

Oracle® Documaker

Documaker Installation Guide

version 12.2

Part number: E41180-01

August 2013

Copyright © 2009, 2013, Oracle and/or its affiliates. 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 USA, Inc., 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, and PeopleSoft are registered trademarks of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

CONTENTS

- xiii Audience
- xiv Documentation Accessibility
 - xiv Accessibility of Links to External Web Sites in Documentation
 - xiv TTY Access to Oracle Support Services
- xiv Related Documents
- xiv Conventions

Chapter 1: Installing Documaker Server on z/OS

2 Overview

- 2 Downloading the Software
- 2 Installing the System
 - 2 Configuring Your Printer Types
 - 3 Configuring Host Archive
 - 3 Checking Your Installation
 - 3 Additional Information

4 Downloading the Software

5 Installing the System

- 6 Binding the Documaker DB2 Plan to Your DB2 Subsystem
- 7 DB2 Bind Considerations

9 Upgrading Your System

- 9 Determining your System's Patch Level

10 Customizing Your System

- 10 Running the COMPSDK Job
- 11 Running the GENDATAL Job
- 11 Running the GENPRTL Job

12 Configuring Your Printer Type

- 12 Creating Xerox Metacode Print Streams
- 12 Changing the Xerox JSL
- 14 Changing the FSISYS INI File
- 15 Pre-compiling FAP Files into Metacode Files

	17 Making Printer Resources Available
	18 Changing the JCL
18	Creating AFP Print Streams
	18 Allocating AFP-related Datasets
	19 Making Printer Resources Available
	20 Pre-compiling FAP Files into AFP Overlays
	20 Changing the FSISYS INI File
	20 Changing the JCL
21	Creating PostScript Print Streams
	21 Changing the FSISYS INI File
	23 Changing the JCL
	23 Printing PostScript Print Streams
24	Creating PDF Files
	24 Changing the FSISYS INI File
	27 Changing the JCL
	27 Creating the PDF Print Stream
28	Configuring Host Archive
29	Configuring Host Archive Using DB2
	29 Creating DB2 Objects
	29 Running the BINDDB2 (or COMPDB2) Jobs
	30 Modifying the FSISYS INI File for Host Archive
	33 Changing the JCL for Host Archive
34	Archiving to Documange
	34 Modifying the FSISYS File
	35 Adding IP Addresses and System/Domain Names
	35 Changing the JCL
36	Archiving Using Stacked DPA
	36 Changing the JCL
37	Checking Your Installation
37	Checking Your FSIUSER and FSISYS Settings
40	Checking the JCL
41	Running DAPRUN (or GenTrnX, GenDataX, and GenPrtX) and DMRUNLM
	41 Running DAPRUNC to Improve Performance

45	Using the DMS1 Resources
46	Checking Your FSIUSERL Settings
48	Checking the JCL
49	Running GenArcDX
51	Running GenArcX
52	Printing the Output
53	The LOADDAPW Job
58	z/OS Libraries
62	z/OS Programs and Utilities
68	Using the Documentation
68	Updating Your Documentation
69	Downloading Patches

Chapter 2: Installing Documaker Server on UNIX

72	Overview
72	Downloading the Software
72	Installing the System
72	Checking Your Installation
72	Using the Documentation
72	Start-Up Tips
72	Downloading Patches
73	Downloading the Software
74	Installing the System
76	Checking Your Installation
76	Testing with the DMS1 Sample Resources
77	Processing the DMS1 Sample Resources
77	Printing the DMS1 Output
77	Testing WIP with DMS1
77	Viewing Archived DMS1 Forms
78	Using the Documentation
78	Updating Your Documentation

79 Start-Up Tips

80 Downloading Patches

Chapter 3: Installing Documaker on Windows

82 Overview

82 Downloading the Software

82 Installing the System

82 Checking Your Installation

82 Using the Documentation

82 Downloading Patches

83 Downloading the Software

84 Installing the System

85 Installation

86 Installing the Documaker Add-In for Word

88 If the Documaker Add-In does not Appear in Word

89 Installing the IStream Migration Utility

90 Checking for .NET Programmability Support

91 Checking Your Installation

91 Testing with the DMS1 Sample Resources

91 Processing the DMS1 Sample Resources

92 Printing the DMS1 Output

92 Testing WIP with DMS1

93 Viewing Archived DMS1 Forms

94 Using the Documentation

94 Updating Your Documentation

95 Downloading Patches

Preface

Oracle Documaker is a powerful, adaptive enterprise document automation platform used by insurance, financial and other institutions worldwide to acquire, create, manage, and present structured, on-demand, and interactive customer communications.

It is designed to put power in the hands of users, giving them the flexibility to create effective document templates quickly and with ease. Spanning the entire business lifecycle, Oracle Documaker helps financial institutions manage customer communications enterprise-wide.

Oracle Documaker offers a cost-effective way to address the design, production, and multi-channel distribution of a broad spectrum of customer facing documents — including statements, billing, correspondence, and marketing campaigns that promote new products and/or cross-sell existing products and services. With robust functionality and cutting-edge technical capabilities, it maximizes efficiencies, ensures compliance, and enhances your ability to serve customers.

This guide provides instructions for installing Oracle Documaker Standard Edition products on computers running z/OS (OS/390), UNIX/Linux, and 32-bit Windows, such as Windows XP and Windows Vista.

This table will help you find the information you need:

To install Documaker on	See
z/OS	<i>Installing Documaker Server on z/OS on page 1</i>
UNIX/Linux	<i>Installing Documaker Server on UNIX on page 71</i>
Windows	<i>Installing Documaker on Windows on page 81</i>

For information on installing Oracle Documaker Enterprise Edition, see the [Oracle Documaker Enterprise Edition Installation Guide](#).

For a summary of the Documaker Standard Edition system requirements, refer to the [Documaker System Requirements Guide](#).

AUDIENCE

This document is designed for system supervisors and is intended to help you install and set up Oracle Documaker Standard Edition.

DOCUMENTATION ACCESSIBILITY

Accessibility of Links to External Web Sites in Documentation

This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

TTY Access to Oracle Support Services

Oracle provides dedicated Text Telephone (TTY) access to Oracle Support Services within the United States of America 24 hours a day, seven days a week. For TTY support, call 800.446.2398.

RELATED DOCUMENTS

For more information, refer to the following Oracle resources:

- Documaker Studio User Guide
- Documaker Administration Guide
- Printers Reference
- Fonts Reference
- Rules Reference
- DAL Reference
- Utilities Reference

CONVENTIONS

The following text conventions are used in this document:

Convention	Description
bold	Indicates information you enter.
<i>italic</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
<code>monospace</code>	Monospace type indicates commands, URLs, code in examples, and text that appears on the screen.

Chapter 1

Installing Documaker Server on z/OS

This chapter provides information on how to install and configure Documaker on a z/OS system.

Note Following MVS version 5.2.2, new versions of this operating system were named OS/390. Following OS/390 version 2.10, new versions were named z/OS. Documaker Server runs on MVS, OS/390, and z/OS. In this manual, MVS, OS/390, and z/OS are referred to as *z/OS* unless otherwise noted.

You will find information on these topics:

- *Overview* on page 2
- *Downloading the Software* on page 4
- *Installing the System* on page 5
- *Upgrading Your System* on page 9
- *Customizing Your System* on page 10
- *Configuring Your Printer Type* on page 12
- *Configuring Host Archive* on page 28
- *Checking Your Installation* on page 37

In addition, you will also find information about...

- *The LOADDAPW Job* on page 53
- *z/OS Libraries* on page 58
- *z/OS Programs and Utilities* on page 62
- *Using the Documentation* on page 68
- *Downloading Patches* on page 69

OVERVIEW

This chapter provides you with steps to install and configure Documaker Server on your z/OS system. This overview discusses the contents of this chapter.

Note See *System Requirements* on page 2 for information about system requirements.

DOWNLOADING THE SOFTWARE

You can download Documaker Server for z/OS by downloading the Oracle Documaker Standard Edition media pack from the Oracle Software Delivery Cloud web site. A media pack is an electronic version of the software products available to Oracle customers. The media pack contains ZIP files for a suite of related products on a single operating system so you can build a complete solution for your business. For more information, see *Downloading the Software* on page 4.

INSTALLING THE SYSTEM

Installing Documaker for z/OS, downloaded from the Oracle Software Delivery Cloud web site, involves these steps:

1. Unzip the ZIP file on Windows and upload the unzipped file to z/OS.
2. Run the LOADDAPW job to perform a RECEIVE of the contents of the file that was uploaded.
3. Run the BINDDB2 job to BIND the DB2 Plan to the DB2 subsystem.

Once you have installed the Documaker software, perform the steps discussed in *Binding the Documaker DB2 Plan to Your DB2 Subsystem* on page 6.

Configuring Your Printer Types

The system produces output for Xerox Metacode, IBM AFP, PostScript, PDF, or compatible printers. The steps differ, depending on the printer you will use.

Configuring your system for Xerox Metacode

Here is an overview of the steps required to configure the system for Xerox Metacode or compatible printers. For detailed instructions, see *Creating Xerox Metacode Print Streams* on page 12.

1. Change the Xerox JSL.
2. Change the FSISYS file.
3. Pre-compile your FAP files. The sections that comprise your forms are called FAP files.
4. Make your printer resources available. This can include uploading Xerox fonts, loading the fonts onto your printer, and converting graphics files.
5. Change the JCL for the DAP procedure (or the GenPrtX job) and the GENERXER job.

Configuring your system for AFP printers

Here is an overview of the steps for configuring the system for AFP or compatible printers. For detailed instructions, see *Creating AFP Print Streams* on page 18.

1. Allocate AFP-related datasets.
2. Make your printer resources available. This can include uploading AFP fonts, loading page segments, setting up FormDefs, and creating overlays.
3. Pre-compile your FAP files into AFP overlays.
4. Modify the FSISYS file.
5. Modify the JCL for the DAP procedure (or the GenPrtX job) and the GENERAFP job.

Configuring your system for PostScript printers

Here is an overview of the steps required to configure the system for PostScript printers. For detailed instructions, see *Creating PostScript Print Streams* on page 21.

1. Change the FSISYS INI file.
2. Modify the JCL.

Configuring your system to produce PDF files

Here is an overview of the steps required to configure the system to produce PDF files. For detailed instructions, see *Creating PDF Files* on page 24.

1. Change the FSISYS INI file.
2. Modify the JCL.

Configuring Host Archive

This topic discusses the steps required to configure host archive. This is an optional step. For detailed instructions, see *Configuring Host Archive* on page 28.

Checking Your Installation

Here is an overview of the steps required to check your installation. This is an optional step. For detailed instructions, see *Checking Your Installation* on page 37.

1. Check your FSIUSER and FSISYS file settings.
2. Check your JCL.
3. Run the DAPRUN procedure (or GenTrnX, GenDataX, and GenPrtX).
4. Run GenArcDX (or GenArcX).
5. Print the output.

Additional Information

At the end of this guide you will also find information about the LOADDAPW job, z/OS libraries, and z/OS programs and utilities.

DOWNLOADING THE SOFTWARE

Oracle Insurance applications are available for download at the Oracle Software Delivery Cloud web site. The process includes:

- Logging in and agreeing to the terms and restrictions
- Searching for the applications you want to download
- Downloading those applications

Go to the Oracle Software Delivery Cloud web site to download Oracle Insurance applications:

<https://edelivery.oracle.com>

INSTALLING THE SYSTEM

Once you download and unzip the files, you will see a file named *DAP121Pxx.xmit*. This file contains several z/OS PDSs and sequential datasets and is in a TRANSMIT (XMIT) format.

Note The object and executable modules were built using IBM's z/OS XL C compiler and corresponding Language Environment (LE) product.

To perform the Documaker z/OS installation, you must transfer this file to z/OS in binary mode, then unzip (RECEIVE) it. The RECEIVE creates and populates the PDSs and sequential datasets that comprise the Documaker installation. Follow these steps to transfer this file to z/OS and perform the RECEIVE.

1. Allocate a dataset on z/OS to which you can upload the DAP121Pxx.xmit file. Allocate this dataset using attributes similar to these:

```
Data Set Name . . . : FSI.DAP.XMIT    <= whichever name you choose
Organization . . . : PS
Record format . . . : FB
Record length . . . : 80
Block size . . . . : 3120
1st extent cylinders: 300              <= primary space needed
Secondary cylinders : 10              <= secondary space needed
```

2. Upload the DAP121Pxx.xmit file from Windows to the file on z/OS you allocated in the previous step. Be sure to upload this file to z/OS in binary mode.
3. Upload the LOADDAPW.JCL file (in text mode) to your JCL library on z/OS. Read the comments in the JCL and make the necessary modifications so it will run on your system.

Note If you have previously installed Documaker z/OS, you may already have the LOADDAPW job in your JCLLIB PDS on z/OS. For more information, see *The LOADDAPW Job* on page 53.

4. Submit the LOADDAPW job. The LOADDAPW job consists of two steps. Each step executes the IKJEFT01 (batch TSO) program and invokes the RECEIVE command.

The first step receives the FSI.DAP.XMIT file into a PDS named *FSI.V121.DAP.XMIT.PDS* (or the name you chose). This PDS contains a member for each PDS or sequential dataset that needs to be installed.

The second step receives each of the members of this PDS and creates a PDS or sequential dataset for each of these members.

Once the LOADDAPW job finishes, it will have created and populated 38 or so Documaker installation datasets.

Now go to *Binding the Documaker DB2 Plan to Your DB2 Subsystem* on page 6.

BINDING THE DOCUMAKER DB2 PLAN TO YOUR DB2 SUBSYSTEM

Perform the steps in this topic after you have installed Oracle Documaker for z/OS. If you are running Documaker on z/OS, in most cases your resources (BDF, GRP, FOR, DAL resources, and so on) are stored in a library. On z/OS, this library is comprised of four DB2 tables. You must perform a DB2 BIND to run Documaker on z/OS and access these tables.

The BIND process creates a DB2 PLAN which the Documaker batch processes, such as GenTrn, GenData, and GenPrint use to access the library tables. Note that the GenArc and GenWIP batch processes also require the use of the DB2 PLAN if you use DB2 to store the archive or WIP data.

The job to perform the DB2 BIND is called *BINDDB2* and is provided in the JCLLIB. This job uses the DBRMLIB dataset created during the installation of Documaker for z/OS. The BINDDB2 job performs these tasks:

- BINDs the DB2LIB member of the DBRMLIB into a COLLECTION named DAPxxx_COL.
- BINDs the DB2LIB2 member of the DBRMLIB into a COLLECTION named DAPxxx_COL.
- BINDs the DAPxxx PLAN, consisting of the PACKAGE List of all PACKAGES that are in the COLLECTION named DAPxxx_COL.

You can assign any name that meets your DB2 requirements to the COLLECTION and the PLAN. The PLAN name must match the PLAN name used when DB2 is invoked in the SYSTSIN DD statement in the GenTrn, GenData, GenPrint, GenArc, and GenWIP batch processes.

For example, the sample job DMRUNLM (Documaker Run using Library Manager) in JCLLIB invokes the DMLM Proc in PROCLIB. The DMRUNLM job contains a JCL override for the SYSTSIN DD statement. The content of this overriding SYSTSIN DD statement invokes the DB2 subsystem named DB8G, then runs the Documaker batch program named *GenTrn* and specifies that the DB2 PLAN named *DAP121* is to be used. The GenTrn program is stored in the library named *FSI.V121.LINKLIB*.

```
//DMRUNLM EXEC DMLM
//GENTRN.SYSTSIN DD *
DSN SYSTEM(DB8G)
RUN PROGRAM(GENTRN) PLAN(DAP121) -
LIB('FSI.V121.LINKLIB')
END
/*
```

Modify the BINDDB2 job to conform to your system requirements, then submit this job.

DB2 Bind Considerations

When you migrate to a new version or patch level of Documaker, you typically need to run both the old and new versions for a period of time. Because of the connection between the Documaker executables (such as GenTrn, GenData, and GenPrint) and the DB2 PLAN they use, you may not be able to simply change the batch job to use the appropriate LINKLIB executable (the new or old version). Instead, you may need to generate a new DB2 PLAN for the new version of the Documaker executables to use.

For example, assume you have Documaker version 12.0, patch 03 installed. When you installed this version of Documaker, you ran the BINDDB2 job and created a PACKAGE LIST named DAP120_COL and a PLAN named DAP120. You now need to install Documaker version 12.0, patch 05.

Using the names from the example above, here are several ways to set up two versions of Documaker on z/OS:

Scenario 1 - Using DBRM versioning

In Documaker 12.0, patch 02 and higher, the DB2LIB and DB2LIB2 DBRMs are versioned. In this case, you can simply run the BINDDB2 job to BIND the new DB2LIB and DB2LIB2 DBRMLIB members (from version 12.0, patch 05) into the same PACKAGE LIST named DAP120_COL.

The PLAN named DAP120 will then use the updated PACKAGE LIST. You should still point the LIB parameter in the SYSTSIN DD statement to the LINKLIB of the Documaker version that you want to run with. You can use this *versioning* method if either your current Documaker installation (such as 12.0 P03) or the new Documaker installation (such as 12.0, patch 05) has package versioning.

Scenario 2 - Creating a new PACKAGE and binding the PLAN with the PACKAGE

In the BINDDB2 job, BIND the new DB2LIB and DB2LIB2 DBRMLIB members (from version 12.0, patch 05) into a new PACKAGE LIST named *DAP120P05_COL*. For instance...

```
//SYSTSIN DD *
DSN SYSTEM(DB8G)
BIND PACKAGE(DAP120P05_COL) MEMBER(DB2LIB) ACT(REP) -
ISOLATION(CS)
END
/*
//SYSTSIN DD *
DSN SYSTEM(DB8G)
BIND PACKAGE(DAP120P05_COL) MEMBER(DB2LIB2) ACT(REP) -
ISOLATION(CS)
END
/*
```

BIND the current PLAN (DAP120) and include both the old PACKAGE LIST (DAP120_COL) and the new PACKAGE LIST (DAP120P05_COL).

```
//SYSTSIN DD *
DSN SYSTEM(DB8G)
BIND PLAN(DAP120) ACT(REP) -
ISOLATION(CS) RETAIN -
PKLIST(DAP120_COL.*,DAP120P05_COL.*)
END
```

```
/*
```

If you choose this approach, be sure to include this INI option when running the new version (12.0, patch 05):

```
< DBHandler:DB2 >  
  CurrentPackageSet = DAP120P05_COL
```

The CurrentPackageSet option tells DB2 to use only the PACKAGES in the DAP120 PLAN that are associated with the PACKAGE LIST named DAP120P05_COL. These PACKAGES are the new ones you bound in step 1.

Scenario 3 - Creating a new PACKAGE and a new PLAN

As in scenario 2, in the BINDDB2 job, BIND the new DB2LIB and DB2LIB2 DBRMLIB members (from version 12.0, patch 05) into a new PACKAGE LIST named DAP120P05_COL. For instance...

```
//SYSTSIN DD *  
DSN SYSTEM(DB8G)  
BIND PACKAGE(DAP120P05_COL) MEMBER(DB2LIB) ACT(REP) -  
ISOLATION(CS)  
END  
/*  
  
//SYSTSIN DD *  
DSN SYSTEM(DB8G)  
BIND PACKAGE(DAP120P05_COL) MEMBER(DB2LIB2) ACT(REP) -  
ISOLATION(CS)  
END  
/*
```

BIND a new PLAN (D120P05) and include only the new PACKAGE LIST (DAP120P05_COL).

```
//SYSTSIN DD *  
DSN SYSTEM(DB8G)  
BIND PLAN(D120P05) ACT(REP) -  
ISOLATION(CS) RETAIN -  
PKLIST(DAP120P05_COL.*)  
END  
/*
```

If you choose this approach, you must also change the SYSTSIN DD statement input to point to the new PLAN name, D120P05. Here is an example:

```
//GENTRN.SYSTSIN DD *  
DSN SYSTEM(DB8G)  
RUN PROGRAM(GENTRN) PLAN(D120P05) -  
LIB('FSI.V120.P05.LINKLIB')  
END  
/*
```

Note Regardless of which approach you choose, you must point the LIB parameter in the SYSTSIN DD statement to the new LINKLIB that contains the new version of Documaker (GenData, GenPrint, GenArc, and so on).

UPGRADING YOUR SYSTEM

Upgrading to a new version of Documaker or applying patches to your Documaker system requires that you perform a standard Documaker installation.

DETERMINING YOUR SYSTEM'S PATCH LEVEL

You can determine the patch level of your Documaker system by running a Job provided in the FSIVERX member of JCLLIB. This Job (see below) runs the FSIVER utility which examines the contents of a *LINKLIB* dataset and looks for patch identifiers contained in the Documaker program modules in the LINKLIB dataset.

