

Oracle® Documaker

Documaker Studio

User Guide

version 11.5

Part number: E16256-01

March 2012

Copyright © 2009, Oracle and/or its affiliates. All rights reserved.

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.

THIRD PARTY SOFTWARE NOTICES

This product includes software developed by Apache Software Foundation (<http://www.apache.org/>).

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright © 2000-2009 The Apache Software Foundation. All rights reserved.

This product includes software distributed via the Berkeley Software Distribution (BSD) and licensed for binary distribution under the Generic BSD license.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright © 2009, Berkeley Software Distribution (BSD)

This product includes software developed by the JDOM Project (<http://www.jdom.org/>).

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE JDOM AUTHORS OR THE PROJECT CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (C) 2000-2004 Jason Hunter & Brett McLaughlin. All rights reserved.

This product includes software developed by the Massachusetts Institute of Technology (MIT).

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright © 2009 MIT

This product includes software developed by Jean-loup Gailly and Mark Adler. This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Copyright (c) 1995-2005 Jean-loup Gailly and Mark Adler

This software is based in part on the work of the Independent JPEG Group (<http://www.ijg.org/>).

This product includes software developed by the Dojo Foundation (<http://dojotoolkit.org>).

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 2005-2009, The Dojo Foundation. All rights reserved.

This product includes software developed by W3C.

Copyright © 2009 World Wide Web Consortium, (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University). All Rights Reserved. (<http://www.w3.org/Consortium/Legal/>)

This product includes software developed by Mathew R. Miller (<http://www.bluecreststudios.com>).

Copyright (c) 1999-2002 ComputerSmarts. All rights reserved.

This product includes software developed by Shaun Wilde and distributed via Code Project Open License (<http://www.codeproject.com>).

THIS WORK IS PROVIDED "AS IS", "WHERE IS" AND "AS AVAILABLE", WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES OR CONDITIONS OR GUARANTEES. YOU, THE USER, ASSUME ALL RISK IN ITS USE, INCLUDING COPYRIGHT INFRINGEMENT, PATENT INFRINGEMENT, SUITABILITY, ETC. AUTHOR EXPRESSLY DISCLAIMS ALL EXPRESS, IMPLIED OR STATUTORY WARRANTIES OR CONDITIONS, INCLUDING WITHOUT LIMITATION, WARRANTIES OR CONDITIONS OF MERCHANTABILITY, MERCHANTABLE QUALITY OR FITNESS FOR A PARTICULAR PURPOSE, OR ANY WARRANTY OF TITLE OR NON-INFRINGEMENT, OR THAT THE WORK (OR ANY PORTION THEREOF) IS CORRECT, USEFUL, BUG-FREE OR FREE OF VIRUSES. YOU MUST PASS THIS DISCLAIMER ON WHENEVER YOU DISTRIBUTE THE WORK OR DERIVATIVE WORKS.

This product includes software developed by Chris Maunder and distributed via Code Project Open License (<http://www.codeproject.com>).

THIS WORK IS PROVIDED "AS IS", "WHERE IS" AND "AS AVAILABLE", WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES OR CONDITIONS OR GUARANTEES. YOU, THE USER, ASSUME ALL RISK IN ITS USE, INCLUDING COPYRIGHT INFRINGEMENT, PATENT INFRINGEMENT, SUITABILITY, ETC. AUTHOR EXPRESSLY DISCLAIMS ALL EXPRESS, IMPLIED OR STATUTORY WARRANTIES OR CONDITIONS, INCLUDING WITHOUT LIMITATION, WARRANTIES OR CONDITIONS OF MERCHANTABILITY, MERCHANTABLE QUALITY OR FITNESS FOR A PARTICULAR PURPOSE, OR ANY WARRANTY OF TITLE OR NON-INFRINGEMENT, OR THAT THE WORK (OR ANY PORTION THEREOF) IS CORRECT, USEFUL, BUG-FREE OR FREE OF VIRUSES. YOU MUST PASS THIS DISCLAIMER ON WHENEVER YOU DISTRIBUTE THE WORK OR DERIVATIVE WORKS.

This product includes software developed by PJ Arends and distributed via Code Project Open License (<http://www.codeproject.com>).

THIS WORK IS PROVIDED "AS IS", "WHERE IS" AND "AS AVAILABLE", WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES OR CONDITIONS OR GUARANTEES. YOU, THE USER, ASSUME ALL RISK IN ITS USE, INCLUDING COPYRIGHT INFRINGEMENT, PATENT INFRINGEMENT, SUITABILITY, ETC. AUTHOR EXPRESSLY DISCLAIMS ALL EXPRESS, IMPLIED OR STATUTORY WARRANTIES OR CONDITIONS, INCLUDING WITHOUT LIMITATION, WARRANTIES OR CONDITIONS OF MERCHANTABILITY, MERCHANTABLE QUALITY OR FITNESS FOR A PARTICULAR PURPOSE, OR ANY WARRANTY OF TITLE OR NON-INFRINGEMENT, OR THAT THE WORK (OR ANY PORTION THEREOF) IS CORRECT, USEFUL, BUG-FREE OR FREE OF VIRUSES. YOU MUST PASS THIS DISCLAIMER ON WHENEVER YOU DISTRIBUTE THE WORK OR DERIVATIVE WORKS.

This product includes software developed by Erwin Tratar. This source code and all accompanying material is copyright (c) 1998-1999 Erwin Tratar. All rights reserved.

THIS SOFTWARE IS PROVIDED "AS IS" WITHOUT EXPRESS OR IMPLIED WARRANTY. USE IT AT YOUR OWN RISK! THE AUTHOR ACCEPTS NO LIABILITY FOR ANY DAMAGE/LOSS OF BUSINESS THAT THIS PRODUCT MAY CAUSE.

This product includes software developed by Sam Leffler of Silicon Graphics.

THE SOFTWARE IS PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EXPRESS, IMPLIED OR OTHERWISE, INCLUDING WITHOUT LIMITATION, ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

IN NO EVENT SHALL SAM LEFFLER OR SILICON GRAPHICS BE LIABLE FOR ANY SPECIAL, INCIDENTAL, INDIRECT OR CONSEQUENTIAL DAMAGES OF ANY KIND, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER OR NOT ADVISED OF THE POSSIBILITY OF DAMAGE, AND ON ANY THEORY OF LIABILITY, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE

Copyright (c) 1988-1997 Sam Leffler
Copyright (c) 1991-1997 Silicon Graphics, Inc.

This product includes software developed by Guy Eric Schalnat, Andreas Dilger, Glenn Randers-Pehrson (current maintainer), and others. (<http://www.libpng.org>)

The PNG Reference Library is supplied "AS IS". The Contributing Authors and Group 42, Inc. disclaim all warranties, expressed or implied, including, without limitation, the warranties of merchantability and of fitness for any purpose. The Contributing Authors and Group 42, Inc. assume no liability for direct, indirect, incidental, special, exemplary, or consequential damages, which may result from the use of the PNG Reference Library, even if advised of the possibility of such damage.

This product includes software components distributed by the Cryptix Foundation.

THIS SOFTWARE IS PROVIDED BY THE CRYPTIX FOUNDATION LIMITED AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE CRYPTIX FOUNDATION LIMITED OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE

Copyright © 1995-2005 The Cryptix Foundation Limited. All rights reserved.

This product includes software components distributed by Sun Microsystems.

This software is provided "AS IS," without a warranty of any kind. ALLEXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE HEREBY EXCLUDED. SUN AND ITS LICENSORS SHALL NOT BE LIABLE FOR ANY DAMAGES SUFFERED BY LICENSEE AS A RESULT OF USING, MODIFYING OR DISTRIBUTING THE SOFTWARE OR ITS DERIVATIVES. IN NO EVENT WILL SUN OR ITS LICENSORS BE LIABLE FOR ANY LOST REVENUE, PROFIT OR DATA, OR FOR DIRECT, INDIRECT, SPECIAL, CONSEQUENTIAL, INCIDENTAL OR PUNITIVE DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF THE USE OF OR INABILITY TO USE SOFTWARE, EVEN IF SUN HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Copyright (c) 1998 Sun Microsystems, Inc. All Rights Reserved.

This product includes software components distributed by Dennis M. Sosnoski.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright © 2003-2007 Dennis M. Sosnoski. All Rights Reserved

It also includes materials licensed under Apache 1.1 and the following XPP3 license

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright © 2002 Extreme! Lab, Indiana University. All Rights Reserved

This product includes software components distributed by CodeProject. This software contains material that is © 1994-2005 The Ultimate Toolbox, all rights reserved.

This product includes software components distributed by Geir Landro.

Copyright © 2001-2003 Geir Landro (drop@destroydrop.com) JavaScript Tree - www.destroydrop.com/hjavadscripts/tree/version0.96

This product includes software components distributed by the Hypersonic SQL Group.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE

Copyright © 1995-2000 by the Hypersonic SQL Group. All Rights Reserved

This product includes software components distributed by the International Business Machines Corporation and others.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 1995-2009 International Business Machines Corporation and others. All rights reserved.

This product includes software components distributed by the University of Coimbra.

University of Coimbra distributes this software in the hope that it will be useful but DISCLAIMS ALL WARRANTIES WITH REGARD TO IT, including all implied warranties of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. In no event shall University of Coimbra be liable for any special, indirect or consequential damages (or any damages whatsoever) resulting from loss of use, data or profits, whether in an action of contract, negligence or other tortious action, arising out of or in connection with the use or performance of this software.

Copyright (c) 2000 University of Coimbra, Portugal. All Rights Reserved.

This product includes software components distributed by Steve Souza.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright © 2002, Steve Souza (admin@jamonapi.com). All Rights Reserved.

This product includes software developed by the OpenSymphony Group (<http://www.opensymphony.com/>.)"

Copyright © 2001-2004 The OpenSymphony Group. All Rights Reserved.

CONTENTS

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

Chapter 1: Using Documaker Studio

2 Introduction

- 2 Documaker System Overview
 - 3 Documaker Server
 - 5 Documaker Server Logical System flow
 - 7 Documaker Studio Workspace Concept

8 Studio Options

10 Creating a Multi-user Development Environment

12 Starting Studio

16 Using System Menus

- 18 File Menu
- 20 Edit Menu
- 21 View Menu
- 23 Manage Menu
- 26 Help Menu

27 Creating Workspaces

34 Joining a Workspace

Chapter 2: Working with Settings, Users, and Security

38 Working With INI Options

- 40 Setting INI Options
- 43 Choosing Options by Topic

46 Defining Studio Settings

- 47 Working with Configuration Options
- 50 Defining Workspace Options
- 52 Working with Common/Document View Settings
- 54 Working with Form Options
- 56 Working with Section Options
 - 56 Setting Document View Options
 - 58 Setting Section Filter Options
- 60 Working with Graphic Options
- 62 Working with Trigger Options
- 64 Setting Up Printstream Analyzer

66 Managing Users

- 67 Adding Users
- 69 Changing User Information
- 69 Deleting Users
- 69 Setting Rights Using the Security Wizard
- 75 Cloning User Rights
- 76 Logging In

77 Storing User Information in Another Database**78 Adding Security at the Resource Level****80 Using the LDAP Security Model****Chapter 3: Working with Application Definition Files****90 Overview****91 Defining an Application Definition****94 Defining Form Lists**

- 94 Adding a Form List
- 96 Deleting a Form List

97 Defining Recipients

- 98 Adding a Recipient
- 98 Deleting a Recipient

99 Defining Categories

- 100 Adding a Category
- 100 Deleting a Category

101 Defining Transaction Types

- 102 Adding a Transaction Code
- 102 Deleting a Transaction Code

103 Using Regional Date Processing

- 103 Setting up Regional Date Processing
 - 105 Adding Tokens and Shortcuts
- 107 Setting Up Regional Date Processing Rules

112 Generating an Application Definition Report**Chapter 4: Working with Form Lists****116 Creating a Form List****118 Adding a Form to a Form List****120 Setting Up Triggers**

- 121 Assigning Form Level Triggers
- 123 Assigning Triggers at Key2 Levels
 - 123 Defining a Key3 Trigger

124 SetRecip Options**Chapter 5: Managing Forms****126 Overview**

- 127 Using Embedded Sections
 - 128 Editing an Embedded Section
 - 129 Unembedding an Embedded Section

132 Using the Screen

- 133 Using the Menu Bar
 - 133 Using the Section Menu
 - 134 Using the Trigger Menu
- 135 Using the Workspace Menu

- 136 Using the Toolbar
- 138 Creating a Form**
- 140 Modifying Forms**
 - 143 Forcing Page Breaks in Subforms
 - 144 Generating Readability Statistics
- 147 Using Form Templates**
 - 147 Using Template Manager
 - 149 Using Form Manager
 - 150 Saving a Form as a Template
 - 150 Controlling Access to Templates
 - 151 Library Manager and Reports
- 153 Setting Up Triggers**
 - 155 Editing a Trigger
 - 155 Using the DAL Trigger Report
- 157 Handling Overflow**
 - 160 Overflow Examples
 - 161 User Functions
 - 163 Overflow Variables
- 165 SetRecip Options**
 - 166 Understanding Recipient Counts
 - 168 Using Variables to Set Copy Counts

Chapter 6: Working with Sections

- 172 Overview**
- 173 Using the Screen**
 - 174 Using the Menu Bar
 - 174 Using the Edit Menu
 - 175 Using the View Menu
 - 176 Using the Insert Menu
 - 177 Using the Format Menu
 - 179 Using the Arrange Menu
 - 180 Using the Tools Menu

- 182 Using the Right Click Menus
- 185 Using the Toolbar
- 189 Creating a Section**
- 191 Opening a Section**
 - 194 Working with Embedded Sections
- 195 Adding Objects**
- 197 Creating Bar Codes**
 - 198 Adding a Bar Code Object
 - 198 Adding a Bar Code as a Field
 - 199 Using Data Matrix 2-D Bar Codes
 - 202 Using PDF417 Bar Codes
 - 203 Inserting PLANET Code Bar Codes
 - 204 Generating a USPS Intelligent Mail Bar Code (4-State Customer Bar Code)
- 206 Adding Bookmarks**
- 208 Adding Charts**
 - 209 Creating a Chart
 - 213 Entering Data
 - 214 Assigning Variable Fields
 - 215 Adding a Series
 - 217 Setting Graph Properties
 - 218 Setting Legend Properties
 - 220 Combining Charts
- 222 Setting Header and Footer Print Options**
- 224 Creating Headers and Footers for Multipage Sections**
- 225 Defining Paragraphs**
 - 225 Creating Paragraphs
 - 226 Inserting Fields
 - 227 Formatting Paragraphs
 - 229 Defining Paragraph Properties
 - 230 Defining Tab Stops
 - 230 Defining Border Properties

231 Defining Shade Properties

232 Defining Bullet Properties

234 Creating Paragraph Lists

235 Inserting a Paragraph List into a Section

237 Defining Tables

239 Setting Table Options

240 Understanding XDD Name References

241 Creating Tables of Contents, Tables of Figures, and Indexes

242 Defining Tables of Contents, Tables of Figures, and Indexes

244 Defining Levels

246 Generating Tables of Contents, Tables of Figures, or an Index

247 Adding Comments to Documaker Objects

249 Using the Task List

251 Creating Text Labels

254 Printing a Form, Section, or Paragraph

256 Using Printcommander to Create Sections

256 Downloading and Installing Printcommander

257 Creating a Virtual Printer

258 Configuring a Docucorp Virtual Printer

258 Installing Tagcommander

258 Installing Docucreate

259 Configuring NetMove

262 Creating a FAP File

263 Possible Errors

263 Uninstalling Control Panel, Common Objects, and
Printcommander

263 Updating Control Panel, Common Objects, and
Printcommander

Chapter 7: Working with Graphics

266 Using the Screen

267 Using the Menu Bar

267 Using the Graphic Menu

269 Using the Right-Click Menu

272 Using the Toolbar

274 Managing Graphics

275 Positioning JPEG Objects

275 Importing Color Bitmaps

276 Converting Files

277 Opening a Graphic

278 Importing a Signature or Graphic

278 Editing a Graphic

280 Setting Rotation Names

281 Changing the Resolution

281 Setting Ruler and Grid Options

Chapter 8: Creating Scripts

284 Using the Screen

285 Using the Menu Bar

285 Using the Edit Menu

286 Using the Script Menu

287 Using the Right-Click Menu

288 Using the Toolbar

290 Overview

292 Checking Syntax

Chapter 9: Managing Resources

298 Overview

299 Terminology

301 Concepts

301 Understanding Libraries

303 Processing with Effective Dates

303 Retrieving Resources with Version and Revision Numbers

305 Stringently Checking Resources

307	Understanding Run Dates
309	How It All Works
309	In Documaker Workstation
312	In Documaker Server
314	In Docupresentation (IDS)
315	Managing Workflow
316	Creating Libraries
317	Using xBase and CARFiles
317	Creating the CARFile and Index Files
319	Using the DB2 Native Driver
319	Creating the Database and Tables
321	Sample INI Options
324	Using the DB2 ODBC Driver
324	Creating the Database and Tables
326	Sample INI Options
329	Using the SQL Server ODBC Driver
329	Creating the Database and Tables
329	Sample INI Options
332	Using the Oracle ODBC Driver
332	Data Format Definition (DFD) Requirements
333	Creating the Database and Tables
333	Sample INI Options
337	Using Documanager
344	Working with Libraries
345	Opening a Library
347	Adding Resources to a Library
350	Importing Files
351	Importing Libraries
352	Migrating a Workspace Library
354	Checking Out Resources
356	Checking In Resources
358	Unlocking Resources
359	Promoting Resources

- 362 Performing the Promotion
- 363 Filtering Resources
- 365 Editing Resource Information
- 367 Deleting Resources
- 368 Searching the Library
 - 371 Performing the Search
- 372 Extracting Resources
 - 375 Performing the Extraction
- 376 Expiring Resources
- 377 Reviewing a History of Resource Changes
- 379 Running Response Files
- 380 Defining Mode, Status, Class, and Project Options
- 382 Using the LBYPROC Utility**
- 384 Troubleshooting**
 - 384 Turning on Tracing
 - 385 Handling Error Messages

Chapter 10: Using Dictionaries

- 388 Working with the Extract Dictionary**
- 389 Opening an Extract Dictionary**
 - 390 Sorting XDD Elements
 - 391 Importing XML Schemas into the XDD
 - 394 Importing XML Extract Files into the XDD
 - 394 Importing XML Attributes
 - 397 Using Addressee Records
 - 397 Adding an Addressee Record
 - 398 Adding the Address Map in the BDF
- 400 Using the Common Fields Dictionary**
 - 401 Working with the Common Fields Dictionary
 - 401 Importing CSV Files
 - 402 Importing TGA Files

Chapter 11: Creating Tables

404 Overview

- 405 Transaction File DFD
- 406 Recipient Batch File DFD
- 406 Application Index File DFD
- 407 Field Section
- 408 Key Section

Chapter 12: Converting Files

410 Overview

412 Converting Word and RTF Files

- 419 Understanding Font Mapping
- 421 Capturing Information from Imported Word Files

422 Converting Files into Sections

- 423 Converting IStream Migration RTF Files
- 425 Converting PDF Files

427 Compiling Sections into Print Files

429 Converting Sections into Paragraphs

430 Converting Graphic Files

431 Changing Templates

433 Making Changes to Multiple Sections

- 434 Section Conversions
- 435 Field Conversions
- 436 Font Conversions
- 437 General Conversions

439 Converting Bitmap Files into LOG Files

440 Converting System Files into XML Files

442 Converting Documerge Resources

445 Changing Recipient Information

447 Importing XML Schemas into the XDD

449 Finishing a Conversion**Chapter 13: Producing Reports****452 Overview****455 Printing Reports**

457 Selecting Files

461 Printing a Finishing Report

463 Using the Finishing Report

465 Printing the Documerge Field Report

Chapter 14: Handling Fonts**468 General Font Concepts**

468 Font Terminology

471 How Characters are Represented

471 Bitmap Fonts

471 Scalable Fonts

472 How Computers and Printers Use Fonts

474 Using Code Pages

475 ASCII Code Pages

477 EBCDIC Code Pages

478 Character Sets

478 Code Page Names

480 Types of Fonts

480 Using Screen Fonts

480 Font Substitution in Windows

481 Installing Screen Fonts in Windows

481 Using Printer Fonts

481 AFP

482 Metacode

482 PCL

482 PostScript Fonts

482 TrueType Fonts

- 482 Adding Printer Fonts to a Font Cross-reference File
- 483 Using System Fonts**
 - 484 Font Cross-reference Files for Monotype Fonts
 - 487 Using Custom Fonts
- 489 Using Font Cross-Reference Files**
 - 490 How FXR Settings Affect Display and Print Quality
 - 491 Maintaining FXR Files
 - 491 Choosing a Font Cross-reference File
- 493 International Language Support**
 - 493 Using the ANSI Code Page for PC Platforms
 - 494 Using Code Page 37 for EBCDIC Platforms
- 495 Using International Characters**
 - 496 Converting Text Files from one Code Page to Another
- 497 Setting Up PostScript Fonts**
 - 500 Fonts for PDF Files
 - 500 Importing PostScript Symbol Fonts
- 502 Font Naming Conventions**
- 503 Mapping Fonts for File Conversions**
- 504 Using Font Manager**
 - 504 Defining Basic Font Properties
 - 506 Defining AFP Font Properties
 - 506 Defining Metacode Font Properties
 - 508 Defining PCL Font Properties
 - 509 Defining PostScript Font Properties
 - 510 Defining PDF Font Properties
 - 512 Defining Font Dimensions
 - 513 Importing Fonts

Chapter 15: Testing Your Work

- 518 Overview**

520 Using the Screen

- 520 Using the Menu Bar
 - 521 Using the View Menu
 - 521 Using the Test Menu
- 523 Using the Right-Click Menus
- 525 Using the Toolbar

527 Creating a Test Profile**530 Running a Test**

- 533 Running a Trigger-Run Only Test
- 533 Manually Triggering

534 Changing Test Properties**535 Modifying the AFGJOB File****536 Deleting a Test****537 Using the Printstream Analyzer**

- 539 Using the Record View and Display Panes
- 540 Using the Find Option
- 541 Finding a Specific Page or Record
- 542 Displaying Information about the Record
- 543 Printing What is in the Panes

Chapter 16: Deploying a Library**546 Overview****547 Creating or Running a Deployment**

- 549 Creating a Library Deployment
- 551 Creating a Flat-File Deployment

552 Additional Resources**553 INI Settings****554 Processing the Deployment****Appendix A: Using the Documaker Add-in for Microsoft Word**

556 Introduction

557 Candidate Document Types

558 Understanding the Workflow

560 Finding Information

561 Synchronizing Fonts

562 Providing Graphic Files to the Add-In

564 Sharing Studio Resource Information with the Add-In

565 Generating a Workspace Definition File

566 Understanding the Workspace Definition File

568 Importing Documents Created in Word

575 Understanding the Import Process

578 Completing Imported Documents

581 Summary

: Glossary

Preface

Documaker Studio is a multi-user forms creation system that allows multiple users to work together to efficiently create form sets. Studio helps you manage the development process and maintain large electronic forms libraries.

Documaker Studio is designed for forms and business analysts who work to meet the requirements defined the compliance group and create the dynamic document applications which are then provided to production operations teams.

This document describes how to use Documaker Studio to create these documents.

AUDIENCE

This document is intended for the person who will use Documaker Studio to create dynamic documents.

DOCUMENTATION ACCESSIBILITY

Accessibility of Links to External Web Sites in Documentation

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

TTY Access to Oracle Support Services

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

RELATED DOCUMENTS

For more information, refer to the following Oracle resources:

- Documaker Server Installation Guide
- Documaker Add-in for Microsoft Word User Guide
- Rules Reference
- DAL Reference
- Docutoolbox Reference

CONVENTIONS

The following text conventions are used in this document:

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

Chapter 1

Using Documaker Studio

Documaker Studio is a forms creation system that allows multiple users to work together to efficiently create form sets. Studio helps you manage the development process and maintain large electronic forms libraries.

This chapter includes information on these topics:

- *Introduction on page 2*
- *Creating a Multi-user Development Environment on page 10*
- *Starting Studio on page 12*
- *Using System Menus on page 16*
- *Creating Workspaces on page 27*
- *Joining a Workspace on page 34*

INTRODUCTION

Welcome to Documaker Studio, the application used for building source files for a Documaker implementation. Documaker Studio is a multi-user forms development system that promotes workgroup and team-based development methodologies. Documaker Studio tackles the complex development process for building and maintaining large electronic forms libraries. Documaker Studio is designed for forms and business analysts who work to meet the requirements defined by the compliance group, and those who create the dynamic document applications provided to the production operations teams.

Studio lets multiple users work together in a library environment to construct graphics, documents, and rules for data and forms assembly. Studio lets you check in and check out the various resources you create so you don't have to worry about overwriting another user's work. Studio also lets you control who has access to the different tools within the product. Furthermore, Studio links user IDs to every resource you create and can track who made each modification. As a system administrator, you can even control who sets recipient copy counts, uses the scripting language, and runs testing scenarios. We will look at these features in greater detail throughout this User Guide.

Note Documaker creates and processes documents for a variety of businesses, including insurance companies, utility companies, and financial companies. Throughout this guide an insurance analogy is often used to provide examples. The documents may differ, but the process of creating and processing the documents is similar.

DOCUMAKER SYSTEM OVERVIEW

There are many different ways to implement the Documaker Server system. Documaker Server can operate on multiple platforms, with other Oracle Corporation products, and with third-party applications. The simplest solution entails a policy administration system supplying data on a company's customers to a Documaker system. The Documaker system incorporates these data elements into customized form sets, which will eventually produce a customer document set. These document sets include copies for other recipients, rendered in a variety of formats for different delivery methods (printers, online content, fax).

Documaker Server can also change platforms and configuration settings as it moves from a *test* phase to a *production* phase. The typical Documaker implementation cycle includes the following:

- Building system resources
- Configuring the processing system
- Testing resources in a runtime environment
- Deploying the tested system in a production (live) environment

Documaker Studio assists in all of these stages of construction, from building resources and the modification of configuration files, to the testing of form rules and recipient printing options. Documaker Studio also has a deployment feature for moving resources to another location for use in production.

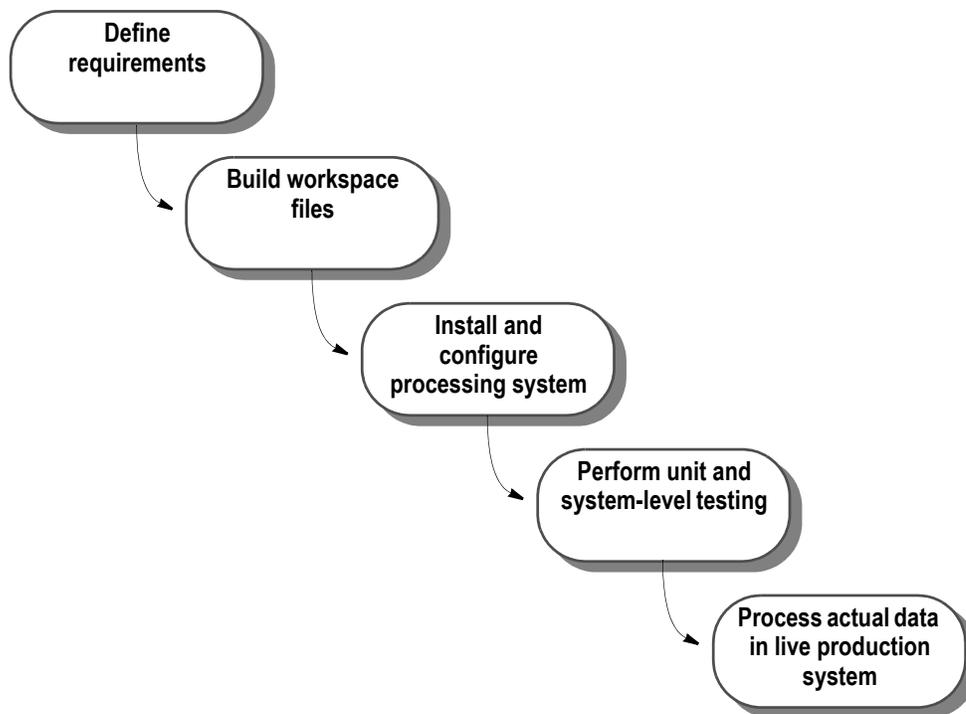


Figure 1: Documaker standard implementation process

Note Documaker Studio is used to build a *workspace*, with all the forms, recipients, tables, fonts, graphics and rules used to fulfill a business unit's requirements for publishing. See *Documaker Studio Workspace Concept* on page 7 for a discussion workspaces.

Documaker Server

As mentioned before, there are many possible configurations for a Documaker implementation. However, every implementation reads and processes certain input and output files. Other Documaker guides, including the Rules Reference, discuss configuration settings and job rules for producing specific output from a Documaker system. Your Insurance Global Business Unit professional services team can also help you customize your Documaker system.

Here is a high-level Documaker processing workflow. Keep in mind your system may use different file types and may combine some of the processing steps shown here:

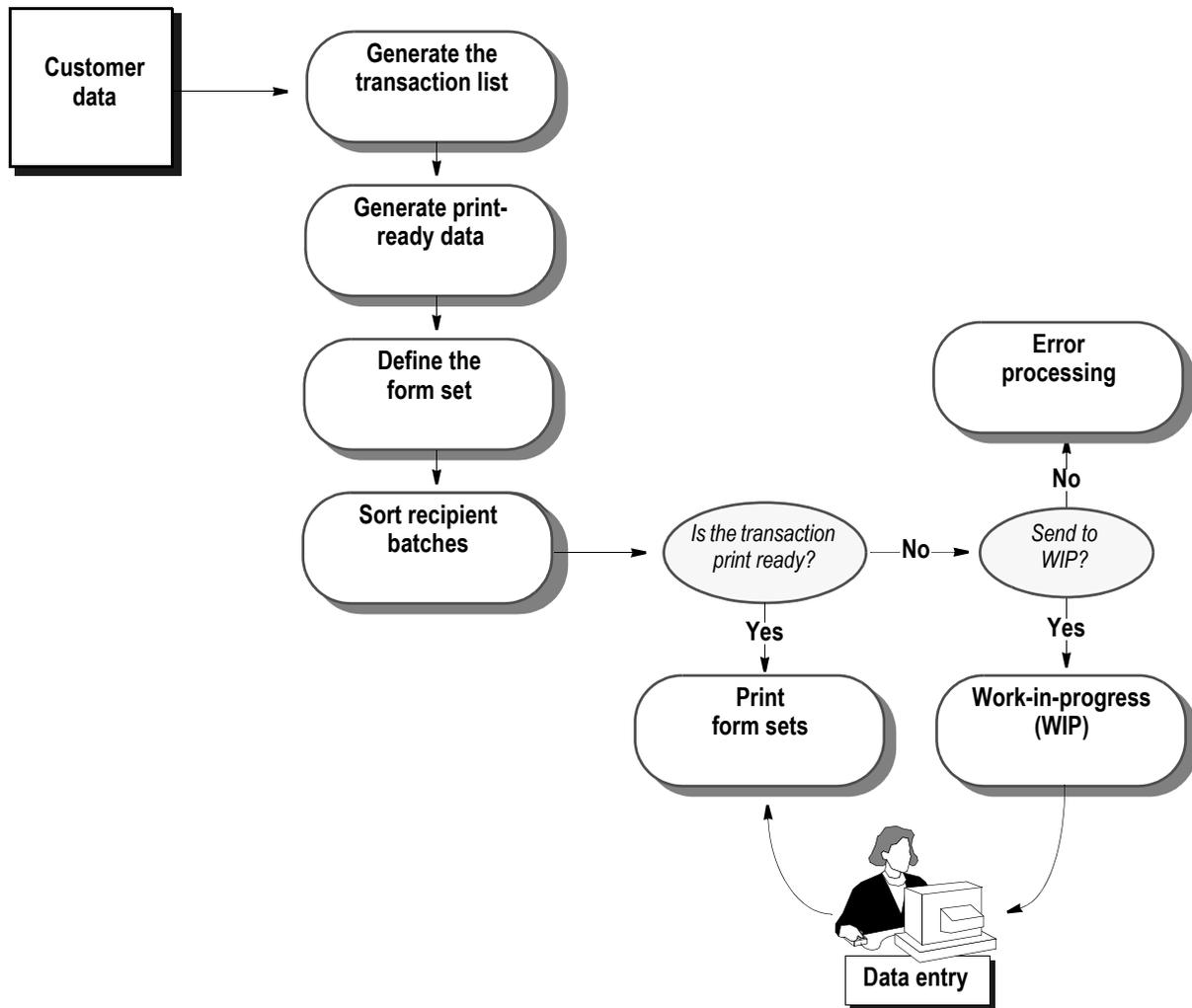


Figure 2: Documaker rules processing overview

In this figure, customer data is used by Documaker to generate a list of transactions (a transaction being any purpose for interacting with that customer's data: new business, policy updates, policy correspondence, and so on). Documaker Server then formats the data based on rules you specify. These rules include calculations and can specify alignment, decimal placement, and spacing. Based on rules and triggers you add via Documaker Studio, Documaker Server generates a set of forms for output which have a particular transaction type. Your Documaker configuration also determines which recipients receive a copy of the documents, and how many copies to produce. If the transaction is of a type that is print-ready, it will be produced as a print file, ready for a high-speed printer, or a PDF file, ready for online viewing.

Another possibility is that the transaction needs to be viewed or further manipulated in an interactive system prior to publishing. This is done through the Documaker Workstation. These transactions may simply be checked for accuracy or be completed manually (for missing data, or adding personal correspondence). Transactions in WIP can be printed directly from WIP queues, or sent back to the original print output stack.

Documaker Server Logical System flow

The rules processing part of Documaker Server consists of these main programs: GenTran, GenData, GenPrint, and GenWIP.

- *GenTran* reads an input data file generated by a policy administration system. Based on configuration settings in the Documaker system, GenTran will identify each transaction in the data file, and write it out to an output file: the TRNFILE.
- *GenData* reads the TRNFILE to interpret the data input file, produces a forms lists for each transaction, and writes output files: a batch file for each Recipient, a NAFILE containing transaction data, and a POLFILE with form and print information. Any WIP transactions are also written to a separate batch file.
- *GenPrint* reads the GenData output files, and produces the print file in the specified format (Metacode, AFP, PDF and so on), using available device fonts and printer settings to publish the documents.
- *GenWip* is an optional program that reads any WIP transactions flagged by GenData and allows manipulation of these files via Documaker Workstation programs.

Each program produces log, error, and message files. The Documaker Batch Processing classes offered through Oracle University discuss these output files in detail. In the figure below is a high-level process flow for the Documaker Server programs.

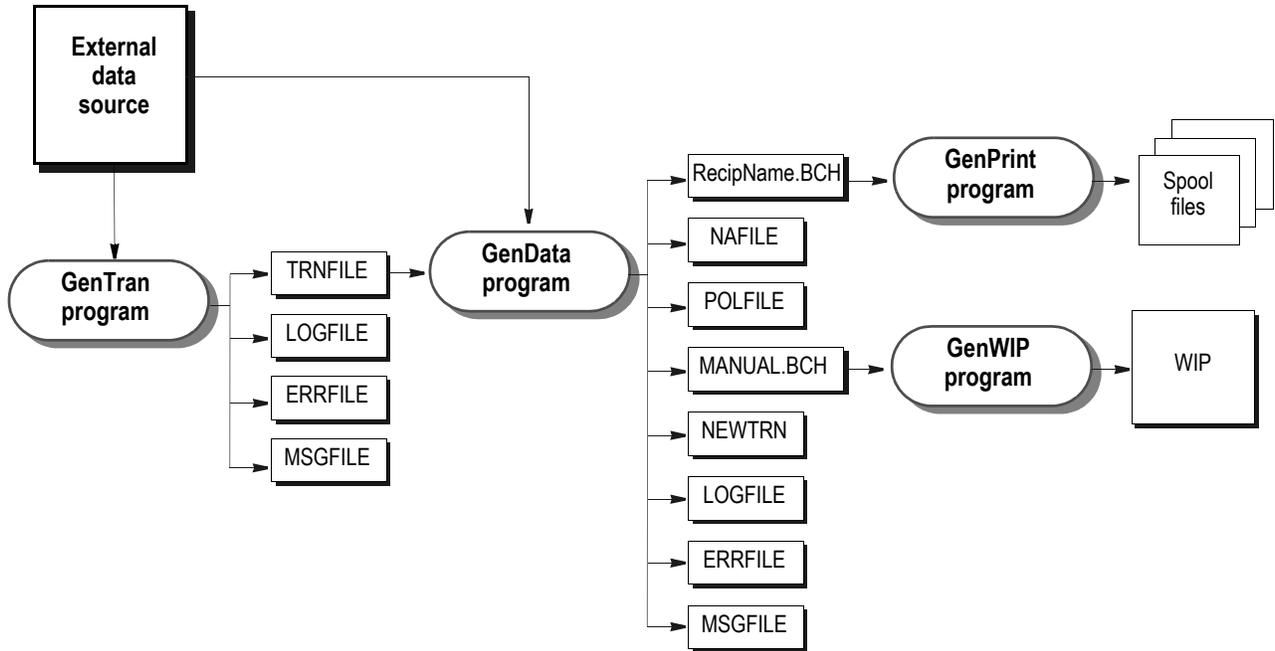


Figure 3: Documaker Server logical system flow

Documaker Studio Workspace Concept

For most Documaker implementations, there is great flexibility in how resources are built: some may already exist in other formats, which can be easily used by Documaker Studio, either as is or through conversion wizards (See the chapter *Converting Files*).

Some resources may be built entirely within the Studio. A particular group within your organization may work exclusively on documents and graphics, while other groups can format data and test form assembly rules.

Regardless of how work is assigned in your organization, all business groups can use Studio to design and implement a Documaker system.

Documaker Studio uses *workspaces* to build and maintain the resources used for a Documaker implementation. A workspace can be defined as the files created within the Studio, as well as files generated by the Documaker Server programs discussed earlier.

Workspaces are based on a multi-user module, where user IDs are linked to resources, so different users can create and update resources within a workspace at the same time without overwriting each other's work.

Before you begin working in Documaker Studio you must create or join a workspace. You can create a workspace from scratch or build one based on the resources already defined in a master resource library (MRL). You can import MRLs created by legacy Documaker tools, such as Docucreate, or from another Studio workspace.

As you create a workspace you must determine the following:

- If the workspace will be shared
- Where the workspace will be located
- What type of storage method you want for the workspace files

See *Creating Workspaces on page 27* for more information.

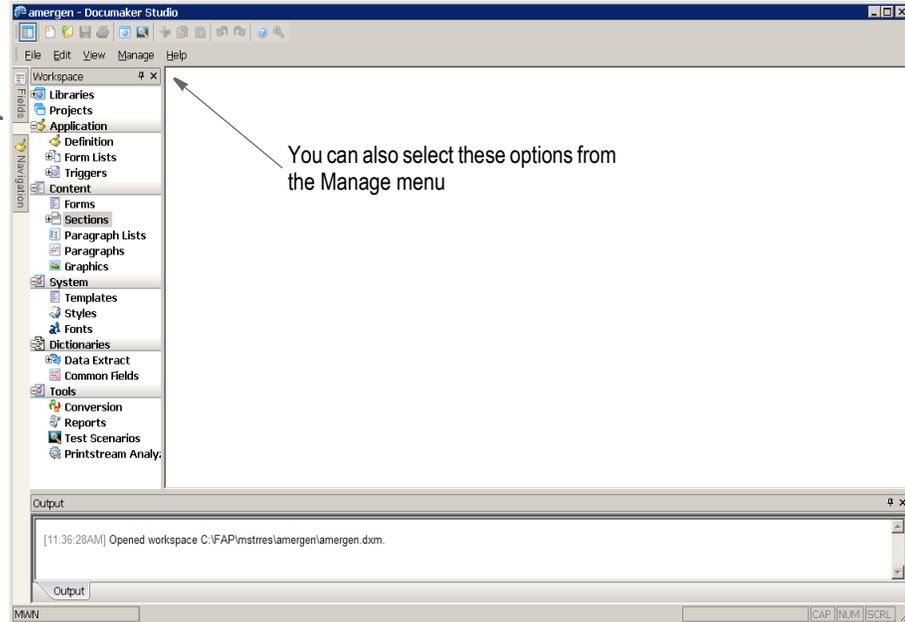
If the workspace was created on a shared drive for others to join, then anyone wishing to be added to the new workspace, with their own set of reference files, is required to go through a process similar to the creation step. See *Joining a Workspace on page 34* more information.

After the workspace is created, you can begin using the various tools within Documaker Studio. However, before proceeding, define security for your workstation environment. By doing so, you can become familiar with the Documaker Studio options and the various ways you can limit access rights to functions within the software. See the section on user security in this guide for details on establishing security.

STUDIO OPTIONS

Most options in Documaker Studio are used to create the resource objects that make up a *Master Resource Library (MRL)*. The options appear when you open a workspace:

The options you can use are listed here.



These resource objects are eventually used during the form entry process (Documaker Workstation) or in the forms processing cycle (rules processor). In some cases, the objects are used during both processes.

Option	Used to	Used during form		
		Entry	Processing	Required?
Libraries	Manage the libraries that contain the forms, sections, graphics, and other resources.	Yes	Yes	Yes
Projects	Associate a resources with a particular job or task.	No	No	No
Application				
Definition	Define a key combination comprised of a Key1 and Key2. Also defines recipients, categories, and transactions.	Yes	Yes	Yes
Form Lists	Define the list of forms available to an application definition.	Yes	Yes	Yes
Triggers	Create scripts that automate tasks within the processing environment. These scripts are created using Document Automation Language (DAL).	Yes	Yes	No
Content				

Option	Used to	Used during form		
		Entry	Processing	Required?
Forms	Create a list of the sections that comprise the form and to store triggering information.	Yes	Yes	Yes
Sections	Create and manage the sections that make up the forms.	Yes	Yes	Yes
Paragraph Lists	Build a list of paragraphs (PSL files) that are used for paragraph selection at run-time.	Yes	Yes	No
Paragraphs	Create and manage selectable paragraphs (PAR files) that can be assembled for a field at run-time.	Yes	Yes	No
Graphics	Resize, reverse, rotate, and manipulate bitmap graphics used on sections.	Yes	Yes	No
System				
Templates	Build form templates (TPL files) that can be used when creating forms.	Yes	Yes	No
Styles	Build style lists (STY files) that can be used when making sections. You define which style file to use in your application definition.	Yes	Yes	No
Fonts	Create and maintain the cross-reference table that serves as the bridge between the Documaker programs and the physical fonts.	Yes	Yes	Yes
Dictionaries				
Data Extract	Create and maintain mapping information for runtime data files.	No	Yes	No
Common Fields	Create and maintain common field information (FDB) to make setting up and creating variable fields on sections (FAP) faster and more consistent.	Yes	Yes	No
Tools				
Conversion	Convert files.	No	No	No
Reports	Create reports	No	No	No
Test Scenarios	Define and process test situations that simulate your production environment.	No	No	No
Printstream Analyzer	Analyze print files.	No	No	No

CREATING A MULTI-USER DEVELOPMENT ENVIRONMENT

Documaker Studio provides for multi-user development cycles. For several users to work on a given set of resources, the system must be able to manage the resources to prevent conflicts.

The key to effectively managing libraries of resources is in the files used to store information. These files are designed to facilitate multiple users throughout the life cycle of a project so you can move resources from development to testing and into production.

Note File names, types, and extensions, as well as menu names, options, and screen names can change during the development cycle.

Application definition files

The first component is a file where you maintain the *lines of business*. The extension and library type given to this file is BDF.

An application definition file defines a valid key combination comprised of a Key1 and a Key2. These keys are typically known as Company, Lines of Business, and usually State, in the Insurance industry.

When you need to add a new line of business or business unit to a set of resources, you first check out the BDF file and then make the necessary changes to define a new business unit.

This file is relatively simple in design because there are not many options at this level. But because it is now a file type you can manage, you can now introduce new business units starting at a given date, which was not possible before.

See *Working with Application Definition Files* on page 89 for more information.

Form list files

The next file type in the workspace tree is called the *form list* file. Once you define your business units, you then associate each business unit with a list of candidate forms. This file is where you define the list of forms available to a given business unit. And you can arrange the forms in the order in which you provide them.

The extension and library type for these files is *GRP*.

To add or remove a form from an application, you simply check out the appropriate form list (*GRP*) file and make those changes.

Since unit form order files are versioned and given effective dates, the list of candidate forms can vary based upon the date associated with a given transaction.

See *Working with Form Lists* on page 115 for more information.

Form files

The final file type created from the FORM.DAT is known as the *form* file. These files have the extension and library type of FOR, which stands for *form*.

As a new file type in the library, you can maintain versions and revisions of a form. Effective dates are used to make sure you get the correct rendition of the form for that date.

In addition to form files containing the list of sections that comprise the form, these files also contain the triggering information formerly stored in the SETRECIP.DAT file. This means that in addition to maintaining the section components, by checking this file out and back in, you automatically maintain the list of triggers used to generate specific layouts of the document.

The form file maintains the section options associated with the sections that comprise the form and maintains the specific location (SetOrigin) information used to place the sections correctly on the form.

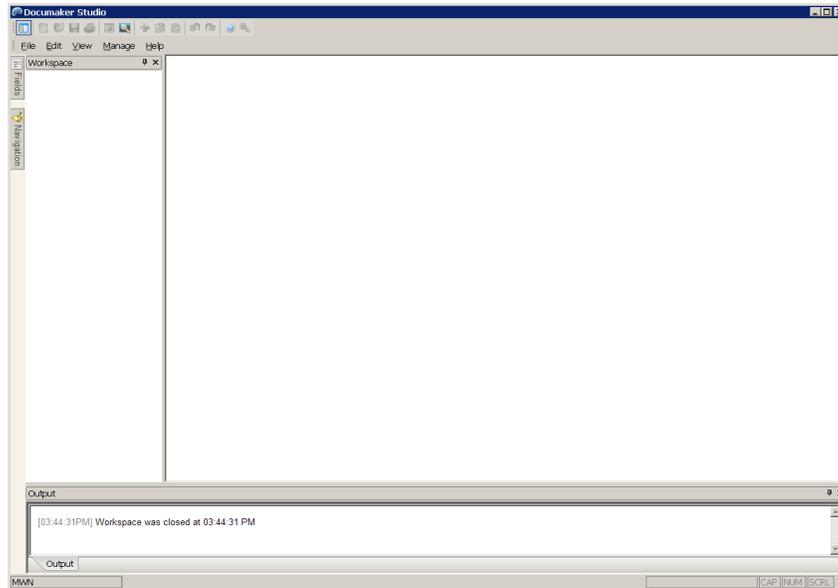
As mentioned, via the Libraries option, Studio supports these new file types (BDF, GRP, and FOR). This means you can check the files in and out and provide effective dates for when each are to become available (or expire).

The date associated with a transaction is used to select the appropriate version/revision of the files to build the document set and each transaction can therefore differ if they have differing dates.

See *Managing Forms on page 125* for more information.

STARTING STUDIO

When you first start Studio, the following window appears.

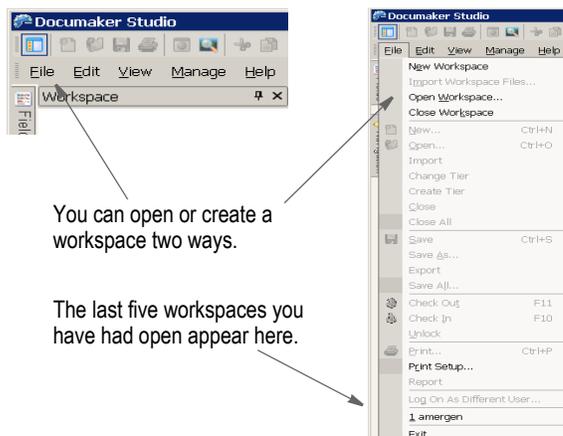


The first step is to open an workspace or create a new workspace.

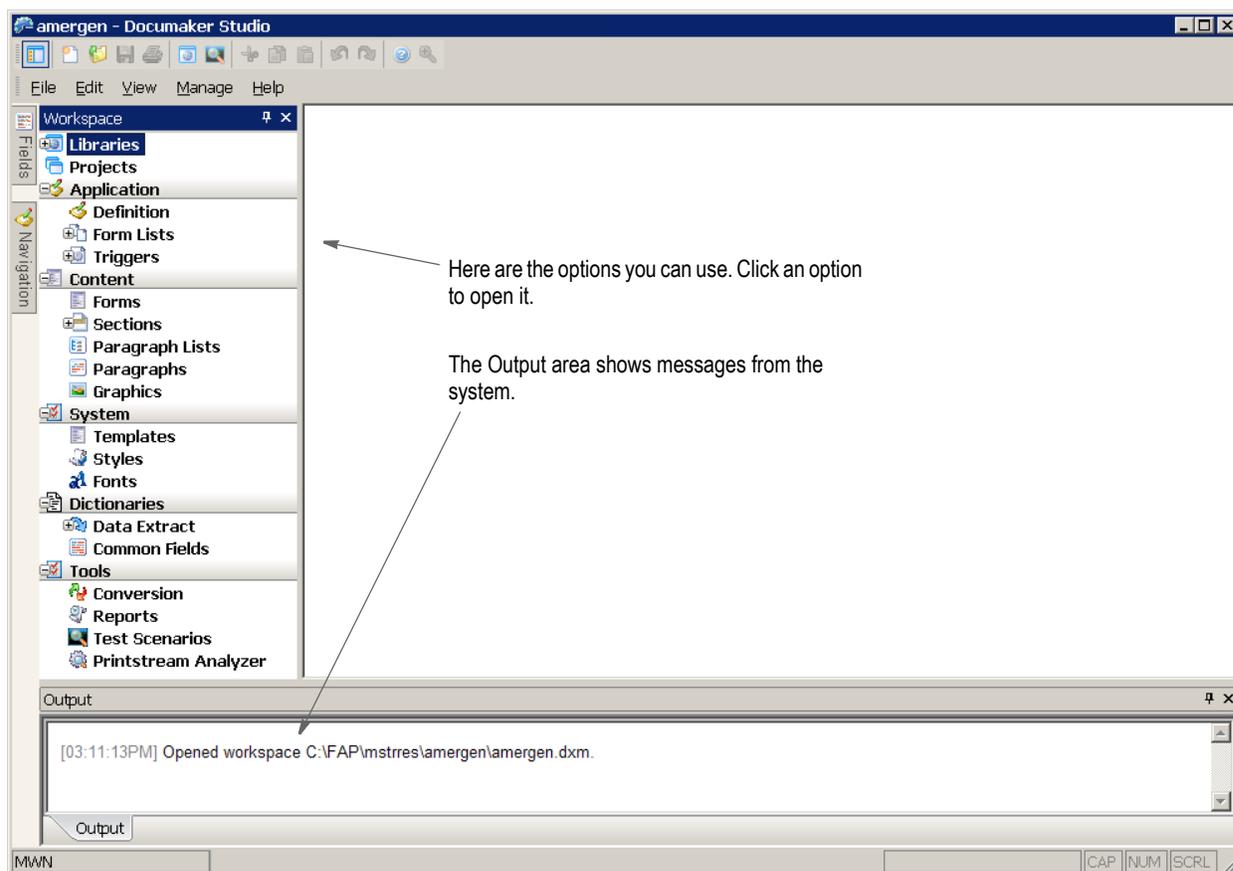
- For detailed instructions on creating a workspace, see *Creating Workspaces on page 27*.
- For more information on joining a workspace, see *Joining a Workspace on page 34*.

Note During installation, sample resource files are stored in the FMRes master resource library. Although you will likely set up your own resource libraries, do not delete these resources because the system may continue to use some of the files, such as the font cross-reference files (FXR)

You'll find the options to open or create workspaces on the File menu and on the toolbar.



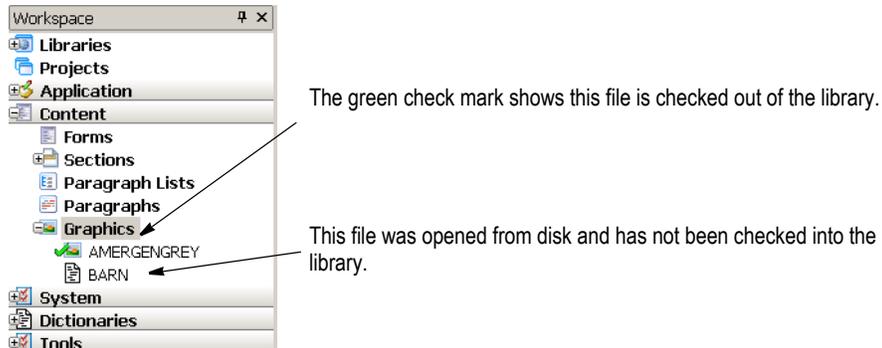
Once open, Studio shows the options you can use to work in that workspace:



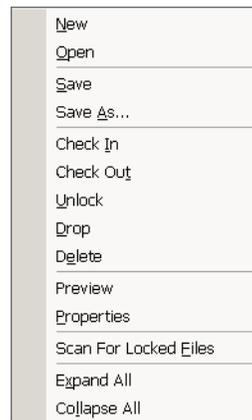
Having the tools displayed in a *tree* arrangement provides an easy way to see the files you checked out from the library. It also serves as a most recently opened file list for all files, including those not in the library.

You can remove a file from this list by highlighting the file then right clicking to display the context menu. Use the...

- Check In option to check the file into the library
- Drop option to remove it from the list.



If you right click in this area, the following menu appears:



Option	Description
New	Use to create a business unit, form lists, form, section, script, dictionary or table entry, or font.
Open	Use to open a business unit, form lists, form, section, graphic, script, library dictionary or table entry, conversion, report, font, or test.
Save	Use to save your work.
Save As	Used to save your work with another name.
Check In	Use to save the changes and check in the item you are working on.
Check Out	Use to check out an item from the library.
Unlock	Use to unlock an item that is open when you want to release it back to the library without saving changes.

Option	Description
Drop	Use to temporarily remove an item that appears in the list that is a library item. Permanently remove from the list a section that is not a library item. Unlock or check in permanently removes a library member from the working list.
Delete	Use to delete the highlighted item. For items already in the library, this option will be unavailable.
Preview	Use to preview the highlighted item.
Properties	Use to display properties of the item.
Scan for Locked Files	Display all locked files.
Expand All	Expand the tree to show all opened items.
Collapse All	Collapse the tree to show no opened items.

USING SYSTEM MENUS

This topic discusses the pull down menus available when you are working in Documaker Studio. The following topics discuss the options on these menus. When you first start Studio, you see these menus:

Select	To
File	Create, open, close, save, check in, check out, unlock, create tiers, change tiers, print files and reports, and log on as a different user. You can also use this menu to exit the system.
Edit	Perform normal editing functions such as undo, redo, cut, copy, paste, and delete.
View	Set to turn on or off the display of the toolbar, workspace bar, output bar, status bar, field bar, extract bar, navigation bar, and scripts bar. You can also use this menu to define how documents appear on your screen.
Manage	Work with various aspects of the project, including settings, libraries, dictionaries, scripts, fields, form sets, and other resources such as fonts, graphics, help, tables, and users. You can also use this menu to convert files, print reports, test forms, and work with user profiles.
Help	Display Help contents, view the Help index, or search Help topics. You can also use this menu to check for system updates, go to the support web site, and view product information.

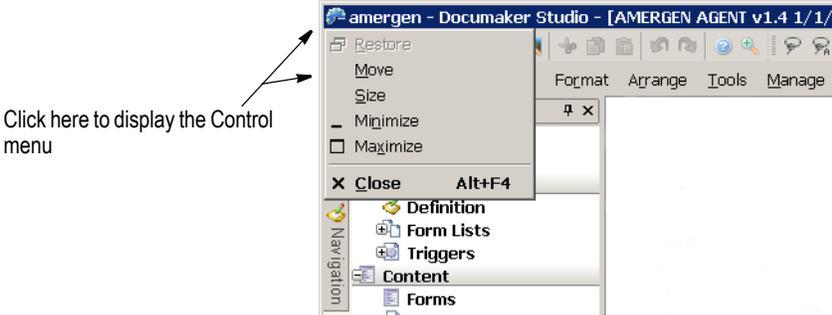
These menus can change, depending on which tool you are working with. For instance, if you are working with sections, you have these additional menus to choose from:

Select	To
Insert	Insert the various objects that comprise a section, such as boxes, text labels, text areas, charts, lines, graphics, fields, shade, and so on.
Format	Convert objects into text labels or areas, or specify the formatting for paragraphs, including setting tabs, defining columns, numbering, specifying shading and borders, and so on.
Arrange	Align, space, resize, and position objects.
Tools	Run a spell or grammar checker, generate readability statistics, print a section report, run a data entry check or a section report, and compile or normalize the section.
Window	Use to control the display of your windows. You can elect to have the windows display in tile, stack, or cascade format. You can also open a new window or arrange icons.

In addition, some of the options on the standard menus change. For instance, if you are working with sections you will find additional Edit menu options which let you bold or italicize text. The chapters that discuss working with sections or form, for example, document the addition menus and options that become available.

Using the Control menu

In addition in the top, left corner of most windows you can click on the Control menu icon:

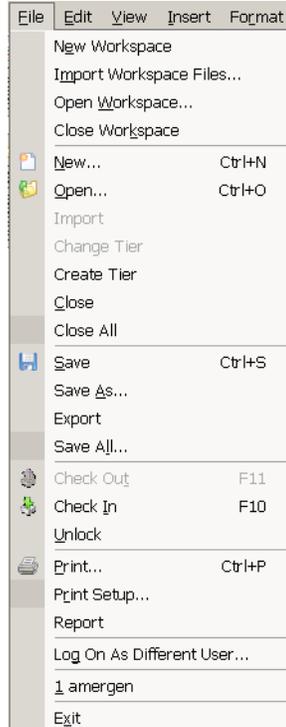


This menu provides these options:

Option	Description
Restore	Restores the window to a smaller view
Move	In the smaller view, the Forms icon appears in the upper left hand corner by the form name. The Move option lets you move the Forms window around.
Size	Lets you resize the window.
Minimize	Minimizes the window.
Maximize	Maximizes the window.
Close	Closes the window.

FILE MENU

The File menu controls creating, opening, closing, saving, checking in or out, and printing your files. You also exit the system and return to your operating system from the File menu. When you select File, this menu appears:



A brief summary of each option appears below.

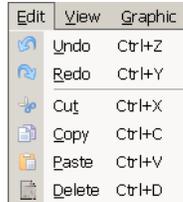
Select	To
New Workspace	Create a new workspace (DXM file).
Import Workspace Files	Import a workspace file.
Open Workspace	Open a workspace.
Close Workspace	Close the current workspace. The system prompts you to save your work.
New	Create a new object, such as a section.
Open	Open as object, such as a section.
Import	Import an object.
Close	Close the open object window. The system prompts you to save the object if changes have been made.
Close All	Close all object windows. The system prompts you to save each object if changes have been made.
Save	Save the object you are working with.

Select	To
Save As	Save the object under a new name. This option lets you make a copy, without changing the original.
Export	Export an object
Save All	Save all open windows.
Check Out	Retrieve an object from a library and lock it, so no other user can check it out while you work with it.
Check In	Return an object to a library and unlock it. Other users can then check out the object.
Unlock	Remove the lock placed on a document when you retrieved it with Check Out, but did not use the Check In option to return it to the library.
Print	Print the current section or page.
Print Setup	Select the printer you will print to and set up printer-specific options.
Report	Generate reports.
Log On as a Different User	Log onto the system under another user ID.
<i>(recently opened files)</i>	Choose from a list of the last five files you had open. This lets you quickly reopen a workspace, for instance.
Exit	Close and exit Studio.

Note The Change Tiers and Create Tiers options are unavailable unless you have the Beta Project Workspace option checked (View, Options, Workspace Information). The Tier options are part of the project management tools currently under development and are only available to Beta testers.

EDIT MENU

The Edit menu lets you make changes to objects. You can undo, copy, cut, delete, and paste individual objects or select all the objects for editing. When you select Edit, this menu appears:



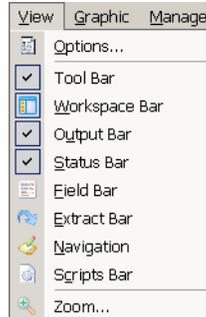
The options on the Edit menu can vary, depending on which tool you are using. This example shows you the standard Edit menu options. Here is a brief summary of these options:

Select...	To...
Undo	Cancel or reverse your last action or choice.
Redo	Repeat the most recent change.
Cut	Remove a selection and place it on the clipboard. Use Paste to insert the selection into, for instance, another part of the section or into a different section.
Copy	Make a copy of the selection and place it in the clipboard. Use Paste to insert the selection into, for instance, another part of the section or into a different section.
Paste	Insert the contents of the clipboard at the cursor location.
Delete	Erase the selection. The selection is not stored on the clipboard so you must immediately select Undo if you change your mind.

VIEW MENU

The View menu controls lets you quickly turn on or off the display of the toolbar, workspace, output, or the status bar. When you select View, this menu appears:

The check marks indicate bars which have already been selected.

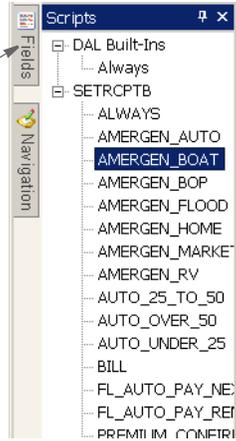


A brief summary of each option appears below:

Select...	To...
Options	Specify grid and ruler settings.
Toolbar	View the toolbar.
Workspace Bar	View the workspace tree.
Output Bar	View the Output bar. Here Studio displays information on the tasks it is performing. This could include error, warning, or confirmation messages.
Status Bar	View the Status bar. Here Studio provides information about the status of the system
Field Bar	View the Field bar. You can use this bar to quickly locate a field in the Common Fields dictionary.
Extract Bar	View the Extract bar. You can use this bar to quickly locate a field in the Data Extract dictionary.
Navigation	View the Navigation bar. The Navigation bar shows you all open objects and provides an quick way to go from one object to another.
Scripts Bar	View the Scripts bar. The Scripts bar shows you all sections which have scripts and provides a quick way to view or edit a script in a section.
Zoom	Zoom in or out.

Note Depending on the tool you are using, other view windows can appear on the menu. The ones shown here are for the main view windows and are always available.

Here is an example of the Scripts bar.



This icon lets you automatically hide a bar or “pin” it to the window. As shown here, the Scripts bar is pinned to the window.

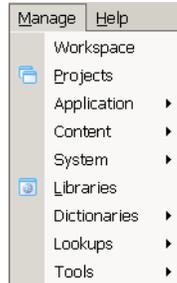


Click the auto hide icon again and Studio hides the bar and adds a tab on which you can click to maximize the bar when you want to work with it again.

Studio rotates the icon to show you that it will hide the bar when you finish working with it. In this example, both the Fields and Navigation bars have been “auto hid”.

MANAGE MENU

The Manage menu lets you work with the various items which make up a project, such as libraries and scripts. When you select Manage, this menu appears:



A brief summary of each option appears below.

Select...	To...
Settings	Define the INI and other configuration settings used by the system.
Business Units	Define the various business units.
Group Forms	Define the characteristics of a specific group of forms. For instance all of the forms which comprise a specific type of loan or insurance policy would be listed.
Forms	Define the selection and arrangement of sections that comprise the form set.
sections	Create and manage the sections you use to make up forms.
Graphics	Work with the bitmap graphics you use in your sections. This gives you an easy way to resize, reverse, rotate, and manipulate the graphic to fit your needs.
Scripts	Create scripts to automate certain tasks. You create these scripts using the Document Automation Language (DAL).
Libraries	Manage the libraries that contain the forms, sections, graphics, and other resources which comprise the form set.
Dictionaries	Create and maintain mapping information for runtime data files.
Extract	Create and maintain the XDB.DBF file.
Rule	Create and maintain the MASTER.DDT file.
Fields	Create and maintain the FDB.DBF file.
Tables	
Batch	Create and maintain tables (TBL files) used in a batch processing.
Entry	Create and maintain tables used in an entry processing.
Definitions	Define tables used by both the batch and entry processing.
Conversion	Convert files from one type to another, such as RTF to FAP and to change characteristics, such as a font ID) on multiple FAP files.
Reports	Print base definition, group, form, or section reports.

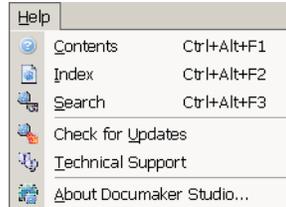
Select...	To...
Fonts	Organize your fonts into sets of fonts you use for section creation and printing.
Tests	Test your forms in a simulated production environment.
Users	Create or work with user profiles.
Deploy	Put resources into production.

Select...	To
Workspace	Manage your workspace.
Projects	Associate a resources with a particular job or task.
Application	
Definition	Define a key combination comprised of a Key1 and Key2. Also defines recipients, categories, and transactions.
Form Lists	Define the list of forms available to an application definition.
Triggers	Create scripts that automate tasks within the processing environment. These scripts are created using Document Automation Language (DAL).
Content	
Forms	Create a list of the sections that comprise the form and to store triggering information.
Sections	Create and manage the sections that make up the forms.
Paragraph Lists	Build a list of paragraphs (PSL files) that are used for paragraph selection at run-time.
Paragraphs	Create and manage selectable paragraphs (PAR files) that can be assembled for a field at run-time.
Graphics	Resize, reverse, rotate, and manipulate bitmap graphics used on sections.
System	
Templates	Build form templates (TPL files) that can be used when creating forms.
Styles	Build style lists (STY files) that can be used when making sections. You define which style file to use in your application definition.
Fonts	Create and maintain the cross-reference table that serves as the bridge between the Documaker programs and the physical fonts.
Dictionaries	
Data Extract	Create and maintain mapping information for runtime data files.
Common Fields	Create and maintain common field information (FDB) to make setting up and creating variable fields on sections (FAP) faster and more consistent.
Tools	

Select...	To
Conversion	Convert files.
Reports	Create reports
Test Scenarios	Define and process test situations that simulate your production environment.
Printstream Analyzer	Analyze print files.

HELP MENU

The Help menu controls access to Help screens. When you select Help, this menu appears:



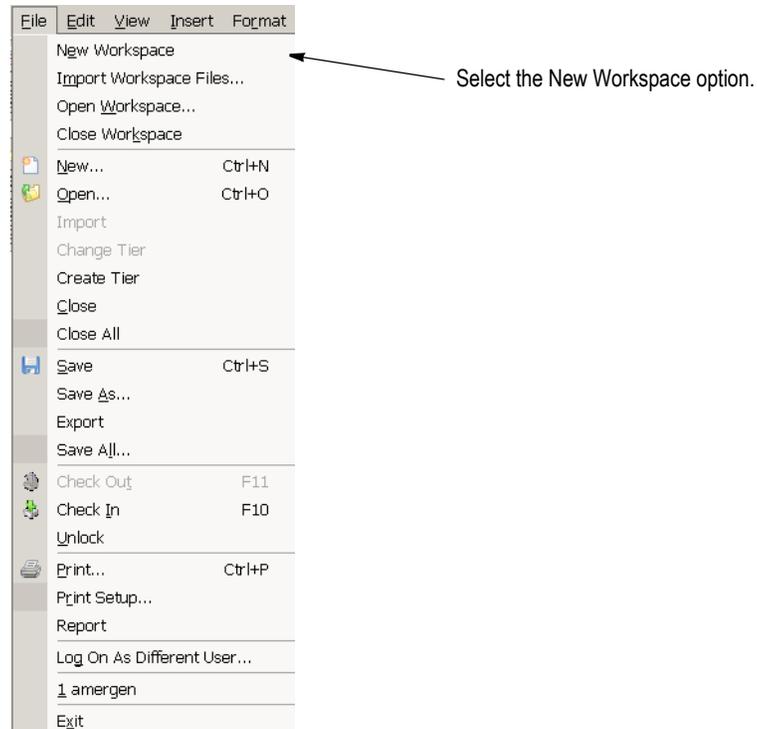
A brief summary of each option appears below.

Select...	To...
Contents	See a general table of contents to the Help system.
Index	View an index of Help topics.
Search	Search the Help system for a word or phrase.
Check for Updates	Contact Oracle's support site to check for updates.
Technical Support	Go to Oracle's support site.
About Documaker Studio	Review product information such as the version and patch number. You may need to refer to this information if you contact Support.

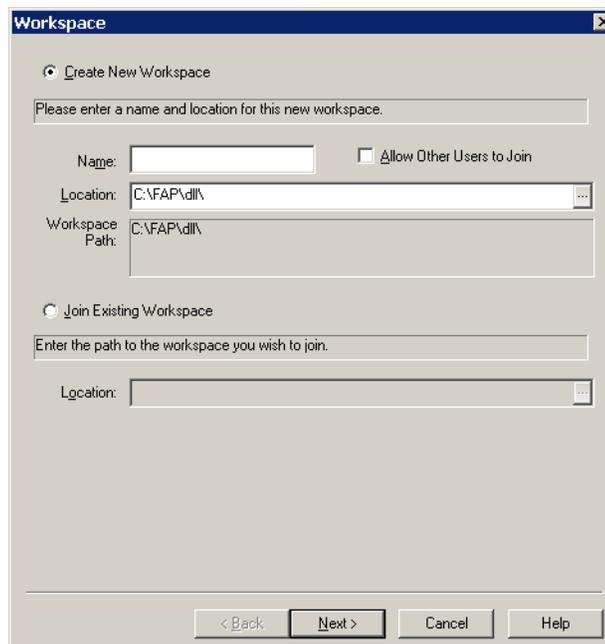
CREATING WORKSPACES

Follow these steps to create a workspace for multiple users:

1. Choose the File, New Workspace option.



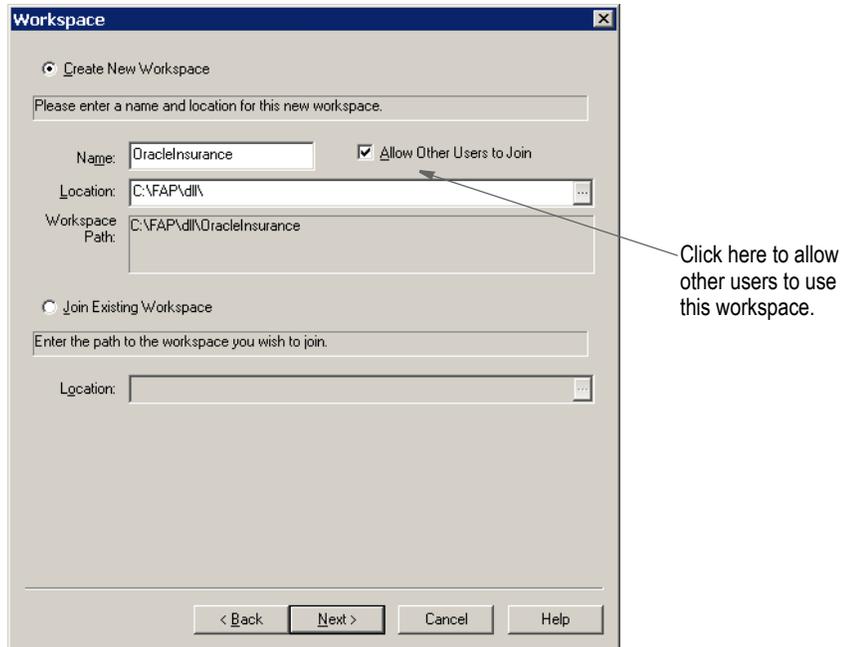
This turns on the Workspace wizard. The first page of the wizard asks whether you are creating a new workspace or joining an existing workspace.



2. Choose the Create New Workspace option. Then enter the name and location where you want the workspace to be created. If the path you specify does not exist, the subdirectories you specify will be created (if possible). For others to join the workspace, it will have to be located on a network or in a shared directory that allows access by the other users.

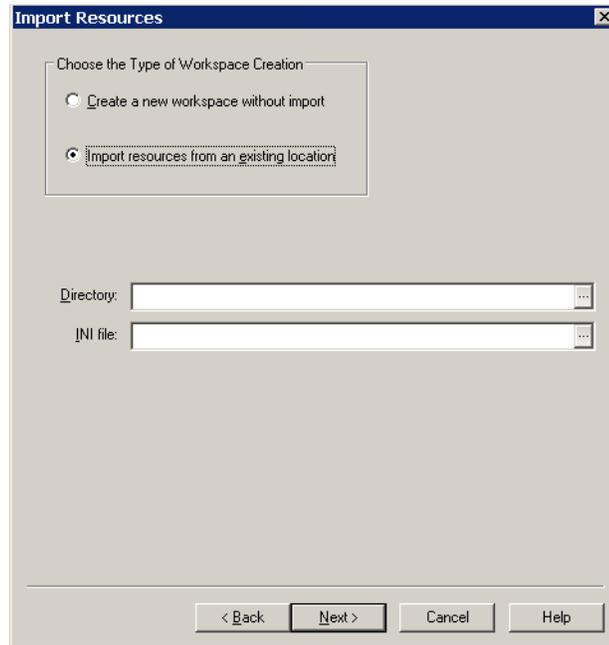
Note The name you enter for the workspace cannot begin with a space or any of these characters: \ / : * ? " < > |

3. Click the Allow Other Users to Join field. Then click Next.



The Import Resources page appears.

4. On the Import Resources page, you indicate whether you want to import resources from another master resource library (MRL) into your new workspace.
 - If you choose to create a new workspace without importing resources, click the Create a New Workspace Without Import option and then click Next. Keep in mind that you can later manually import resource files.
 - If you choose to import resources, click the Import Resources from an Existing Location option and then indicate the directory that contains those resources and also the INI file (usually named FSIUSER.INI) that identifies the components of the resources you want to import.



Note Remember, in some setups, the working directory and the location of the INI file for a resource library setup are different. If the INI file uses relative paths, the working directory must be set correctly, or the components of the library may not be found.

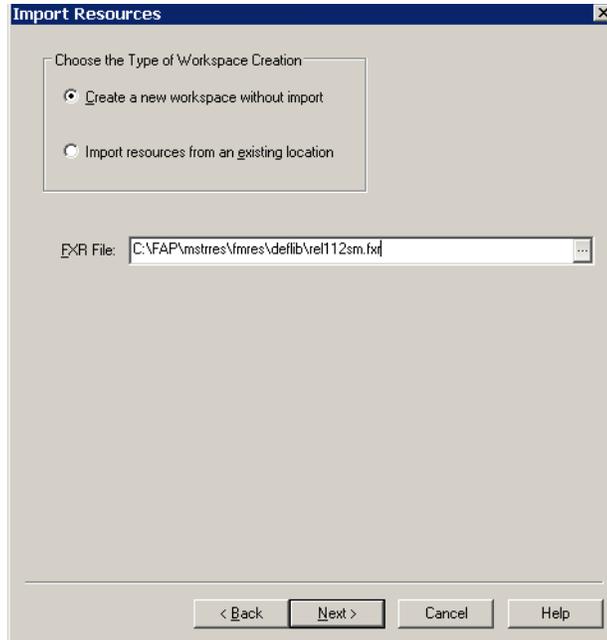
Relative paths are those that have dots to represent the current or prior directory locations, as shown here:

```
.. \DEFLIB\  
. \DEFLIB\  

```

Two dots means to back up a directory from the current working directory. One dot means that the next value is relative to the working directory. It is important to set the working directory correctly if you use these types of relative path settings in the INI file.

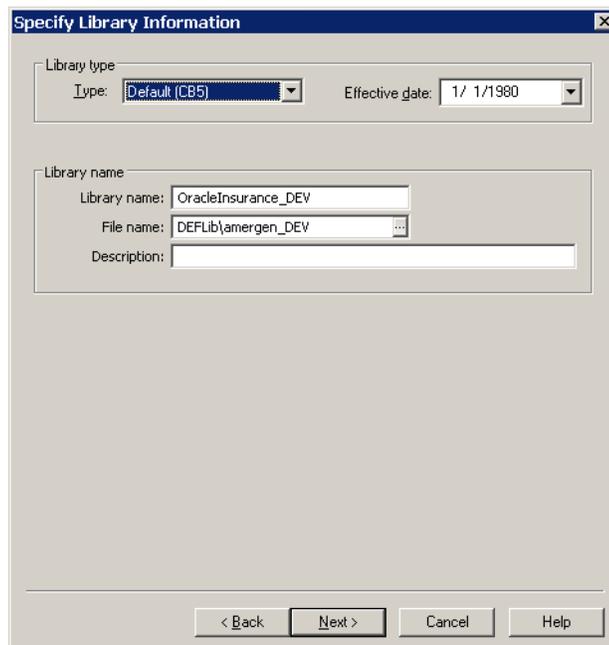
If you choose to import from a resource library, the wizard asks additional questions. If you select the Create a New Workspace Without Import option, the following field appears:



5. Use the FXR File field to enter the font cross-reference (FXR) file you want to use with this new workspace setup.

If you have installed the product according to the default procedures, Studio scans for and displays a font cross-reference (FXR) file. If the file shown is not the one you want, use the Browse button to locate that FXR. Once located, click Next.

The Specify Library Information page appears:

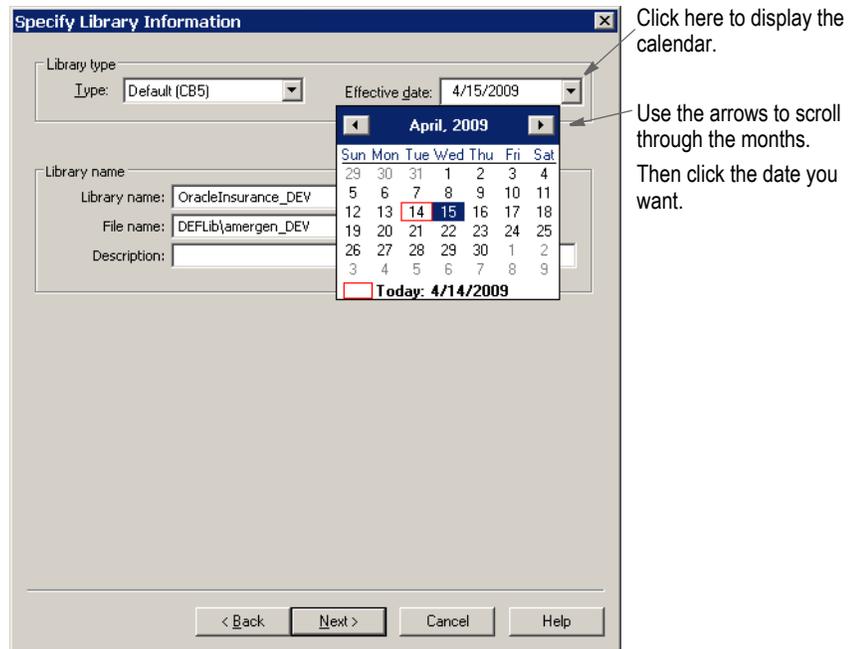


6. Here you define the library management method you will use. The default method (CB5) uses a separate index and compressed library file. This is the method used by many legacy environments. You can also use...

- An ODBC table as the index and location to store the compressed resource data
- A DB/2 SQL table as the index and location to store the compressed resource data
- Documange, Oracle Insurance’s powerful document management repository

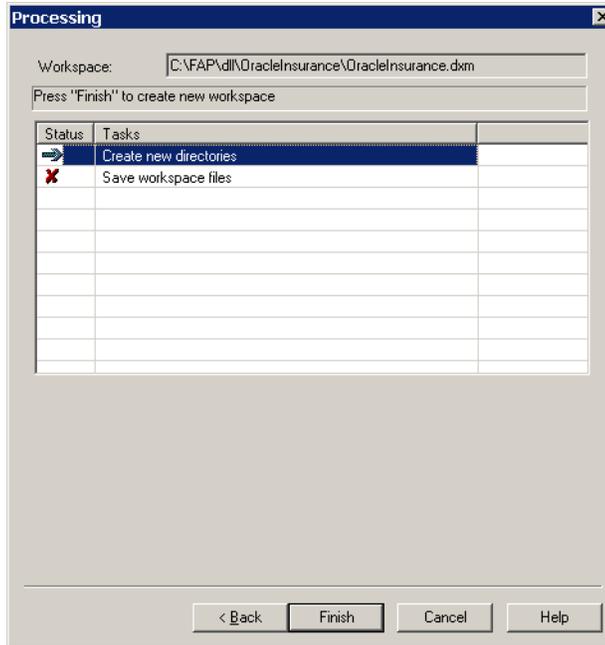
Note Depending on your library management choice, Studio may ask additional questions.

7. Use the Effective Date field to specify the default date for resources that can be imported or created for this new workspace. Remember that transactions are assumed to have a date-of-record — sometimes referred to as the *run date*, *effective date*, or *policy date*. Library management uses this transaction date to locate the proper resource version/revision that was effective on that date.



Note When creating a new workspace, it is important to set the date on this page back far enough to cover the transaction date range that you expect to process.

Click Next when finished. The Processing page appears:

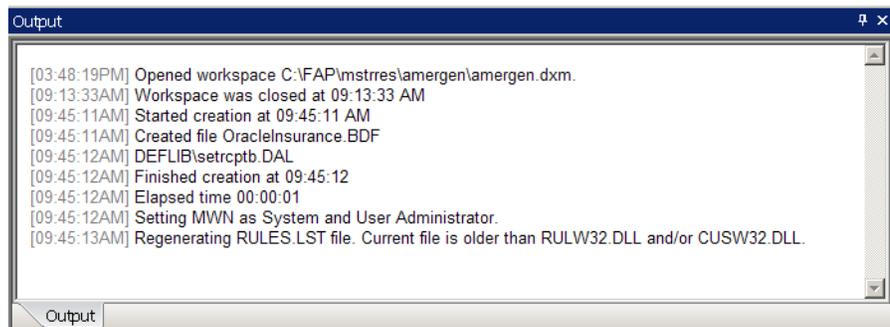


This page shows what steps the creation process has registered to activate. In this example, there are only two entries:

- Creating the new directory structure for the workspace
- Saving the workspace file

Note If you had chosen to import resources from an existing setup, this page would list additional tasks to accomplish the creation of the new workspace.

8. Click Next. The tasks listed on the window are checked off as they occur. You may notice some additional messages appear in the Output window. Here is an example of the output messages that can appear:



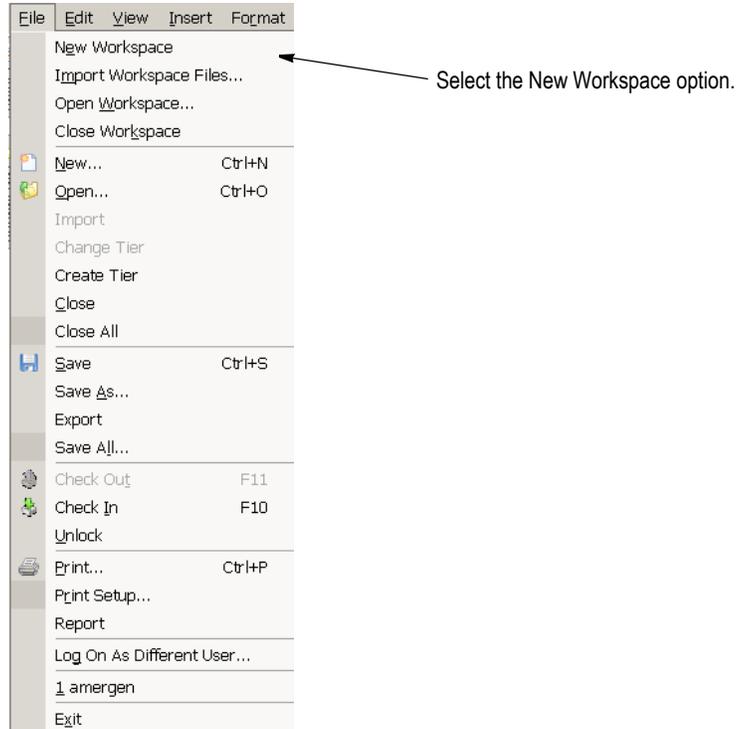
When finished, Studio opens your new workspace.

Once created, you are designated as the administrator of the user database (USERINFO). As administrator, you can set security rights for other users who join the workspace. You do not, however, have to predefine additional users. New users who join a workspace are automatically inserted into the user database and inherit the attributes of USER1. Therefore, as administrator, you should change USER1 to have the level and security rights you want new users to have.

JOINING A WORKSPACE

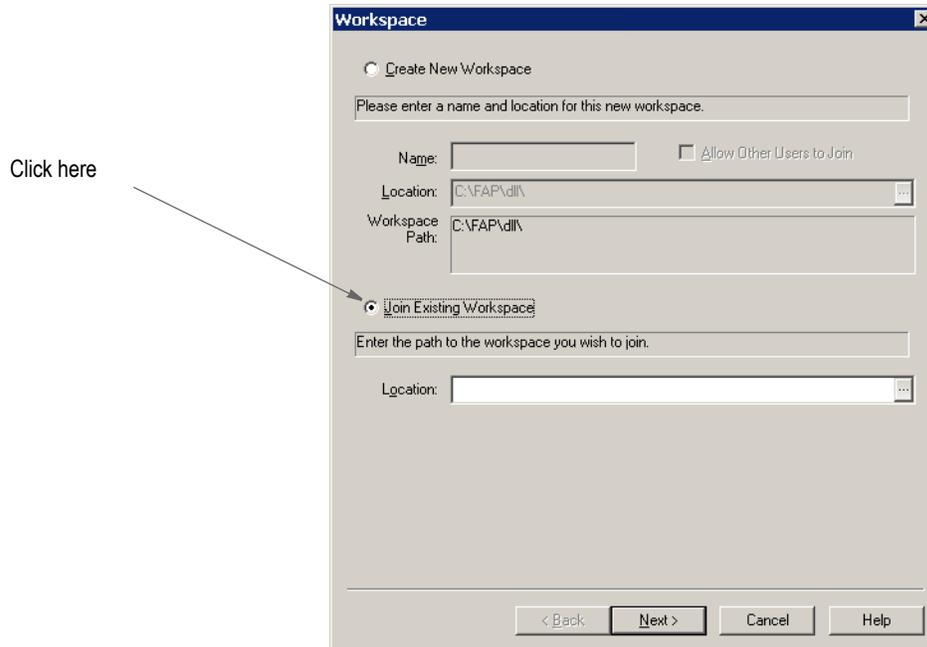
To join a workspace created for multi-user development, follow these steps:

1. Choose the File, New Workspace option.



This turns on the workspace creation wizard which leads you through the steps necessary to join an existing workspace.

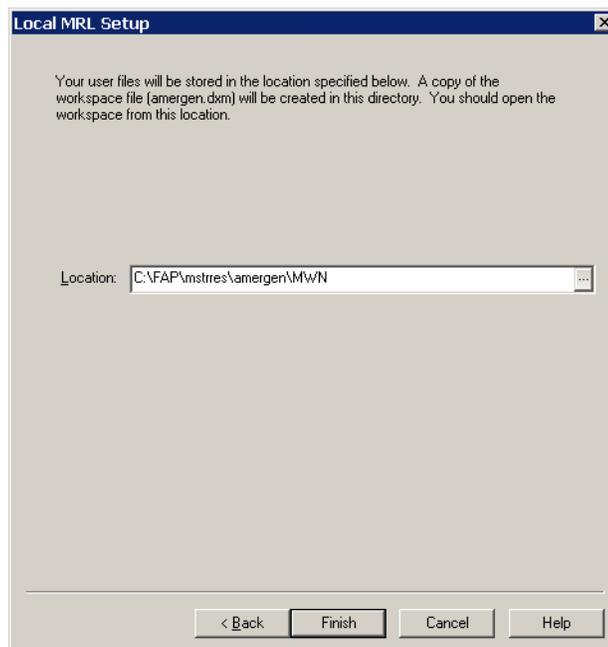
2. The first page of the wizard asks whether you want to create a new workspace or to join an existing workspace. Think of joining an existing workspace as you do creating a shortcut on your Windows desktop. You are creating a local reference to a shared workspace. Select the Join Existing Workspace option.



- Next, specify the location of the shared workspace you want to join in the Location field. You can use the browse button to pick the location, or simply enter the path and file name.

Note When selecting the workspace to join, specify the DXM file and not the DXS file.

Click Next. This page appears:



- On this page, you specify where you want a local scratch pad (where your checked-out resources are maintained) located.

Click Finish. Studio creates your local directory, if necessary, and opens the workspace.

Note A new user in a workspace, may not have sufficient security rights to do certain tasks. It is up to the administrators to assign security rights to users, including what rights new users inherit.

Troubleshooting

If you accidentally choose to create a new, shared workspace on the first wizard window and then enter a location that already has a workspace in it, you will see a message similar to this one.



You can open the workspace if you are already a member, or you can join the workspace and be prompted for the location to store your files. You can also choose to re-create the workspace. This last option only applies if you are the administrator of the workspace. If you are not the administrator, the system makes you start over.

Chapter 2

Working with Settings, Users, and Security

Use the Settings option on the Manage, System menu to work with INI options and tool settings. The Settings window provides an easy way to set the INI options which define default values and other user-defined parameters.

Tool settings let you define how your forms appear on your screen. For instance, you can choose the display units and colors for the rulers that border forms, sections and graphics. You can also define the grid settings and specify colors for different elements in DAL scripts.

This chapter discusses...

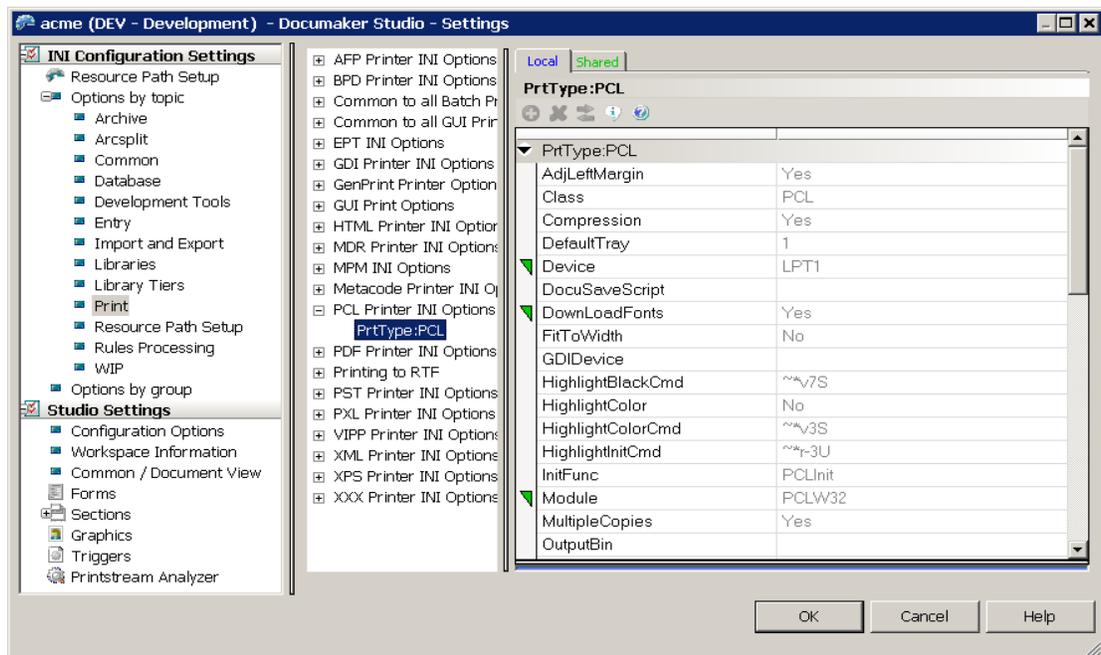
- *Working With INI Options on page 38*
- *Defining Studio Settings on page 46*
- *Managing Users on page 66*
- *Storing User Information in Another Database on page 77*
- *Adding Security at the Resource Level on page 78*
- *Using the LDAP Security Model on page 80*

WORKING WITH INI OPTIONS

INI options tell the system how you want it to operate. These options are stored in INI files. An INI file is simply a text file consisting of control groups and options. A control group organizes the various options and is denoted by brackets (< >) or braces ([]). The individual INI options appear below each control group. The settings for each option appear after an equals sign (=). Here is an example:

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

Instead of editing the text file, Studio lets you modify your INI settings without leaving the system. You simply choose the Manage, System, Settings option and use the Settings window to make changes. Your changes are applied as soon as you click Ok. Here is an example of the Settings window:



How INI files are used

Each resource library uses two INI files: FSIUSER.INI and FSISYS.INI.

- FSIUSER.INI - controls settings which can vary between resource libraries, such as sorting options, archival mode, and import/export ability, as well as individual user options. To work with these options in Studio, you click the Local tab.



- FSISYS.INI - controls information related to the entire system, such as system settings and program function calls. To work with these options in Studio, you click the System tab.

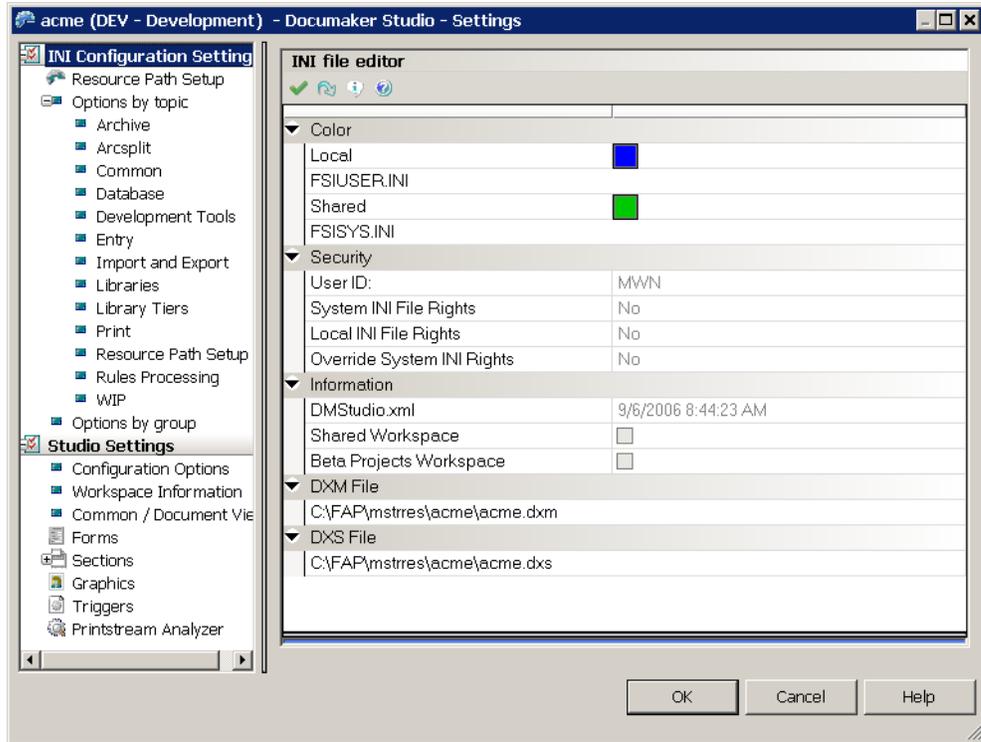


Studio loads the FSIUSER first, then finds the name and location of the FSISYS file from the ENVIRONMENT group, and loads it. Since the FSIUSER is loaded first, the options within this group usually override the FSISYS options.

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 allow a user to change local settings (FSIUSER), but not change the system settings (FSISYS). Or, you could let a user review these settings (local and system), but not change them.

SETTING INI OPTIONS

When you choose the Manage, System, Settings option, the Settings window appears:

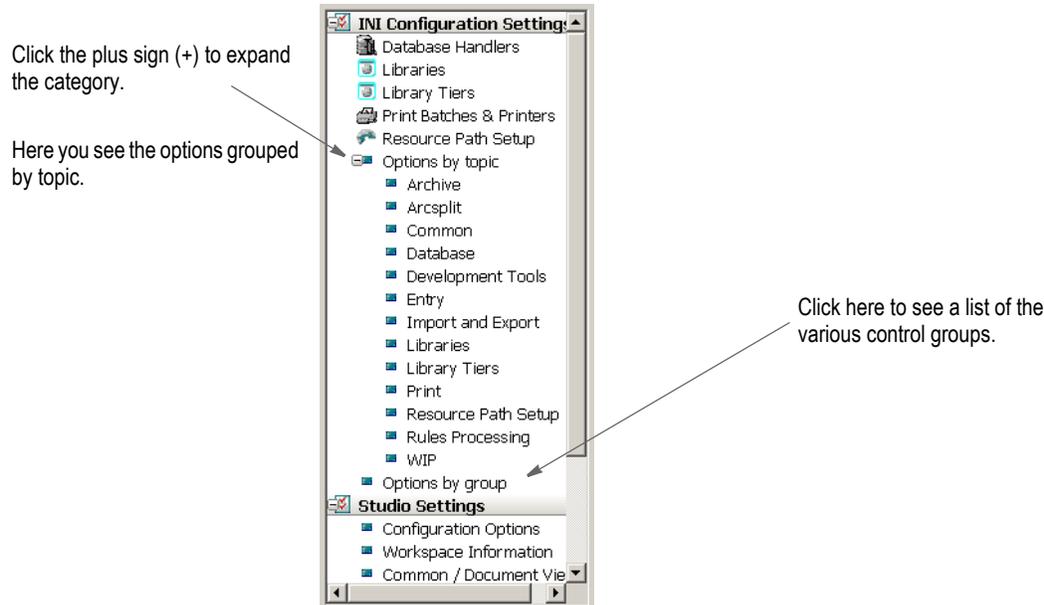


The various categories of settings appear to the left while the individual options appear on the right.

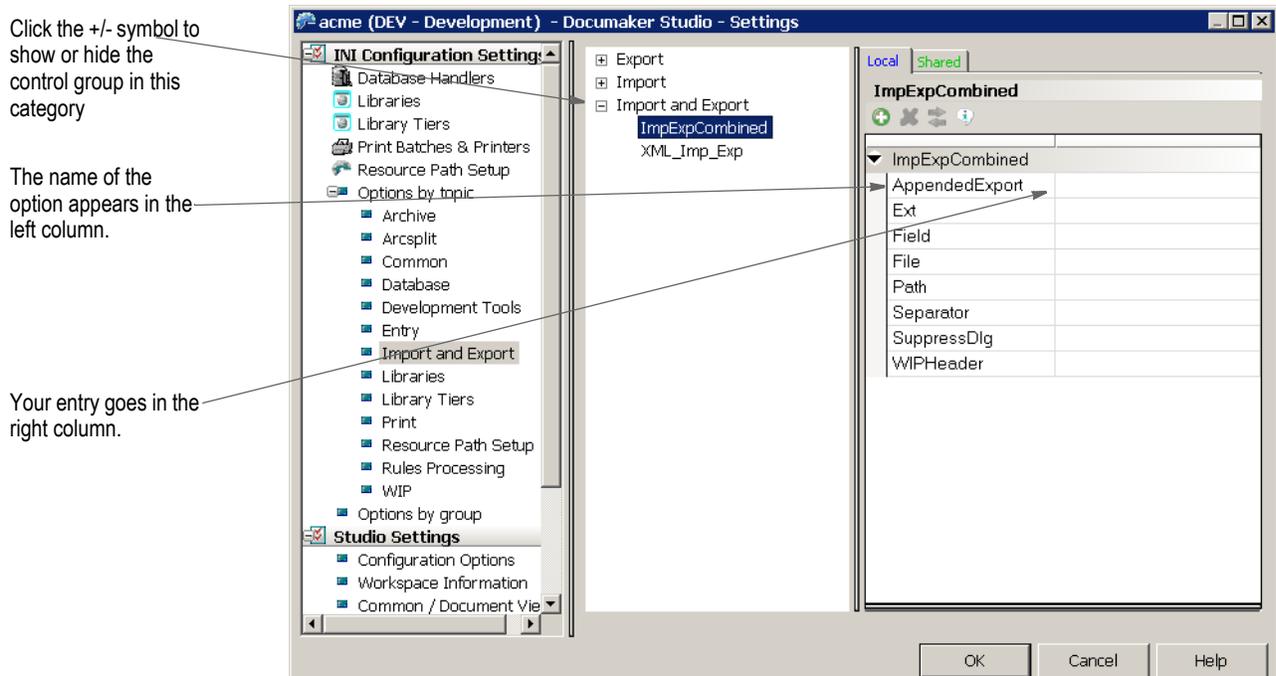
Note You can use the File, Log On As Different User option to change to another user ID.

You can find individual INI options a number of ways:

To	Click on
Go to a specific control group	Options by group.
Choose from lists of related options	The appropriate category or topic.



For instance, if you expand the Import and Export topic, you see the following control groups and options:

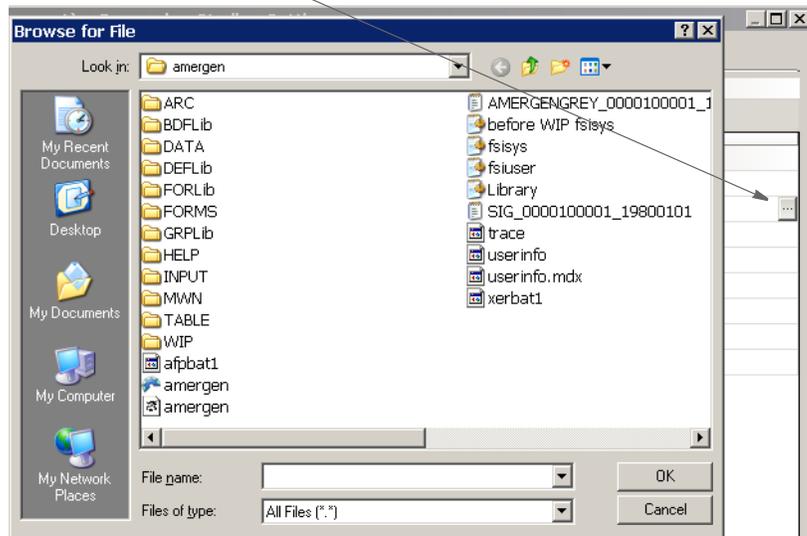


To make an entry for an INI option, click in the entry area beside the option. For some options, you simply enter text. For others, an icon appears to let you browse for a file, choose from a list of pre-set options, or to specify a parameter comprised of several elements.

Here are some examples:

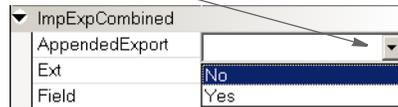
Browsing for a file

Click here to display the Browse for File window.



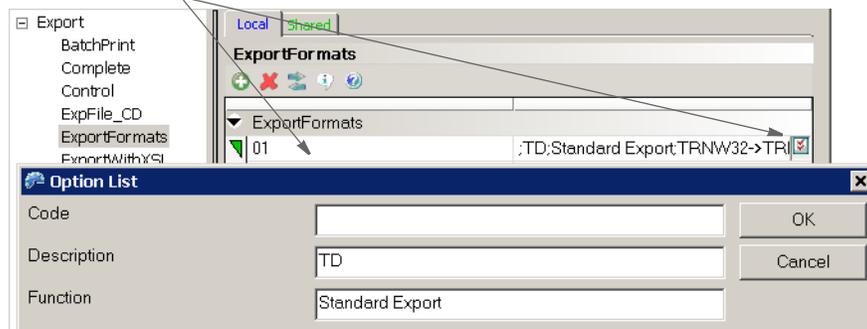
Choosing from a list of options

Click here to display the list of pre-set options.



Specifying a parameter comprised of several elements

Click here to display the Option List window.



CHOOSING OPTIONS BY TOPIC

Studio groups the INI options into these topics:

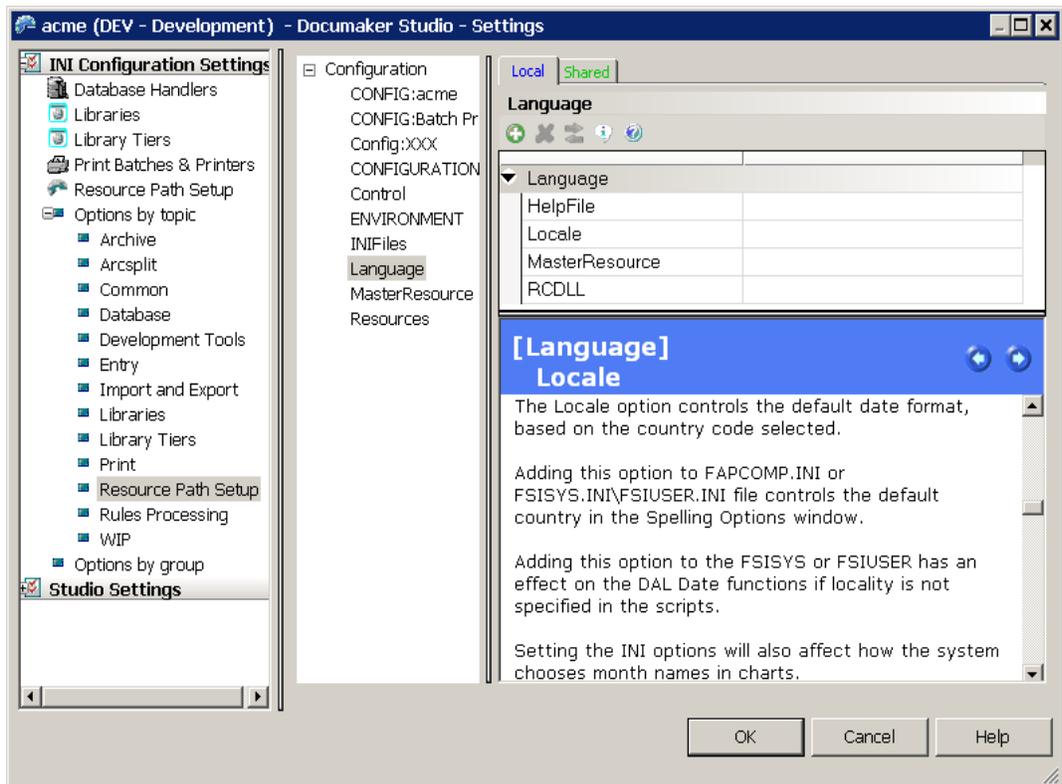


This table describes the topics:

Topic	See
Archive	Here you can define options common to all archives, archive utilities, the GenArc program, and archiving via Documaker Workstation.
ArcSplit	Use these options to define what should happen when you split an archive (CAR) file. You can perform this task from within Documaker Workstation or using the ARCSPLIT utility. The INI options are grouped based on those two approaches
Common	This is where you define settings common to all facets of Studio. This includes interface options, miscellaneous options, and print options
Database	Some of the information you work with, such as archive information, must be stored in database tables. Studio lets you use several types of databases, such as ODBC, DB2 and Oracle. From this topic you can add and delete database handlers. You can also customize the properties of these handlers. When adding a database handler, Studio provides a wizard to guide you through the setup process.
Development Tools	These options let you control how you work with Studio From here you can set how often Studio should automatically save your work or if it should automatically run a spell check when you close an applicable object. You can also set the default colors for the objects you place on sections, such as boxes and bar codes and set up font family matching.
Entry	Use these options to define how the data entry facilities work.
Import and Export	This is where you define how Studio will import and export information.
Libraries	Here you set up the library you want to work with. Libraries store the resources you use as you build and process information on form sets. These resources include sections, graphics, and forms. You can also set up modes, statuses, classes, and projects using these options.
Library Tiers	The Library Tiers options are part of the project management tools currently under development and are only available to Beta testers.

Topic	See
Print	Click the Print Batches & Printers options to set up the print batches and printer options you will use to publish your forms.
Resource Path Setup	This is where you tell Studio where to find the resources you will work with and it will use to build form sets. This is also where you set the Locale option. The Locale option, un the Language control group, determines the default units of measure, date format, and so on.
Rules Processing	This is where you define options that affect how rules are processed via the GenData and GenTm programs.
WIP	This is where you define options that affect work-in-progress. This includes options for the GenWIP program and options that affect how WIP is handled via a workstation.

For instance, if you choose Resource Path Setup, expand the Configuration category and highlight the Language control group, you see these INI options:



The Locale option is selected and in the Information pane, Studio provides information on the highlighted option.

If you right-click within the middle panel, you can choose from these options:

Option	Description
Add Option	Tells Studio to add a new INI option. Studio asks for the name of the control group, the name of the option, and the value you want to assign to the option.
Expand All	Expand all collapsed categories.

Option	Description
Collapse All	Collapse all expanded categories.

If you right-click in a option field that does not offer a list, you can choose from these options:

Option	Description
Undo	Restore the field's contents.
Cut	Remove the highlighted text and place it on the clipboard.
Copy	Copy the highlighted text.
Paste	Insert the contents of the clipboard.
Delete	Delete the highlighted text.
Select All	Highlight the entire contents of the field.

DEFINING STUDIO SETTINGS

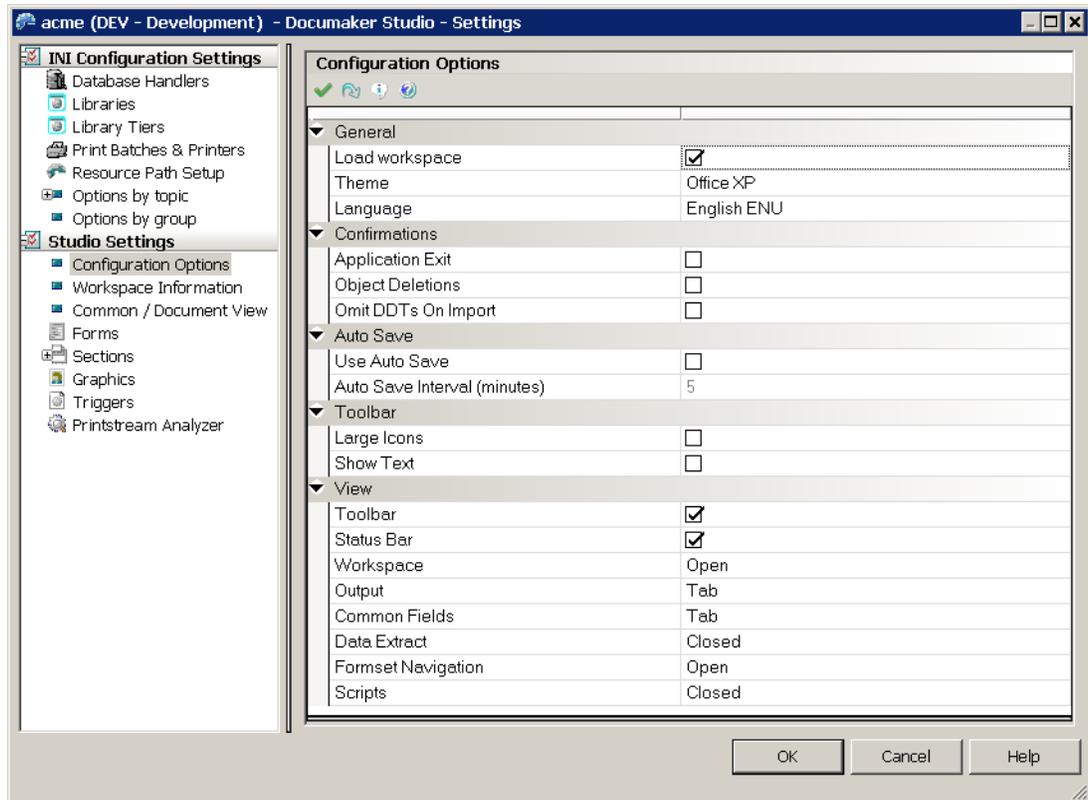
Defining Studio settings is often one of the first tasks you will do. These settings affect how all documents appear on your screen. You can define Studio settings by choosing the Manage, System, Settings option from the main menu. Then click Studio Settings.

From Studio Settings you can also work with options that control the following:

To	See
Set Studio configuration options	<i>Working with Configuration Options on page 47</i>
Define workspace options	<i>Defining Workspace Options on page 50</i>
Set common and document viewing options	<i>Working with Common/Document View Settings on page 52</i>
Work with form options	<i>Working with Form Options on page 54</i>
Work with section options	<i>Working with Section Options on page 56</i>
Work with graphic options	<i>Working with Graphic Options on page 60</i>
Work with trigger options	<i>Working with Trigger Options on page 62</i>
Set Printstream Analyzer options	<i>Setting Up Printstream Analyzer on page 64</i>

WORKING WITH CONFIGURATION OPTIONS

To set basic Studio configuration options, click Configuration Options. This window appears:



This table explains your options:

Option	Description
General	
Local Workspace	Check this box if you want Studio to automatically open the last workspace you were working with when you start Studio.
Theme	Choose the Windows theme you want to use. You can choose from: <ul style="list-style-type: none"> Office 2003 Office 2007 Office XP Whidbey
Language	Select the appropriate language from the list. English is the default.
Confirmations	
Application Exit	Check this box if you want to see a confirmation message each time you exit an application.
Object Deletions	Check this box if you want to see a confirmation message each time you delete an object.

Option	Description
Omit DDTs on Import	Check this box if you want to omit DDT files on import. Some import systems do not have DDT files. This also applies to the trigger file when you are importing a form definition file.
Auto Save	<p>Use these options to define Studio's Auto Save feature.</p> <p>The naming convention Studio uses differs from the one used in Section manager. For instance, assume the name of the FAP file is <i>test.fap</i>.</p> <ul style="list-style-type: none"> • Section manager creates a file named <i>test.fap~</i>. • Studio creates a file named <i>test.fap~\$\$</i>. <p>Studio also automatically saves other file types such as BDF, GRP, and FOR files. The naming convention used for these files is the same, for example <i>form1.for~\$\$</i>.</p> <p>Database files are not affected by the Use Auto Save option. Database files are updated after every record change.</p>
Use Auto Save	Check this box if you want Studio to automatically save your work at timed intervals.
Auto Save Interval	<p>If you checked the Use Auto Save box, enter the number of minutes you want to elapse between auto saves. The default is five minutes. The Auto Save interval is applied individually to each document you have open. For instance...</p> <p>You open document A at 10:00 You open document B at 10:02 Studio saves document A at 10:05 Studio saves document B at 10:07</p>
Toolbar	
Large Icons	Check this box if you want Studio to display large icons on the toolbars.
Show Text	Check this box if you want Studio to display the names of the icons.
View	
Toolbar	Check this box if you want Studio to display the toolbar.
Status Bar	Check this box if you want Studio to display the status bar.
Workspace	<p>Choose the window mode for your workspace. You can choose from these options:</p> <ul style="list-style-type: none"> • Closed • Open • Tab
Output	<p>Choose the window mode for the output area. You can choose from these options:</p> <ul style="list-style-type: none"> • Closed • Open • Tab <p>If an error occurs, Studio displays the corresponding message in the output area even if it was hidden.</p>
Common Fields	<p>Choose the window mode. You can choose from these options:</p> <ul style="list-style-type: none"> • Closed • Open • Tab

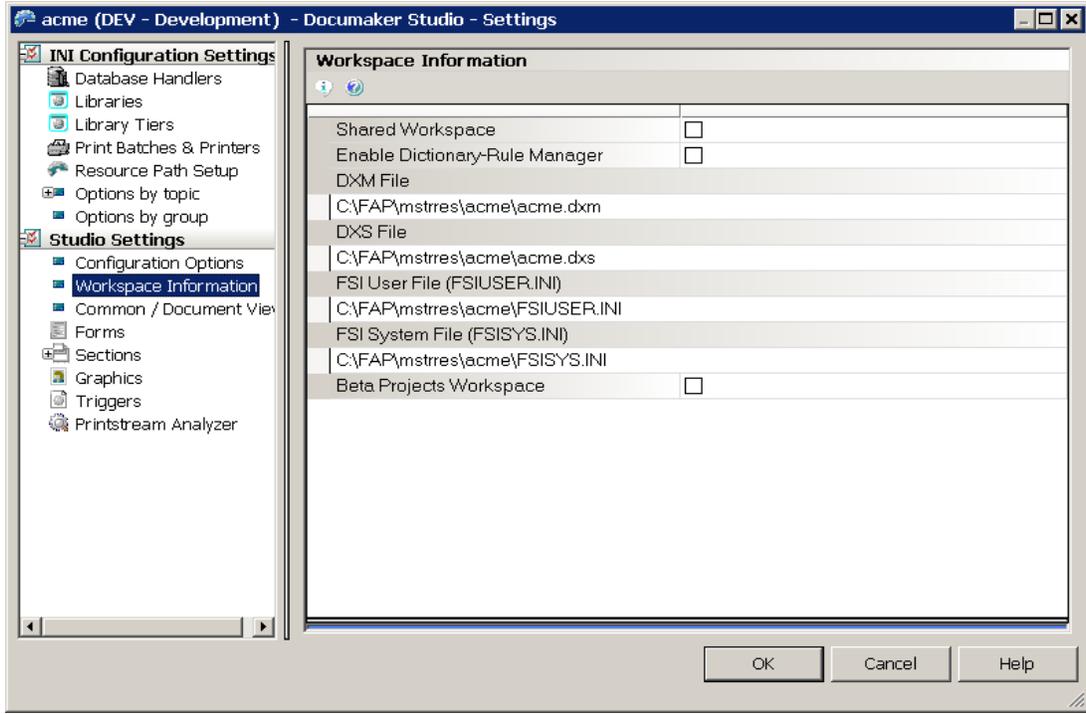
Option	Description
Data Extract	Choose the window mode. You can choose from these options: <ul style="list-style-type: none"> • Closed • Open • Tab
Formset Navigation	Choose the window mode. You can choose from these options: <ul style="list-style-type: none"> • Closed • Open • Tab
Scripts	Choose the window mode. You can choose from these options: <ul style="list-style-type: none"> • Closed • Open • Tab

When you right-click in this window you can choose from these options:

Option	Description
Print this Window	Prints the contents of the window.
Export this Window to File	Lets you copy the contents of this window to a comma-separated value (CSV) file.
Apply Changes	Lets you apply your changes.
Default Settings	Restores the default settings.
Toggle Information Pane	Shows or hides the information pane. This pane provides information about the INI option currently highlighted.
Help	Displays Help information.

DEFINING WORKSPACE OPTIONS

To work with configuration settings, click Configuration Options. This window appears:



General information about your workspace files is shown here, including the location of the DXM and DXS workspace files and the names of the INI files in use. This table explains your options:

Option	Description
Shared Workspace	Studio checks this box if the current workspace is a shared workspace.
Enable Dictionary-Rule Manager	Studio checks this box if this workspace uses the Dictionary Rule manager. If you use a master.ddt file, be sure to check this option.
DXM File	Here Studio shows you the path and file name for the DXM file currently in use.
DXS File	Here Studio shows you the path and file name for the DXS file currently in use.
FSI User File	Here Studio shows you the path and file name for the FSIUSER.INI file currently in use.
FSI System File	Here Studio shows you the path and file name for the FSISYS.INI file currently in use.
Beta Projects Workspace	A check indicates this workspace is project based.

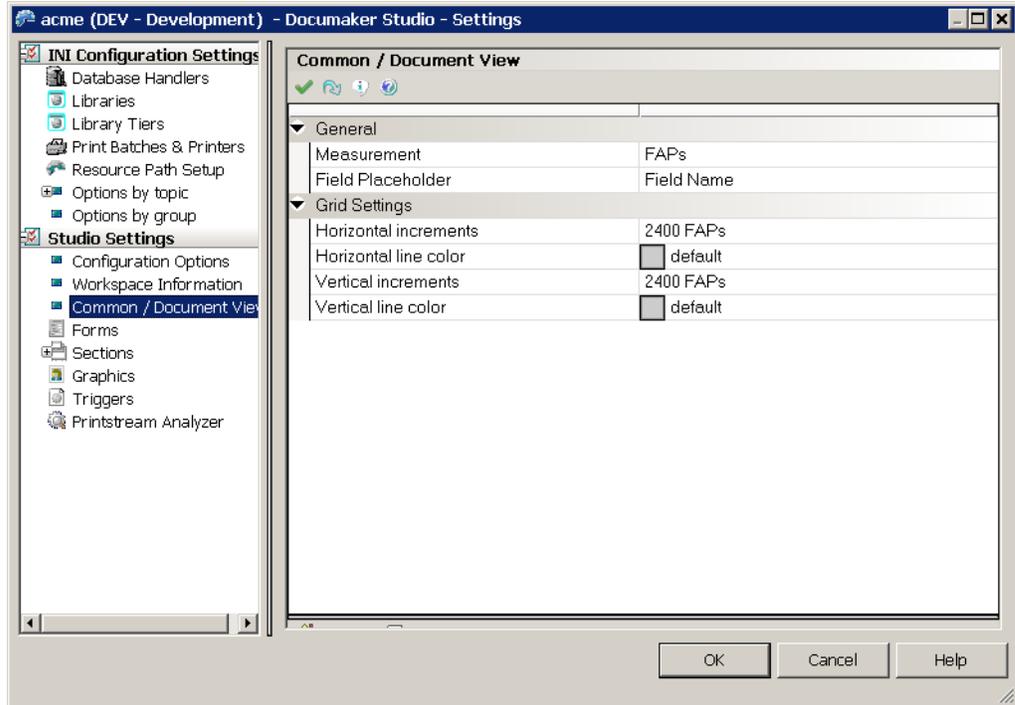
When you right-click in this window you can choose from these options:

Option	Description
Print this Window	Prints the contents of the window.

Option	Description
Export this Window to File	Lets you copy the contents of this window to a comma separated value (CSV) file.
Apply Changes	Lets you apply your changes.
Default Settings	Restores the default settings.
Toggle Information Pane	Shows or hides the information pane. This pane provides information about the INI option currently highlighted.
Help	Displays Help information.

WORKING WITH COMMON/DOCUMENT VIEW SETTINGS

To work with common document viewing settings, click Common/Document View. This window appears:



This table explains your options:

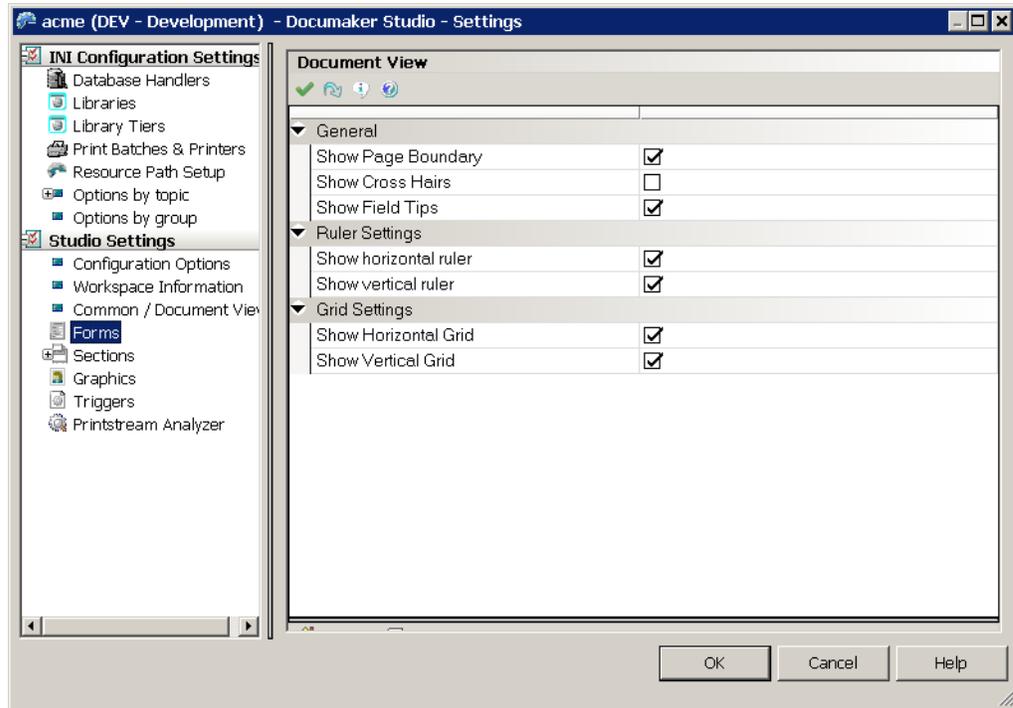
Option	Description
General	
Measurement	Select unit of measurement in which display will be made. You can choose from centimeters, FAP units (2400 per inch), inches, picas, or points.
Field Placeholder	Choose the style of placeholder you want to use to indicate entry fields. You can choose from Template or Field Name style.
Grid Settings	
Horizontal increments	Lets you set the horizontal size of a grid block.
Horizontal line color	Click the icon in this field to choose the color for the horizontal grid lines.
Vertical increments	Lets you set the vertical size of a grid block.
Vertical line color	Click the icon in this field to choose the color for the vertical grid lines.

When you right-click in this window you can choose from these options:

Option	Description
Print this Window	Prints the contents of the window.
Export this Window to File	Lets you copy the contents of this window to a comma separated value (CSV) file.
Apply Changes	Lets you apply your changes.
Default Settings	Restores the default settings.
Toggle Information Pane	Shows or hides the information pane. This pane provides information about the INI option currently highlighted.
Help	Displays Help information.

WORKING WITH FORM OPTIONS

Choose Forms to define how you view documents on your screens. This includes setting up the grid and rulers. This window appears:



This table explains your options:

Option	Description
General	
Show Page Boundary	Check this box to have Studio highlight areas that may fall outside a page boundary.
Show Cross Hairs	Check this box to have Studio draw cross hairs to indicate the mouse position.
Show Field Tips	Check this box to have Studio display a information about the fields that appear in the window.
Ruler Settings	
Show Horizontal Ruler	Check this box to show the horizontal ruler.
Show Vertical Ruler	Check this box to show the vertical ruler.
Grid Settings	
Show Horizontal Grid	Check this box to display the horizontal grid.
Show Vertical Grid	Check this box to display the vertical grid

When you right-click in this window you can choose from these options:

Option	Description
Print this Window	Prints the contents of the window.
Export this Window to File	Lets you copy the contents of this window to a comma separated value (CSV) file.
Apply Changes	Lets you apply your changes.
Default Settings	Restores the default settings.
Toggle Information Pane	Shows or hides the information pane. This pane provides information about the INI option currently highlighted.
Help	Displays Help information.

WORKING WITH SECTION OPTIONS

The section options fall into two categories:

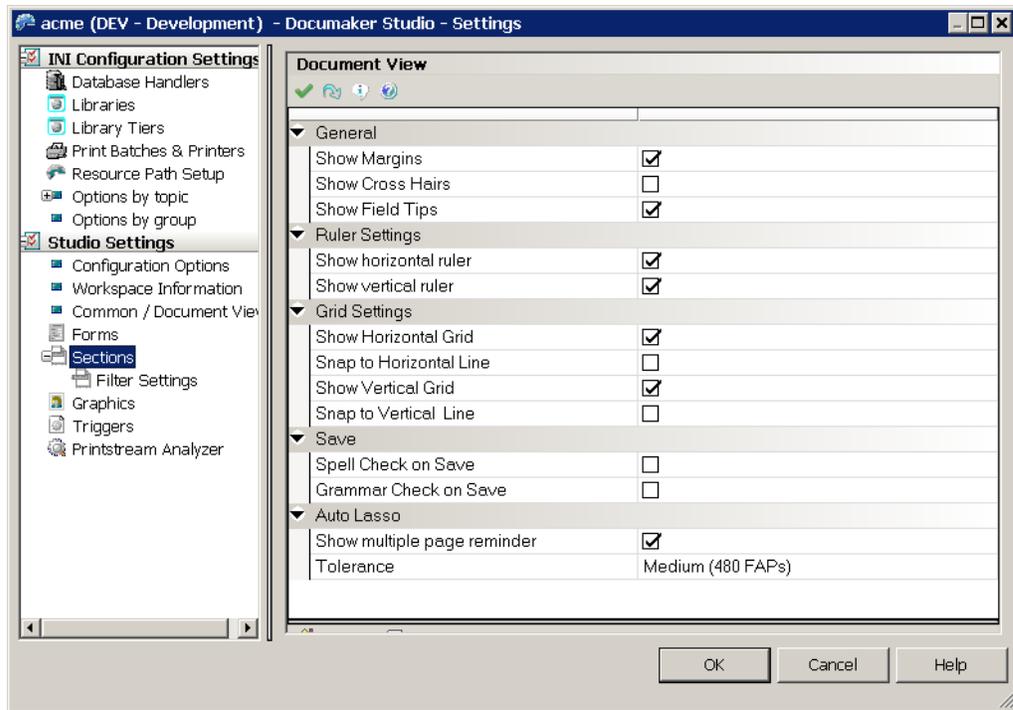
- Document view options
- Filter settings

The following topics discuss these options.

Setting Document View Options

These options control how Studio displays documents. For instance, you use these options to turn on or off rulers, show or hide the grid, and turn on or off automatic spell and grammar checking.

When you click Sections, this window appears:



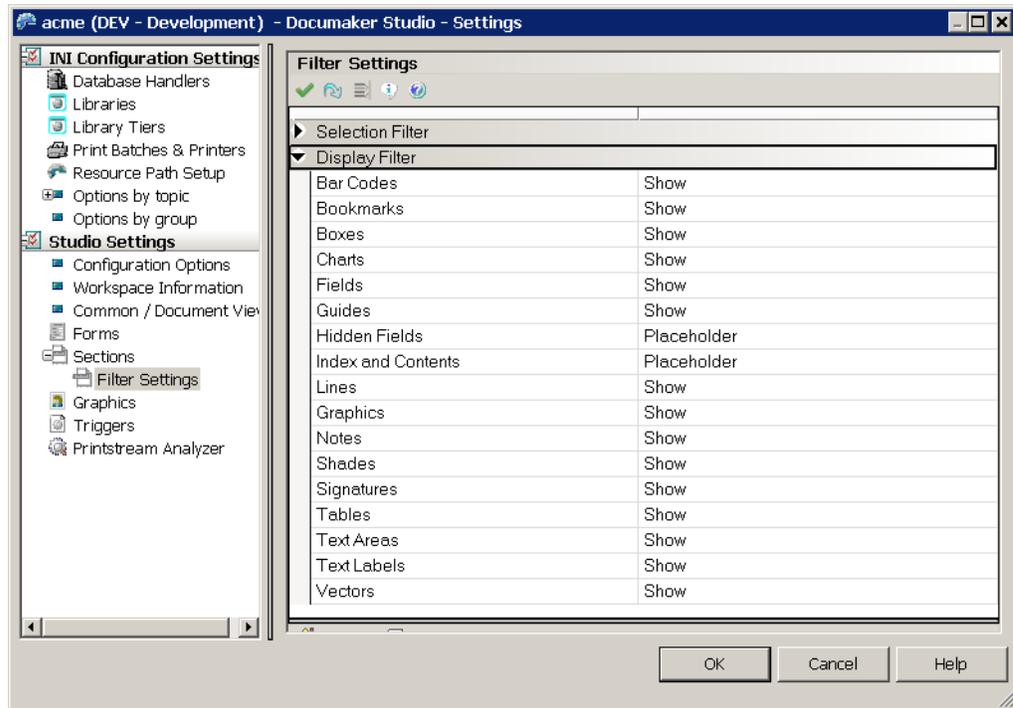
This table explains your options:

Option	Description
General	
Show Margins	Check this box to have Studio draw a red line on the page to indicate the page margins.
Show Cross Hairs	Check this box to have Studio draw cross hairs to indicate the mouse position.
Show Field Tips	Check this box to have Studio display a information about the fields that appear in the window.
Ruler Settings	

Option	Description
Show Horizontal Ruler	Check to show the horizontal ruler.
Show Vertical Ruler	Check to show the vertical ruler.
Use a Ruler in this Tool	Check to use a ruler when working with sections.
Grid Settings	
Show horizontal grid	Check to display the horizontal grid.
Snap to horizontal line	By checking, the object entered will snap (be placed) to the closest horizontal line below it.
Show vertical grid	Check to display the vertical grid.
Snap to vertical line	By checking, the object entered will snap (be placed) to the closest vertical line to the left of it.
Save	
Spell check on Save	Check this option to have Studio automatically check spelling when you save a section.
Grammar check on Save	Check this option to have Studio automatically check grammar when you save a section.
Auto Lasso	
Show multiple page reminder	Check this option to have Studio show a reminder when Auto Lasso is clicked for a section that is comprised of multiple pages.
Tolerance	<p>Here you indicate the amount of separation items can possess and still be grouped together. You can choose from: Low, Medium, and High, with High indicating a more aggressive lasso than Low.</p> <p>The values associated with Low, Medium, and High reflect the unit of measurement you chose. This is what you get if you chose FAP units:</p> <ul style="list-style-type: none"> • Low = 240 FAPs • Medium = 480 FAPS • High = 960 FAPS <p>240 FAPs is equal to 1/10th of an inch.</p> <p>For example, if you choose High, objects separated by less than 960 FAPs (4/10s of an inch) are automatically lassoed.</p>

Setting Section Filter Options

Choose Filter Settings to define the filter settings that apply when you are working with sections. This window appears:



For each type of object, you can choose from these selection filter settings:

Option	Description
Cannot move	Check this option if you do not want anyone to move this type of object.
Non-selectable	Check this option if you do not want anyone to select this type of object.
Selectable	Check this option to let people select this type of object.

Display filter

For each type of object, you can choose from these display filter settings:

Option	Description
Hide	Tells Studio to list the object in the object tree but hide it in the work area.
Show	Tells Studio to show the object as you create it.
Placeholder	Tells Studio to display a placeholder instead of the actual object.

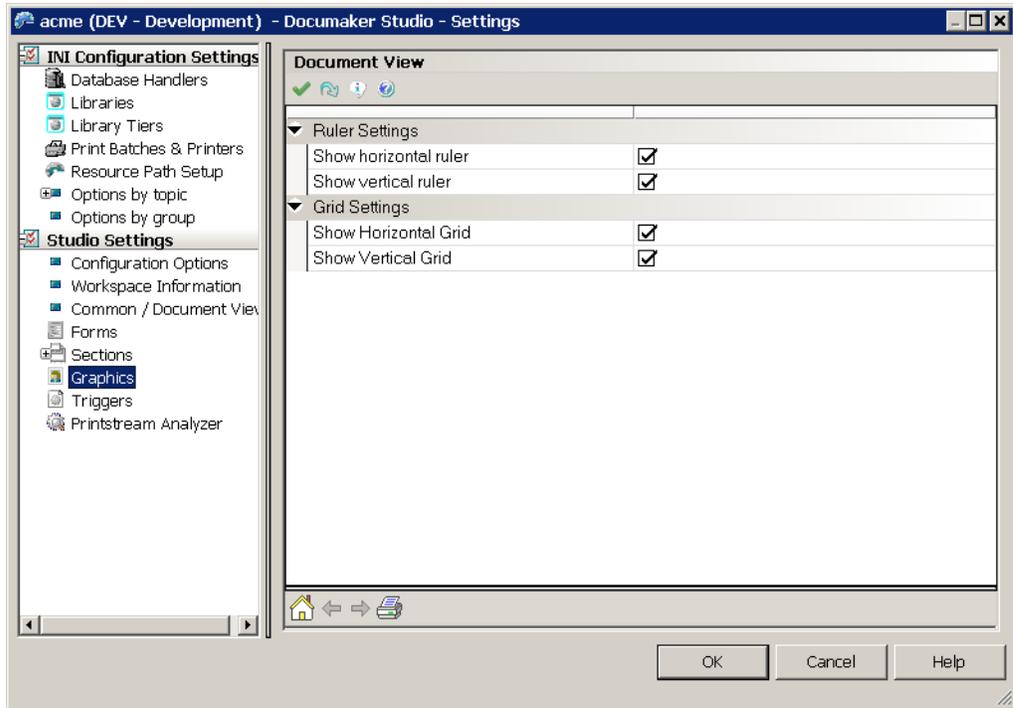
In the example shown previously, Hidden Fields, Indexes, and Contents would be shown with a placeholder while Guides and Notes would be hidden. Studio would show the other objects.

