.00	CSR		CALL 'XP0101'			10.11.92		
164.00	C*		-----			10.11.92		
165.00	CSR		FRAM	FRAM	P0001	10.11.92		
166.00	CSR				I0101	10.11.92		
167.00	CSR		@@ICR	COMP 'HE'		87	10.11.92	
168.00	CSR		@@ICR	COMP 'HL'		99	10.11.92	
169.00	C*		-----			10.11.92		
170.00	C*					17.11.92		
171.00	C*		1.1.8 Read Previous:			17.11.92		
172.00	C*					17.11.92		
173.00	C*		Old Code:			17.11.92		
174.00	CSR			NRADPI0901B		9982	17.11.92	
175.00	C*						17.11.92	
176.00	C*		New Code:				17.11.92	
177.00	CSR			MOVEL 'A61'		@@PMT	17.11.92	
178.00	CSR			MOVEL 'GMY02'		@@KLT	17.11.92	
179.00	CSR			MOVEL 'NRADP'		@@OPR	17.11.92	
180.00	CSR			MOVE 'H'		@@LCK	05.12.92	
181.00	CSR			CALL 'XP0901'			17.11.92	
182.00	C*			-----			17.11.92	
183.00	CSR			FRAM	FRAM	P0001	17.11.92	
184.00	CSR					I0901	17.11.92	
185.00	CSR			@@ICR	COMP 'ROP'		52	17.11.92
186.00	CSR			@@ICR	COMP 'RL'		99	17.11.92
187.00	C*			-----			17.11.92	
188.00	C*						10.11.92	
189.00	C*		1.1.9 Read Previous Equal:				10.11.92	
190.00	C*						10.11.92	
191.00	C*		Old Code:				10.11.92	
192.00	CSR			ARXY04	NRDPI0101C		9987	10.11.92
193.00	C*						10.11.92	
194.00	C*		New Code:				10.11.92	
195.00	CSR			MOVEL 'A61'		@@PMT	10.11.92	
196.00	CSR			MOVEL 'ARXY02'		@@KLT	10.11.92	
197.00	CSR			MOVEL 'NRDPR'		@@OPR	10.11.92	
198.00	CSR			MOVE 'H'		@@LCK	05.12.92	
199.00	CSR			CALL 'XP0101'			10.11.92	
200.00	C*			-----			10.11.92	
201.00	CSR			FRAM	FRAM	P0001	10.11.92	
202.00	CSR					I0101	10.11.92	
203.00	CSR			@@ICR	COMP 'HE'		87	10.11.92
204.00	CSR			@@ICR	COMP 'HI'		99	10.11.92
205.00	C*			-----			10.11.92	
206.00	C*						10.11.92	
207.00	C*		1.1.10 Set Greater Than:				10.11.92	
208.00	C*						10.11.92	
209.00	C*		Old Code:				10.11.92	
210.00	CSR			GRKEY	SETGTI0902A		5498	10.11.92
211.00	C*						10.11.92	
212.00	C*		New Code:				10.11.92	
213.00	CSR			MOVEL 'A61'		@@PMT	10.11.92	
214.00	CSR			MOVEL 'GRKY01'		@@KLT	10.11.92	
215.00	CSR			MOVEL 'SETGT'		@@OPR	10.11.92	
216.00	CSR			2-ADD2		@@KHU	10.11.92	
217.00	CSR			CALL 'XP0902'			10.11.92	
218.00	C*			-----			10.11.92	
219.00	CSR			FRAM	FRAM	P0001	10.11.92	
220.00	CSR					I0902	10.11.92	
221.00	CSR			@@ICR	COMP 'HF'		54	10.11.92
222.00	CSR			@@ICR	COMP 'HR'		99	10.11.92
223.00	C*			-----			10.11.92	
224.00	C*		Old Code:				10.11.92	
225.00	CSR			*RIVAL	SETGTI0902A		99	10.11.92
226.00	C*						10.11.92	
227.00	C*		New Code:				10.11.92	
228.00	CSR			MOVEL 'A61'		@@PMT	10.11.92	
229.00	CSR			MOVEL 'GRKY01'		@@KLT	10.11.92	
230.00	CSR			MOVEL 'SETTR'		@@OPR	10.11.92	
231.00	CSR			CALL 'XP0902'			10.11.92	
232.00	C*			-----			10.11.92	
233.00	CSR			FRAM	FRAM	P0001	10.11.92	
234.00	CSR					I0902	10.11.92	
235.00	CSR			@@ICR	COMP 'HR'		99	10.11.92
236.00	C*			-----			10.11.92	
237.00	C*						10.11.92	
238.00	C*		1.1.11 Set Lower Limit:				10.11.92	
239.00	C*						10.11.92	
240.00	C*		Old Code:				10.11.92	
241.00	CSR			ARX01	SETLLI0101C		549985	10.11.92
242.00	C*						10.11.92	
243.00	C*		New Code:				10.11.92	
244.00	CSR			MOVEL 'A61'		@@PMT	10.11.92	
245.00	CSR			MOVEL 'ARXY02'		@@KLT	10.11.92	

246.00	CER		NOVEL'SETILL'	@@CFER		10.11.92
247.00	CER		Z-ADD1	@@CHUM		10.11.92
248.00	CER		CALL 'XF0101'			10.11.92
249.00	C*		-----			10.11.92
250.00	CER		PARM	@@@81		10.11.92
251.00	CER		PARM	I0101		10.11.92
252.00	CER	@@SIOR	COMP 'EOF'		84	10.11.92
253.00	CER	@@SIOR	COMP 'EQ'		85	10.11.92
254.00	CER	@@SIOR	COMP 'ERR'		99	10.11.92
255.00	C*					10.11.92
256.00	C*	Old Code:				10.11.92
257.00	CER	*LOCAL	SETILLI0101C		99	10.11.92
258.00	C*					10.11.92
259.00	C*	New Code:				10.11.92
260.00	CER		NOVEL'AS1'	@@SPMT		10.11.92
261.00	CER		NOVEL'ARBY03'	@@CLGT		10.11.92
262.00	CER		NOVEL'GETLV'	@@CFER		10.11.92
263.00	CER		CALL 'XF0101'			10.11.92
264.00	C*		-----			10.11.92
265.00	CER		PARM	@@@81		10.11.92
266.00	CER		PARM	I0101		10.11.92
267.00	CER	@@SIOR	COMP 'ERR'		99	10.11.92
268.00	C*		-----			10.11.92
269.00	C*					10.11.92
270.00	C*	1.1.12 Update:				10.11.92
271.00	C*					10.11.92
272.00	C*	Old Code:				10.11.92
273.00	CER		UPDATI0902A		99	10.11.92
274.00	C*					10.11.92
275.00	C*	New Code:				10.11.92
276.00	CER		NOVEL'AS1'	@@SPMT		10.11.92
277.00	CER		NOVEL'ARBY01'	@@CLGT		10.11.92
278.00	CER		NOVEL'UPDAT'	@@CFER		10.11.92
279.00	CER		CALL 'XF0902'			10.11.92
280.00	C*		-----			10.11.92
281.00	CER		PARM	@@@81		10.11.92
282.00	CER		PARM	I0902		10.11.92
283.00	CER	@@SIOR	COMP 'ERR'		99	10.11.92
284.00	C*					10.11.92
285.00	C*	Old Code:				10.11.92
286.00	C*		Read...			10.11.92
287.00	C*		Unlock...			10.11.92
288.00	C*		Chain...		82	10.11.92
289.00	C*					10.11.92
290.00	CER		UPDATI0902A		99	10.11.92
291.00	C*					10.11.92
292.00	C*	New Code:				10.11.92
293.00	C*		Read with no lock...			10.11.92
294.00	C*					10.11.92
295.00	CER		NOVEL'AS1'	@@SPMT		10.11.92
296.00	CER		NOVEL'ARBY01'	@@CLGT		10.11.92
297.00	CER		NOVEL'UPDAT'	@@CFER		10.11.92
298.00	CER		Z-ADD4	@@CHUM		10.11.92
299.00	CER		CALL 'XF0902'			10.11.92
300.00	C*		-----			10.11.92
301.00	CER		PARM	@@@81		10.11.92
302.00	CER		PARM	I0902		10.11.92
303.00	CER	@@SIOR	COMP 'NF'		82	10.11.92
304.00	CER	@@SIOR	COMP 'ERR'		99	10.11.92
305.00	C*		-----			10.11.92
306.00	C*					09.11.92
307.00	C*	1.1.13 Write:				10.11.92
308.00	C*					09.11.92
309.00	C*	Old Code:				09.11.92
310.00	CER		WRITEI0101K		99	10.11.92
311.00	C*					09.11.92
312.00	C*	New Code:				09.11.92
313.00	CER		NOVEL'AS1'	@@SPMT		10.11.92
314.00	CER		NOVEL'ARBY11'	@@CLGT		10.11.92
315.00	CER		NOVEL'WRITE'	@@CFER		10.11.92
316.00	CER		CALL 'XF0101'			10.11.92
317.00	C*		-----			09.11.92
318.00	CER		PARM	@@@81		09.11.92
319.00	CER		PARM	I0101		10.11.92
320.00	CER	@@SIOR	COMP 'ERR'		99	10.11.92
321.00	C*		-----			09.11.92
322.00	C*					09.11.92
323.00	C*	1.1.14 Unlock:				10.11.92
324.00	C*					09.11.92
325.00	C*	Old Code:				09.11.92
326.00	CER		EXCPTURLOCK			10.11.92
327.00	C*					10.11.92
328.00	OI0101A	E	UNLOCK			10.11.92

229.00	C*					09.11.92
230.00	C*	New Code:				09.11.92
231.00	CSE		MOVEL 'A61'	08PMT		10.11.92
232.00	CSE		MOVEL 'ASHY01'	08CLST		10.11.92
233.00	CSE		MOVEL 'UNLCK'	08OPRR		10.11.92
234.00	CSE		CALL 'XF0101'			09.11.92
235.00	C*		-----			09.11.92
236.00	CSE		PARM	PO001		09.11.92
237.00	CSE		PARM	I0101		10.11.92
238.00	CSE	0010R	COMP 'ERR'		99	10.11.92
239.00	C*		-----			09.11.92
240.00	C*					10.11.92
241.00	C*	2.1.1	X0901:			19.01.93
242.00	C*					19.01.93
243.00	CSE		CALL 'X0901'			19.01.93
244.00	C*					19.01.93
245.00	CSE		PARM '2'	WCALC 1		19.01.93
246.00	CSE		PARM	WCO 5		19.01.93
247.00	CSE		PARM	WCO 60		19.01.93
248.00	CSE		PARM	WFM 20		19.01.93
249.00	CSE		PARM	WFT 20		19.01.93
250.00	CSE		PARM	WCTY 20		19.01.93
251.00	CSE		PARM	WEDT 1		19.01.93
252.00	CSE		PARM '1'	WGGY 1		19.01.93
253.00	C*					19.01.93
254.00	C*					19.01.93
255.00	C*					19.01.93
256.00	C*	2.2.1	X0901:			19.01.93
257.00	C*					19.01.93
258.00	CSE		CALL 'X0901'			19.01.93
259.00	C*		-----			19.01.93
260.00	CSE		PARM '1'	PG01N 1		19.01.93
261.00	CSE		PARM EPAM	PG01M 1		19.01.93
262.00	CSE		PARM '1'	PG1M0 1		19.01.93
263.00	CSE		PARM EPGLA	PGANI 29		19.01.93
264.00	CSE		PARM *BLANK	PGNCU 12		19.01.93
265.00	CSE		PARM *BLANK	PGOBJ 6		19.01.93
266.00	CSE		PARM *BLANK	PGSUB 8		19.01.93
267.00	CSE		PARM	PGTRM 4		19.01.93
268.00	C*					19.01.93
269.00	C*					19.01.93
270.00	C*					19.01.93
271.00	C*	2.3.1	X0006:			19.01.93
272.00	C*					19.01.93
273.00	CSE		CALL 'X0006'			19.01.93
274.00	C*		-----			19.01.93
275.00	CSE		PARM 'I'	PGMOD 1	output mode	19.01.93
276.00	CSE		PARM	PGMOD 1	input mode	19.01.93
277.00	CSE		PARM SPNCU	PGNCU 12	cost center	19.01.93
278.00	CSE		PARM	PGTRM 4	error flag	19.01.93
279.00	CSE		PARM	I0006	F0006 record	19.01.93
280.00	C*					19.01.93
281.00	C*					19.01.93

Commonly Used File Servers

The following is a list of commonly used file servers:

File Server	Description	Notes
X0005	User Defined Codes Server	Retrieve Only
X0006	Retrieve Cost Center Master	Retrieve & Scrub
XF0006	Cost Center I/O	Add/Change/Delete
X0010	Automatic Next Numbering	Retrieve & Increment
X9203	DD Alpha Description	Retrieve Only
X9800E	Data Dictionary Info	Editing Info
XF0101	Address Book I/O	Add/Change/Delete
XS0101LA	Address Book	Retrieve Only
X0901	Account Master	Retrieve & Formats
XF0901	Account Master I/O	Add/Change/Delete
X41LOCN	Location Format	

File Server	Description	Notes
X41LOT	Lot Number Assignment	
X41DUP	Lot Master Duplicate	Edits
X4101	Item Master	Retrieve & Edit
X4108	Lot Master Update	Creates & Updates
X4111	Write to Item Ledger	Writes Only
XF4111	CARDEX I/O	Retrieve Only
XF42119	Sales History I/O	Add/Change/Delete
XF42199	Sales Detail Ledger I/O	Add/Change/Delete
XF43199	Purchasing History I/O	Add/Change/Delete

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

This section contains the following:

- [What Are Functional Servers?](#)
- [What Are the Advantages of Using a Functional Server?](#)
- [Setting Up Business Rules for an Entry Program](#)
- [How Does a Functional Server Function?](#)

What Are Functional Servers?

A functional server is a program that performs all transaction validation and database updates.

The functional server removes from the application program the burden of performing edit and update operations. Rather, this functionality is placed within the server.

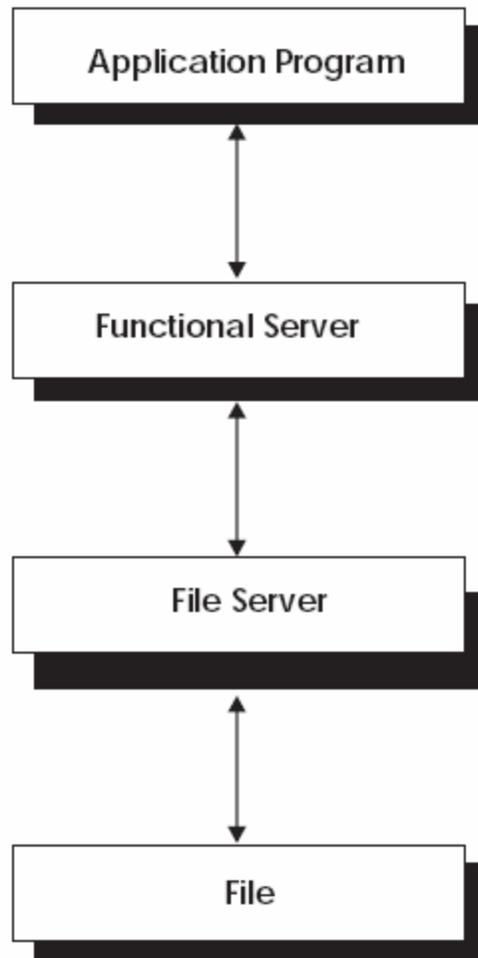
A functional server is a called program. The application program calling the server must tell the server what action to perform for every transaction. In turn, the functional server will record error flags and update flags and return error messages to the application program for use when determining the result of a call to a server.

Functional servers have the following naming convention:

XT (file name) (server version)

For example: The functional server for file F0411 would appear as XT0411Z1.

The following diagram depicts the flow of a typical program using a functional server:



What Are the Advantages of Using a Functional Server?

Advantages of using a functional server include:

- Minimized maintenance and versioning of your software.
- Ability to isolate data editing routines and file updates.
- Increased flexibility because multiple programs can use the same functional server.
- Smoother transition from an old database to a new database. Instead of modifying all programs, you will only have to apply a new set of functional servers.
- Ability to implement one functional server at a time without affecting your entire system.

What Are the Disadvantages of Using a Functional Server?

Disadvantages of functional servers include:

- 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

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. JD Edwards World provides DREAM Writer version ZJDE0001 as the default functional server version for your entry programs.

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.

Caution: 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 receives the data that the user entered and loads it into a user space.
- It performs its functionality on the data.
- Finally, it returns the requested data back to the calling program using the user space. If any errors occur, the system loads them into a user index.

The system uses the following interfaces to communicate with the functional server:

- The call parameters
- The control fields within each user space line
- The error index

Functional Server Highlights

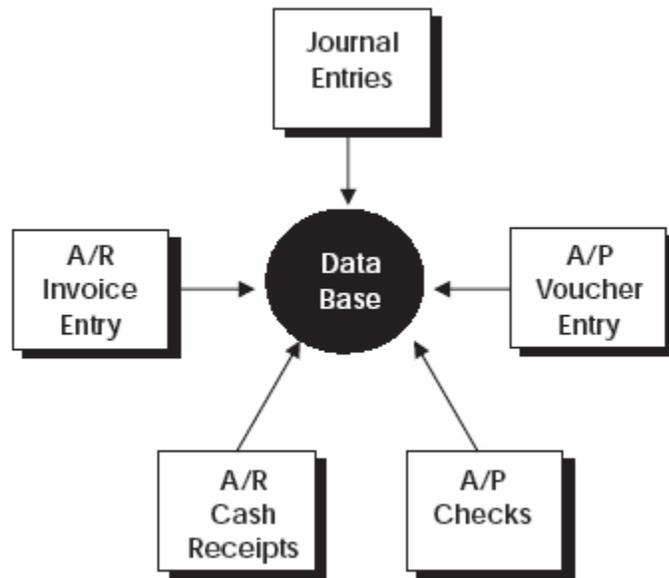
Functional servers are usefully for many things, highlights include:

- 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:

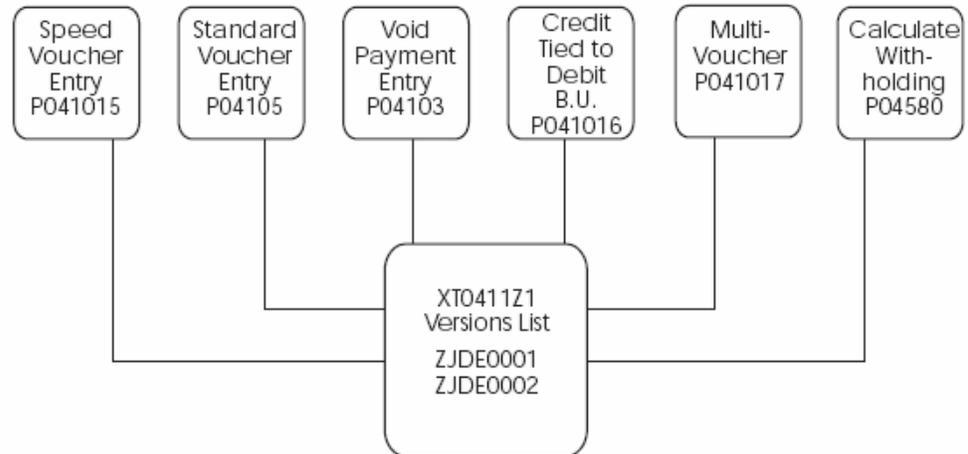
- Journal Entries
- A/P Voucher Entry
- A/P Checks
- A/R Invoice Entry
- A/R Cash Receipts



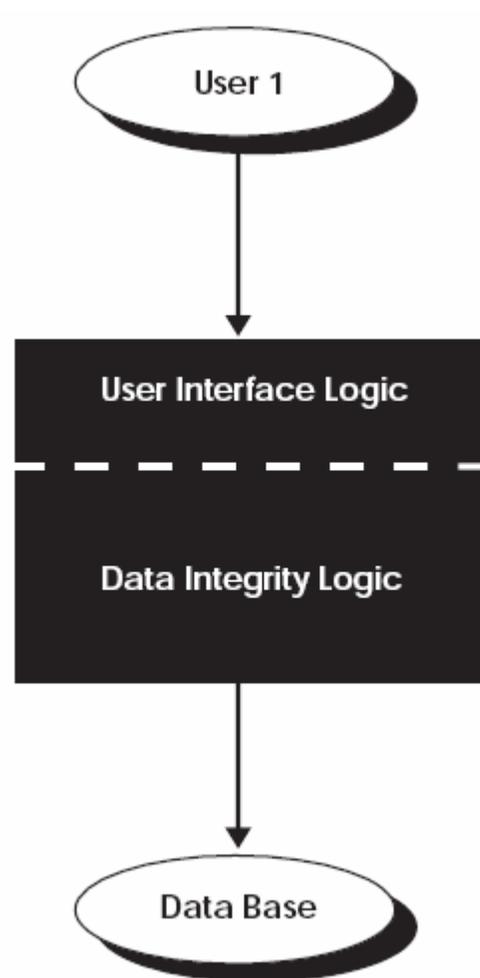
JD Edwards World uses one program for each part or transaction of the system.