The FSIVER utility then produces a report listing the patches it found. Unless instructed otherwise, FSIVER produces both a detailed patch report and a summary patch report. See the Utilities Reference for more information on the FSIVER utility.

```
//ZDA      JOB      (33005), 'FSIVER - 121 ', CLASS=T, MSGCLASS=X,
//          NOTIFY=&SYSUID
// *
//          SET HLQ='FSI.V121'    <== SET HIGH LEVEL QUALIFIER
//          SET RES='RPEX1'       <== SET RESOURCE (E.G. RPEX1, UTEX1)
// *
//          JCLLIB  ORDER=&HLQ..PROCLIB
// *
// *****
// * PROGRAM   : FSIVER
// * PURPOSE   : CREATES A REPORT THAT LISTS WHICH PATCHES HAVE BEEN
// *              APPLIED TO THE PROGRAMS IN THE LINKLIB REFERENCED BY
// *              THE LINKLIB DD STATEMENT.
// *
// * PARMs      : /I=PROGRAM (NAME OF MEMBER IN DD:LINKLIB)
// *              OR '*' TO LIST PATCH LEVEL OF ALL PROGRAMS IN
// *              DD:LINKLIB.
// *
// *****
// FSIVER EXEC PGM=FSIVER, PARM='/ / I=*'
// *
// STEPLIB DD DSN=&HLQ..LINKLIB, DISP=SHR
//          DD DSN=SYS1.SCEERUN, DISP=SHR
// LINKLIB DD DSN=&HLQ..LINKLIB, DISP=SHR
// SYSPRINT DD SYSOUT=*
```

CUSTOMIZING YOUR SYSTEM

This step is optional.

Documaker is designed as a rules-based engine. Many rules are provided in the base product. These rules are located in the various source modules that make up a segment of the product called RULLIB. The Documaker architecture also lets you add your own rules, called *custom* rules. Some custom rules are provided, as examples, in a segment of the product called CUSLIB. If you want to modify these custom rules or add custom rules of your own, you should make your changes or additions to the CUSxxxxx modules that are provided in the Documaker Software Developer's Kit (SDK).

After making changes to any source modules in the SDK, you will need to compile these source modules into object modules and then link a new GenData program. The GenData program is also known as the Rules Processor.

Running the COMPSDK Job

To compile the modules provided in the Documaker SDK, modify and submit the job in COMPSDK. The COMPSDK job invokes the CBC compile proc for approximately 60 source modules. Each source module is compiled into an object module, and the object modules are written to the OBJLIB. The COMPSDK job should receive a return code of 04 or less in each step. A portion of the COMPSDK job is shown below.

```
//*** COPY JOBCARD HERE...  
//*  
//* COMPSDK - COMPILE JCL FOR THE DOCUMAKER SOFTWARE DEVELOPER KIT  
//*          (SDK), WHICH IS A SUBSET OF THE COMPLETE DOCUMAKER  
//*          SOURCE.  
//*  
//CUSARC    EXEC CBC, MEM=CUSARC  
//CUSBANNR  EXEC CBC, MEM=CUSBANNR  
//CUSBAT    EXEC CBC, MEM=CUSBAT  
//CUSBITMP  EXEC CBC, MEM=CUSBITMP  
//CUSCALLB  EXEC CBC, MEM=CUSCALLB  
... 
```

Running the GENDATAL Job

You can link-edit a new GenData program to incorporate any changes or additions you have made to the CUSxxxxx source modules. Run the job in JCLLIB member GENDATAL to link-edit GenData. The GENDATAL job first invokes the OBJUTIL proc to rebuild the object library directory, then it invokes the LINK proc to link-edit GenData. GENDATAL should receive a return code of 04 or less in each step. The GENDATAL job is shown below.

```
//* COPY JOBCARD HERE
//* * * * *
//*
//*          LINK-EDIT GENDATA
//*
//* * * * *
//*
//ALIAS      EXEC OBJUTIL
//*
//GENDATA    EXEC LINK, MEM=' GENDATA'
```

Running the GENPRTL Job

You can link-edit a new GenPrint program to incorporate any changes or additions you have made to the CUSxxxxx source modules. Run the job in JCLLIB member GENPRTL to link-edit GenPrint. The GENPRTL job first invokes the OBJUTIL proc to rebuild the object library directory, then it invokes the LINK proc to link-edit GenPrint. GENPRTL should receive a return code of 04 or less in each step. The GENPRTL job is shown below.

```
//* COPY JOBCARD HERE
//
*****
//*
//*          LINK-EDIT GENPRINT
//*
//
*****
//*
//ALIAS      EXEC OBJUTIL
//*
//GENPRINT    EXEC LINK, MEM=GENPRINT
```

Note If you purchased a license for PDF or PDF417 or both, the object modules you received include these features and will enable the resulting GenData and GenPrint to have the capabilities you licensed.

CONFIGURING YOUR PRINTER TYPE

On z/OS, you can use the system to create print streams for these printing environments:

- Xerox Metacode (see *Creating Xerox Metacode Print Streams* below)
- AFP (see *Creating AFP Print Streams* on page 18)
- PostScript (see *Creating PostScript Print Streams* on page 21)
- PDF (see *Creating PDF Files* on page 24)

The following topics tell you how to configure your system for these environments.

CREATING XEROX METACODE PRINT STREAMS

Configuring your system to produce Xerox Metacode print streams involves these tasks:

- Making changes to the Xerox JSL
- Making changes to the FSISYS INI file
- Pre-compiling your FAP files into MET files (this task is optional)
- Making printer resources available
- Making modifications to the JCL

Changing the Xerox JSL

Configuring the system to print to a Metacode printer involves setting various initialization file settings, specifically those in the PrtType:XER control group in the FSISYS initialization file, which is usually a member of the DEFLIB PDS.

To know what values need to be used for each setting, look at the Xerox printer's configuration file. Xerox configuration settings are kept on the printer in a JSL file. The JSL file contains the initial settings under which the printer operates.

You can have multiple JSL files resident on a single printer, with the desired JSL file specified when you start the printer. It is not necessary to start the printer with a specific JDE,JSL combination to print Metacode streams. You must, however, have a JDE,JSL combination which can be dynamically switched to for the duration of that print stream that satisfies the requirements explained below.

Specific values from the JDE,JSL combination must be available to the GenPrint program. You make these values available by placing them in the FSISYS file.

Viewing the JSL

Before you look at any JSL files, first determine which JSL file will be active when print files are sent to the printer. The JSL file name can contain up to six characters and always has a JSL extension. Most Xerox Metacode printers (except the 4235) contain a built-in console editor for creating, viewing, modifying, and printing JSL files.

Note You may want to print your Xerox printer's current JSL file before you continue.

JSL requirements

To print a Metacode stream on your Xerox printer, you must have a Job Description Entry (JDE) with the parameter *VOLUME CODE=NONE*. If your JSL's default JDE entry or any other JDE entry in the JSL contains this parameter, you do not need to add another JDE entry. If the parameter is missing, you must add it. Follow your accepted procedures for making such JSL additions and modifications.

Once there is a JDE entry (or the default entry) that satisfies the *VOLUME CODE* requirement, enter values in the table below that correspond either to this JDE entry specifically or to the JSL file in general.

Use the table below to note the values in the Xerox JSL you will use. You will enter these values when you modify the FSISYS file.

JSL entry	Description	Values
DJDEIden	Corresponds to the IDEN statement's PREFIX= keyword (or PRE=). Indicates E for EBCDIC, A for ASCII, X for hexadecimal and a string value. If there is no E,A or X before the string value, an <i>E</i> is assumed by default.	
DJDEOffset	Corresponds to the IDEN statement's OFF= keyword.	
DJDESkip	Corresponds to the IDEN statement's SKIP= keyword.	
ImageOpt	If the JSL contains a "GRAPHICS = YES" statement, set ImageOpt to Yes. Otherwise, set it to No. You must set this option to Yes if you need the system to dynamically draw charts.	
JDEName	If you had to create a new JDE entry to specify <i>VOLUME CODE = NONE</i> , use the name of the new JDE entry. If a JDE entry already existed that contained <i>VOLUME CODE = NONE</i> , specify the name of that JDE (if it's the default, it's probably called <i>DFLT</i>).	
JDLName	The name of the JSL file.	
JDLHost	Corresponds to the VOLUME parameter's HOST= keyword. This value will be either IBMONL or IBMOS. If not specified, the default is <i>IBMOS</i> .	
JDLCode	Corresponds to the VOLUME statement's CODE= keyword for the initial JSL setting (at the top of the JSL file). This value will be either EBCDIC, ASCII, or NONE.	
JDLData	Corresponds to the LINE statement's DATA= keyword. Should be two numbers separated by a comma (omit the parentheses).	

JSL entry	Description	Values
PrinterInk	If the printer supports a highlight color, specify that color here. Otherwise, leave blank.	
JDLRStack	To find this value, find the RSTACK command (if there is one) and work backwards. The RSTACK command refers to a TEST label, which sets up a string test and refers to a CONSTANT, string, or value. For example... RSTACK: TEST=RST,DELIMITER=YES; RST: CRITERIA CONSTANT =(0,10,EQ,RSTK); RSTK: TABLE CONSTANT =X'13131313131313131313'; ...the value you would use would be: JDLRStack=0,10,EQ,X'13131313131313131313'	
JDLROffset	Locate the ROFFSET command (if there is one) and work backwards. See JDLRStack above.	
JDLRPage	Locate the RPAGE command (if there is one) and work backwards. See JDLRStack above.	

Changing the FSISYS INI File

Note Two sets of resources (MRLs) are included in the installation. The RPEX1 set of resources is designed for the insurance market. The UTEX1 resources are designed for the utility market. You can verify your installation by running the system using one or both of these sets of resources.

The following instructions may ask you to modify the FSISYS initialization member of DEFLIB, so the DEFLIB you must modify is the one for the set of resources you plan to use, such as *hlq.v103.RPEX1.DEFLIB* or *hlq.v103.UTEX1.DEFLIB*, or both if you plan to use both.

Metacode data sets on z/OS should be created with a Variable Blocked Machine character (VBM) Record Format (RECFM) and should have a Logical Record Length (LRECL) that is four bytes longer than the length of the longest Metacode record you plan to write.

z/OS uses four bytes of each record in a Variable Blocked data set to store data about that record. For example, in your INI file you may have the JDLDData option set to

```
0,250
```

which indicates the longest Metacode record will be 250 bytes long. If this is the case, your Metacode data set should be allocated with an LRECL of 254.

PrtType:XER control group

You must place the values you gathered from or added to the Xerox printer's JSL file into the PrtType:XER control group in the FSISYS INI file. This file is generally stored as a member called *FSISYS* within the sample DEFLIB PDS.

Below are the options in the PrtType:XER control group relevant to z/OS environments. Using the table you completed earlier, make changes as necessary to the following values. The example below shows the default FSISYS values.

Note You can comment out an INI control group or option by placing a semicolon (;) before the control group or option.

```
< PrtType:XER >
  ColorCharts      = Yes
  CompileInstream  = No
  DownloadFonts    = No, Disabled
  InitFunc         = XERInit
  Module           = XERW32
  OutputFunc       = XEROutput
  OutMetFunc       = XEROutMet
  PageNumbers      = Yes
  PrintFunc        = XERPrint
  PrinterInk       = Blue
  TermFunc         = XERTerm
  SendColor        = Yes
  DJDEIden         = A'@@@DJDE'
  DJDEOffset       = 0
  DJDESkip         = 8
  Environment      = MVS
  OutMode          = JES2
  ImageOpt         = Yes
  JDEName          = DFLT
  JDLData          = 0,255
  JDLHost          = IBMONL
  JDLName          = DFAULT
  JDLCode          = NONE
  JDLRStack        = 0,10,EQ,X'13131313131313131313'
  JDLRPage         = 0,5,EQ,X'FFFF26FFFF'Printer control group
```

The Printer control group tells the GenPrint program what type of output to produce. For the RPEX1 sample resources, the PrtType option is set to *XER*. For the UTEX1 sample resources, the PrtType option is set to *AFP*. Make sure the option is set correctly for your environment. For example, this setting tells the GenPrint program to produce Xerox Metacode print streams:

```
< Printer >
; PrtType = AFP
PrtType = XER
```

Printer1 control group

The Printer1 control group contains PORT options which specify a DD name to which the system writes output for that recipient batch.

```
< Printer1 >
  Port = DD:PRTBAT1
```

There may be other options present but they will be commented out with a semicolon. Make sure the PORT option is set appropriately for your environment.

Pre-compiling FAP Files into Metacode Files

During the GenPrint process, FAP files (sections) are compiled into Metacode files and arranged into the print stream. To speed execution, you can pre-compile—compile before running GenPrint—these FAP files.

Note The FSISYS file included with the sample resources assumes you will be using pre-compiled Metacode files, or *PMETs*, since the use of PMETs increases performance. If, however, you *do not* want to run with PMETs, change the CompileInStream option in the PrtType:XER control group and DownloadFAP INI option in the RunMode control group to *Yes* and skip to the topic *Making Printer Resources Available* on page 17.

You can use the FAP2MET utility to compile FAP files into Metacode files. To run the system using pre-compiled Metacode files you must run the FAP2MET utility for every FAP file in FORMLIB. Here's how:

1. Run the FAP2METS job in JCLLIB. This job deletes or re-allocates the PMETLIB PDS and runs the FAP2MET utility for every FAP file in FORMLIB or FAPLIB. This creates a corresponding *PMET* and stores it in the PMETLIB PDS. The FAP2MET job should receive a return code of 00 for each step.
2. Make sure the CompileInstream option in the PrtType:XER control group is set to *No* and that the DownloadFAP option in the RunMode control group is set to *No*.
3. Modify the JCL for the DAP procedure or the GenPrtX job (please first read the topic *Changing the JCL* on page 18) so the PMETLIB DD statement is uncommented (it is included in the JCL but is commented out). This lets the GenPrint program open and read the PMETs you created.

Making Printer Resources Available

The resources your Xerox Metacode printer needs fall into two categories:

- Fonts
- Logos

The following topics tell you how to make these resources available to your printer.

Fonts

By default, the verification test (see *Checking Your Installation* on page 37) uses the fonts included with the system. The fonts were loaded into this dataset when you ran the LOADFONT job:

```
&HLQ..ORACLE.XEROX.FONTLIB
```

If that is the case, go to the topic, *Loading fonts onto your printer* on page 17.

Loading fonts onto your printer

To load the system fonts onto a Xerox printer, follow these steps:

Note The names of the Xerox fonts begin with the characters *FX*. You may want to list the existing fonts on the printer to see whether any of your existing fonts have the same names as the fonts you are about to send to the printer.

1. Run the XERDNLDS member in JCLLIB. This job uses the FSISYS file you customized earlier. It converts the fonts into a format your Xerox printer can store. XERDNLDS should complete all steps with a return code of 00.
2. Run the GENDNLDS member in JCLLIB. This job runs the IBM utility IEBGENER to copy each font to the Xerox printer. Modify GENDNLDS as necessary to fit your standards. Be sure to set the SYSOUT output class to the JES2 output class for the Xerox printer. GENDNLDS should complete all steps with a return code of 00.

Logos

The verification test references FAP files which contain logos (bitmap-type graphics). You must place these logos on the Xerox printer either as FNT files or as IMG files.

If your Xerox printer has a graphics card, it can print IMG files and you should place the LOGOs on the printer as IMG files. If your printer does not have a graphics card, it cannot print IMG files—place the logos on the printer as FNT files.

When you installed the system, two PDSs were created called *&HLQ..LOG2IMG* and *&HLQ..LOG2XFNT*. These PDSs contain logos which have been converted into IMG and FNT files, respectively. You should download the contents of one of these PDSs to your Xerox printer.

For steps 1 and 2, choose either to load the IMG files (choose DD statement for LOG2IMG) or the FNT files (choose DD statement LOG2XFNT) to your Xerox printer:

1. Run the XERLOGOS job in JCLLIB. Edit the job to read from and write to the appropriate type of file (LOG2IMG or LOG2XFNT). This job should complete with return codes of 00 in all steps.
2. Run the GENLOGOS job in JCLLIB. Edit the job so it will read from the dataset you created in step 1 and so it will send the output to the correct JES sysout class. This makes sure the Xerox resources are sent to the Xerox printer. This job should complete with a completion code of 00 in all steps.

Changing the JCL

DAP procedure (or the GenPrtX job)

In the next topic you will run the system with the sample resources to make sure it has been installed and configured correctly. You must modify the jobs you will run to conform to your local JCL and environment standards. The last job, GenPrint, will reference the DD names you set up in the topic, *Printer1 control group* on page 15.

When you run the system with the sample resources, you can run each step as a separate job (GenTrnX, GenDataX, and GenPrintX) or in sequence together by running a job called DAPRUN. The DAPRUN job invokes the DAP procedure which executes GenTrn, GenData, and GenPrint in this order.

For the verification test, you should use the DAPRUN job and edit the DAP procedure in PROCLIB. If, however, you want to run each step separately, edit the GenPrtX member in JCLLIB.

GENERXER job

You use the GENERXER job in JCLLIB to send the print streams created by the GenPrint program to the JES spool to be printed by your Xerox printer. Modify this job as necessary to match the file names you have used.

CREATING AFP PRINT STREAMS

Configuring your system to produce AFP print streams involves these tasks:

- Allocating AFP-related datasets
- Making printer resources available.
- Pre-compiling FAP files into overlays (this task is optional)
- Making changes to the FSISYS initialization file
- Making changes to the JCL

Allocating AFP-related Datasets

To allocate the datasets required to perform AFP printing, use the ALLOCAFP job in the JCLLIB member. This job allocates the Form Definition Library (FDEFLIB), the Page Segment Library (PSEGLIB), and the Overlay Library (OVERLIB). The job should complete with a return code of 00.

Making Printer Resources Available

The resources your IBM AFP printer needs fall into these categories:

- Fonts
- Page segments
- FormDefs
- Overlays

The following topics tell you how to make these resources available to your printer.

Fonts

By default, the verification test (see *Checking Your Installation* on page 37) uses the fonts included with the system. If you will print using IBM's Advanced Function Presentation (AFP), provided by the IBM Print Services Facility (PSF), you need to use these system fonts. The installation includes 240-dpi and 300-dpi fonts.

Page segments

The system generates bitmap-type graphics into files which, on the PC, have an extension of *.LOG. The verification test uses sections that contain these LOG files. For these graphics files to print correctly, you must convert these files into AFP Page Segments (PSEGs). You can run the LOG2PSEG utility on either the workstation or the z/OS system to perform this task.

To convert graphics files for the verification test, run the LOG2PSGS job in JCLLIB. This job converts several graphics files from the sample FORMLIB or FAPLIB into page segments and places them into a PSEGLIB. The job should complete all steps with a return code of 00.

FormDefs

Next, submit the AFPFMDFX job in JCLLIB to compile a *source* Form Definition File into an AFP-ready FormDef. This job should complete with a return code of 00.

Overlays

To speed performance in the GenPrint process, you can pre-compile FAP files (sections) into AFP overlays before you run the GenPrint program. Follow the directions in the next topic, *Pre-compiling FAP Files into AFP Overlays*, to create the AFP overlays.

Note The FSISYS file included with the sample resources assumes you will be using AFP overlays, since the use of overlays increases performance. If, however, you do not want to run with overlays, change the SendOverlays option to No and the DownloadFAP option to Yes and skip to the topic *Changing the FSISYS INI File* on page 20.

Pre-compiling FAP Files into AFP Overlays

The FAP2OVL utility pre-compiles FAP files into AFP overlays, which can improve performance. To pre-compile FAP files into AFP overlays, run the FAP2OVLS job in JCLLIB. This job converts the FAP files in the sample FAPLIB PDS into overlays and places them into the OVERLIB PDS.

Changing the FSISYS INI File

PrtType:AFP control group

Here are the options in the PrtType:AFP control group in the FSISYS file which apply to z/OS environments. The FSISYS INI file should have been supplied with the value below:

```
< PrtType:AFP >
  SendOverlays=Yes
```

If you chose not to create AFP overlays (in the prior topic above), set the SendOverlays option to *No*. If you created overlays, make sure the option is set to *Yes*.

Printer control group

The Printer control group tells the GenPrint program what type of output to produce. For the RPEX1 sample resources, the PrtType option is set to *XER*. For the UTEX1 sample resources, the PrtType option is set to *AFP*. Make sure the option is set correctly for your environment. For example, this setting tells the GenPrint program to produce AFP print streams:

```
< Printer >
  PrtType = AFP
; PrtType = XER
```

Printer1 control group

The Printer1 control group contains PORT options which specify a DD name to which the system writes output for that recipient batch.

```
< Printer1 >
  Port = DD:PRTBAT1
```

There may be other options present but they will be commented out with a semi-colon. Make sure the PORT option is set appropriately for your environment.

Changing the JCL

Using the DAP procedure (or the GenPrtX job)

In the topic, *Checking Your Installation* on page 37, you will run the system using the sample resources. This will help you make sure your system is installed and configured correctly. Before you run the test, you must modify the jobs you will use during the test to match your local JCL and environment standards. The last job, GenPrint, references the DD names you may have added or uncommented (see *PrtType:AFP control group* on page 20).

When you run the system with the sample resources, you can run each step as a separate job (GenTrnX, GenDataX, and GenPrintX) or in sequence together by running the DAPRUN job. The DAPRUN job invokes the DAP procedure which executes GenTrn, GenData, and GenPrint in this order.

For the verification test, we recommend you use the DAPRUN job. If you choose to use the DAPRUN job, edit the DAP procedure in PROCLIB. If, however, you want to run each step separately, edit the GenPrtX member in JCLLIB.

Note The DAP procedure included in PROCLIB (or the GenPrtX member of JCLLIB) for the UTEX1 resources is set up to produce AFP output file names such as AFPBAT1, AFPBAT2, and so on. You may want to inspect the procedure to make sure these names are correct, but you should not have to change the names.

