

**Oracle® Documaker**  
**IStream to Documaker Migration**  
**User Guide**

**Edition 1.0.1**  
**April 2016**

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this software or related documentation is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. 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, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

This software is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications which may create a risk of personal injury. If you use this software in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure the safe use of this software. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software in dangerous applications.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

This software and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Where an Oracle offering includes third party content or software, we may be required to include related notices. For information on third party notices and the software and related documentation in connection with which they need to be included, please contact the attorney from the Development and Strategic Initiatives Legal Group that supports the development team for the Oracle offering. Contact information can be found on the Attorney Contact Chart.

The information contained in this document is for informational sharing purposes only and should be considered in your capacity as a customer advisory board member or pursuant to your beta trial agreement only. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described in this document remains at the sole discretion of Oracle.

This document in any form, software or printed matter, contains proprietary information that is the exclusive property of Oracle. Your access to and use of this confidential material is subject to the terms and conditions of your Oracle Software License and Service Agreement, which has been executed and with which you agree to comply. This document and information contained herein may not be disclosed, copied, reproduced, or distributed to anyone outside Oracle without prior written consent of Oracle. This document is not part of your license agreement nor can it be incorporated into any contractual agreement with Oracle or its subsidiaries or affiliates.

This documentation is in prerelease status and is intended for demonstration and preliminary use only. It may not be specific to the hardware on which you are using the software. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to this documentation and will not be responsible for any loss, costs, or damages incurred due to the use of this documentation.

ORACLE CONFIDENTIAL. For authorized use only. Do not distribute to third parties.

# CONTENTS

<i>CONTENTS</i> .....	3
Figures .....	4
Tables .....	4
<b>GETTING STARTED</b> .....	<b>6</b>
Prerequisites for this Document .....	6
Documentation Audience .....	6
<i>About this Guide</i> .....	7
<i>IStream Products to Documaker Products</i> .....	8
<i>Oracle Documaker Overview</i> .....	10
<i>Migration Steps</i> .....	14
<b>STEP 1. REQUIREMENTS STUDY</b> .....	<b>15</b>
<i>Current and Proposed Workflows</i> .....	16
<i>Analysis of Model Documents and Data</i> .....	17
Model Document Requirements .....	18
Related Documentation .....	20
<b>STEP 2. PROJECT PLANNING</b> .....	<b>21</b>
<i>Resource Requirements</i> .....	22
<i>System Requirements</i> .....	23
<i>Oracle Education</i> .....	24
<i>Oracle Support</i> .....	25
<b>STEP 3. MIGRATION OVERVIEW</b> .....	<b>26</b>
<i>Pre-migration Planning</i> .....	27
<i>Migration Considerations</i> .....	28
<i>Migration Process and Checklist</i> .....	29
<b>STEP 4. MODEL DOCUMENT MIGRATION UTILITY</b> .....	<b>31</b>
<i>Migration Utility Parameters</i> .....	32
<i>Migration Output Files</i> .....	34
<i>Migration Reports</i> .....	35
<i>Run Migration Utility</i> .....	37
<b>STEP 5. DOCUMAKER PRE-IMPORT TASKS</b> .....	<b>38</b>
<i>Recommended Directories</i> .....	39
<i>Create Workspace</i> .....	40
Related Documentation .....	40
<i>Font Conversion</i> .....	41
Importing Fonts .....	43
Related Documentation .....	46
Best Practice .....	46
<b>STEP 6. DOCUMAKER IMPORT</b> .....	<b>47</b>
<i>Import Common Field Dictionary</i> .....	48
Best Practice .....	48
Create FDB Import File .....	49
Related Documentation .....	50
<i>Import Data Extract Dictionary</i> .....	51
Best Practice .....	51
IStream Rules Map To .....	52
Create Data Dictionary .....	54
Related Documentation .....	54
<i>Converting RTF Files</i> .....	55
<i>Import Forms List and Form Attachment Rules</i> .....	59
Best Practice .....	59

<b>STEP 7. STUDIO UPDATES.....</b>	<b>60</b>
<i>Creating the Table of Contents Document .....</i>	<i>61</i>
<i>Review and Modify Graphics .....</i>	<i>63</i>
<i>Section Position.....</i>	<i>64</i>
<i>Converting Conditional Field Logic .....</i>	<i>65</i>
Best Practice .....	66
Related Documentation.....	66
<b>STEP 8. DOCUMAKER SYSTEM CONFIGURATION.....</b>	<b>67</b>
<i>Define Users and Responsibilities.....</i>	<i>68</i>
Best Practice .....	68
Adding Users .....	69
Changing Passwords.....	69
Adding Security .....	69
Related Documentation.....	69
<i>Configuring Distribution Channels and Recipients .....</i>	<i>70</i>
Adding Distribution Channels .....	71
Adding Recipients .....	72
Related Documentation.....	72
Document Rendering .....	73
Adding Document Rendering Options.....	74
Related Documentation.....	74
Print Types.....	75
<i>Configure Options.....</i>	<i>77</i>
Related Documentation.....	77
<i>Configure System Files.....</i>	<i>78</i>
Related Documentation.....	78
<i>IStream to Documaker Scenarios .....</i>	<i>79</i>
Scenario 1: Typical Publisher Scenario .....	79
Scenario 2: Scenario using Dispatcher Components.....	83
<b>STEP 9. TEST AND COMPARE .....</b>	<b>85</b>
<i>Run Documaker to Produce PCL and PDF Files .....</i>	<i>86</i>
<b>STEP 10. DEPLOYMENT .....</b>	<b>88</b>
<i>Library Deployment .....</i>	<i>89</i>
Deployment Steps.....	89
<b>GLOSSARY OF TERMS.....</b>	<b>90</b>

## FIGURES

Figure 1: IStream Products .....	9
Figure 2: Complete Document Automation Solution.....	10
Figure 3: Documaker Overview .....	12
Figure 4: Oracle Documaker Architecture .....	13
Figure 5: Migration Steps.....	14
Figure 6: Project Planning Tasks.....	21
Figure 7: IStream to Documaker Migration .....	29
Figure 8: IStream Migration Utility.....	32
Figure 9: Font ID Naming Standards .....	42

## TABLES

Table 1: Sample IStream Form Types and Features.....	17
Table 2: Requesting a CSI Number .....	25
Table 3: Registration on the Oracle Support Site .....	25
Table 4: Migration Checklist.....	30
Table 5: Migration Output Files Map to Studio Files.....	34

Table 6: IStream Rules Map to Documaker Rules .....52

Table 7: Completed Migration Checklist .....87

# Getting Started

Welcome to the IStream to Documaker Migration Guide for migrating IStream deployments to Oracle Documaker. This guide describes how to use the migration utility, what files you will need to update after running the migrating utility, and identifies Documaker configuration files. In addition, the guide maps product-by-product the IStream products to Documaker products and points out the integration requirements. The guide provides you with the information you will need to develop, plan, and execute your migration project.

## PREREQUISITES FOR THIS DOCUMENT

Before reading this guide, you should complete the IStream to Documaker Questionnaire, located at the end of this document.

## DOCUMENTATION AUDIENCE

This document is intended for the following audiences:

**Documaker trained specialists** — Project team members who will focus on helping IStream customers migrate to Oracle Documaker. Project team members will have an in-depth working knowledge of Documaker Server, Documaker Desktop, and other Documaker products.

**IStream customers** — Individuals who are responsible for the IStream to Documaker Migration project for their organization. Customer project team members will have an in-depth working knowledge of IStream products and have taken or plan to take Oracle Documaker training.

## ABOUT THIS GUIDE

This guide provides an overview and explanation of the specific steps and requirements needed to migrate IStream model documents to Oracle Documaker. This guide is designed to explain the IStream to Documaker framework and to document the migration steps to deployment.

This guide begins with a high-level description of Documaker; including an overview of Oracle's Document Automation solution, an overview of the Documaker architecture and then gradually dives into how to migrate IStream model documents to Documaker. This guide provides information on the following topics:

- How to map IStream products to Documaker products
- An overview of Documaker
- A high-level IStream to Documaker migration checklist
- An IStream to Documaker step-by-step migration guide
- Best practices
- A guide to assist with system setup and library development
- References to related documentation for each topic
- A glossary of Documaker terminology

# ISTREAM PRODUCTS TO DOCUMAKER PRODUCTS

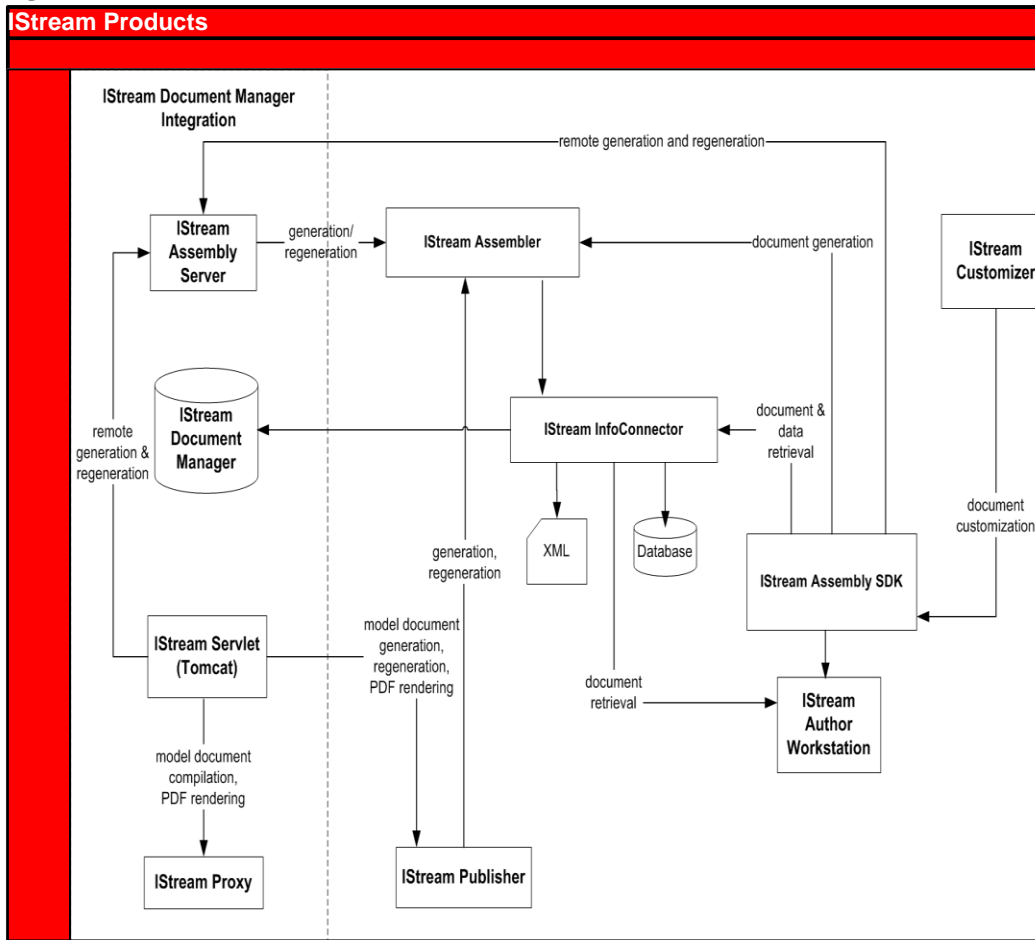
This topic discusses how IStream products relate to Documaker products.

**Note:** This topic does not describe which applications you should install on which systems. Some systems may be able to contain more than one feature. For more information, see the Documaker documentation.

IStream products		Documaker products
IStream Communicator	integrates with	Docupresentment, Documaker Interactive
IStream Document Manager DMS	maps to	UCM
IStream Document Manager (Assembly Engine)	maps to	Documaker Server
IStream Author	maps to	Studio, Documaker Add-In for Word
IStream Customizer	maps to	Documaker Desktop, Documaker Interactive
IStream InfoConnector	integrates with	Transall Studio
IStream Assembly	maps to	Documaker Server
IStream Promoter	maps to	Studio
IStream Admin Console	maps to	Documaker Configuration files
IStream Publisher Test Console	maps to	Studio Test manager
IStream Publisher Worker	maps to	Documaker Server
IStream Web Service Interface, SDK, and Queue Messages	maps to	Docupresentment, EWPS
IStream Writer	maps to	Documaker Studio, Documaker Add-In for Word



**Figure 1: IStream Products**

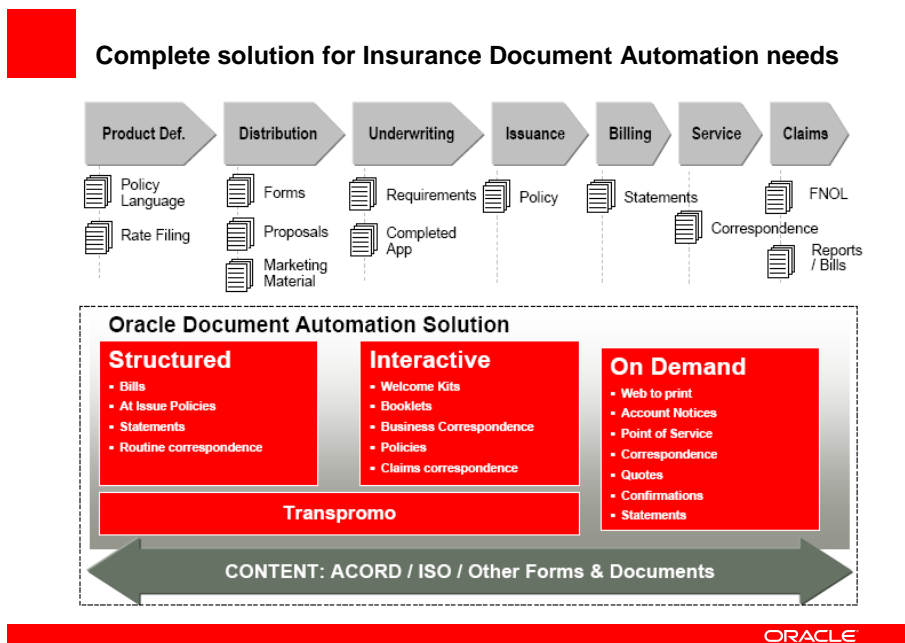


# ORACLE DOCUMAKER OVERVIEW

## Document Automation Solution

Documaker contains software products, product integration, system interfaces, application processes, and user interfaces to provide a complete document automation solution across the entire enterprise.

Figure 2: Complete Document Automation Solution



- Resource conversion
- Development tools
- Resource library management
- Assignment and routing
- Published output
- Archive
- Documaker Desktop
- High-volume processing with Documaker Server
- Data-driven processing and exception handling
- Transall
- Enterprise messaging infrastructure
- Web-enabling Documaker

*Streamline Forms Development:* Documaker streamlines the labor-intensive forms development process with innovative resource library management capabilities.

*Collaborative Work in Process (WIP):* Documaker work-in-process technology facilitates collaborative e-business information solutions and enables documents to receive additional review by a work group.

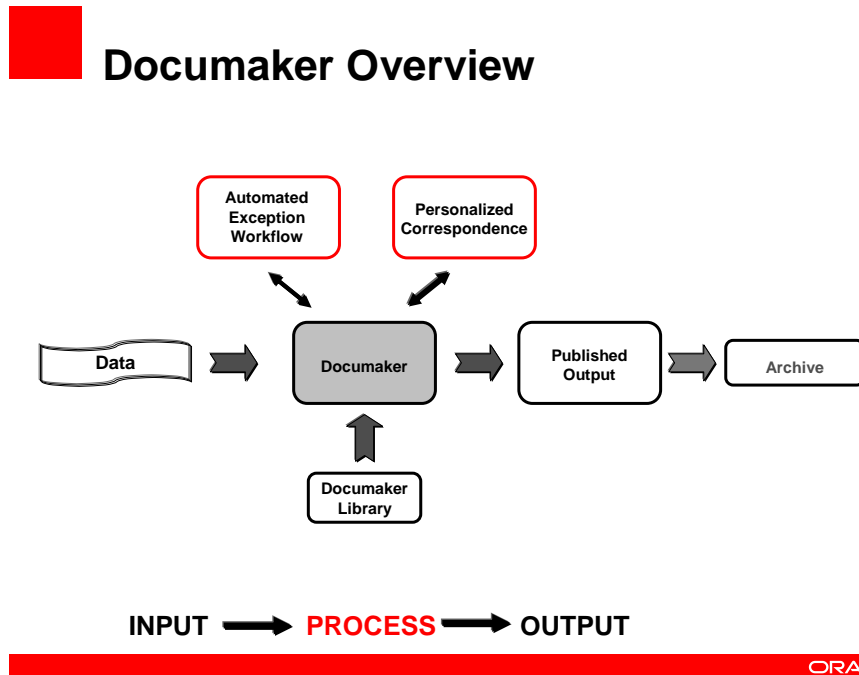
*True Fidelity:* Documaker technology enables electronic documents to be brought to the web without sacrificing compliance and fidelity.

*Return on Investment:* Documaker solutions yield a strong return on investment for the user in many areas, such as: faster development and deployment of forms and applications; reduction in paper and postage; web deployment without costly and error-prone print stream-scraping methods; automation and elimination of manual steps; reduction of errors or compliance violations; reduction of customer support calls.

*Ease of Use:* Documaker solutions focus on true ease of use and total cost of ownership.

## Documaker Overview

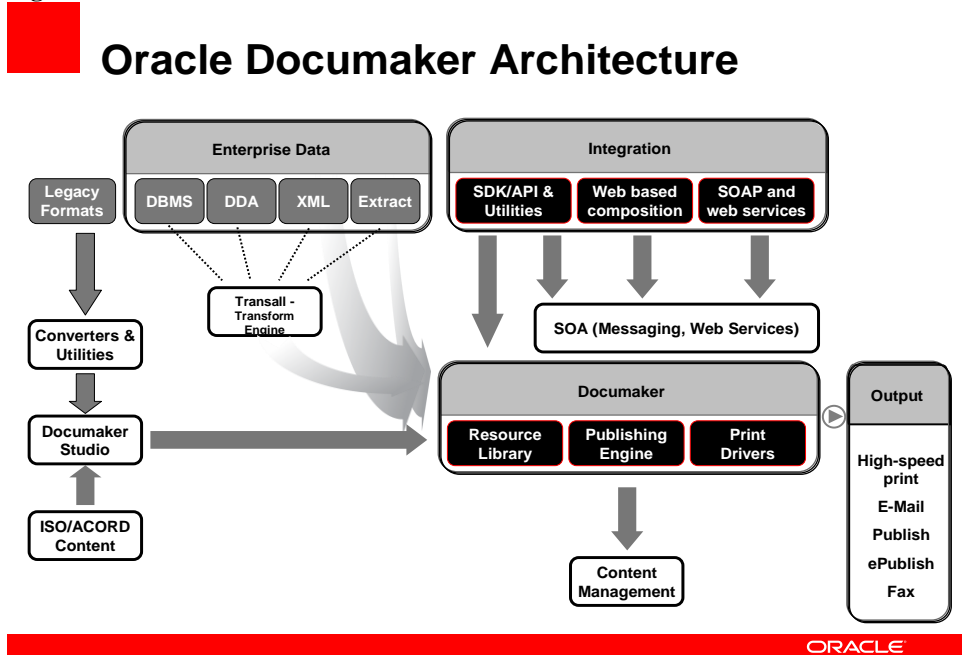
Figure 3: Documaker Overview



## Oracle Documaker Architecture

Documaker components are built using a multi-layered design, with each layer accessible via documented APIs. The layers range from the lowest level system calls and web services, and support SOA standards. Software Developer Kits (SDKs) consisting of toolboxes and libraries are available. They contain collections of common APIs. All of the components of Documaker use the same architecture and invoke common system APIs to read and write data, load and unload library resources, and to manage the objects and attributes inside the intelligent documents.