Example: Voucher Processing Functional Server

The following graphic shows the programs that use the voucher processing functional server. JD Edwards World 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

Aspects of the user interface logic include:

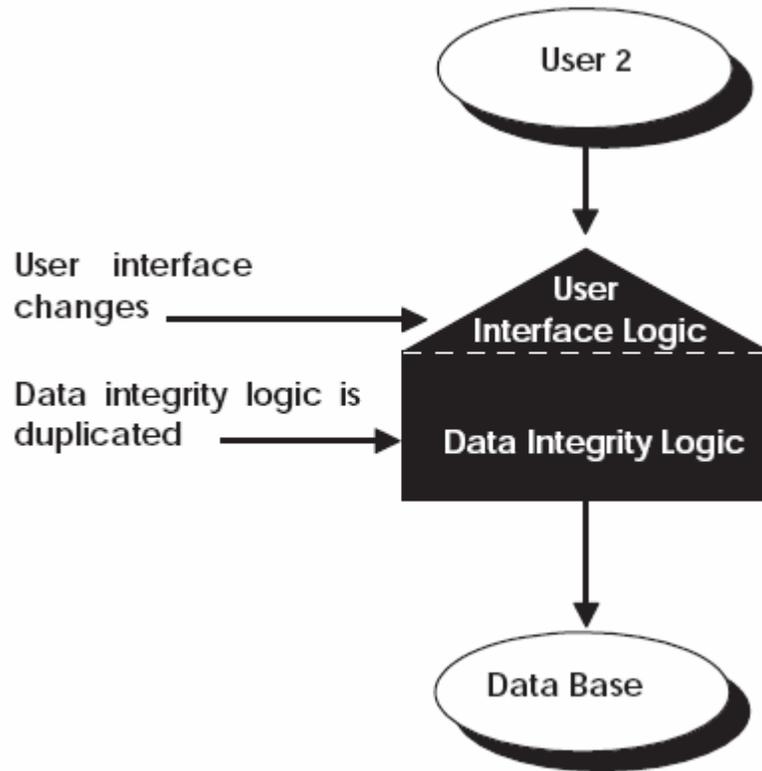
- Screen format
- Field formatting
- Help functions
- Error message display
- Touch and feel

Data Integrity Logic

Aspects of the data integrity logic include:

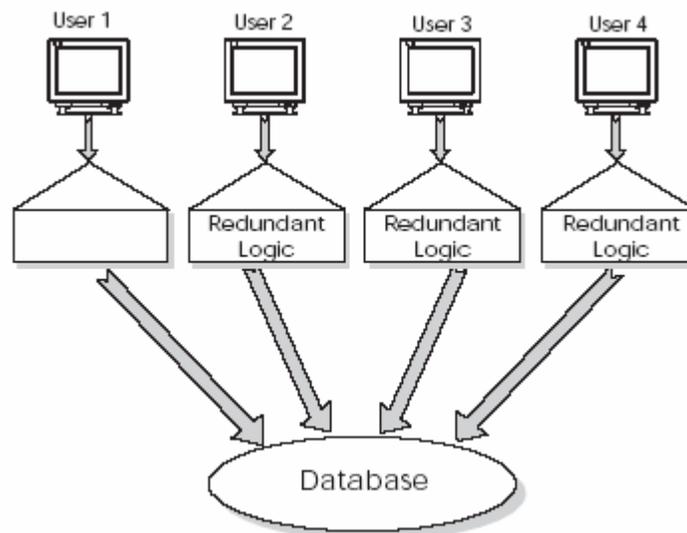
- Field editing
- Multi-field editing
- Transaction editing
- Default logic
- Error message selection
- Tax processing
- Currency processing
- Database update

Example - Traditional Architecture - Alternative Method of Entry



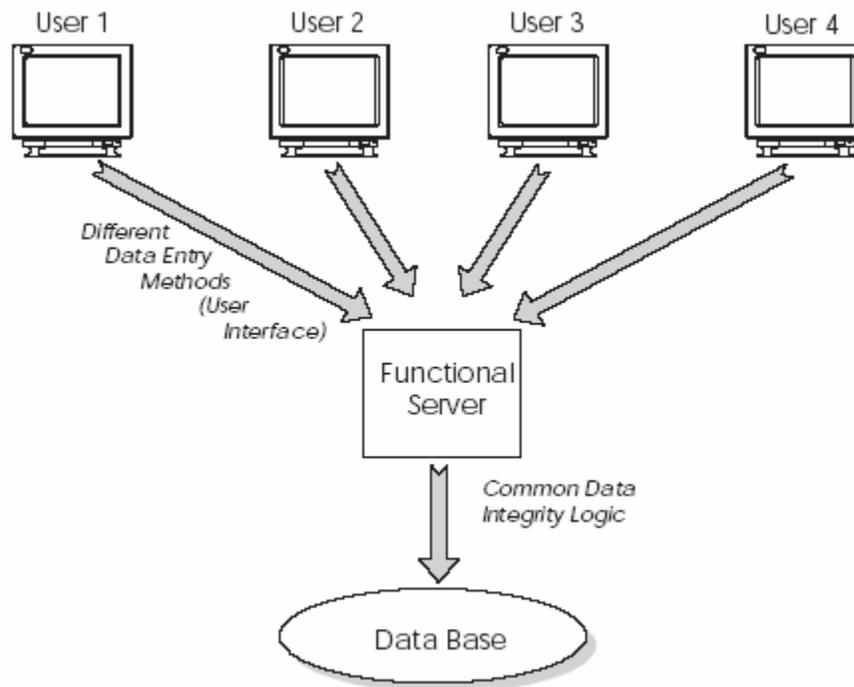
If a user wanted the screen to look different, the User Interface Logic would have to change. The Data Integrity Logic would be duplicated.

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 numerous times.

Example - JD Edwards World Open Application Architecture - Various Entry Methods

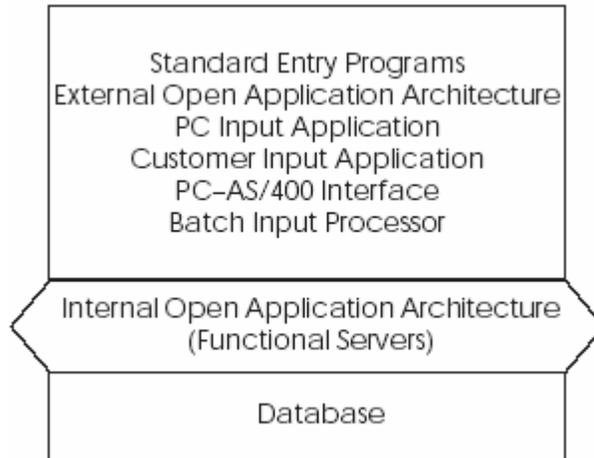


The creation of a Functional Server allows you to maintain the Data Integrity Logic in one common program. The Functional Server becomes separated from each User Integrity Logic program. All of the User Integrity Logic programs access one Functional Server to interface with the database. This concept 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. Advantages of the Open Application Architecture include:

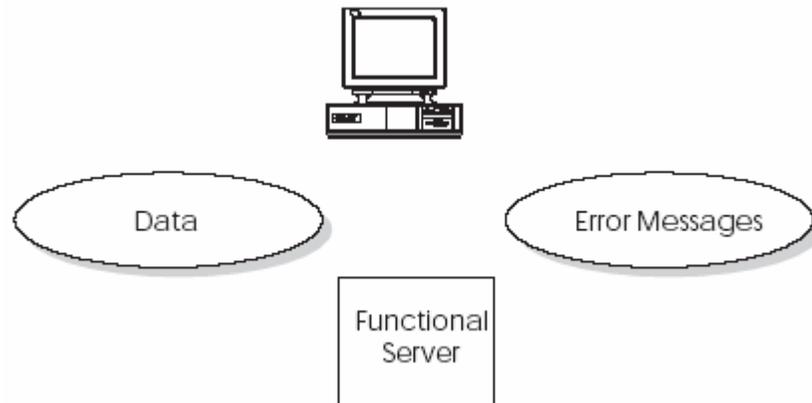
- Automatic consistency
- Reduced maintenance burden
- Stability of custom code
- Separation of development efforts
- Performance enhancements



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.

Parameters that are 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
 - 1) Control section
 - 2) Application specific section
 - 3) 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
 - 1) Key
 - Application ID
 - Line number (assigned by application program)
 - Data item in error
 - Error code
 - 2) 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 the 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.
 - .JD Edwards World version of QUSCHGUS API
 - Updates a user space beginning at offset x for length
 - Similar to CHGDTAARA command
- Reading from the user space QUSRTVUS.
 - 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 applies to all transaction lines in the input user space.

The first parameter is a Data Structure. The following fields are from the #PPARM Data Structure defined in the I00FS@@ copy module.

PARAM (Length)	Description
#PFUNC (1)	Specifies a function code. The valid values are: 0 Edit and Update 1 Edit only 2 Update only I Inquire
#PVERS (10) (3 before A6)	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 ZJDE0001. This allows global processing options to be set at the server level, instead of for each program.
#PSPCN (20)	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.
#PSPCB (9,0)	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.
#PNBRL (5,0)	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.

PARM (Length)	Description
#PWARN (1)	This parameter contains a code explaining how you want warnings to be handled. The valid values are: 0 Normal warning processing 1 Treat warnings as errors 2 Ignore warnings
#PCYCL (1)	This parameter is only used if the #PWARN parameter specifies normal warning processing. The valid values are: 0 No cycle, all cycle processing ignored 1 First cycle, all warning messages are sent to the program 2 Second cycle, only warning messages not previously sent are sent to the program
#PDFTC (1)	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.
#PXATP (3)	The application specific transaction type.
#PLVL (1)	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.
#PPROG (10)	The name of the calling program. This is used by the server to update the program name field in the updated database records.
#PAPPL (10)	The application ID value used for writing entries to the error index. Generally, this may be the same value as the calling program.
#PFLDS (4,0)	The number of fields which have been loaded to the Field Names Array parameter.
#PFMT (10)	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.
#PEDIT (1)	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.
#PUPDT (5,0)	The number of database updates which occurred. This will allow the program to know whether any updates actually occurred.
#PERR (4)	Specifies any errors that occurred within the server. A non-blank value indicates a fatal error occurred.

PARM (Length)	Description
#PFERR (4)	Contains the first error message found during editing.
#PFDTA (4)	Contains the data item of the first field which had an error during editing.
#P#MDE (1)	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.
#PCRCD (3)	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.
#PCRR (15,7)	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.
#PIDXN (20)	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.
#PSPCL (5,0)	The total length of each user space record. This includes both the user space control fields and the database record format.
#PSPEC (250)	This is a data structure which is redefined by each server. Generally, this will contain the application key fields which a specific server uses.

The second parameter will be an array.

PARM (Length)	Description
Variable Array	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.

Control Fields within the User Space

The input user space can contain multiple lines for each control field. The following fields are defined in the #SSPCR Data Structure in the I00FS@@ copy module.

PARAM (Length)	Description
#SPCAC (1)	The line action code. The valid values are: A Add the record D Delete the record C Change the record U Change the record if it already exists, otherwise add the record V Void the record
#SPCID (15,0)	Used by the program to uniquely identify each line in the user space. (optional)
#SPCER (1)	The line error code. X = the line is OK 1 = some warnings 2 = errors.
#SPCUP (1)	The line update code. 0 = the line was not updated 1 = updated
#SPCRR (9,0)	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.
#SPCMN (2,0)	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.
#SPCPG (12)	Allows the program to store up to 12 bytes of information with each user space record.
#SPCAP (200)	Any application specific information which must be passed to the server for each transaction line, but is not contained within the transaction record format.
Application fields	Externally described record format for the transaction record.

Error Message Index Line (C00RIX)

The output error message index contains warning and error messages issued for each line in the user space. The following fields are from the #IIDXR Data Structure defined in the I00FS@@ copy module. The structure of the message index line is as follows:

Field (Length)	Explanation
#IDXAP (10)	The application identifier from the input parameter. Allows a program to access only its error messages.
#IDXID (15,0)	The line identifier from the input user space.
#IDXFN (10)	The data item portion of the field name.
#IDXER (4)	Contains the data dictionary error message code.
#IDXWN(1)	Line warning code 1 = Warning, 2 = Error.
#IDXMD (87)	Not used.

Example - Functional Server Program Sections

<pre> ***** E* Copy Composite Member for Functional Server E* E/COPY JDECPY,E00PS@@ ***** </pre>	Copy module containing generic data structures for functional server.
<pre>I/COPY JDECPY,I00XPSRV</pre>	Contains control parameter list for file servers
<pre>I/COPY JDECPY,I010161</pre>	Contains record image of F0101 version A6.1 for file servers.
<pre> CLEARPS@@ MOVE\$SVCO KY@@ 91 CALL 'XS0010' ----- PARM PS@@ PARM DS0010 </pre>	Call to file server XS0010 to retrieve company currency code.
<pre> *IN\$1 IFEQ '0' RT@@ ANDEQ ' ' MOVE *BLANKS PS@@ MOVE\$COCRCO KY@@ CALL 'XS0013' 91 ----- PARM PS@@ DS0013 PARM DS0013 PS0013 RT@@ IFNE 'H' CVCDEC ANDNE*BLANK MOVE CVCDEC \$CDO ENDIF ENDIF </pre>	Call to file server XS0013 to retrieve display decimals.

```

MOVEL'A61'      @@PMT
MOVEL'ABKY01'  @@KLST
MOVEL'CHAIN'   @@OPER
MOVE 'Y'       @@LOCK
Z-ADD1         @@KNUM
CALL 'XF0101'
-----
PARM          PS@@1
PARM          I0101
@@IOR        COMP 'NF'                               81
    
```

Call to file server XF0101 to retrieve record

```

*IN81
IFBQ '0'
ADD $#FC       ABAFCY
MOVEL'A61'     @@PMT
MOVEL'ABKY01' @@KLST
MOVEL'UPDAT'   @@OPER
CALL 'XF0101'
-----
PARM          PS@@1
PARM          I0101
@@IOR        COMP 'ERR'                               98
    
```

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.

MOVEL#XIDXN   #PIDXN      index name
MOVEL#SPAR    #PSPEC      applicatio
MOVE #EDOP    #PFUNC      function
MOVEL$#311    #FVERS      DW version
Z-ADD1        #ENBERL     number of lines
Z-ADD$#ARBG   #PSPCB      space offset
MOVE ##IGNW   #PWARN      warning handler
MOVE ##OPF    #PLVL       detail level
MOVE ##OPF    #PDFTC      default on chg
MOVE ##PROG   #PPROG      program name
MOVE 'INV'    #FXATP       type
MOVE #ARSN    #PSPCN      space name
Z-ADD#ARSL    #PSPCL      space length
Z-ADD$@AR     #PFLDS      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
      MOVEL#SUGL   #SSPCD
Current user space offset

      Z-ADD$#GLBG  #SPCOF
Set update flag
      MOVE #OFF    #SPCUP
General Ledger record
      MOVEAGL01   @#SSPC
Application specific line data
      MOVEL#SSGL  #SPCAP
Write record to user space
      CALL 'X98CHGUS' #PCHUS      01
      -----
      ENDIP                          ##edit
```

Write records to user space
for functional server.

```

Load G/L Functional Server Specific Parameters

          MOVE *ZERO      #GLDOC      One-to-One Rel
          MOVE 'RP'       #GLDCT      Document Type
          MOVE $SVKCO     #GLKCO      Document Co.
          Z-ADD$GLDG      #GLDG       G/L Date
          Z-ADD$GLDG#     #GLDG#      G/L Date
$FICU    IPEQ ' '
          Z-ADD*ZERO      #GLICU      Batch Number
          ELSE
          Z-ADD$ICU       #GLICU      Batch Number
          ENDIF
          MOVE 'I'        #GLICT      Batch Type
          MOVE $SVC0      #GLCO       Company
          MOVE *BLANKS    #GLMOD      Add a Model
          MOVE *BLANKS    #GLIMD      Change a Model
          MOVE *BLANKS    #GLRDI      Redistribute JE
          MOVE #ARSN      #GLCSN      A/R Spc Name
          MOVE #ARL       #GLCFM      A/R Spc Pmt
          MOVE #ARSL      #GLCLN      A/R Spc Length
          MOVE #OFF       #GLCDG      #GLCD
          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       #FVERS      DW version
          MOVE #GLSN      #FSPCN      space name
          Z-ADD$GLBG      #FSPCB      space offset
          Z-ADD1          #FNBRL      number of lines
          MOVE #IGNW      #FWARN      warning handler
          Z-ADD*ZERO      #PCRR      Exchange Rate
          MOVE #AOFF      #PCYCL      cycle nes
          MOVE #AOFF      #PDPTC      default on chg
          MOVE 'INV'      #PYATP      type
          MOVE #AOFF      #PLVL      detail level
          MOVE #PROG      #PFRG      program name
          Z-ADD$GGL       #PFLDS      number of field
          MOVE *BLANKS    #PFMT      format name
          MOVE$#GGL       #PFMT
          MOVE *BLANKS    #PAMDE      mode of entry
          MOVE *BLANKS    #PCRCO      currency code
          MOVE *BLANKS    #PCRR      exchange rate
          MOVE$XIDXN      #PIDXN      index name
          Z-ADD$GLSL      #PSPCL      space length
          MOVE$SPGL       #PSPCC      application par

          CALL 'XT0911Z1'          91
          -----
          PARM          #PPARM
          PARM          @GLN
    
```

Call functional server
XT0911Z1

```

User space description

          MOVE$#SUAR      #SSPCD

Current user space offset

          Z-ADD$#ARBG     #SPCOP

Read record from user space

          CALL 'QOSRTVUS'#PRTUS    91
          -----
    
```

Retrieve record from user
space.

```

Create Functional Server Objects for XT0311Z1

      CLEAR#PCRT
      MOVE #HAD      #PCRTF
      MOVE *BLANK    #PCRTN
      MOVE#XT0311Z1'#PCRTN

      CALL 'X00991 '  ----- 81
-
      PARM          #PCRT

```

Create user space and
user index for XT0311Z1.

```

Create Functional Server Objects for XT0911Z1

      CLEAR#PCRT
      MOVE #HAD      #PCRTF
      MOVE *BLANK    #PCRTN
      MOVE#XT0911Z1'#PCRTN

      CALL 'X00991 '  ----- 81
-
      PARM          #PCRT

```

Create user space and
use index for
XT0911Z1.

Available Functional Servers

Functional Server	Description	Notes
XT0101Z1	Address Book	
XT0311Z1	Accounts Receivable	
XT0311Z1E	Accounts Receivable	User Exit
XT0411Z1	Accounts Payable	
XT0411Z1E	Accounts Payable	User Exit
XT06116Z1	Payroll Time Entry	
XT0901Z1	Account Master	
XT0911Z1	Journal Entry	
XT0911Z1E	Journal Entry	User Exit
XT4102Z1	Item Balance	

Source Debugger

About Source Debugger

There are two types of programs that can be executed under the JD Edwards World 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 JD Edwards World Source Debugger is a tool designed to help you determine where an error 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 cannot 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.

This section contains the following:

- [Using Debugger with an Interactive Program](#)
- [Using Debugger with a Batch Program](#)

Before You Begin

- The JD Edwards debug tool is based on IBM debug objects. You must have authority to the IBM commands STRDBG, ADDPGM, ADDBKP at a minimum. IBM has a STRISDB command which interacts with the program source. As of V5R1, the STRDBG command allows use of F10 to step through the program statements once the initial breakpoint is set.

Note: In order for the program to be debugged to be visible, it must have been compiled on the current machine.

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 JD Edwards World Source Debugger
- Execute the program being debugged

Source File	Type the file name that contains the source code of your program. If *OBJECT is entered, the source file name and library will be taken from the program object. If *SPLF is entered, the compile listing will be used for the source statements. This will include the statements included from COPY members which allows them to be debugged as well.
Library	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.

2. Enter the correct values in the proper fields and press Enter to start the Source Debugger.
- Any time the program being debugged is executed, the source code will display in debug mode, until you end the Source Debugger. (ENDDDBG in all cases.)

To execute the program being debugged

If 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 and include the required parameters
- Selection/Menu

After you have executed the program, the first thing you will see is the program source code.