When you right-click in this window you can choose from these options:

Option	Description
Print this Window	Prints the contents of the window.
Export this Window to File	Lets you copy the contents of this window to a comma separated value (CSV) file.
Apply Changes	Lets you apply your changes.
Default Settings	Restores the default settings.
Toggle Information Pane	Shows or hides the information pane. This pane provides information about the INI option currently highlighted.
Help	Displays Help information.

WORKING WITH GRAPHIC OPTIONS

These options control how Studio displays graphics. For instance, you use these options to select the unit of measurement, turn on or off rulers, set ruler colors, and so on. When you click Document View, this window appears:



This table explains your options:

Option	Description
Ruler Settings	
Show horizontal ruler	Check this option if you want Studio to show the horizontal ruler.
Show vertical ruler	Check this option if you want Studio to show the vertical ruler.
Grid Settings	
Show Horizontal Grid	Check this option if you want Studio to display the horizontal grid.
Show Vertical Grid	Check this option if you want Studio to display the vertical grid.

When you right-click in this window you can choose from these options:

Option	Description
Print this Window	Prints the contents of the window.
Export this Window to File	Lets you copy the contents of this window to a comma separated value (CSV) file.
Apply Changes	Lets you apply your changes.
Default Settings	Restores the default settings.

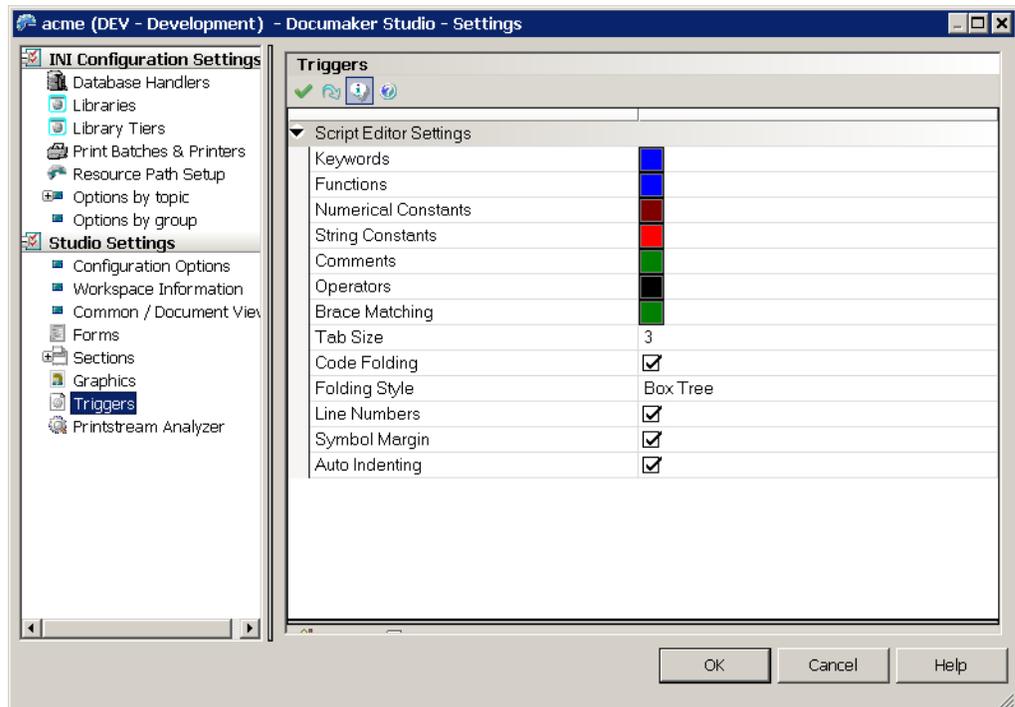
Option	Description
Toggle Information Pane	Shows or hides the information pane. This pane provides information about the INI option currently highlighted.
Help	Displays Help information.

WORKING WITH TRIGGER OPTIONS

Use these options to set up the color for the various elements of a DAL script. For instance, for readability purposes you can define different colors for the following:

- Keywords and functions
- Numeric and string constants
- Comments
- Operators
- Brace Matching

When you choose Triggers, this window appears:



This table explains the syntax color options:

Option	Description
Keywords	Click the icon in this field to display the Color Selection window so you can select the color you want to use.
Functions	Click the icon in this field to display the Color Selection window so you can select the color you want to use.
Numerical Constants	Click the icon in this field to display the Color Selection window so you can select the color you want to use.
String Constants	Click the icon in this field to display the Color Selection window so you can select the color you want to use.
Comments	Click the icon in this field to display the Color Selection screen and select the color you want to use.

Option	Description
Operators	Click the icon in this field to display the Color Selection screen and select the color you want to use.
Brace Matching	Click the icon in this field to display the Color Selection screen and select the color you want to use.
Tab Size	Enter the number of spaces you want Studio to use to represent a TAB character. The default is three (3).
Code Folding	Check this box to enable or disable code folding. Code folding lets you collapse or expand blocks of code.
Folding Style	When code folding is enabled, use this option to select the folding style. You can choose from Arrow, Simple, Circle Tree, or Box Tree.
Line Numbers	When selected, a line number appears to the left of each line of code. Remove this check mark to hide line numbers.
Symbol Margin	When selected, a margin is reserved to the left of each line of code for symbol display. If not selected, then markers change the background color of the line rather than displaying a symbol.
Auto Indenting	When selected and you press ENTER, the new line of text is automatically indented to the same tab stop as the line preceding it. Remove this check mark to tell Studio to turn off auto indenting.

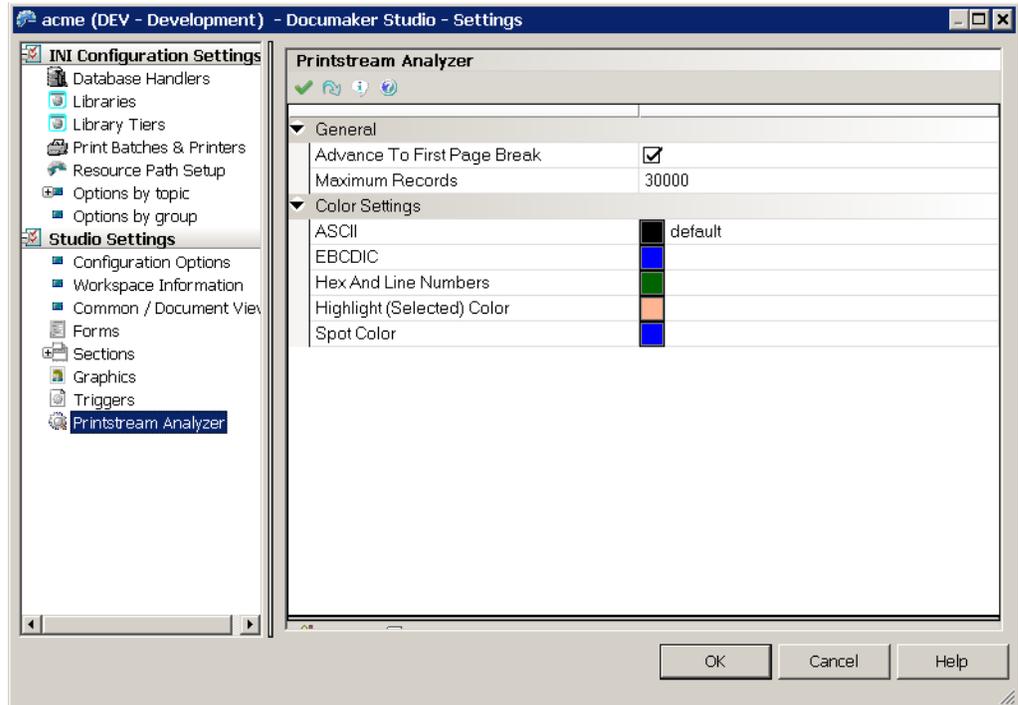
When you right-click in this window you can choose from these options:

Option	Description
Print this Window	Prints the contents of the window.
Export this Window to File	Lets you copy the contents of this window to a comma separated value (CSV) file.
Apply Changes	Lets you apply your changes.
Default Settings	Restores the default settings.
Toggle Information Pane	Shows or hides the information pane. This pane provides information about the INI option currently highlighted.
Help	Displays Help information.

SETTING UP PRINTSTREAM ANALYZER

You can use the Printstream Analyzer to open and analyze AFP or Metacode print streams. The Printstream Analyzer is available from the workspace tree and from the Manage, Tools menu.

To set options that affect Printstream Analyzer, click Printstream Analyzer. This window appears:



This table explains your options:

Option	Description
Advance to First Page Break	Check this box to tell Printstream Analyzer to move the selection in the Record View pane to the first page break record when a print file is opened. Some print files contain comments or other records before the first page. These comments or records are considered page zero (0). The Display pane is blank when you select page zero (0).
Maximum Records	Limits the number of records read into the Record View pane. A smaller number of records means the print file loads more quickly.
ASCII	The color used for ASCII records in the Record View pane.
EBCDIC	The color used for EBCDIC records in the Record View pane.
Hex and Line Numbers	The color used for the hexadecimal and line numbers on the sides of the Record View pane.
Highlighted (Selected) Color	The color used for highlighted or selected text and images in the Display pane.
Spot Color	The color used for spot color text and lines in the Display pane

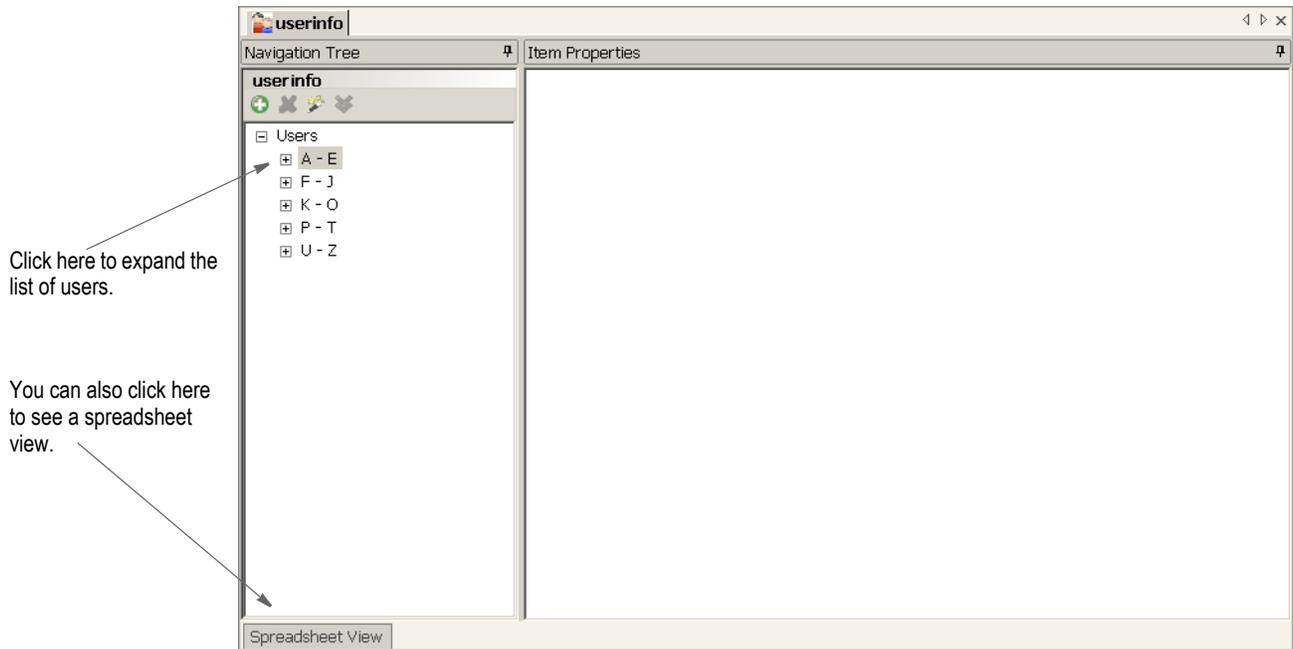
When you right-click in this window you can choose from these options:

Option	Description
Print this Window	Prints the contents of the window.
Export this Window to File	Lets you copy the contents of this window to a comma separated value (CSV) file.
Apply Changes	Lets you apply your changes.
Default Settings	Restores the default settings.
Toggle Information Pane	Shows or hides the information pane. This pane provides information about the INI option currently highlighted.
Help	Displays Help information.

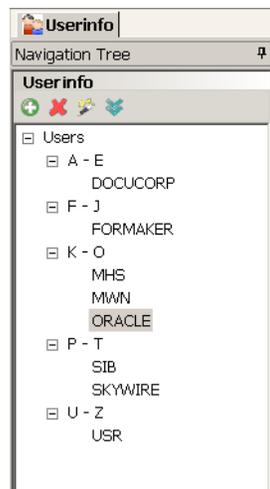
MANAGING USERS

Choose the Manage, System, Users option to work with the user profiles you have set up. Here is an example of the window that appears if you are authorized to work with user profiles:

Note You must have an adequate security level to work with users. Otherwise, these options are not available to you.



To see a list of users you can right click under Users to expand the items in the tree, as shown here:



Or click the Spreadsheet View tab, as shown here:

User ID:	User Name:	Password:	Workstation Rights:	In Use:	Report To:	Security	Message:
DOCUCORP	Docucorp	Change Password	0	Yes		Security	
FORMAKER	Formmaker	Change Password	0	No		Security	
MHS	Mary Hallie	Change Password	0	No		Security	
MWN	Michael Wayne	Change Password	0	No		Security	
ORACLE	Oracle	Change Password	0	No		Security	
SIB	Susan Ingram	Change Password	0	No		Security	
SKYWIRE	Skywire	Change Password	0	No		Security	
USR	Ulysses Simpson	Change Password	9	No		Security	

To work with a specific user, click on that user. You can work with user information directly from the Spreadsheet View or from the Item Properties window.

Once you select a user, Studio shows you the basic information for that user.

Item Properties	
User ID:	DOCUCORP
User Name:	Docucorp
Message:	
Report To:	
Workstation Rights:	0
In Use:	Y
Password:	XXXXXXXXXX
Verify Password:	XXXXXXXXXX
Security	Configure...

Studio shows you information about...

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

It also tells you if there is a message set up to appear to the user when he or she logs in. The tools on the User Info toolbar let you quickly add new users, delete users, open the security wizard, or clone a user.

Click here to add a user

Click here to delete a user

Click here to open the Security wizard

Click here to clone a user's security rights.

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

ADDING USERS

To add a user, click the Add User tool on the toolbar:

Click here to add a user.

Then fill in the following fields to add a user:

Item Properties	
User ID:	NEWUSER000
User Name:	
Message:	
Report To:	DOCUCORP
Workstation Rights:	9
In Use:	N
Password:	*****
Verify Password:	*****
Security	Configure...

Field	Description
User ID	Enter an ID for the user. User IDs can contain up to 64 alphanumeric characters. Each ID must be unique.
User Name	Enter the user's full name.
Message	Enter the message you want Studio to display to the user when he or she starts Studio.
Report To	Select the user's supervisor from the list that appears when you click the arrow.
Workstation Rights	Enter a number between 1 and 9 for the workstation access level. Level zero (0) is reserved for the system supervisor. <i>Docucorp</i> is the default system supervisor. Studio makes no distinction between levels 1-9, so you can define these levels any way you like.
In Use	This field shows whether a user is logged into Studio. With proper rights, you can change the In Use status, which may be necessary if you need to reset a locked user ID. User IDs can become locked if the system is shut down inadvertently, such as when there is a loss of power. Once locked, Studio will display a message stating the user ID is already in use the next time that user attempts to log on.
Password	Enter the password you want to assign to this user ID. Studio displays asterisks (*) as you type the password.
Verify Password	Enter the password again to verify it.
Security	Click the Configure button to open the Security wizard. From the Security wizard you can... <ul style="list-style-type: none"> • Determine if the user is a system administrator. • Set general user rights, such as the right to manage other users, access your own record, change passwords, and so on. • Grant access to INI options. • Control library access and the actions the user can perform within the library. • Limit access to the various managers, such as Forms, Section, and Trigger manager. • Define creation rights for fields, triggers, templates, or all resources. • Assign a project role, such as an administrator, developer, tester, reviewer, and so on. For more information, see <i>Setting Rights Using the Security Wizard on page 69</i> .

Note Studio's Conversion wizards check the user's security settings before saving adjustments to INI files.

During conversions, Studio will still adjust INI values as necessary, but the security check prevents Studio from automatically saving those changes, unless you have the proper security settings.

For example, performing conversions from normalized Metacode or AFP to a section (FAP) can alter some INI settings. Those settings are not saved, however, unless the user has the appropriate security rights.

CHANGING USER INFORMATION

To change a user's information, select the user and then simply type into the appropriate field. Studio records your changes when you click to another field or user.

DELETING USERS

To delete a user, highlight the user you want to delete then click the Delete User tool on the toolbar:



Click here to delete a user.

Studio lets you confirm the deletion before it actually deletes the user.

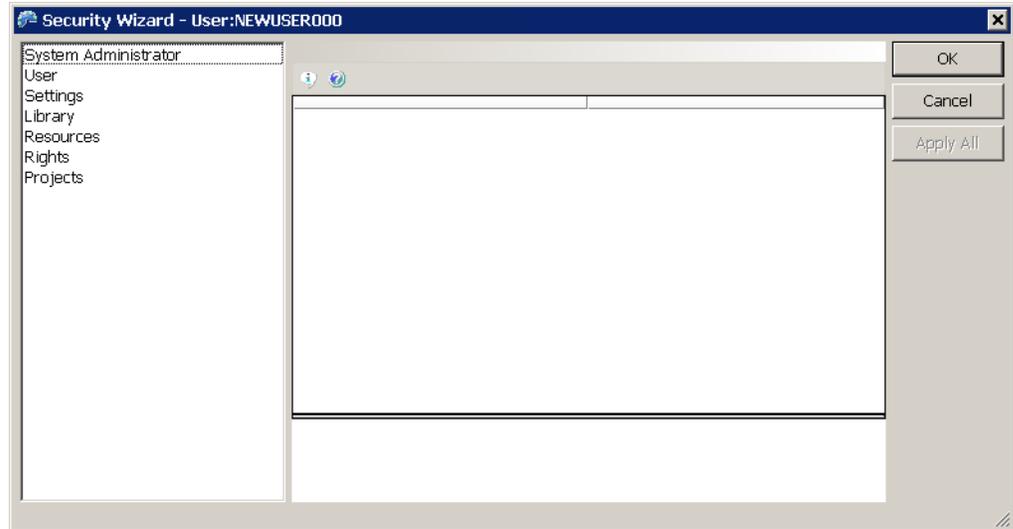
SETTING RIGHTS USING THE SECURITY WIZARD

Use this wizard to modify any security settings for a user. When adding or changing user information, you can click the Configure button in the Security field to open the Security wizard. You can also click the Security wizard icon.



Click here to open the Security wizard.

The Security wizard appears.



From the Security wizard you can control rights in these areas:

Area	Description
System Administrator	Determines if the user is a system administrator. See <i>System Administrator rights on page 70</i> for information about these settings.
User	Sets general user rights, such as the right to manage other users, access your own record, change passwords, and so on. See <i>User rights on page 71</i> for information about these settings.
Settings	Grants access to INI options. See <i>Settings rights on page 71</i> for information about these settings.
Library	Controls library access and the actions the user can perform within the library. See <i>Library rights on page 71</i> for information about these settings.
Resources	Limits access to the various resource managers, such as Forms, Section, and Trigger manager. See <i>Resource rights on page 72</i> for information about these settings.
Rights	Defines creation rights for fields, triggers, templates, or all resources. See <i>Rights on page 74</i> for information about these settings.
Projects	Assigns a project role, such as an administrator, developer, tester, reviewer, and so on. See <i>Project rights on page 75</i> for information about these settings.

System Administrator rights

Click on System Administrator to set these rights:

System Administrator	<input type="checkbox"/>
Access shared workspace file	<input type="checkbox"/>

Field	Description
System Administrator	Check this box to indicate that this user is a system administrator. This gives the user access to all workspace functionality.
Access Shared Workspace File	Check this box if the user is not a System Administrator, but still needs to be able to manage a shared workspace. Typical users will not need this level of access.

User rights

Click on User to set these rights:

User Administrator	<input type="checkbox"/>
Manage Users	<input type="checkbox"/>
Access own record	No Access
Change password on next log in	<input type="checkbox"/>
Cannot change sandbox location	<input type="checkbox"/>

Field	Description
User Administrator	Check this box to indicate that this user is an administrator of the user database. This lets this user activate or deactivate security settings for all users except system administrators.
Manage Users	Check this box to indicate that this user is a manager of other users.
Access Own Record	Check this box if you want to allow this user to access his or her own record in the database. By checking this box, you give this user the right to change the descriptive name and password, but not other attributes.
Change Password on Next Log In	Check this box to give this user the right to change his or her password on the next log in.
Cannot Change Sandbox Location	<p>The default location for a user's sandbox in a shared environment is a directory under the workspace location named for the user. Check this option to prevent this user from changing that location when joining the workspace.</p> <p>The <i>sandbox</i> is the location where the files you have checked out are saved when you are part of a shared workspace. Normally, each user has his or her own sandbox (subdirectory). This subdirectory can be on a local drive or it may be relative to the shared workspace location — which is usually created using the user ID as the name. It is possible for several users to share a sandbox, but that increases the chance that another user might modify or delete files you are working with.</p>

Settings rights

Click on Settings to set these rights:

INI Access	Unable to change INI Settings
Override Settings	<input type="checkbox"/>

Field	Description
INI Access	Check this box if you want this user to change INI option settings. INI options tell the system how you want it to operate. These options are stored in these INI files: FSISYS.INI and FSIUSER.INI. For more information, see <i>Working With INI Options on page 38</i> .
Override Settings	Check this box if you want to let this user override system settings by adding local settings.

Library rights

Click on Library to set these rights:

Library Administrator	<input type="checkbox"/>
Limited Property Modifications	<input type="checkbox"/>
Perform Promotions	<input type="checkbox"/>
Delete Library Entries	<input type="checkbox"/>
Make Effective Date Changes	<input type="checkbox"/>
Modify Prior Versions	<input type="checkbox"/>
Collapse Revisions	<input type="checkbox"/>
Expire/Unexpire Versions	<input type="checkbox"/>
Secure Resources	<input type="checkbox"/>
Purge History	<input type="checkbox"/>
Library create	<input type="checkbox"/>
Check out from non-Dev tier	<input type="checkbox"/>

Field	Description
Library Administrator	Check this box if you want to make this user a library administrator. Library administrators have access to all library functionality.
Limited Property Modifications	Check this box if you want this user to modify the description, mode, status, class and project attributes of library resources.
Perform Promotions	Check this box if you want this user to promote resources within the library.
Delete Library Entries	Check this box if you want this user to delete library entries.
Make Effective Date Changes	Check this box if you want this user to change the effective dates of resources stored in the library.
Modify Prior Versions	Check this box if you want this user to modify prior versions of a library resource.
Collapse Revisions	Check this box if you want this user to collapse library revisions. Note: Collapsing library revisions deletes those revisions.
Expire/Unexpire Versions	Check this box if you want this user to expire and unexpire library resources.
Secure Resources	Check this box if you want this user to have the ability to secure resources. Securing a resource means that it can only be checked out by a certain person or certain group if you are using the LDAP security model. For instance, suppose you have many users that have the right to check out FOR files. But you have a certain FOR file that should be restricted to a certain individual. In that case, you would secure the FOR file to that user ID using Library manager. Then, only that user ID would have the ability to lock that file. See <i>Adding Security at the Resource Level on page 78</i> for more information.
Purge History	Check this box if you want this user to purge library history.
Library Create	Check this box if you want this user to create new library destinations.
Check out from non-Dev tier	Check this box if you want this user to check resources out of a non-development library.

Resource rights

For each resource manager, you can choose to give this user one of these levels of access:

- Full Access — Gives the user full access to this manager.

- **No Access** — Removes references to this manager from the software and does not allow the user to start this manager.
- **View Only Access** — Lets the user to view a resource but not modify it.

Click on Resource to set these rights:

Application Manager	Full Access
Form List Manager	Full Access
Form Manager	Full Access
Section Manager	Full Access
Paragraph List Manager	Full Access
Paragraph Manager	Full Access
Style Manager	Full Access
Template Manager	Full Access
Graphics Manager	Full Access
Trigger Manager	Full Access
Data Extract Manager	Full Access
Common Field Manager	Full Access
Dictionary-Rule Manager	Full Access
Definition Lookup Manager	No Access
Font Manager	No Access
Deployment Manager	No Access

Manager	Description
Application manager	Choose from Full Access, No Access, or View Only Access to set this users ability to access Application manager. This manager lets you work with the business definition file (BDF). If you select No Access, this user will not see options to start this manager.
Form List manager	Choose from Full Access, No Access, or View Only Access to set this users ability to access Form List manager. This manager lets you work with form lists. If you select No Access, this user will not see options to start this manager.
Forms manager	Choose from Full Access, No Access, or View Only Access to set this users ability to access Forms manager. This manager lets you work with forms. If you select No Access, this user will not see options to start this manager.
Sections manager	Choose from Full Access, No Access, or View Only Access to set this users ability to access Sections manager. This manager lets you create, modify, and delete the sections that comprise your forms. If you select No Access, this user will not see options to start this manager.
Paragraph List manager	Choose from Full Access, No Access, or View Only Access to set this users ability to access Paragraph List manager. This manager lets you work with lists of paragraphs. If you select No Access, this user will not see options to start this manager.
Paragraph manager	Choose from Full Access, No Access, or View Only Access to set this users ability to access Paragraph manager. This manager lets you create, modify, and delete individual paragraphs. If you select No Access, this user will not see options to start this manager.
Style manager	Choose from Full Access, No Access, or View Only Access to set this users ability to access Style manager. This manager lets you create, modify, and delete individual styles that appear on your templates. If you select No Access, this user will not see options to start this manager.

Manager	Description
Template manager	Choose from Full Access, No Access, or View Only Access to set this users ability to access Template manager. This manager lets you work with the templates you use when creating forms. If you select No Access, this user will not see options to start this manager.
Graphics manager	Choose from Full Access, No Access, or View Only Access to set this users ability to access Graphics manager. This manager lets you work with graphics (LOG) files. If you select No Access, this user will not see options to start this manager.
Trigger manager	Choose from Full Access, No Access, or View Only Access to set this users ability to access Triggers manager. This manager lets you create, modify, and delete the triggers that can determine if a form is printed. If you select No Access, this user will not see options to start this manager.
Data Extract manager	Choose from Full Access, No Access, or View Only Access to set this users ability to access Data Extract manager. This manager lets you work with the fields in the extract dictionary. If you select No Access, this user will not see options to start this manager.
Common Fields manager	Choose from Full Access, No Access, or View Only Access to set this users ability to access Common Fields manager. This manager lets you work with the fields stored in the common fields dictionary. If you select No Access, this user will not see options to start this manager.
Dictionary-Rule manager	Choose from Full Access, No Access, or View Only Access to set this users ability to access Dictionary Rule manager. If you select No Access, this user will not see options to start this manager.
Definition Lookup manager	Choose from Full Access, No Access, or View Only Access to set this users ability to access Definition Lookup manager. This manager lets you work with the definitions used in lookups. If you select No Access, this user will not see options to start this manager.
Font manager	Choose from Full Access, No Access, or View Only Access to set this users ability to access Font manager. This manager lets you work with the font cross-reference (FXR) file. If you select No Access, this user will not see options to start this manager.
Deployment manager	Choose from Full Access or No Access to set this users ability to access Deployment manager. This manager lets you handle the deployment of resources. If you select No Access, this user will not see options to start this manager.

Rights

Click on Rights to change these settings:

Modify existing library resources only	<input type="checkbox"/>
Limit to pre-defined fields	<input type="checkbox"/>
Limit to pre-defined triggers	<input type="checkbox"/>
Limit to using templates	<input type="checkbox"/>

Field	Description
Modify existing library resources only	This right restricts the ability to create resources. If you check this box, the user can only check out and change existing resources.

Field	Description
Limit to pre-defined fields	This right restricts the ability to create a variable field. If you click this option, the user can only choose entries from the field database or the extract dictionary.
Limit to pre-defined triggers	This right restricts the ability to create a trigger. If you click this option, the user can only select triggers from the trigger list.
Limit to using templates	This right restricts the ability to create a form without using a template. If you click this option, the user must choose from pre-defined form templates.

Note Users who are system administrators are not affected by these options.

Project rights

Click on Project to set these rights:

Note Project management features will be implemented in a future release and will use these project rights. The project rights are included in this release for beta testing purposes. If you are not a beta tester, you can ignore these rights.

Project Administrator	<input type="checkbox"/>
Project Developer	<input type="checkbox"/>
Project Tester	<input type="checkbox"/>
Project Reviewer	<input type="checkbox"/>
Project Promotion	<input type="checkbox"/>
Project Outside	<input type="checkbox"/>

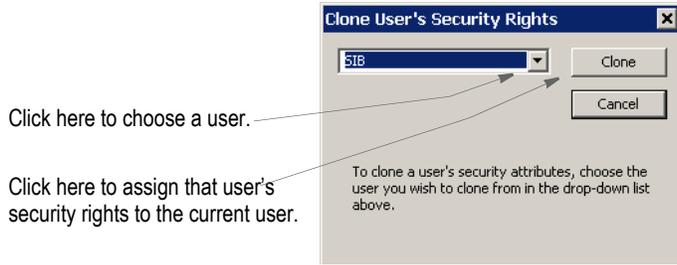
Field	Description
Project Administrator	Check this box to indicate that this user is a project administrator and can create, modify, and delete projects and perform other project management roles.
Project Developer	Check this box to indicate that this user is a project developer.
Project Tester	Check this box to indicate that this user is a project tester.
Project Reviewer	Check this box to indicate that this user is a project reviewer.
Project Promotion	Check this box to indicate that this user is allowed to promote project resources.
Project Outside	Check this box to indicate that this user is allowed to work outside of defined projects.

CLONING USER RIGHTS

Cloning a user's security rights is a quick way to give one user the same level of access rights as another. To clone a user, click the Clone User's Security Rights tool on the toolbar:

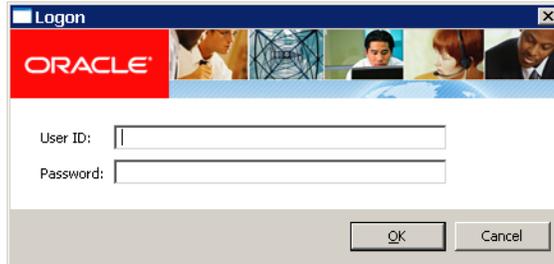


When you click the clone tool, this window appears:



LOGGING IN

Here is an example of the login window:



STORING USER INFORMATION IN ANOTHER DATABASE

By default, the system stores user information in xBase format. You can, however, store user information in SQL or another database format. For example, to use SQL via an ODBC connection to store user IDs, here is how you would set up your INI file:

```
< DBHandler:ODBC >
    Debug           = Yes
    InstallFunc     = SQInstallHandler
    InstallMod      = SQW32
    CreateIndex     = No
    CreateTable     = Yes
    UserID          = sa
    Passwd          = password
    Qualifier       = dms1
    Server          = wipdata
```

Use these options to specify the database type:

Option	Description
Qualifier	Enter the name of the database that will hold the table.
Server	Enter the name of the ODBC connection you made to connect to the database.

The DBTable:USERINFOSQL control group defines the USERINFOSQL table. This is the custom SQL table the system will create if it does not already exist:

```
< DBTable:USERINFOSQL >
    DBHandler      = ODBC
    UniqueIDTag    = UNIQUEIDTAG
    UniqueTag      = IDTAG
    DefaultTag     = UNIQUEIDTAG
    Debug          = Yes
```

If you are using ODBC, the File option should specify the name of the table in the database to use. USERINFOSQL is the custom SQL table that will be created if not present.

```
< UserInfo >
    File           = USERINFOSQL
    SupportSuperUser = Yes
```

Use these options to import user IDs from a default xBase userinfo.dbf file, a comma-delimited text file, or an SQL table:

```
< UserImportFunctions >
    01 = ;Text file;USRMAINT->USRImportText;
    02 = ;Another UserInfo database;USRMAINT->USRImportDBF;
    03 = ;Another database using ODBC;USRMAINT->USRImportODBC;
\DBTable:UserInfo_1
```

Note that the 01 option specifies the name of the table you are importing.

ADDING SECURITY AT THE RESOURCE LEVEL

You can secure resources in Library manager so only the users you authorize can check them out. For instance, this lets you control who can change a graphics (LOG) file that contains a signature.

Studio provides broad categories for security, such as access by file type (BDF, GRP, FOR, FAP, and so on). For instance you can give a user ID full, access, no access, or view-only access to all forms and images. You may, however, have resources that need specific restriction, such as a need to restrict access to LOG files to specific individuals, or groups if you are using the Lightweight Directory Access Protocol (LDAP) security model.

Note For more information about LDAP, see *Using the LDAP Security Model on page 80*.

Studio lets you restrict access to individual library resources within a given type. To expand on the first example, suppose you have graphics (LOG files) in your library that represent signatures. You might secure those resources by restricting them to a particular user or group of users when using the LDAP model. That way, even though other users could access other LOG files, only specific individuals would have the ability to modify the LOG files that contain signatures.

Note If you are not using the LDAP model, an ID represents a single user. In the LDAP model, the ID could represent a single user or a group of users.

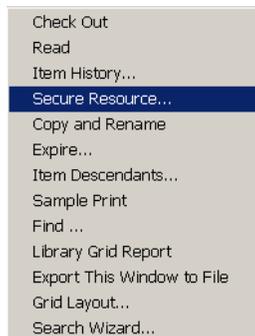
Here is another example within the LDAP security model. Suppose you have two groups of developers who share the same library. You want them to share the library because all of the business runs from a central location. You may, however, want to make sure one group does not accidentally check out or change forms that belong to the other group.

For instance, group one might be distinguished by the line of business they work on, such as Life and Property. To meet this need, you could secure the appropriate resources using an ID that represents the *group* to which individual users belong. Group affiliation is a feature of LDAP security Studio can use.

So, when a user logs into a system that belongs to specific group, that user can check out the resources for that group. But if the user does not belong to the group identified on the secured resource, the user cannot check out the resource.

To secure resources, open Library manager and highlight the resources you want to secure. Then right click and choose the Secure Resource option.

After you highlight the resources you want to secure, right click and choose this option.



The Secure Library Resource window appears:

Choose the user ID to assign to the selected resources.

Use the Release button to remove the security setting.



Choose the user ID you want to assign to the selected resources.

USING THE LDAP SECURITY MODEL

Documaker Studio works with LDAP (Lightweight Directory Access Protocol) to provide group-level security. Studio supports any LDAP-compliant Directory Information Tree, such as Active Directory.

When using LDAP security, Studio automatically queries the server for a list of groups to which the user belongs. Studio then looks up each of those group names in its USERINFO database.

For each group name that exists in the USERINFO database, Studio merges the security rights of that group into the current user. This way, you can belong to multiple groups and will end up with combined security options — so you get the most security rights possible.

Note Any group name returned that does not exist in the USERINFO database is ignored.

If no matching group for a given user is found in the USERINFO database, that user will not be allowed into the workspace. This differs from the standard user model where an unknown user is permitted access by copying the USER1 security rights, which are presumably set with the lowest possible settings. Using the LDAP model, however, unknown users are not allowed into the workspace.

For instance, suppose when using LDAP that JOE is a member of these groups:

- USERS
- FORMDESIGN
- ADMIN

Further suppose the following LDAP group *identities* were defined in the USERINFO database. Next to each name is listed some security attributes assigned.

Identity	Description
FORMDESIGN	Normal access to FOR, FAP, and other files, but not a library administrator, or system administrator.
ADMIN	A library administrator, but not a system administrator
SYSADMIN	A system administrator

When JOE enters the workspace, Studio queries his groups from LDAP and gets USERS, FORMDESIGN, and ADMIN. USERS does nothing for him since it is not defined in the USERINFO database. After finding FORMDESIGN and ADMIN, JOE is accorded the rights of a library administrator and normal file access to the rest of the system. He is not, however, recognized as a system administrator and that might have other restrictions outside of library management — depending on the other settings applied.

Suppose tomorrow, JOE's job changes and you assign him to the SYSADMIN group, but remove him from the FORMDESIGN group. Now, when JOE logs in, he is assigned as a SYSADMIN, which gives him the ability to do anything in the system with no restrictions.

But if you remove some groups from JOE and only leave him in USERS, then he would not be able to enter the workspace any longer because USERS is not a known user within Studio's USERINFO database.

Setting up LDAP

To enable LDAP support, set the LDAP_Enabled option to Yes:

```
< Environment >
  LDAP_Enabled = Yes
```

When you do this, Studio ignores its normal logon procedures and verifies users via LDAP. The LDAP interface requires the following INI options:

Option	Description
LDAP.HOST	(Optional) The host name or IP address of the LDAP server. The default is localhost.
LDAP.PORT	(Optional) The port in which the LDAP server is listening on. The default is 389 when SSL is not used, 636 otherwise (see the LDAP.USE.SSL option).
LDAP.URL	(Optional) The URL the LDAP server is listening on. If a value is specified for this option, it overrides the values specified for LDAP.HOST and LDAP.PORT.
LDAP.UID	(Optional) The user ID for logging onto the LDAP server. If this value is provided and LDAP.USER is not provided, the user ID is derived from this value and the value provided for LDAP.DOMAIN option. Here is an example: <code>Administrator@Oracle.com</code>
LDAP.USER	(optional) An explicit value to use for the user ID for the purpose of login into the LDAP server. Define this option to override the behavior used to determine the user ID when LDAP.UID and LDAP.DOMAIN are defined — see LDAP.DOMAIN.
LDAP.PWD	(Optional) The password used to login into the LDAP server.
LDAP.AUTHENTICATION.MODE	(Optional) The method of authentication used to login into the LDAP server. You can choose from these options: <ul style="list-style-type: none"> • (simple) which provides clear-text password authentication • (none) which provides anonymous authentication. The default is (simple).
LDAP.TIMEOUT	(Optional) The amount of time (in milliseconds) after which a connection attempt or query should expire. The default is 10000 (10 seconds).
LDAP.SEARCH.BASE	(Optional) The base of the search in the DIT (Directory Information Tree). This is the starting point (node location) of a search in the DIT. If you omit this option, the system looks for the LDAP.DOMAIN option and builds a search base from it.

Option	Description
LDAP.DOMAIN	<p>(Optional) This is the domain of the LDAP server. It is used to build the user ID for login into the LDAP server by appending the at symbol (@) plus the value of this option to the LDAP.UID value.</p> <p>The value of LDAP.DOMAIN is further parsed into domain components which are used as the default value for LDAP.SEARCH.BASE, if not already defined.</p>
LDAP.OBJECTS	<p>(Optional) A semicolon-delimited filter list of object classes to search in the LDAP server. If defined, it overrides the default filter list of object classes to search: group and groupOfNames.</p>
LDAP.OBJECTS.SEARCH.STRING	<p>(Optional) An explicit string value used as the filter of object classes to search. If defined, it overrides any value provided for LDAP.OBJECTS option.</p> <p>The value provided for this option must be specified in the appropriate LDAP protocol filter format. Also, if the search filter contains a question mark (?), the system replaces it with the user ID passed in as an argument to this function.</p> <p>Here are some examples:</p> <pre>((objectClass=group) (objectClass=groupOfNames)) . Cn=?</pre>
LDAP.OBJECT.ATTRIBUTES	<p>(Optional) The name of the attributes to retrieve for each object class which contain a value used to determine a match for USERID specified. The default values are <i>member</i> and <i>cn</i> (<i>cn</i> is always included).</p>
LDAP.MATCH.ATTRIBUTES	<p>(Optional) The name of one or more attributes contained within the value returned by a search for the LDAP.OBJECT.ATTRIBUTES option. This is the name of an attribute whose value is used to compare as opposed to the USERID specified to determine a match.</p> <p>For example, if LDAP.OBJECTS contains a value of <i>groupOfUniqueNames</i> and LDAP.OBJECT.ATTRIBUTES contains a value of <i>uniqueMember</i> and the value returned for the uniqueMember attribute of groupOfUniqueNames object class is:</p> <pre>uid=admin,ou=people,dc=mycompany,dc=com</pre> <p>and you want to match the USERID value with the value for uid, you would supply a value of uid for this option.</p> <p>The default is cn.</p>
LDAP.SEARCH.SCOPE	<p>(Optional) The scope of the search. You can choose from:</p> <ul style="list-style-type: none"> • (base) - search only the named context • (one) - search one level below the named context but not the named context • (sub) - search the entire subtree, including the named context. <p>The default is (sub).</p>
LDAP.DEREF.LINK	<p>(Optional) Enter Yes or No to indicate whether or not to remove reference links to other nodes during a search. The default is No.</p>
LDAP.VERSION	<p>(Optional) An integer value that indicates the LDAP protocol version to use. You can choose from:</p> <ul style="list-style-type: none"> • 2 - Version 2 • 3 - Version 3 <p>The default is three (3).</p>

Option	Description
LDAP.SEARCH.LEVEL	<p>(Optional) An integer value that indicates the search level. You can choose from:</p> <ul style="list-style-type: none"> • 1 - User type objects • 2 - Group type objects • 3 - Any objects <p>The default is one (1), user type objects.</p>
LDAP.DN.IDENTIFIER	<p>(Optional) The value for this option is used these ways:</p> <ul style="list-style-type: none"> • Where LDAP.SEARCH.LEVEL is equal to 1 (USER) and there is no LDAP.OBJECTS.SEARCH.STRING value specified, the system generates a default search filter of the format identifier=UserID, where identifier is the value of this option and UserID is the user ID passed in to this function. • Where LDAP.SEARCH.LEVEL is equal to 2 (GROUPS) and there is no LDAP.OBJECTS.SEARCH.STRING value specified, the system generates a default search filter from LDAP.OBJECTS and LDAP.OBJECT.ATTRIBUTES, where each attribute value in the search filter is an asterisk (*), which tells the system to match any value for the attributes specified. If the LDAP.RDNDS option is also provided, the asterisk (*) is replaced with identifier=UserID, followed by a comma and the LDAP.RDNDS value to fine tune the search, where identifier is the value for this option and UserID is the user ID passed in to this function. Here is an example of a default search filter: <pre>(& ((objectClass=groupOfNames) (member=*)))</pre> <p>If a value of <pre>CN=Users,DC=PDDC,DC=DOCUCORP,DC=COM</pre> is specified for LDAP.RDNDS and this option contains a value of <i>cn</i>, the search filter generated would look like this: <pre>(& ((objectClass=groupOfNames) (member=CN=Administrator, CN=Users,DC=PDDC,DC=DOCUCORP,DC=COM))) .</pre> The default is <i>cn</i>.</p>
LDAP.RDNDS	<p>(Optional) This option is only used when LDAP.SEARCH.LEVEL is equal to 2 (GROUPS) and when LDAP.OBJECTS.SEARCH.STRING is not specified. In this situation, the system builds a default search filter from LDAP.OBJECTS and LDAP.OBJECT.ATTRIBUTES.</p> <p>Attribute values specified in the default search filter contain an asterisk (*), which tells the system to match any value for the attributes specified.</p> <p>When you specify this option, the system uses the value along with the value for LDAP.DN.IDENTIFIER to replace the asterisk and fine tune the search, thereby speeding the process. Here is an example of a default search filter: <pre>(& ((objectClass=groupOfNames) (member=*)))</pre> <p>In a case were a value of <pre>CN=Users,DC=PDDC,DC=DOCUCORP,DC=COM</pre> is specified for this option and LDAP.DN.IDENTIFIER contains a value of <i>CN</i>, the search filter generated would look like this: <pre>(& ((objectClass=groupOfNames) (member=CN=Administrator, CN=Users,DC=PDDC,DC=DOCUCORP,DC=COM)))</pre> </p> </p>

Option	Description
LDAP.USE.SSL	(Optional) Enter Yes to enable encrypted communication through an SSL channel. For SSL connections to work, the LDAP server must be configured for SSL with a certificate from a trusted certification authority. This configuration is vendor specific — please consult your vendor documentation.
LDAP.DEBUG	(Optional) Enter Yes to log debugging information to a trace file.

You can specify the LDAP options in a properties file named `openldap.msg` or in the LDAP control group in the INI file for the configuration being used. If you have defined these options in both, the options defined in the LDAP control group override the ones defined in the properties file.

Here is an example of a properties file:

```
ldap.host=10.1.10.101
ldap.port=389
ldap.timeout=10000
ldap.uid=jroberts
ldap.pwd=ElCamino
ldap.authentication.mode=simple
ldap.domain=PDDC.pd.com
ldap.objects.search.string=cn=?
ldap.object.attributes=memberOf
ldap.match.attributes=cn*
ldap.debug=yes
ldap.dn.identifier=cn
```

In this example, the `ldap.objects.search.string` is used to query an object in Active directory that corresponds to the user ID supplied by Documaker Studio. The question mark (?) is replaced at run time by the user ID supplied. Once the object is found, all values for the `memberOf` attribute are retrieved and returned as a list of the groups to which the user belongs.

Keep in mind:

- The `ldap.uid` or `ldap.user` and `ldap.pwd` options should correspond to an account that has permission to connect to the Active Directory and to query the objects in the Active Directory Tree. This account could be the same as that of the user ID that will be queried for group membership. If this is the case, that account must have access rights to the Active Directory and permission to query its own object.
- If the same Studio workstation is used by more than one person, the `ldap.uid` or `ldap.user` account should be one that has permission to query all objects corresponding to all users who will use Studio on that workstation. This is also true for other LDAP servers besides Active Directory, which do not support the `memberOf` attribute. In this situation, use the `ldap.user` option instead of `ldap.uid` and the account used must have permission to query all the objects that define group information (see the OpenLDAP example).

When using LDAP, Studio does not keep records of individual users in the user database. Instead, you must store specific group identities in the user database. Studio queries LDAP for a list of groups the user belongs too. It then looks up each group name in the user database.

If the group name returned does not exist in the database, Studio does nothing for the user. If the group name does exist in the database, then the security rights from that group (user) are merged into the current user. If a user belongs to multiple groups defined in the database, the user ends up with combined security options.

On the other hand, if the user does not have any groups in common with the user database, the user is not allowed to log onto the system.

Unlike the normal (non-LDAP) user situation, Studio does not support a guest or default log on when LDAP is enabled.

Here is an example of how to search a user object in an Active Directory server schema that supports the `memberOf` attribute:

Here is the Distinguished Name of the user *Administrator* in an Active Directory Schema:

```
CN=Administrator,CN=Users,DC=PDDC,DC=pd,DC=com
```

Here are the INI options:

```
< LDAP >
  ldap.host           = localhost
  ldap.port          = 389
  ldap.timeout       = 10000
  ldap.uid           = userID@PDDC.pd.com
  ldap.pwd           = 123456xxx
  ldap.objects.search.string = cn=?
  ldap.authentication.mode = simple
  ldap.domain        = PDDC.pd.com
  ldap.dn.identifier = cn
  ldap.debug         = Yes
```

In this example a search is conducted for a user named *Administrator* and all the `memberOf` attribute values are retrieved for it, composing a list of the groups *Administrator* belongs to.

Here is an example of how to search group objects in an OpenLDAP server schema that does not support the `memberOf` attribute:

Here is the schema for the OpenLDAP server:

```
dn: dc=mycompany,dc=com
objectClass: top
objectClass: dcObject
objectClass: domain
dc: mycompany

dn: ou=roles,dc=mycompany,dc=com
objectClass: top
objectClass: organizationalUnit
ou: roles

dn: ou=people,dc=mycompany,dc=com
objectClass: top
objectClass: organizationalUnit
ou: people

dn: cn=Test Users,ou=roles,dc=mycompany,dc=com
objectClass: groupOfUniqueNames
cn: Test Users
uniqueMember: uid=sspecial,ou=people,dc=mycompany,dc=com
uniqueMember: uid=jbloggs,ou=people,dc=mycompany,dc=com

dn: cn=Special Users,ou=roles,dc=mycompany,dc=com
```

```
objectClass: groupOfUniqueNames
cn: Special Users
uniqueMember: uid=sspecial,ou=people,dc=mycompany,dc=com
```

```
dn: cn=Admin Users,ou=roles,dc=mycompany,dc=com
objectClass: groupOfUniqueNames
cn: Admin Users
uniqueMember: uid=admin,ou=people,dc=mycompany,dc=com
```

```
dn: uid=admin,ou=people,dc=mycompany,dc=com
objectClass: person
objectClass: inetOrgPerson
cn: State App
displayName: App Admin
givenName: App
mail: admin@fake.org
sn: Admin
uid: admin
userPassword: adminpassword
```

```
dn: uid=jbloggs,ou=people,dc=mycompany,dc=com
objectClass: person
objectClass: inetOrgPerson
cn: Joe Bloggs
displayName: Joe Bloggs
givenName: Joe
mail: jbloggs@fake.org
sn: Bloggs
uid: jbloggs
userPassword: password
```

```
dn: uid=sspecial,ou=people,dc=mycompany,dc=com
objectClass: person
objectClass: inetOrgPerson
cn: Super Special
displayName: Super Special
givenName: Super
mail: sspecial@fake.org
sn: Special
uid: sspecial
userPassword: password
```

Here are the INI options:

```
< LDAP >
ldap.host           = localhost
ldap.port           = 389
ldap.timeout        = 5000
ldap.user           = uid=admin,ou=people,dc=mycompany,dc=com
ldap.pwd            = adminpassword
ldap.authentication.mode = simple
ldap.objects        = groupOfUniqueNames
ldap.search.base    = ou=roles,dc=mycompany,dc=com
ldap.object.attributes = uniqueMember
ldap.match.attributes = uid
ldap.search.scope   = sub
ldap.search.level   = 2
ldap.dereference.link = Yes
ldap.version        = 3
ldap.debug          = Yes
UserID              = admin
```

In this example a search is conducted for all objects of type *groupOfUniqueNames* which contain attributes of name *uniqueMember*. All attribute values returned are then filtered matching the uid *Distinguished Name* part to the user ID supplied in the search. For example if the user ID searched was *admin*, then the list of groups returned would be *Admin Users*.

Chapter 3

Working with Application Definition Files

Choose Manage, Application, Definition to maintain your lines of business. For example, suppose you are creating forms for an insurance company which underwrites automobile and homeowner's insurance. Each different type of insurance could be considered a separate line of business.

This chapter discusses the following topics:

- *Overview on page 90*
- *Defining an Application Definition on page 91*
- *Defining Form Lists on page 94*
- *Defining Recipients on page 97*
- *Defining Categories on page 99*
- *Defining Transaction Types on page 101*
- *Using Regional Date Processing on page 103*
- *Generating an Application Definition Report on page 112*

OVERVIEW

The application definition (BDF) file defines the key combinations used to locate a specific form set. These key combinations are comprised of a Key1 and Key2 (sometimes referred to as Unit1 and Unit2; or Group1 and Group2). In the insurance world, these keys are typically called: *company* and *line of business* (LOB).

Other information stored in the BDF file includes the following:

- List of recipients
- Form categories (if used)
- Transaction type information
- Primary extract dictionary (XDD) file (if used)
- Default font cross-reference (FXR) file
- Default style (STY) file

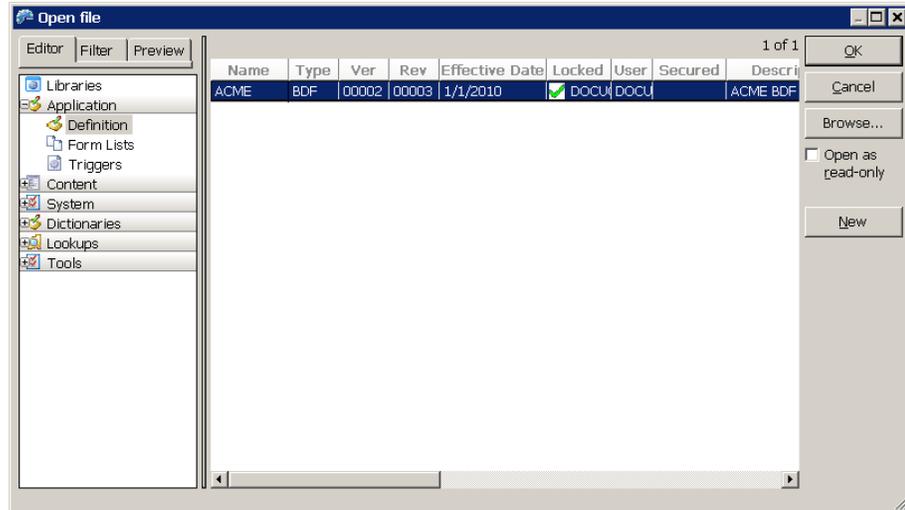
Here is an example of a BDF file:

```
<DOCUMENT TYPE="RPWIP" VERSION="11.4">
<DOCSET NAME="ACME">
  <FXRFILE NAME="rel103sm.fxr"/>
  <XDDFILE NAME="Symbol"/>
  <STYFILE NAME="PropNCas"/>
  <RECIPIENT NAME="BROKER" COPYCOUNT="0" CODE="003" SEQUENCE="3">
    <DESCRIPTION>Broker Copy</DESCRIPTION>
  </RECIPIENT>
  <RECIPIENT NAME="HOMEOFF" COPYCOUNT="0" CODE="002" SEQUENCE="2">
    <DESCRIPTION>Home Office Copy</DESCRIPTION>
  </RECIPIENT>
  <RECIPIENT NAME="ORIGINAL" COPYCOUNT="0" CODE="001" SEQUENCE="1">
    <DESCRIPTION>Original</DESCRIPTION>
  </RECIPIENT>
  <GROUP NAME="ACME_ACCIDENT" NAME1="ACME" NAME2="ACCIDENT"/>
  <CATEGORY NAME="GA" DESCRIPTION="Georgia"/>
  <CATEGORY NAME="MS" DESCRIPTION="Mississippi"/>
  <TRANSACTION TRNCODE="NB" DESCRIPTION="New Business" DLLFUNC="TRNW32->;TRNNew"/>
  <TRANSACTION TRNCODE="EN" DESCRIPTION="Endorsement" DLLFUNC="TRNW32->;TRNEndorse"/>
  <TRANSACTION TRNCODE="RN" DESCRIPTION="Renewal" DLLFUNC="TRNW32->;TRNRenew"/>
  <TRANSACTION TRNCODE="QU" DESCRIPTION="Quote" DLLFUNC="TRNW32->;TRNRenew"/>
</DOCSET>
</DOCUMENT>
```

Note This example is included for illustrative purposes. You should always let Studio create and maintain BDF files. *Do not* manually edit this file.

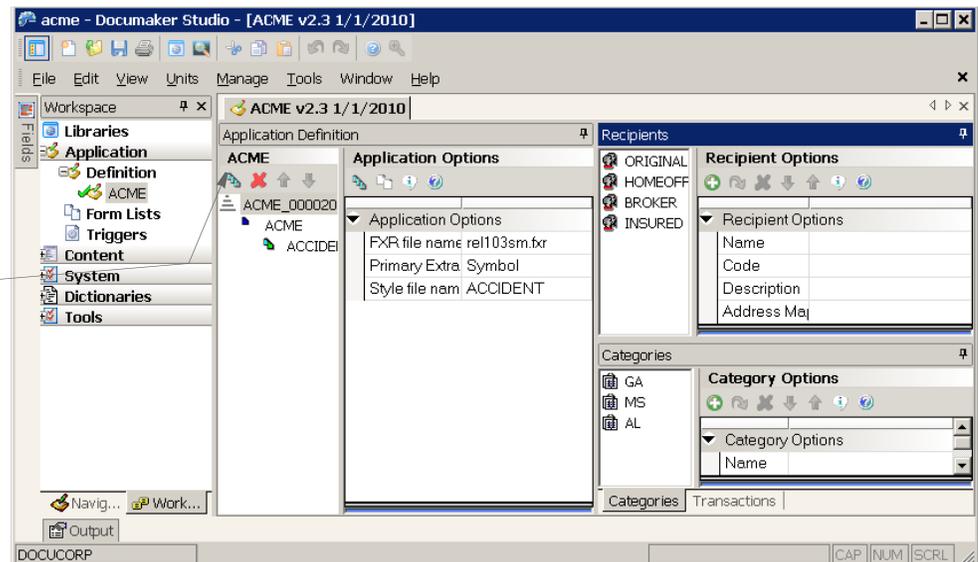
DEFINING AN APPLICATION DEFINITION

To define a line of business for a set of resources, you first check out the BDF file and then make the necessary entries to define a new line of business. Here is an example of the window you use to select a BDF file from the library:



Once you select the one you want, the Application Definition view appears:

For Acme, one line of business already exists, Accident.

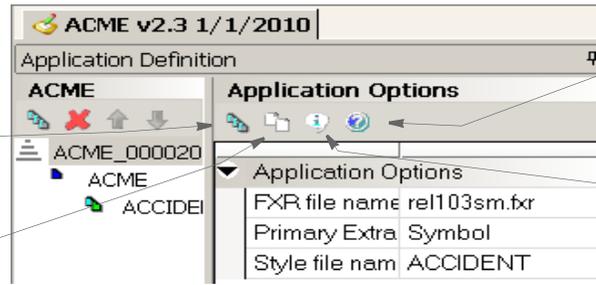


Notice the toolbar above the application options. You will see similar toolbars throughout Studio.

Here you can add an application definition, look up a BDF file, or toggle the information pane.

Click here to add a definition.

Click here to look up a definition.



Click here to display Help.

Click here to display or hide the information pane.

The toolbar provides these options:

To	Click
Add an item.	
Look up an item. When you click this icon, the Open File window appears. You can choose the item you want from the list or browse to another location.	
Display or hide the information pane.	
Display Help.	

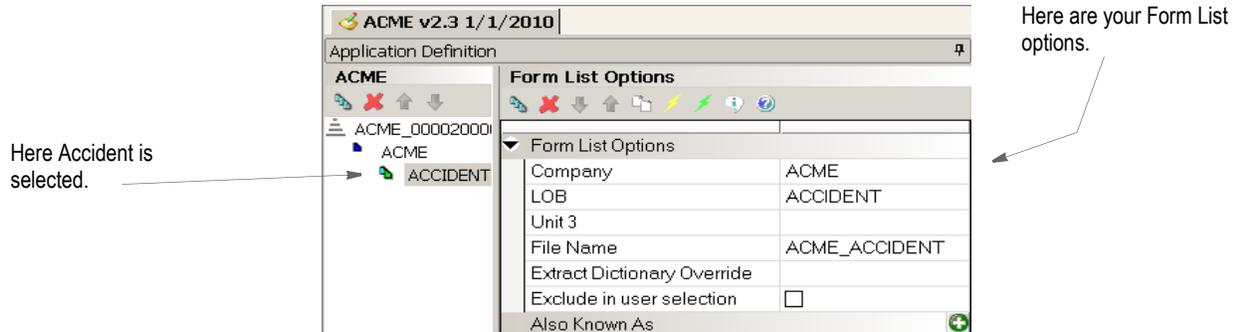
For each application definition, you must define a font cross-reference (FXR) file and the primary extract dictionary (XDD) file. You can also associate a style sheet (STY) with a line of business so other users will have access to those styles when they create sections.

Field	Description
FXR File Name	<p>Enter the name of the font cross-reference file you want to use with this business unit definition. The font cross-reference file lets you organize the fonts you use for display and printing.</p> <p>The FXR provides all necessary font information. It does not contain the actual font files; rather, it contains information about the font attributes. Font attribute information includes formatting styles (bold, italic, and so on), point size (10 point, 14 point, and so on), and font stroke eight (heavy, light, and so on).</p> <p>When you click in this field, a browse icon appears which you can use to browse to the FXR file you want to select.</p> <p>Note: If an FXR file is specified in your INI files, Studio uses that file instead of the one you name in this field.</p>
Primary Extract Dictionary (XDD)	<p>Enter the name of the XDD file you want to use with this business unit definition.</p> <p>The XDD file contains information on how to transfer the data from external files (sometimes called extract files) into fields defined within your documents. If your setup is not a batch implementation, you would not normally need an XDD file.</p> <p>When you click in this field, a browse icon appears which you can use to browse to the XDD file you want to select.</p>

Field	Description
Style File Name	Enter the name of the style (STY) file you want to associate with this line of business. Style files let you specify default settings for text labels, text areas, boxes, shade objects, vectors, lines, bar codes, graphics, signatures, bookmarks, notes, indexes, and charts and associate those settings with named styles. You can click the Browse icon to search for the file you want.

DEFINING FORM LISTS

If you click on a specific LOB (line of business), the view changes to show you the appropriate fields for defining a form list for that line of business. Here is an example:



The toolbar provides these options:

To	Click
Add an item.	
Delete an item. Please note there is no confirmation message. As soon as you click this icon, the item is deleted.	
Change the order of an item. Press the up arrow to move the item higher in the tree. Press the down arrow to move the item lower in the tree. The order of the items defined within the BDF file is the same order used by the entry programs, such as Documaker Workstation and iDocumaker. For batch processing, the order of the items at the BDF level is not usually important, but you may experience a minor improvement in performance if the items used most often appear at the top of the tree.	
Look up an item. When you click this icon, the Open File window appears. You can choose the item you want from the list or browse to another location.	
Display or hide the information pane.	
Add a trigger.	
Insert a manual trigger.	
Display Help.	

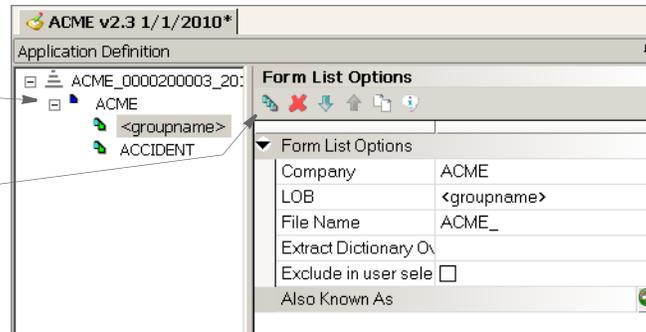
ADDING A FORM LIST

*** Start here

To add a form list, select the company and then click the Add icon.

First, select the company.

Click here to add a form list for that company.



Then enter the appropriate information in these fields:

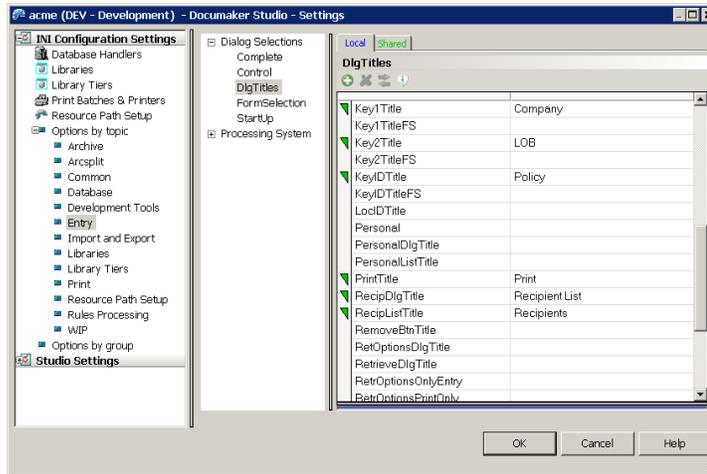
Field	Description
Company	Here you enter the Key1 value you want to use. Studio defaults this based on information you have already entered when you add a line of business.
LOB	Here you enter the Key2 value you want to use. Studio defaults this based on name you entered when you added the line of business.
File Name	Here you define the library file name that contains the available forms list for this Company/LOB combination. This file is referred to as a Forms List (GRP) file. You can enter a name or select the GRP file from the library by clicking on the associated button. Remember that certain platforms can have file naming requirements that are not automatically enforced by Studio. You must make sure the file names you assign are compatible with the platform you will eventually process on.
Extract Dictionary Override	Enter the name of the XDD file you want to use for this line of business instead of the primary XDD file defined at the BDF or root level. Leave this field blank if you want to use the primary XDD file. You can browse through the library to select an XDD file. During batch processing, when the system maps fields it check the associated group level for a specifically defined XDD file. If omitted, the system looks in the BDF file for the name of the primary XDD file. If no primary XDD file is defined, the system uses the value assigned to this INI option: <pre>< MasterResource > DictionaryFile =</pre>
Exclude in User Selection	Check this field to exclude this line of business and the forms associated with it from Documaker Workstation's Form Selection window. This can reduce the clutter of having too many forms to choose from and it can lessen the chance users will forget to include some if otherwise triggered. For example, you might want to exclude a cover letter, list of customer instructions, or addresses from the Forms Selection window yet still include them in the form set by triggering them. A form excluded from the Forms Selection window can still be included in the form set for users to see and work with by, for example, marking the form Pre-selected in Studio's Form List area. Note that if you exclude from user selection all the Key2 children under a parent Key1, the Key1 is also excluded from the Forms Selection window.
Also Known As	Click in this field to set up the automatic mapping of group alias names into the Key1Table and Key2Table control groups. These groups are used to map a short name in your extract file to a long name in your BDF file. Studio lets you identify aliases that, when encountered in a batch operation, should map to the standard group name. When the system starts a batch run, it loads the BDF file and automatically assigns the defined alias names for each group to the appropriate INI option.

Note You can use the DlgTitles control group to change the text that appears on the various windows. This includes field labels, window titles, column headings and button captions. You can use these options to change the text for the Key1 and Key2 fields:

```
< DlgTitles >
  Key1Title = Company
  Key2Title = LOB
```

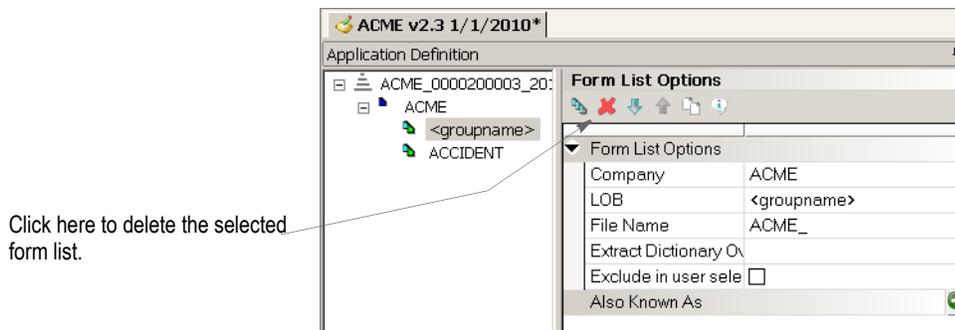
If you leave these options blank, the text will be blank. If you omit these options, the system uses its defaults.

To change these options, choose Manage, System, Settings. Then, under Options by Topic, choose Entry and then Dialog Selections, as shown here:



DELETING A FORM LIST

If you need to delete a form list, select the form list you want to delete and click the Delete Form List icon, as shown here:

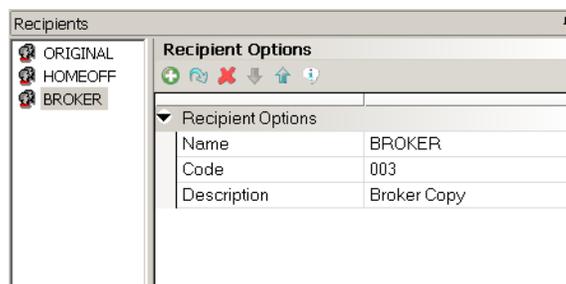


Studio deletes the form list immediately.

DEFINING RECIPIENTS

Use the Recipient Options panel to set up recipients for this application definition. Recipients are those who will get copies of the various forms in the form set. Defining a recipient includes setting up the following:

- A recipient type name, such as *Insured* or *Broker*.
- A sorting code. This code is used in batch implementations. Documaker's GenData program uses the code to know which recipient to queue at print time.
- A brief description. This description appears in any Send Copy To fields defined in your form when you print that form for that recipient. For instance, this is where the text *Insured's Copy* comes from when you print forms. This description is used by both entry systems, like Documaker Workstation or iDocumaker, and the GenPrint program.



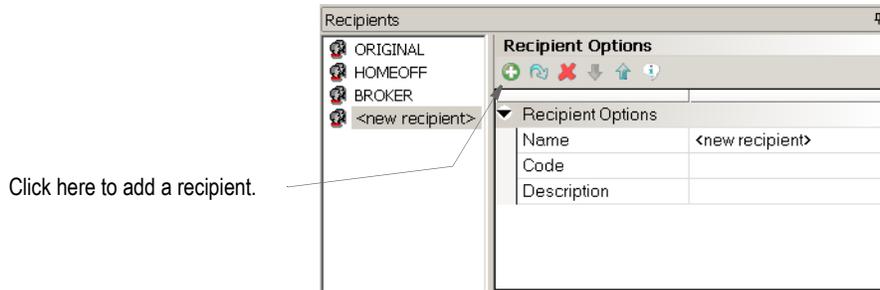
The toolbar provides these options:

To	Click
Add an item.	
Clear the information for an item. Choose this option to delete all of the information entered in the various fields used to define this item.	
Delete an item. Please note there is no confirmation message. As soon as you click this icon, the item is deleted.	
Change the order of an item. Press the up arrow to move the item higher in the tree. Press the down arrow to move the item lower in the tree. The order of the items defined within the BDF file is the same order used by the entry programs. For batch processing, the order of the groups at the BDF level is not usually important, but you may experience a minor improvement in performance if the items used most often appear at the top of the tree.	
Display or hide the information pane.	
Display Help.	

Note When a batch processing cycle starts, the list of recipients defined in the Business Definition file is inserted into the Recip_Names control group. This control group is used by several of the Documaker programs.

ADDING A RECIPIENT

If you need to add a recipient, select the Recipients pane and click the Add Recipient icon, as shown here:

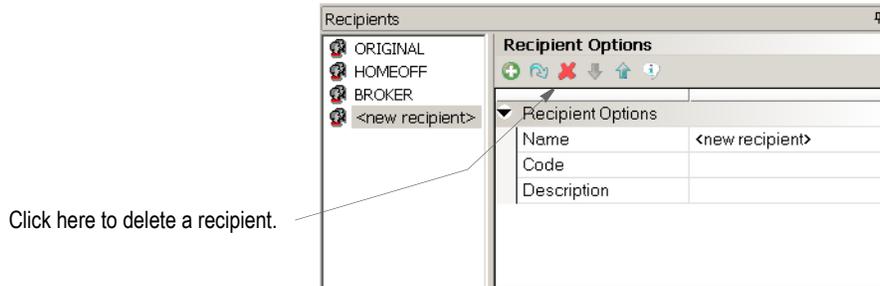


Then enter the appropriate information in these fields:

Field	Description
Name	Enter the name of this recipient you want to add. For example, you might enter Home Office Insured, or Broker.
Code	Use this field to assign a code to this recipient. This code is used by the GenData program.
Description	Enter a description of this recipient. For instance, you could enter <i>Insured's Copy</i> or <i>Broker's Copy</i> .

DELETING A RECIPIENT

If you need to delete a recipient, select the recipient you want to delete and click the Delete Recipient icon, as shown here:



Studio deletes the recipient immediately.

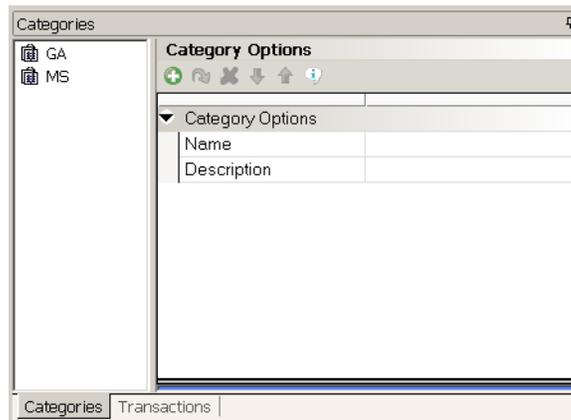
DEFINING CATEGORIES

Categories can help you organize large sets of forms by letting you associate similar forms to make the list more manageable. For instance, you could set up geographical or market-based categories that indicate where certain forms are used.

Setting up categories is optional. If you define them, the form categories appear when you work with group (GRP) files.

Note Categories are not the same as the higher level key groups (Key1 and Key2) and are not used during form triggering or selection.

Click the Categories tab to work with categories.

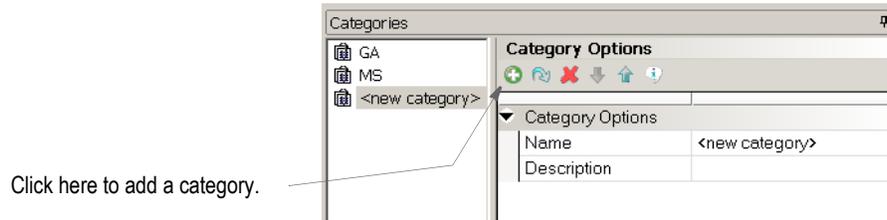


The toolbar provides these options:

To	Click
Add an item.	
Clear the information for an item. Choose this option to delete all of the information entered in the various fields used to define this item.	
Delete an item. Please note there is no confirmation message. As soon as you click this icon, the item is deleted.	
Change the order of an item. Press the up arrow to move the item higher in the tree. Press the down arrow to move the item lower in the tree. The order of the items defined within the BDF file is the same order used when a user selects forms in via the Entry program. For batch processing, the order of the groups at the BDF level is not usually important, but you may experience a minor improvement in performance if the items used most often appear at the top of the tree.	
Display or hide the information pane.	
Display Help.	

ADDING A CATEGORY

If you need to add a category, click the Add Category icon, as shown here:

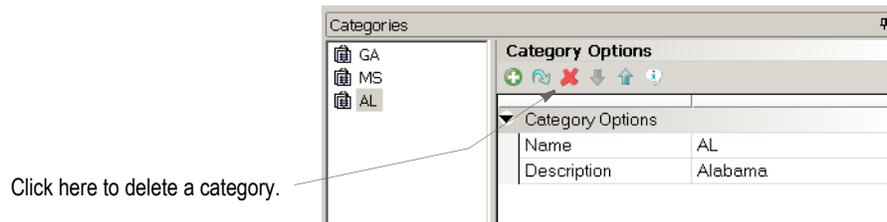


Then enter the appropriate information in these fields:

Field	Description
Name	Enter the name of the category you want to add.
Description	Enter a description of the category you are adding.

DELETING A CATEGORY

If you need to delete a category, select the category you want to delete and click the Delete Category icon, as shown here:



Studio deletes the category immediately.

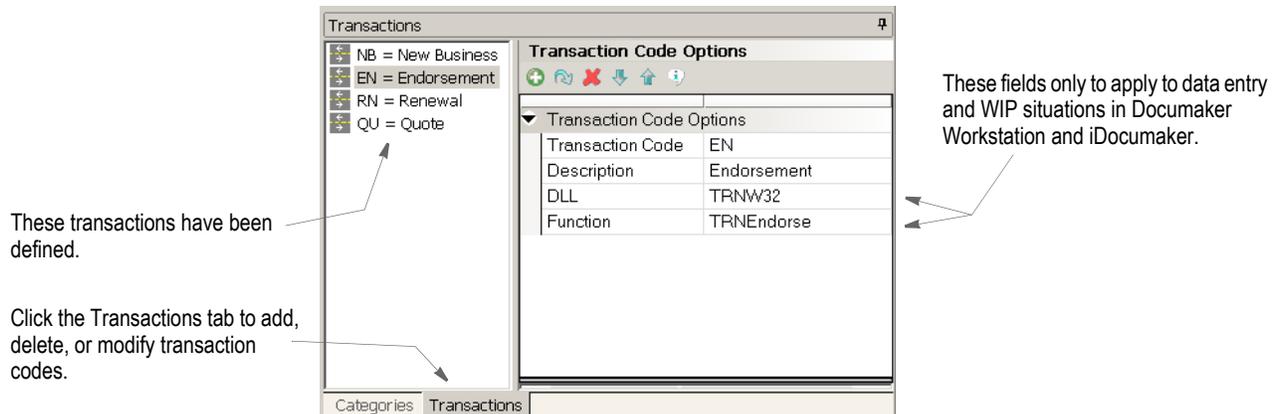
DEFINING TRANSACTION TYPES

Click the Transactions tab to assign codes to the various types of transactions, such as new business, renewals, or quotes.

In batch processing, you can use the transaction codes you define as one level of form filtering. In data entry and WIP situations, transaction codes can identify the type of transaction being created.

Also for Documaker Workstation and iDocumaker, you can use the DLL and Function fields to specify internal or external hook functionality you want the system to execute on that transaction code. For example, you could specify a function that validates the list of forms that were selected.

Note Batch implementations do not use the hook function specified via the DLL and Function fields.

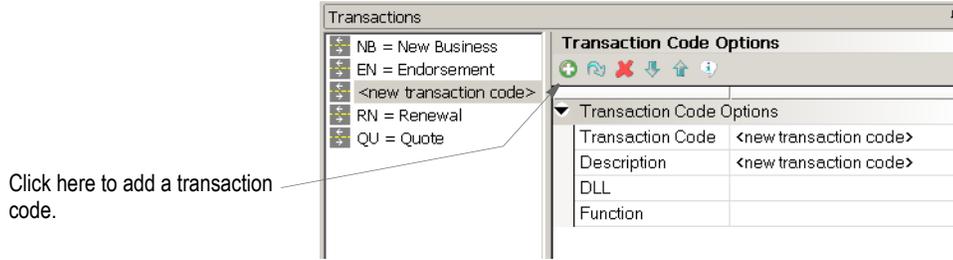


The toolbar provides these options:

To	Click
Add an item.	
Clear the information for an item. Choose this option to delete all of the information entered in the various fields used to define this item.	
Delete an item. Please note there is no confirmation message. As soon as you click this icon, the item is deleted.	
Change the order of an item. Press the up arrow to move the item higher in the tree. Press the down arrow to move the item lower in the tree. The order of the items defined within the BDF file is the same order used when a user selects forms in via the Entry program. For batch processing, the order of the groups at the BDF level is not usually important, but you may experience a minor improvement in performance if the items used most often appear at the top of the tree.	
Display or hide the information pane.	
Display Help.	

ADDING A TRANSACTION CODE

If you need to add a transaction code, click the Add Transaction icon, as shown here:

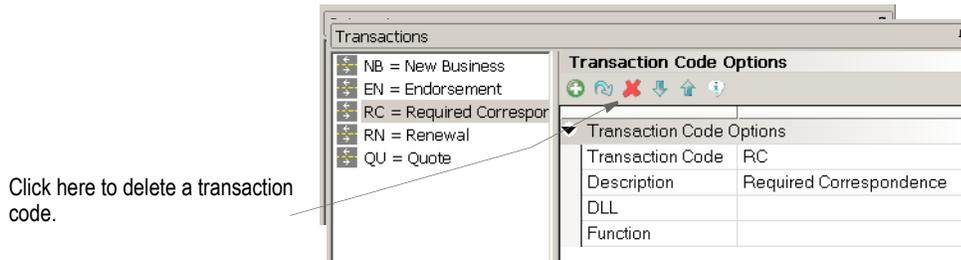


Then enter the appropriate information in these fields:

Field	Description
Transaction Code	Enter the transaction code you want to add.
Description	Enter a description of the transaction code.
DLL	Enter the name of the DLL you want the entry system to execute to validate the list of forms.
Function	Enter the name of the function you want the entry system to execute to validate the list of forms.

DELETING A TRANSACTION CODE

If you need to delete a transaction code, select the transaction code you want to delete and click the Delete Transaction icon, as shown here:



Studio deletes the transaction code immediately.

USING REGIONAL DATE PROCESSING

In the U.S. insurance industry, certain forms must comply with a regional authority (usually a state) to be approved for use within that area. The process of getting approval to use forms in each location is often referred to as *submitting for state compliance*.

Because of the various jurisdictions involved, you may have a form which is accepted by some states, but not by others. Alternatively, the form might be accepted by multiple states, but as of different dates. And to add another layer of complexity, states specify which document date must be used when activating this form.

To understand this last point, consider that almost all insurance policies have a date when coverage becomes *effective* – typically referred to as the *policy effective date*. Likewise, a policy usually has a *written* date that identifies when the document was actually drawn up. It is not unusual for the written date to be different from the policy effective date. For instance, you might buy your hurricane insurance today (the written date), but the policy does not become effective for 30 (or more) days. Each regional authority specifies which date determines the compliance of a given form.

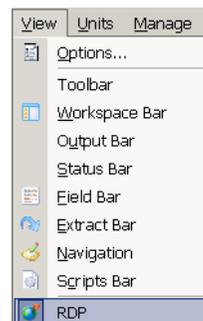
This necessity to only activate the use of a form in a given region after a specific date complicates the creation of trigger conditions. Not only do you have to consider the typical transaction information that would cause you to include the form, you also have to calculate the various details to comply with the regional authorities described above.

To help you more easily manage this process, Studio lets you define regional date processing (RDP) rules that you can assign to each form. Part of the support is accomplished in Studio by defining the appropriate regional tests for each form. The remaining part occurs during the batch transactional process via the `RegionalDateProcess` rule.

Note For more information on the `RegionalDateProcess` rule, see the [Rules Reference](#).

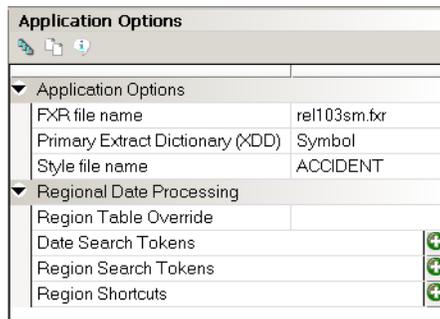
SETTING UP REGIONAL DATE PROCESSING

To begin using regional date processing (RDP) rules you must activate them for your application definition (BDF file). To activate, you first check out the appropriate application definition and then select the View, RDP option.



Once activated, new items are added to the Applications Options pane. These options let you change the default Region Table and set up Date Search Tokens, Region Search Tokens, and Region Shortcuts.

The fields under Regional Date Processing are added when you choose View, RPD



Use these Plus (+) buttons to define search tokens and shortcuts.

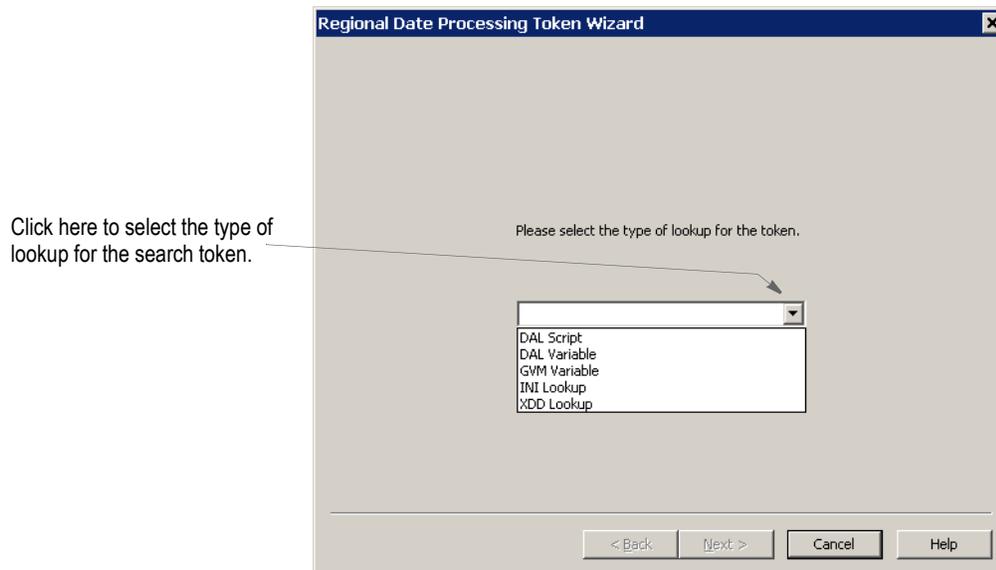
Field	Description
Region Table Override	<p>Studio includes a basic region table (RDPTABLE.TBL) that contains the fifty U.S. states and the District of Columbia. It is included to help you select regions when setting up RDP rules.</p> <p>If you need to use a different region table, click the Browse icon beside the Region Table Override field and select the region table you want to use.</p> <p>This table is not required for run-time processing.</p>
Date Search Tokens	<p>Use this field to identify the token names that represent the date searches you will use within an RDP rule.</p>
Region Search Tokens	<p>Use this field to list the tokens you want the user to use when creating an RDP rule.</p> <p>Click the Plus (+) button to add a region search token. To remove an entry, click on it and then use the right mouse button to access the popup menu to delete the item. When adding a new Region Search Token, you can use the Look Up button to select from your available fields defined in the Extract Dictionary (XDD) for your setup.</p>
Region Shortcuts	<p>These date and region search tokens are used during the associated batch rule to execute the search on the transaction data and return the appropriate values found.</p> <p>Use the Region Shortcuts to create macros or collections of regions identified with a single name. For instance, you might declare a shortcut named Atlantic and then choose the appropriate states that are identified by that location. Later, when defining an RDP rule, you can select a Region Shortcut token instead of individually selecting each state.</p> <p>The name of the shortcut and the regions included in it are entered in a similar window to the Search tokens defined previously. Clicking the Plus (+) button on this section title displays a window on which you can name your shortcut and select the regions it should contain.</p>

Note You can see the format of a region table by exploring the RDPTABLE.TBL file that Studio exports for you. There are two columns. The first column contains a descriptive name for the region — like *Alabama*. The second column identifies the data representation that is found when attempting to match with mapped data — like *AL*. You can edit the content of a region table using Studio's Batch Table manager.

Adding Tokens and Shortcuts

To add date or region search tokens, first click the Plus (+) button in the applicable field. The Regional Date Processing Token wizard appears so you can name the token and enter its search information.

Note You can also right click while in the Date Search Token, Region Search Token, or Region Shortcut field and choose the Add RDP option.



You can choose from these types of lookups:

Lookup	Description
DAL scripts	If you choose DAL scripts as the type of lookup, the wizard prompts you to enter the name of the DAL script. Click on the Browse button to choose a script from the Open File window. After you choose a script and click Next, the wizard lets you assign a name to the token.
DAL variables	If you choose DAL variables as the type of lookup, the wizard prompts you to enter the name of the variable. Once you enter this information and click Next, the wizard lets you assign a name to the token.
GVM variables	If you choose GVM variables as the type of lookup, the wizard prompts you to enter the name of the variable. Once you enter this information and click Next, the wizard lets you assign a name to the token.
INI lookups	If you choose INI lookup as the type of lookup, the wizard prompts you to enter the name of the INI control group, the INI option, and the option's default value. Once you enter this information and click Next, the wizard lets you assign a name to the token.
XDD lookups	If you choose XDD lookups as the type of lookup, the wizard prompts you to enter the name of the lookup. Click the Browse button to choose from the Retrieve from Extract Dictionary window. After you click OK, the wizard lets you assign a name to the token.

Note The date search tokens you list here become the selection choices available to the user who is creating the RDP rule.

To remove an entry, click on it and then use the right mouse button to display the popup menu and delete the item.

Region search tokens are entered the same way within their section. Here you list the available tokens you want the user to use when creating an RDP rule.

Region Search Tokens		+
STATE	?STATE/MASTER	
LOC_SLOC	?Location	
CONTROL_STATE	?Controlling_ST	

Click on the Plus (+) button to add a new region search token. To remove an entry, click on it and then use the right mouse button to access the popup menu to delete the item.

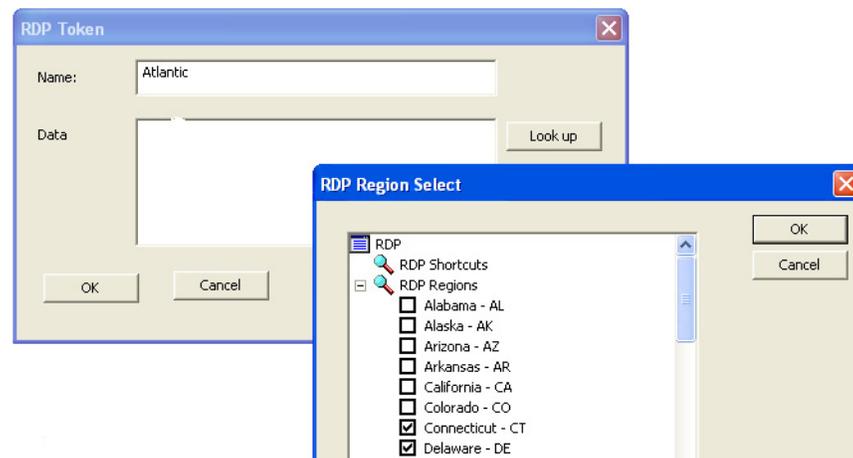
When adding a new Region Search Token, you can use the Look Up button to select from your available fields defined in the Extract Dictionary (XDD) for your setup.

These date and region search tokens are used during the associated batch rule to execute the search on the transaction data and return the appropriate values found.

Note The date in the extract file is in 4Y2M2D format, such as *19880217* for February 17, 1988.

Use the Region Shortcuts to create *macros* or collections of regions identified with a single name. For instance, you might declare a shortcut named *Atlantic* and then choose the appropriate states that are identified by that location. Later, when defining an RDP rule, you can select a Region Shortcut token instead of individually selecting each state.

The name of the shortcut and the regions included in it are entered in a similar window to the Search tokens defined previously. Clicking the Plus (+) button on this section title displays a window on which you can name your shortcut and select the regions it should contain.



Click the Look Up button to display a list of regions from the Region Table along with any other shortcuts you have already created. A region shortcut can contain other shortcuts and these are finally resolved at runtime when the RDP rules execute.

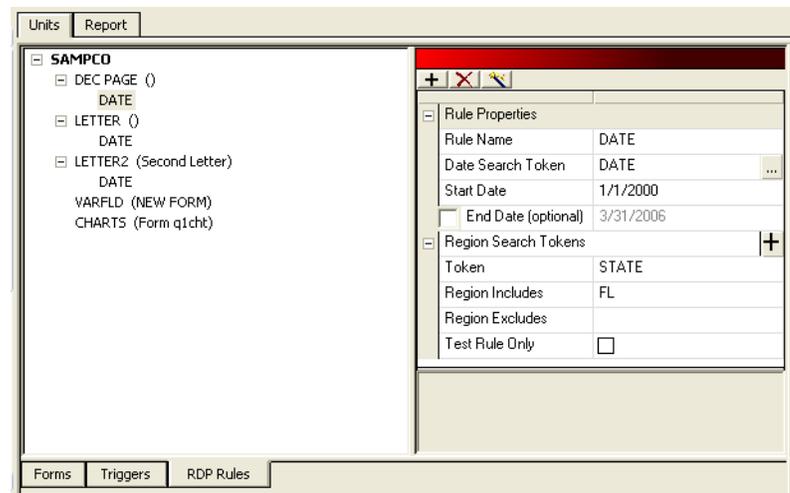
When using the Lookup selection window, simply check, or uncheck, the items you want to include or exclude from this Shortcut definition.

Once you have created all your necessary Date and Region Search tokens and optionally created any Regional Shortcut names you want to provide, save your BDF and check the file back into the library.

SETTING UP REGIONAL DATE PROCESSING RULES

Regional date processing (RDP) rules are defined at the form level using the Form Lists manager. Once you have added date and region search tokens in the RDP settings in the Business Definition manager (and saved your Business Definition File) a new tab appears in your Form Lists manager for RDP rules.

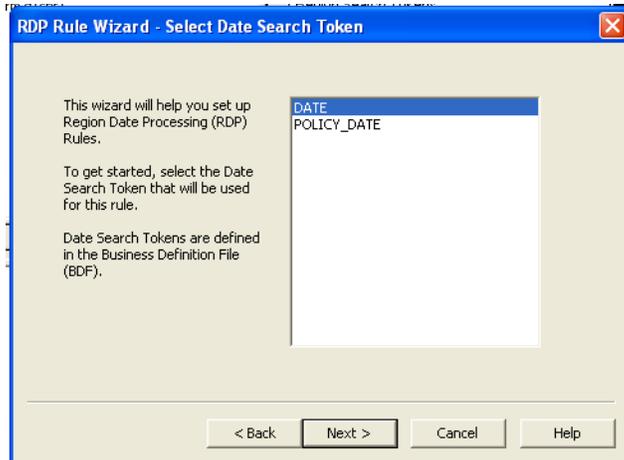
Regional date processing rules are separate from normal trigger rules you might assign to forms. RDP rules operate as a *filter* that occurs before normal form triggering commences. Thus, RDP rules are specified separate to any necessary trigger condition you also would use to identify when you want to include this form.



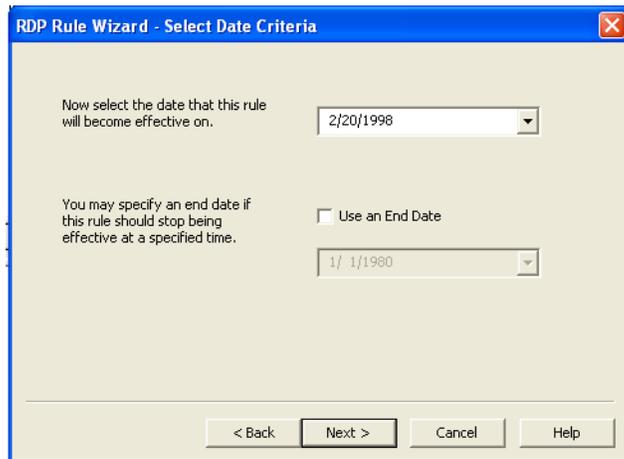
Select the RDP Rules tab to manage your RDP rules assigned to forms. RDP rules are optional. Any form that does not specify an RDP rule is not subject to filtering before the triggering process begins. Also, a form can have multiple RDP rules if necessary. During runtime, each rule will execute in sequence until a matching condition is found, or until the list is exhausted. At that point, the form is excluded from further triggering.

Select the form you want to add a rule to and click the Plus (+) button in the properties area. This adds a rule to the form and starts the RDP Rule wizard.

The first page of the wizard will ask for the Date Search Token (DST) you want to use for this rule. Only one Date Search Token can be used per rule.



During rule execution, the Date Search Token is used to find and perform the appropriate search of the transaction data. At that point, it is necessary to compare this date to an appropriate range that you will define next. Click Next.



On this page, you identify the date this form becomes effective for this rule. Optionally, you can also specify an end date if the form is only available within a given range. If you do not specify an end date, the form remains active indefinitely. You can use the pop-up calendar control to select your date.



Click Next.

This next step is where you select your region search tokens. You can select all the region search tokens that apply this rule. This means if it is required, you can get search data from multiple locations within the transaction to satisfy this single rule.

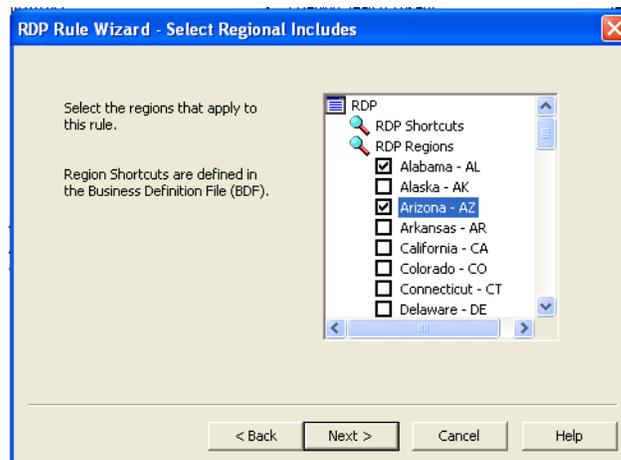


To select an item, click on it in the list. It will remain highlighted as you continue to click other items. To deselect an item, simply click on it again.

Click Next.

Similar to how you specified a date for comparison to the Date Search Token; the next two wizard pages let you specify the range of values that match your region search tokens.

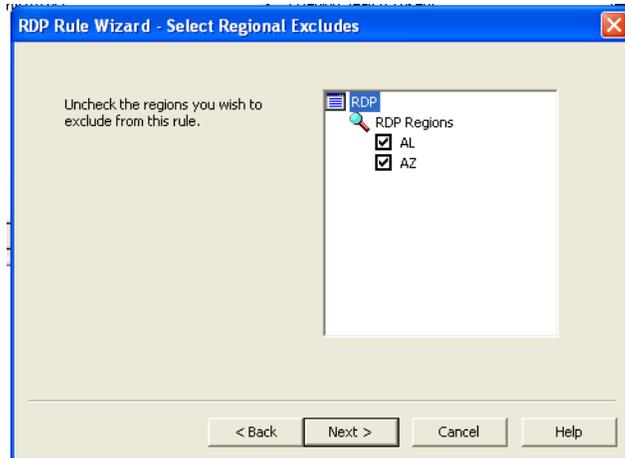
This page lets you identify all the states or Region Shortcut names you want to include with this rule evaluation.



Although you can select multiple region values here, it is only necessary for one of those regions (such as states) to match the resulting search data to satisfy the rule. In other words, specifying multiple selections does not mean they all have to match to satisfy the rule.

Click Next.

Because you can select shortcuts on the previous window that define multiple regions, this page lets you refine your list by excluding certain individual regions.



For instance, suppose you had selected a Region Shortcut named *SouthEast* that contained GA, AL, FL, SC, and NC. Although the previous page would show the selection of *SouthEast*, this exclusion page shows the expanded list of regions included therein. This lets you turn off certain states within that token if they do not apply to this rule.

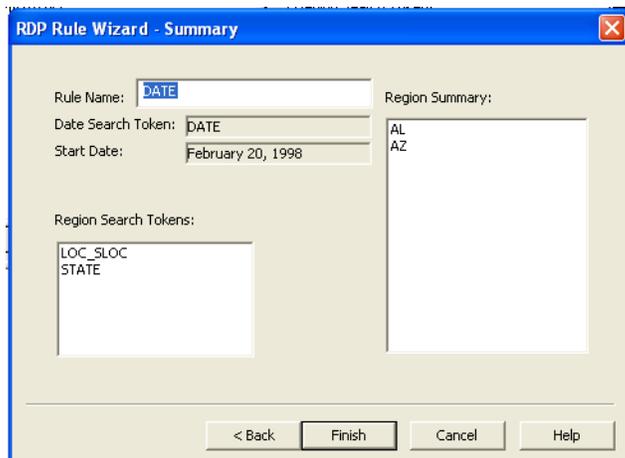
Another example might be a shortcut called *Nation* that includes all 50 states and DC. But if this rule were to require that California be excluded, you could then deselect (uncheck) CA from the resolved list and that region would be omitted when the rule executes.

Therefore any regions and shortcuts you selected from the first page are resolved to show the individual regions on this page. Then you can uncheck any region, or regions, you want to exclude from the rule.

The final page of the wizard is a summary of your rule's settings. The name of the rule defaults to the Date Search Token selected, but you can override it on this page. Included is a resolved region list of which regions are used for this rule. Click Finish if you are satisfied with the rule, or Back to make changes. Click Cancel to discard all changes and return to a blank Rule Properties screen.

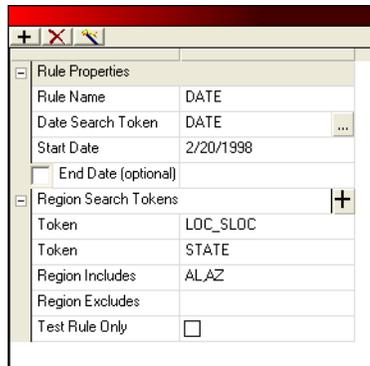
Click Next.

This is the last page of the RDP Rule wizard and represents a summary of the rule definition you have built.



Once you confirm that the information is correct, click Finish and the rule information is saved to the form you previously selected.

The rule definition appears in the property area as you click on them under each associated form.



Rule Properties	
Rule Name	DATE
Date Search Token	DATE ...
Start Date	2/20/1998
<input type="checkbox"/> End Date (optional)	
Region Search Tokens +	
Token	LOC_SLOC
Token	STATE
Region Includes	AL,AZ
Region Excludes	
<input type="checkbox"/> Test Rule Only	

You can change certain information in the property window or click on the Magic Wand toolbar icon to go through the RPD Rule wizard again.

Note there is one property (at the bottom) that is not shown via the wizard. The Test Rule Only field lets you define a rule for testing purposes only. If you check this field, this rule only executes when you are running Studio's Test Manager. This makes it easy to define your rules and work on them without interfering with other users who might be running the batch program externally.

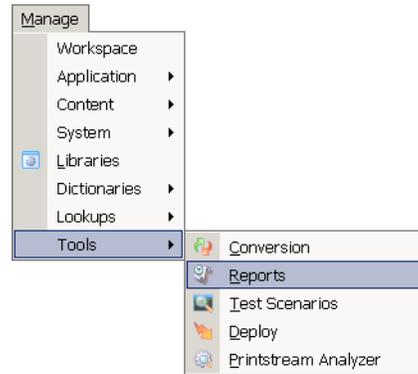
When you are ready for the RDP rule to be used during all runs, simply uncheck the Test Rule Only field.

When you have added all the RDP rules you want, save and check in the Group file.

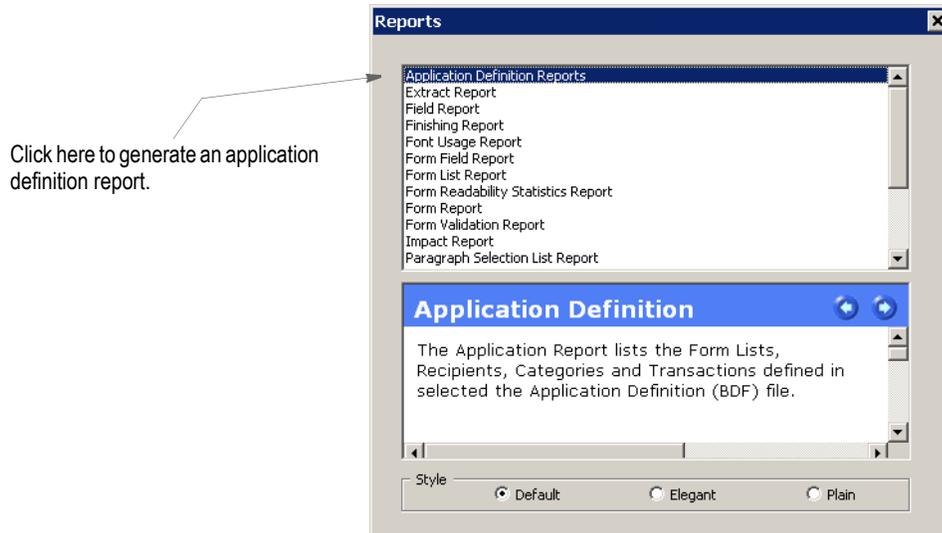
Note You can see information about the RDP rules you have created when you run an Application Definition Report.

GENERATING AN APPLICATION DEFINITION REPORT

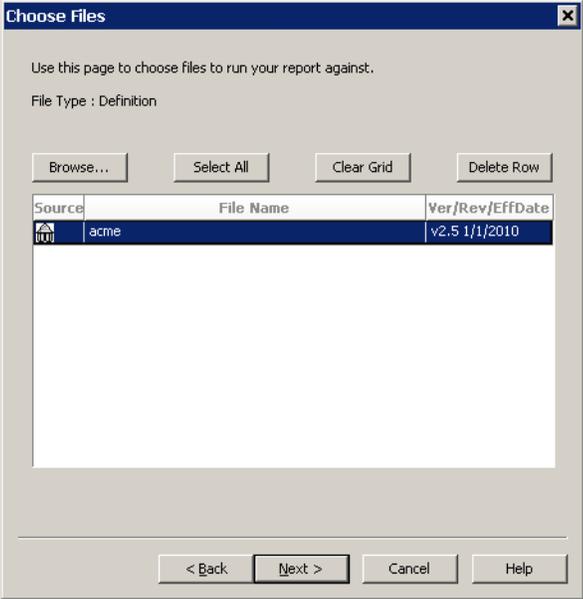
To generate a report which shows the options defined for this application definition, choose the Manage, Tools, Reports option and follow these steps:



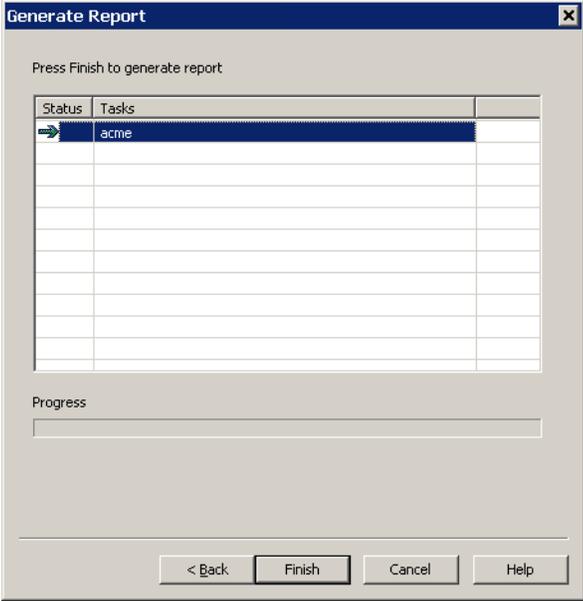
1. This window appears when you click the Reports option. Click Application Definition Reports.



Next, choose the style of report you want. You can see examples of the styles in *Printing Reports* on page 455. The Choose File window appears:



- 2. Choose the application definition file you want and click Next. The Generate Report window appears.



- 3. Click Finish to generate the report. Here is a sample report:



You can right click within the report to choose from these options:

Option	Description
Clear Contents	Clears the contents of the report window.
Select All	Select all of the text within the report.
Copy	Copy all selected text within the report, so you can paste it into another application.
Save As	Save the report as an HTML file.
Send To	Send the report as an attachment in an email.
Print	Print the report.
Print Preview	Show the report in the Print Preview window.

Chapter 4

Working with Form Lists

Click Form Lists to define a list of the forms that apply to each business unit and to maintain form-level triggering information. You can also define the order in which these forms should be provided.

Information about each group of forms is stored in a file with a *GRP* extension. For each group (Key1/Key2), you have a separate GRP file. Group name (Key1/Key2) information is stored in the BDF file, whereas specific information for each of the forms that make up the group is stored in a GRP file.

This topic discusses:

- *Creating a Form List on page 116*
- *Adding a Form to a Form List on page 118*
- *Setting Up Triggers on page 120*
- *SetRecip Options on page 124*

CREATING A FORM LIST

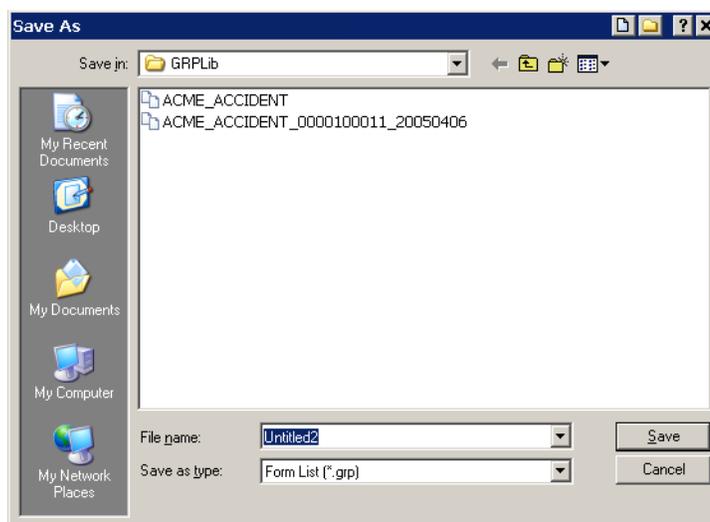
To begin the process of grouping forms, you must first create a form list. Follow these steps:

1. To create the group, double click on Form Lists. Then click New on the Open File window. You can also right click on the Form Lists option and select New. The Form List Options appear.
2. Enter the following information to set up the forms you want included in the group in the following fields. Keep in mind that the forms do not have to exist at this point, you are merely creating a list of form names and defining some of the options that apply to those forms.

Field	Description
Form Name	Enter the name of the form you want to include in the group. You can enter up to 100 characters. This name appears on the Forms Selection window in Documaker Workstation
Form Description	Enter a description of the form. You can enter up to 100 characters. This description appears in the object tree beside the form name. This description can help a Documaker Workstation user select the correct form.
Dec Options	Choose the appropriate declaration page option. Your choices are: <ul style="list-style-type: none"> • None - Any page other than a policy declaration page • Dec Page - A policy declaration page • Master Dec - A master declaration page for the form set (package policy) • Sub Dec - A sub dec page for one group in the form set This option is only used by Documaker Workstation. What you select here determines the manner in which Documaker Workstation presents declaration pages during entry. This option is ignored during batch processing.
Pre-selected	Check this option if you want Documaker Workstation to automatically select this form during entry.
File Name	Enter a file name for this form. Usually, you would keep the form name and the file name the same, but some platforms will not let you use certain characters in file names.

Note Because Dec Options and the Pre-selected fields are stored at the group level, you can reuse the same form in different groups.

3. Choose File, Save to save your work. The Save As window appears.



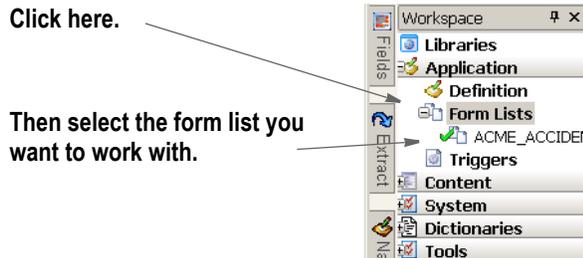
Enter the name of the group in the File Name field.

Note Another way to create a form list is to click on the Add Unit Names icon when working with Application Definition files.

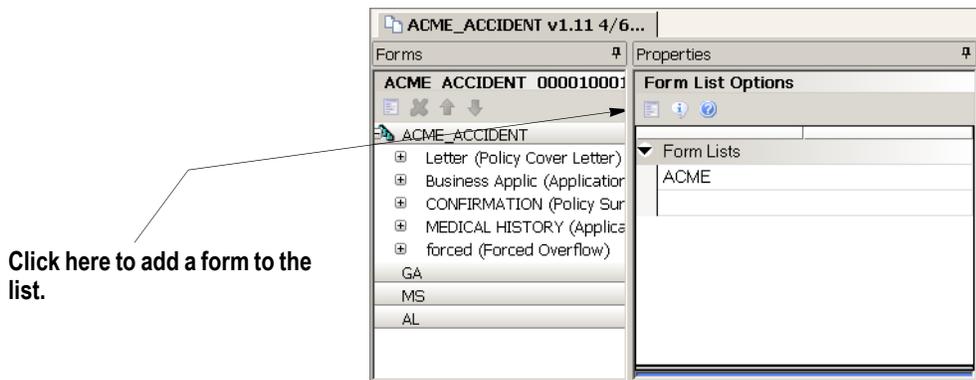
ADDING A FORM TO A FORM LIST

To add a form to a form list, follow these steps:

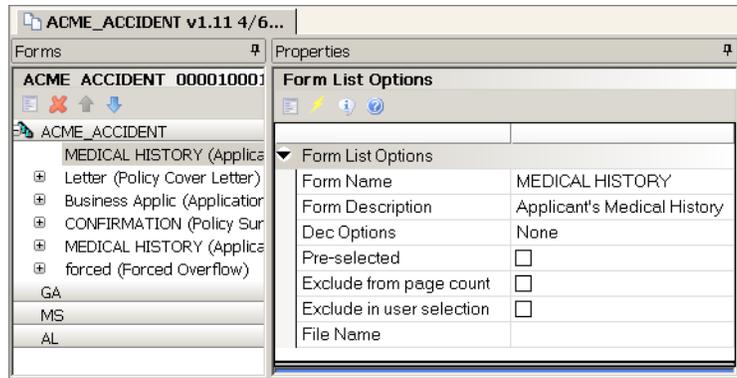
1. Click the plus sign (+) to the left of Form Lists and then select the form list you want to work with.



The following window appears.



2. Click the Add New Form icon to add a form to the list. The Form List Options appear for the form you added.



3. Enter information into the following fields:

Field	Description
Form Name	Enter the name of the form you want to include in the group. You can enter up to 100 characters. This name appears on the Forms Selection window in Documaker Workstation

Field	Description
Form Description	Enter a description of the form. You can enter up to 100 characters. This description appears in the object tree beside the form name. This description can help a Documaker Workstation user select the correct form.
Dec Options	Choose the appropriate declaration page option. Your choices are: <ul style="list-style-type: none"> • None - Any page other than a policy declaration page • Dec Page - A policy declaration page • Master Dec - A master declaration page for the form set (package policy) • Sub Dec - A sub dec page for one group in the form set This option is only used by Documaker Workstation. What you select here determines the manner in which Documaker Workstation presents declaration pages during entry. This option is ignored during batch processing.
Pre-Selected	Check this option if you want Documaker Workstation to automatically select this form during entry.
Exclude from Page Count	Check this option to exclude the form from the page count. Then it will not be counted in the page numbering for the entire document. Also, any Page Num fields that you place on the pages of this form will be blank.
Exclude from User Selection	Check this option to prevent end users from selecting this form.
File Name	Enter a file name for this form. Usually, you would keep the form name and the file name the same, but some platforms will not let you use certain characters in file names.

Note Because Dec Options and the Pre-selected fields are stored at the group level, you can reuse the same form in different groups.

SETTING UP TRIGGERS

You can create triggers at both the form and section level. Form level triggers are stored at the group level and specify the condition that would cause a form to be included (or triggered) with a transaction. You should always specify a trigger for a form.

Note Section level triggers are stored at the form level and specify the condition that would cause the section to be included on the form. Only set up section level triggers if the section's presence on the form is conditional.

Studio assumes you do not want to trigger any of the sections that comprise a form if the form level trigger is false. Therefore, section level triggers are automatically ignored when the form level trigger for the transaction being processed is false. This makes the use of an *M* occurrence flag unnecessary if you want the system to only evaluate section level triggers when the form level trigger is true, as was the case in Docucreate's Image Editor.

If you omit a form level trigger, the system does not operate as efficiently since every section level trigger must then be evaluated by the system.

You can assign triggers to include or exclude:

- An entire form
- Sections within a form
- Text areas within a section
- Optional paragraphs in a paragraph list

You can apply triggers to sections on forms, text areas on sections, and paragraphs in paragraph list files. This table summarizes assigning triggers in both Studio and the Documaker Add-In for Word.

To trigger	Type	Studio	Documaker Add-In for Word
An entire form		Add the converted form to the Form List file using Form List manager. Apply the Form level trigger to the Form in Form List manager.	n/a
Sections within a form	Form	Create a for which includes sections. Section triggers are listed against the sections to which they were applied.	Insert a section. Apply triggers to applicable sections.
Text areas within a section	Section	Create a section which includes text areas. Text area triggers are listed against the sections to which they were applied in the Word Add-In	Insert a text area. Apply triggers to applicable text areas.
Optional paragraphs in a paragraph list	Paragraph list	Create a paragraph list file which includes paragraphs. Paragraph triggers are listed against the paragraphs to which they were applied in the Word Add-In.	Add paragraphs. Apply triggers to the applicable paragraphs.

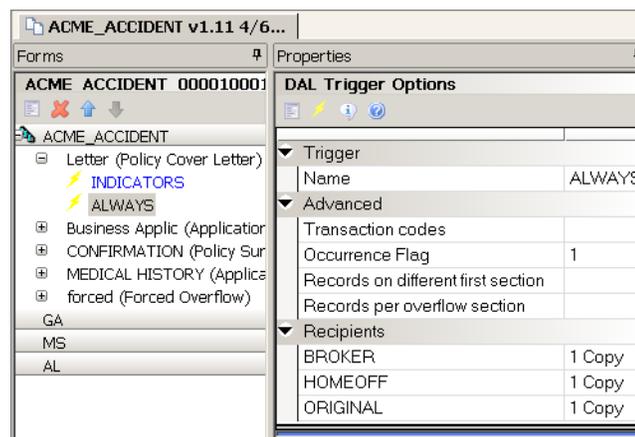
In Studio you cannot assign a trigger to a standalone section, a standalone form or a standalone paragraph list. Form triggers are applied to forms in Form List manager. Section triggers are applied to sections in Form manager. Triggers cannot be applied to a paragraph list file because there is no triggering at the paragraph list level.

If, using the Add-In, you assign a trigger at the form level, section level, or paragraph list level on Documaker documents with the type of form, section, and paragraph list respectively, Studio removes these triggers during the conversion to Studio.

ASSIGNING FORM LEVEL TRIGGERS

Follow these steps to add form level triggers.

1. Highlight Form Lists, right click and select Open.
2. Select the group file in which the form is located and click Ok.
3. Choose the Insert, Triggers option.



4. Use the Trigger Options fields to set up the triggers for your form.

Field	Description
Trigger	
Name	Select the trigger name from the trigger list or select Manual Trigger to create your own. Additional fields appear if you choose Manual Trigger.
Advanced	(Some options may not be present in all managers.)
Transaction Codes	By including one or more transaction codes in this field, you trigger a particular form (or section) only if the extract file record includes that particular transaction code.
Occurrence Flag	To use the search mask as a counter, set the Occurrence Flag to one (1). Use with the Search Mask Counter, Records on Different First Section, and Records Per Overflow Section fields. (0=no overflow and 1=overflow) Performance issue flags in this field include: <ul style="list-style-type: none"> • Master (M) form level trigger flag • Form (F) form level trigger flag • Subordinate (S) Section level trigger flag

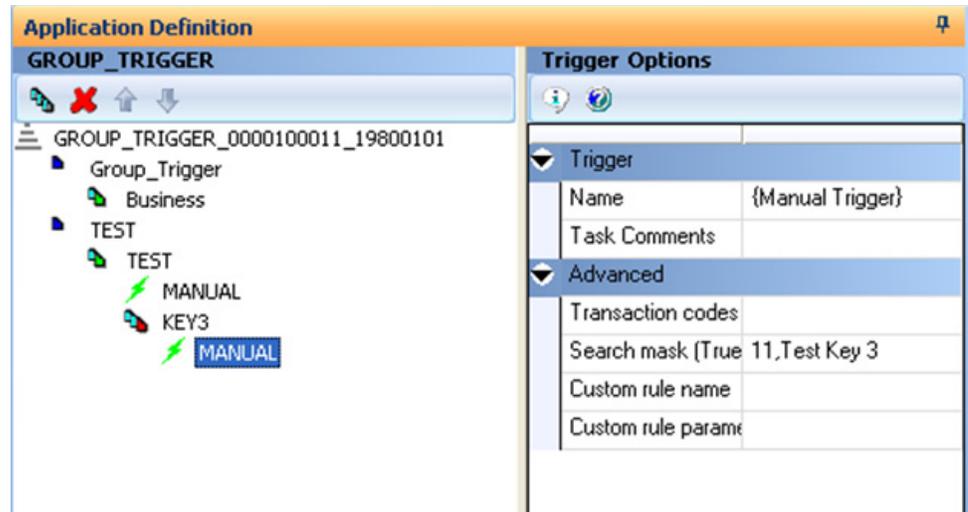
Field	Description
Records on Different First Section	Lets you indicate the number of records that appear on the first section before overflowing to a new section.
Records per Overflow Section	Lets you specify the number of records matching the search mask that will fit on the specified overflow section. If the value is zero (0) and the Occurrence Flag is set to one (1) with the Search Mask Counter set, the system automatically sets this flag to one (1) to perform the overflow calculations.
Search Mask (True/False)	This field only appears if you choose Manual Trigger in the Name field. Define the search criteria to use in determining whether this event should occur. If the search yields a result, a True result is returned. If the search finds no data, the trigger result will be False.
Search Mask (Counter)	This field only appears if you choose Manual Trigger in the Name field. Define the search criteria to use in determining how many duplications of this item to include in the resulting document. The trigger will be applied once for each occurrence found by the search mask. If no matching data is found, the trigger result is 0 (False) and the item is normally excluded.
Custom Rule Name	This field only appears if you choose Manual Trigger in the Name field. This field defines an alternate (custom) rule to execute on this manual trigger to determine its result. For most manual triggers, this field should be left blank.
Custom Rule Parameter	This field only appears if you choose Manual Trigger in the Name field. This field defines an alternate (custom) rule to execute on this manual trigger to determine its result. For most manual triggers, this field should be left blank.
Recipients	(Some options may not be present in all managers.)
(recipient name)	Specify the copy count for each recipient.

ASSIGNING TRIGGERS AT KEY2 LEVELS

You can assign multiple lines of business within a single document, such as Package Policies, and trigger multiple group files at a Key2 or Key3 level.

Defining a Key3 Trigger

Here is an example of how you could define a Key3 trigger. This example adds a Key3 group (Key3) with a manual trigger (“11,Test Key 3”).



The system only evaluates the Key3 trigger if a corresponding Key2 group (TEST_TEST) has been triggered. If the system evaluates the search mask as True and the corresponding Key2 group has been triggered, it evaluates the form in the Key3 form list.

Keep in mind...

- Group triggers only work with the RunTriggers rule.
- The system only evaluates a Key3 trigger if a corresponding (matching Key1 and Key2 group names) Key 2 group has been triggered. There are several ways a Key2 group can be triggered:
 - The Key1 and Key2 definition from the INI is mapped for the transaction.
 - Key2 triggers may explicitly include the group.
 - An import adds a Key2 group. If the system then executes the RunTriggers rule to get more forms, it would then be able to run the Key3 triggers for any Key2s brought in via import.

SETRECIP OPTIONS

The SetRecip Table lets you specify the conditions under which a form or a section will get triggered. Depending upon the requirement, not every field is completed. This table discusses each field:

Section	Field	Description
SetRecip	Transaction Codes <TRANSCDS>	By including one or more Transaction codes in this field, you trigger a particular form (or section) only if the extract file record includes that particular Transaction Code.
	<RECIPS>	
	Search Mask (Counter) <MASK1>	Lets you set the criteria to determine whether or not a form (or section) belongs in the form set. The criteria allows the Rules Processor to get specific data from the extract file. Adds one form (or section) for every occurrence of the search mask per transaction. The Occurrence Flag must be set to one (1) for the search mask to act as a counter.
	Occurrence Flag <OCCURRENCE>	To use the search mask as a counter, the Occurrence Flag (overflow) must be set to one (1). Use with the Search Mask Counter, Records on Different First Section, and Records Per Overflow Section fields. (0=no overflow and 1=overflow) Performance issue flags in this field include: - Master (M) form level trigger flag - Form (F) form level trigger flag - Subordinate (S) section level trigger flag
	Records on different first section <RECSPER1ST>	Lets you indicate the number of records that appear on the first section before overflowing to a new section.
	Records per overflow section <RECSPERIMG>	Lets you specify the number of records matching the search mask that will fit on the specified overflow section. If the value is zero (0) and the Occurrence Flag is set to one (1) with the Search Mask Counter set, the system automatically sets this flag to one (1) to perform the overflow calculations.
	Search Mask (True/False) <CONDITIONAL>	Similar to Search Mask (Counter), but only one form (or section) is triggered, regardless of how many occurrences of the condition exists.
	Custom Rule Name <FUNCTIONNAME>	Normally used for custom implementation requirements. Only exceptions are RECIPIF and RECIPCONDITION rules.
	Custom Rule Parameters <DATA>	Specifies parameters for the custom rule used in the Custom Rule Name field.
Recipient	Form Level Trigger Recipient Copy Count <RECIPCYCT>	
	Section Level Trigger Recipient Copy Count <RECIPCYCT> Recipient names listed with no check or a check	Defaults the recipient copy of each recipient defined at the form level If recipient was eligible in the form file, it is checked here.

Chapter 5

Managing Forms

Click Forms to work with the list of sections that comprise the form and maintain section-level triggering information. The triggering information determines what criteria must exist for this form to be included in a form set.

This chapter discusses the following topics:

- *Overview on page 126*
- *Using the Screen on page 132*
- *Creating a Form on page 138*
- *Modifying Forms on page 140*
- *Using Form Templates on page 147*
- *Setting Up Triggers on page 153*
- *Handling Overflow on page 157*
- *SetRecip Options on page 165*

OVERVIEW

A form is a single document containing one or more pages or sections. Most forms contain multiple pages that are usually printed on both sides of a single sheet (duplex). Some forms are printed only on one side (simplex). Typical forms include insurance policies, tax returns, and mortgage documents.

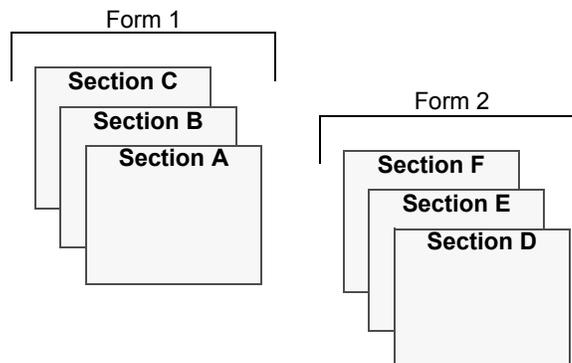
The information for a form is stored in a file with an FOR extension. For each form you create, you have a separate FOR file. A form is listed in a GRP file, whereas the specific information for each form is stored in a separate FOR file.

A form includes two types of data: *fixed* and *variable*.

- *Fixed* data is the same on every copy of the form. This includes items such as graphics. This information remains constant regardless of the data entry.
- *Variable* data may differ from form to form. This includes items such as individuals' names, addresses and policy numbers. This information relates to the specific data processed on each form.

A single form consists of one or more sections. Since multiple forms and sections make up a form set, you can view and navigate through each form and section individually.

Forms consist of one or more sections



A form may consist of multiple sections and, consequently, you can view the varying sections within a form by displaying different pages of the form.

Objects

Objects are the individual items which comprise your section. Examples of objects are boxes, bar codes, lines, graphics, and text. All objects have unique attributes within the section. Attributes include items such as position, size, font type, and color.

USING EMBEDDED SECTIONS

Few other document composition tools use sections as building blocks for forms. In most, you work on individual forms as a whole. Since Studio users sometimes import content from other tools, Studio provides a way to break imported or converted forms into sections. These sections are called *embedded sections*.

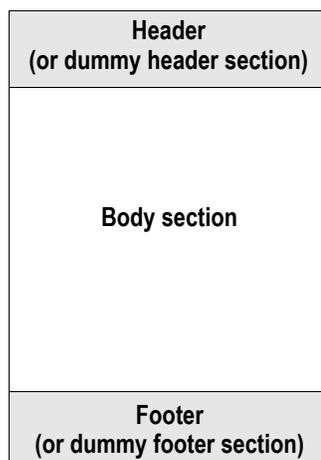
Embedded sections provide a way to edit the content of a form as though it was a section. You can also *unembed* an embedded section if you want to replace it with another section or make that content available for use on another form.

Embedded sections also provide a way to maintain the content of a form without having to create a separate library resource and check out the section to edit it. This lets you see the edits to a section in context with the remainder of the form. Using embedded sections can be appropriate when you do not plan to reuse or maintain separate versions of the content.

You can import these types of files into Studio:

- RTF files
- DOC and DOCX files created in Microsoft Word
- DOCX and DOCM files created with the Documaker Add-In for Word which have a document type of *form*

These files are imported as forms, with at least three sections:



By default, these sections are embedded into the form during the conversion. For more information on converting and importing files, see *Converting Files* on page 409.

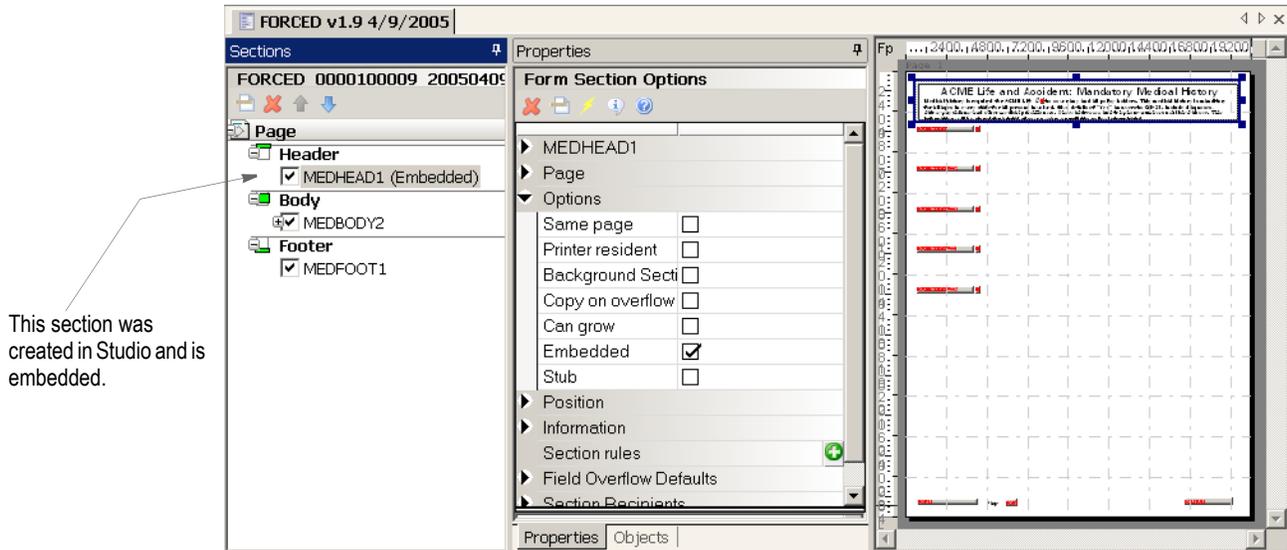
Note You can also manually embed sections you create in Studio after you insert them into a form.

When you use embedded sections, each section can still be named, but will not load from the library because its content is part of the form. The sections are assigned names derived from the name of the document or the use of the content within the document — such as *header* or *footer*.

If you specified the section name in Studio or you converted a section from the library into an embedded section, the name of that section would appear in Form or Template manager as shown here:

SectionName (Embedded)

Here is an example:



This section was created in Studio and is embedded.

Editing an Embedded Section

You edit embedded sections using the Forms or Template manager, instead of the Section manager.

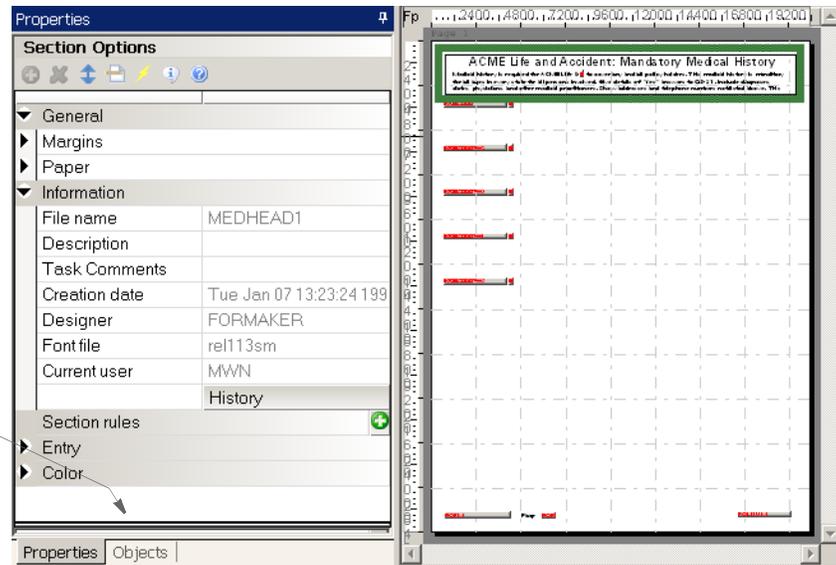
To edit an embedded section in the Forms manager, follow these steps:

1. Select a section in the Sections object tree
2. Right-click and select Edit on the right-click menu.

Note You can also double-click the section in the work area.

Once you are in section edit mode in Forms manager, the menu and toolbar changes to reflect the Section manager options. You also see Section options on the Properties tab.

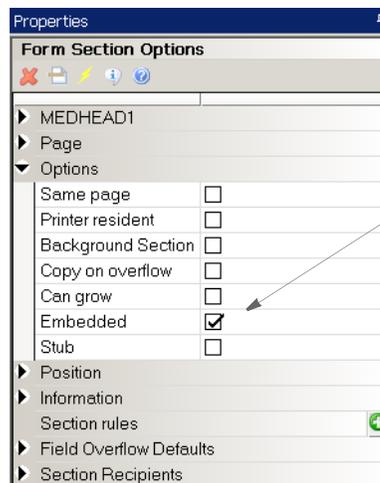
Section objects are listed on the Objects tab. Just click on an object in the list to display its properties.



Note Press ESC, when in section edit mode but without an object selected, to exit section edit mode.

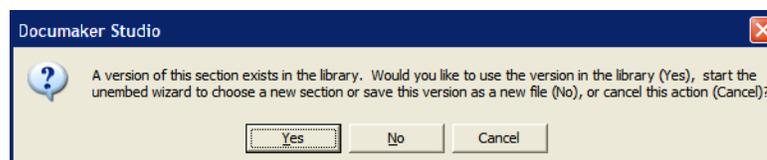
Unembedding an Embedded Section

The Embedded option lets you indicate whether the section should be part of the form or its own file in the library or on disk. To unembed an embedded section, simply uncheck the Embedded option.



Click this option to unembed an embedded section.