Figure 4: Oracle Documaker Architecture



# MIGRATION STEPS

The steps below illustrate the key tasks needed to successfully complete the IStream to Documaker migration. The migration begins once Step 1 has been completed and continues through Step 10. The complete project timeline will vary based on factors such as: customer specific requirements, stringency and length of testing by the customer. Each step is detailed in subsequent pages.

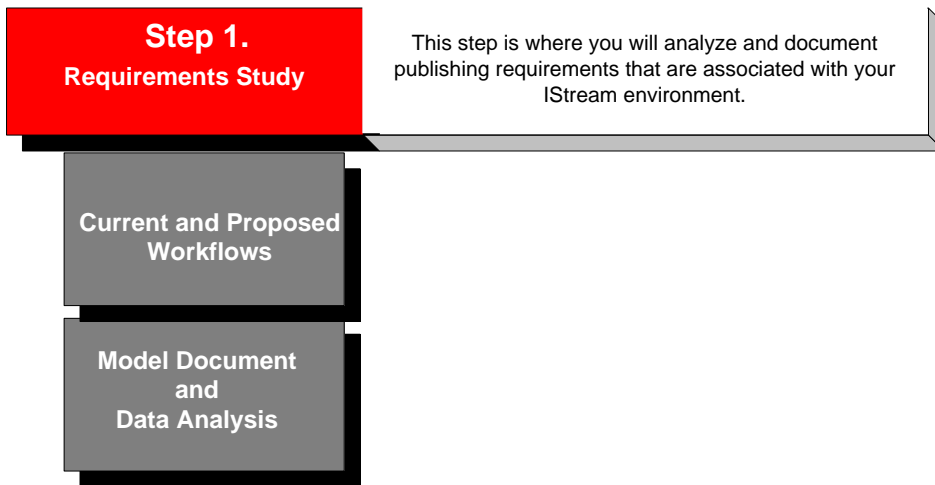
**Figure 5: Migration Steps**

<b>Step 1.</b> Requirements Study	This step is where you will analyze and document publishing requirements that are associated with your IStream environment.
<b>Step 2.</b> Project Planning	This step identifies: project resources and skills, system requirements, Documaker education needs, and changes to the Oracle support process.
<b>Step 3.</b> Migration Overview	This step identifies: migration pre-analysis steps and an overview of the migration process and utility.
<b>Step 4.</b> Model Document Migration Utility	During this step, you will run the migration utility and verify output files.
<b>Step 5.</b> DMStudio Pre-Import Tasks	This step identifies DMStudio tasks that need to be complete prior to importing migrated files.
<b>Step 6.</b> DMStudio Import	This step guides you through importing files into DMStudio.
<b>Step 7.</b> DMStudio Post Import Tasks	This step will walk you through updating and verifying DMStudio files.
<b>Step 8.</b> Documaker Configuration	This step will point out the how to configure Documaker system files.
<b>Step 9.</b> Test and Compare Output	This step you will compare IStream PCL/PDF files against the same files created using Documaker.
<b>Step 10.</b> Library Deployment	Final step, uses Documaker's deployment tool to deploy your library to the next environment in your document automation life cycle.

## Step 1. Requirements Study

---

The requirement study should be completed before running the IStream Migration Utility. The requirement study is critical to the success of the project. Each requirement must be measurable, testable, and, where possible, each requirement should be mapped from the old system to the new system. Each requirement must be defined sufficiently so the developer can construct from the requirement study document and testers can define a test matrix.



## CURRENT AND PROPOSED WORKFLOWS

The requirement study will gather information about current system workflows (functional and technical), system specifications including detailed forms analysis, recipient/batch processing analysis, production volumes analysis, business design requirements, forms composition requirements, data analysis (inputs, outputs, file layouts), archive and retrieval specifications and work-in-process requirements.

The current and proposed workflow analysis will help map IStream products and components to Documaker products and components



## ANALYSIS OF MODEL DOCUMENTS AND DATA

The analysis of model documents and data is critical to the success of a migration project. The model document analysis will define form types and map IStream model document requirements to Studio features and functionality.

To help with the migration effort we have identified several IStream document requirements that include; the ability to edit fields and paragraphs, repeating paragraphs, grouping requirements, if fields and paragraphs can be edited by the user, page numbering, and conditional field logic, and so on.

**Table 1: Sample IStream Form Types and Features**

Form Type	Editable	Grouping	Unlimited Fields	Repeating Sections	Pagination	Page Numbers	Table of Contents	Conditional Logic
Static Document	No	No	No	No	Possibly	Yes	Yes	Yes
Basic Variable Document	Yes	No	Yes	No	Yes	Yes	Yes	Yes
Table of Contents	Yes	No	No	Yes	Yes	Yes	Yes	Yes
Dynamic Document, no grouping	Yes	No	Yes	Yes	Yes	Yes	Yes	Yes
Dynamic Document, with grouping	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes

# MODEL DOCUMENT REQUIREMENTS

This topic compares the known IStream model document requirements to a corresponding Studio approach and then maps the Studio approach to one ore more Documaker runtime products.

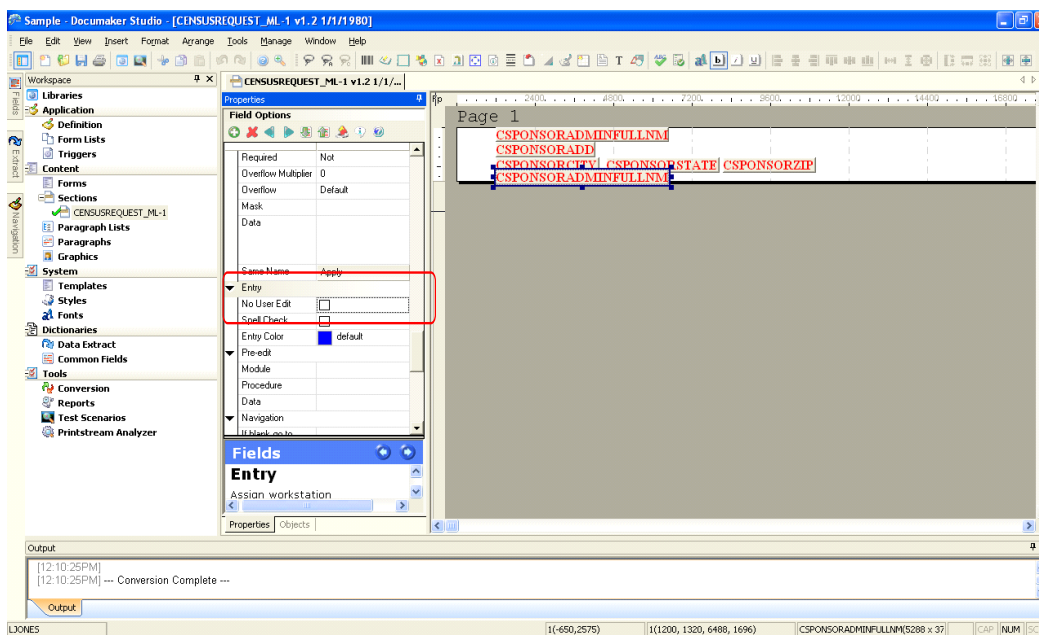
## Editable Content

### IStream Requirement

IStream model documents can be constructed to allow the user to edit parts of a document or the entire document.

### Studio Approach

Using Studio you can construct a form to have editable fields, and paragraphs. By default fields are flagged as editable by the user. To disable a field from user editing, simply enable the No User Edit option in Studio. Editable paragraphs will use a multiline text variable field and Studio's paragraph assembly functionality.



### Documaker Products

Documaker Desktop and Documaker Interactive let you edit variable fields and paragraphs.

## Grouping

### IStream Requirement

IStream model documents can be grouped, processed and assembled by plan, classes and other criteria.

### Studio Approach

Grouping requirements will use Studio to define groups and sections and if needed, Transall to organize the data. Oracle defines a *group* as a set of sections delimited by a begin image and an end image that is processed as a single unit. Studio uses section rules (GroupBegin and GroupEnd) and sub forms to define a collection of sections within a group. A group is a set of sections delimited by a begin section and an end section that is processed as a single unit.

### Documaker Products

Documaker, Transall

---

**Note:** For information about section level rules and sub forms, refer to the Rules Reference and the Documaker Administration Guide.

---

## Unlimited Fields

### IStream Requirement

IStream model documents do not have the concept of a field size.

### Studio Approach

Studio uses multiline variable fields to accommodate unlimited fields. All other field types require a definitive field size. During the requirement study phase, you will be asked to supply field sizes for each field referenced in your model documents.

### Documaker Products

Documaker Server, Documaker Desktop, and Documaker Interactive support multiline variable fields.

## Repeating Sections

### IStream Requirement

IStream model documents can repeat paragraphs or table column and row based on source data.

### Studio Approach

Studio support repeating paragraphs, table columns and rows using internal counters, Documaker rules and scripts.

### Documaker Products

Documaker Server, Documaker Desktop, and Documaker Interactive

## Pagination

### IStream Requirement

IStream model documents automatically paginates based on data using Microsoft Word pagination rules.

### Studio Approach

When constructing forms using Studio, you will need to consider how the form should paginate when Documaker applies headers, footers, sections, and how to paginate groups. Studio provides multiple section level and system level pagination rules that you can use when constructing forms that require pagination. Pagination rules are applied after the system has finished processing the entire form set, meaning that all data is complete.

## **Documaker Products**

Documaker Server, Documaker Desktop, and Documaker Interactive

## **Page Numbers**

### **IStream Requirement**

IStream generated documents can have page numbers and can also have documents within a document set that are not excluded in the total page count.

### **Studio Approach**

Studio supports page numbering at the form level and at the form set (entire document set) and forms can be excluded from the total page count.

## **Documaker Products**

Studio, Documaker Server, Documaker Desktop, and Documaker Interactive

## **Table of Contents**

### **IStream Requirement**

IStream model documents can repeat paragraphs and text based on source data.

### **Studio Approach**

Studio supports repeating paragraphs, text, variable fields, table columns and rows using internal counters, rules and scripts.

## **Documaker Products**

Documaker Server, Documaker Desktop, and Documaker Interactive

## **Conditional Logic**

### **IStream Requirement**

IStream model documents may have embedded If/Then/Else statements that are used to determine what and how data will populate the document.

### **Studio Approach**

Studio uses Document Automation Language (DAL) scripts to perform If/Then/Else statements. You can use DAL scripts to create field calculations, form and section attachment rules, and to apply If/Then/Else logic when populating fields.

## **Documaker Products**

Documaker Server, Documaker Desktop, and Documaker Interactive

## **RELATED DOCUMENTATION**

[Documaker Studio User Guide](#)

[Documaker Studio Help](#)

[Frequently Asked Questions](#)

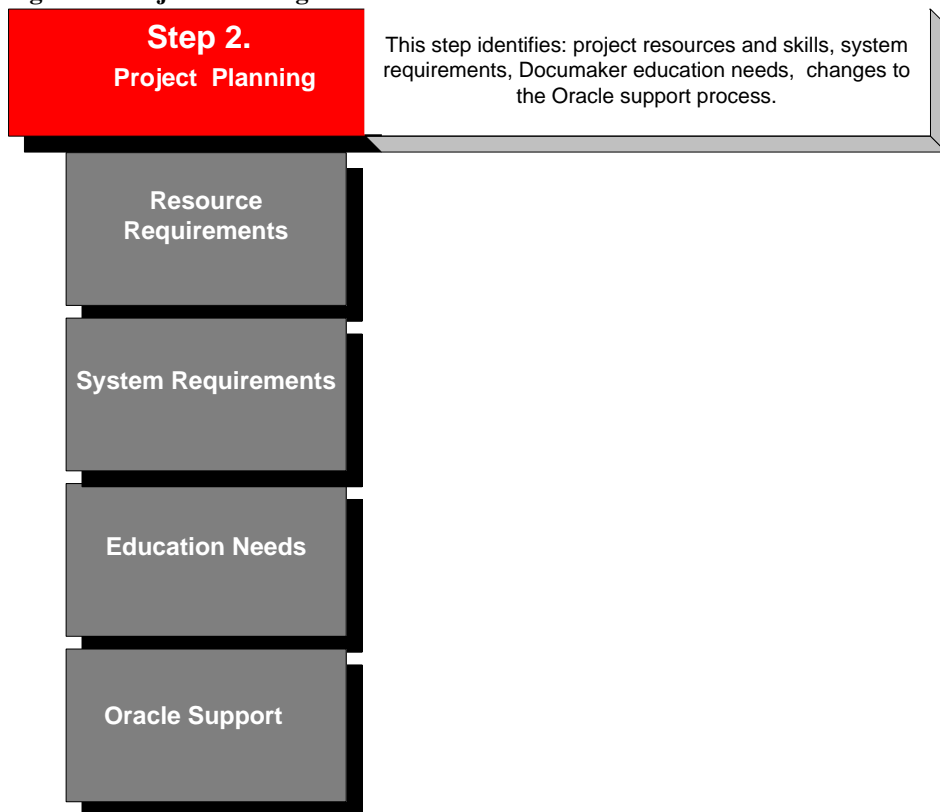
[Documaker Administration Guide](#)

## Step 2. Project Planning

---

The key to a successful project is in the planning. The project goal is to provide a seamless migration to Documaker by defining roles, responsibilities and deliverables.

**Figure 6: Project Planning Tasks**



# RESOURCE REQUIREMENTS

It is recommended that a project team be comprised of both Documaker trained specialists and IStream trained personnel. The team's goal is to provide a seamless migration from IStream to Documaker. Definitions of project personnel roles and responsibilities are provided below:

## **Documaker Trained Resources**

### Project Manager (PM)

The central point of contact for your IStream to Documaker migration is the Project Manager. This person will manage all phases of the implementation and control project planning, task assignments, project issues and resolution, status reporting as well as any direct customer contact necessary to manage and control the project deliverables.

### Studio SME

The role of the Studio SME (Subject Matter Expert) is to work with you and your staff to help migrate and implement your document libraries. The Studio SME is involved in all phases of the project from pre-analysis, to running the migration utility, importing migrated files into Studio, and testing.

### Technical Contact

The technical contact reviews the Requirements Questionnaire responses and performs any technical tasks such as database connection configuration and any other non-business functions.

## **IStream Trained Resources**

### Project Manager (PM)

The Project Manager will be the central contact point for Oracle. Project Managers work together to set key deliverable dates for the migration project. The PM manages resources and task assignments at the customer side, and provides status reporting.

### IStream SME

The IStream SME is someone who is familiar with your model documents, new model document processes, and business level requirements. This person will assist with resolution of mapping questions, provide test matrices and test xml files.

### Technical Contact

The technical contact is the contact for database connectivity issues, technical questions and technical deliverables as needed from the customer.

## SYSTEM REQUIREMENTS

This guide does not outline the IStream Migration Utility software pre-requisites. These pre-requisites are included in the Documaker Installation Guide and need to be installed separately before running the IStream Migration Utility.

## ORACLE EDUCATION

Oracle University is the premier provider of training for Oracle technologies and products. One of the largest global corporate training organizations, Oracle University offers thousands of in-class, onsite, web-based, and CD-ROM courses for learners around the world. Oracle University can help customize a complete training plan that matches the customer's requirements. We train entire project teams according to their learning needs, timeline, and budget, or help individual learners find the right training by using our easy job-role learning paths. Oracle education advisors are experienced in identifying learning objectives and the needs of customers. Their mission is to provide customers with end-to end education approaches that meet the unique education requirements of each organization and exceed the expectations of our customers.

Oracle University courses are available in a variety of training formats to provide students with flexible learning options. In addition to courses taught at Oracle Learning Centers, we offer private events, training at customer sites, and custom training. Delivery methods include Instructor-Led, Live, or Recorded WebClass training, Self-Study Online, and CD-ROM. The courses and their associated training formats and locations are subject to availability. For additional information please visit [http://education.oracle.com/pls/web\\_prod-plq-dad/db\\_pages.getpage?page\\_id=501](http://education.oracle.com/pls/web_prod-plq-dad/db_pages.getpage?page_id=501)

We recommend that system administrators take the three-day course for document designers, Oracle Insurance: Introduction to Documaker Studio. In addition, there are several courses which are recommended for system administrators to support Oracle Documaker in an operational environment. Additional administrative training is available via Oracle Consulting and may be provided as part of a customer's Oracle Documaker implementation.



# ORACLE SUPPORT

My Oracle Support is a personalized, collaborative support platform that incorporates service request creation, product knowledge base, access to patches and updates, and a community forum, all in one place. More information on each function is available from the Help option, located in the top-right of your screen after logging into My Oracle Support.

The IStream to Documaker migration will not affect your Oracle support. It does, however, change the product numbers used when requesting support. Documaker product names, product offerings, product numbers, and components are listed below:

Product Name	Product number	Description
Oracle Documaker	5477	Docupresentment, Docusave, Documaker Server, Documaker Studio, Forms Integrity Manager, fonts, Documaker Interactive, iPPS, PPS Reporting Tool, Transall, Transall Lite, Transall Studio, Transall Studio Lite
Oracle Documaker Desktop	5478	Documaker Workstation, fonts, Policy Print System (PPS), PPS Reporting Tool
Oracle for ACORD Forms	5482	ACORD Forms
Oracle Library Express	5487	ISO P&C Commercial and Personal Lines of Business Forms. Converted, tagged, and ready to use by Documaker products.

**Note:** There are no changes to your Customer Support Identifier (CSI) number however, if you do not already have a CSI number, the following topics discuss how to request a CSI number and register on Oracle's support site.

## Requesting a SI (Support Identifier) Number

You need an SI number to log a support inquiry. The SI number identifies your licensed products and maintenance level. You can request your SI number by contacting the Oracle Support Hub at 1-800-223-1711 (in North America).

**Table 2: Requesting a CSI Number**

Steps	Description
Contact Oracle Support Hub	1-800-223-1711
Indicate the type of support request	Select 2, non-technical support
Provide Oracle Support Hub with your company name	Your company name is used to locate your licensed products, maintenance levels and contract duration.
Provide Oracle Support Hub with the product name	Oracle Documaker

## Registration on the Oracle Support Site

The registration process uses a number of pages to gather relevant information so Oracle Support can contact you. To register, go to <https://support.oracle.com>, then click New User?