```

3701                                     JDE Visual Debug      JDESRC      JDFSRC
Scan:                                     - Address Book - Who's Who      P01051
Current Breakpoint: /0001
01.00  H/TITLE P01051 - Address Book - Who's Who
02.00  H* -----
03.00  H*
04.00  H*      Copyright (c) 1985,1986
05.00  H*      J. D. Edwards & Company
06.00  H*
07.00  H*      This unpublished material is proprietary to
08.00  H*      J. D. Edwards & Company. All rights reserved.
09.00  H*      The methods and techniques described herein are
10.00  H*      considered trade secrets and/or confidential.
11.00  H*      Reproduction or distribution, in whole or in part,
12.00  H*      is forbidden except by express written permission
13.00  H*      of J. D. Edwards & Company.
14.00  H* -----
15.00  H*
16.00  H*
17.00  F*      PROGRAM REVISION LOG
18.00  F*

F2=JDE Command Line  F5=ADDBKP  F6=ADDBKP w/prompt  F7=DSPPGMVAR
F8=CHGPGMVAR  F13=Display Indicators  F16/15=Scan Fwd/Bkwd  F24=More

```

Note: The source code is displayed in browse mode, so you cannot edit or change any code.

Using Debugger with a Batch Program

To use Debugger with a batch program you should complete the following tasks:

- Sign on to the program environment
- Initiate the JD Edwards World Source Debugger
- Execute the program
- Set the break point
- Continue execution

To initiate the JD Edwards World 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 JD Edwards World 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.

Debug Program (JDEDBG)

Type choices, press Enter.

Program Name:	<u>J928400</u> _____	Name
Source File:	<u>JDESFC</u> _____	Name, *OBJECT, *SPLF
Library:	<u>JDFSRC</u> _____	Name

Bottom

F3=Exit F4=Prompt F5=Refresh F10=Additional parameters F12=Cancel
F13=How to use this display F24=More keys

2. Enter the JD Edwards World 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.

```

                                Debug Program (JDEDBG)

Type choices, press Enter.

Program Name: . . . . . P928400      Name
Source File: . . . . . JDESRC        Name, *OBJECT, *SPLF
Library: . . . . . JDESRC            Name

                                Bottom
F3=Exit   F4=Prompt  F5=Refresh  F10=Additional parameters  F12=Cancel
F13=How to use this display  F24=More keys

```

To execute the program

Since you are executing a batch program interactively, you must call the CL Program from a command line.

1. CALL CL program (The parameters are usually “program name” and “version”).
2. The CL Program source code displays.

```

93701                                JDE Visual Debug      JDESRC      JDFSRC
Scan:                                /0001                J928401
Current Breakpoint: /0001
0001.00 /******
0002.00 /*
0003.00 /* Program. . . . . J928401
0004.00 /*
0005.00 /* Description. . . . . Inventory by Cost Center
0006.00 /*
0007.00 /* Program Revision Log
0008.00 /* -----
0009.00 /*
0010.00 /*
0011.00 /*      Date      Programmer      Description
0012.00 /*      -----      -----      -----
0013.00 /*      11/10/93      FB908300      SAR # 00365595
0014.00 /******
0015.00 J928401: PGM      (&PSPID &PSVERS)
0016.00 /*
0017.00 /* ----- Define program file(s) and variable(s). -----
0018.00 /*

F2=JDE Command Line  F5=ADDBKP  F6=ADDBKP w/prompt  F7=DSPPGMVAR
F8=CHGPGMVAR  F13=Display Indicators  F16/15=Scan Fwd/Bkwd  F24=More

```

Note: 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 &JOBTYPE 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 &JOBTYPE.

```

93701          JDE Visual Debug          JDESRC          JDFSRC
Scan:          Current Breakpoint: /0001          J928401
0044.00 /*
0045.00 /* ----- Override Printer files to one spool file. ----- *
0046.00 /*
0047.00 OVRPRTF FILE(R98COVER ) TOFILE(R928401) SHARE(*YES)
0048.00 OVRPRTF FILE(R98RPTH ) TOFILE(R928401) SHARE(*YES)
0049.00 OVRPRTF FILE(R928401) SHARE(*YES)
0050.00 /*
0051.00 /* ----- Retrieve job name and submitting message queue. ----- *
0052.00 /*
0053.00 RTVJOBA JOB(&JOBID) SEMMSGQ(&PMSGQ) TYPE(&JOBTYPE)
0054.00 IF COND(&JOBTYPE='1') THEN(DO)
0055.00 SNDPGMMSG MSGID(JDE9991) MSGF(QJDEMSG) +
0056.00 MSGDTA('J928401') TOPGMQ(*EXT)

F2=JDE Command Line F5=ADDBKP F6=ADDBKP w/prompt F7=DSPPGMVAR
F8=CHGPGMVAR F13=Display Indicators F16/15=Scan Fwd/Bkwd F24=More

```

2. Press F5 anywhere on the line containing &JOBTYPE 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 the &JOBTYPE variable to something other than 1 so that the compare to '1' fails.
2. To change the value of &JOBTYPE, 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 . . . . . '2'
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 JD Edwards World Source Debugger

Function	Description
F2	To display a JD Edwards World command line window, press F2.
F3	Once the program hits a breakpoint or when you first enter the source, F3 will allow the program to continue processing.
F5	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.

Example F5

```

93701                               JDE Visual Debug      JDESRC      JDFSRC
Scan:                               P01051
Current Breakpoint: /0001
0319.00 C*
0320.00 C      $AUTO      CASEQ'1'      S003      24
0321.00 C*
0322.00 C      END
0323.00 C*
0324.00 C*      Begin normal program processing.
0325.00 C*
0326.00 C*
0327.00 C      *INLR      DOWEQ'0'
0328.00 C*
0329.00 C*      If subfile page display not set, set subfile page display.
0330.00 C*
0331.00 C      #SFRNO      IFLE 0
0332.00 C      Z-ADD1      #SFRNO
0333.00 C      END
0334.00 C*
0335.00 C*      Write video screen.
0336.00 C*      @$MEMO determines which format is written

F2=JDE Command Line F5=ADDBKP F6=ADDBKP w/prompt F7=DSPPGMVAR
F8=CHGPGMVAR F13=Display Indicators F16/15=Scan Fwd/Bkwd F24=More

```

Function	Description
F6	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.

Example F6

```

Add Breakpoint (ADDBKP)

Type choices, press Enter.
Statement identifier . . . . . > 62100      Character value
+ for more values
Program variables:
Program variable . . . . . *NONE
-
Basing pointer variable . . .
-
+ for more values
-
Output format . . . . . *CHAR          *CHAR, *HEX
Program . . . . . > P01051          Name, *DPTPGM

F3=Exit  F4=Prompt  F5=Refresh  F10=Additional parameters  F12=Cancel  More...
F13=How to use this display  F24=More keys
    
```

Use the prompt, after pressing F10, to assign a skip value or breakpoint conditions.

Function	Description
F7	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.

Example F7

```

Display Program Variables

Program . . . . . : P01051
Recursion level . . . . . : 1
Start position . . . . . : 1
Format . . . . . : *CHAR
Length . . . . . : *DCL

Variable . . . . . : *IN99
Type . . . . . : CHARACTER
Length . . . . . : 1
*...+...1...+...2...+...3...+...4...+...5
'0'

Variable . . . . . : *IN93
Type . . . . . : CHARACTER
Length . . . . . : 1
*...+...1...+...2...+...3...+...4...+...5
'0'

Press Enter to continue.

F3=Exit  F12=Cancel
    
```

Function	Description
F8	To change the value of a variable, press F8 and type the correct values in the prompt screen.

Example F8

```

Change Program Variable (CHGPGMVAR)

Type choices, press Enter.

Program variables:
Program variable . . . . .
-
- Basing pointer variable . . .
-
- + for more values
-
New value . . . . .
Program . . . . . > P01051      Name, *DFTPGM

                                           Bottom
F3=Exit  F4=Prompt  F5=Refresh  F10=Additional parameters  F12=Cancel
F13=How to use this display  F24=More keys

```

Function	Description
F10	Move Line to Top of Page.
F12	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	To display the current values of all indicators, press F13.

Example 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'
 11 '0' '0' '0' '0' '0' '0' '0' '0' '0'
 21 '0' '0' '0' '0' '0' '0' '0' '0' '0'
 31 '0' '0' '0' '0' '0' '0' '0' '0' '0'
 41 '0' '0' '0' '0' '0' '0' '0' '0' '0'
 51 '0' '0' '0' '0' '0' '0' '0' '0' '0'

Press Enter to continue.

F3=Exit  F12=Cancel

```

Function	Description
F15	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	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	To display a command line, press F21.

ENDDBG End Debug

To stop the JD Edwards World Source Debugger, enter ENDDBG from a command line. You can not enter ENDDBG 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.

Note: You can remove a single program from debug mode by using the RMVPGM (remove program) command.

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.

This section contains the following:

- [To Work with Software Scan and Replace](#)
- [Report](#)
- [Guidelines](#)

To Work with Software Scan and Replace

1. From the Computer Assisted Programming menu (G93), choose Developer's Workbench. From the Developer's Workbench menu (G9362), choose Software Scan and Replace.

```
98810                               Software Scan & Replace

System code. . . . . 55                (Blank = all)
Function code. . . . . RPG            (Blank = all)
Specific object. . . . .              (Generic = *)    (Blank = all)
File ID. . . . . JDESRC
Source library . . . . . DEVSRCL      (Defaults to source libr in member master)
Scan argument:
  I00SC
  (If search argument contains imbedded blanks enclose argument with >.)
Replacement argument:
  I00RSC
  (If argument contains imbedded blanks enclose argument with >.)
Column replacement: Beginning column - ____ Ending column - ____
Replacement Overflow Code . . . . . _
Insert Source From: File. . ____ Libr. . ____ Member. ____
```

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 33) 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 Ovrfl:
      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 009200

```

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 on the job queue.

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

General Performance Issues

Following are some performance issues you should consider when executing JD Edwards World software, changing current JD Edwards World programs, or writing new programs:

- Purge your files on a regular basis to avoid excess, unnecessary records existing in files. The REUSEDLT *YES parameter on a Physical file can be used on files where records are deleted to minimize file expansion and contraction. G9645/14 will present a list of P98999 versions which can be used to RGZPFM the named files. The RGZPFM command, in general, removes deleted records and rewrites the remaining records in a file.
- 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 than a 1 and 0.
- Consider flexibility versus performance when using User Defined Codes, Vocabulary Overrides, and loading Data Dictionary values extensively.

DREAM Writer

One definition of “Performance Problem” is that some Dream Writers take a long time. The usual symptom is that the user does a Work with Submitted Jobs and sees the job spending a long time “indexing”. Programs which have a very large based-on file are the prime suspects. P09800 is not a suspect!

The problem in these cases often involves the IBM Query File Optimizer. Simply stated, the optimizer tries to find an existing access path (logical file) that it can use to select records. If it can't find one or finds one that will return more than 20% of the file, it builds a whole new access path which takes a long time for large files.

Possible solutions:

Solution 1

The first assumption is that the applicable IBM PTFs have been applied.

1. On the DW Additional Parameters screen, there is a File Output Type field. An F1 on this field shows that values 2 and 4 use a logical file. If an existing logical should be used, enter a 2 in the File Output Type and the file name in the Override Logical File field. This setup often makes the DW run faster since the IBM Query File Optimizer is bypassed by using an LF rather than the OPNQRYF.
2. To have DW create the Logical File for you, enter a 4 in the File Output Type and blanks in the Override Logical file field. This option will use the DW Data Selection and Sequencing parameters to create a Logical File over the based-on physical, and will name it by replacing the P in the program name with an F and adding a 00n suffix.
3. **IMPORTANT:** You must then run the DW so the LF will be created. Then, before going back into the DW with a 2 to change, (which will cause the LF to be deleted) you must find the file using a DSPDDBR on the based-on file. Do RNMOBJ so DW will not find this new file. Then, go back to the Versions list and enter a 2 to change the DW version. DW will automatically have changed the File Output Type to a 2 and have entered the name of the logical file it created. Enter the file name from the above RNMOBJ command.
4. When the File Output Type is a 2, the options for Data Selection Values and Data Sequencing Values will not show up on the change window. This is because the values in the LF will be used. If the values need to be changed, go to Step 3C.
5. On the DW version list, if you use option 7, you will see either the OPNQRYF statement or the DDS for the logical file.

Solution 2

The based-on file can be changed to a logical file attached to the Physical file, but then the File Output Type must be set to 1. This reintroduces the OPNQRYF command into the process using the Query File Optimizer. The Data Selection and Sequencing options are now available.

7 Group Jobs

Overview to Group Jobs

About Group Jobs

Group Jobs is an IBM concept which allows the user to switch between sessions on the Series i. This involves communication with the Series i. Most users now use IBM Client Access which can have several sessions active. Switching between these sessions does not require Series i communication. The following is a description of JD Edwards's implementation of IBM 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 sign on
- Execute (or run) CL and fast path commands from a single command line
- Execute (or run) JD Edwards World 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.

This section covers the following tasks:

- Access the .JD Edwards World 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 JD Edwards World Group Job window

About the JD Edwards World Group Job window

This section contains the following:

- [Accessing the JD Edwards World Group Job window](#)
- [Creating New Group Jobs](#)
- [Activating Suspended Group Jobs](#)
- [Terminating Group Jobs](#)
- [Changing to Non-Group Mode](#)
- [Signing Off with Suspended Group Jobs](#)
- [Work with Non-JD Edwards World Group Jobs](#)
- [Advanced Functions of the JD Edwards World Group Job window](#)

Before You Begin

For a user to access the JD Edwards World Group Job window at any time, the Esc (ATTN) key program should be set to call the JD Edwards World Group Job window program (P98GRP).

To set the ATTN key program

1. From the Security Officer Menu (G94), choose User Information.

```
0092                User Information                Action Code. . . . . I

User ID. . . . . TEACH
Library List . . . . . QTEMP JDFOBJ COMMON PRODDATA JDFSRC OGPL

User Security:
  User Key . . . . . A J K DP F
  Initial Menu to Execute. . . . . A          Allow Command Entry (Y/N). Y
  Initial Program to Execute . . . . .          Allow Menu Traveling (Y/N) Y
  Allow Fast Path (Y/N) . . . . .          Allow Fast Path (Y/N) . . . Y
Menu Level. . . . .
User Type. . . . . TEACHER
User Class/Group . . . . .
Batch Job Queue. . . . . QBATCH
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
```


To sign off with suspended group jobs

Select one of the following methods:

1. Press F18 within the JD Edwards World Group Job window.
2. Enter SIGNOFF, 90, or “..” on the JD Edwards World Group Job Cmd/HS command line.

Note: Because group jobs are created under one sign on, all group jobs are terminated when the signoff command is executed.

Work with Non-JD Edwards World Group Jobs

To work with non-JD Edwards World group jobs

To create group jobs that call a program outside the JD Edwards World software, the JD Edwards World Group Job window allows an external program to be executed. In addition, the Esc (ATTN) key can be pressed within the external program and still allow access to the JD Edwards World group jobs.

1. To call an external program, press F11 within the JD Edwards World Group Job window.

The following illustrates what will be displayed when F11 is pressed.

```

Change Library List (CHGLIBL)

Type choices, press Enter.

Libraries for current job . . . > QTEMP           Name, *SAME, *NONE
                                > TCA302OBJ
                                > JDFOBJ
                                > TCA302DTA
                                > A3SHARE
                                > TRNSHARE
                                > TCA302SRC
                                > JDPSRC
                                > VAPAY2JLIB
                                > VBPAY2JLIB
                                > VCPAY2JLIB
                                > VPAYLIB
                                > QPRT5225
                                + for more values > QGPL
Current library . . . . . *SAME           Name, *SAME, *CRTDFT

Bottom
F3=Exit  F4=Prompt  F5=Refresh  F12=Cancel  F13=How to use this display
F24=More keys

```

2. Complete the Change Library List screen.

You can add 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 JD Edwards World group jobs:

- QTEMP
- Library containing F9220 (JD Edwards World Vocabulary Overrides)
- Library containing F0082 (JD Edwards World Menu Master)
- Library containing F0092 (JD Edwards World User Information)

- Library containing JD Edwards World 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.

```

                                Call Program (CALL)
Type choices, press Enter.
Program . . . . . _____ Name
Library . . . . . *LIBL _____ Name, *LIBL, *CURLIB
Parameters . . . . . _____
                                + for more values _____

                                Bottom
F3=Exit  F4=Prompt  F5=Refresh  F12=Cancel  F13=How to use this display
F24=More keys

```

When the CALL command is executed, the external program will be executed.

- To work with a JD Edwards World group job, the Esc (ATTN) key can be pressed to display the JD Edwards World Group Job window.
- Any suspended group job can be activated from the JD Edwards World Group Job window.

Advanced Functions of the JD Edwards World Group Job window

JD Edwards World Hidden Selections

Hidden Selections are commands and features of the JD Edwards World products that are not available through a menu selection.

- Most JD Edwards World Hidden Selections (31+) can be executed from the command line at the bottom of the JD Edwards World Group Job window.
- The JD Edwards World Hidden Selection window (HS) can be used to display and execute hidden selections.
- JD Edwards World Hidden Selection (Menu) Security is used when users execute hidden selections.
- No JD Edwards World Menus or JD Edwards World Hidden Selections related to menus are allowed, (for example 27 or 29).

Entering Commands

You can enter any command on the command line at the bottom of the JD Edwards World Group Job window.

- Press F4 to prompt for a command.
- Place a “?” in front of a command to prompt.
- 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.
- JD Edwards World 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 screen from the JD Edwards World Group Job window (For example, JD Edwards World 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 JD Edwards World User Information option found on A94.

Securing the CHGJOB Command (HS33)

HS 33 uses the IBM command WRKSBJOB. On this command screen, a user can access the CHGJOB command to move jobs to a different queue or change priorities.

Caution: Securing the CHGJOB command is problematic since JD Edwards World uses this command during the sign on process.

As of JD Edwards World release A81CU5 or A73CU15, the following workaround is available.

1. Objects J00WSJ, P00WSJ, V00WSJ and X00WSJ should be in the JD Edwards World object library.
2. Change the ZHIDDEN menu selection with SELECTION33 on it so the Option Key has J00WSJ rather than J00SBMJOB.
3. Sign out of the environment and back in. HS33 will present the information on a JD Edwards World menu which enables Function key and Option key security.

Note: Use Knowledge Document WST-00-0023 for IBM releases below V4R5.

JD Edwards World Group Job window Summary

The program allows you to:

- Create up to 16 jobs per sign on
- Execute commands, JD Edwards World hidden selections, JD Edwards World Fast Path Command, and JD Edwards World Fast Path Menu Execution

Function Key	Description
F3	Exit the JD Edwards World Group Job window
F4	Prompt a command
F5	Create a new JD Edwards World group job
F6	Submit job to batch
F8	JD Edwards World Menu Word Search
F9	Retrieve previous command
F11	Create a new Non-JD Edwards World group job
F13	Display all fast path commands
F18	SIGNOFF all group jobs

Selection Exits	Description
4	Activate a suspended group job
9	End a group job

The JD Edwards World Group Job window 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 window

About the Attention MENU Window

The JD Edwards World Attention Menu window program is a generic program that allows you to access up to 15 predefined programs via the Esc (ATTN) Key. The 15 predefined programs are associated with options on a JD Edwards World Menu.

Note: Each user can be assigned a different JD Edwards World Menu

Before You Begin

To access the JD Edwards World Attention Menu window at any time, the Set Attention Program field on the JD Edwards User Information video should be set to a JD Edwards World Menu, for example *G92.

The following illustrates how the Esc (ATTN) key program is set in the JD Edwards World software. The User Information screen can be accessed from the Security Officers Menu (G94).

0092	User Information	Action Code. I
User ID.	TEACH	
Library List	QTEMP JDFOBJ COMMON PRODDATA JDESRC OGPL	
<u>User Security:</u>		
User Key	A J K DP F	
Initial Menu to Execute.	A	Allow Command Entry (Y/N). Y
Initial Program to Execute		Allow Menu Traveling (Y/N) Y
Menu Level.		Allow Fast Path (Y/N). . . Y
User Type.	TEACHER	
User Class/Group		
Batch Job Queue	QBATCH	
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.	*G92	
F6=Display/Lang Pref F9=Library Inquiry F21=Print Lib List F24=More Keys		

Note: An * (asterisk) must precede the menu name.

Accessing the JD Edwards World Attention Menu window

After the Esc (ATTN) key program has been set up for the JD Edwards World software you can access the JD Edwards World attention menu window.

To access the JD Edwards World attention menu window

1. Sign off and sign back on to reset the Esc (ATTN) key program within the JD Edwards World Menu Driver.
2. Press the Esc (ATTN) key and the menu options for the menu will be displayed as follows.