GENERAFFP job

You will use the GENERAFP job in JCLLIB to send the print streams created by the GenPrint program to the JES spool to be printed by your AFP printer. Modify this job as necessary to match the file names you have used.

CREATING POSTSCRIPT PRINT STREAMS

Configuring the system to produce PostScript print streams involves these tasks:

- Making changes to the FSISYS INI file
- Making changes to the JCL
- Printing the PostScript print stream

Changing the FSISYS INI File

Here are the options in the PrtType:PST control group in the FSISYS INI file that are relevant to z/OS environments:

```
< PrtType:PST >
  Module      = PSTW32
  PageNumbers = Yes
  PrintFunc   = PSTPrint
; PrinterModel = (PPD file)
  SendOverlays = Yes
  SendColor   = Yes
  DownloadFonts = Yes
```

If you are using a PostScript Printer Description (PPD) file, upload your PPD file (as text) to DEFLIB, uncomment the PrinterModel option by removing the semicolon (;), and set it to your PPD file in DEFLIB. For example, if your PPD file in DEFLIB is HP8000, set your option as shown here:

```
PrinterModel = HP8000
```

Setting the SendOverlays option to Yes tells the GenPrint program to use PostScript overlays to speed its process. To create PostScript overlays, first run the OVLCOMP utility to convert your FAP files into PostScript overlays (the OVLCOMP utility only runs on Windows). You then upload (as binary) the PostScript overlays to the dataset referred to by OVERLIB DD in your JCL.

The Printer control group tells the GenPrint program what type of output to produce. For the RPEX1 sample resources, the PrtType option is set to XER. For the UTEX1 sample resources, the PrtType option is set to AFP. Make sure the option is set correctly for your environment. For example, this setting tells the GenPrint program to produce PostScript print streams:

```
< Printer >
;  PrtType                      = AFP
;  PrtType                      = PST
;  PrtType                      = XER
;  PrtType                      = PDF
;  EnableTransBanner            = Yes
;  EnableBatchBanner           = Yes
;  BatchBannerBeginScript       = PreBatch
;  TransBannerBeginScript       = PreTrans
;  BatchBannerEndScript         = PstBatch
;  TransBannerEndScript         = PstTrans
```

The Printer1 control group contains PORT options which specify a DD name to which the system writes output for that recipient batch.

```
< Printer1 >  
    Port = DD:PRTBAT1
```

There may be other options commented out with semicolons (;). Make sure the PORT option is set appropriately for your environment.

Changing the JCL

In the next topic, *Checking Your Installation* on page 37, you run the system with sample resources to make sure it has been installed and configured correctly. You must modify the jobs you will run to conform to your local JCL and environment standards. The last job, GenPrint, references the DD names you set up in the Printer1 control group.

When you run the system with the sample resources, you can run each step as a separate job (GenTrnX, GenDataX, and GenPrtX) or in sequence together by running a job called DAPRUN. The DAPRUN job invokes the DAP procedure which executes GenTrn, GenData, and GenPrint in this order.

For the verification test, use the DAPRUN job and edit the DAP procedure in PROCLIB. If, however, you want to run each step separately, edit the GenPrtX member in JCLLIB.

Printing PostScript Print Streams

To print PostScript print streams to a PostScript-capable printer, download (as binary) the print streams created by the GenPrint program to your Windows environment and do a binary copy (see the command below) of the downloaded file to your PostScript printer.

```
>copy /b your_file your_printer
```

CREATING PDF FILES

Configuring the system to produce PDF files involves these tasks:

- Making changes to the FSISYS INI file
- Making changes to the JCL
- Creating the PDF file

Changing the FSISYS INI File

In the FSISYS INI file, below are the options in the PrtType:PDF control group relevant to z/OS environments. The FSISYS INI file includes the values shown here:

```
< PrtType:PDF >
; Device           = E:\TEST.PDF
Bookmark          = Yes,Page
DownloadFonts     = Yes
LanguageLevel     = Levell
Module            = PDFW32
PageNumbers       = Yes
PrintFunc         = PDFPrint
SendOverlays      = No
SendColor         = Yes
Encrypt           = No
SecurityGroup     = PDFEncryption
; PrintViewOnly    = No
; SplitText        = No
; SplitPercent     = 50
; Class           = PDF
; PrePrintedPaper= (not supported)
; PaperSize       = 0
Compression       = 0
FontCompression   = 0
< PDFEncryption >
KeyLength         = 128
User              = M
Owner             = Oracle
AllowModify       = False
```

To produce secured PDF print streams, set the Encrypt option to Yes. Under the PDFEncryption control group, set the KeyLength, User, and Owner options as appropriate for your implementation.

The Printer control group tells the GenPrint program what type of output to produce. For the RPEX1 sample resources, the PrtType option is set to XER. For the UTEX1 sample resources, the PrtType option is set to AFP. Make sure the option is set correctly for your environment.

Generating PDF files using DAL scripts

For example, these options and settings tell the GenPrint program to produce a PDF file for each transaction in your print stream:

```
< Printer >
; PrtType           = AFP
; PrtType           = PST
; PrtType           = XER
; PrtType           = PDF
; EnableTransBanner = Yes
; EnableBatchBanner = Yes
; BatchBannerBeginScript = PreBatch
; TransBannerBeginScript = PreTrans
; BatchBannerEndScript = PstBatch
; TransBannerEndScript = PstTrans
```

Note that these INI settings require the use of a BANNER DAL library and a WRITE DAL script. The BANNER DAL library contains the PreBatch, PreTrans, PstBatch, and PstTrans DAL scripts specified in the INI file. The WRITE DAL script is called from PreTrans.

These DAL scripts split a recipient batch PDF file into separate PDF files for each transaction in the batch. You can see examples of the BANNER DAL library and the WRITE DAL script in the DEFLIB of RPEX1 sample resources.

You also need to add this option:

```
< DALLibraries >
LIB = Banner
```

Generating PDF files using the MultiFilePrint rule

These options and settings show another way to produce a PDF file for each transaction in your print stream:

```
< Printer >
; PrtType           = AFP
; PrtType           = PST
; PrtType           = XER
; PrtType           = PDF
; EnableTransBanner = Yes
; EnableBatchBanner = Yes
; BatchBannerBeginScript = PreBatch
; TransBannerBeginScript = PreTrans
; BatchBannerEndScript = PstBatch
; TransBannerEndScript = PstTrans
< Print >
; CallbackFunc       = MultiFilePrint
; MultiFileLog        = DD:PDFLOG
```

These INI settings tell the system to use the MultiFilePrint rule to split a recipient batch PDF file into separate PDF files, one for each transaction in the batch. For more information about the MultiFilePrint rule, see the Rules Reference.

Note The MultiFileLog option is optional. Include it if you want the system to create a log of the PDF file names written to the PDFLOG DD statement in the JCL. You can see an example of the PDFLOG DD statement in the DAP member of PROCLIB.

The Printer1 control group contains Port options which specify the DD name to which the system writes output for that recipient batch. The first transaction in the batch is written to member BAT10000 (or BAT10001 if you are using the MultiFilePrint rule) of PDFLIB PDS. The next transaction is written to member BAT10001 (or BAT10002 if you are using the MultiFilePrint rule), and so on.

The member name can consist of any four characters followed by 0000 (or 0001 if you are using the MultiFilePrint rule) for the first transaction of each batch. You only need to specify this first member name for each recipient batch in the INI file. The BANNER DAL library and the WRITE DAL script generate the subsequent member names and increment the numbers for that particular batch.

```
< Printer1 >
  Port = DD:PDFLIB(BAT10000)
< Printer2 >
  Port = DD:PDFLIB(BAT20000)
```

If there are other options present, comment out those options with semicolons (;). Make sure the Port option is set appropriately for your environment.

Including PostScript and TrueType fonts

To include PostScript fonts or TrueType fonts or both in PDF print streams, include these INI options and settings:

```
< PrtType:PDF >
  DownloadFonts = Yes
< MasterResource >
  FontLib      = DD:FONTLIB()
```

Add a FONTLIB DD statement into your JCL (for the GenPrint step) and point to the FONTLIB PDS that has PostScript fonts and the PDS that has TrueType fonts. The FONTLIB for both PostScript and TrueType can have a record format (RECFM) of FB and logical record length (LRECL) of 128. You can see an example of a FONTLIB DD statement in the DAP member of PROCLIB.

PostScript and TrueType fonts can be uploaded as binary into the FONTLIB PDS. Keep in mind that you should change underscores (_) in the font name to @ symbols when uploading. For example, *ALBB____.PFB* is uploaded as *ALBB@@@@* since z/OS can not have underscore in the member name. The PDF Print Driver produces linearized PDF files. Linearized PDF files support page-at-a-time downloading which makes it faster to view a PDF file over the Internet. The nature of PDF files requires each transaction be in a separate PDF file.

The RunMode control group in the FSISYS.INI file includes these options and values:

```
< RunMode >
  NAUnload      = Yes
  DownloadFAP    = No
  CompiledFAP    = Yes
  ChkDstLenExceeded = No
  LoadCordFAP    = No
; MergedDDTs    = No
```

Be sure to set the DownloadFAP option to Yes when you are producing PDF files.

Understanding how PDF Files are Created

In PDF files, information is added at the beginning of the PDF file that tells Acrobat where every object (file offsets for page, text, bitmap, fonts, and so on) is located inside the PDF file. The only way to gather this information is to have the entire file stored in memory before writing it out.

In Documaker, because of the volume of transactions that may be processed in a batch, only a single transaction is loaded into memory at a time. The print driver is then called. That transaction is processed and then the next transaction is loaded into memory. The PDF Print Driver only has information about the current transaction in memory, not all of the transactions that comprise a batch.

Typically, you use the PDF Print Driver to create a separate PDF file for each transaction in the batch. You can, however, use the SpoolBatches option to generate a single PDF file that includes all of the transactions in a batch. See Using the PDF Print Driver for more information on generating separate PDF files or generating a single PDF file.

Note If you do not generate separate PDF files and you do not set the SpoolBatches option, you can end up with multiple PDF files concatenated into a single file, but the beginning of the file only contains information (file offsets for page, text, bitmap, font, and so on) about the last transaction. Since the last transaction can look a lot like the first transaction, Acrobat may be fooled into displaying the first transaction. It will not, however, display the other transactions in the file.

Changing the JCL

In the topic, *Checking Your Installation* on page 37, you will run the system with sample resources to make sure it has been installed and configured correctly. You must modify the jobs you will run to conform to your local JCL and environment standards. The last job, GenPrint, references the DD names you set up in the Printer1 control group.

When you run the system with the sample resources, you can run each step as a separate job (GenTrnX, GenDataX, and GenPrtX) or in sequence together by running a job called DAPRUN. The DAPRUN job invokes the DAP procedure which executes GenTrn, GenData, and GenPrint in this order.

For the verification test, use the DAPRUN job and edit the DAP procedure in PROCLIB. If, however, you want to run each step separately, edit the GenPrtX member in JCLLIB.

Creating the PDF Print Stream

To print PDF print streams to the printer, download (as binary) the print streams created by the GenPrint program to your Windows environment. Then use Adobe Reader to open the downloaded file and send it to your printer.

CONFIGURING HOST ARCHIVE

The system lets you archive form sets that have been created by the GenData program. Archiving is performed by running the GenArc program and can be configured to archive the form sets in one of several ways:

- To the standard DAP archive. For more information, see *Configuring Host Archive Using DB2* on page 29.
- To Documange. For more information, see *Archiving to Documange* on page 34.
- To Stacked DPA files. For more information, see *Archiving Using Stacked DPA* on page 36.

CONFIGURING HOST ARCHIVE USING DB2

In addition to (or instead of) simply printing form sets, you can optionally archive your form sets and data using the GenArc program. The GenArc program on z/OS systems supports archival to IBM's DB2 database management system.

Configuring a host archive system involves performing these tasks:

- Creating DB2 objects
- Running BINDDB2 or the ALLOCDB2 and COMPDB2 jobs
- Modifying the FSISYS file
- Making JCL changes

Creating DB2 Objects

The GenArc program, as configured for the sample library, uses these DB2 tables:

Table	Description
APPIDX	An application index table which contains a row for every archived form set.
ARCHIVE	An archive table which contains the actual NAFILE and POLFILE data for each archived form set.
CATALOG	A table which stores unique catalog IDs which the system uses as it builds the ARCKEY used in the APPIDX and ARCHIVE tables.
RESTART	A table in which GenArc restart information is stored.

You should create these tables using the ARCSQLR Job for the RPEX1 resources and the ARCSQLU Job for the UTEX1 resources. Modify the Jobs to suit your site's DB2 naming conventions and submit. The job should receive a return code of 00.

Running the BINDDB2 (or COMPDB2) Jobs

If you have a standard license, run the BINDDB2 job in JCLLIB as described below. If you have a source code license, skip to the following topic, *Running ALLOCDB2 and COMPDB2* on page 30.

The BINDDB2 Job performs DB2 binds for the DB2LIB and DB2LIB2 packages and the DAP121 plan. In the BINDDB2 job, change the DB2 subsystem, package names, and plan name to fit your standards. The BINDDB2 job should finish with return codes of 00 in all steps. Proceed to *Modifying the FSISYS INI File for Host Archive* on page 30.

Running ALLOCDB2 and COMPDB2

The COMPDB2 job invokes the DB2 pre-compiler to pre-compile source files and place the resulting DBRMs into a PDS called DBRMLIB. This DBRMLIB dataset should exist, but if it is not available, you can allocate it by running the job ALLOCDB2. The ALLOCDB2 job should finish with a return code of 00.

The GenArc program uses programs called DB2LIB and DB2LIB2 to make all SQL calls. These SQL calls are prepared dynamically and are not static SQL calls.

To prepare the GenArc program for execution, run the job in member COMPDB2 of JCLLIB. The COMPDB2 member uses the DSNHC procedure in PROCLIB. Essentially, the COMPDB2 job will...

- DB2- pre-compile the DB2LIB and DB2LIB2 programs
- Compile the DB2LIB, DB2LIB2, and the GenArc programs
- Bind DB2LIB and DB2LIB2 into packages
- Bind the DB2LIB and DB2LIB2 packages into a plan
- Link-edit GenArc into an executable program, placing it in LINKLIB.

Depending on your DB2 system file names and C compiler file names, you will probably have to modify these procedures to meet your needs. The COMPDB2 job should finish with return codes of 00 or 04 in all steps.

Modifying the FSISYS INI File for Host Archive

Just as the FSIUSER and FSISYS initialization files provide the GenTrn, GenData, and GenPrint programs with configuration information, they also provide configuration information to the GenArc program.

The following control group and option settings must exist in either the FSIUSER or FSISYS members of DEFLIB. The FSISYS file shipped with the base product contains the values shown below. You only need to change these options if you have modified the names of the DB2 objects, such as the database name, table name, and so on.

ArcRet control group

The ArcRet control group contains information about the file (or table) names used for archiving information. The values for table names in this group cannot exceed eight characters. To specify a table name with more than eight characters use the DB2_FileConvert control group to map the eight character name you enter here to a longer (up to 18 characters) name.

```
< ArcRet >
  APPIDXDFD      = APPIDX
  APPIDX         = APPIDX
  CARFile        = ARCHIVE
  CATALOG        = CATALOG
  RestartTable   = RESTART
```

Archival control group

The Archival control group indicates whether archived information is stored via a DBMS. For DAP archive to DB2, specify:

```
< Archival >  
    ArchiveMem = Yes
```

DBHandler:DB2 control group

The DBHandler:DB2 control group tells the GenArc program that DB2 will be used to access the tables used during the GenArc process.

The Database option identifies the name of the DB2 database that contains the tables used by the GenArc program. Database names can consist of up to eight characters.

The CreateTable option indicates whether the GenArc program can create the tables it needs or if the tables must exist before it begins.

Enter Description

No	Do not allow the GenArc program to create the DB2 tables.
Yes	Allow the GenArc program to create the DB2 tables.

The CreateIndex option indicates whether indexes are to be created for the APPIDX DB2 table if the APPIDX (DFD) member of DEFLIB contains keys information.

Enter Description

No	Do not allow the GenArc program to create the DB2 indexes.
Yes	Allow the GenArc program to create the DB2 indexes.

```
< DBHandler:DB2 >  
  CreateTable = No  
  CreateIndex = No
```

DB2_FileConvert control group

The DB2_FileConvert control group lets you use DB2 table names that are longer than eight characters. The option on the left (APPIDX) is the value associated with the corresponding option in the *ArcRet control group* on page 30. The value on the right (DAP121_APP_R1) is the name of the DB2 table as it exists in the DB2 subsystem. If your table names are not longer than eight characters, you do not need to make any entries in the DB2_FileConvert control group.

```
< DB2_FileConvert >  
  APPIDX      = DAP121_APP_R1  
  Archive     = DAP121_ARC_R1  
  Catalog     = DAP121_CAT_R1  
  Restart     = DAP121_RES_R1
```

DBTable control group

Use the DefaultTag option to specify the default tag for ODBC and DB2. This tag is used by the ORDER BY clause in the SQL database to sort records.

```
< DBTable:MYTABLE >  
  DefaultTag =
```

For the DefaultTag option, enter the name of the key from the DFD file. Keep in mind this only works with ODBC and DB2. It does not work with xBase files.

Trigger2Archive control group

The Trigger2Archive control group designates the fields in the NEWTRN DAT file (right side) that are copied to the columns in the APPIDX table (left side) during the GenArc process. Remember that the APPIDX table columns are defined by the APPIDX member of DEFLIB and that this APPIDX member is in a DFD format. For the sample resources, the Trigger2Archive control group might be set up as follows:

```
< Trigger2Archive >
  Company    = Company
  LOB        = Lob
  PolicyNum  = PolicyNum
  RunDate    = RunDate
```

Changing the JCL for Host Archive

Examine the GenArcDX member of JCLLIB. Make sure the JCL is appropriate for your site and that the dataset names correspond to the names you have used for your files. Additionally, make sure the DB2 subsystem name and other DB2 object names, such as the plan names, are correct.

ARCHIVING TO DOCUMANAGE

Configuring the system to archive to Documange involves performing these tasks:

- Setting up Documange (see the Documange manuals for more information)
- Modifying the FSISYS file (FSISYSM member in DEFLIB)
- Adding IP addresses and system/domain names of the Documange machine to the TCPIP host data sets
- Making JCL changes

Modifying the FSISYS File

Note All references of the FSISYS file in this topic refer to FSISYSM member in DEFLIB.

In the ArcRet control group, the FSISYS file includes the values shown here:

```
< ArcRet >
  APPIDXDFD      = AppIdx
  APPIDX         = APPIDX
  CARFile        = ARCHIVE
  CARFileDFD     = CARFILE
  CARPath        =
;  Catalog       = CATALOG
;  RestartTable  = RESTART
```

For archiving to Documange, you need to set the CARFileDFD option to *CARFILE* (which resides in DEFLIB) and comment out both the Catalog and RestartTable options.

The Archival control group in the FSISYS file should have these values:

```
< Archival >
  ArchiveMem      = Yes
  UseRestartTable = No
```

For archiving to Documange, set the UseRestartTable option to No.

In the following DBTable control groups, the FSISYS file should have these values:

```
< DBTable:APPIDX >
  DBHandler = DMIA
< DBTable:ARCHIVE >
  DBHandler = DMIA
```

Note You must use *DMIA* as a DBHandler for archiving to Documange.

The DBHandler:DMIA control group tells the GenArc program that DMIA is used to access Documanage during the GenArc process. Your FSISYS file includes these values:

```
< DBHandler:DMIA >
  RPCHost    = (IP address of the Documanage machine)
  Cabinet    = RPEX1OS390
  Domain     = (system/domain name)
  UserID     = (Documanage user ID)
  Password   = (Documanage password)
;  Debug     = Max
```

You must add the IP address, system/domain name, user ID, and password to the appropriate INI options.

Note For whichever cabinet name is being set in the DBHandler:DMIA control group, you must include that name in the DMIA control group name. RPEX1OS390 is the cabinet name in this case. So, the DMIA control group name would be *DMIA:RPEX1OS390*.

For the DMIA:RPEX1OS390 control group, the FSISYS file includes these values:

```
< DMIA:RPEX1OS390 >
  FileType   = DPA
  FolderBy   = company,lob,policynum
  NameDocBy  = arckey
  StackedDPA= Yes
```

Set the StackedDPA option to No when archiving to Documanage.

Adding IP Addresses and System/Domain Names

You will need to add the IP address and system/domain name of your Documanage machine to the local TCPIP host data sets on your z/OS. The steps below are examples you could follow to accomplish this task.

Note You first may need to consult your z/OS system programmer on how to get this task done as the steps you need to do may differ from below.

1. Add an entry with your IP and system/domain name to hlq.HOSTS.LOCAL.
2. Execute a TSO MAKESITE command to generate hlq.HOSTS.SITEINFO and hlq.HOSTS.ADDRINFO datasets from your hlq.HOSTS.LOCAL.

Changing the JCL

Examine the GenArcX member of JCLLIB. Make sure the JCL is appropriate for your site and that the dataset names correspond to the names you have used for your files.