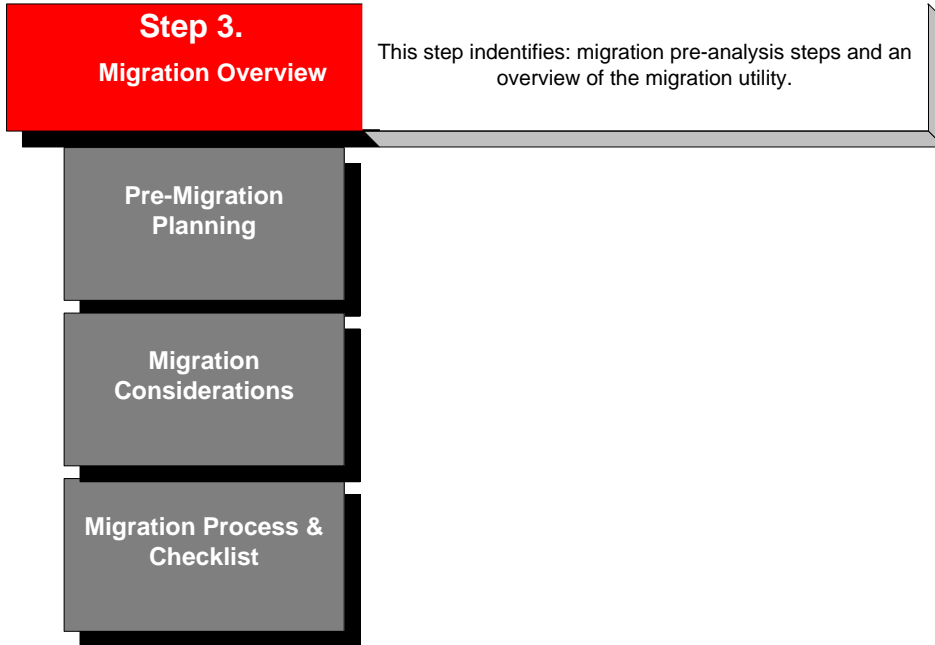
**Table 3: Registration on the Oracle Support Site**

Prompt	Description
Email	An email address is needed to login to Oracle Support and is used to receive communication from Oracle Support.
Support Identifier, Terms of Use	You must have a valid support identifier which is also called SI number. SI numbers are associated with Oracle products so you may have multiple SI numbers.  For IStream to Documaker customers, you need the SI number for Oracle Documaker.
Contact Information	Contact information is gathered so Oracle Support can communicate with you regarding your support requests.
Send and Wait	Send and wait page lets you review information entered before submitting your registration request.

## Step 3. Migration Overview

---

The IStream Migration utility and Documaker Studio are the main IStream to Studio migration components. The utility is used to create the same output in Studio that was achieved from IStream. The IStream Migration utility extracts content, logic, and variable fields from IStream files, creates RTF files and reports. The Studio output, when assembled, will produce the formatting, content, graphics, rules, logic and variables originally created in IStream. This will be accomplished by converting the IStream CMS/CDS files into Studio compatible files.



## PRE-MIGRATION PLANNING

Oracle IStream and Oracle Documaker use different file structures, data acquisition models, and logic constructs. The migration to Oracle Documaker requires that the content, logic, and variable definitions be extracted from the IStream format and imported into Documaker. A large portion of the migration is handled by utilities and product functions; however, some manual preparation of environments and post-import tasks are required.

- Gather models documents to be converted to a staging area
- Remove any \*.C?S tmp files from the staging area
- Create an output destination folder
- Ensure adequate drive space on temporary storage (size of models \* 2)
- Identify all required external resources
- Printers that are used by IStream and will be used by Documaker
- Microsoft Word templates (.DOT) referenced in the Model Documents (.CMS/.CDS)
- Fonts used within the documents
- True Type fonts for display
- Printer/device specific fonts
- Install IStream Migration Utility
- Install Documaker Studio
- Identify sample data sets and produce printed and electronic output for comparison

## MIGRATION CONSIDERATIONS

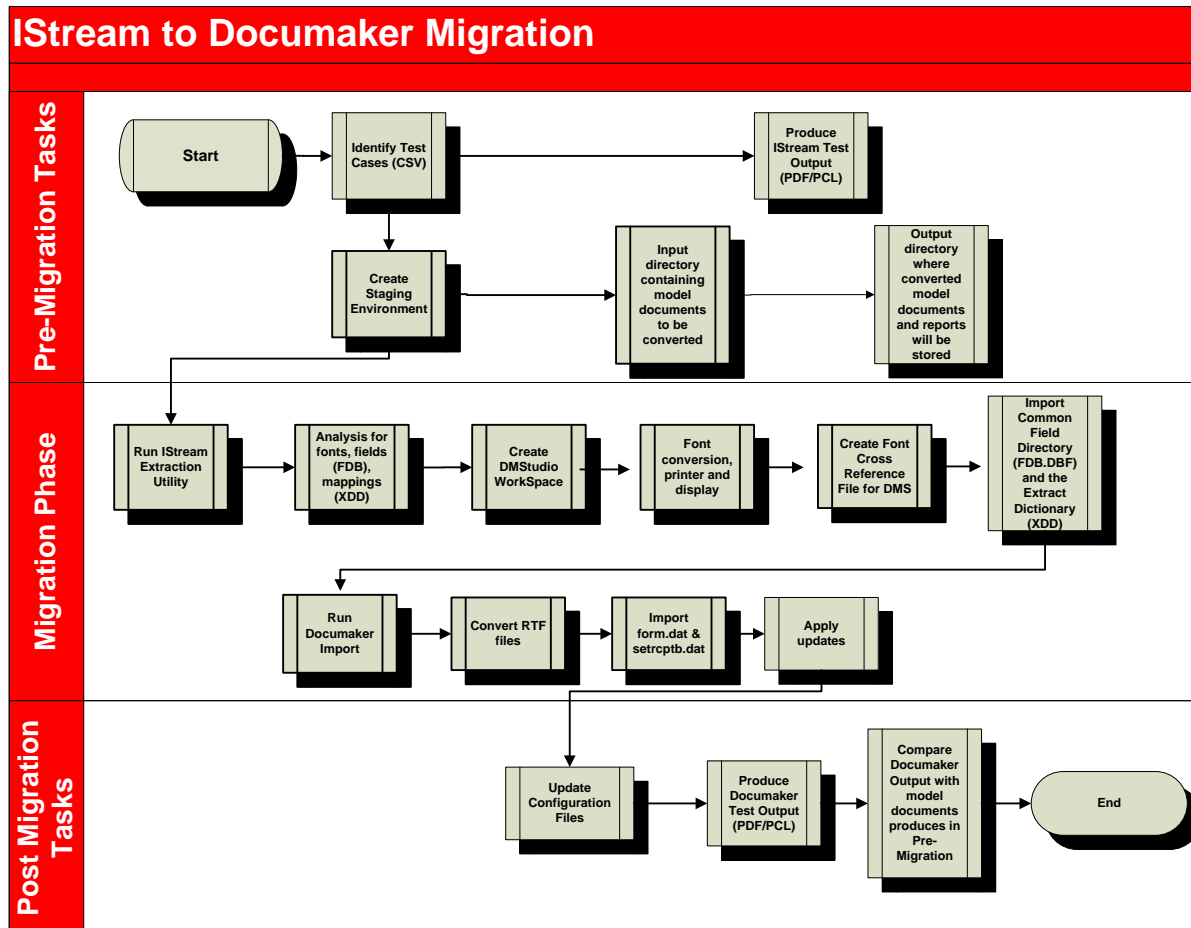
Each Model Document should be processed independently.

- A new Documaker form will be set up for each IStream Document Section (CDS).
- A new Documaker section will be set up for each RTF fragment that is exported from IStream Document Sections (CDS).

# MIGRATION PROCESS AND CHECKLIST

The key to a successful migration is to map functionality from the old system to the new system, which the Insurance Global Business Unit, also known as, IGBU, completed for the IStream to Documaker migration project. The second building block to a successful migration is having a roadmap that defines migration best practices and steps. The IGBU created a proven migration roadmap (see the checklist on the next page) that will help you manage your migration project. This guide is a blend of the IStream to Documaker functionality mappings and the migration best practices and steps defined by the IGBU. These migration steps will overlap with the implementation steps defined by the IGBU Consulting Services group.

Figure 7: IStream to Documaker Migration



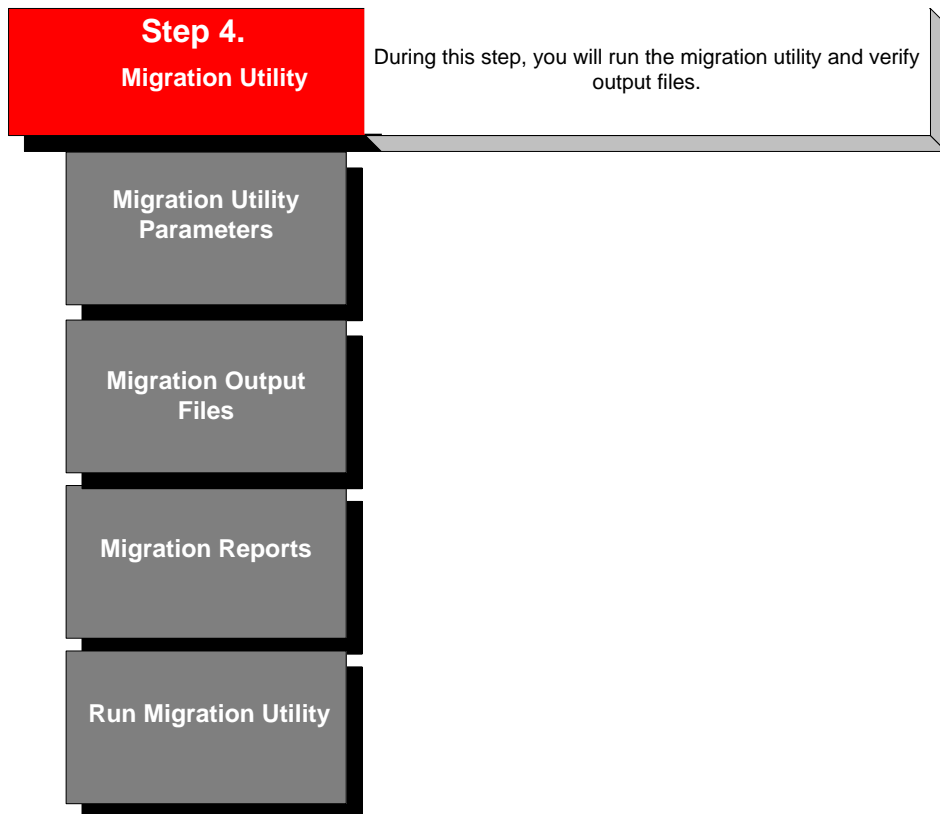
**Table 4: Migration Checklist**

Step	Sub-Step	Complete
<b>1. Requirement Study</b>		
	Current and proposed workflows Model document and data analysis	
<b>2. Project Planning</b>		
	Resource requirements System requirements Education needs Oracle support	
<b>3. Migration Overview</b>		
	Pre-migration planning Migration considerations Migration utility parameters Migration process and checklist	
<b>4. Migration Utility</b>		
	Migration utility parameters Migration output files Migration reports Run Migration utility	
<b>5. Studio Pre-Import Tasks</b>		
	Create recommended directories Create workspace Convert fonts Create font cross reference file (FXR)	
<b>6. Studio Import</b>		
	Import FDB & XDD Convert RTF files Import forms list and form attachment rules	
<b>7. Studio Post-Import Tasks</b>		
	Apply header and footer sections Create Table of Contents (Optional) Review and modify graphics (Optional) Apply section positions Convert expressions into DAL scripts	
<b>8. Documaker Configuration</b>		
	Define user roles and responsibilities Configure distribution channels and recipients Document rendering Configure options Configure system files Configuration scenarios	
<b>9. Test and Compare Output</b>		
	Run Documaker to produce PCL and PDF Compare output Use Forms Integrity Manager	
<b>10. Deployment</b>		
	Deploy library	

## Step 4. Model Document Migration Utility

---

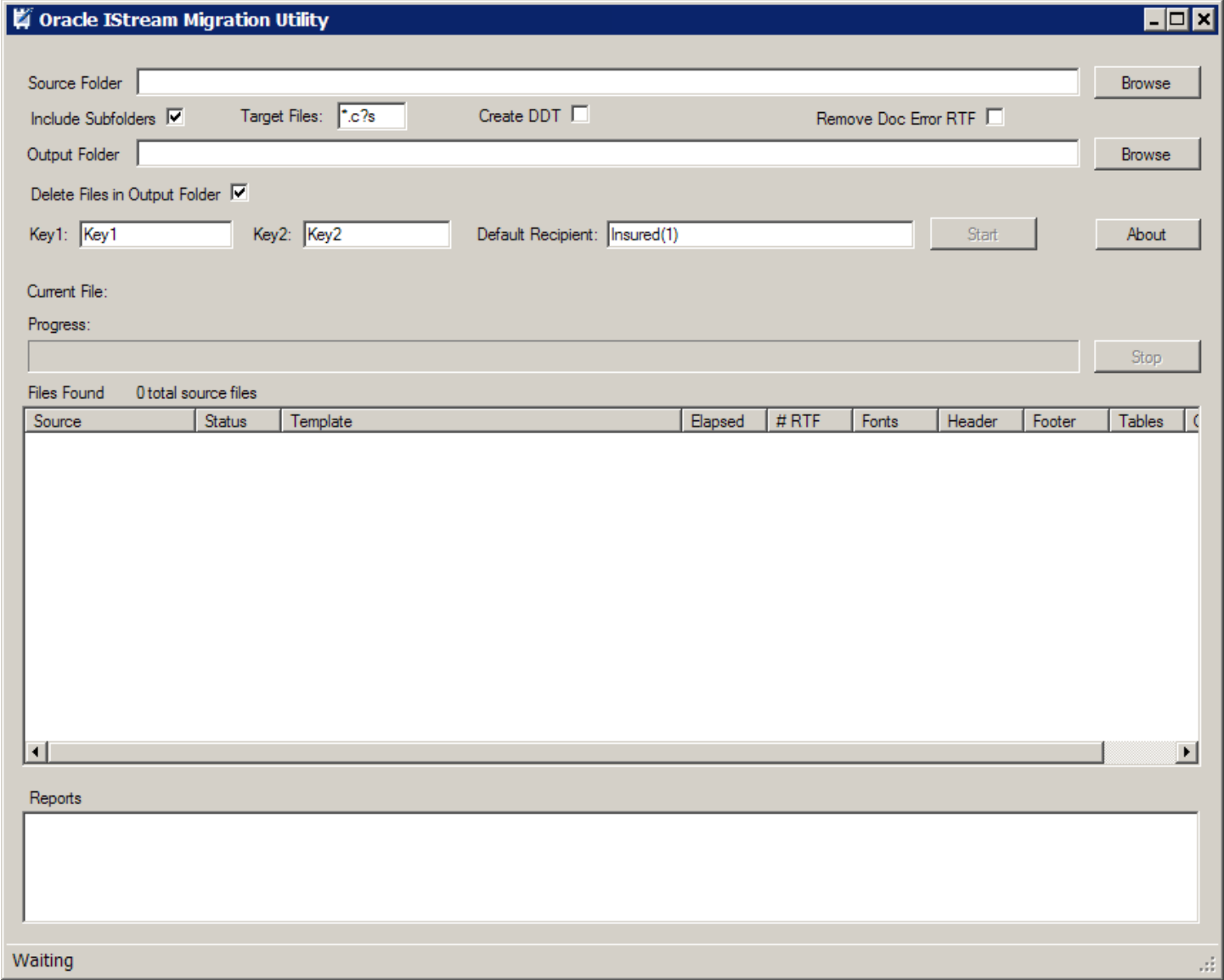
The migration utility is designed to mine the content, logic, and variable definitions from IStream Model Documents (CMS/CDS) and create the files needed to migrate to Documaker. This step walks you through setting up and running the migration utility.



# MIGRATION UTILITY PARAMETERS

When you start the utility, this window appears:

Figure 8: IStream Migration Utility



Field	Description
Source Folder	Browse to the location where you stored your target Model Documents.
Include Subdirectories	Check this box if you have target Model Documents in subdirectories.
Target Files	This is where you can change the extension of the target model directories. The default is *.c?s, which lets the utility get both CMS and CDS files.
Create DDTs	Check this box to have the utility create DDT files.
Remove Doc Error RTF	If you want the utility to Identify and separate error handling TEXT/ENDTEXT blocks from regular content in model documents (CDS) when exporting RTF files, check this box. The utility will only separate these blocks if there is a CALL DOCERR followed by TEXT/ENDTEXT along with an ENDJOB.



Output Folder	Browse to the location where the documents generated from the utility are stored.
Delete files in output folder	Check this box if you have previously stored documents in the output folder that are no longer needed.
Key1 and Key2	<p>Keys define a business group of forms, such as, proposals, letters, or policy forms. Insurance companies use the Key1 field to define companies and the Key2 field to define lines of business. Keys are a way of categorizing transactions. Each Key1 may have multiple Key2 subsets.</p> <p>Note: You must run the utility for each Key1 and Key2 combination.</p>
Default Recipient	<p>Default recipient name and count.</p> <p>Note: You can add recipients after you import the files into Studio.</p>

Click Start to convert your model documents.

## MIGRATION OUTPUT FILES

The IStream Migration Utility evaluates a collection of IStream Model Documents (CMS and CDS) and produces output files that facilitate the migration to Documaker. This utility produces:

**Rich Text Format (RTF):** Each TEXT/ENDTEXT rule within the model is extracted to a RTF file to isolate the data acquisition and inclusion logic from the textual content of the models. If a model contains multiple text rules a separate output sub-folder will be created to contain the RTF files for that model.

**Logic Files:** All inclusion logic and data acquisition rules are extracted into plain text files with a logic extension. Use the information in the logic files to trigger the forms in Documaker based on specific conditions. One logic file is produced for each model that contains rules outside of the TEXT/ENDTEXT areas. The logic file information appears on the spreadsheet under the Text Block Inclusion Logic tab.

**Define Files:** Variable define statements are extracted into plain text files with a define extension. You can use these files, along with the expression files, to help set up the proper field definitions within Documaker. One define file is produced for each model that contains define statements.

**Expression Files:** All expressions and variables used with the models are extracted into plain text files with an .expr extension. You can use these files, along with the define files, to help set up the proper field definitions within Documaker. One expression file is produced for each model that contains variable references or other expressions within the TEXT/ENDTEXT areas.

**Report Files:** Various report files are produced that provide additional information about the models.

**DDT Files:** All variable fields are output to a data definition table (DDT file) and can be used to define data extract fields in Documaker. One DDT file is produced for each model that contains variable fields.

**DAT File:** The DAT file is a grouping of forms, sections and has references to the default recipient assigned within the IStream Migration Utility. This file is used to create the Documaker Studio group file.

**DAL File:** The DAL file is a collection of form attachment rules extracted from model documents. This file used to create the Documaker Studio trigger file.

**Table 5: Migration Output Files Map to Studio Files**

IStream Migration Output Files	Process	Studio Files
Rich Text Format (RTF)	Conversion	Sections, FDB, XDD
Logic Files	Analysis	Inclusion rules
Define Files	Analysis	Common Field Database (FDB), Data Extract (XDD)
Expression Files	Analysis	DAL scripts and Documaker rules
DAT File	Import	Group file, forms list
DAL File	Import	Trigger file

# MIGRATION REPORTS

The IStream Migration Utility creates several migration reports. Reports are created in plain text format and are used throughout the migration steps. Here are descriptions of the reports:

## Summary Report

The summary report provides summary information on the models it processed. The report provides the number of model documents processed, number of fonts, headers, footers, tables, and graphics used in the model document. The summary report also provides elapsed processing time in seconds for each model document.

## Fonts Report

This report identifies all fonts used within the models documents. The report lists the font name, font attribute, and point size. Use the Font report to identify fonts that may need to be converted and added to the Documaker system. Font conversion is discussed in step 5.

## Headers Report

This report identifies whether a model document uses Microsoft Word headers. Use this report to identify Documaker forms which include headers. Header usage is discussed in step 7.