```

G0                                J.D. Edwards & Company
                                General Business Systems

... GENERAL BUSINESS SYSTEMS
2. Address Book
3. General Accounting
4. Accounts Payable
5. Accounts Receivable
6. Financial Reporting
7. Modeling & Allocatio! 00AMNU-----Group Jobs -----JDED
8. Fixed Asset           O . . . Description . . . . . _Status_
9. Payroll              - Original Job           Active
10. Human Resources     - Software Versions Repository
11. Electronic Mail     - Data Dictionary
                        - CASE Profiles
                        - Function Key Definitions
                        - Vocabulary Overrides
                        - Processing Options
                        - Help Instructions

Selection or command
====>>-----

                                Cmd/HS:
                                Opt: 4=Sel  9=End    F3=Exit  F24=More Keys

```

Note: *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 JD Edwards World Attention Menu window Functions

The program allows you to:

- Access 15 predefined programs via the Esc (ATTN) Key
- Execute commands, JD Edwards World Hidden Selections, JD Edwards World Fast Path Commands, and JD Edwards World Fast Path Menu Executions

Function Key	Description
F3	Exit the JD Edwards World Attention Menu window
F4	Prompt a command
F6	Submit a job to batch
F8	JD Edwards World Menu Word Search
F9	Retrieve previous command
F13	Display all fast path commands
F18	SIGNOFF all group jobs

Selection Exits	Description
4	Activate a menu selection
9	End a group job(Ignored on a Menu window)

The JD Edwards World Attention Menu window is not accessible while using

- SysReq (Source Machine Only)
- a program that has reset the Esc (ATTN) Key program (for example OFFICE/400)

Work with IBM Pass-Through

About Working with IBM Pass-Through

Note: This section is predicated on Series i network setup enabling use of the IBM command STRPASTHR. The steps for that implementation will not be covered in this manual.

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:

- [Setting Up Access to Remote Locations](#)
- [Using IBM Pass-Through with Group Jobs](#)

Setting Up Access to Remote Locations

To set up access to remote locations

To set up access to remote locations, go to the DREAM Writer versions list for Form ID P98GRP5.

Q	Version	Description	User	Chg Date
—	XJDE0001	Denver A	DEMO	08/23/93
—	XJDE0002	Denver C	DEMO	08/23/93
—	XJDE0003	Denver D	DEMO	08/23/93
—	XJDE0004	Denver E	DEMO	08/23/93
—	XJDE0005	Denver I	DEMO	08/23/93
—	XJDE0006	Atlanta	DEMO	11/13/91
—	XJDE0007	Chicago	DEMO	11/13/91
—	XJDE0008	New York	DEMO	11/13/91
—	XJDE0009	Dallas	DEMO	11/13/91
—	XJDE0010	Houston	DEMO	11/13/91
—	XJDE0011	San Francisco	DEMO	11/13/91
—	XJDE0012	Washington DC	DEMO	11/13/91

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 provide setup parameters for the STRPASTHR command allowing access to a remote location. The following illustrates the processing options.

```

98312                Processing Options Revisions  Form ID. . . . P98GRP5
Denver C              Version. . . . 0002

This job has various options described below. Enter the desired values and
press ENTER to continue.

Destination Virtual Control Unit          V5251

Enter ONE of the following:

1) Destination Location:                  JDEC
   (If APPN routing can be used.)

2) APPC Device(s):
   Communication Device 1:
   Communication Device 2:

   (If S/38's are involved, an APPN
   cannot be used.)

                                Bottom          +

                                F5=Printer Overrides

```

Option	Description
<i>Destination Virtual Control Unit</i>	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.
<i>Destination Location</i> (Used in AS/400 Environment)	This is the APPN network name for the remote location.
<i>APPC Device(s)</i> (Used in S/38 Environment)	These are the APPC devices that identify the route to the remote location. <ul style="list-style-type: none"> Only one intermediate node is supported.

Using IBM Pass-Through with Group Jobs

To use IBM Pass-Through with Group Jobs

1. Use the JD Edwards World menu B98P to start an IBM Pass-Through session to a remote machine.
2. Use the JD Edwards World Menu Revisions (1/G901) to attach your user defined DREAM Writer Form ID P98GRP5 versions to a 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.

```

B98P                               J.D. Edwards & Company           E
                                   JDE Passthru Network

... DENVER
2.  A . . . . . S/38
3.  C . . . . . AS/400
4.  D . . . . . AS/400
5.  E . . . . . AS/400
6.  I . . . . . AS/400

... BRANCH OFFICES
14. Atlanta . . . . . AS/400
15. Chicago . . . . . S/38
16. Dallas . . . . . AS/400
17. Houston . . . . . AS/400
18. New York . . . . . AS/400
19. San Francisco . . . . . AS/400
20. Washington DC. . . . . AS/400

Selection or command
===>> _____
_____

```

The mechanism used to attach remote locations to the JD Edwards World Group Job window on the source machine is a parameter on the STRPASTHR (Start Pass-Through) command. The following illustrates the link to the source machine.

```

                                Start Pass-Through (STRPASTHR)

Type choices, press Enter.

Remote location . . . . . _____ Name, *CNNDEV
APPC device . . . . . *LOC _____ Name, *LOC
+ for more values
Virtual controller . . . . . *NONE _____ Name, *NONE
Virtual display device . . . . . *NONE _____ Name, *NONE
+ for more values
Mode . . . . . *NETATR _____ Name, *NETATR
Local location . . . . . *LOC _____ Name, *LOC, *NETATR
Remote network identifier . . . . . *LOC _____ Name, *LOC, *NETATR, *NONE
System request program . . . . . *SRQMNU _____ Name, *SRQMNU
Library . . . . . _____ Name, *LIBL, *CURLIB

Bottom
F3=Exit F4=Prompt F5=Refresh F10=Additional parameters F12=Cancel
F13=How to use this display F24=More keys

```

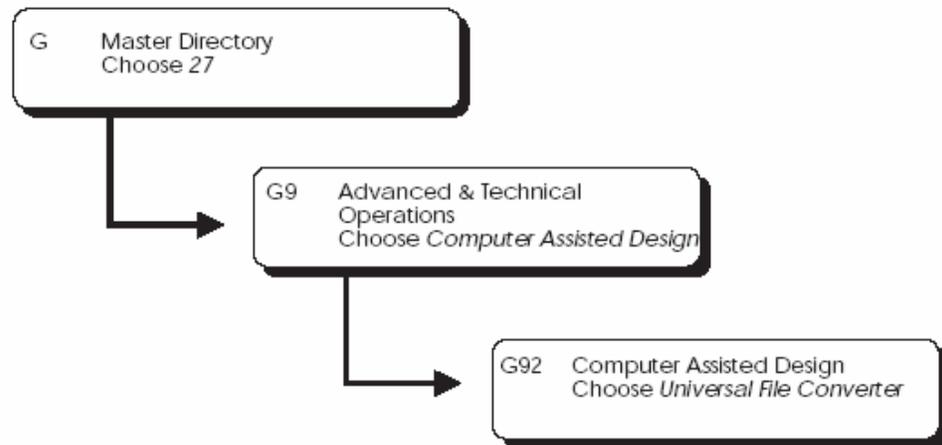
The SRQ10PGM (SysReq 10) parameter allows a program to be called on the source machine from the remote location. By entering the JD Edwards World Group Job window program (P98GRP) in this parameter, the JD Edwards World Group Job window can be displayed on the remote location by pressing SysReq 10, NOT the ATTN Key. This allows access to all suspended group jobs on the source machine and other remote locations.

8 Universal File Converter

Overview to Universal File Converter

About Universal File Converter

JD Edwards World Universal File Converter maps data from one data file resident on the Series i to another file on the same Series i using crossover rules. A DREAM Writer version can be set up so that the conversion can be repeated many times for a standing process or just once for a conversion process.



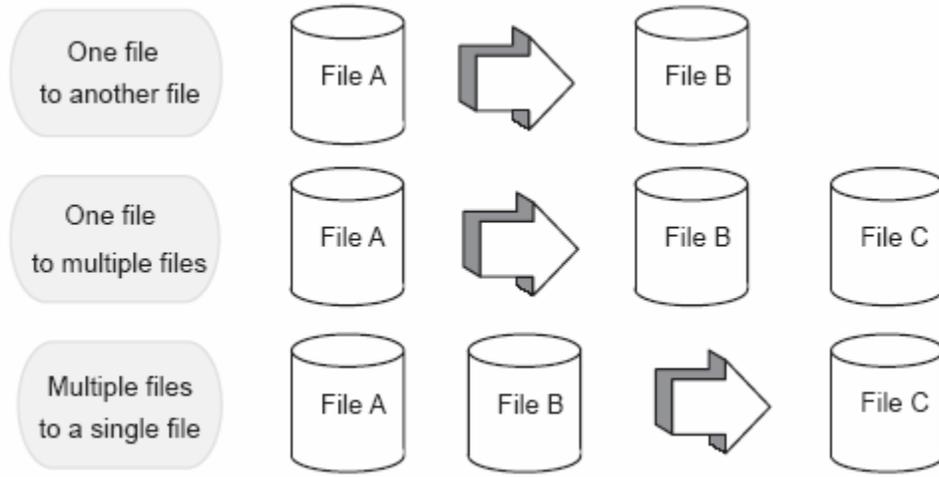
JD Edwards World Universal File Converter uses Crossover rules to map data:

- From one file to another file
- From one file to multiple files
- From multiple files to a single file

Caution: Do not attempt to use UFC 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.

The UFC does a character at a time move of the data from the input file data location into the output file data location. For this reason it is not suited to large, repeated data transfers. It is suited to one time data file conversions or small scale frequent data import situations.

The Cross Over Rules file defines the association between two files and includes data field information.



Step 1

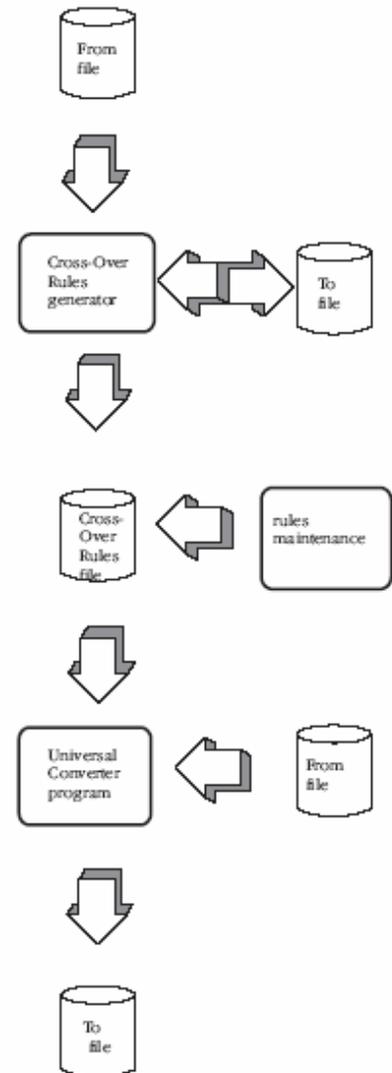
You specify *From* files and *To* files through DREAM Writer processing options. You can specify up to four *To* files. If you require multiple *From* files, specify a join logical as the *From* file in the DREAM Writer “based on” file. The system returns file field information and pre-loads the Cross-Over 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:

Field	Explanation
Dates	The converter uses a keyword to decide what date translation is necessary.
Numeric Fields	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.
Business Unit	The converter processes the field through the Business Unit scrub routine. This routine right adjusts and fills the field with blanks.
Data Dictionary Default	The converter uses the reference field in the To file to access the data dictionary and retrieve the default value for the field.
Initialization	Fields in the To file are initialized to blanks for alphanumerics and zeros for numerics if no fields are defined to map to them.
Next Number	You can specify to have a next number value assigned to a field.
Check Data Dictionary	You can specify to have the value of the field validated against the data dictionary values, ranges, and user defined codes.
User Defined Code Lookup	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.
Default Constant	Specify constant value, up to six characters, for the To field value.

Database Considerations

The system creates records in the Cross-Over Rules file for each version of cross-over rules you specify. This file contains information about the fields in the From file and the To file and how the two fields 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 field 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 Versions
- Work with the Data Dictionary Glossary by File

Set Up Universal File Converter

About Universal File Converter (UFC)

The Universal File Converter is used to import data from one AS/400 physical file to another AS/400 physical file. 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.

A major benefit of using UFC is that once the process has been tested, it can be made into a production process using a Dream Writer version.

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 JD Edwards World and non-JD Edwards World files.

This section contains the following:

- [Understanding the Universal File Converter Setup](#)
- [Setting Up Universal File Converter](#)

Before You Begin

- Before you run the setup procedure make sure the To files exist.

Caution: 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, F0031.

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. **FILLER fields will be ignored during the conversion process. 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.

Considerations

UFC is written to map data character-by-character from any physical file resident on the iSeries (AS/400) to any other resident physical file. If the input file resides on a PC, it must be transferred to the iSeries via a data transfer utility.

If you are using UFC to populate Z1 files, Z1 batch processing is recommended for uploading data to the appropriate production files. Areas where these processes are available include:

Area	Menu
Address Book	G01313
General Accounting	G09311
Accounts Payable	G04311
Accounts Receivable	G03311
Time Card Entry	G07121
Budgeting	G1421
EDI	G47 menus
Batch Sales Order Entry	G4212

To protect your production data, create a test library and put a copy of the “from file” and “to file” with attached logical files in it. There may be a logical file attached to the to file with a unique key requirement. By having this file attached during the UFC process, all the key field requirements will be met. Now gather the following two pieces of information. Type command DSPFP on the from file in the test library. Page down to the last panel. Write down the Format name and the number of records in the file. The number of records should be 100 or less for the initial testing. During testing, the number of records can be used to make sure all records were read. When the conversion has been tested successfully a full-file test can be made. The record format name will be used in the last step.

4. 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.

Note: If either file is set to JD Edwards World file = Y, numeric field names will be looked up for decimal point information. If either file is not created with Data Dictionary data item name, make sure to set the JD Edwards World file = N. Page down to enter the test library name.

5. Return to the Versions List.

6. Execute the version.

After entering the file names and the libraries in the processing options, you do need to submit the DREAM Writer version. P00120 will find the “from” and “to” files and create a record in F0031 for each field in these files. If there is no print file, the F0031 will have version records added for each field in the setup in field name sequence.

98312	Processing Options Revisions	Form ID. . . . P00120
		Version. . . . APCS
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.		<u>F92801</u>
JDE File?		<u>Y</u>
2. Enter the name of the file OR files to convert the data to.		
File 1		<u>F92801U</u>
JDE File?		
File 2		
JDE File?		
File 3		
JDE File?		
File 4		
JDE File?		
F5=Printer Overrides		+

Option	Description
Enter the name of the file	The name of the <i>From</i> file to convert the data from.
JD Edwards World File?	Y if the <i>From</i> file is a JD Edwards World file, or N if it is not. If Y, the first two characters of the field name will be stripped and the remaining field name will be looked up in the JD Edwards Data Dictionary.
Enter the name of the file OR files to convert the data to.	The names of the <i>To</i> files in the spaces provided.
JD Edwards World File?	Y if the <i>To</i> file is a JD Edwards World file, or N if it is not.

```

98312                Processing Options Revisions  Form ID. . . . P00120
                                                Version. . . . APCS
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                DEMO
   "from" file. If left blank the
   library list will be searched for
   the "from" files.
4. Enter the library containing the                DEMOG
   "To" file. If left blank the
   library list will be searched for
   the "To" file.
                                                F5=Printer Overrides

```

Option	Description
Enter the library containing the From file.	The name of the <i>From</i> file library, or leave blank to search your library list.
Enter the library containing the To file.	The name of the <i>To</i> file library, or leave blank to search your library list.

To set up crossover rules

This is where the mapping details are entered. There are many conversion rules in UFC to cover most situations, try them first before using custom Xxxxx programs. You need only map the fields that you are concerned with.

- From menu G9841, take Selection 3 to go to the Crossover Rules.
- Inquire on the version name you created and ran in Step 2.
P00120 is pre-loaded in the Form ID field.

Note: The program in Step 2 above attempts to match field names. Any field with no match will have ****FILLER** for a Field Name. If ****FILLER** is on either the "From" or "To" side, the data will not be mapped. If the "From" side has fewer ****FILLER** fields, press F8 to clear them. If the "To" side has fewer, press F9.

The remaining fields will be in alphabetical order, not by position within the record. UFC is usually used to bring data into a JD Edwards World World file, so that assumption will be used in this exercise.

- Pick a field on the "To" side that you want to map into.
Start with the field name on the "From" side and key in a meaningful field name. Use F14 to enter additional documentation.
- Set the "From" field "type" to A for Alpha.
This is the most inclusive data type and will be used for numeric fields as well.
You will need to know where the data to be mapped starts in the input record. UFC moves data character-by-character, so if the input data has leading spaces, increment the "Begin Pos" field to skip over them. Similarly, on the "To" side, manipulate the "Begin Pos" as necessary so the data will be mapped correctly.

The field names are not important to the process. The type, position and length values are important.

5. On the “To” side, the type can be “A”, “S”, or “P” for Alpha, Signed or Packed. “A” fields will have the “Byte” field set to the length of the data and the “Dig” field should be 0. “S” fields should have the “Bytes” and “Dig” fields set to the same value. “P” fields should have the “Dig” set to $((\text{“Bytes”} \times 2) - 1)$. It takes 8 bytes to hold 15 packed digits. In most cases, the “Dec” field will be 00.
6. Press F4 to open the fold area. Press F1 on the “Conv Rule” field. The most used functions are the date formats, *DFT and *NN. Most file dates are stored in Julian format in JD Edwards World World. The incoming data should be 6 digits long and will probably be in a *MDY or *DMY format. On the “To” side, the field will also be 6 digits long but will have the *JUL format. The *DFT rule has room for 6 contiguous digits of data. *NN uses the first four digits for system code and the next 2 for the bucket number.
7. When all the incoming fields with data have been mapped, type C in the action code and press enter. All fields in the “To” file will be initialized to the proper empty format if not mapped.

See Also

- Work with Cross Over Rules

To create a new version of P00111

1. On menu G9843, take Selection 4 to go to the Versions List for P00111.
2. Make a copy of the XJDE0001 version and give it the same name and title as in Step 2A for continuity purposes.
3. On the Additional Parameters screen:
 - Enter the input file name in the Based on file field.
 - Press Enter.
 - Press F12 to return to the Add'l Parameters screen.
 - Ensure the format name is the one from the DSPFD command in the preparation step.
 - Press Enter.
4. Display the Processing Options.
5. Set the Processing Options as follows:
 - Processing Option 1 needs the version of P00120 from Step 2 so it can find the right crossover rule records in the F0031.
 - Enter the appropriate file and library names for options 2, 3, and 4.
 - Option 5 is usually set to 1 to clear the file so just the records from the latest conversion will be there.
 - Option 6 is usually blank.

Caution: Do not change Data Selection and Data Sequencing. They are used for the OPNQRYP statement. Since we want to use the full file, they don't apply.

6. Return to the Versions List.
7. Submit your version.

To review your results

1. Go to Work with Spooled Files and look for R00111.
At the bottom of this spooled file will be how many records were read and written.
2. Type command DSPPFM on the "to" file to see if the mapping is correct.
3. In order to see all the data, press F10, then F11 to see the hexadecimal representation of the data in an over/under presentation.

Trouble Shooting

- Sometimes P00111 will have problems with the input file.
 - On the Additional Parameters screen, if the file output type is 1, change it to 2 and vice versa.
- Make sure the P00111 based on file and format names are correct.
 - See To create a new version of P00111.
- There may be just one record in the "to file" and a message in the Joblog about a duplicate record.
 - Type command DSPFD on the file and its logicals to see which one has the "unique" key requirement. You will need to map data to each of the key fields. If there is no matching data, maybe a *NN conversion rule will have to be attached to one of the key fields as a tie breaker.

Key Words

Valid conversion rules keywords include:

Function	Key Words
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, *UDCssssr where ssss is the System Code and rr is the Code Type
Default Constant	*DFTcccccc where ccccc is the constant

Function	Key Words
Next Number	*NNssssxx where ssss is the System Code and xx is the Number

Work with Crossover Rules

Working with the Crossover Rules Screen

The Crossover Rules screen 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. You need only map the fields that you are concerned with.

Note: There are many conversion rules in UFC to cover most situations, try them first before using custom Xxxxx programs.

Lines with ****FILLER** in either the from file or the to files are ignored enabling you to view *From* file fields with no corresponding *To* file fields or view *To* file fields with no corresponding *From* file fields.

This section contains the following:

- [Displaying Field Descriptions](#)
- [Adding Fields](#)
- [Deleting Records](#)
- [Keywords](#)
- [About the Conversion Rule Program](#)
- [Available Functions and Options](#)

To work with the Crossover Rules form

1. From the Universal File converter menu, choose Crossover Rules.

```

0031                                Crossover Rules
Action Code . . . . . I
Form Id . . . . . P00120           From File . . . . F4001Z
Version . . . . . 0001
To File Name . . . . F4011Z
Skip to . . . . From . . . . . To . . . . .