If the section was manually embedded, you are finished. If the embedded section was created via a conversion, the Embedded Section wizard appears. This wizard guides you through a series of questions to determine how to handle the section. If a section with the same name exists in the library, you see this message:



You have these options:

If you want to	Click
Replace the section on the form with the one found in the library	Yes
Start the wizard to determine what to do	No
Leave as embedded	Cancel

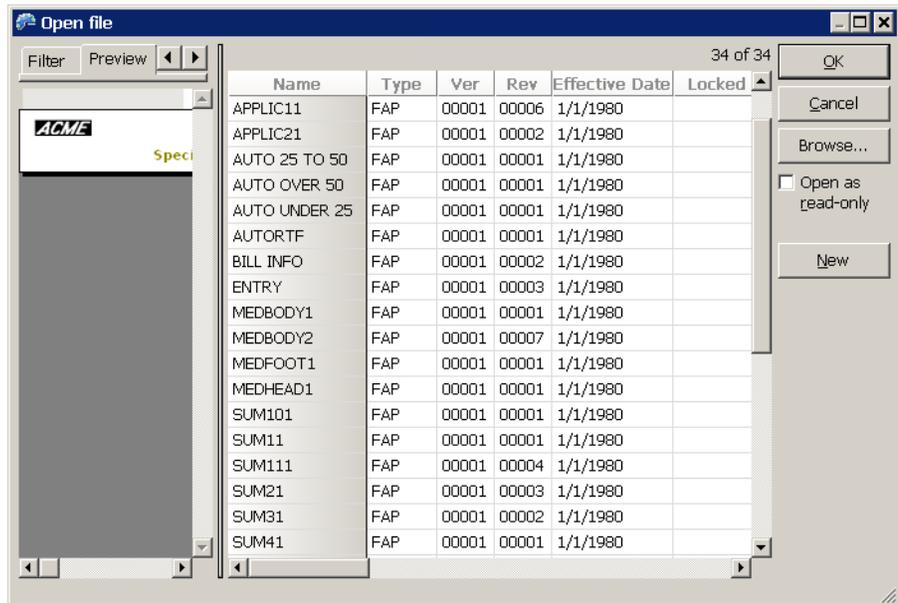
If you click No, or if there was no section already in the library by that name, the Embedded Section wizard appears:



You have these choices:

- Replace the embedded section with one from the library or disk.

If you choose to replace the section with a another section from the library or disk, the Open File window appears:



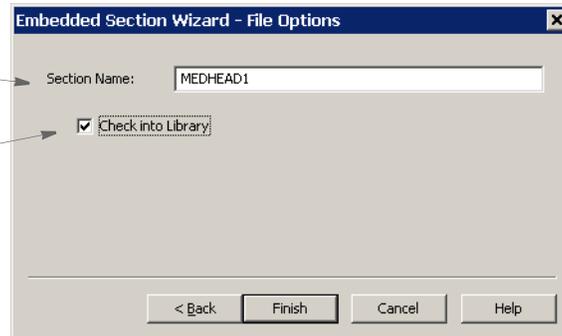
Choose the section you want and click Ok. Click Browse to choose a section from disk.

- Save this section.

This option creates a section in the library with the currently embedded content. You provide a name for the section and tell Studio whether it should check this new section into the library or have it remain on disk.

Enter the name for the section here.

Remove this check mark if you want the section to remain on disk and not be checked into the library.



The screenshot shows a dialog box titled "Embedded Section Wizard - File Options". It features a text input field for "Section Name" containing the text "MEDHEAD1". Below this is a checkbox labeled "Check into Library" which is currently checked. At the bottom of the dialog, there are four buttons: "< Back", "Finish", "Cancel", and "Help".

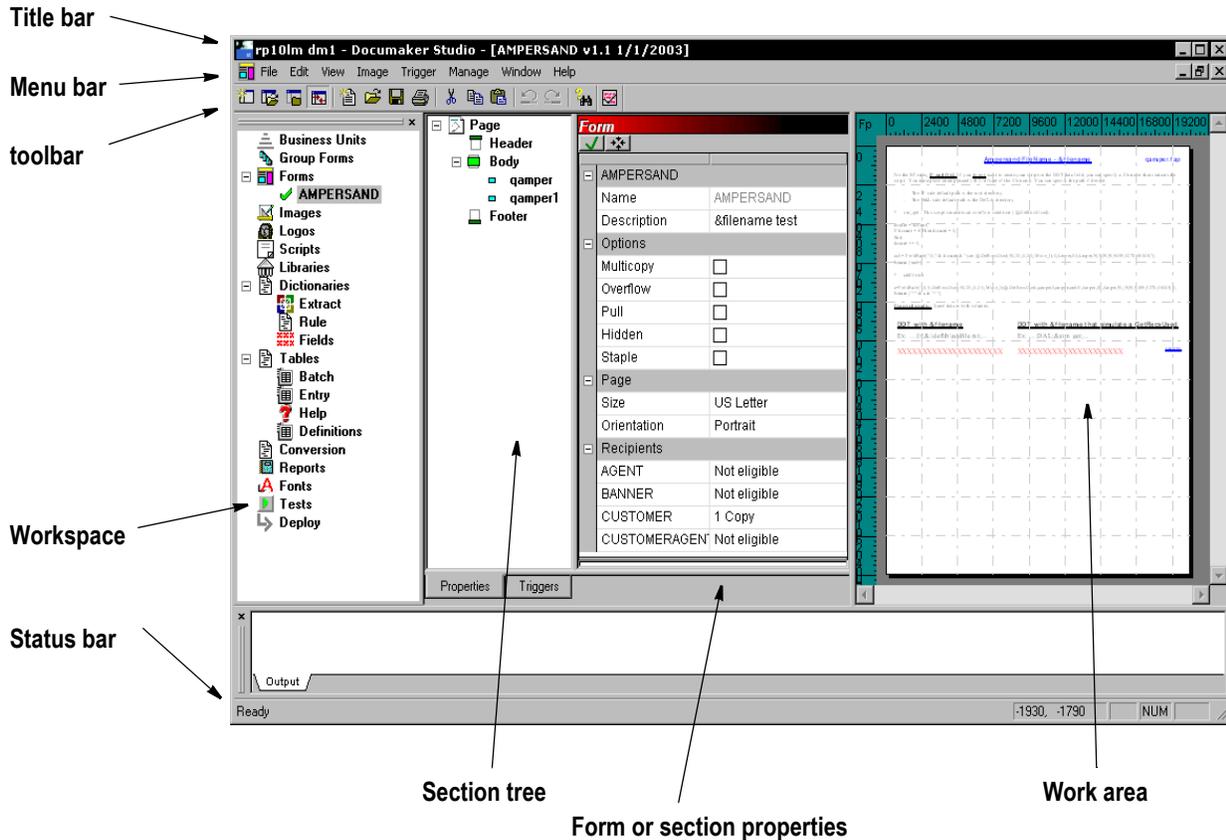
- Unembed the section.

By choosing to unembed the section, the content is removed and not replaced. You will see the result after you close the form and then reopen it.

USING THE SCREEN

Studio places all the tools you need to manage your forms at your fingertips. The screen is your forms work area. It is important to become familiar with the general screen layout and parts of the screen. Understanding the screen layout will help you work quickly and efficiently.

The first window that appears when working with forms is shown here.



Item	Description
Title bar	The title bar displays the name of the workspace you have open, followed by Documaker Studio, and then the name of the form you have open.
Menu bar	The menu bar provides the list of available pull-down menus.
Toolbar	The toolbar contains a row of icons that provide quick access to common options.
Status bar	The Status bar gives the coordinates of the mouse pointer in the work area. The mode of operation, such as ready or edit, also appears here.
Workspace	The workspace lets you quickly access different items. It also shows which specific resources that are checked out (green check mark), which resources are checked out by another user that you would only have read-only access to (red check mark), and which resources are open in read-only mode or have never been checked into the library.
Section tree	Shows the sections that comprise the form.

Item	Description
Form or section properties	Depending upon whether you highlighted a form or a section, if you click the Properties tab, Studio shows you the properties for that form or section. If you click the Triggers tab, Studio shows you the triggers for that form or section.
Work area	This is where you add, delete, or re-arrange the sections that comprise the form. You can also test entering data onto the form.

USING THE MENU BAR

This topic introduces you to the pull-down menus which include additional options or are only available when you are working with forms. A summary of each of these menu appears below. The menus are listed in the order they appear on the menu bar.

Note For information on the standard menus and menu options which are always available, see *Using System Menus* on page 16

Menu	Description
Section	The Section menu lets you add, delete or move sections on the form.
Trigger	The Trigger menu lets you add, delete or rearrange the triggers that tell the system when to include or exclude a form.

Using the Section Menu

The Section menu provides options useful when you are adding a section to a form.



Option	Description
Add	
Page	Adds a page to the form.
Section	Adds a section to the form.
Group	Adds a group begin section. When you add a group begin section, a group end section is automatically added. This lets you associate sections and optionally assign a common trigger to all of them. You can drag and drop or create sections between the <i>group begin</i> and <i>group end</i> sections to indicate they are part of the group. A form group can optionally have a group rule associated with the <i>group begin</i> section.
Delete	Deletes the highlighted section.
Move up	Moves the highlighted section up.

Option	Description
Move down	Moves the highlighted section down.
Data Entry Check	Lets you perform form level data entry check.

Using the Trigger Menu

The Trigger menu provides options for working with section triggers.

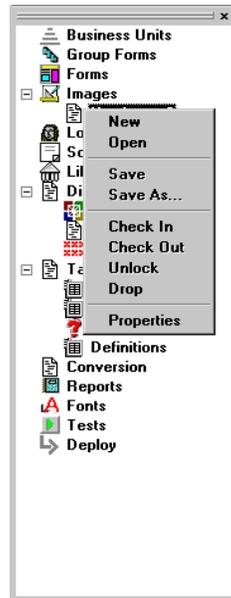


Option	Description
Add	Adds a trigger to the highlighted section.
Delete	Deletes the highlighted trigger.
Move up	Moves the highlighted trigger up.
Move down	Moves the highlighted trigger down.

USING THE WORKSPACE MENU

The workspace area is the area to the far left of the screen. From the workspace, you can open different modules within Documaker Studio. It shows which resources are...

- Checked out (green check mark) by you
- Checked out by another user and you would only have read-only access to (red check mark)
- Open in read-only mode or have never been checked into the library



Option	Description
New	Use to create a form.
Open	Use to open a form.
Save	Use to save an opened form.
Save As...	Used to save an opened form with another name.
Check In	Use to check in a form that has been checked out when you want to save the changes.
Check Out	Use to check out a form that is in the library when changes need to be made.
Unlock	Use to unlock a form that is open when you want to release it back to the library without saving changes.
Drop	Use to temporarily remove a form that appears in the list that is a library item. Permanently remove from the list a form that is not a library item. Unlock or check in permanently removes a library member from the working list.
Properties	Use to display properties of the form.

USING THE TOOLBAR

The toolbar is useful because it serves as a quicker route for performing some functions that may be listed on a drop down menu. Here is an example of the toolbar shown when you are working with forms:



Standard toolbar icons

Shown below are the toolbar icons that are always available. The icons are listed as they appear, from left to right.



Icon	Name	Description
	New Workspace	Creates a workspace.
	Open Workspace	Opens workspace.
	Close Workspace	Closes a workspace.
	Toggle Workspace	Toggles between displaying and hiding the workspace.
	New	Creates a file.
	Open	Opens a file.
	Save	Saves the open file.
	Print	Prints the current object.
	Cut	Removes an object and places it on the clipboard.
	Copy	Copies an object and places it on the clipboard.
	Paste	Places an object from clipboard onto the current file.
	Undo	Reverses your last action

Icon	Name	Description
	Redo	Reverses last undo.
	Help	Displays the Help Context window

Form toolbar icons

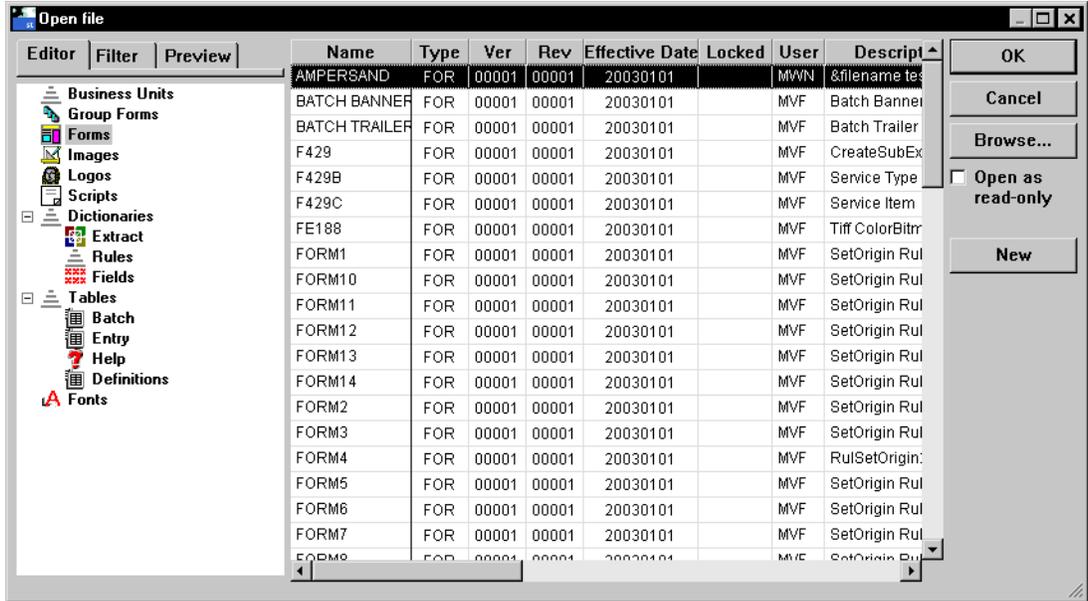
Shown below are the toolbar icons that appear when you are working with forms.

Icon	Name	Description
	data entry check	Lets you test entering data onto a form.

CREATING A FORM

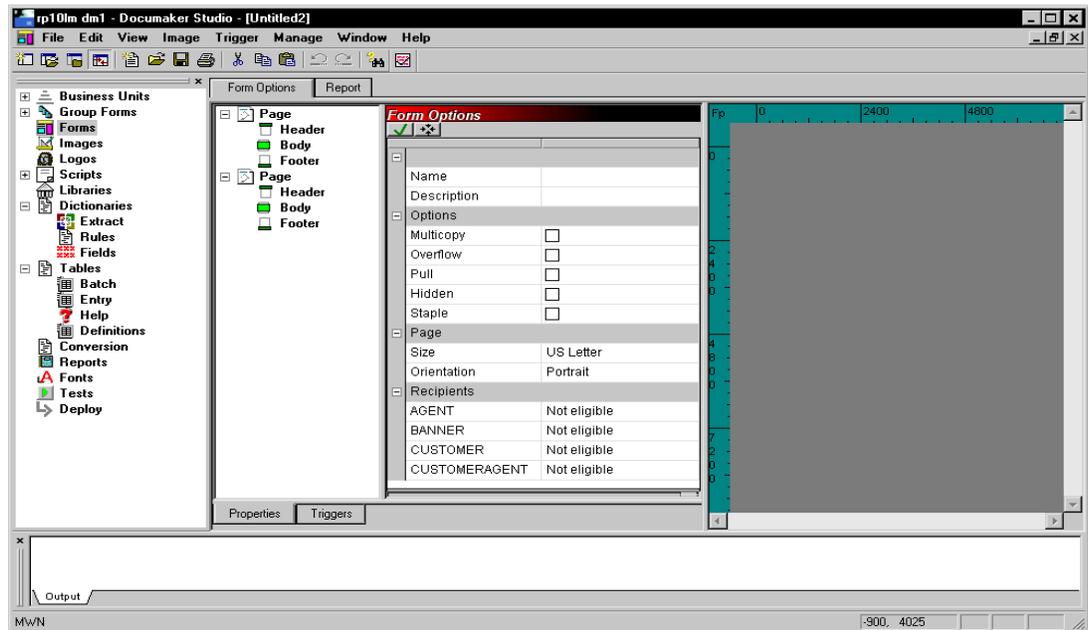
To create a form, follow these steps:

1. Double click on Forms. The Open File window appears.



Note You can also right click on Forms and select New, then click Ok on the Forms window.

2. Click New on the Open File window. This window appears.



3. To define the form, make the appropriate entries in these fields:

Field	Description
Name	Enter the form name. You can enter up to 100 characters. This name appears on the Form Selection window in Documaker Workstation.
Description	(optional) Enter a description of the form. You can enter up to 100 characters. This description appears beside the form name when you are working with Groups. It also can help Documaker Workstation users make sure they have the correct form.
Multicopy	Check this field if more than one copy of the form can be used for a transaction. This only affects Documaker Workstation.
Overflow	This field is for legacy use.
Pull	Check this field if the form is manually inserted into the form set as needed — there is no electronic section. This only affects Documaker Workstation. To add a pull form, you must check this field, then on the Section List window, highlight the blank line and click List. Highlight End of List on the Recipient window and click Add. Next add a recipient with a copy count of one (1) and click Ok.
Hidden	Form is hidden from view in the entry environment but data can be embedded on the form for later use.
Staple	Forms on certain Metacode and PCL printers can be stapled.
Size	Choose the paper size, Only the most common are listed: Letter - prints on default paper size (8.5 x 11) Legal - prints on legal paper (8.5 x 14) A4 - prints on standard European paper (210mm x 297mm) Executive - prints on executive paper (7.25 x 10.5)
Orientation	Choose from portrait or landscape. The default is portrait.
Recipients	Studio shows you the names of the recipients defined by the Recip_Names INI option. For each recipient, you can specify whether the recipient should receive: <ul style="list-style-type: none"> - One copy - Two copies - Three copies - A number of copies set via DAL (if not overridden by a trigger) - A number of copies set via a GVM variable (if not overridden by a trigger) - Eligible - The recipient is not eligible to receive copies of this form.

4. To save the form, choose File, Save. Once you have saved the form, you can check it into the library using the File, Check in option.

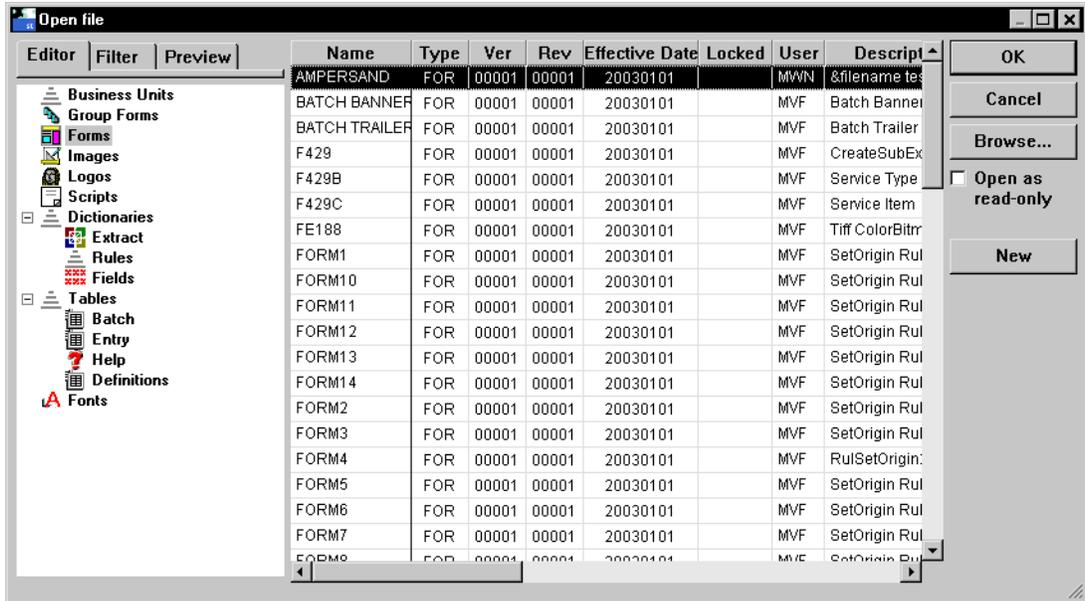
Note You can use the icons at the top of the option area to save the changes or insert a section from the library or disk.

To define section-level triggers, see *Setting Up Triggers on page 153* and *SetRecip Options on page 165*.

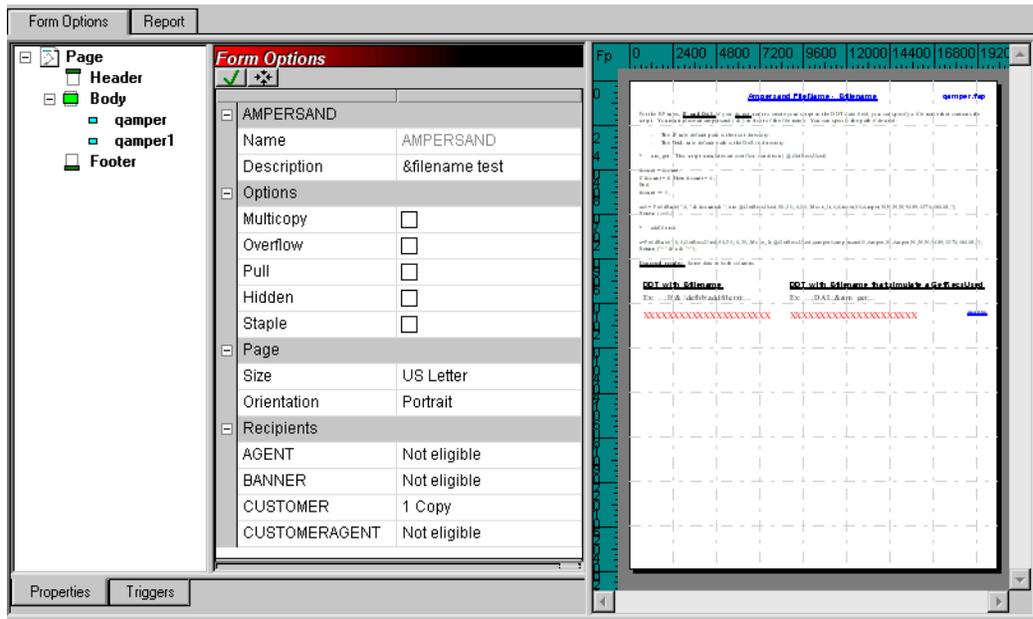
MODIFYING FORMS

Follow these steps to open and modify a form:

1. Double click on Forms. The Open File window appears:



2. Select the form you want to work with and click Ok. When you check out a form for editing or viewing, the Form View window appears. Inside the Form View there are several panels that show information about that form.



The first panel shows a tree view of the sections that comprise the form. The sections are segmented by page and whether they are designated as a *header* or *footer* sections.

Note There are two types of triggers. The ones shown in the examples are the standard triggers which have been in use through prior releases. Newer triggers are actually given names that appear in the tree. The attributes displayed in the Property window will differ.

Note you can resize the various components of the view to suit your needs. For instance, you can zoom the Preview panel to make sections larger or smaller and you can scroll to view a particular page.

Note For more information on section-level triggers, see *Setting Up Triggers on page 153* and *SetRecip Options on page 165*.

3. When you finish modifying a form, select the File, Check in option to save the form and check it into the library. The File Information window appears.

The screenshot shows a 'Check In' dialog box with the following fields and values:

- Name: AMPERSAND
- Description: Ampersand Form
- Version: 00001 (with an unchecked 'Increment version' checkbox)
- Revision: 00002
- Effective date: 01/01/2003
- Mode: DEV
- Status: (blank)
- Class: TX
- Project: P001

Buttons: OK, Cancel

4. Set the library attributes that apply to the form and the changes you made.

Note To learn more about the Mode, Status, Class, and Project fields see *Managing Resources on page 297*.

This window shows you the previous library settings for this form and lets you set new values. You set up the various modes, statuses, classes, and project codes elsewhere, but this window lets you choose the appropriate entry from predetermined lists.

The revision number is automatically set for you. When you specify that you intend to create a new version of the document, you can then enter a new effective date for that version.

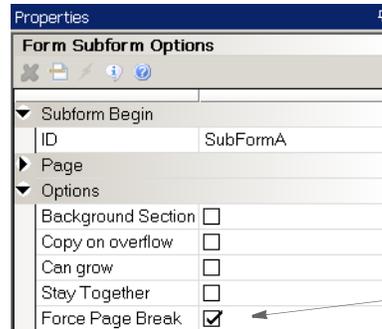
5. Click Ok to check the document back into the library and make it available for others sharing the same resources.

Dropping a form

Dropping a form does not delete the actual FOR file, but does remove it from the library and from the list. To drop a form, highlight the form and then right-click. Select Drop.

FORCING PAGE BREAKS IN SUBFORMS

Use the Force Page Break form-level option to force a page break on the second occurrence of a subform, within the current form or parent subform.



Click this box to force a page break for this subform.

Studio treats any header or footer sections added to the subform as group headers and footers for the subform.

Option	Description
Background Section	Check if you want this subform's origin to not affect subsequent section origins.
Copy on Overflow	Check if this subform can be copied onto additional pages. If this subform is a header or footer, checking this option tells the system to make sure this subform appears on the top or bottom on each subsequent page.
Can Grow	Check if this subform can grow in size. This option is useful when a subform contains a multiline text field which lets the end user enter varying amounts of text.
Stay Together	If you check this box for a subform, the system will not split the subform between pages unless the subform is larger than the calculated page size (page size minus headers and footers). If the subform is larger than the page size, the system splits the subform.
Force Page Break	Check this box to force a page break on the second occurrence of the subform, within the current form or parent subform.

You add headers and footers to subforms just as you would to a normal form. In addition, you can also specify the print control options. Keep in mind, however, that if you select the Not First, Not Last, or Last Only print control option, your selection affects the subform, not the form.

Note The Force Page Break option is not supported by the GroupBegin rule's GroupPagination function. It is supported by the UpdatePOLFile and PaginateAndPropagate rules.

GENERATING READABILITY STATISTICS

Studio lets you generate readability statistics, including Flesch scoring, when you are previewing a form or section in Library manager or when working with sections, forms, tests, and reports.

Note You can also generate readability statistics when working in Documaker Workstation or the WIP Edit plug-in.

.In Studio, Flesch scoring considers information in text areas and multi-line text (MLT) fields. Since the Flesch scoring formula works on sentences, text labels are ignored. Sentences are typically entered within a text area or multi-line text field but generally not in a text label.

Flesch/Flesch–Kincaid Readability tests are designed to indicate how difficult a reading passage is to understand. Readability is based on factors such as the number of words in sentences and the number of letters or syllables per word.

Higher scores indicate material that is easier to read; lower numbers indicate harder-to-read passages. Here is the formula for the Flesch Reading Ease Score test:

$$206.835 - 1.015 \left(\frac{\text{total words}}{\text{total sentences}} \right) - 84.6 \left(\frac{\text{total syllables}}{\text{total words}} \right)$$

where total syllables/total words = average number of syllables per word (ASW) and total words/total sentences = average sentence length (ASL).

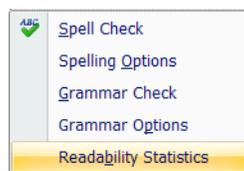
Note This test is designed for English. The scores may not be valid when you run the test on non-English text.

Working with forms and sections

When working with a form or section or running a test scenario, you can generate readability statistics by right-clicking and choosing the Readability Statistics option.

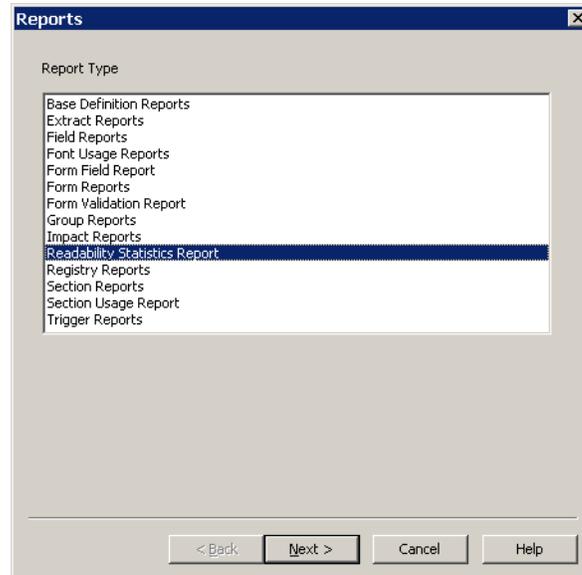
When running a test scenario, this option is only available when sections and forms are displayed as part of the test run. When generated during a test run, the readability statistics are based on all of the forms and sections in the form set.

If you are working in a specific text area, right-click, then choose this option from the Tools menu:

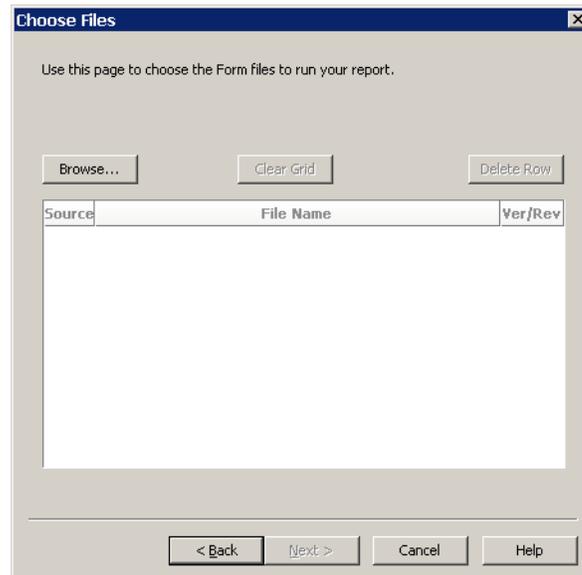


Printing the Readability Statistics Report

You can also choose the Readability Statistics Report from the Report menu. This report is available for both forms (FOR) and sections (FAP) files.

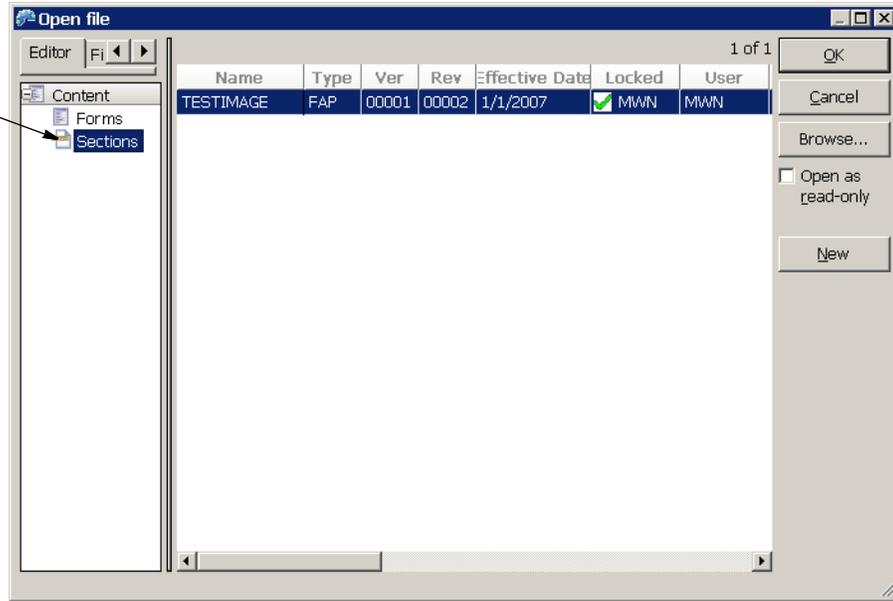


When you click Next, the Choose Files window appears:

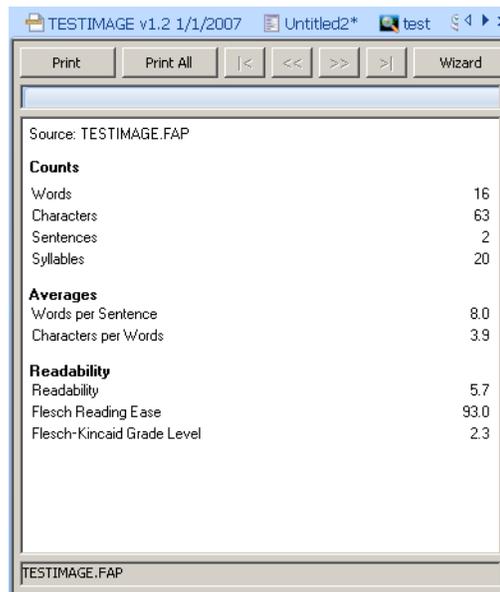


Click Browse to display the Open File window and select the forms or sections for which you want readability statistics generated:

To switch from forms to sections, click here



Once you have chosen the forms or sections you want included, click Next to continue and then Finish to generate the report. Here is an example of the Readability Statistics Report:



USING FORM TEMPLATES

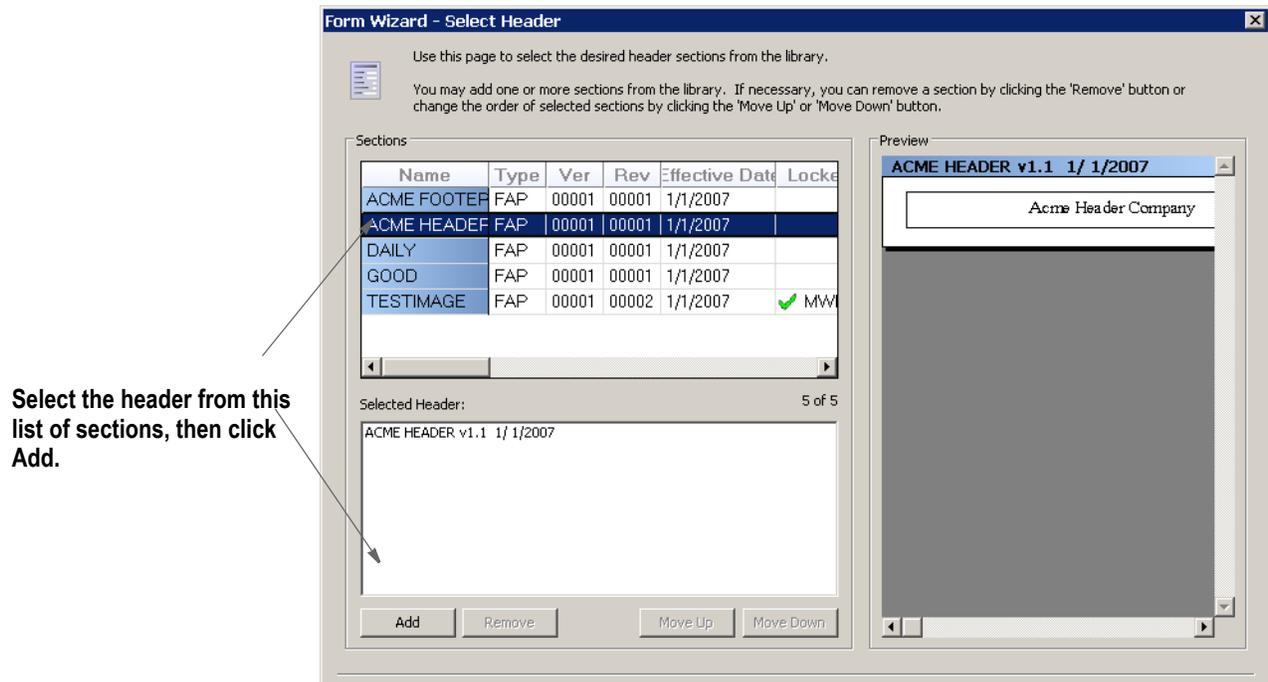
Studio lets you create templates for the forms you create with Studio. You can use templates to more easily implement and maintain document standards while also speeding the process of creating new forms. For instance, you can use templates to make changing your company's name or logo easier.

To add templates, Studio includes the Template manager which you can start from the Manage menu. Studio also includes a new file type (*.tpl), which is used for templates.

Using Template Manager

Use the Template manager to create and maintain your templates. A form template contains headers and footers. You add body sections when you create the actual form.

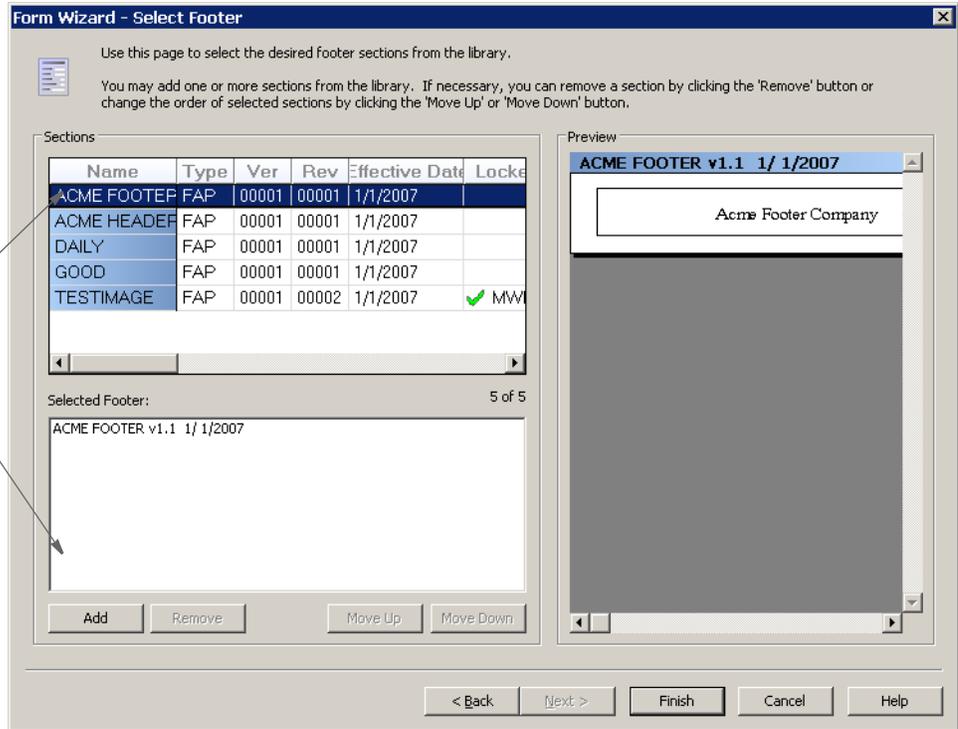
When creating a template, the Form wizard prompts you to select sections for the template's header and footer.



You can see a preview of the section by clicking on it.

Double clicking on the desired sections adds them to the Selected Header list area. You can also use the Add button to add selected sections to the Selected Header area.

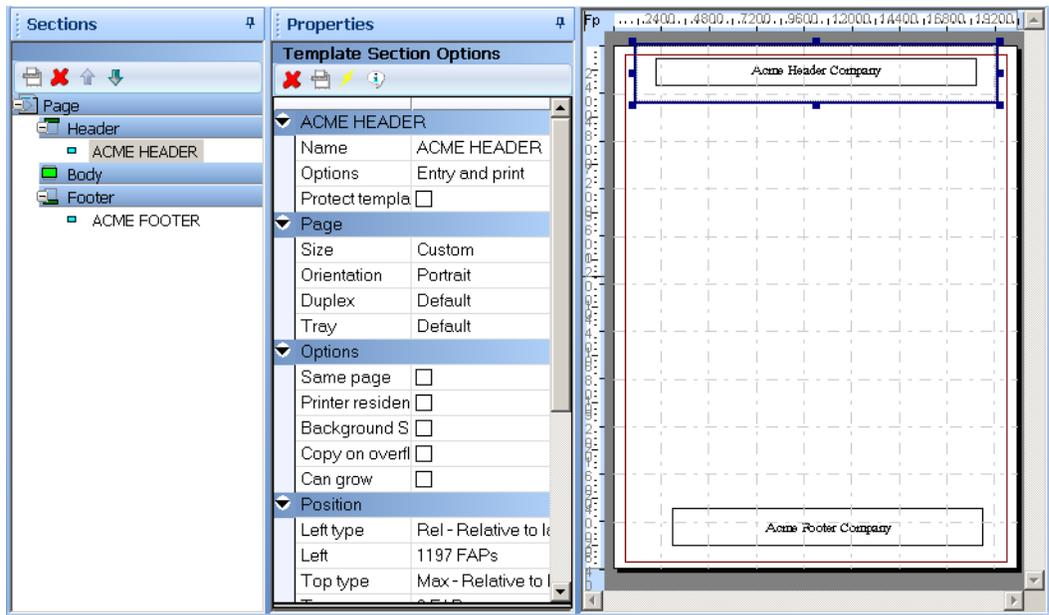
Click Next to display the Footer window and select the sections for the template's footer.



Like you did for the header, just select the footer from this list of sections, then click Add.

Click Finish to close the Template wizard and open Template manager.

Set the header and footer section's options, rules, triggers, recipients, and so on to have the values you would expect a form derived from this template to use.



The Template Section Options includes the new Protect template item field:

This option lets you prevent users from changing the settings you choose for this template object.

▼ ACME HEADER	
Name	ACME HEADER
Options	Entry and print
Protect template item	<input type="checkbox"/>

This option is available for each section that makes up the header and footer.

If you check the Protect template item field, the person using the template to create a form cannot delete or change any of the attributes on that section in the form.

If you protect a section in the header or footer area, then the user will not be able to add any other headers (or footers) into that area on the resulting form.

Although you can define a template that contains more than one page, it is probably not necessary for most forms. If, however, you flag template sections as Copy On Overflow, those sections are automatically copied to new pages created on the resulting form.

Sections not set as Copy On Overflow are included on the corresponding page of the resulting form, but are not included on subsequent page additions.

Using Form Manager

To make it easier to use templates, a creation wizard automatically starts when you begin to create a new form. If there are templates in the library (TPL resources), the first page of the wizard lets you select one.

Note If in your user setup, the *Limit to using Templates* option has been turned on, Studio requires you to use a template when you create a form.

When you create a new form based on a template, you essentially get a copy of the defined template sections. From this starting point, you can add and arrange additional form sections to complete the form.

If the template sections have been protected, you cannot change any of the attributes or options related to those sections. In addition, if the header area or footer area contains a protected template section, you cannot add sections in those areas.

If the template sections were not protected, you can modify attributes or even remove objects from the form. If, however, you reapply the template later, the objects you deleted will reappear.

Here Studio shows you the name of the template used to create the form, if one was used.

Form Options	
  	
▼	
Name	
Description	
Template	LETTERTEMPLATE
Reapply Template	

If a form is created from a template, this information appears on the Form Options window. You can use the Reapply Template button to update your form with the latest definitions from the template.

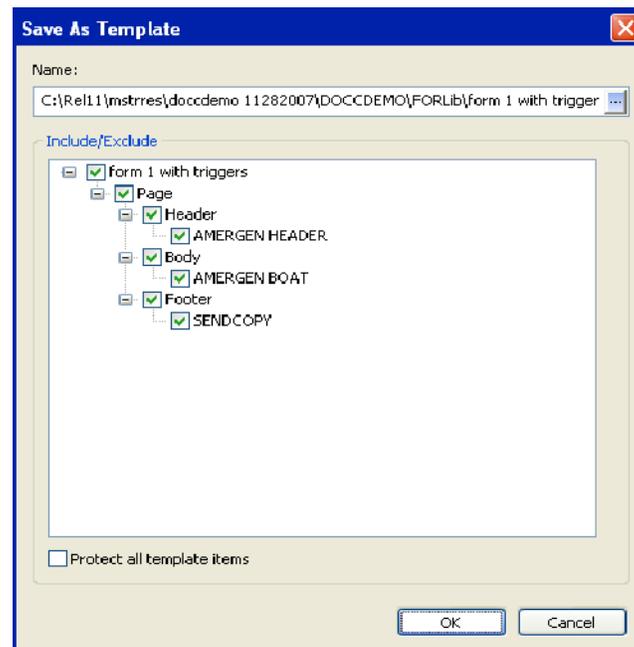
Note that when you reapply a template, this can rearrange, add, remove, and alter the previous template items. In addition, if the previous template definition did not protect the header or footer sections and the new template does, any sections in those areas that are not part of the template are removed.

If you reapply a template and body images are involved, the templated body images move to the top of the body list. This is in keeping with the original default layout of the template had you created a new form.

Finally, if your form contains more pages than the template defines, the pages that exceed the template definition will inherit only the Copy On Overflow template sections. If there are no such sections in the template, your additional pages are not altered.

Saving a Form as a Template

You can save a form as a template by right clicking and selecting the Save As Template option or by selecting Save As Template from the Tools menu. The Save As Template window appears.

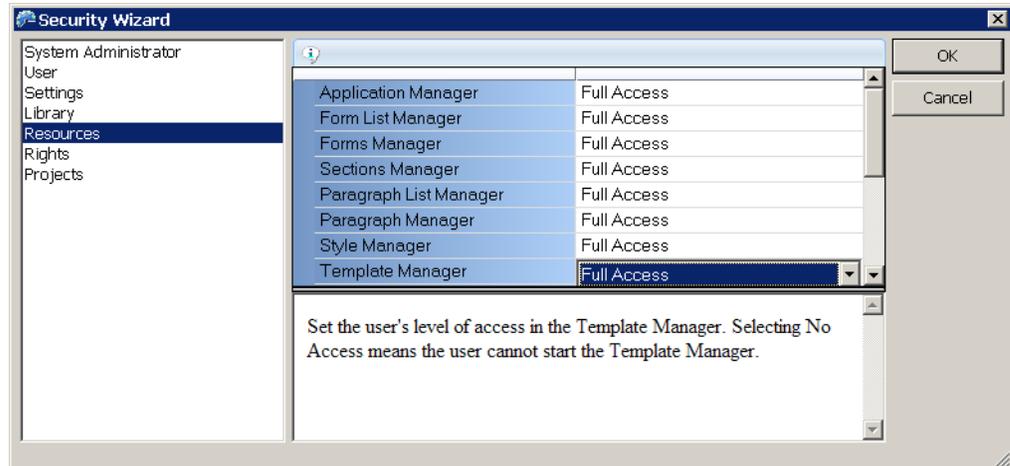


You can select or deselect items by clicking on the top item in the tree (or any other parent item).

You can flag all of the selected items as protected in the template by selecting Protect all template items.

Controlling Access to Templates

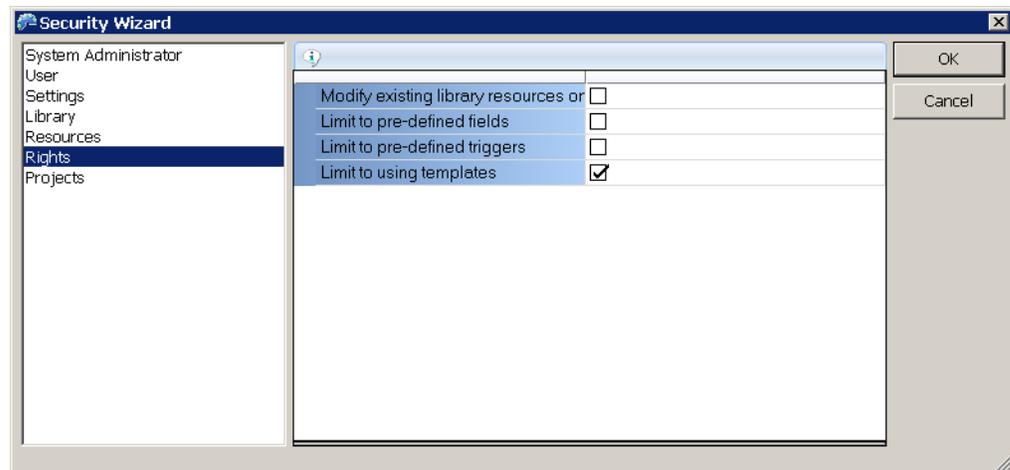
Via User manager, you can specify who can have access to the templates. You can set up the same attributes for templates as are set up for other resources.



Option	Description
--------	-------------

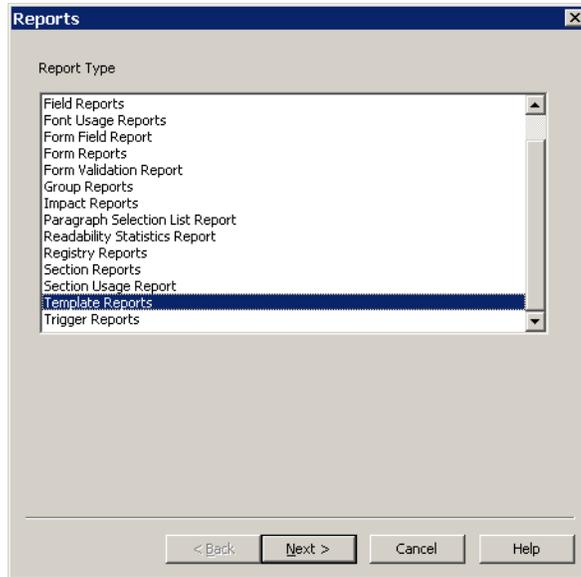
Full Access	Choose this option if you want the user to check out templates and modify them.
No Access	Choose this option if you want to prevent the user from checking out templates. If you choose this option, the user cannot open Template manager.
View Only Access	Choose this option if you want to prevent the user from checking out templates, but allow the user to open Template manager in read-only mode and view the contents of the template.

You can also specify whether a user must use a template when creating a new form by checking the Limit to using Templates field. This field is available under Rights:



Library Manager and Reports

Templates (*.tpl) are a resource type you can select, filter, and report on. Since templates are essentially forms, the information provided on these resources is similar to that for forms.



SETTING UP TRIGGERS

You can create triggers at both the form and section level. Section-level triggers are stored at the form level and specify the condition that would cause a section to be included (or triggered) with a transaction.

Section level triggers are stored at the form level and specify the condition that would cause the section to be included on the form. Only set up section level triggers if the section's presence on the form is conditional.

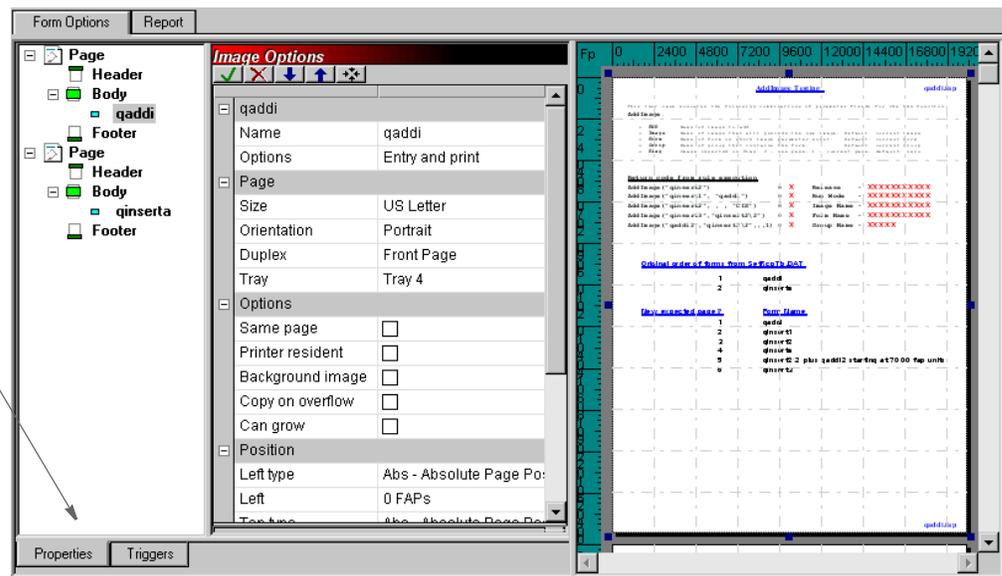
Studio assumes you do not want to trigger any of the sections that comprise a form if the form-level trigger is false. Therefore, section level triggers are automatically ignored when the form level trigger for the transaction being processed is false.

Note If you omit a form-level trigger, the system does not operate as efficiently since every section-level trigger must then be evaluated by the system.

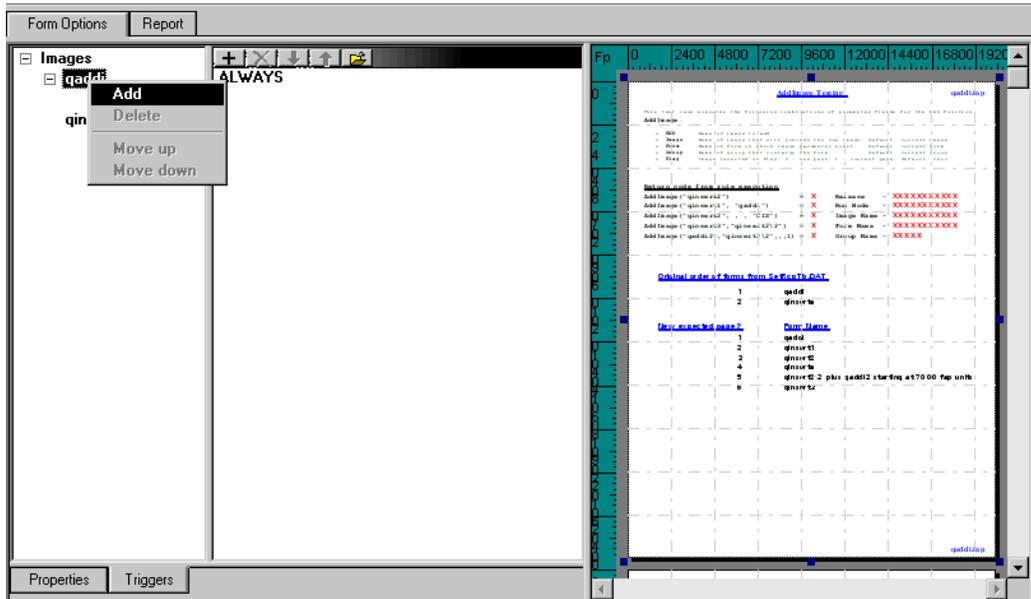
Follow these steps to add section-level triggers.

1. Highlight Forms, right click and select Open.
2. Select the form and click Ok.
3. Highlight the section to which you want to assign a trigger and click on the Triggers tab.

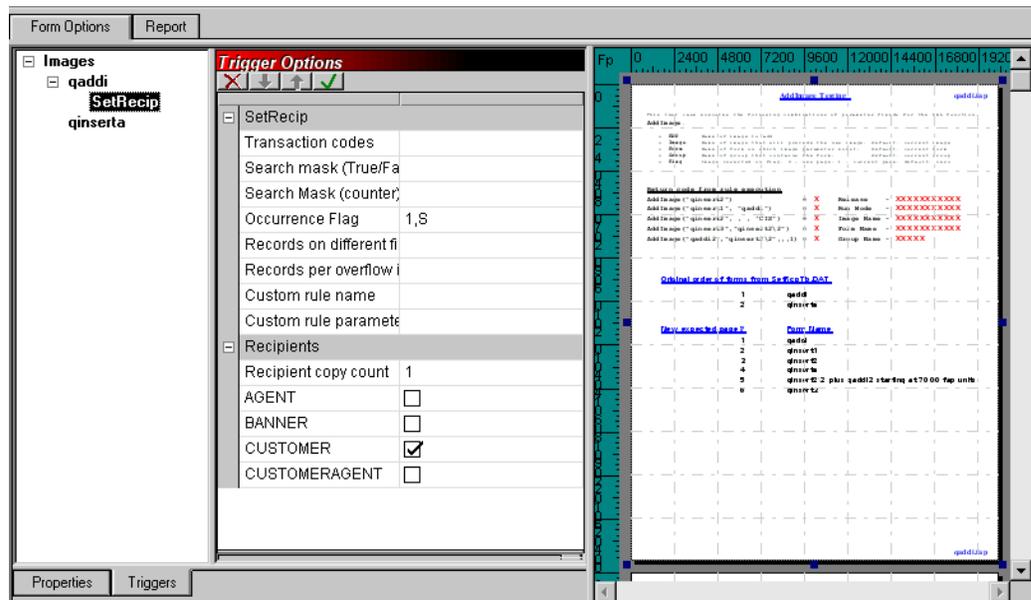
Click the Triggers tab.



- On the Triggers tab, select the trigger you want to edit then right click and choose Add.



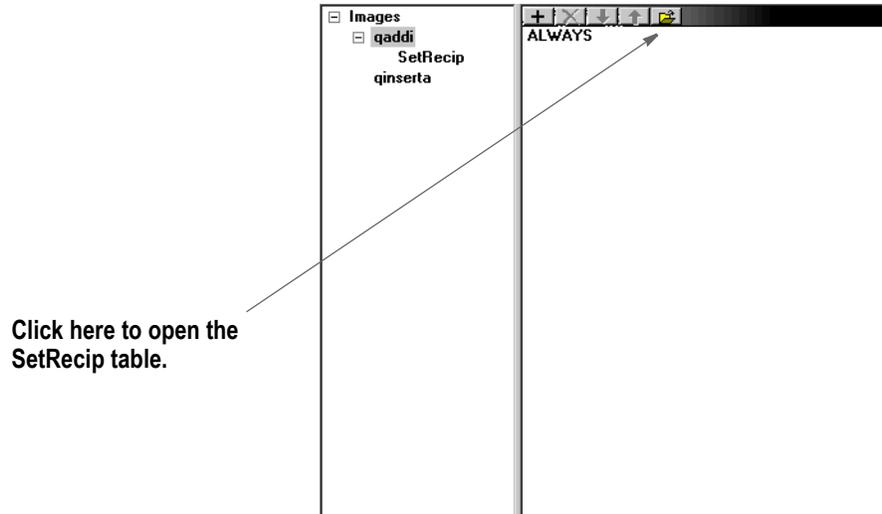
- The SetRecip table appears, on which you can edit or set up the triggers for the section.



EDITING A TRIGGER

If you need to modify a trigger, right click on the trigger and select Edit Trigger File. The SetRecip table appears.

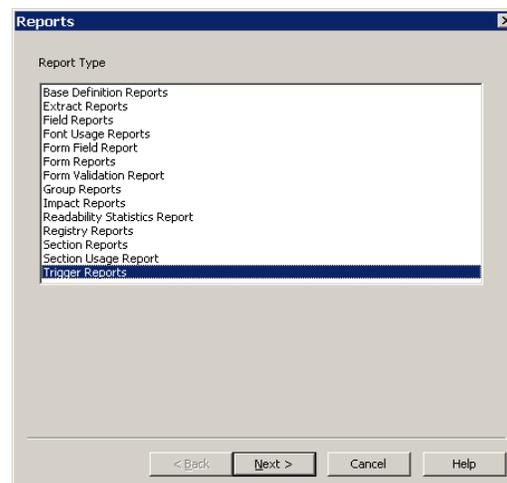
You can also highlight the trigger and click on this icon:



USING THE DAL TRIGGER REPORT

You can use the DAL Trigger Report to see which DAL triggers have been referenced and which have not been referenced in the business definition you select.

You can print this report by choosing Reports and then choosing the Trigger Reports option from the Reports window.



The Reports wizard then takes you through the steps necessary to identify the business definition (BDF) from which to pull the information and to generate the report.

Here is an example of the report:

Business Definition: MYNEWORKSPACE

Subroutines Referenced	
Subroutines	Number of References

Subroutine Referenced But Not in DAL Trigger Library Script	
Subroutine	Target

Subroutines not Referenced
Subroutine
ALWAYS

The report has three sections, similar to the other usage reports:

- The first section lists all of the referenced DAL triggers with a count of how many times that trigger was encountered.
- The second section lists the unrecognized DAL triggers along with the name of target they intend to trigger.
- The third section lists the DAL triggers not directly referenced in any trigger in the business definition

HANDLING OVERFLOW

Handling overflow is based upon the assumption that there is usually a relationship between the data occurrence you want to map and the form and/or page segment (FAP file) occurrence where the fields are defined. All you have to do is set a few field and image options to define the relationship for Studio.

Note Version 11.2 changed the way you mapped overflow. Overflow mappings defined using the prior method are not affected. The prior method may, in fact, prove useful to handle some specific and more complicated overflow situations, such as when the variables are not being mapped onto repeating page segments of the same name.

Keep in mind that there are two types of overflow situations, sometimes referred to as:

- Regular overflow
- Forced overflow

The main difference is how many occurrences of similar data appear on a single page segment.

Regular overflow

In regular overflow, there is one occurrence of each field on the form or page segment. For instance, you might have a page segment that represents an invoice line on a form. Here is an example:

Page segment 1

Classification	Code No	Prem Basis	Rate	Prem/Ops	Commiss	Extended

In this example, seven fields are defined on the page segment. This might represent a single row of information from the input stream you are mapping from. Using this type of page segment, you would trigger one page segment for each row of data found in the extract file. For example, if your extract contained 12 rows of data, you would trigger 12 copies of this page segment. Each page segment you include will need to map a subsequent occurrence of the data.

Forced overflow

In forced overflow, there are multiple occurrences of similar data to be mapped onto the page segment. Here is an example:

Page segment 1

Classification1	Code No1	Prem Basis1	Rate1	Prem/Ops1	Commiss1	Extended1
Classification2	Code No2	Prem Basis2	Rate2	Prem/Ops2	Commiss2	Extended2
Classification3	Code No3	Prem Basis3	Rate3	Prem/Ops3	Commiss3	Extended3
Classification4	Code No4	Prem Basis4	Rate4	Prem/Ops4	Commiss4	Extended4

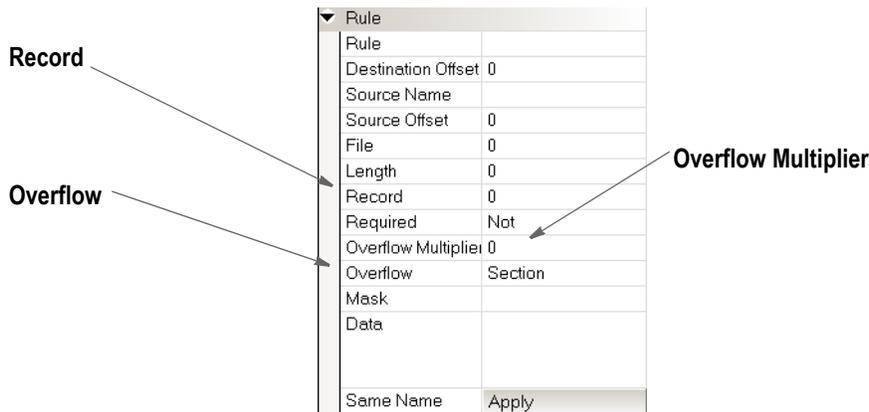
Classification5	Code No5	Prem Basis5	Rate5	Prem/Ops5	Commiss5	Extended5
------------------------	-----------------	--------------------	--------------	------------------	-----------------	------------------

Although similar in layout, in this example there are five pre-defined rows of fields on the page segment. For each page segment you trigger, you will be able to show five additional rows of data.

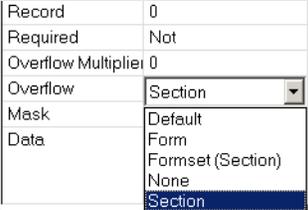
Notice that the names of the fields differ slightly on each row. Fields on a segment must have a unique name, but the underlying mapping of each similarly named field can be the same. Using the previous example of 12 rows of data in the extract file, you would trigger three occurrences of this page segment to hold that data. The last page segment would only map the first two rows of data and the remaining three rows would be empty.

Keeping these overflow examples in mind, let’s look at how you can define the field mappings for these segments.

At the field level, you must tell Studio that repeating data elements will be mapped into the field's location using the Overflow, Overflow Multiplier, and Record fields.



Field	Description
Record	<p>Use this field in the forced overflow situation.</p> <p>Studio adds the value you enter to the current overflow calculation to reach the correct occurrence of the data to retrieve from the source file.</p> <p>If you enter 1 (or 0), Studio uses the occurrence calculation without modification— it uses the first record it finds.</p> <p>If you enter 2 or higher, it tells Studio that you want to reach the second occurrence value at the current calculation and so on.</p> <p>The exact overflow calculation is determined by using the Overflow Multiplier and Overflow fields.</p> <p>The default is zero (0).</p>
Overflow Multiplier	<p>When using forced overflow, there are multiple occurrences of the same data on the page segment.</p> <p>Use this field to indicate the number of times the field is repeated on the page segment. Enter a value greater than 1 to tell Studio you require forced overflow calculations.</p> <p>The resulting calculation is then added to the Record field to determine the exact overflow occurrence you are trying to map.</p> <p>The default is zero (0).</p>

Field	Description
Overflow	<p>Use this field to indicate overflow is required.</p> <p>This applies to both regular and forced overflow. The option you select here defines how the occurrence will be calculated.</p>  <ul style="list-style-type: none"> • Default - This is the default setting. A value of Default means that this field is subject to the image-level definition for this Overflow value. For example, if you select Default and the image properties define anything other than None, Studio uses the image-level overflow definition. • Form - Choose Form if this field appears on a repeating form. This implies there is a different form for each instance of the data. • Formset (Section) - Choose Formset (Section) if the image page segment that contains the fields is repeated within the document, but is not limited to the same form. This means counting the matching image names from the start of the document, without regard to the actual form, determines the value to use in the overflow occurrence calculation. • Image - Choose Image when the repeating page segments are included on the same form. This means the count of the image within the parent form is used in calculating the correct overflow occurrence. • None - Choose None if you want to exclude this field from overflow calculations. This tells Studio that you will handle this field's mapping independent from any overflow that might be specified at the image page segment level. You do not have to set the field Overflow value to None if the image Overflow property is set to None. <p>The default is Default.</p>

If you use the defaults for these fields at the field level, these fields inherit what is defined at the section level. If you choose different settings at the field level, the system uses your choices instead of the section level settings. So, unless you are mixing overflow and non-overflow variables on a section, you can just accept the defaults for the fields. If, however, you are using forced overflow, you do have to define the *record member* which is added in addition to any overflow occurrence calculated.

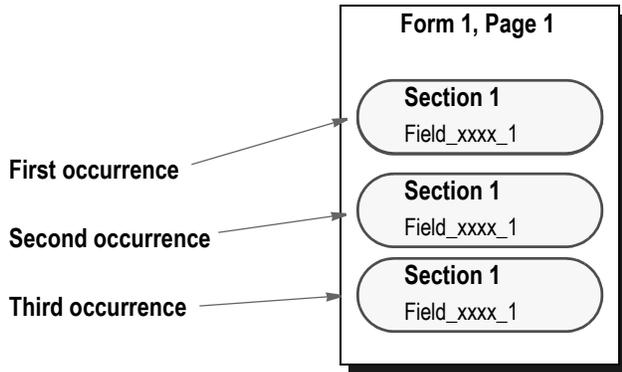
Note If you use this method, be sure to remove the old overflow rules from your AFGJOB.JDT file if you want to see an improvement in performance.

Overflow Examples

Here are some examples of how you can use this overflow method.

Section-level overflow

In this example of section-level overflow, Field_xxx_1 repeats on Section_1.

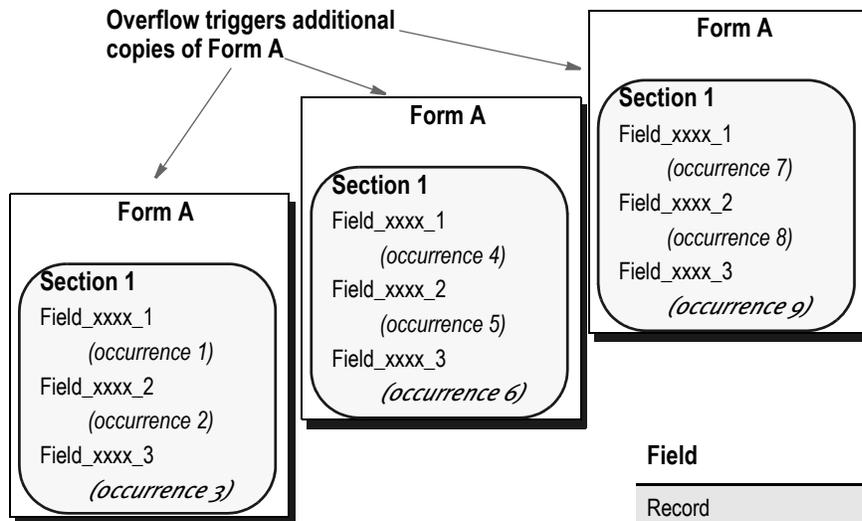


Field	Value
Record	1
Overflow	Section
Overflow Multiplier	1

If you have additional occurrences, the system would create Form 1, Page 2 and so on until all the occurrences are handled.

Form-level overflow

In this example of form-level overflow, the fields that comprise Section 1 (Field_xxx_1, Field_xxx_2, and Field_xxx_3) repeat three times, triggering three copies of Form A.

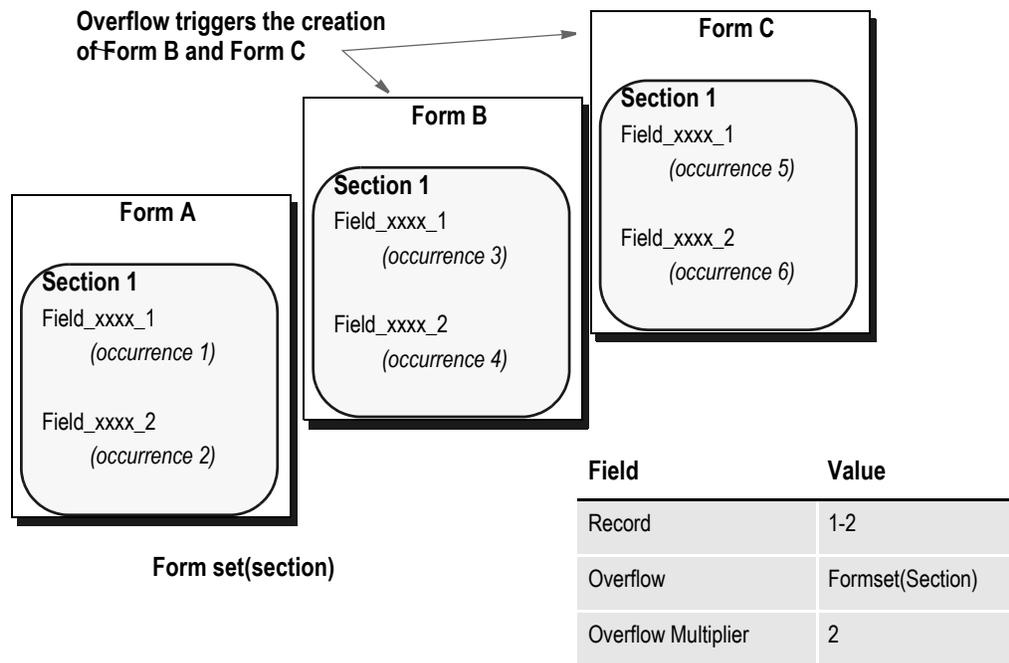


Field	Value
Record	1-3
Overflow	Form
Overflow Multiplier	3

If you have additional occurrences, the system would create additional copies of Form A until all the occurrences are handled.

Form set (section)-level overflow

In this example of form set(section)-level overflow, Section_1, which contains Field_xxx_1 and Field_xxx_2, repeats three times, triggering Forms A, B, and C.



If you have additional occurrences, you must set up forms to handle those additional occurrences.

User Functions

In addition, you can use these user functions:

Function	Description
GetImage	Use this function to return the image's sequence number of like named sections within the current form. For compatibility with the field rule record functionality, all counts are zero (0) based.
GetForm	Use this function to return the form's sequence number of like named forms within the form set.
GetFormSetImage	Use this function to return the section's sequence number of like named sections within the form set.

If you are using XML extract files, you can use these user functions to set up incrementers at the form and section level. Otherwise, Studio handles the use of these functions for you when you enter overflow information on the properties window.

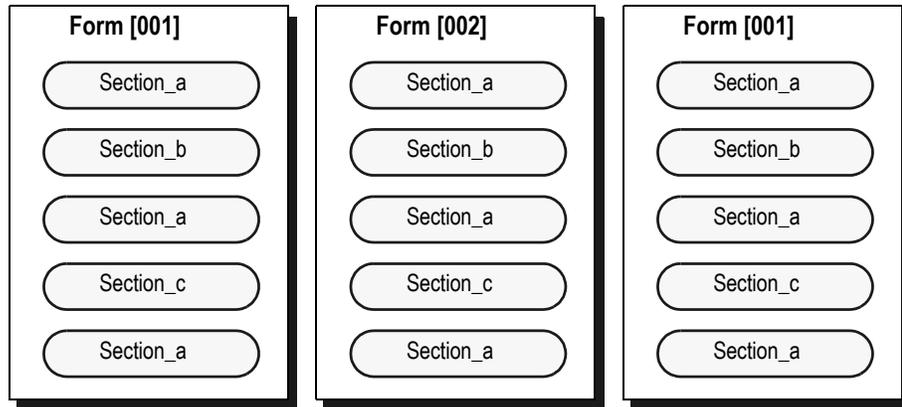
Note These functions correspond to the options you can choose for the Overflow field: Section, Form, Formset(Section).

You can assign these user functions to fields. There are additional options when assigning them to fields:

Option	Description
None	Choose None to indicate the field does not use overflow.
Default	Choose Default to use overflow assigned at the section level.

In the Form manager, you can assign overflow at the section level. The three user functions and the None option are available to the section. Any function set for a section will be used on all overflow fields that are set to Default.

Keep in mind that the value for @GetImage, @GetForm, and @GetFormSet, is not the section or form sequence number, but the section or form sequence number for sections or forms with the same name. For example, if you have a form set with these forms:



These values are returned when using the @GetImage (“Image”), @GetFormsetImage (“Formset(Image)”) and @GetForm:

Form	Section	GetImage	GetFormsetImage	GetForm
001				
	a	1	1	1
	b	1	1	1
	a	2	2	1
	c	1	1	1
	a	3	3	1
002				
	a	1	4	1
	b	1	2	1
	a	2	5	1

Form	Section	GetImage	GetFormsetImage	GetForm
	c	1	2	1
	a	3	6	1
001				
	a	1	7	2
	b	1	3	2
	a	2	8	2
	c	1	3	2
	a	3	9	2

The RunTriggers rule evaluates the form trigger, if the form is triggered. The rule then triggers each of the section triggers before it evaluates the next form trigger. This lets you do regressive form triggers from the section level.

These examples show the rule data entry for an XML file:

```
!/status/summary/staff[**@GETIMAGE**]/inProcess
```

This returns the *inProcess* data element for the X occurrence of the staff section where X is equal to the section's sequence number in the form (zero based) plus the field rule record count.

```
!/status/summary/staff[**@GETFORMSETIMAGE**]/assigned
```

This returns the *assigned* data element for the X occurrence of the staff section where X is equal to the section's sequence number in the form set (zero based) plus the field rule record count.

```
!/status/summary/staff[**@GETFORM**]/onHold
```

This returns the *onHold* data element for the X occurrence of the staff section where X is equal to the form's sequence number (zero based) plus the field rule record count.

```
!/status/summary[**GETFORM**]/staff[**@GETIMAGE**]/closed
```

This returns the *closed* data element for the X occurrence of the summary section where X is equal to the form's sequence number (zero based) plus the field rule record count and the Y occurrence of the staff section where Y is equal to the section is equal to the section's sequence number in the form (zero based) plus the field rule record count.

Overflow Variables

With Studio, overflow variables that require only a single increment per use are created for you. You no longer have to declare an overflow variable for your form sets in the AFGJOB.JDT file.

For example, you can omit these lines from your AFGJOB.JDT file:

```
;SetOvFlwSym;1;DATI0OVF,QDATI0,1;
;SetOvFlwSym;1;MNUM1OVF,QMVNUM,4;
;SetOvFlwSym;1;AGDTDATAOVF,QAGDTD,1;
;SetOvFlwSym;1;CGDECBDOVF,QCGDCB,1;
```

Each of these lines declares a simple overflow variable with a single increment step. When using Studio, these variables are automatically created if you omit them from the AFGJOB.JDT file.

The system creates single increment overflow symbols regardless of whether you are using the legacy or Studio model.

If *all* of your overflow situations can be handled using this overflow methodology, also remove the InitOvFlw and ResetOvFlw rules from the AFGJOB.JDT file. These rules are only necessary if you have an overflow situation that uses the legacy method of handling overflow -- not the method described in this topic. For instance, even if you are only using the legacy method in one scenario, you must still include the SetOvFlwSym, InitOvFlw, and ResetOvFlw rules in your AFGJOB.JDT file.

Note You can also omit the IncOvSym section-level rule.

SETRECIP OPTIONS

The SetRecip Table lets you specify the conditions under which a form or a section will get triggered. Depending upon the requirement, not every field is completed. This table discusses each field:

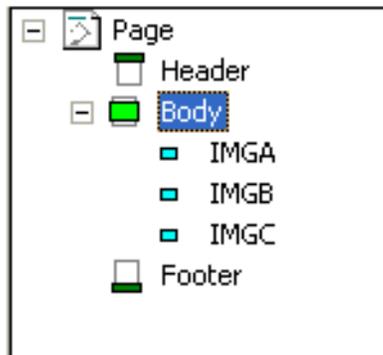
Field	Description
SetRecip Information	
Transaction Codes <TRANSCDS>	By including one or more Transaction codes in this field, you trigger a particular form (or section) only if the extract file record includes that particular Transaction Code.
<RECIPS>	???
Search Mask (Counter) <MASK1>	Lets you set the criteria to determine whether or not a form (or section) belongs in the form set. The criteria allows the Rules Processor to get specific data from the extract file. Adds one form (or section) for every occurrence of the search mask per transaction. The Occurrence Flag must be set to one (1) for the search mask to act as a counter.
Occurrence Flag <OCCURRENCE>	To use the search mask as a counter, the Occurrence Flag (overflow) must be set to one (1). Use with the Search Mask Counter, Records on Different First Section, and Records Per Overflow Section fields. (0=no overflow and 1=overflow) Performance issue flags in this field include: - Master (M) form level trigger flag - Form (F) form level trigger flag - Subordinate (S) section level trigger flag
Records on different first section <RECSPER1ST>	Lets you indicate the number of records that appear on the first section before overflowing to a new section.
Records per overflow section <RECSPERIMG>	Lets you specify the number of records matching the search mask that will fit on the specified overflow section. If the value is zero (0) and the Occurrence Flag is set to one (1) with the Search Mask Counter set, the system automatically sets this flag to one (1) to perform the overflow calculations.
Search Mask (True/False) <CONDITIONAL>	Similar to Search Mask (Counter), but only one form (or section) is triggered, regardless of how many occurrences of the condition exists.
Custom Rule Name <FUNCTIONNAME>	Normally used for custom implementation requirements. Only exceptions are RECIPIF and RECIPCONDITION rules.
Custom Rule Parameters <DATA>	Specifies parameters for the custom rule used in the Custom Rule Name field.
Recipient Information	
Form Level Trigger Recipient Copy Count <RECIPCPYCT>	

Field	Description
Section Level Trigger Recipient Copy Count <RECIPCYCT> Recipient names listed with no check or a check	Defaults the recipient copy of each recipient defined at the form level If recipient was eligible in the form file, it is checked here.

UNDERSTANDING RECIPIENT COUNTS

Although you can specify a recipient count at the form level, technically, recipient counts can only occur at the section level. The count shown at the form level actually represents the first occurrence of a recipient on a section contained within the form.

For instance, suppose you have these sections:



with these possible recipients: Agent, Insured, and Memo, as shown in this table:

IMGA recipients		IMGB recipients		IMGC recipients	
Agent	not eligible	Agent	1 copy	Agent	2 copies
Insured	1 copy	Insured	not eligible	Insured	2 copies
Memo	not eligible	Memo	not eligible	Memo	not eligible

Assuming you have these sections defined in this order: IMGA, IMGB, and IMGC on the form, your form-level recipient list will show the following:

Recipients	
AGENT	1 Copy
INSURED	1 Copy
MEMO	Not eligible

Notice that even though IMGC has a different copy count for Agent and Insured, the form shows the count of the *first occurrence* of that recipient. In this case, Insured first occurs on IMGA. Agent first occurs on IMGB and Memo does not occur on any section. Therefore, the form shows one copy for both Insured and Agent and *not eligible* for Memo, since it is not used anywhere.

When you change the recipient count at the form level, the system scans that form's sections. The system changes the recipient count for any section which shared the same recipient count value as the form to match the new setting defined at the form level. If the section did not share the same recipient value as the form or if that recipient had a different count, the system does not change the value.

Using the example above, assume you are changing the form level definition of Agent as shown here.

Recipients	
AGENT	3 Copies
INSURED	1 Copy
MEMO	Not eligible

The resulting changes at the section level are shown in this table:

IMGA recipients		IMGB recipients		IMGC recipients	
Agent	not eligible	Agent	3 copies	Agent	2 copies
Insured	1 copy	Insured	not eligible	Insured	2 copies
Memo	not eligible	Memo	not eligible	Memo	not eligible

Notice that only IMGB changed to show Agent getting three copies. IMGA did not define the recipient and therefore did not change. IMGC did define the recipient, but had a different count (2) from what was shown at the form level. Therefore, it did not change either.

A value of *not eligible* means the recipient is not defined for a given section. If no sections are using a given recipient, the form level will also show *not eligible* as the current value. In this example, the recipient Memo is not used on any of the sections.

When you change a recipient at the form level that was formerly *not eligible* to have another value, you are adding that recipient to all subsequent sections with the count you specify. For example, assume you define Memo at the form level as shown below:

Recipients	
AGENT	3 Copies
INSURED	1 Copy
MEMO	1 Copy

The resulting change to the sections is shown in this table:

IMGA recipients		IMGB recipients		IMGC recipients	
Agent	not eligible	Agent	3 copies	Agent	2 copies
Insured	1 copy	Insured	not eligible	Insured	2 copies
Memo	1 copy	Memo	1 copy	Memo	1 copy

Since none of the sections previously defined a value for the Memo recipient, all of the sections accepted the new value you assigned at the form level. In effect, all of the sections had the same prior value for this recipient, as shown at the form level and therefore all matched. At this point, if you decide a section should not define this recipient or should have a different count, you can change that section independent of the others.

There is one exception to the matching rule where only those sections that define the same value as shown on the form level will change. This occurs if you should change a recipient to *not eligible* at the form level.

If you set a recipient to *not eligible* at the form level, it does not matter what value a section has for that recipient. All the sections will accept and assign *not eligible* to that recipient, if you assign that value at the form level. Consider this example where Agent is changed at the form level:

Recipients	
AGENT	Not eligible
INSURED	1 Copy
MEMO	1 Copy

If you look back at the prior table for the sections, you will see that only IMGB and IMGC defined the Agent recipient, but both had different counts. The resulting change by assigning *not eligible* at the form level is show in this table:

IMGA recipients		IMGB recipients		IMGC recipients	
Agent	not eligible	Agent	not eligible	Agent	not eligible
Insured	1 copy	Insured	not eligible	Insured	2 copies
Memo	1 copy	Memo	1 copy	Memo	1 copy

Again, note that assigning *not eligible* at the form level for a recipient is an exception that changes all of the sections contained by that form.

Keep in mind...

- The values shown at the form level for each recipient merely represent the first occurrence of that recipient found within the form.
- It is not necessary that all of the sections have the same value for each recipient.
- If you change the first occurrence of a given recipient at the section level, the form level value for that recipient changes to reflect the definition of that first instance.

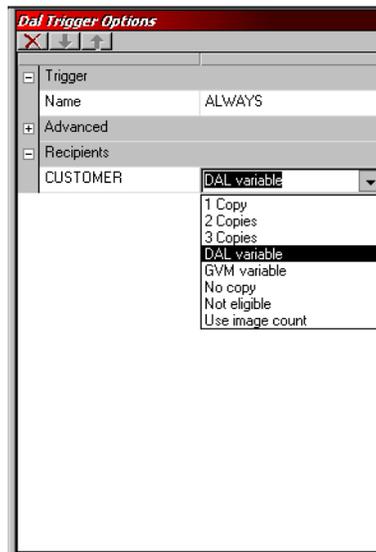
Using Variables to Set Copy Counts

You can use a GVM or DAL variable to set the copy count for sections and forms triggered using the method known as DAL triggers. This makes it possible for the trigger script to assign the count based on user data, or in the case of GVM variables, to get a value directly from the data feed.

The key is that the DAL or GVM variable *must* have the same name as the recipient being mapped. For instance, if you are mapping the *INSURED* recipient and want to use a DAL variable, then the DAL variable must be named *INSURED* as well. Or if you want to use a GVM variable to map the copy count to the recipient *AGENT3*, then the GVM variable must be named *AGENT3*.

If the requested variable type cannot be found with the correct name, a warning message appears and the count is set to one (1).

To use a GVM variable to assign the count, select *GVM variable* on triggers tab as the copy count. Select *DAL variable* if you want to assign the count from a DAL variable. Here is an example:



You must establish the values assigned to the recipient named GVM and/or DAL variables referenced by the triggers prior to their use. You can do this in the trigger script or at any point before the trigger executes. Subsequent values assigned to the recipient named GVM or DAL variables do not affect forms and/or sections that have already been triggered and assigned a value.

Chapter 6

Working with Sections

Studio makes it easy to create and maintain the sections that comprise your forms.

A section (image) is a group of text or graphics or both which make up all or part of a form. You create sections with the Sections option. Each section is stored in a separate file, so you can reuse sections in multiple forms and form sets.

For example, a three-page form with text and graphics printed on both sides of each page, could contain a total of six sections.

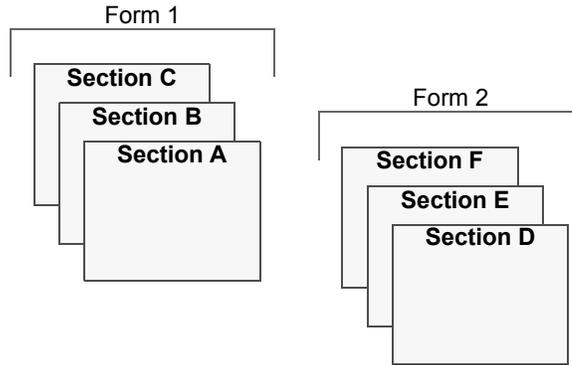
This topic discusses:

- *Overview on page 172*
- *Using the Screen on page 173*
- *Creating a Section on page 189*
- *Opening a Section on page 191*
- *Adding Objects on page 195*
- *Creating Bar Codes on page 197*
- *Adding Bookmarks on page 206*
- *Adding Charts on page 208*
- *Setting Header and Footer Print Options on page 222*
- *Creating Headers and Footers for Multipage Sections on page 224*
- *Defining Paragraphs on page 225*
- *Creating Paragraph Lists on page 234*
- *Defining Tables on page 237*
- *Creating Tables of Contents, Tables of Figures, and Indexes on page 241*
- *Adding Comments to Documaker Objects on page 247*
- *Using the Task List on page 249*
- *Creating Text Labels on page 251*
- *Printing a Form, Section, or Paragraph on page 254*
- *Using Printcommander to Create Sections on page 256*

OVERVIEW

Sections can consist of both static and non-static objects. When saved, a section is stored with the extension of *FAP*. Each form is comprised of a minimum of one FAP file. Typically, however, a single form consists of one or more sections. Since multiple forms and sections make up a form set, you can view and navigate through each form and section individually.

Forms consist of one or more sections



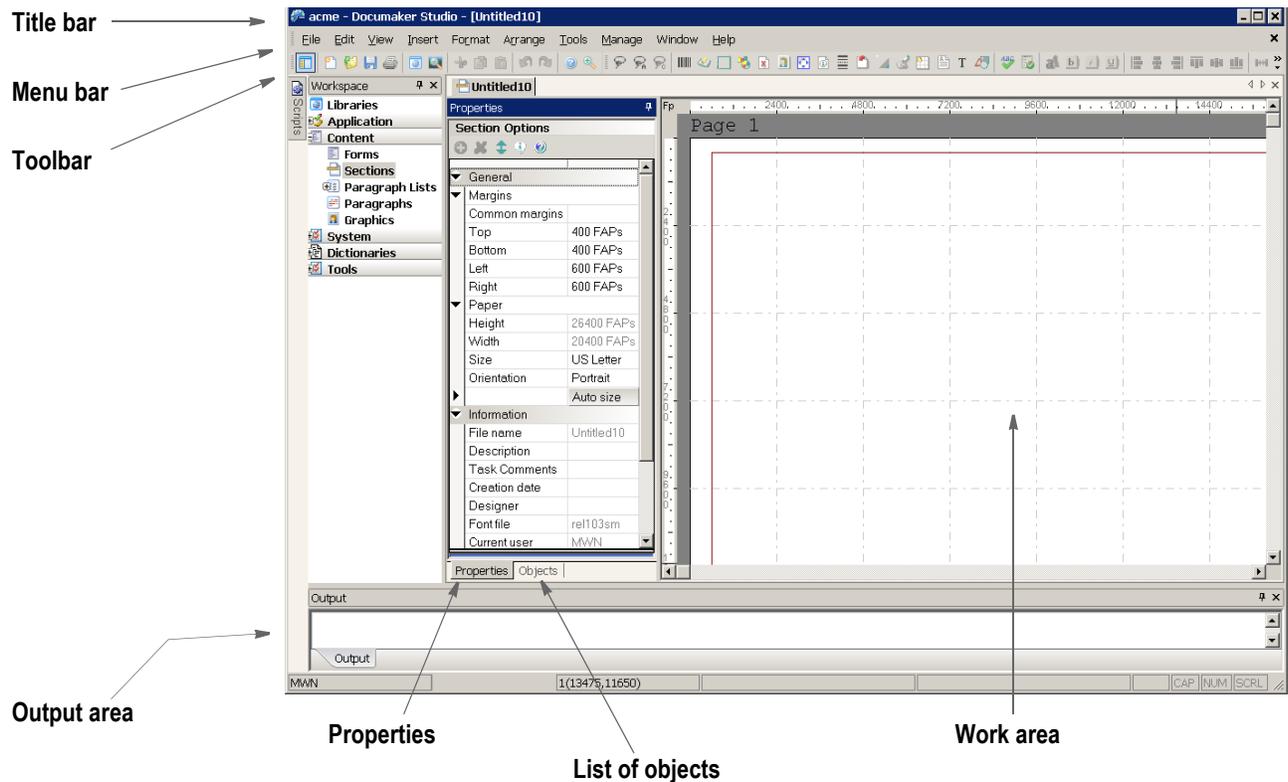
A form may consist of multiple sections, and consequently you can view the varying sections within a form by displaying different pages of the form.

Objects are the individual items which comprise your section. Examples of objects are boxes, bar codes, lines, graphics, and text. All objects have unique attributes within the section. Attributes include items such as position, size, font type, and color.

USING THE SCREEN

Studio places all the tools you need for professional document creation at your fingertips. The screen is your document work area. It is important to become familiar with the general screen layout and parts of the screen. Understanding the screen layout will help you work quickly and efficiently.

The first window that appears when working with sections is shown here.



Item	Description
Title bar	The title bar displays the name of the workspace you have open, followed by Documaker Studio, and then the name of the section you have open.
Menu bar	The menu bar provides the list of available pull-down menus.
toolbar	The toolbar contains a row of icons that provide quick access to common options.
Output area	The Output area is where Studio displays messages to you.
Properties	Click this tab to work with the properties for the section or the selected object within the section.
Objects	Click this tab to see and choose from a list of the objects contained in this section.
Work area	This is where you create the section. Objects are placed in this area as the section is built.

USING THE MENU BAR

This section introduces you to the pull-down menus which include additional options or are only available when you are working with sections. A summary of each of these menu appears below. The menus are listed in the order they appear on the menu bar.

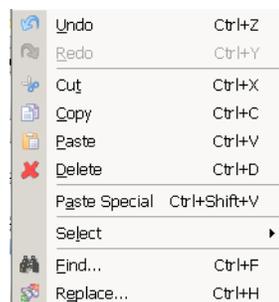
Note For information on the standard menus and menu options which are always available, see *Using System Menus on page 16*.

Menu	Description
Edit	The Edit menu provides options you use as you create and modify sections and the objects that make up sections. You copy, cut, delete, and paste individual objects, or select all objects for editing. You can also delete pages and undo changes from the Edit menu.
View	The View menu controls the appearance of your window as you create or edit sections. It lets you hide or show various toolbars and panes.
Insert	The Insert menu lets you create objects and place them in your section. You can also insert objects by clicking on the object icon on the toolbar.
Format	The Format menu provides options that let you format objects and text and set up the specific formatting properties.
Arrange	The Arrange menu lets you align, space, size, and center objects.
Tools	The Tools menu controls features used to check spelling and grammar on sections and lets you enter the data entry check option. These types of features are used most often when a section is complete or near completion.

Using the Edit Menu

The Edit menu controls modifying sections and objects within sections. You can copy, cut, delete, and paste individual objects or select all the objects for editing. Edit options also let you find and replace text, make global font changes, and undo certain editing functions.

When you select Edit when working with sections, this menu appears:



Option	Description
Undo	Cancels or reverses your last action or choice.
Redo	Reverses your last undo.

Option	Description
Cut	Removes a section of a section and places it on the clipboard.
Copy	Duplicates a selected section of a section and places it on the clipboard.
Paste	Places a section stored on the clipboard into an open section.
Delete	Deletes the selected object.
Paste Special	Only available for text areas, lets you copy from the clipboard.
Select	
Character	Selects a character.
Word	Selects a word.
Line	Selects an entire line.
Paragraph	Selects a paragraph.
Column	Selects a column.
All	Selects everything.
Deselect	Deselects everything selected
Find	Locate words, phrases, or character strings.
Replace	Locate words, phrases, or character strings and, if necessary, replace them.

Using the View Menu

The View menu controls the appearance of your screen as you work on a section. You toggle on and toggle off the display of various features through the use of items on the View menu.

When you select View, this menu appears:



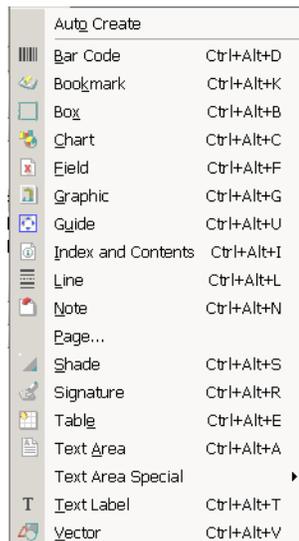
Option	Description
Options	Lets you change general document view options. (Manage, System, Settings, Sections)
Toolbar	Lets you show or hide the Toolbar

Option	Description
Workspace Bar	Lets you show or hide the Workspace pane
Output Bar	Lets you show or hide the Output area
Status Bar	Lets you show or hide the Status bar
Field Bar	Lets you show or hide the Common Fields pane
Extract Bar	Lets you show or hide the Data Extract pane
Navigation	Lets you show or hide the Form Set Navigation pane
Scripts Bar	Lets you show or hide the Scripts pane
Non-printing Tokens	Display or hide formatting symbols such as paragraph markers and tab markers.
Refresh	Lets you update and redisplay the section.
Zoom...	Lets you zoom in or out.

Using the Insert Menu

The Insert menu controls the creation and placement of objects in your section. Boxes, bar codes, lines, and other types of objects are drawn and positioned in your section. You can also import a page, whether it is blank or full of text. You can select to create an object from the Insert menu or you select to create an object from the toolbar.

When you select Insert, this menu appears:



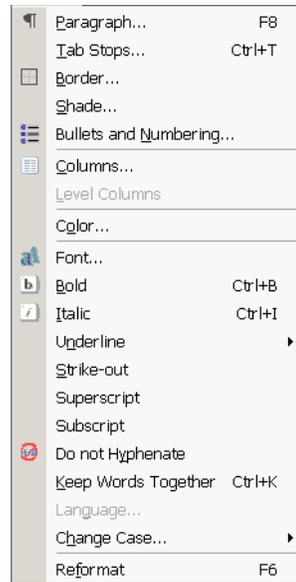
Option	Description
Autocreate	Lets you create objects repeatedly without entering the object's options.
Bar Code	Lets you insert a bar code.

Option	Description
Bookmark	Used to add bookmarks to FAP files. A bookmark defines the text you want to print in the table of contents. A level number (TOC1, TOC2, and so on) can be used to determine the formatting to use. You specify whether the bookmark is to be used by a table of contents, table of figures, or index.
Box	Lets you insert a box.
Break	Lets you insert a a break into a text area.
Chart	Starts the Chart wizard so you can insert a chart.
Field	Lets you insert a variable field.
Graphic	Lets you insert a graphic.
Guide	Lets you create a guide box, which you can use to help you align objects.
Index and Contents	Lets you insert a add an index or a table of contents or figures.
Line	Lets you insert a line.
Note	Lets you insert a note.
Page	Lets you insert a page.
Shade	Lets you insert a shaded area.
Signature	Lets you insert a signature.
Table	Starts the Table wizard so you can create a table.
Text Area	Lets you insert a text area.
Text Area Special	Activates text area draw to insert from a file or the clipboard.
Text Label	Lets you insert a text label.
Vector	Lets you insert a vector.

Using the Format Menu

The Format menu controls formatting functions associated with the section and the objects in the section. You can view or change the properties for the page, section, or individual objects from this menu.

When you select Format, this menu appears:



Option	Description
Paragraph	Text area feature – allows for formatting of paragraph
Tab Stops	Text area feature – allows for formatting of tab stops
Border	Text area feature – allows for formatting of border
Shade	Text area feature – allows for formatting of shading
Bullets and Numbering	Text area feature – allows for formatting of bullets and numbering used in a text area
Columns	Text area feature – allows for formatting of columns
Level Columns	Level the text in the columns.
Color	Change the color of the selected object. Highlight the object, then choose this option.
Font	Select or change fonts.
Bold	Bold the selected text.
Italic	Italicize the selected font.
Underline	Underline the selected text.
Strike-out	Strike-out the selected text.
Superscript	Turns the selected text into superscript.
Subscript	Turns the selected text into subscript.
Do not Hyphenate	Prevents hyphenation of text.
Keep Words Together	Keeps words on the same line.
Language	Select or change the language.

Option	Description
Change Case	Changes the case of the selected text.
Reformat	Reformat the text. Use after you have made significant formatting changes

Using the Arrange Menu

The Arrange menu controls options that let you align, space, size, and center the objects you are working with

When you select Arrange, this menu appears:

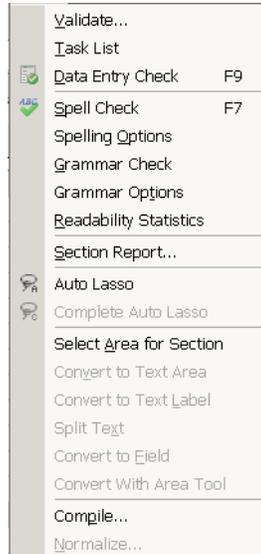


Option	Description
Align	
Lefts	Aligns the left edges of the selected objects.
Centers	Aligns the centers of the selected objects.
Rights	Aligns the right edges of the selected objects.
Tops	Aligns the tops of the selected objects.
Middles	Aligns the middles of the selected objects.
Bottoms	Aligns the bottoms of the selected objects.
Space	
Evenly Across	Spaces the selected objects evenly across the section.
Evenly Down	Spaces the selected objects evenly down the section.
Custom	Displays the Custom Spacing window so you can specify custom spacing intervals.
Make Same Size	
Height	Makes the selected objects the same height.
Width	Makes the selected objects the same width.
Both	Makes the selected objects the same height and width.
Center in Page	
Horizontal	Centers the selected objects horizontally on the page.
Vertical	Centers the selected objects vertically on the page.

Using the Tools Menu

The Tools menu controls options often used when your section is complete or near completion. Spell check and Data Entry check are some of the features available.

When you select Tools, this menu appears:

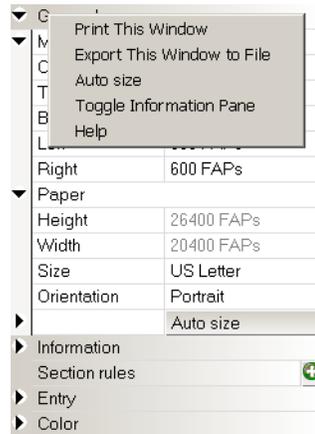


Option	Description
Validate	Lets you validate your section.
Task List	Displays the task list for the selected object.
Data Entry Check	Activates the section check feature. Lets you test the data entry and navigation rules you assigned to variable fields.
Spell Check	Turns on the spell check feature. Lets you check the spelling of all text in your current section.
Spelling Options	Lets you specify spell check options.
Grammar Check	Gives you the option as to whether you wish to check the grammar on the entire document or only items that you have selected.
Grammar Options	Let you pick from the grammar checking settings that you want to use.
Readability Statistics	Generates readability statistics for the text in the selected objects.
Section Report	Generate a report about the current section. You can view and print the report.
Auto Lasso	Automatically groups text labels that are likely to be paragraphs.
Complete Auto Lasso	Converts the text labels selected using Auto Lasso into paragraphs.
Select Area for Section	Lets you select an area in the current section and convert it to a new section.
Convert to Text Area	Combines selected text labels or text areas into a single text area.
Convert to Text Label	Combines selected text labels into a single text label.

Option	Description
Split Text	Splits a single text area into individual text labels.
Convert to Field	Converts a text label into a variable field.
Convert with Area Tool	Converts selected objects or a selected area to a new section.
Compile	Lets you compile your section file for a particular printer. Compile creates a section file for a PCL, AFP, PostScript, or Metacode printer.
Normalize	Lets you convert a FAP file into a normalized Metacode or AFP file.

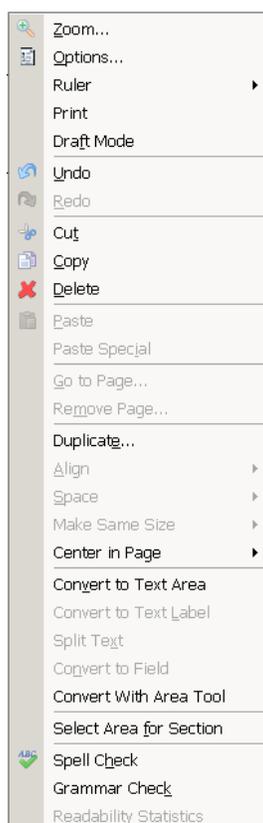
USING THE RIGHT CLICK MENUS

When working with section properties, you can right click to see the following menu:



Option	Description
Print this Window	Use to print the section.
Export This Window to File	Export the contents of this window to a CSV (comma separated value) file.
Auto size	Use to automatically size the section.
Toggle Information Pane	Display or hide the information pane.
Help	Display Help information.

When working with a section, you can right click to see this menu.



Option	Description
Zoom	Lets you see a larger or smaller version of the section.
Options	Opens Sections settings so you can changes document view and other options.
Ruler	Lets you show or hide horizontal and vertical rulers and set the unit of measure on those rulers.
Horizontal	Select to display the horizontal ruler. Select again to remove the ruler.
Vertical	Select to display the vertical ruler. Select again to remove the ruler.
FAPs	Select to use FAP units (2400 per inch) on the ruler.
Inches	Select to use inches on the ruler.
Centimeters	Select to use centimeters on the ruler.
Picas	Select to use picas on the ruler.
Points	Select to use points on the ruler.
Print	Prints a copy of the section.

Option	Description
Draft Mode	Lets you review all the sections sequentially. In draft mode, the sections appear left-justified in the same order as they appear in the list area, not how they are positioned on the form. If you choose to print while in draft mode, the items print in that order. You cannot drag a section to position it while draft mode is on. You can change the sections position using the properties, but you will not see the effect of your changes until you turn of draft mode.
Undo	Cancel or reverse your last action or choice.
Redo	Repeat the most recent change.
Cut	Saves the changes you have made
Copy	Removes part of a section and places it on the clipboard.
Delete	Places what's in the clipboard into an open section.
Paste	Duplicates part of a section and places it on the clipboard.
Paste Special	Duplicates part of a section and places it on the clipboard.
Go to Page	Jumps to a specific page in the section.
Remove Page	Removes a page in the section.
Duplicate	Deletes the selected object.
Align	Aligns according to section margins or a specific object.
Space	Places selected objects at specified intervals either vertically or horizontally in relation to each other
Make Same Size	Lets you make the selected objects the same size
Center in Page	Lets you center the selected object horizontally or vertically on the page.
Convert to Text Area	Combines selected text labels or text areas into a single text area.
Convert to Text Label	Combines selected text labels into a single text label.
Split Text	Splits a single text area into individual text labels.
Convert to Field	Converts a text label into a variable field.
Convert with Area Tool	Converts selected objects or a selected area to a new section.
Select Area for Section	Lets you select an area in the current section and convert it to a new section.
Spell Check	Checks spelling
Grammar Check	Checks grammar.
Readability Statistics	Provides statistics on the readability of the selected text.

USING THE TOOLBAR

The toolbar is useful because it serves as a quicker route for performing some functions that may be listed on a menu.

Standard toolbar icons

Shown below are the toolbar icons that are always available. The icons are listed as they appear, from left to right.



Icon	Name	Description
	Workspace	Shows or hides the workspace pane.
	New	Displays the Create New window so you can create an object.
	Open	Displays the Open File window so you can open or create a file.
	Save	Saves the open file.
	Print	Prints the current object.
	Libraries	Opens the master library.
	Test Scenarios	Displays the Open File window so you can select a test scenario.
	Cut	Removes an object and places it on the clipboard.
	Copy	Copies an object and places it on the clipboard.
	Paste	Places an object from clipboard onto the current file.
	Undo	Reverses your last action
	Redo	Reverses last undo.
	Help	Displays the Help window
	Zoom	Zoom in

Icon	Description
	Insert a note
	Insert a shaded area
	Insert a signature
	Insert a table
	Insert a text area
	Insert a text label
	Insert a vector drawing

These icons let you edit text in objects. See also the Format menu.



Icon	Description
	Lets you check spelling
	Lets you run a data entry check
	Opens the Select Font window so you can change fonts.
	Changes the font of the selected text to a bold version of that font.
	Changes the font of the selected text to an italic version of that font.
	Underlines the selected text.

These icons let you align objects. See also the Arrange menu.



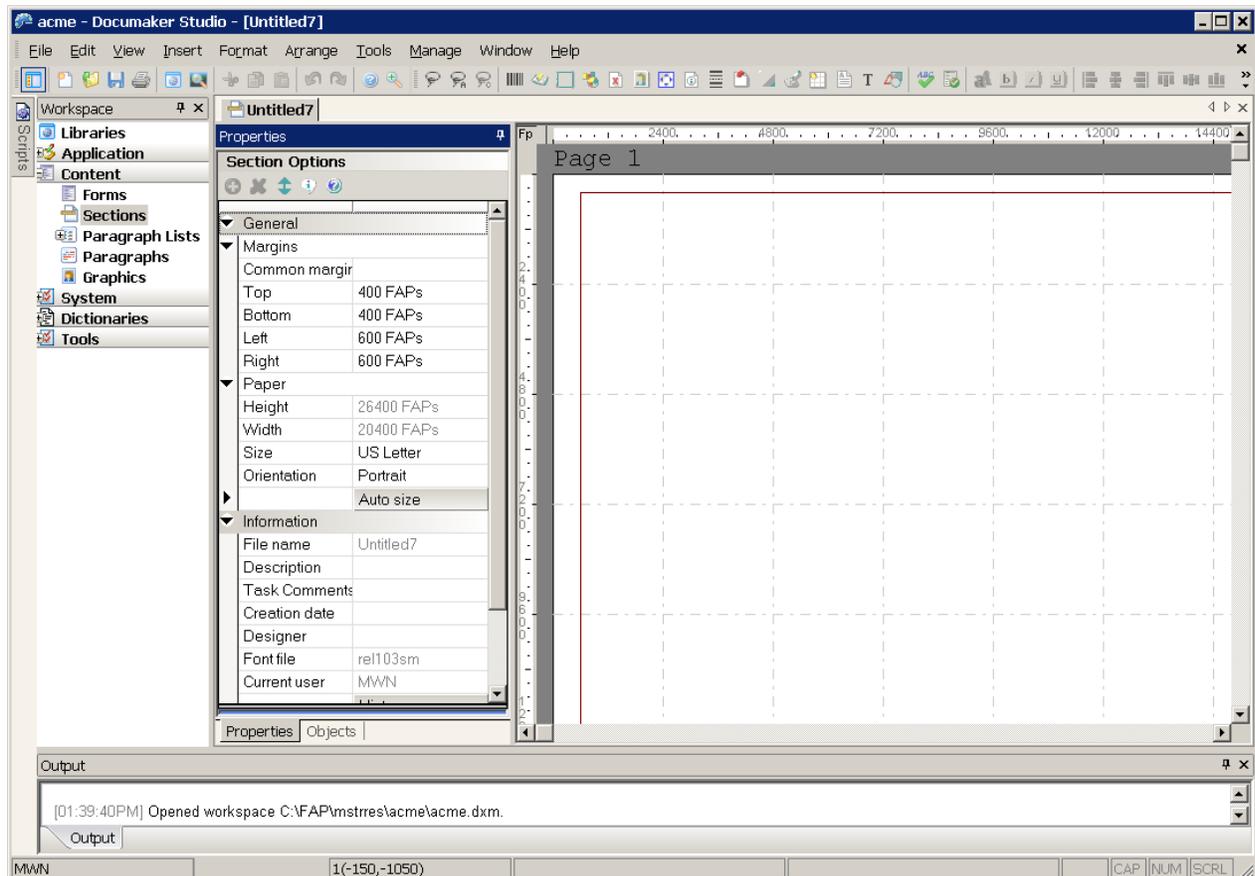
Icon	Description
	Aligns the left edges of the selected objects.
	Aligns the centers of the selected objects.
	Aligns the right edges of the selected objects.
	Aligns the tops of the selected objects.
	Aligns the middles of the selected objects.
	Aligns the bottoms of the selected objects.
	Space selected items evenly across.
	Space selected items evenly down.
	Displays the Custom Spacing window so you can specify custom spacing intervals.
	Make the selected objects the same height.
	Make the selected objects the same width.
	Make the selected objects the same height and width.
	Centers the selected objects horizontally on the page.
	Centers the selected objects vertically on the page.

CREATING A SECTION

You can create a section several ways, such as

- From the toolbar
- Using the File, New option

Studio then opens a new, untitled section:



Modify the section options as necessary and begin adding the objects that will comprise your section. See *Adding Objects on page 195* for more information.

Pagination

Keep in mind that the positioning information you establish in Studio is reapplied to the form each time changes in a text area cause the dimensions of the section to change. You can, however, use the AutoPagination option to disable automatic re-pagination:

```
< Control >
    AutoPagination = Yes
```

Option	Description
AutoPagination	Enter No if you do not want the system to automatically re-paginate when image dimensions change. The default is Yes.

Keep in mind...

- The system will honor positioning information designed into the form via Studio. Positioning information is stored via SetOrigin rules.
- If a form consists of multiple sections on a page, but those sections comprise more space than defined for the page size, the system automatically paginates that page and moves the sections that did not fit to a new page.
- If a section grows to push another section such that its positioning rule causes it to encroach on a defined footer or the bottom of the page, that section is moved to the next page and the entire form will be have the SetOrigin rules reapplied.
- When designing a form, avoid having a footer section that uses a relative position. This ultimately means there can only be one section on the page that is not a header or footer. Footers, typically should be placed using a rule that makes sure it has a relationship to the bottom of the page.
- When sections shrink (due to text area shrinking) a section from the next page may be brought back to the current page. In other words, sections can not only flow to the next page, but they can come back when space allows.
- Remember that the positioning (SetOrigin) defined in the form is applied. So although there may appear to be a space large enough to hold a section, you also have to account for any additional adjustments applied by the SetOrigin rules.

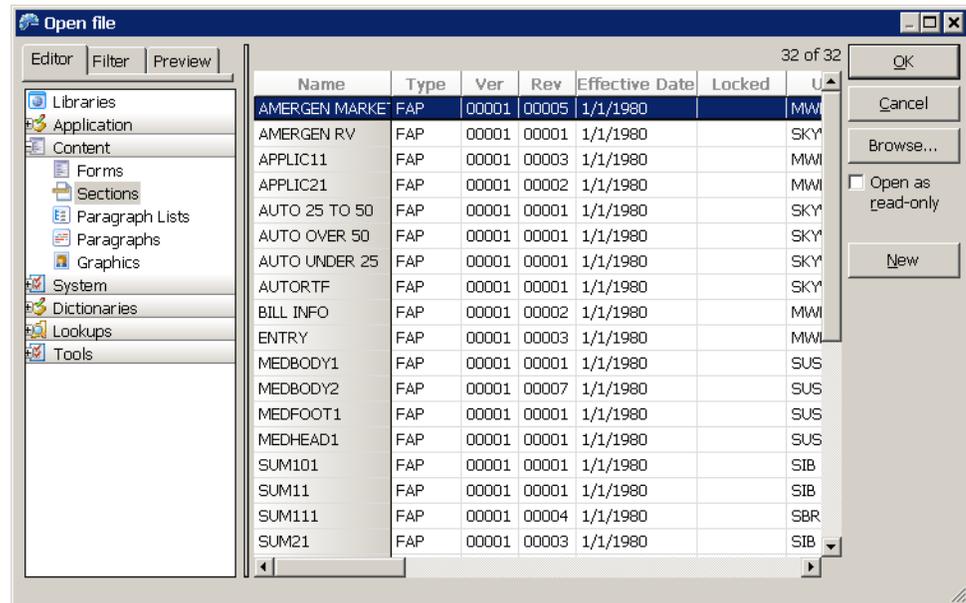
For example, suppose you have a section that is two inches in height, but the SetOrigin rule for that image specifies a relative placement 1/2 inch from the previous section. In this case, 2 1/2 inches of space is required for that section to fit on the page. If there is less than 2 1/2 inches remaining before encountering a footer section or the bottom of the page layout, then that section moves to the next page.

- Sections can only flow to and from pages that were created during pagination. If a page was specifically designed into the form via Studio, then no sections will move onto that page from a prior page. Sections can only move to or from pages that were created by overflowing their defined page.

OPENING A SECTION

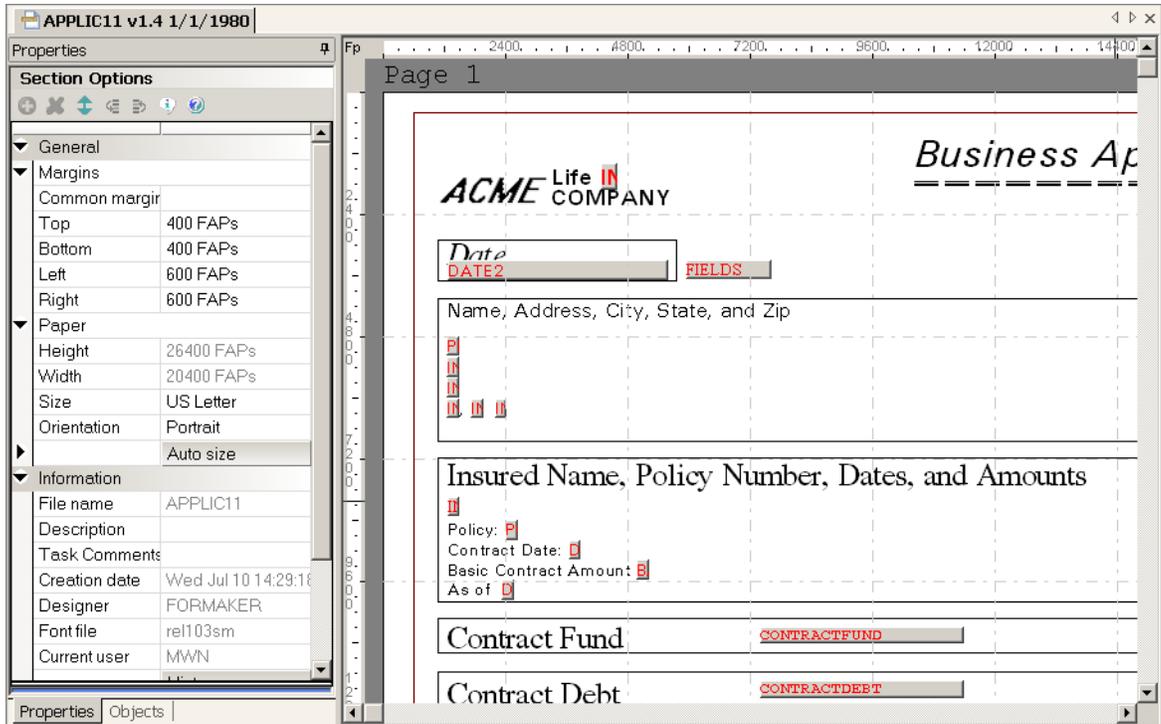
Follow these steps to select and open a section:

1. Double-click sections. The Open File window appears. You can also right click on Sections and choose Open or Check out.



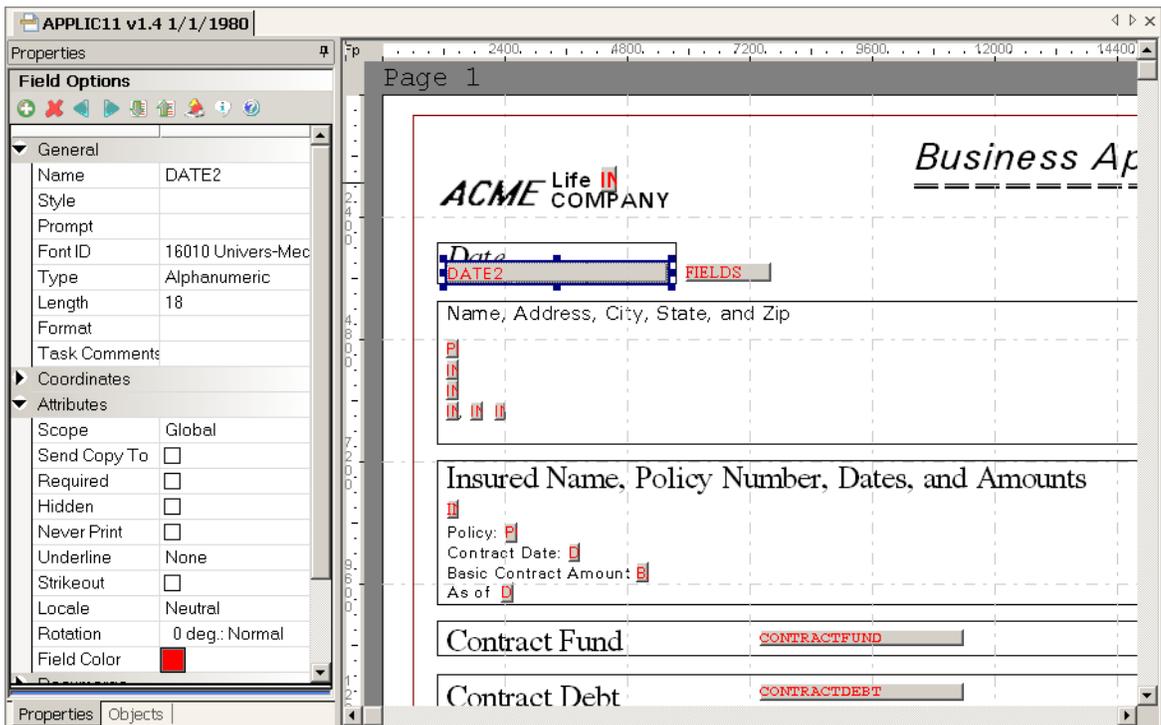
2. Highlight the section you want to edit or view and click Ok. The section and its properties appear.

Note If you check Open as read only and then click Ok, the file will only be available for viewing and someone else at another workstation would still be able to check it out.



Studio shows you the section and its properties.

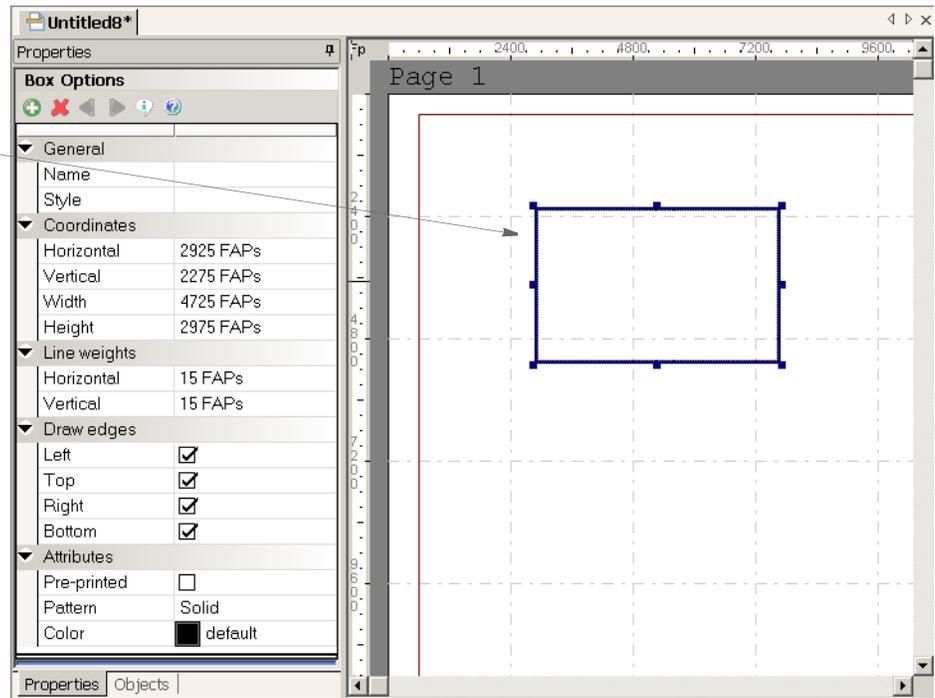
For example, if you click on a field, the properties are shown for that field. If you click on a box or static text label, the property panel changes to show the attributes for those objects. Here is an example of field properties:



Here is an example of how you edit box properties:

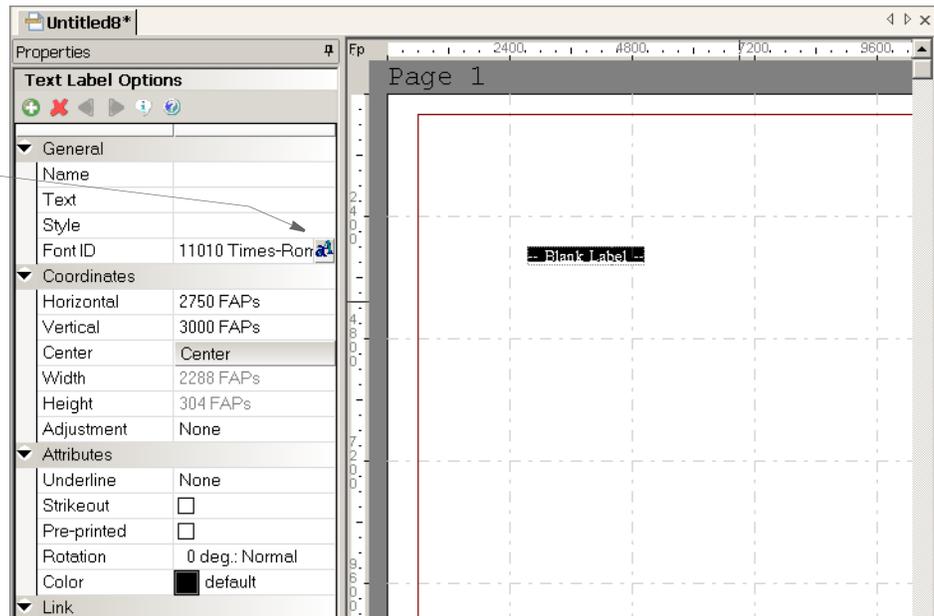
Here is the box. Notice that it is selected.

Here are the properties for that box.



Here is an example of how you choose a font for a static text label:

Click here to choose a font.
The list of fonts is defined in
your FXR file.



Creating new fields and other objects on your section is as simple as clicking on the menu or toolbar to select what type of item you want to create and then using the mouse to designate the location on the section.

When you create a new object, the property panel changes to reflect the current settings for your new object and lets you modify those settings to complete your definition.

3. Once you have completed the changes to a section, you can save your work and check the section back into the library for other users to access.

Working with Embedded Sections

Few other document composition tools use sections as building blocks for forms. In most, you work on individual forms as a whole. Since Studio users sometimes import content from other tools, Studio provides a way to break imported or converted forms into sections. These sections are called *embedded sections* and are maintained using Form or Template manager.

Embedded sections provide a way to edit the content of a form as though it was a section. You can also *unembed* an embedded section if you want to replace it with another section or make that content available for use on another form.

For more information on using embedded sections, see *Using Embedded Sections* on page 127.

ADDING OBJECTS

Adding objects to a section is easy in Studio. You simply open a section or create a new section, then follow the steps below. You can select the type of object you want to add two ways:

- From the Insert menu
- From the toolbar

From the Insert menu

Choose the type of object you want to add and draw it on your section. Then move it where you want it and adjust the properties as necessary.

From the toolbar

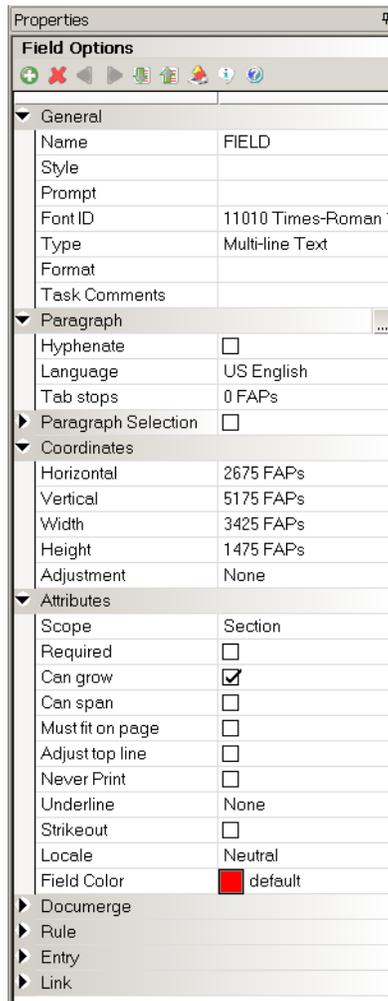
Use these icons:

Click	To
	Insert a bar code
	Insert a bookmark
	Insert a box
	Insert a chart
	Insert a field
	Insert a graphic
	Insert a guide*
	Insert an index, table of contents, or table of figures
	Insert a line
	Insert a note*
	Insert a shaded area
	Insert a signature
	Insert a table
	Insert a text area

Click	To
	Insert a text label
	Insert a vector drawing

Note All section objects have default settings. You can override these settings by using the Manage, System, Settings option. Select Options by Group and scroll down to the Default groups, such as DefaultBarCode or DefaultBox.

Once you select the object and position it in your section, use the properties to customize the object to meet your needs. Here is an example of the properties for a variable field:



CREATING BAR CODES

Studio lets you add the following types of bar codes to a section:

- Codabar Compressed
- Codabar Standard
- Code 39 1:1:2:2
- Code 39 1:1:3:3
- Code 39 1:2:4:5
- Code 39 3:1 Mod 43 Check Digit
- Code 93
- Code 128 Automatic Add-On
- Code 128 Start Code A
- Code 128 Start Code B
- Code 128 Start Code C
- Data Matrix. For more information, see *Using Data Matrix 2-D Bar Codes* on page 199.
- EAN-13 Digit
- EAN-8 Digit
- Intelligent Mail Bar Code (4-State Customer Bar Code). For more information, see *Generating a USPS Intelligent Mail Bar Code (4-State Customer Bar Code)* on page 204.
- Interleaved 2 of 5 1:1:3:3
- Interleaved 2 of 5 Mod 10
- MSI Mod 10 Check Digits
- MSI Mod 10/10 Check Digits
- MSI Mod 11/10 Check Digits
- MSI no Check Digit
- PDF417. For more information, see *Using PDF417 Bar Codes* on page 202.
- PLANET Code® bar codes. For more information, see *Inserting PLANET Code Bar Codes* on page 203.
- UPC 2 Digit Add-On
- UPC 5-Digit Add-On
- UPC-A 1:2:3:4 11 Digit
- UPC-EO Zero Suppress 10 Digit
- UPC-EO 6 Digit

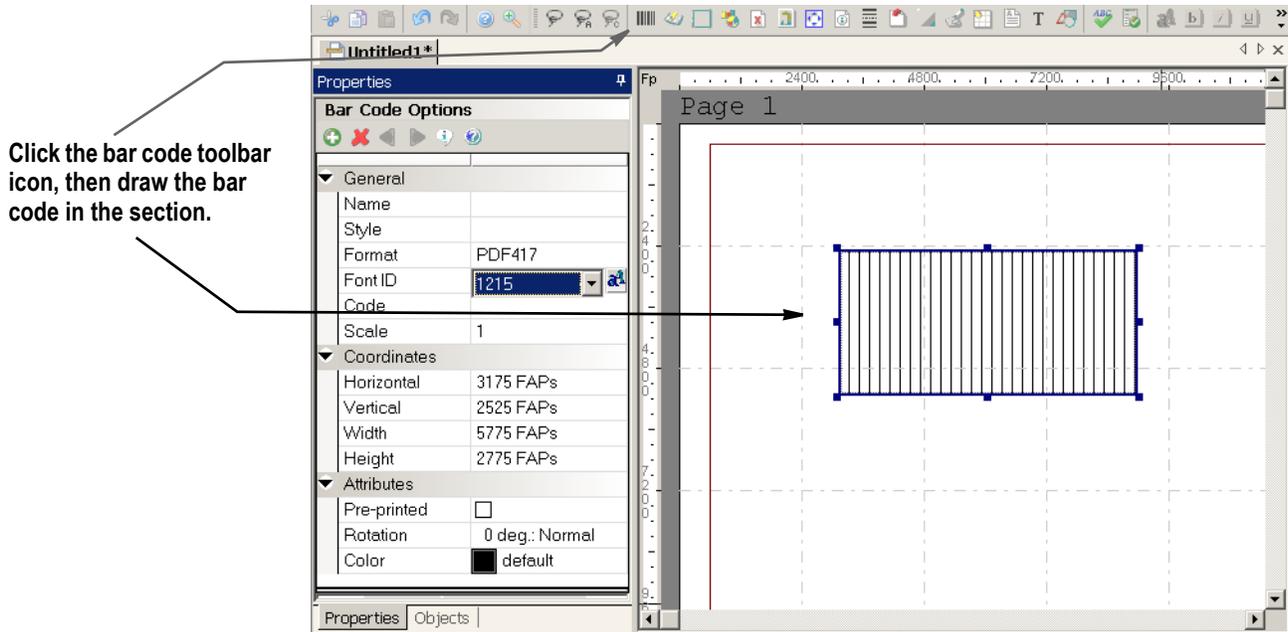
- UPC-EI 6 Digit
- ZIP Code (can be 5, 9, or 11 digits)

There are several ways to create a bar code. Here is a summary of the approaches:

ADDING A BAR CODE OBJECT

Follow these steps to add a bar code object:

1. Add a bar code object to your section.



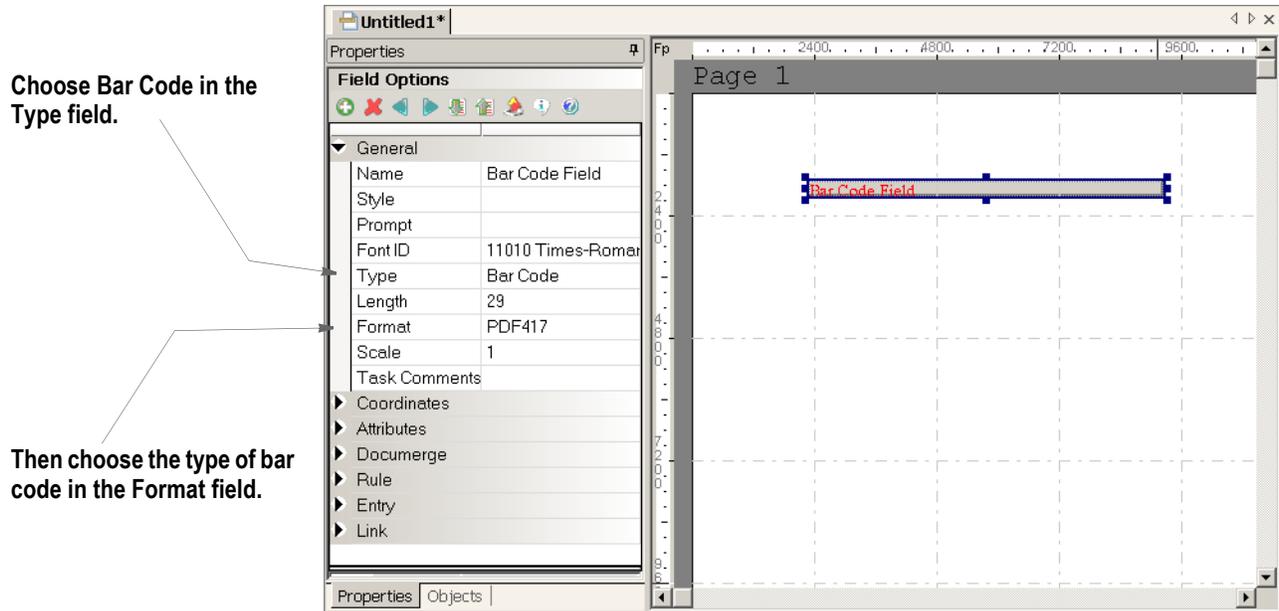
2. Specify the property information for the bar code. You can choose from numerous bar code types in the Format field. Be sure to specify the font and enter the data.

Adding a Bar Code as a Field

Follow these steps to add a bar code as a field:

1. Add a field to your section.
2. As you specify the property information for the field, select Bar Code in the Type field and then choose from numerous bar code types in the Format field. Be sure to specify the font and enter the data.

Here is an example:



Using Data Matrix 2-D Bar Codes

A Data Matrix bar code consists of black and white squares arranged in either a square or rectangular pattern. Depending on the size of the bar code, you can encode up to two kilobytes of text or raw data.

You can use the Data Matrix bar code with printer finishing equipment, such as equipment from manufacturers like Gunther or Pitney Bowes. Here is an example of a Data Matrix 2-D bar code:



Note While the maximum number of alphanumeric characters for some symbol sizes, such as 88 x 88, in the Data Matrix specification can exceed 1024, the maximum number of alphanumeric characters for a variable field in a Documaker section (FAP) is 1024. So these larger symbol sizes are effectively restricted to 1024 characters.

The following table shows the complete list of Data Matrix symbol sizes, the number of regions in that size, and the maximum number of numeric characters stored in it.

With ASCII encodation, two numerics (1, 2, 3, 4, and so on) can be encoded into a single *data code word*, while a single alphabetic character or symbol (A, B, C, @, #, \$, and so on) requires a single data code word. So, if you are storing only numbers in the bar code, you can store more information than if you were storing alphabetic characters.

For example, the 12 x 12 symbol size allows for five data code words. This means you can encode 10 numeric or five alphabetic characters or symbols or some combination, such as eight numerics and one alphabetic character or symbol.

Data Matrix Bar Code

Symbol Size	Regions	Max Numeric	Data Code Words
10 x 10	1	6	3
12 x 12	1	10	5
14 x 14	1	16	8
16 x 16	1	24	12
18 x 18	1	36	18
20 x 20	1	44	22
22 x 22	1	60	30
24 x 24	1	72	36
26 x 26	1	88	44
32 x 32	4	124	62
36 x 36	4	172	86
40 x 40	4	228	114
44 x 44	4	288	144
48 x 48	4	348	174
52 x 52	4	480	204
64 x 64	16	560	280
72 x 72	16	736	368
80 x 80	16	912	456
88 x 88	16	1,152	576
96 x 96	16	1,392	696
104 x 104	16	1,632	816
120 x 120	36	2,100	1050
132 x 132	36	2,608	1304
144 x 144	36	3,116	1558
8 x 18	1	10	5
8 x 32	2	20	10
12 x 26	1	32	16
12 x 36	2	44	22

Data Matrix Bar Code

Symbol Size	Regions	Max Numeric	Data Code Words
16 x 36	2	64	32
16 x 48	2	98	49

Data encoding

Data mapped to the bar code symbol is encoded using ASCII encodation. The error-correcting code ECC 200 is used so damaged bar code symbols can still be read.