## Footers Report

This report identifies whether a model document uses Microsoft Word footers. Use this report to identify Documaker forms which include footers. Footer usage is discussed in step 7.

## Tables Report

This report identifies whether a model document uses Microsoft Word tables. Use this report to identify Documaker sections using tables.

## Logos Report

This report identifies whether a model document contains graphics, such as company logos and signatures. Use this report to identify Documaker sections which include graphics. Graphics are discussed in step 7.

## Exceptions Report

This report tells you if the migration utility encountered an unexpected problem. The report displays the path and the model document name that encountered the problem and includes a system exception message. Use this report to identify model document problems. You need to review and resolve issues in this report before moving to the next step in the migration process.

## DAL Scripts Report

This report identifies model documents which use curly braces ( { } ) to insert a value of an expression into the text. You must replace the braces with brackets ( < > ) and a valid field name to migrate to Documaker. Use this report to identify model documents which have braces. You need to review and resolve issues in this report before moving to the next step in the migration process. DAL scripts are discussed in step 7.

## Tags Report

This report identifies references in model documents that use the TEXT TAG attribute. The report is not generated when a library does not use the TEXT TAG attribute. Only tags using the TEXT TAG syntax are recognized by the migration utility. Use this report to identify tags which use the TEXT TAG syntax in model documents.

## Include Summary Report

This report identifies lists the references used in model documents. The report shows the model document name and the includes the statement reference. Use this report to identify include statements contained in the model documents. Review this report to determine if additional sections are needed to complete the migrated form.

## Plus Report

(plus.txt) This report contains RTF file names. It is used to implement the IStream Document Manager (IDM) ENDTEXT+ rule, which indicates that the RTF file that follows the one mentioned in the Plus report should be appended without any spaces or lines.

## Language Report

(language.txt) This report tells you what language is used in the RTF file. Here is an example of the format:

```
<RTF file name>, Language: <language_letter>
```

where *language\_letter* can be:

E - English

F - French

S - Spanish

G - German

I - Italian

## Issues Report

(issues.txt) This report notes issues that occurred during the processing of an IDM document. The issues can contain descriptions of problems which occurred while parsing a particular IDM rule. The Exceptions report (exceptions.txt) notes problems that are typically caused by settings and missing prerequisites.

# RUNNING THE MIGRATION UTILITY

## Step 1. Start the IStream Migration Utility

During the load of the form the application folder is checked for a simple text file containing the last known source and destination folders. If the file is found the source and destination are pre-populated.

## Step 2. Select Source Folder

Click Browse to select the root folder of the Model Document files.

If the Include Sub Folders box is checked, all child folders are processed through all levels of the folder structure. If it is not checked only the source folder is evaluated.

The Target Files setting is used to filter the files selected for processing within the source folder structure. The default is \*.c?s which selects all CMS and CDS files. You can use any appropriate filter.

## Step 3. Select Output Folder

Select the root output folder using the Browse button. The Folder Select window provides a new folder method if required by the user. The root folder must exist before execution. The utility creates all of the necessary sub folders for the output.

## Step 4. Select Start to Begin Processing

Click Start. The utility checks for text length greater than zero in the source and output folder properties and displays a message and stops processing if either is blank.

A message reminds you that Microsoft Word templates (\*.DOT) and fonts must be properly positioned before proceeding. Click Ok to continue.

---

**Warning:** Proper processing requires that this system have access to the Microsoft Word templates (.DOT) referenced in the Model Documents (.CMS/.CDS) files being processed. The fonts used in the documents (.CMS/.CDS) must be properly installed and available to Microsoft Word during processing. Failure to properly provide the templates (.DOT) or fonts will result in the creation of improperly formatted output (.RTF) documents and, subsequently, poorly formatted Forms within Documaker. Please make sure you have properly positioned the templates (.DOT) and any required fonts before proceeding.

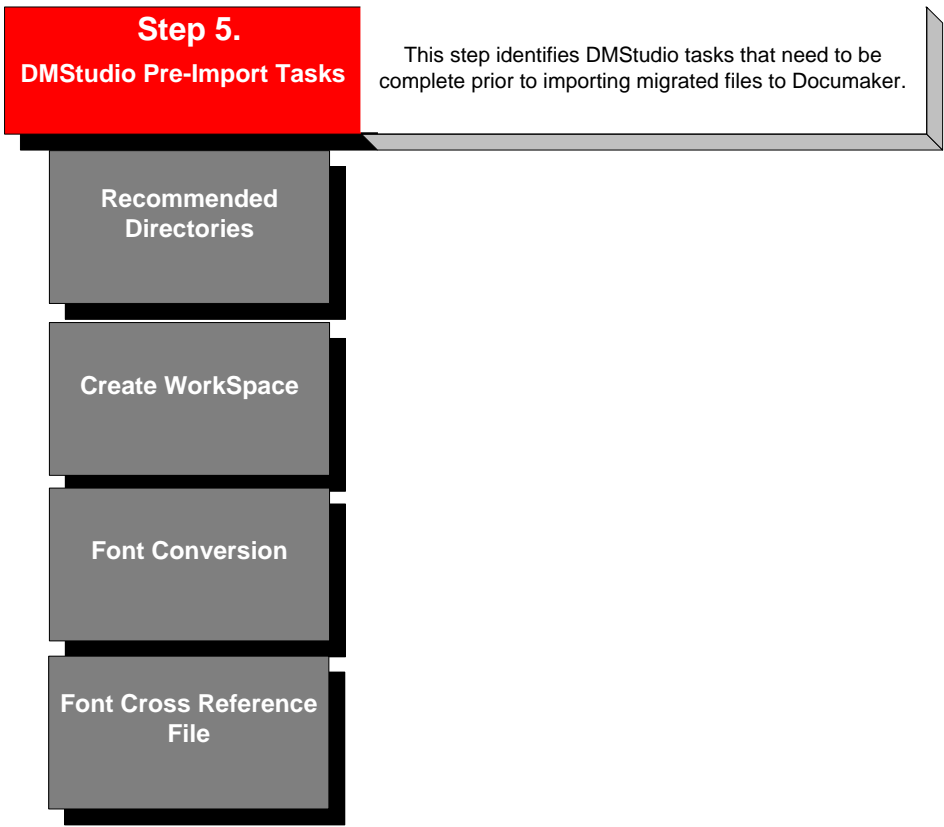
---

## Step 5. Review Output and Reports

Best practice with any migration or conversion project is to review reports and verify output. If necessary, update source files and re-run the utility.

# Step 5. Documaker Pre-Import Tasks

This step will guide you through creating your Studio workspace and preparing to import migrated files into Studio.



## RECOMMENDED DIRECTORIES

After running the migration utility, you will want to move common file types into a staging structure to help with the import process. We recommend that you create staging folders so you can easily access files when importing into Studio.

- Create staging folder, possibly use Key1 and Key2 combination.
- Create a RTFLIB folder under the staging root folder and copy \*.RTF files to the RTFLIB folder.
- Create a DDTLIB folder under the staging root folder and copy \*.DDT files to the DDTLIB folder.
- Create a DEFLIB folder under the staging folder and copy \*.expr, \*.logic, to the DEFLIB folder.
- Create a LOGO folder under the staging folder and copy graphics (logo and signatures) to the LOGO folder.

## CREATE WORKSPACE

Go to the New Workspace option and enter the name of the workspace and location where you want the workspace to be created. If the subdirectories do not exist they will be created. Also, copy TrueType fonts to the font folder defined in Studio. Step 6 guides you through manually importing IStream files using the File menu and Import Workspace Files option.

---

**Note:** Once created, you are designated as the administrator of the user database (USERINFO).

---

## RELATED DOCUMENTATION

[Documaker Studio User Guide](#)

[Documaker Studio Help](#)



# FONT CONVERSION

For a successful migration to Documaker Studio, you must complete a font analysis, a font conversion, and set up a font cross reference (FXR) file before converting the RTF files. Every font present in your model documents will need to have an entry in the FXR file.

## Font Analysis

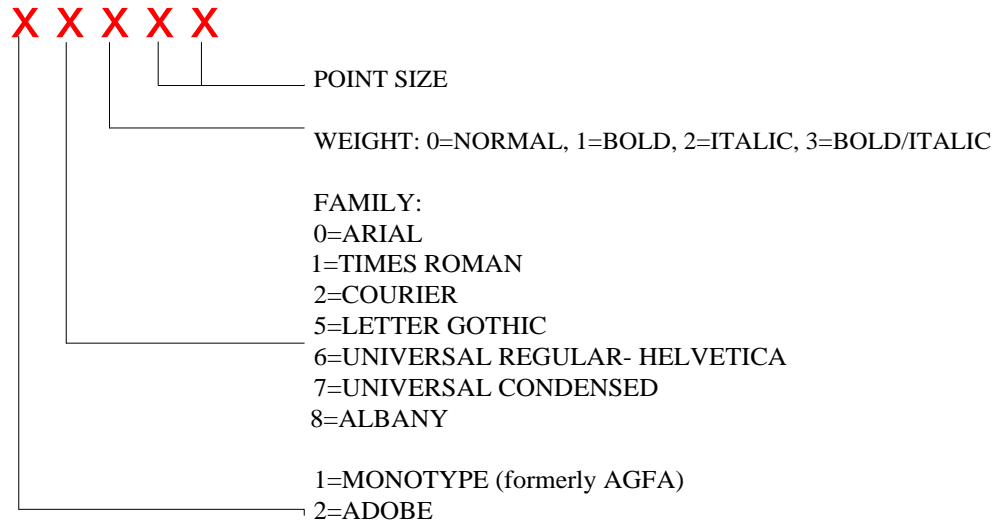
This includes the following tasks:

- Identify the font environment
- What font families are used and what are the font characteristics
- Evaluate the composition environment
- Understanding screen versus printer fonts
- Evaluate the print platform, such as Xerox, IBM, PCL, and so on
- Understanding Documaker font migration utilities
- Graphic-to-font, Metacode-to-PCL

## Font ID Naming Standards

Documaker references fonts using a five character font ID that is resolved at print time. Oracle recommends that all fonts adhere to the five character naming standards.

Figure 9: Font ID Naming Standards



## IMPORTING FONTS

To import fonts, select the Import under Fonts menu or right click and select Import from the pop-up menu in the Font List window. The Import wizard appears. Font types that can be imported are as follows:

- AFP
- PCL bitmap
- PostScript/TrueType
- Xerox Metacode

Use the Import option to add font information to your font set files. Import lets you add multiple font information or individual font information to the font set or FXR file. If necessary, you can edit font information to change the font ID, typeface, family, and printer specific information.

Step 1. Start Studio, open your workspace.

---

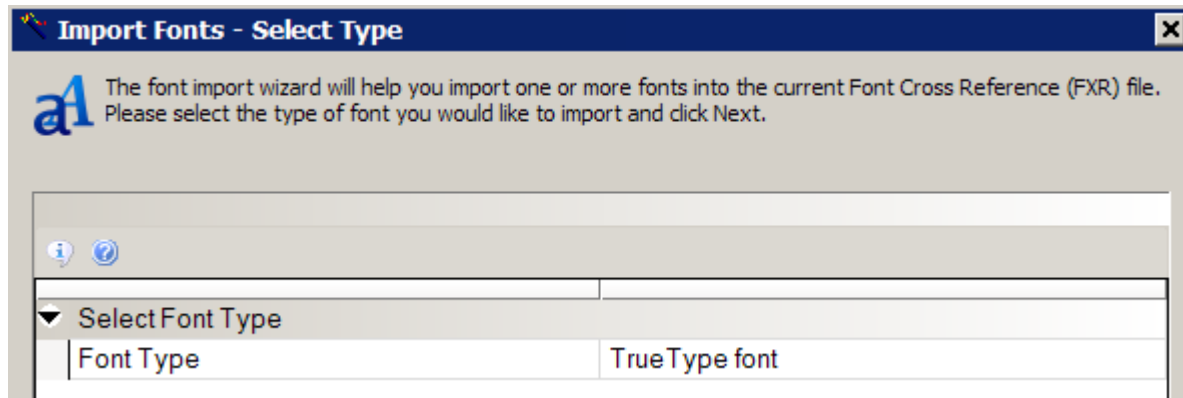
**Note:** Make sure TrueType fonts are in the font folder defined in Studio.

---

Step 2. Click the Fonts tab to open the FXR file in the Font manager.

Step 3. On the Font tab, click the Import option to display the Import Fonts – Select Type window.

Step 4. Select TrueType font as the type of font to import.



Step 5. Click Next. The Font Support Files Required window appears.

Step 6. Locate the font support files and click Ok.

---

**Note:** Font support files should be stored in the font folder defined in Studio.

---

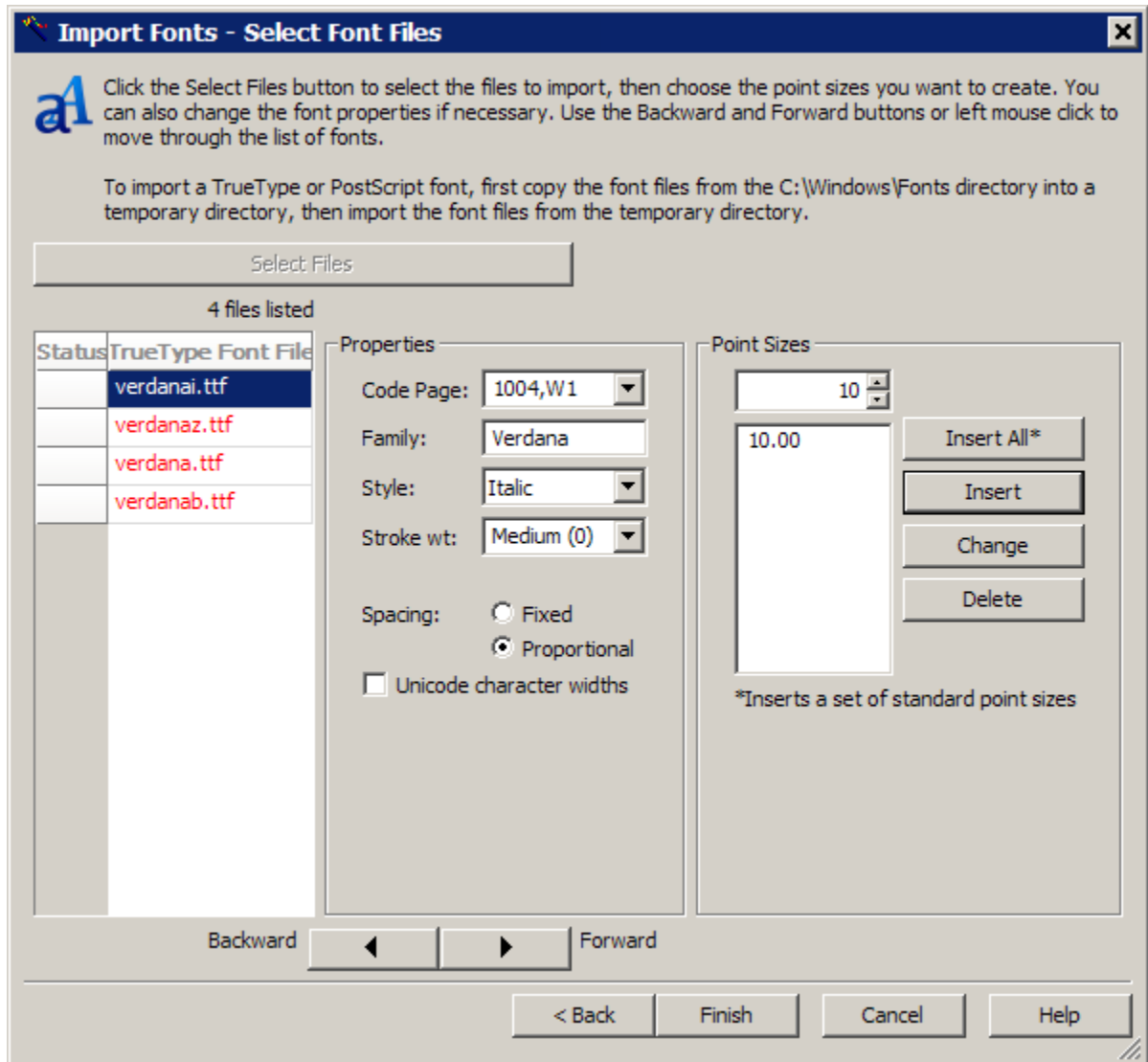
Step 7. Click Next, The Import Fonts – Select Font Files window appears.

Step 8. Select the TrueType fonts to import.

- a. Select 1004, W1 as the code page to use during the import.
- b. Enter the point size you want to import and click Insert.

If you want to import additional point sizes, repeat this step by typing in the point size you want to import and clicking Insert again.

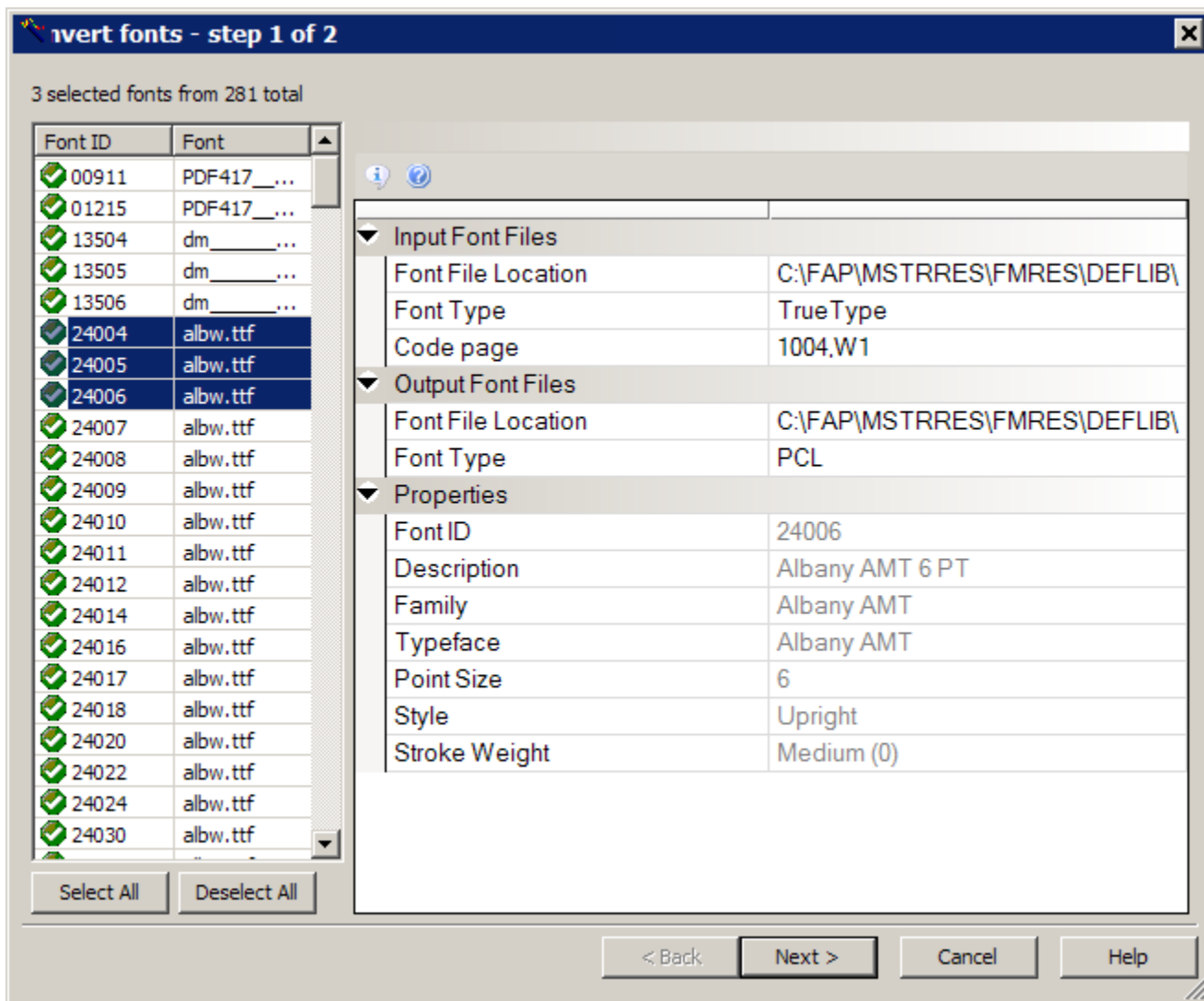
- c. Click Finish when you are finished.