  From File . . . . .                To File . . . . .
  Field   T  Begin  ... Field ...   Field   T  Begin  ... Field ...   O
  Name   -  Pos    Bytes Dig Dec   Name   -  Pos    Bytes Dig Dec   -
**FILLER A   1     1     1 00 00   SZACOM  A   988     1 00 00   -
**FILLER A   1     1     1 00 00   SZAEXP  P   593     8 15 02   -
**FILLER A   1     1     1 00 00   SZAID   A  1039     8 00 00   -
**FILLER A   1     1     1 00 00   SZAISL  A  1131     8 00 00   -
**FILLER A   1     1     1 00 00   SZAITM  A   316    25 00 00   -
**FILLER A   1     1     1 00 00   SZANI   A  1010    29 00 00   -
**FILLER A   1     1     1 00 00   SZAOPN  P   601     8 15 02   -
**FILLER A   1     1     1 00 00   SZAPTS  A   755     1 00 00   -
**FILLER A   1     1     1 00 00   SZATXT  A   750     1 00 00   -
**FILLER A   1     1     1 00 00   SZBIN   A  1139     8 00 00   -
**FILLER A   1     1     1 00 00   SZCADC  P   685     4 07 03   -
**FILLER A   1     1     1 00 00   SZCDCD  A   756    15 00 00   -
Opt:  9=Del  F4=Del1 F6=Add  F8=From Fill F9=To  Fill F13=File F14=Text
  
```

The form above displays illustrative data only. This is where the mapping details are entered. The *From* file fields appear on the left. The *To* file fields 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.
 - Use F6 to pull up full information about both fields on the Crossover Rules screen. See *To add a field*.
 - 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 detail area.

Field	Explanation
Form Id	This will always be P00120, the Version Setup program.
From File Name	The file that data is being transferred “from” in the file conversion process.
Version	This is the version name you created and ran in the Setup step.
To File Name	The file that data is being transferred “to” in the file conversion process.
Skip to...	Enter either the FROM field name or the TO field name to skip to in the subfile.
From File	Heading for the fields that data is being transferred “from” in the file conversion process.
To File	Heading for the fields that data is being transferred “to” in the file conversion process.

```

0031                                Crossover Rules
Action Code . . . . . I
Form Id. . . . . P00120      From File . . . . . F92801
Version. . . . . APCS
To File Name . . . . . F92801U
Skip to . . . From . . . . . To. .

      From File . . . . .      To File . . . . .

Field      T Begin  ... Field ...      Field      T Begin  ... Field ...  O
Name      _ Pos   Bytes Dig Dec      Name      _ Pos   Bytes Dig Dec  -
OXXCC    _ A    47    12  00  00      OXXCC    _ A    47    12  00  00  -
Desc Bus Unit Conv Rule
Key Pos  _ Ref XCC      Array N      Key Pos 00 Ref XCC      Array N
OXXDS    _ A     9     30  00  00      OXXDS    _ A     9     30  00  00
Desc Descriptio Conv Rule
Key Pos  _ Ref XDS      Array N      Key Pos 00 Ref XDS      Array N
OXXDT    _ S    41     6  06  00      OXXDT    _ S    41     6  06  00  -
Desc Date Last Conv Rule
Key Pos  _ Ref XDT      Array N      Key Pos 00 Ref XDT      Array N
OXXIT    _ S     1     8  08  00      OXXIT    _ S     1     8  08  00
Desc Item ID. . Conv Rule
Key Pos  _ Ref XIT      Array N      Key Pos 00 Ref XIT      Array N
OPT: 9=Del F4=Detail F6=Add F8=From Fill F9=To Fill F13=File F14=Text
    
```

Field	Explanation
<i>From File</i> Field name	The name of the field in the FROM file for source data that is used in the conversion process.
<i>From File</i> Field Data Type	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'. (F1) Usually A for fields in a PC (ASCII) file.
<i>From File</i> Field Begin Pos	The beginning position of the field in the “from” file.
<i>From File</i> Field Size In Bytes	The number of bytes for the field in the “from” file.
<i>From File</i> Field Number of Digits	The actual number of digits in the “from” file field. In a non-packed field this is the same as the number of bytes. (Numeric fields)
<i>From File</i> Field Decimal Positions	The number of decimal positions in the “from” file field. (Numeric fields)
<i>From File</i> Field Description	The description of the “from” file field. If the file is declared a JD Edwards World file, this will be the first part of the Data Dictionary description.
REF (Reference)	Use reference field for those conversion rule entries that refer to the Data Dictionary.

```

Field      T Begin  ... Field ...      Field      T Begin  ... Field ...  O
Name      _ Pos   Bytes Dig Dec      Name      _ Pos   Bytes Dig Dec  -
**FILLER  _ A     1     1  00  00      SZACOM    _ A    988     1  00  00  -
Desc FILLER Conv Rule
Key Pos  _ Ref FILLER  Array N      Desc Apply Comm Conv Rule
Key Pos 00 Ref ACOM      Array N
    
```

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 <i>To</i> 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.
“Skip to” capabilities	Two “skip to” capabilities are available on this form. You can skip to a field in either the <i>From</i> file or the <i>To</i> file.
*DFT	When using *DFT if the literal is over 6 positions then part it out as if dealing with two fields.
Packed numeric	<p>Packed numeric values use ½ their length plus one in a file. A numeric field defined as a length of 15 only uses 8 bytes.</p> <p>Negative, or signed fields, must have the ‘sign’ character in the FROM file.</p> <p>In the Crossover Rules, when using a User Defined Code Lookup keyword (*UDCsssrr) and the system code is only 2 numbers, such as 55, enter it as __55 (with leading blanks), not 0055 or 5500.</p>
Negative, or signed fields	Negative, or signed fields, must have the ‘sign’ character in the FROM file.
Using lookup keyword *UDCsssrr	In the Crossover Rules, when using a User Defined Code Lookup keyword (*UDCsssrr) and the system code is only 2 numbers, such as 55, enter it as __55 (with leading blanks), not 0055 or 5500.

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 displays, as shown below.

98FFD		File Field Descriptions		S/FMT
File and Libr:	F4011Z	PGPPTA71	PF	
- I4011Z	- Batch Receiver File - Order De			
- SZEDTY	Record Type	A	1	1
- SZEDSQ	Record Sequence . . .	P	2 0	2
- SZEKCO	Document Key Company	A	5	4
- SZEDOC	Document Number . . .	S	9 0	9
- SZEDCT	Document Type	A	2	18
- SZEDLN	Line Number	P	7 0	20
- SZEDST	Transaction Set . . .	A	6	24
- SZEDFT	Translation Format . .	A	10	30
- SZEDDT	EDI - Transmission D S		6 0	40
Opt: 2=Dictionary 4=Sel F15=Resequence F3=Return				

- 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 *Computer 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 *Technical Foundations Manual*.
2. Enter 4 in the option field. The program returns the field description to the associated field as shown in this example.

0031		Crossover Rules						
Action Code	I							
Form Id.	P00120	From File	F4001Z					
Version	0001							
To File Name	F4011Z							
Skip to	From	To						
From File		To File						
Field Name	T	Begin Pos	... Field Bytes Dig Dec	Field Name	T	Begin Pos	... Field Bytes Dig Dec	O
**FILLER	A	1	1 00 00	SZACOM	A	00004	00001 00 00	-
**FILLER	A	1	1 00 00	SZAXEP	P	593	8 15 02	-
**FILLER	A	1	1 00 00	SZAIID	A	1039	8 00 00	-
**FILLER	A	1	1 00 00	SZAISL	A	1131	8 00 00	-
**FILLER	A	1	1 00 00	SZAITM	A	316	25 00 00	-
**FILLER	A	1	1 00 00	SZANI	A	1010	29 00 00	-
**FILLER	A	1	1 00 00	SZAPNP	P	601	8 15 02	-
**FILLER	A	1	1 00 00	SZAPTS	A	755	1 00 00	-
**FILLER	A	1	1 00 00	SZATYT	A	750	1 00 00	-
**FILLER	A	1	1 00 00	SZBIN	A	1139	8 00 00	-
**FILLER	A	1	1 00 00	SZCADC	P	685	4 07 03	-
**FILLER	A	1	1 00 00	SZCADC	A	756	15 00 00	-
Opt: 9=Del F4=Del1 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

```

This window presents most of the fields on the Crossover Rules screen.

2. With the cursor in the Field Name field on the Crossover Rules screen, press F13 to open the File Field Descriptions form.
3. Select a file using the 4 option.
4. Press Enter.

The program returns the 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 skipped in the conversion process.

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.

Caution: With the exception of the date keywords listed below, specify conversion rules for either *From* field or *To* field, never for both.

Keyword	Description
Dates - *MDY, *DMY, *YMD, *JUL, *SYSVAL	These keywords activate a date conversion between the <i>From</i> file field and the <i>To</i> file field. You must type keywords into both the <i>From</i> file Conversion Rule field and the <i>To</i> file Conversion Rule field. Each keyword on the <i>From</i> field specifies how the field is stored in the <i>From</i> file. The keyword on the <i>To</i> field conveys the output format on the <i>To</i> 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 <i>From</i> file field before moving it to the <i>To</i> field.
Initialize - *ZEROES, *BLANKS	These keywords move either zeroes or blanks to the <i>From</i> 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 <i>To</i> file field, using the Reference field in the Data Dictionary, and loads it to the <i>From</i> file field before it is transferred.
User Defined Code - *UDC $ssrr$	This keyword retrieves the definition of the user defined code used in a specific system and loads it to the <i>To</i> 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 <i>To</i> field. When typing your request, $cccccc$ is the default constant.
Next Number - *NN $ssssxx$.	This keyword computes a next number and loads it to the <i>To</i> field. When typing your request, $ssss$ is the system and xx is the bucket 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:

PARM	Description
First parameter	Must be 50 bytes and contains the value of the field being processed. Use it to send the value to the converter program and receive the value 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 <i>To</i> file. <ul style="list-style-type: none"> ▪ If the error flag returns with 2, the data in parameter 1 is not transferred to the <i>To</i> file. Use this error if you are writing multiple <i>To</i> file records and different <i>From</i> file fields are used for a single field in the <i>To</i> file. ▪ If the error flag returns with 3, a <i>record</i> will not be written to the <i>To</i> 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 <i>To</i> 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 <i>To</i> file records.
Fourth parameter	Ten-byte field for the <i>From</i> file field name. This lets your “X” program know which field you are processing if multiple fields in the <i>From</i> file are updating a single <i>To</i> file field.
Fifth parameter	Ten-byte field for the <i>To</i> file field name. This lets your “X” program know which field you are processing if multiple fields in the <i>From</i> file are updating a single <i>To</i> file field.

Available Functions and Options

Function	Description
F6	<p>Add Instructions</p> <p>To add fields to be converted, press F6 to access the Add Cross Over Instructions Form. The required fields for adding a field are <i>Field Name</i>, <i>Field Data Type</i>, <i>Field Beginning Position</i>, and <i>Number of Bytes</i>.</p>

```

0031                                Crossover Rules
Action Code. . . . .
Form Id. . . . . 00312             Add Cross Over Instructions
Version. . . . .
To File Name . . . . .
Skip to . . . From
      From File . . .
Field      T Begi
Name      Pos
Field Name . . . . .
Field Data Type. . . . .
Field Begin Pos. . . . .
Number of Bytes. . . . .
Number of Digits . . . . .
Field Dec Pos. . . . .
Field Description
Conversion Rule . . . . .
      To File . . . . .
Field Name . . . . .
Field Data Type. . . . .
Field Begin Pos. . . . .
Number of Bytes. . . . .
Number of Digits . . . . .
Field Dec Pos. . . . .
Field Description
Conversion Rule . . . . .
eld ... 0
Dig Dec -
OPT:          F3=Exit

```

Function	Description
F8	<p><i>Suppress From</i> **FILLER Fields</p> <p>Will not display those lines with **FILLER values in the <i>From</i> field.</p>
F9	<p><i>Suppress To</i> **FILLER Fields</p> <p>Will not display those lines with **FILLER values in the <i>To</i> field.</p>
F13	<p>File Field Description</p> <p>Place cursor on any <i>Field Name</i> field and press F13 to display the File Field Description form.</p>

Function	Description
F14	<p>User Defined Text</p> <p>Highlight to indicate that there is generic text associated with this field.</p> <ul style="list-style-type: none">▪ Press F14 in the top area of the form to enter text about the conversion.▪ Press F14 in the <i>From Field</i> area (left side of the form) to enter text describing the <i>From Field</i>.▪ Press F14 in the <i>To Field</i> area (right side of the form) to obtain text describing the <i>To Field</i>. <p>The field will highlight to indicate that there is generic text associated with this field.</p>

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.

```
G9841                J.D. Edwards & Company
                   Universal File Converter

... DATA FILE CONVERSION
 2. Version Setup
 3. Crossover Rules
 4. File Conversion
 5. Report

Selection or command
====> _____
_____
```

Note: When creating an execution form, be sure the Based on File and the Format Name fields on the Additional Parameters screen contain your Based-on filename and the correct Format name for that file. The Data selection and sequence records should be left as-is since the converter reads the entire from file.

2. Add your own version from a Demo version and go to the processing options of your new version.

```

98312                Processing Options Revisions  Form ID. . . . P00111
Execute File Conversion - Sample                 Version. . . . APCS

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
    
```

Option	Description
Enter the name of the Form ID and version containing the Initial Setup step.	The Form ID will be P00120. Enter your version from the conversion specifications.
Enter the name and library of "from" file, if different than the Form ID and version specified.	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. This file must have the same organization as the file used in the crossover rules.

```

98312                Processing Options Revisions  Form ID. . . . P00111
Execute File Conversion - Sample                 Version. . . . APCS

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
    
```

Option	Description
Enter the name of the file OR files to convert the data to.	Type the names of the <i>To</i> files. Up to four files can be specified. If these fields are left blank, all files entered in the setup version are converted. The files must have the same organization as the file used in the crossover rules.
Enter the library the to files are in.	Type the name of the library containing the <i>To</i> files, or leave blank to have the library list searched.

```

98312                Processing Options Revisions  Form ID. . . . P00111
Execute File Conversion - Sample                    Version. . . . APCS

This job has various options described below. Enter the desired values and
press ENTER to continue.

File Preparation:
5. Enter a '1' to clear the file data               _____ -
   is being transferred to.

TO FILE FORMAT:
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

```

Option	Description
Enter a "1" to clear the file data is being transferred to.	Enter "1" to clear the <i>To</i> file. The <i>To</i> file will be filled only with converted records. If this field is left blank, the converted data records are added to the <i>To</i> file.
Enter the number of to file records to be created for each from file record.	Enter the number of <i>To</i> file records you want to create for each <i>From</i> file record. If this field is left blank, only a single <i>To</i> file record will be created for each <i>From</i> file record.

3. Enter the correct values on Processing Options and submit your version to complete the conversion process.

What You Should Know About

Multiple "From" files	If you are using multiple <i>From</i> files, remember to create a join logical over all the <i>From</i> files you want to use.
New versions	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 Ixxxx and may not be appropriate.
"From" file name and the "To" file formats	The <i>From</i> file name and the <i>To</i> file formats should be the same as used to set up the conversion rules in Step 1.
Error conditions	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.

Trouble Shooting

Problem	Explanation / Resolution
Incorrect Value in the Format Name field.	<p>The most common UFC error is an incorrect value in the Format Name field on the Additional Parameters screen for P00111. The Format Name is for the Based On File (the from file).</p> <p>To verify the format name do a DSPFD on the from file. The file format is listed at the bottom of the display.</p> <p>Always exit the version and re-inquire on the Additional Parameters to verify that the Format Name change was accepted.</p>
Not writing to the output file	<p>Problems writing to output file (in P00111) may be because of duplicate keys.</p> <ul style="list-style-type: none"> ▪ Check if the to file has a logical attached that has a unique key and verify the user is not populating that field with blanks. ▪ The File Output Type field on the Additional Parameters screen for P00111 DREAM Writer should be a 2 (logical). Use a file output type of a 1 (OPNQRYP) for files that contain DDS (JD Edwards World files). If one type does not work, try the other. ▪ Data Definition Specifications (DDS) means the file is formatted (the lengths of the fields in the record are defined). Files used in UFC do not have to be formatted (contain DDS) but they must be orderly (all fields in each record must be in the same place.) Use DSPPFM to see the data in each record. Use F10, F11 to see the hexadecimal value and the ASCII value. ▪ If all setup options look correct check for multiple F0031 files – you may be executing out of the wrong Crossover Rules file. The data in this file is created when running the first DREAM Writer, P00120 (2/G9841).
Job log “version for P00111 cannot be found”	<p>If you receive a job log after running P00111 that states the version for P00111 cannot be found, check that only one set of DREAM Writer files exist in your library list. Otherwise unpredictable results may occur.</p>

3. Select one of the following print options:

Field	Explanation
Enter a "1" to print Data Dictionary Glossary for each item.	Prints Data Dictionary Glossary for each <i>To</i> field.
Enter "1" to print File Specific Glossary for each item.	Prints file specific glossary from Generic Text file (F00163) for each <i>To</i> file.
Enter "1" to print generic text instructions for each item	Prints any generic text associated with either <i>To</i> fields or <i>From</i> fields.

Create Conversion Versions

Creating Conversion Versions

Universal File Converter enables you to create conversion forms to use for planning purposes when you convert your non-JD Edwards World files into JD Edwards World 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.
- JD Edwards World supplies a special Data Dictionary glossary relating to specific fields in specific files in your JD Edwards World 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 Universal File Converter (G9841), choose either Versions Setup or Report.

```
98312                Processing Options Revisions  Form ID. . . . P00120
Generate Cross Over Instruction - Sample          Version. . . . XJDE0001
                                                  Display Level. 9

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?                                   _____
2. Enter the name of the file OR files
   to convert the data to.                      _____
   File 1                                       F92801U
   JDE File?                                   Y
   File 2                                       _____
   JDE File?                                   _____
   File 3                                       _____
   JDE File?                                   _____
   File 4                                       _____
   JDE File?                                   _____
                                                  +

F5=Printer Overrides
```

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 JD Edwards World files or N if they are not.

98312	Processing Options Revisions	Form ID. . . .	<u>P0031P1</u>
		Version. . . .	<u>XJDE0001</u>
		Display Level.	<u>2</u>
File Converter Report			
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)	<u>1</u>	_____
2)	Enter a "1" to print File Specific Glossary for each data item. Leave blank to not print. (Prints for "TO" fields only)	<u>1</u>	_____
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)	<u>1</u>	_____
		Bottom	+
F5=Printer Overrides			

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, JD Edwards World 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:

- [Accessing the Data Dictionary Glossary by File](#)
- [Adding a File Specific Glossary Item](#)
- [Printing the Data Dictionary Glossary Information](#)

Accessing the Data Dictionary Glossary by File

To access the Data Dictionary Glossary by file

1. From Universal File Converter (G9841), type DD on the command line and press Enter.

The Data Dictionary Repository screen appears.

```
9201 Data Dictionary Rls Last Chg A61
Action Code . . . . I Item Parent.
Data Item . . . . .MCU
Glossary Group . . D
-----
Alpha Desc . . . . Business Unit
Reporting System . 09
System Code . . . . 09 Type . A Size . 12 Data File Decimals ___
Data Item Class. . COSTCTRSEC Item Occurrences ___ Display Decimals . _
-----
Row Description. . Business Unit
Column Title . . . Business
Unit
-----
Default Value. . . . .
Data Display Rules *RAB Justify. _
Data Edit Rules. . SERVER X0006
-----
Search Program . . . . .
Next Nbr System. . . . . Next Number Index . . . . .
-----
F4=Search F8=UDC F9=Prev F10=Glossary F11=Descriptions F15=Where Used
```

2. Press F10 to display the glossary definition of the data item you selected.

92001	Data Item Glossary Revisions	Language	_____
		Applic Override	_____
		Scrn/Rpt	F4102
Action Code	I		
Data Item	MCU	Desc Business Unit	
System Code	09	Reporting System Code . 09	
Glossary Group	D		
<p>Identifies a separate entity within a business for which you wish to track costs, for example, a warehouse location, job, project, work center, or branch/plant. The business unit field is alphanumeric.</p> <p>You can assign a business unit to a voucher, invoice, fixed asset, and so forth, for responsibility reporting. The system provides reports of open A/P and A/R by business units, for example, to track equipment by responsible department.</p> <p>Business Unit security can prevent you from inquiring on business units for which you have no authority.</p> <p>In the Inventory Management System, MCU represents a branch or plant</p>			
<p>F4=Search F9=Redisplay Prev F19/F20=Prev/Next Item F24=More</p>			

3. 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, as needed.

Adding a File Specific Glossary Item

To add a File Specific Glossary item

From the Data Item Glossary Revisions form

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.

98312	Processing Options Revisions	Form ID.	P0031P1
		Version.	XJDE0001
		Display Level.	2
File Converter Report			
<p>This job has various options described below. Enter the desired values and press ENTER to continue.</p>			
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)		1	_____
2) Enter a "1" to print File Specific Glossary for each data item. Leave blank to not print. (Prints for "TO" fields only)		1	_____
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)		1	_____
	Bottom		+
F5=Printer Overrides			

Printing the Data Dictionary Glossary Information

To print the Data Dictionary Glossary information

1. From Universal File Converter (G9841), choose Report.
2. Complete the Processing Options Revisions form.

3. Type 1 next to all three options to print the Data Dictionary glossary.
4. Use Option 2 to print the File-Specific glossary text.

9 Appendices

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.

Common Library Files Automatically Created by JD Edwards World Build Programs

The following chart lists files automatically generated from the Data Dictionary as a result of a build program that JD Edwards World offers from a menu. It is recommended that these files be maintained in your common library along with the Data Dictionary.