ARCHIVING USING STACKED DPA

Configuring the system to use stacked DPA to archive to a CARDATA file involves performing these tasks:

- Checking the FSISYS file (FSISYSM member of DEFLIB)
- Making JCL changes

For the DBHandler:DMIA control group below, you do not need to add IP address, system/domain name, user ID, or password to run GenArc with stacked DPA.

```
< DBHandler:DMIA >
    RPCHost    = (IP address of the Documanage machine)
    Cabinet    = RPEX1OS390
    Domain     = (system/domain name)
    UserID     = (Documanage user ID)
    Password   = (Documanage password)
;    Debug     = Max
```

In the DMIA:RPEX1OS390 control group, make sure the StackedDPA option is set to Yes.

```
< DMIA:RPEX1OS390 >
    FileType   = DPA
    FolderBy   = company,lob,policynum
    NameDocBy  = arckey
    StackedDPA = Yes
```

Note All other INI options for archiving to Documanage are the same for stacked DPA.

Changing the JCL

Examine the GenArcX member of JCLLIB. Make sure the JCL is appropriate for your site and that the data set names correspond to the names you have used for your files.

CHECKING YOUR INSTALLATION

To make sure the base system has been installed correctly you should:

- Install DMS1 test resources.
- Check the INI files (FSIUSER, FSISYS, and FSIUSERL).
- Check the JCL for the DAPRUN and DAP procedure (or GentrnX, GenDataX, GenPrtX), and the DMRUNLM and DMLM procedure.
- Run DAPRUN (or GenTrnX, GenDataX, GenPrtX) and DMRUNLM.

Note You can verify your installation by running the system using the DMS1 resources. When you run the DAPRUN job later in this guide, you will need to specify as part of the jobcard, which set of resources the DAPRUN job is using.

You indicate the resource set by specifying this statement in your JCL:

```
//      SET RES=xxxxxx
```

where *xxxxxx* is *DMS1*. By default, the JOBCARD is set up to use DMS1 resources.

The DMS1 resources are in Documaker Studio format. These resources are stored in the DB2 library. Use the DMRUNLM job to run with DMS1 resources. Set the JOBCARD to use DMS1 resources.

CHECKING YOUR FSIUSER AND FSISYS SETTINGS

During installation, you loaded the FSIUSER and FSISYS members into the DEFLIB PDS during installation. These two files have similar control groups and options but different purposes. Generally, the FSIUSER member contains customer-specific initialization options while the FSISYS member contains general system-wide options.

Among other things, the FSIUSER and FSISYS members contain the names of various input and output files. On the PC, these file names include the path and file name. On z/OS, these names consist of DD statement names or PDS member names or both.

The FSIUSER and FSISYS you uploaded as part of the sample resources are in a format you can run on z/OS. To verify these values, consider the following but note that this list of options is not complete.

The relevant value in the FSIUSER INI file is...

```
< Environment >
  FSISYSINI  = DD:FSISYS
```

The relevant values in FSISYS that indicate the DD names of various libraries, the PDS members, or are required for the system to run are:

```
< BATCH1 >
  Printer = Printer1
< BATCH2 >
  Printer = Printer2
< BATCH3 >
  Printer = Printer3
< Manual >
  Printer = Pmanual
< Error >
```

```
Printer = Perror
< CONFIG:Batch Processing >
  CompLib = DD:COMPLIB()
  DDTLib = DD:DDTLIB()
  DefLib = DD:DEFLIB()
  FormDef = FORM
  FormLib = DD:FAPLIB()
  XRFFile = REL121
< Control >
  ImageExt = .FAP
  LogoExt = .LOG
  XrfExt = .FXR
< Data >
  DataPath =
  ExtrFile = DD:EXTRACT
  DbLogFile = DD:DBLOGFILE
  ErrFile = DD:ERRFILE
  LogFile = DD:LOGFILE
  MsgFile = DD:MSGFILE
  NAFile = DD:NAFILE
  NewTrn = DD:NEWTRN
  PolFile = DD:POLFILE
  TrnFile = DD:TRNFILE
  WIPBatch = DD:MANUAL
  Manual = DD:MANUAL
  TrnDfdFile = TrnDfdFl
  AfgJobFile = DD:AFGJOB
  RcbDfdFile = RcbDfdFl
  SetRcpTb = SetRcpTb
  TextTbl = TextTbl
  TblFile = TblFile
< Print_Batches >
  Batch1 = DD:BATCH1
  Batch2 = DD:BATCH2
  Batch3 = DD:BATCH3
  Manual = DD:MANUAL
  Error = DD:ERROR
< Printer >
  PrtType = AFP
; PrtType = XER
< Printer1 >
  Port = DD:PRTBAT1
< Printer2 >
  Port = DD:PRTBAT2
< Printer3 >
  Port = DD:PRTBAT3
< PManual >
  Port = DD:PMANUAL
< PError >
  Port = DD:PERROR
< PrinterInfo >
  Printer = Printer1
  Printer = Printer2
  Printer = Printer3
  Printer = PManual
  Printer = PError
< PrtType:XER >
  ColorCharts = Yes
  CompileInstream = No
  DownloadFonts = No,Disabled
  InitFunc = XERInit
  Module = XERW32
  OutputFunc = XEROutput
  OutMetFunc = XEROutMet
```

```
PageNumbers = Yes
PrintFunc = XERPrint
PrinterInk = Blue
TermFunc = XERTerm
SendColor = Yes
DJDEIden = A'@@@DJDE'
DJDEOffset = 0
DJDESkip = 8
Environment = MVS
OutMode = JES2
ImageOpt = Yes
JDEName = DFLT
JDLData = 0,255
JDLHost = IBMONL
JDLName = DEFAULT
JDLCode = NONE
JDLRStack = 0,10,EQ,X'13131313131313131313'
JDLRPage = 0,5,EQ,X'FFFF26FFFF'Printer control group
< PrtType:AFP >
    SendOverlays = Yes
< TRN_File >
    MaxExtRecLen = 119 (for RPEX1 only)
    MaxExtRecLen = 450 (for UTEX1 only)
    BinaryExt = N
```

CHECKING THE JCL

As stated in the topic, *Configuring Your Printer Type* on page 12, when you run the system with the RPEX1 and UTEX1 sample resources you can run each step as a separate job (GenTrnX, GenDataX, GenPrintX) or together in sequence using the DAPRUN job which invokes the DAP procedure in PROCLIB.

For the verification test, we recommend you use the DAPRUN job. If you want to use the DAPRUN job and DAP procedure, inspect and modify those files. If you want to run GenTrnX, GenDataX, and GenPrtX separately, inspect and modify those files.

If you plan to use the DAPPRUN job...

1. Check the DAPRUN job in JCLLIB. This job invokes the DAP procedure. Make sure DAPRUN has a valid jobcard and that the PROCLIB name is correct.
2. Check the DAP procedure in PROCLIB. This procedure contains these steps:

Description	
GENTRND	Deletes files in preparation for the GenTrn program
GENTRN	Runs the GenTrn program
GENDATAD	Deletes files in preparation for the GenData program
GENDATA	Runs the GenData program
GENPRTD	Deletes files in preparation for the GenPrint program
GENPRT	Runs the GenPrint program

3. Check the JCL statements in the DAP procedure and change the dataset names to match the names you chose during installation. Also, change any UNIT names to conform to your site's standards.

Remember that the DD names for various input files (DEFLIB, FAPLIB, TRNFILE, and so on) and output files (BATCH1, NAFILE, and so on) need to correspond to the values that point to these files in the FSISYS file.

You should not need to modify the actual DD names for these files either in the DAP procedure nor in the FSISYS member because these files have been set up to run the verification test as is.

Running GenTrnX, GenDataX, and GenPrtX separately

Look at the GenTrnX, GenDataX and GenPrtX jobs in JCLLIB and modify any jobcards, dataset names, unit names, and so on, so these jobs will run on your system.

Remember that the DD names for various input files (DEFLIB, FAPLIB, TRNFILE, and so on) and output files (BATCH1, NAFILE, and so on) should correspond to the values that point to these files in the FSISYS member. You should not need to modify the actual DD names for these files either in the JCL jobs or in the FSISYS member because these files are set up to run the verification test as is.

RUNNING DAPRUN (OR GENTRNX, GENDATA, AND GENPRTX) AND DMRUNLM

Submit the job DAPRUN (or DMRUNLM). DAPRUN (or DMRUNLM) should receive return codes of 00 or 04 in each step. The output for DAPRUN (or DMRUNLM) should include logging messages, which are sent to SYSPRINT. Depending on whether you are using the RPEX1 or UTEX1 sample resources, the messages will differ. The messages for DMS1 are similar to those for RPEX1. Examples are shown below and on the following pages.

Note If you are running GenTrnX, GenDataX, and GenPrtX separately you should submit each job in sequence and check the return codes.

Running DAPRUNC to Improve Performance

The DAPRUNC job invokes the DAPC procedure in PROCLIB. The DAPC procedure contains JCL that assumes the GenTrn and GenPrint steps have been combined into the GenData step. The GenTrn and GenPrint steps are run as rules within the GenData step. This is called *single-step processing* and can improve system performance. To use single-step processing, submit DAPRUNC. It should receive a return code of 00 or 04 in the GenData step.

Note When using single-step processing, only the SYSOUT messages for GenData are displayed.

Logging messages when running RPEX1

```

--- GenTrn ---
Filtering extract file...
Building transaction file...
==> Processing: TransactionId-GroupName1-GroupName2-GroupName3-
TransactionType
==> Processing: 1234567-SAMPCO-LB1--T1
==> Processing: 2234567-SAMPCO-LB1--T1
==> Processing: 5SAMPCO-SAMPCO-LB2--T1
==> Processing: 6SAMPCO-SAMPCO-LB2--T1
==> Processing: 7SAMPCO-SAMPCO-LB2--T1
==> Processing: 8SAMPCO-SAMPCO-LB2--T1
==> Processing: 9SAMPCO-SAMPCO-LB2--T1
==> Processing: 4234567-FSI-CPP--T1
==> Processing: 5234567-FSI-GL--T1
==> Warning count:    0
==> Error   count:    0
Elapsed Time: 3 seconds
--- GenTrn Completed ---

--- GenData ---
==> Processing: TransactionId-GroupName1-GroupName2-GroupName3-
TransactionType
==> Processing: 1234567-SAMPCO-LB1--T1
==> Processing: 2234567-SAMPCO-LB1--T1
==> Processing: 5SAMPCO-SAMPCO-LB2--T1
==> Processing: 6SAMPCO-SAMPCO-LB2--T1
==> Processing: 7SAMPCO-SAMPCO-LB2--T1
==> Processing: 8SAMPCO-SAMPCO-LB2--T1
==> Processing: 9SAMPCO-SAMPCO-LB2--T1
==> Processing: 4234567-FSI-CPP--T1
==> Processing: 5234567-FSI-GL--T1

```

```
==> Warning count:    0
==> Error   count:    0
Elapsed Time: 44 seconds
--- GenData Completed ---

--- GenPrint ---
Processing Batch:<BATCH1> File:<DD:BATCH1> Port:<DD:XERBAT1>
==> Processing: TransactionId-GroupName1-GroupName2-GroupName3-
TransactionType
==> Processing: 1234567-SAMPCO-LB1--T1
==> Processing: 2234567-SAMPCO-LB1--T1
==> Processing: 5SAMPCO-SAMPCO-LB2--T1
==> Processing: 6SAMPCO-SAMPCO-LB2--T1
==> Processing: 7SAMPCO-SAMPCO-LB2--T1
==> Processing: 8SAMPCO-SAMPCO-LB2--T1
==> Processing: 9SAMPCO-SAMPCO-LB2--T1
==> Processing: 4234567-FSI-CPP--T1
==> Processing: 5234567-FSI-GL--T1
9 records processed in BATCH1 batch.
Processing Batch:<BATCH2> File:<DD:BATCH2> Port:<DD:XERBAT2>
==> Processing: 1234567-SAMPCO-LB1--T1
==> Processing: 2234567-SAMPCO-LB1--T1
==> Processing: 5SAMPCO-SAMPCO-LB2--T1
==> Processing: 6SAMPCO-SAMPCO-LB2--T1
==> Processing: 7SAMPCO-SAMPCO-LB2--T1
==> Processing: 8SAMPCO-SAMPCO-LB2--T1
==> Processing: 9SAMPCO-SAMPCO-LB2--T1
==> Processing: 4234567-FSI-CPP--T1
==> Processing: 5234567-FSI-GL--T1
9 records processed in BATCH2 batch.
Processing Batch:<BATCH3> File:<DD:BATCH3> Port:<DD:XERBAT3>
==> Processing: 1234567-SAMPCO-LB1--T1
==> Processing: 2234567-SAMPCO-LB1--T1
==> Processing: 5SAMPCO-SAMPCO-LB2--T1
==> Processing: 6SAMPCO-SAMPCO-LB2--T1
==> Processing: 7SAMPCO-SAMPCO-LB2--T1
==> Processing: 8SAMPCO-SAMPCO-LB2--T1
==> Processing: 9SAMPCO-SAMPCO-LB2--T1
7 records processed in BATCH3 batch.
Processing Batch:<ERROR> File:<DD:ERROR> Port:<DD:PERROR>
Processing Batch:<MANUAL> File:<DD:MANUAL> Port:<DD:PMANUAL>
==> Warning count:    3
==> Error   count:    0
Elapsed Time: 49 seconds
--- GenPrint Completed ---
```

Logging messages when running UTEX1

```

--- GenTrn ---
Filtering extract file...
Building transaction file...
==> Processing: TransactionId-GroupName1-GroupName2-GroupName3-
TransactionType
==> Processing: 2222-3445-UTIL-BILL--
==> Processing: 1234-5555-UTIL-BILL--
==> Processing: 3456-7777-UTIL-BILL--
==> Processing: 3999-9999-UTIL-BILL--
==> Processing: 4444-1234-UTIL-BILL--
==> Processing: 4999-R777-UTIL-BILL--
==> Processing: 5111-8888-UTIL-BILL--
==> Processing: 5555-7010-UTIL-BILL--
==> Processing: 6750-R758-UTIL-BILL--
==> Processing: 7000-7000-UTIL-BILL--
==> Processing: 2428-1927-UTIL-BILL--
==> Processing: 8234-5555-UTIL-BILL--
==> Warning count:    0
==> Error   count:    0
Elapsed Time: 3 seconds
--- GenTrn Completed ---

```

```

--- GenData ---
==> Processing: TransactionId-GroupName1-GroupName2-GroupName3-
TransactionType
==> Processing: 2222-3445-UTIL-BILL--
==> Processing: 1234-5555-UTIL-BILL--
==> Processing: 3456-7777-UTIL-BILL--
==> Processing: 3999-9999-UTIL-BILL--
==> Processing: 4444-1234-UTIL-BILL--
==> Processing: 4999-R777-UTIL-BILL--
==> Processing: 5111-8888-UTIL-BILL--
==> Processing: 5555-7010-UTIL-BILL--
==> Processing: 6750-R758-UTIL-BILL--
==> Processing: 7000-7000-UTIL-BILL--
==> Processing: 2428-1927-UTIL-BILL--
==> Processing: 8234-5555-UTIL-BILL--
==> Warning count:    0
==> Error   count:    0
Elapsed Time: 148 seconds
--- GenData Completed ---

```

```

--- GenPrint ---
Processing Batch:<BATCH1> File:<DD:BATCH1> Port:<DD:AFPBAT1>
==> Processing: TransactionId-GroupName1-GroupName2-GroupName3-
TransactionType
==> Processing: 2222-3445-UTIL-BILL--
==> Processing: 1234-5555-UTIL-BILL--
==> Processing: 3456-7777-UTIL-BILL--
==> Processing: 3999-9999-UTIL-BILL--
==> Processing: 4444-1234-UTIL-BILL--
==> Processing: 4999-R777-UTIL-BILL--
==> Processing: 5111-8888-UTIL-BILL--
==> Processing: 5555-7010-UTIL-BILL--
==> Processing: 6750-R758-UTIL-BILL--
==> Processing: 7000-7000-UTIL-BILL--
==> Processing: 2428-1927-UTIL-BILL--
==> Processing: 8234-5555-UTIL-BILL--
12 records processed in BATCH1 batch.
Processing Batch:<BATCH2> File:<DD:BATCH2> Port:<DD:AFPBAT2>
Processing Batch:<BATCH3> File:<DD:BATCH3> Port:<DD:AFPBAT3>
Processing Batch:<ERROR> File:<DD:ERROR> Port:<DD:PERERROR>
Processing Batch:<MANUAL> File:<DD:MANUAL> Port:<DD:PMANUAL>

```

```
==> Warning count:    5  
==> Error   count:    0  
Elapsed Time: 75 seconds  
--- GenPrint Completed ---
```

USING THE DMS1 RESOURCES

After you download and install the Oracle Documaker media pack, you will have these files on your workstation:

- install.txt
- loaddms1.jcl
- loaddb2.jcl
- DMS1.zip

Note The install.txt file contains much of the same information included in this topic.

Follow these instructions to install DMS1 resources for z/OS.

1. Unzip the DMS1.zip file. It contains a file named *DMS1.xmit*. This file contains several z/OS datasets and is in a TRANSMIT (XMIT) format.

To install the DMS1 resources, you must transfer this file to z/OS in binary mode and then unzip (RECEIVE) it. The RECEIVE creates and populates the datasets that comprise the DMS1 test resources. Follow the remaining steps to transfer this file to z/OS and perform the RECEIVE.

2. Allocate a dataset on z/OS into which you can upload the DMS1.xmit file. Allocate this dataset using attributes similar to these:

```
Data Set Name . . . : FSI.DMS1.XMIT <= any name you choose

Organization . . . : PS
Record format . . . : FB
Record length . . . : 80
Block size . . . . : 3120
1st extent cylinders: 10      <= primary space needed
Secondary cylinders : 2      <= secondary space needed
```

3. Upload the DMS1.xmit file from Windows to z/OS. Be sure to upload this file in binary mode.
4. Upload LOADDMS1.JCL as text to your JCL library on z/OS. Read the comments in the JCL and make the necessary modifications for your system. Submit the LOADDMS1 job. It should create several datasets that contain the DMS1 sample resources.

Note Make sure you have completed the steps in *Binding the Documaker DB2 Plan to Your DB2 Subsystem* on page 6 before you continue to the next step.

5. Use LBYSQLR member in the JCLLIB to create DB2 tables for the Documaker library. Make the necessary modifications, including changing table names for your system, before submitting.

Note You can have all of the tables in one tablespace or you can place each table in a different tablespace.

6. Edit the FSI.LBYD.SYSPUNCH and FSI.LBYI.SYSPUNCH datasets, replacing the user ID *ZD* and the table names with the user ID and table names you used in the LBYSQLR member. Do not change anything else in this file. This file loads the DMS1 sample resources into your DB2 tables.

```
INTO TABLE "ZD"."DMS1_LBYD"  <= this line is in FSI.LBYD.SYSPUNCH
INTO TABLE "ZD"."DMS1_LBYI"  <= this line is in FSI.LBYI.SYSPUNCH
```

7. Upload the LOADDB2.JCL to your JCLLIB. Make the necessary modifications, including changing the name of the SYSPUNCH and SYSREC datasets for your system.

Submit LOADDB2 to load the tables. You will need to run this job twice — once with the SYSIN DD statement set to FSI.LBYD.SYSPUNCH and once with the SYSIN DD set to FSI.LBYI.SYSPUNCH.

Note A certain level of DB2 authority may be needed to run this job.

Checking Your FSIUSERL Settings

The FSIUSERL is a member of the DEFLIB PDS in the DMS1 installation. Besides those options as described in FSIUSER and FSISYS, the FSIUSERL member contains initialization options to retrieve resources from DB2 library.

The following partial list of options is from the FSIUSERL member. The DB2 database is set with the Database INI option. The DB2 tables are set in the DB2_FileConvert control group.