Studio displays a message to show the number of fonts it imported.

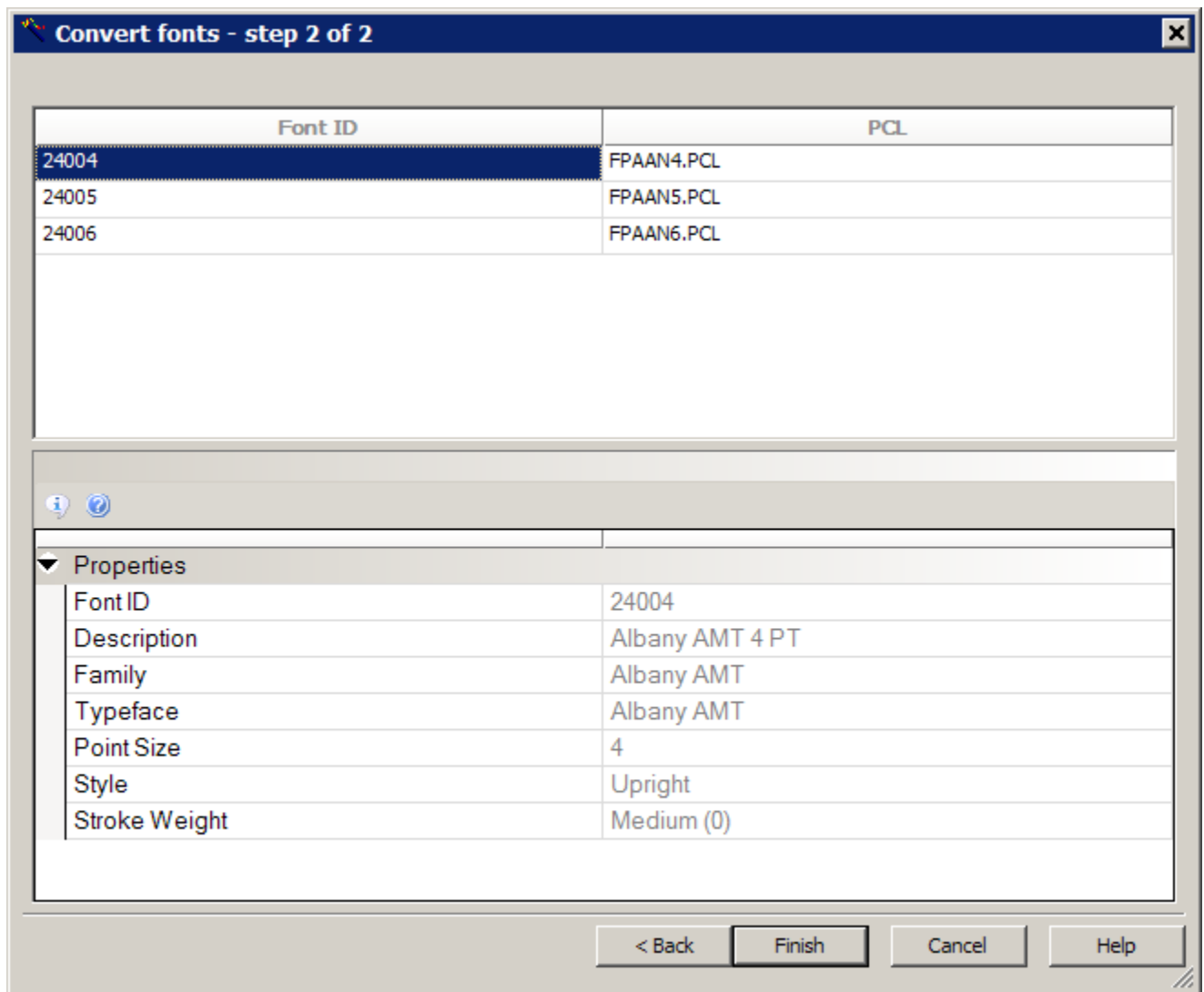
## Font Conversion

- Step 10. From the Font tab select Convert to display the Convert Fonts Step 1 of 2 window.
- Scroll down to the bottom of the list and select the fonts you just imported.



- Click the fonts you want to convert. Make sure you are converting from TrueType fonts and you are using code page 1004.W1.
- Select the fonts you want to convert. Make sure that PCL is selected as the type of font to convert to. Then click Next

**Note:** If you want to change the file name of the PCL font to be converted, click the PCL font name in the second column and change the file name as needed.



d. Click Finish to finalize the migration. Studio shows you the number of fonts converted.

Step 11. Save changes to the FXR file. Then perform test print using converted PCL font.

## RELATED DOCUMENTATION

Documaker Studio User Guide

Documaker Studio Help: key words – Fonts, Converting Fonts, FXR

Frequently Asked Questions

Documaker Administration Guide

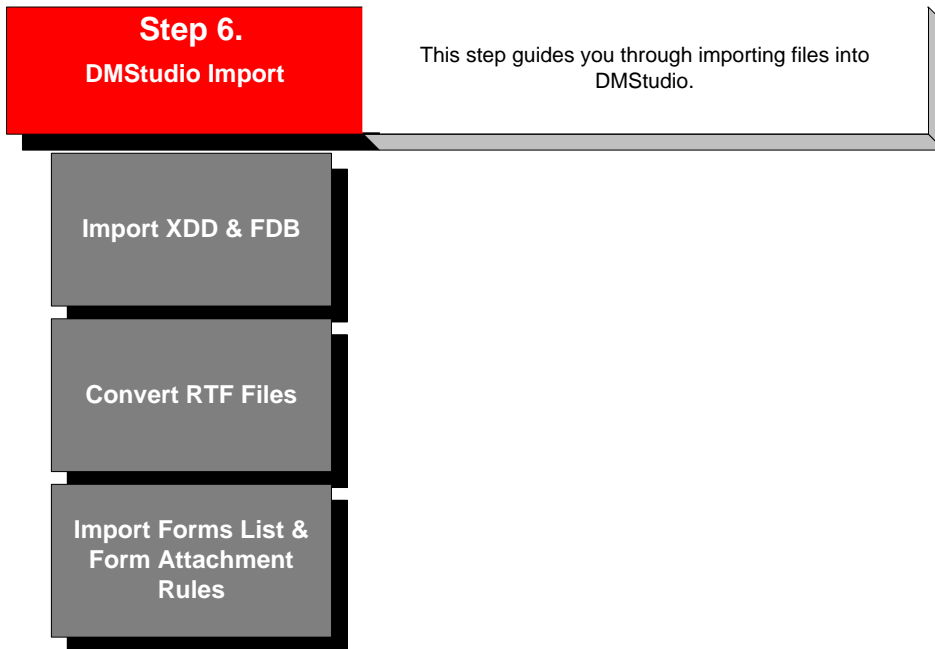
## BEST PRACTICE

A well organized font set makes document creation fast and efficient.

## Step 6. Documaker Import

---

This step shows you how to import the files created by the migration utility.



## IMPORT COMMON FIELD DICTIONARY

The field dictionary lets you maintain and manage field naming standards across a single resource library or across many libraries. The field dictionary (also known as the FDB.DBF) lets you store and manage field settings in a database file format. Each record in this database contains most of the settings that appear for the field in the section.

The field dictionary contains a record for each unique variable field name, indexed in ascending order. You can use the Field Dictionary manager to ensure consistency in form sets. For example, if you have a Name field, you can pull the attributes for that field into the form from the Field Dictionary manager. The database contains such information as name of field, font, type of field, and so on.

There are several ways to create a common field dictionary.

- Import fields using the RTF conversion utility. The RTF conversion utility automatically inserts fields into the FDB.

---

**Note:** Oracle recommends that you let the RTF conversion utility automatically create FDB entries.

---

- Import the FDB using a CSV file. The CSV file is created using the \*.DDT files generated by the IStream Migration Utility.
- Add fields directly to the common field database via Studio.
- Create fields at document creation time and then add the fields to the common field dictionary.

## BEST PRACTICE

For ease of maintenance and re-usability, fields should be stored in the field database. This makes updating easier and faster.



## CREATE FDB IMPORT FILE

### Step A: Locate Resources

1. Locate the \*.DDT files created by the IStream Migration Utility
2. Create a spreadsheet named FDB.CSV which has the following columns, then go to Step B.

	A1				
	A	B	C	D	E
1	NAME	PROMPT	LENGTH	FONTID	SCOPE
2					
3					

The FDB.CSV file will contain general field information. All other field attributes will be added after you import the fields into Studio.

Column Name	Description
NAME	You can not use semicolons, backslashes, and quotation marks when creating a field name.
PROMPT	Displayed to the user when processing inter-active documents using Documaker Desktop or Documaker Interactive. Prompts can be used to provide descriptive information about a field and can be used to provide simple instructions for entering data, such as, "Enter the proposal date in MM/DD/YYYY format."
FONTID	Use one of the valid font IDs created in step 5.
LENGTH	Enter the maximum field length. The field length is defined by the source Admin System and also the available space on a document.
SCOPE	The scope controls the distribution or propagation of data to variable fields with the same name. Enter 0, if the field should propagate across a single document/form. Enter 1, if the field should propagate at the section level. Enter 3, if the field should propagate across an entire document set/form set.

### Step B: Open FDB spreadsheet (CSV)

1. In the NAME column, paste fields names from the \*.DDT files into the FDB.CSV in column A – NAME.
2. In the PROMPT Column, enter a descriptive definition of the field or you can enter helpful information about how to enter the data. The prompt appears to the end user during data entry time when using Documaker Desktop and Documaker Interactive.
3. In the FONTID column, enter a font ID for each field.
4. In the LENGTH column, enter a length for each field.
5. In the SCOPE column, enter a scope (see table above) for each field.
6. Save the FDB.CSV file, ensuring the file is saved as a CVS file.
7. Import the FDB.CSV file into Studio.

**Note:** The Common Field Database automatically saves therefore, you are not required to save or check-in this file.

## RELATED DOCUMENTATION

Documaker Studio User Guide

Documaker Studio Help file; key words – Users, Security

## IMPORT DATA EXTRACT DICTIONARY

The Extract Dictionary (also known as the SYMBOL.XDD) lets you store information about the input file, as well as field information, in a database file format. Using the Data Extract Dictionary increases productivity and reliability when mapping since each field does not have to be mapped at a detailed level. Instead of entering information for all the fields in the Rule section for each field in an section, you just have to specify the XDD rule as the field rule.

Typically, there is one SYMBOL.XDD file per resource library. By default, this file is stored in the DEFLIB directory, under the specified workspace. The XDD is used in the Rule options of the section at rules processing time. The link to the XDD database from the Rule options of the section is the use of the XDD rule.

The XDD file has a parent-child structure. The top level (the parent) is comprised of individual records in the extract file. Fields within that record are children. There are several ways to move data into the XDD:

- Import XDD entries using the RTF conversion utility. The RTF conversion utility automatically inserts fields into the XDD.
- Add XDD entries directly to the Data Extract (XDD) using Studio.

---

**Note:** Oracle recommends that you let the RTF conversion utility automatically create XDD entries. See Documaker Server and Documaker Studio documentation for additional data extract import file types.

---

## BEST PRACTICE

For ease of maintenance and re-usability, field mappings and conditional field logic should be stored in the Extract Dictionary.

## ISTREAM RULES MAP TO

Studio lets you use assign rules to control how information is merged onto forms, how that information is then processed, and how the information and those forms are output when using Documaker.

**Note:** Rules can be specific to an input file format, such as a flat file or XML. Review the Rules Reference to determine if the rule is appropriate for your implementation.

**Table 6: IStream Rules Map to Documaker Rules**

IStream rule	Documaker rule or DAL function
!EMPTY	Rules and DAL
AADD	Custom DAL or Transall
ABS	ABS
ALLTRIM	Trim
ASORT	Custom DAL or Transall
AT	StringFunction
BREAK	DAL
CALLAPP	Transall
CASEPHRASE	Move_it
CASEWORD	Move_it
CDOW	FmtDate
CHECK_DAY	DAL
CHECK_MTH	DAL
CHECKUP	TbLkUp
CHR	DAL (CHAR)
CMONTH	FmtDate
CTOD	FmtDate
DATE	SysDate
DAY	FmtDate
DEFINE	Retain
DISVPAR	Print_It
DLL	Transall
DMY	FmtDate
DOCERR	DAL
DOL_AMT	FmtNum, Round
DOL_AMT2	FmtNum, BankRound
DOW	FmtDate
DO WHILE	DAL (While and Wend)
DTOC	Date2Date
DTOS	DateFmt, StrngFmt, Date2Date, STR, DeFormat,
ELSEIF	Elseif
EMPTY	Rules and DAL
ENDIF	End
ENDJOB	GenDataStopOn INI option
ENDTEXT	Not Needed
ENDTEXT+ EXP	AppendText, AppendTxm, ConnectFields POW
FIELD	Tbllkup
FINDFILE	CheckImageLoaded
FORCE NEXT	Use a While statement
GETDAY	FmtDate, RunDate, SysDate
GETFILE	TblTxt and AppendTxt
GETIMAGE	Logo
GETMONTH	FmtDate
GETYEAR	FmtDate
IF	If
INCLUDE	Trigger
INITIALIZE	While...Wend, Continue
INT	INT
ISALPHA	Rules and DAL

<b>IStream rule</b>	<b>Documaker rule or DAL function</b>
ISFIELDNULL	Most rules, DAL
ISDIGIT	DAL
ISIMAGE	HaveLogo, HaveImage
ISLOWER	DAL
ISUPPER	Combination of DAL functions
LEFT	Left
LEN	Len
LOG	DAL
LOOKUP	LookUp
LOWER	Lower
LOWTRIM	Trim
LTRIM	MoveNum
MAX	MAX – numeric
MDY	DateFmt
MDY	Date2Date
MIN	MIN – numeric
MONTH	FmtDate
MYSTR1	MoveNum
MYUSERID	UserID
NEXT	Break
NUMBER	FmtNum
NUMBER2	FmtNum
PADC	PAD
PADL	PAD
PADR	PAD
POWER	RPN function
PROMPT	GVM variable definitions in the RCBDFFDL TriggerToWip
QUERY	Combination of database related DAL functions
RAT	DAL
REPLACESTR	DAL (INSERT)
REPLICATE	DAL (PAD)
RIGHT	DAL
ROUND	Round
RTRIM	Trim
SELECT	DBOpen
SELECT ALL	DBOpen
SETFORMAT	Locale format string
SETIMAGEFORMAT	ResetImageDimensions
SETLANGUAGE	Locale format string
SHOWRULES	RPWarningMsg, RPLogMsg
SPACE	Len
SPELLNUMBER	NumText
SQRT	RPN function
STR	MoveNum
SUBSTR	SUB
TEXT	DAL
TEXT {}	Hidden field
TEXT DEFAULT {}	Not needed
TRIM	Trim
TRUNCATE	FmtNum
UNUSEIS	Combination of database related DAL functions
UPPER	Upper
UPTRIM	FormatString
USEIS	Combination of database related DAL functions
VAL	CharV, FmtNum
YEAR	FmtDate, RunDate, SysDate
YMD	DateFmt, Date2Date

## CREATE DATA DICTIONARY

### Step A: Locate Resources

Locate the DDT files created by the IStream Migration Utility and copy those DDT files into the DEFLIB folder created in step 4.

### Step B: Import DDT Files

1. Start Studio and open a workspace.
2. Select Dictionary, Data Extract. The Open File window appears. Open the data dictionary (XDD file).
3. Go to the File menu and select Import.
  - a. On the Open File window, highlight Data Extract to display the XDD file.
  - b. Highlight the XDD file.
  - c. Click Browse to display the Data Extract window. From the Data Extract window change the file type to \*.DDT.
  - d. Locate the DDT files. Select the files you want to include and start the import process.
5. Save and check in the XDD file.

## RELATED DOCUMENTATION

[Documaker Studio User Guide](#)

[Documaker Studio Help: key words – XDD, Dictionary](#)

[Frequently Asked Questions](#)

[Documaker Administration Guide](#)

# CONVERTING RTF FILES

The RTF conversion step does more than simply convert RTF files into Documaker sections. This step also:

- Converts IStream RTF fragments to Documaker sections.
- Converts IStream tags which have brackets (<>) to Documaker field objects.
- Reads the field database (FDB) and verifies the field is in the FDB.
- Creates an entry in the common field database and the data extract (XDD) if the field is not found in the field database (FDB).

The common field database (FDB) entry uses the default field attributes selected to the Conversion Options window. The data extract (XDD) entry uses the default values selected on the field and XDD Options window.

- Writes an entry to a finishing report when a field is created by the RTF conversion tool. The report displays these possible messages:

If the field name is not found you will get this message:

```
Field <cFieldname> was not found in Common Fields dictionary
```

If the field is a multiline text field, you will get this message:

```
Field <cFieldname> type 'Multi-line Text' has been changed to Alphanumeric
```

If the field is a bar code field, you get this message:

```
Field <cFieldName> type 'Bar Code' has been changed to Alphanumeric
```

- Increments a field counter (FIELD ###) when the RTF conversion finds an IStream tag that occurs multiple times in one RTF fragment.

IStream Tag	Action	Documaker Field
<cPolicyNumber>	1st occurrence converts to	cPolicyNumber
<cPolicyNumber>	2nd occurrence converts to	cPolicyNumber #002
<cPolicyNumber>	3rd occurrence converts to	cPolicyNumber #003

- Automatically resizes sections by removing top and bottom white space.

---

**Note:** Before you can convert RTF files, you must convert your TrueType fonts, update your FXR rile, and import your common field data and data dictionary.