File Name	File Description	System Code
F98FRF@	Field Reference - "@" Data Items	98
F98FRF\$	Field Reference - "\$" Data Items	98
F98FRFA thru	Field Reference - "A" Data Items through	98
F98FRFZ	Field Reference - "Z" Data Items	98

Physical and Logical Files Created in a Common Library

On the SVR screen, there are two fields which govern the location and content of data files in user libraries during an install.

- The first field is Common File. If a file has this field set to "Y", the file will be created in the user's Common library, if one is specified.
- The second field is Copy Data (Y/N). Most files to be located in the Common library will have Copy Data Y but there are some exceptions such as F0016. Most non-Common (User data) files will be Copy Data N but there are exceptions such as F0010 and F0009. No data will be copied for a logical file.
- Inquire on a file in SVR to see if it should be located in Common or not.
- Inquire on a file in SVR to see if data from the pristine environment should be copied into it.

Appendix B – Upgrading Customized Source Code

JD Edwards World provides access to several complementary products. If you have customized JD Edwards World source code, the following products will help you upgrade your source code.

This section contains the following:

- [S/Compare](#)
- [Harmonizer](#)
- [About Harmonizer Plus](#)

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 before and after 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 JD Edwards World original program, the JD Edwards World 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 JD Edwards World original source file, the JD Edwards World 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 Function

The Project Manager function 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. . . N

O Source Object Source SAR Version S D User Date
P Library Library File Number ID C P ID Modified
_ JDFSRC73 JDFOBJ73 JDESRC 981283 A73 1 BECK 07/07/95

Opt: 1=Browse 2=Edit 3=Copy 5=SAR 8=Print 9=Dlt 10=Design 14=Crt
```

J98MODEL2 - 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. . . .
Maint/RSTDSP . . . Omit Option. . . O Generation Sev . .
Copy Data (Y/N). N Optional File. . N Common File. . . N

O Source Object Source SAR Version S D User Date
P Library Library File Number ID C P ID Modified
_ JDFSRC73 JDFOBJ73 JDESRC 867923 A73 1 BECK 07/07/95

Opt: 1=Browse 2=Edit 3=Copy 5=SAR 8=Print 9=Dlt 10=Design 14=Crt
```

J98MODEL3 - 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. . . .
Maint/RSTDSP. . . . Omit Option. . . . Q Generation Sev. . .
Copy Data (Y/N). N Optional File. . . N Common File. . . N

O Source Object Source SAR Version S D User Date
P Library Library File Number ID C P ID Modified
_ JDFSRC73 JDFOBJ73 JDESRC 867923 A73 1 _ BECK 07/07/95

Opt: 1=Browse 2=Edit 3=Copy 5=SAR 8=Print 9=Dlt 10=Design 14=Crt
    
```

J98MODEL4 - Interactive/Batch with Processing Options

```

9801                               Software Versions Repository
Action Code. . . . I
Member ID. . . . J98MODEL4
Description. . . . Model CL Program - Interactive/Batch with Processing Options
Function Code. . . . CLP CL Programs
Function Use. . . . 198 Model Source Member
System Code. . . . 98 Technical Tools
Reporting System 98 Technical Tools
Base Member Name J98MODEL4 File Prefix. . . .
Maint/RSTDSP. . . . Omit Option. . . . Q Generation Sev. . .
Copy Data (Y/N). N Optional File. . . N Common File. . . N

O Source Object Source SAR Version S D User Date
P Library Library File Number ID C P ID Modified
_ JDFSRC73 JDFOBJ73 JDESRC 867923 A73 1 _ BECK 07/07/95

Opt: 1=Browse 2=Edit 3=Copy 5=SAR 8=Print 9=Dlt 10=Design 14=Crt
    
```

J98MODEL5 - Batch Report Writer - No DDS File

```

9801                               Software Versions Repository
Action Code. . . . I
Member ID. . . . J98MODEL5
Description. . . . Model CL Program - Batch Report Writer - No DDS 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 J98MODEL5 File Prefix. . . .
Maint/RSTDSP. . . . Omit Option. . . . Q Generation Sev. . .
Copy Data (Y/N). N Optional File. . . N Common File. . . N

O Source Object Source SAR Version S D User Date
P Library Library File Number ID C P ID Modified
_ JDFSRC73 JDFOBJ73 JDESRC 867923 A73 1 _ BECK 07/07/95

Opt: 1=Browse 2=Edit 3=Copy 5=SAR 8=Print 9=Dlt 10=Design 14=Crt
    
```

J98MODEL6 - Batch Report Writer OPNQRYF

```

9801                Software Versions Repository
Action Code. . . . I
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           File Prefix. . . .
Maint/RSTDSP . . . Omit Option. . . O Generation Sev . .
Copy Data (Y/N). N   Optional File. . N Common File. . . N

O Source   Object   Source   SAR   Version   S D   User   Date
P Library  Library  File     Number ID     C P   ID     Modified
-- JDFSRC73 JDFOBJ73 JDESRC   867923 A73     1 -   BECK   07/07/95

Opt: 1=Browse 2=Edit 3=Copy 5=SAR 8=Print 9=Dlt 10=Design 14=Crt

```

J98MODEL7 - Batch Report Writer OPNQRYF w/OQF Reset

```

9801                Software Versions Repository
Action Code. . . . I
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           File Prefix. . . .
Maint/RSTDSP . . . Omit Option. . . O Generation Sev . .
Copy Data (Y/N). N   Optional File. . N Common File. . . N

O Source   Object   Source   SAR   Version   S D   User   Date
P Library  Library  File     Number ID     C P   ID     Modified
-- JDFSRC73 JDFOBJ73 JDESRC   867923 A73     1 -   BECK   07/07/95

Opt: 1=Browse 2=Edit 3=Copy 5=SAR 8=Print 9=Dlt 10=Design 14=Crt

```

J98MODEL8 - Control File Driven Batch Process

```

9801                Software Versions Repository
Action Code. . . . I
Member ID. . . . . J98MODEL8
Description. . . . Model CL Program - Control File Driven Batch Process
Function Code. . . CLP   CL Programs
Function Use . . . 198   Model Source Member
System Code. . . . 98   Technical Tools
Reporting System  98   Technical Tools
Base Member Name  J98MODEL8           File Prefix. . . .
Maint/RSTDSP . . . Omit Option. . . O Generation Sev . .
Copy Data (Y/N). N   Optional File. . N Common File. . . N

O Source   Object   Source   SAR   Version   S D   User   Date
P Library  Library  File     Number ID     C P   ID     Modified
-- JDFSRC73 JDFOBJ73 JDESRC   867923 A73     1 -   BECK   07/07/95