Fonts

Documaker draws the Data Matrix bar code using fonts instead of graphic commands. The fonts are listed below and referenced in the REL113.FXR and REL113SM.FXR font cross-reference (FXR) files. The font IDs for the Data Matrix fonts are numbered 13504, 13505, and 13506.

The TrueType, PostScript, PCL, AFP (240 and 300 DPI), and Metacode fonts you need to produce the bar code are included in version 11.3 and listed below:

Size	Font name
TrueType	
All sizes	dm_____.ttf
Postscript	
All sizes	dm_____.pfb
PCL	
4 point	fpdmn4.pcl
5 point	fpdmn5.pcl
6 point	fpdmn6.pcl
AFP 240 DPI	
4 point	x0dadmn4.fnt, c0fadmn4.240 (Coded Font, Character Set)
5 point	x0dadmn5.fnt, c0fadmn5.240
6 point	x0dadmn6.fnt, c0fadmn6.240
AFP 300 DPI	
4 point	x0dadmn4.fnt, c0fadmn4.300 (Coded Font, Character Set)
5 point	x0dadmn5.fnt, c0fadmn5.300
6 point	x0dadmn6.fnt, c0fadmn6.300
Metacode	

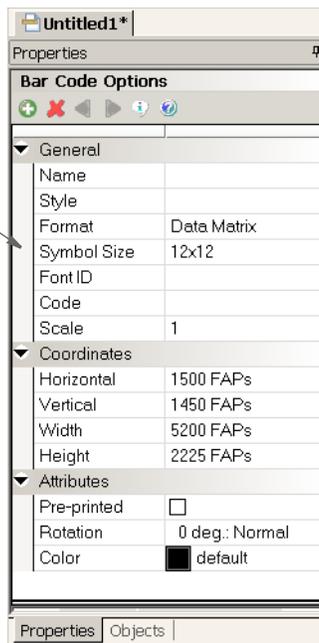
Size	Font name
4 point	fxdmn4.fnt Rotations: f9dmn4.fnt, f1dmn4.fnt, f2dmn4.fnt (90, 180, 270)
5 point	fxdmn5.fnt Rotations: f9dmn5.fnt, f1dmn5.fnt, f2dmn5.fnt
6 point	fxdmn6.fnt Rotations: f9dmn6.fnt, f1dmn6.fnt, f2dmn6.fnt

Adding a Data Matrix bar code

In Studio, you can place a Data Matrix bar code in a field by following these steps:

1. Insert a field in a section. For the field type, choose Bar Code.
2. Choose Data Matrix in the Format field.
3. Select the symbol size.

Select the symbol size.



4. In the Font ID field, select one of the Data Matrix bar code fonts (13504, 13505, or 13506) which are included with Documaker.
5. Use the Move_It rule to map data to this field.

Using PDF417 Bar Codes

A PDF417 bar code can contain any type of information. For instance, the New York State Insurance Department’s (NYSID) regulation requires PDF417 bar codes on driver ID cards. To add a PDF417 bar code, follow these steps:

Note The REL113.FXR font cross-reference file references the PDF417 fonts. You can select this FXR file using the Manage, System, Settings, Resource Path Setup option.

If you are a system supervisor, you can also import the PDF417 bar code fonts into an FXR. For example, choose the Manage, System, Fonts option. Then choose the Fonts, Import, option and use the Import Fonts wizard to import the PDF417 fonts. Use this table to select the fonts for your printer type:

	240 DPI 9x12	300 DPI 9x12	240 DPI 2x16	300 DPI 2x16
AFP	X0P09X12.FNT	X0P09X12.FNT	X0P12X16.FNT	X0P12X16.FNT
Metacode				
0 degrees	P09X12.FNT	P12X16.FNT	P12X16.FNT	P15X20.FNT
90 degrees	J09X12.FNT	J12X16.FNT	J12X16.FNT	J15X20.FNT
180 degrees	I09X12.FNT	I12X16.FNT	I12X16.FNT	I15X20.FNT
270 degrees	L09X12.FNT	L12X16.FNT	L12X16.FNT	L15X20.FNT
PCL	P09X12.PCL	P12X16.PCL	P12X16.PCL	P15X20.PCL
PostScript	PDF417__.PFB	PDF417__.PFB	PDF417__.PFB	PDF417__.PFB
PDF	PDF417__.TTF	PDF417__.TTF	PDF417__.TTF	PDF417__.TTF

1. Insert a field in a section. For the field type, choose Bar Code.
2. Choose PDF417 in the Format field.
3. In the Font ID field, select one of these font IDs: 911, 912, 1215, or 1216.
4. Enter the rest of the Bar Code properties as necessary.

Note The minimum size for a PDF417 bar code is 1.250" (width) x 0.583" (height). If you insert a PDF417 bar code or a PDF417 bar code field and attempt to specify either a width or a height that is less than the minimum size, Studio resizes the bar code to the minimum size.

Inserting PLANET Code Bar Codes

You can add PLANET Code® bar codes to your forms. These bar codes, along with the Confirm® service offered by the United States Postal Service (USPS), let you track your mail electronically.

You uniquely identify and track your mail by placing the PLANET Code bar code on the front of the mail piece. As the mail piece moves through the postal system, USPS sorting equipment notes the processing facility, operation number, and processing date and time. This information is returned to you in the form of a response file.

For more information, see the PLANET Codes web site:

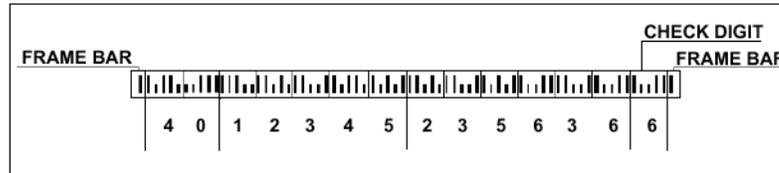
www.planetcodes.com

The PLANET Code is essentially an inverse image of the POSTNET bar code and is defined as follows:

Like POSTNET Codes, PLANET Codes have 14 digits and consist of tall and short bars. PLANET symbology is the inverse of POSTNET because each POSTNET digit has a combination of two tall and three short bars and each PLANET digit had three tall and two short bars.

All PLANET bar codes include a five-bar checksum digit or correction character. This digit is always the number which, when added to the sum of the other digits in the bar code, results in a total that is a multiple of 10.

Here is an example of the PLANET Code bar code:



You can download a true type font from the USPS at:

www.planetcodes.com/confirm/common/planetFonts.ttf

In addition:

- The first two digits comprise an ID Confirm Service number, such as 4 which indicates First-Class Mail Letters.
- The next five digits comprise a Subscriber ID assigned by a post office.
- The next six digits comprise the Mailing ID which is assigned by the system and identifies each mail piece in a batch cycle.
- The last digit is a Check Digit which is calculated based on the sum of the previous 13 digits. For example, if the sum of the previous 13 digits is 44, the Check Digit would be 6 ($44 + 6 = 50$ which is divisible by 10).

Generating a USPS Intelligent Mail Bar Code (4-State Customer Bar Code)

You can generate the Intelligent Mail® bar code, formerly referred to as the *4-State Customer bar code* on your forms. The Intelligent Mail® bar code is a height-modulated bar code using four distinct vertical bar types.

Note You must follow the guidelines set by US Postal Service to implement an Intelligent Mail® bar code. You can download these guidelines from the following web site:

<http://ribbs.usps.gov/>

Since Documaker handles the encoding and printing of the bar code and does not use the font provided by the US Postal Service, you do not have to download that font. You do, however, need to get a Mailer ID from the US Postal Service.

It encodes a 20, 25, 29, or 31-digit string into 65 vertical bars, each representing one of four possible states: full bar, ascender, tracker, and descender. Intelligent Mail bar code expands the ability to track individual mail pieces and provide its customers with greater visibility in the mail stream.

Type	Field	Digits
Tracking Code	Bar Code ID	2 (The 2nd digit must 0-4)
	Service Type ID	3
	Customer ID	6
	Sequence Number	9
Routing Code	Delivery Point ZIP Code	0, 5, 9, or 11
Total		31 maximum

Here is an example of the 4CB bar code:



To generate a USPS Intelligent Mail bar code, insert a bar code and select the USPS 4CB option in the Format field.

ADDING BOOKMARKS

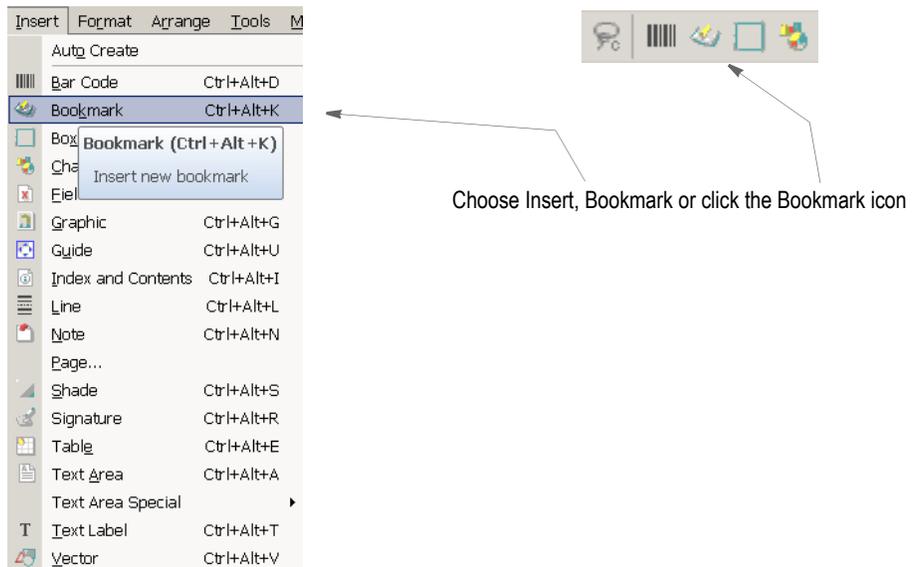
Studio lets you add bookmarks to the sections that make up the form set. These bookmarks define the text you want to print in the table of contents, table of figures, or index. When adding a bookmark, you can also specify a level number that determines the formatting to use.

Note For information about compiling the bookmarks you create into tables of contents, tables of figures, and indexes, see *Creating Tables of Contents, Tables of Figures, and Indexes on page 241*.

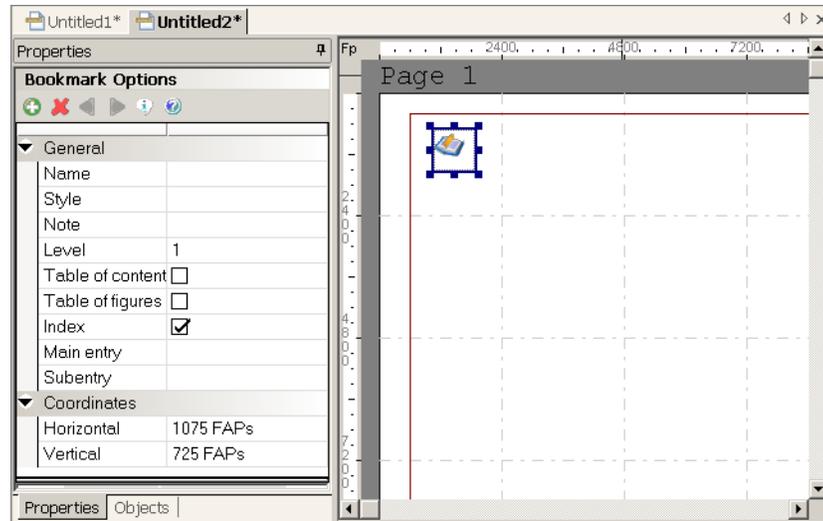
If the bookmark is for an index, specify the main entry and subentry. Index bookmarks are sorted by main entry, subentry, page number, and coordinate. Tables of contents and tables of figures are sorted by page number and coordinate.

Follow these instructions to add a bookmark to a section:

1. Open the section to which you want to add a bookmark.
2. Choose the Insert, Bookmark option.



The Bookmark Options window appears



3. Use these fields to define the bookmark:

Property	Description
General	
Name	(Optional) Enter the name you want to assign to the bookmark.
Style	This field only appears if you specified a style file in the BDF file. Select the style you want to use.
Note	Define the text you want to print within your index, or table of contents.
Level	Define a level number which determines the formatting to use (like ToC1, ToC2, and so on).
Table of Contents	Check this option if this bookmark is used by the table of contents.
Table of Figures	Check this option if this bookmark is used by the table of figures.
Index	Check this option if this bookmark is used by the index.
Main Entry	(Index only) Enter the text to appear as the main entry of the index.
Sub Entry	(Index only) Enter the text to appear as the sub entry of the index.
Coordinates	
Horizontal	Set the horizontal (left) starting position.
Vertical	Set the vertical (top) starting position.

Note If you later need to edit the bookmark, just double-click the bookmark icon in your section and then modify its properties as needed.

ADDING CHARTS

Studio lets you include graphical charts in your sections. You can choose from a variety of 2-dimensional and 3-dimensional charts including pie charts, bar charts, area charts, and percentage charts. You can customize the look of your chart by selecting various properties to change object positioning, dimensions, line characteristics, chart background, chart colors, and so on.

Studio provides two ways for you to show data on your charts: statically or dynamically. A chart can show static data you enter when you create the chart. This data does not change. A chart can also show dynamic data added via Documaker Server at run time or during data entry.

Note There are several rules you can use with dynamic data. These rules let you populate the chart with data from an extract file or from data in variable fields on the section. You can find more information on these rules in the Rules Reference:

- CreateChartSeries
 - FieldVarsToChartSeries
 - PurgeChartSeries
 - SetCustChartAxisLabels
-

After the chart has been created and the look of the chart is established, you can fill the chart with data. There are several ways to populate a chart with data. You can...

- Enter static data onto the chart by entering permanent data values when you create the chart. These data values are stored along with the chart as part of the image file.
- Populate the fields with data at runtime or when using the Entry module to update the dynamic data.

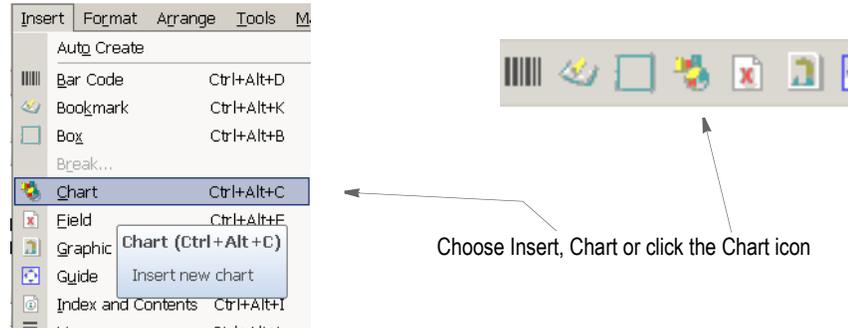
Here is a summary of the questions to consider and the steps required to create a chart:

1. Consider what type of data will fill the chart.
2. Consider what type of chart best shows the data.
3. Add the chart using the Chart toolbar icon or Insert menu option.
4. Select the type of chart and customize its appearance.
5. Exit the chart editor and save your work.

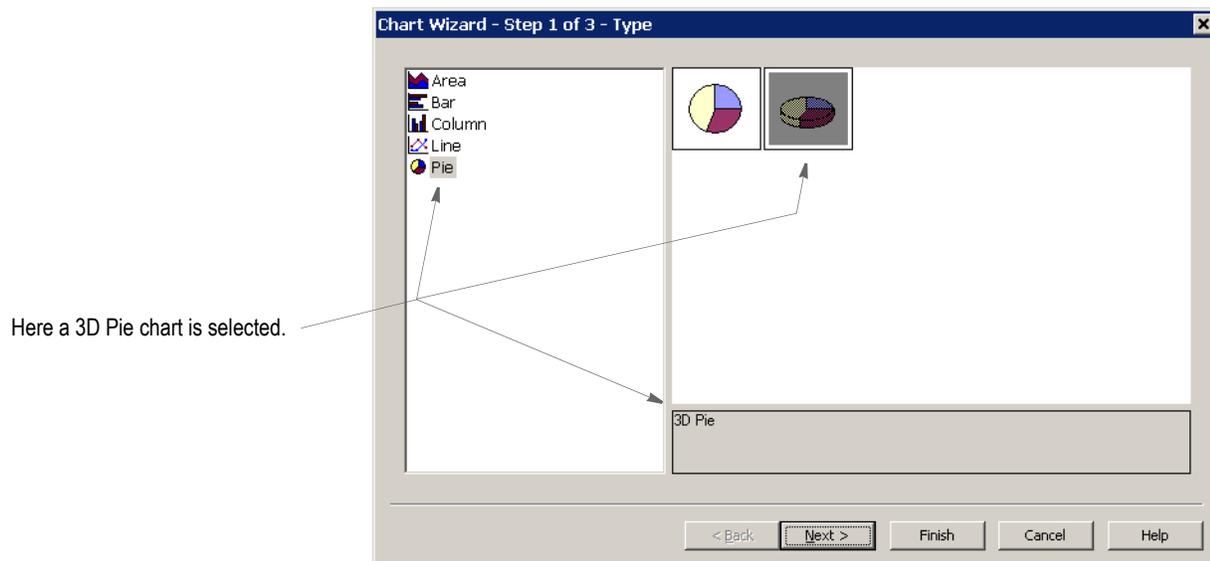
CREATING A CHART

Once you have determined the type of data you will be using and the type of chart that can best display the data, you can begin creating a chart meeting your data needs. To create a chart, follow these instructions:

1. Open the section that will contain the chart.
2. Choose Insert, Chart or click the chart icon.



3. Place the mouse pointer where you want the top left corner of the chart to be and press and hold the left mouse button. Drag the mouse to draw the approximate area where you want Studio to insert the chart. Release the mouse button. The Chart wizard's Type window appears.



Note You can change all of these settings after the Chart wizard creates your chart.

4. Choose the type of chart you want to create. Each type is comprised of subtypes, examples of which appear on the right side of the screen. Select the type and subtype you want and click Next.

Type	Subtypes
Area	2D Area, 2D Percent Area 3D Area, 3D Percent Area, 3D Split Area
Bar	2D Horizontal Bar, 2D On-top Horizontal Bar, 2D Percent Horizontal Bar 3D Horizontal Bar, 3D On-top Horizontal Bar, 3D Percent Horizontal Bar, 3D Split Horizontal Bar
Column	2D Vertical Bar, 2D On-top Vertical Bar, 2D Percent Vertical Bar 3D Vertical Bar, 3D On-top Vertical Bar, 3D Percent Vertical Bar, 3D Split Vertical Bar
Line	2D Line and 3D Ribbon
Pie	2D Pie and 3D Pie

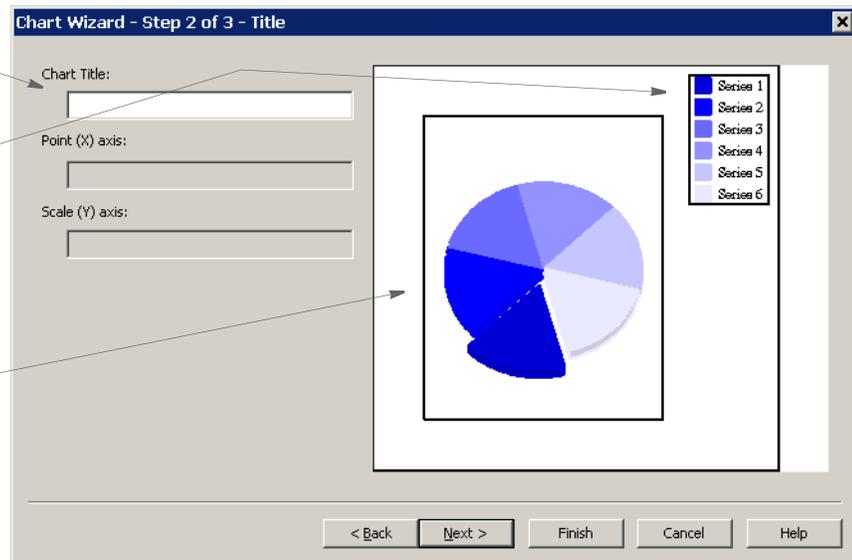
Note The default is a 3D Pie chart.

The Chart wizard's Title window appears.

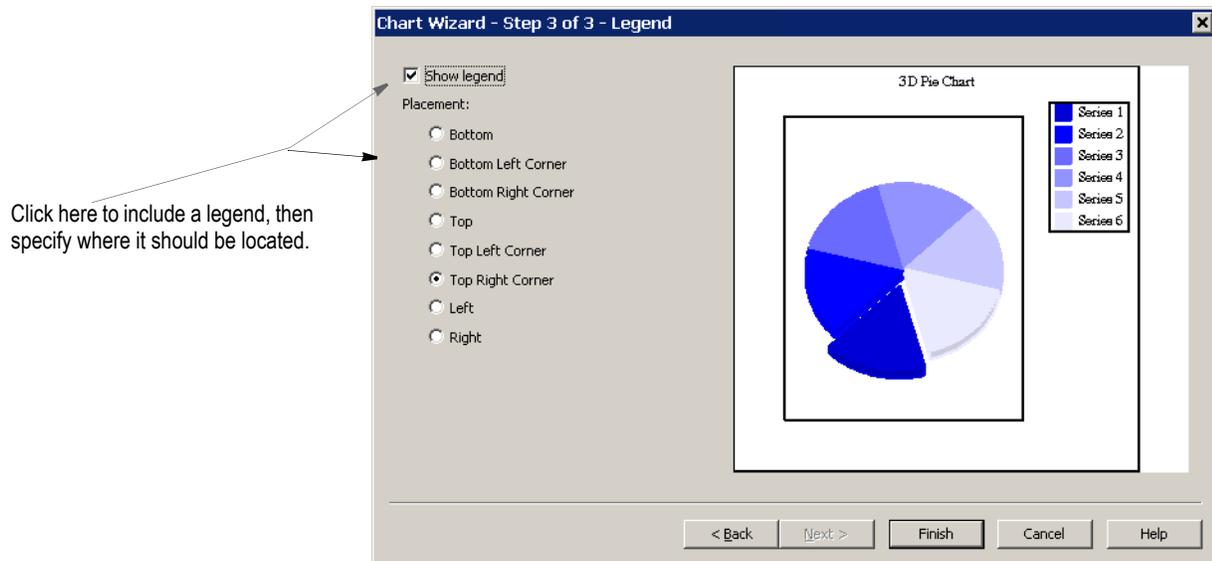
Enter the title here.

This is the chart's legend.

This is the chart's graph.

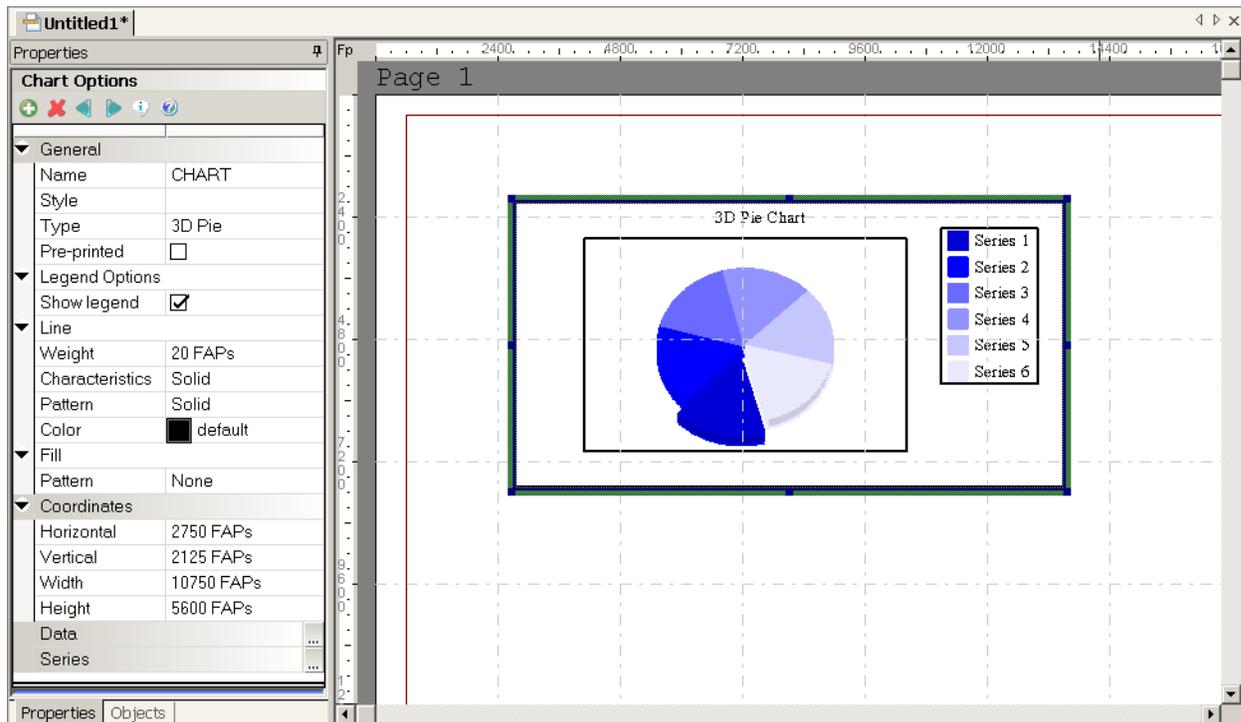


5. Enter the title of the chart and click Next. The Legend window appears.



- Specify if you want to show the legend and, if so, where you want it to appear. Then click Finish. Studio displays your chart.

Note You can further define the legend by modifying the legend's properties. For more information, see *Setting Legend Properties* on page 218.



- Once the Chart wizard creates your chart, use the properties on the chart's Properties tab to customize its position, dimensions, border, and the background of the entire chart area.

Note The Style field only appears if you specified a style file in the BDF file.

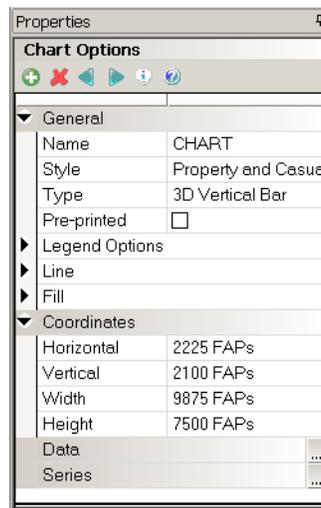
Property	Description
General	
Name	Here you can assign a name to the chart.
Style	This field only appears if you specified a style file in the BDF file. Select the style you want to use.
Type	Select the type from the list.
Preprinted	Check this property to tell the system <i>not</i> to print this chart when you print the section.
Legend	
Show Legend	Check this property to display the legend. For more information, see <i>Setting Legend Properties</i> on page 218.
Line	
Weight	Specify the thickness in FAP units (2400 per inch) of the line that comprises the chart's outside border. Enter zero (0) if you do not want an outside border.
Characteristics	Specifies whether the chart's outside border is solid or dashed and, if dashed, how long the dashes and the spaces between the dashes should be.
Pattern	Choose the pattern of the chart's outside border.
Color	Specifies the color of the chart's outside border.
Fill	
Pattern	Specifies the background pattern for the fill area — the area inside the chart's outside border. Choose None if you do not want a pattern fill.
Color	Specifies the color of the fill area.
Coordinates	
Horizontal	Specifies the horizontal position of the upper left hand corner of the chart. You can change the chart's position by entering coordinates or by using the Left and Right arrow keys.
Vertical	Specifies the vertical position of the upper left hand corner of the chart. You can change the chart's position by entering coordinates or by using the Up and Down arrow keys.
Width	Specifies the width of the chart. You can change the chart's width by entering a value or by dragging the chart handles.
Height	Specifies the height of the chart. You can change the chart's height by entering a value or by dragging the chart handles.
Data	Use this option to display the window where you modify series and data information. You can enter static data associated with a series for printing a static chart or associate the data with variable fields in the series. See <i>Assigning Variable Fields</i> on page 214 for more information.

Property	Description
Series	Use this option to manipulate a data element called a <i>series</i> . A series is a set of data, shown on the chart as a bar, a line, or a piece of the pie. You can use this option to add another piece of pie, a new bar, or a new line or to format the data in a series and its legend area. See <i>Adding a Series</i> on page 215 for more information.

ENTERING DATA

You can enter static data associated with a series for printing a static chart or you can associate the data with variable fields in the series. Follow these steps to define how Studio will get the data that appears on the chart.

1. From the chart's Properties window, click the ellipsis icon in the Data field.



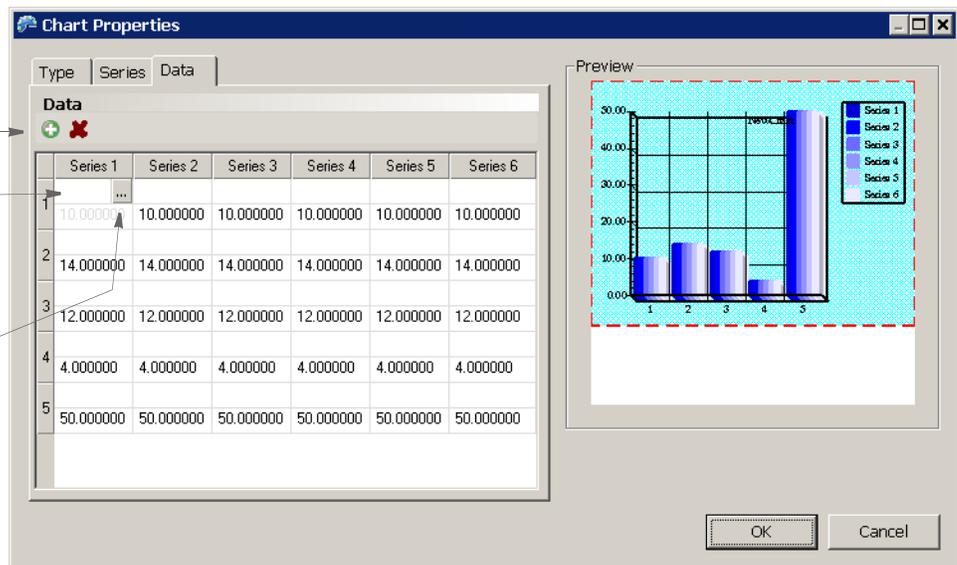
Click here to define the data for your chart.

This window appears.

Use these buttons to add or delete the data in a series.

Double-click here to enter static data for this series.

Click here to choose a variable field as the source for the data in this series.



Note The Add and Delete icons on the Data tab let you add or delete the *data* in a series. To add or delete a series, you have to go to the Series tab.

- When you insert a data point to a series, you can either enter static data or use the ellipsis button to insert a variable field. Variable field data is added at processing time. See *Assigning Variable Fields* on page 214 for more information.

Click Ok when you are finished defining the data for the chart.

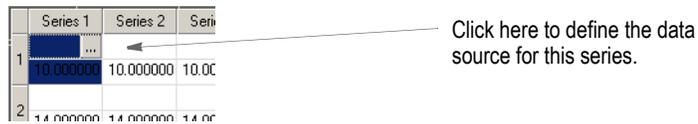
ASSIGNING VARIABLE FIELDS

Follow these steps to use a variable field as the source of data for a chart series.

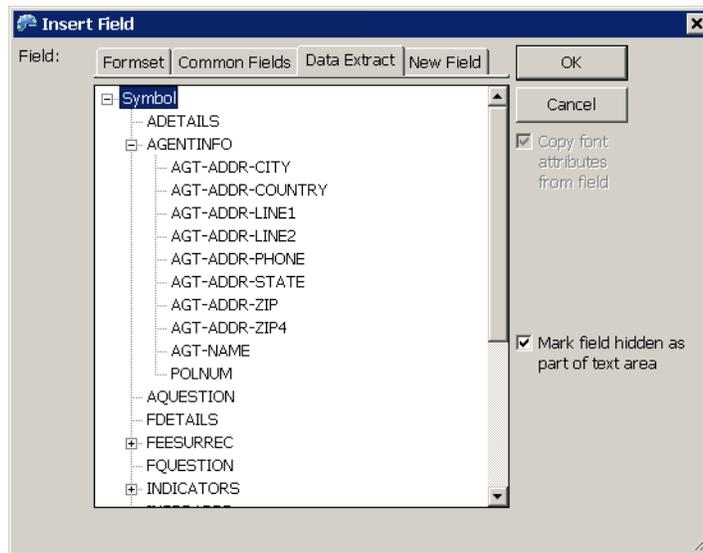
- Click the ellipsis button in the Data field.



- On the Data tab, click in the series for which you want to define a variable field, then click the ellipsis button.



The Insert Field window appears. This example shows the fields defined in the Data Extract dictionary.



- Choose the variable field you want to serve as the source of the data for the series. You can choose from variable fields defined for the current form set (section), fields in the Common Fields dictionary, fields in the Data Extract dictionary, or define a new field.

Note There are section-level rules that must be used when you use variable data to propagate a chart. Refer to the Rules Reference for a description of the following rules: FieldVarToChartSeries, CreateChartSeries, PurgeChartSeries, DeleteDefaultSeriesData.

ADDING A SERIES

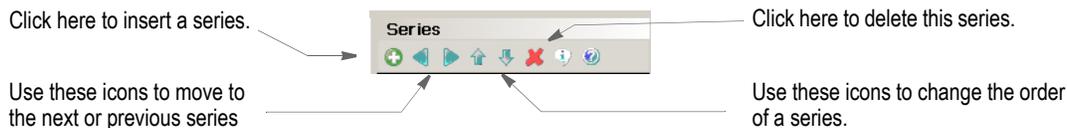
A series is a set of data, shown on the chart as a bar, a line, or a piece of the pie. You can use this option to add another piece of pie, a new bar, or a new line, or to format the data in a series and its legend area.

To add a series, follow these steps.

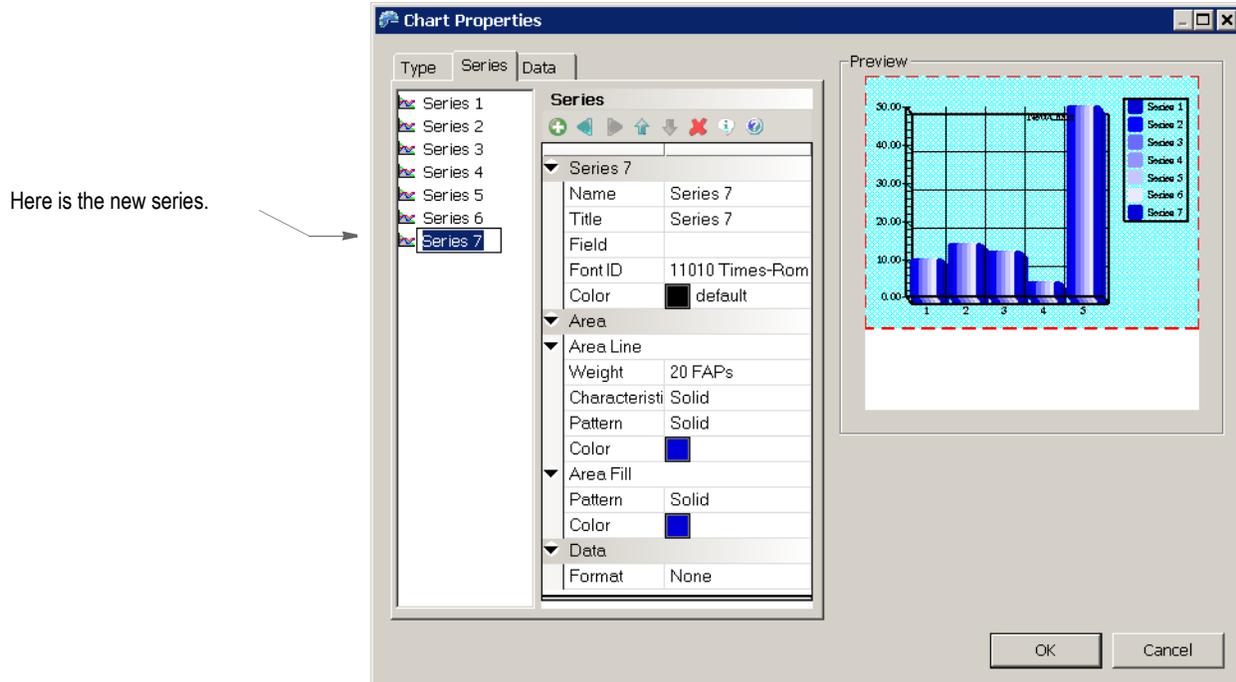
1. Click the ellipsis button in the Series field.



2. On the Series tab, click the Add icon to add a series to your chart.



Studio appends the new series at the end of the list and lets you define its properties. In this example, *Series 7* is the new series.



3. Use these properties to define the series.

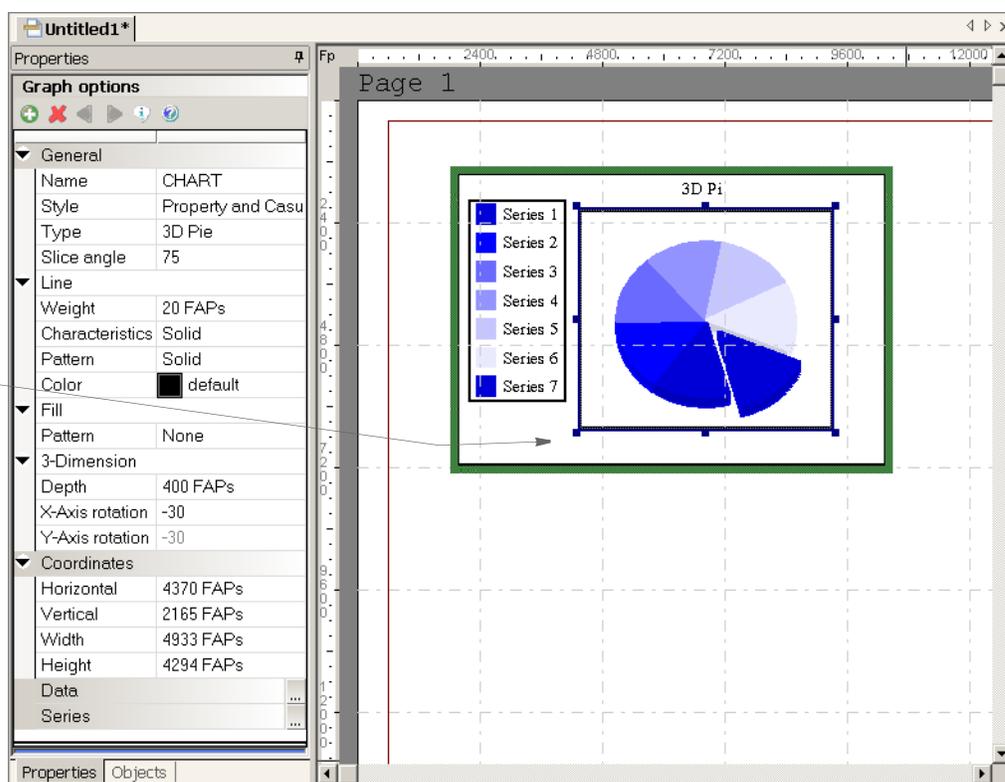
Property	Description
<i>Series X</i>	
Name	Enter the name of the series.
Title	Enter the title you would like to appear in the legend area for this series.
Field	Use the ellipsis button to select a field, if the data for this series comes from a variable field. See <i>Assigning Variable Fields</i> on page 214 for more information.
Font ID	Specifies the ID of the font you want to use for this series' legend text.
Color	Choose the color for this series' legend text.
Explode %	(Pie charts only) Used to define the degree of separation from other pie slices. Enter zero (0) if you want the slice to touch its adjoining slices. Enter a larger number, such as 20%, to pull a slice out of the pie for emphasis purposes.
Size %	(Line charts only) Used to define the size of the symbols representing a data point for the series. Enter a larger number to emphasize a specific series.
Area Line	
Weight	Specifies the weight, in FAP units (2400 per inch), of the line, bar, or pie slice that represents this series.
Characteristics	Specifies whether the line, bar, or pie slice that represents this series is solid or dashed and, if dashed, how long the dashes and the spaces between the dashes should be.
Pattern	Specifies the pattern of the line, bar, or pie slice that represents the series.
Color	Specifies the color of the line, bar, or pie slice that represents the series.
Area Fill	
Pattern	Specifies the pattern for the series in the chart area.
Color	Specifies the color for the series in the chart area.
Data	
Format	If you set this property to None, the following, related properties do not appear. Specifies the format of the numeric legend to the data point (series).
Align	Specifies where the data prints with respect to the data point (series), such as center, left, or right.
Accumulate	Check this property if you want to add the values of the data points in a series.
Font ID	Specifies the ID of the font you want to use for the numeric legend to the data point (series).
Color	Specifies the color of the numeric legend of the data point (series).
Apply to All	Applies the characteristics of all the fields in this section to all of the series.
Data Line	

Property	Description
Weight	Specifies the weight, in FAP units (2400 per inch), of the line that highlights the data.
Characteristics	Specifies whether the line is solid or dashed and, if dashed, how long the dashes and the spaces between the dashes should be.
Pattern	Specifies the pattern of the line that highlights the data.
Color	Specifies the color of the line that highlights the data.
Data Fill	
Pattern	Specifies the pattern fill of the area inside the line that surrounds the data.
Color	Specifies the color of the area inside the line that surrounds the data.

SETTING GRAPH PROPERTIES

To define the way the graph appears, first click on the graph to select it, then modify its properties as necessary.

Click on the graph to display the graph's properties.



Use these properties to define the chart's graph:

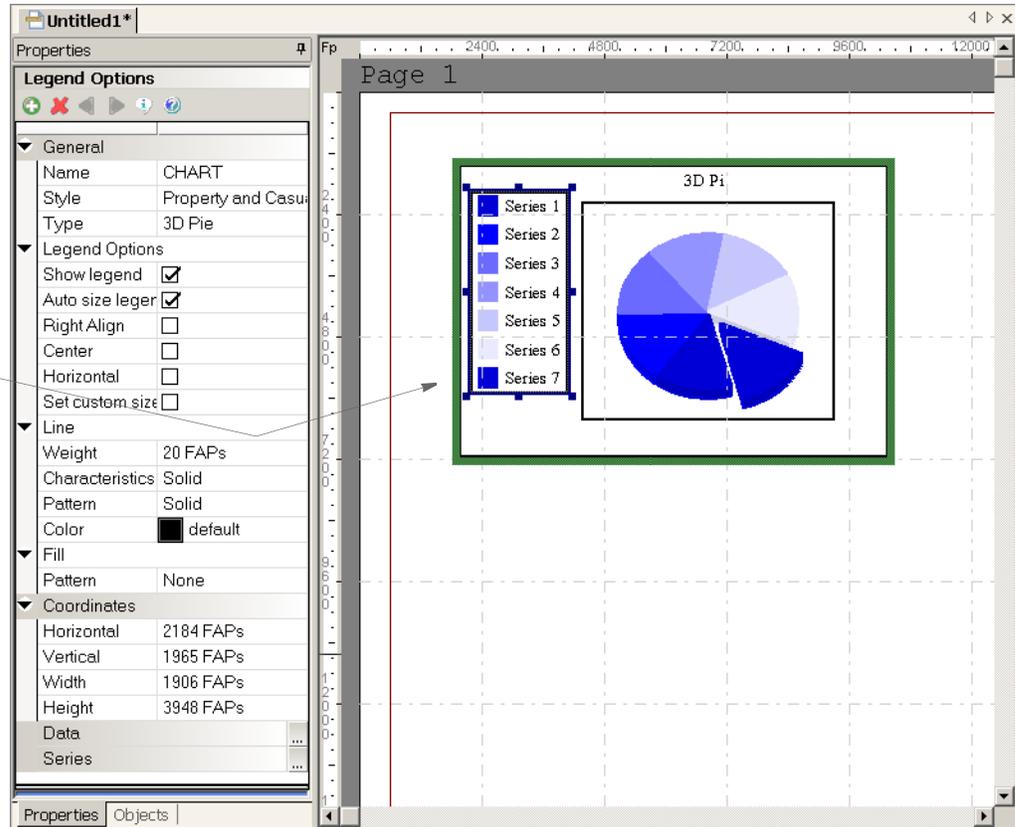
Property	Description
General	
Name	The chart name appears here.

Property	Description
Style	This field only appears if you specified a style file in the BDF file. Select the style you want to use.
Type	Select the type from the list.
Line	
Weight	Specifies the weight, in FAP units (2400 per inch), of the line around the graph.
Characteristics	Specifies whether the line is solid or dashed and, if dashed, how long the dashes and the spaces between the dashes should be.
Pattern	Specifies the pattern of the line.
Color	Specifies the color of the line.
Fill	
Pattern	Specifies the pattern of fill area for the graph.
Coordinates	
Horizontal	Specifies the horizontal position of the upper left hand corner of the graph. You can change the graph's position by entering coordinates or by using the Left and Right arrow keys.
Vertical	Specifies the vertical position of the upper left hand corner of the graph. You can change the graph's position by entering coordinates or by using the Up and Down arrow keys.
Width	Specifies the width of the graph. You can change the graph's width by entering a value or by dragging the graph's handles.
Height	Specifies the height of the graph. You can change the graph's height by entering a value or by dragging the graph's handles.

SETTING LEGEND PROPERTIES

To set properties for the chart's legend, first click on the legend to select it, then modify its properties as necessary.

Click on the legend to display the legend's properties.



Use these properties to define the chart's legend:

Property	Description
General	
Name	The chart name appears here.
Style	This field only appears if you specified a style file in the BDF file. Select the style you want to use.
Type	Select the type from the list.
Legend Options	
Show legend	Check this box to show the legend.
Auto size legend	Check this box if you want the system to automatically size the legend.
Right Align	Check this box to right align the legend.
Center	Check this box to center the legend.
Horizontal	Check this box to have legend display horizontally. Remove the check to display the legend vertically.
Set custom size	Check this box to custom size the legend. If you check this box, the Width and Height parameters appear. Enter a value for these parameters, which Studio will use to size the legend.

Property	Description
Line	
Weight	Specifies the weight, in FAP units (2400 per inch), of the line around the legend.
Characteristics	Specifies whether the line is solid or dashed and, if dashed, how long the dashes and the spaces between the dashes should be.
Pattern	Specifies the pattern of the line.
Color	Specifies the color of the line.
Fill	
Pattern	Specifies the pattern of fill in legend area.
Coordinates	
Horizontal	Specifies the horizontal position of the upper left hand corner of the legend. You can change the legend's position by entering coordinates or by using the Left and Right arrow keys.
Vertical	Specifies the vertical position of the upper left hand corner of the legend. You can change the legend's position by entering coordinates or by using the Up and Down arrow keys.
Width	Specifies the width of the legend. You can change the legend's width by entering a value or by dragging the legend's handles.
Height	Specifies the height of the legend. You can change the legend's height by entering a value or by dragging the legend's handles.

Note At runtime, the system starts drawing the first slice of a pie chart at zero (0) degrees. To specify the angle at which the system starts drawing slices, make the following entry in the INI file:

```
< Control >
  FirstPieSliceAngle = XXX
```

Where XXX represents the angle degree, from zero (0) to 359.

You can use the Manage, System, Settings option to set this INI option.

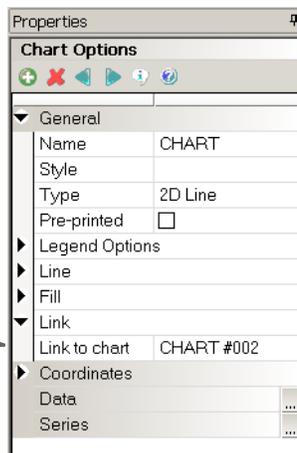
Combining Charts

Studio lets you generate output that combines certain types of charts. For instance, you can link a 2D line chart to a 2D vertical bar, 2D line, or a 3D vertical bar chart.

To combine charts, create the two charts you want to link then follow these instructions:

1. Double click on one of the charts to display the chart's properties.

This chart is linked to CHART #002.



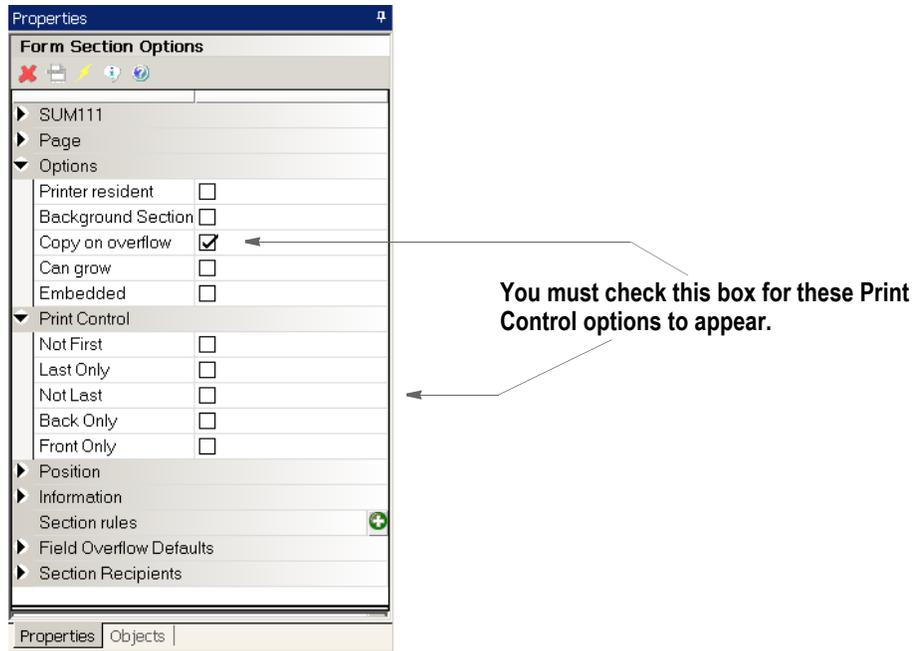
2. Select the chart you want to link to in the Link to Chart field.

Once you link a chart to another chart, the chart will then use the position and size information from the other chart. For instance, if you link chart A to chart B, chart A then uses chart B's position and size information and those fields are unavailable for chart A.

SETTING HEADER AND FOOTER PRINT OPTIONS

Use the Print Control options to specify how you want a section to print. Typically, these options are used to specify how sections that contain headers or footers print. To set print control options, follow these steps:

1. Open the form in Studio and double click on the section for which you want to set print controls.
2. Make sure the Copy on Overflow box is checked for the section.



3. Choose from these options to control when the section prints:

Option	Description
Not First	Do not print the first occurrence of this section.
Last Only	Only print the last occurrence of this section.
Not Last	Do not print the last occurrence of this section.
Back Only	If it is a duplex page, print only on the back (even-numbered) pages. If it is a simplex page, do not print.
Front Only	If it is a duplex page, print only on the front (odd-numbered) pages. If it is a simplex page, print on all pages.

You can select multiple print options, such as Not Last and Back Only. If an occurrence of this section was on all pages, those selections would tell Documaker Server to print the section on all back pages, except for the last page of the form.

Note When you select multiple print options for a section, all options must be true for the section to print.

Keep in mind...

- All simplex pages are considered front pages.
- The print control options do not determine the page duplex settings. For instance, if you select the Back Only option on a simplex form, the section is never printed.
- When you choose Last Only, Not Last, or Not First, it affects the first and last occurrence of the section in the form.
- For pagination, the header size is based on the largest possible header. Footer size is based on the largest possible footer.
- During processing, Documaker Server does not repaginate the form set to accommodate varying size header and footer images.

Page numbering is turned on by default. If you set the PageNumbers INI option to No to turn off page numbering, the system ignores your header/footer print options.

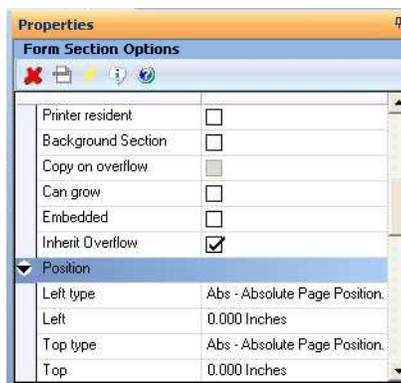
CREATING HEADERS AND FOOTERS FOR MULTIPAGE SECTIONS

To create headers and footers for multipage sections, follow these steps:

1. Open a multipage section.

Note For best results, define the multipage section to be smaller than a full page or else leave white space at the top and bottom of the page to allow for the placement of the header or footer. Also keep in mind that the way you design the multipage section dictates the type of origin rules you would apply to the header, footer, and the multipage section.

2. Add headers and footers to the first page of the multipage section.
3. Open the form in which the section resides and select the section. In the Form Selection Options, check the Inherit Overflow field.



Option	Description
Inherit Overflow	Check this box if the headers and footers on the first page of a multipage section should be copied onto the subsequent pages of the multipage section.

With this setting, during processing Documaker Server will add the copy on overflow sections defined on the first page to all subsequent pages of the multipage section.

Note If the Inherit Overflow option is checked and you remove the check mark, the copy on overflow sections are removed from subsequent pages of the multipage section.

DEFINING PARAGRAPHS

Studio lets you create canned paragraphs that can be inserted into a form. The system then uses the selected paragraph to fill a multiline text field.

The way paragraphs are selected differs between Documaker Workstation and Documaker Server. With Documaker Workstation, a user selects which paragraph to use. With Documaker Server, trigger processing makes that determination.

Defining paragraphs to later be inserted into a form, either by a Documaker Workstation user or during a Documaker Server processing cycle, involves these steps:

- Creating paragraphs
- Creating paragraph lists. For more information, see *Creating Paragraph Lists* on page 234.
- Creating a multiline text field in a section that uses paragraph selection. For more information, see *Inserting a Paragraph List into a Section* on page 235.

CREATING PARAGRAPHS

Creating a paragraph (PAR file) is very similar to creating a text area in a section. Like a text area, a paragraph is primarily text and you can insert fields, graphics, line breaks, symbols, and boxes into a paragraph. You can also perform a spell check, grammar check, and generate readability statistics for your paragraphs.

Follow these steps to create a paragraph:

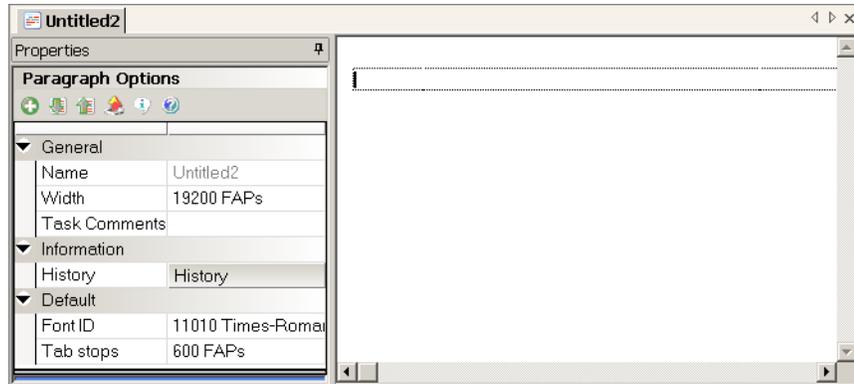
1. Double-click on Paragraphs in the Workspace tree.

Double-click here to open or create a paragraph.



The Open Files window appears.

2. Select a paragraph from the list or click New to create a paragraph. The Paragraph Options window appears.



Use these properties to define the general characteristics of your paragraph:

Property	Description
General	
Name	The name of this paragraph.
Width	Specify the width of this paragraph.
Task Comments	Click the ellipsis button to display the Task Comments window and enter any comments you want associated with this paragraph. For instance, you could use this property to enter a to do list of tasks for this paragraph. Your entries appear on the Task List or Finishing Report.
Information	
History	Click the History button to display the History window and view any comments. Studio shows you the ID of the user who entered the comments and the date and time the comments were entered. You can also enter new comments.
Default	
Font ID	Select the default font ID for this paragraph.
Tab Stops	Specify the tab stop interval for this paragraph.

Once created, you can then add text, fields, boxes, breaks, graphics, symbols and other objects to your paragraph.

To delete a field, box, break, graphic, or symbol that has been inserted in the paragraph, use the Delete option on the Edit menu or the right-click menu.

Inserting Fields

Use the Insert, Field option or click the Insert icon in the Paragraph Options area to add a variable field. Once you insert a field, Studio shows you that field's properties. These properties are the same as those for a field in a section with one exception, a field in a paragraph has this additional property:

Property	Description
Represent With	Enter the characters you want to use to fill the field until the system adds the actual field data.

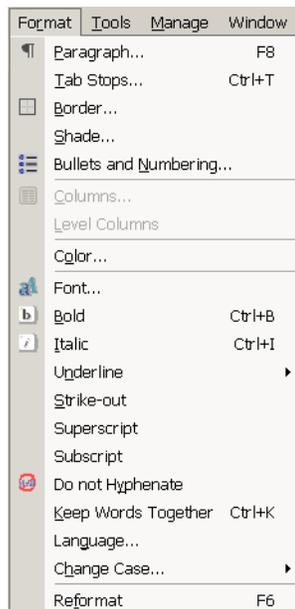
Note You cannot insert bar code and multiline text variable fields into a paragraph.

To modify field properties, place your cursor on the field and then make any necessary changes to the properties.

FORMATTING PARAGRAPHS

To further define your paragraph, Studio provides a number of properties that let you control how paragraphs appears and prints. To set these properties, Choose Format, Paragraph.

Use these options to set paragraph properties.

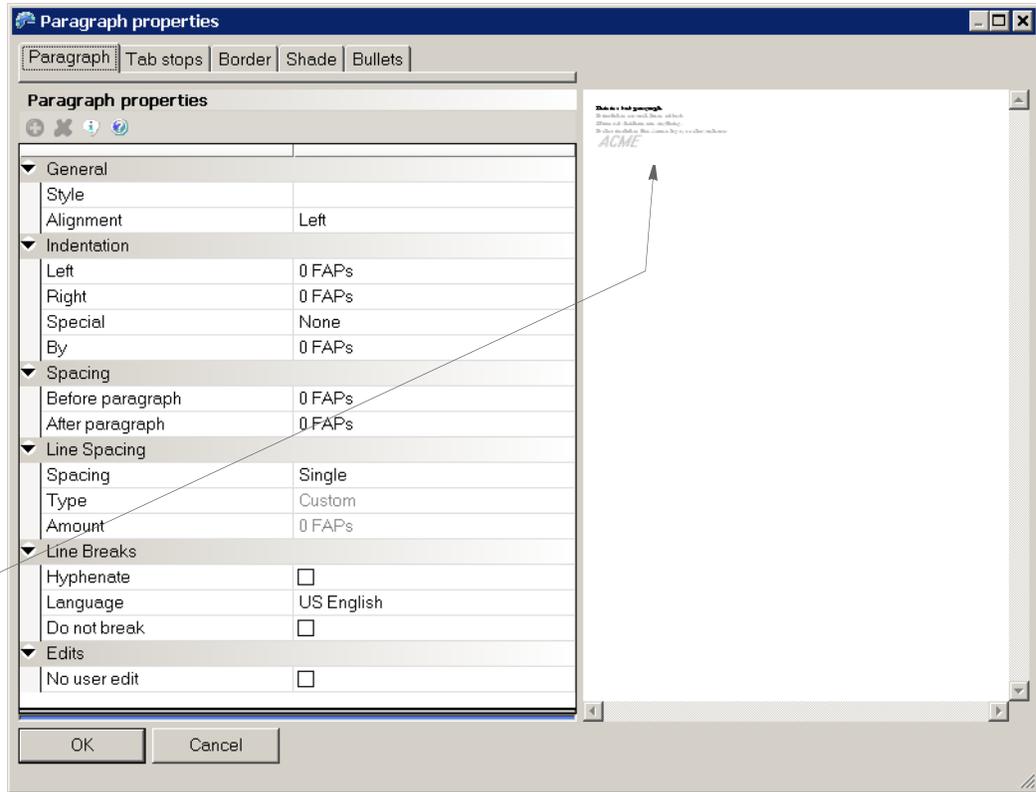


Use these options to format the text in your paragraph.

Whether you choose the Paragraph, Tab Stops, Border, Shade, or Bullets and Numbering option, the Paragraph Properties window appears. All of these properties are available from the tabs on this window.

Use these properties to define how the paragraph displays and prints.

Here Studio shows you what your choices will look like.



These paragraph properties are presented on the following tabs:

- Paragraph. For more information, see *Defining Paragraph Properties* on page 229.
- Tab Stops. For more information, see *Defining Tab Stops* on page 230.
- Border. For more information, see *Defining Border Properties* on page 230.
- Shade. For more information, see *Defining Shade Properties* on page 231.
- Bullets. For more information, see *Defining Bullet Properties* on page 232.

Note For those properties where you are entering an amount of space, your entry will be in the unit of measure you chose for the Measurement option (see Manage, System, Settings, Common/Document View). For instance, if you chose FAP units (2400 per inch) and you want to right indent .25 inches, you would enter 600.

Defining Paragraph Properties

Use these properties on the Paragraph tab to define the paragraph layout.

Paragraph properties	
<div style="border: 1px solid black; padding: 2px;"> Paragraph Tab stops Border Shade Bullets </div>	
<div style="border: 1px solid black; padding: 2px;"> + ✕ ↑ ↓ </div>	
<div style="border: 1px solid black; padding: 2px;"> General </div>	
Style	
Alignment	Left
<div style="border: 1px solid black; padding: 2px;"> Indentation </div>	
Left	0 FAPs
Right	0 FAPs
Special	None
By	0 FAPs
<div style="border: 1px solid black; padding: 2px;"> Spacing </div>	
Before paragraph	0 FAPs
After paragraph	0 FAPs
<div style="border: 1px solid black; padding: 2px;"> Line Spacing </div>	
Spacing	Single
Type	Custom
Amount	0 FAPs
<div style="border: 1px solid black; padding: 2px;"> Line Breaks </div>	
Hyphenate	<input type="checkbox"/>
Language	US English
Do not break	<input type="checkbox"/>
<div style="border: 1px solid black; padding: 2px;"> Edits </div>	
No user edit	<input type="checkbox"/>

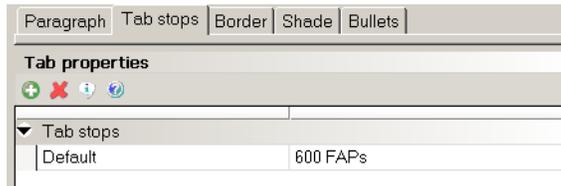
Property	Description
General	
Style	This field only appears if you specified a style file in the BDF file. Select the style you want to use.
Alignment	Choose from Left, Right, Centered, or Left and Right to set the paragraph alignment.
Indentation	
Left	Enter the amount of space you want to indent from the left.
Right	Enter the amount of space you want to indent from the right.
Special	Choose from these options: None, First Line, or Hanging.
By	Enter the amount of indentation to apply to First Line or Hanging.
Spacing	
Before Paragraph	Enter the amount of space you want before the paragraph.
After Paragraph	Enter the amount of space you want after the paragraph.
Line Spacing	
Spacing	Choose from these options: Single, Double, Exactly, At Least, Multiple.
Type	Choose from these options: Lines Per Inch or Custom.
Amount	Enter the amount to complete your line spacing choice.

Property	Description
Line Breaks	
Hyphenate	Check this box to turn on automatic hyphenation.
Language	Select the language to use for spelling checks and hyphenation.
Do Not Break	Check this box to keep paragraph lines together.
Edits	
No User Edit	Check this box to prevent users from editing the content when it is imported.

Click Ok to save your entries or Cancel to exit without making changes.

Defining Tab Stops

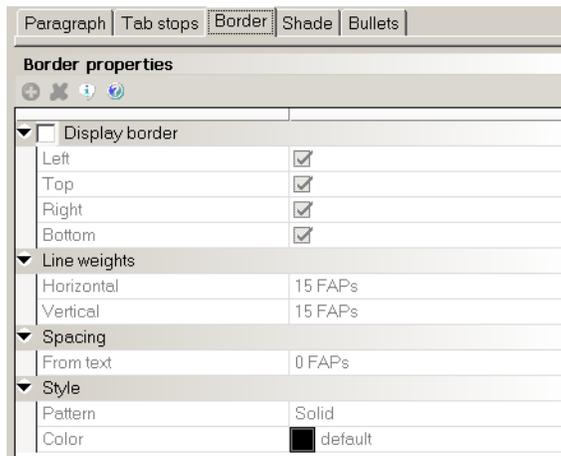
Use these properties on the Tab Stops tab to define tab settings for your paragraph.



Property	Description
Default	Enter the amount of space you want to set as the default tab spacing. For instance, if you are using FAP units (2400 per inch) and you want your tabs spaced .25" apart, you would enter 600.

Defining Border Properties

Use these properties on the Border tab to define the border around the paragraph.

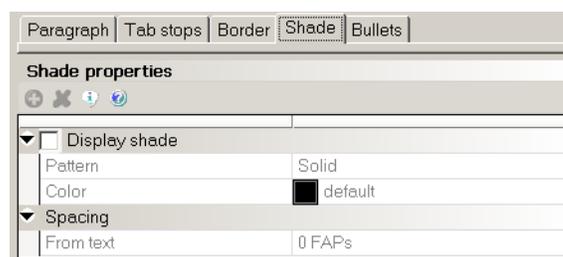


Property	Description
Display Border	Check this box if you want the system to display and print a border around the paragraph. Once you check this box, the Left, Top, Right, and Bottom properties are enabled.
Left	Check this box to display and print a border to the left of the paragraph.
Top	Check this box to display and print a border above the paragraph.
Right	Check this box to display and print a border to the right of the paragraph.
Bottom	Check this box to display and print a border below the paragraph.
Line Weights	
Horizontal	Enter the line weight you want for horizontal lines. The greater the number, the thicker the line.
Vertical	Enter the line weight you want for vertical lines. The greater the number, the thicker the line.
Spacing	
From Text	Enter the amount of space you want between the border and the text of the paragraph.
Style	
Pattern	Choose a pattern for the fill of the border lines.
Color	Choose a color for the border lines.

Click Ok to save your entries or Cancel to exit without making changes.

Defining Shade Properties

Use these properties on the Shade tab to define background shading for the paragraph.



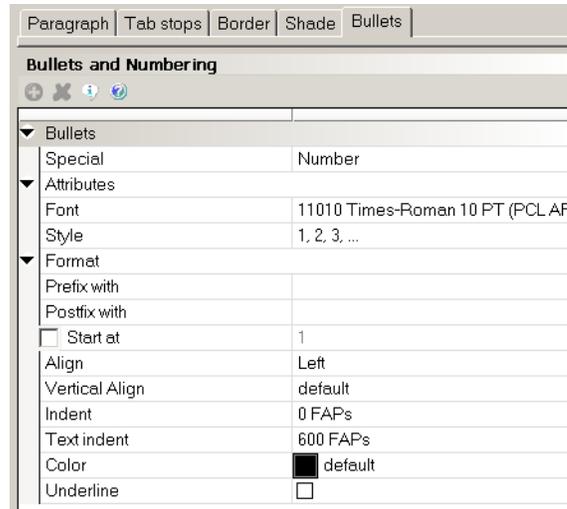
Property	Description
Display Shade	Check this box if you want the system to display and print a shaded area behind the paragraph. Once you check this box, the Pattern and Color properties are enabled.
Pattern	Choose a pattern for the fill of the shaded area.
Color	Choose a color for the shaded area.
Spacing	

Property	Description
From Text	Enter the amount of space you want between the shaded area and the text of the paragraph.

Click Ok to save your entries or Cancel to exit without making changes.

Defining Bullet Properties

Use these properties on the Bullet tab to define how bullets appear in the paragraph.



Depending on your entry in the Special field, different properties appear. Shown here are the properties for Number bullets.

Property	Description
Bullets	
Special	Choose the kind of bullet you want. You can choose from None, Font, Graphic, Number, or Symbol.
Attributes	
Font	(Font, Number, and Symbol bullets only) Select the ID of the font you want to use
Character	(Font bullets only) Enter the bullet character. Click the ellipsis button to display the Symbol window and choose a character from the ones shown there.
Graphic	(Graphic bullets only) Enter the name of the graphic file you want to use as a bullet. Click the button to select a file from the library.
Symbol	(Symbol bullets only) Click the ellipsis button to select from a list of system-supplied symbol bullets. Your choices include filled and hollow round bullets and filled and hollow square bullets.
Style	(Number bullets only) Select the numbering style you want to use. You can choose from numeric (1, 2, 3), lowercase alphabetic (a, b, c), uppercase alphabetic (A, B, C), lowercase roman numerals (i, ii, iii) or uppercase roman numerals (I, II, III).
Prefix With	(Number bullets only) Enter up to two characters to precede the number, letter, or roman numeral.

Property	Description
Postfix With	(Number bullets only) Enter up to two characters to follow the number, letter, or roman numeral.
Start At	(Number bullets only) Check this box and then enter the number, letter, or roman numeral at which you want the system to start numbering. The default is 1, a, A, i, or I, depending on your choice in the Style property.
Align	Specify the alignment for the bullet. You can choose from Center, Decimal, Left, and Right.
Vertical Align	Specify the vertical alignment for the bullet. You can choose from the default or Top.
Indent	Enter the amount of space you want to indent the bullet from the left.
Text Indent	Enter the amount of space you want to indent the text from the bullet.
Color	Choose a color for the bullet. Click the ellipsis button to display the Color Selection window. On this window you can choose a color and specify whether you want the system to print the bullet in that color.
Underline	(Font and Number bullets only) Check this box if you want an underline beneath the bullet.

Click Ok to save your entries or Cancel to exit without making changes.

CREATING PARAGRAPH LISTS

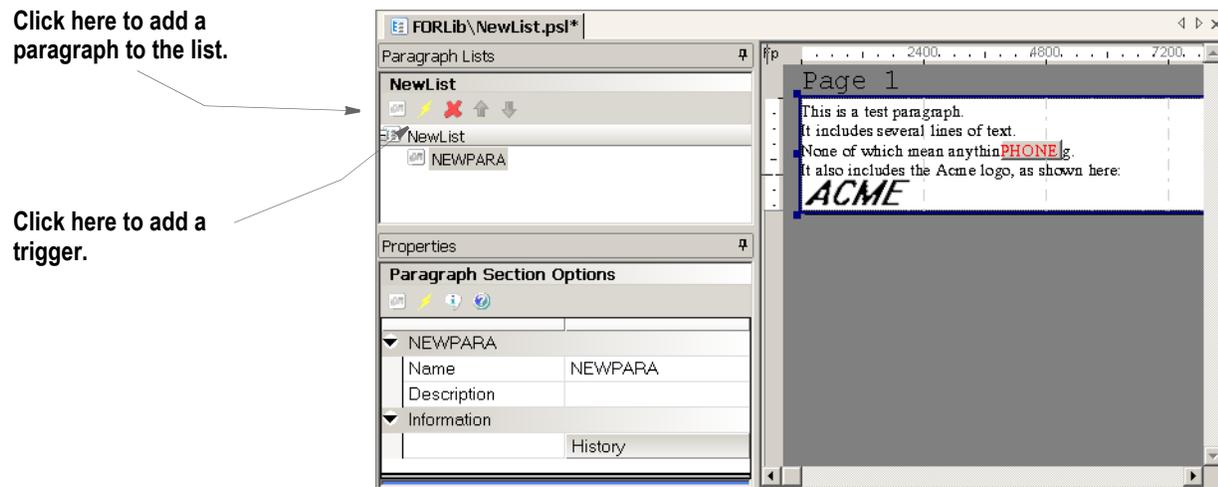
Studio lets you create a list of paragraphs (PSL file) which lists all of the paragraphs from which either a Documaker Workstation user or Documaker Server can select at processing time. You insert this paragraph list into a multiline text field on a section. Follow these steps to create a paragraph list:

1. Double-click on Paragraph Lists in the Workspace tree.



The Open Files window appears.

2. Select a paragraph list from the list or click New to create a paragraph list. The Paragraph Options window appears.



3. Click the Add Paragraph icon to add a paragraph to the list. As you add a paragraph, Studio shows you that paragraph and displays its options. You can change the name and description if necessary.

Click the History button to view or add information about the paragraph.

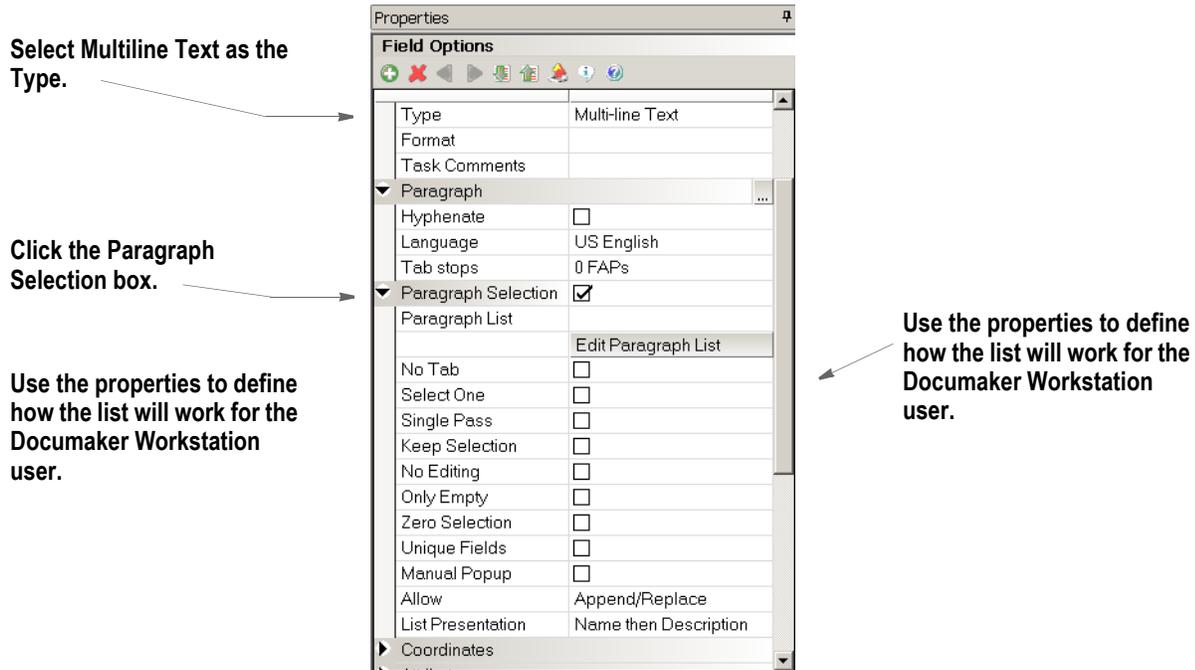
Note Your entries in the Name and Description fields appear to the Documaker Workstation user during data entry to help that user choose which paragraph to insert.

4. Once you add all the applicable paragraphs to your list, choose the File, Check In option to add the paragraph list to the library. You can also choose File, Save to save your list if you are not yet ready to check it in.

INSERTING A PARAGRAPH LIST INTO A SECTION

Once you create paragraphs and a paragraph list, you must then add the list to a section. To do this, you create a multiline text field in the section. Follow these steps:

1. Open a section and insert a field into that section.
2. In the Field properties, select Multiline Text in the Type field.



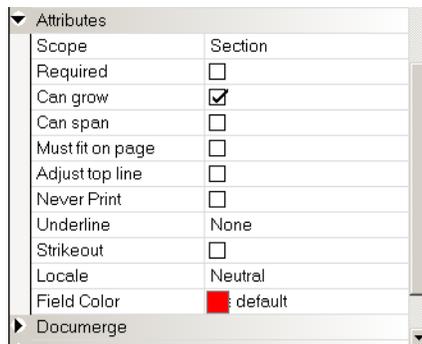
3. Click in the box next to Paragraph Selection. Then enter or browse for the appropriate paragraph list name in the Paragraph List field. Click the Edit Paragraph List button if you need to edit the list.
4. Use these properties to define how the list will work for the Documaker Workstation user:

Property	Description
No Tab	Check this option to prevent Documaker Workstation from automatically advancing to the next field after selecting a paragraph.
Select One	Check this option if you want the Paragraph Selection window in Documaker Workstation to allow only one of the available paragraphs to be selected in the Selected Paragraphs area. Depending on your intent, you may also want to check the Hide Append option to prevent users from reentering the field, selecting a second paragraph, and appending it to the first.
Single Pass	Check this option to prevent the user from re-entering the field and making changes after he or she selects a paragraph.
Keep Selection	Check this option to make the Paragraph Selection window retain the previously selected paragraphs in the Selected Paragraphs area.

Property	Description
No Editing	Check this option to prevent users from manually editing the field beyond paragraph selection. Changes to the field are allowed only through Paragraph Selection window.
Only Empty	Check this option if you want the Paragraph Selection window to appear only when the text area is empty. If the system detects that the field contains data, it will not display the Paragraph Selection window.
Zero Selection	Check this option to enable the Paragraph Selection window's Replace button to work even when the selected paragraphs area is left blank, which will clear the field.
Unique Fields	Checking this option causes an embedded field within this field to be assigned a unique name. This prevents matching or duplicate field names on a section. A rolling number is assigned to the end of the embedded field name.
Manual Popup	Checking this option disables the Paragraph Selection window from automatically opening whenever the field is selected and users must then press F4 to open it.
Allow	Use this option to determine which buttons will be visible on the Paragraph Selection window. You can choose from Append only, Replace only, or Append/Replace. The default is Append/Replace.
List Presentation	Select how the Paragraph Selection window displays Paragraphs. Your choices include Description Only, Description then Name, Name only, or Name then Description.

5. In the Attributes, check the Can Grow property.

Check this property.



If it is possible for the paragraphs you select to span to another page, you must also check the Can Span property.

DEFINING TABLES

Use the Table wizard in Studio to create tables in your sections. This wizard guides you through the process and optionally lets you select a data extract record to link to a table. This record is then used to trigger the table. The record's children are linked to the table columns. Studio uses the existing XDD rule, so you do not have to set up field mapping rules for the table and the Extract Data Dictionary.

Keep in mind...

- You can only select records which have children.
- You can only select one record per table.
- If the field is defined in your common fields (FDB) database, Studio assigns the type defined there. If the field is not in the FDB, Studio assigns alphanumeric as the type.
- For each column, Studio assigns the XDDTrigger rule in the Custom Rule Name field. In the Custom Rule Parameters field, Studio prefixes a question mark (?) to the child name. Here is an example:

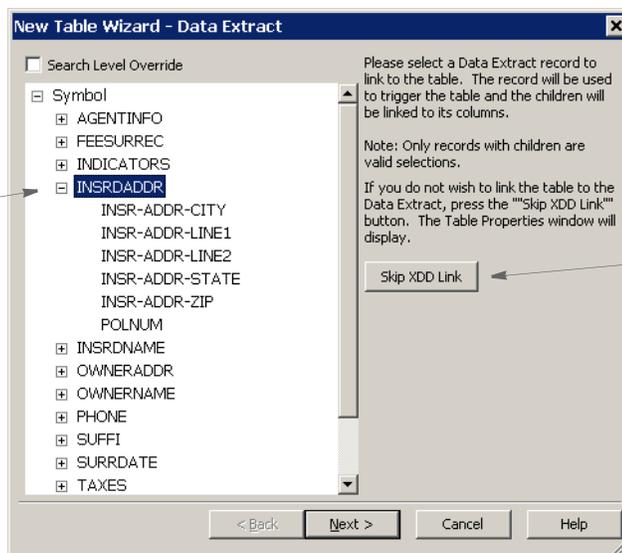
Custom rule name	XDDTrigger
Custom rule parameters	?INSRDADDR

To create a table using the Table wizard, follow these steps:

1. Open a section. Choose the Insert, Table option. The New Table Wizard - Data Extract window appears:

Right-click in the tree to choose a different XDD.

You can expand the data extract parent record to the children within that parent record.

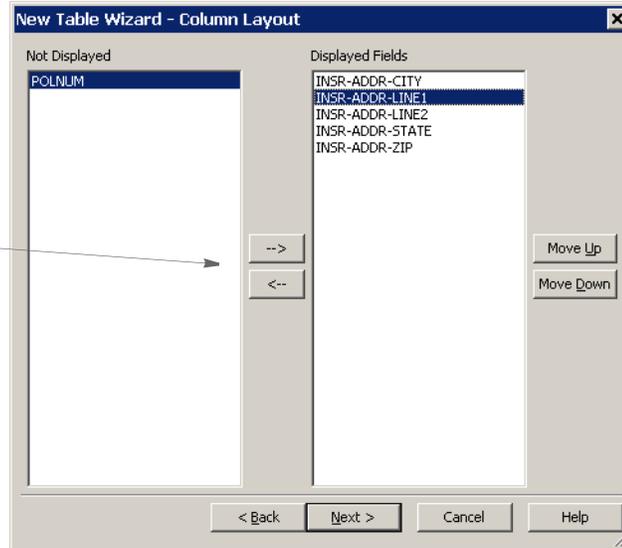


Click here if you do not want to link the table to the Data Extract Dictionary record.

2. Highlight the data extract parent record you want to link to the table, then click Next. The New Table Wizard - Column Layout window appears.

Note If you click the Skip XDD Link button, Studio takes you to the Table window where you can define the appearance of the table. For more information, see *Setting Table Options* on page 239.

Use these buttons to display or hide a field on the table.



Use these buttons to choose the order in which the fields appear on the table.

- Use this window to choose the fields you want to display or hide and to specify the order in which the fields appear on the table. Keep in mind you have to display at least one field to have a table linked to the XDD. Click Next when finished. The New Table Wizard - Summary window appears.



- You have these choices:

Click this box	If you want
Generate XDD Lookups using Name/Parent	XDD searches to use the name/parent lookup instead of the name/unique ID lookup.
Generate Labels for Table Columns	Studio to include a row for column labels. Studio includes the name of the corresponding field you selected for that column as the label. The label text may be truncated to fit in the column. You can change this text as needed. You can also later delete this row.
Open the Table Properties Window	To customize how the table will appear.

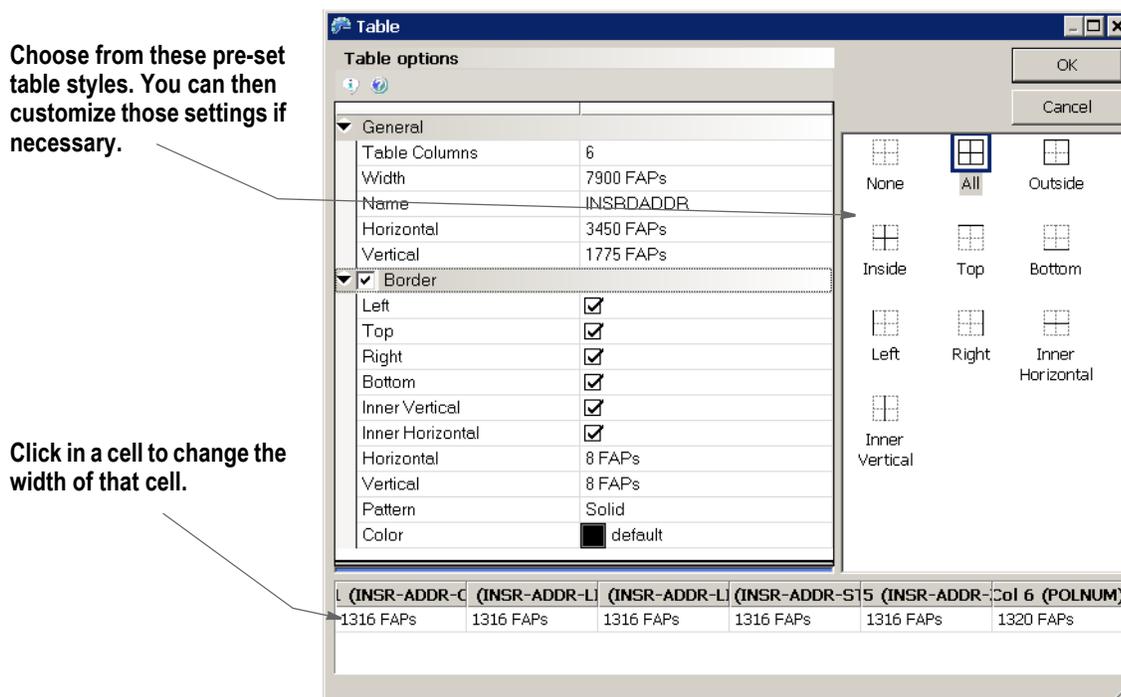
Click Finish to tell Studio to create the table and map the appropriate triggers and fields. The Table window appears.

Note See *Understanding XDD Name References* on page 240 for more information on Name/Unique ID lookup.

- Use the properties on the Table window to customize the appearance of your table. See *Setting Table Options* on page 239 for more information.

SETTING TABLE OPTIONS

Use the table options to customize the position, size, and appearance of the tables you create.



Choose from these pre-set table styles. You can then customize those settings if necessary.

Click in a cell to change the width of that cell.

Field	Description
-------	-------------

General

Table Columns	Specifies the number of columns in the table.
Width	Specifies the width of the table.
Name	Specifies the name of the table. If you use the Table wizard to link the table to the Extract Data Dictionary, the name of the parent record you chose appears here.
Horizontal	Defines the position of the left side of the table.
Vertical	Defines the position of the top of the table.

Border

Field	Description
Left	Click this box to display an exterior left border.
Top	Click this box to display an exterior top border.
Right	Click this box to display an exterior right border.
Bottom	Click this box to display an exterior bottom border.
Inner Vertical	Click this box to display interior vertical column borders.
Inner Horizontal	Click this box to display interior horizontal row borders.
Horizontal	Enter the width of the horizontal rules used in interior and exterior borders.
Vertical	Enter the width of the vertical rules used in interior and exterior borders.
Pattern	Choose the fill pattern for the rule.
Color	Choose the color for the rule.

Understanding XDD Name References

To have fully-qualified names, a Unique ID field was added in version 11.5. The system generates this value and you cannot edit it. When you open an XDD file that has not been converted to include the Unique ID, the system automatically generates this entry.

When you select an XDD item to map, the item appears with the name of the child, a forward slash (/), a number sign (#), and the Unique ID. Here is an example:

```
CHILD/#UNIQUEID
```

The lookup wizards in Studio automatically fill in the proper syntax for you.

The system supports previously-defined mappings that use the name or child/parent naming convention. Documaker Server caches child/parent combinations for legacy support, and also caches the child/#unique ID identifier to enhance performance.

If you are using multiple XDDs with the same child/parent combinations, you will still want to use child/parent search masks. For example, you would use the XDD override at the form list level if a section with XDD mapping appears in multiple lines of business and the extract differs for each line of business. In this case, use the child/parent combination to make sure the proper nodes are found in each file.

Note If you do not use XDD overrides at the form list level and are certain you are not using the older, Child/Parent mapping method, you can add this INI option for better performance:

```
< SymLookup >
  UniqueID = Yes
```

This option tells the system you do not need child/parent name caching for XDD members and therefore saves some time when the system is loading and parsing the extract dictionary during processing.

CREATING TABLES OF CONTENTS, TABLES OF FIGURES, AND INDEXES

You can add tables of contents, tables of figures, and indexes to your form sets to help readers navigate to specific topics. This process consists of these main steps:

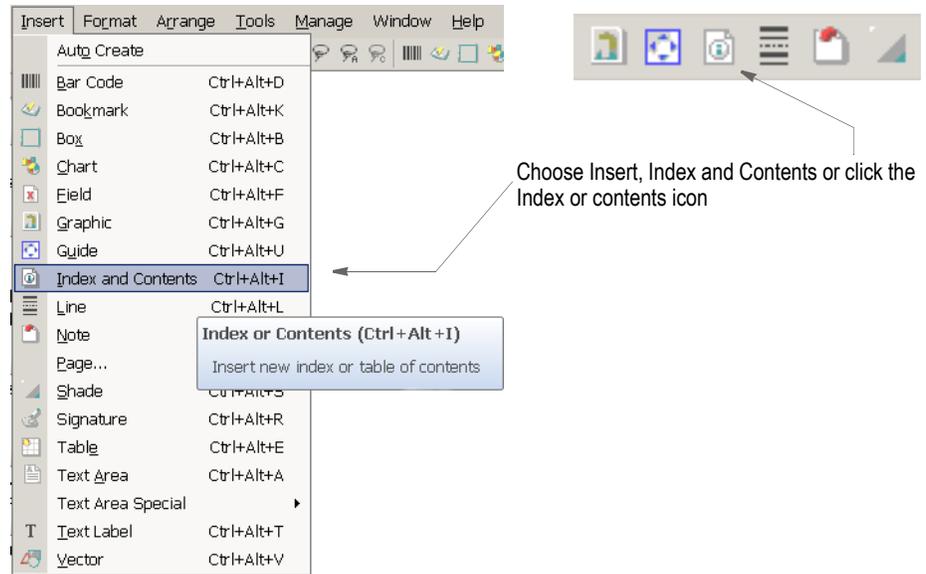
- Adding bookmarks to the sections that comprise the form set. Adding bookmarks is the process of placing markers in the places you want to appear in the table of contents, table of figures, or index and adding the appropriate text for those markers. For more information, see *Adding Bookmarks* on page 206.
- Defining the how you want the generated result to appear. For more information, see x.
- Generating the table of contents, table of figures, or index. For more information, see *Generating Tables of Contents, Tables of Figures, or an Index* on page 246.

DEFINING TABLES OF CONTENTS, TABLES OF FIGURES, AND INDEXES

Before you generate a table of contents, table of figures, or index, you must tell Studio where and how you want the result to appear.

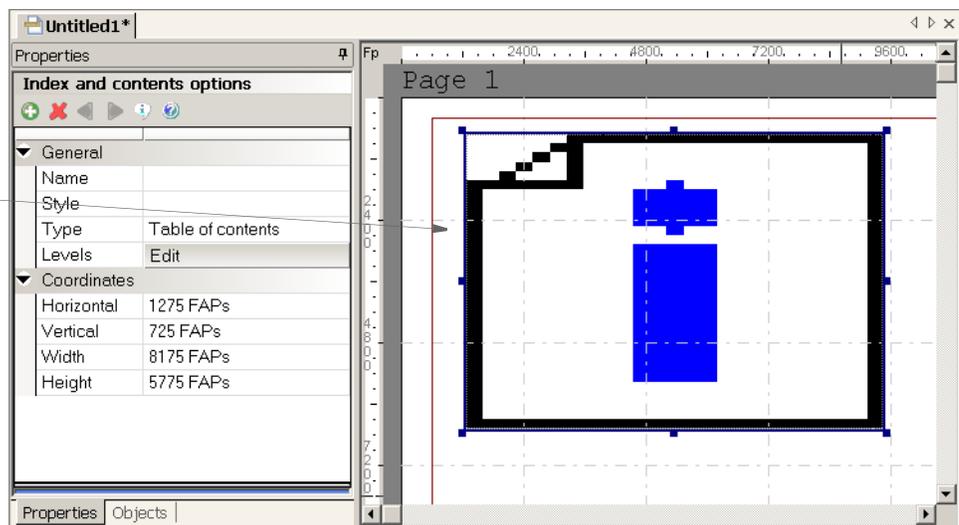
To add a table of contents, table of figures, or index, follow these steps:

1. Open the section that will contain the table of contents, table of figures, or index. This section should be included in a form in the form set.
2. Choose Insert, Index and Contents or click the Index or Contents icon.



The Index and Contents properties window appears.

This icon is the placeholder for the table of contents, table of figures, or index.



Note This placeholder is used for tables of contents, tables of figures, and indexes.

3. Use the Index and Contents options to define the table of contents, table of figures, or index:

Property	Description
General	
Name	(Optional) Enter the name you want to assign to this table of contents, table of figures, or index.
Style	This field only appears if you specified a style file in the BDF file. Select the style you want to use.
Type	Table of Contents is the default, but you can also choose to create a table of figures or an index.
Caption	(Table of Figures only) Enter the caption you want to appear with the table of figures.
Language	(Index only) Select the appropriate language from the list.
Columns	(Index only) Specify how many columns you want for the index.
Column Gap	(Index only) Specify the amount of space between the columns.
Levels	Click the Edit button in this field to specify how you want the system to format each level of your table of contents, table of figures, or index. You can specify up to nine levels for each. For more information, see <i>Defining Levels</i> on page 244.
Coordinates	
Horizontal	Specifies the horizontal position of the upper left hand corner of the table of contents, table of figures, or index. You can change the chart's position by entering coordinates or by using the Left and Right arrow keys.
Vertical	Specifies the vertical position of the upper left hand corner of the table of contents, table of figures, or index. You can change the chart's position by entering coordinates or by using the Up and Down arrow keys.
Width	Specifies the width of the table of contents, table of figures, or index. You can change the width by entering a value or by dragging the chart handles.
Height	Specifies the height of the table of contents, table of figures, or index. You can change the height by entering a value or by dragging the chart handles.

Note The Coordinates options define where the tables or index are generated. You can 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.

4. Click Ok to save your entries or Cancel to exit without making changes.

Defining Levels

Studio lets you specify the formatting for each level in the table of contents, table of figures, or index. The content of these levels is determined by the bookmarks you set. You can control the formatting of each level.

For	The system...
Tables of contents	Matches the level number of the bookmark to the level paragraph format. You can define up to nine table of contents levels. The system also lets you define leading and page numbering.
Indexes	Uses the first three formats defined for the index. When you choose index, you can set the number of columns and the gap between columns. You can also specify the locale for sorting purposes and include Unicode text.
Tables of figures	Uses the first format defined for the figure. When you create a table of figures, you define the prefix text to display and print. For a table of figures, the system generates prefix text, figure number, and a colon, before the text and page number.

Here are some examples:

Here is an example of a two-level index.

projects	
assigning roles	71
granting rights	76

Here is an example of a three-level table of contents.

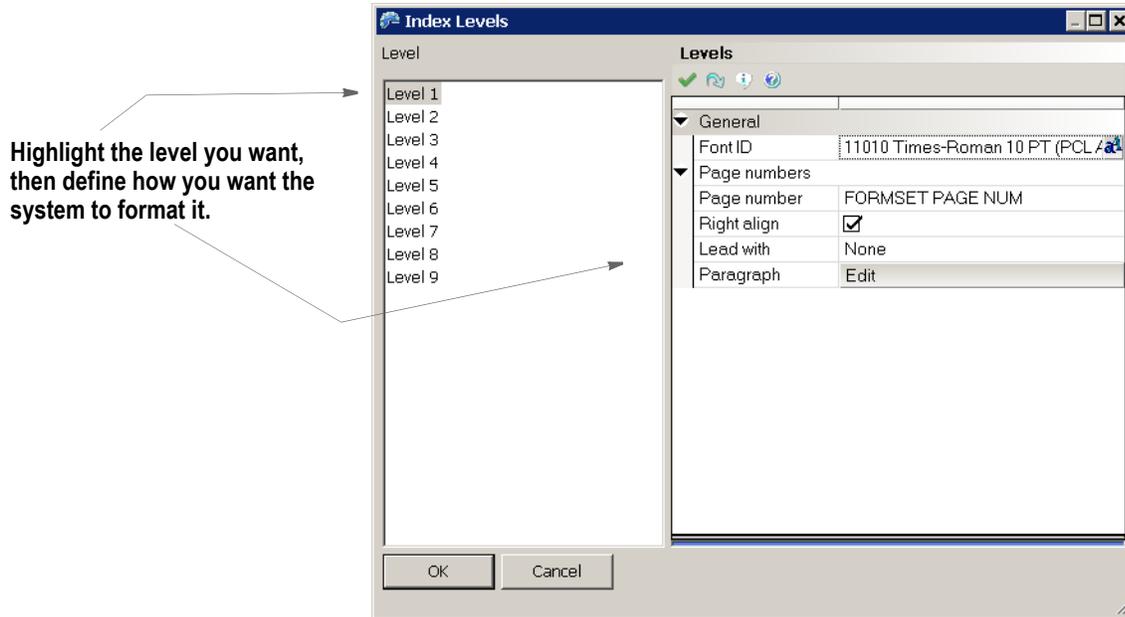
Using Docuflex	33
Charts	33
Creating Docuflex Charts	34
Chart Types	36
Overflow Processing	40
Using Overflow Definitions	41
Step 1: Create the Sections and Members	42
Step 2: Create an Overflow Definition	43
Step 3: Reference the Overflow Definition	44

Here is an example of a table of figures.

Chart 1: Mammal Habitats	2
Chart 2: Intelligence Chart	3

Follow these steps to define levels for your table of contents, table of figures, or index:

1. On the Index and Contents Options window, click the Edit button in the Levels field. The Index Levels window appears.



- Use these fields to define each level. Remember that indexes have two levels (main entries and sub entries) and tables of figures have one level. Tables of contents can have up to nine levels.

Property	Description
General	
Font ID	Select the ID of the font you want to use.
Page Numbers	
Page Number	Select the kind of page numbering you want. You can choose from these options: <ul style="list-style-type: none"> None - no page numbering Form Page Num - Tracks page numbers within the form Formset Page Num. - Tracks page numbers within the form set
Right Align	Check this box if you want the page number right aligned.
Lead With	Select the type of leading you want. Your selection determines what characters fill the space between the end of the bookmark text and the page number. You can choose from None, Dashes (--), Periods (...), or Underscores (___).
Paragraph	Click the Edit button to define the paragraph properties for this level. For more information see <i>Defining Paragraph Properties</i> on page 229.

- Click Ok to save your entries or Cancel to exit without making changes.

GENERATING TABLES OF CONTENTS, TABLES OF FIGURES, OR AN INDEX

Once you add your bookmarks and define where and how you want the compiled table or index to appear, the system will generate the table of contents, table of figures, or index for you when the form set is processed.

Keep in mind that to create tables of contents or figures or an index, all sections must be loaded *before* the print operation executes. Otherwise, the system does not have all the content available and cannot create a complete table of contents, table of figures, or index.

Because 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 Workstation, you would include this option:

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

For Documaker Server (GenPrint), you would include this option:

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

ADDING COMMENTS TO DOCUMAKER OBJECTS

You can attach comments to Documaker objects created in Studio or via the Documaker Add-In for Microsoft Word. Use these comments to note additional work you feel should be completed or for any other purpose.

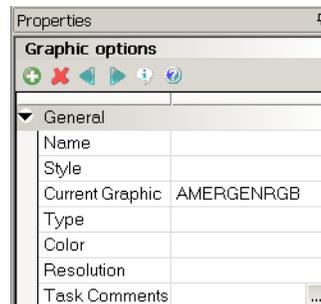
You can add comments to these Studio files and objects:

- Forms
- Sections
- Paragraphs
- Paragraph lists
- Graphics
- Templates
- Fields
- Triggers

To add comments to forms, sections, paragraphs, paragraph lists, templates, and fields, follow these steps:

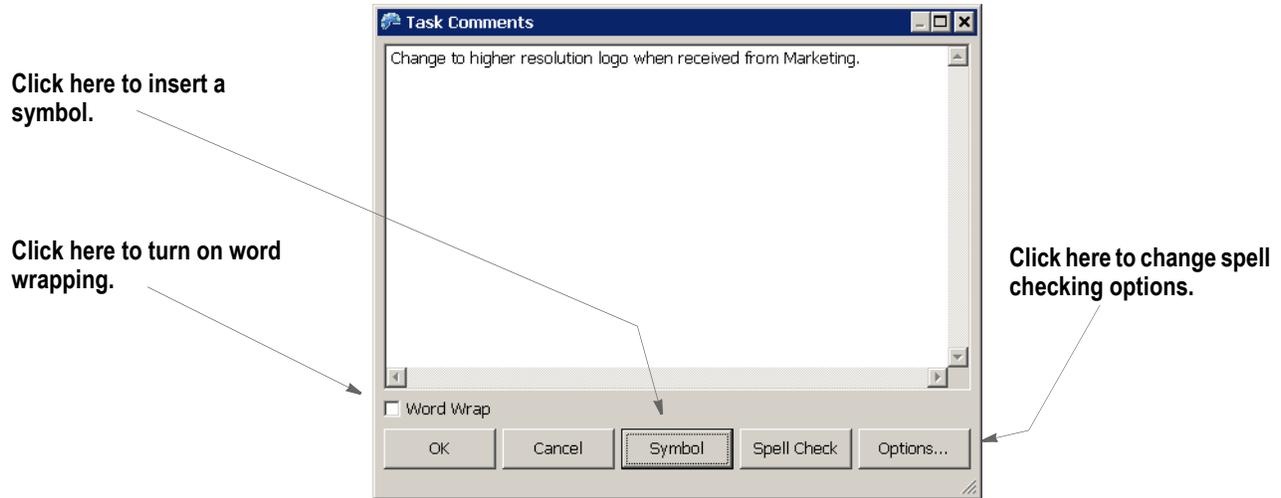
Note For graphics and triggers, you add the comment after you have added the graphic to a section or assigned the trigger to a section. For information on adding comments using the Documaker Add-In, see the Documaker Add-In for Microsoft Word User Guide.

1. Open the item to which you want to add comments.
2. Click the item's Properties tab. Here is an example of the Properties tab for a graphic which has been added to a section:



Click here to display the Task Comments window.

3. Click the ellipsis button in the Task Comments field to display the Task Comments window:



4. Enter up to 1024 characters, including spaces.

Note The actual number of characters you can enter may be fewer in some languages.

You can also use these buttons on the Task Comments window:

Button	Description
Ok	Click Ok to save your changes and close the Task Comments window.
Cancel	Click Cancel to discard any changes and close the Task Comments window.
Symbol	Click Symbol to insert typographical symbols using the Symbol window.
Spell Check	Click Spell Check to check the spelling of the text you entered in the Task Comments window.
Options	Click Options to set spell checking options.

5. Click Ok when you finish entering comments.

Note Use the Finishing Report (Manage, Reports, Finishing Reports) option or the Task List (Tools, Task List) option to see a listing of all the comments entered for a form, section, paragraph, paragraph list, or template.

USING THE TASK LIST

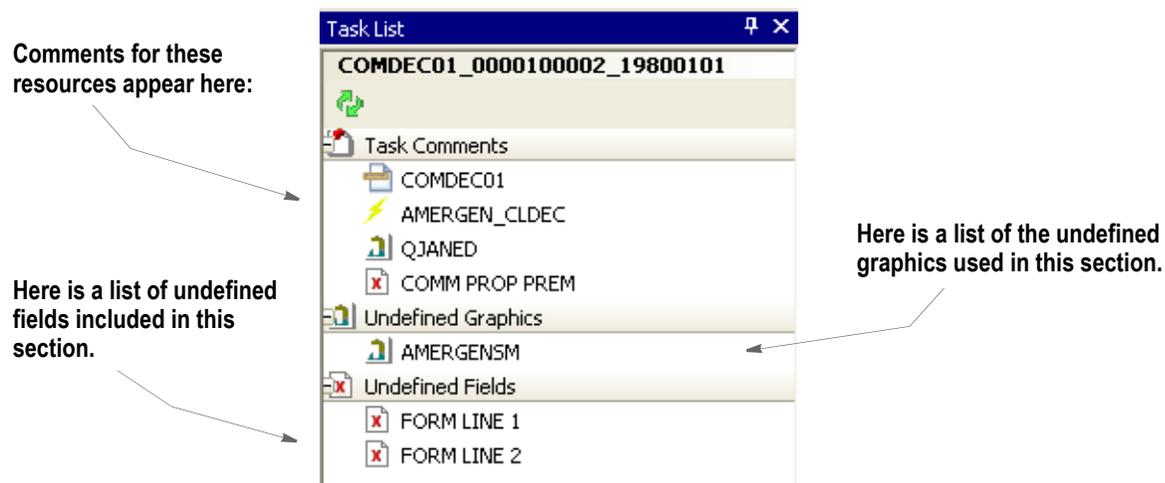
The Task List lets you see a list of the task comments and undefined fields, recipients, graphics, and triggers for the Studio resource you are viewing. You should review each comment and define any undefined items.

To view a Task List, select the Task List option on the Tools menu when you are working with any of these objects:

- Templates
- Forms
- Sections
- Paragraph lists
- Paragraphs

Note You add task comments in Studio's Task Comments field or in the Documaker Add-In for Microsoft Word's Comments field.

Here is an example of a task list for a section:



Note When you use the Manage, Tools, Conversion, Convert Documents to Forms option to convert a Word Add-In document, Studio converts any entries in the Comments field into task comments.

Depending on the resource you are viewing, you could see comments about a field, a graphic, a trigger, or a recipient. Comments could also pertain to the specific resource (section, paragraph, paragraph list, form, or template) you are working with, such as a reminder to add a logo or signature.

Note For more information, see *Adding Comments to Documaker Objects* on page 247.

Below the task comments, Studio shows a list of all undefined objects associated with the current resource.

Object	Appears on the task list if...
Field	An entry for the field does not exist in the Common Fields Dictionary.
Trigger	The trigger is not defined in the SETRCPTB.DAL trigger file.
Recipient	The recipient is not defined in the Business Definition (BDF) file.
Graphic	The graphic does not exist in the workspace library.

Note The Finishing Report provides another way to view task comments. For more information, see *Printing a Finishing Report* on page 461.

CREATING TEXT LABELS

A text label is a short block of text placed in a section. Create a text label when you want to write and place a minimal amount of text such as field labels or headings. A text label can contain up to one line of text.

Note Use a text area if you need to position a longer amount of text and apply word processing features to your text.

If you are inserting several text labels with the same basic properties, you can simplify the task by entering the information for the text label color, effects, font, length, and orientation in the DefaultTextLabel control group. Use the Manage, System, Settings, Options by Group option to locate this control group. Each text label you create will then default to the settings you specify.

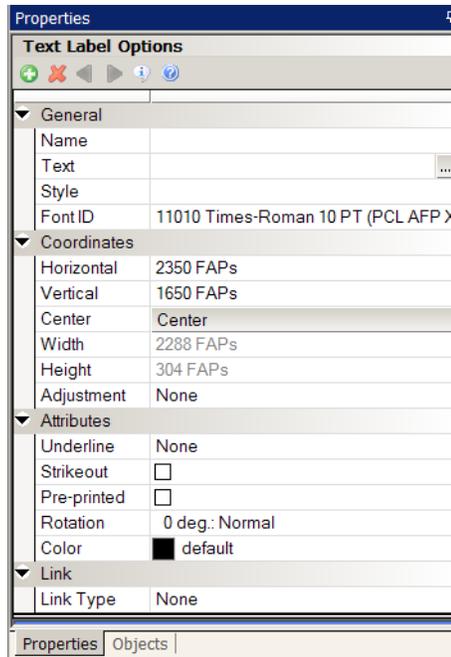
Follow these steps to create a text label:

1. Open a section, then choose Insert, Text Label. After you position your text label the Text Label Options appear.
2. You can type the text for a text label directly into the label or you can type it into the Text property. If you are editing a text label, double-click on the label and then press Esc to exit text label edit mode when you finish.

An active text label remains active when you zoom or rotate the label and will appear inside a border when you are editing it. Once you press Esc, Studio removes the border.



If zooming or rotating the text label makes it difficult to see the text, you can make entries or changes using the Text property in the Properties window. Use these properties to customize the location and appearance of the text label.

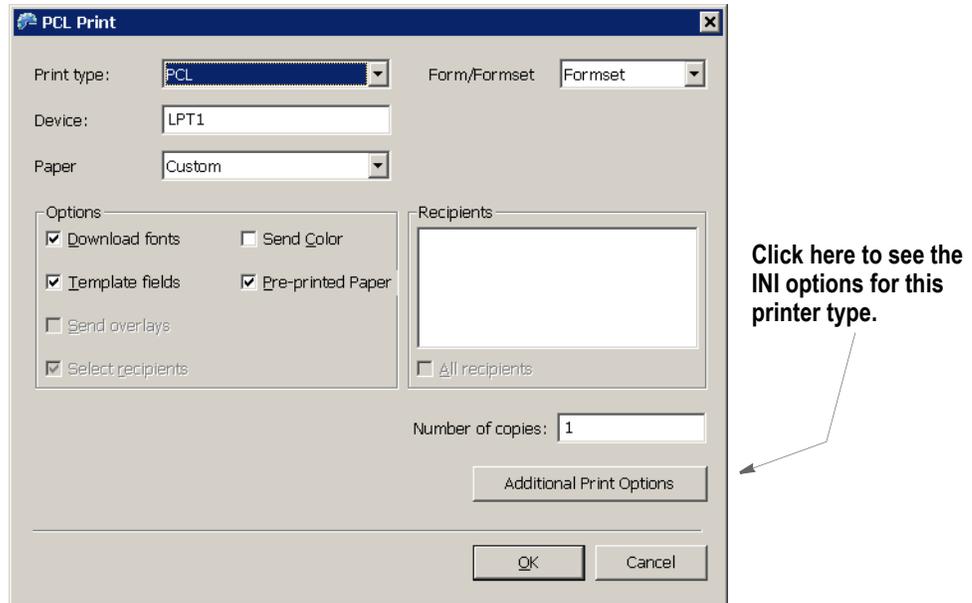


Property	Description
General	
Name	Enter a name for the text label in the Name field. Assign a unique name to facilitate future reference to the text label for editing or other purposes.
Text	Enter the text for the label. If you do not enter text, the system displays Blank Text Label in the image to show the location of the text label. The special characters that you use depend on the available code page.
Style	This field only appears if you specified a style file in the BDF file. Select the style you want to use.
Font ID	Select the font you want to use and click Ok.
Coordinates	
Horizontal	Set the left edge of where you want the label to begin.
Vertical	Set the top edge of where you want the label to begin.
Center	Click Center to move the label to the left/right center of the section. This does not affect the vertical placement of the label.
Width	Specify the width of the text label.
Height	Specify the height of the text label.
Adjustment	Choose from None, Superscript, or Subscript.
Attributes	
Underline	Choose from None, Double line, Single line, Double line (words only) or Single line (words only).

Property	Description
Strikeout	Check this box if you want the text stricken.
Pre-printed	Check this box to tell the system not to print this object when you print the section. Checking this box can be useful during forms creation or when test printing your forms. You may be using letterhead or special paper that has the company name or logo already embossed on the paper. Inserting a text label or graphic on your form in the same location as the company name or logo helps to make sure the space is not inadvertently written over by other objects. These objects would reserve that space on the form.
Rotation	This lets you rotate the text label. You can choose from zero (0), 90, 180, or 270 degree rotation.
Color	Click the ellipsis button to select the color for the label text.
Link	
Link Type	Choose from either none, external, internal, or target hyperlinks. If you chose an option other than none, additional properties appear.
Link Border	Select the type of border (if any) to draw around the hyperlink (affects PDF output only). You can choose from Dash, Solid, Underline, or None.
HTML Reference	Enter the address of the web page. Here is an example: <code>http://www.oracle.com</code>
Parameters	Use this field when producing HTML output to specify additional parameters to an HREF type link, such as a target frame or mouseover behavior. This example causes a web page to open in a new browser window: <code>target="new"</code>

PRINTING A FORM, SECTION, OR PARAGRAPH

To print a form, section, paragraph, or other object you have open, right-click and choose the Print option. Studio shows you the Print window:



Note The fields on this window may vary, depending on the type of printer you chose. This discussion focuses on the PCL printer type.

You can use these fields to define your print job:

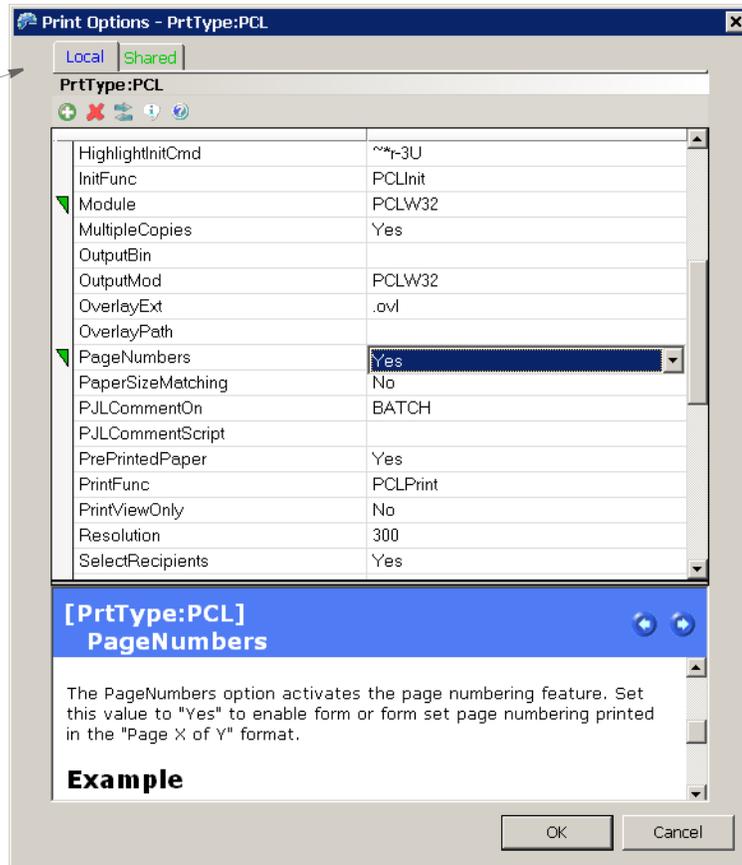
Field	Description
Print Type	Select the printer type. Your choices include: PCL, AFP, Metacode, PostScript, VIPP.
Device	Select the name of the file or device (LPT1) where the print stream should be written.
Form/Form Set	Choose whether you want to print the current form or the entire form set.
Download Fonts	Downloads to the printer, the fonts needed to correctly print the format
Template Fields	Display Xs (or whatever replacement character you chose) in place of actual variable field data on printed policy. Each X indicates a field character. You can see the actual field size by turning this option on.
Send Overlays	Overlays are FAP images compiled into HP printer language. This enhances print speed. Usually, this option is automatically selected.
Select Recipients	Click this box to select recipients.
Fit to Width	Prints the full width of the image on the page.
Send Color	Enables color printing if a supported color printer is used.

Field	Description
Pre-printed Paper	Check this box to include objects marked as pre-printed. All objects except fields can be marked pre-printed on the object's Properties window.
Recipients	Here Studio shows you the recipients selected
Number of Copies	Enter the number of copies you want to print.

Click the Additional Print Options button to see the INI options associated with the current printer type.

The Local tab lets you view and change settings that apply to your computer. The Shared tab lets you view and change settings that serve as defaults for all of your Studio users.

Here Studio provides information about the various INI options.



Note You can also modify these INI options by choosing, Manage, System, Settings. Under Options by Topic, choose Print, the select the printer type you want to modify. The Additional Print Options button just provides a shortcut to these settings from the Print window.

You must be logged in with a user ID which has the appropriate rights to change these settings. Contact your system administrator for more information.

Once you finish reviewing the INI settings and making any necessary changes, click Ok to return to the Print window.

When you have the print options set the way you want them, click Ok to print or Cancel to exit without printing.

USING PRINTCOMMANDER TO CREATE SECTIONS

To use Printcommander to create FAP files (sections), your system must meet the following requirements:

- Windows NT 4.0 or Windows 2000 or higher
- Administrator rights

Setting up Printcommander to create FAP files involves these steps:

- Downloading and installing Printcommander
- Creating a virtual printer
- Configuring the virtual printer
- Installing Tagcommander
- Installing Docucreate
- Configuring NetMove

Once you have performed these steps, you can open a Windows document processor, create a file, and then convert that file into a FAP file by simply printing it to the virtual printer you set up. The following topics guide you through the set up process. See the Printcommander documentation for additional information.

Downloading and Installing Printcommander

Follow these steps to download and install Printcommander:

1. Open a browser and go to the My Oracle Support site at this address:
`http://metalink.oracle.com`
2. Sign in, then click the Patches and Updates tab.
3. Click Simple Search and search for patch number 8483517.

Enter the patch number here.

Simple Search

Advanced Search Quick Links Saved Searches

Search By  Patch Number/Name
(ex. 2123967, 2517300, 11i.FND.H, R12.AD.A.delta.1)

Platform or Language  

Tip  To quickly find the latest patchsets go to Quicklinks.

Results

Patch	Description	Release	Updated	Size
*** No search conducted ***				

- Then click Go. The search results appear.

After you enter the criteria, click Go.

Patchset 8483517

Simple Search Advanced Search Quick Links Saved Searches

Description	PRINTCOMMANDER 6.0.10 INSTALLER
Product	Skywire PrintCommander
Release	6.0
Platform or Language 	Microsoft Windows (32-bit) 
Last Updated	01-MAY-2009
Size	126M (132505934 bytes)
Support Level 	General Support
Classification 	General

Download View Readme View Digest

- Click View Readme to read information about this patch. Click Download to download the patch.
- Once downloaded, unzip the file, then go to the following directory and run the setup.exe program:

```
..\Docucreate_Workstation_FP_6_0_10\Printcommander\6.0.10 - 20090309
```
- Follow the instructions on that appear on your screen to complete the installation.

Creating a Virtual Printer

When you finish installing Printcommander, follow these steps to set up a virtual printer:

- From the Start menu, select Settings, then click on Printers. Next, double-click on the Add Printer icon.
- Select My Computer, then click Next.
- Select DPT1: (DocuCorp Port) from the available ports, then click Next.
- Click the Have Disk button. Go to the `..\docucorp\docucreate\ras_drvr` directory in the Copy From Address field (or where you installed the files) and click Ok.
- Select `oemsetup4.inf` (Docucreate Virtual Raster Printer), click Ok. Then click Next.

Note If the system prompts you to keep an existing driver, select Keep Existing Driver and click Next.

- Enter a name for the printer.
 - If prompted to share the printer driver, select Not Shared. Then click Next.
 - If prompted to print a test page, select No. Then click Next.

This sets up a virtual printer.

Configuring a Docucorp Virtual Printer

Follow these steps to configure your virtual printer:

1. A new printer should appear in the Printers folder. Right click on it and select the Properties option.
2. Click on *Print Processor* and then select *dfxprint* as the processor and *NT EMF 1.003* as the default data type. Click Ok twice. Close the printer window.

Installing Tagcommander

Tagcommander lets you add variable fields in Windows applications.

To install Tagcommander, go to the following director and run the setup.exe program:

```
..\Docucreate_Workstation_FP_6_0_10\Tagcommander\6.3.1.7 - 20090212
```

Follow the instructions that appear on your screen and be sure to restart your computer if requested.

Installing Docucreate

If you have not installed Docucreate or Documaker Studio, go to E-Delivery and select the option to install Oracle Documaker. Follow the instructions that appear on your screen.

Configuring NetMove

Note Windows NT uses the `\winnt\profiles\` directory as the default directory for storing user profiles. Windows 2000 uses the `\documents and settings\` directory as the default for storing user profiles. Printcommander looks for its action file (DFXOCTRL.ACT) in the `\winnt\profiles` directory.

If you are using Windows 2000, create a `\profiles` directory under the Windows directory. Then create a subdirectory which has the name of the logon ID of the Printcommander user, such as: `\winnt\profiles\SJackson`.

Follow these steps to configure NetMove:

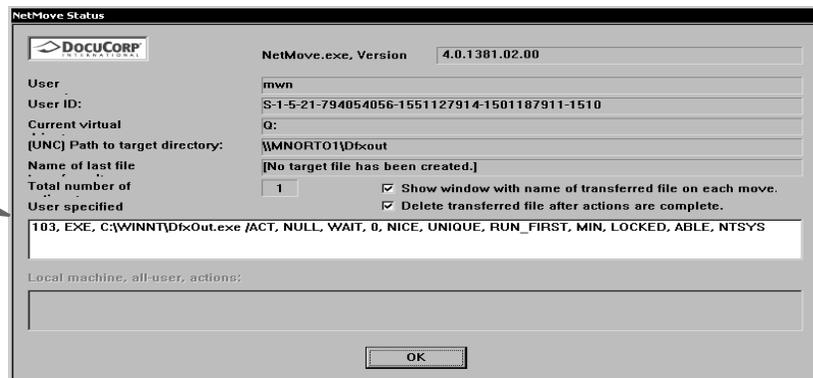
1. Hold down the Ctrl and Shift keys and left click on the green Docucreate Output Control icon in the system tray on the bottom right hand side of your screen.

Click here to start Docucreate's Output Control

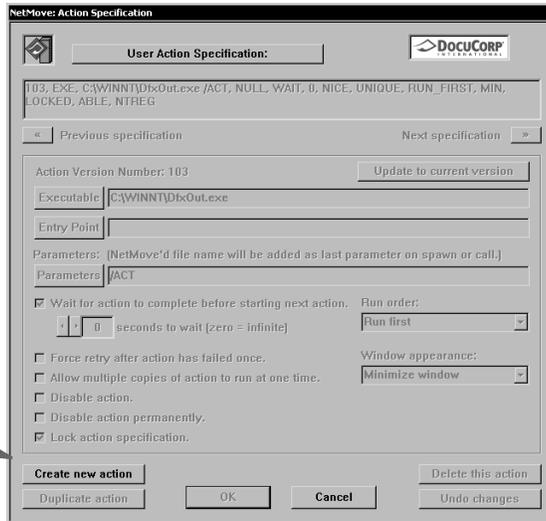


The NetMove Status window appears.

Here is a list of user-specified actions.

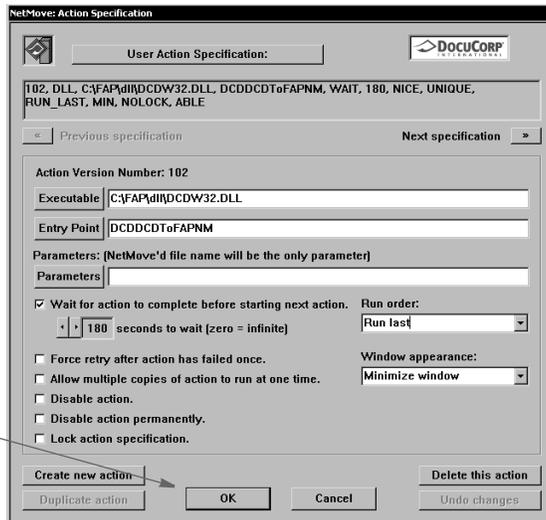


2. Double-click in the box which displays the list of user-specified actions. The Action Specification window appears. On the Action Specification window, click the Create New Action button.



Click the Create New Action button.

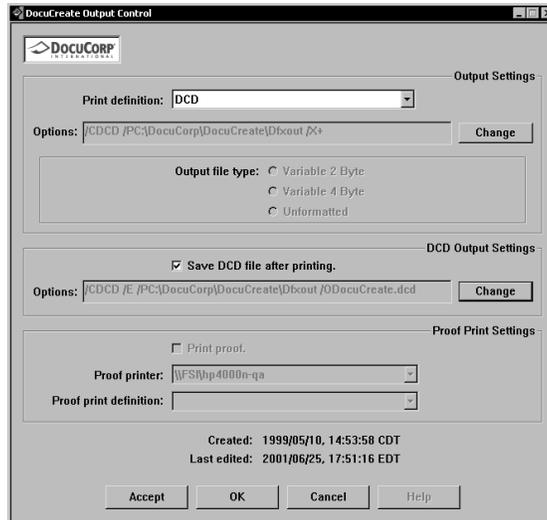
3. Next, uncheck the Lock Action Specification field, then click the Executable button. An Open window appears. In the Files of Type field, select Dynamic-link Library (*.dll).
4. Go to the directory where you installed Documaker Server, such as c:\fap\dll. Select the DCDW32.DLL file in that directory and click Ok.
5. Click the Entry Point button. The Select Exported Entry Point window appears and shows a list of functions from the DCDW32.DLL file. Select this function: DCDDCDTToFAPNM
6. Go to the Run Order field and choose the Run Last option. Your window should look like this:



Click the Ok button when your window looks like this example.

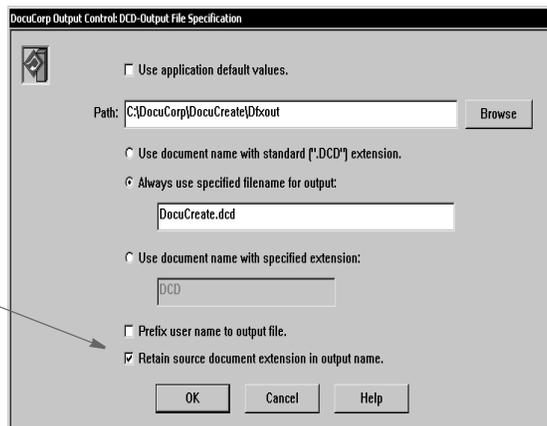
Then click Ok on the Action Specification window and on the NetMove Status window.

7. Left click on the green NetMove icon (without holding down Ctrl+Shift). The Docucreate Output Control window appears. Select DCD as the printer definition.



8. In the DCD Output Settings section, click the Change button.

Make sure this field is checked.



Make sure the option to retain source document extensions in the output name is checked. Then click Ok twice and go back to the Windows desktop.

NetMove is now configured correctly.

Note You must have a FAPCOMP.INI file in the fap\dll directory and your selected library and paths must be valid for the system to create FAP files. Also, make sure the option to retain source document extensions in the output name is turned on.

Creating a FAP File

Follow these steps to create a FAP file using Printcommander:

Note Printcommander is designed to work with word processing systems such as Word, WordPad, WordPerfect, WordPro, and so on. You can also use Excel as long as you do not include BPSD tag references. Inserted objects cannot be viewed by Tagcommander.

1. If you installed Documaker Studio or the Docucreate tools, start Studio or one of the tools, such as Image Editor, and make sure a valid master resource library (MRL) is selected.
 - In Studio, after opening a workspace, you can check this by choosing Manage, System, Settings and then looking under the Resource Path Setup settings.
 - In Image Editor, use the File, Library Setup option to check this information.
2. Open a Windows application such as Microsoft Word. Open or create a sample document.
3. Select Print from the File menu and change the printer to the Docucorp virtual printer. Select Ok. This generates a FAP file. You will see the green NetMove program added to the status bar at the bottom of the screen.

The new FAP file will be created in the c:\docucorp\docucreate\dfxout directory. If you installed to a drive other than C:, FAP and DCD files will be located on that drive.

4. Using a tool such as Windows Explorer, move the FAP file to the forms\ directory of your MRL.

Once you have moved the FAP file, you can delete the DCD files and any SPL files. Docucreate no longer needs these files.

If you receive an error similar to...

```
PFMNAM . Error: Font Not Present for Orientation 0 in PFM: ...
```

Left click on the Netmove icon on the task bar and change the printer definition to DCD. See the topic, *Operating DocuCreate Output Control*, in the Printcommander documentation for more information.

Note For optimal fidelity when converting Printcommander DCD files to FAP files, the system uses static text labels for every word instead of for every line. This helps make sure justified paragraphs come out looking justified in the FAP files. This can also increase the size of the FAP files.

After you convert the file and open it in Documaker Studio, you can select a group of text labels, right click, and choose the Convert to Text Area option to combine the individual text labels into larger, more manageable text areas.

If you notice problems with underlined text, go to the printer settings (File, Print, Properties) and turn on the option to print text as graphics.

The location of this option varies, depending on the type of printer you have selected. This tells the system to treat the underlined text as a graphic and to append the FORMSX.FNT file to the Metacode. A side effect is that by having this option checked, as well as the option to use printer metrics to display the document (Tools, Options, Compatibility), you may see some garbage at the top of a tag. This will not appear in the print out.

Possible Errors

If, when printing a doc file to the raster printer, you get a message similar to the one shown here:



It indicates the system cannot find the DLL files it needs. To correct this error, set up a path to your Documaker DLL files. Normally, the installation will update the path for you, but there are situations where this may not happen, such as if the software is installed on a network drive.

Uninstalling Control Panel, Common Objects, and Printcommander

To uninstall, reverse the order in which you installed the various software tools. Uninstall Printcommander, then Control Panel, then Common Objects.

Updating Control Panel, Common Objects, and Printcommander

To update to a newer version of Control Panel, Common Objects, and Printcommander, uninstall the currently installed version and then install the newer version. You do not lose your raster printer setup but you will lose all NetMove settings so some of these steps would need to be redone.

Chapter 7

Working with Graphics

Graphics are bitmap objects you can place on a section. Studio does not create graphics, but it does let you manipulate them. Graphics are created in graphics applications such as Microsoft Paint or by scanning artwork.

Click Graphics to work with graphics.

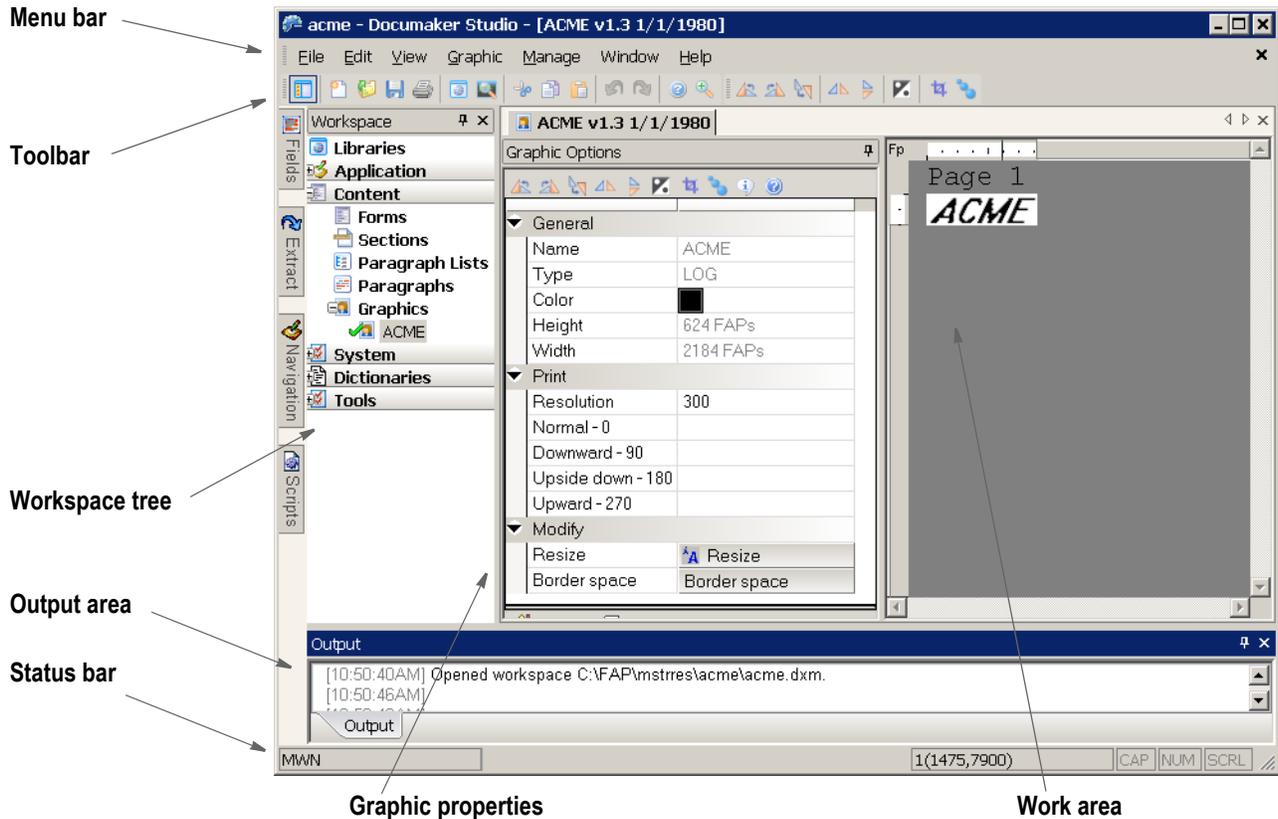
This chapter discusses these topics:

- *Using the Screen on page 266*
- *Managing Graphics on page 274*

USING THE SCREEN

Studio places all the tools you to work with graphics files at your fingertips. The screen is your graphic work area. It is important to become familiar with the general screen layout and parts of the screen. Understanding the screen layout will help you work quickly and efficiently.

The first window that appears when working with graphics is shown here.



Item	Description
Menu bar	The menu bar provides the list of available pull-down menus.
Toolbar	The toolbar contains a row of icons that provide quick access to common options.
Workspace tree	The workspace tree lets you quickly access different items. It also shows which specific resources that are checked out (green check mark), which resources are checked out by another user that you would only have read-only access to (red check mark), and which resources are open in read-only mode or have never been checked into the library.
Output area	The output area whows messages from the system.
Status bar	The Status bar gives the coordinates of the mouse pointer in the work area. The mode of operation, such as ready or edit, also appears here.
Graphic properties	Here Studio shows you the properties for the graphic you are working on.
Work area	This is where you work with the graphic.

USING THE MENU BAR

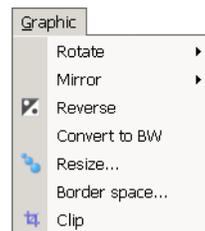
This section introduces you to the pull-down menus which include additional options or are only available when you are working with graphics.

Note For information on the standard menus and menu options which are always available, see *Using System Menus* on page 16.

Menu	Description
Graphic	The Graphic menu provides you with tools to manipulate a graphic image.

Using the Graphic Menu

The Graphic menu provides you with tools to manipulate a graphic. You can rotate, mirror, reverse, convert to black and white, resize, and crop a graphic. When you select Graphic, this menu appears:



Option	Description
Rotate	
Left	Rotates the graphic 90 degrees in a clockwise motion.
Right	Rotates the graphic 90 degrees in a counter-clockwise motion.
Flip	Rotates the graphic 180 degrees.
Mirror	
Horizontal	Lets you create a mirror image of the graphic by flipping it horizontally.
Vertical	Lets you create a mirror image of the graphic by flipping it vertically.
Reverse	Creates a negative image of the graphic file, where dark becomes light and vice versa.
Convert to BW	Converts a color graphic into a black and white graphic. Reducing the number of colors typically reduces the size of the graphic.
Resize	Lets you change the size of the graphic, either by entering a new height and width or by entering a percentage. For instance, if you enter 200%, Studio makes the graphic twice as big. If you enter a new height and width, make your entry in FAP units (2400 per inch).
Border Space	Lets you specify the size of the border that surrounds the graphic. You enter the size of the top, bottom, left, and right borders in FAP units (2400 per inch).

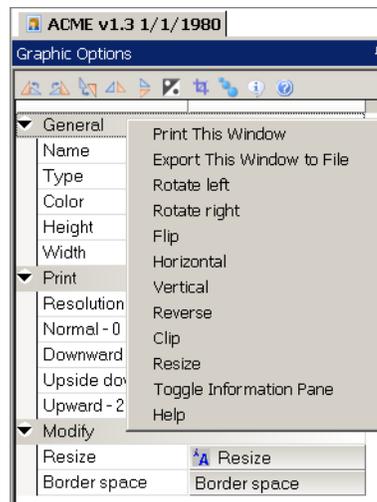
All of these options are available if you right-click in the work area.

Option	Description
Clip	Lets you crop a graphic. Using the Clip option, you draw a box around the part of the graphic you want to keep. Studio deletes everything outside the box.