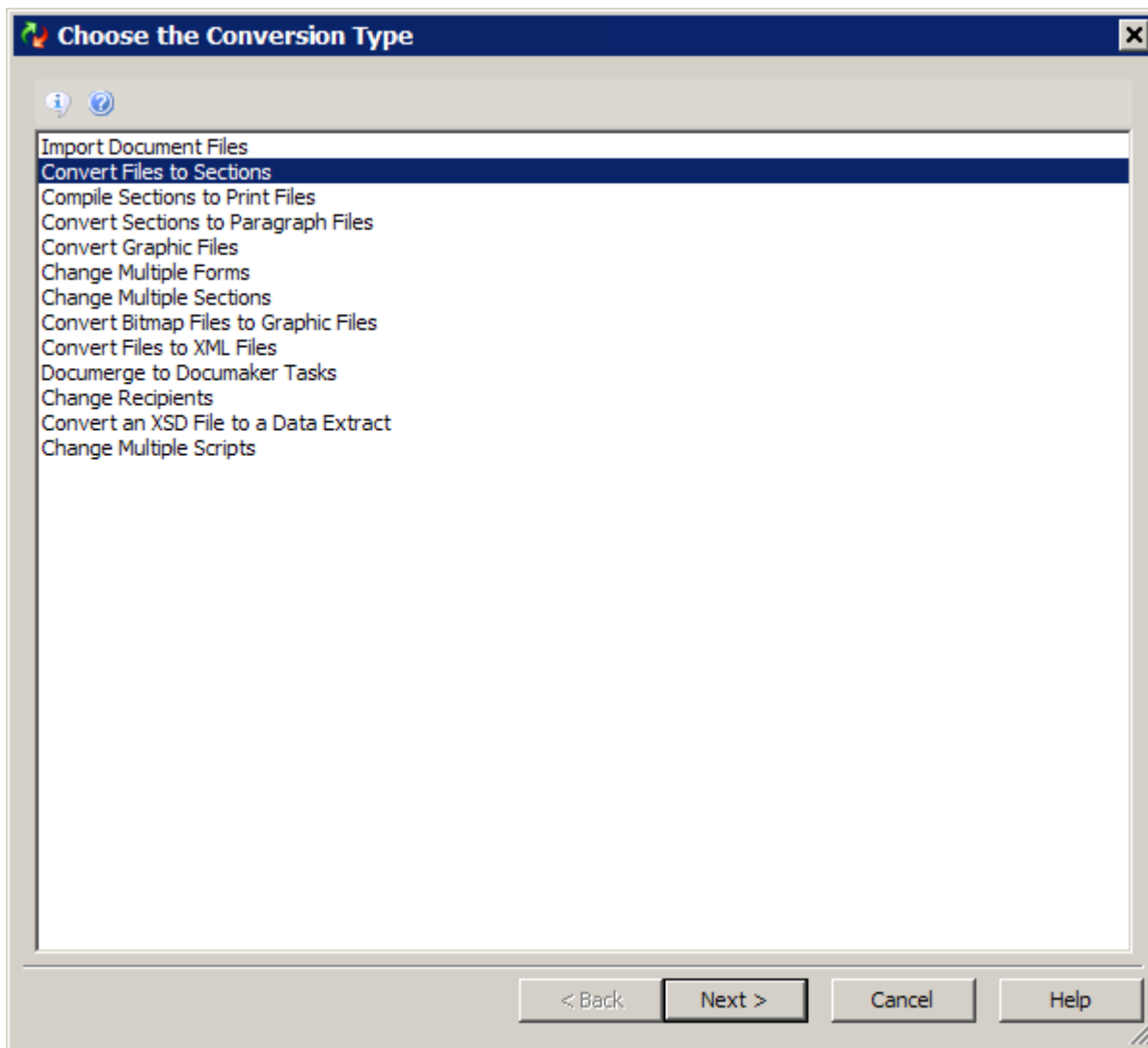
---

## Documaker RTF to Section Steps

Follow these steps to convert Documaker RTF files into sections:

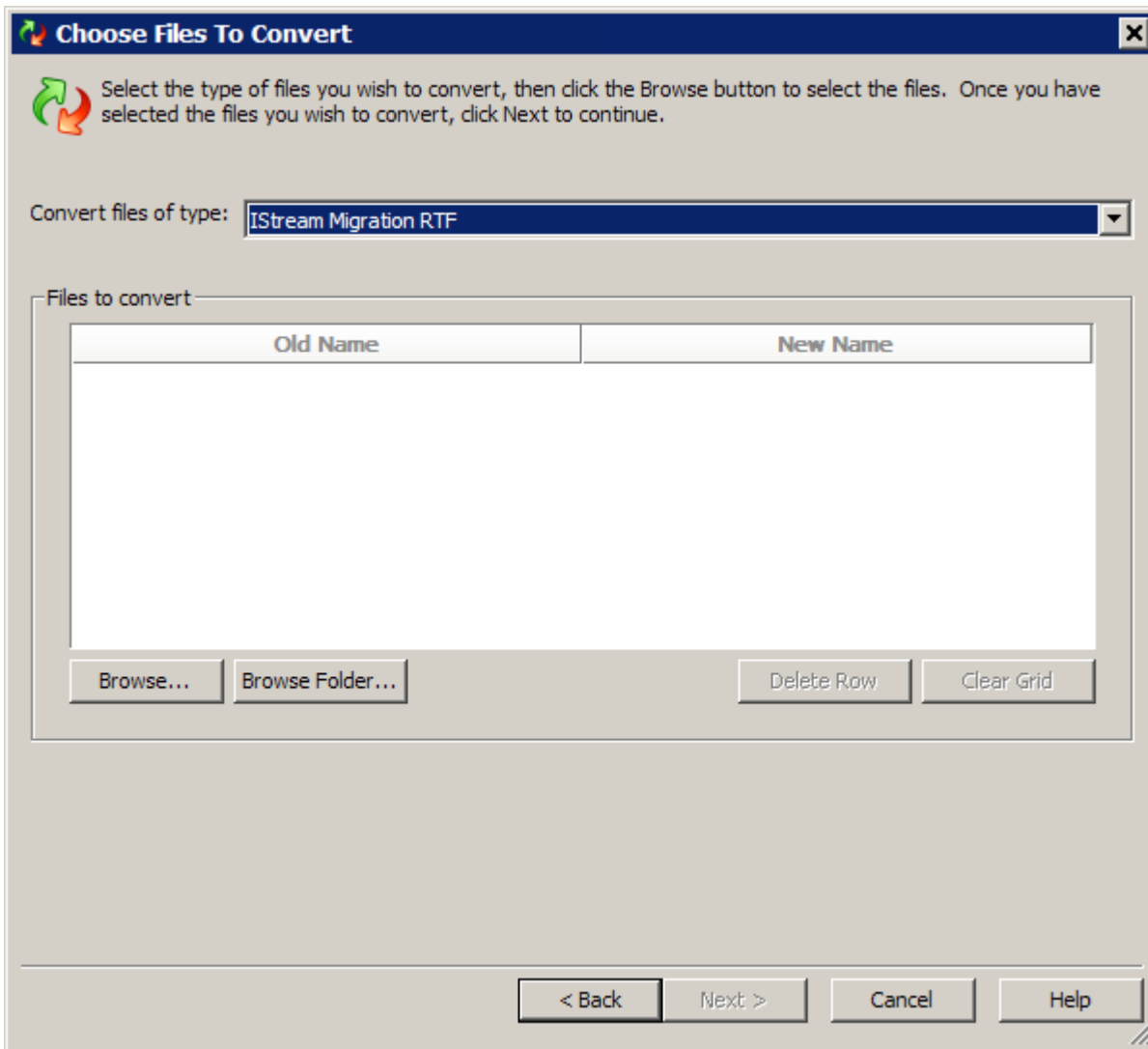
Start Studio and open a workspace.

Select Conversions. The Choose the Conversion Type window appears.



Highlight the Convert Files to Sections option and click Next. The Choose Files To Convert window appears.





Select IStream Migration RTF from the list of file types. Then click Browse to locate the RTF files created by the IStream Migration Utility.

Highlight the RTF files you want to convert and click Open to update the Files to Convert list. Then click Next. The Conversion Options window appears.

Accept the defaults for these fields:

- Field Length
- Field Scope
- File Options

Complete the options and click Next. The Fields and XDD Options window appears.

Check and select the default rule name for each field. This option inserts the rule name needed to populate the field.

---

**Note:** The IStream Migration RTF conversion option creates a Data Extract Dictionary so you will need to select the XDD rule.

---

Check the option to Update the Common Field Dictionary.

Select the Advanced option under Common Fields Dictionary (FDB) and select to Skip the record if the field already exists in the FDB.

Check the option to Update the Extract Data Dictionary

Select the Advanced option under Extract Data Dictionary (XDD) and select the Skip the Record if the Field Already Exists in the XDD option.

Accept the defaults for all other fields:

- Using XML Paths
- Not using XML Paths
- Field Name Offset
- Data Offset
- Field Name Value
- Adjust Field Name for Block Tag Fields
- Adjust Data Offset Values for Block Tag Fields

Complete the options and click Next.

The Convert Files and Finish window appears. This window lets you apply an effective date and an initial check-in description for the conversion. In addition, you can set mode, status, class, and project values.

Click Finish to start the conversion process.

Open the Common Field Database and the XDD and review the fields added by the conversion utility. Apply any needed updates.

## IMPORT FORMS LIST AND FORM ATTACHMENT RULES

The following steps guide you through the process of importing into Studio the forms list (IMUForm.dat) and form attachment rules (SETRCPTB.DAL) files created by the IStream Migration Utility.

### IMPORT FORMS LIST (IMUFORM.DAT)

- Step 1. Select the Import Workspace Files option. The Import Files window appears.
- Step 2. Select the Forms Definition File option and click Browse to locate the IMUForm.dat file.
- Step 3. Select the IMUForm.dat file and click Open. The Enter Trigger File Path window appears.
- Step 4. Check Skip to bypass this step. Then click Ok to start the import process.

### IMPORT ATTACHMENT RULES (SETRCPTB.DAL)

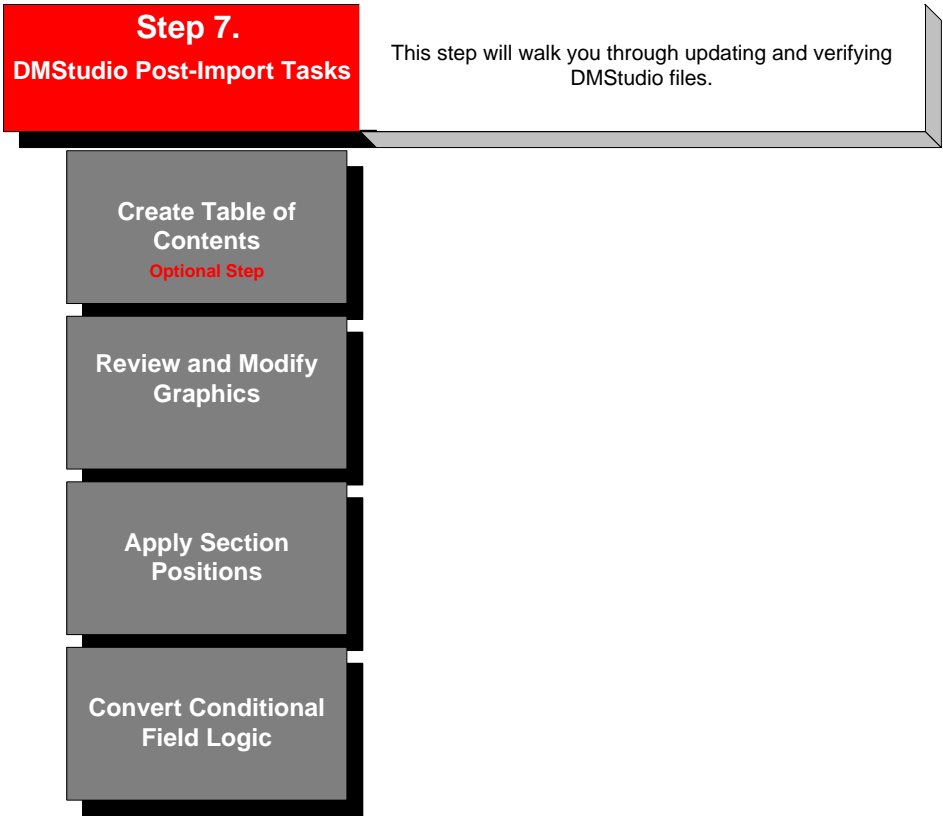
- Step 1. Select the Import Workspace Files option. The Import Files window appears.
- Step 2. Select the Triggers option and click Browse to locate the SETRCPTB.DAL file.
- Step 3. Select the SETRCPTB.DAL file and click Open.
- Step 4. Click Ok to start the import process.

## BEST PRACTICE

For ease of maintenance, check and validate attachment rules and remove duplicates.

# Step 7. Studio Updates

After importing files into Studio, the next step in the migration process is to apply final library updates. This includes reviewing graphics and analyzing and converting conditional field logic. This step will guide you through updating Studio files.



## CREATING THE TABLE OF CONTENTS DOCUMENT

You can add a Table of Contents object (or Table of Figures or Index) to your form set. Place the Table of Contents object on a section that is used on a form in the form set. The coordinates define where the generated Table of Contents will be placed. Size the table based on how large you want it to be. The system creates additional pages when necessary, but the table does not flow dynamically.

Add bookmarks to the sections in the form set to define the text you want to print in the Table of Contents, Table of Figures, or Index. You can define a level number that determines the formatting to use. If the bookmark is for an index entry, specify the main entry and subentry. An Index marker lets you add a main entry and subentry that can be sorted in alphabetical order. Indexes are sorted by main, sub, text, page number, and coordinate. Tables of Contents and Tables of Figures are sorted by page number and coordinate.<sup>1</sup>

To use this feature, all sections must be loaded before the print operation executes. Otherwise, the system does not have all the content available and is not able to create a complete table of contents, list of figures, or index. Since some print drivers do not force the loading of all sections until necessary, this means you may have to include an additional INI option.

For Documaker Desktop, include this option:

```
< Control >  
  LoadPrintOnly = Yes
```

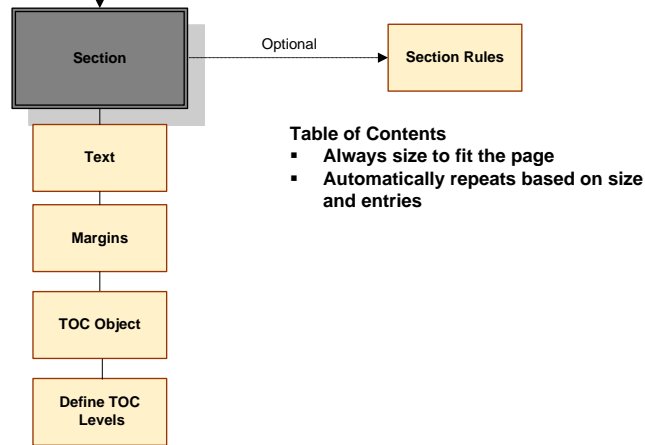
For Documaker Server (GenPrint), include this option:

```
< RunMode >  
  DownloadFAP = Yes
```

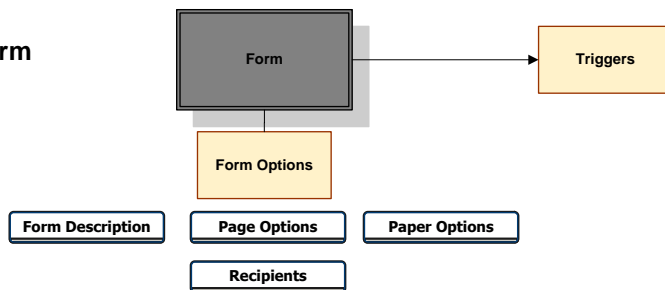
## Table of Contents



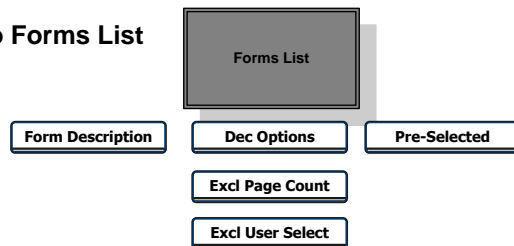
### Step A - Create TOC Section



### Step B - Add Form



### Step C - Add to Forms List



## REVIEW AND MODIFY GRAPHICS

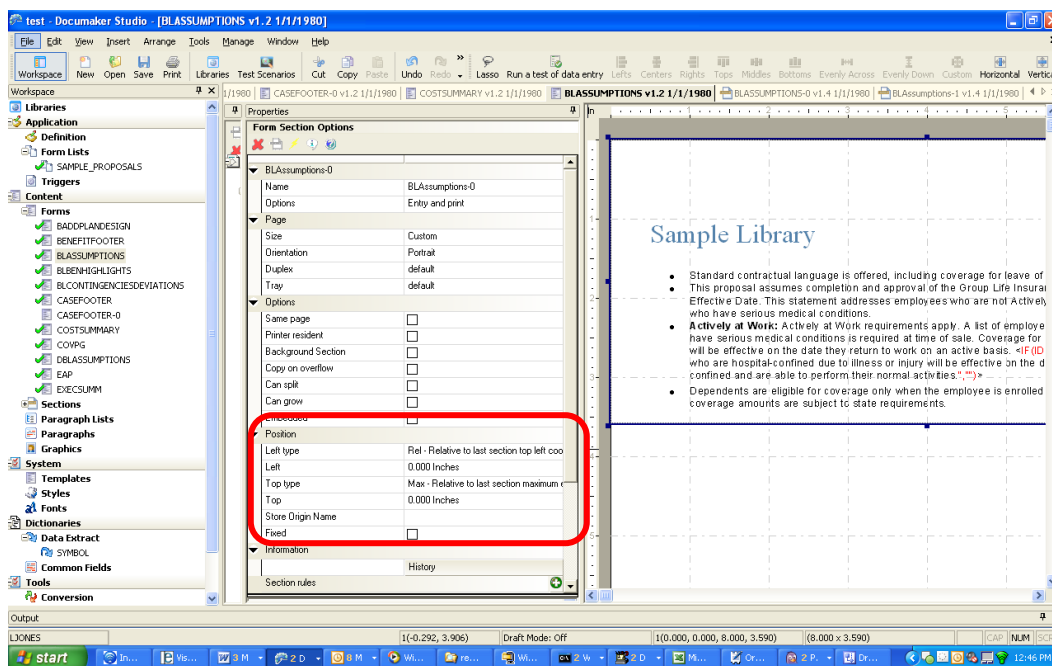
When files are converted from model documents into sections, the graphics are converted along with the document content. Documaker Studio, however, lets you reuse these resources by storing them in the Forms directory and letting you apply the graphics to other forms.

## SECTION POSITION

The RTF conversion shrinks each section by removing extra white space above and below the text. After the RTF fragments are converted to sections, you will need to apply section positions.

### Step A. Applying Section Positions

1. Highlight Forms, then right-click and select Open.
2. Highlight the forms you want, then click Ok to display those forms.
3. You can position sections these ways:
  - Click in a section, then drag the section to the appropriate position
  - Use the position options:



**Note:** You can also store the origin name for future use or indicate a fixed position.

5. Choose the File, Save option. Then choose the File, Check-in to Library option.



# CONVERTING CONDITIONAL FIELD LOGIC

IStream expression and conditional field logic statements are extracted from model documents and referenced in the DAL Script report and \*.expr files. Each conditional statement needs to be analyzed to determine if a Documaker rule can be used or if a DAL script is needed.

## Documaker Rules

Studio lets you use assign rules to control how information is merged onto forms, how that information is then processed, and how the information and those forms are output when using Documaker.

**Note:** Rules can be specific to an input file format, such as a flat file or XML. Use the Rules Reference to determine if the rule is appropriate for your implementation.

### Sample 1: IStream Conditional Statement

The IStream uses conditional statements to omit blank lines from an address. The migration to Documaker replaces the conditional if statement with the RemoveWhiteSpace rule to omit blank address lines.