Opt: 1=Browse 2=Edit 3=Copy 5=SAR 8=Print 9=Dlt 10=Design 14=Crt

```

Appendix D – Universal File Converter

About Universal File Converter

This is a simple, single file to single file walk-through of the JD Edwards World World Universal File Converter (UFC). A major benefit of using UFC is that once the process has been tested, it can be made into a production process using a Dream Writer version.

The UFC is written to map data character-by-character from any physical file resident on the System I (iSeries / AS400) to any other resident physical file. If the input file is from a PC spreadsheet, that file will have to be ported (FTP or other process) to the iSeries.

Caution: To protect your production data, create a test library and put a copy of the “from file” and the “to file” with its attached logicals into it. There may be a logical file attached to the “to file” with a “unique” key requirement. By having this file attached during the UFC process, all the key field requirements will be fulfilled.

Gathering Information

There are 2 pieces of information to gather:

- The format name
- The number of records in the file

The record format name will be used in the last step. The number of records should be 100 or less for the initial testing. During testing, the number of records can be used to make sure all records were read. When the conversion has been tested successfully a full-file test can be made.

To gather information

1. Do a DSPFD on the “from file” in your test library.
2. Page down to the last panel.
3. Write down the Format name and the number of records in the file.

To identify from and to files

1. You will need to be signed on to JD Edwards World World. You must have an F0031 file in this environment. Go to menu G9841.

2. Selection 2 will take you to a versions list for P00120.
 - Make a copy of the XJDE0001 version. Use your version name and title to tie it to the data you are converting, for example ABCONV and “Address book conversion.”
 - Do NOT make any changes on the Additional Parameters screen. The based on file should be *NONE and the Format name should be INONE.
 - On the processing Options screen, enter the “from” and “to” file names. If either file is set to JD Edwards World file = “Y”, numeric field names will be looked up for decimal point information. Page down to enter the test library name.

Processing options three and four pertain to the library for test data.
 - Upon returning to the Versions List, run your version. P00120 will find the “from” and “to” files and create a record in the F0031 for each field in these files. If P00120 has no print files after it runs, it was successful.

To map data

Selection 3 takes you into the crossover rules. This is where the mapping details are entered.

- P00120 is pre-loaded. Inquire on the version name you created and ran in Step 2.
- The program in Step 2 above attempts to match field names. Any field with no match will have **FILLER for a Field Name. If **FILLER is on either the “From” or “To” side, the data will not be mapped. If the “From” side has fewer **FILLER fields, do an F8 to remove them. If the “To” side has fewer, do an F9.
- The remaining fields will be in alphabetical order, not by position within the record. UFC is usually used to bring data into a JD Edwards World file, so that assumption will be used in this exercise.
- Pick a field on the “To” side that you want to map into. Start with the field name on the “From” side and key in a meaningful field name. Use F14 to enter additional documentation.
- The system highlights Fields that have additional documentation.
- Set the “From” field “type” to A for Alpha. This is the most inclusive data type and will be used for numeric fields as well.
- You will need to know where the data to be mapped starts in the input record. UFC moves data character-by-character, so if the input data has leading spaces, increment the “Begin Pos” field to skip over them. Similarly, on the “To” side, manipulate the “Begin Pos” as necessary so the data will be mapped correctly. The field names are not important to the process. The type, position and length values are important.
- On the “To” side, the type can be “A”, “S”, or “P” for Alpha, Signed or Packed. “A” fields will have the “Byte” field set to the length of the data and the “Dig” field should be 0. “S” fields should have the “Bytes” and “Dig” fields set to the same value. “P” fields should have the “Dig” set to ((“Bytes” x 2) – 1). It takes 8 bytes to hold 15 packed digits. In most cases, the “Dec” field will be 00.

- Open the fold with an F4. Do F1 on the “Conv Rule” field. The most used functions are the date formats, *DFT and *NN. Most file dates are stored in Julian format in JD Edwards World. The incoming data should be 6 digits long and will probably be in a *MDY or *DMY format. On the “To” side, the field will also be 6 digits long but will have the *JUL format. The *DFT rule has room for 6 contiguous digits of data. *NN uses the first four digits for system code and the next 2 for the bucket number.
- When all the incoming fields with data have been mapped, put “C” in the action code and press enter. All fields in the “To” file will be initialized to the proper empty format if not mapped.
- At the top of the screen, you can inquire with the name of an additional file that you’re mapping data to. Go through the above steps to map the data.
- F3 back to the menu.

To convert data

1. Selection 4 takes you to the versions list for P00111.
 - Copy version XJDE0001 and give it the same name and title as in Step 2a for continuity purposes.
 - On the Additional Parameters screen:
 - Enter the input file name in the Based on file field.
 - Press Enter.
 - Press F12 to return to the Additional Parameters screen and make sure the format name is the one from the DSPFD command in the preparation step.
 - Press Enter.
 - Processing option 1 needs the version of P00120 from step 2 so it can find the right crossover rule records in the F0031.
 - For options 2, 3, and 4, enter the appropriate file and library names.
 - Option 5 is usually set to 1 to clear the file so just the records from the latest conversion will be there.
 - Option 6 is usually blank.
 - Leave the data selection and sequencing screens alone. They are used for the OPNQRYF statement. Since we want to use the full file, they don’t apply.
2. From the versions list, run your version.
 - There should be a R00111 in the spool files. At the bottom it will tell you how many records were read and written.
 - Do a DSPPFM on the “to” file to see if the mapping is correct. In order to see all the data, you may want to do F10 followed by F11 to see the hexadecimal representation of the data in an over/under presentation. There will be an “F” or “D” sign in the bottom row at the right hand end of a packed field. Read that data by reading backwards from the sign in an up and down manner.

Trouble Shooting

Questions to ask yourself

Did you:

1. Set up and RUN step 2 on the menu 9841(P00120).
 - You should not change anything on the DW additional parameters.
 - You must enter the file and library names in the Proc Opt.
2. Map only the fields they want in the FROM and TO files on the crossover rules screen. **FILLER fields will be ignored and the fields will be created empty, either blanks or zeros.
3. Enter the correct format name for the based-on file(FROM) in the Additional parameters screen of the P00111 DW.
 - Check this with DSPFD of based-on file.
4. Try changing the file output type for P00111 from 1 to 2 or from 2 back to 1.

Note: Sometimes LF works, sometimes OPNQYF.

5. Check the output file with a DSPPFM to see if there are any records created.

Factors to consider

1. What JD Edward World release level are you on?
2. What Step is having problems?
 - If Step 1: Check the Additional Parameters for the Based On File to be *NONE and format INONE.
 - If Step 2: Check if the you ran the Dream Writer in Step 1 to ensure you are setting the action code to C when changing cross over rules.
 - If Step 3: Are you getting any records in the To file?
 - Yes: Check the cross over rules.
 - No: Use DSPFD on the from file to identify the format ID and verify the correct format is in the additional parameters of the conversion Dream Writer (P00111).

Note: If you are on release A7.1 or above and using *DFT in the cross over rules, you needs a paper fix.

- After making any changes to steps 2 or 3, rerun the conversion.

Things to remember

1. The all time leader in UFC errors is an incorrect format name for the Based on (input) file on the Additional Parameters screen for P00111. Do a DSPFD on the From file and go to the bottom of the display to determine the format. Option 3 from G9841.
2. Do not change anything in the additional parameters screen on the P00120 form. After entering the file names and libraries in the Processing options, you need to run the Dream Writer version. Option 1 from G9841.
3. Problems writing to output file (in P00111) may be because of duplicate keys. Check if the To file has a logical attached that has a unique key and verify the user is not populating that field with blanks.
4. File output type in P00111 Dream Writer additional parameters should be a 2 for logical: for some JD Edwards World file to JD Edwards World file conversions (DDS files), a 1 for OPNQRYF is necessary.
5. There are many conversion rules in UFC to cover most situations, try them first before using custom Xxxxx programs.
6. Use F6 to pull up full information about both fields on the Cross Over Rules screen.
7. Use reference field in the fold area of cross over for those conversion rule entries that refer to the Data Dictionary.
8. You need only map the fields that you are concerned with. If some fields are not being initialized correctly to *ZEROES or *BLANKS, you might have to find a one byte field of the right persuasion and move it into that field (longer length will be accommodated) or use *DFT.
9. The UFC does a character at a time move of the data from the input file data location into the output file data location. For this reason it is not suited to large, repeated data transfers. It is suited to one time data file conversions or small scale frequent data import situations.
10. When using *DFT if the literal is over 6 positions then part it out as if dealing with two fields.
11. When having problems with going from zoned or signed to a packed field, call it alpha to packed forcing UFC to go through the C0012 to edit the data.
12. When having problems with going from packed to alpha, try packed to signed.
13. When having problems with alpha to signed, try alpha to alpha.
14. *If all looks correct, check for multiple F0031 files, you may be executing out of the wrong cross over rules.
15. If you get a duplicate key message in the joblog, it is because there is a unique key on the physical or logical. You may not be populating one of the fields in the key and this may create the problem.
16. Set most of the FROM fields to type A for alpha.
 - Numeric TO fields can be P for packed or S for signed.
17. Sometimes P00111 will have problems with the input file.

- On the Add'l parameters screen, if the file output type is 1, change it to 2 and vice versa.
- 18.** Make sure the P00111 based-on file and format names are correct. See Step 4(b).
- 19.** There may be just one record in the “to file” and a message in the Joblog about a duplicate record.
- Do a DSPFD on the file and its logicals to see which one has the “unique” key requirement. You will need to map data to each of the key fields. If there is no matching data, maybe a *NN conversion rule will have to be attached to one of the key fields as a tie breaker.

Appendix E – Dates

About Date Programs

JD Edwards World uses four key RPG programs to work with dates in the system.

- X0027 is used to calculate a new date from a reference date.
- X0028 is the primary date formatting program which converts dates from Julian to Gregorian or Gregorian to Julian.
- X0035 determines the difference in days between two dates or can calculate the second date given one date and the difference in days
- X98DAY determines the day of the week and the day and month names for a given date.

Date Formats

Julian Format

The JD Edwards World Julian (*JUL) format is CYYDDD where:

- C is the value added to 19 to create the century (0=>19, 1=>20).
- YY is the year within the century.
- DDD is the day in the year which can range from 001 to 731(next year).

For Example:

- 098185 is July 4, 1998
- 100001 is Jan 1, 2000
- 099666 is Oct 27, 2000

Gregorian Formats

The JD Edwards World Gregorian formats are *MDY, *DMY, and *YMD where:

- M represents the 2 character month
- D represents the 2 character day
- Y represents the 2 or 4 character year

Note: Edited means punctuation such as: 12/31/98, 22.11.45, or 2000-01-01.

X0027

X0027 is an RPG program the system uses to calculate a new date from a reference date including *TODAY. The interval can be + or -, the unit can be *DAY, *WEEK, *MONTH, or *YEAR.

The information here is intended to cover the usual usage of this program. The source code has samples and more documentation. Specifically, *FORMULA is used to pass a string like “*TODAY + 1”.

Using X0027

The reference date must be scrubbed (usually copy module C0012) before being passed to X0027. X0027 calls x0028 which uses the Data Dictionary item #CYR to establish the last year of the floating 100-year business window. See the X0028.

#CYR DD default	Digits entered = (*MDY)	Julian date = CYYDDD	IBM QRY = (no leading 0's)
10	01/01/10	110001 (2010)	110,001
10	12/31/11	011365 (1911)	11,365
30	07/29/55	055241 (1955)	55,241
60	07/29/55	155241 (2055)	155,241
98	09/14/98	198257 (2098)!	198,257
97	09/14/98	098257 (1998)	98,257

Parameters

The parameters to pass to X0027 are ALL type Alpha. If the calling program is written in CL, be sure the variables are *CHAR.

The first parameter will accept a 4 digit year, for example 01011999(MMDDCCYY).

You must pass the first 8 parameters. If you want the day of the week and that day name, pass 10 parameters.

Parameter	Description
SSIDAT	SSIDAT is a 25 character field which can be any format as long as it agrees with the From format. To fetch today's date use *TODAY (+/- optional) and use *FORMULA in SFFMT.
SCENT	SCENT is a 2 character field which should be sent blank.
SFFMT	SFFMT is an 8 character field describing the FROM format. The usual values will be *JUL, *MDY, *DMY, *YMD, *SYSVAL or *FORMULA. See the program code for other allowed values.

Parameter	Description
\$OFSET	\$OFSET is the 6 character number (+/-) of the units you have chosen in parameter 5.
\$INTVL	\$INTVL is the 8 character calendar unit used to calculate the number of days. The usual values are *DAY, *WEEK, *MONTH or *YEAR. See the program code for other allowed values.
\$SODAT	\$SODAT is a 6 character field which will contain the output date. This date will need to be passed to X0028 for any editing. It should be sent blank.
\$TFMT	\$TFMT is an 8 character field describing the TO format. The usual values will be *JUL, *MDY, *DMY, *YMD, or *SYSVAL. See the program code for other allowed values.
RTNCDE	RTNCDE is a single character error code. A '1' will be placed in this in this field if X0027 receives an invalid parameter. Your program should check this field right after the call to X0027.
\$DAY# (Optional)	\$DAY# is a one character field for the day name, 1 = Sunday, 2 = Monday, and so on.
\$DAYS (Optional)	\$DAYS is a ten character field which, if it is sent, will have the day name loaded.

X0028

X0028 is the primary JD Edwards World date formatting program. Briefly, it is an RPG program which converts dates from Julian to Gregorian or Gregorian to Julian. It will do some editing, but that is not its purpose. This program will NOT convert edited dates to unedited.

Using X0028

The edited date must be scrubbed (usually copy module C0012) before being passed to X0028. The Data Dictionary item that X0028 uses is #CYR. It establishes the last year of the floating 100-year business window.

#CYR DD default	Digits entered = (*MDY)	Julian date = CYYDDD	IBM QRY = (no leading 0's)
10	01/01/10	110001 (2010)	110,001
10	12/31/11	011365 (1911)	11,365
30	07/29/55	055241 (1955)	55,241
60	07/29/55	155241 (2055)	155,241

#CYR DD default	Digits entered = (*MDY)	Julian date = CCYYDDD	IBM QRY = (no leading 0's)
98	09/14/98	198257 (2098)!	198,257
97	09/14/98	098257 (1998)	98,257

If the 2 digits for the year are less than or equal to #CYR, a 1 will be placed in the Julian century field so that when it is added to 19, a century of 20 will result.

Caution: Setting #CYR to 60 so that your business will go from 1961 to 2060 means entering a date of birth for 07/29/55 will be converted to 2055. Setting #CYR to 98 means you have a range from 1999 to 2098. Think about it!

In the prior paragraph, the stored Julian values were shown. Once the data has been stored in the file, no comparison or decision is necessary. The value of the century digit is added to 19 and the rest of the data will be used to compute the day and month. Remember, when displaying a file with QRY, the leading zero will not be displayed. DSPPFM will show you all the digits.

Parameters

The parameters to pass to X0028 are ALL type Alpha. If the calling program is written in CL, be sure the variables are *CHAR.

If you do not need 4 position years, pass the first 6 parameters. If you need a 4 position year, pass all 10 or 11 parameters.

Parameter	Description
SSIDAT	SSIDAT is a 6 character field which can be any format as long as it agrees with the From format field and has only digits from 0-9 or is all zeros. X0028 moves the converted unedited date back into this field.
#EDAT	#EDAT is the 8 character edited field. It has two extra positions to allow for the #SEP characters inserted by X0028. It will have the format specified by #TFMT. X0028 blanks this field upon receipt.
#FFMT	#FFMT is a 7 character field describing the FROM format. It can be *SYSVAL, *JUL, *MDY, *DMY, or *YMD.
#TFMT	#TFMT a 7 character field describing the TO format. It can be *SYSVAL, *JUL, *MDY, *DMY, or *YMD.
#SEP	#SEP is a 7 character field for the separator character to be used in editing the date. It can be *SYSVAL, *NONE, blank or a single editing character like '-', '/' or '.'.

Parameter	Description
\$ERTST	\$ERTST is a single character error code. A '1' will be placed in this in this field if X0028 receives an invalid date or invalid data. Your program should check this field right after the call to X0028.
\$CTRY (Optional)	\$CTRY is a two character field which will contain the century, 19 or 20. It should be sent with blanks.
#FJPN (Optional)	#FJPN is a one character field used for Japanese Era dates.
#TJPN (Optional)	#TJPN is a one character field used for Japanese Era dates.
#EDAT2	#EDAT2 is a 10 character field used for edited dates with 4 digits for the year. You MUST use this parameter if you want a 4 digit year date. It has two extra positions to allow for the #SEP characters inserted by X0028. It will have the format specified by #TFMT.
#SIDT2	#SIDT2 is an optional 8 character field intended for use with 4 digit year dates. It can be any format as long as it agrees with the From format field and has only digits from 0-9. If this parameter is used and the from format is Julian, the Julian date needs to be loaded here with 2 leading zeros and #SIDAT should be loaded with zeros. X0028 moves the converted unedited date back into this field as well as #SIDAT.

X0035

X0035 is an RPG program which determines the difference in days between two dates or can calculate the second date given one date and the difference in days. It uses X0028 for date conversion.

Using X0035

The Data Dictionary item that X0028 uses is #CYR. It establishes the last year of the floating 100-year business window. See X0028.

Parameters

The parameters to pass to X0035 are ALL type Alpha. If the calling program is written in CL, be sure the variables are *CHAR.

Parameter	Description
#SIDA1	#SIDA1 is a 6 character date field which can be any format as long as it agrees with the #SIFM1 format and has only digits from 0-9. It can be sent as blanks if a reverse date is to be calculated from #SIDA2.

Parameter	Description
#SIDA2	#SIDA2 is a 6 character date field which can be any format as long as it agrees with the #SIFM2 format and has only digits from 0-9. It can be sent as blanks if a forward date is to be calculated from #SIDA1.
#SIFM1	#SIFM1 is a 7 character field describing the #SIDA1 format. It can be *SYSVAL, *JUL, *MDY, *DMY, or *YMD.
#SIFM2	#SIFM2 is a 7 character field describing the #SIDA2 format. It can be *SYSVAL, *JUL, *MDY, *DMY, or *YMD.
#SITYE	#SITYE is a one-character field which is used when calculating the difference in days between two dates. It is usually sent blank. If the difference between two dates is to be calculated, a blank means one date is subtracted from the other. If this field is a 1, a 1 is added to the difference so that the beginning and ending date are included in the count. If this field is a 2, a 1 is subtracted from the difference which means neither date will be included in the count.
\$ERTST	\$ERTST is a one character field which should be sent blank. It will contain a 1 if two blank dates are sent, an invalid format is sent, or X0028 finds a problem with either date.
\$#X	\$#X is a 9 character field which contains the number of day difference to be calculated between #SIDA1 and #SIDA2. If #SIDA1 is blank, the reverse date will be placed in #SIDA1. If #SIDA2 is blank, the future date will be placed in #SIDA2.
#SICT1 (Optional)	#SICT1 is a 2 character century field for #SIDA1.
#SICT2 (Optional)	#SICT2 is a 2 character century field for #SIDA2.

X98DAY

X98DAY is an RPG program which determines the day of the week and the day and month names for a given date. It uses X0028 for date conversion.

Using X98DAY

The Data Dictionary item that X0028 uses is #CYR. It establishes the last year of the floating 100-year business window. See the User's Guide for X0028.

Note: X98DAY only works inside this 100-year window.

Parameters

The parameters to pass to X0035 are ALL type Alpha. If the calling program is written in CL, be sure the variables are *CHAR.

Parameter	Description
#SIDAT	#SIDAT is a 6 character field which can be any format as long as it agrees with the From format field and has only digits from 0-9.
#SIFMT	#SIFMT is a 7 character field describing the FROM format. It can be *SYSVAL, *JUL, *MDY, *DMY, or *YMD.
#SIDAY	#SIDAY is a 29 character field which will contain the converted day name, month name, day and year. If an 'A' is sent in this field, the month and day abbreviations will be used for a maximum size of 17 characters.
#SNDAY	#SNDAY is a one character field which will contain the number for the day of the week, 1 = Sunday, 7 = Saturday.
#SOFMT	#SOFMT is a 16 character field which is used to send an override output format. The edit codes are MM, DD, YY, YYYY, AM and AD where AM and AD are the abbreviated month and day names. The entries can be separated by comma, :, ., /, or a blank.

Miscellaneous

Fields Selected
defined, 4-65

*

*ALL
using, 4-57
*BOTH
using, 4-56
*BOTH and *ALL features
using, 4-56

A

About attention MENU window, 7-11
About CASE profiles, 3-71
About common subroutines, 6-2
About creating libraries, 3-6
About data dictionary repository, 4-13
About designing the report, 4-89
About development environments, 3-5
About file servers, 6-87
About functional servers, 6-101
About group jobs, 7-1
About performance issues, 6-135
About program specifications, 5-3
About programming tools, 4-1
About record formats, 4-65
About report design aid, 4-83
About SAR log, 3-77
About SAR system setup, 3-39
About screen design aid, 4-43
About setting up universal file converter, 8-7
About software scan and replace, 6-133
About source debugger, 6-123
About standard subroutines, 6-1
About subroutines, 6-11
About the conversion rule program, 8-22
About the data file design aid, 4-33
 automatic reference to field reference files, 4-33
 data dictionary validation, 4-33
 enforced naming conventions, 4-33
 enforced prefixes, 4-33
 resequencing, 4-34
About the field reference file, 4-29
About the JD Edwards World message file, 4-30

About the program overview, 6-1
About user indexes, 6-57
About user spaces, 6-49
About using IBM pass-through with group jobs, 7-15
About version control, 3-1
Accessing CASE profiles, 3-71
Accessing report design aid, 4-90
Accessing SAR log inquiry, 3-80
Accessing the JD Edwards World attention MENU window, 7-11
Accessing the JD Edwards World group job window, 7-4
Accessing the software versions repository, 3-45
Accessing the user index, 6-112
Accessing the user space, 6-112
Accessing versions setup, 8-9
Action Code
 defined, 4-51
Activating suspended group jobs, 7-5
Add
 promotion path, 3-88
Add Cross Over Instructions form (00312), 8-20
Adding a literal video field
 adding, 4-56
Adding fields, 8-20
Adding record types, 3-36
Address numbers
 searching for, 3-26
Advanced functions of the JD Edwards World group job window, 7-8
 entering commands, 7-9
 hidden selections, 7-8
Application development cycle, 1-1
Assign
 project SABs, 3-95
 promotion paths, 3-94
Attention MENU window, 7-11
 accessing, 7-11
 functions summary, 7-12
ATTN key program
 setting, 7-3

B

Based On File
 defined, 4-40
Basic accounting transactions

graphic, 6-104
 Build Transfer Library form (P98312), 3-106
 Building blocks
 JD Edwards World software, 1-1

C

Calculation specifications, 5-6
 Call Program form (CALL), 7-8
 CASE profiles, 3-71
 accessing, 3-71
 summary, 3-75
 CASE Profiles form (P98009), 3-72, 3-78
 Change Library List form (CHGLIBL), 7-7
 Changing compile option defaults for reports, 4-97
 Changing record types, 3-37
 Changing subfile boundaries, 4-77
 to make larger, 4-78
 to make smaller, 4-77
 Changing to non-group mode, 7-6
 Compile option defaults for reports
 changing, 4-97
 Compiling a report, 4-96
 Compiling your video, 4-60
 Conducting a search, 4-10
 Continue execution
 source debugger, 6-128
 Control fields
 user space, 6-116
 Control Files Copy form (P924127), 3-103
 Control parameters, 6-89
 Control specifications, 5-4
 Conventions
 naming, 3-52
 Conversion forms
 creating, 8-31
 Conversion rule program, 8-22
 fifth parameter, 8-22
 first parameter, 8-22
 fourth parameter, 8-22
 second parameter, 8-22
 third parameter, 8-22
 Copy DD, VO, DW, UDC, SVR, Menus form
 (P99630), 3-22, 3-111
 Copy File form (CPYF), 3-14, 3-20
 Copy modules, 3-63
 COPY modules
 functional servers, 6-111
 Copying a file, 3-19
 Copying a library, 3-18
 Copying a record, 3-20
 Copying data to your development environment,
 3-18
 Copying JD Edwards World record types, 3-21
 Create Library form (CRTLIB), 3-14
 Create New Screen form (P92510), 4-51
 Create Source Physical File form (CRTSRC PF), 3-16

Creating a development environment, 3-5
 Creating a development source library, 3-14
 Creating conversion forms, 8-31
 Creating development object libraries, 3-13
 Creating JDESRC with JD Edwards World
 program generator, 3-14
 Creating JDESRC without the program generator,