Be sure to make the necessary changes for your implementation.

```
< CONFIG:DMS1 >
  ArcPath          =
  BaseDef           = DMS1
  BDFFile           = LBYI
  BDFLib            =
  CARPath           =
  DALFile           = LBYI
  DDTFile           = LBYI
  DefLib            = DD:DEFLIB()
  DictionaryFile    = xdb.dbf
  FieldBaseFile     = fdb.dbf
  FNTFile           =
  FontLib           = DD:FONTLIB()
  FORFile           = LBYI
  FORLib            =
  FormDef           = FORM
  FormFile          = LBYI
  FormLib           =
  GRPFile           = LBYI
  GRPLib            =
  HelpLib           =
  LbyLib            = DD:DEFLIB()
  LogoFile          = LBYI
  TableLib          = DD:DEFLIB()
  WIPPath           =
  XDDFile           = LBYI
  XRFFile           = rel121
< Configurations >
  Config            = DMS1
< DBHANDLER:DB2 >
  Database          = LBYDB
```

```

        CreateTable = No
        CreateIndex = No
        Debug        = No
    < DBTable:LBYSI >
        DBHandler    = DB2
    < DBTable:LBYSI >
        DBHandler    = DB2
        UniqueTag    = ARCKEY+SEQ_NUM
    < DBTable:LBYSI >
        DBHandler    = DB2
        UniqueTag    = DATE+TIME
    < DBTable:Catalog >
        DBHandler    = DB2
        UniqueTag    = CatalogID
    < DB2_FileConvert >
        LBYSI        = ZD.DMS1_LBYSI
        LBYSI        = ZD.DMS1_LBYSI
        LBYSI        = ZD.DMS1_LBYSI
        Catalog      = ZD.DMS1_CAT
    < Library:LBYSI >
        DBTable      = LBYSI
        LBYSIFile    = LBYSI
    < LibraryManager >
        Class        = GA;Georgia resource
        Class        = TX;Texas resource
        Class        = MD;Maryland resource
        Library      = LBYSI
        Mode         = DEV;Development
        Mode         = TEST;Test
        Mode         = PROD;Production
        Project      = P001;Project 001
        Project      = P002;Project 002
        Project      = P003;Project 003
        Status       = TEST;Needs to be tested
        Status       = FAILED;Failed testing
        Status       = PASSED;Passed testing
        Status       = PROMOTED;Has been promoted
    < MasterResource >
        BaseDef      = <CONFIG:DMS1> BaseDef =
        BDFFile      = <CONFIG:DMS1> BDFFile =
        BDFLib       = <CONFIG:DMS1> BDFLib =
        COMPLIB      = <CONFIG:DMS1> CompLib =
        DALFile      = <CONFIG:DMS1> DalFile =
        DDTFile      = <CONFIG:DMS1> DDTFile =
        DDTLib       = <CONFIG:DMS1> DDTLib =
        DefLib       = <CONFIG:DMS1> DefLib =
        DictionaryFile = <CONFIG:DMS1> DictionaryFile =
        FieldBaseFile = <CONFIG:DMS1> FieldBaseFile =
        FNTFile      = <CONFIG:DMS1> FntFile =
        FontLib      = <CONFIG:DMS1> FontLib =
        FORFile      = <CONFIG:DMS1> FORFile =
        FORLib       = <CONFIG:DMS1> FORLib =
        Form7x       = <CONFIG:DMS1> Form7x =
        FormDef      = <CONFIG:DMS1> FormDef =
        FormFile     = <CONFIG:DMS1> FormFile =
        FormLib      = <CONFIG:DMS1> FormLib =
        GRPFile      = <CONFIG:DMS1> GRPFile =
        GRPLib       = <CONFIG:DMS1> GRPLib =
        HelpLib      = <CONFIG:DMS1> HelpLib =
        LbyLib       = <CONFIG:DMS1> LbyLib =
        LogoFile     = <CONFIG:DMS1> LogoFile =
        LogoLib      = <CONFIG:DMS1> LogoLib =
        TableLib     = <CONFIG:DMS1> TableLib =
        TablePath    = <MASTERRESOURCE> TableLib =

```

```
XDDFile           = <CONFIG:DMS1> XDDFile =  
XRFFFile          = <CONFIG:DMS1> XrffFile =
```

Checking the JCL

When you run the system with the DMS1 sample resources, use the DMRUNLM job which invokes the DMLM procedure in PROCLIB. Make sure DMRUNLM has a valid jobcard and that the PROCLIB name is correct.

Also in DMRUNLM job, check the override statement for GenTrn, GenData, and GenPrint to make sure you have the correct DB2 SUBSYSTEM, PLAN, and LINKLIB from your system installation.

See *Binding the Documaker DB2 Plan to Your DB2 Subsystem* on page 6 for more information on the DB2 SUBSYSTEM and PLAN. For a description of the DMLM procedure, see the description of the DAP procedure above.

Here is an example of the DMRUNLM job:

```
//* COPY JOBCARD HERE  
//*****  
//*                               - DMRUNLM -  
//*  
//*****  
//*  
//DMRUNLM EXEC DMLM  
//GENTRN.SYSTSIN DD *  
DSN SYSTEM(DB8G)  
RUN  PROGRAM(GENTRN) PLAN(DAP121) -  
      LIB('FSI.V121.LINKLIB')  
END  
//*  
//GENDATA.SYSTSIN DD *  
DSN SYSTEM(DB8G)  
RUN  PROGRAM(GENDATA) PLAN(DAP121) -  
      LIB('FSI.V121.LINKLIB')  
END  
//*  
//GENPRT.SYSTSIN DD *  
DSN SYSTEM(DB8G)  
RUN  PROGRAM(GENPRINT) PLAN(DAP121) -  
      LIB('FSI.V121.LINKLIB')  
END  
//*
```


RUNNING GENARCDX

This task is optional.

The JCLLIB member GenArcDX (GenArcX prior to version 10.3) executes the GenArc program in two steps. The first step, called *GenArcD*, executes the IEFBR14 program and contains DD statements that delete various files which will be created in the following GenArc step. The second step, called *GenArc*, executes the TSO Terminal Monitor Program and starts the DB2 Command Processor program (DSN) which runs the GenArc program and DB2 plan.

At the bottom of the GenArcDX job, the SYSTSIN DD statement has some in-stream control cards, one of which indicates the LINKLIB you loaded during the installation. Change FSI.V121.LINKLIB to match the name you gave LINKLIB during the installation. Also, you change the subsystem name from *DB8G* to match your DB2 subsystem name.

```
//SYSTSIN DD *
DSN SYSTEM(DB8G)
RUN PROGRAM(GENARC) PLAN(DAP121) -
LIB('FSI.V121.LINKLIB')
END
```

To test, submit the GenArcDX job. The GenArcDX job should receive a return code of *00* or *04*. The output for the GenArcDX job should include logging messages (sent to SYSPRINT) similar to the following messages.

RPEX1 messages

```
--- GenArc ---
==> Processing: TransactionId-GroupName1-GroupName2-GroupName3-
TransactionType
==> Processing: 1234567-SAMPCO-LB1--T1
==> Processing: 2234567-SAMPCO-LB1--T1
==> Processing: 5SAMPCO-SAMPCO-LB2--T1
==> Processing: 6SAMPCO-SAMPCO-LB2--T1
==> Processing: 7SAMPCO-SAMPCO-LB2--T1
==> Processing: 8SAMPCO-SAMPCO-LB2--T1
==> Processing: 9SAMPCO-SAMPCO-LB2--T1
==> Processing: 4234567-FSI-CPP--T1
==> Processing: 5234567-FSI-GL--T1

==> Transactions Read      :      9
==> Transactions Archived :      9
==> Transactions In Error  :      0
==> Transactions Rolled Back:      0

==> Warning count:      1
==> Error count:      0

Elapsed Time: 44 seconds
--- GenArc Completed ---
```

UTEX1 messages

```
--- GenArc ---

==> Processing: TransactionId-GroupName1-GroupName2-GroupName3-
TransactionType
==> Processing: 2222-3445-UTIL-BILL--
==> Processing: 1234-5555-UTIL-BILL--
==> Processing: 3456-7777-UTIL-BILL--
==> Processing: 3999-9999-UTIL-BILL--
==> Processing: 4444-1234-UTIL-BILL--
==> Processing: 4999-R777-UTIL-BILL--
==> Processing: 5111-8888-UTIL-BILL--
==> Processing: 5555-7010-UTIL-BILL--
==> Processing: 6750-R758-UTIL-BILL--
==> Processing: 7000-7000-UTIL-BILL--
==> Processing: 2428-1927-UTIL-BILL--
==> Processing: 8234-5555-UTIL-BILL--

==> Transactions Read      :      12
==> Transactions Archived :      12
==> Transactions In Error  :        0
==> Transactions Rolled Back:        0

==> Warning count:      0
==> Error   count:      0

Elapsed Time: 34 seconds
--- GenArc Completed ---
```

RUNNING GENARCX

This task is optional. The JCLLIB member GenArcX executes the GenArc program in two steps. The first step, called GenArcD, executes the IEFBR14 program and contains DD statements that delete various files which will be created in the following GenArc step. The second step, called GenArc, executes the GenArc program to archive to Documanage or to a CARDATA file if you are using stacked DPA.

To test, submit the GenArcX job. The GenArcX job should receive a return code of 00 or 04. The output for the GenArcX job should include logging messages (sent to SYSPRINT) similar to the following messages if you are using stacked DPA.

RPEX1 messages

```

--- GenArc ---
==> Processing: TransactionId-GroupName1-GroupName2-GroupName3-
TransactionType
==> Processing: 1234567-SAMPCO-LB1--T1
==> Processing: 2234567-SAMPCO-LB1--T1
==> Processing: 5SAMPCO-SAMPCO-LB2--T1
==> Processing: 6SAMPCO-SAMPCO-LB2--T1
==> Processing: 7SAMPCO-SAMPCO-LB2--T1
==> Processing: 8SAMPCO-SAMPCO-LB2--T1
==> Processing: 9SAMPCO-SAMPCO-LB2--T1
==> Processing: 4234567-FSI-CPP--T1
==> Processing: 5234567-FSI-GL--T1
==> Transactions Read      :      9
==> Transactions Archived :      7
==> Transactions In Error  :      0
==> Transactions Rolled Back:      0
==> Warning count:      1
==> Error count:      0
Elapsed Time: 27 seconds
--- GenArc Completed ---

```

UTEX1 messages

```

--- GenArc ---
==> Processing: TransactionId-GroupName1-GroupName2-GroupName3-
TransactionType
==> Processing: -UTIL-BILL--
==> Processing: -UTIL-BILL--
==> Processing: -UTIL-BILL--
==> Processing: -UTIL-BILL--
==> Processing: -UTIL-BILL--
==> Processing: -UTIL-BILL--
==> Processing: -UTIL-BILL--
==> Processing: -UTIL-BILL--
==> Processing: -UTIL-BILL--
==> Processing: -UTIL-BILL--
==> Processing: -UTIL-BILL--
==> Transactions Read      :      12
==> Transactions Archived :      12
==> Transactions In Error  :      0
==> Transactions Rolled Back:      0
==> Warning count:      1
==> Error count:      0
Elapsed Time: 66 seconds
--- GenArc Completed ---

```

PRINTING THE OUTPUT

When you performed the tasks discussed in the topic, *Running DAPRUN (or GenTrnX, GenDataX, and GenPrtX) and DMRUNLM* on page 41, DAPRUN should have produced three print-ready files. These files would be referenced as shown in the following table:

If you configured your system for...	You will see these default DD names...
Xerox Metacode	PRTBAT1, PRTBAT2, and PRTBAT3
IBM AFP	PRTBAT1, PRTBAT2, and PRTBAT3
PostScript	PRTBAT1, PRTBAT2, and PRTBAT3
PDF	PDFLIB(BAT10000), PDFLIB(BAT20000), and PDFLIB(BAT30000)

Consult the appropriate topic below to send the output to the type of printer you are configured for.

Xerox Metacode printer

If you configured the system to create Xerox Metacode, use the GENERXER member of JCLLIB to *IEBGENER* the print files to JES. Make any changes necessary in this member to match your z/OS environment.

BARR-attached Metacode printer

If you have a Metacode printer attached to your local area network (LAN) and you are using the BARR interface to connect to the mainframe, you must first *BARRWRAP* the output print stream, then send this print stream to the printer's spool area.

To BARRWRAP the output print stream, run the BARRWRPX utility. You'll find an example of this utility in JCLLIB. Then download the resulting file to your LAN. Be sure to download without carriage returns or line feeds (CRLF) and without ASCII translation. Once the file is downloaded, copy it to the BARR SPOOL area.

IBM AFP printer

If you have configured the system to create an IBM AFP data stream, use the GENERAFP member of JCLLIB to *IEBGENER* the print files to JES. Make any changes necessary in this member to match your z/OS environment.

THE LOADDAPW JOB

The LOADDAPW job in JCLLIB is listed below. This job creates the...

- System datasets
- RPEX1, UTEX1, and PDF417 sample resources (for the verification test)

Here is the listing of the system and sample resources datasets:

- JCLLIB
- PROCLIB
- SOURCE
- INCLIB
- OBJLIB
- SOURCE (NY State Source)
- INCLIB (NY State Header Files)
- OBJLIB (NY State Object Library)
- INCLIB (Symbol Technology Header Files)
- OBJLIB (Symbol Technology Object Library)
- LINKLIB
- DBRMLIB (DB2)
- DEFLIB (RPEX1)
- DDTLIB (RPEX1)
- FAPLIB (RPEX1)
- COMPLIB (RPEX1)
- EXTRACT (RPEX1)
- LOG2IMG (RPEX1 Xerox Logo Images)
- LOG2XFNT (RPEX1 Xerox Logo Fonts)
- PFRMLIB (RPEX1 Xerox Forms)
- PMETLIB (RPEX1 Xerox Precompiled Metacode)
- OVERLIB (RPEX1 PostScript Overlays)
- DEFLIB (UTEX1)
- DDTLIB (UTEX1)
- FAPLIB (UTEX1)
- COMPLIB (UTEX1)
- EXTRACT (UTEX1)

- LOG2IMG (UTEX1 Xerox Logo Images)
- LOG2XFNT (UTEX1 Xerox Logo Fonts)
- PMETLIB (UTEX1 Xerox Precompiled Metacode)
- OVERLIB (UTEX1 PostScript Overlays)
- DEFLIB (PDF417)
- DDTLIB (PDF417)
- FAPLIB (PDF417)
- COMPLIB (PDF417)
- EXTRACT (PDF417)
- PMETLIB (PDF417 Xerox Precompiled Metacode)
- OVERLIB (PDF417 PostScript Overlays)

In addition, the LOADDAPW job reads from the installation and populates these datasets.

```
/** COPY JOBCARD HERE
//
*****
/** LOADDAPW -
/**
/** COPY THE DAP BATCH SYSTEM FILES, RPEX1 AND UTEX1 RESOURCES
/** FROM A TRANSMIT/XMIT FILE TO A Z/OS SYSTEM. THE JOB DOES THE
/** FOLLOWING:
/**
/** 1. RECEIVE - RECEIVES A FILE THAT HAS BEEN UPLOADED FROM
/**                WINDOWS (THE FILE IS IN TRANSMIT/XMIT FORMAT)
/**                INTO A PARTITIONED DATASET.
/**
/** 2. RECVDAP - RECEIVES A PARTITIONED DATASET FROM STEP 1
/**                TO LOAD DAP RESOURCES.
/**
/** 3. EXPORT - ALLOCATES AN EXPORT DATASET NEEDED FOR LINKING C++
/**                WITH EXPORTALL.
/**
/** 4. SYSCPRT - ALLOCATES A SYSCPRT DATASET NEEDED ONLY FOR
/**                SOURCE COMPILE.
/**
/** REPLACE 'FSI.DAP.XMIT' IN RECEIVE STEP WITH YOUR UPLOADED
/** DATASET NAME (IN ITEM 1 OF README.TXT)
/** - THIS DATASET CAN BE DELETED WHEN THE JOB FINISHES -
/**
/** REPLACE 'FSI.DAP.XMIT.PDS' IN RECEIVE STEP AND SET PDS=
/** WITH YOUR DATASET NAME (MUST BE A NON-EXISTING DATASET NAME)
/** - THIS DATASET CAN BE DELETED WHEN THE JOB FINISHES -
/**
/** REPLACE '&HLQ.' IN RECVDAP STEP, EXPORT STEP, AND SYSCPRT STEP
/** WITH YOUR HIGH-LEVEL QUALIFIERS
/**
//
*****
/**
/**RECEIVE EXEC PGM=IKJEFT01,REGION=0M,DYNAMNBR=175
/**SYSUADS DD DISP=SHR,DSN=SYS1.UADS
/**SYSLBC DD DISP=SHR,DSN=SYS1.BROADCAST
```

```

// *
//XMITFILE DD DISP=SHR,DSN=FSI.DAP.XMIT
// *
//SYSPRINT DD SYSOUT=*
//SYSOUT DD SYSOUT=*
//SYSTSOUT DD SYSOUT=*
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
RECEIVE INDDNAME(XMITFILE)
        DSN('FSI.DAP.XMIT.PDS')
// *
// SET PDS='FSI.DAP.XMIT.PDS'
// *
//RECVDAP EXEC PGM=IKJEFT01,REGION=0M,DYNAMNBR=175
//SYSUADS DD DISP=SHR,DSN=SYS1.UADS
//SYSLBC DD DISP=SHR,DSN=SYS1.BROADCAST
// *
//XMITPD1 DD DISP=SHR,DSN=&PDS(JCLLIB)
//XMITPD2 DD DISP=SHR,DSN=&PDS(PROCLIB)
//XMITPD3 DD DISP=SHR,DSN=&PDS(SOURCE)
//XMITPD4 DD DISP=SHR,DSN=&PDS(INCLIB)
//XMITPD5 DD DISP=SHR,DSN=&PDS(OBJLIB)
//XMITPD6 DD DISP=SHR,DSN=&PDS(SRCNY)
//XMITPD7 DD DISP=SHR,DSN=&PDS(INCNY)
//XMITPD8 DD DISP=SHR,DSN=&PDS(OBJNY)
//XMITPD9 DD DISP=SHR,DSN=&PDS(INCSYM)
//XMITPD10 DD DISP=SHR,DSN=&PDS(OBJSYM)
//XMITPD11 DD DISP=SHR,DSN=&PDS(LINKLIB)
//XMITPD12 DD DISP=SHR,DSN=&PDS(DBRMLIB)
//XMITPD13 DD DISP=SHR,DSN=&PDS(DEFLIB)
//XMITPD14 DD DISP=SHR,DSN=&PDS(DDTLIB)
//XMITPD15 DD DISP=SHR,DSN=&PDS(FAPLIB)
//XMITPD16 DD DISP=SHR,DSN=&PDS(COMPLIB)
//XMITPD17 DD DISP=SHR,DSN=&PDS(EXTRACT)
//XMITPD18 DD DISP=SHR,DSN=&PDS(LOG2IMG)
//XMITPD19 DD DISP=SHR,DSN=&PDS(LOG2XFNT)
//XMITPD20 DD DISP=SHR,DSN=&PDS(PFRMLIB)
//XMITPD21 DD DISP=SHR,DSN=&PDS(PMETLIB)
//XMITPD22 DD DISP=SHR,DSN=&PDS(OVERLIB)
//XMITPD23 DD DISP=SHR,DSN=&PDS(DEFLIBU)
//XMITPD24 DD DISP=SHR,DSN=&PDS(DDTLIBU)
//XMITPD25 DD DISP=SHR,DSN=&PDS(FAPLIBU)
//XMITPD26 DD DISP=SHR,DSN=&PDS(COMPLIBU)
//XMITPD27 DD DISP=SHR,DSN=&PDS(EXTRACTU)
//XMITPD28 DD DISP=SHR,DSN=&PDS(LOG2IMGU)
//XMITPD29 DD DISP=SHR,DSN=&PDS(LOG2XFTU)
//XMITPD30 DD DISP=SHR,DSN=&PDS(PMETLIBU)
//XMITPD31 DD DISP=SHR,DSN=&PDS(OVERLIBU)
//XMITPD32 DD DISP=SHR,DSN=&PDS(DEFLIB4)
//XMITPD33 DD DISP=SHR,DSN=&PDS(DDTLIB4)
//XMITPD34 DD DISP=SHR,DSN=&PDS(FAPLIB4)
//XMITPD35 DD DISP=SHR,DSN=&PDS(COMPLIB4)
//XMITPD36 DD DISP=SHR,DSN=&PDS(EXTRACT4)
//XMITPD37 DD DISP=SHR,DSN=&PDS(PMETLIB4)
//XMITPD38 DD DISP=SHR,DSN=&PDS(OVERLIB4)
// *
//SYSPRINT DD SYSOUT=*
//SYSOUT DD SYSOUT=*
//SYSTSOUT DD SYSOUT=*
//SYSTSPRT DD SYSOUT=*
//
*****
// *
// * NOTE THAT SYMBOLIC SUBSTITUTION IS NOT ALLOWED IN