All of these options are available if you right-click in the work area.

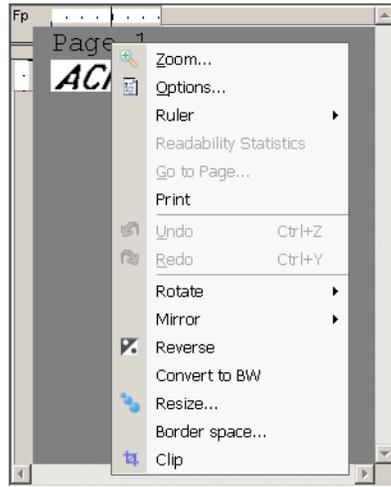
USING THE RIGHT-CLICK MENU

When working with graphic properties, you can right click to see the following menu.



Option	Description
Print	Prints a copy of the properties for this graphic.
Export this Window to File	Lets you save the contents of this window in a CSV file.
Rotate left	Rotates the graphic 90 degrees to the left.
Rotate right	Rotates the graphic 90 degrees to the right.
Flip	Rotates the graphic 180 degrees.
Horizontal	Lets you create a mirror image of the graphic by flipping it horizontally.
Vertical	Lets you create a mirror image of the graphic by flipping it vertically.
Reverse	Creates a negative image of the graphic file, where dark becomes light and vice versa.
Clip	Lets you crop a graphic. Using the Clip option, you draw a box around the part of the graphic you want to keep. Studio deletes everything outside the box.
Resize	Displays the Resize window so you can change the size of the graphic by entering new dimensions or by specifying a percentage increase or decrease.
Toggle Information Pane	Lets you display or hide the information pane.
Help	Lets you display Help information

When working with the actual graphic in the work area, you can right click to see the following menu.



Option	Description
Zoom	Lets you see a larger or smaller version of the graphic.
Options	Lets you modify document viewing options.
Ruler	
Horizontal	Select to display the horizontal ruler. Select again to remove the ruler.
Vertical	Select to display the vertical ruler. Select again to remove the ruler.
FAPs	Select to use FAP units (2400 per inch) on the ruler.
Inches	Select to use inches on the ruler.
Centimeters	Select to use centimeters on the ruler.
Picas	Select to use picas on the ruler.
Points	Select to use points on the ruler.
Readability Statistics	If applicable, generates readability statistics for the current text.
Go to Page	Lets you go to a specific page.
Print	Prints a copy of the graphic.
Undo	Cancel or reverse your last action or choice.
Redo	Repeat the most recent change.
Rotate	
Left	Rotates the graphic 90 degrees in a clockwise motion.
Right	Rotates the graphic 90 degrees in a counter-clockwise motion.
Flip	Rotates the graphic 180 degrees.

Option	Description
Mirror	
Horizontal	Lets you create a mirror image of the graphic by flipping it horizontally.
Vertical	Lets you create a mirror image of the graphic by flipping it vertically.
Reverse	Creates a negative image of the graphic file, where dark becomes light and vice versa.
Convert to BW	Converts a color graphic into a black and white graphic. Reducing the number of colors typically reduces the size of the graphic.
Resize	Lets you change the size of the graphic, either by entering a new height and width or by entering a percentage. For instance, if you enter 200%, Studio makes the graphic twice as big. If you enter a new height and width, make your entry in FAP units (2400 per inch).
Border space	Lets you specify the size of the border that surrounds the graphic. You enter the size of the top, bottom, left, and right borders in FAP units (2400 per inch).
Clip	Lets you crop a graphic. Using the Clip option, you draw a box around the part of the graphic you want to keep. Studio deletes everything outside the box.

USING THE TOOLBAR

The toolbar provides a quicker way to select options that may be listed on a drop down menu. Here is an example of the toolbar shown when you are working with graphics:



Standard toolbar icons

Shown below are the toolbar icons that are always available. The icons are listed as they appear, from left to right.



Icon	Name	Description
	New Workspace	Creates a workspace.
	Open Workspace	Opens a workspace.
	Close Workspace	Closes an open workspace.
	Toggle Workspace	Toggles between displaying and hiding the workspace.
	New	Creates a file.
	Open	Opens a file.
	Save	Saves the open file.
	Print	Prints the current object.
	Cut	Removes an object and places it on the clipboard.
	Copy	Copies an object and places it on the clipboard.
	Paste	Places an object from clipboard onto the current file.
	Undo	Reverses your last action
	Redo	Reverses last undo.
	Help	Displays the Help window

Graphic toolbar icons

Shown below are the toolbar icons that appear when you are working with graphics.



Icon	Name	Description
	Rotate left	Rotates the graphic 90 degrees to the left.
	Rotate right	Rotates the graphic 90 degrees to the right.
	Flip	Rotates the graphic 180 degrees.
	Mirror horizontally	Lets you create a mirror image of the graphic by flipping it horizontally.
	Mirror vertically	Lets you create a mirror image of the graphic by flipping it vertically.
	Reverse	Creates a negative image of the graphic file, where dark becomes light and vice versa.
	Clip	Lets you crop a graphic. Using the Clip option, you draw a box around the part of the graphic you want to keep. Studio deletes everything outside the box.
	Resize	Lets you change the size of the graphic.

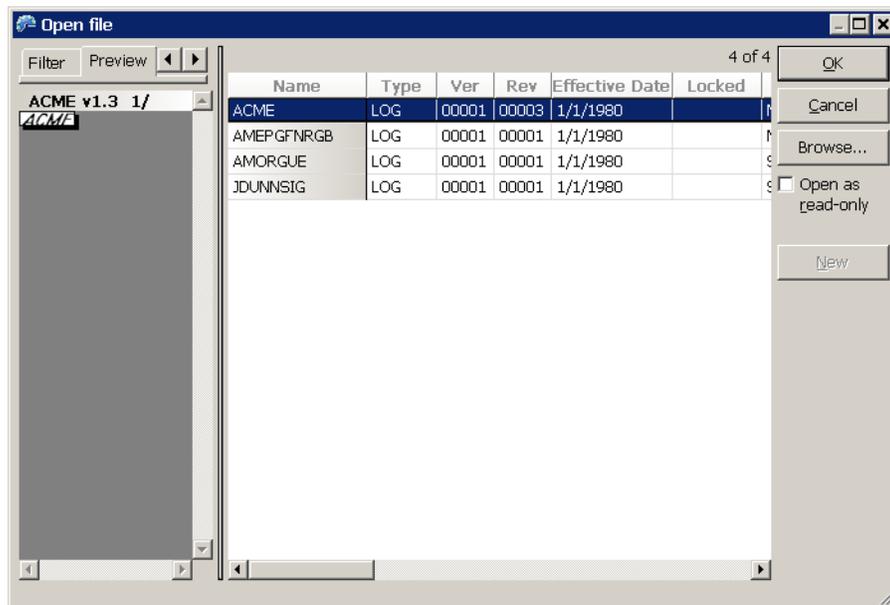
MANAGING GRAPHICS

Once you create a graphic, you can modify it using this option. This gives you an easy way to resize, reverse, rotate, and manipulate the graphic to fit your needs.

Note Studio embeds non-LOG format bitmap files into a multi-line text field. By embedding the graphic file, you can pass the WIP or archived form set to another user who only has a copy of the basic master resource library (MRL), which does not include a copy of the graphic.

The system does not embed LOG type bitmaps because those are Documaker standard format graphics files and would normally be stored in the MRL.

When you choose Graphics, here is an example of the window that appears:



You can use the Browse button to open the following types of files:

- Documaker graphic (.LOG)
- Bitmap files (.BMP)
- IOCA (.ICA)
- TIFF (.TIF)
- JPEG (.JPG)
- Portable Network Graphics (.PNG)
- Xerox font (.FNT)
- Xerox font data (.FNT)
- Xerox image (.IMG)
- Xerox logo (.LGO)
- Paintbrush (.PCX)

- Overlay (.OVL)
- Windows Metafile (.WMF)
- Encapsulated PostScript (.EPS)
- AFP page segment (.SEG)

You can use the Manage, System, Settings option to specify these DPI settings in the BitmapLoaders control group:

```
< BitmapLoaders >
  WMFImportDPI =
  VectorGraphicImportDPI =
```

Option	Description
WMFImportDPI	(Optional) Enter the DPI (dots per inch) at which the graphic will be rendered into raster format. The default is 100. The maximum DPI is 2400.
VectorGraphicImportDPI	(Optional) Enter the DPI (dots per inch) at which the graphic will be rendered into vector format. The default is 100. The maximum DPI is 2400.

Note The level of support varies from format to format.

Not all PNG formats are supported. Specifically, PNG supports a transparency attribute that is not supported. Studio only supports opaque (non-transparent) bitmaps.

PNG also supports a variety of color bit patterns, such as - 1, 2, 4, 8, 16, 24, and 32 bits per pixel. Studio does not support all of these formats, but a 3rd-party PNG library included with Studio will convert the bitmaps into a pixel format the system does support.

Studio converts PNG files into bitmaps when printing. It handles monochrome (1 bit), 16-color (4-bit), 256-color (8-bit), and full color (24-bit) bitmaps. However, not all printers can support these, so make sure you use bitmaps appropriate for your printer.

PDF only supports 1-bit and 24-bit bitmaps. So, some types of PNG files may look different when you create PDF files.

Positioning JPEG Objects

You can use the Z-Index option to position JPG objects:

```
< PrtType:HTML >
  IMG_ZIndex = 100
```

The z-index indicates the stacking order of objects based on the order in which those objects appear in the HTML file. Higher values place objects closer to the front while lower values place them further to the back. Objects with the same value are stacked based on the order in which they appear in the HTML source.

For instance, a positive value positions an object above text that has no defined z-index. A negative value would place the object below the same text.

If you omit this option or leave it blank, Studio will not layer objects.

Importing Color Bitmaps

Studio can import the following types of color bitmaps:

- TIFF

- BMP
- LOG
- JPEG

To import another type of bitmap, first convert it to monochrome (black and white). Both gray-scale and color bitmaps can be converted to monochrome bitmaps.

GDI and PCL print support color printing. Select File, Print to display the Print window. Select the printer type and device and click the Send Color option to print in color.

Note Color bitmap and JPG files are only supported on Windows and UNIX systems if you are using the GenPrint program to print to color GDI, PCL, or PostScript printers or if you are producing color PDF or RTF output.

Color graphics and TIF files are converted to monochrome for Metacode and AFP output, but are supported on Windows, UNIX, and z/OS systems.

Reverse black and white file types

Studio can also import reversed black and white bits. A flag tells you whether a bitmap was stored in the opposite method. You can automatically reverse the bit if necessary.

Converting Files

You can use the File, Save As, option to convert a graphic to one of the following file types:

- Documaker graphic file (*.LOG)
- TIFF (*.TIF)
- Bitmap (*.BMP)
- Xerox font (*.FNT)
- Xerox image (*.IMG)
- VIPP image (*.VPP)
- AFP page segment (*.SEG)
- JPEG (*.JPG)

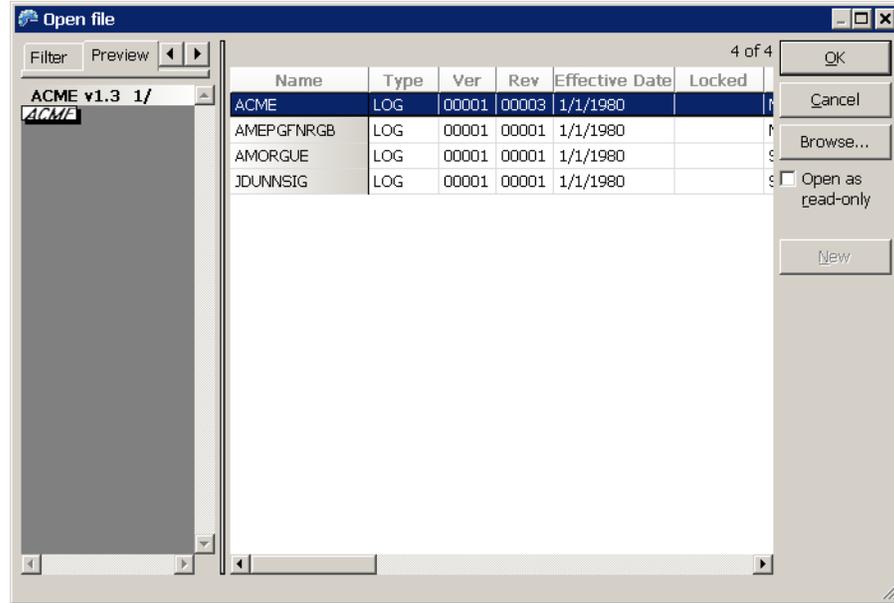
To do so, open the graphic and choose the File, Save As option. Select the type of file you want to convert to in the Save File As Type field. You can also convert files using the Manage, Conversion, Convert Graphic Files option. If you have a large number of LOG files to convert into JPEG files, see the [Docutoolbox Reference](#) for information on using the LOG2JPG utility.

Note An AFP page segment can contain a mixture of text, section objects, and graphics data objects and can be placed anywhere on a presentation page. Programs can request page segments for presentation in a page or overlay. Page segments are used for graphics, signatures, and boilerplate. A Xerox font can also include multiple graphics or signatures.

OPENING A GRAPHIC

Use the following instructions to open a graphic file for editing.

1. Choose Graphics from the Manage menu or from the Workspace tree. The Open File window appears.

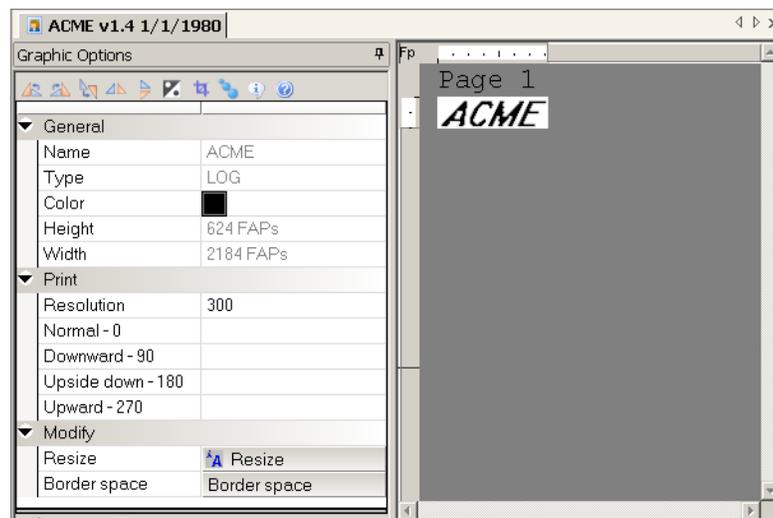


Note This option specifies the directory for storing graphics:

```
< MasterResource >
  LogoLib =
```

If you omit this option, Studio looks for graphics in the path defined for FormLib.

2. Highlight the file you want and click Ok. The graphic appears. If you want to open a file that has not been checked into the library, click Browse.



Importing a Signature or Graphic

You can import a signature or graphic out of a Xerox font that contains multiple signatures or graphics by clicking Browse and following these steps:

1. Select Xerox font data in the Files of Type field. For the import to be successful, you must know the characters which comprise the signature or graphic and you must match case when entering the characters to be imported.

Be sure you do not select Xerox font files (*.FNT) as the file type by mistake.

2. Select the file containing the signature or graphic and click Open.
3. Studio asks you to enter the characters to convert from the font file. You must enter the exact characters in the exact case for the system to properly import the signature or graphic. For example, if the characters JHNDOE are used to print a signature from John Doe, you can't enter *jhndoe*.
4. Click Ok.

EDITING A GRAPHIC

Once you open a graphic, you have these options when you right click on a graphic:

- Undo
- Redo
- Rotate
- Mirror
- Reverse
- Convert to BW
- Resize
- Border Space
- Clip

Option	Description
Undo	Use the Undo option to cancel or reverse your last action or choice, and restore the graphic to its prior appearance. Since the system keeps track of all actions, you can select Undo several times to undo your most recent changes one at a time.
Redo	Use this option to repeat an action you have just reversed. For instance, if you rotate a graphic, then click Undo to restore it to its original position, you can then click Redo to rotate the graphic again.
Rotate	The Rotate option takes the current graphic and turns it in the direction you choose. You can rotate the graphic left or right or flip the graphic.
Mirror	Use this option to create a mirror image of the graphic. If you want to create a mirror image of the graphic, you can select either vertical or horizontal.

Option	Description														
Reverse	The Reverse option creates a <i>negative</i> of the graphic by taking the open graphic and switching the black and white colors.														
Convert to BW	Use the Convert to BW option to convert a graphic to black and white. Reducing the number of colors typically reduces the size of the graphic. The system displays the Color Selection window when you choose this option. If you select Print in Color, the system prints the graphic in the color you selected.														
Resize	Displays the Resize window so you can change the size of the graphic, either by entering a new height and width or by entering a percentage. For instance, if you enter 200%, Studio makes the graphic twice as big. <div data-bbox="646 575 1268 1052" data-label="Image"> <table border="1" data-bbox="667 680 1247 911"> <tr> <td>Resize</td> <td>By Dimensions</td> </tr> <tr> <td>Height</td> <td>624 FAPs</td> </tr> <tr> <td>Width</td> <td>2184 FAPs</td> </tr> <tr> <td>Percent</td> <td>0</td> </tr> </table> </div> <p data-bbox="651 1058 1360 1085">If you enter a new height and width, make your entry in FAP units (2400 per inch).</p>	Resize	By Dimensions	Height	624 FAPs	Width	2184 FAPs	Percent	0						
Resize	By Dimensions														
Height	624 FAPs														
Width	2184 FAPs														
Percent	0														
Border space	Displays the Border Size window so you can specify the size of the border that surrounds the graphic. You enter the size of the top, bottom, left, and right borders in FAP units (2400 per inch). <div data-bbox="646 1220 1268 1696" data-label="Image"> <table border="1" data-bbox="667 1325 1247 1556"> <tr> <td>Border</td> <td>White Border</td> </tr> <tr> <td>Adjust space type</td> <td>White Border</td> </tr> <tr> <td>Top</td> <td>72 FAPs</td> </tr> <tr> <td>Left</td> <td>72 FAPs</td> </tr> <tr> <td>Right</td> <td>72 FAPs</td> </tr> <tr> <td>Bottom</td> <td>72 FAPs</td> </tr> <tr> <td>Clear values</td> <td>Clear values</td> </tr> </table> </div>	Border	White Border	Adjust space type	White Border	Top	72 FAPs	Left	72 FAPs	Right	72 FAPs	Bottom	72 FAPs	Clear values	Clear values
Border	White Border														
Adjust space type	White Border														
Top	72 FAPs														
Left	72 FAPs														
Right	72 FAPs														
Bottom	72 FAPs														
Clear values	Clear values														
Clip	Lets you crop a graphic. Using the Clip option, you draw a box around the part of the graphic you want to keep. Studio deletes everything outside the box.														

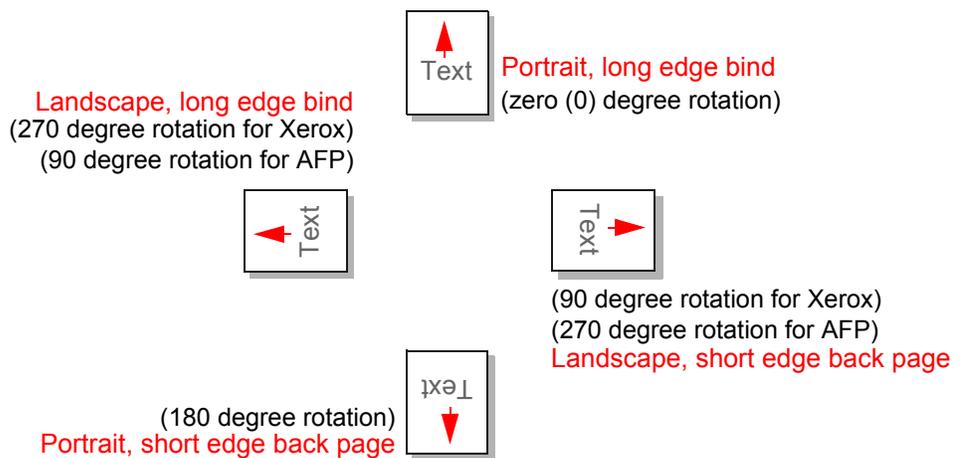
SETTING ROTATION NAMES

Use the Normal, Downward, Upside Down, and Upward fields in properties to enter graphic file names if you want to print a rotated graphic using an AFP or Xerox Metacode printer. The names you enter here are the names of the printer resources (files stored on the printer) which are to be used when printing a graphic which has a rotation of 0, 90, 180, or 270 degrees. One system graphic file can equate to four printer resource files.

Note PCL and PostScript printers do not have printer resource files, so they can automatically rotate a graphic. You only need to enter these names if you are using an AFP or Xerox Metacode printer.

AFP and Metacode printers cannot automatically print a rotated graphic. Therefore, you must create four printer resource files to support a graphic printed in any rotation. Since you cannot rotate a graphic directly on a section, you need rotated graphics when using graphics on a non-portrait long edge section. For example,

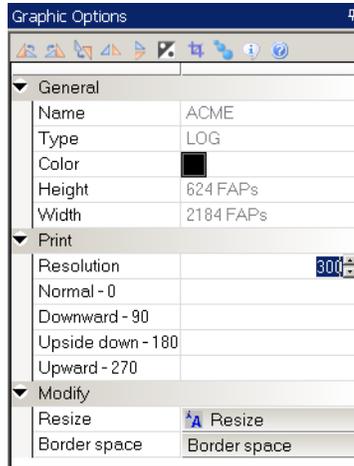
- Portrait, long edge bind requires a zero (0) degree (normal) rotation printer resource.
- Portrait, short edge back page requires a 180 degree (upside down) rotation printer resource.
- Landscape, long edge bind requires a 270 degree (upward) rotation printer resource for Xerox printers and a 90 degree rotation printer resource for AFP printers.
- Landscape, short edge back page requires a 90 degree (downward) rotation printer resource for Xerox printers and a 270 degree rotation printer resource for AFP printers.



Note If you are using a Metacode printer, *do not* enter more than six characters per name.

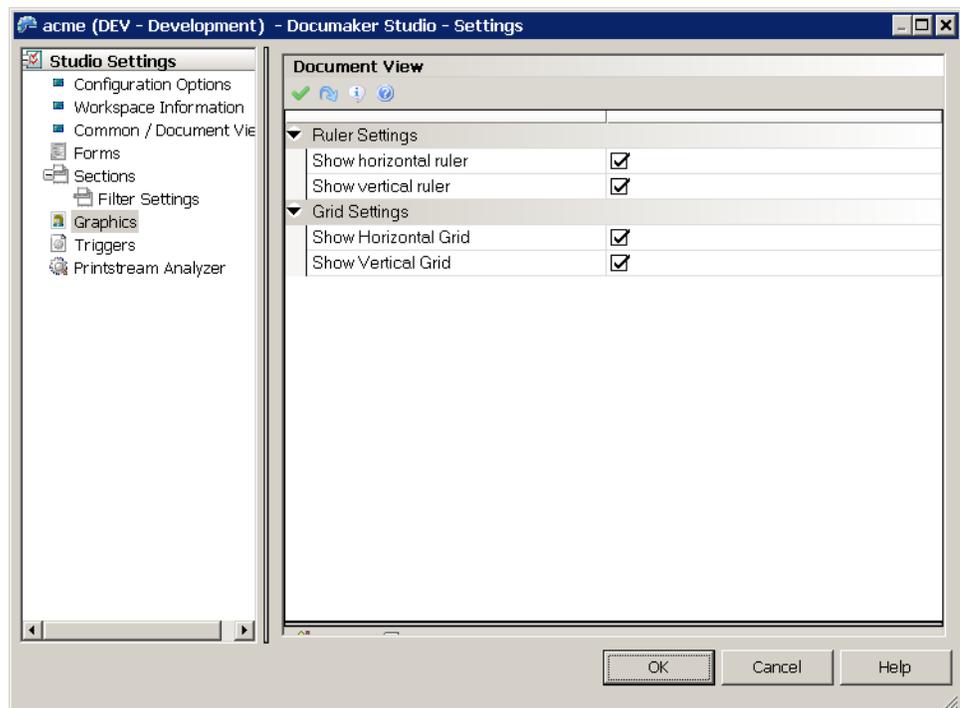
Changing the Resolution

The Resolution field in the Properties lets you change the dpi (dots per inch) of the graphic. Resolution determines the clarity of the object for use with different printer types. The more dots per inch, the sharper and more defined the object appears.



SETTING RULER AND GRID OPTIONS

You can right-click and select Options to turn on or off rulers and the grid.



The Grid Settings let you display or hide horizontal and vertical grid lines on the graphic. Grid lines show you the edges of the graphic. Select the appropriate Show field to display a grid line. Click Ok to exit and apply your changes.

By right clicking on a graphic and choose the Rulers option, you can set the display options for the horizontal and vertical rulers. Select the ruler's units of measure in the field to the right. You can choose between FAP units ($1/2400$ of an inch), points, inches ($1/6$, $1/8$, or $1/10$), centimeters, or picas.

Chapter 8

Creating Scripts

You can write DAL scripts to automate system tasks. Click on Scripts to open the DAL script editor.

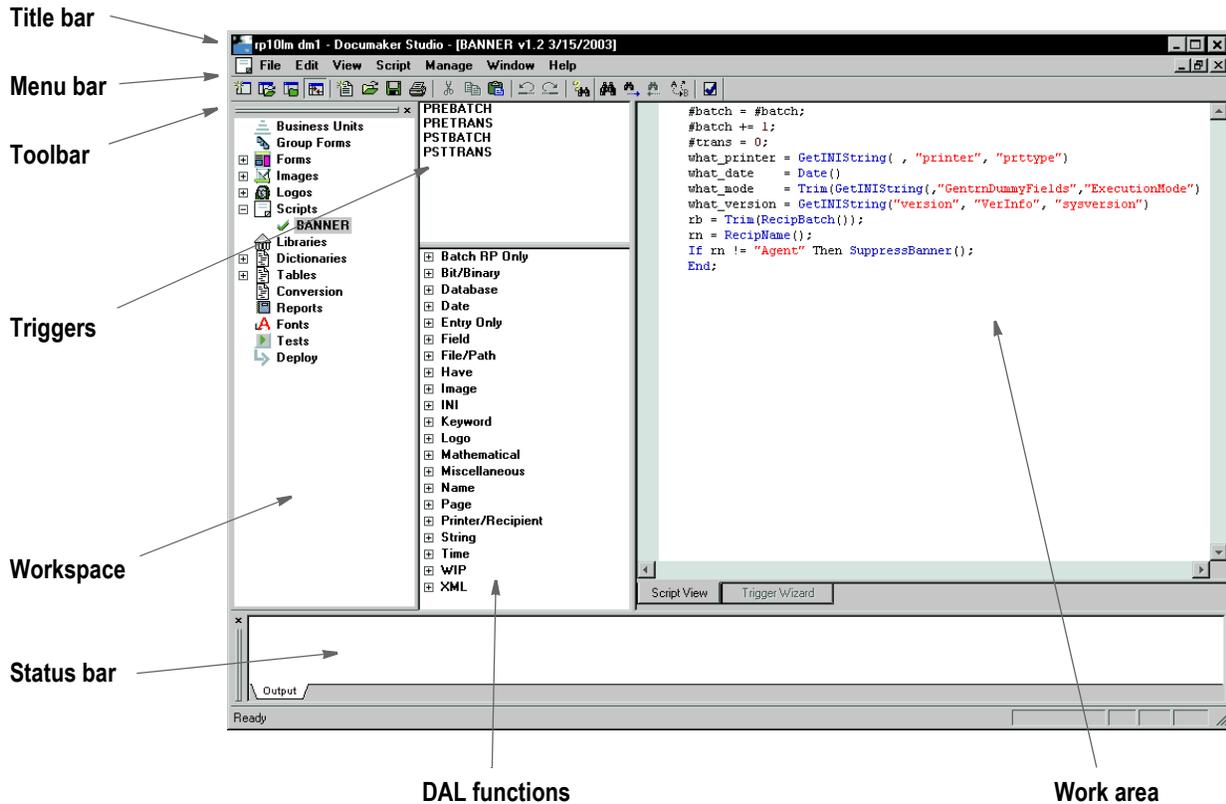
This chapter discusses this topics:

- *Using the Screen on page 284*
- *Overview on page 290*
- *Checking Syntax on page 292*

USING THE SCREEN

Studio places all the tools you need to work with DAL scripts at your fingertips. The screen is your work area. It is important to become familiar with the general screen layout and parts of the screen to work quickly and efficiently.

The window that appears when working with a DAL script is shown here.



Item	Description
Title bar	The title bar displays the name of the workspace you have open, followed by Documaker Studio, and then the name of the DAL script you have open.
Menu bar	The menu bar provides the list of available pull-down menus.
Toolbar	The toolbar contains a row of icons that provide quick access to common options.
Triggers	Here Studio lists the triggers associated with this DAL script.
Status bar	The Status bar gives the coordinates of the mouse pointer in the work area. The mode of operation, such as ready or edit, also appears here.
Workspace	The workspace lets you quickly access different items. It also shows which specific resources that are checked out (green check mark), which resources are checked out by another user that you would only have read-only access to (red check mark), and which resources are open in read-only mode or have never been checked into the library.

Item	Description
DAL functions	This panel lists the various categories of DAL functions and procedures. You can expand each category to see the DAL functions and procedures included within. To insert a function or procedure, simply highlight it and then drag it to the appropriate location within the script.
Work area	This is where you work with the DAL script.

USING THE MENU BAR

This section introduces you to the pull-down menus which include additional options or are only available when you are working with DAL scripts.

Note For information on the standard menus and menu options which are always available, see *Using System Menus* on page 16.

Menu	Description
Edit	To this menu, Studio adds the Syntax Check option.
Script	The Script menu provides you with options you can use when writing or editing a DAL script.

Using the Edit Menu

The Edit menu provides standard word processing options you can use as you create and modify DAL scripts. These options let you copy, cut, delete, and paste text or select all of the text in the script. Edit options also let you find and replace text.

When you select Edit when working with DAL scripts, this menu appears:

Undo	Ctrl+Z
Redo	Ctrl+Y
Cut	Ctrl+X
Copy	Ctrl+C
Paste	Ctrl+V
Delete	Ctrl+D
Select All	Ctrl+A
Find...	Ctrl+F
Replace...	Ctrl+H
Syntax Check	Ctrl+K

The grayed-out options are standard Edit menu options.

Option	Description
Syntax Check	Lets you check your script for syntax errors. See x for more information.

Using the Script Menu

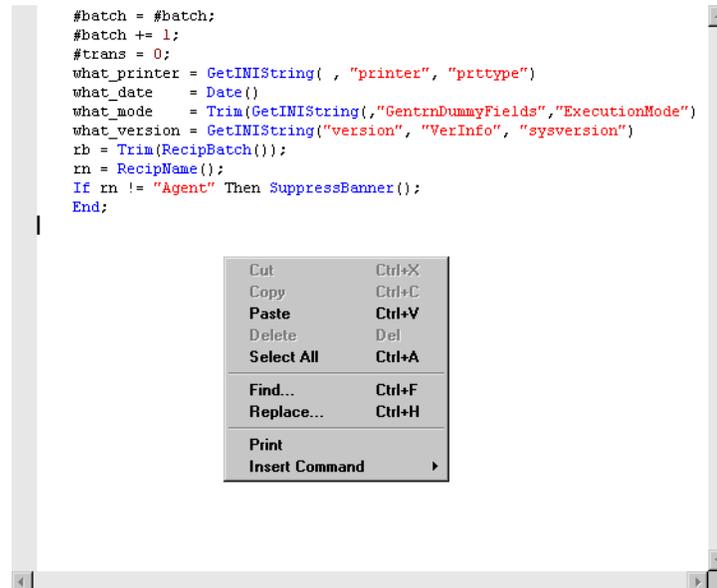
The Script menu provides you with options you can use as you work with DAL scripts. You can insert and delete functions or convert the file into a library, trigger file, or script. When you select Script, this menu appears:



Option	Description
Insert New Function	Use to insert a DAL function.
Delete Function	Use to remove a DAL function.
Convert to Library	Use to convert this file into a library file.
Convert to Trigger File	Use to convert this file into a trigger file.
Convert to Script File	Use to convert this file into a script file.

USING THE RIGHT-CLICK MENU

When working with scripts, you can right click to see the following menu.



Option	Description
Cut	Remove a highlighted selection and place it on the clipboard. Use Paste to insert the selection into another part of the script.
Copy	Make a copy of the selection and place it in the clipboard. Use Paste to insert the selection into another part of the script.
Paste	Insert the contents of the clipboard at the cursor location.
Delete	Erase the selection. The selection is not stored on the clipboard so you must immediately select Undo if you change your mind.
Select All	Select all of the script. You can then cut or copy the script.
Find	Select to find a text string in the file.
Replace	Select to find a text string and replace it with another text string.
Print	Send the current file to the printer.
Insert Command	Use to insert a DAL function or procedure. When you choose this option a list of categories appears. You can then choose the appropriate function or procedure from the appropriate category.

Note To learn more about individual DAL functions and procedures, see the [DAL Reference](#).

USING THE TOOLBAR

The toolbar provides a quicker way to select options that may be listed on a drop down menu. Here is an example of the toolbar shown when you are working with scripts:



Standard toolbar icons

Shown below are the toolbar icons that are always available. The icons are listed as they appear, from left to right.



Icon	Name	Description
	New Workspace	Creates a workspace.
	Open Workspace	Opens a workspace.
	Close Workspace	Closes an open workspace.
	Toggle Workspace	Toggles between displaying and hiding the workspace.
	New	Creates a file.
	Open	Opens a file.
	Save	Saves the open file.
	Print	Prints the current object.
	Cut	Removes an object and places it on the clipboard.
	Copy	Copies an object and places it on the clipboard.
	Paste	Places an object from clipboard onto the current file.
	Undo	Reverses your last action
	Redo	Reverses last undo.
	Help	Displays the Help window

Script toolbar icons

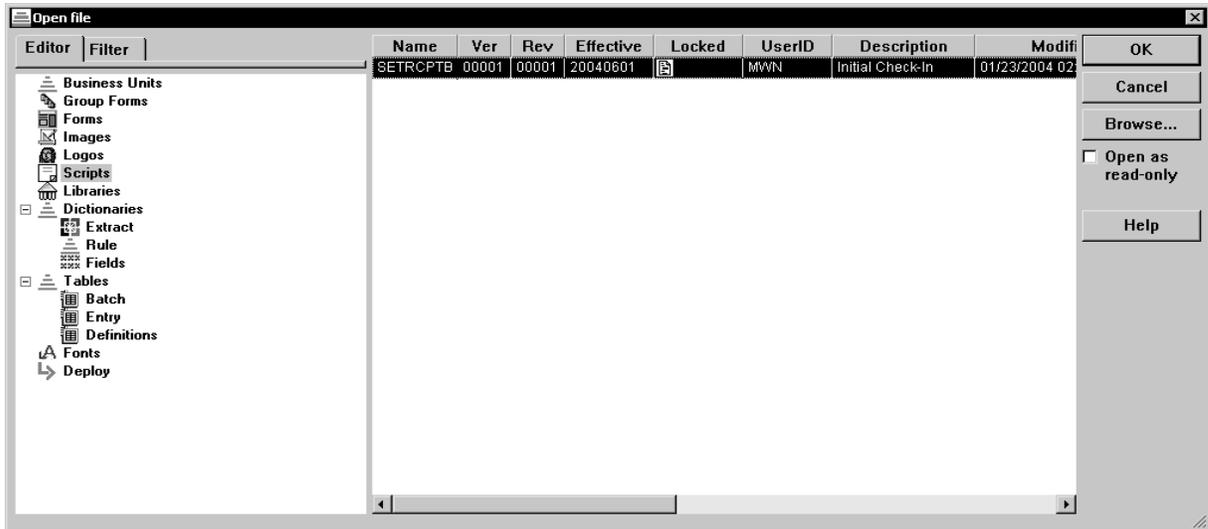
Shown below are the toolbar icons that appear when you are working with scripts.



Icon	Name	Description
	Find	Select to find a text string in the file.
	Repeat	Select to repeat the last action.
	Find previous	Select to return to reverse the search.
	Replace	Select to find a text string and replace it with another text string.
	Check syntax	Select to check your script for syntax errors.

OVERVIEW

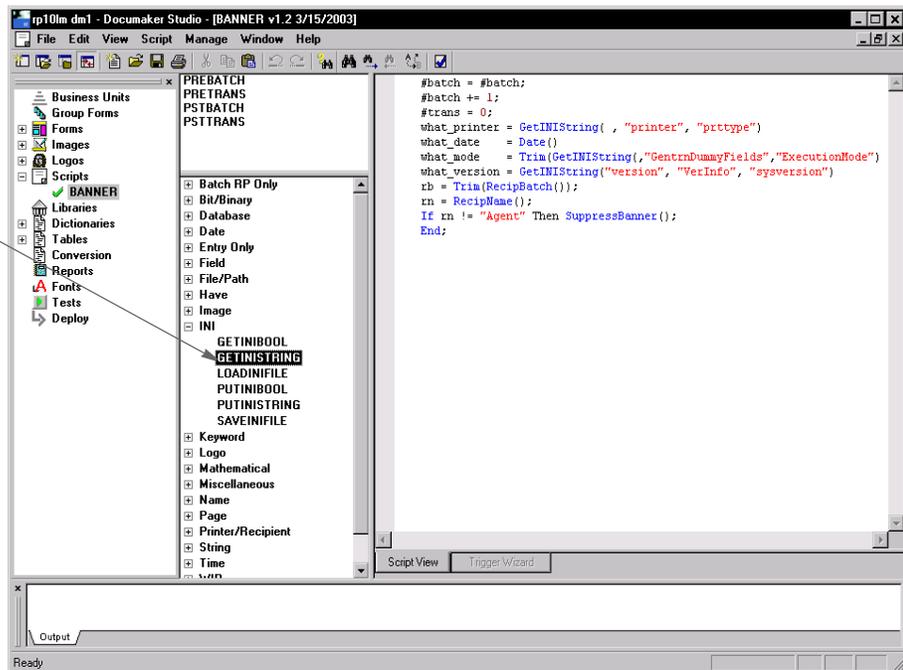
Here is an example of the window that appears:



After you select the script you want to edit, the system displays in a window similar to the one shown here:

You can insert DAL functions and procedures from here.

Simply highlight the function or procedure you want to insert, then drag it to the appropriate location in the script.



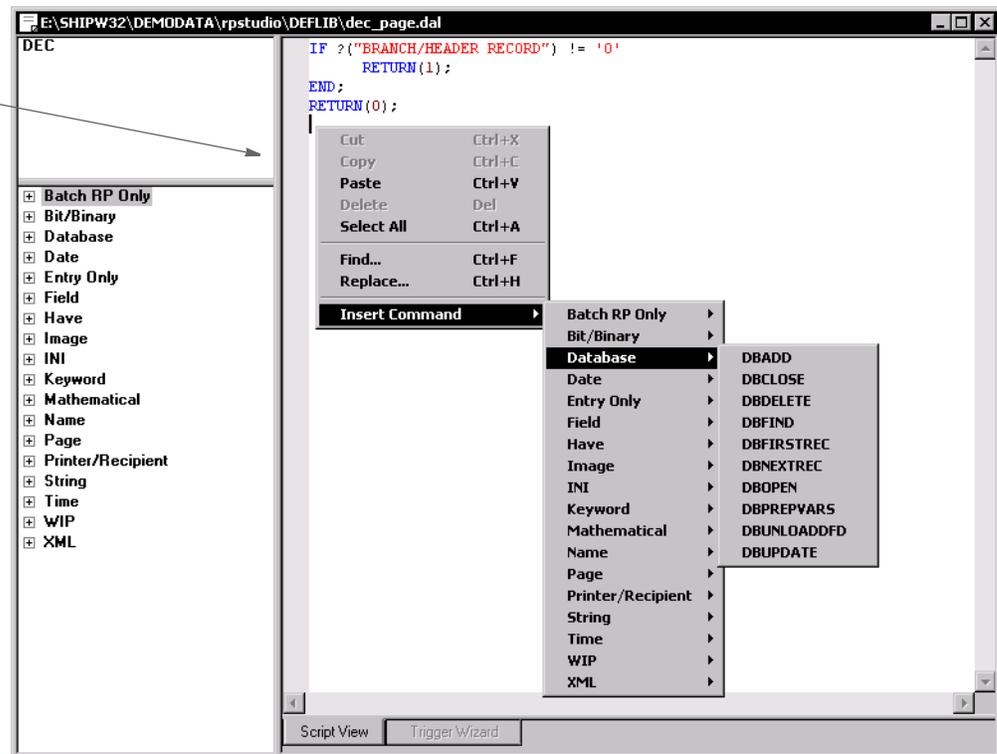
When editing a DAL script, you can right click to display a menu of editing options. This menu provides another way to insert a DAL function or procedure, as shown below:

Right click to display the context menu, then highlight the Insert Command option.

You can choose from a list of DAL function categories.

When you highlight a category, the applicable functions appear.

Highlight the one you want to insert it into your script.



Note To learn more about individual DAL functions and procedures, see the [DAL Reference](#).

CHECKING SYNTAX

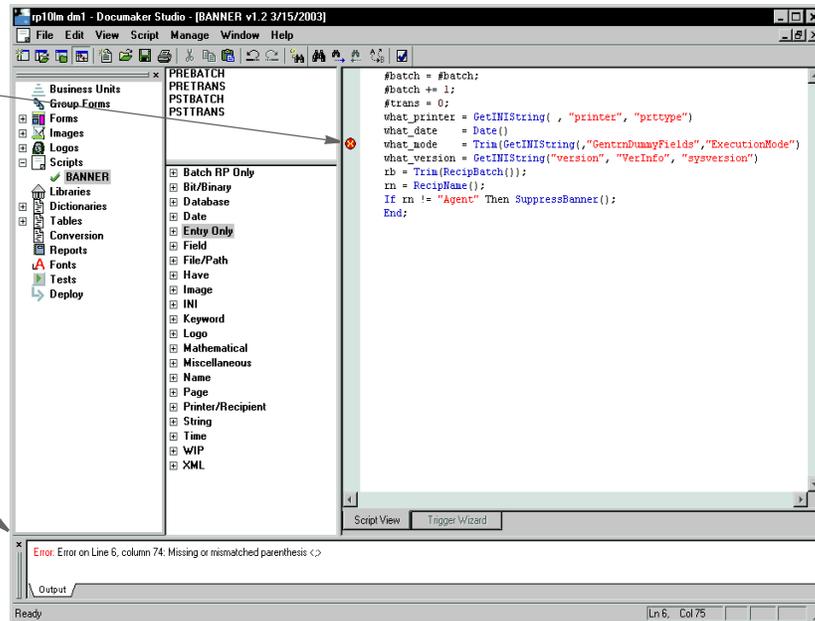
When creating a DAL script, you can easily check for syntax errors by choosing the Edit, Check Syntax option.

Note You can also check for syntax errors by clicking this icon on the toolbar: 

Here is an example of what you will see when you check the syntax of a DAL script:

This icon tells you there is a syntax error on this line of the DAL script.

Here, the system explains the error



The following icon marks each line with a syntax error:



An explanation of the syntax error appears in the status bar



Studio tells you the line and column number to make it easier to find the error. This table discusses the types of things Studio checks for:

Error	Description
Out of memory	This can indicate you are running low on memory or have some other internal memory problem. Try closing any other applications or open documents within the program to free memory before trying again.
Open failure on script file	This indicates the program could not locate or open the referenced script file. This could mean the file name or path is incorrect or that the file did not contain the expected DAL information. You would typically not receive this error during a syntax check.

Error	Description
Syntax error	<p>This message indicates that something on or about the line where it was encountered that did not meet the syntax requirements.</p> <p>Check parentheses for matches; that quoted strings are defined correctly; and that semicolons are used to separate multiple statements on a single line.</p>
Wrong number of parameters	<p>This is an unlikely message to receive during a syntax check. The syntax check does not actually execute internal functions and therefore only minimal parameter validation occurs. This message indicates that a DAL function or procedure was called with the incorrect number of parameters. Check the documentation of the function for more information.</p>
Wrong type of parameter	<p>This message indicates that a parameter passed to a DAL function or procedure is not of a type that can be used by that routine. In most cases, DAL automatically handles the conversion of variables types, however, there are some variable types it cannot convert.</p> <p>For instance, a list variable can be assigned to another list variable, but it cannot be converted to a number or string. Therefore passing a list variable to a function expecting a string might cause such an error.</p> <p>This is an unlikely message to receive during a syntax check. The syntax check does not actually execute internal functions and therefore only minimal parameter validation occurs.</p>
Invalid or unknown symbol	<p>During a syntax check, Studio can only verify that the defined symbol follows the correct naming requirements. The syntax check cannot verify whether functions or variables you might use will exist at runtime.</p> <p>Encountering this error during a syntax check means the defined item does not conform to naming requirements. If a DAL variable or function you reference does not exist at the point where it is referenced during the runtime execution of the script, you will not receive this message at that time.</p>
Invalid assignment statement	<p>This means you have created an invalid arithmetic or assignment statement. Typically this indicates you forgot the assignment operator or omitted an expected arithmetic operator.</p> <p>For instance, AVAR=BVAR is correct syntax; using AVAR==BVAR would generate the invalid assignment statement message because the operator == is unknown. AVAR BVAR (with no operator specified) also generates the invalid assignment statement message.</p>
Cannot modify target	<p>This message indicates the script is incorrectly trying to assign or change data associated with a constant (or keyword) item that cannot be changed.</p> <p>An example of this is to use a statement like 1 = AVAR. You cannot change the constant value of 1 in this case. A situation where you might encounter this error is when you attempt use a variable that happens to have the same name as a defined function or DAL routine.</p>
Unexpected internal error	<p>This message indicates the script processing has resulted in a condition that was not expected. It is difficult to predict when you might receive this error, although it is probably more likely to be encountered during runtime execution of the script than during a syntax check.</p> <p>Typically, this message indicates you have an expression or function parameters that do not resolve to an expected state.</p>
Missing or mismatched parenthesis	<p>This message indicates the number of opening and closing parentheses used within the statement do not match. For each open parenthesis, there should be a closing parenthesis.</p>

Error	Description
Invalid IF statement	This message indicates there is some problem with the construction of the IF statement. It could mean you have forgotten a parenthesis, or a quote around a constant, or that a subsequent keyword limited to use within an IF statement was encountered outside of the IF structure. This would include these keywords: ELSE, ELSEIF, and END.
Unexpected after ELSE	(24) The ELSE statement is the final branch of an IF statement. This error will be generated if another ELSE or ELSEIF condition is found that appears to belong to the same IF statement.
CONTINUE outside of WHILE	(25) The CONTINUE statement is only valid when used within a WHILE-WEND statement group. This error may not always be caught by syntax checking, but will be during runtime execution.
BREAK outside of WHILE	(26) The BREAK statement is only valid when used within a WHILE-WEND statement group. This error may not always be caught by syntax checking, but will be during runtime execution.
Invalid WHILE statement	(27) This message indicates a WHILE statement did not end with a WEND or that a WEND statement was encountered without a WHILE.
Unexpected WEND statement	(28) Similar to the invalid WHILE statement message, this message indicates a WEND statement was encountered incorrectly. This error may not always be caught by syntax checking, but will be during runtime execution.
Unexpected end of script	This message indicates the end of the script was encountered before finding an expected keyword or that the script was empty. Such a condition can occur if you fail to terminate an IF statement with an END or a WHILE statement with a WEND before encountering the end of the script. This error may not always be caught by syntax checking, but will be during runtime execution.
Invalid expression syntax	This error indicates the expression did not result in a value as expected. Such a situation can occur if you called a function or procedure that expected a value and none was returned. Since the syntax check does not actually execute internal functions, it is not always possible to identify this problem prior to runtime. This error may not always be caught by syntax checking, but will be during runtime execution.
Attempt to divide by zero	This message indicates the expression results in an attempt to divide a numerator by zero. This is an undefined mathematical situation and is flagged as an error by most processors. In most cases, this message appears during runtime execution. The syntax checker only catches this situation if the expression explicitly uses a zero constant as a divisor.
No result value returned	This message is unlikely to be generated during a syntax check. The message indicates that a function or called DAL script or routine did not place a return value on the internal stack. If the code is calling a DAL routine, make sure all the RETURN statements in the called script include a return parameter and that there is a RETURN statement at the end of the script.
Statement label already used	This message indicates the same label used as a destination for a GOTO statement has been defined in multiple places within the same script. This error may not always be caught by syntax checking, but will be during runtime execution.

Error	Description
Unknown statement label	<p>This message indicates the label defined as the destination for a GOTO statement could not be located within the defined script. Typically, this means you have incorrectly identified the label in the GOTO statement or have omitted the label destination in the script. Make sure to name the destination label on the GOTO statement with the trailing colon.</p> <pre>GOTO BOB (incorrect) GOTO BOB: (correct)</pre>
Invalid statement label	<p>This message indicates the defined label does not conform to the definition requirements of a GOTO label. Statement labels have the same requirement as string variables and must begin with a letter and be no more than 64 characters in length, including the terminating colon on the end of the name.</p>
Function out of place	<p>This message indicates you have called a function which must return a value, but did not define the expression in a way to use the resulting value.</p> <p>For instance, the HaveForm function must return a zero or one to indicate if the defined form is included in the document set. If you use this function without capturing or testing the returned value, this message appears.</p> <p>Since the syntax checker does not actually execute internal functions, it is unable to identify when such a situation might exist. These errors occur during the runtime execution of the script.</p>
Illegal parameter value	<p>This message indicates you have called a function or procedure with an invalid parameter value.</p> <p>Since the syntax checker does not actually execute internal functions, it is unable to identify when such a situation might exist. These errors occur during the runtime execution of the script.</p>
Table has not been opened	<p>(22) This message indicates a function or procedure was called with a parameter naming a table, such as a database file, that was not explicitly opened.</p> <p>Since the syntax checker does not actually execute internal functions, it is unable to identify when such a situation might exist. These errors occur during the runtime execution of the script.</p> <p>If you encounter this error, make sure that the DBOPEN statement was not skipped due to an IF statement or GOTO operation.</p>

Chapter 9

Managing Resources

This chapter discusses how you can store the resources that comprise your forms in libraries and use Studio and other tools to manage those resources.

Included in this chapter is information on:

- *Overview on page 298*
- *Creating Libraries on page 316*
- *Working with Libraries on page 344*
- *Using the LBYPROC Utility on page 382*
- *Troubleshooting on page 384*

OVERVIEW

You use a variety of resources to build a form set. These resources can include sections (FAP files), graphics (LOG files), and processing scripts (DAL scripts). Studio creates a library into which you can place the resources and provides features that let you manage those resources. For instance, Studio lets you date stamp a resource so it will not be used until its *effective date* is reached.

As you work with resources, such as sections or graphics, Studio lets you check resources into and out of the library. When a resource is checked out, Studio marks that resource as being locked and prevents others from checking it out until it is checked back in.

Documaker Workstation uses the library management capabilities to get the appropriate version of a resource, based on the effective date of the documents being generated.

Likewise, Documaker uses the library management capabilities to load the correct resources from the library when needed. And, the Documaker Bridge for Docupresentation uses library management capabilities to retrieve the correct version of each of the resources used when the document was archived.

To better understand library management, it is important for you to become familiar with the following topics:

- *Terminology on page 299*
- *Concepts on page 301*
- *How It All Works on page 309*
- *Managing Workflow on page 315*

TERMINOLOGY

Before you begin managing your resources in libraries, there are some terms you should understand and concepts you should be familiar with.

Check in

The process of putting a resource back into the library after you have checked that resource out.

Check out

The process of selecting a resource from a library. When you select the resource, it is locked. It remains locked until you check the resource back in.

Effective dates

A field in the library index that is associated with a resource in the library. This field indicates the date at which the resource is available for use, or when it will become effective.

Expire

The process of designating a library resource as being unavailable for use beginning with a supplied expiration date.

Extract

The process of making a copy of a resource in a library and writing that copy to disk.

Libraries

Refers to the physical files in which the resources are stored. Logically, a library is divided into an index portion and a data portion.

Modification dates

A field in the library index that contains a timestamp indicating the date and time the resource was last modified or checked in to the library. This field is in hexadecimal format in the index but is formatted to a more readable format when displayed in the tools.

Revisions

A field in the Library Index that indicates the minor change number. The revision number is generally incremented by one each time a resource is checked out and back in. If, however, the version number is incremented, the revision number is reset to 1 for that version. A revision number consists of five digits, such as *00001*.

Note When a resource is expired, the revision field is set to *EXP*.

Response files

In Documaker terms, a file created by the LBRYMGR utility that contains commands and data. These commands and data are then read by Studio (or the LBRYMGR utility) and processed accordingly.

Promotion

The process of copying (or promoting) resources from one library to another library, based on criteria you specify. Resources from the source library that have a newer modification date than resources, with a corresponding name, in the target library, are copied to the target library.

Unlock

The process of removing the lock from a resource that you have previously checked out. Only the user who locked the resource can unlock it.

Versions

A field in the library index that indicates the major change number. The version number is only incremented upon check in when you check the Increment Version field. If you want to change the effective date for the resource you are checking in, you are forced to check the Increment Version field before you are allowed to type a new effective date.

In Documaker 10.2 and higher, a version number consists of five digits, such as *00001*.

CONCEPTS

In addition to the terminology you should be familiar with, there are several concepts:

- *Understanding Libraries on page 301*
- *Processing with Effective Dates on page 303*
- *Retrieving Resources with Version and Revision Numbers on page 303*
- *Stringently Checking Resources on page 305*
- *Understanding Run Dates on page 307*

Understanding Libraries

A Documaker library consists of an index and its corresponding data. Generally, the index is contained in one table and the data portion in another table. The index table contains one row for each version and revision of a resource with a specific name and of a specific type.

The index table includes these fields. The maximum length is indicated in parentheses:

Field	Description
FileType (3)	This field can contains a value that defines the type of resource this row refers to, such as FAP, LOG, BDF, GRP, or DAL.
FileSTyp (3)	This field indicates the sub type of the resource this row refers to. This field is not currently used and is usually set to the value of the FileType field.
FileName (100)	This field indicates the name of the resource, such as Q1SNAM, Q1ADDR, or IMAGE1. The name can consist of up to 100 characters.
Resource (25)	Reserved for future use.
Descript (100)	This field contains a description of the resource or the last change made to it. You can enter up to 100 characters.
Effectiv (10)	This field contains the date on which you want this resource to become available for processing. This date is stored in D4 format (YYYYMMDD) in the library index. Depending on your locale setting (Choose Settings, then go to the Locale option in the Language control group), you may see the date displayed in other formats like MMDDYYYY or DDMMYYYY.
ModifyTm (10)	This field indicates the date and time the resource was last checked in or modified. This date appears in the appropriate format for your locale setting, along with the time.
FileIndx (8)	This field contains a number that, for xBase implementations, connects this index record to a compressed file in the data portion of the library.
RecStat (3)	This field indicates whether this resource is locked or not. If locked, the field contains LOC, otherwise it is blank.
Version (5)	This field contains the version number of this resource. The version number starts at 00001 and, as the resource is checked out and checked in, is incremented if you have checked the Increment Version field.

Field	Description
Revision (5)	This field contains the revision number of this resource. The revision number starts at 00001 and, as the resource is checked out and checked in, is incremented. If you have checked the Increment Version field, the version number is incremented instead of the revision number and the revision number is reset to 00001. When a resource is expired, the revision field is set to <i>EXP</i> .
UserID (64)	This field shows you the user ID of the user who last modified the resource, or who has the resource checked out. This ID can consist of up to 64 characters.
UsrLevl (2)	Reserved for future use.
Passwd (64)	Reserved for future use.
Unique_ID (26)	This field contains a unique identifier used for internal purposes.
ArcKey (18)	This key is used with DBMS implementations of the library to connect this index record to a row in the data table of the library. When using a DBMS, this field is used instead of the FILEINDX field, which is used in xBase implementations of the library.
LibName (129)	This is the name of the library. This name can consist of up to 129 characters.
AppData (10)	This field can contains up to 10 characters of additional, customer-specific application data.
Mode (25)	This field indicates the mode of the resource. You define the modes using the Manage, Settings options. For instance, you could set up modes to denote milestones in the development process such as Development, Testing, and Production. You can enter up to 25 characters. For more information, see <i>Defining Mode, Status, Class, and Project Options on page 380</i> .
Status (25)	This field indicates the status of the resource. You define the status codes using the Manage, Settings options. For instance, you could have codes like Pass or Fail. You can enter up to 25 characters. For more information, see <i>Defining Mode, Status, Class, and Project Options on page 380</i> .
Class (25)	This field indicates the class of the resource. You define classes using the Manage, Settings options. You can use classes to group resources by product lines or by geographical regions, such as GA, TX, or MD. You can enter up to 25 characters. For more information, see <i>Defining Mode, Status, Class, and Project Options on page 380</i> .
Project (25)	This field indicates the project code that last modified this resource. You define project codes using the Manage, Settings options. You can enter up to 25 characters. For more information, see <i>Defining Mode, Status, Class, and Project Options on page 380</i> .

While the index portion of the library contains attributes for each version/revision combination of a resource that is stored in the library, the data portion of the library contains the resources themselves.

The structure of the data portion of the library depends on how the library is defined (xBase, DBMS, Documanage), but in general, the data portion contains each of the version/revision combinations of the resources listed in the library index.

Processing with Effective Dates

Because document requirements change over time, your forms will evolve. Studio lets you assign effective dates to resources so you can make changes ahead of time while making sure those changes are put in effect only when appropriate. You can change the content of a form as necessary and control the date at which the new form takes effect. Using effective dates also lets you preserve prior versions of a form so when you need to retrieve a version of a form that was effective, for example, a year ago, you can.

In the Documaker system, a form may be composed of one or more sections. A section is also called a FAP file because *fap* is the extension the Studio applies. So, if a form needs to be changed, then one or more of the sections (FAP files) that comprise the form need to be changed.

For example, you have a form called *DEC PAGE*. This form consists of these sections...

- Q1SNAM
- Q1MDC1
- Q1MDC2
- Q1MDC3

The Q1SNAM section contains the company's corporate address. The company's corporate address is going to change next month. and you want to go ahead and change the address to reflect the change.

You can check out the Q1SNAM section (FAP), make the address change, then check in the Q1SNAM section and set the effective date for this new version of Q1SNAM to the first of next month.

If the DEC PAGE form is assembled at any time this month, the version of Q1SNAM that is effective this month (the older one) is used. If the DEC PAGE form is assembled next month, the version of Q1SNAM that is effective next month (the newer one) is used.

This example uses a FAP file to demonstrate how the effective date of a resource can be used to cause a different version of that resource to be used at the appropriate time. You can also manage graphics (LOG files) and DAL scripts in the same way.

Note Keep in mind if you are triggering forms based on effective date processing:

If possible, avoid creating forms with effective dates that require updated printer resource files, such as fonts, form overlays, bitmap files, and so on.

Otherwise, make sure you have the corresponding set of printer resource files for the forms based on the effective date to be processed.

Retrieving Resources with Version and Revision Numbers

When a resource is added to the library it is assigned a *version* number and a *revision* number. A resource added to the library for the first time is given a version number of *00001* and a revision number of *00001*. The version number and revision number change as the resource is checked out, changed, and checked back in.

Generally, if you check out a resource to make a small change, the revision number of the resource is incremented by one when you check in the resource. If you make a major change to the resource, or if this change requires you also change the effective date, the version number is incremented by one. The version number is considered a *major* number and the revision number is considered a *minor* number.

Typically, version/revision numbers are shown in this format:

version.revision

and leading zeros are omitted. So version *00001*, revision *00003* is referred to as *1.3*.

Note Documaker (versions 10.2 and higher) enforces the rule that all revisions of a particular version of a resource must have the same effective date. Prior to version 10.2, Documaker allowed revisions of a particular version of a resource to have different effective dates.

When you request a resource, the request generally supplies a resource type (FAP, LOG, or DAL), a name (such as Q1SNAM) and a run date (such as 20071225, which is December 25, 2007). Studio searches the library index for items that match that name and type, then chooses the one which has an effective date closest to, but not exceeding, the run date.

If Studio finds there are two or more versions of the same resource with the same effective date, it chooses the one with the higher version/revision number. For instance, the version/revision 2.1 is considered to be higher than the version/revision 1.2, so between these two resources, Studio would choose 2.1.

Stringently Checking Resources

In Documaker 10.3

In Documaker 10.3 and higher, the StringentChecking option defaults to Yes. Stringent checking means that if Studio finds versions of a resource in the library but none are effective yet — their effective date is greater than the run date being used— it will not return any of those resources.

Although you should leave the StringentChecking option set to Yes, you can, however, override it, as shown here:

```
< LibraryManager >
  StringentChecking = No
```

When StringentChecking is set to No, Studio tries to return some version of the resource, even if that resource is not yet effective. If Studio finds versions of a resource in the library but none are effective yet, Studio returns the resource with the oldest effective date.

If the StringentChecking option is set to No and Studio does not find any versions of the resource in the library, it tries to find the resource on disk, in the location designated for that resource type in the INI file. If it finds the resource on disk, it returns that resource.

In Documaker 10.2 and earlier

In Documaker 10.2 and earlier, the StringentChecking option defaults to No. If the system finds versions of a resource in the library but none of those versions are effective yet, it returns the oldest version/revision of that resource.

If the system does not find any versions of the resource in the library, it tries to find the resource on disk, in the location designated for that resource type in the INI file. If it finds the resource on disk, it returns that resource.

For DDT files only, if you want the system to issue an error message or to perform specialized processing, you can set the StringentChecking option to Yes. In Documaker 10.2, stringent checking is only available for DDT resources and only affects the GenData program.

You can use these INI options to control the stringent checking of DDT resources:

```
< DDTResource >
  ErrorOnMissingFile      = Yes
  RemoveImageMissingDDT  = No
  StringentChecking       = No
  WarnOnMissingFile      = Yes
```

Option	Description
ErrorOnMissingFile	The system normally generates an error if a DDT file cannot be found for a triggered section. Errors are written to a file, named using the ErrFile option in the Data control group. This file is typically called ERRFILE.DAT. Set this option to No to prevent the system from generating an error message when it cannot find the DDT file. The default is Yes.

Option	Description
RemoveImageMissingDDT	<p>A DDT file is normally required for any triggered section and not finding a section is a fatal error for a transaction. Setting this option to Yes tells the system that when it cannot find a DDT file for a triggered section, it should remove the section from the form set.</p> <p>This can result in the removal of the form as well if all sections are removed. An empty form set can result if all of the forms are removed. If this results in an empty form set, the system generates an error message. The default is No.</p>
StringentChecking	<p>Normally, if the system does not find an effective version of the file, it returns the oldest version/revision of the file.</p> <p>If, however, you set this option to Yes, only an effective version of the file will be returned. If no such file exists, nothing is returned. Stringent checking is only applicable when you use the system to control effective date eligibility of DDT files. The default is No.</p>
WarnOnMissingFile	<p>Normally, the system generates an error if it cannot find a DDT file for a triggered section. You can, however, suppress this error by setting the ErrorOnMissingFile option to No.</p> <p>When you suppress this type of error, the system assumes you still want a warning message but by setting this option to No, even the warning message is suppressed. The default is Yes.</p>

Understanding Run Dates

The idea behind effective date processing is that, within a library, there can be multiple versions of a resource, each with a different effective date. When the system creates a form set, it is built with the version (and revision) of the resources that were, or will be, effective on a given date. In Documaker, this date is called the *run date*. In Documaker Workstation it is called the *create date*.

When using Studio to manage your resources, you must use run dates for the system to construct, archive, and retrieve your form sets correctly. Be sure to:

- Define the RunDate field in your TRNDFDFL.DFD, RCBDFDFL.DFD, and APPIDX.DFD Data Format Definition (DFD) files.
- Create the extract file to include the run date for each transaction's set of records.
- Use the TRN_FIELDS control group to indicate the location of the RunDate and other fields in the extract file. The GenTrn program uses this control group.
- Set the RunDate option in the Trigger2Archive control group as shown here to make the GenArc program populate the APPIDX file with the RunDate from the NEWTRN file:

```
< Trigger2Archive >
  RunDate = RunDate
```

- Set the RunDate option in the AFEWIP2ArchiveRecord control group to make Documaker Workstation (PPS) populate the APPIDX file with the value of the creation date in the WIP index:

```
< AFEWIP2ArchiveRecord >
  RunDate = CreateTime,X
```

If you follow the guidelines above, the system will behave as described here:

- When the GenTrn program runs, the value in the RunDate field for each transaction is read from the extract file and copied into the TRNFILE.
- When the GenData program runs, it uses the value in the RunDate field in the TRNFILE for each transaction and loads the appropriate resources from the library.
- When the GenPrint program runs, it uses the value in the RunDate field, if necessary, in the recipient batch files for each transaction.
- When the GenArc program runs, it copies the value in the RunDate field from the NEWTRN file into the APPIDX file, using the RunDate option in the Trigger2Archive control group.
- When you archive a form set in Documaker Workstation, the value in the CreateTime field is converted from hexadecimal format to D4 format (YYYYMMDD) and copied into the APPIDX file.

When you retrieve that form set, the value in the CreateTime field that was saved into the APPIDX file can be used to load objects, such as FAP files and graphics, from the library as required to reconstruct the form set.

Keep in mind the run date should be in D4 format, which is YYYYMMDD. If your run date is not in D4 format, you can convert it using additional INI options.

To make sure the value in the RunDate field is part of the TRNDFDFL.DFD, RCBDFDFL.DFD, and APPIDX.DFD files, make sure these DFD files contain entries for the RunDate field, as shown below. The DFD files shipped with your system contain similar entries. DFD files are generally stored in the \DEFLIB directory. In each of these DFD files, you should have an entry for the RunDate field and an entry that describes the attributes of the RunDate field. These entries will look similar to those shown here:

```
< Fields >
  FieldName      = RunDate
< FIELD:RunDate >
  EXT_Type       = CHAR_ARRAY
  EXT_Length     = 8
  EXT_Precision  = 0
  INT_Type       = CHAR_ARRAY
  INT_Length     = 8
  INT_Precision  = 0
  Key            = No
  Required       = Yes
```

If you did not place the RunDate field in the TRNDFDFL.DFD, RCBDFDFL.DFD, and APPIDX.DFD files, you can use the GetRunDate rule to get the current system date and use it as the RunDate for each transaction. Keep in mind that using the GetRunDate rule limits you to using the current date as the run date, which may not always be the date you want to use.

HOW IT ALL WORKS

The following topics describe how Documaker software uses these concepts in everyday processing.

In Documaker Workstation

Entry

When you use Documaker Workstation (or PPS) to create a new form set or transaction, the current system date is used to note when the form set was created. This value is stored in the CreateTime field, which is equivalent to the RunDate field in Documaker.

You can change this date by clicking on the Effective Date control and choosing a different date from the calendar. You can hide the Effective Date control using this option:

```
< Control >
  ShowEffectiveDate = No
```

The effective date specified — whether the current date or another date — is the date compared against the effective dates associated with the resources, such as FAP files and graphics, pulled from the library.

Example 1 - Multiple versions, different effective dates

Say today's date is 10/25/2007 and the form set you are creating requires a form called *DEC PAGE*. The DEC PAGE form is comprised of these FAP files:

- Q1SNAM
- Q1MDC1
- Q1MDC2
- Q1MDC3

Assume the following versions of the Q1SNAM FAP are in the library:

Form set	Version	Revision	Effective Date
Q1SNAM	00001	00001	20070131
Q1SNAM	00001	00002	20070731
Q1SNAM	00002	00001	20071231

Since the run date is 20071025 (10/25/2007), Studio chooses version 00001, revision 00002 (version 1.2) of Q1SNAM because it has an effective date (20070731) and that is the latest date that does not exceed the run date (20071025).

Note This example shows two revisions, 1.1 and 1.2, of version 1 which contain different effective dates. Though Documaker 10.2 and higher enforce the rule that all revisions of a particular version of a resource must have the same effective date, if you have migrated a library from a prior version, your library may contain resources that do not adhere to this restriction. This situation will not cause errors.

Example 2 - Multiple versions, same effective dates

Say today's date is 10/25/2007 and the form set you are creating requires a form called *DEC PAGE*. The DEC PAGE form is comprised of these FAP files:

- Q1SNAM
- Q1MDC1
- Q1MDC2
- Q1MDC3

Assume the following versions of the Q1SNAM FAP are in the library:

Form set	Version	Revision	Effective Date
Q1SNAM	00001	00001	20070731
Q1SNAM	00001	00002	20070731
Q1SNAM	00001	00003	20070731
Q1SNAM	00002	00001	20071231

The run date is 20071025 (10/25/2007), so the latest effective for Q1SNAM that does not exceed the run date is 20070731. There are three version/revisions of Q1SNAM that have the most correct effective date, versions 1.1, 1.2, and 1.3. Studio chooses version 1.3 because it has the highest version/revision number.

Sometimes, you might create a form set and decide to base that new form set on an older one from archive. To do this, you first choose the New, Retrieve Data option to select the form set from archive. By default, the new form set would consist of the same version/revisions of the forms from the archived form set.

If, instead, you want the new form set to be comprised of the same forms as the archived form set, but with newer versions/revisions of those forms — if they exist — you must use this INI option:

```
< FormSelection >
  RetrieveVersionInfo = No
```

WIP

When you save a form set to WIP (by choosing File, Save or WIP, Save), the system writes a record to the WIP index table and two WIP data files are written to the \WIP directory. By default, the WIP index table is named *WIP* and is composed of these files:

- WIP.DBF
- WIP.MDX

The two WIP data files written to the \WIP directory are essentially a NAFILE and a POLFILE. These two files may be named something like:

```
D4234FF15243414FB1B504379EC76D0D.dat    (this is the NAFILE)
      D4234FF15243414FB1B504379EC76D0D.pol    (this is the POLFILE)
```

Prior to version 10.2 of Documaker Workstation, the WIP data files had names such as:

```
00000001.dat    (this is the NAFILE)
00000001.pol    (this is the POLFILE)
```

If you migrated to version 10.2 of Documaker Workstation from a prior version and have a WIP index (WIP.DBF and WIP.MDX), your WIP data files have the shorter names. Depending on the configuration, the resources in the NAFILE.DAT file may have version, revision, and effective date information in the \NA=... record. Here is an example:

```
\NA=q1snam, LN=1, DUP=OFF, SIZE=C, TRAY=U, X=0, Y=0, PA=1, OPT=D\
\ENDIMAGE\
\NA=q1mdc1, LN=1, DUP=OFF, SIZE=C, TRAY=U, X=0, Y=3360, PA=1, OPT=DS, V=1, R=1
, D=20020911\
ENDIMAGE\
```

Notice the \NA=q1mdc1... record contains these options and values:

- V=1
- R=1
- D=20020911

These values represent the specific version, revision, and effective date of Q1MDC1, pulled from the library when this form set was created. Notice that the \NA=q1snam... record does not contain these values.

After saving this form set to WIP, you might later return to retrieve this form set. When reconstructing the form set, for those sections listed in the NAFILE.DAT file that contain the V, R, and D options and values, the system retrieves the specific version and revision of the section listed.

For those sections listed in the NAFILE.DAT file that do not contain the V, R, and D options, Documaker Workstation gets a run date and uses that date to retrieve the correct version/revision of the sections. Documaker Workstation gets its run date by looking for a value in these locations:

- The RunDate field in the WIP index
- The CreateTime field in the WIP index
- The current system date

If the RunDate field does not exist or is empty, Documaker Workstation looks in the CreateTime field. If the CreateTime field does not exist or is empty, it uses the current system date.

Archive and retrieval

When a form set is archived, its NAFILE and POLFILE contents are stored in an archive file or table (called a CARFile — Compressed Archive File) and some (or all) of the information that was in the WIP index (if it was archived from WIP) or in the NEWTRN file (if it was archived using the GenArc program) is stored into the application index (APPIDX) table. One of the fields in the APPIDX table is called *RunDate*.

You can retrieve a form set from archive in Documaker Workstation by choosing the Retrieve, Formset option. When reconstructing the form set, for those sections listed in the NAFILE file that contain the V, R, and D options and values, the system retrieves the specific version and revision of the section that is listed.

For those sections listed in the NAFILE that do not contain the V, R, and D options, Documaker Workstation gets a run date and uses that date to retrieve the correct version/revision of the sections. Documaker Workstation gets its run date by looking for a value in these locations:

- The RunDate field in the application index (APPIDX)
- The CreateTime field in the application index
- The current system date

If the RunDate field does not exist or is empty, Documaker Workstation looks in the CreateTime field. If the CreateTime field does not exist or is empty, it uses the current system date.

In Documaker Server

The GenData and GenPrint programs in Documaker Server will generally need to load resources (such as FAP files, LOG files, and DAL scripts). To retrieve the resource with the appropriate effective date, the system needs to know the run date for each form set to be generated.

The GenTrn program gets the run date from the extract file, for each transaction and writes the run date to the TRNFILE.

The GenData program gets the run date from a Global Variable Manager (GVM) variable named *RunDate*. The RunDate GVM is set by:

- a value in the TRNFILE file or
- the GetRunDate rule, which copies the current system date into the RunDate GVM

As the GenData program processes transactions, the run date is written to the NEWTRN file and to the recipient batch files, provided the TRNDFDFL.DFD and RCBDFFDFL.DFD files contain the RunDate field.

The GenPrint program reads the NAFILE, POLFILE, and recipient batch files. When it needs to load a resource referenced in the NAFILE, such as a FAP or graphic file, it asks for the specific version and revision of the resource if those values are listed in the NAFILE.

For example, in this NAFILE excerpt:

```
\NA=q1snam, LN=1, DUP=OFF, SIZE=C, TRAY=U, X=0, Y=0, PA=1, OPT=D\
\ENDIMAGE\
\NA=q1mdc1, LN=1, DUP=OFF, SIZE=C, TRAY=U, X=0, Y=3360, PA=1, OPT=DS, V=1, R=1
, D=20070911\
\ENDIMAGE\
```

The Q1MDC1 FAP file contains these options:

- V=1
- R=1
- D=20070911

These options and values represent version 1, revision 1 (1.1) and an effective date of 9/11/2007. If the GenPrint program needs to load Q1MDC1, it asks for version 1.1 of this file.

On the other hand, for Q1SNAM, the V, R, and D options and values do not exist in the NAFILE, so if the GenPrint program needs to load this file, the system uses the run date specified in the recipient batch file to determine the correct version and revision of the FAP file to retrieve.

In Docupresentation (IDS)

When you use the Documaker Bridge to retrieve archived form sets, you can also use library management to retrieve the correct version and revision of any objects, such as FAP and graphic files, referenced in the form set.

To configure Docupresentation to use library management, follow these steps:

1. For each request type in the DOCSERV.INI file you want to use, specify the DPRInitLby rule in the rules list. Here is an example of how you can use the DPRInitLby rule and its location in the rule list for the PRT request type:

```
[ ReqType:PRT ]
function = atcw32->ATCLogTransaction
function = atcw32->ATCLoadAttachment
function = dprw32->DPRSetConfig
function = dprw32->DPRInitLby
function = atcw32->ATCUnloadAttachment
function = dprw32->DPRRetrieveFormset
function = dprw32->DPRPrint
function = dprw32->DPRProcessTemplates
```

2. For each configuration you want to use, specify the library name in that configuration's INI file. For example, if you are using the RPEX1 configuration, you would specify options similar to these in the RPEX1.INI file:

```
< MasterResource >
  FormFile = master.lby
  LogoFile = master.lby
  DALFile = master.lby
  LbyLib = e:\fap\mstrres\rpex1\deflib\
```

The FormFile, LogoFile, and DALFile options name the library and the LbyLib option names the location of the library.

3. Set the ARCEFFECTIVEDATE attachment variable before calling any rules that use the library, such as DPRRetrieveFormset. Refer to the [SDK Reference](#) to determine which rules use the ARCEFFECTIVEDATE attachment variable.

The ARCEFFECTIVEDATE attachment variable is used by the Documaker Bridge to locate the appropriate version/revision of a resource if a specific version/revision is not indicated in the retrieved NAFILE. If you are using the Docupresentment CGI client, you can set the ARCEFFECTIVEDATE by specifying the following in the appropriate HTML templates:

```
<FORM METHOD="POST" ACTION="#EXENAME, #">
<INPUT NAME="USERID" VALUE="#USERID,%s#" TYPE="HIDDEN">
<INPUT NAME="DOCTYPE" VALUE="#DOCTYPE,%s#" TYPE="HIDDEN">
<INPUT NAME="REQTYPE" VALUE="PRT" TYPE="HIDDEN">
<INPUT NAME="CONFIG" VALUE="#CONFIG,%s#" TYPE="HIDDEN">
<INPUT NAME="ARCEFFECTIVEDATE" VALUE="#RUNDATE,%s#" TYPE="HIDDEN">
<-- sets the ARCEFFECTIVEDATE attachment variable
```

This copies the value of the RunDate attachment variable into the ARCEFFECTIVEDATE attachment variable. The RunDate attachment variable comes from the RunDate field of the archive application index (APPIDX) file. See *Understanding Run Dates on page 307* to see how to make sure the APPIDX RunDate field is populated correctly.

MANAGING WORKFLOW

You can also manage the development, testing, and promotion of resources. Using one or more libraries, you can coordinate the creation and modification of section (FAP), graphics (LOG), and DAL script files. Locking of resources as they are checked out, prevents multiple users from trying to change a particular resource at the same time. Prior versions of a resource are kept safe and can be restored if necessary.

As you test resources checked into a library, you can mark the resources as having passed or failed using the Status field on the File Information window. You can also promote them to another library for additional testing or for production use.

CREATING LIBRARIES

The way you create a resource library differs depending on how the library is stored. You have these choices:

- *Using xBase and CARFiles on page 317*
- *Using the DB2 Native Driver on page 319*
- *Using the DB2 ODBC Driver on page 324*
- *Using the SQL Server ODBC Driver on page 329*
- *Using the Oracle ODBC Driver on page 332*
- *Using Documanager on page 337*

Specifying a catalog table for each library

You can specify a catalog table for each library stored in a database management system (DBMS). If you do this, Studio will only update the catalog table once per library per session, which reduces the chances of you receiving the duplicate index or -803 SQL error message.

You specify the catalog table using INI options. Here is an example:

```
< Library:LBYI >
  Catalog    = LBYC
< DBTable:LBYC >
  DBHandler  = DB2
< DB2_FileConvert >
  Catalog    = DAP112_LBYC_R1
```

USING XBASE AND CARFILES

You can use the xBase or CARFile format for the library on Windows, AIX, Solaris, and Linux operating systems. By default, a Documaker library is stored in this file format:

- The index portion is stored as a xBase file (actually two files)
- The data portion is stored in a compressed file format referred to as a *CARFile*

The default library name is *MASTER.LBY*. Using this name, the system creates these files:

File	Description
MASTER.DBF	The DBF and MDX files make up the index portion of the library. The DBF component contains the index data and the MDX component contains tag information.
MASTER.MDX	The DBF and MDX files make up the index portion of the library. The DBF component of the index contains the index data and the MDX component contains tag information.
MASTER.LBY	The LBY file makes up the data portion of the library. This file contains the actual resources the index refers to. This file is in a compressed archive file (CARFile) format.

Creating the CARFile and Index Files

On Windows, AIX, Solaris, and Linux

If you are using the xBase/CARFile format for the library index and data, Studio can create the library index and data files.

Sample INI options

Use INI options like the following to create a library in a CARFile format (index in xBase format) and to load resources from that library:

```
< MasterResource >
  BDFFile   = master.lby
  DALFile   = master.lby
  DDTFile   = master.lby
  FORFile   = master.lby
  FormFile  = master.lby
  GRPFile   = master.lby
  LbyLib    = ..\mstrres\deflib\
  LogoFile  = master.lby
< LibraryManager >
  LbyLogFile= lbylog
```

Option	Description
MasterResource control group	
BDFFile	This option tells the system you want to retrieve business definition (BDF) resources from a library named MASTER.LBY.
DALFile	This option tells the system you want to retrieve DAL scripts and DAL script libraries from a library named MASTER.LBY.

Option	Description
DDTFile	This option tells the system you want to retrieve DDT files from a library named <i>MASTER.LBY</i> .
FORFile	This option tells the system you want to retrieve form (FOR) resources from a library named <i>MASTER.LBY</i> .
FormFile	This option tells the system you want to retrieve FAP files from a library named <i>MASTER.LBY</i> .
GRPFile	This option tells the system you want to retrieve group form (GRP) resources from a library named <i>MASTER.LBY</i> .
LbyLib	This option tells the system that the <i>MASTER.LBY</i> file and the two files that make up its index (<i>MASTER.DBF</i> and <i>MASTER.MDX</i>), reside in the location specified by the relative path <i>..lmstrres\deflib</i> .
LogoFile	This option tells the system you want to retrieve graphics (LOG) files from a library named <i>MASTER.LBY</i> .
LibraryManager control group	
LbyLogFile	This option tells the system the name of the library log file is <i>LBYLEG</i> . The library log contains information about resources added to, deleted from, or updated in the library. The library log file does not have to use the same type of database handler as the library index and data portions.

Specifying a DFD file for the library data table

A Data Format Definition (DFD) file is used to describe the layout of the fields in another file. The description includes the name, data type, and length of each field in the file.

Several files used by Documaker adhere to pre-defined internal structure definitions so there is usually no need to create and specify DFD files for those files. The library's data and index portions are two examples of files that usually adhere to a pre-defined internal structure.

There are cases, however, when you must specify the structure of the data portion of the library using a DFD file. Prior to version 11.2, you made this DFD specification via the *CARFileDFD* option. For example, if your library data table was called *LYD* and the DFD was named *CARDB2.DFD*, you would have specified the DFD as shown here:

```
< ArcRet >
  CARFileDFD= DEFLIB\cardb2.dfd
```

This *CarFileDFD* option, however, is also used by the GenArc program and if a different value was needed for archival purposes, you had to maintain a different INI file for that process. In version 11.2 and in subsequent versions, instead of using the *CARFileDFD* option, you can specify the library data file's DFD using the *DFD* option, as shown here:

```
< DBTable:LYD >
  DBHandler = DB2
  DFD       = DEFLIB\LYD_DB2.dfd
```

This *DFD* option does not conflict with the *CARFileDFD* option used by the GenArc program.

USING THE DB2 NATIVE DRIVER

You can use the DB2 native driver — using DB2 but not going through ODBC — with the Windows, AIX, Solaris, Linux, and OS390 operating systems.

Creating the Database and Tables

On Windows, AIX, Solaris, and Linux

On Windows, AIX, Solaris, and Linux you can have the LBRYMGR utility create the library index, data and catalog tables or you can create them beforehand. To tell the LBRYMGR utility to create the library and catalog tables, specify this INI option:

```
< DBHandler:DB2 >
  CreateTable = Yes
```

Note For more information about the LBRYMGR utility, see the [Docutoolbox Reference](#).

To create the DB2 library and catalog tables manually, execute an SQL script like the one provided with the RPEX1 sample resources in this directory:

```
..rpex1\deflib\db2\lbysqlr.cmd
```

This file is also listed below:

```
CONNECT TO LBYLIB;
```

```
-----
-- DDL Statements for table "DAP110_CAT_R1"
-----
```

```
CREATE TABLE "DAP110_CAT_R1" (
    "CATALOGID" CHAR(10) ,
    "CARFILE" CHAR(8) ,
    "MEDIAID" CHAR(11) ,
    "STATUS" CHAR(1) )
IN "USERSPACE1" ;
```

```
-----
-- DDL Statements for table "DAP110_LBYI_R1"
-----
```

```
CREATE TABLE "DAP110_LBYI_R1" (
    "FILETYPE" CHAR(3) ,
    "FILESTYP" CHAR(3) ,
    "FILENAME" CHAR(100) ,
    "RESOURCE" CHAR(25) ,
    "DESCRIPT" CHAR(100) ,
    "EFFECTIV" CHAR(10) ,
    "MODIFYTM" CHAR(10) ,
    "FILEINDX" CHAR(8) ,
    "RECSTAT" CHAR(3) ,
    "VERSION" CHAR(5) ,
    "REVISION" CHAR(5) ,
    "USERID" CHAR(64) ,
    "USRLEVL" CHAR(2) ,
    "PASSWD" CHAR(64) ,
    "UNIQUE_ID" CHAR(26) ,
    "ARCKEY" CHAR(18) ,
    "MODE" CHAR(25) ,
    "STATUS" CHAR(25) ,
```

```
"CLASS" CHAR(25) ,  
"PROJECT" CHAR(25) )  
IN "USERSPACE1" ;
```

```
-----  
-- DDL Statements for table "DAP110_LBYD_R1"  
-----
```

```
CREATE TABLE "DAP110_LBYD_R1" (  
    "ARCKEY" CHAR(18) ,  
    "SEQ_NUM" CHAR(5) ,  
    "CONT_FLAG" CHAR(1) ,  
    "TOTAL_SIZE" INTEGER ,  
    "CARDATA" LONG VARCHAR FOR BIT DATA )  
IN "USERSPACE1" ;
```

```
COMMIT WORK;
```

```
CONNECT RESET;
```

```
TERMINATE;
```

If you manually create the DB2 tables by running using this script, set the CreateTable option to No when you later run Documaker. Here is an example:

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

On z/OS

On z/OS, run the job located in FSI.V110.JCLLIB(LBYSQLR) to create the library index, data, and catalog tables. This job also creates the library log (LBYLOG) table, which contains entries of items that are added to, deleted from, or updated in the library.

Sample INI Options

Here is an example of how you can set up your INI options to use the DB2 native driver to load resources from a library defined in DB2:

```
< MasterResource >
  BDFFile   = LBYI
  DALFile   = LBYI
  DDTFile   = LBYI
  FORFile   = LBYI
  FormFile  = LBYI
  GRPFile   = LBYI
  LogoFile  = LBYI
< LibraryManager >
  LBYLogFile= LBYLOG
< Library:LBYI >
  DBTable   = LBYD
< DBTable:LBYI >
  DBHandler = DB2
< DBTable:LBYD >
  DBHandler = DB2
  UniqueTag = ARCKEY+SEQ_NUM
< DBTable:LBYLOG >
  DBHandler = DB2
< DBTable:CATALOG >
  DBHandler = DB2
  UniqueTag = CATALOGID
< DBHandler:DB2 >
  Class     = DB2
  CreateIndex= No
  CreateTable= Yes
  Database  = LBYDBASE
  Debug     = No
  Passwd    = password
  UserID    = userID
< DB2_FileConvert >
  LBYI      = DAP110_LBYI_R1
  LBYD      = DAP110_LBYD_R1
  LBYLog    = DAP110_LBYLOG_R1
```

Option	Description
--------	-------------

MasterResource control group

BDFFile	This option tells the system you want to retrieve business definition (BDF) resources from a library named <i>LBYI</i> .
DALFile	This option tells the system you want to retrieve DAL scripts and DAL script libraries from a library named <i>LBYI</i> .
DDTFile	This option tells the system you want to retrieve DDT files from a library named <i>LBYI</i> .
FORFile	This option tells the system you want to retrieve form (FOR) resources from a library named <i>LBYI</i> .
FormFile	This option tells the system you want to retrieve FAP files from a library named <i>LBYI</i> .
GRPFile	This option tells the system you want to retrieve group form (GRP) resources from a library named <i>LBYI</i> .
LogoFile	This option tells the system you want to retrieve graphics (LOG) files from a library named <i>LBYI</i> .

Option	Description
--------	-------------

LibraryManager control group

LbyLogFile	This option tells the system the name of the library log file is <i>LBYLOG</i> . The library log contains information about resources added to, deleted from, or updated in the library. The LbyLogFile does not have to use the same type of database handler as the library index and data portions.
------------	--

Library:LBYI control group

DBTable	This option tells the system the data component of the library named <i>LBYI</i> is called <i>LBYD</i> . In this example, the names LBYI and LBYD have been chosen to emphasize that one table, LBYI, represents the library index and one table, LBYD represents the library data. You can call these tables any name you like but the name cannot exceed eight characters. See the DB2_FileConvert control group if you need to map these eight-character names to longer table names.
---------	---

DBTable:LBYI control group

DBHandler	This option tells the system to access the table known as <i>LBYI</i> using the database handler named <i>DB2</i> . Based on this option, the system expects to find a control group named <i>DBHandler:DB2</i> .
-----------	---

DBTable:LBYD control group

DBHandler	This option tells the system to access the table known as <i>LBYD</i> using the database handler named <i>DB2</i> . Based on this option, the system expects to find a control group named <i>DBHandler:DB2</i> .
-----------	---

UniqueTag	This option tells the system the columns ARCKEY and SEQ_NUM can be combined to represent a unique tag for the table. This unique tag is only used for internal purposes. If you do not specify a unique tag for this table, and a column with the name UNIQUE_ID does not exist within the table, you receive warning messages indicating no unique tag is defined. Include this value to avoid those warning messages.
-----------	--

DBTable:LBYLOG control group

DBHandler	This option tells the system to access the table known as <i>LBYLOG</i> using the database handler named <i>DB2</i> . Based on this option, the system expects to find a control group named <i>DBHandler:DB2</i> .
-----------	---

DBTable:CATALOG control group

DBHandler	This option tells the system to access the table known as <i>CATALOG</i> using the database handler named <i>DB2</i> . The CATALOG table temporarily stores the CATALOGID values used to construct an ARCKEY.
UniqueTag	This option tells the system the column CATALOGID represents a unique tag for this table. This unique tag is only used for internal purposes. If you do not specify a unique tag for this table, and a column with the name UNIQUE_ID does not exist within the table, you receive warning messages indicating there is no unique tag defined. Include this value to avoid those warning messages.

DBHandler:DB2 control group

Class	This option tells the system this database handler represents a handler for IBM's DB2 database management system. The Class option is unnecessary if the name of the handler (DB2 in this case) matches one of the Documaker pre-defined values, such as DB2, ODBC, ORA, or DMS.
-------	--

Option	Description
CreateIndex	This option tells the system not to create database indexes. This option should <i>always</i> be set to No.
CreateTable	This option tells the system to create any missing required tables at run time.
Database	This option tells the system the name of the database for this database handler is <i>LBYDBASE</i> .
Debug	This turns off tracing for the Documaker DB2 database handler. Normally you would omit the Debug option or set it to No. In troubleshooting situations, set this option to Yes and examine the trace messages written to the trace file.
Passwd	This option tells the system the password to use when connecting to the database management system.
UserID	This option tells the system the user ID to use when connecting to the database management system.

DB2_FileConvert control group

LBYI	This option tells the system the table referenced in INI options as LBYI is really named <i>DAP110_LBYI</i> on the database management system. This control group lets you map table names of eight characters or less to table names longer than eight characters. The table name you specify must adhere to the table naming conventions for the database management system.
LBYD	This option tells the system the table referenced in INI options as <i>LBYD</i> is really named <i>DAP110_LBYI</i> on the database management system.
LBYLog	This option tells the system the table referenced in INI options as <i>LBYI</i> is really named <i>DAP110_LBYI</i> on the database management system.

USING THE DB2 ODBC DRIVER

You can use the DB2 ODBC driver on the Windows operating system.

Creating the Database and Tables

You can let the LBRYMGR utility create the library index, data, and catalog tables or you can create them beforehand. To have the LBRYMGR utility create the library and catalog tables, specify this INI option:

```
< DBHandler:DB2 >
    CreateTable = Yes
```

Note For more information about the LBRYMGR utility, see the [Docutoolbox Reference](#).

To create the DB2 library and catalog tables manually, execute an SQL script like the one provided with the RPEX1 sample resources in this directory:

```
..rpex1\deflib\db2\lbysqlr.cmd
```

This file is also listed below:

```
CONNECT TO LBYLIB;
```

```
-----
-- DDL Statements for table "DAP110_CAT_R1"
-----
```

```
CREATE TABLE "DAP110_CAT_R1" (
    "CATALOGID" CHAR(10) ,
    "CARFILE" CHAR(8) ,
    "MEDIAID" CHAR(11) ,
    "STATUS" CHAR(1) )
IN "USERSPACE1" ;
```

```
-----
-- DDL Statements for table "DAP110_LBYI_R1"
-----
```

```
CREATE TABLE "DAP110_LBYI_R1" (
    "FILETYPE" CHAR(3) ,
    "FILESTYP" CHAR(3) ,
    "FILENAME" CHAR(100) ,
    "RESOURCE" CHAR(25) ,
    "DESCRIPT" CHAR(100) ,
    "EFFECTIV" CHAR(10) ,
    "MODIFYTM" CHAR(10) ,
    "FILEINDX" CHAR(8) ,
    "RECSTAT" CHAR(3) ,
    "VERSION" CHAR(5) ,
    "REVISION" CHAR(5) ,
    "USERID" CHAR(64) ,
    "USRLEVL" CHAR(2) ,
    "PASSWD" CHAR(64) ,
    "UNIQUE_ID" CHAR(26) ,
    "ARCKEY" CHAR(18) ,
    "MODE" CHAR(25) ,
    "STATUS" CHAR(25) ,
    "CLASS" CHAR(25) ,
    "PROJECT" CHAR(25) )
IN "USERSPACE1" ;
```

```
-----  
-- DDL Statements for table "DAP110_LBYD_R1"  
-----  
  
CREATE TABLE "DAP110_LBYD_R1" (  
    "ARCKEY" CHAR(18) ,  
    "SEQ_NUM" CHAR(5) ,  
    "CONT_FLAG" CHAR(1) ,  
    "TOTAL_SIZE" INTEGER ,  
    "CARDATA" LONG VARCHAR FOR BIT DATA )  
IN "USERSPACE1" ;  
  
COMMIT WORK;  
  
CONNECT RESET;  
  
TERMINATE;
```

If you manually create the DB2 tables by running using this script, set the CreateTable option to No when you later run Documaker. Here is an example:

```
< DBHandler:ODBC >  
    CreateTable = No
```

Sample INI Options

Use INI options like the following to create a library in DB2, using the DB2 ODBC driver, and to load resources from that library:

```
< MasterResource >
  BDFFile   = LBYI
  DALFile   = LBYI
  DDTFile   = LBYI
  GRPFile   = LBYI
  FORFile   = LBYI
  FormFile  = LBYI
  LogoFile  = LBYI
<LibraryManager>
  LBYLogFile= LBYLOG
< Library:LBYI >
  DBTable   = LBYD
< DBTable:LBYI >
  DBHandler = ODBC
< DBTable:LBYD >
  DBHandler = ODBC
  UniqueTag = ARCKEY+SEQ_NUM
< DBTable:LBYLOG >
  DBHandler = ODBC
< DBTable:CATALOG >
  DBHandler = ODBC
  UniqueTag = CATALOGID
< DBHandler:ODBC >
  Class      = ODBC
  CreateIndex= No
  CreateTable= Yes
  Debug      = No
  Passwd     = password
  Qualifier  = LBYDBASE
  Server     = LBYDB2
  UserID     = userID
< ODBC_FileConvert >
  LBYI       = DAP110_LBYI_R1
  LBYD       = DAP110_LBYD_R1
  LBYLog     = DAP110_LBYLOG_R1
```

Option	Description
--------	-------------

MasterResource control group

BDFFile	This option tells the system you want to retrieve business definition (BDF) resources from a library named <i>LBYI</i> .
DALFile	This option tells the system you want to retrieve DAL scripts and DAL script libraries from a library named <i>LBYI</i> .
DDTFile	This option tells the system you want to retrieve DDT files from a library named <i>LBYI</i> .
FORFile	This option tells the system you want to retrieve form (FOR) resources from a library named <i>LBYI</i> .
FormFile	This option tells the system you want to retrieve FAP files from a library named <i>LBYI</i> .
GRPFile	This option tells the system you want to retrieve group form (GRP) resources from a library named <i>LBYI</i> .
LogoFile	This option tells the system you want to retrieve graphics (LOG) files from a library named <i>LBYI</i> .

Option	Description
--------	-------------

LibraryManager control group

LbyLogFile	This option tells the system the name of the library log file is <i>LBYLEG</i> . The library log contains information about resources added to, deleted from, or updated in the library. The LbyLogFile does not have to use the same type of database handler as the library index and data portions.
------------	--

Library:LBYP control group

DBTable	This option tells the system the data component of the library named <i>LBYP</i> is called <i>LBYPD</i> . In this example, the names <i>LBYP</i> and <i>LBYPD</i> have been chosen to emphasize that one table, <i>LBYP</i> , represents the library index and one table, <i>LBYPD</i> represents the library data. You can call these tables any name you like but the name must be eight characters or less. See the <i>ODBC_FileConvert</i> control group to map these eight-character names to longer table names.
---------	---

DBTable:LBYP control group

DBHandler	This option tells the system to access the table known as <i>LBYP</i> using the database handler named <i>ODBC</i> . Based on this option, the system expects to find a control group named <i>DBHandler:ODBC</i> . Microsoft's SQL Server is an ODBC-compliant database.
-----------	--

DBTable:LBYPD control group

DBHandler	This option tells the system to access the table known as <i>LBYPD</i> using the database handler named <i>ODBC</i> . Based on this option, the system expects to find a control group named <i>DBHandler:ODBC</i> .
-----------	--

UniqueTag	This option tells the system the columns <i>ARCKEY</i> and <i>SEQ_NUM</i> can be combined to represent a unique tag for the table. This unique tag is only used for internal purposes. If you do not specify a unique tag for this table, and a column with the name <i>UNIQUE_ID</i> does not exist, you receive warning messages indicating no unique tag is defined. Include this option to avoid those warning messages.
-----------	---

DBTable:LBYLEG control group

DBHandler	This option tells the system to access the table known as <i>LBYLEG</i> using the database handler named <i>ODBC</i> . Based on this option, the system expects to find a control group named <i>DBHandler:ODBC</i>
-----------	---

DBTable:CATALOG control group

DBHandler	This option tells the system to access the table known as <i>CATALOG</i> using the database handler named <i>ODBC</i> . The <i>CATALOG</i> table temporarily stores the <i>CATALOGID</i> values used to construct an <i>ARCKEY</i> .
-----------	--

UniqueTag	This option tells the system the column <i>CATALOGID</i> represents a unique tag for this table. This unique tag is only used for internal purposes. If you do not specify a unique tag for this table, and a column with the name <i>UNIQUE_ID</i> does not exist within the table, you receive warning messages indicating no unique tag is defined. Include this option to avoid those warning messages.
-----------	--

DBHandler:ODBC control group

Class	This option tells the system this database handler represents a handler for Microsoft's Open Data Base Connectivity (ODBC) standard. The <i>Class</i> parameter is unnecessary if the name of the handler, <i>ODBC</i> in this case, matches one of the Documaker pre-defined values, such as <i>DB2</i> , <i>ODBC</i> , <i>ORA</i> , or <i>DMS</i> .
-------	---

Option	Description
CreateIndex	This option tells the system not to create database indexes. This option should <i>always</i> be set to No.
CreateTable	This option tells the system that if a table needed by the system does not exist, to create it.
Debug	This option turns off tracing for the Documaker ODBC database handler. Normally you should omit the Debug option or set it to No. In troubleshooting situations, set this option to Yes and examine the trace messages that are written to the trace file.
Passwd	This option tells the system the password to use when connecting to the database management system.
Qualifier	This option tells the system that the name of the database for this database handler is <i>LBYDBASE</i> . If you omit this option, the system uses the database set up as the default database for the <i>LBYDB2</i> ODBC data source.
Server	This option tells the system the name of the ODBC data source for this database handler is <i>LBYDB2</i> . You will need to have defined an ODBC data source by this name.
UserID	This option tells the system the user ID to use when connecting to the database management system.

ODBC_FileConvert

LBYI	This option tells the system the table referenced in INI options as <i>LBYI</i> is really named <i>DAP110_LBYI</i> on the database management system. This control group lets you map table names of eight characters or less to table names longer than eight characters. The table name you specify must adhere to the table naming conventions for the database management system.
LBYP	This option tells the system the table referenced in INI options as <i>LBYP</i> is really named <i>DAP110_LBYI</i> on the database management system.
LBYPLog	This option tells the system the table referenced in INI options as <i>LBYP</i> is really named <i>DAP110_LBYI</i> on the database management system.

USING THE SQL SERVER ODBC DRIVER

You can use the SQL Server ODBC driver on Windows.

Creating the Database and Tables

On Windows, you can use Studio to create the library index and data files. You can also use the LBRYMGR utility do it.

Note For more information about the LBRYMGR utility, see the [Docutoolbox Reference](#).

Sample INI Options

Use INI options like the following to create a library in SQL Server, using the SQL Server ODBC driver, and to load resources from that library:

```
< MasterResource >
  BDFFile    = LBYI
  DALFile    = LBYI
  DDTFile    = LBYI
  FORFile    = LBYI
  FormFile   = LBYI
  GRPFile    = LBYI
  LogoFile   = LBYI
< LibraryManager >
  LBYLogFile= LBYLOG
< Library:LBYI >
  DBTable    = LBYD
< DBTable:LBYI >
  DBHandler  = ODBC
< DBTable:LBYD >
  DBHandler  = ODBC
  UniqueTag  = ARCKEY+SEQ_NUM
< DBTable:LBYLOG >
  DBHandler  = ODBC
< DBTable:CATALOG >
  DBHandler  = ODBC
  UniqueTag  = CATALOGID
< DBHandler:ODBC >
  CreateIndex = No
  CreateTable= Yes
  Debug       = No
  Passwd      = password
  Qualifier   = LBYDBASE
  Server      = LBYSQL
  UserID      = userID
< ODBC_FileConvert >
  LBYI        = DAP110_LBYI
  LBYD        = DAP110_LBYD
  LBYLog      = DAP110_LBYLOG
```

Option	Description
--------	-------------

MasterResource control group

BDFFile	This option tells the system you want to retrieve Business Definition (BDF) resources from a library named <i>LBYI</i> .
DALFile	This option tells the system you want to retrieve DAL scripts and DAL script libraries from a library named <i>LBYI</i> .

Option	Description
DDTFile	This option tells the system you want to retrieve DDT files from a library named <i>LBYI</i> .
FORFile	This option tells the system you want to retrieve form (FOR) resources from a library named <i>LBYI</i> .
FormFile	This option tells the system you want to retrieve FAP files from a library named <i>LBYI</i> .
GRPFile	This option tells the system you want to retrieve group form (GRP) resources from a library named <i>LBYI</i> .
LogoFile	This option tells the system you want to retrieve graphics (LOG) files from a library named <i>LBYI</i> .

LibraryManager control group

LbyLogFile	This option tells the system the name of the library log file is <i>LBYLOG</i> . The library log contains information about resources that are added to, deleted from, or updated in the library. The LbyLogFile does not have to use the same type of database handler as the library index and data portions.
------------	--

Library:LBYI control group

DBTable	This option tells the system the data component of the library named <i>LBYI</i> is called <i>LBYD</i> . In this example, the names <i>LBYI</i> and <i>LBYD</i> emphasize that one table, <i>LBYI</i> , represents the library index and one table, <i>LBYD</i> , represents the library data. You can call these tables any name you like but the name must be eight characters or less. See the <i>ODBC_FileConvert</i> control group to map these eight-character names to longer table names.
---------	--

DBTable:LBYI control group

DBHandler	This option tells the system to access the table known as <i>LBYI</i> using the database handler named <i>ODBC</i> . Because of this INI value, the system later expects to find a control group named <i>DBHandler:ODBC</i> . Microsoft's SQL Server is an ODBC-compliant database.
-----------	---

DBTable:LBYD control group

DBHandler	This option tells the system to access the table known as <i>LBYD</i> using the database handler named <i>ODBC</i> . Because of this INI value, the system later expects to find a control group named <i>DBHandler:ODBC</i> .
UniqueTag	This option tells the system the columns <i>ARCKEY</i> and <i>SEQ_NUM</i> can be combined to represent a unique tag for the table. This unique tag is only used for internal purposes. If you do not specify a unique tag for this table, and a column with the name <i>UNIQUE_ID</i> does not exist within the table, you receive warning messages indicating no unique tag is defined. Include this option to avoid those warning messages.

DBTable:LBYLOG control group

DBHandler	This option tells the system to access the table known as <i>LBYLOG</i> using the database handler named <i>ODBC</i> . Because of this INI value, the system later expects to find an INI control group named <i>DBHandler:ODBC</i> .
-----------	---

DBTable:CATALOG control group

Option	Description
DBHandler	This option tells the system to access the table known as <i>CATALOG</i> using the database handler named <i>ODBC</i> . The <i>CATALOG</i> table is used to temporarily store <i>CATALOGID</i> values which are used to construct an <i>ARCKEY</i> .
UniqueTag	This option tells the system the column <i>CATALOGID</i> represents a unique tag for this table. This unique tag is only used for internal purposes. If you do not specify a unique tag for this table, and a column with the name <i>UNIQUE_ID</i> does not exist within the table, you receive warning messages indicating no unique tag is defined. Include this option to avoid those warning messages.

DBHandler:ODBC control group

Class	This option tells the system this database handler represents a handler for Microsoft's Open Data Base Connectivity (ODBC) standard. This option is unnecessary if the name of the handler (<i>ODBC</i> in this case) matches one of the Documaker pre-defined values such as, <i>DB2</i> , <i>ODBC</i> , <i>ORA</i> , or <i>DMS</i> .
Debug	This option turns off tracing for the Documaker ODBC database handler, which is the default. Normally you should omit the Debug option or set it to No. In troubleshooting situations, set this option to Yes and examine the messages written to the trace file.
CreateIndex	This option tells the system not to attempt to create database indexes (always set to No).
CreateTable	This option tells the system that if a table needed does not exist at run time, it should create it.
Passwd	This option tells the system the password to use when connecting to the database management system.
Qualifier	This option tells the system the name of the database for this database handler is <i>LBYDATABASE</i> . If you omit this option, the database set up as the default database for the <i>LBYSQL</i> ODBC data source is used.
Server	This option tells the system the name of the ODBC data source for this database handler is <i>LBYSQL</i> . You must define an ODBC data source by this name.
UserID	This option tells the system the user ID to use when connecting to the database management system.

ODBC_FileConvert control group

LBYP	This option tells the system the table referenced in INI options as <i>LBYP</i> is really named <i>DAP110_LBYP</i> on the database management system. This control group lets you map table names of eight characters or less to table names longer than eight characters. The table name you specify must adhere to the table naming conventions for the database management system.
LBYPD	This option tells the system the table referenced in INI options as <i>LBYPD</i> is really named <i>DAP110_LBYP</i> on the database management system.
LBYPLog	This option tells the system the table referenced in INI options as <i>LBYP</i> is really named <i>DAP110_LBYP</i> on the database management system.

USING THE ORACLE ODBC DRIVER

You can use the Oracle ODBC driver on Windows.

Note You can have up to 32 SQL static variables.

Data Format Definition (DFD) Requirements

Sample CARFILE.DFD file

To use a library using the Oracle ODBC driver, you must use an Oracle Insurance-supplied CARFILE.DFD file that differs from the standard (internal) DFD definition. The supplied CARFILE.DFD file is included in the sample RPEX1 resources in the directory:

```
..\DEFLIB\ODBC_ORA\CARFILE.DFD
```

The contents of the CARFILE.DFD file are listed below:

```
; CARFILE.DFD - use this DFD when referencing a library or archive
with the Oracle ODBC driver.
;
< Fields >
  FieldName = ARCKEY
  FieldName = SEQ_NUM
  FieldName = CONT_FLAG
  FieldName = TOTAL_SIZE
  FieldName = CARDATA
< Field:ARCKEY >
  INT_Type = CHAR_ARRAY
  INT_Length = 18
  EXT_Type = CHAR_ARRAY
  EXT_Length = 18
  Key = N
  Required = N
< Field:SEQ_NUM >
  INT_TYPE = CHAR_ARRAY
  INT_LENGTH = 5
  EXT_TYPE = CHAR_ARRAY
  EXT_LENGTH = 5
  KEY = N
  REQUIRED = N
< Field:CONT_FLAG >
  INT_TYPE = CHAR_ARRAY
  INT_LENGTH = 1
  EXT_TYPE = CHAR_ARRAY
  EXT_LENGTH = 1
  KEY = N
  REQUIRED = N
< Field:TOTAL_SIZE >
  INT_TYPE = LONG
  INT_LENGTH = 4
  EXT_TYPE = DOUBLE
  EXT_LENGTH = 4
  KEY = N
  REQUIRED = N
< Field:CARDATA >
  INT_TYPE = BLOB
  INT_LENGTH = 252
  EXT_TYPE = BLOB
  EXT_LENGTH = 252
  KEY = N
  REQUIRED = N
```

```

< Keys >
    KEYNAME = ARCKEY
    KEYNAME = SEQ_NUM
    KEYNAME = CAR_KEY
< Key:ARCKEY >
    EXPRESSION = ARCKEY+SEQ_NUM
    FIELDLIST = ARCKEY,SEQ_NUM
< Key:SEQ_NUM >
    EXPRESSION = SEQ_NUM
    FIELDLIST = SEQ_NUM
< Key:CAR_KEY >
    EXPRESSION = ARCKEY
    FIELDLIST = ARCKEY

```

To use the supplied CARFILE.DFD file, follow these steps:

1. Copy the CARFILE.DFD file into the directory where you store other DFD files (generally the \DEFLIB directory).
2. Tell the system to use the CARFILE.DFD file by adding this option to the INI file:

```

< ArcRet >
    CARFileDFD = ..\DEFLIB\CARFILE.DFD

```

Creating the Database and Tables

On Windows, you can use Studio to create the library index and data files. You can also use the LBRYMGR utility do it.

Note For more information about the LBRYMGR utility, see the [Docutoolbox Reference](#).

Sample INI Options

Use INI options like the following to create a library using the Oracle ODBC driver and to load resources from that library:

```

< MasterResource >
    BDFFile    = LBYI
    DALFile    = LBYI
    DDTFile    = LBYI
    FORFile    = LBYI
    FormFile   = LBYI
    GRPFile    = LBYI
    LogoFile   = LBYI
< LibraryManager >
    LBYLogFile= LBYLOG
< Library:LBYI >
    DBTable    = LBYD
< DBTable:LBYI >
    DBHandler  = ODBC
< DBTable:LBYD >
    DBHandler  = ODBC
    UniqueTag  = ARCKEY+SEQ_NUM
< DBTable:LBYLOG >
    DBHandler  = ODBC
< DBTable:CATALOG >
    DBHandler  = ODBC
    UniqueTag  = CATALOGID
< DBHandler:ODBC >
    CreateIndex = No
    CreateTable = Yes

```

```

Debug          = No
Passwd         = password
Qualifier      = LBYDBASE
Server         = LBYORA
UserID         = userID
< ODBC_FileConvert >
LBYI           = DAP110_LBYI
LBYD           = DAP110_LBYD
LBYLog         = DAP110_LBYLOG
    
```

Option Description

MasterResource control group

BDFFile	This option tells the system you want to retrieve business definition (BDF) resources from a library named <i>LBYI</i> .
DALFile	This option tells the system you want to retrieve DAL scripts and DAL script libraries from a library named <i>LBYI</i> .
DDTFile	This option tells the system you want to retrieve DDT files from a library named <i>LBYI</i> .
FORFile	This option tells the system you want to retrieve form (FOR) resources from a library named <i>LBYI</i> .
FormFile	This option tells the system you want to retrieve FAP files from a library named <i>LBYI</i> .
GRPFile	This option tells the system you want to retrieve group form (GRP) resources from a library named <i>LBYI</i> .
LogoFile	This option tells the system you want to retrieve graphics (LOG) files from a library named <i>LBYI</i> .

LibraryManager control group

LbyLogFile	This option tells the system the name of the library log file is <i>LBYLOG</i> . The library log contains information about resources added to, deleted from, or updated in the library. The LbyLogFile does not have to use the same type of database handler as the library index and data portions.
------------	---

Library:LBYI control group

DBTable	This option tells the system the data component of the library named <i>LBYI</i> is called <i>LBYD</i> . In this example, the names <i>LBYI</i> and <i>LBYD</i> emphasize that one table, <i>LBYI</i> , represents the library index and one table, <i>LBYD</i> represents the library data. You can call these tables anything you like but the name must be eight characters or less. Use the <i>ODBC_FileConvert</i> control group if you need to map these eight character names to longer table names.
---------	--

DBTable:LBYI control group

DBHandler	This option tells the system to access the table known as <i>LBYI</i> using the database handler named <i>ODBC</i> . Because of this INI option, the system later expects to find a control group named <i>DBHandler:ODBC</i> . Microsoft's SQL Server is an ODBC-compliant database.
-----------	--

DBTable:LBYD control group

DBHandler	This option tells the system to access the table known as <i>LBYD</i> using the database handler named <i>ODBC</i> . Because of this INI option, the system later expects to find a control group named <i>DBHandler:ODBC</i> .
-----------	---

Option	Description
UniqueTag	This option tells the system the columns ARCKEY and SEQ_NUM can be combined to represent a unique tag for the table. This unique tag is only used for internal purposes. If you do not specify a unique tag for this table, and a column with the name UNIQUE_ID does not exist, you receive warning messages indicating no unique tag id defined. Include this option to avoid those warning messages.

DBTable:LBYLOG control group

DBHandler	This option tells the system to access the table known as <i>LBYLOG</i> using the database handler named <i>ODBC</i> . Based on this INI option, the system expects to find a control group named <i>DBHandler:ODBC</i> .
-----------	---

DBTable:CATALOG control group

DBHandler	This option tells the system to access the table known as <i>CATALOG</i> using the database handler named <i>ODBC</i> . The <i>CATALOG</i> table temporarily stores the <i>CATALOGID</i> values used to construct an <i>ARCKEY</i> .
UniqueTag	This option tells the system the column <i>CATALOGID</i> represents a unique tag for this table. This unique tag is only used for internal purposes. If you do not specify a unique tag for this table, and a column with the name <i>UNIQUE_ID</i> does not exist, you receive warning messages indicating no unique tag is defined. Include this option to avoid those warning messages.

DBHandler:ODBC control group

Class	This option tells the system this database handler represents a handler for Microsoft's Open Data Base Connectivity (ODBC) standard. This option is unnecessary if the name of the handler (ODBC in this case) matches one of the Documaker pre-defined values, such as DB2, ODBC, ORA, or DMS.
CreateIndex	This option tells the system not to attempt to create database indexes (always set to No).
CreateTable	This option tells the system that, if a table needed does not exist at run time, it should create it.
Debug	This option turns off tracing for the Documaker ODBC database handler, which is the default. Normally you should omit the Debug option or set it to No. In troubleshooting situations, set this option to Yes and examine the trace messages written to the trace file.
Passwd	This option tells the system the password to use when connecting to the database management system.
Qualifier	This option tells the system the name of the database for this database handler is <i>LBYDBASE</i> . If you omit this option, the system uses the database set up as the default database for the <i>LBYORA</i> ODBC data source.
Server	This option tells the system the name of the ODBC data source for this database handler is <i>LBYORA</i> . You must define an ODBC data source by this name.
UserID	This option tells the system the user ID to use when connecting to the database management system.

ODBC_FileConvert control group

Option	Description
LBYI	<p>This option tells the system the table referenced in INI options as <i>LBYI</i> is really named <i>DAP110_LBYI</i> on the database management system.</p> <p>This control group lets you map table names of eight characters or less to table names longer than eight characters.</p> <p>The table name you specify must adhere to the table naming conventions for the database management system.</p>
LBYD	<p>This option tells the system the table referenced in INI options as <i>LBYD</i> is really named <i>DAP110_LBYI</i> on the database management system.</p>
LBYLog	<p>This option tells the system the table referenced in INI options as <i>LBYI</i> is really named <i>DAP110_LBYI</i> on the database management system.</p>

USING DOCUMANAGE

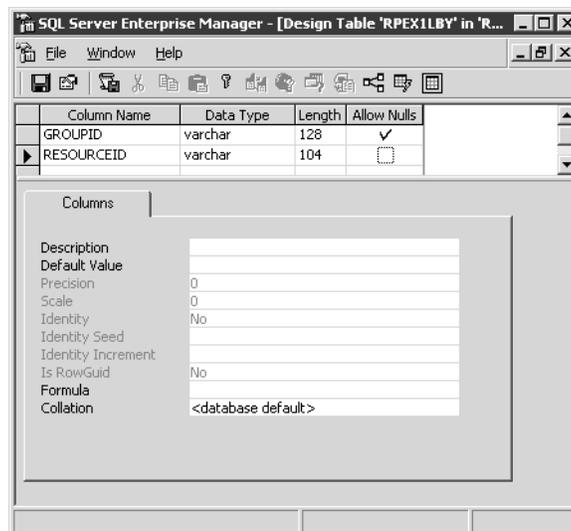
Studio supports installable interfaces to alternate document management systems (DMS). The default DMS is the DAP archive system, but you can also use Documanager. To use Documanager, follow these steps:

Note To store an 11.0 resource library in Documanager, you must have Documanager version 6.3 SR 2 or version 6.4 SR 1 or higher.

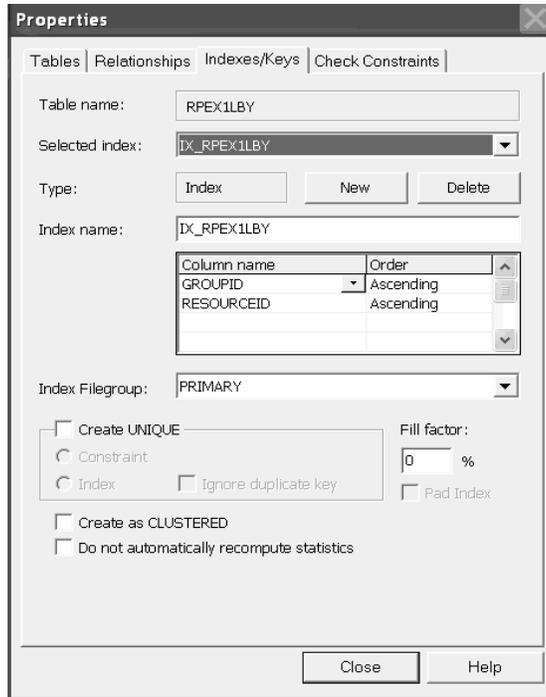
1. Create a table in an ODBC database. You can name the table anything you like. This example uses RPEX1LBY. Include in the table the following fields. All fields should be of the VARCHAR type:

Field	Description
GROUPID	varchar 128
RESOURCEID	varchar 104

This table maps to the internally-defined table for Studio. The table corresponds to a cabinet in the Documanager environment.

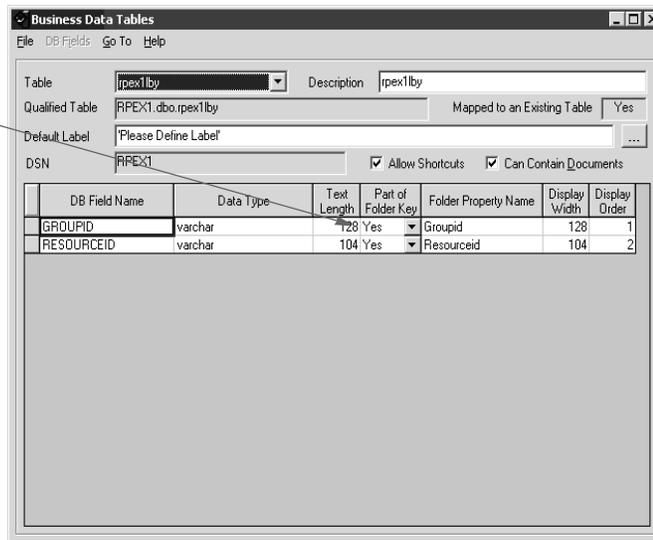


For performance reasons, it is best to add an index on the new table by GROUPID and RESOURCEID, as shown on the following window.



2. Set up an ODBC connection for this database using Data Sources (ODBC) in Administrative Tools in Windows 2000 Server. This lets Documange Administrator recognize this database and table for *powermapping*.
3. Powermap the database table you created (RPEX1LBY in this example) using Documange Administrator.
4. After you powermap the database table, edit the Business Data table to set the Part of Folder Key values to Yes for the GROUPID and RESOURCEID database fields.

Make sure these values are set to Yes.



5. Create a cabinet. The name of this cabinet (RPEX1LBY in this example) must be used in the FAPCOMP.INI file and other Documaker INI files that access the library from Documanager. To create the cabinet in Documanager Administrator, select Cabinet, File, New, and enter the name of your new cabinet (RPEX1LBY in this example).

On the Cabinet Definition window, select the table you powermapped from the Table list then select the Folder Properties tab and set the Editable values to Yes for the GROUPID, RESOURCEID, and UNIQUE_ID database fields.

Click here to select the table you powermapped

The screenshot shows the 'Cabinet Definition - Released' dialog box. The 'Cabinet' field is set to 'rpex1lby'. Below it is a table with the following data:

Level	Table	Show	Documents	Label Formula
1	rpex1lby	Yes	Yes	rpex1.dbo.rpex1lby.GROUPID + ' ' + rpex1.dbo

The 'Folder Properties' tab is active, showing a table of database fields:

Database Field Name	Display	Editable	Required
GROUPID	Yes	Yes	Yes
RESOURCEID	Yes	Yes	Yes

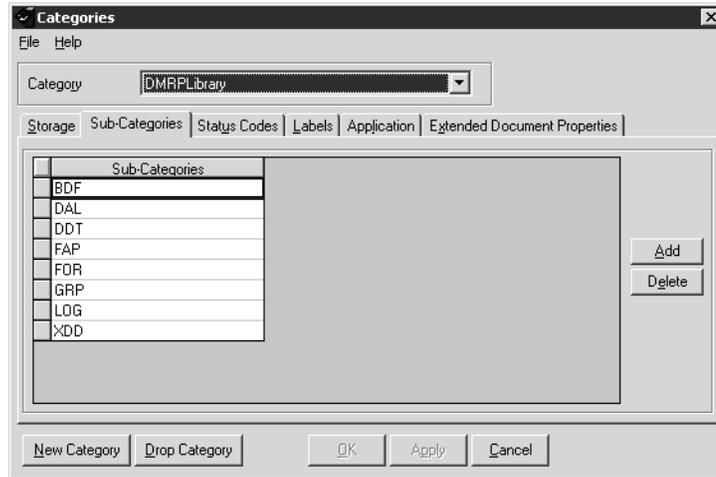
The 'Distribution List' section shows 'Public Access' selected and a 'Group' field with 'Add' and 'Delete' buttons.

Once you define the cabinet, select Save, Release, and then Refresh Servers from the File menu.

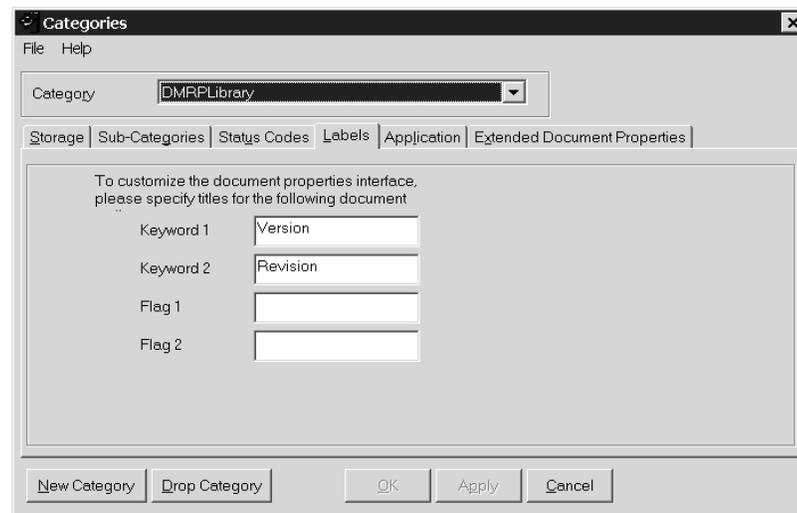
6. Define a category called *DMRPLibrary* using the Documanager Administrator. Set up a storage location under the Storage tab. This is usually *DEFAULT/Permanent* or *Temporary*. Under the Sub-Categories tab, add these sub-categories:
 - BDF
 - DAL
 - DDT
 - FAP
 - FOR
 - GRP
 - LOG

- XDD

Here is an example:



7. Next, select the category *DMRPLibrary*, click the Labels tab, and enter **Version** in the Keyword1 field and **Revision** in the Keyword2 field.

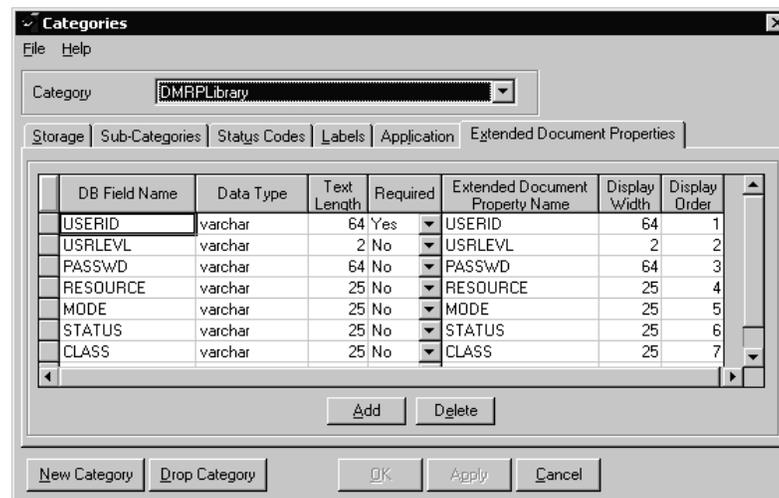


8. Next, select the category *DMRPLibrary*, click on the Extended Document Properties tab, and add the following information:

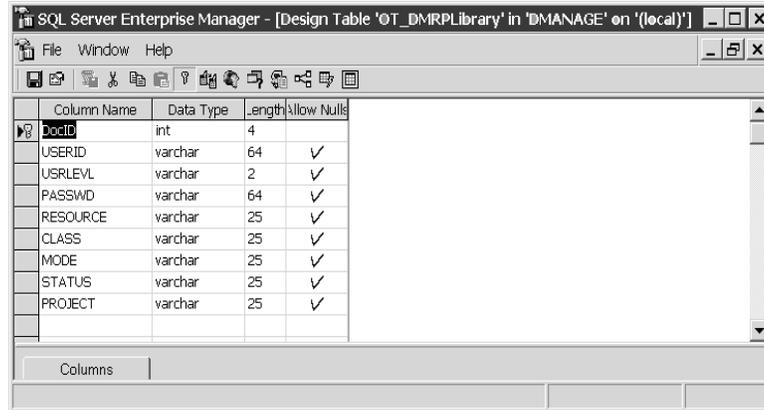
Note The display order is not important.

Field name	Data type	Length	Required	Extended document property name	Display Width	Display Order
USERID	varchar	64	Yes	USERID	64	1
USRLEVL	varchar	2	No	USRLEVL	2	2
PASSWD	varchar	64	No	PASSWD	64	3
RESOURCE	varchar	25	No	RESOURCE	25	4
MODE	varchar	25	No	MODE	25	5
STATUS	varchar	25	No	STATUS	25	6
CLASS	varchar	25	No	CLASS	25	7
PROJECT	varchar	25	No	PROJECT	25	8

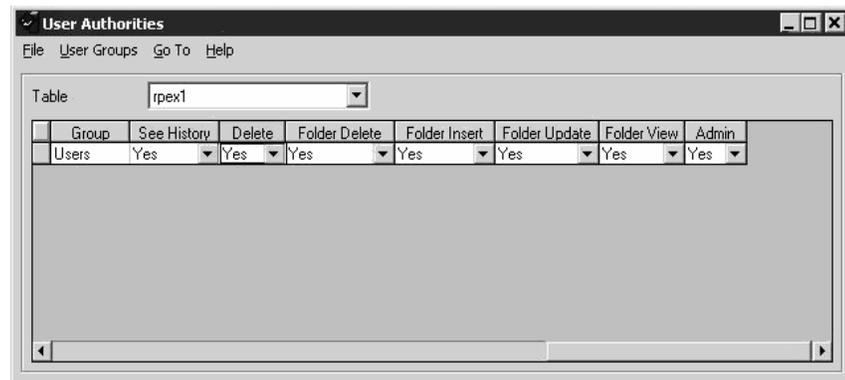
As shown here:



Documange creates the table OT_DMRPLibrary in the DMANAGE (DM administrative tables) database.



Be sure to select Yes for Folder Update, Folder Insert, Folder Delete, and Administrator Authorities in the Authorities section of Documange Administrator.



- Load the test master resource libraries (MRLs) into the config RPEX1. Here are the necessary FAPCOMP.INI options for Documaker: (RPEX1LBY is the Documange cabinet name that contains the objects in this example)

```
< Config:RPEX1 >
DALFile   = RPEX1LBY
DDTFile   = RPEX1LBY
DDTLib    = RPEX1LBY
FormFile  = RPEX1LBY
LbyLib    = RPEX1LBY
LogoFile  = RPEX1LBY
LogoLib   = RPEX1LBY
FormLib   = FORMS
BDFFile   = RPEX1LBY
GRPFile   = RPEX1LBY
FORFile   = RPEX1LBY
< DBHandler:LBYSETUP >
Class     = DMIL
Domain    = FSI
RPCHost   = 10.1.10.228
UserID    = administrator
PassWord  = 1234589
Debug     = No
< DBTable:RPEX1LBY >
SelfIndex = Yes
DBHandler = LBYSETUP
```

Note The value in the DBHandler option must match the name in the DBHandler:XXX control group. For instance, in the above example *LBYSETUP* is used in both places.

10. Create a response file for RPEX1. This response file will add FAP, DDT, DAL, and LOG files into a file called RPEX1.RSP. To create the response file, run the LBRYMGR utility in the directory where the FAPCOMP.INI file resides. Use the following syntax:

```
LBRYMGRW /RSP=RPEX1.RSP /INI=FAPCOMP.INI /  
FAP=D:\FAP\MSTRRES\RPEX1\FORMS\*.FAP /  
LOGO=D:\FAP\MSTRRES\RPEX1\FORMS\*.LOG /  
DDT=D:\FAP\MSTRRES\RPEX1\DEFLIB\*.DDT /  
DAL=D:\FAP\MSTRRES\DEFLIB\*.DAL
```

11. Finally, run the response file. For more information on running the response file, see *Running Response Files on page 379*.

WORKING WITH LIBRARIES

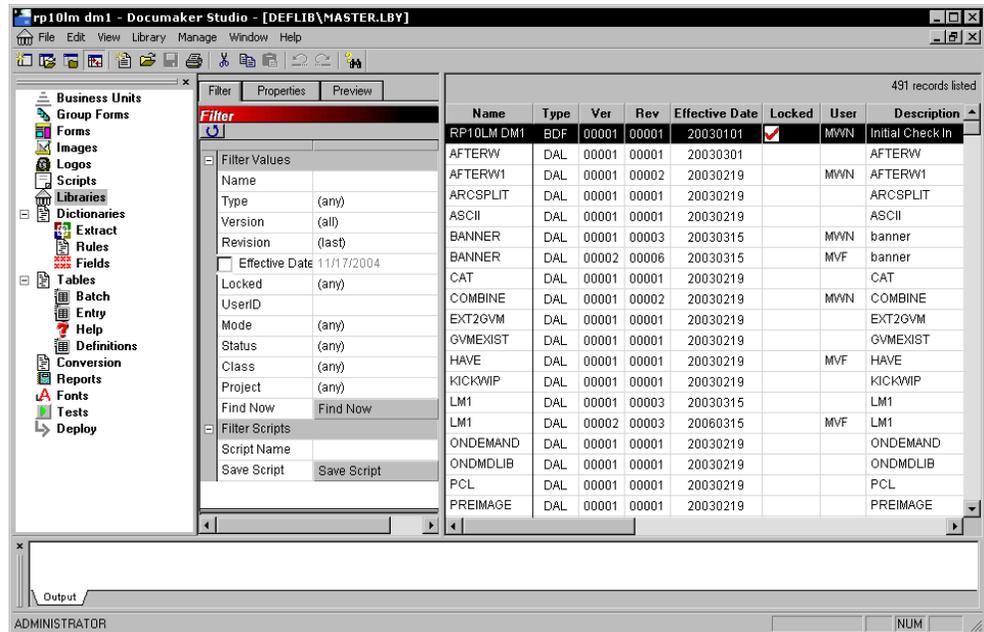
Studio lets you open a library, add resources to a library, delete, or extract resources from a library and edit some of the properties of a resource. You can also run a response file, print a list of resources, and view a history of changes made to a resource. For initialization settings, Studio uses the INI files referenced in the Studio workspace file. The Studio workspace file usually references the FSIUSER.INI and FSISYS.INI INI files.

- *Opening a Library on page 345*
- *Adding Resources to a Library on page 347*
- *Importing Files on page 350*
- *Importing Libraries on page 351*
- *Checking Out Resources on page 354*
- *Checking In Resources on page 356*
- *Unlocking Resources on page 358*
- *Promoting Resources on page 359*
- *Filtering Resources on page 363*
- *Editing Resource Information on page 365*
- *Deleting Resources on page 367*
- *Searching the Library on page 368*
- *Extracting Resources on page 372*
- *Expiring Resources on page 376*
- *Reviewing a History of Resource Changes on page 377*
- *Running Response Files on page 379*

OPENING A LIBRARY

A workspace is associated with a specific library. When you click on Library, the contents of the library appear, as shown below. This is known as the Library view.

Note Once you are in Studio, there is no visible difference between a standard library, a DBMS library, or a Documanager library. You are only concerned with the library type when you are creating a new workspace or using Manage, Settings, Libraries to define a new library.



Here you see all the resources in the library along with their type, version/revision, effective date, description, last modified date, mode, status, class, and project, along with information about the user who created the resource and whether or not it is locked.

From this window you can perform these tasks:

To	Do this
Check out a resource	Highlight the resource and right click. Then choose Check Out.
Check in a resource	Highlight the resource and right click. Then choose Check In. The Check In window appears.
Unlock a resource	Highlight the resource and right click. Then choose Unlock.
See a history of the changes made to the resource	Highlight the resource and right click. Then choose History. The Library Log window appears.
Save a copy of the resource to disk	Highlight the resource and right click. Then choose Extract. The Extract Library Resources window appears. From this window you can specify where to extract the resource and other options.