 3-16
 Creating libraries, 3-11
 Creating new job groups, 7-4
 Creating record type codes, 3-39
 Creating the transfer library, 3-105
 Creating user space and user index, 6-111
 Cross Reference form (P980014), 3-66
 Crossover rules
 working with, 8-15
 Crossover Rules form (0031), 8-15, 8-19

D

D

defined, 4-68
 Data dictionary
 security, 4-21
 user defined help instructions, 4-26
 working with, 4-16
 Data Dictionary form (P9201), 4-16, 8-33
 Data dictionary glossary
 groups, 4-22
 working with, 4-22
 working with by file, 8-33
 Data dictionary repository, 4-13
 Data dictionary structure
 data field display text, 4-14
 data field specifications, 4-14
 data item aliases, 4-15
 data item alpha description, 4-14
 data item master, 4-14
 error message program ID, 4-15
 glossary text file, 4-15
 key index file, 4-15
 understanding, 4-13
 Data Field Alias form (P9204), 4-22
 Data field descriptions
 working with, 4-27
 Data Field Descriptions form (P9202), 4-27
 Data file design aid, 4-33
 entering, 4-35
 Data File Design Aid form (P92102), 4-35, 4-38
 Data file design aid summary, 4-41
 Data integrity logic, 6-106
 Data item alias
 revisions, 4-21
 Data Item Glossary Revisions form (P92001), 4-24,
 4-26, 8-34
 Data item name
 locating, 4-15

- Data Item Size
 - defined, 4-68
 - Data Item Type
 - defined, 4-68
 - Data modeling
 - accessing, 4-3
 - selection exits, 4-6
 - working with, 4-3
 - Data Modeling form (P98042), 4-3, 4-5
 - Database considerations, 8-5
 - Database fields
 - selecting, 4-66
 - Debugger
 - using with interactive program, 6-123
 - with batch program, 6-126
 - Define
 - project, 3-106
 - promotion path, 3-106
 - promotion path for control tables, 3-91
 - promotion path for source code members, 3-89
 - Defining a project, 3-92
 - Defining a promotion path, 3-87
 - Defining access for a user profile using J98INIT, 3-17
 - Defining access for a user profile using J98INITA, 3-17
 - Defining record type titles, 3-41
 - Deleting records, 8-20
 - Description
 - defined, 4-51, 4-68
 - Design Aid Exit/Save form (P92590), 4-59
 - Designing the report, 4-89
 - Detail Spec. Over Titles form (P48002), 3-37
 - Detail Spec. Types form (P00051), 3-36
 - Detailed explanation of a line, 4-5
 - Determining
 - program environment, 6-124
 - Developer's Workbench form (G9362), 3-80
 - Development cycle, 1-1
 - Development environment
 - copying data to, 3-18
 - Development environments, 3-5
 - rules for creating, 3-5
 - Direction
 - defined, 4-5
 - Displaying field descriptions, 8-18
 - documentation, 6-6
 - Domestic Field Begin Pos
 - defined, 8-17
 - Domestic Field Data Type
 - defined, 8-17
 - Domestic Field Decimal Positions
 - defined, 8-17
 - Domestic Field Description
 - defined, 8-17
 - Domestic Field Name
 - defined, 8-16
 - Domestic Field Size In Bytes
 - defined, 8-17
 - Domestic File Name
 - defined, 8-16
 - Domestic Number of Digits
 - defined, 8-17
- E**
- Edit and Promote form (P92412), 3-97
 - Editing commands, 4-45
 - End debug (ENDDBG), 6-132
 - ENDDBG (end debug), 6-132
 - Entry program
 - setting up business rules, 6-103
 - Error handling, 6-3
 - Error message
 - index line (COORIX), 6-117
 - Examples
 - production and development, 3-8
 - Executing
 - program being debugged, 6-125
 - source debugger, 6-127
 - Existing field
 - updating, 4-48
 - Exits
 - function key, 4-48
 - Extension specifications, 5-5
- F**
- Fast path create
 - accessing for a new form, 4-51
 - Fast Path Create
 - defined, 4-51
 - Fast Path File
 - defined, 4-65, 4-94
 - Features
 - system integration, 2-1
 - Field definition
 - revising, 4-80
 - Field Definition report (P928400), 4-91
 - Field definition window
 - report design aid vs. screen design aid, 4-84
 - Field descriptions
 - displaying, 8-18
 - Field in RDA
 - updating, 4-90
 - Field Name
 - defined, 4-68
 - Field name standards, 4-46
 - Field reference file, 4-29
 - rebuild, 4-30
 - Field Selection List form (P92524), 4-67, 4-69
 - Fields
 - # Fields Selected, 4-65
 - Action Code, 4-51

- add hidden, 4-75
 - adding, 8-19
 - adding without using pick list, 4-54
 - Based On File, 4-40
 - cover page
 - graphic, 4-85
 - D, 4-68
 - Data Item Size, 4-68
 - Data Item Type, 4-68
 - Description, 4-51, 4-68
 - Direction, 4-5
 - Domestic Field Begin Pos, 8-17
 - Domestic Field Data Type, 8-17
 - Domestic Field Decimal Positions, 8-17
 - Domestic Field Description, 8-17
 - Domestic Field Name, 8-16
 - Domestic Field Size In Bytes, 8-17
 - Domestic File Name, 8-16
 - Domestic Number of Digits, 8-17
 - Fast Path Create, 4-51
 - Fast Path File, 4-65, 4-94
 - Field Name, 4-68
 - File Description, 4-34, 4-40
 - Fld Pfx, 4-65, 4-94
 - Foreign Field Name, 8-16
 - Foreign File Name, 8-16
 - Format Name, 4-65, 4-94
 - HDG, 4-68
 - Logical Files, 4-41
 - Opt, 4-64, 4-94
 - Ordering of Fields, 4-41
 - OVERLAY, 4-52
 - PUTOVR, 4-52
 - Quantifier, 4-5
 - Related Record, 4-65, 4-94
 - report header
 - graphic, 4-85
 - Selection Exits, 4-52
 - Seq No, 4-67
 - Start/End Lines, 4-65, 4-94
 - Subfile, 4-51
 - Subfile Clear, 4-52
 - Subfile Fold, 4-52
 - Subfile portion of screen, 4-6
 - Type, 4-6, 4-65, 4-94
 - Unique Keys, 4-40
 - Use, 4-68
 - Version, 8-16
 - Wide Screen, 4-51
 - Window, 4-51
 - work, 6-7
- File conversion
 - working with, 8-25
 - File Description
 - defined, 4-34, 4-40
 - File description specifications, 5-4
 - File prefix, 4-34
 - File Selection List form (P92522), 4-69
 - File servers, 6-87
 - advantages, 6-88
 - commonly used, 6-99
 - disadvantages, 6-88
 - how does it function, 6-89
 - implementing, 6-92
 - tips when using, 6-94
 - types, 6-88
 - what is a, 6-87
 - Fld Pfx
 - defined, 4-65, 4-94
 - Flow of typical program
 - graphic, 6-102
 - Foreign Field Name
 - defined, 8-16
 - Foreign File Name
 - defined, 8-16
 - Format Keyword Maintenance form (P92537), 4-72
 - Format Name
 - defined, 4-65, 4-94
 - Forms
 - Add Cross Over Instructions, 8-20
 - Build Transfer Library, 3-106
 - Call Program, 7-8
 - CASE Profiles, 3-72, 3-78
 - Change Library List, 7-7
 - Control Files Copy, 3-103
 - Copy DD, VO, DW, UDC, SVR, Menus, 3-22, 3-111
 - Copy File, 3-20
 - Copy File (CPYF), 3-14
 - Create Library (CRTLIB), 3-14
 - Create New Screen, 4-51
 - Create Source Physical File (CRTSRCPF), 3-16
 - Cross reference, 3-66
 - Crossover Rules, 8-15, 8-19
 - Data Dictionary, 4-16, 8-33
 - Data Field Alias, 4-22
 - Data Field Descriptions, 4-27
 - Data file design Aid, 4-35
 - Data File Unit, 4-38
 - Data Item Flossary Revisions, 4-24, 4-26
 - Data Item Glossary Revisions, 8-34
 - Data Modeling, 4-3, 4-5
 - Design Aid Exit/Save, 4-59
 - Detail Spec. Over Titles, 3-37
 - Detail Spec. Types, 3-36
 - Developer's Workbench, 3-80
 - Edit and Promote, 3-97
 - field Selection List, 4-67, 4-69
 - File Selection List, 4-69
 - Format Keyword Maintenance, 4-72
 - FRF & JD Edwards World Msg File, 4-31
 - Function Key/Opt Definition, 4-73
 - Item Maintenance, 4-52, 4-53
 - Item Master Information, 4-54, 4-56, 4-57, 4-58, 4-70, 4-76, 4-78, 4-79

- Item Search, 4-47, 4-48, 4-65
 - JD Edwards World Passthru Network, 7-17
 - JD Edwards World Visual Debug, 6-125, 6-127, 6-129
 - Library List Revisions, 3-17
 - List of Defined Fields, 4-75
 - Load Transferred Library, 3-109
 - Manage Program Paths, 3-87
 - Manage Projects, 3-93
 - Next Numbers, 4-28
 - Next Numbers by Company/Fiscal, 4-29
 - Object Cross Reference Repository, 4-9, 4-10, 4-11
 - Pre Promotion Edit Details, 3-100
 - Pre Promotion Edit History, 3-100
 - Print Transfer Report, 3-107
 - Processing Options Revisions, 7-16, 8-10, 8-25, 8-31, 8-34
 - Project Elements, 3-95
 - Project Promotion Paths, 3-94, 3-99
 - Promotion Path, 3-88
 - Promotion Path Control Files, 3-91
 - Promotion Path Members, 3-89
 - Record Formats List, 4-64, 4-66, 4-68
 - Record Type Titles, 3-41
 - Remove Member (RMVM), 3-15
 - Restore Library, 3-107
 - SAR Log Inquiry, 3-81
 - SAR Log Transfer, 3-98
 - Save Library, 3-107
 - Single D/B Relation Entry, 4-62
 - Software Scan and Replace, 6-133
 - Software Transfer, 3-103
 - Software Versions Repository, 3-45, 3-52, 3-62, 3-63, 3-64, 3-65, 3-72, 3-81, 4-34, 4-60, 4-90, 4-96, 6-124
 - Start Pass-Through, 7-17
 - Universal File Converter, 8-8, 8-25
 - User Defined Code Revisions, 3-40
 - User Defined Codes Window, 4-11
 - User Information, 3-18, 7-3, 7-11
 - User Signon List Revisions, 3-17
 - Versions List, 8-9
 - Versions Setup, 7-15
 - FRF & JD Edwards World Msg File form (P98FRF), 4-31
 - Function key
 - exits from screen design aid, 4-70
 - Function key exits
 - return to previous panel, 4-48
 - Function key exits from screen design aid, 4-70
 - Function Key/Opt Definition form (P9601), 4-73
 - Function keys, 3-66
 - data dictionary, 4-21
 - flow program/illustrate file models, 3-67
 - function key/opt definition, 4-73
 - JD Edwards World command line, 4-70
 - subfile drop, 4-71
 - toggle monochrome/color display, 4-72
 - Functional server
 - interactive program cycle, 6-112
 - Functional server error messages
 - graphic, 6-110
 - Functional server highlights, 6-104
 - Functional server interface, 6-110
 - graphic, 6-109
 - Functional server parameters, 6-110
 - Functional server program sections
 - graphic, 6-117
 - Functional server transaction data
 - graphic, 6-109
 - Functional server user index, 6-111
 - Functional server user space, 6-110
 - Functional server/COPY modules, 6-111
 - Functional servers, 6-101
 - advantages, 6-102
 - call parameters, 6-113
 - COPY modules, 6-111
 - disadvantages, 6-103
 - highlights, 6-104
 - how they function, 6-103
 - parameters, 6-110
 - user indexes, 6-111
 - user space, 6-110
 - what are they, 6-101
 - Functions
 - navigation, 3-64
 - Functions and options
 - delete records, 8-24
- ## G
- General aesthetics, 4-62
 - alignment, 4-62
 - grouping fields, 4-62
 - spacing, 4-62
 - Graphics
 - project attributes, 3-86
 - Group job window
 - accessing, 7-4
 - advanced functions, 7-8
 - entering commands, 7-9
 - hidden selections, 7-8
 - working with, 7-3
 - Group job window summary, 7-10
 - Group jobs, 7-1
 - activating suspended, 7-5
 - signing off with suspended, 7-6
 - terminating, 7-6
 - working with non-JD Edwards World, 7-7
 - Guidelines, 3-92, 6-6
 - scan and replace, 6-134

H

- HDG
 - defined, 4-68
- Hidden fields, 4-75
- How does a file server function?, 6-89
- How does a functional server function?, 6-103
- How does a user index function?, 6-58

I

- IBM pass-through
 - setting up access to remote locations, 7-15
 - using with group jobs, 7-16
 - working with, 7-15
- Identifiers
 - file information, 3-48
 - maintenance on a logical file, 3-48
 - member, 3-45
 - member relationship and compiling information, 3-47
 - processing a screen, 3-48
 - type, use, and associated systems, 3-46
 - where members are maintained, 3-49
- Implementing a file server, 6-92
- Index line (COORIX)
 - error message, 6-117
- Initiating
 - source debugger, 6-124, 6-126
- Input specifications, 5-5
- Interactive program cycle using a functional server, 6-112
- Internal RPG subroutines within
 - JD Edwards World programs, 6-11
- Item description
 - graphic, 4-88
- Item Maintenance form (P92700), 4-52, 4-53
- Item Master Information form (P928011), 4-54, 4-56, 4-57, 4-58, 4-70, 4-76, 4-78, 4-79
- Item Search form (P928200), 4-47, 4-48, 4-65

J

- J98INIT
 - defining access for a user profile, 3-17
- J98INITA
 - defining access for a user profile, 3-17
- JD Edwards World group job window summary, 7-10
- JD Edwards World open application architecture
 - graphic, 6-108
- JD Edwards World Passthru Network form (B98P), 7-17
- JD Edwards World Visual Debug form (P3701), 6-125
- JD Edwards World Visual Debug form (P93701), 6-127, 6-129

- Job groups
 - creating new, 7-4
- Join logical files, 3-63

K

- Key lists
 - searching for, 6-93
- Keywords, 8-21
 - business unit, 8-21
 - check data dictionary, 8-21
 - data dictionary default, 8-21
 - dates, 8-21
 - default constant, 8-21
 - initialize, 8-21
 - next number, 8-21
 - user defined code lookup, 8-21

L

- Libraries
 - CLTSEC, 3-7
 - copying, 3-18
 - creating, 3-6, 3-11
 - creating common, 3-11
 - creating development object, 3-13
 - data, 3-7
 - install, 3-7
 - JD Edwards World, 3-6
 - JDEINSTAL, 3-7
 - JDFDATA, 3-7
 - JDFINS, 3-7
 - JDFOBJ, 3-7
 - JDFSRC, 3-6
 - object, 3-7
 - plans, 3-7
 - security, 3-7
 - source, 3-6
 - understanding development source, 3-13
- Library List Revisions form (P0094), 3-17
- List of Defined Fields form (P92540), 4-75
- Literal field
 - adding, 4-56
- Load Transferred Library form (P98312), 3-109
- Loading the transfer library, 3-109
- Locate
 - a project, 3-93
 - promotion path, 3-88
- Locating a data item name, 4-15
- Locating the next numbers facility, 4-28
- Locating the rebuild FRF and JD Edwards World
 - Msg file form, 4-31
- Logic
 - data integrity, 6-106
 - user interface, 6-106
- Logical file
 - sample, 4-37

Logical files, 3-62
 join, 3-63
 Logical Files
 defined, 4-41
 Logical files with omits
 sample, 4-38
 Logical files with selects
 sample, 4-38

M

Manage Projects form (P92413), 3-93
 Manage Promotion Paths form (P92403), 3-87
 Member identifiers, 3-45
 Menu flow, 4-34
 Merge functions for PTFs and reinstalls, 4-41
 Message file
 JD Edwards World, 4-30
 locating rebuild FRF & JD Edwards World Msg,
 4-31
 rebuilding, 4-30
 Miscellaneous items, 6-7
 key list (KLIST), 6-7
 naming conventions, 6-7
 Modules
 copy, 3-63

N

Naming conventions, 3-52
 Naming conventions for objects, 3-54
 Next Numbers by Company/Fiscal form (P00021),
 4-29
 Next numbers facility
 locating, 4-28
 working with, 4-27
 working with by company & fiscal year, 4-29
 Next Numbers form (P0002), 4-28
 Non-group mode
 changing to, 7-6

O

Object cross reference repository
 working with, 4-9
 Object Cross Reference Repository form (P980014),
 4-9, 4-10, 4-11
 Open application architecture
 graphic, 6-108
 Opt
 defined, 4-64, 4-94
 Optional files workbench, 3-62
 Ordering of Fields
 defined, 4-41
 Output specifications, 5-7
 OVERLAY
 defined, 4-52

P

Parameters
 call, for the functional server, 6-113
 control, 6-89
 functional servers, 6-110
 returned, 6-92
 Performance issues, 6-135
 Pick list
 placing fields on a form using, 4-70
 Placing fields on a form using a pick list, 4-70
 Pre Promotion Edit Details form (P92431), 3-100
 Pre Promotion Edit History form (P9243), 3-100
 Prefix standards, 4-45
 Prepare
 SAR system, 3-106
 Print Transfer Report form (P98312), 3-107
 Printing a report, 8-29
 Printing the transfer library report, 3-107
 Process overview - placing selected fields, 4-78
 Process overview - revising the field definition, 4-
 79
 Processing Options Revisions form (P98312), 7-16,
 8-10, 8-25, 8-31, 8-34
 Production and development examples, 3-8
 basic development environment, 3-9
 basic production environment, 3-8
 common shared library, 3-9
 no source in production environment, 3-9
 one source and object library, 3-10
 Profiles
 defining access for using J98INIT, 3-17
 understanding user, 3-16
 Program and file names, 3-61
 Program being debugged
 executing, 6-125
 Program environment
 determining, 6-124
 Program generator
 creating JDESRC, 3-14
 creating JDESRC without, 3-16
 Program overview, 6-1
 Program specifications, 5-3
 Program Structure, 6-11
 Programs and IDs
 0031 (crossover rules), 8-15, 8-19
 00312 (add cross over instructions), 8-20
 B98P (JD Edwards World passthru network), 7-
 17
 CALL (call program), 7-8
 CHGLIBL (change library list), 7-7
 CPYF (copy file), 3-14, 3-20
 CRTLIB (create library), 3-14
 CRTSRCPF (create source physical file), 3-16
 G9362 (developer's workbench), 3-80
 G9841 (universal file converter), 8-8, 8-25
 P0002 (next numbers), 4-28

- P00021 (next numbers by company/fiscal), 4-29
- P00051 (detail spec. types), 3-36
- P00051 (user defined code revisions), 3-40
- P0092 (user information), 3-18, 7-3, 7-11
- P0093 (user signon list revisions), 3-17
- P0094 (library list revisions), 3-17
- P08332 (single d/b relation entry), 4-62
- P3701 (JD Edwards World visual debug), 6-125
- P48002 (detail spec. over titles), 3-37
- P48002 (record type titles), 3-41
- P81QM (user defined codes window), 4-11
- P90630 (Copy DD, VO, DW, UDC, SVR, Menus), 3-111
- P92001 (data item glossary revisions), 4-24, 4-26, 8-34
- P9201 (data dictionary), 4-16, 8-33
- P9202 (data field descriptions), 4-27
- P9204 (data field alias), 4-22
- P92102 (data file design aid), 4-35, 4-38
- P9240 (promotion path), 3-88
- P92401 (promotion path members), 3-89
- P92402 (promotion path control files), 3-91
- P92403 (manage program paths), 3-87
- P92411 (project promotion paths), 3-94
- P92411W (project promotion paths), 3-99
- P92412 (project elements), 3-95
- P92412 (edit and promote), 3-97
- P924124 (software transfer), 3-103
- P924127 (control files copy), 3-103
- P92413 (manage projects), 3-93
- P9242 (SAR log transfer), 3-98
- P9243 (pre promotion edit history), 3-100
- P92431 (pre promotion edit details), 3-100
- P92510 (create new screen), 4-51
- P92520 (record formats list), 4-64, 4-66, 4-68
- P92522 (file selection list), 4-69
- P92524 (field selection list), 4-67, 4-69
- P92537 (format keyword maintenance), 4-72
- P92540 (list of defined fields), 4-75
- P92590 (design aid exit/save), 4-59
- P92700 (item maintenance), 4-52, 4-53
- P928011 (item master information), 4-54, 4-56, 4-57, 4-58, 4-70, 4-76, 4-78, 4-79
- P928200 (item search), 4-47, 4-48, 4-65
- P928400 (field definition report), 4-91
- P93701 (JD Edwards World visual debug), 6-127, 6-129
- P9601 (function key/opt definition), 4-73
- P980014 (cross reference), 3-66
- P980014 (Object Cross Reference Repository), 4-9, 4-10, 4-11
- P98009 (CASE profiles), 3-78
- P98009 (CASE Profiles), 3-72
- P9801 (software versions repository), 3-45, 3-52, 3-62, 3-63, 3-64, 3-65, 3-72, 3-81, 4-34, 4-60, 4-90, 4-96, 6-124
- P98042 (data modeling), 4-3, 4-5
- P9810 (SAR log inquiry), 3-81
- P98300 (versions list), 8-9
- P98300 (versions setup), 7-15
- P98312 (build transfer library), 3-106
- P98312 (load transferred library), 3-109
- P98312 (print tranfer report), 3-107
- P98312 (processing options revisions), 7-16, 8-10, 8-25, 8-31, 8-34
- P98810 (software scan and replace), 6-133
- P98FRF (FRF & JD Edwards World Msg file), 4-31
- P99630 (copy DD, VO, DW, UDC, SVR, Menus), 3-22
- RMVM (remove member), 3-15
- RSTLIB (restore library), 3-107
- SAVLIB (save library), 3-107
- STRPASTHR (start pass-through), 7-17
- Project
 - define, 3-106
 - defining, 3-92
 - promote, 3-106
 - promoting, 3-97, 3-102
 - to locate, 3-93
 - to promote, 3-102
- Project Elements form (P92412), 3-95
- Project Promotion Paths form (P92411), 3-94
- Project Promotion Paths form (P92411W), 3-99
- Project SABs
 - to assign, 3-95
- Project updates
 - promoting, 3-105
- Promote
 - project, 3-102, 3-106
- Promoting a project, 3-97, 3-102
- Promoting project updates, 3-105
- Promotion path
 - define, 3-106
 - validate, 3-106
 - validating, 3-99
- Promotion Path Control Files form (P92402), 3-91
- Promotion Path form (P9240), 3-88
- Promotion Path Members form (P92401), 3-89
- Promotion paths
 - defining, 3-87
 - to add, 3-88
 - to assign, 3-94
 - to define for control tables, 3-91
 - to define for source code members, 3-89
 - to locate, 3-88
 - to validate, 3-99
 - understanding, 3-85
- Promotion paths and projects
 - working with, 3-85
- PTFs
 - merge functions, 4-41
- PUTOVR
 - defined, 4-52

Q

Quantifier
defined, 4-5

R

Reading from a user space, 6-54
Record formats, 4-65
Record Formats List form (P92520), 4-64, 4-66, 4-68
Record type codes
creating, 3-39
Record type titles
defining, 3-41
verifying, 3-42
Record Type Titles form (P48002), 3-41
Record types
adding, 3-36
changing, 3-37
copying, 3-21
Records
copying, 3-20
deleting, 8-20
Reinstalls
merge functions, 4-41
Related Record
defined, 4-65, 4-94
Remove Member form (RMVM), 3-15
Report
compiling, 4-96
printing, 8-29
scan and replace, 6-134
Report design aid, 4-83
accessing, 4-90
Report design standards, 4-88
general aesthetics, 4-89
JD Edwards World standards/record formats, 4-88
RDA features, 4-88
Report formats, 4-86
Reports
Field Definition, 4-91
Print Install Records, 3-109
Software Scan and Replace, 6-134
Universal File Converter, 8-29
Restore Library form (RSTLIB), 3-107
Restoring the transfer library from tape, 3-107
Retrieving data from a user index, 6-64
Returned parameters, 6-92
Review an RPG program's source, 6-16
RPG program's source
review, 6-16
Rules for creating development environments, 3-5

S

SAR information

selecting types to log, 3-79
SAR log, 3-77
SAR log inquiry
accessing, 3-80
selection exits, 3-82
summary, 3-83
SAR Log Inquiry form (P9810), 3-81
SAR Log Transfer form (P9242), 3-98
SAR logging
setting up user input options, 3-78
SAR system
setup, 3-39
to prepare, 3-106
SARs
updating, 3-97
Save Library form (SAVLIB), 3-107
Saving the transfer library to tape, 3-107
Scan and replace, 6-133
guidelines, 6-134
report, 6-134
working with, 6-133
Screen design aid, 4-43
summary, 4-82
working with, 4-47
Screen design standards and tips, 4-60
alpha fields, 4-61
default cursor, 4-61
description fields, 4-61
fold area, 4-61
line 24, 4-60
title, 4-60
window, 4-61
SDAExit/save function key, 4-59
Search
conducting, 4-9, 4-10
Searching for key lists, 6-93
Selected fields
placing, 4-78
Selecting database fields, 4-66
Selecting types of SAR information to log, 3-79
Selection exits
data modeling, 4-6
fields, 4-7
software versions repository, 3-67
where used, 4-7
Selection Exits
defined, 4-52
Selection exits from the scheduling workbench, 3-36
processing options, 3-36
Seq No
defined, 4-67
Setting the break point
source debugger, 6-127
Setting up universal file converter, 8-8
Setting up user input options for SAR logging, 3-78
Signing off with suspended group jobs, 7-6

Single D/B Relation Entry form (P08332), 4-62
Software Versions Repository form (P9801), 3-81
Software Scan and Replace form (P98810), 6-133
Software Scan and Replace report (P98810), 6-134
Software Transfer form (P924124), 3-103
Software versions repository
 accessing, 3-45
 selection exits, 3-67
Software Versions Repository
 working with, 3-43
Software Versions Repository form (P9801), 3-45, 3-52, 3-62, 3-63, 3-64, 3-65, 3-72, 4-34, 4-60, 4-90, 4-96, 6-124
Source debugger, 6-123
 continue execution, 6-128
 execute the program, 6-127
 features, 6-129
 initiating, 6-124, 6-126
 set the break point, 6-127
Standards
 field name, 4-46
 prefixes, 4-45
 screen design, 4-60
Start Pass-Through form (STRPASTHR), 7-17
Start/End Lines
 defined, 4-65, 4-94
Subfile
 defined, 4-51
Subfile boundaries
 changing, 4-77
Subfile Clear
 defined, 4-52
Subfile Fold
 defined, 4-52
Subfile portion of screen
 defined, 4-6
Subroutines, 6-1
 common, 6-2
 internal RPG, within JD Edwards World programs, 6-11
 standard, 6-1
Summary of CASE profiles, 3-75
Summary of JD Edwards World attention MENU window functions, 7-12
Summary of screen design aid, 4-82
Summary of the SAR log inquiry, 3-83
Suspended group jobs
 signing off with, 7-6
System integration, 2-1
 features, 2-1

T

Terminating group jobs, 7-6
Tips when using file servers, 6-94
To work with software scan and replace, 6-133

Tracking information if writing variable length records, 6-54
Traditional architecture
 graphic, 6-105, 6-107
Transferring individual control table records, 3-111
Type
 defined, 4-6, 4-65, 4-94

U

Understanding development source libraries, 3-13
Understanding promotion paths, 3-85
Understanding the data dictionary structure, 4-13
Understanding the SDA exit/save function key, 4-59
Understanding the select all function key, 4-67
Understanding the universal file converter setup, 8-7
Understanding user profiles, 3-16
Unique Keys
 defined, 4-40
Universal building blocks of JD Edwards World software, 1-1
Universal file converter, 8-1
 business unit, 8-4
 check data dictionary, 8-4
 data dictionary default, 8-4
 database considerations, 8-5
 dates, 8-4
 default constant, 8-4
 graphic, 8-2
 initialization, 8-4
 next number, 8-4
 numeric fields, 8-4
 setting up, 8-7, 8-9
 special processing, 8-4
 understanding set up, 8-7
 user defined code lookup, 8-4
 user responsibilities, 8-5
Universal File Converter form (G9841), 8-8, 8-25
Universal File Converter report (P98300), 8-29
Updating a field in RDA, 4-90
Updating an existing field, 4-48
Updating the SARs, 3-98
Updating/adding fields through SDA, 4-47
Usage
 indicator, 6-4
Use
 defined, 4-68
User Defined Code Revisions form (P00051), 3-40
User Defined Codes Window form (P81QM), 4-11
User defined help instructions
 data dictionary, 4-26
User indexes, 6-57
 accessing, 6-112
 advantages, 6-58
 creating, 6-59, 6-111

- functional servers, 6-111
- how does it function, 6-58
- retrieving data from, 6-64
- writing to, 6-62
- User Information form (P0092), 3-18, 7-3, 7-11
- User interface logic, 6-106
- User responsibilities, 8-5
- User Signon List Revisions form (P0094), 3-17
- User space, 6-49
 - accessing, 6-112
 - advantages, 6-49
 - control fields, 6-116
 - creating, 6-50, 6-111
 - functional servers, 6-110
 - how it functions, 6-50
 - reading from, 6-54
 - writing to, 6-53
- Using *ALL, 4-57
- Using *BOTH, 4-56
- Using debugger with an interactive program, 6-123
- Using IBM pass-through with group jobs, 7-15

V

- Validate
 - promotion path, 3-99, 3-106
- Validating a promotion path, 3-99
- Variable length records
 - tracking information, 6-54
- Version
 - defined, 8-16
- Version control, 3-1
- Versions List form (P98300), 8-9
- Versions Setup form (P98300), 7-15
- Video
 - compiling, 4-60
- Video fields
 - adding, 4-64
 - record formats list, 4-64

W

- W.O. Detail form
 - accessing, 3-26
- What are calculation specifications?, 5-6
- What are control specifications?, 5-4
- What are extension specifications?, 5-5

- What are file description specifications?, 5-4
- What are functional servers?, 6-101
- What are input specifications?, 5-5
- What are output specifications?, 5-7
- What are the advantages of using a file server?, 6-88
- What are the advantages of using a functional server?, 6-102
- What are the advantages of using a user index?, 6-58
- What are the advantages of using a user space?, 6-49
- What are the disadvantages of using a file server?, 6-88
- What are the disadvantages of using a functional server?, 6-103
- What are the report design standards, 4-88
- What is a file server?, 6-87
- What is a user space?, 6-49
- Wide Screen
 - defined, 4-51
- Window
 - defined, 4-51
- Windows, 3-64
- Work fields, 6-7
 - option files, 6-8
- Work with non-JD Edwards World group jobs, 7-7
- Workbench
 - optional files, 3-62
- Working with data field descriptions, 4-27
- Working with data item alias revisions, 4-22
- Working with data modeling, 4-3
- Working with file conversion, 8-25
- Working with next number by company and fiscal year, 4-29
- Working with object cross reference repository, 4-9
- Working with promotion paths and projects, 3-85
- Working with screen design aid, 4-47
- Working with Software Versions Repository, 3-43
- Working with the crossover rules form, 8-15
- Working with the data dictionary, 4-16
- Working with the data dictionary glossary, 4-22
- Working with the data dictionary glossary by file, 8-33
- Working with the next numbers facility, 4-27
- Working with user defined help instructions, 4-26
- Writing to a user space, 6-53