IStream conditional statements map to	Documaker
<cGrpOffice> Group Office <cGrpOfficeAddLine1><IF(!EMPTY(cGrpOfficeAddLine2), CHR(13)+cGrpOfficeAddLine2,"")> <IF(!EMPTY(cGrpOfficeAddLine3), CHR(13)+cGrpOfficeAddLine3,"")>	RemoveWhiteSpace rule

## DAL Scripts

DAL scripts can be used by Documaker Server, Documaker Desktop, and Documaker Interactive. The DAL scripting language is used to create field calculations, form and section attachment rules, and to apply If/Then/Else logic when populating fields.

### Sample 2: IStream Conditional Statement

IStream Conditional Statements Map To	Documaker
<IF(!Empty(nRepresTelNum), "Tel. " + nRepresTelNum, "")>	DAL script

## Sample DAL Scripts

```
BeginSub RepresTelNumText
nRepresTelNum =?("nRepresTelNum")
If nRepresTelNum = ""
    Return(" ")
Else
    Return ("Tel." & " " & nRepresTelNum);
End
```

### Sample 3: if statement DAL Script

IStream Conditional Statements Map To	Documaker
{IF(nSLSpecAggrDedAmtValidPlansCntr > 1,"Plans " + cAssumpPlans ,"Plan " + cAssumpPlans)}	DAL script

```
BeginSub PlansText
#SLSSpecPlansCnt =? ("nSLSpecAggrDedAmtValidPlansCnt")
cAssumPlans = ? ("cAssumpPlans");
If #SLSSpecPlansCnt > 1;
    Return ("Plans" & " " & cAssumPlans);
Else
    Return ("Plan" & " " & cAssumPlans);
End
```

## BEST PRACTICE

For ease of maintenance and reuse, store business rules separately from forms, making updates and deployment easier and faster.

Use DAL libraries to share scripts across multiple forms.

## RELATED DOCUMENTATION

[Documaker Studio User Guide](#)

[Documaker Studio Help: key words, the name of the rule, such as SetAddr2 or DAL.](#)

[Frequently Asked Questions](#)

## Step 8. Documaker System Configuration

---

Documaker system configuration is the process of setting up your Documaker options. This includes defining users, print types, and rendering options. This step will guide through configuring your Documaker environment.

### **Step 8. Documaker System Configuration**

This step guides you through configuring your Documaker system files.

Defining User Roles &  
Responsibilities

Configure Distribution  
Channels &  
Recipients

Document Rendering

Configure Options

Configure System  
Files

Configuration  
Scenarios

## DEFINE USERS AND RESPONSIBILITIES

In Studio, only a System Administrator or User Administrator can access the Users module to set up user profile information. Important questions regarding user roles are:

- What is your team structure?
- How many users will use the product?
- What rights/access levels/roles will the users have?

## BEST PRACTICE

**Administrators** – Studio accommodates four types of administrators: System Administrator, User Administrator, Library Administrator, and Project Administrator. Depending on the organization of your company, you may have one type of administrator or all four. It is recommended that you have a System Administrator with access to all workspace functionality. This Administrator may grant rights to other users.

**User Administrator** is the administrator of the user database and activates and deactivates all security settings for all users except the System Administrator.

**Library Administrator** is responsible for promoting forms to the appropriate environment, manages the list of forms needed for deliveries, and has the authority to perform all tasks within the library and maintains the Field Dictionary. Some companies see this role as the gatekeeper.

**Developer** is responsible for building forms, coordinating section reuse, and maintaining the XDD extract dictionary.

**Reviewer/Tester** is responsible for final reviews and testing of the library.

## ADDING USERS

Choose the Manage, Users option to add user profiles. Studio shows you information about:

- The user's name and ID
- Who the user reports to
- Access rights and In Use status
- Security settings

Sample Users List

User ID	User Name	Password	Access Rights	In Use	Report To	Security	Message
JDD	John Demo	Change Password	9	No		Security	
JMD	Jane Demo	Change Password	0	No	JDD	Security	
USER1	User1	Change Password	9	No		Security	

---

**Note:** Your default user ID is the same as your Windows NT user ID.

---

## CHANGING PASSWORDS

A user that has access to his or her own record may change the descriptive name and password, but not the other attributes.

## ADDING SECURITY

Studio provides a Security wizard for adding and modifying user security records. There are several security options in Studio. Security options may correlate with project team roles and responsibilities.

Security Options	Description
System Administrator	Assign system administrator rights to a user.
User	Assign user administrator rights to a user.
Setting	Indicates if a user can change INI files and if a user has override rights.
Library	Assign library administration rights to a user.
Resource	Assign resource administration rights to a user.
Rights	Used to limit and restrict user rights.
Project	Assign overall project rights to a user.

## RELATED DOCUMENTATION

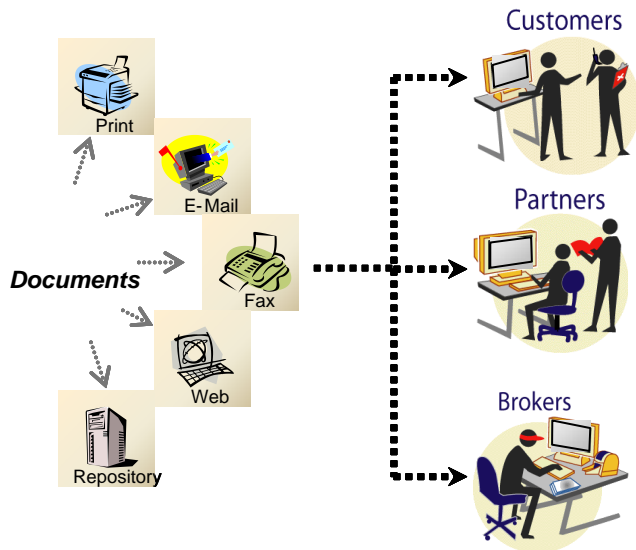
Documaker Studio User Guide

Documaker Studio Help: key words – users, security, password

## CONFIGURING DISTRIBUTION CHANNELS AND RECIPIENTS

Documaker uses Distribution Package options when delivering to multiple recipients through various delivery channels, through one XML request. Take an example of a document that needs to be delivered by fax to a group of customers. In addition, another version of the document needs to be sent by email to 2,000 managers. A third version of the same document needs to be saved to a web site for use by sales agents. All of these requirements can be handled by one request using Documaker.

Each recipient within a distribution chain has specific needs which Documaker takes into account. Documaker supports multiple delivery methods, such as, print, email, fax, web, and of course, archiving document sets to your repository.



Documaker writes recipient records for each member of the delivery chain and prints a separate copy of the document set for each member and, if required, with a unique mailer page. The mailer page can contain the name and address information for each recipient.

---

**Note:** Each delivery chain recipient receives identical copies of the document set (Proposal Documents).

---

## ADDING DISTRIBUTION CHANNELS

Documaker stores multiple sets of addressee information for a single transaction or document set. This addressee information is used to distribute the document set to various addressees once approved for distribution.

Studio lets you define the layout of addressee data that will be stored with the document set. The addressee data layout defines the fields, lengths, and attributes associated with each addressee. There may be multiple addressees that are identified by a single recipient classification, such as brokers, partners, and so on.

## ADDING RECIPIENTS


Studio associates a recipient name to forms within your document library, such as Insured or Customer. A sorting code is also defined which is used by batch implementations to sort output into a number of different files for delivery, such as grouping the recipient broker transactions into one file versus individual files. The description option is used with the field attribute Send Copy To option and displays the recipient name at print time in the Send Copy To field.

When adding, modifying, or deleting recipients, first open the workspace definition file and then click the recipient tab.

### Step 1. Open Workspace Definition File

- a. Open the workspace definition file.
- b. Click the Recipient tab to display the Recipient Options pane.

### Step 2. Adding Recipients

- a. Click  to add a recipient
- b. Add the recipient options:
  - ✓ Enter the Recipient Name
  - ✓ Enter the Recipient Code
  - ✓ Enter the Recipient Description
  - ✓ Enter the Address Map

---

**Note:** Repeat step 2 for each recipient identified in your requirement study document. The recipients added are used to assign recipients to documents and are used with Documaker, Documaker Desktop, and Documaker Interactive.

---

### Step 3. Save and Check-in the Workspace Definition file after adding recipients.

## RELATED DOCUMENTATION

Documaker Studio User Guide

Documaker Studio Help: key words – Fonts, Converting Fonts

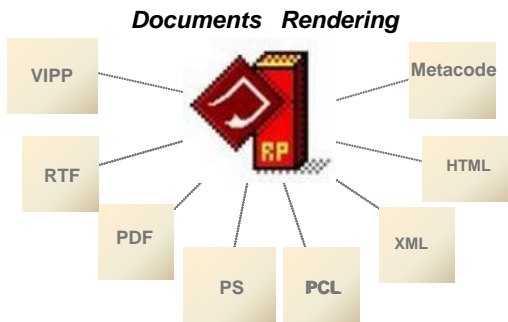
Frequently Asked Questions

Documaker Administration Guide



## DOCUMENT RENDERING

Documaker creates and renders documents to multiple formats and any document can be delivered by one or more delivery methods to one or more recipients. Documaker support the IStream rendering options plus others.



Document rendering options are defined within the system's INI files.

Documaker	IStream	Description
PDF	PDF	Oracle supports compression and page-at-a-time downloading. With page-at-a-time downloading (byte-serving), a web server sends only the requested page of information to the user, not the entire PDF document.
PostScript	PostScript	For the most part, system-produced PostScript output will run on any printer that supports PostScript Level 2.
PCL	PCL	The PCL language has evolved over time. For the most part, system-produced PCL output will run on any printer that supports PCL 5. To support color printing, the printer must support PCL 5c, which contains color extensions. To support more than two paper trays, the printer must support PCL 6.
BPD	TIF	The Bitmap Print Driver lets you specify a bitmap format to use, such as LOG, TIF, JPG, or BMP.
HTML	HTML	The HTML print driver includes support for: Boxes - solid and shaded colors only Bar codes Charts Vectors - solid and shaded colors only Logos - converted to JPG files by the driver Lines - solid and shaded colors only. Dashed lines are supported but do not take the line characteristics as specified. The spacing and length of dashed lines are defaulted by the HTML 4 specification. Shaded areas - solid and shaded colors only Text areas Text Variable fields
RTF	DOC	This is a format designed by Microsoft and supported by Word as well as several other editors or viewers.
Metacode	N/A	The Metacode language is the native mode language for Xerox 4000 and 9000 series printers. This language is superior to printing using line data with Xerox Laser Printing Systems (LPS). The advantages of using Metacode over line data printing include support for portrait and landscape text on the same page, support for different fonts on the same line, precise text positioning, and text justification. In addition, Metacode lets you merge multiple forms onto a single page.
VIPP	N/A	Variable Data Intelligent PostScript PrintWare (VIPP) is based on PostScript and works by extending

Documaker	IStream	Description
		<p>the PostScript programming language. VIPP can be used on any PostScript compatible printer, including Xerox and third-party network, workgroup, and production devices licensed for VIPP.</p> <p>VIPP is supported on these devices:</p> <ul style="list-style-type: none"> <li>• DocuPrint NPS (monochrome and color)</li> <li>• DocuPrint N-series</li> <li>• DocuSP controllers, including iGen3</li> <li>• DocuColor, EFI, and Creo controllers, (including iGen3)</li> </ul> <p>The Documaker VIPP print driver requires that you have VIPP version 5.3 or later installed on your printer's controller. To use the Documaker VIPP print driver, any supported device must have a local file system you can access to transfer resource files.</p>
PXL	N/A	Documaker includes a printer driver for PCL 6. PCL 6 is a newer version of the PCL language, not an superset of PCL 5. It is a completely different page description language (PDL). Documaker includes a PCL 6 driver because of its Unicode support.
XML	N/A	This driver lets you use Documaker Server to create Documaker standard XML output.

## ADDING DOCUMENT RENDERING OPTIONS

Group	Where used	Description
Printer	Documaker Desktop Documaker Interactive	Defines available printers used by Documaker Desktop.
Printers	Documaker Server	Defines all available printers used by Documaker Server.
PrtType:XXX	Documaker Desktop Documaker Server	Used to define the settings for each printer type listed under the Printer and Printers control groups, where xxx represents the name of the printer type. You can name it anything you like, but it must match with the value assigned to the Printer and Printers control groups.

The example Printer control group below has four output types that would be available to when using Documaker Desktop. Documaker Server, however, uses the Printers control group and has five output types. Based on this setup, RTF output is only provided with Documaker Desktop printing.

```

< Printer >
  PrtType      = EPT
  PrtType      = PDF
  PrtType      = PCL
  PrtType      = RTF
< Printers >
  PrtType      = EPT
  PrtType      = PDF
  PrtType      = PCL
  PrtType      = XER
  PrtType      = HTML
< PrtType:XXX >

```

## RELATED DOCUMENTATION

[Documaker Studio User Guide](#)

[Documaker Studio Help](#)

[Frequently Asked Questions](#)

[Documaker Administration Guide](#)

## PRINT TYPES

### PDF

Documaker produces PDF files which adhere to the PDF file format version 1.2. This version supports compression and page-at-a-time downloading. With page-at-a-time downloading (byte-serving), a web server sends only the requested page of information to the user, not the entire PDF document.

Set the LoadPrintOnly option to Yes when you are printing PDF output from Entry when there are Print Only forms in the form set. If you produce an invalid PDF file, check your form set definition file (.BDF, .GRP, or .FOR) to see if you have any print only forms. If so, change the LoadPrintOnly option to Yes. The default is No.

PDF viewers can only view one transaction in a file. So, you must generate a separate PDF file for every transaction in a batch. For two or three step processing mode (GenTrn, GenData, GenPrint) you must use the Print control group CallbackFunc option set to MultiFilePrint to accomplish this. For single-step processing mode (GenData), you use the PrintFormSet control group's MultiFilePrint INI option instead.

### PostScript

Documaker supports PostScript Type 1 fonts. The PST print driver supports up to nine paper trays. A PostScript Printer Definition (PPD) file is supplied with a PostScript printer. This file contains information about printer-specific features. If the PrinterModel option contains the name of a PPD file, this file must be in the directory specified in the DefLib option in the XXXX control group.

Documaker supports PostScript Type 1 fonts. Postscript fonts must reside in the directory specified in the FontLib option under the MasterResource control group when the DownloadFonts option is set to Yes.

### PCL

The system-produced PCL output will run on any printer that supports PCL 5. To support color printing, the printer must support PCL 5c, which contains color extensions. To support more than two paper trays, the printer must support PCL 6. The PCL print driver supports up to nine paper trays.

### BPD

The Bitmap Print Driver runs on Windows implementations of Documaker Desktop, Documaker Server, and Documaker Studio. You can view the entire list of configuration options in Studio.

The Bitmap Print Driver can produce 1-bit, 8-bit, and 24-bit bitmaps. The number of bits per pixel generated is based on the highest number of bits used. For example, if the page includes a 1-bit bitmap, an 8-bit bitmap, and a 24-bit bitmap, the Bitmap Print Driver will produce a 24-bit bitmap of the page. You can use Graphics manager in Studio to find out the type of color bitmap you have.

---

**Note:** Other Oracle print drivers may accept bitmaps which are not 1-bit or 24-bit graphics and produce different results.

---

### HTML

The HTML print driver includes support for:

- Boxes - solid and shaded colors only

- Bar codes
- Charts
- Vectors - solid and shaded colors only
- Logos - converted to JPG files by the driver
- Lines - solid and shaded colors only. Dashed lines are supported but do not take the line characteristics as specified. The spacing and length of dashed lines are defaulted by the HTML 4 specification.
- Shaded areas - solid and shaded colors only
- Text areas
- Text
- Variable fields

To make sure the page looks the same to all users, only use fonts that are common to all users. Otherwise, fonts are substituted. The look of the printed HTML file depends on the settings in effect for your browser.

### **Email (EPT)**

Documaker allows subject and message information and use the email address book when printing from Documaker Desktop using the EPT print driver.

## CONFIGURE OPTIONS

Configuration options tell the system how you want it to operate. These options are stored in configuration files. The configuration files consist of control groups and options. Here is an example:

```
< ControlGroup >  
  Option1 = Parameters  
  Option2 = Parameters  
  OptionN = Parameters
```

### **FSIUSER (Local tab)**

The FSIUSER controls settings which can vary between resource libraries, such as sorting options, archival mode, and import/export ability, as well as individual user options.

### **FSISYS (System tab)**

The FSISYS controls information related to the entire system, such as system settings and program function calls.

The system loads the FSIUSER first, then finds the name and location of the FSISYS file from the ENVIRONMENT group, and loads it. Since the FSIUSER is loaded first, the options within this group usually override the FSISYS options.

In a multi-user environment, you may not want to let all users change system-level settings. In some cases, you may not want some users to change individual settings. The system lets you determine which configuration options are available to a specific user. This is done via security definitions which are assigned to these options by the system administrator. For instance, you can let a user change local settings (FSIUSER), but not change the system settings (FSISYS). Or, you can let a user review these settings (local and system), but not change them.

## RELATED DOCUMENTATION

[Documaker Studio User Guide](#)

[Documaker Studio Help](#)

[Documaker Administration Guide](#)

## CONFIGURE SYSTEM FILES

During processing, several intermediary files are created to identify transactions and recipients that will be used to generate print streams. Documaker also generates files or updates indexes of database tables as a final result of processing. These intermediary files and indexes can take several formats -- including xBase, ASCII, and SQL database tables -- in Oracle, DB2, or MS SQL. Many of these files or index values have a default or internal layout known to the system, but others require the system administrator to create a data format definition (DFD) file which tells Documaker to generate the desired layout. The files or index values used during system processing are:

- Transaction - trndfdfl.dfd or this INI setting
- New transaction - trndfdfl.dfd or this INI setting
- Recipient batch - rcbfdfdfl.dfd or this INI setting
- Archive index - appidx.dfd or this INI setting
- WIP index - wip.dfd (not required) or this INI setting
- WIP data - wipdata.dfd (not required) or this INI setting
- Help files are no longer available within the system
- Table files are now table look-up XML files stored in the Documaker master resource library

Of the files listed above, only these files require an external DFD file. The use of an external DFD file for the other files is optional.

File	External DFD File
Transaction files	TRNDFDFL.DFD
New transaction files	TRNDFDFL.DFD
Recipient batch files	RCBDFDFL.DFD
Manual batch files	RCBDFDFL.DFD
Application index files	APPIDX.DFD
WIP files	WIP.DFD (optional)

The WIP file may optionally have an external DFD. If there is no external WIP DFD file, the internal record structure as coded in the system is used. The help and table files do not support the use of external DFD files.

## RELATED DOCUMENTATION

[Documaker Studio User Guide](#)

[Documaker Studio Help](#)

[Documaker Administration Guide](#)

# ISTREAM TO DOCUMAKER SCENARIOS

This topic illustrates four scenarios and how each customer's requirements and scope can be mapped to a Documaker products or component. Each scenario illustrates how to map:

- IStream products to Documaker products
- IStream workflow diagram to Documaker workflow diagram

## SCENARIO 1: TYPICAL PUBLISHER SCENARIO

Scenario 1 depicts a typical IStream Publisher environment that relies on other applications from front-end processing and a third-party repository for document storage. This scenario is repeated at many IStream sites with variations including how the data used for document generation is stored (ODBC or XML), what outputs are produced, when output is printed, and how the Publisher response is processed.

The Admin Policy System submits a message to a queue that is picked up by the Publisher Integration Client (PIC). PIC translates the message into the Publisher XML Request syntax and submits the request to IStream Publisher's submission queue. PIC also posts a status update in the tracking table.