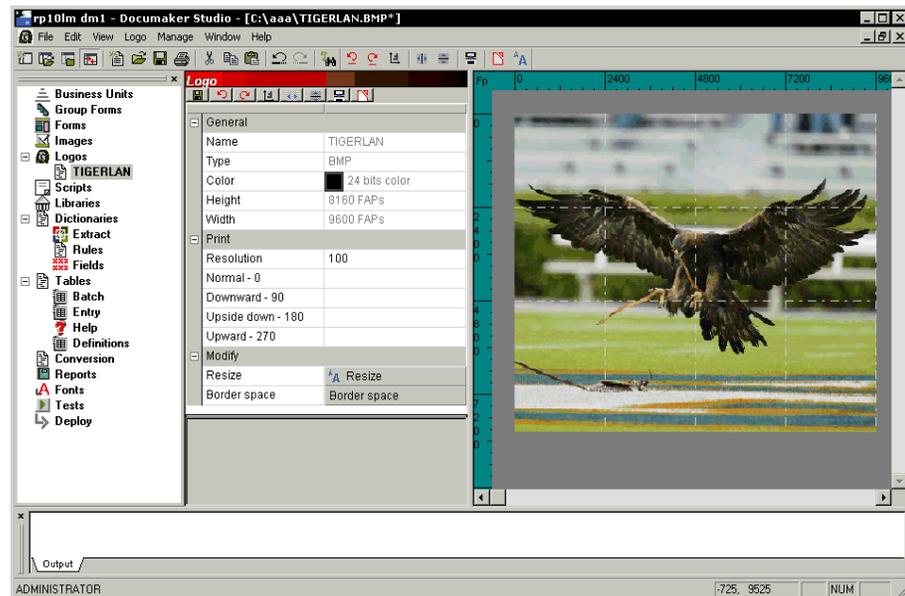
To	Do this
Print a listing of the resources	Right click in the list of resources and choose Print this Window. This prints a list of all the resources currently being shown. You can use the Filter tab to limit that list to only those resources you want to see. For instance, you can choose to see only DAL scripts or FAP files.

ADDING RESOURCES TO A LIBRARY

When you add resources to a library, the system writes information about the resource into the index portion of the library and places the resource itself in the data portion of the library.

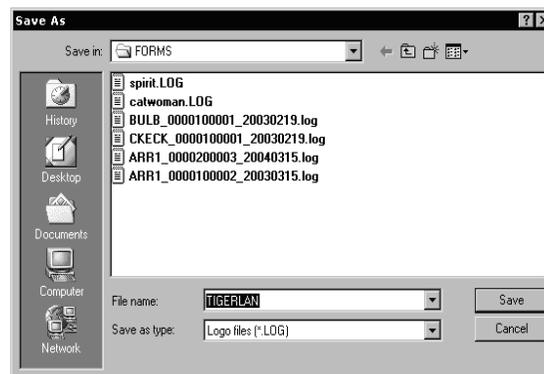
There are several ways to add a resource to your library. For instance, you can use the Library, Import Files options to add several files at once. See *Importing Files on page 350* for more information. You can also add a single resource to your library.

To add a single resource to your library, you simply choose the File, Check In option. For example, suppose you have opened a bitmap file and you want to check it into the library as a LOG file.



Follow these steps:

1. Choose File, Check In. The Save As window appears.



1. Make sure you have the name and location you want to assigned to the resource and click Save. The Check In window appears.

2. Make entries into the following fields as necessary:

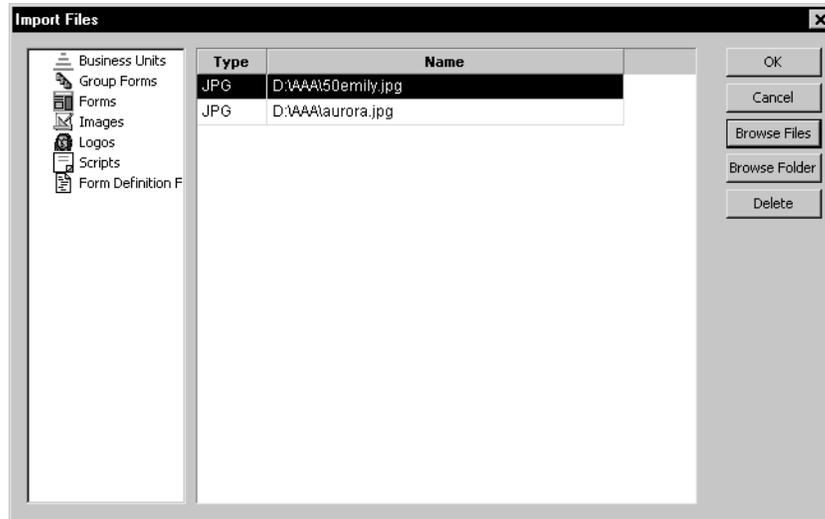
Field	Description
Description	<p>You can enter up to 100 characters to indicate what type of change was made to the resource. This field defaults to the contents of the Description field of the resource checked out, but you can enter a different description if you like.</p> <p>The Description field appears on the Check Out Section window that appears when you select a file to check out.</p>
Increment Version	<p>Check this box if you want Studio to increment the version number.</p> <p>If you only make a minor change you will probably want to just increment the revision number and not the version number. The revision number is automatically incremented by one if you do not check the Increment Version field.</p> <p>If you make a major change, or if you need to change the effective date of the resource that you are checking in, check the Increment Version field. This tells Studio to increment the version number by one and set the revision number to one.</p> <p>Once this field is checked, you can modify the Effective Date field. This makes sure all revisions of a resource have the same effective date. When a resource is checked out, changed, and checked back in, the older version/revision combinations of that resource remain in the library and can be loaded by the GenData and GenPrint programs or retrieved by Documaker Workstation or Docupresentation as needed.</p> <p>Keep in mind, however, you can only check out the latest revision of each version of a resource.</p>
Effective Date	<p>All revisions of a specific version of a resource must have the same effective date. This field is only available if you check the Increment Version field.</p> <p>The effective date defaults to MM/DD/YYYY, but may have a different format depending on your locale setting.</p>
Mode	<p>Use this field to assign a mode to the resource as it is checked in. For instance, you can use the Mode field to specify where in the development cycle the resource is.</p> <p>See <i>Defining Mode, Status, Class, and Project Options</i> on page 380 for information on how to set up modes.</p>
Status	<p>Use this field to assign a status to the resource as it is checked in. For instance, you could use the Status field to indicate whether a resource has passed or failed testing.</p> <p>See <i>Defining Mode, Status, Class, and Project Options</i> on page 380 for information on how to set up statuses.</p>

Field	Description
Class	Use this field to assign a class to the resource as it is checked in. For instance, you could use the Class field to indicate the market in which a resource was applicable. See <i>Defining Mode, Status, Class, and Project Options on page 380</i> for information on how to set up classes.
Project	Use this field to assign a project ID to the resource as it is checked in. For instance, you could use the Project field to indicate which project a resource was associated with. See <i>Defining Mode, Status, Class, and Project Options on page 380</i> for information on how to set up projects.

3. Click Ok when finished or Cancel to exit without checking in the file.

IMPORTING FILES

You can use the Library, Import Files option to add a number of files into your library. When you choose this option, the Import Files window appears:



You can use the Browse buttons to select the files you want to import. When you finish, click Ok and Studio adds them to the library. You can then modify the property settings as necessary.

Note If you experience delays when checking in or out forms or sections or when importing files, choose Manage, System, Settings and set the LibAutoRefresh option in the LibraryManager control group to No. Keep in mind that if you set this option to No, the system will not update the contents of the Library window you currently have open. You can manually refresh the display by clicking the Filter Now button.

IMPORTING LIBRARIES

You can use the Library, Import Library option to import resources from another library to the workspace library. You can only import resources from a library that is in xBase/CARFile format. You cannot import resources from a library that is in a DBMS format or one that is in Documanage.

When you choose this option, the Open window appears:



When you select a library and click Open, the following window appears:



You can click Cancel to stop the import. Studio lists the resources as it imports them in the Status bar. If you right click in the Status bar and select Print, you can print a list of the resources it imported.

MIGRATING A WORKSPACE LIBRARY

You can easily change the format of a library within a workspace. For instance, if you have administrator level security, you can use the migration wizard to quickly change a library from xBase to SQL or Documange.

In addition to moving the contents of the library, the wizard changes the applicable workspace INI options so the new location becomes the primary library for the workspace. This is often useful if you find the default library is too limited for the solution you are creating.

Note Be sure to back up your library and workspace settings before migrating a library. Also, make user no users are actively using the system. Once the library changes, users will need to load the new workspace settings.

To migrate a library, choose the Migrate option from the Library menu. The Library Migration window appears.

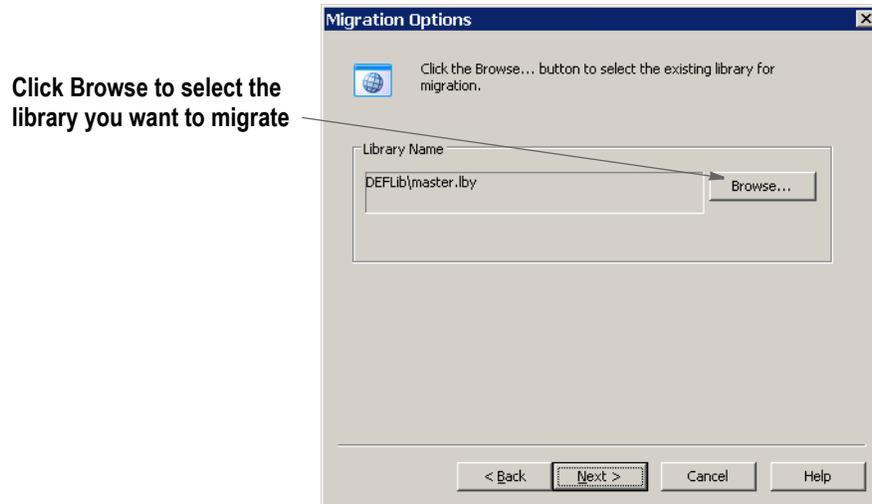
Choose from these options to tell Studio how you would like to migrate the library



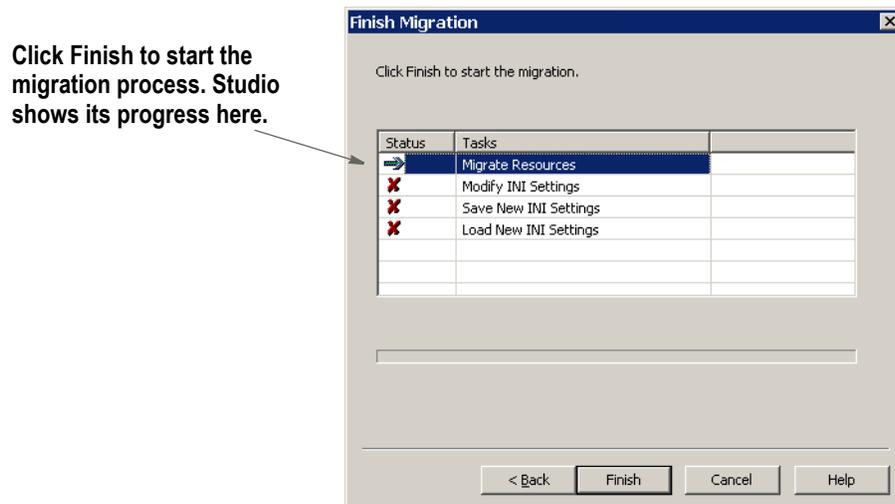
You can choose from these options:

Migration option	Description
Create a new library and copy the current library contents	When creating a new library, you specify the name, location, and description. If you specify an existing library, Studio copies all of the resources into that library. If there are existing resources with the same name, Studio treats this occurrence the same as a promotion and will not migrate the resource if it has the same modification date.
Merge the current library contents into an existing library	Keep in mind the library must exist on disk or in the workspace's list of associated libraries. If there are existing resources with the same name, Studio treats this occurrence the same as a promotion and will not migrate the resource if it has the same modification date.
Change the associated workspace library without copying contents	When choosing a library, you can choose from the workspace's list of associated libraries or browse for another library on disk. Studio does not try to migrate resources but assumes the library is already populated for this workspace.

After you choose the appropriate migration option, click Next. The Migration Options window appears so you can select the library you want to migrate.



Click Next and the Finish Migration window appears.



Click Finish to start the migration process.

CHECKING OUT RESOURCES

Studio lets you check out resources, edit them and check them back in. While you have a resource checked out, it is *locked* and others cannot check it out, although they can open it in view-only mode.

There are several ways to check out resources. For instance, to check out a section you simply double click on Sections, then select the section you want to check out on the Open File window and click Ok.

Another way to check out a resource is to double click Library in the workspace to bring up the library view, then right click on a resource in the library view and select Check Out.

Red check marks indicate resources checked out by other users.

Green check marks indicate resources you have checked out.

Name	Type	Ver	Rev	Effective Date	Locked	User	De
102SMFNT	FAP	00001	00001	20030219		MVF	102SM
APPENDS	FAP	00001	00002	20030219		MVN	APPEN
ARRWDMY	FAP	00001	00001	20030219		MVF	ARRW
BASE1	FAP	00001	00001	20030219		MVF	BASE1
BASE2	FAP	00001	00001	20030219		MVF	BASE2
BASE3	FAP	00001	00001	20030219		MVF	BASE3
BASE4	FAP	00001	00001	20030219		MVF	BASE4
BASE5	FAP	00001	00001	20030219		MVF	BASE5
BASE6	FAP	00001	00001	20030219	✓	MVF	BASE6
BASE7	FAP	00001	00001	20030219		MVF	BASE7
BASE8	FAP	00001	00001	20030219		MVF	BASE8
BCKGRND	FAP	00001	00001	20030219	✓	MVN	BCKGI
BILLCGAR	FAP	00001	00001	20030219		MVF	BILLC
BILLCSWR	FAP	00001	00001	20030219	✓	MVF	BILLC
BILLCWT3	FAP	00001	00001	20030219		MVF	BILLC

Note You can only check out the latest revision of a specific version of a resource. You cannot, for example, check out version 1.4 of Q1SNAM if a version 1.5 exists.

When you check out a resource, this is what happens:

- In the library index record for the resource you are checking out, the RecStatus column is set to LOC and your user ID is placed into the UserID column.

In the library index, the Locked column will now contain a green check mark to indicate you have checked out the resource (a red check mark indicates another user has checked out the resource).

- A long (or *versioned*) resource name for the resource is created and the resource is retrieved from the library and placed onto disk with the long file name. The long file name consists of the resource name concatenated with the version, revision and effective date of the resource. For example, if version 2, revision 3 of Q1ADDR, with an effective date of 4/21/2007 is checked out, it will be written to disk with this name:

```
Q1ADDR_0000200003_20070421.FAP
```

The file is marked as a read/write file. The version, revision, and effective date for the resource appear on the title bar so you can tell which version/revision of the resource you are editing.

The resource is stored on disk in the appropriate directory. You specify these directories in the MasterResource control group, as shown below:

```
< MasterResource >
  BDFLib = \BDFLIB
  GRPLib = \GRPLIB
  FORLib = \FORLIB
  FormLib= \FORMS
  DDTLib = \DEFLIB
  LogoLib= \FORMS
  DEFLib = \DEFLIB
```

Option	Description
BDFLib	BDF (business definition) files are stored here.
GRPLib	GRP (group) files are stored here.
FORLib	FOR (form) files are stored here.
FormLib	FAP files (sections) are stored here.
DDTLib	DDT files are stored here.
LogoLib	Graphics (LOG) files are stored here. This is often the same directory used for FAP files.
DEFLib	DAL (Document Automation Library) files are stored here.

Note If you experience delays when checking in or out forms or sections or when importing files, choose Manage, System, Settings and set the LibAutoRefresh option in the LibraryManager control group to No. Keep in mind that if you set this option to No, the system will not update the contents of the Library window you currently have open. You can manually refresh the display by clicking the Filter Now button.

CHECKING IN RESOURCES

Once you finish making changes, you can check in the resource.

When you check in the resource, Studio asks for information the system needs to assign the appropriate version, revision and effective dates.

To check in a resource follow these steps:

1. With the resource open, choose File, Check In. The Check In window appears.

2. Make entries into the following fields as necessary:

Field	Description
Description	<p>You can enter up to 100 characters to indicate what type of change was made to the resource. This field defaults to the contents of the Description field of the resource checked out, but you can enter a different description if you like.</p> <p>Your entry in the Description field appears in the grid displayed on the Open File window and in the Library view.</p>
Increment Version	<p>Check this box if you want Studio to increment the version number.</p> <p>If you only make a minor change you will probably want to just increment the revision number and not the version number. The revision number is automatically incremented by one if you do not check the Increment Version field.</p> <p>If you make a major change, or if you need to change the effective date of the resource that you are checking in, check the Increment Version field. This tells Studio to increment the version number by one and set the revision number to one.</p> <p>Once this field is checked, you can modify the Effective Date field. This makes sure all revisions of a resource have the same effective date. When a resource is checked out, changed, and checked back in, the older version/revision combinations of that resource remain in the library and can be loaded by the GenData and GenPrint programs or retrieved by Documaker Workstation or Docupresentment as needed.</p> <p>Keep in mind, however, you can only check out the latest revision of each version of a resource.</p>
Effective Date	<p>All revisions of a specific version of a resource must have the same effective date. This field is only available if you check the Increment Version field.</p> <p>When you enter an effective date, keep in mind the date must be equal to or later than the effective date of the prior version of the resource.</p> <p>The effective date defaults to MM/DD/YYYY, but may have a different format depending on your locale setting.</p>

Field	Description
Mode	Use this field to assign a mode to the resource as it is checked in. For instance, you can use the Mode field to specify where in the development cycle the resource is. For more information, see <i>Defining Mode, Status, Class, and Project Options on page 380</i> .
Status	Use this field to assign a status to the resource as it is checked in. For instance, you could use the Status field to indicate whether a resource has passed or failed testing. For more information, see <i>Defining Mode, Status, Class, and Project Options on page 380</i> .
Class	Use this field to assign a class to the resource as it is checked in. For instance, you could use the Class field to indicate the market in which a resource was applicable. For more information, see <i>Defining Mode, Status, Class, and Project Options on page 380</i> .
Project	Use this field to assign a project ID to the resource as it is checked in. For instance, you could use the Project field to indicate which project a resource was associated with. For more information, see <i>Defining Mode, Status, Class, and Project Options on page 380</i> .

3. Click Ok when finished or Cancel to exit without checking in the resource.

This is what happens when you check in a resource:

- The contents of the long (or versioned) resource file are written to the data portion of the library and the long file name is deleted. For example, if you had originally checked out version 2, revision 3 of Q1ADDR FAP, with an effective date of 4/21/2007, the file written to disk during check out would have a long file name of:

```
Q1ADDR_0000200003_20070421.FAP
```

During check in, after this file is added to the data portion of the library, the file is deleted from disk.

- A new library index record for the resource is added to the library index file. This record contains the name, type, version, revision, effective date, modification date, description, mode, status, and so on of the resource.

Note the version or the revision, or both, of this library index record differs from those of the library index record for the resource originally checked out.

- The library index record for the resource originally checked out is updated so the RecStatus column is changed from *LOC* to blank. This shows the resource is no longer checked out.

On the Open File window and in the Library view, Studio removes the green check mark that was in the Locked column.

Note If you experience delays when checking in or out forms or sections or when importing files, choose Manage, System, Settings and set the LibAutoRefresh option in the LibraryManager control group to No. Keep in mind that if you set this option to No, the system will not update the contents of the Library window you currently have open. You can manually refresh the display by clicking the Filter Now button.

UNLOCKING RESOURCES

When a resource is checked out it becomes locked. You can tell which resources are locked and by whom by looking at the Locked and User columns of the Open File window, shown here:

This file has been checked out by the current user (ADMIN)

These files have been checked out by other users

Name	Type	Ver	Rev	Effective Date	Locked	User	Description
102SMFNT	FAP	00001	00001	20030219		MVF	102SMFNT 0E
APPENDS	FAP	00001	00002	20030219		MVWN	APPENDS 0E
ARRWDMY	FAP	00001	00001	20030219		MVF	ARRWDMY 0E
BASE1	FAP	00001	00001	20030219		MVF	BASE1 0E
BASE2	FAP	00001	00001	20030219	✓	ADMINI	BASE2 0E
BASE3	FAP	00001	00002	20030219		ADMINI	BASE3 11
BASE4	FAP	00001	00001	20030219		MVF	BASE4 0E
BASE5	FAP	00001	00001	20030219		MVF	BASE5 0E
BASE6	FAP	00001	00001	20030219	✓	MVF	BASE6 0E
BASE7	FAP	00001	00001	20030219		MVF	BASE7 0E
BASE8	FAP	00001	00001	20030219		MVF	BASE8 0E
BCKGRND	FAP	00001	00001	20030219	✓	MVWN	BCKGRND 0E
BILLCGAR	FAP	00001	00001	20030219		MVF	BILLCGAR 0E
BILLCSWR	FAP	00001	00001	20030219		MVF	BILLCSWR 0E
BILLCWT3	FAP	00001	00001	20030219		MVF	BILLCWT3 0E

Only the user who has locked the resource can resume editing the resource or unlock it.

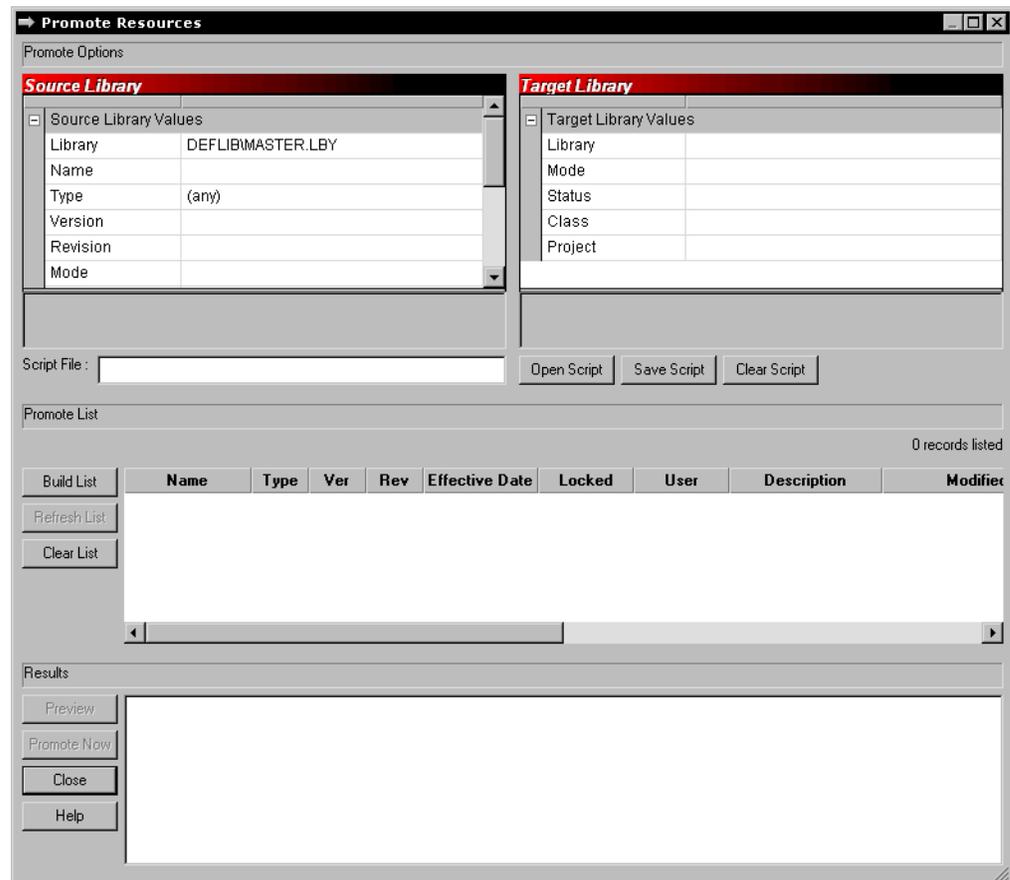
There are several ways to unlock a resource:

- If you are editing a resource and you want to unlock the resource, choose the File, Unlock option. A window appears telling you your changes will be lost if you unlock the resource and asks whether you want to unlock it. Click Yes to unlock the resource.
- Double click on Library in the workspace. From the Library view, right click on the resource you want to unlock and choose Unlock from the popup menu.
- If the resource is in the workspace and has a green check mark next to it, you can either right click on the resource and choose Unlock from the popup menu or highlight the resource and choose Unlock from the File menu.

PROMOTING RESOURCES

You can promote one or more resources from one library to another and in the process have Studio modify the Mode, Status, Class, and Project fields. When you choose the Library, Promote option, the Promote Resources window appears.

Note You must have sufficient access rights to promote resources. If this option is unavailable to you, contact your system administrator.



Identifying the resources to promote

On the Promote Resources window, you use the Source Library options to tell Studio which resources you want to promote.

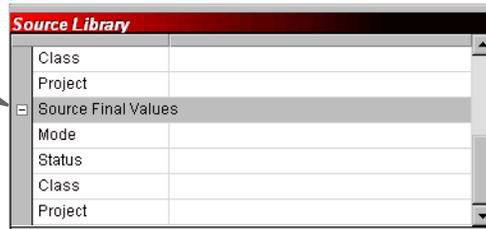
Field	Description
Library	Select the name of the library that contains the resources you want to promote.
Name	Enter the name of the resource you want to promote. You can use wildcards and enter a partial name to have Studio promote a group of similarly named resources. For instance, <i>brt*</i> tells Studio to select all resources with names that start with <i>brt</i> .
Type	Select the type of resource you want to promote. You can choose from any type of resources stored in a library or specify that you want any resource that meets the other criteria.
Version	You can specify that you want Studio to include all versions or just the last version.

Field	Description
Revision	You can specify that you want Studio to include all revisions, just the last revision, or only expired (EXP) revisions.
Mode	Select which mode you want Studio to look for when selecting resources to promote. For more information, see <i>Defining Mode, Status, Class, and Project Options on page 380</i> .
Status	Select which status you want Studio to look for when selecting resources to promote. For more information, see <i>Defining Mode, Status, Class, and Project Options on page 380</i> .
Class	Select which class you want Studio to look for when selecting resources to promote. For more information, see <i>Defining Mode, Status, Class, and Project Options on page 380</i> .
Project	Select which project ID you want Studio to look for when selecting resources to promote. For more information, see <i>Defining Mode, Status, Class, and Project Options on page 380</i> .

Changing the Mode, Status, Class, and Project values in the source library

In addition, you can have Studio change the Mode, Status, Class, and Project values for resources in the source library *after* they are successfully promoted. For instance, you can tell Studio to select resources with a Mode of DEV, promote them, then change the Mode value in the source library to TEST. You do this in the Source Library fields, as shown here:

Scroll down to define the final values in the source library

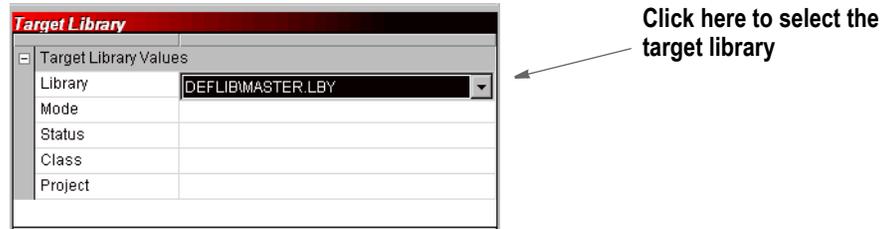


Field	Description
Mode	Select which mode you want Studio to assign after it moves a copy of the resource into the target library. For more information, see <i>Defining Mode, Status, Class, and Project Options on page 380</i> .
Status	Select which status you want Studio to assign after it moves a copy of the resource into the target library. For more information, see <i>Defining Mode, Status, Class, and Project Options on page 380</i> .
Class	Select which class you want Studio to assign after it moves a copy of the resource into the target library. For more information, see <i>Defining Mode, Status, Class, and Project Options on page 380</i> .
Project	Select which project ID you want Studio to assign after it moves a copy of the resource into the target library. For more information, see <i>Defining Mode, Status, Class, and Project Options on page 380</i> .

If you do not want these values to change, simply ignore the Source Final Value fields.

Defining the target library

Next, you must define the target library. Click in the Library field to select the library to which you will promote resources.



After you identify the target library, you can tell Studio the values you want assigned to the resources it will promote after it moves those resources into the target library.

Field	Description
Mode	Select which mode you want Studio to assign after it moves the resource into the target library. For more information, see <i>Defining Mode, Status, Class, and Project Options on page 380</i> .
Status	Select which status you want Studio to assign after it moves the resource into the target library. For more information, see <i>Defining Mode, Status, Class, and Project Options on page 380</i> .
Class	Select which class you want Studio to assign after it moves the resource into the target library. For more information, see <i>Defining Mode, Status, Class, and Project Options on page 380</i> .
Project	Select which project ID you want Studio to assign after it moves the resource into the target library. For more information, see <i>Defining Mode, Status, Class, and Project Options on page 380</i> .

If you do not want these values to change, simply ignore these Target Library fields.

Using a promotion script

You can save your resource promotion settings and reuse them. To save your settings, enter a name for the script in the Script File field and click Save Script. Studio appends an LSC extension onto the name you entered and stores the script in the DEFLIB directory.

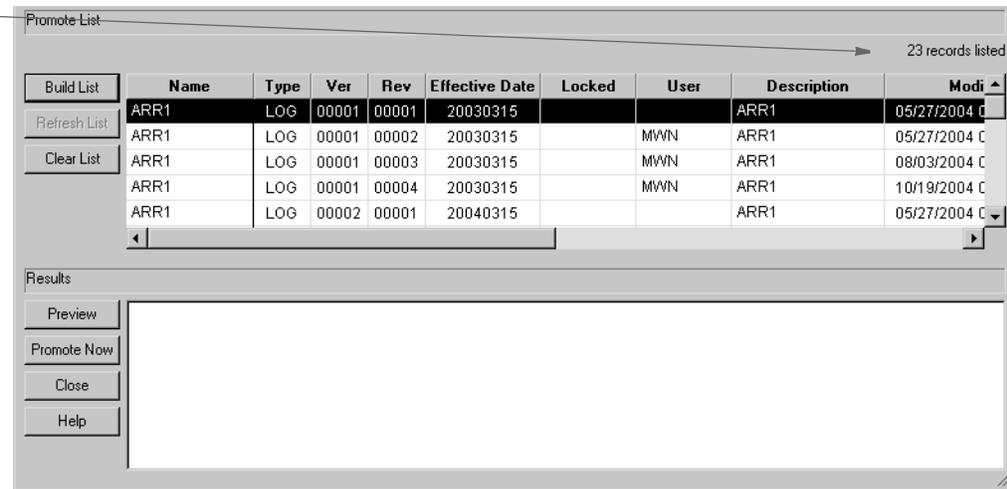


To open a script, click Open Script and choose the script you want to run or edit. You can clear the settings from a script by clicking Clear Script.

Performing the Promotion

Once you define the promotion settings, click Build List to have Studio create a list of the resources that qualify for promotion.

Studio tells you how many resources it selected and lets you scroll through the list



Inspect the resources shown in the list. If necessary, make changes to your selection criteria and build the list again. When you have the list as you want it, click Preview. Studio runs a test promotion and shows you the results.

Note At this point, no changes have been made. Preview simply shows you what the result of your promotion script would be if you actually ran it.

If necessary, make changes to your selection criteria and preview the promotion again. When you are satisfied with the results, click Promote Now to promote the resources.

FILTERING RESOURCES

To make it easier to work with resources, you can apply a filter to select a subset of your resources. For instance, you can apply a filter to see only FAP files (sections) or only graphics with a given effective date. The filter tab is available on the Open File window and in the Library view.

To create a filter, click the Filter tab:

Field	Description
Name	Enter the name or partial name of a resource or group of resources. You can use wildcards in your entry. For instance, <i>bil*.fap</i> will give you all the FAP files with names beginning with <i>bil</i> .
Type	Select the type of file you want to filter on. You can choose from all types of files stored in the library.
Version	Choose either all or last to see all versions or only the last version.
Revision	Choose all, last, or EXP to see all revisions, only the last revision, or only expired revisions
Effective Date	Check this field and then select the effective date you want to filter on.
Locked	Choose from yes, no, or any to see only locked files, only files that are not locked, or any file.
User ID	Enter a user ID to see only files assigned to that user.
Mode	Select which mode you want to filter on.
Status	Select which status you want to filter on.
Class	Select which class you want to filter on.
Project	Select which project ID you want to filter on.
Script Name	Enter a name for the filter script you are creating or enter the name of the script you want to retrieve.

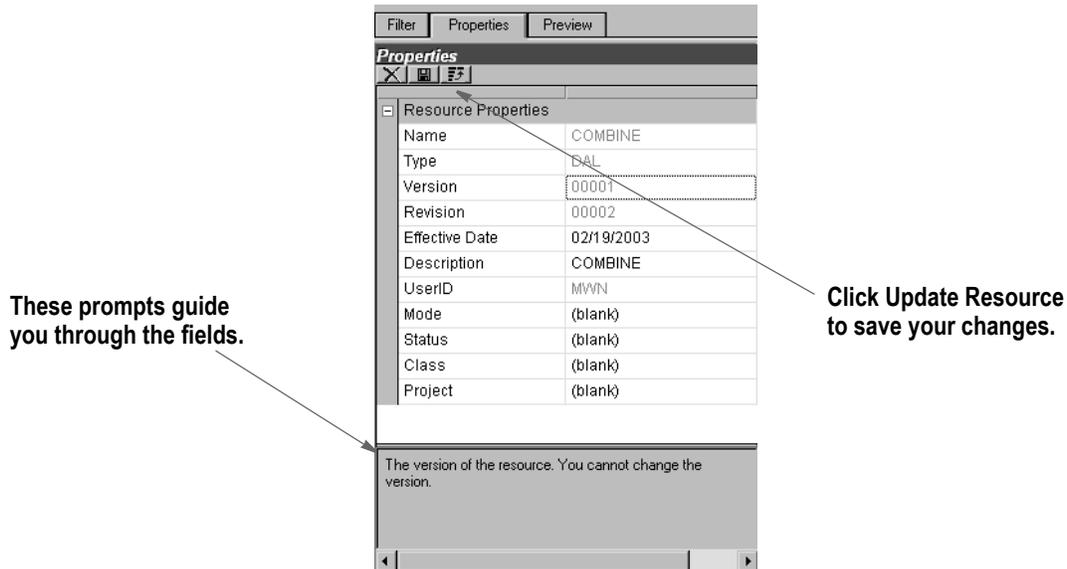
Click Save Script to save this filter script so you can reuse it.

Note If you save a filter script, Studio remembers the script name and tries to use that script the next time you open the Open File window or the Library view. To prevent Studio from remembering the script and trying to use it the next time, remove the script name before you exit this window.

EDITING RESOURCE INFORMATION

The steps below tell you how to edit the Description, Effective Date, Mode, Status, Class, and Project fields for a resource.

1. From the Library view, highlight the resource whose properties you want to modify and click the Properties tab.



2. Make your changes. You cannot edit some fields, such as the version number. This table discusses the fields you can edit:

Field	Description
Effective Date	All revisions of a specific version of a resource must have the same effective date. The effective date for a resource must equal or fall after the effective date for the prior version of that resource. Studio prevents you from entering a prior date. The effective date defaults to MM/DD/YYYY, but may have a different format depending on your locale setting.
Description	You can enter up to 100 characters to indicate what type of change was made to the resource. This field defaults to the contents of the Description field of the resource checked out, but you can enter a different description if you like. Your entry in the Description field appears in the grid displayed on the Open File window and in the Library view.
Mode	Use this field to assign a mode to the resource as it is checked in. For instance, you can use the Mode field to specify where in the development cycle the resource is. For more information, see <i>Defining Mode, Status, Class, and Project Options</i> on page 380.
Status	Use this field to assign a status to the resource as it is checked in. For instance, you could use the Status field to indicate whether a resource has passed or failed testing. For more information, see <i>Defining Mode, Status, Class, and Project Options</i> on page 380.

Field	Description
Class	Use this field to assign a class to the resource as it is checked in. For instance, you could use the Class field to indicate the market in which a resource was applicable. For more information, see <i>Defining Mode, Status, Class, and Project Options on page 380</i> .
Project	Use this field to assign a project ID to the resource as it is checked in. For instance, you could use the Project field to indicate which project a resource was associated with. For more information, see <i>Defining Mode, Status, Class, and Project Options on page 380</i> .

3. Click the Update Resource icon on the toolbar to record your changes.

DELETING RESOURCES

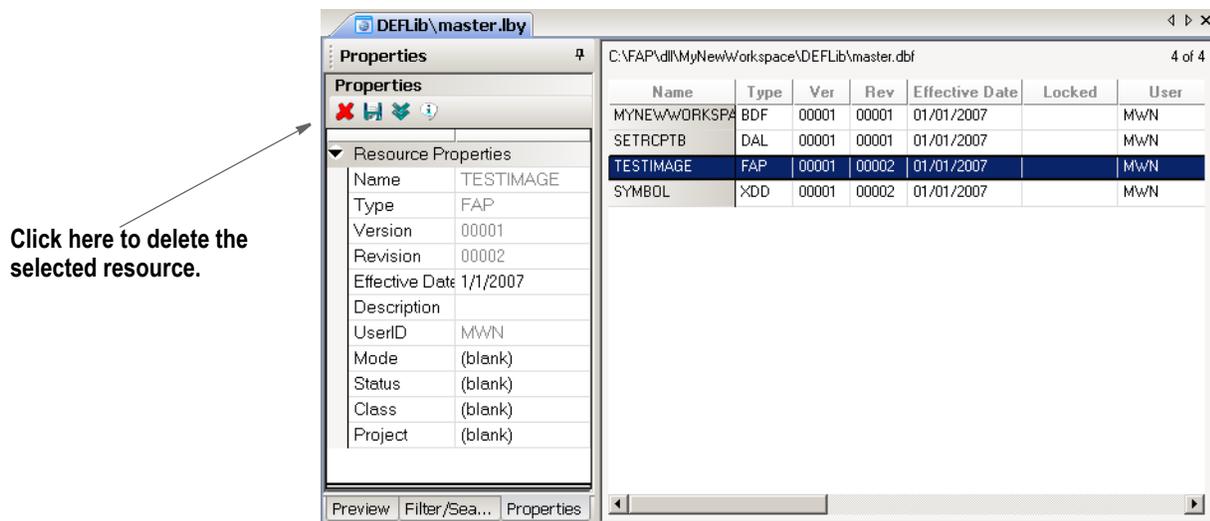
You must be a library or system administrator or have been assigned the specific right to delete library resources. If you do not have the appropriate rights, the red X icon shown below will not be enabled.

In all but very rare circumstances, if you have ever used the resource, you should not delete it. For example, suppose a form was used when a policy was archived and you later delete that form. If you later try to retrieve that form from archive, you will encounter problems because the system will not be able to load the form you deleted.

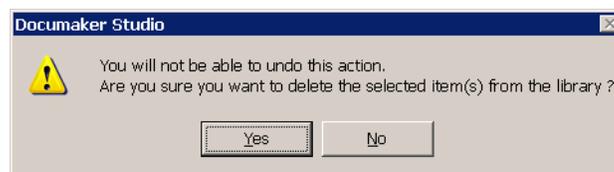
If you delete a resource from the library while it is checked out, you cannot later check in the resource. If you try, Studio tells you the resource has not been checked out. You would then have to close the resource and delete it from disk.

Note When you delete a resource from the library, Studio does not remove the file from disk if you had it checked-out or extracted a copy there. Studio merely removes the resource from the library. Once removed, you cannot undo this action.

To delete a resource from Library Manager, first select the resource. Then, on the Properties tab for the resource, click the red X, as shown below.



Studio warns you that you cannot undo this action and lets you confirm the deletion.



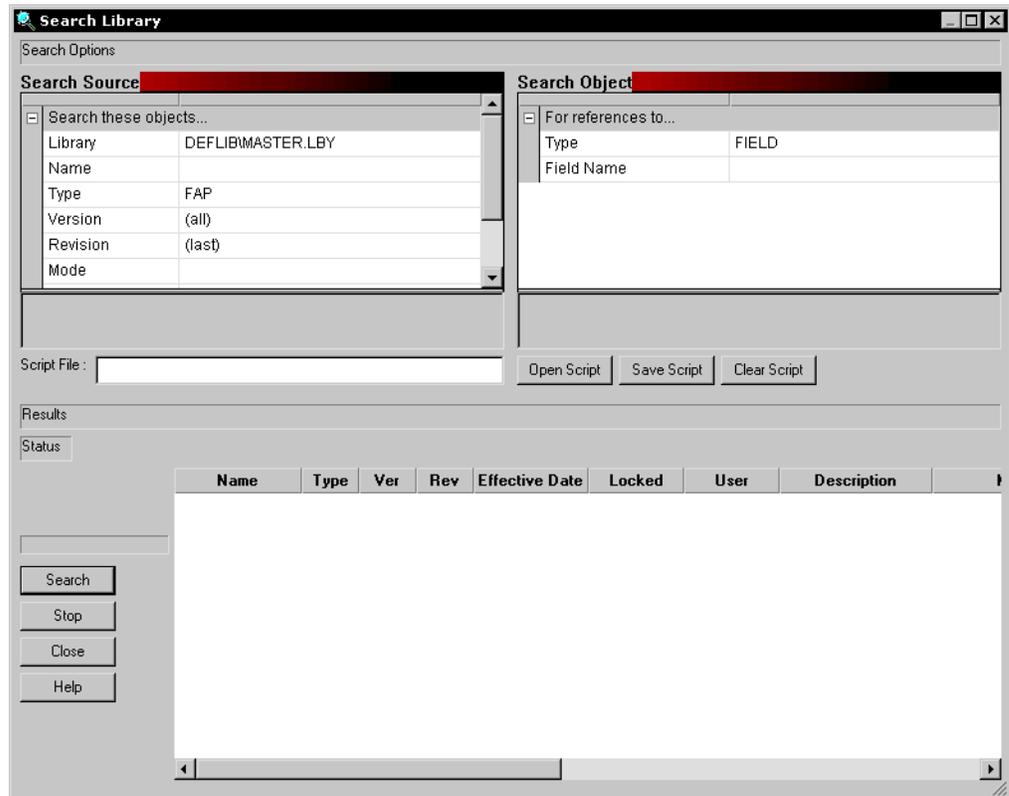
Click Yes to delete the resource you selected. Studio then deletes the resource from the library and from your hard disk.

Note There is a Delete option on the pop-up menu for the Workspace window and a Delete button on the Workspace List window. These options delete a file from disk and do not affect the library storage of the selected item — assuming it came from the library. Unlike deleting a resource from the library, you do not have to have special rights to use these Delete options.

SEARCHING THE LIBRARY

You can use the Library, Search option to identify resources that reference (or contain) other resources or resource elements. For instance, since FAP files can contain references to graphics (LOG resources), variable fields, rule names, and text labels, you can use this option to search the library for FAP files that contain a given field name or that use a certain rule.

When you choose the Library, Search option, the Search Library window appears.



Defining the resources

Use the Search Source fields to identify the resources you want to search. You must limit your search to a specific type of resource.

Field	Description
Library	Select the library in which the resources are stored.
Name	Enter the name or partial name of a resource or group of resources. You can use wildcards in your entry. For instance, specifying a type of FAP and a name value of <i>bil*</i> tells Studio to search all FAP files with names that begin with the letters <i>bil</i> .
Type	Select the type of resource you want to filter on. You can choose from these types of resources: BDF, GRP, FOR, and FAP.
Version	Choose either all or last to see all versions or only the last version.
Revision	Choose all, last, or EXP to see all revisions, only the last revision, or only expired revisions

Field	Description
Mode	Use this field to tell Studio to include only those resources assigned the mode you specify.
Status	Use this field to tell Studio to include only those resources assigned the status you specify.
Class	Use this field to tell Studio to include only those resources assigned the class you specify.
Project	Use this field to tell Studio to include only those resources assigned the project ID you specify.

Defining the objects

Use the Search Objects fields to identify specific objects within the resources you want to search for. For instance, these fields let you search for a specific field on FAP files or a specific group within business unit files.

The fields change, depending on what you select in the Type field in the Search Source fields. For instance, if you chose FAP files in the Type field, you see these fields:

Field	Description
Type	You can choose from Field, Text, Font, Log, Rule (section), Rule (field). Your selection tells Studio to search for this type of object in the FAP files selected using the criteria you specified on the Search Source fields.
Name	The name of this field changes, depending on what you chose in the Type field. For instance, if you chose Field in the Type field, the name of this field is Field Name. If you chose Section Rule, the name will be Section Rule Name. Enter the name of the field, the font ID, the graphic, the section rule, or the field rule. If you chose Text in the Type field, the following fields appear:

Field	Description
Text Element Name	Enter the name of the text element you want to search for. If you leave this field blank, Studio identifies all text elements.

Field	Description
Text String	Enter the text string you want to search for. If you leave this field blank, Studio identifies all text elements that match the Text Element Name field. If you left the Text Element Name field blank, Studio searches for text elements with any name.
Text Case	Choose Insensitive if you do not want Studio to consider the case when reaching. Choose Sensitive if you do want the case considered.

Using a search script

You can save your search settings and reuse them. To save your settings, enter a name for the script in the Script File field and click Save Script. Studio appends an LSC extension onto the name you entered and stores the script in the DEFLIB directory.

A screenshot of a software interface showing a text input field labeled "Script File:" followed by three buttons: "Open Script", "Save Script", and "Clear Script". The input field is currently empty.

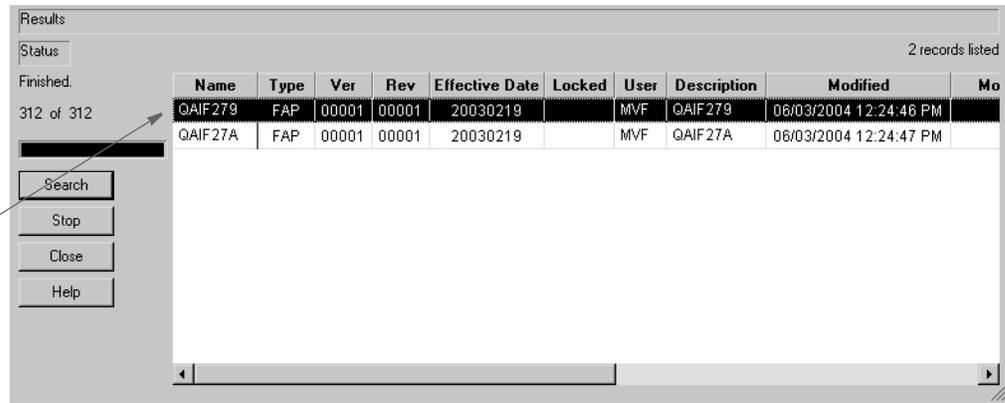
To open a script, click Open Script and choose the script you want to run or edit. You can clear the settings from a script by clicking Clear Script.

Performing the Search

Once you define the search settings, click Search to have Studio create a list of the resources it found based on the criteria you defined.

Click here to start the search

The results appear here



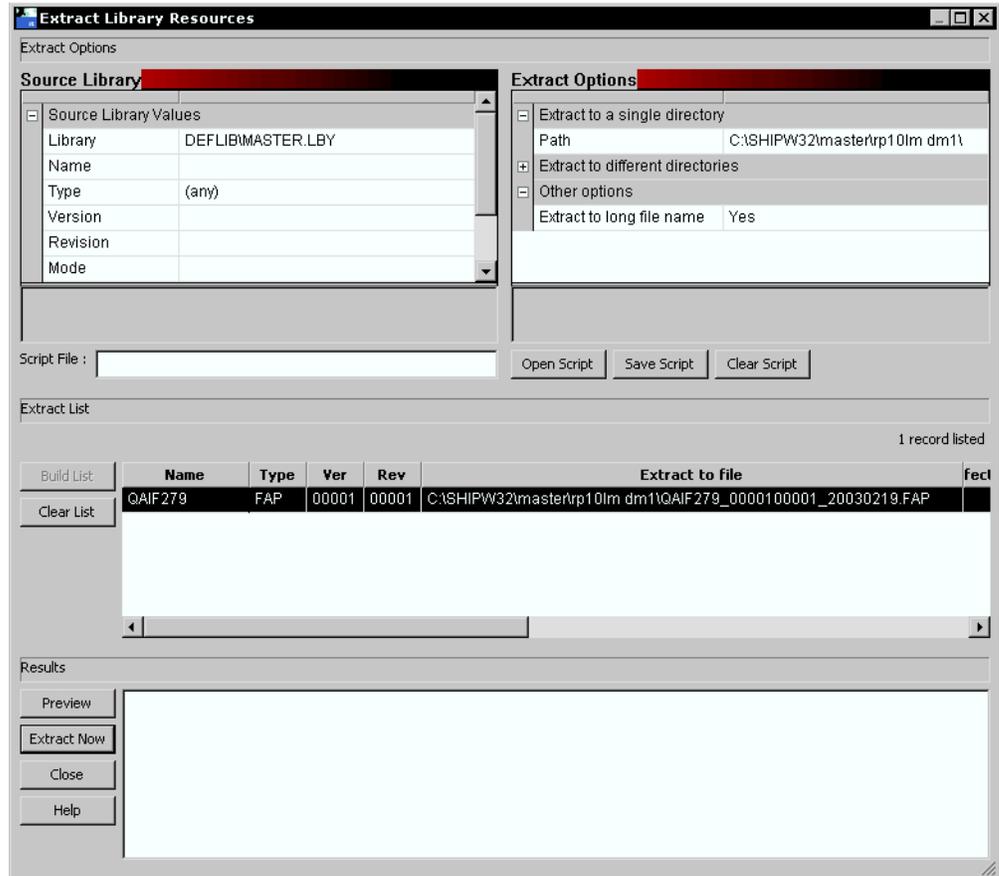
You can highlight on any selected resource, right click, and perform these tasks:

To	Select this option
Check out the resource	Check Out. See <i>Checking Out Resources on page 354</i> for more information.
Open the resource without checking it out	Read
See the history of changes to the resource	History. See <i>Reviewing a History of Resource Changes on page 377</i> for more information.
Extract the resource from the library	Extract. See <i>Extracting Resources on page 372</i> for more information.
Promote the resource	Promote. See <i>Promoting Resources on page 359</i> for more information.
Expire the resource	Expire. See <i>Expiring Resources on page 376</i> for more information.
Print a list of the resources found	Print this Window

EXTRACTING RESOURCES

You can extract a resource from a library using the Library, Extract option. You can also highlight the resource on the Library View or from a list of resources found via a search, right click, and select the Extract option.

The Extract Library Resources window appears.



Identifying the resources to extract

On the Extract Library Resources window, you use the Source Library options to tell Studio which resources you want to extract.

Field	Description
Library	Select the name of the library that contains the resources you want to extract.
Name	Enter the name of the resource you want to extract. You can use wildcards and enter a partial name to have Studio promote a group of similarly named resources. For instance, <i>brt*</i> tells Studio to select all resources with names that start with <i>brt</i> .
Type	Select the type of resource you want to extract. You can choose from any type of resources stored in a library or specify that you want any resource that meets the other criteria.
Version	You can specify that you want Studio to include all versions or just the last version.
Revision	You can specify that you want Studio to include all revisions, just the last revision, or only expired (EXP) revisions.

Field	Description
Mode	Select which mode you want Studio to look for when selecting resources to extract.
Status	Select which status you want Studio to look for when selecting resources to extract.
Class	Select which class you want Studio to look for when selecting resources to extract.
Project	Select which project ID you want Studio to look for when selecting resources to extract.

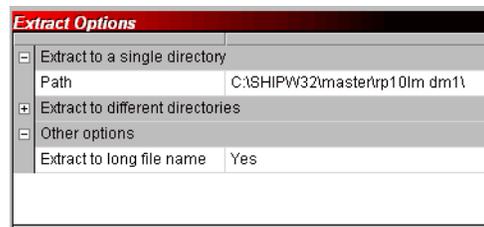
Defining extract options

Use the Extract Options fields to tell Studio which directory or directories to extract the resources to. By default, Studio extracts the resources to the location you identified using the Path field. You can change this location by clicking on the field and navigating to a different location.

If you are extracting several types of resources and you want each resource type to be extracted to a specific directory, you can specify the directories by expanding the Extract to Different Directories item and specifying a directory for each type.

By default, the extracted resources are written to disk with their long file name. The long file name is constructed by taking the resource name, appending an underscore, appending the version and revision (five digits each), followed by another underscore, followed by the effective date (in YYYYMMDD format) and followed by a file extension for the specific resource type, such as BDF or GRP. If you want to extract to the standard name of the resource, select No in the Extract to Long File Name field.

Click here to define a directory for each type of resource.



Enter a path to define a single directory into which the extracted resources will be copied.

Field	Description
Path	Enter a path to extract the selected resources to a specific path.
Extract to different directories	Click to display fields into which you can define individual directories for each type of resource.
Extract to long file name	Select Yes to use long file names. Select No to use the 8.3 file naming convention on extracted resources. Long file names are created by taking the resource name, appending an underscore followed by the version and revision, then appending another underscore followed by the date in YYYYMMDD format, and then adding an extension to indicate the resource type. Here is an example: FileName_0000100001_20041201.fap

Using an extraction script

You can save your resource extraction settings and reuse them. To save your settings, enter a name for the script in the Script File field and click Save Script. Studio appends an LSC extension onto the name you entered and stores the script in the DEFLIB directory.



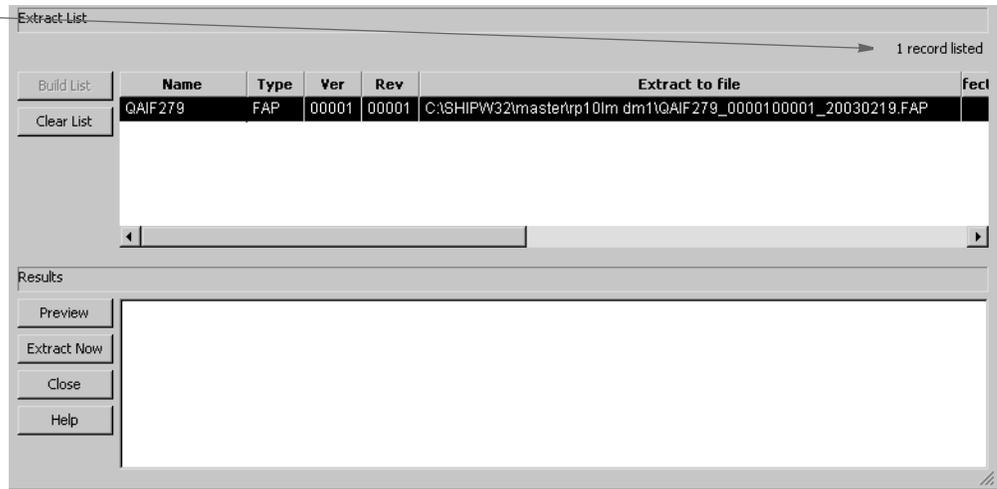
The image shows a user interface for managing extraction scripts. It consists of a horizontal bar with a text input field on the left labeled 'Script File:' and three buttons on the right: 'Open Script', 'Save Script', and 'Clear Script'. The buttons are arranged in a row and are separated by small gaps.

To open a script, click Open Script and choose the script you want to run or edit. You can clear the settings from a script by clicking Clear Script.

Performing the Extraction

Once you define the extraction settings, click Build List to have Studio create a list of the resources that qualify for extraction.

Studio tells you how many resources it selected and lets you scroll through the list.



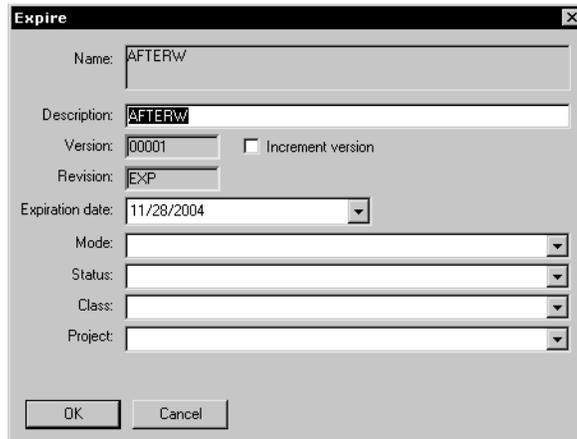
Inspect the resources shown in the list. If necessary, make changes to your selection criteria and build the list again. When you have the list as you want it, click Preview. Studio runs a test extraction and shows you the results.

Note At this point, no changes have been made. Preview simply shows you what the result of your extraction script would be if you actually ran it.

If necessary, make changes to your selection criteria and preview the extraction again. When you are satisfied with the results, click Extract Now to extract the resources.

EXPIRING RESOURCES

You can expire a resource you no longer wish to maintain. In essence, expiring a resource is like retiring a resource. The resource remains in the library, but has *EXP* as its revision number. To expire a resource, highlight the resource on the Library view or from a search list and right click. Then choose the Expire option. The Expire window appears.



The screenshot shows the 'Expire' dialog box with the following fields and values:

- Name: AFTERW
- Description: AFTERW
- Version: 00001 Increment version
- Revision: EXP
- Expiration date: 11/28/2004
- Mode: [Dropdown]
- Status: [Dropdown]
- Class: [Dropdown]
- Project: [Dropdown]

Buttons: OK, Cancel

Expiring a resource tells Studio to check into the library another revision of the resource, but with *EXP* as the value of the Revision field.

By default, the expiration date is set to today's date but you can change this date if necessary. The expiration date you specify must be greater than or equal to the effective date of other revisions within this version and must be less than or equal to the effective date of the next version of the resource, if one exists.

On the Expire window, you can also enter a description and set the Mode, Status, Class, and Project fields. Once you are satisfied with the values, click Ok to expire the resource.

REVIEWING A HISTORY OF RESOURCE CHANGES

To see what changes have occurred to the resources in your library, choose the Library, History option. The Library Log window appears.

Name	Action	Library	Date	Time	Type	Ver	Rev
PHONES	CHECKED OUT	MASTER.LBY	20041019	16:59:44	LOG	00001	00001
CATWOMAN	UNLOCKED	MASTER.LBY	20041019	16:59:25	LOG	00001	00001
JPG	UNLOCKED	MASTER.LBY	20041019	16:59:19	LOG	00001	00001
JPG	CHECKED OUT	MASTER.LBY	20041019	16:59:00	LOG	00001	00001
CATWOMAN	CHECKED OUT	MASTER.LBY	20041019	16:58:34	LOG	00001	00001
CHECK	CHECKED IN	MASTER.LBY	20041019	16:58:28	LOG	00001	00002
BULB	CHECKED IN	MASTER.LBY	20041019	16:58:18	LOG	00001	00002
ARR1	CHECKED IN	MASTER.LBY	20041019	15:01:29	LOG	00002	00004
ARR1	CHECKED IN	MASTER.LBY	20041019	15:01:21	LOG	00001	00004
ARR1	CHECKED OUT	MASTER.LBY	20041019	15:01:12	LOG	00001	00003
CATWOMAN	UNLOCKED	MASTER.LBY	20041007	12:57:56	LOG	00001	00001
CATWOMAN	CHECKED OUT	MASTER.LBY	20041007	12:55:26	LOG	00001	00001
CATWOMAN	UNLOCKED	MASTER.LBY	20041007	12:55:10	LOG	00001	00001
CATWOMAN	CHECKED OUT	MASTER.LBY	20041007	12:55:04	LOG	00001	00001
CATWOMAN	UNLOCKED	MASTER.LBY	20041007	12:55:00	LOG	00001	00001
CATWOMAN	CHECKED OUT	MASTER.LBY	20041007	12:54:29	LOG	00001	00001
CATWOMAN	CHECKED IN	MASTER.LBY	20041007	12:53:31	LOG	00001	00001
RP10LMDM1	CHECKED OUT	MASTER.LBY	20041004	15:33:03	BDF	00001	00001

Studio shows you a log of all changes to all resources. You can filter this list by assigning filter values.

For instance, you can use a filter to see only FAP files which have been checked in by a specific user. Here are the fields you can define to create a filter:

Field	Description
Name	Enter the name or partial name of a resource or group of resources. You can use wildcards in your entry. For instance, specifying a type of FAP and a name of <i>bil*</i> tells Studio to display the log entries for all FAP files whose names begin with the letters <i>bil</i> .
Action	Choose from these actions: any, added, checked in, checked out, deleted, promoted from after, promoted from before, promoted to, unlocked, updated from, or updated to.
Library	Select the library in which the resources are stored.
Type	Select the type of file you want to filter on. You can choose from all types of files stored in the library.

Field	Description
User ID	Enter the user ID for the user on whom you want to filter.
Mode	Select which mode you want to filter on.
Status	Select which status you want to filter on.
Class	Select which class you want to filter on.
Project	Select which project ID you want to filter on.

Click Filter Now to apply the filter.

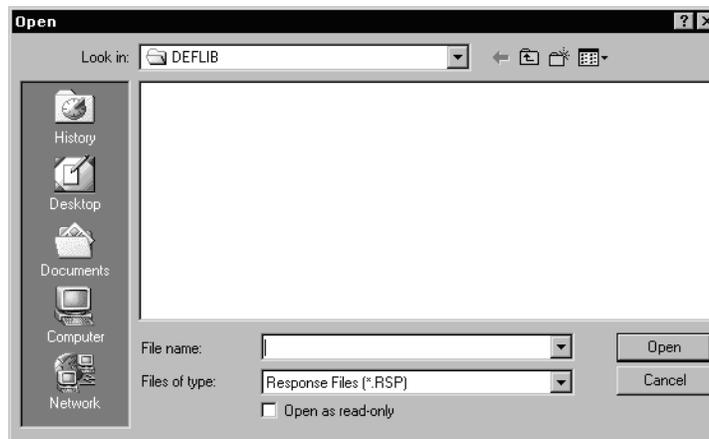
RUNNING RESPONSE FILES

A response file is a file created by the LBRYMGR utility which contains commands and data. These commands and data are then read and processed accordingly. There are two ways to process a response file:

- Using the LBRYMGR utility (see the [Docutoolbox Reference](#) for more information)
- Using Studio

To read and process a response file using Studio, follow these steps:

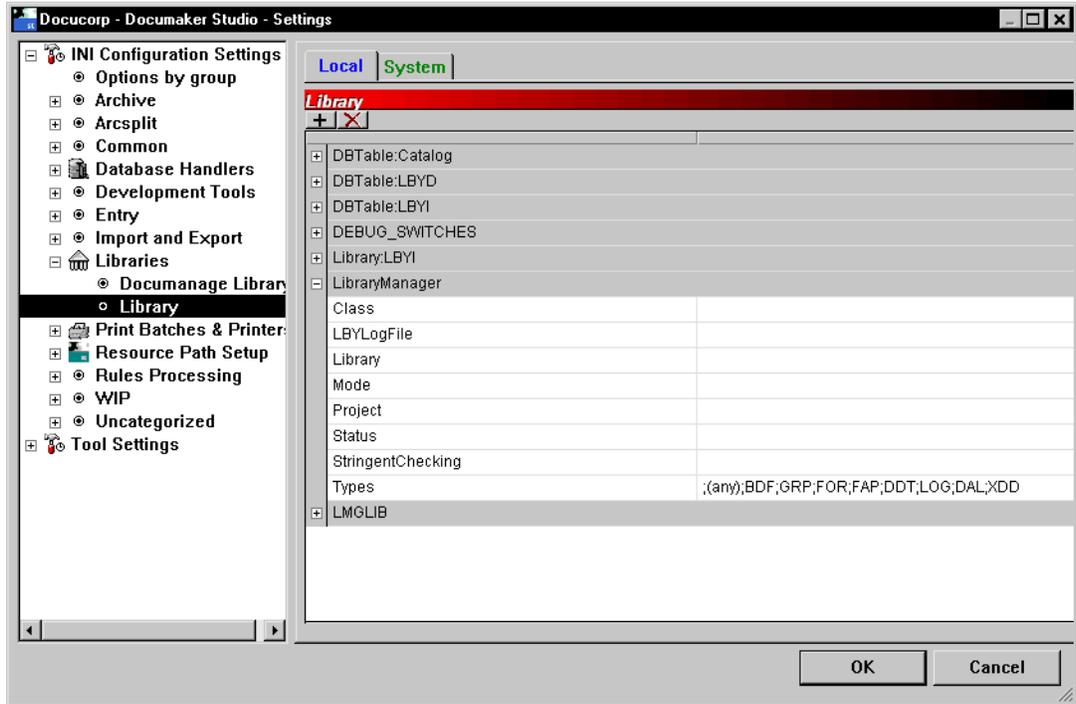
1. Choose the Library, Run Response File option. The Open window appears from which you can select the response file you want to run.



2. Select the response file to process and click Ok. Studio displays messages to show you its progress.

DEFINING MODE, STATUS, CLASS, AND PROJECT OPTIONS

Use the Manage, Settings option to define the Mode, Status, Class, and Project options users choose from when checking in a resource. These options are grouped under Library manager, as shown here:



You can have multiple modes, classes, statuses, and projects.

Option	Description
Mode	<p>Use this option to set up the modes users will select from in the Mode field as they edit resource information. For instance, you can use the Mode field to specify where in the development cycle the resource is. You can set up multiple options, as shown here:</p> <pre>Mode = DEV;Development Mode = TEST;Testing Mode = PROD;Production</pre> <p>These entries set up these modes: DEV, TEST, and PROD. The text after the semicolon is an optional description.</p>
Status	<p>Use this option to set up the statuses users will select from in the Status field as they edit resource information. For instance, you could use the Status field to indicate whether a resource has passed or failed testing. You can set up multiple statuses, as shown here:</p> <pre>Status = TEST;Test Status = PASSED;Passed testing Status = FAILED;Failed testing</pre> <p>These entries set up these statuses: TEST, PASSED, and FAILED. The text after the semicolon is an optional description.</p>

Option	Description
Class	<p>Use this option to set up the classes users will select from in the Class field as they edit resource information. For instance, you could use the Class field to indicate the market in which a resource was applicable. You can set up multiple classes, as shown here:</p> <pre>Class = GA;Georgia resource Class = TX;Texas resource Class = MD;Maryland resource</pre> <p>These entries set up these classes: GA, TX, and MD. The text after the semicolon is an optional description.</p>
Project	<p>Use this option to set up the projects users will select from in the Project field as they edit resource information. For instance, you could use the Project field to indicate which project a resource was associated with. You can set up multiple projects, as shown here:</p> <pre>Project = P001;Project 001 Project = P002;Project 002 Project = P003;Project 003</pre> <p>These entries set up these projects: P001, P002, and P003. The text after the semicolon is an optional description.</p>

USING THE LBYPROC UTILITY

The LBYPROC utility processes library scripts. Library scripts are XML-based files that let you perform actions on a resource library. You can use these scripts for...

- Adding resources to a library
- Deleting resources from a library
- Extracting resources from a library (writing the contents to a disk file)
- Promoting resources from one library to another
- Searching a library for specific elements
- Producing a list of resources that match a designated set of filter values

The scripts are designed so a script can cause the indicated action to be performed on more than one resource. For example, a single Promote script can cause many resources to be promoted from one library to another and an Extract script can cause many resources in a library to be extracted from the library and written to disk.

Note For more information on the LBYPROC utility, see the [Docutoolbox Reference](#).

Here are some example scripts:

```
<LBYSCRIPT>
<FILTER>
<LIBRARY VALUE=". \DEFLIB\MASTER.LBY"/>
<NAME VALUE=""/>
<TYPE VALUE=""/>
<MODE VALUE=""/>
<STATUS VALUE=""/>
<CLASS VALUE=""/>
<PROJECT VALUE=""/>
<DESC VALUE=""/>
<VERSION VALUE=""/>
<REVISION VALUE="last"/>
<USERID VALUE=""/>
<EFFDATE VALUE=""/>
</FILTER>

<PROMOTE>
<LIBRARY SRC=". \deflib\MASTER.LBY" TGT=". \deflib\V.LBY"/>
<NAME VALUE=""/>
<TYPE VALUE=""/>
<VER VALUE=""/>
<REV VALUE=""/>
<USERID VALUE=""/>
<MODE SRC="" TGT=""/>
<STATUS SRC="" TGT="*"/>
<CLASS SRC="" TGT="*"/>
<PROJECT SRC="" TGT="*"/>
</PROMOTE>

<ADD>
<LIBRARY VALUE=". \DEFLIB\MASTER.LBY"/>
<FILENAME VALUE=". \forms\Q1ADDR.fap"/>
<NAME VALUE="Q1ADDR"/>
<DESC VALUE="Added this FAP using LBYPROC"/>
<TYPE VALUE="FAP"/>
<SUBTYPE VALUE=""/>
```

```
<EFFDATE VALUE="20030701"/>
<VER VALUE="00001"/>
<REV VALUE="00001"/>
<MODE VALUE=""/>
<STATUS VALUE=""/>
<CLASS VALUE=""/>
<PROJECT VALUE=""/>
</ADD>

<SEARCH>
<LIBRARY VALUE=". \DEFLIB\MASTER.LBY"/>
<NAME VALUE=""/>
<TYPE VALUE="FAP"/>
<MODE VALUE=""/>
<STATUS VALUE=""/>
<CLASS VALUE=""/>
<PROJECT VALUE=""/>
<OBJECTTYPE VALUE="LOG"/>
<OBJECTNAME VALUE=""/>
<OBJECTTEXT VALUE=""/>
<OBJECTTEXTCASE VALUE=""/>
</SEARCH>

<EXTRACT>
<LIBRARY VALUE="DEFLIB\MASTER.LBY"/>
<NAME VALUE="" TGTNAME="LONG"/>
<ALLLIB VALUE=". \EXT\"/>
<BDFLIB VALUE=". \EXT\BDFLIB\"/>
<GRPLIB VALUE=". \EXT\GRPLIB\"/>
<FORLIB VALUE=". \EXT\FORLIB\"/>
<FAPLIB VALUE=". \EXT\FAPLIB\"/>
<DDTLIB VALUE=". \EXT\DDTLIB\"/>
<LOGLIB VALUE=". \EXT\FAPLIB\"/>
<DALLIB VALUE=". \EXT\DEFLIB\"/>
<TYPE VALUE=""/>
<VERSION VALUE=""/>
<REVISION VALUE=""/>
<MODE VALUE=""/>
<STATUS VALUE=""/>
<CLASS VALUE=""/>
<PROJECT VALUE=""/>
</EXTRACT>

<DELETE>
<LIBRARY VALUE=". \DEFLIB\W.LBY"/>
<NAME VALUE=""/>
<TYPE VALUE="FOR"/>
<MODE VALUE=""/>
<STATUS VALUE=""/>
<CLASS VALUE=""/>
<PROJECT VALUE=""/>
</DELETE>

</LBYSRIPT>
```

TROUBLESHOOTING

To help you resolve problems, this topic discusses how to turn on tracing and various error messages you may encounter.

TURNING ON TRACING

If you feel Studio is not retrieving the correct version/revision of a resource, or if you are experiencing other problems that seem to be related to the use of a library, you can use INI options to create a trace file.

You can then use the trace file to determine the date of the resource you requested and the effective date of the resource returned by Studio, as well as other library-related information. To turn on tracing, specify these options:

```
< Debug_Switches >
  Enable_Debug_Options= Yes
  LbyLib                = Yes
```

This table shows you where to place the options and the default file name.

For	Specify the options in	Default file name
Documaker (GenData)	FSIUSER.INI or FSISYS.INI	trace
Documaker Workstation	FSIUSER.INI or FSISYS.INI	trace
Docucreate	FAPCOMP.INI	trace
Docupresentment (IDS)	DAP.INI or RPEX1.INI *	dprtrc.log

* RPEX1.INI is commonly-used as the name of the configuration INI file. If the name you use differs, substitute that name.

For Documaker, Documaker Workstation, and Docupresentment, you can specify the name of the trace file using this option:

```
< Data >
  TraceFile =
```

Note Turning on the tracing causes a large amount of data to be written to the trace file. For optimal performance, only do this when you are testing or troubleshooting an implementation.

You may also want to retain the NAFILE.DAT and the POLFILE.DAT files. During retrieval, these files are retrieved from archive and are initially written to disk then later deleted. To keep these files on disk during retrieval from Documaker Workstation, include this option:

```
< ArcRet >
  KeepFiles = Yes
```

To keep the NAFILE and POLFILE on disk during retrieval from Docupresentment (IDS), include this option:

```
< Attachments >
  Debug = Yes
```

Note For optimal performance, only turn on the KeepFiles or Debug option when you are testing or troubleshooting a system.

HANDLING ERROR MESSAGES

Here are some of the error messages you may encounter:

ORA-00904

The following error message:

```
[Oracle] [ODBC] [Ora]ORA-00904: invalid column name
```

indicates one of the column names in the table is invalid. To determine which table is invalid, turn on LBYLIB tracing (See *Turning on Tracing on page 384*), recreate the problem, then examine the trace file and look for the ORA-00904 error message. The table referenced immediately before the ORA-00904 message is the one with the invalid column.

Note there are column names reserved by Oracle which cannot be used. To remap a column name use the ODBC_FieldConvert control group as described in a previous section. If you are setting up the tables for the first time, after re-mapping the column, you may need to remove the existing table so it can be recreated with the correct column names.

This message can also indicate you are trying to reference a column in the table that does not exist. Look closely at the trace file and compare the columns referenced in the SQL statement, such as SELECT, to the columns of the table as it is defined in the database.

ORA-01401

The following error message:

```
[Oracle] [ODBC] [Ora]ORA-01401: inserted value too large for column
```

indicates you are trying to insert a value that is too large for the column. To try to determine which table the problem is with, turn on LBYLIB tracing (See *Turning on Tracing on page 384*), recreate the problem, then examine the trace file and look for the ORA-01401 error message. The table referenced immediately before the ORA-01401 message should be the table that contains the column with the problem.

SQL0104N

The following DB2 ODBC error message:

```
42601 -104 [IBM] [CLI Driver] [DB2/NT] SQL0104N An unexpected token ")"  
was found following "on DAP110_LBYI_R1 ("  
Expected tokens may include: "<index_col_list>". SQLSTATE=42601
```

indicates that the DB2 driver attempted to create an index for the table. When using the DB2 ODBC driver, you should always set the following INI option to No.

```
< DBHandler:ODBC >  
    CreateIndex = No
```


Chapter 10

Using Dictionaries

Use the Dictionaries option to work with the fields, rule files, and extract files you will use in your implementation. This topic discusses the following topics:

- *Working with the Extract Dictionary on page 388*
- *Sorting XDD Elements on page 390*
- *Importing XML Schemas into the XDD on page 391*
- *Importing XML Extract Files into the XDD on page 394*
- *Using Addressee Records on page 397*
- *Using the Common Fields Dictionary on page 400*
- *Importing CSV Files on page 401*
- *Importing TGA Files on page 402*

WORKING WITH THE EXTRACT DICTIONARY

The extract dictionary (also known as the SYMBOL.XDD) lets you store information about extract file records, as well as fields within records, in a database file format. Each record in this database contains all of the information that appears in the rule section for a variable field. Other information is also stored in the XDD. You can use the extract dictionary in both the field mapping process and the triggering process. This topic discusses the creation of and the use of the XDD in the mapping process.

Use of the XDD 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 variable field on a section, all that is required is that the rule XDD be specified for the field rule.

Typically, there is one (see note below) SYMBOL.XDD file per resource library. It is stored by default in the DEFLIB directory, under the specified workspace. The XDD is used with the rule section of the section at rules processing time. The link to the XDD database from the rule section of the section is the use of the XDD rule.

When the rules processor runs, items for the XDD are stored in cache. A thousand entries is the default cache number. There are INI options to change this number and to control the order of fields being removed (less frequent or less recent) from cache. Use the extract dictionaries option to build the SYMBOL.XDD file.

The XDD file is laid out in a parent-child structure. The top level, parents, are individual records in the extract file. Fields within the record are children.

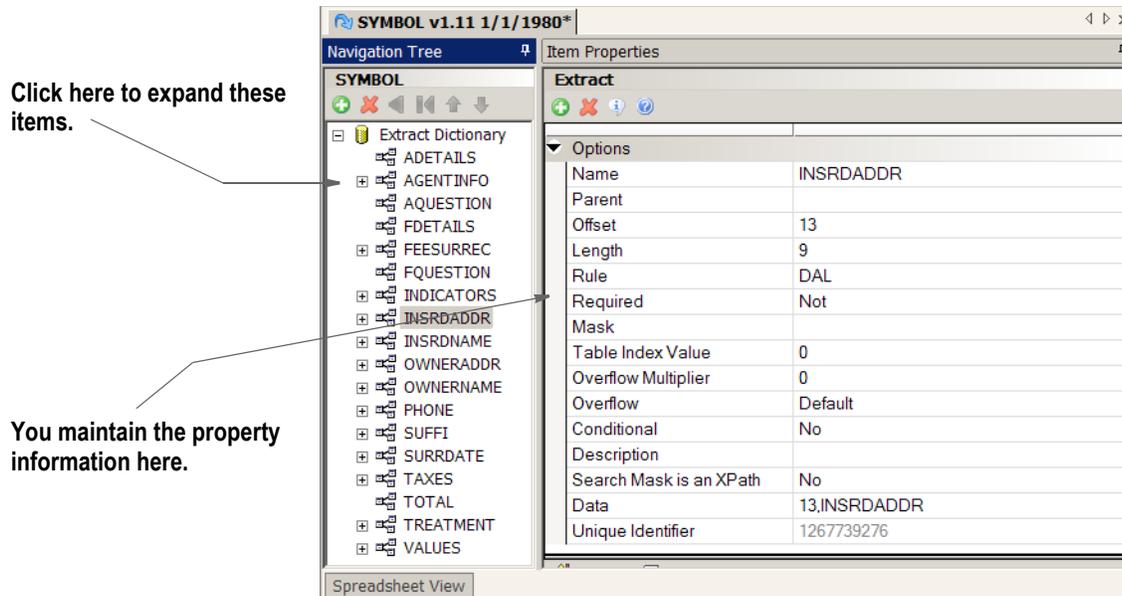
Note You can have a separate XDD file for each line of business. This is specified in the BDF file. At conversion time, the MASTER.DDT file, if one exists, and the XDB.DBF file, if one exists, automatically import into the SYMBOL.XDD file.

OPENING AN EXTRACT DICTIONARY

There are two ways to open an extract dictionary:

- Using the Manage, Dictionaries, Data Extract option
- By clicking on Data Extract in the Workspace tree

The main window that appears when you are managing an extract dictionary is shown here.



When you highlight an extract dictionary or the fields it contains, its property information appears so you can modify it.

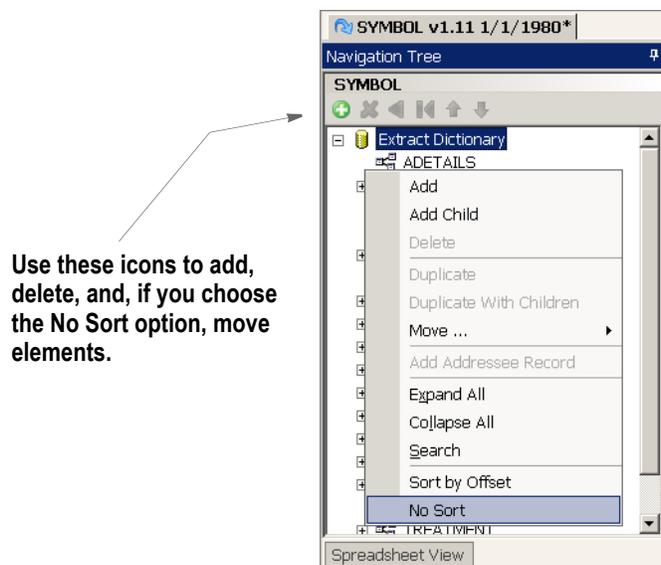
Property	Description
Name	Enter a unique name for this entry.
Offset	Enter the offset of the data in the extract.
Length	Enter the length of the data in the extract.
Rule	List any optional rule to run on this data. You can only use the DAL ? rule and the GlobalFld rule.
Required	Choose one of these options: Not, Host, Operator, Either.
Mask	Include the formatting mask for the data returned by the rule.
Table Index Value	Used with the TableLookUp rule.
Overflow Multiplier	Specifies the overflow multiplier for this entry.
Overflow	Specifies the overflow, if any, for this entry.
Conditional	Set to Yes if this entry can be used as a trigger.

Property	Description
Description	Reserved for future use.
Search Mask is an XPath	Set to Yes if the search mask is an XPath. Otherwise, choose No.
Data	Use this field for search masks and rule parameters.
Unique Identifier	Indicates a unique ID for this record.

SORTING XDD ELEMENTS

You can view XDD elements by name, by offset, or in the order in which they were manually entered or imported (No Sort). The No Sort option gives you a view which can help you compare elements when you are trying to match the XDD to a sample import XML file.

To change the sort order when working in the XDD, right-click and choose the appropriate option. Studio defaults to sorting by name so the following example omits the Sort by Name option and only shows the Sort by Offset and No Sort options:



Note During an XML import, the No Sort option is analogous to the order in which the XML presented the fields. For the No Sort option, it is the underlying import that determines the order of the created fields within the XDD.

Choose the option you want and Studio re-orders the elements.

You can use the Add and Delete icons to enter or remove elements. When you choose the No Sort option, you can use the Move icons to re-arrange elements within the XDD navigation view.

IMPORTING XML SCHEMAS INTO THE XDD

You can import an XML schema (XSD) into an extract data dictionary (XDD). From the schema, the XDD recognizes parent and children elements and also inherits any attributes as children.

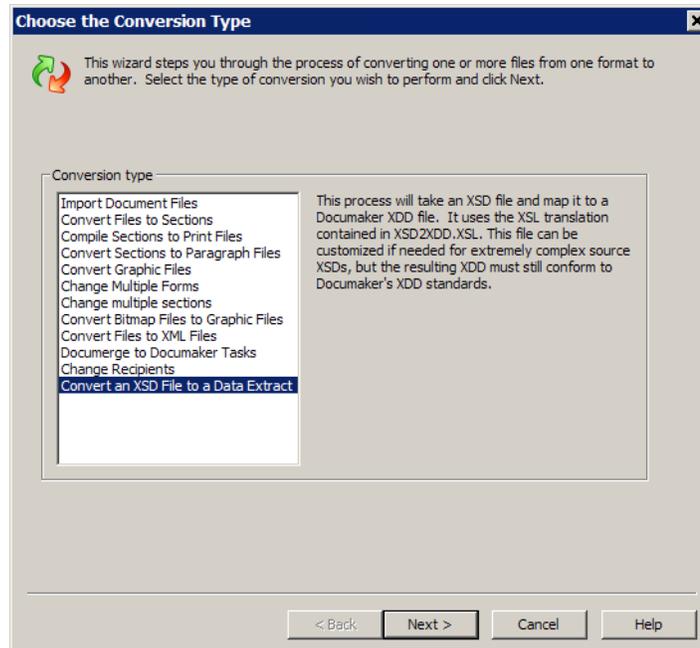
Importing an XML schema to populate the XDD provides these benefits:

- Lets you build the XDD before you have sample XML input data
- Helps reduce errors when creating the XDD
- Ensures compatibility with the input file structure
- Makes it easier to synchronize changes to the input file schema
- Helps reduce mapping errors as XPath expressions are easily derived from XML schema

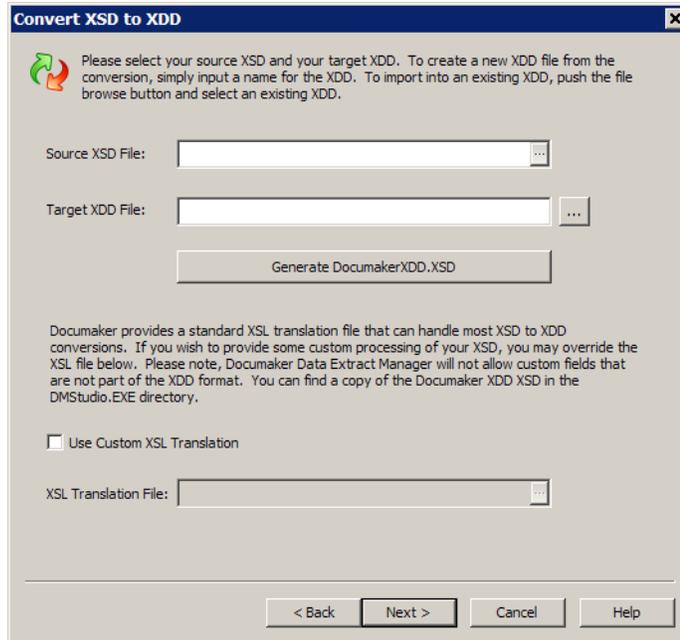
Note For the conversion, Studio uses the XSL translation contained in the XSD2XDD.XSL file. This file is located in the directory that contains the DMStudio.exe file, usually c:\fap\dl. You can customize this file if needed to handle complex source XSD files, but your changes must conform to Documaker's XDD standards.

To import an XML schema, follow these steps:

1. Choose the Tools, Conversion option. The Choose the Conversion Type window appears.
2. Select the Convert an XSD File to a Data Extract option.

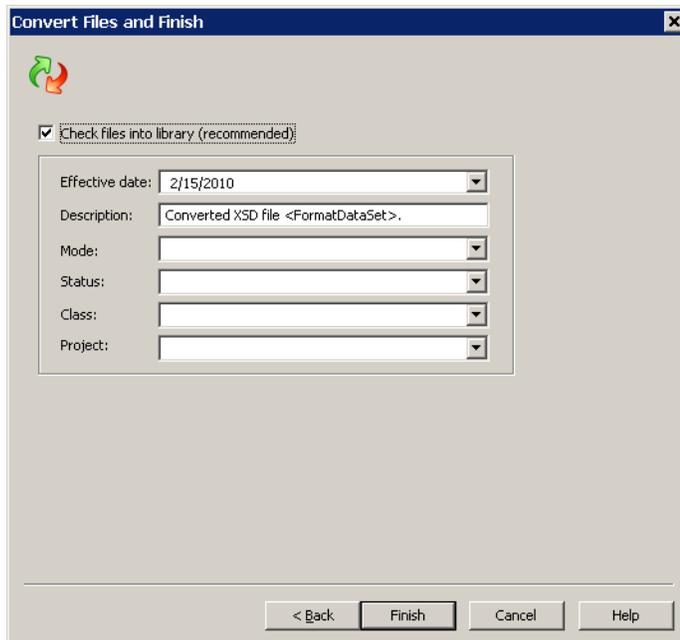


The Convert XSD to XDD window appears.



- Use the Browse button to select the source XSD file. Enter the name you want to assign to the target XDD or click Browse to select an XDD file. Studio lets you use a custom XSL translation file if necessary. Click Next when you finish specifying the files to use.

The Convert Files and Finish window appears.



- By default, Studio checks the new XDD into the library for you. You can change this option if necessary. On this window you can also make or change the entries in the following fields:

Field	Description
Effective Date	Studio shows you the current date as a default. You can enter another date if necessary.
Description	Change the default description if necessary.
Mode	Use this field to assign a mode to the resource as it is checked in. For instance, you can use the Mode field to specify where in the development cycle the resource is.
Status	Use this field to assign a status to the resource as it is checked in. For instance, you could use the Status field to indicate whether a resource has passed or failed testing.
Class	Use this field to assign a class to the resource as it is checked in. For instance, you could use the Class field to indicate the market in which a resource was applicable.
Project	Use this field to assign a project ID to the resource as it is checked in. For instance, you could use the Project field to indicate which project a resource was associated with.

You define modes, statuses, classes, and projects using the Manage, Settings options. For instance, you could set up modes to denote milestones in the development process such as Development, Testing, and Production. For more information, see *Defining Mode, Status, Class, and Project Options on page 380*.

5. Click Finish to convert the XML schema file.

Keep in mind...

- Studio imports all XDD fields as strings
- Studio assigns the Move_It rule to these fields in the XDD
- Documaker does not validate the XSD file during processing

IMPORTING XML EXTRACT FILES INTO THE XDD

If you are using an XML extract files, there are two main ways to represent data. The first minimizes the use of attributes in favor of explicit nodes. Here's an example.

```
< Transaction >
  <Customer>
    < Name >Your Name Here</Name>
    <Phone>111-222-3333</Phone>
  </Customer>
</Transaction
```

The second minimizes the number of nodes by using attributes. Here's the same data reformatted to use attributes:

```
< Transaction >
  <Customer Name="Your Name Here" Phone="111-222-3333" </Customer>
</Transaction
```

Depending upon how your XML extract file is defined, you may need to modify the ImportXMLAttributes option.

```
< Symbol >
  ImportXMLAttributes = Yes
```

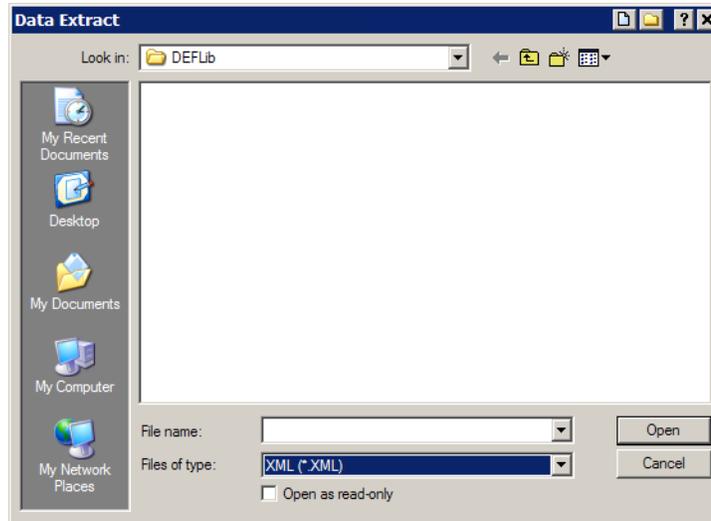
Option	Description
ImportXMLAttributes	Enter Yes if you want Studio to import XML attributes into your extract dictionary (XDD) and automatically generate the proper mappings. The default is No. Use the default if your XML files minimize the use of attributes in favor of explicit nodes.

Note You can check this option using the Manage, System, Settings option and then selecting Options by Group.

Importing XML Attributes

If your XML extract file is defined to use attributes when representing data, make sure the ImportXMLAttributes option is set to Yes. Once set, follow these steps to import XML attributes and have the node attributes become fields:

1. Open the Extract Dictionary and select the File, Import option. The Open File window appears.
2. Click Browse and load the XML file you want to import into the XDD file.



Note Be sure to select XML as the file type.

During the import, if a field matches an existing XDD field, Studio lets you decide to add, replace, or skip that field. If you add the field, you will have fields in the XDD which have duplicate names. You can later decide which of these fields to discard.

Once Studio imports the XML file, you return to the XDD file navigation tree, where you will see a new tag in the XDD file called *InterfaceRequest*. This tag represents the contents of the XML file you just imported.

For instance, here is an example which assumes you have these nodes, each with an attribute assigned:

- AUTO_Make
- AUTO_Model
- AUTO_Year

These nodes are listed under a parent node called AUTO. All three attributes nodes have truncated names so the new attribute name can be added to it. In addition, the attribute name is prefixed with an at sign (@) and is placed in the Data name section. Here is an example:

Information for each node appears here. Here you see the options for the AUTO_Make node.

Extract	
Options	
Name	AUTO_Make
Parent	AUTO
Offset	0
Length	80
Rule	MOVE_IT
Required	Not
Mask	
Table Index Value	0
Overflow Multiplier	0
Overflow	Default
Conditional	No
Description	
Search Mask is an XPath	Yes
Data	/@Make
Unique Identifier	1268921188

Here is an example of the XML:

```
<InterfaceRequest>
  <AUTO Make="Toyota" Model="Solara" Year="2001">01</AUTO>
  <AUTO Make="Ford" Model="Taurus" Year="2002">02</AUTO>
  <AUTO Make="Chevrolet" Model="Silverado" Year="2004">03</AUTO>
  <AUTO Make="Dodge" Model="Durango" Year="2005">04</AUTO>
</InterfaceRequest>
```

USING ADDRESSEE RECORDS

Addressee records in the extract dictionary support class recipients and addressee-based processing, for use with Documaker and the iDocumaker 4.0 Correspondence application.

To support addressee processing, the system stores multiple sets of addressee information for a single transaction or document set. You can then use this information distribute documents to various addressees.

Note See also the [Documaker Administration Guide](#) for information on printing addressee information and using addressee information in recipient batch files.

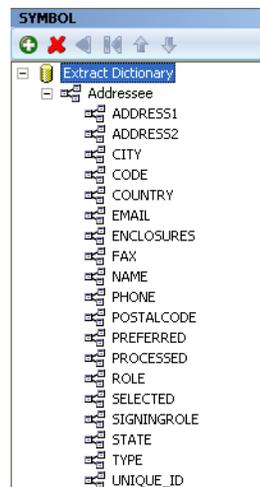
In a Documaker environment, this record data is mapped the same way other data elements are mapped to the data dictionary. In an iDocumaker environment, the data can come from end user edits.

Adding an Addressee Record

To enable the use of Addressee records, first create the Extract Data Dictionary (XDD) entry that will be associated with the recipient that uses the Address Map.

To do this, check the XDD out of the workspace, then click Add to add a parent record for the addressees, such as *Addressee* or *Lienholder*. You can also select an existing XDD entry that has no defined children.

Right-click and select the Add Addressee Record option to flood the data schema for Addressees into the field. You can also select this option from the Action menu.



The Move_It rule is used on all addressee records. You can modify the rule on applicable addressee records as needed. You do not have to map every field added to the Addressee record and you can delete fields you are not using.

Note If you add children to this record, those entries are not stored as information linked to addressees.

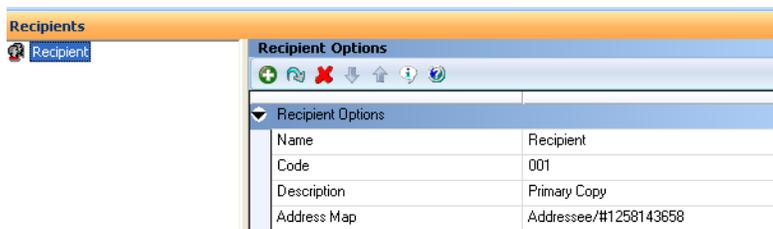
Adding the Address Map in the BDF

The Address Map field under the Recipient entry in the Business Definition (BDF) stores the Extract Data Dictionary member that defines the mapping for this recipient's addressee records. Use this field in situations where you want to store multiple sets of address information for a single class of recipients on a single transaction.

After adding the addressee record to the Data Dictionary, add the Address Map field to the selected recipient in the Definition file. To do this, follow these steps:

1. Check out the BDF file.
2. Click the Ellipsis button in the Address Map field to access the Parent Name entry in the XDD for this BDF recipient. The Unique Identifier number that appears with the Parent Record name comes from the Extract Dictionary. This ID ensures the selection of the correct parent record if there are duplicate entries in the XDD.

You can also manually type the name of the parent record into the Address Map entry, without adding any Unique ID. Remember that any XDD record selected for the Address Map should contain the addressee record.



Note If you select the XDD Parent Name record using the Search functionality, then any modification of the Unique ID portion of the Address Map can cause run-time errors.

Testing an Address Map

You can test your Address Map in Studio. To do this, simply create a test profile and step through your test.

Note You can choose the Test, Show Print Batches option in Studio's Test manager to verify the batch records after they have been written.

You can automatically map addressee records using the RunTriggers rule in your AFGJOB.JDT file. If you map addressee records this way, the print stream for that recipient includes a copy for each addressee member. See Writing Addressee Records to the Recipient Batch Files for more information.

If using the RunTriggers rule, then under the Test menu, there is a new option to Show Addressee List. When stepping through Test, you must have executed the RunTriggers rule before the Addressee List will show. If not using RunTriggers, you may still view addresses in Test without the Addressee List.

▶ Step Into ⏸ Pause ▶ Step Out Of ⏹ Stop 🔄 Refresh Format 👤 Profile Properties
▶ Run ▶ Step ▶ Show Print Batches ▶ Stop on DAL Script
▶ Show Addressee List

Test Scenarios

Address Record

Formset Tree

- CENTRAL - ACCOUNT_STATUS
 - AM-472
 - HDR ALL CORR
 - FORMSET PAGE NUM
 - FORMSET PAGE NUM OF
 - RJUST DATE
 - POSITION
 - ADDRESS
 - SALUTATION POL
 - AM-472 A
 - CLOSE LTR
 - SIGNATURE JONES
 - FTR ALL CORR
 - XDD TABLE - ADDRESSEE REC

Formset View

Page 2

1	Workfield Lead	
1	Workfield Lead	
1	Workfield Lead	

andy.jones@gmail.com	
andy.jones@gmail.com	
andy.jones@gmail.com	

USING THE COMMON FIELDS DICTIONARY

The Common Fields Dictionary provides an easy way to store common variable field information to make setting up and creating sections and paragraphs faster and more consistent. This file contains a record for each unique variable field name. Each record contains information such as...

- Field name
- Field length
- Field type
- Font ID

Be sure to add all of the known fields to the Common Field Dictionary. This lessens the need to create fields when creating sections, either in Studio or in the Documaker Add-In for Word

When naming fields, avoid using cryptic abbreviations. When possible, use easily identifiable, field names that others who work with Studio can quickly recognize. This makes selecting the correct field in Studio or the Documaker Add-In for Word easier.

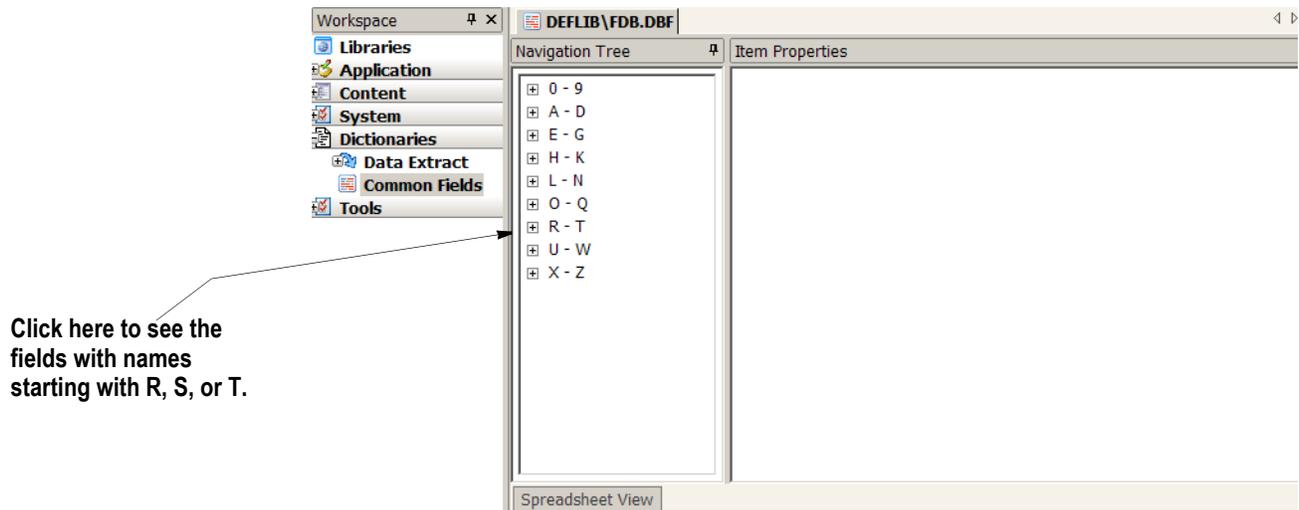
You can enter field information directly into Studio's Common Field Dictionary or by use the File, Import option to import various files, such as comma separated value (CSV) or Tagcommander (TGA) files.

WORKING WITH THE COMMON FIELDS DICTIONARY

Click on Common Fields in the tree to work in a centralized database where the system stores variable field attributes for the master resource library. Here you can edit and manage common field information.

Storing common variable field information can make setting up and creating FAP files faster and more consistent.

Here is an example of the window that appears when you are working with fields:

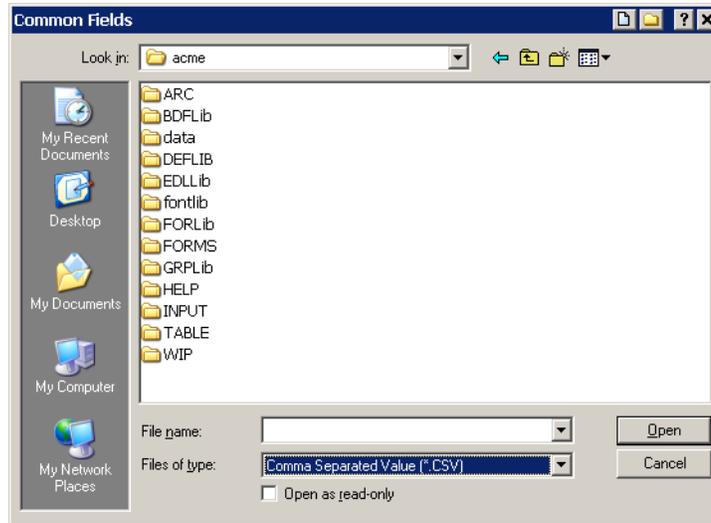


Use the Navigation Tree to locate the field you want to work with. When you highlight a field, its property information appears so you can modify it.

IMPORTING CSV FILES

Follow these steps to import a comma separated value file:

1. Choose Manage, Dictionaries, Common Fields.
2. Choose File, Import (not Import Workspace files). On the Open File window, double click Fields.
3. In the Field window, change the Files of Type field to CSV, as shown here:

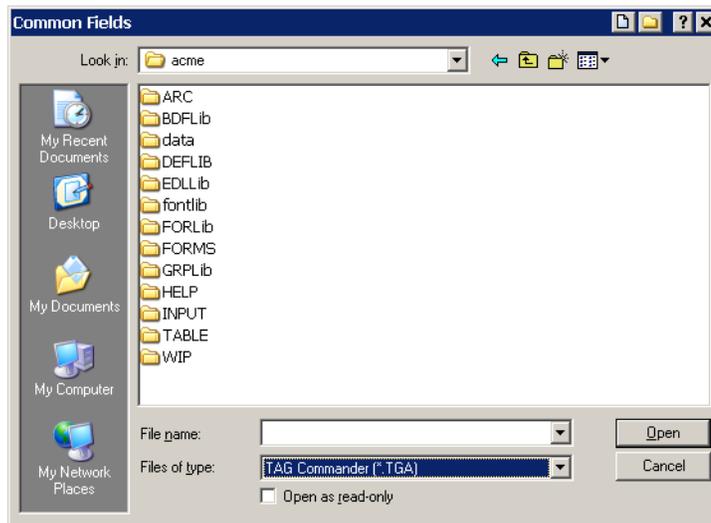


4. Browse for the file you want to import and then click Ok.

IMPORTING TGA FILES

Follow these steps to import a Tagcommander (TGA) file:

1. Choose Manage, Dictionaries, Common Fields.
2. Choose File, Import (not Import Workspace files). On the Open File window, double click Fields.
3. In the Field window, change the Files of Type field to TGA, as shown here:



4. Browse for the file you want to import and then click Ok.

Chapter 11

Creating Tables

Click on Tables to attach table data to a variable field so it is available during data entry.

There are two ways to create a table file: using the Tables option, or when you create a variable field using the Properties window. The entry screens are the same. The Tables option lets you create tables for the variable fields you create.

Tables make the entry process quicker and more efficient for the end user. Users choose from data in a table rather than keying in information. This is especially useful, for example, when entering lengthy codes. Tables reduce data entry errors and increase speed.

OVERVIEW

The data for a table can either be manually entered or merged from a client data source. A table file can contain many tables. Each table contains many entries. The entries in the tables are the choices available to the data entry user. Each table entry contains two parts: a key and a description. A key might be a short abbreviation, such as GA, and the description provides a longer and more complete explanation of the key, such as Georgia.

Table files reside in the selected master resource library (typically in the Table directory). A table file (DBF) can contain multiple tables identified by unique names. You can create a new table file, or you can open an existing table file to correspond with your section and its variable fields. After you open or create a file, you can add or edit tables in the file to meet the particular requirements of the section.

Documaker uses a variety of database files as input and output for various software modules. For example, the GenTrn program outputs a transaction file that serves as an input to the GenData program. The transaction file is stored as a database file, and as such, the transaction file has a pre-defined record structure. This record structure specifies the length, type, and order of the fields that contain the information in the file.

The software modules that use these database files must have access to the record structure in order to write data in the proper format and to read that data correctly. This is true for all database files. The purpose of DFD (data format definition or just definition) files is to provide record structure information to the software modules.

DFD (Data Format Definition) files contain information defining the structure of the data contained within a database file. Many common system files are stored in database format. For example, in addition to the transaction file already discussed, the new transaction, application index, recipient batch, WIP (work in progress), help, and table files are all stored in database format. These system database files can be in a variety of formats, including xBase, DB/2, ODBC, and standard sequential files (such as flat ASCII files). The record structure defined in the DFD files remains independent, regardless of the type of database being used — although there are occasionally exceptions for some database specific records.

Not all of these database files require an external DFD file. In some cases, the file's record structure is coded directly into the software modules that access the file. There are currently no external DFD files for Documaker help and table database files for this reason. However, external DFD files are used with other system database files, in many cases, because DFD files allow for easier modification of the database file's record structure, without having to modify the software modules directly.

DFD Files

There are several system database files, meaning that these files are written and read via calls to the DBLIB data base software library. As already mentioned, these database files can be in several formats, including xBase, DB/2, and flat ASCII. Also, not all system database files require a corresponding DFD file because their record structure is coded in the software modules that access them.

The following are Documaker database files:

- Transaction files

- New transaction files
- Recipient batch files
- Manual batch files
- Application index files
- WIP files
- Help files
- Table files

Of the files listed above, only five require an external DFD file. The use of one other DFD file 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 – see below)

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 program is used. The help and table files do not support the use of external DFD files.

Of the system database files that require external DFD files, only these actual DFD files are required:

- Transaction file DFD
- Recipient batch file DFD
- Application index file DFD

The transaction file DFD is used by both the transaction file and the new transaction file. The recipient batch file DFD is used by both the recipient batch files and the manual batch files. Finally, the application index file DFD is used by the application index file.

So, for installations, these are the only DFD files that possibly need to be configured.

TRANSACTION FILE DFD

The transaction file DFD, commonly referred to as the TRNDFDFL, is used by these modules:

- GenTrn
- GenData

- GenArc

The GenTrn program writes out the transaction file using the TRNDFDFL. The GenData program reads the transaction file and writes out the new transaction file using TRNDFDFL. The GenArc program reads the new transaction file using TRNDFDFL.

The name of the TRNDFDFL is set in the initialization file in the Data control group as follows:

```
< Data >
  TrnDfdFile = TrnDfdFl.Dfd
```

RECIPIENT BATCH FILE DFD

The recipient batch file DFD, commonly referred to as the RCBDFDFL, is used by these modules:

- GenData
- GenPrint
- GenWIP

The GenData program writes out the recipient and manual batch files using the RCBDFDFL. The GenPrint program reads the recipient batch files using RCBDFDFL. The GenWIP program reads the manual batch files using RCBDFDFL.

The name of the RCBDFDFL is set in the initialization file in the Data control group as follows:

```
< Data >
  RcbDfdFile = RcbDfdFl.Dfd
```

APPLICATION INDEX FILE DFD

The application index file DFD, commonly referred to as the APPIDXDFD is used by these modules:

- GenArc
- AFEMAIN

The GenArc program writes out the application index file using the APPIDXDFD. The AFEMAIN program reads the application index file using APPIDXDFD.

The name of the APPIDXDFD may be set in the initialization file in the ArcRet control group as follows:

```
< ArcRet >
  AppIdxDfd = AppIdx.Dfd
```

However, the APPIDXDFD name does not have to be set as shown above provided you are running the system in a Windows environment. If the APPIDX.DFD name is not specified as shown, the system automatically appends a DFD extension to the APPIDX name specified in the same group, which is specified as follows:

```
< ArcRet >
```

AppIdx = AppIdx

This will not work in an environment that does not support file name extensions, such as z/OS.

In addition to the specifying the name of the APPIDXDFD, other related settings must be made in the initialization file to use the GenArc and AFEMAIN programs. These changes will be discussed in the Archive/Retrieval setup section.

FIELD SECTION

The FIELD section lists all the fields in the record structures and the order those fields appear in the storage media. The fields are automatically stored internally in the same order they appear externally.

Field	Format Type	Field / Format Description
Name	FIELDNAME can have length limitations, based upon each database type. For instance, when using xBase, the FIELDNAME length limit is 10 characters.	Name of the field used by applications to reference data in the DFD record. Defaults to FIELDXXX, where XXX is the next sequential field number.
Internal Type	BLOB CHAR_ARRAY CHAR_ARRAY_NO_NULL_TERM DATETIME DECIMAL DOUBLE FLOAT LONG LONG DOUBLE NOT_PRESENT SHORT SIGNED CHARACTER TIMESTAMP UNKNOWN UNSIGNED CHARACTER UNSIGNED LONG UNSIGNED SHORT VARCHAR	Data format of field used internally by Documaker Binary Large Object NULL terminated string character array not NULL terminated A Documaker date/time field expressed as a decimal value double precision float single precision 32-bit signed integer long double precision not present in this record 16-bit signed integer a signed character a Documaker time stamp data type is unknown an unsigned character 32-bit unsigned integer 16-bit unsigned integer variable length character array
Internal Length		Same as External Length except one additional byte is added to length to store null termination byte. Maximum of 255.
Internal Precision		Same as External Precision. Maximum of 255.
External Type	Same as Internal Type format types.	Data format of the field on storage media. Same as Internal Type format descriptions.
External Length		Length of field on storage media. This is valid for data types CHAR_ARRAY & CHAR_ARRAY_NO_NULL_TERM only. It is ignored for all other data types. Maximum of 255 characters.
External Precision		Number of digits after decimal point. This is valid for data types FLOAT, DOUBLE, and LONG_DOUBLE only. It is ignored for all other data types. Maximum of 255 characters.

Note The external record definition must match the actual records written to or read from the database. The internal record definition is provided for easier programming use.

KEY SECTION

The KEY section lists fields that are key fields. This field is only used by DB/2 and indicates that the field is required. The purpose of defining keys is to define columns if a database is being used when WIP and/or Archive is being implemented.

Field	Description
Name	Defaults to KEY01, KEY02, and so on. Can be changed.
Field list	Used for associating the KEYXX with a field already defined in the DFD.
Descending	If using SQL databases, descending works by issuing the correct <i>order by</i> command.

Chapter 12

Converting Files

Use the Conversion option to convert, change, or compile one or more files.

This chapter covers these topics:

- *Overview on page 410*
- *Converting Word and RTF Files on page 412*
- *Converting Files into Sections on page 422*
- *Compiling Sections into Print Files on page 427*
- *Converting Sections into Paragraphs on page 429*
- *Converting Graphic Files on page 430*
- *Changing Templates on page 431*
- *Making Changes to Multiple Sections on page 433*
- *Converting Bitmap Files into LOG Files on page 439*
- *Converting System Files into XML Files on page 440*
- *Converting Documerge Resources on page 442*
- *Changing Recipient Information on page 445*
- *Importing XML Schemas into the XDD on page 447*
- *Finishing a Conversion on page 449*

OVERVIEW

The first step is to start a conversion by choosing the Manage, Conversion option from the menu or by double clicking on Conversion in the Workspace tree. The Conversion Type window appears:



Highlight the type of conversion you want to perform and click Next to go to the next window. Click Cancel to exit Conversions.

To	See
Import Documaker document files created in Microsoft Word	Appendix A, <i>Using the Documaker Add-in for Microsoft Word</i> on page 555
Convert Microsoft Word (DOC or DOCX) files or RTF files into forms	<i>Converting Word and RTF Files</i> on page 412
Convert files into sections	<i>Converting Files into Sections</i> on page 422
Compile sections into PCL, AFP, Metacode, PostScript, or VIPP print files	<i>Compiling Sections into Print Files</i> on page 427
Convert sections into paragraph (PAR) files	<i>Converting Sections into Paragraphs</i> on page 429
Convert graphics files into another format	<i>Converting Graphic Files</i> on page 430
Assign, reapply, or remove templates from form (FOR) files.	<i>Changing Templates</i> on page 431

To	See
Make changes to sections. This includes: <ul style="list-style-type: none"> • Fixing negative coordinate on a section • Rotating a section 90 degrees • Retaining section dimensions • Adjusting X and Y coordinates • Changing section names • Changing DAL calcs into DAL scripts • Telling Studio to use 4-digit year date formats • Updating variable fields on the section from the FDB • Requiring unique field names • Deleting fields • Recalculating coordinates • Changing fonts Updating DDT files from FAP files • Updating DDT files from the XDB • Combining adjacent text labels • Combining overlapping text labels • Returning to the default colors • Changing to print in color • Mapping alternative font characters 	<i>Making Changes to Multiple Sections on page 433</i>
Convert graphics files into the Documaker LOG file format	<i>Converting Bitmap Files into LOG Files on page 439</i>
Convert application definition (BDF), form lists (GRP), form (FOR), template (TPL), or section (FAP) files into XML format.	<i>Converting System Files into XML Files on page 440</i>
Convert Documerge resources into Documaker resources	<i>Converting Documerge Resources on page 442</i>
Change a recipient in selected resources	<i>Changing Recipient Information on page 445</i>
Import an XML schema (XSD) into an extract data dictionary (XDD)	<i>Importing XML Schemas into the XDD on page 447</i>

Note Other conversion options are available by running command line utilities. Refer to the [Docutoolbox Reference](#) for more information.

CONVERTING WORD AND RTF FILES

You can convert Microsoft Word (DOC or DOCX) and Rich Text Files (RTF) into Documaker form (FOR) files. And optionally, during the conversion Studio can create separate files (FAP files) for each section and separate files (LOG files) for each graphic in the converted form.

Note Microsoft Word must be available on the workstation where the Studio conversion is run. If Microsoft Word is not available, the convert process will fail and you will see this message in the output area:

Failed to Start Microsoft Word.

To open Microsoft Office Word 2007 .DOCX files with Microsoft Office Word 2003, Word 2002, or Word 2000, you need to install the Microsoft Office Compatibility Pack for 2007 Office Word. For more information, go to this web site:

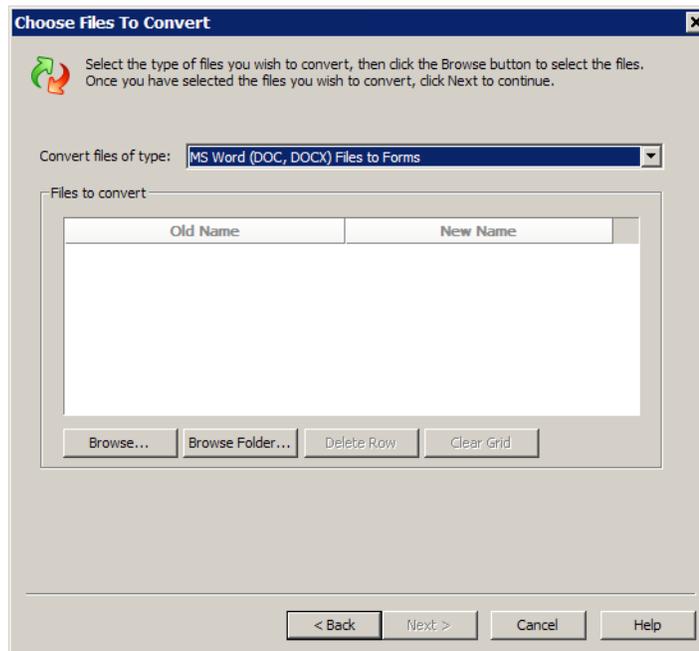
www.microsoft.com

Understanding the Import Process on page 575 provides general information on what is and is not converted. For information on how fonts are handled, see *Understanding Font Mapping* on page 419.

Note You can also use the RTF2FAP utility to convert RTF files into FAP files. See the [Docutoolbox Reference](#) for more information.

Follow these steps to convert Word and RTF files into Documaker forms:

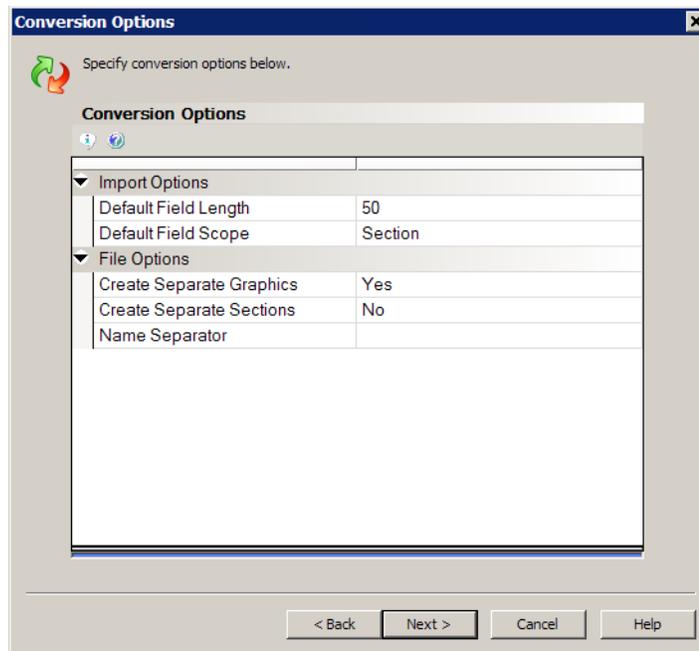
1. Choose the Import Document Files option on the Choose the Conversion Type window. The Choose Files to Convert window appears.



2. Select the type of conversion you want, then select the files you want to convert.

Click	To select
Browse	The files you want to convert.
Browse Folder	The folder that contains the files you want to convert. You can also select Search Sub Folders to find files in sub folders. The Browse Folder button changes to a Stop button that lets you end a search.

Use the Delete Row or Clear Grid buttons modify your file selections. You can also give a file a new name on this screen. Once you select the files to convert, click Next. The Conversion Options window appears:

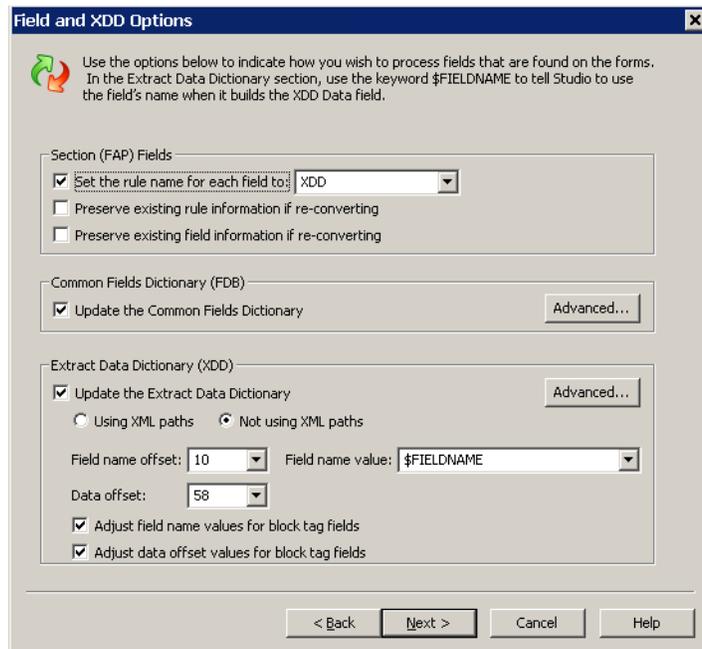


Field	Description
Import Options	
Default Field Length	Enter the field length you want Studio to assign to the fields it finds during the conversion. The default is 50 characters.
Default Field Scope	Choose the field scope you want Studio to assign to the fields it finds during the conversion. You can choose from these options: <ul style="list-style-type: none"> • Section – data is only written to the same named variable field within a section. • Form – data is written to the same named variable field throughout all sections within the form. • Global – data is written to the same named variable field throughout all sections and all forms in the form set. The default is Section.

File Options

Field	Description										
Create Separate Graphics	<p>Enter Yes if you want Studio to create separate graphic (LOG) files and assign a name based on the name of the input document and the separator you specify in the Name Separator field.</p> <p>Enter No if you want Studio to embed any graphics into the section. You can later remove embedded graphics by editing the file and replacing the graphic or unchecking the embedded option for the graphic.</p> <p>The default is Yes.</p>										
Create Separate Sections	<p>Enter Yes if you want Studio to create a FAP file (section) for each page and for each header and footer in the input document. If you answer Yes, Studio assigns a name to each page of the document based on the name of the input document and the separator you specify in the Name Separator field.</p> <p>Enter No if you want Studio to place the content into embedded sections on the form. You can later unembed these sections in the Form manager by unchecking the Embedded option.</p> <p>The default is No.</p>										
Name Separator	<p>Enter the character you want Studio to insert between elements of the file name. The separator can be nothing, a blank, an underscore, or any other valid Windows file name character.</p> <p>For example, if the name of the input document is Declaration and you choose to create separate sections you would get these results for page 1:</p> <table border="1"> <thead> <tr> <th>Separator</th> <th>Result</th> </tr> </thead> <tbody> <tr> <td>(nothing)</td> <td>Declaration1.FAP</td> </tr> <tr> <td>Blank</td> <td>Declaration1 .FAP</td> </tr> <tr> <td>_</td> <td>Declaration_1.FAP</td> </tr> <tr> <td>#</td> <td>Declaration#1.FAP</td> </tr> </tbody> </table>	Separator	Result	(nothing)	Declaration1.FAP	Blank	Declaration1 .FAP	_	Declaration_1.FAP	#	Declaration#1.FAP
Separator	Result										
(nothing)	Declaration1.FAP										
Blank	Declaration1 .FAP										
_	Declaration_1.FAP										
#	Declaration#1.FAP										

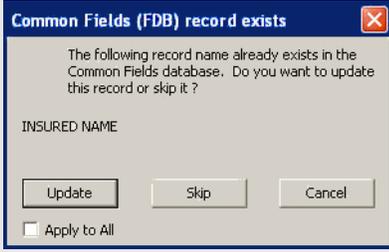
Click Next when finished. The Field and XDD Options window appears:

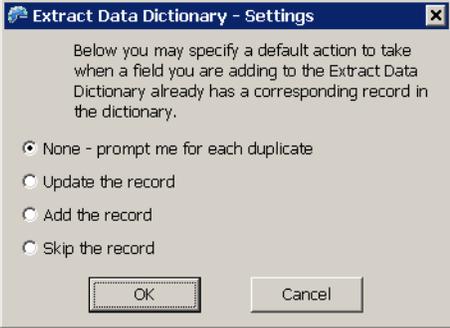


- The Field and XDD Options window lets you tell Studio what to do with the fields it encounters during the conversion.

Field	Description
Section (FAP) Fields	
Set the Rule Name for each Field to	Check this box if you want to set the rule name for each field in the section to a specific value. Typically, you would set the rule name to XDD. This tells the system that at runtime, it should retrieve field information from the Extract Data Dictionary (XDD) record for this field.
Preserve Existing Rule Information if Re-converting	If you have previously converted this file and if you have updated that section with rule information you want to preserve, check this box to preserve the rule information during this conversion.
Preserve Existing Field Information if Re-converting	If you have previously converted this file and if you have updated the fields in the section with information you want to preserve, such as field length, type and so on, check this box to preserve that field information during this conversion.

Common Fields Dictionary (FDB)

Update the Common Fields Dictionary	<p>Check this box if you want Studio to update the Common Fields Dictionary (FDB) with each field name Studio finds in the source file. If a field in the source file already exists in the Common Fields Dictionary, you can choose to update the field, skip the field, or cancel the conversion.</p> <p>Click Advanced to specify a default action Studio should take when the field to be added already exists in the Common Fields Dictionary.</p> <p>You can choose to be prompted each time for which action to take, have Studio update the record, or have Studio skip the record.</p>  <p>If you choose to have Studio prompt you each time it finds a duplicate record, you will see a window similar to the following, during conversion:</p>  <p>Click Update to update the record in the FDB or Skip to skip this record in the FDB. If you want to cancel the conversion, click Cancel.</p> <p>Check the Apply to All box to apply your selection of Update or Skip to the rest of the duplicate records Studio encounters during the conversion.</p>
-------------------------------------	--

Field	Description
<p>Extract Data Dictionary</p> <p>Update the Extract Data Dictionary</p>	<p>Check this box if you want Studio to update the Extract Data Dictionary (XDD) with each field name it finds in the input document.</p> <p>Click Advanced to specify a default action Studio should take when the field to be added already exists in the XDD. You can choose to update the record, add the record, skip the record, or be prompted each time for what action to take.</p>  <p>If you choose to have Studio prompt you each time it finds a duplicate field, you will see a window similar to the Common Fields (FDB) Record Exists window that prompts you to add, replace, skip the record, or cancel the conversion.</p>
<p>Using XML Paths</p>	<p>Click this button if your extract file is in XML format.</p> <p>If the extract file is in XML format, you will locate field data in your extract file using an XPath statement. This XPath statement is specified in the Field Name Value field. Make sure the Field Name Value field is set as shown here:</p> <p style="text-align: center;"><code>\$/FIELDNAME</code></p> <p>This tells the system to set up an XPath-compatible search token that references the field's name, such as the one shown here:</p> <p style="text-align: center;"><code>/FieldName</code></p>

Field	Description
Not Using XML Paths	<p>Click this button if your extract file is <i>not</i> in XML format, but instead is a standard sequential or flat file.</p> <p>If the extract file is a standard sequential file, you will locate field data in your extract file using an offset, value statement. The offset, value statement is specified in the Field Name Offset and Field Name Value fields.</p> <p>Make sure the Field Name Value field is set as shown here:</p> <pre>\$FIELDNAME</pre> <p>If you set the Field Name Offset field to 10, this creates a result similar to the one shown here:</p> <pre>10,FieldName</pre> <p>Note: Whether you want the slash before the name depends on whether it is appropriate for the type of search your data requires. If your field names do not appear in the extract rows, specify whatever text identifies the record where the field data occurs. For example, with these settings:</p> <pre>Field Name Offset:1 Field Name Value: InputData</pre> <p>The result in your data area will be:</p> <pre>1,InputData</pre> <p>This means your fields will be in a record identified by finding <i>InputData</i> starting in the first text position. Then the data offset from the XDD definition of the field would be used to find the actual field value. You should only have to set the Field Name Value the first time you do a conversion. After you set the field, Studio remembers your entry.</p> <p>Studio also sets the Data Offset field to a number you specify, such as 58, for an extract file created with the VRF2EXP utility</p>
Field Name Offset	<p>This field is only applicable if your extract file is <i>not</i> in XML format.</p> <p>Here you can specify the offset at which the system can expect (or test for) a particular value. You specify the value the system should test for in the Field Name Value field. The default is 10.</p>
Field Name Value	<p>This field is applicable for both types of extract file.</p> <p>If your extract file is not in XML format, the Field Name Value field lets you specify the value to test for at the offset specified in the Field Name Offset field. For example, if the Field Name Offset field contains 10 and the Field Name Value field contains <i>MYFIELD</i>, the Data field in the XDD is set to <i>10,MYFIELD</i>. When Documaker processes this field, it looks at offset 10 in the record in the extract file and tests for the string <i>MYFIELD</i> at that offset. If the test is successful, Documaker goes to the offset specified in the Data Offset field and copies the data at that offset into the field.</p> <p>If your extract file is in XML format, the Field Name Value field lets you specify the XPath to use to search for data for the field.</p> <p>For example, if the Field Name Value field contains <i>/MYFIELD</i>, the Data field in the XDD is set to <i>/MYFIELD</i>. When Documaker processes this field, it would use this XPath (<i>/MYFIELD</i>) to locate data to copy into the field being processed.</p>

Field	Description
Data Offset	<p>This field is only applicable when your extract file is <i>not</i> in XML format.</p> <p>Use this field to indicate the offset in the record in the extract file at which to expect field data.</p> <p>For example, if you enter 58 here, the Field Name Offset field contains 10, and the Field Name Value field contains <i>\$FIELDNAME</i>, during conversion, an XDD record would be created whose Data portion contains 10,<i>fieldname</i>, where <i>fieldname</i> is the actual field name (since the <i>\$FIELDNAME</i> keyword parameter was used) and the whose Offset portion contains 58.</p> <p>Documaker would process the field named <i>ADDRESS1</i> by reading the contents of the XDD record for that field (<i>ADDRESS1</i>), then looking in the extract file at offset 10 for the value <i>ADDRESS1</i>. If the string <i>ADDRESS1</i> is found at offset 10, Documaker would look at offset 58 and copy the data at that offset to the field on the section.</p>

Use these boxes when you are performing a Documerge to Documaker conversion and you are using the VRF2EXP utility to generate a Documaker extract file from a Documerge VRF file.

Adjust Field Name Values for Block Tag Fields	Check this box if you want the system to update the Data field in the XDD with the field's root name (for example, ADDRESS) rather than with the unique field name (ADDRESS#001_01) that Documaker uses. The extract file produced by the VRF2EXP utility generates records that contain the root field name for the field at offset 10, then multiple data values concatenated one after the other beginning at offset 58.
Adjust Data Offset Values for Block Tag Fields	Check this box if you want the system to increment the Offset field in the XDD to the next portion of the block tag's data.

Once you have selected the files you want to convert, click Next to go to the Convert Files and Finish window. See *Finishing a Conversion on page 449* for more information.

Documerge to Documaker Implementations

When the wizard encounters a block tag in a normalized file, it creates a field for each line of the block. Each field is given a name consisting of the tag name followed by:

- a space
- an octothorpe (#)
- a three-digit number that corresponds to the numerical sequence of that line in the block
- an underscore (_)
- a two-digit number, incremented for each block tag on the page that has the same name

When the wizard creates the Data Extract Dictionary (XDD) entry for this field, the renamed field, along with an offset value, is placed into the Data portion of the XDD record. If you are planning to use this XDD as part of a Documerge to Documaker Tier 2 implementation, the name placed into the Data portion of the record should be the original tag name, not the renamed field. To set the field name to the original tag name, check the Adjust field name values for the block tag fields field.

Additionally, when the wizard creates the XDD entry for this record, the Offset portion of the XDD record is assigned the value you provide in the Data Offset field on this page of the wizard. In a Documerge to Documaker Tier 2 implementation, the offset is usually 58, so this value is provided as the default.

In some Documerge to Documaker Tier 2 implementations, a single record in the extract file that this XDD entry is mapped to may contain values for all the lines of a block tag field rather than just the value for a particular line of the block tag field. If that is the case, you would want the Offset portion of the XDD record to be incremented (by the tag length) for each field corresponding to lines two and higher of the block tag. To make this adjustment, check the Adjust data offset values for a block tag field's field.

Note Studio saves your choices so they will be available the next time you run this conversion.

Understanding Font Mapping

Handling fonts is an important part of all conversion options available when you choose the Import Document Files option. When converting Word (.DOC or .DOCX) and Word Documaker document files from the Documaker Add-In for Word, Studio temporarily converts the source files into RTF format. While in RTF format, Studio compares the fonts used in the source document to fonts in the font cross-reference (FXR) file. To find the best match, Studio evaluates font attributes and calculates a score that reflects how well the attributes match. It then uses the font with the highest score.

Note If you choose the Rich Text (RTF) Files to Forms option, no initial conversion to RTF format is required, but the font matching process is the same.

The most important criteria in calculating the font score is the ability to display text in the RTF file. Matching the font name to the font family name is the second highest priority in calculating a font score. Matching the point size of the text with a font in the FXR is considered next. Matching the bold or italic attribute is one of the lowest priorities in calculating the font substitution score for a font.

Note If a font uses Unicode characters, a Unicode font in the FXR is always chosen over a non-Unicode font.

Here is a description of how the font matching process works:

For example purposes, the font Studio is trying to match from the RTF file will be called the *RTF font*. The fonts in the FXR will be called *FXR fonts*.

- Studio first compares the RTF font name to the font family name of the FXR font. If a full match is not found, partial matches are allowed for some fonts, such as Times and Courier.

You can specify matches in the RTFFontSubs control group. Here is an example of an entry:

```
< RTFFontSubs >  
    Arial = Albany AMT:UNIVERS:HELV
```

The name on the left is the facename of the RTF font. On the right, each matching FXR font's family name is separated with a colon (:). Preference is given to the names in the order they are listed.

You can specify a default font name in the INI file. If there is a tie for the best match, this font name is used if possible. Here is an example of how you could set this INI option:

```
< RTFDefaultFont >  
    RTFDefaultFont = Courier
```

If you omit this INI option or leave it blank, Studio uses Albany AMT as the default.

- Studio next evaluates font sizes and considers the difference between the font sizes. If the difference is too large — over 2 points in most cases— then a facename match is disregarded. This allows fonts with different names to be substituted if their font size is a closer match.
- Studio evaluates other attributes, such as if the font is bold or italic.

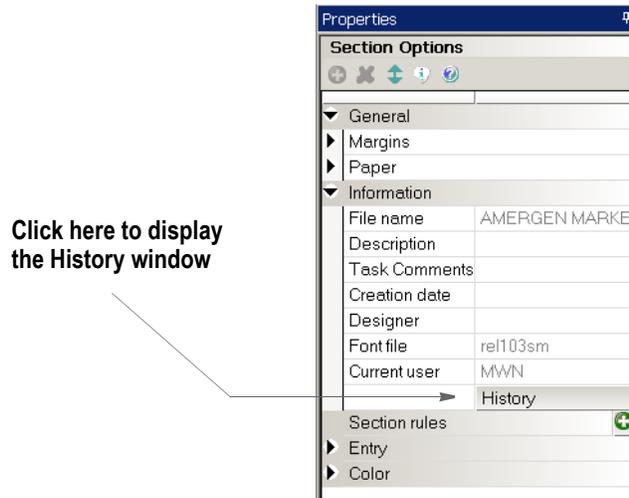
If there is no exact match between a font in the RTF file and a font in the FXR, Studio substitutes the font in the FXR that has the highest score for the font in the RTF file.

Note Studio places font substitution messages in the Output area and the trace file if you set the following options to Yes:

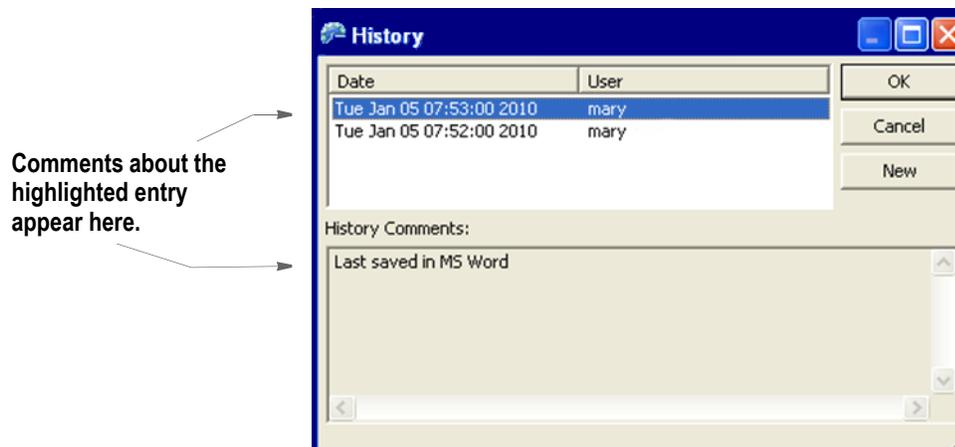
```
< Debug_Switches >  
    Enable_Debug_Options = Yes  
    Debug_FontSubs      = Yes
```

Capturing Information from Imported Word Files

When Studio imports a Word (DOC/DOCX or RTF) file, it captures any author, last author, creation date, last saved date, and application name information stored in the Word file (under File, Properties in Word). Studio attaches this information to the section or paragraph and stores it in history where you can view it from within Studio.