```

```
/**      DSN('&HLQ..    . . . ')
/**      YOU MUST REPLACE '&HLQ.' WITH ACTUAL QUALIFIER NAMES
/**
/** YOU CAN DELETE ANY OF THE FOLLOWING RECEIVE INDDNAME/DSN PAIRS
/**      THAT YOU DON'T NEED TO LOAD.  EACH DELETION SHOULD INCLUDE
/**      RECEIVE INDDNAME AND DSN TOGETHER AS A PAIR
/**
/**
*****
//SYSTSIN DD *
RECEIVE INDDNAME(XMITPD1)
  DSN('&HLQ..JCLLIB') SPACE(2,2) CYLINDERS
RECEIVE INDDNAME(XMITPD2)
  DSN('&HLQ..PROCLIB') SPACE(2,1) CYLINDERS
RECEIVE INDDNAME(XMITPD3)
  DSN('&HLQ..SOURCE') SPACE(40,5) CYLINDERS
RECEIVE INDDNAME(XMITPD4)
  DSN('&HLQ..INCLIB') SPACE(10,1) CYLINDERS
RECEIVE INDDNAME(XMITPD5)
  DSN('&HLQ..OBJLIB') SPACE(30,5) CYLINDERS
RECEIVE INDDNAME(XMITPD6)
  DSN('&HLQ..NYAAMVA.SOURCE') SPACE(1,1) CYLINDERS
RECEIVE INDDNAME(XMITPD7)
  DSN('&HLQ..NYAAMVA.INCLIB') SPACE(1,1) CYLINDERS
RECEIVE INDDNAME(XMITPD8)
  DSN('&HLQ..NYAAMVA.OBJLIB') SPACE(1,1) CYLINDERS
RECEIVE INDDNAME(XMITPD9)
  DSN('&HLQ..SYMBOL.INCLIB') SPACE(1,1) CYLINDERS
RECEIVE INDDNAME(XMITPD10)
  DSN('&HLQ..SYMBOL.OBJLIB') SPACE(1,1) CYLINDERS
RECEIVE INDDNAME(XMITPD11)
  DSN('&HLQ..LINKLIB') SPACE(200,10) CYLINDERS
RECEIVE INDDNAME(XMITPD12)
  DSN('&HLQ..DBRMLIB') SPACE(1,1) CYLINDERS
RECEIVE INDDNAME(XMITPD13)
  DSN('&HLQ..RPEX1.DEFLIB') SPACE(2,1) CYLINDERS
RECEIVE INDDNAME(XMITPD14)
  DSN('&HLQ..RPEX1.DDTLIB') SPACE(1,1) CYLINDERS
RECEIVE INDDNAME(XMITPD15)
  DSN('&HLQ..RPEX1.FAPLIB') SPACE(2,1) CYLINDERS
RECEIVE INDDNAME(XMITPD16)
  DSN('&HLQ..RPEX1.COMPLIB') SPACE(1,1) CYLINDERS
RECEIVE INDDNAME(XMITPD17)
  DSN('&HLQ..RPEX1.EXTRACT') SPACE(1,1) CYLINDERS
RECEIVE INDDNAME(XMITPD18)
  DSN('&HLQ..RPEX1.LOG2IMG') SPACE(1,1) CYLINDERS
RECEIVE INDDNAME(XMITPD19)
  DSN('&HLQ..RPEX1.LOG2XFNT') SPACE(1,1) CYLINDERS
RECEIVE INDDNAME(XMITPD20)
  DSN('&HLQ..RPEX1.PFRMLIB') SPACE(1,1) CYLINDERS
RECEIVE INDDNAME(XMITPD21)
  DSN('&HLQ..RPEX1.PMETLIB') SPACE(1,1) CYLINDERS
RECEIVE INDDNAME(XMITPD22)
  DSN('&HLQ..RPEX1.PST.OVERLIB') SPACE(1,1) CYLINDERS
RECEIVE INDDNAME(XMITPD23)
  DSN('&HLQ..UTEX1.DEFLIB') SPACE(1,1) CYLINDERS
RECEIVE INDDNAME(XMITPD24)
  DSN('&HLQ..UTEX1.DDTLIB') SPACE(1,1) CYLINDERS
RECEIVE INDDNAME(XMITPD25)
  DSN('&HLQ..UTEX1.FAPLIB') SPACE(1,1) CYLINDERS
RECEIVE INDDNAME(XMITPD26)
  DSN('&HLQ..UTEX1.COMPLIB') SPACE(1,1) CYLINDERS
RECEIVE INDDNAME(XMITPD27)
  DSN('&HLQ..UTEX1.EXTRACT') SPACE(1,1) CYLINDERS
```



```
RECEIVE INDDNAME(XMITPD28)
  DSN('&HLQ..UTEX1.LOG2IMG') SPACE(1,1) CYLINDERS
RECEIVE INDDNAME(XMITPD29)
  DSN('&HLQ..UTEX1.LOG2XFNT') SPACE(1,1) CYLINDERS
RECEIVE INDDNAME(XMITPD30)
  DSN('&HLQ..UTEX1.PMETLIB') SPACE(1,1) CYLINDERS
RECEIVE INDDNAME(XMITPD31)
  DSN('&HLQ..UTEX1.PST.OVERLIB') SPACE(1,1) CYLINDERS
RECEIVE INDDNAME(XMITPD32)
  DSN('&HLQ..PDF417.DEFLIB') SPACE(2,1) CYLINDERS
RECEIVE INDDNAME(XMITPD33)
  DSN('&HLQ..PDF417.DDTLIB') SPACE(1,1) CYLINDERS
RECEIVE INDDNAME(XMITPD34)
  DSN('&HLQ..PDF417.FAPLIB') SPACE(1,1) CYLINDERS
RECEIVE INDDNAME(XMITPD35)
  DSN('&HLQ..PDF417.COMPLIB') SPACE(1,1) CYLINDERS
RECEIVE INDDNAME(XMITPD36)
  DSN('&HLQ..PDF417.EXTRACT') SPACE(1,1) CYLINDERS
RECEIVE INDDNAME(XMITPD37)
  DSN('&HLQ..PDF417.PMETLIB') SPACE(1,1) CYLINDERS
RECEIVE INDDNAME(XMITPD38)
  DSN('&HLQ..PDF417.PST.OVERLIB') SPACE(1,1) CYLINDERS
/*
/* ALLOCATE EXPORT DATASET FOR LINKING C++ WITH EXPORTALL.
/*
//EXPORT EXEC PGM=IEFBR14
//SYSPRINT DD SYSOUT=*
//DD1 DD DSN=&HLQ..EXPORT,
// DISP=(,CATLG),
// UNIT=SYSDA,DCB=(RECFM=FB,LRECL=80,BLKSIZE=3200),
// SPACE=(TRK,(10,2,5)),DSNTYPE=LIBRARY
//SYSIN DD DUMMY
/*
/* ALLOCATE SYSCPRT FOR SOURCE COMPILE.
/*
//SYSCPRT EXEC PGM=IEFBR14
//SYSPRINT DD SYSOUT=*
//DD1 DD DSN=&HLQ..SYSCPRT,
// DISP=(,CATLG),
// UNIT=SYSDA,RECFM=VBA,LRECL=137,
// SPACE=(CYL,(25,25,100)),DSNTYPE=LIBRARY
//SYSIN DD DUMMY
```

z/OS LIBRARIES

The system begins as many source code modules which are separated into directories or libraries. Each library performs a specific task. For instance, INILIB handles Initialization file tasks and GVMLIB performs global variable manager functions.

Some of these libraries perform functions that are relevant only on the PC platform. You do not need to upload these libraries to your z/OS system. The libraries you must upload to your z/OS system are listed in the table below, along with the various print drivers.

An asterisk (*) indicates the library is needed only to perform a particular function. For instance FAP2MET is required only if you are converting FAP files into PMETs and running with the CompileInstream INI option set to *No*.

Library	Required for					
	The system?	Xerox?	AFP?	PostScript?	PCL?	PDF?
A2WBLIB	Y*					
AFP2MVS			Y			
AFPCOPY			Y			
AFPFMDEF			Y			
AFPLIB			Y			
AFPOPT			Y*			
AFPRESRC			Y*			
AFPPRT			Y			
ARCLIB	Y*					
ASCLIB	Y					
BARR2MVS						
BARRWRAP		Y*				
CARLIB	Y					
CFA2FAP	Y					
CPCNV	Y					
CUSLIB	Y					
DALLIB	Y					
DB2DB	Y					
DB2LIB	Y					
DBLIB	Y					

Library	Required for					
	The system?	Xerox?	AFP?	PostScript?	PCL?	PDF?
DMILIB	Y*					
DSLIB	Y					
DXMLIB	Y					
ERRLIB	Y					
FAP2AFP	Y*					
FAP2CFA						
FAP2MET		Y*				
FAP2OVL			Y*			
FAP2PDF						Y*
FAPLIB	Y					
FDT2CFA	Y					
FIXOFFS						
FNTLIB	Y					
FSILIB	Y					
FSIVER	Y					
GenArc	Y*					
GenData	Y					
GENLIB	Y					
GenPrint	Y					
GenTrn	Y					
GenWIP	Y*					
GLBLIB	Y					
GRFLIB	Y					
GVMLIB	Y					
INILIB	Y					
JPGLIB	Y					
KSDS2SEQ	Y					
LBRYMGR	Y					

Library	Required for					
	The system?	Xerox?	AFP?	PostScript?	PCL?	PDF?
LBYLIB	Y					
LBYPROC	Y					
LBYSYNC	Y					
LINLIB	Y					
LOG2PSEG			Y*			
LOGLIB	Y					
METOPT		Y				
METRESRC		Y*				
MRG2MVS		Y*	Y*			
PALLIB	Y					
PCLLIB					Y	
PDFKEY						Y*
PDFLIB_CPP						Y*
PDS2VSAM	Y*					
PRTLIB	Y					
PSTLIB				Y		
PXLIB	Y					
RCBLIB	Y					
RCP2VSAM	Y*					
RPLIB	Y					
RULLIB	Y					
SAMPAFP			Y			
SEQ2KSDS	Y*					
TEXTDRIV	Y					
TIFLIB	Y					
TRANSLAT	Y					
UTILLIB	Y					
UTLLIB	Y					

Library	Required for					
	The system?	Xerox?	AFP?	PostScript?	PCL?	PDF?
VMMLIB	Y					
VRF2EXP	Y*					
WIPLIB	Y*					
XDBLIB	Y					
XERDNLD		Y*				
XERLIB		Y				
XMCLIB		Y				

Z/OS PROGRAMS AND UTILITIES

The following is a list and a short description of the programs and utilities you can run on a z/OS system. For more information about these utilities, see the Utilities Reference.

AFPCOPY utility

This utility takes an AFP print stream which may contain multiple BEGINDOC and ENDDOC statements and generates an AFP print stream with only one BEGINDOC and one ENDDOC statement—in other words, it converts the print stream from *cut-sheet* to *continuous-form*. No parameters are required. The input AFP print stream is read from the INFILE DD statement and the output AFP print stream is written to the OUTFILE DD statement. Look in member AFPCOPYX of JCLLIB to find an example of this utility.

AFPOPT utility

This utility optimizes an AFP print stream. The AFPOPT utility reads an AFP print stream produced by Documaker and outputs a smaller, optimized AFP print stream. The input AFP print stream is read from the INFILE DD statement. The output AFP print stream is written to the OUTFILE DD statement. Look in member AFPOPTX of JCLLIB to find an example of this utility.

AFPRESRC utility

This utility performs these tasks:

- Determines the AFP resources, such as formdef, fonts, overlays, and page segments, used by an AFP print stream
- Combines the original AFP print stream along with its required AFP resources

Look in member AFPRESC1 (to determine AFP resources) and AFPRESC2 (to combine AFP resources) of the JCLLIB to find examples of this utility.

AFPFMDEF utility

This utility accepts as input a source form definition file and generates an AFP FormDef. Look in member AFPFMDFX of JCLLIB to find an example of this utility.

AFP2MVS utility

This utility accepts as input an AFP resource, such as a font, page segment, or print-ready file, that's been uploaded to the z/OS system from the PC with no ASCII and no CRLF. It generates that resource to another file such that the record-oriented format of the resource will be restored. Look in the AFP2MVSS or AFP2MVSX member of JCLLIB to find an example of this utility.

BARR2MVS utility

This utility program accepts as input a print-ready Metacode file produced by Documaker Server running on the PC (using these INI options: Environment=OS22, OutMode=BARR, or BARRWORD), strips that file of the BARR-encoded information and produces a Metacode print-ready file acceptable in a z/OS environment. Look in the BARR2MVX member of JCLLIB to find an example of this utility.

BARRWRAP utility

This utility accepts as input a print-ready Metacode file produced by Documaker Server on z/OS and produces a Metacode print-ready file *wrapped* with control information the BARR printer-attachment system understands. Look in the BARRWRPX member of JCLLIB to find an example of this utility.

CPCNV utility

This utility converts a source file from one code page to another. The utility reads an input source file, such as an extract file, a FAP file, and so on, which corresponds to a specific code page. You can convert either a single member in the PDS (use /I=member) or all members of the PDS (use /I=*). The utility then generates a target file which corresponds to a different code page. Look in member CPCNVX of JCLLIB to find an example of this utility.

FAP2AFP utility

This utility generates an AFP print-ready file from a FAP file. You can convert a single FAP file in the PDS (/I=FAPNAME) or all FAP files in the PDS (/I=*). Look in the FAP2AFPX member of JCLLIB to find an example of this utility.

You can also convert a FAP file in the library (use /LIB=LIBNAME). You can specify the optional parameter /VER and /REV parameters to specify the version and/or revision of a FAP file in the library you want to convert. Keep in mind that you cannot use wildcards (/I=*) when you include the /LIB parameter.

Look in the FAP2AFPL member of JCLLIB to find an example of this utility.

FAP2CFA utility

This utility accepts as input a FAP file and generates a compiled FAP file. Look in member FAP2CFAX of JCLLIB to find an example of this utility.

FAP2MET utility

This utility accepts as input a FAP file and generates a pre-compiled Metacode member, usually placing the *PMET* into the PMETLIB dataset. You can convert a single FAP in the PDS (use /I=fapname) or all FAPs in the PDS (use /I=*). There are several options you can select when you run the FAP2MET utility. For example, you can save the PMET in the PMETLIB dataset so it is ready to be included by the GenPrint program into a print stream or you can generate the PMET so it can be sent directly to the Metacode printer. You can also select the HMI, Short Edge Bind, or Long Edge Bind options. Look in member FAP2METX or FAP2METS of JCLLIB to find an example of this utility.

FAP2PDF utility

This utility generates a PDF file from a FAP file. The PDF file is written into a member of PDFLIB PDS. You can convert a single FAP in the PDS (use /I=fapname) or all FAPs in the PDS (use /I=*). Look in the FAP2PDFX member of JCLLIB to find an example of this utility.

FAP2OVL utility

This utility generates an AFP overlay from a FAP file, usually placing it into an overlay PDS, which is generally referred to as *OVERLIB*. You can convert a single FAP in the PDS (use /I=fapname) or all FAPs in the PDS (use /I=*). Look in the FAP2OVLX or FAP2OVLS member of JCLLIB to find an example of this utility.

FDT2CFA utility

This utility accepts as input a FSISYS INI file (member of DEFLIB), a FORM file (member of DEFLIB) and the FAPLIB PDS (Partitioned Data Set) and produces a COMPLIB PDS containing compiled FAP files.

This utility is similar to the FAP2CFA utility, except the FAP2CFA utility reads a single FAP and compiles it into a compiled FAP, while the FDT2CFA utility reads a FSISYS INI file, gets the name of the FORM member, opens the FORM member of DEFLIB to get a list of all of the FAP files that might be used, compiles all of these FAP files into compiled FAPs, and places them into the COMPLIB. Look in member FDT2CFAX of JCLLIB to find an example of this utility.

FIXOFFS utility

Use the FIXOFFS utility to correct field offsets after you have uploaded or downloaded a file from an environment where physical offset values differ. For example, you must use this utility if you are running the GenArc program in a z/OS environment using a local area network (LAN) archive of z/OS-generated data.

The utility uses the offsets in the NA and POL files to determine how to update other output files, such as recipient batch files and the NEWTRN file. Look in member FIXOFFSX of JCLLIB to find an example of this utility.

FSIVER utility

This utility accepts as input a LINKLIB PDS and generates a report that lists which patches have been applied to the programs in that LINKLIB. You can generate the report for either a single program (use /I=program name) or for all programs in the PDS (use /I=*). Look in member FSIVERX of JCLLIB to find an example of this utility.

SEQ2KSDS utility

Use the SEQ2KSDS utility to convert a non-VSAM NAFILE or POLFILE dataset into a VSAM copy of that dataset. A 4-byte key is prefixed to each record of the VSAM dataset as it is created. You can find sample JCL for running the SEQ2KSDS utility in the SEQ2KSDX member of JCLLIB.

GenArc program

The GenArc program is an optional part of the system which you run after the GenData program. The GenArc program reads the NEWTRN, the NAFILE, and POLFILE and writes (or archives) this information to a repository (DB2). You can retrieve and print these archived form sets. The GenArcDX member of JCLLIB runs GenArc.

GenData program

The GenData program uses a processing engine referred to as the *Rules Processor* to step through the transaction trigger file (TRNFILE) created by the GenTrn program. This creates an NAFILE which contains section and variable data information, a POLFILE which contains recipient information, and recipient batch files (BATCH1, BATCH2, and so on) which contain trigger records for the different recipient groups. These files are sent to the GenPrint (or GenArc) program. The GenData program is part of the DAP procedure in PROCLIB. There is also an example of GenData in the GenDataX member of JCLLIB.

GenPrint program

The GenPrint program reads the recipient batch files, NAFILEs, and POLFILEs produced by the GenData program and produces print-ready streams for specific printer platforms such as Xerox Metacode or IBM AFP. The print streams can be sent to datasets which you can copy to the JES print SPOOL or directly to JES SPOOL as SYSOUT files. The GenPrint program is part of the DAP procedure in PROCLIB. There is also an example of GenPrint in the GenPrtX member of JCLLIB.

GenTrn program

The GenTrn program performs the first step of the processing cycle. The GenTrn program accepts as input an extract file and produces a transaction trigger file or *TRNFILE*. Each record in the TRNFILE represents a single transaction. The TRNFILE is then passed to the GenData program. The GenTrn program is part of the DAP procedure in PROCLIB. There is also an example of GenTrn in the GenTrnX member of JCLLIB.

LOG2PSEG utility

This utility accepts as input a logo file, which is similar to a bitmap file, and produces an AFP page segment. The page segment is generally placed into a page segment PDS referred to as PSEGLIB. This PSEGLIB must be available to your AFP printer when you print. You can convert a single logo in the PDS (use /I=logoname) or all logos in the PDS (use /I=*). Look in members LOG2PSGX or LOG2PSGS of JCLLIB for an example of this utility.

METOPT utility

This utility optimizes a Metacode print stream. This utility reads a Metacode print stream produced by Documaker and outputs a smaller, optimized Metacode print stream. The input Metacode print stream is read from the INFILE DD statement. The output Metacode print stream is written to the OUTFILE DD statement. Look in member METOPTX of JCLLIB for an example of this utility.

METRESRC utility

This utility performs these tasks:

- Determines the Xerox resources, such as fonts, sections, forms, and logos, used by a Metacode print stream
- Combine the original Metacode print stream along with its required Xerox resources

Look in member METRESC1 (to determine Xerox resources) and METRESC2 (to combine Xerox resources) of JCLLIB for examples of this utility.

PDFKEY utility

Use the PDFKEY utility to generate the encrypted passwords used in the PDF security control group. Look in member PDFKEYX of JCLLIB for an example of this utility.

PDS2VSAM utility

This utility accepts as input a partitioned dataset, such as the DDTLIB or PMETLIB PDSs, and copies those members into a VSAM KSDS. Look in members PDS2VSDX, PDS2VSJX, and PDS2VSPX of JCLLIB for examples of this utility.

RCP2VSAM utility

This utility accepts as input a set recipient table, usually the SETRCPTB member in the DEFLIB PDS, and copies this table into a VSAM KSDS. When you have large set recipient tables, this conversion can speed the reading of the table. Look in member RCP2VSMX of JCLLIB for an example of this utility.

SAMPAFP utility

This utility accepts as input the name of either an AFP overlay or page segment and generates a SYSOUT file that, together with the correct USERLIB reference to this AFP resource, will print this overlay or page segment resource. Look in member SAMPAFPX of JCLLIB for an example of this utility.

TRANSLAT utility

This utility accepts as input a tokenized message file (MSGFILE) that is output from the GenTrn, GenData, GenPrint, or GenArc program. This utility then produces a readable error file. The TRANSLAT utility lets you translate the tokens in the MSGFILE into various languages or simply change the contents to meet your needs. Look in member TRANSLTX of JCLLIB for an example of this utility.

XERDNLD utility

This utility lets you get resources to a Xerox printer if you do not have other software programs to accomplish this task.

The utility accepts as input a Xerox Metacode resource, such as a font, section, or FRM file, encapsulates it with location-specific DJDE information and generally writes the resource back out to a PDS. You can then send the contents of this PDS to the JES SPOOL and target those contents to a specific Xerox printer where the resource will be stored on the printer's local disk device.

The resources will then be available when needed at print time. Look in the XERDNLDX or XERDNLDS members of JCLLIB for examples of this utility.

USING THE DOCUMENTATION

After you download Oracle Documaker and unzip the file you downloaded, you will find a readme file that provides an overview, late-breaking information, and links to the documentation. For more detailed instructions on downloading the documentation, see the following topic.

The documentation available for this release is in Portable Document Format (PDF) files which you can view or print using the Adobe® Acrobat™ Reader™. In the documentation file you download from the OTN web site, you will find the various documentation that accompanies this release, such as:

- Documaker Administration Guide
- Documaker Studio User Guide
- Utilities Reference
- Rules Reference
- DAL Reference
- Release Notes

For best results, make sure you have the latest version of Acrobat Reader. You can download the free reader at Adobe's web site: www.adobe.com.

UPDATING YOUR DOCUMENTATION

You can download the latest Oracle Insurance documentation at the Oracle Technology Network (OTN) web site. The process includes:

- Going to the Oracle Insurance Documentation page on the OTN site
- Searching for the application documentation you want to download
- Downloading that documentation

To download Oracle Insurance application documentation, go to the Oracle Insurance Documentation page:

<http://www.oracle.com/technetwork/documentation/insurance-097481.html>

DOWNLOADING PATCHES

You can download the latest Oracle software patches at the My Oracle Support web site. The process includes:

- Going to the My Oracle Support site (requires registration)
- Searching for the patches you want to download
- Downloading those patches

To download Oracle software patches, go to the My Oracle Support web site:

<https://support.oracle.com>

Chapter 2

Installing Documaker Server on UNIX

This guide provides detailed information on how to install and configure the system on an IBM AIX, Sun Solaris, and Linux environments.