IStream Publisher picks up the request from the queue, processes the request and generates PDF and PCL files. The PDF files are archived and the PCL files are stored in a system folder. After generating the PDF files, PIC posts a status update in the tracking table.

The PCL files are not immediately printed but are produced by the system throughout the day. Before printing, the PCL files are combined in proper order. After printing the PCL files, PIC posts a status update in the tracking table and sends an update message via a queue back to the Policy Admin system.

---

**Note:** The Publisher Integration Client (PIC or IC) is not part of the Publisher product. All Publisher Integration Clients are built by the customer or Services on behalf of customer and are maintained by the customer.

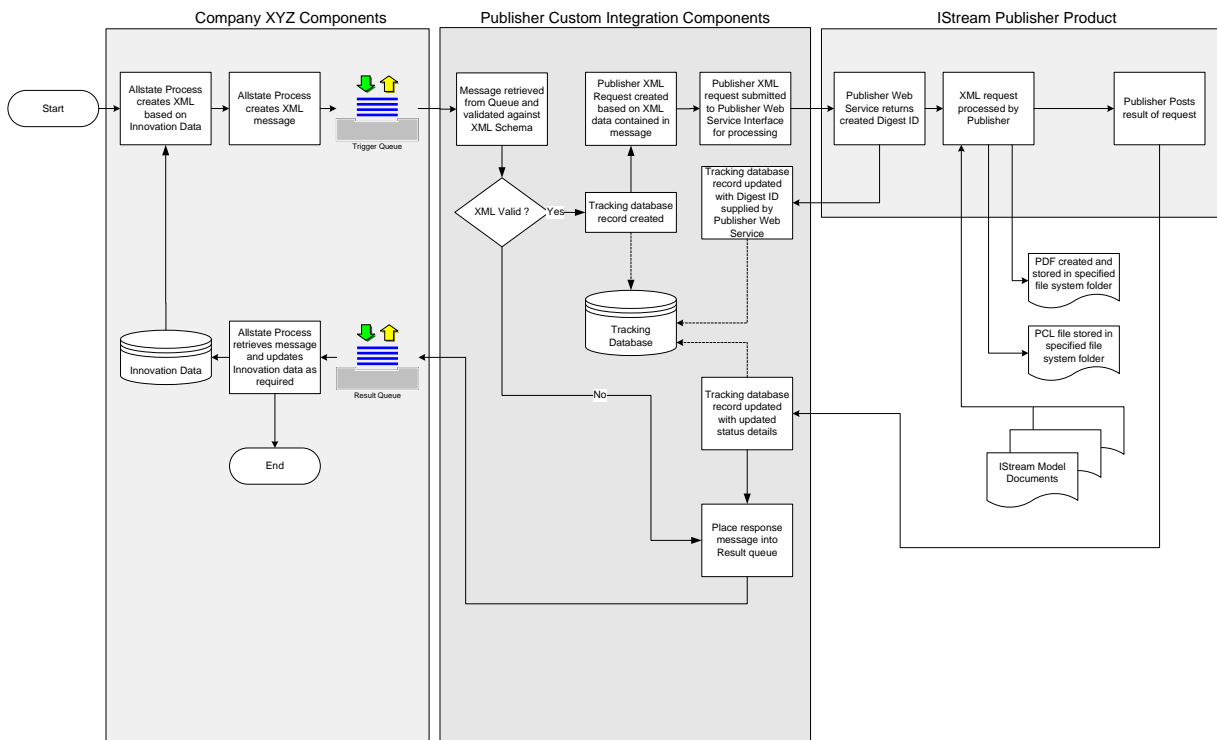
---

## Workflow

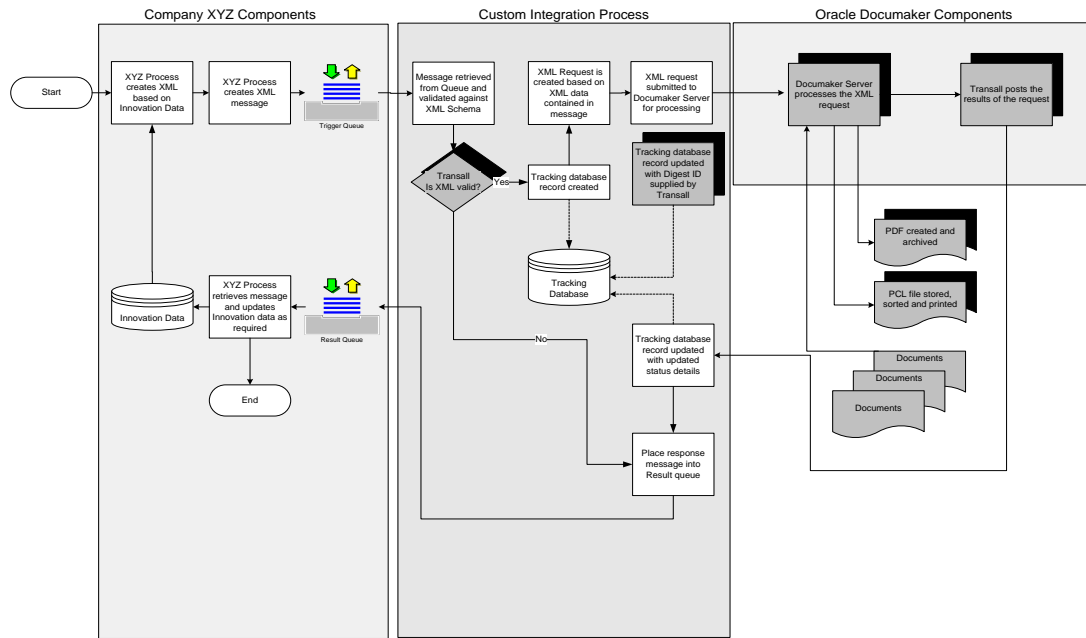
Steps	Activity	IStream	Documaker
1	Admin Policy System submits message to a queue.		
2	The PIC (Publisher Integration Client) translates the message into the Publisher XML Request syntax and submits the request to Publisher's submission queue.	Custom code	Internet Document Server and Transall
3	PIC (Publisher Integration Client) updates the tracking table in a shared data base.	Custom code	Transall
4	Publisher picks up the request from the queue and processes the request.	IStream Publisher	Internet Document Server
5	PDF documents are created and archived.	IStream Publisher	Documaker Server
6	For the PDF creation response: The PIC (Publisher Integration Client) monitors Publisher's response queue for the completion response and sends an update message via a queue back to the Policy Admin system and updates the tracking table.	Custom Code	Internet Document Server and Transall
7	PCL print stream documents are produced by the system and PCL print streams are combined in the proper order for printing.	IStream Publisher and custom code	Documaker Server and custom Code
8	For the PCL creation response: The PIC (Publisher Integration Client) monitors Publisher's response queue for the completion response and sends an update message via a queue back to the Policy Admin system and updates the tracking table.	Custom code	Internet Document Server and Transall



## IStream Workflow



## Oracle Documaker Workflow



## SCENARIO 2: SCENARIO USING DISPATCHER COMPONENTS

Scenario 2 is typical of many smaller IStream customers who used the sunset Dispatcher component that was a database driven batch engine that uses the Assembly engine for document automation. Scenario two uses Integration Client, a prebuilt, configurable IStream Publisher Client produced by the Services department.

The Integration Client (IC) is a software application that can monitor different locations such as folders and database tables to build, deliver, and monitor IStream Publisher requests. The IC has the ability to monitor database tables for data that is used to build and send requests to IStream Publisher. Based on the presence of this data and its configuration, the IC will build the appropriate IStream Publisher requests. The IC provides a robust mechanism for integrating with IStream Publisher through database table updates; similar to Request Builder and the Request Table in Dispatcher. The IC can be configured to support any type of IStream Publisher Request.

The IC uses database Trigger tables as the integration point between the data entry client software application and the IStream document creation process. The data retrieved from the IC trigger tables (by the IC) is processed based on rules defined as part of the IC configuration. Based on these rules, specific XML requests are created from XML templates and submitted to IStream Publisher for processing. Request status details are stored and updated within the IC trigger table and tracking tables.

The IC runs continuously as a Windows service and checks the trigger tables every  $n$  seconds. The IC can be configured to support multiple document automation workflows. The XML templates allow dynamic request creation that can support dynamic file naming and other Policy data based requirements.

---

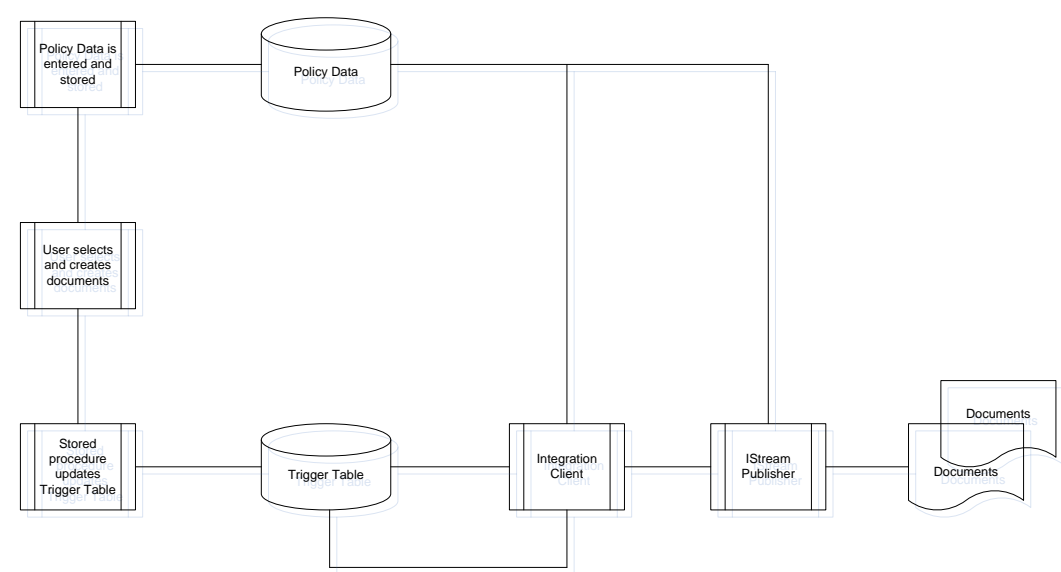
**Note:** The Integration Client (IC) is not part of the Publisher product. All Publisher Integration Clients are built by the customer or Services on behalf of customer and are maintained by the customer.

---

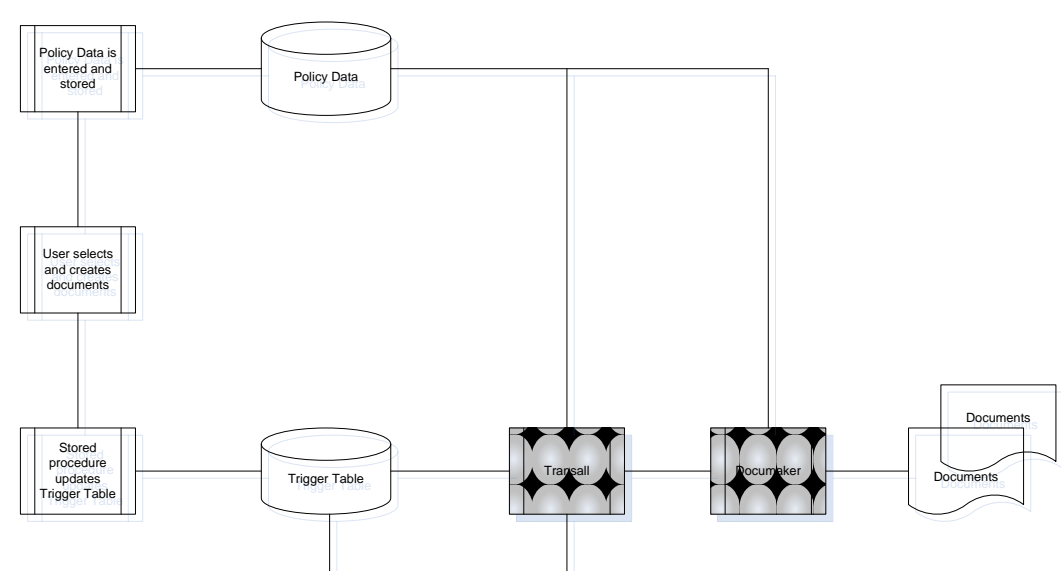
Workflow

Steps	Activity	IStream	Documaker
1	Admin Policy System writes a record to the trigger table.		
2	The IC (Integration Client) checks the trigger table every n seconds and submits a request to Publisher's submission queue.	Custom code	Transall
3	Publisher picks up the request from the queue, processes the request, and produces and delivers the desired output.	IStream Publisher	Documaker Server
4	The IC (Integration Client) monitors Publisher's response queue for the completion response and updates the record in the trigger table with a completion status.	Custom Code	Transall

IStream Workflow



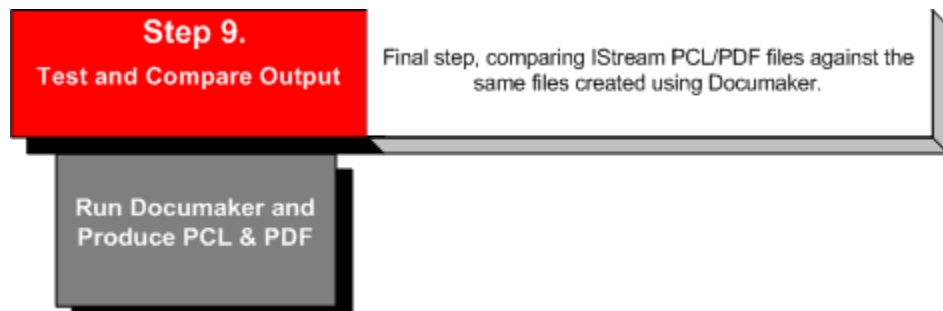
Oracle Documaker Workflow



## Step 9. Test and Compare

---

This is the final step in the IStream to Documaker migration so you should have completed steps 1 through 8. If you have not successfully completed steps 1 – 8, then you are not ready to start this step.



## RUN DOCUMAKER TO PRODUCE PCL AND PDF FILES

This step confirms that the processing results from your new Documaker system are consistent with the processing results from your old IStream application. The same input data should be run through the old and new applications and the output files compared.

The objectives of testing during the migration to Documaker are to:

- Make sure the sections and forms migrated correctly
- Demonstrate consistency between the two applications
- Demonstrate the integrity of the Oracle Documaker applications

Table 7: Completed Migration Checklist

Step	Sub-Step	Complete
<b>1. Requirement Study</b>		
	Current and proposed workflows	✓
	Model document and data analysis	✓
<b>2. Project Planning</b>		
	Resource requirements	✓
	System requirements	✓
	Education needs	✓
	Oracle support	✓
<b>3. Migration Overview</b>		
	Pre-migration planning	✓
	Migration considerations	✓
	Migration utility parameters	✓
	Migration process and checklist	✓
<b>4. Migration Utility</b>		
	Set Migration utility parameters	✓
	Migration output files	✓
	Migration reports	✓
	Run the Migration utility	✓
<b>5. Studio Pre-Import Tasks</b>		
	Create recommended directories	✓
	Create workspace	✓
	Convert fonts	✓
	Create font cross-reference file (FXR)	✓
<b>6. Studio Import</b>		
	Import FDB and XDD	✓
	Convert RTF files	✓
	Import forms list and form attachment rules	✓
<b>7. Studio Post-Import Tasks</b>		
	Apply header and footer sections	✓
	Create Table of Contents (Optional)	✓
	Review and modify graphics (Optional)	✓
	Apply section positions	✓
	Convert expressions to DAL scripts	✓
<b>8. Documaker Configuration</b>		
	Define user roles and responsibilities	✓
	Configure distribution channels and recipients	✓
	Document rendering	✓
	Configure options	✓
	Configure system files	✓
<b>9. Test and Compare Output</b>		
	Run Documaker to produce PCL and PDF	
	Compare output	✓
<b>10. Deployment</b>		
	Deploy library	

## Step 10. Deployment

---

Library deployment is the last development step. Using Documaker's deployment tool you can deploy a library to the next environment in your document automation life cycle.



The diagram consists of a red rectangular box at the top left containing the text 'Step 10. Deployment'. To its right is a light gray rectangular box containing the text 'Final step, uses Documaker's deployment tool to deploy your library to the next environment in your document automation life cycle.' Below the red box is a dark gray rectangular box containing the text 'Library Deployment'. A horizontal line extends from the right side of the red box, passing behind the light gray box and ending at the right edge of the page.

### Step 10. Deployment

Final step, uses Documaker's deployment tool to deploy your library to the next environment in your document automation life cycle.

Library Deployment



# LIBRARY DEPLOYMENT

Studio's deployment feature consists of several interrelated steps with possible transitions between them. Deployment can be interpreted as a *general process* that has to be customized according to specific project requirements or characteristics.

## DEPLOYMENT STEPS

### Requirements Phase

Deployment is an important and critical undertaking with any development project. As a best practice, the deployment requirements phase should take into consideration the following questions:

**Deployment owner**, who will own the deployment process?

**Deployment type**, what type of deployment your company will use?

**Deployment resources**, how are resources tagged for deployment?

**Deployment scripts**, what and how many deployment scripts are needed?

# Glossary of Terms

Group	Term	Definition and usage
Application	Definitions	Used to define a valid key combination comprised of a Key1, Key2, and eventually Key3 into Application Definition (BDF) files. Also used to define recipients, categories, and transactions (for an entry environment). These files are used in both the batch and workstation and are required.
Application	Triggers	Used to create scripts that automate certain tasks within the processing environment. These scripts are created using Document Automation Language (DAL). These files are used in both the batch and workstation.
Content	Forms	Used to build a list of sections (FOR files) that create the form and to store section triggering information.
Content	Graphics	Used to resize, reverse, rotate, and manipulate bitmap graphics that are used on sections.
Content	Paragraph Lists	Used to build a list of paragraphs (PSL files) that are used for paragraph selection at run-time.
Content	Paragraphs	Used to create and manage selectable paragraphs (PAR files) that can be assembled for a field at run-time.
Content	Sections	Used to create and manage the sections (FAP files) that make up the forms. Stores field level and some section level rule information.
Dictionary	Common Fields	Used to create and maintain common field information (FDB) to make setting up and creating variable fields on sections (FAP) faster and more consistent.
Dictionary	Data Extract	Used to create and maintain mapping information for runtime data files (the default extract is symbol.xdd).
Manage	Libraries	Used to manage the libraries that contain the forms, sections, graphics, and other resources.
Manage	Projects	Let you associate a resource with a particular job or task. This module is not fully released and is available for BETA testing.
Manage	Workspace	Used by Studio to manage the Master Resource Library (MRL) you are working with. Workspace files (DXM, DXS) store information about your INI files and paths and resources you are currently working with that appear in your workspace tree.
System	Fonts	Used to create and maintain the cross-reference table that serves as the bridge between the Documaker programs and the physical fonts.
System	Form Lists	Used to define the list of forms available to a given Application Definition and to store form triggering information (GRP files). These files are used in both the batch and workstation and are required.
System	Settings	Used to create or update Studio options and INI files.
System	Styles	Used to build style lists (STY files) that can be used when making sections. You define which style file to use in your application definition.
System	Templates	Used to build form templates (TPL files) that can be used when creating forms.
System	Users	Used to create or update user security.