On the History window, information about the highlighted entry appears in the History Comments field.



Note This information is captured using the locale in use at the time of the import. If the content is later viewed by a user in a different locale, the information retains the original locale settings.

CONVERTING FILES INTO SECTIONS

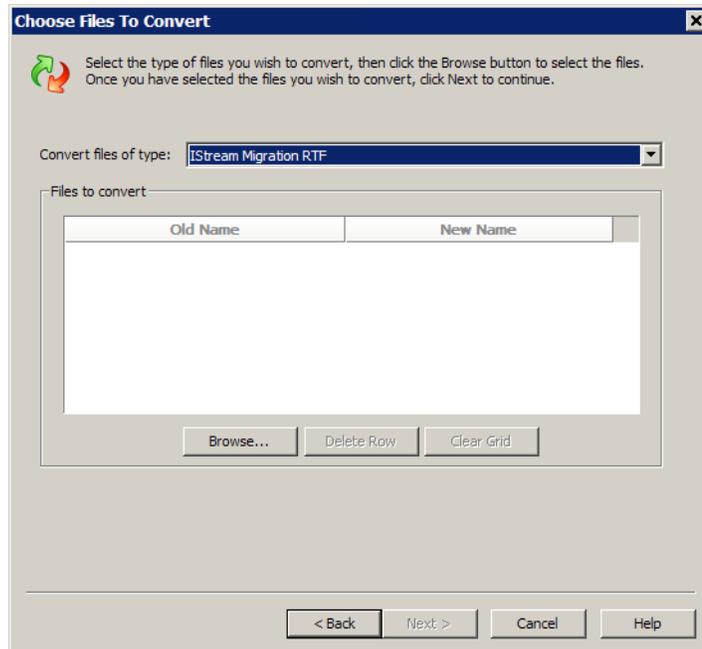
You can choose to convert to several file types including:

- IStream Migration (RTF) files
- PCL print files
- Metacode Form (FRM) files
- Metacode files
- Normalized Metacode files
- AFP print files
- Normalized AFP files
- PDF files
- TIF files
- DCD files
- Compuset files

Note There are several utilities you can use to batch convert files into FAP files, such as MET2FAP (Metacode to FAP), DCD2FAP (DCD to FAP), RTF2FAP (RTF to FAP and FOR) and CSET2FAP (CompuSet to FAP). See the [Docutoolbox Reference](#) for more information.

Follow these steps to convert files into sections:

1. Choose the Convert Files to Sections (FAPs) option on the Choose the Conversion Type window. The Choose Files to Convert window appears.



2. Select the type of conversion you want, then click Browse to select the files you want to convert.

Note Some conversions, such as IStream Migration and PDF conversions, offer additional options specific to that conversion. Refer to these topics or the Help for additional information:

- *Converting IStream Migration RTF Files* on page 423
 - *Converting PDF Files* on page 425
-

3. Once you have selected the files you want to convert and set any applicable options, click Next to go to the Convert Files and Finish window. See *Finishing a Conversion* on page 449 for more information.

Converting IStream Migration RTF Files

You can use the IStream Migration Utility to convert IStream model documents into RTF files. You can then use Studio's option to import an IStream Migration RTF file to convert those RTF files into sections. During this conversion, Studio automatically sets the size of the section, recognizes tags as variable fields from the RTF, and creates XDD entries. Studio also imports property information for these fields from the FDB, if available.

Note You can also use the RTF2FAP utility to convert IStream Migration RTF files. For more information on the RTF2FAP utility and the IStream Migration Utility, see the [Docutoolbox Reference](#).

During the conversion, Studio...

- Creates a section to match the text in the IStream RTF file.
- Sizes the resulting section to the height of the text.
- Recognizes text surrounded by brackets (<>) as fields. Duplicate field names are not permitted in a single section. Therefore, if duplicates are found in the IStream Migration RTF file, Studio appends a suffix to the duplicate field names to make them unique. If the brackets are in a different font than the text, the text inside the brackets are not recognized as a field.
- Looks up the field in the FDB and gets the field length. Sets the type to alphanumeric. If the field is not found, Studio uses the default length set for the conversion and assigns a type based on tag name prefix:

If the tag name begins with Studio assigns it this type

C	Character fields (field size 1024 characters)
D	Date fields (field size 18 characters)
I	Indicator fields (field size one character)
N	Numeric fields (field size 15 characters)

If the first character is not C, D, N or I, the default value is used when creating the FDB entry.

Note For field length, Studio first looks in the FDB. It next looks at the first character of the field's name and uses that character as a guide to the field type and length. If the field is not in the FDB and its name does not start with C, D, I, or N, Studio uses the default field length you specified in the Conversion wizard or via this INI option:

```
< RTFImport >
  DefaultFieldLength =
```

The default is 50.

- Creates task comments for any expressions surrounded by brackets (<>). If an expression or field has an invalid name, Studio substitutes a generic name (*FIELD*) and includes a task comment as a reminder to you to follow-up with the needed information. You should convert these expressions into DAL script functions.

After you select the IStream Migration RTF files you want to convert, this Conversion Options window appears:

Enter the applicable information in these fields.

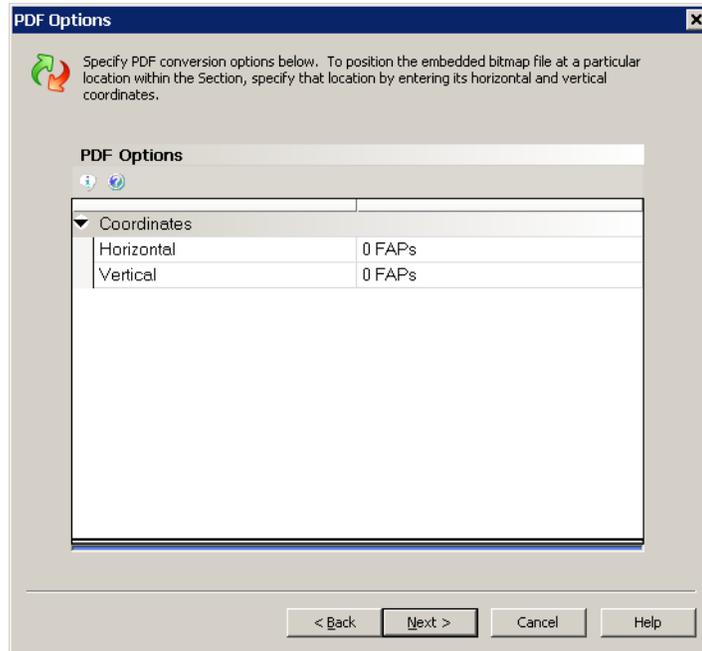
Field	Description										
Import Options											
Default Field Length	Choose the field length you want Studio to assign to the fields it finds during the conversion. The default is 50 characters.										
Default Field Scope	Choose the field scope you want Studio to assign to the fields it finds during the conversion. You can choose from these options: <ul style="list-style-type: none"> • Section – data is only written to the same named variable field within a section. • Form – data is written to the same named variable field throughout all sections within the form. • Global – data is written to the same named variable field throughout all sections and all forms in the form set. The default is Section.										
File Options											
Create Separate Graphics	Enter Yes if you want Studio to create separate graphic (LOG) files and assign a name based on the name of the input document and the separator you specify in the Name Separator field. Enter No if you want Studio to embed any graphics into the section. You can later remove embedded graphics by editing the file and replacing the graphic or unchecking the embedded option for the graphic. The default is Yes.										
Name Separator	Enter the character you want Studio to insert between elements of the file name. The separator can be nothing, a blank, an underscore, or any other valid Windows file name character. For example, if the name of the input document is Declaration and you choose to create separate sections you would get these results for page 1: <table border="1" data-bbox="690 1102 1128 1270"> <thead> <tr> <th>Separator</th> <th>Result</th> </tr> </thead> <tbody> <tr> <td>(nothing)</td> <td>Declaration1.FAP</td> </tr> <tr> <td>Blank</td> <td>Declaration1 .FAP</td> </tr> <tr> <td>_</td> <td>Declaration_1.FAP</td> </tr> <tr> <td>#</td> <td>Declaration#1.FAP</td> </tr> </tbody> </table>	Separator	Result	(nothing)	Declaration1.FAP	Blank	Declaration1 .FAP	_	Declaration_1.FAP	#	Declaration#1.FAP
Separator	Result										
(nothing)	Declaration1.FAP										
Blank	Declaration1 .FAP										
_	Declaration_1.FAP										
#	Declaration#1.FAP										

Once you have set the conversion options, click Next to go to the Convert Files and Finish window. See *Finishing a Conversion on page 449* for more information.

Note You can find additional information about converting IStream model document files in the Oracle IStream Migration Utility User Guide.

Converting PDF Files

When you convert PDF files, Studio converts the PDF file into an embedded graphic and places that graphic in a section. This Conversion Options window appears to let you specify how to position the embedded graphic within the section:



Field	Description
Horizontal	Enter the distance you want Studio to move the embedded graphic from the left edge of the section. For instance, if you are using FAP units as your unit of measure and you want the embedded graphic placed 1/4 inch from the left edge of the section, you would enter 600.
Vertical	Enter the distance you want Studio to move the embedded graphic from the top edge of the section. For instance, if you are using FAP units as your unit of measure and you want the embedded graphic placed 1/4 inch from the top edge of the section, you would enter 600.

Once you have set these conversion options, click Next to go to the Convert Files and Finish window. See *Finishing a Conversion on page 449* for more information.

COMPILING SECTIONS INTO PRINT FILES

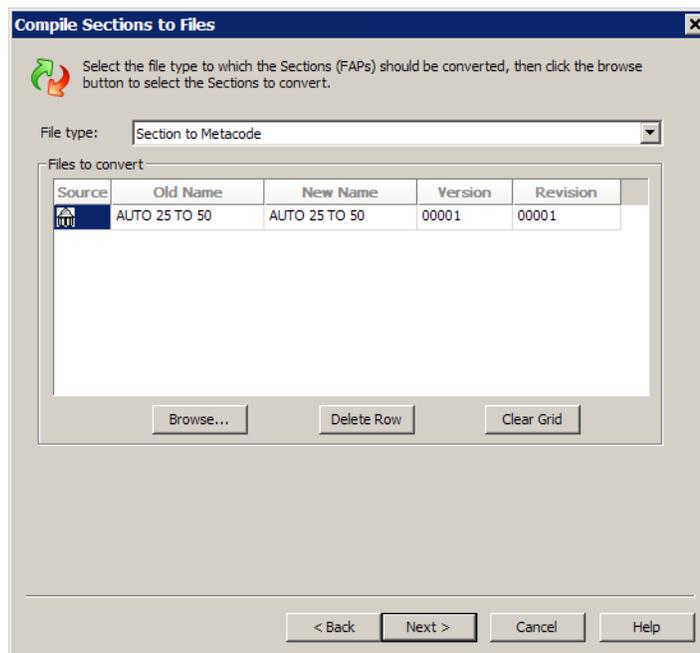
You can choose to compile to several types of print files, including:

- AFP print files
- PCL print files
- Metacode files
- Postscript files
- VIPP files
- FRM file
- RTF files

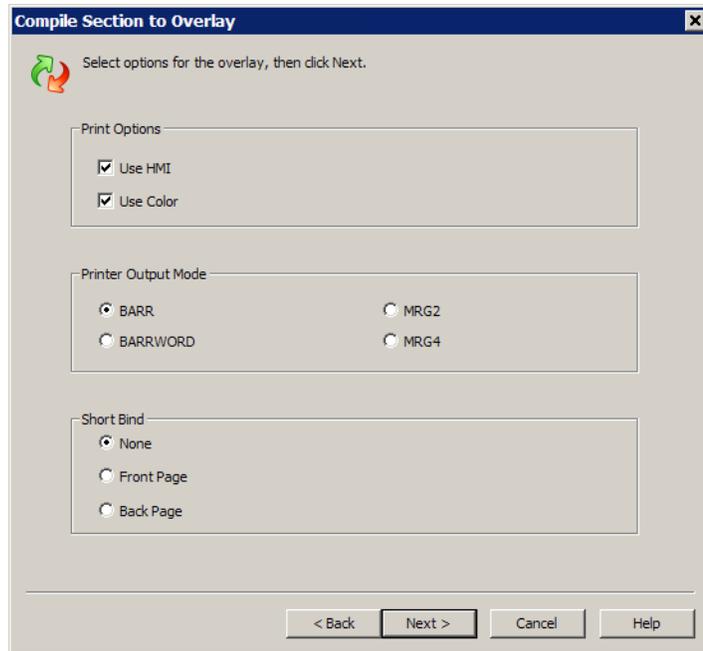
Note Variable Data Intelligent PostScript PrintWare (VIPP) was created by Xerox to enable high-performance variable data printing on PostScript devices.

Follow these steps to convert sections into print files:

1. Choose the Compile Sections to Print Files option on the Choose the Conversion Type window. The Compile Sections to Files window appears.



2. Once you have selected the files you want to change, click Next. The Compile Section to Overlay window appears.



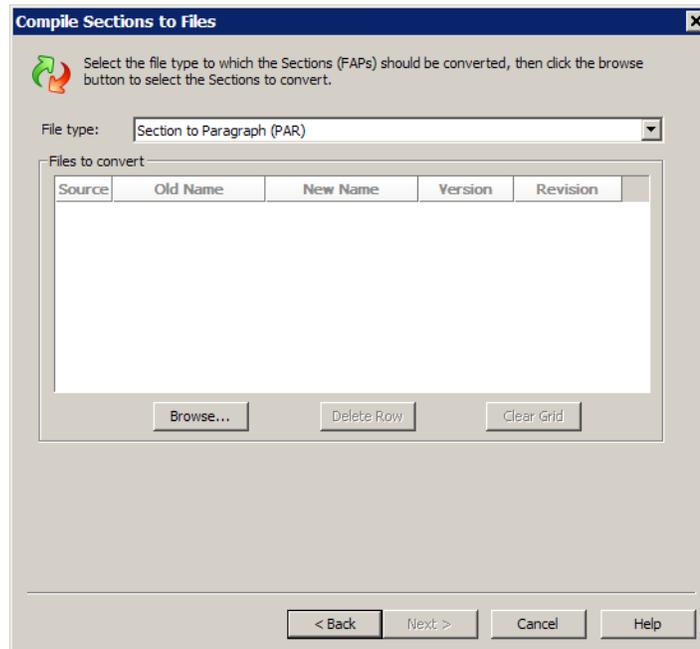
3. Select the options you want, then click Next to go to the Convert Files and Finish window. See *Finishing a Conversion on page 449* for more information.

CONVERTING SECTIONS INTO PARAGRAPHS

Use this option to convert sections into paragraph (PAR) files.

To convert sections into paragraphs, follow these steps:

1. Choose the Convert Sections to Paragraph Files option on the Choose the Conversion Type window. The Compile Sections to Files window appears.



2. Click Browse to select the files you want to convert.
3. Once you have selected the sections you want to change into paragraphs, click Next to go to the Convert Files and Finish window. See *Finishing a Conversion on page 449* for more information.

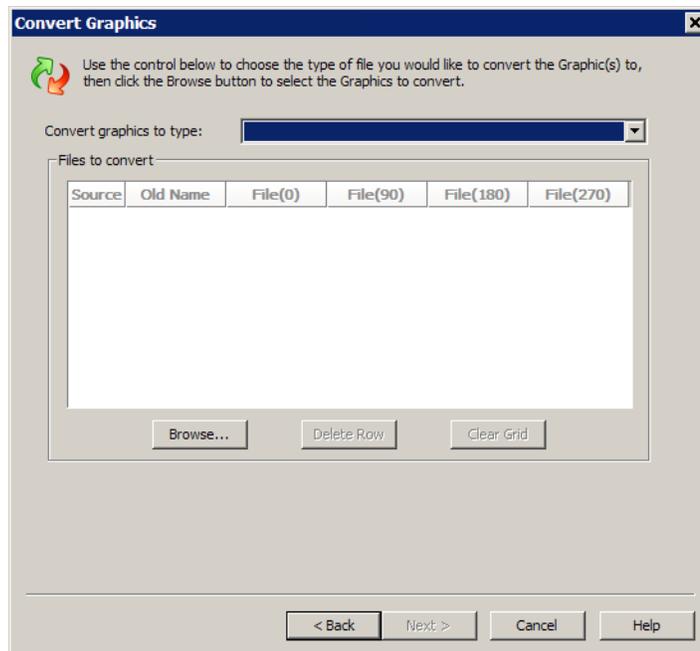
CONVERTING GRAPHIC FILES

You can choose to convert to several file types including:

- Standard Documaker graphic files (LOG)
- JPEG files (JPG)
- Segmented graphics for AFP printers (SEG)
- VIPP image (VPP)
- Xerox image files (IMG)
- Xerox font files (FNT)
- Bitmap files (BMP)
- TIFF files (TIF)

To convert graphic files, follow these steps:

1. Choose the Convert Graphic Files option on the Choose the Conversion Type window. The Convert Graphics window appears.



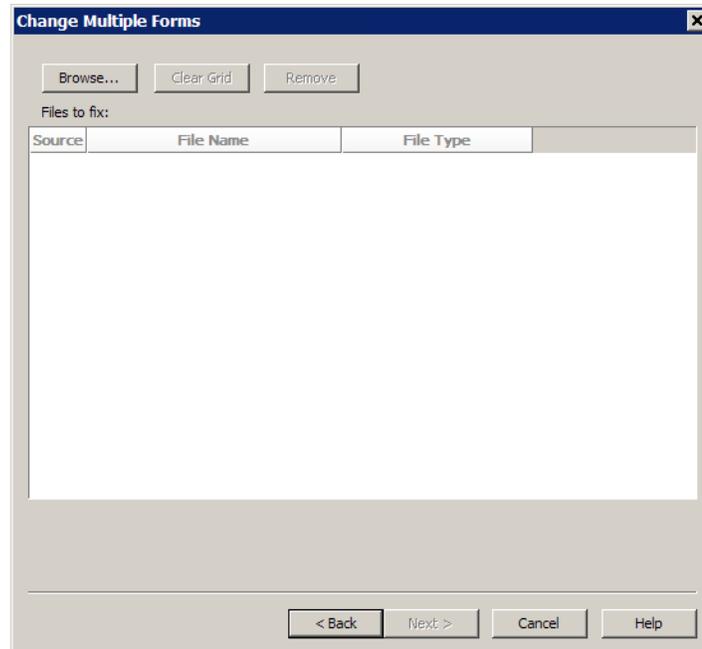
2. Select the graphic type you want to convert to, then click Browse to select the files you want to convert.
3. Once you have selected the files you want to change, click Next to go to the Convert Files and Finish window. See *Finishing a Conversion on page 449* for more information.

CHANGING TEMPLATES

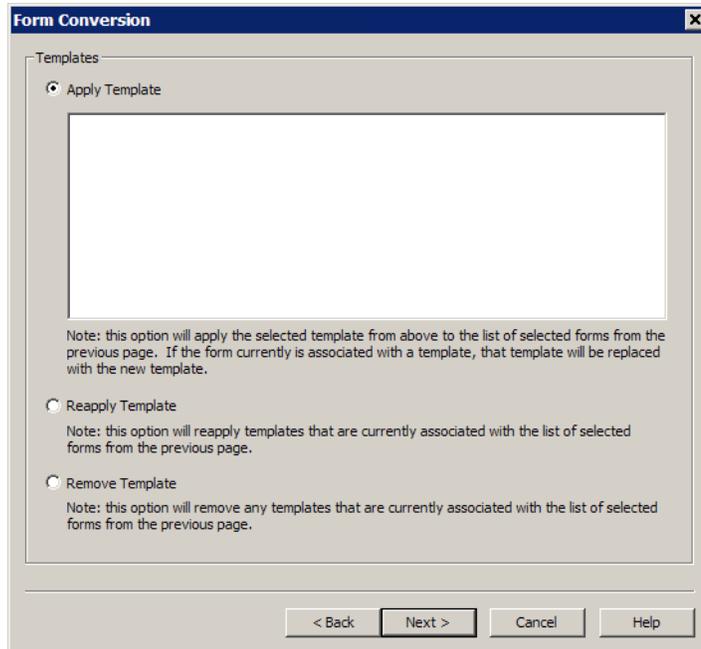
Use the Change Multiple Forms option to assign, reapply, or remove templates from the forms you specify. During these changes, Studio checks the forms out of the library, makes the changes you indicate, and then checks the forms back into the library.

To change form template assignments, follow these steps:

1. Choose the Change Multiple Forms option on the Choose the Conversion Type window. The Change Multiple Forms window appears.



2. Click Browse to select the forms you want to change, then click Next. The Form Conversion window appears.



3. Select from these options and click Next:

Option	Description
Apply Template	Apply the selected template to the forms you selected. If one of the forms already has a template assigned, Studio replaces that template with the one you chose here.
Reapply Template	Reapplies the selected template to the forms you selected. You would choose this option, for instance, if you have made changes to your template and you want to update your forms to incorporate those changes.
Remove Template	Removes any template assignments to the forms you selected.

4. Once you have selected the options you want, click Next to go to the Convert Files and Finish window. See *Finishing a Conversion on page 449* for more information.

MAKING CHANGES TO MULTIPLE SECTIONS

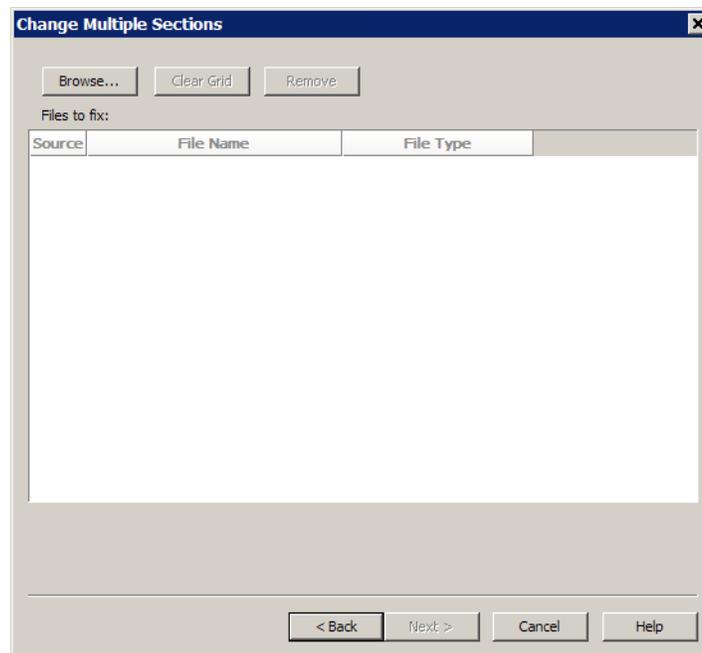
You can make numerous changes to a group of section (FAP) files. For instance, you could use this option to convert FAP files created with the DAP Development System into newer FAP file versions. You can also use this option to make global changes to a group of FAP files you select.

For instance, you can use this option to:

- Change fonts
- Change field names
- Recalculate font coordinates
- Replace specific characters
- Update FAP files from the Field Database

Follow these steps to convert files into sections:

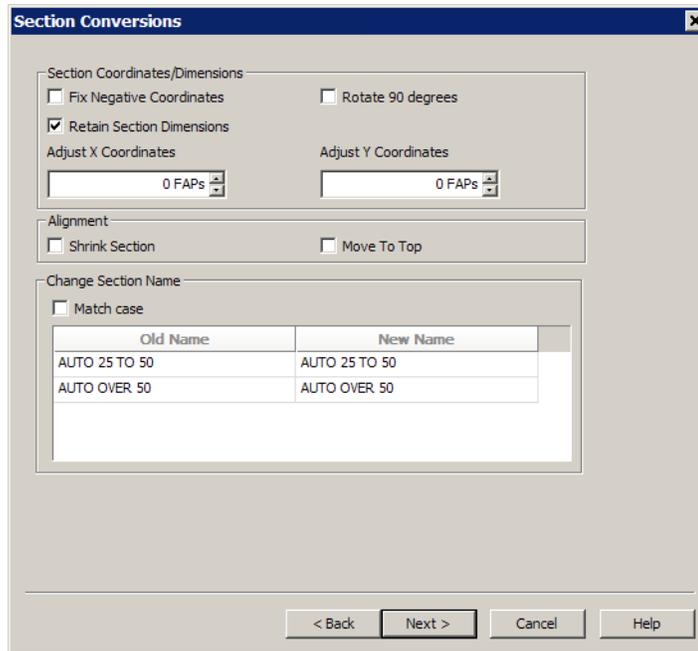
1. Choose the Change Multiple Sections option on the Choose the Conversion Type window. The Change Multiple Sections window appears.



2. Click Browse to find and select the sections you want to run this conversion on. Use the Ctrl key to select multiple sections. Click Next to move to the Section Conversions window.

Section Conversions

The Section Conversions window lets you define changes to apply to the sections you selected.



Use these fields to indicate changes you want made to the sections:

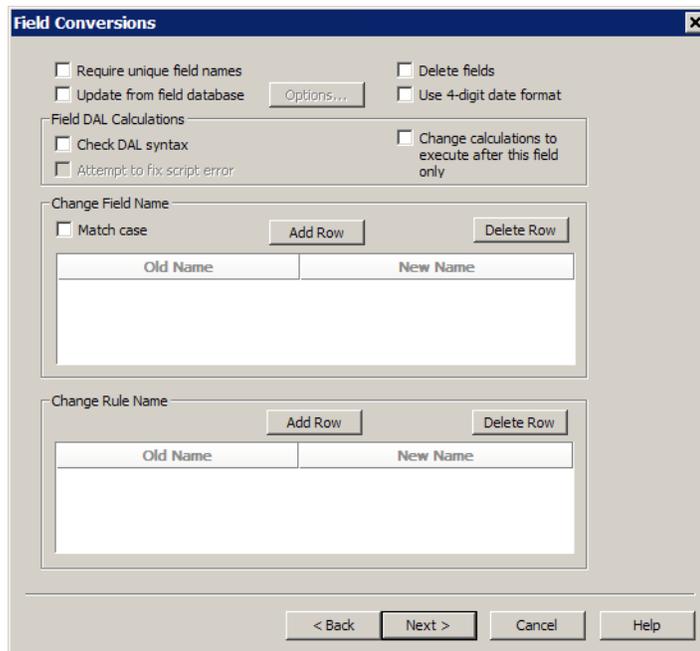
To	Then
Fix Negative Coordinates	Check the Fix Negative Coordinates field. Negative coordinates are not produced by Studio, but some users have encountered them when converting print streams.
Retain Section Dimensions	Studio increases a section's horizontal or vertical dimensions if you place objects outside the section's borders. To prevent this from happening, check the Retain Section Dimensions field.
Rotate 90 Degrees	Check to rotate the section 90 degrees.
Adjust X Coordinates	Enter the appropriate value in FAP units into the Adjust X Coordinates field. FAP units are 2400 units per inch. For instance, to adjust by 1.5 inches, you would enter 3600 in this field.
Adjust Y Coordinates	Enter the appropriate value in FAP units into the Adjust Y Coordinates field. FAP units are 2400 units per inch. For instance, to adjust by 1.5 inches, you would enter 3600 in this field.
Shrink Section	Check to shrink the section bottom coordinates to the bottom of the lowest object.
Move to Top	Check to move all object's coordinates to the top of the section to eliminate white space.
Change Section Name	Enter the current name of the FAP file in the Old Name field and the name you want to assign in the New Name field.

Note Check the Match Case field if you want Studio to consider upper- and lowercase letters when it searches for the FAP files to change.

Once you have selected the options you want, click Next to go to the Field Conversions window.

Field Conversions

The Field Conversions window lets you make changes that apply to several fields at one time.



Use these fields to indicate changes you want made to the fields on the sections:

Field	Description
Require Unique Field Names	Check this field if you want Studio to check for duplicate field names and warn you if any are found. If Studio finds duplicate fields, it appends an octothorp (#) followed by a number such as 001 to the field name.
Update from Field Database	Check this field if you want Studio to retrieve all of the information stored in the field database except for the font ID.
Delete Fields	Check this field if you want Studio to delete all variable fields from the sections you selected. Studio does not delete the fields from the Field Database. You can use this option to create overlays.
Use 4-digit Date Format	Check this field to change all date fields on the selected sections to use 4-digit years. Studio adjusts the fields input length if necessary, but makes no other changes to the field.
Check DAL Syntax	Check the Change DAL calculations to DAL scripts field.
Attempt to Fix Script Error	Check this field if you want Studio to fix common DAL script errors.

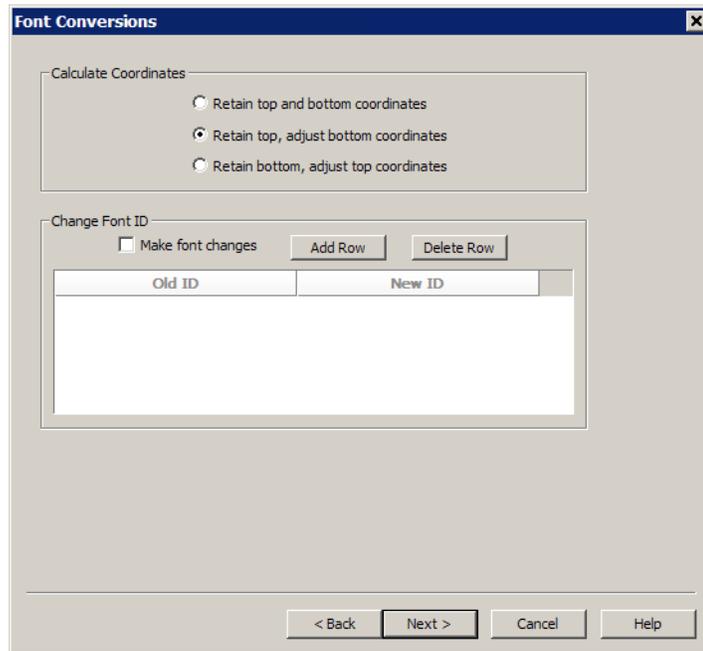
Field	Description
Change Calculations to Execute after this Field Only	Check this field if you want the system to modify the DAL calculations so that they only execute after the current field.
Change Field Names	Enter the current name of the field in the Old Name field and the name you wish to change to in the New Name field.

Note Check the Match Case field if you want Studio to consider upper- and lowercase letters when it searches for the field names to change.

Once you have selected the options you want, click Next to go to the Font Conversions window.

Font Conversions

The Font Conversions window lets you make changes that apply to several fonts at one time.



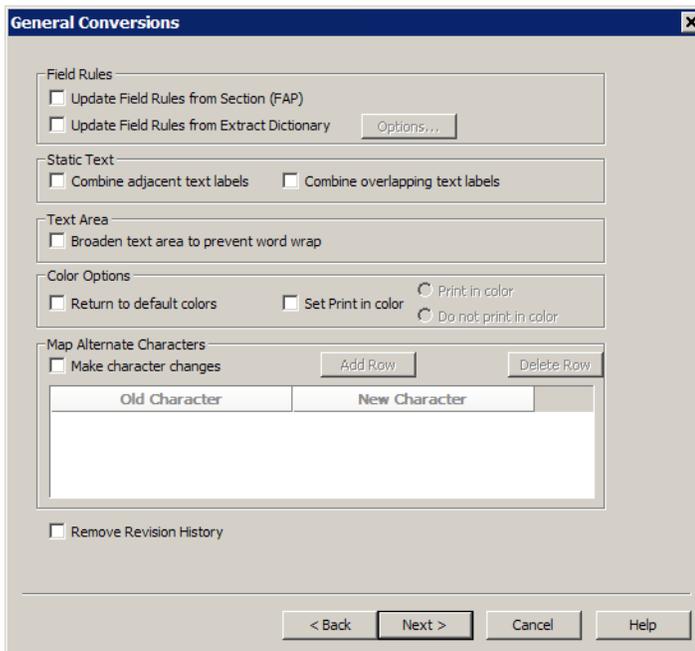
Use these fields to indicate changes you want made to the fonts used in the sections:

Field	Description
Calculate Coordinates	To have Studio recalculate font coordinates, click the appropriate option. Click Retain Top and Bottom Coordinates if you do not want Studio to recalculate font coordinates. Click Retain Top, Adjust Bottom Coordinates if you want to keep the top coordinates. Click Retain Bottom, Adjust Top Coordinates if you want the text would retain its current baselines.
Change Font ID	Enter the current ID of the font in the Old ID field and the ID you want to change to in the New ID field.

Once you have selected the options you want, Click Next to go to the General Conversions window.

General Conversions

Use the General Conversions window to make general changes to the sections you selected.



Use these fields to indicate general changes you want made in the sections:

Field	Description
Update Field Rules from Section (FAP)	Click this field to update the field rules based on the rule assignments in the section.
Update Field Rules from Extract Dictionary	<p>Click this field then click the Options button to display the Update Options from Extract Dictionary window. This window lets you further define how to update field rules based on the contents of your extract dictionary.</p> <p>For instance, you can choose to use the extract dictionary rule or update various parts of the rule definition, such as the offset, length, mask, rule assignment, data, or whether it is required.</p>

Field	Description
Combine Adjacent Text Labels	<p>Check the Combine Adjacent Text Labels field to combine text labels that are very close into one label. Studio combines text labels if the text labels use the same font and have the same baseline. If the text areas are located closer than half the width of a space character, Studio joins the labels without a space.</p> <p>If the labels are more than half the width of a space character apart, but less than the width of a space character, Studio joins the labels with a space between them. If the labels are located more than a space apart, Studio does not combine them.</p>
Combine Overlapping Text Labels	<p>Occasionally, conversions from print streams produce text labels which overlap. Check the Combine Overlapping Text Labels field to have Studio correct these kinds of problems.</p> <p>Note: When working with a section, you can use the Format, Convert to text label or Convert to text area options to combine text areas and labels you select on a specific section. This option is more often used to when print files are converted to FAP files and each character is turned into a separate label.</p>
Broaden Text Area to Prevent Wrapping	Check this option to enlarge your text areas to prevent the text from wrapping.
Return to Default Colors	Check the Return to Default Colors field to reset all colors on the selected sections to the default colors.
Print in Color	Check the Print in Color field if you want Studio to make all colors printable for the sections you selected. This option, in effect, checks the Print in Color field on each objects Color Selection window.
Map Alternate Characters	<p>Enter the numeric value (0-255) of the current character of the font in the Old Character field and the numeric value of the character you want to change to in the New Character field. For instance, using code page 1004, if you want to change a to ä, you would enter 132 in the Old Character field and 133 in the New Character field.</p> <p>Note: You can test the characters by pressing the ALT key and typing the numeric value using the numeric keypad.</p> <p>Typically, you would use this option to replace a character which exists in one code page, but not in another. For instance, some code pages include typographical quotation marks (sometimes called smart quotes) while others do not. You can use this option to replace the typographical quotation marks with standard quotation marks (").</p>
Remove Revision History	Click this box to delete all revision history.

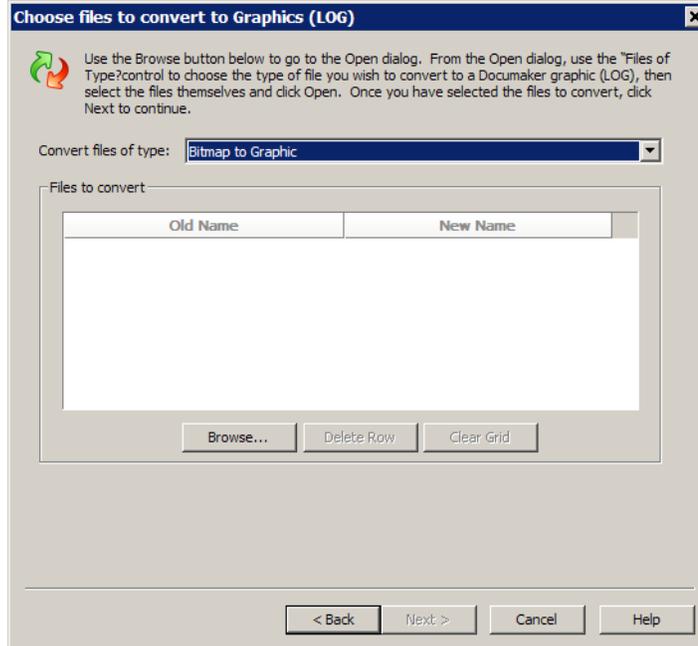
Once you have selected the options you want, click Next to go to the Convert Files and Finish window. See *Finishing a Conversion on page 449* for more information.

CONVERTING BITMAP FILES INTO LOG FILES

Use this conversion option to convert bitmap (BMP) files into Documaker graphic format (LOG) files.

Follow these steps to convert bitmap files into LOG files:

1. Choose the Convert Bitmap Files to Graphic Files option on the Choose the Conversion Type window. The Choose Files to Convert to Graphics (LOG) window appears:



2. Click Browse to select the files to run the conversion on.
3. Once you have selected the files you want to change, click Next to go to the Convert Files and Finish window. See *Finishing a Conversion on page 449* for more information.

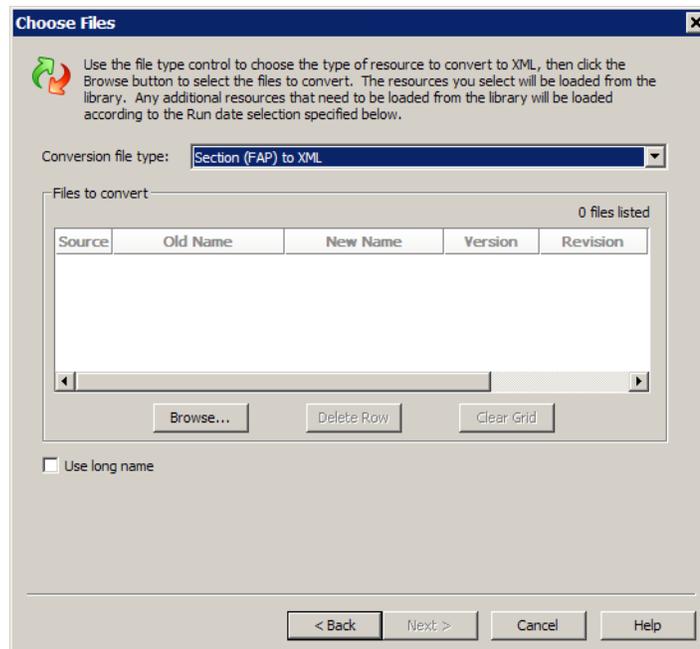
CONVERTING SYSTEM FILES INTO XML FILES

Use this option to convert the following Documaker system files into XML files:

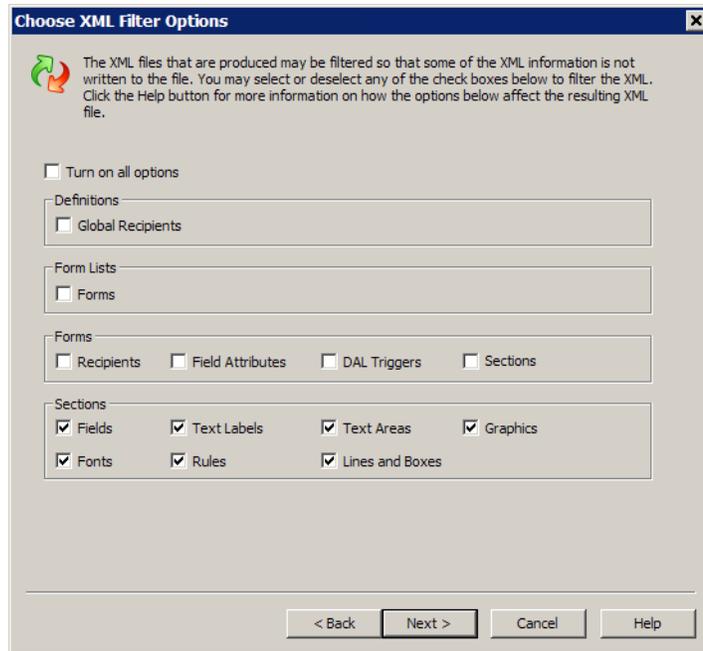
- Application Definition (BDF) files
- Form List (GRP) files
- Form (FOR) files
- Section (FAP) files
- Template (TPL) files

Follow these steps to convert system files into XML files:

1. Choose the Convert Files to XML Files option on the Choose the Conversion Type window. The Choose Files window appears



2. Select the type of conversion you want to run, then click Browse to select the files to run the conversion on.
3. Once you have selected the files you want to change, click Next. The Choose XML Filter Options window appears:



4. Use this window to add or omit elements you want in the XML file, then click Next.
5. Click Next to go to the Convert Files and Finish window. See *Finishing a Conversion on page 449* for more information.

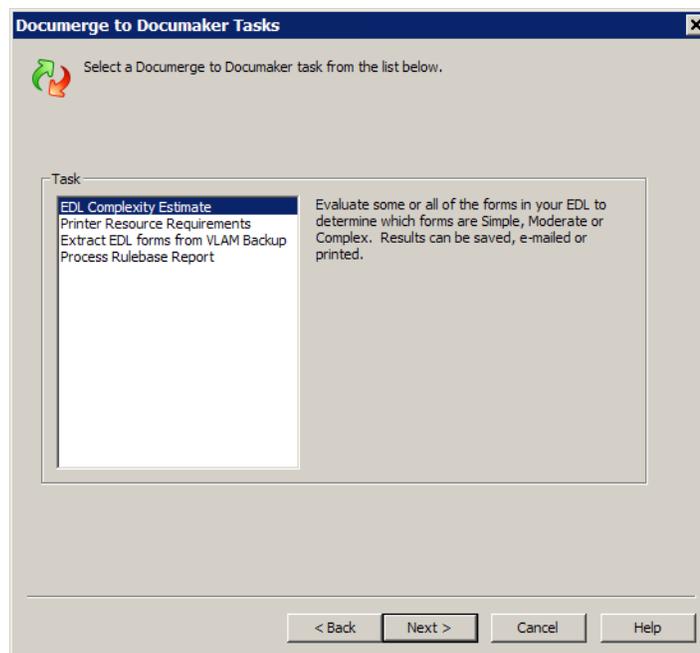
CONVERTING DOCUMERGE RESOURCES

Use this option to convert Documerge resources into Documaker resources. You can extract Documerge forms from a VLAM EDL backup file, determine which printer resources the forms use, and convert the forms into Documaker section (FAP) and form (FOR) files.

You can also use the Rulebase Report to transfer form sort order and recipient information from Documerge to Documaker.

Follow these steps to convert Documerge resources into Documaker resources:

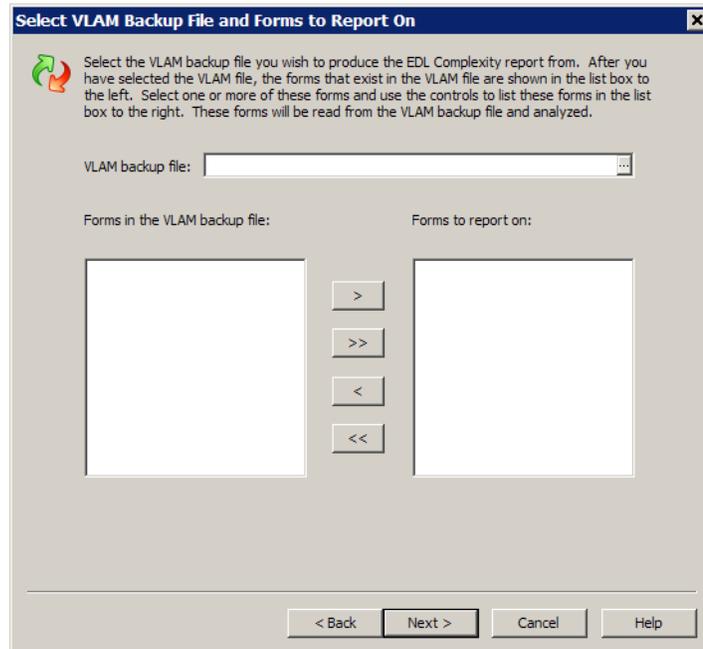
1. Choose the Documerge to Documaker Tasks option on the Choose the Conversion Type window. The Documerge to Documaker Tasks window appears



2. Select the task you want to perform. You can choose from these tasks:

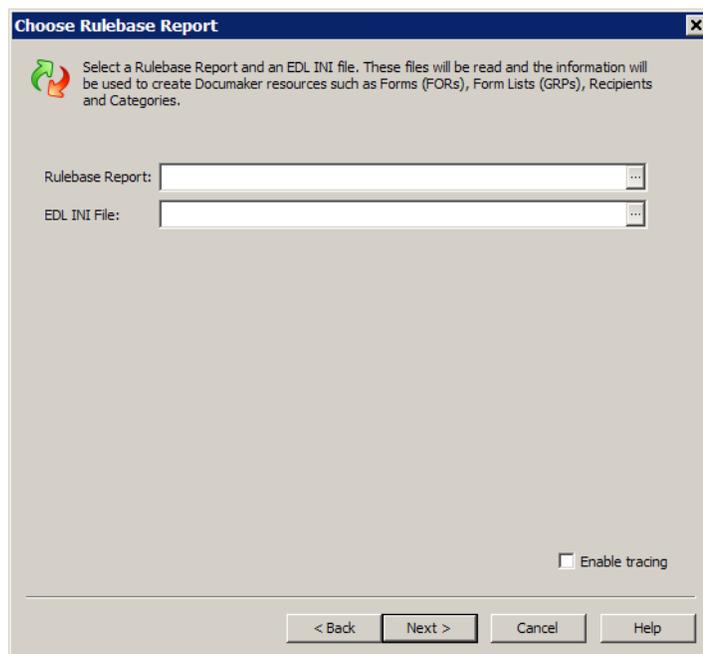
Task	Description
EDL Complexity Estimate	Choose this task to evaluate the forms in your EDL to determine which forms are simple, moderate, or complex. You can save, email, or print the results.
Printer Resource Requirements	Choose this task to create a report that shows which AFP or Metacode printer resources are required to convert your AFP or Metacode forms into Documaker section (FAP) files. These resources can include fonts, IMG files, FRM files, page segments, and so on.
Extract EDL Forms from VLAM Backup	Choose this task to extract EDL forms from a VLAM backup file. The forms can be Metacode, AFP, or DCD forms. Studio extracts the forms and places them in the location you specify. You can then convert the result into section (FAP) files.
Process Rulebase Report	Choose this task to have Studio use a Documerge Rulebase Report to create Documaker resources, such as forms, form lists, recipients, and categories.

Click Next. If you choose any of the first three tasks, the Select VLAM Backup File and Forms to Report On window appears.



Click Browse to select the VLAM backup file, then choose the forms you want to report on.

If you chose the Process Rulebase Report task, the Choose Rulebase Report window appears.



Click Browse to select the Rulebase report you want to process and the EDL INI file to use.

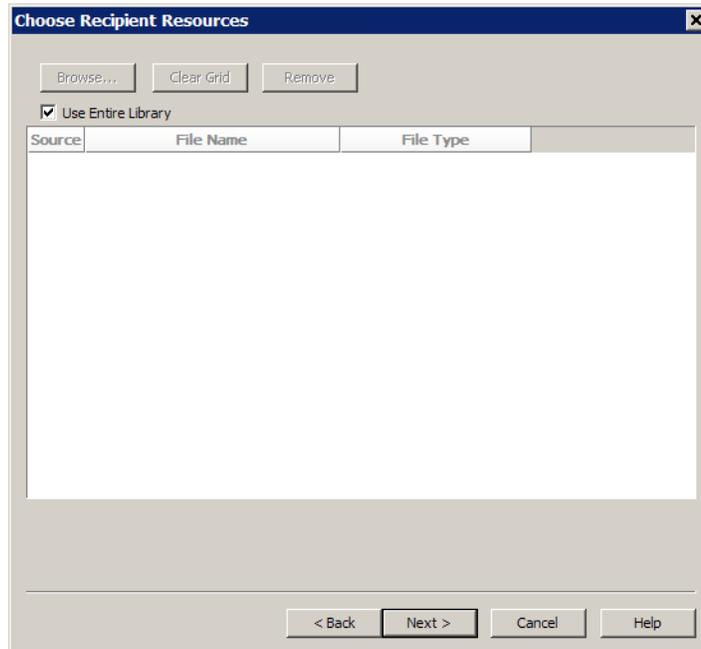
3. Once you have finished, click Next to go to the Convert Files and Finish window. See *Finishing a Conversion on page 449* for more information.

CHANGING RECIPIENT INFORMATION

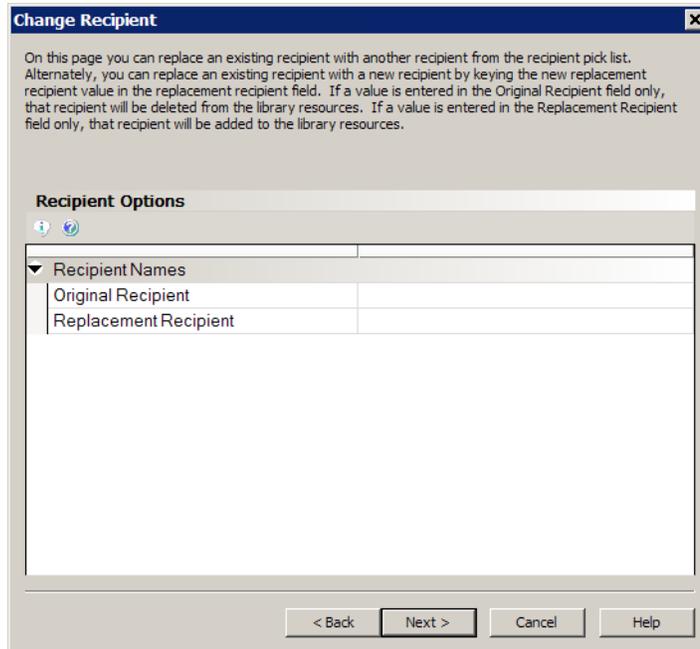
Use this option to change recipient information in the resources you specify.

To change recipient information, follow these steps:

1. Choose the Change Recipients option on the Choose the Conversion Type window. The Choose Recipient Resources window appears.



2. To change recipient information throughout the entire library, just click Next. If you want to choose specific application definition (BDF) files, remove the check from the Use Entire Library field, then click Browse to select the BDF files.
3. Click next when finished. The Change Recipient window appears.



4. You have these choices on the Change Recipient window:

To	Then
Substitute one recipient for another	Select the recipient you want to replace in the Original Recipient field. Then select the recipient you want to change to in the Replacement Recipient field.
Replace one recipient with a new recipient	Select the recipient you want to replace in the Original Recipient field. Then enter the name of the new recipient in the Replacement Recipient field.
Delete a recipient from the library	Enter the name of that recipient in the Original Recipient field. Leave the Replacement Recipient field blank.
Add a recipient to the library	Enter the name of the new recipient in the Replacement Recipient field. Leave the Original Recipient field blank.

Note When you make an entry in the Replacement Recipient field, Studio adds a Copy Count field so you can enter the appropriate copy count.

5. Once you have selected the recipients, click Next to go to the Convert Files and Finish window. See *Finishing a Conversion on page 449* for more information.

IMPORTING XML SCHEMAS INTO THE XDD

You can import an XML schema (XSD) into an extract data dictionary (XDD). From the schema, the XDD recognizes parent and children elements and also inherits any attributes as children.

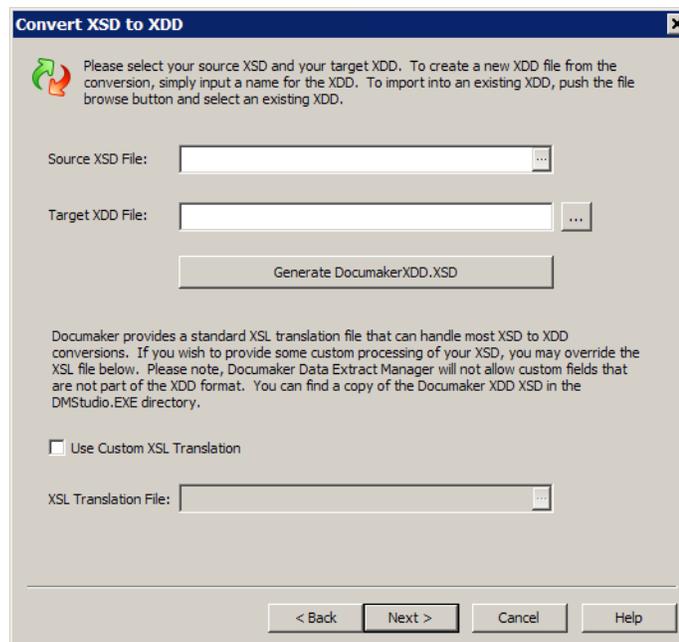
Importing an XML schema to populate the XDD provides these benefits:

- Lets you build the XDD before you have sample XML input data
- Helps reduce errors when creating the XDD
- Ensures compatibility with the input file structure
- Makes it easier to synchronize changes to the input file schema
- Helps reduce mapping errors as XPath expressions are easily derived from XML schema

Note For the conversion, Studio uses the XSL translation contained in the XSD2XDD.XSL file. This file is located in the directory that contains the DMStudio.exe file, usually c:\fap\dl. You can customize this file if needed to handle complex source XSD files, but your changes must conform to Documaker's XDD standards.

To import an XML schema, follow these steps:

1. Choose the Convert an XSD File to a Data Extract option on the Choose the Conversion Type window. The Convert XSD to XDD window appears.



2. Click Browse to select the source XSD file. Enter the name you want to assign to the target XDD or click Browse to select an XDD file. Studio lets you use a custom XSL translation file if necessary. Click Next when you finish specifying the files to use.
3. The Convert Files and Finish window appears. See *Finishing a Conversion on page 449* for more information.

Keep in mind...

- Studio imports all XDD fields as strings.
- Studio assigns the Move_It rule to these fields in the XDD
- Documaker does not validate the XSD file during processing

FINISHING A CONVERSION

On the Convert Files and Finish window you can choose to check files into the workspace library or place them into a directory.

By default, Studio checks the new files into the library for you. You can change this option if necessary. On this window you can also make or change the entries in the following fields:

Field	Description
Effective Date	Typically you would accept the default effective date. The default effective date is set up when the workspace is created. You can enter another date for the converted files if necessary. Documaker uses the effective date to determine which version/revision combination of a resource is selected at processing time. The effective date of a selected resource is always less than or equal to the Documaker processing date.
Description	(Optional) Change the default description if necessary. You can enter up to 100 characters.
Mode	(Optional) Enter a mode if you are using this library feature and you want it to apply to all converted resources. Mode cannot be used if the Projects option is activated for this workspace.
Status	(Optional) Enter a status if you are using this library feature and you want it to apply to all converted resources. Status cannot be used if the Projects option is activated for this workspace.
Class	If the Projects option is activated for this workspace, select a class code or blank from the list. If the Projects option is not turned on, optionally enter a class code if you are using this library feature and you want to apply it to all converted resources.

You define modes, statuses, classes, and projects using the Manage, Settings options. For instance, you could set up modes to denote milestones in the development process such as Development, Testing, and Production. For more information, see *Defining Mode, Status, Class, and Project Options on page 380*.

Field	Description
Project	If the Projects option is turned on for this workspace, select a project code or a value of blank from the list. If the Projects option is not turned on, optionally enter a project code if you are using this library feature and you want to apply it to all converted resources.
Save Files to Disk at Location	This field is only available when the Check File into Library box is not selected. Files from a conversion that could be saved to this location are FAP files (sections) and LOG files (graphics).
Form (FOR)	This option is only available when the Check File into Library box is not selected. Files from a conversion that could be saved to this location are FOR files (forms).
Graphic (LOG)	This option is only available when the Check File into Library box is not selected. Files from a conversion that could be saved to this location are LOG files (graphics). Use this option to specify the location for LOG files when LOG files are not stored in the same folder as FAP files (sections).
Save the FXR	This option is only available when a normalized AFP or Metacode file, where the fonts are embedded in a print stream, is being converted.

You define modes, statuses, classes, and projects using the Manage, Settings options. For instance, you could set up modes to denote milestones in the development process such as Development, Testing, and Production. For more information, see *Defining Mode, Status, Class, and Project Options on page 380*.

Click Finish to convert the files.

Chapter 13

Producing Reports

The Reports option in the Workspace tree lets you print a variety of system reports. A wizard guides you through the process of selecting the information you want to appear on the report.

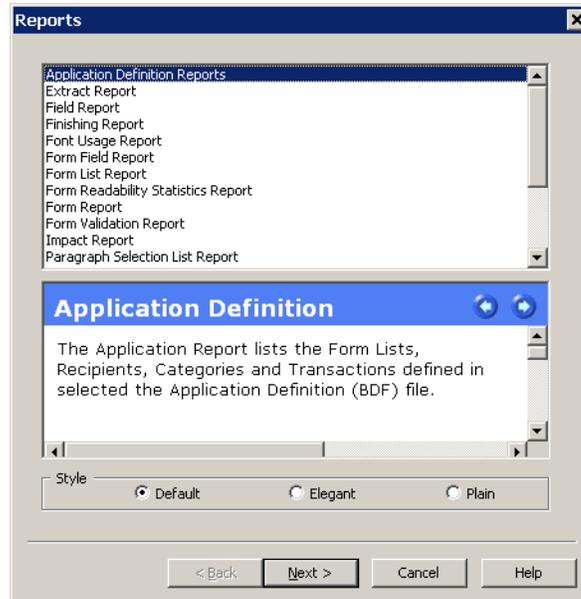
Studio displays the result and gives you the option of viewing it on your screen, sending it to your printer, saving it in HTML format, or emailing the report.

This chapter includes the following topics:

- *Overview* on page 452
- *Printing Reports* on page 455
- *Printing a Finishing Report* on page 461
- *Printing the Documerge Field Report* on page 465

OVERVIEW

When you choose Reports from the Workspace tree, the Reports window appears, as shown here.



You can choose from these reports:

Report	Description
Application Definition Report	This report lists the form lists, recipients, categories and transactions defined in the Application Definition (BDF) file.
Extract Report	This report provides information about the records contained in the extract dictionary (XDD) file. This includes the level, name, offset, length, rule, format, FDB, and data. This report compares the record names in the specified extract file against the record names in the workspaces' FDB file and reports whether the record exists in the FDB.
Field Report	This report provides information about the records contained in the common field dictionary. This includes the field's name, type, format, font ID, length, attributes, and XDD.
Finishing Report	This report identifies objects that need to be reviewed or completed before the object is considered completed. The report shows comments associated with a form, section, field, recipient, or trigger. It also tells you if the following items are undefined: <ul style="list-style-type: none"> • Triggers are not found in the trigger library • Fields are not found in the FDB • Graphics are not found in the library • Recipients are not defined in the global recipient list in the BDF file See <i>Printing a Finishing Report</i> on page 461 for more information.
Font Usage Report	This report provides information about the fonts used for the sections stored in the library. You can create the report for all sections or just for those you select.
Form Field Report	This report provides information on the fields contained in the forms you select. This includes the field name, associated rules and their parameters, and the format mask.

Report	Description
Form List Report	This report provides information for the form list files you select. This includes the form name, form description, DEC options, pre-selected options, file names, triggers, and regional date processing (RDP) rules.
Form Readability Statistics Report	This report shows readability statistics and Flesch score information for the sections that comprise the forms you select. This includes character, sentence, and syllable counts for the text areas contained in the sections defined in the form as well as averages for words per sentence and characters per word,
Form Report	This report provides information about the sections in the forms you selected. This includes information about the sections (name, options, size, orientation and number of pages) and the fields contained in the section (name, rule name, format mask, and rule parameters). The report also provides form section options, such as recipient names, copy counts, and section rule names and parameters.
Form Validation Report	This report provides validation information about the sections defined in the forms you selected. This report can tell you... <ul style="list-style-type: none"> • Whether sections referenced in the form (FAP files) exist in the library • If the sections defined in the form have assigned recipients • Any sections with assigned recipients and a copy count of zero • Any sections that contain text areas or multi-line text fields marked as Can Grow, but the Can Grow attribute is not specified at the form level
Impact Report	This report provides information based on the selected object type (form list, form, trigger, section, and so on) and which areas would be affected if that object were changed or deleted. For example, if you select a form list file, this report lists references to that form list file in the application file (BDF).
Paragraph Selection List Report	The report provides information about the Paragraph List files you select. It includes the Paragraph List file name and description and the names and descriptions of the paragraph contained in the paragraph list file.
Registry Report	This report provides information about Studio's registry settings. This report may be requested by Support.
Section Readability Statistics Report	This report shows readability statistics and Flesch score information for the sections you select. This includes character, sentence, and syllable counts for the text areas contained in each sections as well as averages for words per sentence and characters per word.
Section Report	This report provides information about the sections you select. This includes information about the margins (top, bottom, left, and right); paper (height, width, size, and orientation); section rules (rule name and parameters), fields (name, type, format, font ID, length, rules, and style), graphics (name, graphic, and style) and the fonts used (font ID and description).
Section Usage Report	This report lists all of the sections associated with the application definition (BDF) file you selected. The report lists all sections in use and sections which are not directly referenced on any form.
Template Report	This report provides information about the template including options (name, description, size, and orientation), the sections defined in the template (name, options, size, orientation, number of pages, and protect template item flags), and recipient and copy counts for the sections in the template.
Trigger Report	This report shows which DAL triggers are referenced and which are not referenced in the application definition (BDF) file you selected.
User Security Report	This report shows user security rights.

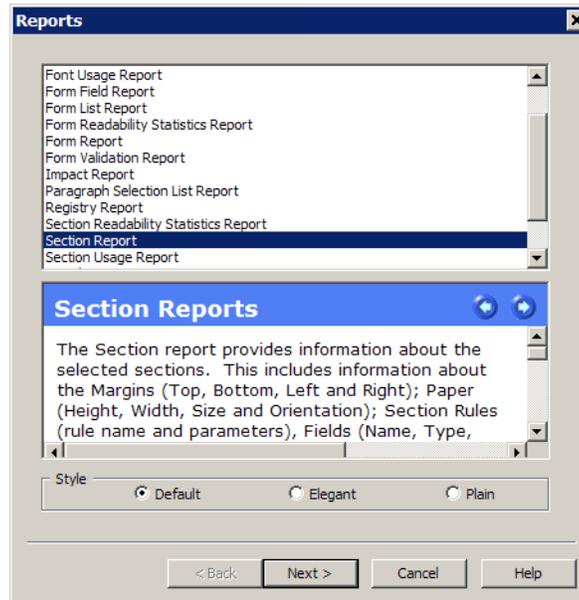
Report	Description
Documerge Field Report	Only available in Documerge Studio. This report shows you information about the Documerge resources you are maintaining. For more information, see <i>Printing the Documerge Field Report</i> on page 465.

Note Studio writes these reports using the HTML 4.01 standard so the report and all formatting can be in a single file. All reports are validated for HTML and CSS compliance and each style is written as CSS embedded in the header. If you are comfortable with HTML, you can edit the style in the report files you generate to suit your needs.

PRINTING REPORTS

The following steps take you through the steps for printing reports. Section reports are used as an example.

1. Highlight the type of report you want to print and choose the report style you prefer.



The differences in the styles are the fonts, borders, and shading used. Keep in mind styles can be interpreted differently by different browsers, so your results may differ. Here are some examples.

Elegant

Source: Q1ADDR v1.5 1/1/1980

Margins (FAP):

Top	Bo
0	0

Paper (FAP):

Height	Width
26400	20400

Rules:

Rule Name
SetOrigin

Default

Source: Q1ADDR v1.5 1/1/1980

Margins (FAP):

Top	B
0	0

Paper (FAP):

Height	Width
26400	20400

Rules:

Rule Name
SetOrigin

Plain

Source: Q1ADDR v1.5 1/1/1980

Margins (FAP):

Top	Bottom
0	0

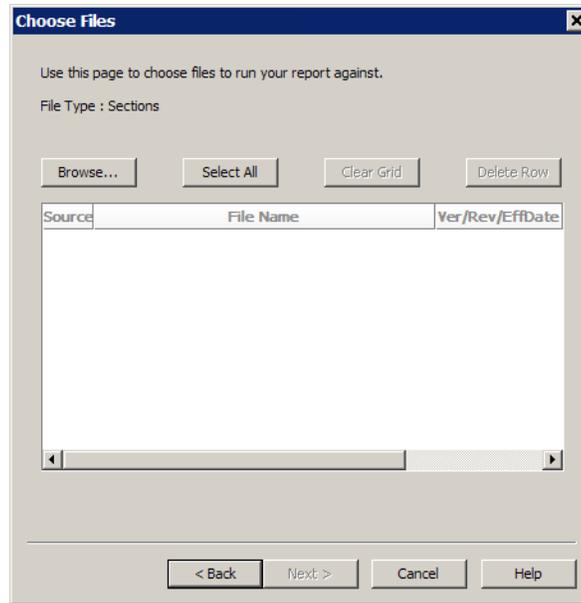
Paper (FAP):

Height	Width
26400	20400

Rules:

Rule Name
SetOrigin

2. Click Next. The Choose Files window appears.

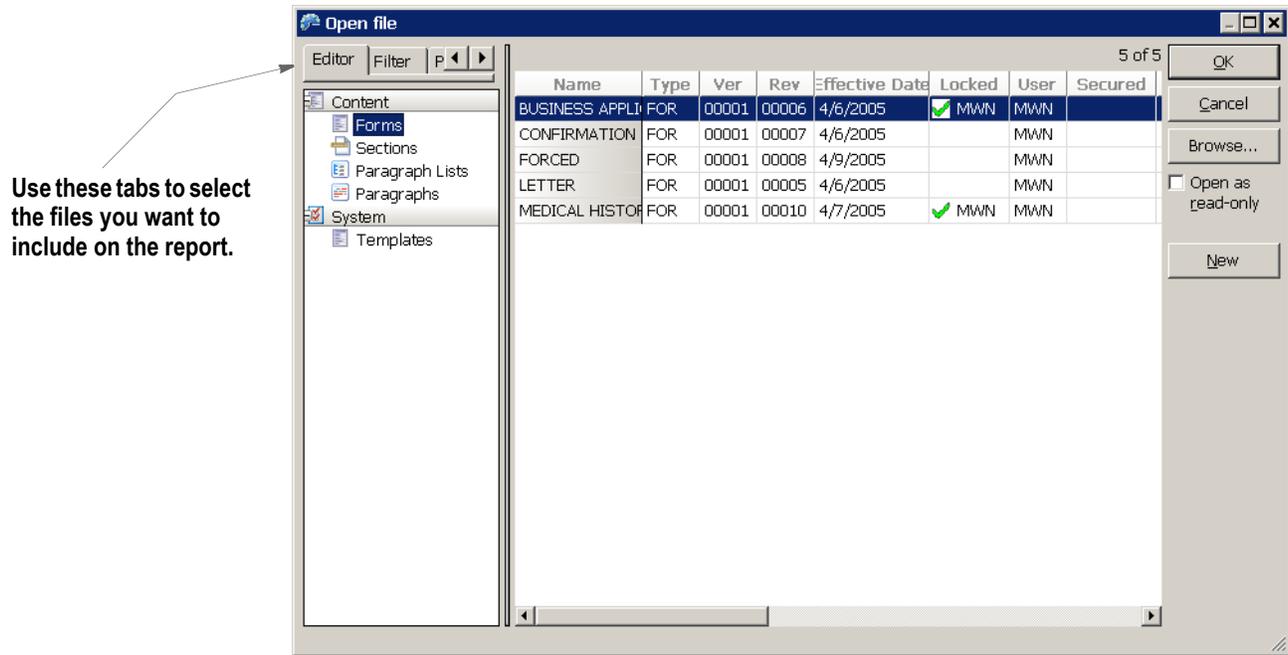


Use this window to choose the files you want to include on the report.

Click	To
Select All	Produce a report that contains an entry for each applicable item. When you click Select All, the Report Options window appears. Go to step 4 on page 459.
Browse	Select the specific files you want included on the report. When you click Browse, the Open Files window appears. Use the Ctrl key to select multiple files. See <i>Selecting Files</i> for more information.

Selecting Files

When you choose Browse, the Open File window appears:

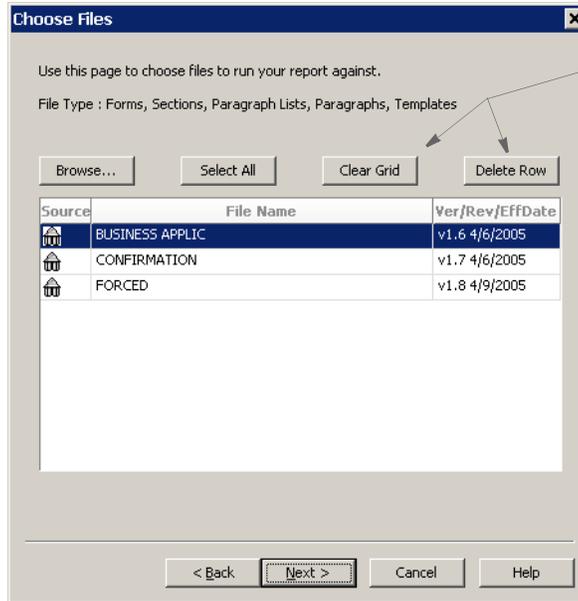


There are several ways to select the files to include on the report:

Click this tab To

Editor	List all of the file types you have to select from. To narrow the list to a specific file type, click that type in the list. For instance, in the example, Forms is Selected. If you wanted to choose from a list of sections, you would click Sections and select the sections you want to include on the report.
Filter	Use the filter fields to set criteria that narrows the list of items. Once you have the criteria set, click Search Now to process the search criteria.
Preview	See a preview of the highlighted item.

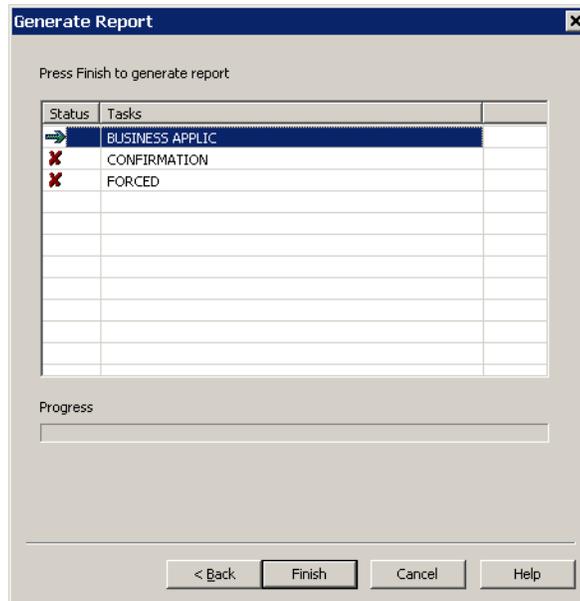
Whether you use the Editor or Filter tab, once you select your files, click Ok to return to the Choose Files window.



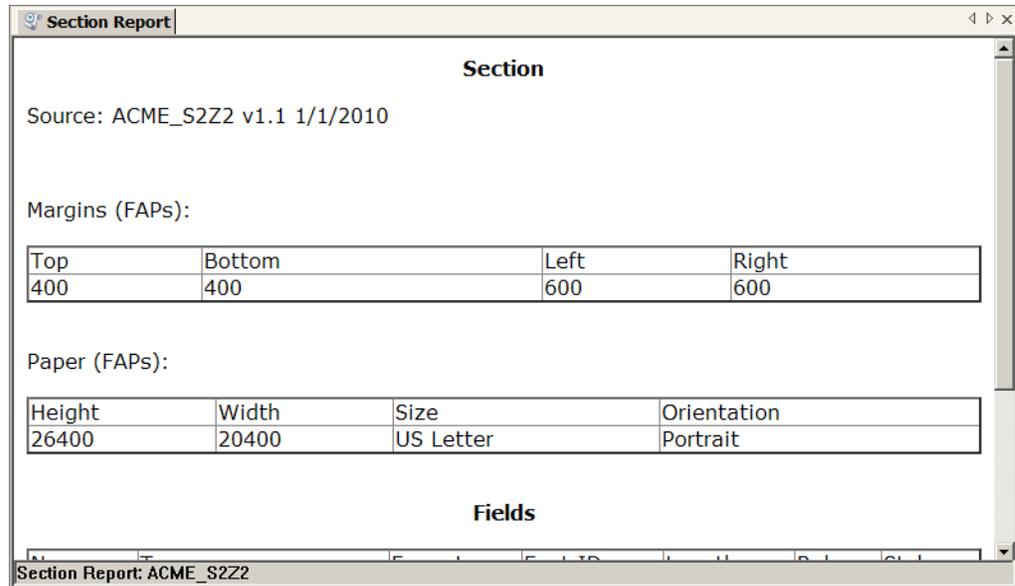
Use these buttons to modify your file selections.

Click Clear Grid or Delete Row to modify the file selection you made on the Open Files window. Click Next to proceed to the Report Options window.

- Once you have selected the files you want to include on the report, click Next. The Generate Report window appears.



- On the Report Options window, click Finish to generate the report. Studio displays the report on your screen. Here is an example of a Section Report:



You can right click within the report to choose from these options:

Option	Description
Clear Contents	Clears the contents of the report window.
Select All	Select all of the text within the report.
Copy	Copy all selected text within the report, so you can paste it into another application.
Save As	Save the report as an HTML file.
Send To	Send the report as an attachment in an email.
Print	Print the report.
Print Preview	Show the report in the Print Preview window.

PRINTING A FINISHING REPORT

Use the Finishing Report to see a list of the remaining tasks that apply to the forms, sections, paragraphs, and paragraph lists you chose to appear on the report. These tasks can include the following:

- Task comments
- Undefined fields
- Undefined graphics
- Undefined recipients
- Undefined triggers

Note These tasks can originate in Studio or the Documaker Add-In for Word.

Each task that appears on the report should be reviewed as part of your normal daily procedures. Follow these steps to print this report:

1. Choose Manage, Reports, and then choose the Finishing Report option.

Note You can also access the report from the Workspace toolbar by clicking Reports and choosing the Finishing Report option.

2. Choose from Default, Elegant, or Plain for the style of the report and then click Next. The Choose Files window appears.
3. Use this window to choose the files you want to include on the report.
 - Click Select All to produce a report that contains an entry for each section, form, paragraph, paragraph list, and template in the workspace’s library. When you click Select All, the Report Options window appears. Go to step 4 on page 459.
 - Click Browse to select the specific files you want included on the report.

When you choose Browse, the Open File window appears. On the Open File window, there are several ways to select the files to include on the Finishing report:

Click the To

Editor tab	Display all of the file types you have to select from. To narrow the list to a specific file type, click Forms, Sections, Paragraph Lists, Paragraphs, or Templates. The select the files you want to include on the report and click Ok.
------------	--

Click the To

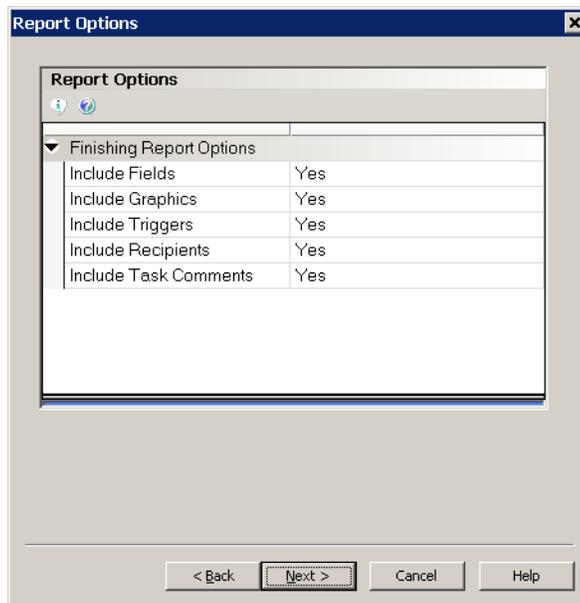
Filter tab Create a report that only shows files which have a remaining task activity.
 In the Type field, select Task. In the Task Type field, select the option for which you want to produce the report.

Search for references to ...	
Type	Task
Task Type	(any)
	(any)
	Task Comments
	Undefined Fields
	Undefined Graphics
	Undefined Recipients
	Undefined Triggers

Click Search Now to process the search criteria. Click Ok to accept the search results.

Whether you use the Editor or Filter tab, once you select your files, click Ok to return to the Choose Files window. Click Next to go to the Report Options window.

4. On the Report Options window, select the content you want to appear on the Finishing Report. Your choices include undefined fields, undefined graphics, undefined triggers, undefined recipients, and task comments.



5. Click Next after you have made your selections. Studio creates the Finishing Report.

Using the Finishing Report

Here is an example of a Finishing Report that includes one file:

Finishing Report

*FDB = C:\FAP\mstrres\amergen\DEFLIB\FDB.DBF.DBF

Source: COMDEC01.FAP v1.2 1/1/1980

COMDEC01

Section Name: COMDEC01
 Comment: This is a task comment on the COMDEC01 section.

Field Name: FORM LINE 1
 Task: Field reference not found in the Common Fields database.

Field Name: FORM LINE 2
 Task: Field reference not found in the Common Fields database.

Graphic Name: AMERGENSM
 Task: Graphic reference not found in library.

Trigger Name: AMERGEN_CLDEC
 Comment: This is a task comment on the AMERGEN_CLDEC trigger assigned to a text area on the COMDEC01 section.

Graphic Name: QJANED
 Comment: This is a task comment on the QJANED logo inserted into the section COMDEC01.

Graphic Name: AMERGENSM
 Comment: This is a task comment on the AMERGENSM logo inserted into the section COMDEC01.

Notice that the finishing activities on the example Finishing Report match those on the task list.

Comments for these resources appear here.

Here is a list of undefined fields included in this section.

Here is a list of the undefined graphics used in this section.

The report presents the information differently, but the information is the same. In the report, the task comments for a given resource appear first. Undefined item information for that resource appears next. Comments associated with objects that are inserted into the resource then appear.

Note See Creating an Implementation Finishing Task List for more information on an alternative way of displaying remaining finishing activities.

For task comments, you should open or check out the resource (section, form, paragraph, paragraph list, or template) review the comment and take the appropriate action.

This table shows you what to do with the various undefined items that can appear on the Finishing Report.

Item	Add undefined
Undefined recipients	Recipients to the Application Definition (BDF) file.
Undefined fields	Fields to the Common Fields dictionary.
Undefined graphics	Graphics to the library.
Undefined triggers	Triggers to the SETRCPTB.DAL file.

Note You can add the undefined item as stated or select a previously-defined element from the appropriate location.

PRINTING THE DOCUMERGE FIELD REPORT

Use the Documerge Field report while in Documerge Studio to see information about the Documerge resources you are maintaining. For each field, this report shows you the following information:

Column	Description
Field	The name of the field.
Type	The field type, either BPSD or VSD (Variable Space Definition, including multi-line fields).
Length	The length of the field.
Replacement	The replacement character used. This character fills space until replaced by variable data.
Multidata	Indicates the number of line feeds between the lines of variable data. An <i>N</i> indicates the system only chooses one data item. A number from zero (0) to nine indicates the number of lines the system inserts between data items.
Line End	The character that indicates a line end in a multi-line field. The default is [.
Delete	Tells you whether the system will delete the tag value after it is used. A <i>Y</i> indicates the data will remain available to Documerge for reuse in other occurrences of the field in the document package (mergeset) after being used once. An <i>N</i> indicates the data will be used once and then deleted. A one (1) indicates the system resets the value of the field, uses the data from the first occurrence, and does not delete the tag data. An <i>R</i> indicates the system resets the value of the field, uses the data from the first occurrence of the field, and then deletes the tag data.
Mandatory	Tells you whether the field is mandatory.
Generate	Tells you whether the default replacement characters are generated or whether the replacement characters are user-defined.

This report reads a specified Common Fields dictionary to display field (tag) information, therefore your workspace must utilize a Common Fields (FDB.DBF) file to run this report. To print this report, choose the Manage, Tools, Reports option, then select Documerge Fields.

Here is an example:

Documerge Field Report								
Fields								
Source: C:\FAP\mstres\normalize metacode\DEFLIB\FDB.DBF								
Name	Type	Length	Replacement	Multidata	Line End	Delete	Mandatory	Generate
ADDENDUM_NO	BPSD - Single line	10	@	N	[N	No	Yes
ADDR_CITYSTZIP	BPSD - Single line	40	@	N	[N	No	Yes
ADDR_LINE_1	BPSD - Single line	10	@	N	[N	No	Yes
ADDR_LINE_2	BPSD - Single line	40	@	N	[N	No	Yes

Chapter 14

Handling Fonts

A font is a collection of letters, symbols, and numbers which share a particular design. This chapter provides general information on font concepts and types.

A font is a collection of letters, symbols, and numbers which share a particular design. Studio provides a font manager which lets you organize sets of fonts for section creation and printing needs.

This chapter provides general information on font concepts and types and moves into the specifics of setting up fonts and using Font manager.

This chapter includes the following topics

Topics included are as follows:

- *General Font Concepts* on page 468
- *Using Code Pages* on page 474
- *Types of Fonts* on page 480
- *Using System Fonts* on page 483
- *Using Font Cross-Reference Files* on page 489
- *International Language Support* on page 493
- *Setting Up PostScript Fonts* on page 497
- *Font Naming Conventions* on page 502
- *Mapping Fonts for File Conversions* on page 503
- *Using Font Manager* on page 504

Note The Documaker system also includes several utilities you can use to work with fonts. These utilities are mentioned where appropriate throughout this chapter and are discussed in detail in the [Docutoolbox Reference](#).

GENERAL FONT CONCEPTS

FONT TERMINOLOGY

The following is a glossary of some common typographic terms you may encounter when working with fonts.

Typography is the art and technique of selecting and arranging type styles, point sizes, line lengths, line spacing, character spacing, and word spacing for typeset applications.

A *typeface* is a unique design of upper- and lower-case characters, numerals, and special symbols. Times-Roman, Arial-Italic, Courier-Bold are examples of typefaces.

A *font* is the implementation, for a specific device, of one typeface. A font contains a group of characters (letters, numbers, punctuation, and so on) which have a specific form and size. As you can see below, a Courier font is one which is designed to look like it was produced by a typewriter.

Courier fonts look like text produced by a typewriter.

A *font family* is family of related font typefaces. Times-Roman, Times-Bold, Times-Italic, and Times-BoldItalic are typefaces which belong to the Times font family.

Font size refers to the vertical point size of a font, where a point is about 1/72 of an inch.

There are several other terms used to describe the characteristics of a font, including:

- Ascender
- Baseline
- Descender

The *ascender* is the portion of a lowercase character that extends above its main body, as in the vertical stem of the character *b*.

The diagram shows the lowercase letters 'bcxy' in a serif font. A red arrow points from the text 'ascender' to the vertical stem of the letter 'b', which rises above the main body of the letters.

The *baseline* is an imaginary line upon which the characters in a line of type rest.

The diagram shows the lowercase letters 'bcxy' in a serif font. A red dashed line is drawn horizontally across the bottom of the letters, labeled 'baseline'.

The *descender* is the portion of a lowercase character that extends below the baseline, as in *y* or *g*.

The diagram shows the lowercase letters 'bcxy' in a serif font. A red arrow points from the text 'descender' to the tail of the letter 'y', which extends below the baseline.

Kerning is the process of decreasing space between two characters for improved readability, such as tucking a lowercase *o* under an uppercase *T*. A variation of kerning, called *tracking*, involves decreasing the amount of space between all characters by a specified percentage.

Leading is the amount of vertical space between lines of text. Leading (pronounced *led-ding*) is measured from baseline to baseline. On old hot-type printing presses, this was done by inserting strips of lead between the cast type.

Fonts are measured in *points*. A point is a typographical unit of measure which equals about 1/72 of an inch. For example, this is a **16 point font** while the rest of the line uses a 10 point font.

A *pica* is another typographical unit of measurement equal to 12 points. There are about 6 picas in one inch.

A *twip* is yet another typographical unit of measurement equal to 1/20th of a point. There are 1440 twips to one inch, 567 twips to one centimeter.

Pitch refers to the amount of horizontal space used for each character of fixed-width fonts. This is often specified in characters-per-inch (CPI). Typically, 10-pitch equals 12 point, 12-pitch equals 10 point, and 15-pitch equals 8 point type, but some fonts use other equivalencies.

Sans serif means without serifs and refers to a character (or typeface) that lacks serifs, such as Arial or Helvetica.

A *serif* is an ornamental aspect of a character. A serif typeface is one whose characters contain serifs (such as Times Roman or Courier).

Spacing can either be fixed or proportional. In a fixed font, such as Courier, every character occupies the same amount of space. In a proportional font, such as Arial or Times, characters have different widths.

Stroke weight refers to the heaviness of the stroke for a specific font. This is usually indicated in font names by including words such as Light, Regular, Book, Demi, Heavy, Black, and Extra Bold.

The *style* of a font is whether it is plain, bold, or italic.

National language terminology

Here are some additional terms you may encounter when working with fonts and supporting international languages.

National character handling is dependent on both the language used, and on the country. In many cases, the language is used only in one country (such as Japanese in Japan). In other cases, there is a national variant of the language (such as Canadian French).

A *code page* is a table which defines the mapping in a computer of each of these characters to a unique hexadecimal number, called a code point. There are three families of code pages: EBCDIC, ASCII, and ISO.

A *character set* defines which characters must be supported for a specific language.

Single byte character sets (SBCS) are character sets which can be defined using a single byte code point (code points range from 0 to 255). Most languages can be defined using an SBCS.

Double byte character sets (DBCS) are character sets which contain so many characters that they require two bytes to define the valid code point range. Languages which require a DBCS are Japanese (Kanji), Korean, and Chinese (both Traditional and Simplified). For example, the Kanji character set uses approximately 6,700 characters out of a total of 65,000 valid code points provided by a DBCS code page.

Multiple byte character sets (MBCS) use both single and double byte code points. This is also referred as a *combined code page*. For example, the combined Japanese code page 932 consists of a SBCS code page 897 and a DBCS code page 301. These code pages use the Shift JIS encoding defined by the Japanese Industry Standard Association, and contains Kanji, Hiragana, and Katakana characters.

Unicode is a character coding system designed to support the interchange, processing, and display of the written texts of the diverse languages of the modern world. In its current version (3.2), the Unicode standard contains over 95,000 distinct coded characters derived from dozens of supported scripts. These characters cover the principal written languages of the Americas, Europe, the Middle East, Africa, India, Asia, and Pacifica. Support for Unicode is growing among operating systems, such as Windows XP, and programming languages, such as Java.

Note Specific information on how to use Unicode is available in a separate document, entitled *Using Unicode*.

Bi-directional (BIDI) languages or *Extended SBCS* languages are languages which display text in a right-to-left manner and numbers in a left-to-right manner. Hebrew and Arabic are BIDI languages.

ANSI is an acronym for the American National Standards Institute. The Windows ANSI character set is based on code page ISO 8859-x plus additional characters based on an ANSI draft standard.

ASCII is an acronym for the American Standard Code for Information Interchange. ASCII is a 7-bit code that is a US national variant of ISO 646.

Program Integrated Information (PII) includes all text in messages, menus, and reports which is displayed to the user. To provide national language support, all PII text must be isolated for easy translation.

Enabled is a term used to indicate an application that has been altered to handle input, display, and editing of double byte languages (such as Japanese) and bi-directional languages (such as Arabic).

Translated is a term used to indicate an application which has been enabled and has had its Program Integrated Information translated into the national language. A translated application must also support various country settings, such as time, date, currency, and sorting.

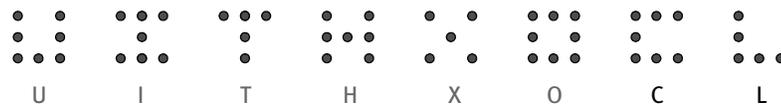
AFM is an extension used with Adobe® PostScript® font files. It stands for *Adobe Font Metrics*. AFM files are text files that describe a PostScript font.

HOW CHARACTERS ARE REPRESENTED

Fonts can use different methods of internally representing characters. Two categories of representing characters in fonts are known as bitmap fonts and scalable fonts.

Bitmap Fonts

Bitmap fonts describe each character as a pattern of black dots. Bitmap fonts were originally used for printer and screen devices because these devices were only capable of drawing dots. Below is crude representation of how the certain letters could be drawn as a series of dots in a 3x3 grid.



Essentially, this is what happens when a character is drawn to the screen or printed on paper. Fortunately, screen and printer fonts use a whole lot more dots per inch so that the distance between the dots becomes nearly invisible to the naked eye. By the way, this is also the reason why printed text looks better than text on the screen. Printed text often uses 300 or 600 dots per inch while your screen's resolution might be 96 dots per inch.

A different font file is required for each point size and different font files are required for different device resolutions (VGA vs. Super-VGA monitors, 300 dpi vs. 600 dpi printers).

Bitmap fonts are used primarily by printers. Bitmap fonts used by printers cannot be used for displaying text on screens because there are different internal formats and different resolutions. Printers which use bitmap fonts include HP® laser printers, IBM® AFP printers, and Xerox® Metacode printers.

Scalable Fonts

A scalable font can be scaled to any size needed. Characters of scalable fonts are internally represented as outlines (a series of straight lines and curves). These outlines can be scaled to allow characters to be rendered at different resolutions and point sizes. For example, the letter O may be represented as outer and inner circular lines whose interior is filled.

Outlines



Final Character



Two types of scalable fonts are TrueType and PostScript fonts.

TrueType

TrueType was designed and developed by Apple Computer and Microsoft for use on the Macintosh computer and PCs running Microsoft Windows. TrueType provides a number of advantages over bitmap fonts. TrueType is WYSIWYG (what you see is what you get). The same font can be used with printers and video displays. Typically, TrueType font files have a file extension of TTF.

PostScript

PostScript fonts were designed and developed by Adobe Systems Incorporated. PostScript fonts are a special implementation of a PostScript language program. PostScript fonts are scalable fonts. PostScript fonts describe each character as a series of straight-line and curved-line segments. These segments (also known as an outline) along with a flexible coordinate structure allow PostScript fonts to be scaled easily and used on different devices (video monitors and printers). PostScript printers support the PostScript language and fonts. There are several types of PostScript fonts:

- PostScript Type 1

When someone refers to a PostScript font, this is the type of font most often referred to. Typically, Type 1 font files have a file extension of PFB.

- PostScript Type 3

A Type 3 font is one whose behavior is determined entirely by the PostScript language procedures built into the font. These fonts are typically larger files than Type 1 fonts and do not take advantage of special algorithms built into the PostScript interpreter for rendering characters. This usually results in inferior output at small sizes and low resolution.

- PostScript Type 0

A Type 0 (zero) font is a composite font program that can contain several thousand characters, accessed by multi-byte codes. They can be used for non-Roman scripts, such as Japanese kanji.

- PostScript Multiple Master

Multiple master font programs are an extension of the Type 1 font format. Multiple master font programs contain a wide variety of typeface variations, such as multiple weights, character widths, and so on.

HOW COMPUTERS AND PRINTERS USE FONTS

What happens to make the letter *A* show up on the screen or print on a printer?

When you press the letter *A* on the keyboard, the keyboard sends a number to computer. On a PC, this number is usually 65 for the letter *A*. The computer uses this number to produce the letter *A*. For simplicity, let's assume you have a bitmap screen font.

As stated before, bitmap fonts describe each character as a pattern of black dots. Let's assume these patterns are stored in the font as a series of slots where slot 0 is followed by slot 1 which is followed by slot 2, and so on. For the number 65 (letter *A*), the computer simply draws the pattern of dots stored in slot 65. When the bitmap is drawn on the screen, we see what looks like the letter *A*.

If you print the letter *A* with a bitmap font, the concept is essentially the same. The printer receives the number 65 and prints the series of dots stored in slot 65 of the printer font.

The numbers which the computer uses to represent characters are called *code points*.

USING CODE PAGES

A code page is a table which defines the mapping in a computer of each of these characters to a unique hexadecimal number, called a code point. There are three families of code pages: EBCDIC, ASCII, and ISO.

A code page is a table that defines how the characters in a language or group of languages are encoded. A specific value is given to each character in the code page. For example, in code page 850 the letter *ñ* (lowercase) is encoded as hex A4 (decimal 164), and the letter *Ñ* (uppercase) is encoded as hex A5 (decimal 165). Of particular interest are these code pages:

- Code Page 850

Code page 850 is also called the Latin-1, multilingual code page. This code page supports the alphabetic characters of the Latin-1-based languages.

- Code Page 437

Code page 437 is the standard personal computer code page. The lower 128 characters are based on the 7-bit ASCII code. The upper 128 characters contain characters from several European languages (including part of the Greek alphabet) and various graphic characters. However, some of the accented characters, such as those used in the Nordic countries, are not represented. The missing characters are available in other code pages (code page 850 will usually contain the desired characters). It contains characters required by 13 languages used in approximately 40 countries.

- Code page 1004

Code page 1004 is the equivalent of the Windows ANSI code page. It contains more international characters than the multilingual code page 850. This character set contains all characters necessary to type all major (West) European languages. This encoding is also the preferred encoding on the Internet.

ISO 8859-x character sets use code points 128 through 255 to represent national characters, while the characters in the 32 to 127 range are those used in the US-ASCII (ISO 646) character set. Thus, ASCII text is a proper subset of all ISO 8859-X character sets.

The code points 128 through 159 are typically used as extended control characters, and are not used for encoding characters. These characters are not currently used to specify anything. This character set is also used by AmigaDOS, Windows, VMS (DEC MCS is practically equivalent to ISO 8859-1) and (practically all) UNIX implementations. MS-DOS normally uses a different character set and is not compatible with this character set.

ASCII Code Pages

ASCII is an acronym for the American Standard Code for Information Interchange. ASCII code pages are used on the PC platform. Code points below 32 for ASCII code pages are considered control characters for internal uses. These code points are usually not displayable characters. Code points from 32 to 127 are usually the same in ASCII code pages and are used for English letters, numbers, and punctuation.

Where ASCII code pages differ is in the characters assigned to code points 128-255. Code points 128-255 are used for international characters, math symbols, and so on. The characters for these code points vary in other code pages.

The characters used in code points below 128 use the English letters, numbers, and punctuation commonly found in ASCII code pages. The upper 128 code points are used for characters from several European languages (including part of the Greek alphabet) and various graphic characters. However, some of the accented characters, such as those used in the Nordic countries, are not represented.

Code page 437 is known as the standard personal computer code page. These characters were originally used in the original IBM PC. This code page is still used today in U.S. English versions of DOS and Windows. The primary code page used for these platforms is also known as the OEM code page.

Code page 850 is also called the multilingual code page. This code page supports many of the characters of the Latin-based alphabet.

The following table shows code page 850. To determine the code point associated with a character, use the numbers in the first row and column in the following table. For example, the letter *A* has a code point of 65 (64 + 1) and the space character has a code point of 32 (32 + 0).

Code Page 850

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0																
16																
32		!	"	#	\$	%	&	'	()	*	+	,	-	.	/
48	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
64	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
80	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
96	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
112	p	q	r	s	t	u	v	w	x	y	z	{		}	~	
128	Ç	ü	é	â	ä	à	á	ç	ê	ë	è	ï	î	ì	Ä	Å
144	É	æ	Æ	ò	ö	ò	ù	ù	ÿ	Ö	Ü	ø	£	Ø	×	f
160	á	í	ó	ú	ñ	Ñ	ª	º	¿	®	¬	½	¼	¡	«	»
176						Á	Â	À	©					¢	¥	
192							ã	Ã								¤
208	ø	Ð	Ê	Ë	È		Í	Î	Ï					Ì	ì	
224	Ó	ß	Ô	Ò	õ	Õ	µ	þ	Ɔ	Ú	Û	Ù	ý	Ý	–	·

240

-	±	¼	¶	§	÷	,	°	¨	•	¹	²	³			
---	---	---	---	---	---	---	---	---	---	---	---	---	--	--	--

There are many more ASCII code pages which are targeted for a specific country and or language. For example, code page 863 is used for Canadian French.

Code page 1004 is the IBM equivalent of the Windows ANSI code page. It contains more international characters than the multilingual code page 850. It contains characters required by 13 languages used in approximately 40 countries. Windows uses the ANSI code page to support most of the languages used in the Western Hemisphere and Western Europe. Keystrokes are translated by Windows from the primary (OEM) code page into the ANSI code page.

The following page shows the Windows ANSI code page. To determine the code point associated with a character, use the numbers in the first row and column in the following table. For example, the letter *A* has a code point of 65 (64 + 1) and the space character has a code point of 32 (32 + 0).

Code Page 1004 (ANSI Code Page)

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0																
16																
32		!	"	#	\$	%	&	'	()	*	+	,	-	.	/
48	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
64	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
80	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
96	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
112	p	q	r	s	t	u	v	w	x	y	z	{		}	~	
128	€		,	f	„	…	†	ç	^	‰	Š	‹	Œ		Ž	
144		‘	’	“	”	•	-	—	~	™	š	›	œ		ž	ÿ
160		ı	¢	£	¤	¥	¦	§	¨	©	ª	«	¬	-	®	¯
176	°	±	²	³	´	µ	¶	·	,	¹	º	»	¼	½	¾	¿
192	À	Á	Â	Ã	Ä	Å	Æ	Ç	È	É	Ê	Ë	Ì	Í	Î	Ï
208	Ð	Ñ	Ò	Ó	Ô	Õ	Ö	×	Ø	Ù	Ú	Û	Ü	Ý	Þ	ß
224	à	á	â	ã	ä	å	æ	ç	è	é	ê	ë	ì	í	î	ï
240	ð	ñ	ò	ó	ô	õ	ö	÷	ø	ù	ú	û	ü	ý	þ	ÿ

EBCDIC CODE PAGES

EBCDIC is an acronym for the Extended Binary Coded Decimal Interchange Code. EBCDIC code pages are used on mainframe (z/OS) and mini computers (AS400). There are many EBCDIC code pages. EBCDIC code pages usually share the same code points for English letters, numbers, and punctuation characters. However, EBCDIC code pages use different code points than ASCII code pages for the same English letters, numbers, and punctuation characters. Code points below 64 for EBCDIC code pages are considered control characters for internal uses. These code points are usually not displayable characters.

Code page 37 is an EBCDIC code page used on many z/OS and AS400 systems. Although the code points are completely different, code page 37 shares most of the same characters as code page 1004 (ANSI). The characters associated with code points 128-159 in the ANSI code page are not defined in code page 37.

Note The system uses some undefined code points (below 64) in code page 37 to try represent these characters. For maximum portability, avoid using code points 128-159 of the ANSI code page when composing forms.

The following page shows a table of code page 37. To determine the code point associated with a character, use the numbers in the first row and column in the following table. For example, the letter *A* has a code point of 193 (192 + 1) and the space character has a code point of 64 (64 + 0).

Code Page 37

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0																
16																
32																
48																
64			â	ä	à	á	ã	å	ç	ñ	ø	.	<	(+	
80	&	é	ê	ë	è	í	î	ï	ì	ß	!	\$	*)	;	¬
96	-	/	Â	Ã	À	Á	Ã	Ä	Ç	Ñ		,	%	_	>	?
112	ø	É	Ê	Ë	È	Í	Î	Ï	Ì	`	:	#	@	'	=	"
128	Ø	a	b	c	d	e	f	g	h	l	«	»	ð	ý	þ	±
144	°	j	k	l	m	n	o	p	q	r	ª	º	æ	,	Æ	¤
160	µ	~	s	t	u	v	w	x	y	z	ı	ı	Đ	Ý	Þ	®
176	^	£	¥	•	©	§	¶	¼	½	¾	[]	—	…	'	×
192	{	A	B	C	D	E	F	G	H	I	-	ô	ö	ò	ó	õ
208	}	J	K	L	M	N	O	P	Q	R	¹	ù	ü	ù	ú	ÿ
224	\	÷	S	T	U	V	W	X	Y	Z	²	Ô	Ö	Ò	Ó	Õ
240	0	1	2	3	4	5	6	7	8	9	³	Û	Ü	Ù	Ú	

CHARACTER SETS

You may have noticed that the largest code point shown in the earlier code page tables is 255 (240 + 15). The reason for this is that 255 is the largest value which can fit into a byte of memory. Code pages like this are said to have a single byte character set (SBCS). Some Asian languages, like Japanese and Chinese, contain so many characters that they must be represented by a double byte character set (DBCS) or a multiple byte character set (MBCS).

CODE PAGE NAMES

One confusing thing about code pages is that different organizations have different names for the same code pages. IBM, Microsoft, and the International Organization for Standardization (ISO) all use different names for essentially the same code page. You may hear a code page referred to by its IBM, Microsoft, or ISO name. For example, the ANSI code page is the same as IBM code page 1004, Microsoft code page 1252, and ISO code page 8859-1.

The following table shows a list of commonly used code pages. For more information, see these books:

- Developing International Software, Second Edition - Microsoft Press
- National Language Design Guide Volume 2 - IBM

Language	Country	Code Pages		
		Windows	OEM	z/OS
U.S. English	USA	1252 (ANSI)	437, 850	037
Canadian English	Canada	1252 (ANSI)	437, 850	037
Western Hemisphere and Western Europe SBCS Code Pages				
U.K. English	UK	1252 (ANSI)	850, 437	?
Brazilian Portuguese	Brazil	1252 (ANSI)	850, 437	?
Canadian French	Canada	1252 (ANSI)	850, 863	?
Danish	Denmark	1252 (ANSI)	850	?
Dutch	Netherlands	1252 (ANSI)	850, 437	?
Finnish	Finland	1252 (ANSI)	850, 437	?
French	France	1252 (ANSI)	850, 437	?
German	Germany	1252 (ANSI)	850, 437	?
Italian	Italy	1252 (ANSI)	850, 437	?
Norwegian	Norway	1252 (ANSI)	850	?
Portuguese	Portugal	1252 (ANSI)	850, 860	?
Spanish	Spain	1252 (ANSI)	850, 437	?
Swedish	Sweden	1252 (ANSI)	850, 437	?
Welsh	Wales	1252 (ANSI)	850	?
Eastern Europe SBCS Code Pages				

Language	Country	Windows	Code Pages	
			OEM	z/OS
Russian	Russia	1251 (Cyrillic)	866, 850	?
Bosnian	Bosnia	?	852, 850	?
Bulgarian	Bulgaria	1250 (Eastern Europe)	?	?
Croatian	Croatia	1250 (Eastern Europe)	852, 850	?
Czech	Czech	1250 (E.E.)	852, 850	?
Estonian	Estonia	1257 (Baltic)	?	?
Greek	Greece	1253 (Greek)	?	?
Hungarian	Hungary	1250 (E.E.)	852, 850	?
Latvian	Latvia	1257 (Baltic)	?	?
Lithuanian	Lithuania	1257 (Baltic)	?	?
Polish	Poland	1250 (E.E.)	852, 850	?
Romanian	Romania	1250 (E.E.)	852, 850	?
Serbian-Latin	Serbia	1250 (E.E.)	852, 850	?
Slovak	Slovak	1250 (E.E.)	852, 850	?
Slovenian	Slovenia	1250 (E.E.)	852, 850	?
Turkish	Turkey	1254 (Turkish)	857, 850	?
Extended SBCS Code Pages				
Arabic	Arabic speaking	1256 (Arabic)	864, 850, 437	?
Hebrew	Israel	1255 (Hebrew)	862, 850, 437	?
Thai	Thailand	874	874, 437	?
Asian DBCS Code Pages				
Japanese	Japan	932	932, 942, 437, 850	?
Korean	Korea	949	949, 850, 437	?
Simplified Chinese	PRC, Singapore	936	1381, 437, 850	?
Traditional Chinese	Taiwan, Hong Kong	950	938, 948, 437, 850 950, 437, 850	? ?

TYPES OF FONTS

Studio uses screen and printer fonts for displaying and printing text on forms. The Family field in the FXR contains the name of the screen font to use for displaying text under Windows.

The Font File fields in the FXR contain the names of the printer fonts to use when printing text. The FXR file provides attributes of the fonts and cross references the various font file names and parameters for different printers. The FXR does not contain any printer or screen fonts, only information about printer and screen fonts. FXR files are referred to in this section but are discussed in detail in the section, *Using Font Cross-Reference Files* on page 489.

USING SCREEN FONTS

Font Substitution in Windows

If the system cannot find a matching screen font using the information in the FXR, it will attempt to substitute a different Windows font. For Windows, the system will automatically try to substitute the following fonts for these missing fonts:

If this font is missing...	The system will substitute this font...
Courier	Courier New
Helv	Arial
Letter Gothic	Courier New
MICR	Courier New (fixed pitch) or Arial (proportional)
OCR A	Courier New (fixed pitch) or Arial (proportional)
OCR B	Courier New (fixed pitch) or Arial (proportional)
Times	Times New Roman
Times Roman	Times New Roman
Tms Rms	Times New Roman
Univers	Arial

Separate INI file control groups are used for Windows 3.1x (16-bit) and Windows 32-bit platforms for defining substitute font names. These control groups are named WINDOWSUBS and WINDOW32SUBS, respectively. Here is an example of the WINDOW32SUBS control group, which shows the defaults settings:

```
< Window32Subs >
  Univers           = Arial
  Helv              = Arial
  Letter Gothic    = Courier New
  Courier           = Courier New
  Tms Rms          = Times New Roman
  Times Roman      = Times New Roman
  Times            = Times New Roman
```

In this example, the system substitutes the native Windows 32-bit font, Times New Roman, if the Times family font is not found. Likewise, it substitutes Courier New for Letter Gothic and Arial for Univers. If you do not have a font installed which matches the original or substituted fonts, a default font will be used instead (usually Courier).

Installing Screen Fonts in Windows

To avoid these font substitutions, you can install fonts into Windows using the Fonts folder (usually located in the Control Panel). After opening the Fonts folder, select the File, Install New Font option. The Add Fonts window appears and asks for the drive and directory in which the new TrueType font files are located. When you finish selecting the fonts you want to install, click Ok to install them.

For the system to correctly match the fonts installed under Windows, the family and face name must be spelled exactly the same as they appear on the Names tab of the Properties window for the font. Use FXR settings for FAP height, FAP width, and so on, to customize the display of a font.

USING PRINTER FONTS

The system supports printer fonts for AFP, Xerox Metacode, PCL, and PostScript printers. Here is some background information you should know about each of these print platforms.

AFP

AFP fonts are designed solely for IBM's AFP printers. In AFP terminology, a font is described by these components:

Coded fonts

A coded font file contains references to specific character set and specific code page. Coded font files always begin with the letter *X*, such as *X0DATIN8*.

Code pages

In IBM AFP terminology, a code page file maps code points to an AFP character name in a character set file. Code page files always begin with the letter *T*, such as *TIDOC037*.

Character sets

A character set file contains the bitmap image of each character in the character set. Character set files always begin with the letter *C*, such as *C0FATIN8.240* or *C0FATIN8.300*. The character set file name extension (240 or 300) indicates whether the bitmap images are drawn at 240 or 300 dots per inch. Each character is given a eight letter AFP character name. For example, the letter *A* has an AFP character name of LA020000.

Metacode

Metacode fonts are designed solely for Xerox Metacode printers. Metacode fonts are bitmap fonts. Typically, Metacode font files have a file extension of FNT, such as *FXTIN8.FNT*. Characters are accessed by code points.

PCL

PCL is the Printer Control Language developed by Hewlett Packard for its LaserJet (and compatible) printers. PCL bitmap fonts are used by the system. PCL bitmap fonts can have any file name extension. The system provides PCL fonts with an extension of PCL, such as *FPTIN8.PCL*. Like Metacode fonts, PCL characters are accessed by code points.

PostScript Fonts

PostScript fonts were designed and developed by Adobe Systems Incorporated. PostScript fonts are actually a special implementation of a PostScript language program. PostScript fonts are scalable fonts and there are several types of PostScript fonts, PostScript Type 1 fonts are most common and are the only type supported by the system. Typically, Type 1 font files have a file extension of PFB, such as *COURIER.PFB*.

Each character in a PostScript font has a PostScript character name. When used as a screen font, the operating system associates code points in a code page with the appropriate PostScript character names.

Note The system uses the CODEPAGE.INI file to associate code points with the appropriate PostScript characters.

TrueType Fonts

TrueType is a scalable font designed and developed by Apple Computer and Microsoft for use on the Macintosh computer and on PCs running Microsoft Windows. The same font can be used with printers and video displays. Typically, TrueType font files have a file extension of TTF.

Adding Printer Fonts to a Font Cross-reference File

Fonts are added to an FXR file using Font Manager. You can insert TrueType, PCL, AFP, Xerox Metacode, certain FormMaker II files, and other FXR files into a font cross-reference file.

USING SYSTEM FONTS

Oracle Insurance has licensed for use and distribution with the system the following Postscript and TrueType fonts from Monotype Imaging, Inc. (formerly Agfa):

- Albany
- Arial Black
- Arial Narrow
- Courier
- Letter Gothic
- Times
- Univers
- Univers Condensed
- DocuDings
- MICR
- OCRA
- OCRB
- ZIPCODE

Albany (an Arial clone), Arial Narrow, Arial Black, and DocuDings (a Wingdings clone) are clones of commonly-used Windows fonts. The fonts are similar in appearance to the corresponding Windows fonts and have the same character width attributes. In addition, you can now use PCL, PostScript, AFP, and Metacode versions of these fonts for printing.

Note Although DocuDings is very similar to Wingdings, there are some differences. For instance, code point 255 in Wingdings is the flying Windows symbol (⚡). The DocuDings font displays a blank space for code point 255. The other code points (characters) are very similar in appearance but are not exact duplicates to the Wingdings font.

The Monotype font sets include the Euro character (€).

From these fonts, we have created fonts to use with AFP, PCL, and Xerox printers. These fonts let you print nearly identical forms on any supported printer. We use the following file naming convention for AFP, PCL, and Xerox printer fonts:

F T F1 F1 S P

For example, a 10 point bold Courier Xerox font would be named *FXCOB0.FNT*.

F	Standard Documaker system font
T	Printer type where A = AFP, P = PCL, X = Xerox 0 degree, 9 = Xerox 90 degree, 1 = Xerox 180 degree, 2 = Xerox 270 degree
F1	Two-character family name where AB = Albany, AL = Arial Black, AN = Arial Narrow, CO = Courier, HV = Helvetica, LG = Letter Gothic, TI = Times, UN = Univers, UC = Univers Condensed, DD = DocuDings, MI = MICR, OA = OCRA, OB = OCRB, ZP = ZIP code
S	Style where B = Bold, I = Italic, O = Bold Italic, N = Normal/Medium
P	Point size where 1 - 9 = point sizes 1-9 and 0 = point size 10 A - Z = point sizes 11-36

Font Cross-reference Files for Monotype Fonts

HPINTL.FXR, HPINTLSM.FXR

These FXR files provide support for Hewlett Packard (PCL) internal fonts using ANSI code page character sets instead using Monotype-based PCL downloadable fonts. The HPINTLSM.FXR file is a subset of the font information contained in the HPINTL.FXR file—*SM* indicates *small*.

REL103.FXR, REL103SM.FXR

These FXRs are similar to the REL102 FXRs but also include these fonts: Albany, Arial Black, Arial Narrow, and DocuDings. Be aware that the REL103SM.FXR file does not include DocuDings or all of the point sizes of the Albany group (including bold and italic), the Arial Narrow group (including bold and italic), and the Arial Black group (including italic).

You can identify these fonts via their names. For example *18010* indicates a 10-point Albany font. The initial *1* indicates Monotype, the *8* indicates Albany, the *0* indicates normal type, and *10* is the point size.

Arial Black fonts are indicated with a nine (9) and Arial Narrow fonts are indicated with a zero (0). DocuDings are indicated with a 34.

Below are the PostScript and TrueType fonts included in REL103SM.FXR:

PostScript Font	PostScript Font Name
ALBB____.PFB	Albany-Bold
ALBBI____.PFB	Albany-BoldItalic

PostScript Font	PostScript Font Name
ALBIT____.PFB	Albany-Italic
ALBR____.PFB	Albany-Regular
AN____.PFB	ArialNarrowMT
ANB____.PFB	ArialNarrowMT-Bold
ANBI____.PFB	ArialNarrowMT-BoldItalic
ANI____.PFB	ArialNarrowMT-Italic
ARBLI____.PFB	ArialMT-BlackItalic
ARIBL____.PFB	ArialMT-Black
DOCUD____.PFB	DocuDings

TrueType Font	TrueType Font Name
ALB.TTF	Albany AMT
ALBB.TTF	Albany AMT Bold
ALBBI.TTF	Albany AMT Bold Italic
ALBI.TTF	Albany AMT Italic
ARBL.TTF	Arial Black
ARBLIT.TTF	Arial Black Italic
ARIALN.TTF	Arial Narrow
ARIALNB.TTF	Arial Narrow Bold
ARIALNBI.TTF	Arial Narrow Bold Italic
ARIALNI.TTF	Arial Narrow Italic
DOCUDING.TTF	DocuDings

REL112.FXR
REL112SM.FXR

These files differ from the REL103.FXR and REL103SM.FXR files in that...

- The PDF417 fonts were added into the base FXR file.
- Character widths were corrected for font records previously created by importing TrueType fonts.
- Font heights were corrected for the Times fonts so Windows will select the correct screen font.

USING CUSTOM FONTS

To the system, custom fonts are simply fonts which are not based on the ANSI code page. This means that the font contains characters which have different code points or which do not exist in the ANSI code page. If you cannot use the system's Monotype fonts (or at least ANSI code page based fonts), you will need to consider these possible issues:

- Viewing Forms

Viewing forms may be the first problem since the characters in the original printer font do not match the characters used in displaying text on the screen. This problem will be seen during forms composition. This will also be a problem if the you have licensed the Entry or Archive Retrieval modules. Keyboard entry becomes a training issue as well. Under Windows, you must use 4-digit Alt key sequences to prevent code point translation.

If possible, you should convert any custom fonts to TrueType fonts for Windows and install the fonts into your operating system. If the font cross-reference file is properly modified to specify these screen fonts, the system will display your forms correctly. However, these characters may not display properly in Documaker Workstation.

- PDF Incompatibility

In addition to the Entry and Archive module problem, PDF or Acrobat files created for Internet archive retrieval use the ANSI code page for displaying forms. Therefore, archived forms based on custom fonts may not display correctly when retrieved through Docupresentment.

- Printing Forms

Another problem concerns using custom fonts on multiple (ASCII and EBCDIC) platforms. The system performs ASCII/EBCDIC translation based on the assumption that the ASCII code page is the ANSI code page and that the EBCDIC code page is code page 37. The system also assumes that PCL, PostScript, and Metacode printers use ASCII (hence ANSI) fonts. The system assumes AFP printers use EBCDIC fonts. The following table shows when the system will translate text (from FAP files) and variable data (from extract files) when printed under different platforms and printers.

Platform / Printer	ASCII (Windows 32-bit) ASCII FAP files and Extract data	EBCDIC (z/OS, AS400) EBCDIC FAP files and Extract data
AFP	ASCII to EBCDIC translation	No translation
PCL	No translation	EBCDIC to ASCII translation
PostScript	No translation	EBCDIC to ASCII translation
Xerox Metacode	No translation	EBCDIC to ASCII translation

On AFP printers

On a PC, text will be translated when printing to an AFP printer. Therefore, the code points used in text or variable data on forms are very important. After these code points are translated to the EBCDIC (code page 37), they must match the code points associated with the desired characters in the AFP code page which will be used.

On EBCDIC platforms, such as z/OS, AS400, text is assumed to be EBCDIC and will not be translated when you print to an AFP printer. The key to correct printing is to make sure the text (FAP files) and variable data (extract files) use the code points associated with the desired characters in the AFP code page you will use. Since FAP files are created as ASCII files on a PC, they will need to be transferred to the EBCDIC platform. Since you are using custom fonts, it is quite likely the file transfer software will not perform the proper code point translation. In this case, you may need to upload the files without translation and use the CPCNV utility to translate the files. This may require defining a special code page in the CODEPAGE.INI file for the CPCNV utility to use to do the proper translation.

On Xerox Metacode printers

On a PC, text (code points) will not be translated when printing to a Metacode printer.

On EBCDIC platforms (z/OS, AS400), text is assumed to be EBCDIC and will be translated to ASCII (ANSI code page) when printing to a Metacode printer. Therefore, the EBCDIC code points used in text or variable data on forms are very important. Since the FAP files are ASCII files created on a PC, they will need to be transferred to the EBCDIC platform. Since you are using custom fonts, it is quite likely that the file transfer software will not perform the proper code point translation. In this case, you may need to upload the files without translation and use the CPCNV utility to translate the files. This may require defining a special code page in the CODEPAGE.INI file for the CPCNV utility to use to do the proper translation.

On PCL printers

On a PC, text (code points) will not be translated when printing to a PCL printer. On EBCDIC platforms (z/OS, AS400), PCL print is not currently supported.

On PostScript printers

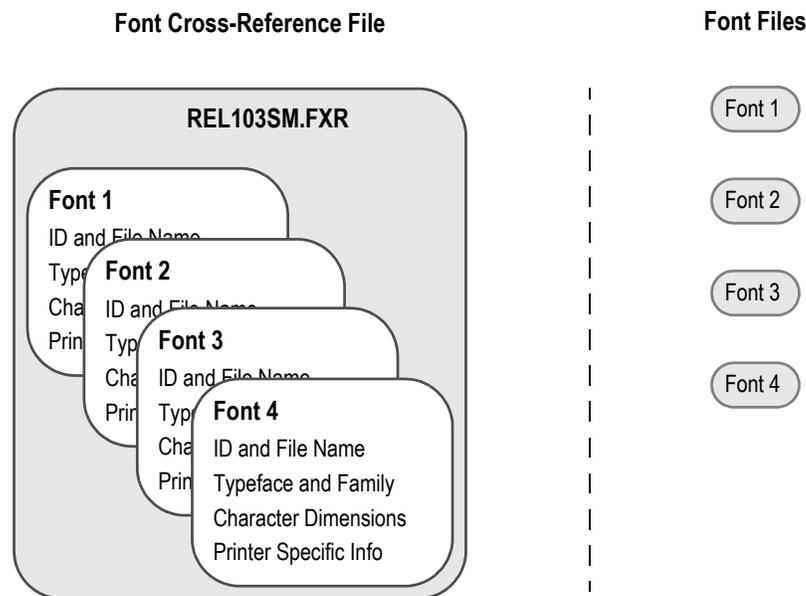
On a PC, text (code points) will not be translated when printing to a PostScript printer. On EBCDIC platforms (z/OS, AS400), PostScript print is not currently supported.

USING FONT CROSS-REFERENCE FILES

The font cross-reference file lets you organize the fonts you use for display and printing. The FXR provides the system with all the necessary font information. It does not contain the actual font files; rather, it contains information about the font attributes. Font attribute information includes formatting styles (bold, italic, and so on), point size (10 point, 14 point, and so on), and font stroke weight (heavy, light, and so on).

Note Storing the cross-reference information separately from the physical fonts affords greater flexibility in printer and font usage. You can convert virtually any font for your individual printer environment, provided you obtain appropriate license agreements for the fonts.

Let's examine the organization of the font cross-reference file and the font files. The illustration below depicts a font cross-reference file named REL103SM.FXR. This file contains a single font set. It includes all the crucial information for each font in the font set. The actual font files are physically separate from the font cross-reference file.



As shown above, the font files are distinct from the font cross-reference file. When you work with the font cross-reference file you affect the stored font information. You do not affect the separate and independent font files. The number of available fonts is limited only by your needs and storage space. If you keep this organizational structure in mind you can easily work with the font cross-reference file.

The font cross-reference file provides the names of your independent font files, but it is more than a simple listing of fonts. The file contains crucial font attribute information along with information specific to your printer types. The printer information is crucial because sections are compiled based on your printer environment.

The font cross-reference file ends in the extension *FXR* (for font cross-reference). The system includes these font cross-reference files:

- FAP\MSTRRES\FMRES\DEFLIB\HPINTL.FXR
- FAP\MSTRRES\FMRES\DEFLIB\HPINTLSM.FXR
- FAP\MSTRRES\FMRES\DEFLIB\REL103.FXR
- FAP\MSTRRES\FMRES\DEFLIB\REL103SM.FXR
- FAP\MSTRRES\FMRES\DEFLIB\REL112.FXR
- FAP\MSTRRES\FMRES\DEFLIB\REL112SM.FXR
- FAP\MSTRRES\FMRES\DEFLIB\REL113.FXR
- FAP\MSTRRES\FMRES\DEFLIB\REL113SM.FXR

Keep in mind these points concerning the FXR file:

- Contains one font set

The font set is the specific group of fonts you choose to include in your font cross-reference file. Each font cross-reference file contains a single font set. You assign each font cross-reference file and font set a unique name. For example, you might organize a font set for creating and printing accounting forms in a font cross-reference file called ACCOUNT.FXR.

- Contains information on multiple fonts

A font set contains numerous fonts. For example, a font set might contain Times New Roman fonts and Gothic fonts of multiple point sizes with bold and italic attributes. A second font set might contain Courier fonts and Helvetica fonts, also of multiple point sizes with bold, italic and regular attributes.

- Independent of your font files

The font cross-reference file works with the printer and window font files. Remember that the font files are separate files from the font cross-reference file.

HOW FXR SETTINGS AFFECT DISPLAY AND PRINT QUALITY

Certain attributes in the FXR file affect how the system displays text. For example, when the system displays text, it uses scalable font technology which exists in Adobe Type 1 Postscript fonts and TrueType fonts. All versions of Windows support TrueType fonts. Windows 2000 also supports PostScript fonts.

These fonts are selected via the family name specified in our FXR, and scaled according to point size, height and width parameters in the FXR. The fonts are spaced according to the character widths specified in the FXR.

Once the font is selected, then it can be zoomed in and out, or additionally scaled as required. Bitmap fonts do not have this scaling ability, which is why scalable fonts are used for display purposes, rather than bitmap fonts.

This means that when the system displays text on the screen, it attempts to mirror how it will look on paper. To achieve the same look on the screen as on paper, the parameters in the FXR are critical. The more accurate the FXR is, the more likely the display will mirror the printed document. The printed document is the standard for the screen display.

Since the system includes Monotype TrueType and PostScript fonts which match its printer fonts, if you install these Monotype fonts on a Windows system, what you see on your screen will more closely match what you print out. The keys are to closely match the printer's fonts and to have the best possible information in the FXR file.

Creating a font cross-reference file is usually done by importing a printer font file using Font manager. Since the font cross-reference file is a representation of information contained in the printer font file, modification of its fields usually does not affect the printed output. However, modifying these FXR fields can improve the system's ability to display forms.

MAINTAINING FXR FILES

Use the Font manager to maintain FXR files. You can start this tool in Studio using the Manage, System, Fonts option. Font manager makes it easy to insert, edit, copy or delete font information in the FXR file.

Choosing a Font Cross-reference File

During library setup, you must choose either REL103.FXR or REL103SM.FXR as the font cross-reference file for an AFP printer. You should also specify the PCL download font file named REL103SM.FNT in the FntFile option of the Resource Library window.

If you have older versions of the AFP coded font and code page files installed in PSF or PSF/2, you can use these versions to print to the same AFP printer. If you do not keep the older AFP coded font and code page files installed, you must recompile AFP page overlays for the current version using REL103.FXR or REL103SM.FXR.

This example shows you how the HPINTL.FXR and HPINTLSM.FXR files use PCL escape sequences in the Setup Data field (on the Font Properties window) to use internal fonts on a PCL printer. If you use Font manager to edit a font in the HPINTL(sm).FXR file, you will see the PCL escape sequence in this field.

For example, if you look at the Setup Data field for font ID 11036 (Times Roman Normal 36 point), you will see:

```
~ (19U~ (s1p36v0s0b5T
```

Where Represents

~	an escape character which must always start a PCL escape sequence
(19U	the primary symbol set or code page (Windows 3.1 Latin 1 in this case)
~	the start of a second PCL escape sequence
(s1p	the spacing of the font (proportional in this case)

Where Represents

36v	the height of the font in point size (36 point in this case)
0s	the style of the font (normal in this case)
0b	the stroke weight of the font (medium in this case)
5T	the typeface family of the font (Times Roman in this case)

There are other values you can use for each of these sequences. For example, the character or symbol set values used in HPINTL.FXR are:

19U for Windows 3.1 Latin 1

This symbol set matches the Windows ANSI code page and IBM code page 1004. You can find a list of character set values in the HP manual entitled, *PCL 5 Comparison Guide*.

Spacing values are *(slp* for proportional fonts and *(s0p* for fixed pitch fonts.

- Point size values are placed before the *v*
- Font styles are *0s* for normal, *1s* for italic
- Font stroke weights are *0b* for medium, *3b* for bold

The typeface family values used in HPINTL.FXR are:

- *5T* for Times Roman
- *3T* for Courier
- *6T* for Letter Gothic
- *52T* for Univers

INTERNATIONAL LANGUAGE SUPPORT

Our goal for international language support is to support the languages you are most likely to need. At the present, we consider these languages to be those used in the Western Hemisphere and Western Europe.

If you need support for Far Eastern languages like Chinese, Japanese, or Korean or if you need support for Eastern European languages, you must use version 10.2 or higher. See *Using Unicode* for more information.

USING THE ANSI CODE PAGE FOR PC PLATFORMS

The Windows operating environment supports languages in these countries via a code page known as the *ANSI code page*. Windows supports different keyboard mappings for these countries by translating the key codes into ANSI code points. Therefore, even though a keystroke for an international character generates different *key codes* on English, Spanish, and French keyboards, a Windows application receives the same *ANSI code point*.

We adopted these standards:

- The ANSI code page is the standard code page for all data files. The text contained in FAP files is stored using the ANSI code page.
- The ANSI code page is the standard for the Monotype fonts included with the system.

Note See *Using International Characters on page 495* for more information.

By adopting these standards, you receive these benefits:

- Support for 13 languages used in approximately 40 countries
- Improved platform resource compatibility (Windows, UNIX, and z/OS).
- You only need one set of Monotype fonts—no need to create separate fonts for each language
- Improved support of other Windows products, such as dictionaries, databases, and so on.

The ANSI code page is used by the World Wide Web and UNIX computers, as well as the Windows operating environment.

There are a few drawbacks to this approach. For instance, although all international alphabetic characters in the Latin character set are supported in the ANSI code page, certain symbols available in other code pages are not supported. These symbols include mathematical, scientific, and line drawing symbols. Greek, Cyrillic, and Asian characters are not supported either. And, in some cases, data files may have to be converted to ANSI code page characters.

USING CODE PAGE 37 FOR EBCDIC PLATFORMS

To support international languages on EBCDIC platforms, such as z/OS and AS400, we use EBCDIC code page 37 as the standard EBCDIC code page. Code page 37 is the native code page for many z/OS systems. By using code page 37, you receive these benefits:

- Code page 37 supports languages used in Europe and North and South America, such as French, Spanish, Italian, German, Portuguese, and Danish.
- This reduces or eliminates the need to convert extract files containing international characters on z/OS.
- This helps reduce or eliminate the need to convert resources before uploading to EBCDIC platforms from Windows.
- Using code page 37 for EBCDIC platforms creates compatibility problems with resources created in earlier versions. This only affects resources created in an earlier version which contain international or desktop publishing characters.
- All characters defined in code page 37 are also contained in code page 1004, the standard ASCII code page. There are, however, characters in code page 1004 which are not in code page 37—mainly desktop publishing characters from code point 128 to 159. To support these characters, we use undefined code points in code page 37 (code points below 64). For maximum portability, *avoid* using characters not defined in code page 37.

AFP print output and resource files normally use EBCDIC characters. The other supported printers, such as Metacode, PCL, and PostScript, normally use ASCII characters.

Note The current AFP code page file is called T1DOC037, the AFP code page for prior versions was called T100ASC4. The current AFP coded font files are called X0DA????.FNT, the AFP coded font files for prior releases were called X0FA????.FNT. The AFP character set files are unchanged and can be used by all versions.

USING INTERNATIONAL CHARACTERS

One method for entering international characters is to install a country/language specific version of Windows. These language-specific versions of Windows map characters from the keyboard differently so that it is easier to enter characters common to that language. In the simplest case, a single keystroke will generate an international character.

For example, if you have a Canadian French version of Windows, pressing the slash character (/) on a U.S. keyboard produces an e-acute letter (é). Many international characters require a two-character keystroke combination. Again using the Canadian French keyboard setup, you must press the left square bracket ([) followed by the letter *e* to generate an e-circumflex letter (ê).

Having to install a special version of Windows would be difficult for those in the U.S. who are trying to compose forms with French characters. Fortunately, there is a simpler solution.

Using the numeric keypad on the right side of your keyboard, you can hold down the Alt key and enter a three-digit number to enter an international character. For example, if your primary (OEM) code page is 437 or 850, you can enter the letter *ñ* (lowercase) by pressing the Alt key while you type 164 on the numeric keypad. When you release the Alt key, the code point 164 will be generated by the keyboard, which Windows will display as the letter *ñ*.

Note If you look at the code page 1004 table you will see that on the ANSI code page code point 164 is not the letter *ñ*. So why is the letter *ñ* being displayed? Windows recognizes that a code point of 164 has been generated by the keyboard and it is associated with the OEM code page (437 or 850). For this code page, code point 164 maps to the letter *ñ*. In Windows, the code point from the keyboard is translated from 164 to 241. A Windows program will actually receive a keystroke code point of 241 instead and that code point will display as the desired letter *ñ*.

You can also use the numeric keypad to enter ANSI code points directly. Using the numeric keypad on the right side of your keyboard, you can press the ALT key and type a four-digit number to key in an international character. For example, you can enter in the letter *ñ* by pressing the Alt key and typing 0241 on the numeric keypad. Entering a four-digit number beginning with a zero tells Windows you are entering a code point for the ANSI code page. Therefore, Windows does not need to translate the code point and passes the keystroke code point directly to the Windows application.

By standardizing on the ANSI code page, a document containing several languages can be read and written by a number of people from different countries. The keystroke code point translation lets Windows support many OEM code pages and keyboard settings.

Note You can use any Windows text editor, such as Notepad, to edit resource files since Windows also uses the ANSI code page.

CONVERTING TEXT FILES FROM ONE CODE PAGE TO ANOTHER

There are two situations where you may need to convert text files from one code page to another.

- If the customer's data (extract) file is not in the ANSI code page and the file contains international characters, you will need to convert the customer data file to use the ANSI code page.
- If you need to upload system resource files, such as FAP, INI, and menu resource (MEN.RES) files, which contain international characters to an EBCDIC platform, such as z/OS, and the file transfer software cannot convert ANSI code page file to EBCDIC code page 37.

To convert a file from one code page to another, you can use the CPCNV code page conversion utility. For more information, see the [Docutoolbox Reference](#).

SETTING UP POSTSCRIPT FONTS

The system includes a standard font set with PostScript fonts. These fonts reside in the FAP\MSTRES\FMRES\DEFLIB\ directory with the sample forms included with Documaker Studio. We devised naming conventions for the bitmap printer fonts that are created from the PostScript fonts supported by the system. PostScript fonts are easily converted to Xerox, AFP, and PCL formats.

Note When you create bitmap printer fonts from PostScript fonts, follow the naming convention outlined in the table below. This will make it easier to track and identify those fonts.

A standard font has a six-character name. Each character indicates a specific piece of data that describes the font. For example, you may take a PostScript font such as Times (Roman), Bold (TIB____.PFB), convert the font to Metacode format, and change the name to the standard FSI bitmap font name (FXTIOM). The font name characters designate the following:

Character	Definition
1	Converted PostScript fonts always begin with the letter F, indicating a system supported font.
2	Indicates the printer platform associated with the converted font: X = Xerox, A = AFP, P = PCL
3 and 4	Indicate the font family, such as Times Roman, Courier, and so on. AB = Albany AL = Arial Black AN = Arial Narrow CO = Courier DD = DocuDings UC = Univers Condensed LG = Letter Gothic MI = MICR TI = Times (Roman) OA = OCRA UN = Univers(al) OB = OCRB ZP = ZIP code
5	Indicates the style of the font: N = Normal (no attributes), B = Bold, I = Italic, O = Bold, Italic
6	Indicates the point size of the font. Use numbers 1 through 9 for point sizes 1 through 9. 0 (zero) = 10 point A = 11 point B = 12 point C = 13 point--through--Z = 36 point

This table lists PostScript fonts and their file names. The list shows the font names before you create and name the fonts using the conventions in the previous table. Point sizes are omitted in the names below. Use the table on the previous page to determine the remaining font file name value for each corresponding font size.

Font	File Name
Albany	ALBR____.PFB
Arial Black	ARIBL____.PFB
Arial Narrow	AN____.PFB
Courier	CO____.PFB
Courier Bold	COB____.PFB
Courier Bold Italic	COBI____.PFB
Courier Italic	COI____.PFB
DocuDings	DOCUD____.PFB
Letter Gothic	LG____.PFB
Letter Gothic Bold	LGB____.PFB
Letter Gothic Bold Italic	LGBSL____.PFB
Letter Gothic Italic	LGSL____.PFB
MICR MT	MICR____.PFB
OCRA MT	OCRA____.PFB
OCRB MT	OCRBMT____.PFB
Times Roman	TIR____.PFB
Times Roman Bold	TIB____.PFB
Times Roman Bold Italic	TIBI____.PFB
Times Roman Italic	TII____.PFB
Univers	UNM____.PFB
Univers Bold	UNB____.PFB
Univers Bold Italic	UNBI____.PFB
Univers Italic	UNMI____.PFB
Univers-Condensed Bold	UNCB____.PFB
Univers-Condensed Medium	UNCM____.PFB
Univers-Condensed Medium Italic	UNCMI____.PFB
ZIPcode Barcode-Regular	ZIPCODE_.PFB

Remember that PostScript fonts are scaleable. You complete font file name by adding the point size values when you convert the font. Here is an example:

CSBD__.PFB = CS Bookman Bold (any point size)

Note AFM files are Adobe Font Metrics files which describe a PostScript font. These files are used when you install PostScript fonts using Adobe Type Manager.

PostScript fonts reference code pages to define window and print characters. In turn, the code page maps to specific characters in the character set. The PostScript fonts included with Documaker Studio reference code page 1004, W1 and are shown here:

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
0																	
16																	
32		!	"	#	\$	%	&	¢	()	*	+	,	-	.	/	
48		0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
64		@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
80		P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
96		`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
112		p	q	r	s	t	u	v	w	x	y	z	{		}	~	
128		€		,		„	°	†	‡	^	%	Š	‹	Œ		Ž	
144			‘	’	“	”	•	–	—	~	™	š	›	œ		ž	ÿ
160			ı	¢	£	¤	¥	¦	§	¨	©	ª	«	¬	-	®	¯
176		°	±	²	³	´	µ	¶	·	¸	¹	º	»	¼	½	¾	¿
192		À	Á	Â	Ã	Ä	Å	Æ	Ç	È	É	Ê	Ë	Ì	Í	Î	Ï
208		Ð	Ñ	Ò	Ó	Ô	Õ	Ö	×	Ø	Ù	Ú	Û	Ü	Ý	Þ	ß
224		à	á	â	ã	ä	å	æ	ç	è	é	ê	ë	ì	í	î	ï
240		ð	ñ	ò	ó	ô	õ	ö	÷	ø	ù	ú	û	ü	ý	þ	ÿ

Bitmap fonts are a specific set of symbols or characters. The maximum number of characters a set of bitmap fonts can reference is 256. Scaleable fonts, such as PostScript fonts, may have more than 256 characters, but only 256 can be used at one time. The system’s font structure is designed to use the standard code page 1004, W1. Code pages are predefined in your system, and reside in the CODEPAGE.INI file in your DEFLIB directory. The path is FAP\MSTRRES\FMRES\DEFLIB.