You will find information on these topics:

- *Overview* on page 72
- *Downloading the Software* on page 73
- *Installing the System* on page 74
- *Checking Your Installation* on page 76
- *Using the Documentation* on page 79
- *Start-Up Tips* on page 80
- *Downloading Patches* on page 81

OVERVIEW

This chapter provides you with steps to install and test the system on your UNIX/Linux system. This overview discusses the contents of this chapter.

Note See *System Requirements* on page 2 for information about requirements.

Downloading the Software

You can download Documaker Server for UNIX/Linux by downloading the Oracle Documaker Standard Edition media pack from the Oracle Software Delivery Cloud web site. A media pack is an electronic version of the software products available to Oracle customers. The media pack contains ZIP files for a suite of related products on a single operating system so you can build a complete solution for your business. For more information, see *Downloading the Software* on page 73.

Installing the System

This topic explains how to install the system once you have downloaded the media pack. For more information see *Installing the System* on page 74.

Checking Your Installation

After you install your system, use the sample resources to check your installation. For detailed instructions, see *Checking Your Installation* on page 76.

Using the Documentation

This topic describes the documentation and explains how to use it. For more information, see *Using the Documentation* on page 79.

Start-Up Tips

This topic provides additional information on using the system in a UNIX/Linux environment. For more information, see *Start-Up Tips* on page 80.

Downloading Patches

This topic tells you how to download the latest patches from Oracle's web site. For more information, see *Downloading Patches* on page 81.

DOWNLOADING THE SOFTWARE

Oracle Insurance applications are available for download at the Oracle Software Delivery Cloud web site. The process includes:

- Logging in and agreeing to the terms and restrictions
- Searching for the applications you want to download
- Downloading those applications

Go to the Oracle Software Delivery Cloud web site to download Oracle Insurance applications:

<https://edelivery.oracle.com>

INSTALLING THE SYSTEM

Follow these steps to install the Oracle Documaker on a computer running AIX, Solaris, or Linux (Intel-based). The following fonts are also included in the media pack you download from the Oracle Software Delivery Cloud, see *Downloading the Software* on page 73 for more information.

- PCL and PostScript fonts
- AFP fonts
- Xerox fonts

Note Re-installation of the run time environment *does not* overwrite your INI files. If you install the system into an existing directory, the current library setup (INI files) is retained. If you install the system into a new directory, you must set up all of your libraries—just as you would for a new installation.

Before you begin

Make sure you have full directory access permissions (read/write/execute) to the directory into which you will be installing Documaker Server.

If you plan to run Documaker Server on AIX using DB2, make sure the following IBM DB2 variable is set:

```
DB2NOEXITLIST = On
```

Otherwise, any Documaker processes that load the DB2 driver, such as shared objects, GenData, GenPrint, GenWIP, or GenArc, will issue an error similar to the following upon exiting.

```
--- GenData Completed ---  
Illegal instruction(coredump)
```

This occurs because the DB2 client DSO's registered exit function is invalid at the time of exit because the DSO was unloaded. This affects the return code from the Documaker process. If you have customizations that check the exit codes of these processes, those customizations will be affected, even though the Documaker processes were successful within their tasks.

INSTALLATION

Note All paths and file names should be in lowercase.

Follow these steps to install the system:

1. Log onto the UNIX/Linux system using the account you want to use for the installation. By default the installation process installs the system into a subdirectory in your home directory with a name based on the release number.

For example, if you are installing the system using an account named *fap* that has a home directory of */home/fap*, the installation installs the system in the */home/fap/rel121* directory. You can, however, override the default.

2. Unzip the file you downloaded from the Oracle Software Delivery Cloud web site. Included in this file is a readme file and an installer. For each type of UNIX, the name of the installer is the same except for the extension. The name of the installer is as follows:

```
setuprterpl21pxx.zzz
```

Where *121* is the version number, *pxx* is the patch number, and *zzz* is one of these extensions:

This extension	Indicates
aix	AIX
spa	Solaris SPARC
lnx	Linux x86

3. Run the installer using either a Korn or a Bash shell prompt.

The initial installation screen describes the type of installation, displays the default directory the installation will use, and lets you override the installation location.

Note In case you encounter a message:

"WARNING: Couldn't flush system prefs: java.util.prefs.BackingStoreException: /etc/.java/.systemPrefs/com create failed." while installing Documaker connector, JAPAC Fonts or the full set of fonts, you can continue without concern.

CHECKING YOUR INSTALLATION

You can also download a sample MRLs (master resource libraries) from the Oracle Software Delivery Cloud web site. You will find these sample MRLs with the downloads for Oracle Documaker for the Microsoft Windows (32-bit) platform.

There are several sets of sample resources. The DMS1 sample resources are designed for the insurance market. You can check your installation by processing the resources in the sample resource library. The sample library includes an example directory which you can use to test against the results you get when you run the following tests.

These examples use scripts such as *run.sh* that pre-set the PATH and library PATH environment variables so the Dynamically Shared Objects (DSOs) and executables can be located at run time.

To manually run the programs or utilities (such as *gentrn*, *gendata*, *genprint*, *ovlcomp*, *fap2cfa*, and so on) at a shell command prompt, set these environment variables by prepending the PATH and the appropriate library PATH environment variables with the location of these files (such as LD_LIBRARY_PATH for Solaris and Linux and LIBPATH for AIX).

The rel121/bin path should be specified during the execution of *run.sh*.

.

Note For more information about the utilities included with the system, see the Utilities Reference.

Some implementations may set these environment variables using a .profile for the shell so they are pre-set automatically when a Korn shell is run. Please check with your system administrator for options.

Here are examples of the command-line shell prompt commands to enter if you install the system into the /home/fap/rel121 directory:

Solaris and Linux

```
PATH=/home/fap/rel121/bin:$PATH
LD_LIBRARY_PATH=/home/fap/rel121/bin:$LD_LIBRARY_PATH
export PATH LD_LIBRARY_PATH
```

AIX

```
PATH=/home/fap/rel121/bin:$PATH
LIBPATH=/home/fap/rel121/bin:$LIBPATH
export PATH LIBPATH
```

TESTING WITH THE DMS1 SAMPLE RESOURCES

The system includes a shell script file (*run.sh*) which lets you process the DMS1 sample resources. There are specific options and arguments you can use to select the Documaker Server modules (*gentrn*, *gendata*, *genprint*, and *genarc*) in single or multiple step mode. The DMS1 sample resources only outputs PCL printer files. To set up other printer types, refer to the [Documaker Administration Guide](#).

The shell script file executes the Documaker Server modules (gentrn, gendata, genprint, and genarc) in the appropriate mode for the PCL printer type and copies the appropriate INI files and printer overlays into the proper directories (see the [Documaker Administration Guide](#) for more information).

Processing the DMS1 Sample Resources

To run the system through a processing cycle, follow these steps:

1. Download the Sample MRLs from the Oracle Software Delivery Cloud and install the DMS1 sample resources into the mstrres\ directory.
2. Go to the mstrres/dms1 directory where you installed the MRL examples. At the Korn or Bash shell prompt, enter this command:

```
run.sh
```

3. Follow the prompts that appear on your screen.

Printing the DMS1 Output

After running the system, the printer ready files can be found in the ../dms1/data directory.

Note See the [Documaker Administration Guide](#) for detailed instructions on printing to AFP, Metacode, PCL, or PostScript printers.

Testing WIP with DMS1

The DMS1 library is set up to place two transactions into WIP. Follow these steps to perform the test:

1. At a UNIX/Linux Korn or Bash shell prompt, go to the mstrres/dms1 directory, enter genwip command..
2. Prerequisite:
CD1-7spDocumakerStudioFullSystemW32RelXXXpYYb#####.exe is installed on a windows machine.
3. Then copy the entire MRL to your Windows machine and run Documaker Desktop (afemnw32.exe) from the DMS1 root directory.
4. Choose the WIP, WIP List option.
5. Select transaction 2234567 and click Ok.

Viewing Archived DMS1 Forms

Optionally, you can archive the bills previously printed by following these steps:

1. At a UNIX/Linux Korn or Bash shell prompt, go to the mstrres/dms1 directory and enter this command:

```
genarc
```

2. Then copy the entire MRL to your Windows machine and run Documaker Desktop (afemnw32.exe) from the DMS1 root directory.
3. Choose the Retrieve, Formset option to view the archived bills.

USING THE DOCUMENTATION

After you download Oracle Documaker and unzip the file you downloaded, you will find a readme file that provides an overview, late-breaking information, and links to the documentation. For more detailed instructions on downloading the documentation, see the following topic.

The documentation available for this release is in Portable Document Format (PDF) files which you can view or print using the Adobe® Acrobat™ Reader™. In the documentation file you download from the OTN web site, you will find the various documentation that accompanies this release, such as:

- Documaker Administration Guide
- Documaker Studio User Guide
- Utilities Reference
- Rules Reference
- DAL Reference
- Release Notes

For best results, make sure you have the latest version of Acrobat Reader. You can download the free reader at Adobe's web site: www.adobe.com.

UPDATING YOUR DOCUMENTATION

You can download the latest Oracle Insurance documentation at the Oracle Technology Network (OTN) web site. The process includes:

- Going to the Oracle Insurance Documentation page on the OTN site
- Searching for the application documentation you want to download
- Downloading that documentation

To download Oracle Insurance application documentation, go to the Oracle Insurance Documentation page:

<http://www.oracle.com/technetwork/documentation/insurance-097481.html>

START-UP TIPS

Here are some additional tips to help you get your system up and running:

Using long file names

File names in UNIX and Linux can be up to 256 characters long or longer. You should, however, limit the names of resource files, such as FAP and LOG files, to 100 characters or less, excluding the path.

Documaker Server software imposes no limits on the length of the names use for the output files, such as print streams, so you can name those files however you like.

plugin.ttf font file

Make sure the plugin.ttf font file is not in your font directory. This font file is only applicable on Windows platforms.

DOWNLOADING PATCHES

You can download the latest Oracle software patches at the My Oracle Support web site. The process includes:

- Going to the My Oracle Support site (requires registration)
- Searching for the patches you want to download
- Downloading those patches

To download Oracle software patches, go to the My Oracle Support web site:

<https://support.oracle.com>

Chapter 3

Installing Documaker on Windows

This chapter provides detailed information on how to install and configure the system in a Windows environment.

You will find information on these topics:

- *Overview* on page 84
- *Downloading the Software* on page 85
- *Installing the System* on page 86
- *Checking Your Installation* on page 95
- *Using the Documentation* on page 98
- *Downloading Patches* on page 99

OVERVIEW

This chapter provides you with steps to install and test the system on your Windows system. This overview discusses the contents of this chapter.

Note See *System Requirements* on page 2 for information about system requirements.

Downloading the Software

You can download Documaker Server for Windows by downloading the Oracle Documaker Standard Edition media pack from the Oracle Software Delivery Cloud web site. A media pack is an electronic version of the software products available to Oracle customers. The media pack contains ZIP files for a suite of related products on a single operating system so you can build a complete solution for your business. For more information, see *Downloading the Software* on page 85.

Installing the System

This topic explains how to install the system. For more information see *Installing the System* on page 86.

Checking Your Installation

After you install your system, use the sample resources to check your installation. For detailed instructions, see *Using the Documentation* on page 98.

Using the Documentation

This topic describes the on-line documentation and explains how to use it. For more information, see *Using the Documentation* on page 98.

Downloading Patches

This topic tells you how to download the latest patches from Oracle's web site. For more information, see *Downloading Patches* on page 99.

DOWNLOADING THE SOFTWARE

Oracle Insurance applications are available for download at the Oracle Software Delivery Cloud web site. The process includes:

- Logging in and agreeing to the terms and restrictions
- Searching for the applications you want to download
- Downloading those applications

Go to the Oracle Software Delivery Cloud web site to download Oracle Insurance applications:

<https://edelivery.oracle.com>

INSTALLING THE SYSTEM

Follow these steps to install the Oracle Documaker on a personal computer running Windows. The following fonts are also included in the media pack you download from the Oracle Software Delivery Cloud, see *Downloading the Software* on page 85 for more information.

- PCL and PostScript fonts
- AFP fonts
- Xerox fonts

Included with the media pack as a separate download are sample resources. The DMS1 sample resource library includes sample data.

Note Re-installation *does not* overwrite your INI files.

If you install the system into an existing directory, the current library setup is retained. If you install the system into a new directory, you must set up all of your libraries—just as you would for a new installation.

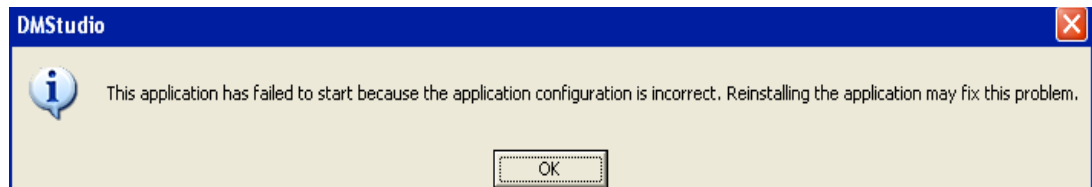
If you need to uninstall Oracle Documaker or one of its registered add-ins, such as Oracle Documaker Add-In for Microsoft Word, use Control Panel's Add or Remove Programs option.

Before you begin

Keep in mind...

- Be sure to log onto your computer as *Administrator* and make sure you have full directory access permissions to the \all users folder, as well as to the registry.
- Admin rights and permissions are required to run the installer and uninstaller because of need for ability to modify registry and system.
- When using NTFS drive formats, the logon name must be assigned all privileges and permissions to the Windows operating system folder and all sub-folders within, as well as to make changes to the Windows registry.
- The installation wizard verifies that you have the Microsoft Visual C++ Redistributable Package (vcredist_x86.exe) before beginning the installation.

This program installs run-time components of the Visual C++ libraries, which are required to run Documaker Studio. If you try to run Studio without the Microsoft Visual C++ Redistributable Package, you will get the following message:



If these run-time components are not installed on your computer, download and install the latest Microsoft Visual C++ 2008 Redistributable Package (x86) from Microsoft's Download Center:

<http://www.microsoft.com/downloads/>

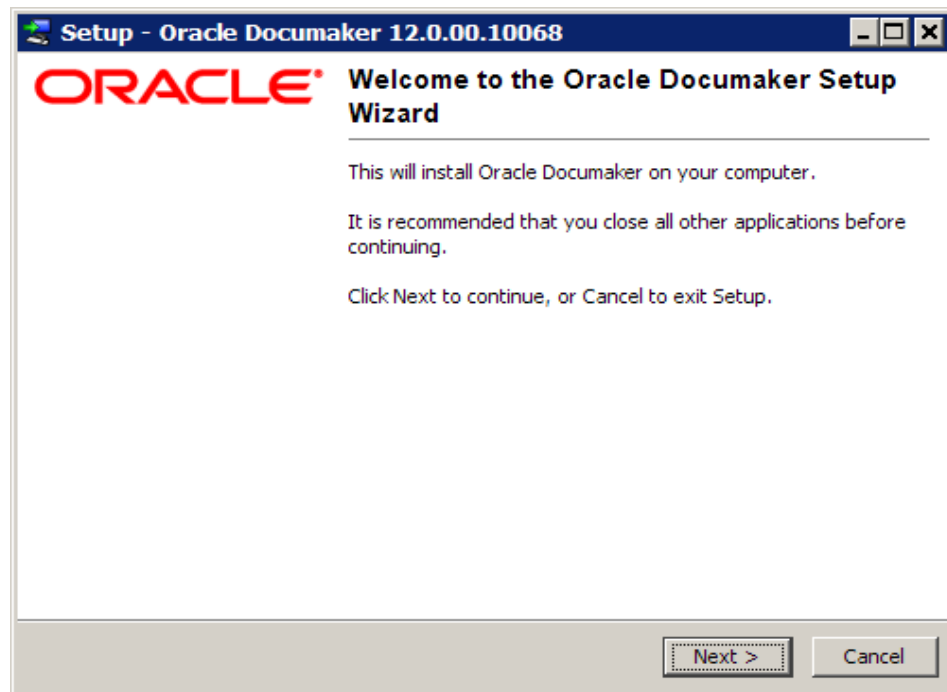
INSTALLATION

Follow these steps to install the system:

Note If you plan to install the optional Documaker Add-In for Microsoft Word, review *Installing the Documaker Add-In for Word* on page 89 and make sure Word is closed before you begin.

If you plan to install the optional IStream Migration Utility, review *Installing the IStream Migration Utility* on page 92 and make sure Word is closed before you begin.

1. From the directory into which you downloaded the Oracle Documaker Standard Edition media pack, unzip the media pack, locate and double click on the following program:



`CD1-7spDocumakerStudioFullSystemW32Rel(version and patch number).exe`

The installation wizard starts.

2. Follow the instructions that appear on your screen to install the software. The installation creates folders on your Start\Programs menu.

Note The short cut icon is not automatically created as a part of the installation process. This can be manually created to display in the Start\Programs menu.

3. At the end of the installation process on a Windows 7 -64bit environment, the system may display the Microsoft Visual C++ 2008 Redistributable Setup dialog for Maintenance mode

Select one of the options below:

- Repair - Repair Microsoft Visual C++ 2008 Redistributable to its original state.
- Uninstall - Uninstall Microsoft Visual C++2008 Redistributable from this computer.

Choose Repair to go ahead with the installation process.

The system does not display this dialog when performing installations in a Win XP environment.

Note If you uninstall the MS VC Redistributable on Windows 7 prior to running the Documaker install, you will be prompted to click through the license agreement for the redistributable. However, both the installations finish successfully.

In a scenario when the redistributable is still installed post Documaker Uninstallation, Maintenance: Repair/uninstall screen appears when you re-install Documaker.

After you install the system, restart your computer so the various system settings can take effect. If you install the software on a network, you will need to set up icons and a program group for each user. Refer to your operating system's documentation for information on how to do this.

Installing the Documaker Add-In for Word

The Documaker Add-In for Microsoft Word is an optional module that lets you create Documaker documents using Microsoft Word.

Note Before installing the Documaker Add-In for Word, you must make sure Microsoft Word 2007 or Word 2010 is installed on the computer.

If you have older versions of Word installed on the computer that will run the Add-In, uninstall those versions of Word *before* you install the Add-In.

After you install Studio and the Documaker Add-In, be sure to run Word *before* you import files into Studio, otherwise the Documaker Add-In will not load when Studio starts Word as part of the import process. If the Documaker Add-In does not appear, see *If the Documaker Add-In does not Appear in Word* on page 91 for more information.

Add-In requirements

In addition to the basic Documaker Windows system requirements outlined on page 7, you need the following applications to run the Add-In:

Item	Version
Microsoft .NET Framework	Version 3.5. You can have other versions of .NET Framework too, but you must have version 3.5.
Microsoft VSTO Runtime	Version 3.0 with Service Pack 1, or higher

Review the Microsoft Download Center site for any additional requirements associated with these applications:

<http://www.microsoft.com/downloads/>

Note that the Microsoft .NET Framework must be installed before the VSTO Runtime.

In addition, Microsoft .NET Programmability Support for Word must be enabled. For more information, see *Checking for .NET Programmability Support* on page 93.

Note You can only install one copy of the Documaker Add-in for Word on a computer. For best results, use the same version of Studio and the Add-In. For example, a new feature in version 12.1 of the Add-In might not be recognized in an earlier version of Studio.

Configuring resource options

During installation, the Setup wizard lets you override default resource location options. You can override these locations at this time or you can change the locations as needed while working in the Add-In by choosing Options from the Documaker ribbon.

The Setup wizard lets you set these configuration options:

Option	Description
Workspace Service URL	(Optional) Enter the default web address for the Docupresentment web service. This service can be used to download workspace definition files.
Workspace Definition Folder	Enter the default location for storing workspace definition files (WDF).
Resource File	Enter the name of the local resource file for this installation. This XML file contains all of your locally created resources. The default is: <code>localresources.xml</code> If this file is missing, the system generates a new copy of it.
Graphics Folder	Enter the default location for storing graphic files.
Default Graphic	Enter the file name and extension for the default graphic, also called the placeholder image. The Add-In displays this graphic if it cannot find a specified graphic.
Log Folder	Enter the default location for storing log files. These files contain error and other operational messages which can be useful when diagnosing problems.

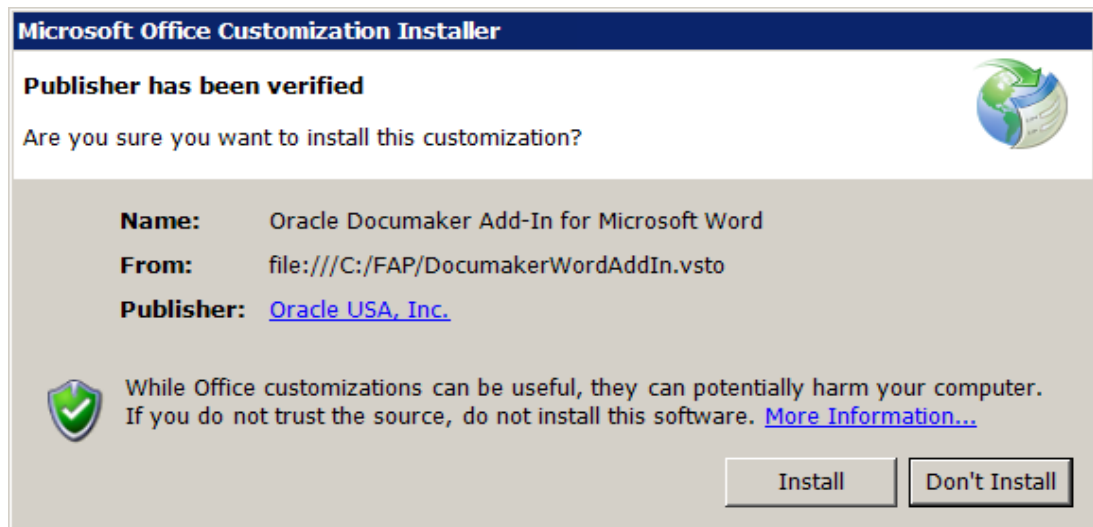
Updating Microsoft Word

For best results, make sure you have installed the latest updates to your version of Microsoft Word.

To check for updates to	Follow these steps
Word 2007	From the Office menu, click Word Options. On the left pane, click Resources, then click the Check for Updates button.
Word 2010	Choose File, Help and then select the Check for Updates option.

Starting the Documaker Add-In for Word

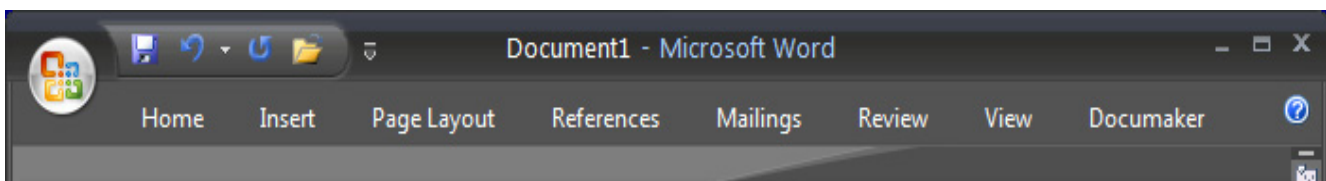
After you install the Documaker Add-In for Microsoft Word, the next time you start Microsoft Word, you will see this message if you have Internet access:



If you do not have Internet access, this message appears instead:

Publisher cannot be verified.
Are you sure you want to install this customization?
To continue, click Install.

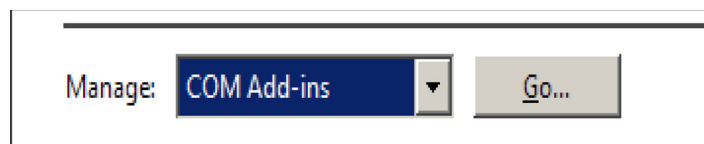
Click Install to continue. Then make sure the Documaker tab appears as the last tab on the Microsoft Word ribbon.



If the Documaker Add-In does not Appear in Word

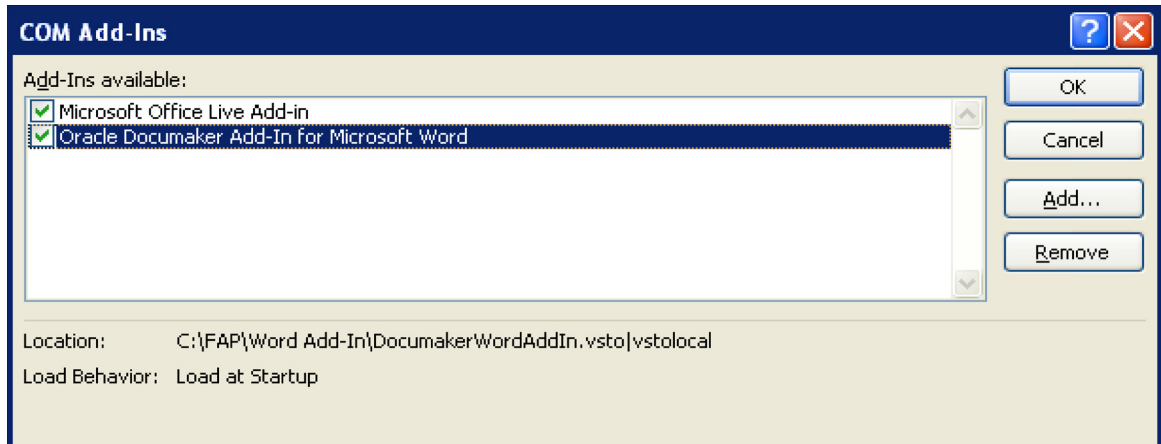
If the Documaker Add-In does not appear after you install Documaker and open Microsoft Word, you probably need to install the certificate for the Add-In. Follow these steps:

1. Click the Microsoft Office button and select Word Options.
2. On the left pane of the Word Options window, click Add-Ins.
3. At the bottom of the Word Options window, for Manage select COM Add-ins, then click Go.



4. Click the Oracle Documaker Add-In for Microsoft Word entry to enable it. Then click Ok.

Note If the Oracle Documaker Add-In for Microsoft Word entry does not appear, see *If the Documaker Add-In does not appear as an inactive add-in* on page 92.



The Documaker tab should now appear in Word. If it does not, close and reopen Word.

If the Documaker Add-In does not appear as an inactive add-in

If Oracle Documaker Add-In for Microsoft Word does not appear in the Inactive Application Add-Ins section follow these steps:

1. Scroll down the list of add-ins until you see the Disabled Application Add-Ins section.

If Oracle Documaker Add-In for Microsoft Word appears in the Disabled Application Add-Ins section, complete these remaining steps:
2. At the bottom of the Word Options window, on the Manage list, select Disabled Items, then click Ok.
3. Click the Oracle Documaker Add-In for Microsoft Word entry to highlight it. Then click Enable.

The Documaker ribbon should now appear in Microsoft Word. If it does not, close and reopen Microsoft Word.

Installing the IStream Migration Utility

The IStream Migration Utility is an optional utility that helps you migrate IStream model documents to Documaker. In addition to the basic Documaker Windows system requirements outlined on page 7, you need the following applications to run this utility:

Item	Version
Operating system	Windows Server 2003, XP, Vista, or Windows 7
Microsoft Word	Word version 2003, with Service Pack 2 or higher Word version 2007, with Service Pack 1 or higher
Microsoft .NET Framework	Version 2.0 or higher

Review the Microsoft Download Center site for any additional requirements associated with these applications:

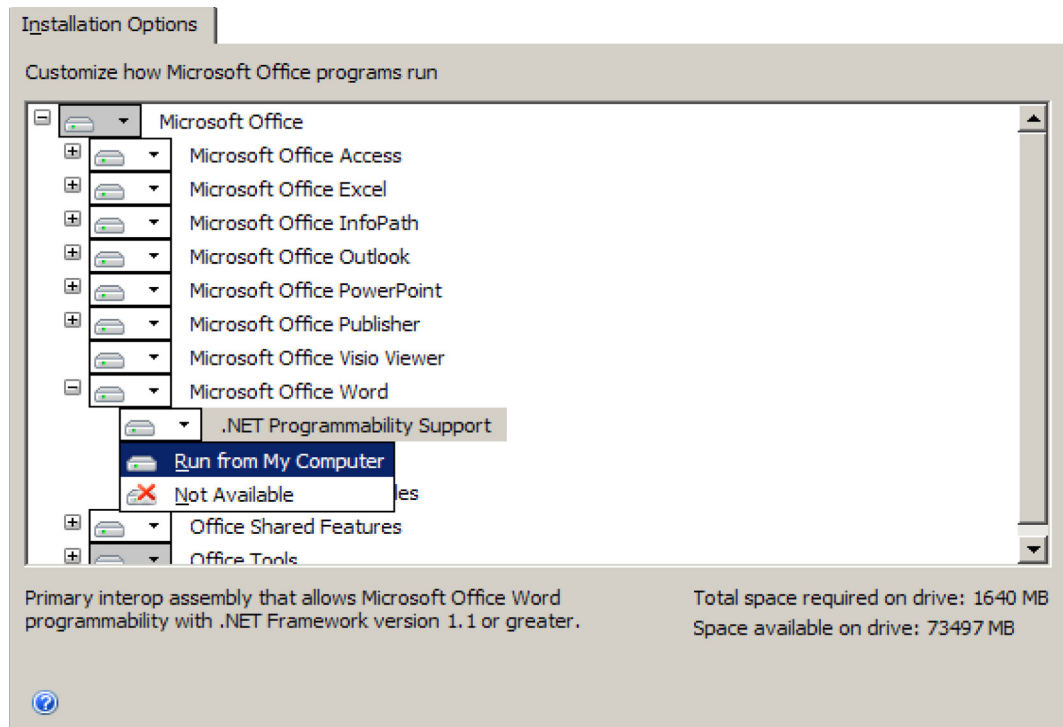
<http://www.microsoft.com/downloads/>

In addition, Microsoft .NET Programmability Support for Word must be enabled. For more information, see *Checking for .NET Programmability Support* on page 93. For information on using the IStream Migration Utility, see the Utilities Reference.

Checking for .NET Programmability Support

Both the Documaker Add-In for Word and the IStream Migration Utility require Microsoft .NET Programmability Support for Microsoft Word to be available. This feature is typically installed and enabled when you install Word. If, however, you need to check, follow these steps:

1. From the Control Panel, choose the Add or Remove Programs option, then highlight Microsoft Word or Microsoft Office and click Change.
2. Select the Add or Remove Features option and click Continue.
3. On the Installation Options tab, expand Microsoft Office Word and make sure Microsoft .NET Programmability Support is available, as shown here:



4. Once you make sure the Run from My Computer option is selected, click Continue and exit Control Panel.

CHECKING YOUR INSTALLATION

You can also download a sample MRLs (master resource libraries) from the same Oracle Software Delivery Cloud page you downloaded Documaker. The DMS1 sample resources are designed for the insurance market. You can check your installation by processing the resources in the sample resource library. The sample library includes an example directory which you can use to test against the results you get when you run the following tests.

TESTING WITH THE DMS1 SAMPLE RESOURCES

The system includes several batch files which let you process the DMS1 sample resources. There is a specific batch file that lets you select Documaker Server modules (GenTrn, GenData, GenPrint, and GenArc) in single or multiple step mode for each type of printer, such as PCL, PostScript, Xerox Metacode, and AFP.

The batch files execute the Documaker Server modules (GenTrn, GenData, GenPrint, and GenArc) in the appropriate mode for the specified printer type and copies the appropriate INI files and printer overlays into their proper directories (see the [Documaker Administration Guide](#) for more information).

Processing the DMS1 Sample Resources

To run the system through a processing cycle, follow these steps:

1. Download the Sample MRLs from the Oracle Software Delivery Cloud and place the DMS1.ZIP file (which contains the DMS1 sample resources), into the mstrres\ directory.
2. Unzip the contents of DMS1.ZIP into the mstrres\ directory. The resulting directory structure should be mstrres\dms1\ . The arc, bdfilib, data and the other directories fall a level below.

Note When you unzip the DMS1.ZIP file from the mstrres directory, it will place the DMS1 sample resources in the dms1 directory.

3. Go to the \dms1 directory. At the operating system prompt, enter this command:

run.bat

This command installs the DMS1 sample resources into the \dms1 directory.

Printing the DMS1 Output

To print the PCL output you generated when you ran the system, follow the instructions below.

Note See the [Documaker Administration Guide](#) for detailed instructions on printing to AFP, Metacode, PCL, or PostScript printers.

Copy the REL121.FNT font file located in \mstrres\finres\deflib directory to the printer along with the PCLBAT1 file, which is located in the \data directory. You can do this by entering the following command from the \dms1 directory:

```
copy /b pclbat1.pcl lpt1
```

Where *pclbat1.pcl* is the name of a printstream created when you ran run.bat to install the DMS1 sample resources and *lpt1* is the printer.

Note Run.bat also creates the pclbat2.pcl and pclbat3.pcl printstream files.

Testing WIP with DMS1

The DMS1 library is set up to place two transactions into WIP. Follow these steps to perform the test:

1. At an operating system prompt, go to mstrres\dms1 directory and enter this command:

```
genwpw32
```

2. Start the Entry module by entering:

```
afemnw32 /mode=wip
```

3. Choose the WIP, WIP List option.
4. Select transaction 2234567 and click Ok.

Viewing Archived DMS1 Forms

Optionally, you can archive the bills previously printed by following these steps:

Note If you are licensed for Archive Retrieval the following steps apply. If, however, you *are not* licensed for Archive Retrieval, do not perform these steps.

1. At an operating system prompt, go to mstrres\dms1 directory and enter this command:

`genacw32`
2. Start the Entry system by entering this command:

`afemnw32 /mode=retrieve`
3. Choose the Retrieve, Formset option. The Retrieve Document window appears. Enter **Sampco** in the Company field and click Ok to view the archived bills.

USING THE DOCUMENTATION

After you download Oracle Documaker and unzip the file you downloaded, you will find a readme file that provides an overview, late-breaking information, and links to the documentation. For more detailed instructions on downloading the documentation, see the following topic.

The documentation available for this release is in Portable Document Format (PDF) files which you can view or print using the Adobe® Acrobat™ Reader™. In the documentation file you download from the OTN web site, you will find the various documentation that accompanies this release, such as:

- Documaker Administration Guide
- Documaker Studio User Guide
- Utilities Reference
- Rules Reference
- DAL Reference
- Release Notes

For best results, make sure you have the latest version of Acrobat Reader. You can download the free reader at Adobe's web site: www.adobe.com.

UPDATING YOUR DOCUMENTATION

You can download the latest Oracle Insurance documentation at the Oracle Technology Network (OTN) web site. The process includes:

- Going to the Oracle Insurance Documentation page on the OTN site
- Searching for the application documentation you want to download
- Downloading that documentation

To download Oracle Insurance application documentation, go to the Oracle Insurance Documentation page:

<http://www.oracle.com/technetwork/documentation/insurance-097481.html>

DOWNLOADING PATCHES

You can download the latest Oracle software patches at the My Oracle Support web site. The process includes:

- Going to the My Oracle Support site (requires registration)
- Searching for the patches you want to download
- Downloading those patches

To download Oracle software patches, go to the My Oracle Support web site:

<https://support.oracle.com>

INDEX

A

- access permissions, 74, 84
- Administrator
 - logging in as, 84
- Adobe Acrobat Reader, 68, 78, 94
- Advanced Function Presentation, 19
- AFP
 - batch files for RPEX1, 91
 - configuring, 12
 - fonts, 74, 84
- AFP2MVS utility, 62
- AFPCOPY utility, 62
- AFPFMDEF utility, 26
- AFPFMDFX job, 19
- AIX
 - IBM DB2 variable, 74
- Allocating AFP-related Datasets, 18
- ALLOCDB2 and COMPDB2, 30
- and host archive, 29
- APPIDX table, 29, 33
- Archival control group, 31
- ARCHIVE table, 29
- archived bills
 - viewing from RPEX1, 93
- ArcRet control group, 30

B

- BARR interface, 52
- BARR2MVS utility, 63
- BARRWRAP utility, 63
- BARRWRPX utility, 52
- batch files
 - for processing RPEX1, 91

C

- changing
 - the FSISYS.INI file (AFP), 20
 - the JCL, 20, 33
 - the JCL (Metacode), 18
 - the Xerox JSL, 12
- checking
 - the JCL, 40
 - your FSIUSER and FSISYS settings, 37
 - your installation, 37
- CompileInStream option, 16
- configuring
 - host archive using DB2, 29
 - your print environment, 12
 - your system for AFP printers, 18
- CPCNV utility, 63

- CreateIndex option, 32
- CreateTable option, 32
- Creating DB2 Objects, 29
- CurrentPackageSet option, 8

D

- DAP procedure, 18, 20
 - GenPrtX job, 20
- DAPRUN job, 18, 21, 40
- Database option, 32
- DB2
 - Command Processor program, 49
 - database management system, 29
 - DefaultTag option, 32
 - table, 32
- DB2_FileConvert control group, 30, 32
- DB2NOEXITLIST variable, 74
- DBHANDLER
 - DB2 control group, 32
- DefaultTag option, 32
- directory access permissions, 84
- DJDEIden entry, 13
- DJDEOffset entry, 13
- DJDESkip entry, 13
- DMS1
 - testing with, 76, 91
- Documaker Add-in for Microsoft Word, 86
- documentation, 68, 78, 94
 - downloading, 68, 78, 94
- DownloadFAP option, 16
- DSNHC procedure, 30

E

- encrypted passwords, 66

F

- FAP2AFP utility, 63
- FAP2CFA utility, 63, 64
- FAP2MET utility, 16, 63
- FAP2OVL utility, 20, 64
- FDT2CFA utility, 64
- file names, 79
 - case, 74
- FIXOFFS utility, 64
- FNT files
 - for logos, 17
- FONTLIB PDS, 26
- fonts
 - AFP, 19
 - Metacode, 17

FormDef, 19

FSISYS

- changing the JSL for Metacode printers, 12

- checking your settings, 37

FSIUSER

- archive settings, 30

- checking your settings, 37

FSIVER utility, 64

G

GenArc program, 29, 77, 91

- and DSN, 49

- defined, 65

GenArcX member, 33

GenData program, 65, 77, 91

GENDATAL job, 11

GENERAFF job, 21

GENERAFF member, 52

GENERXER job, 18

GENERXER member, 52

GENLOGOS job, 18

GenPrint program, 65, 77, 91

GENPRTL job, 11

GenPrtX job, 20

GenTrn program, 65, 77, 91

graphics card, 17

I

IBM

- DB2 database management system, 29

- Print Services Facility, 19

IEBGENER, 52

ImageOpt entry, 13

IMG files

- graphics cards, 17

INI files

- installation, 74

installing the system, 74, 84

IStream Migration Utility, 89

J

JDName entry, 13

JDLCode entry, 13

JDLData entry, 13

JDLHost entry, 13

JDLName entry, 13

JDLROffset entry, 14

JDLRPage entry, 14

JDLRStack entry, 14

Job Description Entry, 13

JSL

- changing the Xerox, 12

- requirements for Metacode, 13

- viewing, 12

L

libraries, 58

- checking your installation, 76, 91

linearized PDF files, 26

LOADDAP job

- listing, 53

LOADFONT job, 2, 5, 53

loading fonts onto a Metacode printer, 17

LOG2PSEG utility, 19, 65

logos

- loading, 17

- making resources available, 17

long file names, 79

lowercase, 74

M

making printer resources available

- for AFP printers, 19

- for Metacode printers, 17

Metacode

- batch files for RPEX1, 91

- configuring, 12

Microsoft

- .NET Programmability Support, 90

modifying the FSISYS.INI file, 30

MultiFileLog option, 25

MultiFilePrint rule, 25

MVS

- utilities, 62

N

NTFS drives, 84

O

ODBC

- DefaultTag option, 32

Oracle Software Delivery Cloud, 4, 73, 83

OTN web site, 68, 78, 94

overlays, 19

Overview, 72

P

page segments, 19

page-at-a-time downloading, 26

passwords, 66

patches

- downloading, 69, 80, 95

paths

- case, 74

PCL

- batch files for RPEX1, 91

- fonts, 74, 84
- PCLBAT1 file, 92
- PDF
 - fonts in print streams, 26
 - print driver, 26
 - using DAL scripts, 25
 - using the MultiFilePrint rule, 25
- PDFKEY utility, 66
- PDS2VSAM utility, 66
- permissions, 74, 84
- plugin.ttf font file, 79
- Port option, 15, 20, 26
- PostScript
 - batch files for RPEX1, 91
 - fonts, 74, 84
 - fonts on z/OS, 26
 - print streams, 23
- pre-compiling FAP files
 - into Metacode files, 15
 - into overlays, 20
- Printer control group, 15, 20
- printer resources, 17
- Printer1 control group, 15, 20
- PrinterInk entry, 14
- printing
 - IMG files, 17
 - RPEX1 output, 77, 92
 - with DAPRUN, 52
- privileges, 84
- processing
 - RPEX1 sample resources, 77, 91
- PrtType control group
 - AFP, 20
 - XER, 12, 14
- PSF
 - and AFP, 19
- R**
- RCP2VSAM utility, 66
- registry, 84
- REL121SM.FNT font file, 92
- RPEX1
 - installation, 84
- run.sh, 76
- RunMode control group, 16
- running, 30
 - DAPRUN, 41
 - GenArcX, 49
 - GenTrnX, GenDataX, and GenPrtX, 41
 - GenTrnX, GenDataX, and GenPrtX separately, 40
- S**
- SAMPAFP utility, 66
- SEQ2KSDS utility, 64
- software
 - downloading, 4, 73, 83
- SQL databases
 - DefaultTag option, 32
- start-up tips, 79
- SYSPRINT
 - running GenArcX, 49
- T**
- testing
 - with RPEX1, 76, 91
- TRANSLAT utility, 66
- Trigger2Archive control group, 33
- TrueType
 - fonts on z/OS, 26
- TSO Terminal Monitor Program, 49
- U**
- uninstall, 84
- UNIT names, 40
- updates
 - downloading, 69, 80, 95
- using
 - on-line documentation, 78, 94
- V**
- viewing
 - archived RPEX1 forms, 77, 93
- viewing the JSL, 12
- VSAM datasets, 64
- X**
- XERDNLD utility, 66
- XERLOGOS job, 18
- Xerox
 - batch files for RPEX1, 91
 - configuring, 12
 - font names, 17
 - fonts, 74, 84