The characters in the code page include foreign language characters and mathematical function characters. When you convert PostScript fonts using Font manager, you always select this code page (1004). You may, however, notice that the PostScript fonts themselves support multiple code pages.

Note If you want to use the internal printer fonts and you will print international characters, your printer must have the character or symbol set named Windows Latin 1 (also known as ANSI code page) on your printer. Be aware that not all PCL printers support this character set.

FONTS FOR PDF FILES

When you are creating PDF files, keep in mind that the following PostScript fonts are included with Adobe Acrobat Reader and do not have to be embedded.

Fixed Pitch Fonts	Proportional Fonts
Courier	Helvetica
Courier-Bold	Helvetica-Bold
Courier-Oblique	Helvetica-Oblique
Courier-BoldOblique	Helvetica-BoldOblique
	Times-Roman
	Times-Bold
	Times-Italic
	Times-BoldItalic
	Symbol
	ZapfDingbats

Importing PostScript Symbol Fonts

You can select a code page when importing PostScript symbol fonts, such as Euro Sans and ITD Zapf Dingbats, which contain characters that do not adhere to a standard Windows code page.

In Font manager, select *9999,WD* as the code page when importing these types of PostScript fonts.

Note For normal fonts, you should continue to select *1004,W1* as the code page.

If you import a PostScript font using code page 1004,W1 and the system produces a font record with only a few non-zero character widths or produces an internal error, try using code page 9999,WD to import the font.

For instance, importing Euro Sans and ITC Zapf Dingbats using code page 1004,W1 produces a font record where only the space and hard space characters (code points 32 and 160) contain non-zero character widths. Importing the same fonts using code page 9999,WD produces a font record with non-zero character widths for virtually every code point from 32 to 255.

When you use the PS2PCL utility to convert PostScript symbol fonts to PCL, specify the symbol set by setting the /S parameter to *WD*. This tells the utility that these PostScript fonts that contain characters that do not adhere to a standard Windows code page.

Note When converting normal text fonts with the PS2PCL utility, continue to set the /S parameter to *W1*.

FONT NAMING CONVENTIONS

When adding fonts to a font set, or when installing new fonts, you must give each font a unique ID. Use this 5-digit naming convention:

The...	Indicates...
First digit	the font provider: 1= Monotype 2= Adobe
Second digit The standard FXR file (REL103SM) defines only Times (Roman), Courier, and Univers. If you add other fonts to your FXR, use these font code naming conventions. (DocuDings is included in 3)	the font type or font family: 1 = Times (Roman) 2 = Courier 3 = OCRA, OCRB, MICR, and ZIPcode* 5 = Letter Gothic 6 = Univers 7 = Univers Condensed 8 = Albany 9 = Arial Black 0 = Arial Narrow
Third digit	the font attributes 0= normal 1= bold 2= <i>italic</i> 3= <i>bold, italic</i>
Fourth and fifth digits	the point size of the font, such as 09 point, 12 point, and so on.

* The OCRA, OCRB, MICR, and ZIPcode fonts do not have bold or italic styles so the second and third digits identify these fonts: 30 = OCRA, 31 = OCRB, 32 = MICR, 33 = ZIPcode, and 34 = DocuDings.

For example, 11010 indicates Times (Roman) Regular 10 point, 11214 indicates Times (Roman) Italic 14 point, and 16110 indicates Universal Bold 10 point.

Note You can only use a font ID from 00001 to 32767 and the font ID must be numeric not alphanumeric.

MAPPING FONTS FOR FILE CONVERSIONS

When converting a file from one format to another, you may need to convert the fonts used in the document. You can use INI control groups and options to map fonts in a source document to the fonts you want to use in the destination document. For instance, if you are converting an RTF file into a FAP file, you can use the following control group:

```
< RTFFontMAP >
  Arial = Swiss
```

This tells the system to convert all Arial fonts into Swiss fonts. Use this control group when converting DCD files into FAP files:

```
< FontFamilyMatching >
  Arial = Swiss
```

Place these control groups and options in the FAPCOMP.INI file.

RTF and DCD files contain font information about the generic font families used. For example, Arial and Univers, both sans serif proportional fonts, belong to a generic font family called *Swiss*.

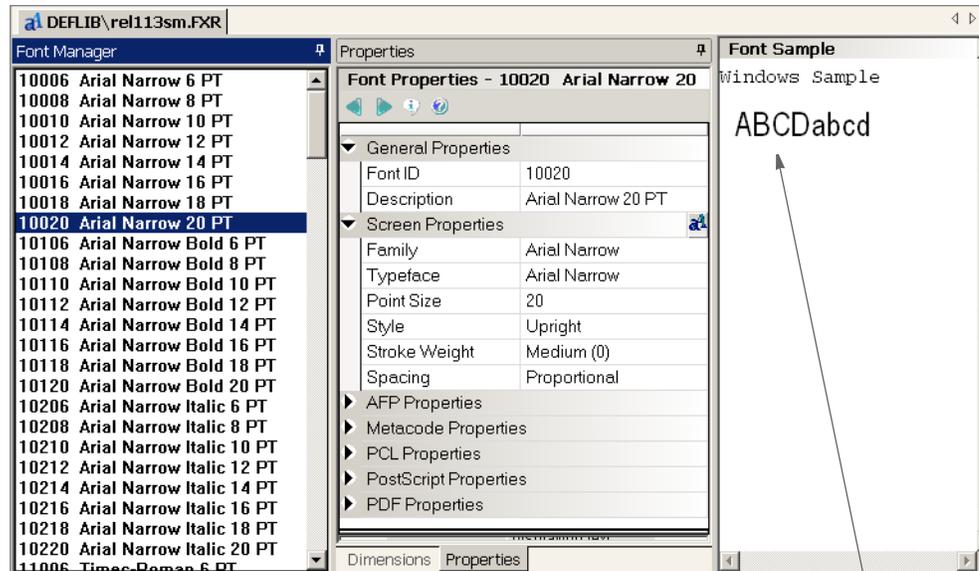
The RTF and DCD converters in the system use the RTFFontMap and FontFamilyMatching control groups to assign a font when other means of mapping fonts from the RTF or DCD file fails.

In Windows environments, there are several generic font families, as shown in this table:

Family	Description
Decorative	Specifies a novelty font, such as Old English.
Dontcare	Specifies a generic family name. This name is used when information about a font does not exist or does not matter. The default font is used.
Modern	Specifies a monospace font with or without serifs. Monospace fonts are usually modern fonts, such as Pica, Elite, and Courier New.
Roman	Specifies a proportional font with serifs, such as Times New Roman.
Script	Specifies a font that is designed to look like handwriting, such as Script and Cursive.
Swiss	Specifies a proportional font without serifs, such as Arial.

USING FONT MANAGER

When you choose the Manage, System, Fonts, option, here is an example of the Font manager window that appears:



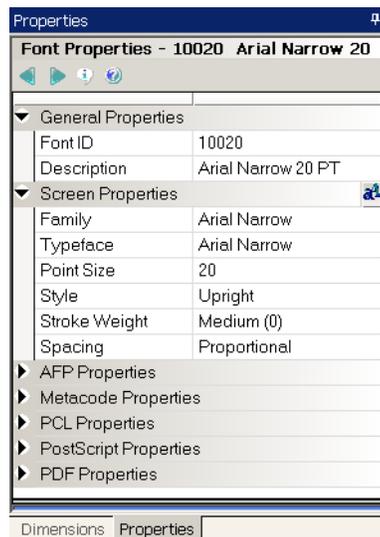
Use these tabs to set the dimensions and other properties.

Here is a sample of the highlighted font.

DEFINING BASIC FONT PROPERTIES

Follow these steps to set general and screen font properties:

1. Choose Manage, System, Fonts and highlight the font you want. Then click the Properties tab.
2. Modify these properties as needed:

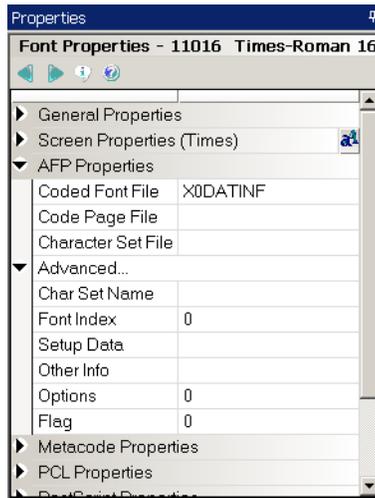


Field	Description
General Properties	
Font ID	When a font is imported into the FXR, it is assigned a unique font ID. The system uses font IDs to track font usage. In addition, some printers require that you refer to a font by its ID number instead of its name. This value is generated when the printer font is imported into the FXR but may be changed if needed. If you change this field, you may need to change the Setup Data field for PCL Properties to use the same font ID.
Description	This description appears when you are selecting fonts.
Screen Properties	
Family	A group of typefaces that share basic design characteristics and encompass many size and style variations such as Courier or Times. The family name helps determine the screen font used for displaying text when running under Windows.
Typeface	A specific member of a typeface family, such as Times-Roman or Times-Bold. The typeface name helps determine the screen font to use for display purposes.
Point Size	A point is a typographical vertical measurement, 72 points are equal to approximately 1 inch. Point size is used similarly in PostScript printing. It does not affect PCL, AFP, or Metacode printing.
Style	Choose Upright or Italic The style helps determine the screen font used for displaying text when running under Windows.
Stroke Weight	Your entry determines the lightness or darkness of the printed typeface. You can choose from Lightest (-7) to Darkest (7). The default is medium (0). The stroke weight helps determine the screen font used for displaying text when running under Windows.
Spacing	Choose Fixed or Proportional The spacing helps determine the screen font used for displaying text when running under Windows

DEFINING AFP FONT PROPERTIES

Follow these steps to set AFP font properties:

1. Choose Manage, System, Fonts and highlight the font you want. Then click the Properties tab.
2. Modify these properties as needed:

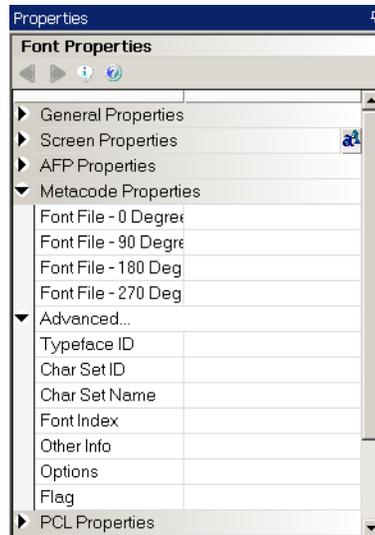


Field	Description
Coded Font File	For printing to AFP printers, this file name must be the name of an AFP coded font file installed on the printer.
Code Page File	Enter the font file name. AFP font file names do not have an extension.
Character Set File	This field may display information, but no entry is required.
Char Set Name	This field may display information, but no entry is required.
Font Index	This field may display information, but no entry is required.
Setup Data	This field may display information, but no entry is required.
Other Info	This field may display information, but no entry is required.
Options	This field may display information, but no entry is required.
Flag	This field may display information, but no entry is required.

DEFINING METACODE FONT PROPERTIES

Follow these steps to set Xerox Metacode font properties:

1. Choose Manage, System, Fonts and highlight the font you want. Then click the Properties tab.
2. Modify these properties as needed:



Field	Description
Font File - 0 Degrees	This is the name of the font file. This name references a font that is not rotated.
Font File - 90 Degrees	This is the name of the font file. This name references a font that is rotated 90 degrees.
Font File - 180 Degrees	This is the name of the font file. This name references a font that is rotated 180 degrees.
Font File - 270 Degrees	This is the name of the font file. This name references a font that is rotated 270 degrees.

The font file name can consist of up to six characters, with no extension. For printing to Metacode printers, these file names must match the name of a Xerox font installed on the printer.

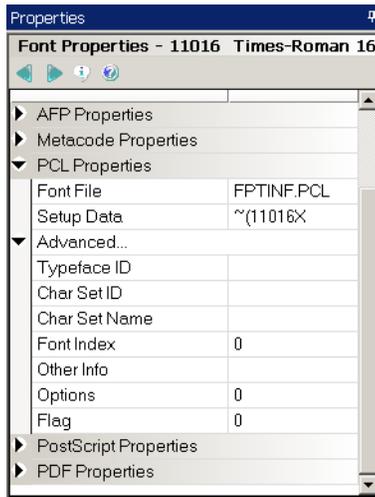
Typeface ID	This field may display information, but no entry is required.												
Char Set ID	<p>A character set (also known as a symbol set) identifies the set of symbols provided by the font. Some printers require that you refer to a character set by its ID number instead of its name. This is used in PostScript printing to build an internal code page. Use W1 for the ANSI code page. This value should match the character set ID specified in the CODEPAGE.INI file. For instance, if you enter 1004 as the code page, enter W1 here.</p> <table border="1"> <thead> <tr> <th>Code page</th> <th>Char Set ID</th> </tr> </thead> <tbody> <tr> <td>1004</td> <td>W1</td> </tr> <tr> <td>863</td> <td>CF</td> </tr> <tr> <td>850</td> <td>PM</td> </tr> <tr> <td>437</td> <td>PC</td> </tr> <tr> <td>37</td> <td>Z1</td> </tr> </tbody> </table>	Code page	Char Set ID	1004	W1	863	CF	850	PM	437	PC	37	Z1
Code page	Char Set ID												
1004	W1												
863	CF												
850	PM												
437	PC												
37	Z1												
Char Set Name	This field may display information, but no entry is required.												
Font Index	This field may display information, but no entry is required.												
Other Info	This field may display information, but no entry is required.												
Options	This field may display information, but no entry is required.												

Field	Description
Flag	This field may display information, but no entry is required.

DEFINING PCL FONT PROPERTIES

Follow these steps to set PCL font properties:

1. Choose Manage, System, Fonts and highlight the font you want. Then click the Properties tab.
2. Modify these properties as needed:



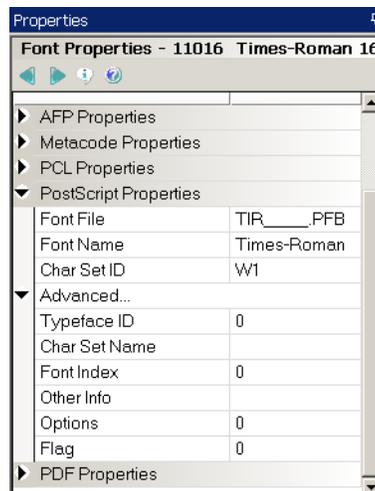
Field	Description
Font File	Enter the name of the font file, including the .PCL extension. For printing to PCL printers, this PCL bitmap font must be located in the FONTLIB master resource directory so it can be downloaded to the printer if requested.
Setup Data	Enter the PCL printer sequence required to select a font. Normally, the setup data must appear in this format: ~ (11018X where 11018 is the font ID. The font ID must match the font ID you defined under General Properties. The X must be uppercase. Your setup data may differ if you are using internal printer fonts. Check your printer manual for the proper setup data sequence for internal printer fonts.
Typeface ID	This field may display information, but no entry is required.

Field	Description
Char Set ID	A character set (also known as a symbol set) identifies the set of symbols provided by the font. Some printers require that you refer to a character set by its ID number instead of its name. This is used in PostScript printing to build an internal code page. Use W1 for the ANSI code page. This value should match the character set ID specified in the CODEPAGE.INI file. For instance, if you enter 1004 as the code page, enter W1 here. Code page Char Set ID 1004 W1 863 CF 850 PM 437 PC 37 Z1
Char Set Name	This field may display information, but no entry is required.
Font Index	This field may display information, but no entry is required.
Other Info	This field may display information, but no entry is required.
Options	This field may display information, but no entry is required.
Flag	This field may display information, but no entry is required.

DEFINING POSTSCRIPT FONT PROPERTIES

Follow these steps to set PostScript font properties:

1. Choose Manage, System, Fonts and highlight the font you want. Then click the Properties tab.
2. Modify these properties as needed:



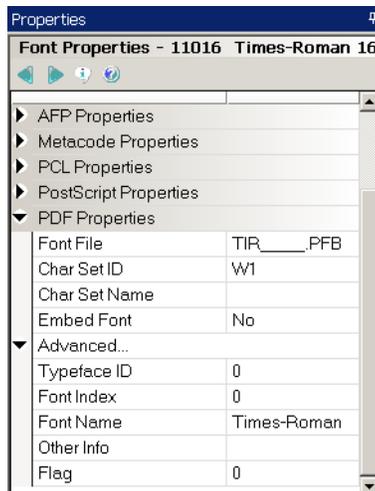
Field	Description
Font File	Enter the name of the font file, including the .PFB extension. For printing to PostScript printers, this PostScript Type 1 font must be located in the FONTLIB master resource directory so that it can be downloaded to the printer if requested.

Field	Description												
Font Name	Enter the name of the font, such as Times-Roman. Studio fills this field when you insert a PostScript font. The font name also appears in the font .AFM file. All type 1 PostScript fonts require two files for each font family name: .AFM and .PFB.												
Char Set ID	A character set (also known as a symbol set) identifies the set of symbols provided by the font. Some printers require that you refer to a character set by its ID number instead of its name. This is used in PostScript printing to build an internal code page. Use W1 for the ANSI code page. This value should match the character set ID specified in the CODEPAGE.INI file. For instance, if you enter 1004 as the code page, enter W1 here. <table border="1"> <thead> <tr> <th>Code page</th> <th>Char Set ID</th> </tr> </thead> <tbody> <tr> <td>1004</td> <td>W1</td> </tr> <tr> <td>863</td> <td>CF</td> </tr> <tr> <td>850</td> <td>PM</td> </tr> <tr> <td>437</td> <td>PC</td> </tr> <tr> <td>37</td> <td>Z1</td> </tr> </tbody> </table>	Code page	Char Set ID	1004	W1	863	CF	850	PM	437	PC	37	Z1
Code page	Char Set ID												
1004	W1												
863	CF												
850	PM												
437	PC												
37	Z1												
Typeface ID	This field may display information, but no entry is required.												
Char Set Name	This field may display information, but no entry is required.												
Font Index	This field may display information, but no entry is required.												
Other Info	This field may display information, but no entry is required.												
Options	This field may display information, but no entry is required.												
Flag	This field may display information, but no entry is required.												

DEFINING PDF FONT PROPERTIES

Follow these steps to set PDF font properties:

1. Choose Manage, System, Fonts and highlight the font you want. Then click the Properties tab.
2. Modify these properties as needed:

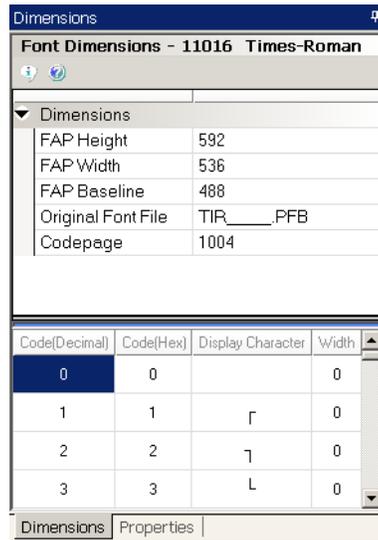


Field	Description												
Font File	This is the name of the font file.												
Char Set ID	<p>A character set (also known as a symbol set) identifies the set of symbols provided by the font. Some printers require that you refer to a character set by its ID number instead of its name. This is used in PostScript printing to build an internal code page. Use W1 for the ANSI code page. This value should match the character set ID specified in the CODEPAGE.INI file. For instance, if you enter 1004 as the code page, enter W1 here.</p> <table><thead><tr><th>Code page</th><th>Char Set ID</th></tr></thead><tbody><tr><td>1004</td><td>W1</td></tr><tr><td>863</td><td>CF</td></tr><tr><td>850</td><td>PM</td></tr><tr><td>437</td><td>PC</td></tr><tr><td>37</td><td>Z1</td></tr></tbody></table>	Code page	Char Set ID	1004	W1	863	CF	850	PM	437	PC	37	Z1
Code page	Char Set ID												
1004	W1												
863	CF												
850	PM												
437	PC												
37	Z1												
Char Set Name	This field may display information, but no entry is required.												
Embed Font	Choose Yes if you want the system to embed this font. The default is No.												
Typeface ID	This field may display information, but no entry is required.												
Font Index	This field may display information, but no entry is required.												
Font Name	This field may display information, but no entry is required.												
Other Info	This field may display information, but no entry is required.												
Flag	This field may display information, but no entry is required.												

DEFINING FONT DIMENSIONS

Follow these steps to define font dimensions:

1. Choose Manage, System, Fonts and highlight the font you want. Then click the Dimensions tab.
2. Modify these properties as needed:



Field	Description
FAP Height	This is the largest font character height (in FAP units, 2400 dots/inch) The font height affects the size of text displayed when running under Windows.
FAP Width	This is the largest font character width (in FAP units) The font width affects the width of text displayed when running under Windows.
FAP Baseline	This is the largest font character base line (in FAP units) The baseline is measured from the top of the largest character to the imaginary line that the character appears to rest on. The font baseline affects the positioning of text displayed when running under Windows.
FXR File	Under Windows, the font file name has the extension FNT. This field is not currently used.
FON File	Under Windows, the font width table file name has the extension FON. This field is not currently used.
Code Page	Under Windows, the system uses the ANSI code page. Normally, this field is set to 1004 or is left blank.
Width Table	The width table is calculated from the printer font file and is used to determine character spacing when displaying text. Fixed pitch fonts use the same width for each character. The width is measured in FAP units (2400 to an inch).

The Dimensions tab provides information about the font file. Information on this window should match the characteristics in the font file. If you enter dimension information which does not match the actual font file, only the bitmap font window display changes. The actual dimensions of the font in the font file do not change, nor do print dimensions of the font change.

IMPORTING FONTS

Studio provides a wizard to guide you through the steps of importing fonts into your font cross-reference (FXR) file. Studio lets you import these types of fonts:

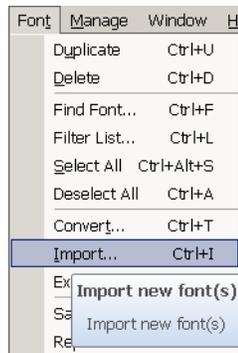
- AFP
- Fonts in another FXR
- PCL
- PostScript
- TrueType
- Metacode

For instance, you can use this wizard to import TrueType fonts into your FXR file.

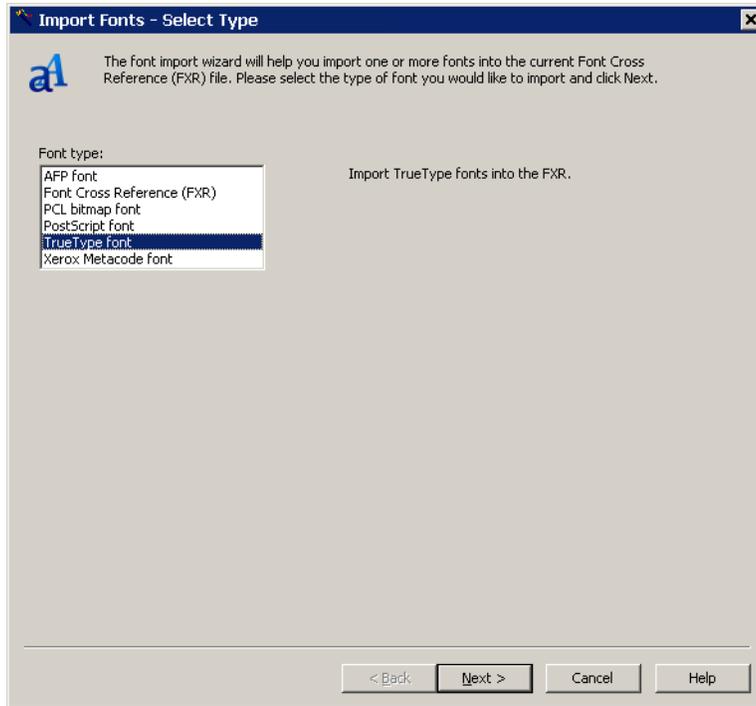
Note To import TrueType or PostScript fonts, first place a copy of the fonts you want to import from the Windows\Fonts directory into a temporary directory. You will then import those fonts from the temporary the directory.

Follow these steps to add fonts to your font cross-reference (FXR) file:

1. Open the workspace which uses the font cross-reference (FXR) file to which you want to add fonts.
2. Choose Manage, System, Fonts.
3. From the Font menu, choose Import



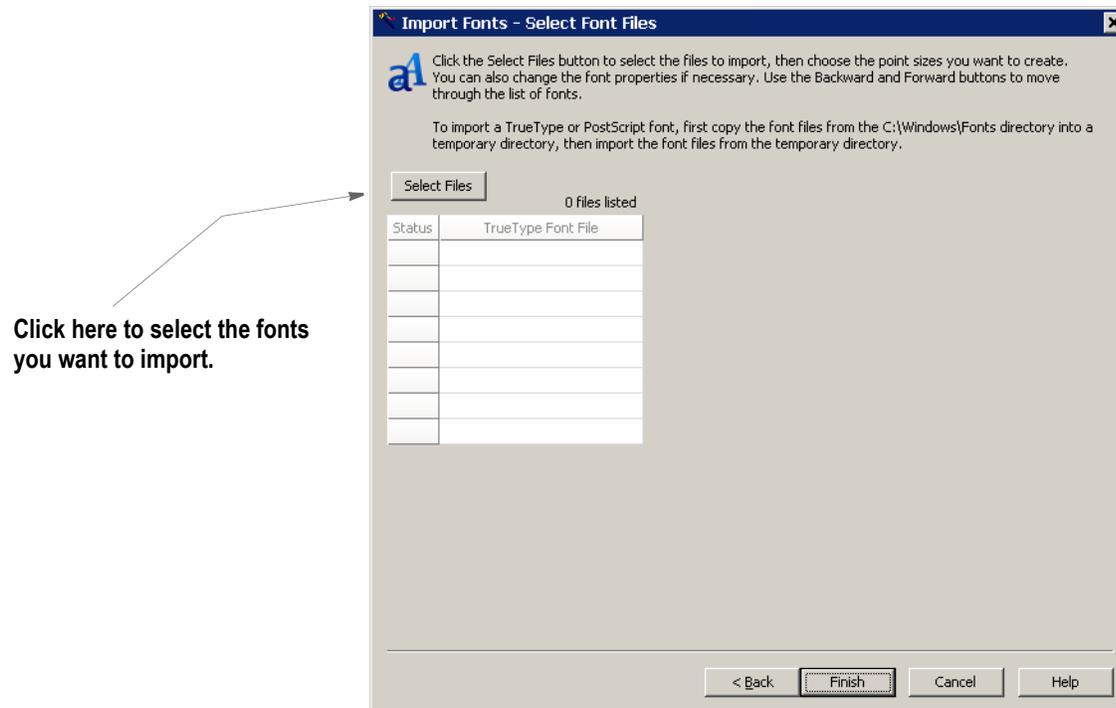
The Import Fonts - Select Type window appears:



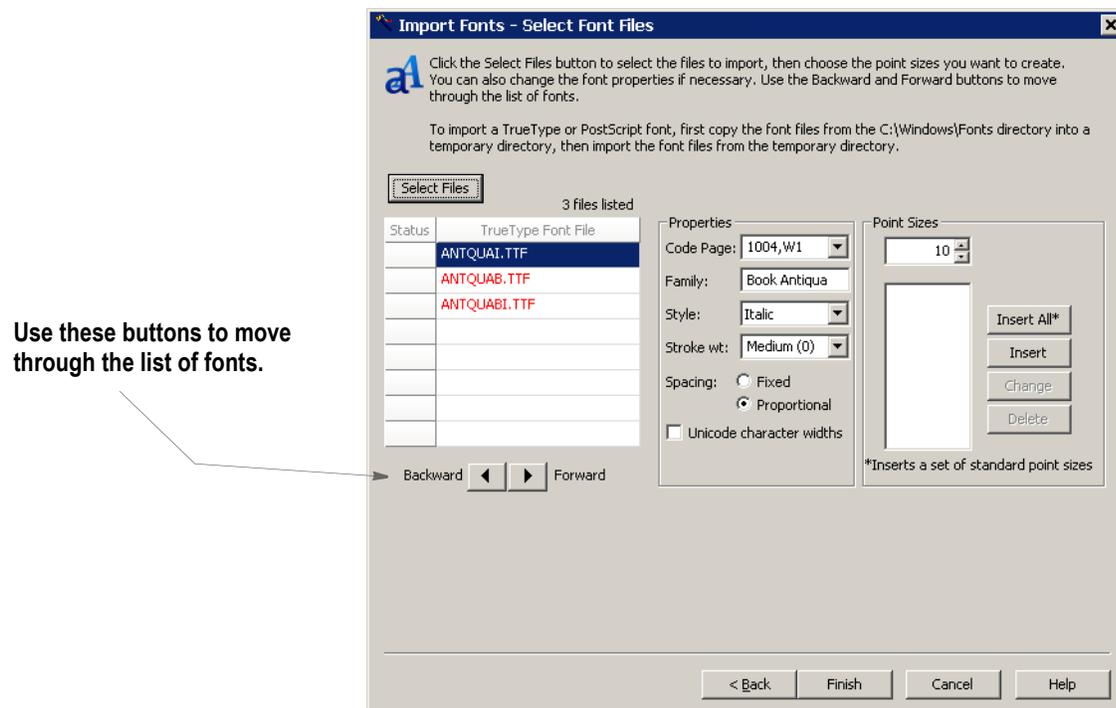
Select the type of font you want to import and click Next. You can choose from these types of fonts:

Font	Description
AFP	AFP fonts consist of a code font file, a code page file, and a character set file. These fonts are typically used on IBM AFP printers.
Font Cross-Reference (FXR)	Lets you import fonts from another FXR file.
PCL	This option imports PCL bitmap fonts into your FXR. These fonts are typically used on PCL printers.
PostScript	This option imports PostScript bitmap fonts into your FXR. These fonts are typically used on PostScript printers.
TrueType	This option imports TrueType fonts into your FXR. These fonts are common on Windows computers.
Xerox Metacode	This option imports Metacode fonts into your FXR. These fonts are typically used on Xerox Metacode printers.

The Import Fonts - Select Font Files window appears:



4. Click Select Files to select the fonts you want to import. The Import Fonts - Select Font Files window changes after you select the fonts you want to import:



5. For each font, specify the point sizes you want to import. Click Insert All to import all point sizes. You can also change the font's properties if necessary.
6. Click Finish to import the fonts into your FXR.

Chapter 15

Testing Your Work

This chapter discusses how you can test the forms you have created and analyze system output. Testing the forms involves using the Test Scenarios manager. Analyzing the output involves using Printstream Analyzer.

This chapter includes these topics:

- *Overview on page 518*
- *Using the Screen on page 520*
- *Creating a Test Profile on page 527*
- *Running a Test on page 530*
- *Changing Test Properties on page 534*
- *Modifying the AFGJOB File on page 535*
- *Deleting a Test on page 536*
- *Using the Printstream Analyzer on page 537*

OVERVIEW

When you use the Test Scenario manager to run a test, keep in mind Studio does not execute the GenTrn and GenPrint processing steps. When you select an AFGJOB.JDT file you want to use for test, Studio creates a copy of that file and modifies it to exclude some rules and include others. One of the included rules eliminates the GenTrn requirement. Aside from those top level rules replaced in the AFGJOB sections, all the other rules remain where they are.

Note Please note that your original AFGJOB.JDT file is not changed. Studio makes a copy of this file and modifies it for testing purposes. Also, unlike an actual run, no files are saved to disk.

These rules are not imported into the AFGJOB.JDT file for a test run:

- LoadExtractData
- UpdatePOLFile
- ServerJobProc
- InitPrint
- InsNAHdr
- PrintFormset
- WriteOutput
- WriteNAFile

They could cause problems with the way Studio runs the test or else are simply not applicable in the case of the ServerJobProc rule.

Note The Test option does not support custom Proc rules in the JDT file.

Also keep in mind that you do not have to check in resources to include them in a test run. This lets you easily experiment with resources you are developing and only check them in once you have corrected any problems.

Breakpoints

The breakpoint process stops on most other rules defined in the AFGJOB.JDT file. The LoadExtractData rule is, however, an exception. Since a test has no GenTrn processing step, the extract data is already loaded. If you execute another LoadExtractData rule, it means you are reading a *second* transaction before processing the first — effectively skipping ever other transaction in your extract.

Establishing origins and pagination are tasks that happen sequentially. The origin for each is determined when processing encounters that section, until that time the section may appear in an incorrect position. Furthermore, the final pagination process does not take place until you get to a form set- or transaction-level rule on the reverse processing, such as the PaginateAndPropagate rule. This means you may see some really long pages, until this occurs.

The breakpoint that occurs when form set reloads is a new one. This happens anytime the list of form set sections changes outside the normal triggering process. So, if you have a DAL rule or some other rule that adds or deletes a form or section beyond normal triggering, this event will be a breakpoint.

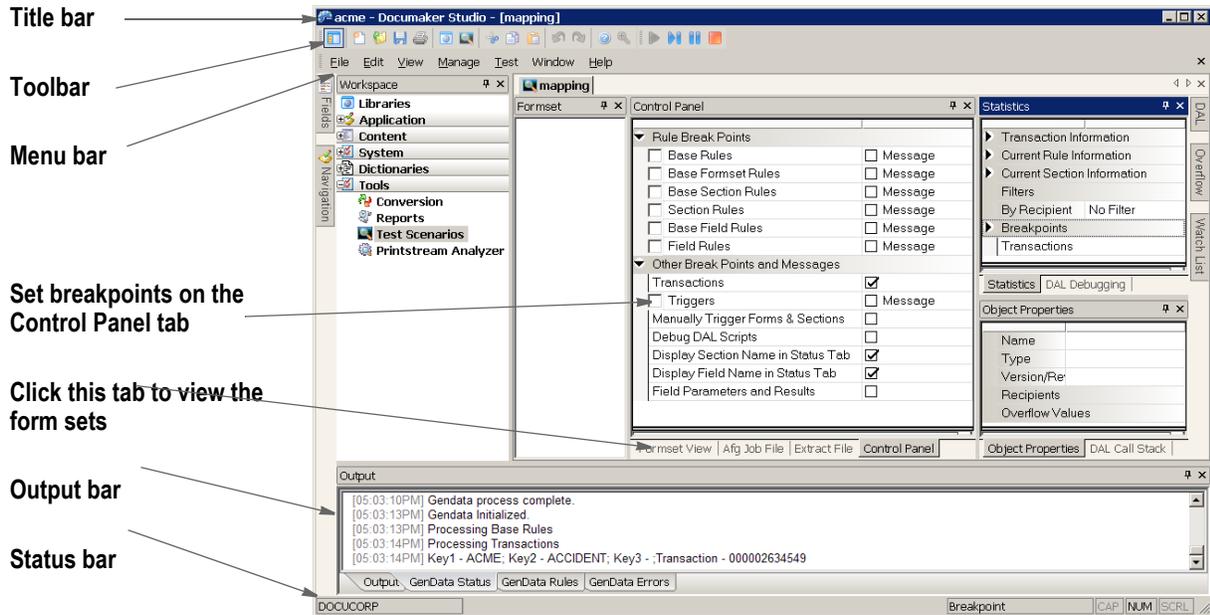
Messaging can give you more feedback of exactly what processed and in what order. When things are going along nicely, you may want to turn messaging off to reduce the clutter you see on the screen. When things are not working as expected, you would probably turn on more messaging to get an idea of what is happening.

Note Changes to a graphic via DAL or a rule may not appear in the Test view until the final document pagination occurs. The change can appear earlier if the section that contains the graphic is also forced *in-lined* due to the growth of a text area.

USING THE SCREEN

Studio places all the tools you need to create test profiles and run tests at your fingertips. The screen is your testing area. It is important to become familiar with the general screen layout and parts of the screen. Understanding the screen layout will help you work quickly and efficiently.

The main window that appears when you are testing your resources is shown here.



Item	Description
Title bar	The title bar displays the name of the workspace you have open, followed by Documaker Studio, and then the name of the test you are running.
Toolbar	The toolbar contains a row of icons that provide quick access to common options.
Menu bar	The menu bar provides access to the pull-down menus.
Status bar	The Status bar shows you any messages generated by the test.
Breakpoints	The Control Panel tab lets you set breakpoints that tell Studio when to pause the test.
View area	Depending on the tab you choose, Studio shows you the forms as they are processed, your AFGJOB.JDT file, your extract file, or the DAL script you are executing.

USING THE MENU BAR

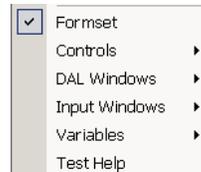
This section introduces you to the pull-down menus which include additional options or are only available when you are working with tests. When you open a test profile, Studio adds the Test menu to your menu bar.

Note For information on the standard menus and menu options which are always available, see *Using System Menus on page 16*.

Menu	Description
View	This menu include additional options you can use when running a test.
Test	The Test menu provides you with options for managing your tests.

Using the View Menu

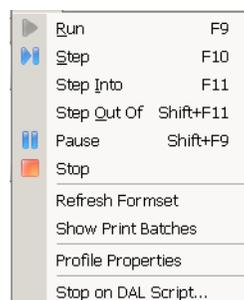
The View menu provides additional options you can use when running a test. When you select Test, this menu appears:



Option	Description
Formset	Use to display or hide the form set view.
Controls	Choose this option to display or hide the form set tree, control panel, statistics, object properties, and advanced debug controls.
DAL Windows	Choose this option to display or hide the DAL call stack, DAL variables, and DAL debugging window.
Input Windows	Choose this option to display or hide the AFGJOB.JDT file and the extract file.
Variables	Choose this option to display or hide DAL, GVM, overflow, and record dictionary variables. You can also display or hide the watch list.
Test Help	Choose this option to see Help information about the Test Scenario manager.

Using the Test Menu

The Test menu provides you with tools to manage the test profiles you create and clear breakpoints. When you select Test, this menu appears:



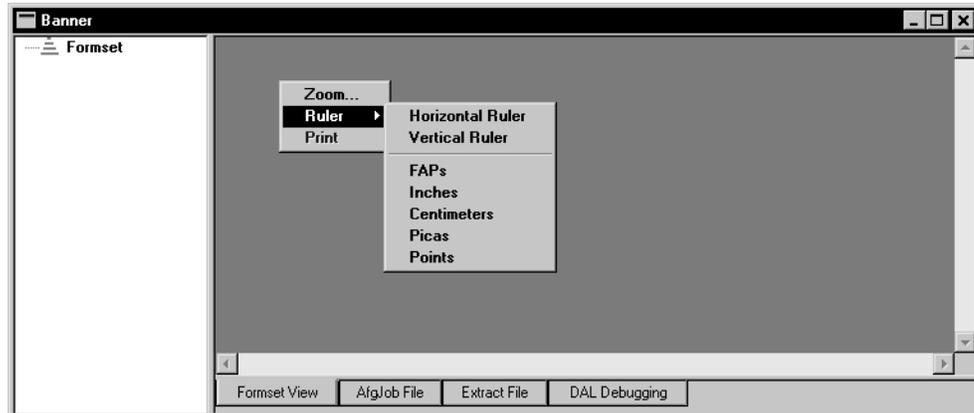
Option	Description
Run	Start the test.
Step	Proceed one step at time

Option	Description
Step Into	
Step Out Of	
Pause	Pause the test.
Stop	Stop the test.
Refresh Formset	
Show Print Batches	
Profile Properties	Use to edit the properties you have defined for a test profile.
Stop on DAL Script	

USING THE RIGHT-CLICK MENUS

Formset View

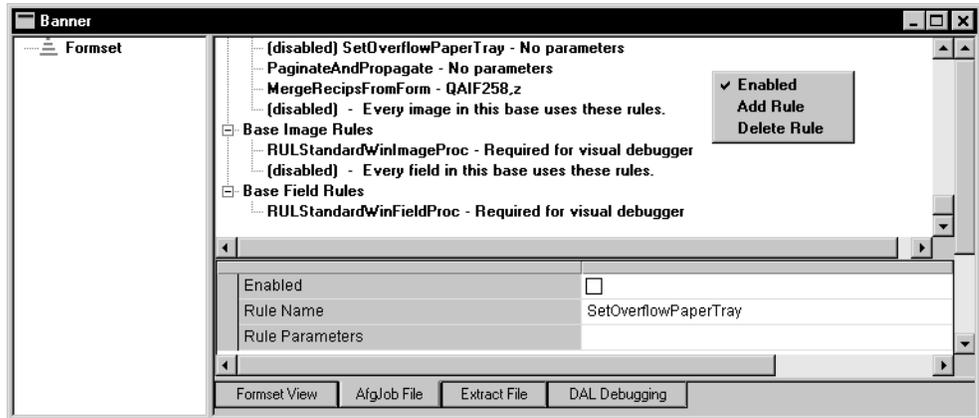
When running tests, if you click on the Formset View tab to watch the form sets change as they are processed, you can right click to see the following menus.



Option	Description
Zoom	Lets you see a larger or smaller version of the form set.
Ruler	
Horizontal	Select to display the horizontal ruler. Select again to remove the ruler.
Vertical	Select to display the vertical ruler. Select again to remove the ruler.
FAPs	Select to use FAP units (2400 per inch) on the ruler.
Inches	Select to use inches on the ruler.
Centimeters	Select to use centimeters on the ruler.
Picas	Select to use picas on the ruler.
Points	Select to use points on the ruler.
Print	Prints a copy of the form set.

AFGJOB File

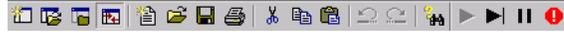
When running tests, if you click on the AFGJOB File tab to view the JDT file, you can right click in the view area to see the following menus.



Option	Description
Enabled	Lets you turn on or off the processing of a rule. The check mark indicates the rule is enabled and will be executed during the test.
Add Rule	Lets you add a rule.
Delete Rule	Lets you remove a rule form the AFGJOB.JDT file.

USING THE TOOLBAR

The toolbar provides a quicker way to select options that may be listed on a drop down menu. Here is an example of the toolbar shown when you are working with test profiles:



Standard toolbar icons

Shown below are the toolbar icons that are always available. The icons are listed as they appear, from left to right.



Icon	Name	Description
	New Workspace	Creates a workspace.
	Open Workspace	Opens a workspace.
	Close Workspace	Closes an open workspace.
	Toggle Workspace	Toggles between displaying and hiding the workspace.
	New	Creates a file.
	Open	Opens a file.
	Save	Saves the open file. For instance, if you are making changes to the AFGJOB.JDT file, this would save those changes to the JDT file used for the test — not your original JDT file.
	Print	Prints the current object.
	Cut	Removes an object and places it on the clipboard.
	Copy	Copies an object and places it on the clipboard.
	Paste	Places an object from clipboard onto the current file.
	Undo	Reverses your last action
	Redo	Reverses last undo.
	Help	Displays the Help window

Test toolbar icons

Shown below are the toolbar icons that appear when you are working with test profiles.



Icon	Name	Description
	Start	Lets you start a test.
	Step	Lets you continue to the next breakpoint
	Pause	Lets you pause a test. Click Start or Step again to resume the test.
	Stop	Lets you cancel a test.

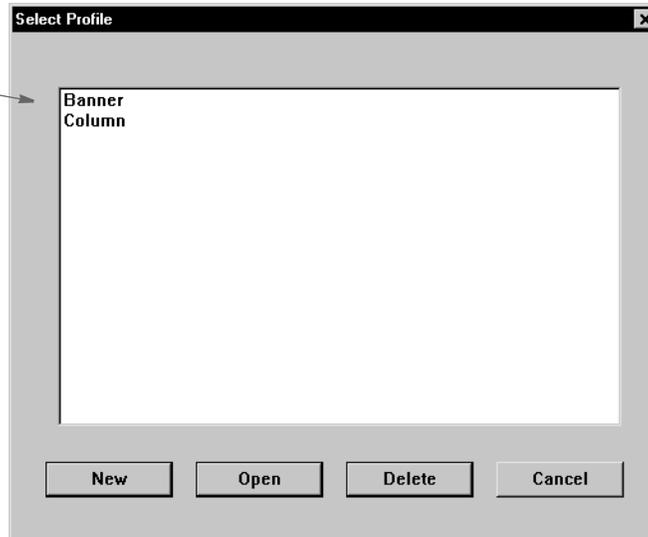
Note You can hide these icons by choosing View, Test Remote Control.

CREATING A TEST PROFILE

Studio includes a wizard to help you set up a test profile to run. This wizard starts automatically if you have no tests set up. If you do have test profiles set up, the following window appears:

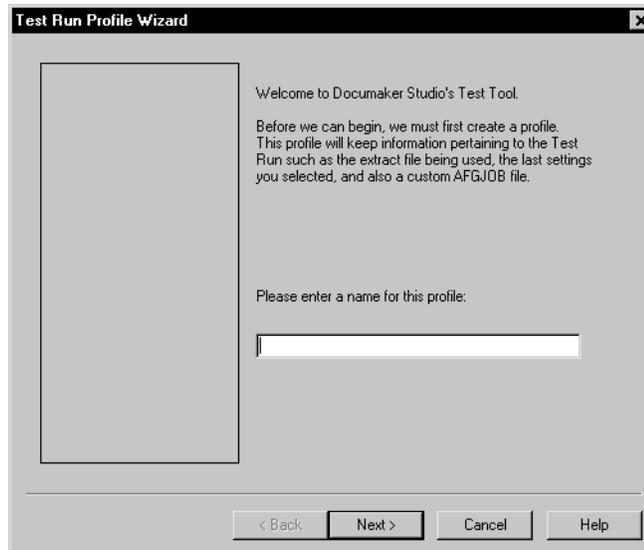
These tests have already been set up.

To choose one, double click on it or highlight it and click Open.



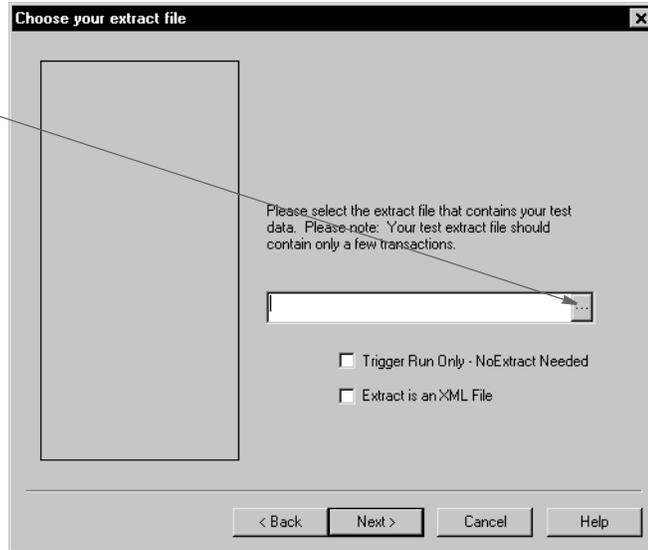
To create a new test profile, click New and follow these steps:

1. When you Click New, the following wizard appears.



Enter a name for the test profile so you can run it later without having to set up all the test parameters again. Click Next. The following window appears.

Click here to browse for an extract file.



2. Here you select an extract file to use in the test. If you only want to test the triggers, check the Trigger Run Only field. For this kind of test, you do not need an extract file.

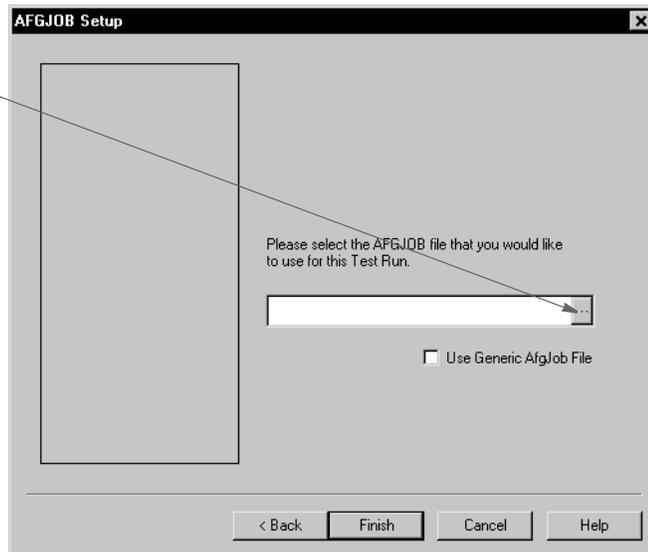
Note See *Running a Trigger-Run Only Test* on page 533 for more information on running a trigger-only test.

If your extract file is in XML format, be sure to check the Extract is an XML File field.

Your extract file can include as many transactions as you like, but keep in mind that the more you include, the longer it will take for the test to run. The transactions you do include should be representative of the data you will actually process to get the best results.

Click Next when you are finished. The following window appears.

Click here to browse for a job definition (JDT) file.



- On this window you select a job definition file (JDT). This file tells the system how to run as it performs the test. A JDT file is a text file which tells the system which job and form set rules to use as it processes your data. The rules defined in the JDT file are run before the system runs rules assigned to specific fields. An example of a JDT file is the AFGJOB.JDT file.

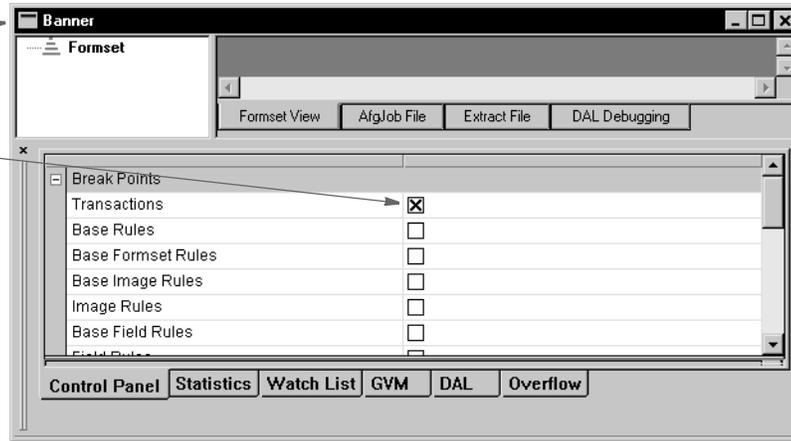
If you do not have a specific JDT file you want to use, click the Use Generic Afgjob File option.

Click Finish to complete the definition of the test profile. The system then shows you the test window:

Studio displays the name of the test profile here.

Click here to define breakpoints.

This part of the Test window shows you statistics.



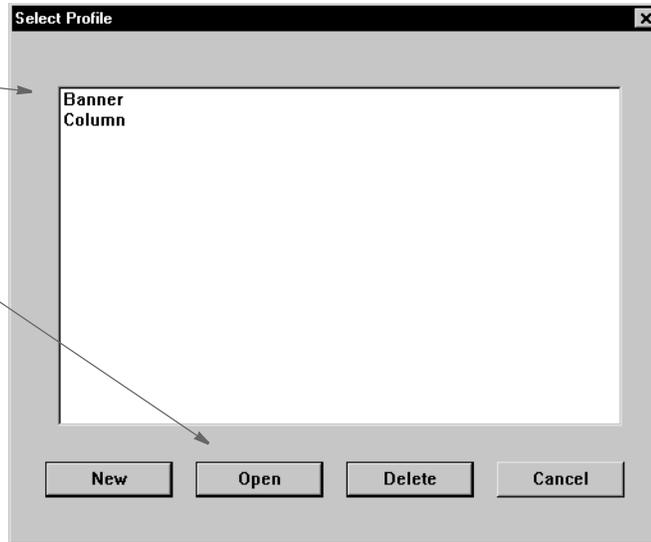
RUNNING A TEST

Follow these steps to run a test profile you have already defined.

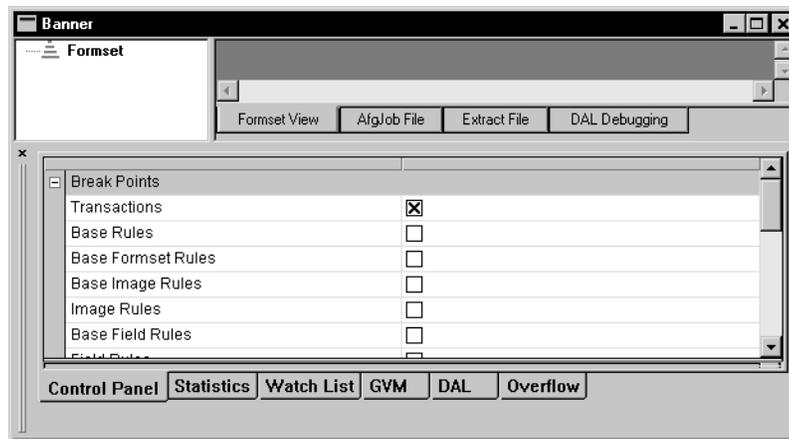
1. Open a workspace, then click on Tests. The following window appears.

These tests have already been set up.

To run a test, highlight it and click Open.

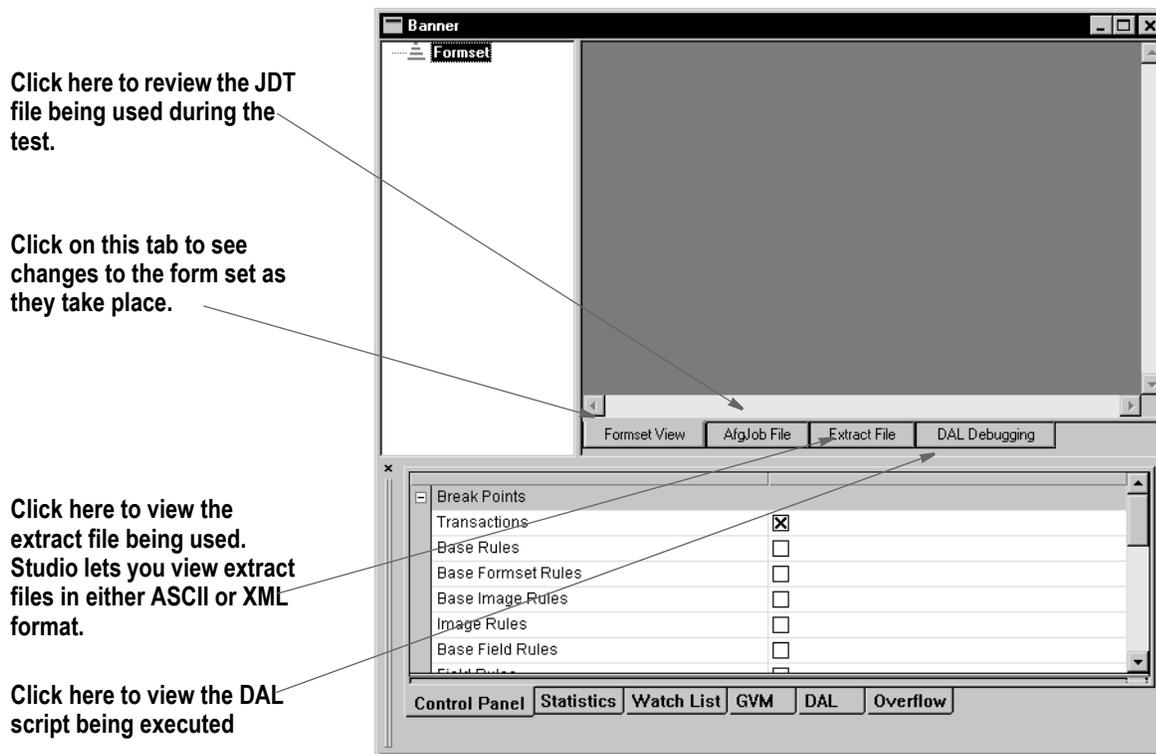


2. Highlight the test profile you want to run, then click Open. Studio displays a window similar to this one.



To better monitor the test, you may want to resize some of the components of the window, like this:

Note These examples only show the Test window itself. You can close the workspace to get additional room. When you need to restore the workspace, just click View, Workspace.

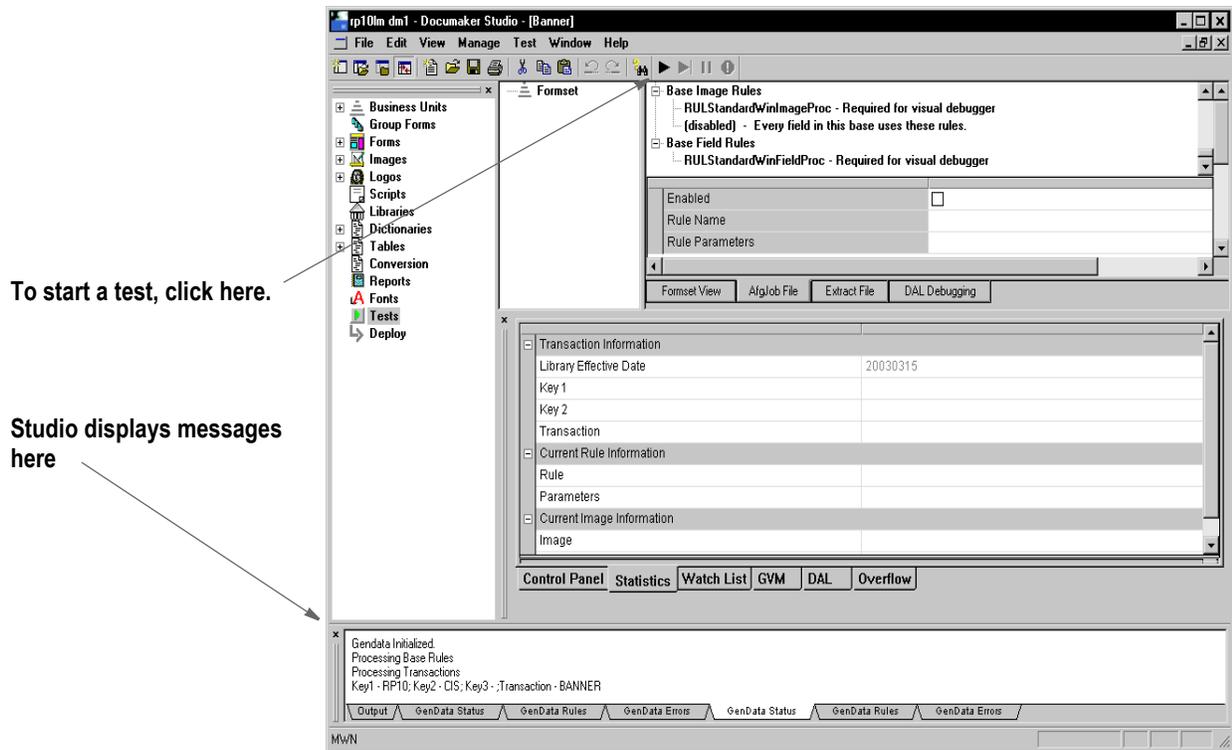


3. You can click on these tabs to change the view:

To...	Click
Set breakpoints on transactions and rules, turn messages on or off, manually trigger form and sections, and debug DAL scripts. Keep in mind you can change breakpoints while running a test.	Control Panel
View information on the current transaction, rule, and section,	Statistics
Set up a list of DAL, GVM, or overflow variables to watch. To add a variable, click on the plus sign (+). To remove a variable, click on the red X .	Watch List
Define the GVM variables for the watch list. To add a variable, click on the plus sign (+).	GVM
Define the DAL variables for the watch list. To add a variable, click on the plus sign (+).	DAL
Define the overflow variables for the watch list. To add a variable, click on the plus sign (+).	Overflow

Note To see a list of the GVM, DAL, and overflow variables, click the Start icon on the toolbar. As Studio runs through the test, it adds the variables it encounters onto the appropriate tab. Once the test finishes, you can then add the variables you want to watch onto your watch list.

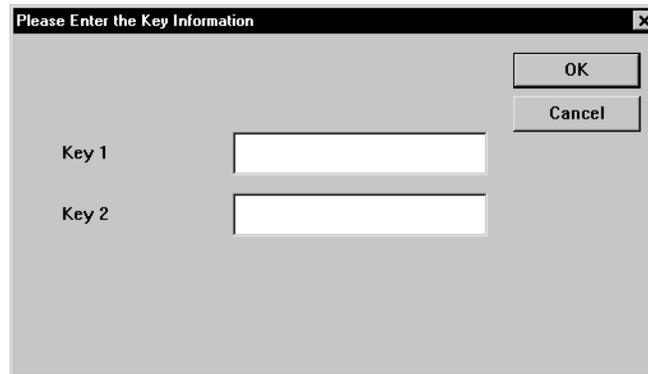
4. To start the test, click Start on the toolbar, as shown here:



When you start the test, the system runs and relevant messages are shown in the status bar.

RUNNING A TRIGGER-RUN ONLY TEST

When you run a trigger-only test, Studio displays the following window:

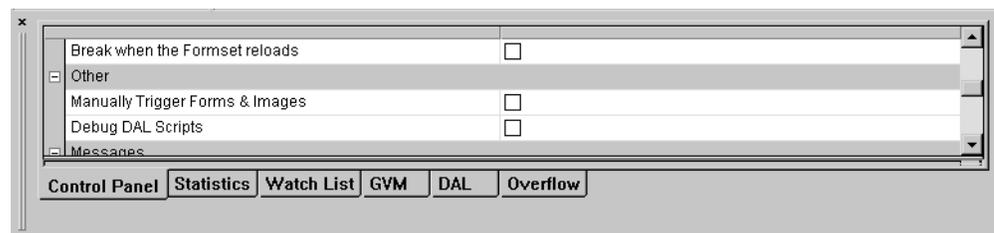


A dialog box titled "Please Enter the Key Information" with a close button (X) in the top right corner. It contains two text input fields labeled "Key 1" and "Key 2". To the right of the input fields are two buttons: "OK" and "Cancel".

Use this window to enter the Key1 and Key2 fields you want to trigger on.

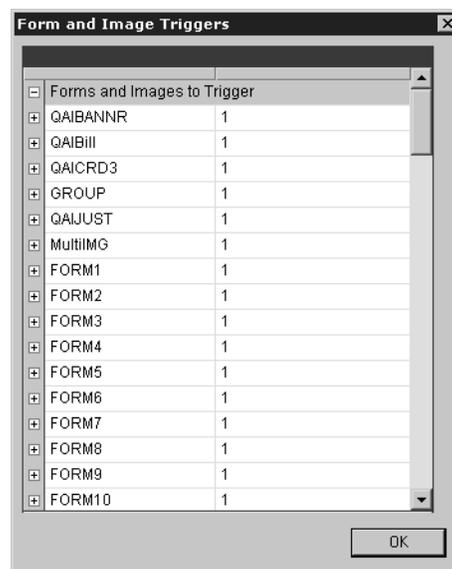
MANUALLY TRIGGERING

You can use the Manually Trigger Forms and Sections option to tell Studio you want to manually trigger copies of the forms that comprise the form set.



A dialog box with a list of options and checkboxes. The options are: "Break when the Formset reloads", "Other", "Manually Trigger Forms & Images", "Debug DAL Scripts", and "Messages". The "Manually Trigger Forms & Images" option is selected. Below the list are several tabs: "Control Panel", "Statistics", "Watch List", "GVM", "DAL", and "Overflow".

When you choose this option and start a test, Studio displays the following window so you can specify the forms and sections you want to trigger on.



A dialog box titled "Form and Image Triggers" with a close button (X) in the top right corner. It contains a list of forms and images to trigger, with a column for the form name and a column for the number of copies to trigger. The list is as follows:

Forms and Images to Trigger	
QAIBANNR	1
QAIBIII	1
QAICRD3	1
GROUP	1
QAJJUST	1
MultIMG	1
FORM1	1
FORM2	1
FORM3	1
FORM4	1
FORM5	1
FORM6	1
FORM7	1
FORM8	1
FORM9	1
FORM10	1

An "OK" button is located at the bottom right of the dialog box.

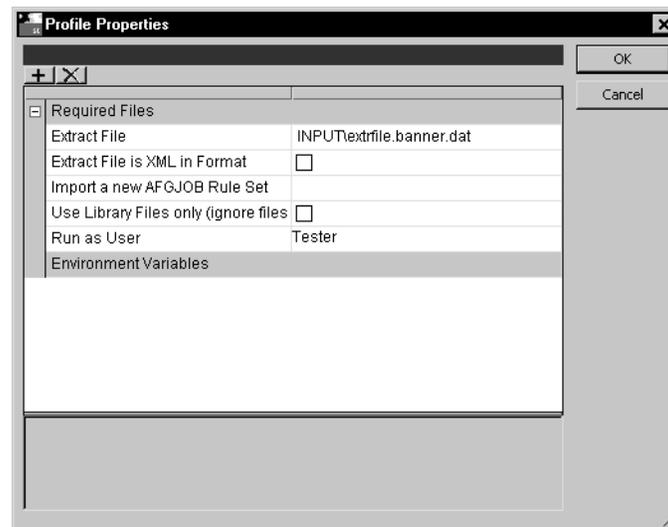
This can be useful when you need to test complicated scenarios without robust data.

CHANGING TEST PROPERTIES

You can change the properties associated with a test using the Test, Profile Properties option. This includes modifying these properties:

Property	Description
Extract File	Here you can specify a different extract file to use during the test.
Extract File is in XML Format	Check this box if the extract file is in XML format.
Import a new AFGJOB Rule Set	Here you can specify a different AFGJOB.JDT file to use during the test.
Use Library Files only	Check this box if you only want to use resource files stored in the library.
Run as User	This lets you specify the user ID you want the test to run under.

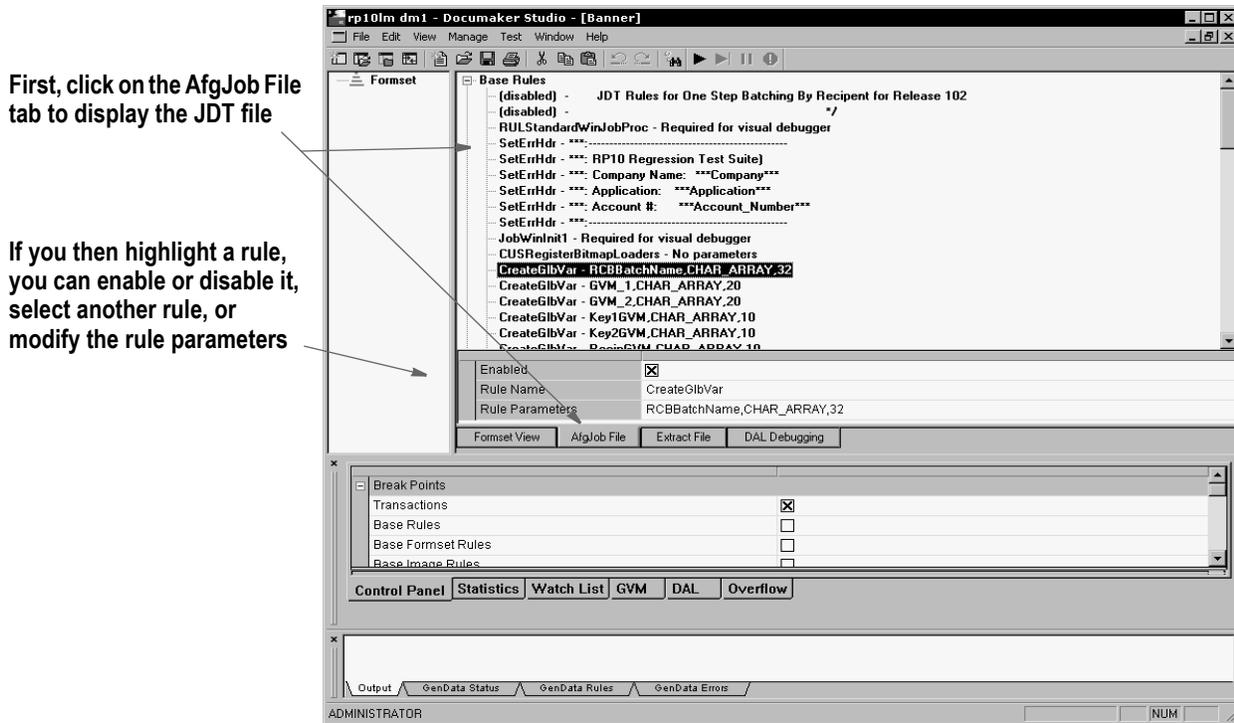
When you choose this option, the following window appears:



Make the appropriate changes and click Ok to accept them or Cancel to exit without making changes.

MODIFYING THE AFGJOB FILE

You can modify the AFGJOB.JDT file Studio uses when running a test by clicking on the AfgJob File tab, as shown here:



First, click on the AfgJob File tab to display the JDT file

If you then highlight a rule, you can enable or disable it, select another rule, or modify the rule parameters

First, highlight the rule you want to modify, then...

To	Do this
Enable or disable a rule	Click the check box in the Enabled field. If you disable a rule, Studio does not remove it from the file, but will skip over it when running a test.
Change the rule	Enter a new rule in the Rule Name field.
Change rule parameters	Enter the new parameters in the Rule Parameters field.
Delete a rule	With the rule highlighted, right click and choose Delete Rule. Remember that you can also disable the rule if you want Studio to skip it during a test. The Delete Rule option removes it from the JDT file.
Save your changes	Click the Save icon on the toolbar.

Note You cannot modify extract files or DAL scripts when running tests.

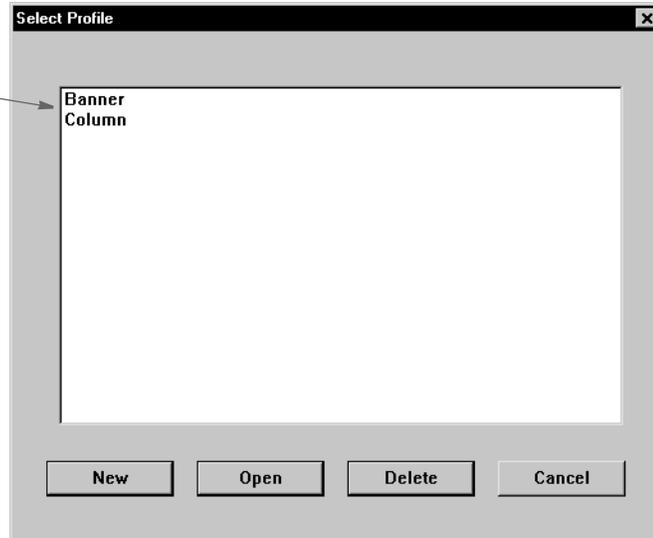
DELETING A TEST

Follow these steps to delete a test.

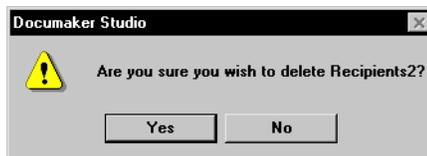
1. Click on Tests in the Workspace. The following window appears.

These tests have already been set up.

To delete a test, highlight it and click Delete.



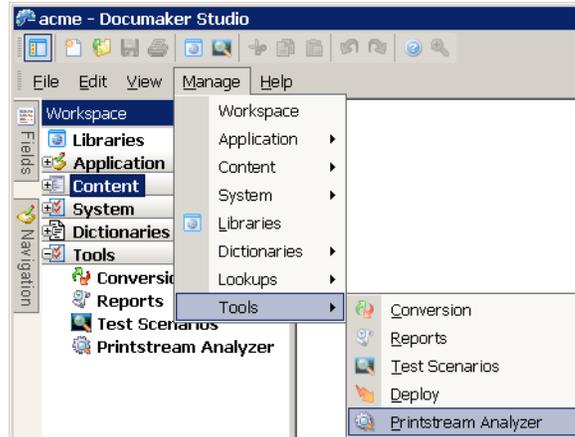
2. Highlight the test you want to delete, then click Delete. Studio lets you confirm your decision to delete the test.



Click Yes to delete the test.

USING THE PRINTSTREAM ANALYZER

You can use the Printstream Analyzer to open and analyze AFP or Metacode print streams. The Printstream Analyzer is available from the workspace tree. You can also start it using the Manage, Tools menu.



Note You do not have to open a workspace to use the Printstream Analyzer. To customize how Printstream Analyzer works, see *Setting Up Printstream Analyzer on page 64*.

Printstream Analyzer recognizes and opens AFP and Metacode print streams that have a variety of record delimiter types.

- AFP may be stream (unformatted), variable 2-byte, variable-blocked, CR-LF, Barr1 and Barr2.
- Metacode may be variable 1-byte, variable 2-byte, variable 2-byte inclusive, variable 4-byte inclusive, variable-blocked, xermet, CR-LF, Barr1, Barr2 or Mobius Metacode. Please note that for Mobius Metacode files, most text appears as Arial since Mobius files do not usually specify fonts in a DJDE command.

These record delimiter types are supported in Documaker by using the indicated OutMode INI option setting:

For this record delimiter type	Set the OutMode option to
AFP stream (unformatted)	This is the default OutMode setting.
AFP variable 2-byte	MRG2
AFP variable-blocked	MRG4
Metacode variable 2-byte	MRG2
Metacode variable-blocked	MRG4
Metacode Barr1	BARR
Metacode Barr2	BARRWORD
Mobius Metacode	Mobius

Note You do not need to make changes to the OutMode setting so Printstream Analyzer can analyze an AFP or Metacode print stream. You only need to set the OutMode option in Documaker so the GenPrint program will produce an AFP or Metacode print stream using a specific record type delimiter.

Printstream Analyzer does not require or access device fonts. It depends on font naming conventions to determine the characteristics of a Windows display font. When a font has an unknown naming convention, Printstream Analyzer uses Arial.

Naming conventions

Type	Example
AFP	
Monotype fonts	X0DAUNN6, C1FAUNN8
Expanded core fonts	C1N20800
Compatibility fonts	X0GT12, X0AOA
Others	C1TR110, C1COD39P
Metacode	
Monotype fonts	FXUNN8
Xerox 9700 fonts	P07TDC, L0112B
Xerox Arial MT fonts	A003GP
Others	UN110E, TR212E, COD39P

Note Printstream Analyzer does not use device font metrics. Size differences between device fonts and Windows display fonts can cause positioning errors. This is expected behavior that will be most apparent in print stream files that contain many relative moves.

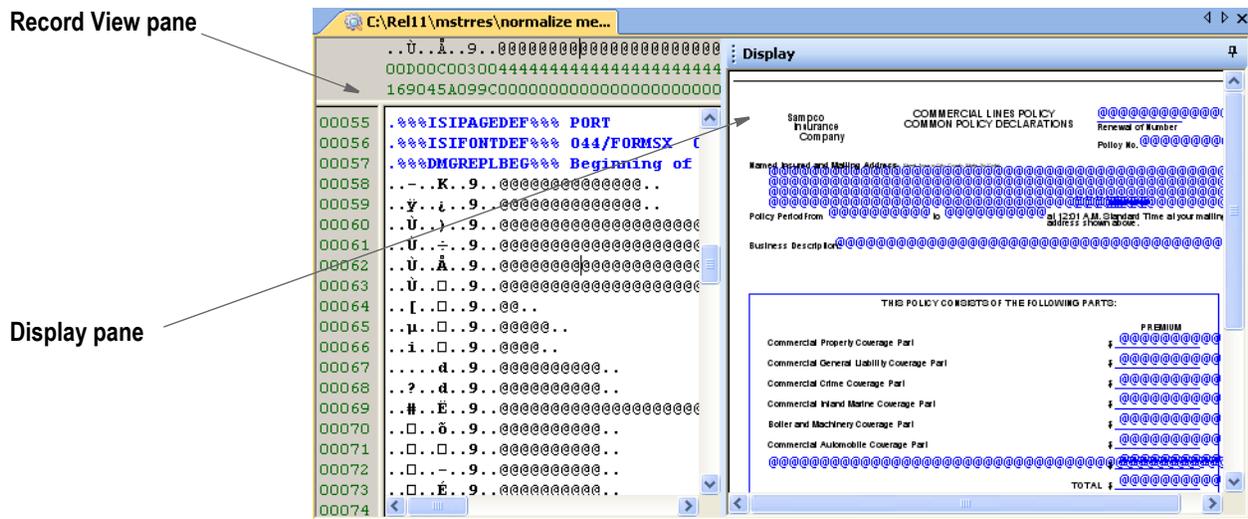
USING THE RECORD VIEW AND DISPLAY PANES

The Record View pane lets you view the records in a print stream. Since AFP and Metacode print streams contain many binary, non-displayable values, much of the content of the print stream appears as a series of periods (. . .). The Record View pane provides a hexadecimal display of the bytes of the selected print stream record so you can see the values shown as periods.

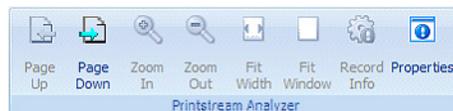
The Display pane, which you can dock and pin, provides a WYSIWIG view of the page that contains the records shown in the Record View pane. The text you select in the Record View pane is highlighted in the Display pane.

The line numbers in the Record View pane show horizontal dividers to indicate page breaks. You can click a line number to select an entire print record. Text contained in the record is then highlighted in the Display pane. A description of current record appears in the status bar.

Here is an example of the Record View and Display panes:



While working with the Record View and Display panes, you can use these toolbar icons:



Icon	Description
Page Up	Click this icon to go to the next page.
Page Down	Click this icon to go to the previous page.
Zoom In	Click this icon to incrementally zoom in on the page.
Zoom Out	Click this icon to incrementally zoom out on the page.
Fit Width	Click this icon to resize the page display so you see the entire width of the page.
Fit Window	Click this icon to resize the page display so you see the entire page in the window.

Icon	Description
Record Info	Click this icon to display information about the record, including its location on the page, the font used, the text that appears, and so on. See <i>Displaying Information about the Record on page 542</i> for more information.
Properties	Click this icon to see information about the print stream, such as its name and type, the number of records, the longest record, and the page you are viewing.

USING THE FIND OPTION

Use the View, Find option to find specific text within the print stream. For Printstream Analyzer, you can choose from a variety of value encodings. The Auto encoding follows the usual encoding for the print stream you selected.

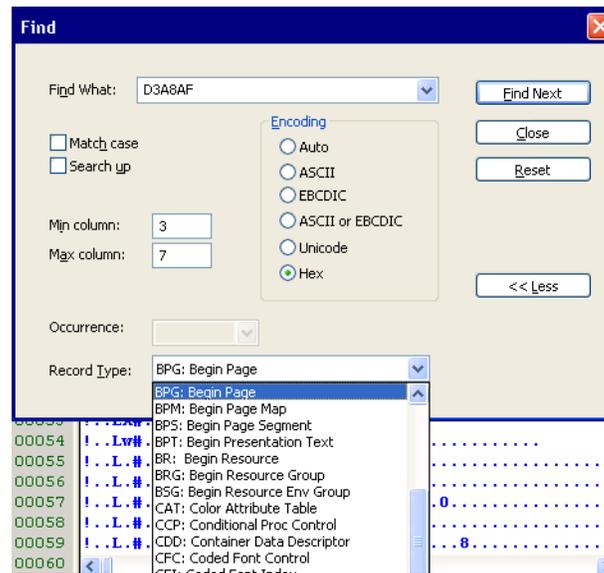
In AFP, text is usually EBCDIC. In Metacode, text is usually ASCII. Comment records in Metacode are EBCDIC so occasionally it is necessary to choose EBCDIC encoding when you are searching in a Metacode print stream.



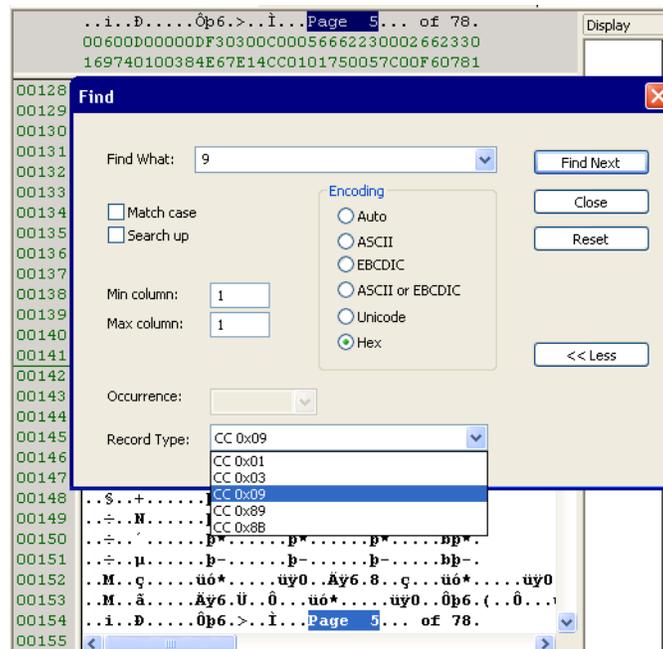
Select the Hex option and enter minimum and maximum column values to find records with a particular identifier type or a carriage control byte. Click More to automatically set the Hex encoding and appropriate minimum and maximum column values for AFP and Metacode.

Here are some examples:

AFP advanced



Metacode advanced



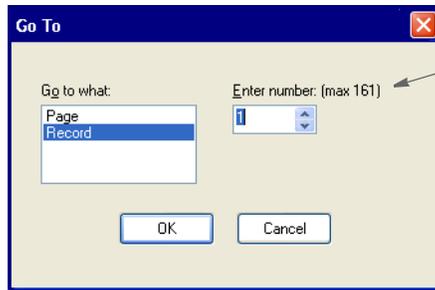
FINDING A SPECIFIC PAGE OR RECORD

Use the View, Go To option to move the selection in the Record View pane to the first record of the page you specify or to a specific record. The Display pane changes to reflect the contents of the page that contains the newly-selected record.

Note When you first open a print stream, you may see comments and other records stored on what is considered page zero. This information does not print and the Display pane is blank for page zero.

To automatically go to the first page, select View, Options, Printstream Analyzer. Then click the Advance to First Page Break option. This tells the system to move to the first page break record when you open a print stream.

You can also right click in the Display pane when on page zero and choose Page Down to move to page one.



Here the system shows you the maximum number of records (or pages) in the print stream.

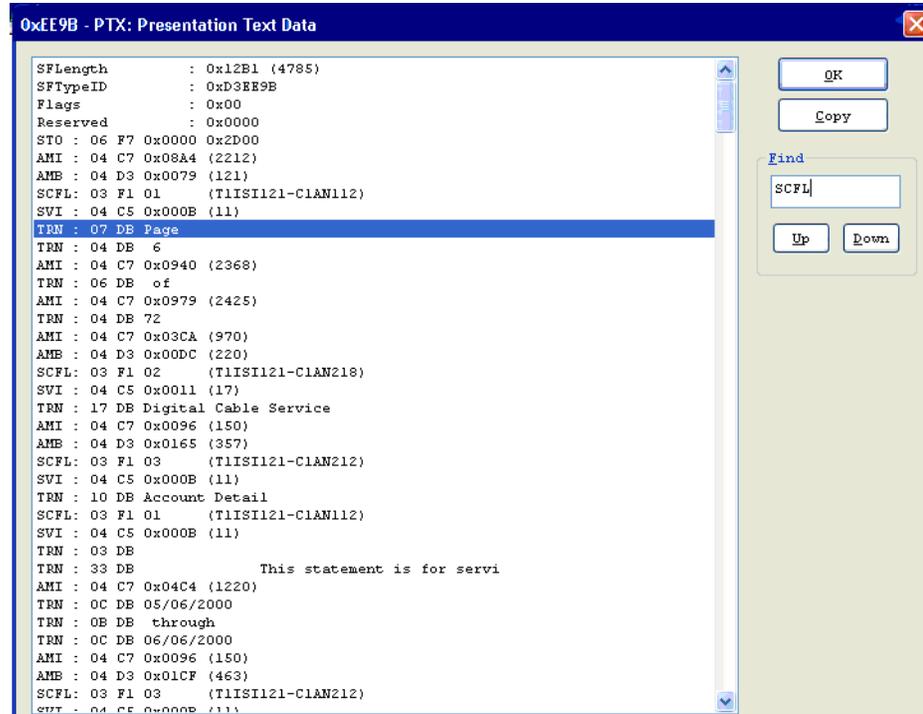
DISPLAYING INFORMATION ABOUT THE RECORD

Use the Record Info option to analyze the commands in the print stream record. The command nearest to the selected position in the current record is selected in the Record Info window when it opens. When you click Ok, the selection position in the current record changes to that of the command selected in the window.

Click Copy to copy the command list to the clipboard so you can paste the command list into other applications.

Use the Up and Down buttons in the Find group to search for commands in the record.

The example below shows that an AFP PTX record is comprised of many absolute move inline, absolute move baseline, and transparent data (text) commands:



PRINTING WHAT IS IN THE PANES

Use the Print option to print the contents of either the Record View or Display panes. When you choose the Print option from the File menu or from the toolbar, the system lets you choose from Display pane or Record View pane so you can indicate which contents to print.



Since print streams can be quite large and printing all pages can take a long time, the Print window lets you print a range of the pages.

Chapter 16

Deploying a Library

You develop resources in Studio within the confines of your workspace and the directories you defined for that workspace. When you decide to move all or part of the resources you have developed into testing or production, you copy or *deploy* the resources to another location.

Studio includes features to make this process quick and easy to do. This chapter discusses these features and includes these topics:

- *Creating or Running a Deployment on page 547*
- *Additional Resources on page 552*
- *INI Settings on page 553*
- *Processing the Deployment on page 554*

OVERVIEW

You can have as many deployments as you need. For instance, you might have an initial test location, a secondary test site, a pre-production site, and a production location. Using the deployment features of Studio, you can easily manage these deployments.

You may also do temporary or occasional deployments for reasons other than testing and production. For instance, you might do a deployment to a secondary location like a backup server or a CD writer. Also, you might deploy to a laptop drive when you are going to take a trip and need to take resources with you. Anytime you intend to make a copy of development resources, it is a deployment.

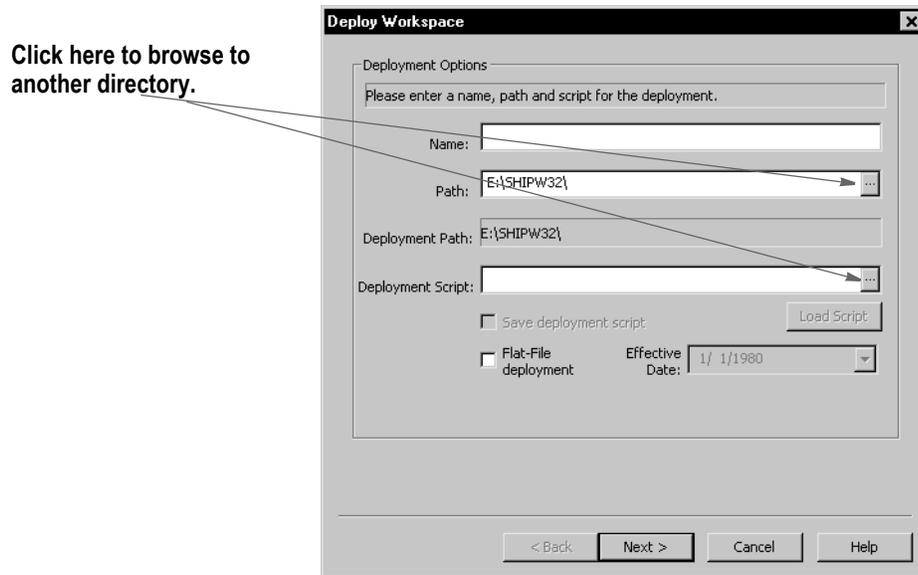
Note A deployment only copies resources that have been checked into the library.

CREATING OR RUNNING A DEPLOYMENT

To run a deployment, double-click on Deploy in the workspace window or choose the Manage, Deploy option from the menu.

The Deployment wizard appears to lead you through the steps necessary to do a deployment of your resources. You can accept the defaults and simply do a total deployment or you can manage the details of the deployment via the wizard.

When you choose Deploy, the Deploy Workspace window appears.



This window includes these fields:

Field	Description
Name	The name for the deployment. This is not the same as naming the workspace. This name is simply a short descriptive identifier for how you want to refer to the deployment you creating or updating.
Path	The location of the deployment. Notice the name of the deployment is combined with the location path to build the appropriate subdirectory name for your deployment.
Deployment Script	The deployment script to use. You can identify a previously saved deployment script to use or assign a name to a new script you want to create.
Save Deployment Script	Indicates whether you want the deployment script to be updated with the choices you make as you continue using the wizard. If you do not check the box, the choices you make are not saved, meaning this is a one-time deployment. If you do save or update the deployment script, the choices you make on the remaining wizard windows are saved and will be available if you decide to deploy the workspace resources again.
Flat File Deployment	Check this box to run a flat-file deployment or leave it unchecked for a library deployment.

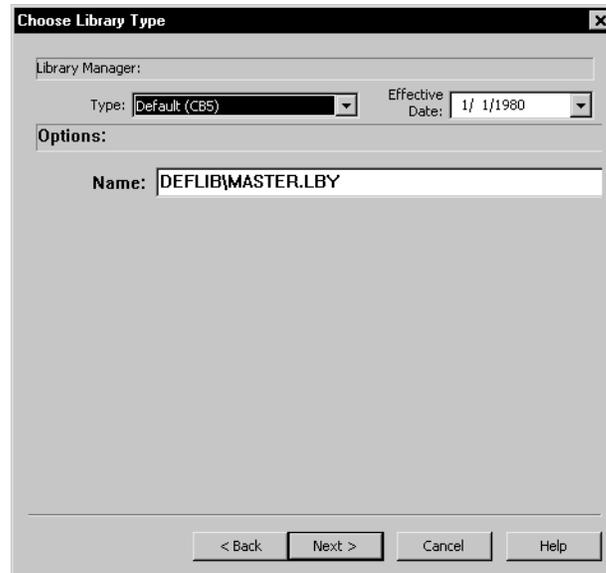
Once you have identified the name and location of the deployment and have either accessed an existing script, entered a new script, and checked whether you want to save the new script, click Next to go to the next step in the wizard. This step differs, depending on whether you chose a flat file or library deployment.

If you chose a	Go to
Library deployment	<i>Creating a Library Deployment on page 549</i>
Flat file deployment	<i>Creating a Flat-File Deployment on page 551</i>

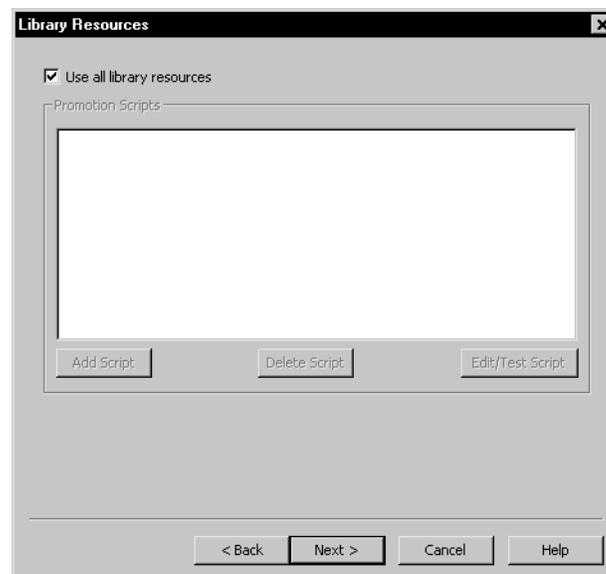
CREATING A LIBRARY DEPLOYMENT

For library deployments, you must choose the library type. On the Choose Library Type window you define the library specifications (type, effective date, and name) you want to use when the resources are deployed.

Depending upon the type of library database you selected, there may be additional questions to answer. For ODBC databases, you must first create the data source names using the Windows Control Panel before Studio can address the library database.



Once you have made your selections, click Next. The Library Resources window appears.



Indicate if you want to use all the library resources. Not checking this option and clicking the Add Script option tells Studio you want to select specific library resources to deploy.

Add Script lets you name one or more promotion scripts you want to use to move resources to the deployment destination. The promotion scripts you choose are shown in a list. You can add or delete them as needed. In addition, you can click Edit Script to edit or create promotion scripts. This takes you to the Library Promotion window. There you can develop your script and test it before including it in your list.

Once you have identified the library resources you want to send to the deployment destination, click Next to move to the Additional Resources window.

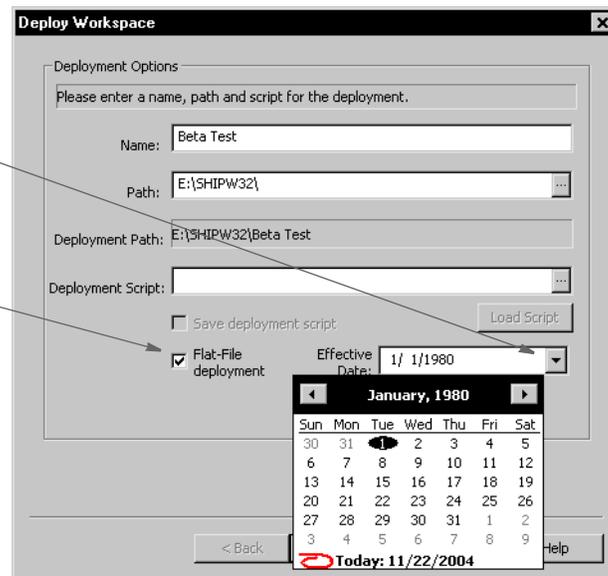
CREATING A FLAT-FILE DEPLOYMENT

In a flat-file deployment, resources are extracted from your development library and copied as external files (not as a library) to the appropriate directory structure in the destination. Studio selects the resources it will move based on the effective date you enter.

To do a flat file deployment, check the Flat-File Deployment option and enter an effective date.

Click here to display the calendar and select an effective date

Check this box to do a flat file deployment



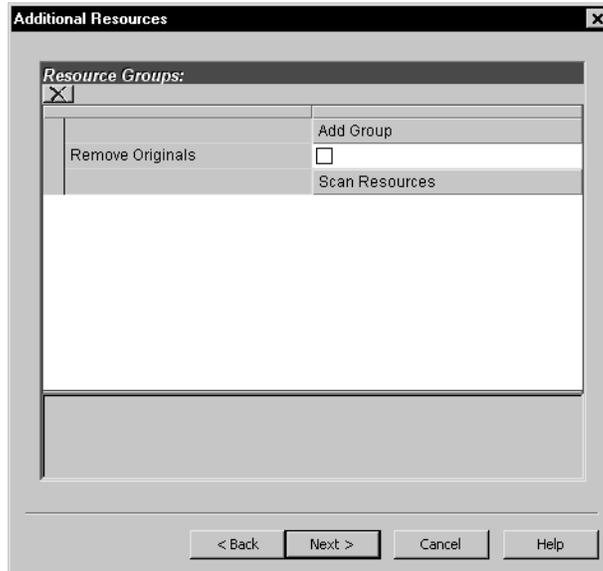
Press Esc to close the calendar.

Once you have finished, click Next to move to the Additional Resources window.

ADDITIONAL RESOURCES

You may have files that are not stored in the library but are required by your runtime setup. You use the Additional Resources window to include these kinds of non-library resources in a deployment.

The Additional Resources window lets you define a list of subdirectories that contain supporting files to copy to the destination deployment.



As a rule, Studio expects to copy a file to the same *relative* location in the destination as the directory name where it resides in the development source. For instance, if there are DFD files in the DEFLIB\ directory of your development workspace, then Studio expects to copy these files into the DEFLIB\ location within the deployment. Likewise, DAT files found in the TABLES\ subdirectory of the workspace would be copied into the TABLES\ subdirectory of the destination location.

This helps ensure that the deployments resemble the development environment and also reduces the number of configuration (INI option) changes you have to make when synchronizing resources.

For each major category shown on the dialog, you can include or exclude those resources from the deployment process by simply checking or unchecking the appropriate box.

If you should want to include only certain files of a given category, you can drill down to the individual file list and check or uncheck the files included in that list.

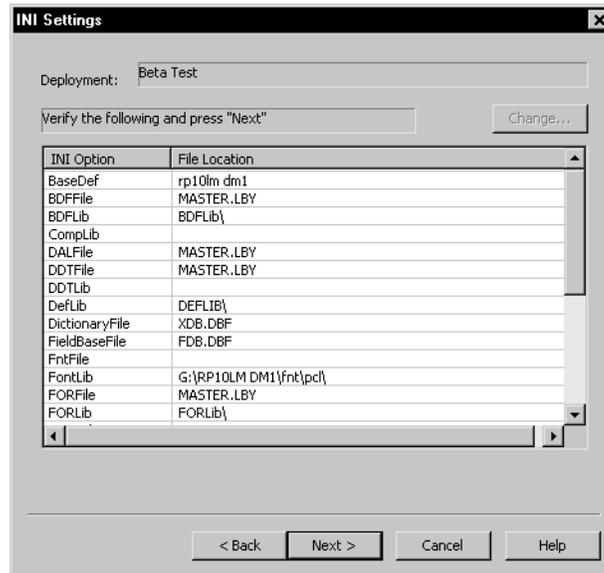
The Other Resources category is where you identify files that are not known to the workspace. Here you can add files including the source location and destination location where the file should be copied.

When you are done identifying the additional resource files you want included in the deployment, click Next to move to the INI Settings window.

INI SETTINGS

You can modify INI options for your deployment. The INI options shown here are for your deployment not the currently open workspace.

By default, if this is your first time to deploy to this location, the workspace settings will be used as the basis for your deployment settings.



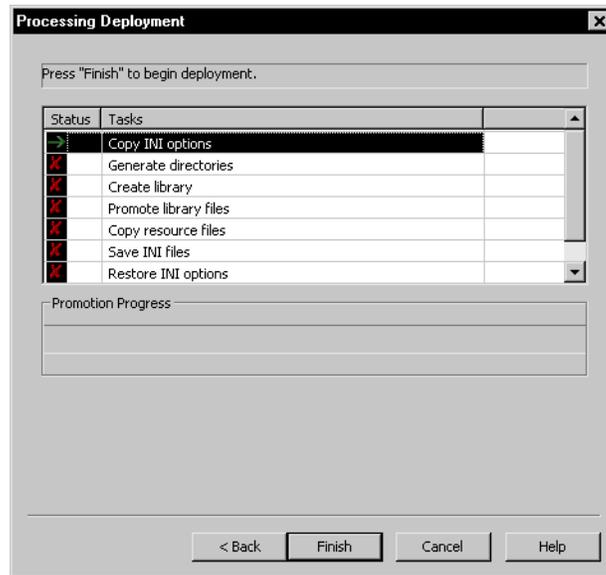
Once you finish, click Next to move to the Processing Deployment window.

PROCESSING THE DEPLOYMENT

The final wizard window shows you the steps that occur to complete the deployment process. When you click Finish, the deployment process begins and as each step is started, a check mark appears next to the task item.

As the deployment proceeds, you may see messages scrolling through the output window. These messages reflect the details of the deployment process to your deployment location.

When all tasks have been completed, the Deployment wizard closes and the success or failure of the deployment is indicated in the Status message.



Appendix A

Using the Documaker Add-in for Microsoft Word

The Documaker Add-In for Microsoft Word lets you use Word to create Documaker forms, sections, and paragraph lists. You can then import your Word Documaker documents into Studio.

With a workspace definition file (WDF), which contains information used in the creation of Documaker documents in Word, you have access to various Studio resources while using the Documaker Add-In for Word. These resources include fields, recipients, fonts, graphics, and triggers. You can use these resources in the documents you create with the Documaker Add-In for Word, which helps promote consistency and efficiency.

This appendix is designed for:

- Project managers implementing the use of the Documaker Add-In for Word
- Personnel responsible for incorporating files imported from the Documaker Add-In into a Studio workspace
- Documaker Add-In users who want to learn what happens when the documents they create in Word are imported into Studio

This appendix discusses these topics...

- *Introduction* on page 556
- *Understanding the Workflow* on page 558
- *Finding Information* on page 560
- *Synchronizing Fonts* on page 561
- *Providing Graphic Files to the Add-In* on page 562
- *Sharing Studio Resource Information with the Add-In* on page 564
- *Importing Documents Created in Word* on page 568
- *Completing Imported Documents* on page 578

INTRODUCTION

Within the insurance carrier organization many departments and individuals are responsible for creating and maintaining document templates. The technical skill levels of these individuals vary greatly as does their knowledge of specific areas of the insurance industry and their company’s processes and procedures.

A common factor within their skill set, however, is the ability to create documents in Microsoft Word. A tool that lets these authors create content in a familiar manner reduces the requirement of additional human resources and lets these individuals operate independently and with greater throughput.

Underwriters, compliance representatives, and others are responsible for ensuring the content of generated documents is accurate. In the current business process, which includes many authoring applications, these individuals first create paragraphs or complete documents in Word. This content is then sent to specialized authors who re-create the document in an advanced authoring tool, such as Studio.

In many cases the sections, variable data, and inclusion rules must be discussed multiple times before the final document template is completed. Additional input from resources such as database administrators is often needed as well.

The Documaker Add-In for Microsoft Word provides non-technical Word users with an easy way to provide Word-based content to the document automation process.

Add-In users can format text using Word to create paragraphs or full documents that can then be imported into Studio without significant editing. When creating documents in Word, the Add-In user has access to variable field names and graphics defined in Studio, can create new, unlinked variables and graphics, and can provide notes and settings for various document attributes such as library tags and recipients.

This figure shows the routing and re-routing of information between Word and Studio users without the functionality provided by the Add-In:

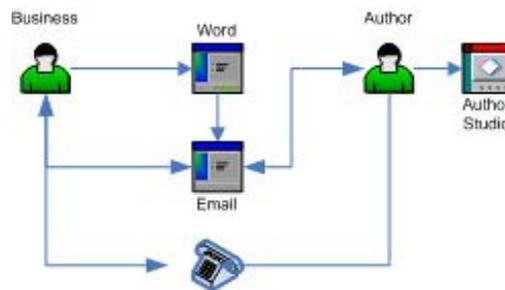


Figure 4: Information flow without the Documaker Add-In for Word

This figure shows the process with the Add-In, where the Word user can select from predefined Studio resources and provide the Studio author additional information directly within the Word Documaker document.

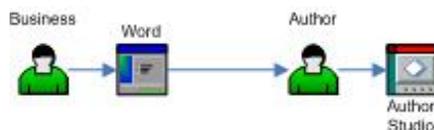


Figure 5: Information flow using the Documaker Add-In for Word

CANDIDATE DOCUMENT TYPES

The Documaker Add-In for Microsoft Word is intended to be used for documents that are predominantly flowing text or distinct blocks of content. The Add-In is intended to provide an alternate authoring environment for the text of documents. It is not intended to be a complete form design tool. Documaker Studio provides excellent, user-friendly form design functionality. The Add-In extends Word functionality to provide its user with access to non-text entities such as pre-defined variables and graphics.

There are several common document types produced within the insurance industry that suggest particular features and accompanying functionality or authoring requirements.

- Bills and statements are typically very structured documents that contain a large number of data variables laid out in a specific structure. These documents tend to include calculations and overflow logic. These documents are generally very form-like and are best authored in Studio.
- Marketing material varies between simple, short text blurbs and highly-structured, full color documents. Simple marketing statements are good candidates for Word authoring, however, the more highly-structured documents should be authored in Studio (or another design tool). The inclusion logic that determines when material is included is, of course, managed within Studio.
- Declarations are often long-flowing documents that contain highly flexible and dynamic tables. These documents may be good candidates for dual-authoring in which the text areas are authored in Word and the dynamic tables are created and managed within Studio. This is the case with many documents in which certain portions that deal with the layout of multiple data items should be authored in Studio.
- Policy documents and contracts generally contain a lot of text and very few areas of heavy data variable inclusion. Like declarations, these documents should be separated into appropriate authoring areas so long paragraphs of flowing text are authored in Word, while dynamic tables based on data variables are authored in Studio.
- Correspondence style documents may be free-flowing or highly structured and as such are candidates for either authoring tool based on the document's structure.

UNDERSTANDING THE WORKFLOW

There are several products involved in creating Documaker documents in Microsoft Word and then importing them into Studio.

Product	Description
Documaker Studio	Lets you create and maintain workspaces and workspace resource files and lets you generate a workspace definition file (WDF), which contains information used in the creation of Documaker documents in Word. Studio is also used to convert Documaker Add-In for Word documents into forms, sections, and paragraph lists.
Documaker Add-In for Word	Lets you create and maintain Documaker documents. You can create these types of Documaker documents using the Add-In: <ul style="list-style-type: none"> • Forms • Sections • Paragraph lists
Microsoft Word 2007	The Add-In works with Microsoft Word 2007 or higher.
Docupresentation Web Service	(Optional) You can set up the Add-in to submit a request to Docupresentation to get a WDF file. This is an alternative to manually generating a WDF file in Studio.

This illustration shows how Studio users and Documaker Add-In for Word users typically interact.

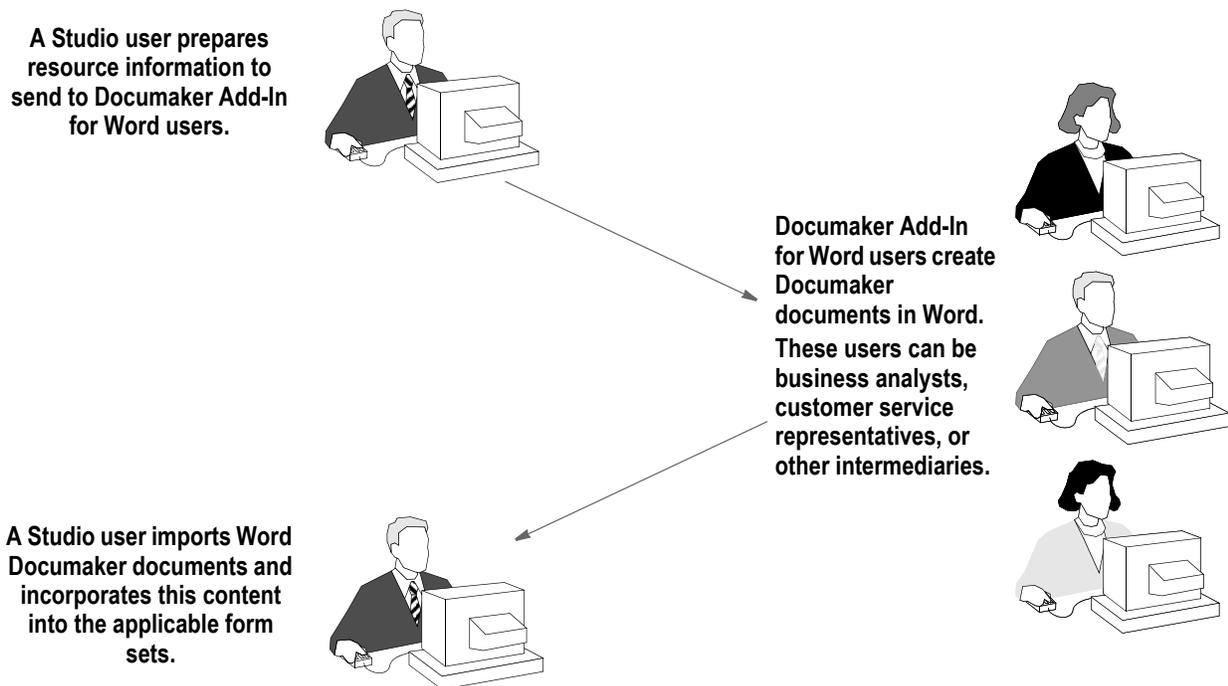


Figure 6: How Studio and the Documaker Add-In for Word Interact

The following illustration provides a more detailed task-oriented view of a Studio and Documaker Add-In for Word workflow. This view does not include the optional use of the Docupresentation web service.

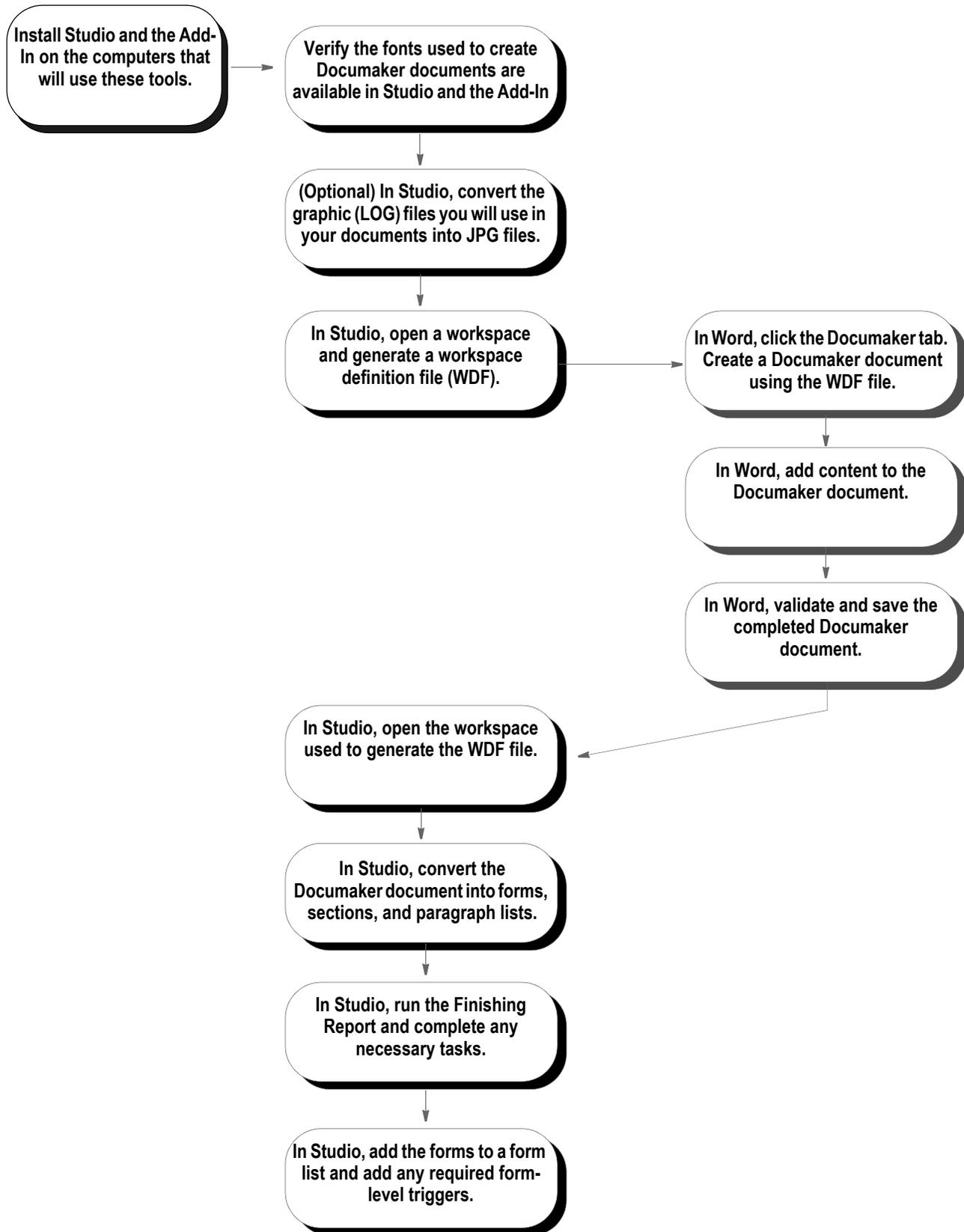


Figure 7: The Studio and Documaker Add-In for Word Workflow

FINDING INFORMATION

Here is an overview of the tasks involved in installing and setting up Studio and the Documaker Add-In for Word and where you can find more information about those tasks:

Task	For more information, see
Install Documaker Studio and the Documaker Add-In for Word	Documaker Installation Guide
(Optional) Set up the Docupresentation Web Service to provide resource information to the Add-In	Introduction to Enterprise Web Processing Services
(Optional) Make sure the fonts available to Studio are also available to Word	<i>Synchronizing Fonts</i> on page 561
(Optional) Convert Studio graphic (LOG) files into JPG files	<i>Providing Graphic Files to the Add-In</i> on page 562
(Optional) Generate a WDF file to use with the Add-in	<i>Generating a Workspace Definition File</i> on page 565
Start Word 2007 and click the Documaker ribbon tab	Documaker Add-in for Microsoft Word User Guide
Configure available WDF files using the Workspaces item in the Tools group	
Create Documaker documents in Word	
Import Documaker documents into Studio	<i>Importing Documents Created in Word</i> on page 568
Finish incorporating the documents into the library	<i>Completing Imported Documents</i> on page 578

SYNCHRONIZING FONTS

When you import Documaker documents created using the Add-In, Studio uses the font cross-reference (FXR) file specified in your application definition file (BDF). Since document layout heavily depends on the choice of fonts, Studio tries to match the fonts in the source document to those defined in the FXR, but substitution can occur and may affect the layout. The best way to avoid layout changes is to make sure all fonts used by the Add-In are referenced in your FXR file.

Note The FXR file contains information about fonts, such as the font family name, typeface, and point size. You specify the FXR you will use for a workspace in the BDF file. When you create a workspace definition file (WDF) in Studio for use by the Add-In, Studio looks in the FXR file specified in the BDF file to get font information. This includes the font family name and the font point size. For more information, see *Generating a Workspace Definition File* on page 565. For more information on BDF files, see *Working with Application Definition Files* on page 89.

To ensure font consistency across all documents, review the font family names and make sure all fonts you will use in document creation are available to both Studio and the Add-In. If there is a discrepancy, you can resolve it by using one or both of these options:

- Import the TrueType fonts included with the Documaker installation into the Font Palette on the computers that run the Add-In.
- Import the Windows TrueType fonts available to the Add-In into your FXR file. For more information, see *Importing Fonts* on page 513.

Note There are legal issues to consider when using fonts from other vendors. Font vendors generally copyright the fonts they create. Make sure your license lets you use the font on more than one computer at a time. Also make sure your license lets you copy converted fonts to other platforms running on the same computer.

In addition, the Add-In provides a validation tool you can use to check the Documaker documents you create. As part of the validation process, it compares the font names defined in the Word Documaker document with the font family names defined in the WDF file imported from Studio. For more information, see the [Documaker Add-in for Microsoft Word User Guide](#).

PROVIDING GRAPHIC FILES TO THE ADD-IN

When you insert a graphic into a Documaker Word document, the Add-In shows you a list of graphic names contained in the WDF file. Graphic file names and sizes are taken from the LOG files stored in the workspace resource library when the WDF file is generated.

If you have JPG format versions of these LOG files, you can insert the actual graphic directly into the Word Documaker document.

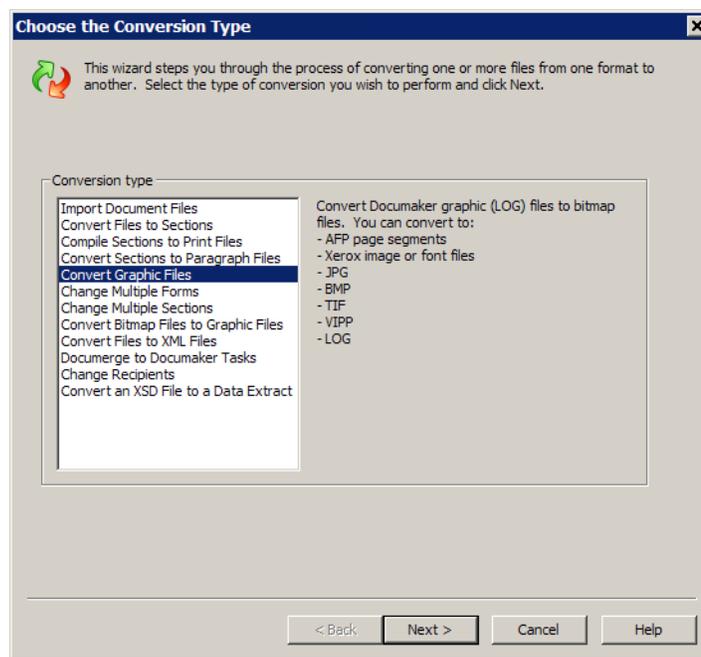
If you do not have JPG format versions of these LOG files, the Add-In inserts a default graphic placeholder that is the same size as the actual graphic. Here is an example of the default graphic:



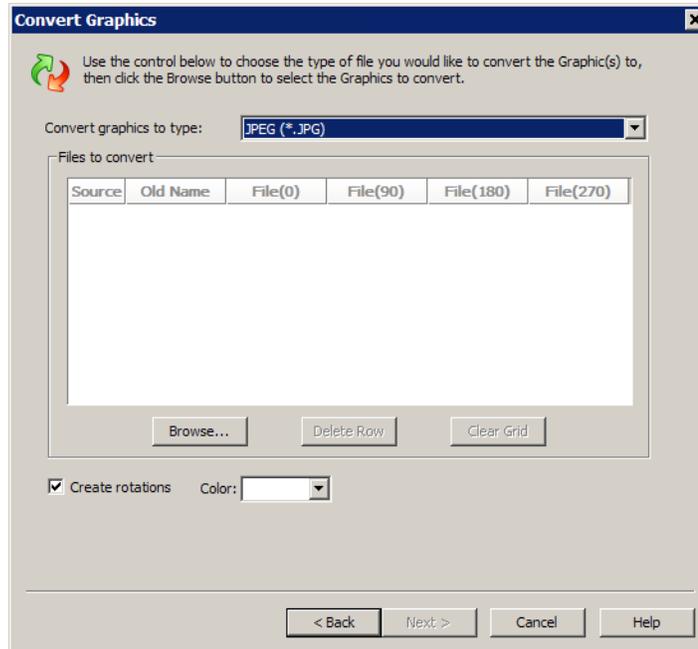
Note If your implementation uses JPG files *instead* of LOG files or if you have no need to see the actual graphic when working in Word, you do not need to convert LOG files.

To create JPG format versions of your LOG files, follow these steps:

1. In Studio, choose the Manage, Tools, Conversion option. The Choose the Conversion Type window appears. Select Convert Graphic Files and click Next.



The Convert Graphics window appears.



2. Click Browse to select the graphic files (LOG) you want to convert. Then click Next. The Convert Files and Finish window appears.

Note In certain situations, such as when using a signature or company logo, you may prefer to use a placeholder rather than an actual graphic in the Add-In. Therefore, do not generate a JPG file for those graphics. This helps safeguard those files and prevents unauthorized redistribution.

3. Click the ellipsis button to specify the location where you want Studio to place the converted JPG files.

After you create the JPG files, enter the location of the graphic files folder into the Add-In's options. For more information on this and inserting graphic files, see the [Documaker Add-in for Microsoft Word User Guide](#).

SHARING STUDIO RESOURCE INFORMATION WITH THE ADD-IN

Using Studio resource information in the Add-In makes the document creation process easier and gives you better results when you convert those documents into Studio forms, sections, and paragraph lists.

You can provide Word with these types of Studio resource information:

- Library class values
- Fields
- Fonts
- Graphics
- Form metadata
- Library project values
- Recipients
- Triggers

The Add-In uses this information to provide lists of options which are available when you are creating forms, sections, or paragraph lists in Word. This lets you insert most of the Studio resources directly into a document at creation time, which makes the process quicker and provides more consistent results.

There are two ways to provide this information to the Add-In. You can...

- Provide Add-In users with access to Studio information via a Docupresentation web service. For more information on connecting the Add-In to Docupresentation web services, see the [Documaker Add-in for Microsoft Word User Guide](#).
- Generate a workspace definition file (WDF) in Studio and access it in the Add-In. For more information, see *Generating a Workspace Definition File* on page 565.

Note Although highly recommended, you do not have to generate a workspace definition file (WDF) to use the Add-In.

GENERATING A WORKSPACE DEFINITION FILE

Follow these steps to generate a workspace definition file (WDF) in Studio:

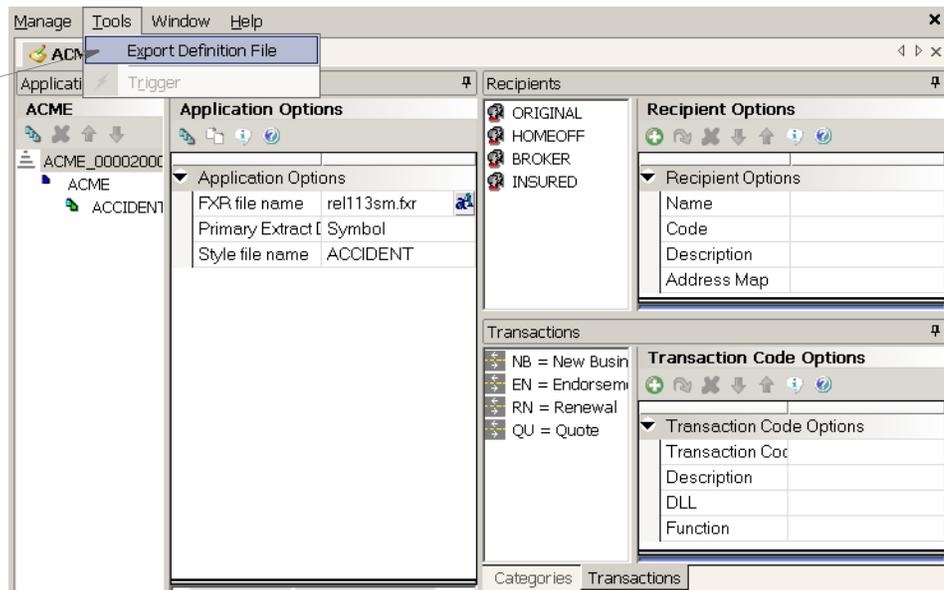
1. Open a workspace and check out the BDF file for that workspace.



Note The ability to open BDF files requires the appropriate security rights. Check with your system administrator for more information.

2. Select the Export Definition File option from the Tools menu.

Choose the Tools, Export Definition File option.



3. Browse to the location where you want to save the file. Accept the default file name (WDF.XML) or enter the name you want to assign to the file and click Save.

Studio tells you the file was successfully generated in the Output area:



Understanding the Workspace Definition File

The workspace definition file (WDF) contains information specific to a workspace and is a snapshot of several types of resources in a workspace at the time the file is generated. The Add-In uses the information in this file to present its users with choices when they are creating Documaker Documents in Word.

The WDF file contains these types of information:

Type	Description
Definition name and time stamp	Includes the name of the workspace definition file (WDF) and a time stamp which tells you the date and time the file was generated. Here is an example of a time stamp: <code><Definition Name="ExampleCo" TimeStamp="2010-03-31 18:23:55"></code>
Common fields	Includes field names, field types, field prompts (if applicable), and field lengths. This information comes from the Common Fields Dictionary in the workspace (FDB.DBF). There are three nodes (field name, type, and length) in the WDF file for each entry in the FDB.DBF file. If a prompt appears for a field in the FDB, it also appears as a node for that field in the WDF file.
DAL triggers	Includes DAL routine (trigger) names. Triggers define the criteria that must exist for content to be included in the form set. This information comes from the trigger files stored in the workspace's library SETRCPTB.DAL file.
Recipients	Includes recipient names and descriptions. This information comes from the recipient information defined in the workspace's BDF file.
Fonts	Includes the font ID, font family, and point sizes. This information comes from the font cross-reference (FXR) file assigned to the workspace in the workspace's BDF file.
Project list	Includes project names and descriptions. This information comes from the INI files of the workspace.
Class list	Includes class names and descriptions. This information comes from the INI files of the workspace.
Graphics	Includes the name of each graphic, plus its height and width. This information comes from the graphic files (LOG) stored in the library of the workspace. It includes the name of the graphic and the height and width of the graphic. The height and width values are specified in inches.
Metadata	Includes the Metadata name. This information comes from form files (FOR) stored in the library of the workspace.

Note This file is in XML format.

Here is an example of a workspace definition file (WDF).

```
<?xml version="1.0" encoding="UTF-8" ?>
  <Definition Name="ExampleCo" TimeStamp="2010-03-31 18:23:55">
    <CommonFields>
      <Field Name="ADDRESS LINE1">
        <Prompt>Please enter the Number and Street of the Address</Prompt>
        <Type>Alphanumeric</Type>
        <Length>30</Length>
      </Field>
    </CommonFields>
    ...
    <DALTriggers>
      <Routine Name="ALWAYS" />
      <Routine Name="WELCOME_LETTER" />
    ...
  </DALTriggers>
  <Recipients>
    <Recipient Name="AGENT">
      <Description>Agent Copy</Description>
    </Recipient>
  ...
  </Recipients>
  <Fonts>
    <Font ID="00911">
      <Family>Pdf417</Family>
      <Size>2.16</Size>
    </Font>
  ...
  <Font ID="19230">
    <Family>Arial Black</Family>
    <Size>30</Size>
  </Font>
  </Fonts>
  <LibraryProjectList>
    <Project Name="P001">
      <Description>Project 001</Description>
    </Project>
  ...
  </LibraryProjectList>
  <LibraryClassList>
    <Class Name="GA">
      <Description>Georgia resource</Description>
    </Class>
  ...
  </LibraryClassList>
  <Graphics>
    <Graphic Name="ExampleCoGrey">
      <Height>0.88</Height>
      <Width>1.51</Width>
    </Graphic>
  ...
  </Graphics>
  <Metadata />
</Definition>
```

Figure 8: Example Workspace Definition File

This example WDF file includes information about fields, DAL triggers, recipients, fonts, projects, classes, and graphics. For more information on how to use a WDF file in Microsoft Word, see the [Documaker Add-in for Microsoft Word User Guide](#).

IMPORTING DOCUMENTS CREATED IN WORD

For best results, import Add-In Documaker documents into Studio resources in the same workspace used to generate the workspace definition file (WDF). Otherwise, resource compatibility issues, such as incorrect fonts or graphics, are likely to occur.

When you import Documaker documents, Studio converts those documents into forms, sections, graphics, and paragraph lists. You should perform several trial Word to Studio conversions without checking the imported document resources into the library. This lets you make sure the document composition process you are using in Word provides the results you want in Studio. See also *Understanding the Import Process* on page 575 for more information.

Depending on the Studio conversion, you may need to refine the way you are composing documents in Word. If a resource is not checked into the library during the trial conversion process, you then have the choice of checking it into the library if it is what you want or deleting it if you want to do more work in Word before importing it again.

Note After documents are imported, you decide whether the imported resources should be checked into the library or stored on disk.

If you elect to store the files on disk, you specify the disk location for the file or files on the last page of the conversion wizard. Any file that is not checked into the library can later be deleted in Windows just like any other file.

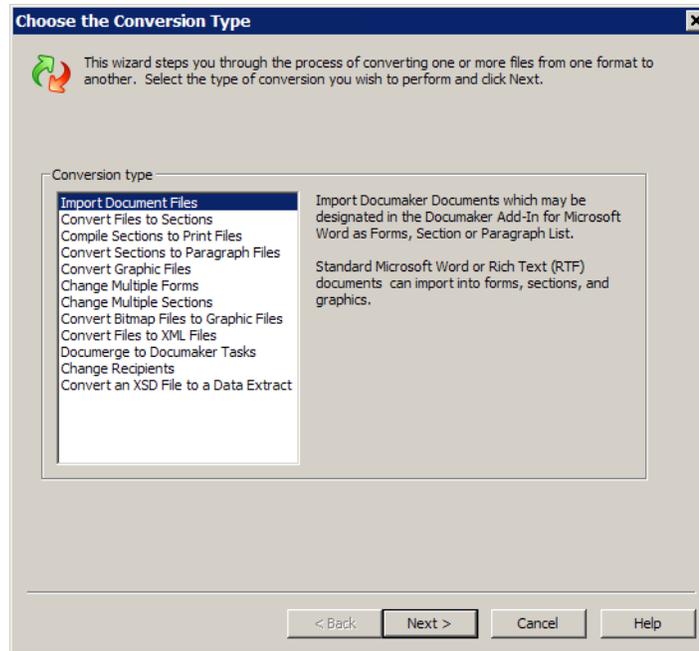
When you are confident that your documents from Word are converting correctly into Studio resources, you may want to continue to use a small number of documents in a batch conversion until you get a good feel for the length of the conversion process. The length of the conversion process is affected by both document complexity and the number of documents.

Microsoft Word 2007 must be installed on the computer where the Word to Studio conversion takes place. Be sure to close all of the documents you are importing before you start the conversion process. Also, avoid using Word on the computer during the conversion as this may conflict with the Word automation used by the conversion.

To import Documaker documents created in the Add-In, follow these steps:

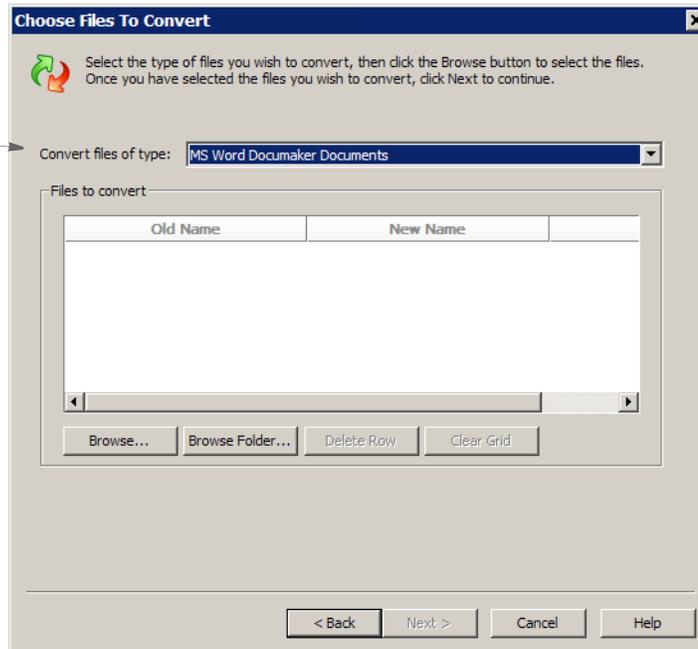
1. Open your Studio workspace. Then select the Manage, Tools, Conversion option. The Choose the Conversion Type window appears.

Note If your Studio theme is set to Office 2007, click the Conversion icon in the Tools group.



2. Select Import Document File option as the conversion type. Click Next. The Choose Files to Convert window appears.

Select the MS Word Documaker Documents option here.



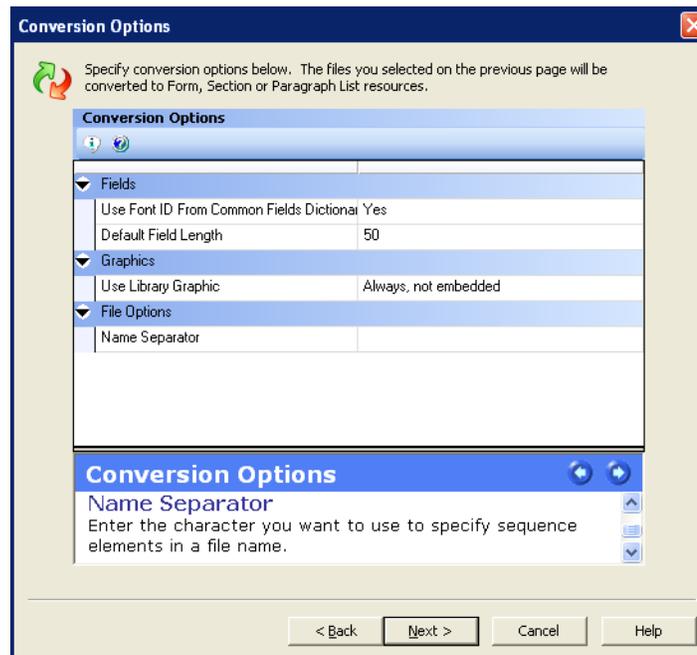
3. Select the MS Word Documaker Documents option. Then click Browse to locate the Word Documaker documents you want to import or use the Browse Folder button to select the folder that contains the files to be imported. You can also select Search Sub Folders to find files in sub folders. The Browse Folder button changes to a Stop button that lets you end a search.

Depending on the size and complexity of the document, it may take several seconds to load a document. After the document is loaded, the Old Name, New Name, and Type fields appear.

Field	Description
Old Name	This is the name of the .DOCX or .DOC file you are selecting to import. You cannot change this name.
New Name	This is the name of the document specified by the user in Word in the Documaker Name field on the Document Properties, General tab. If needed, you can change the default. This name is used to identify subsequent imports of upgraded versions of the document so make sure you inform those using the Add-In user of the proper naming conventions.
Type	This is the type of Documaker document created in Word. Valid document types are: form, section, and paragraph list. You cannot change this field. You specify the document type in the Type field on the Document Properties, General tab in Documaker Add-In for Word when you create the document.

Note If you are importing a form that contains sections, the sections are not listed separately as files to import. If you are importing a paragraph list that contains paragraphs, the paragraphs are not listed separately as files to import.

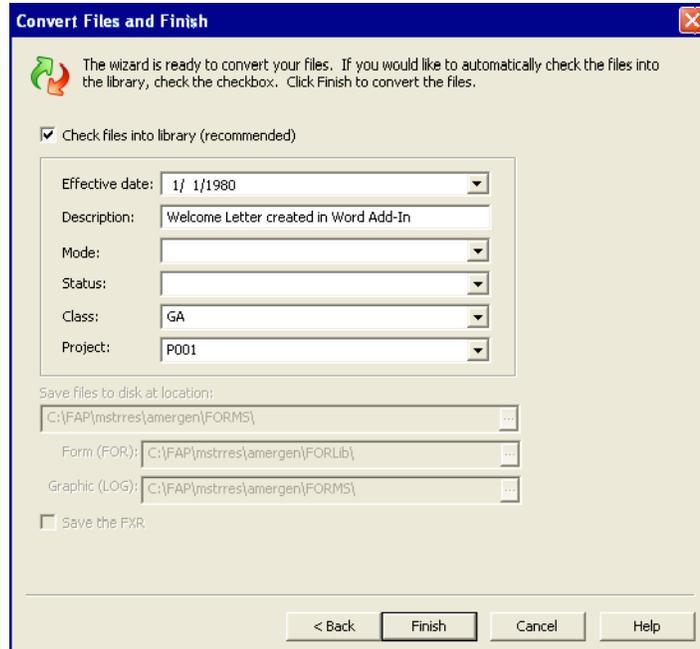
Click Next to continue. The Conversion Options window appears.



4. Use the fields on the Conversion Options window to specify information about field and graphic conversions.

Option	Description										
Use Font ID from Commons Fields Dictionary	<p>Yes - The conversion checks the fields inserted on the Documaker document against fields in the Common Field Dictionary. The font ID of the matching field in the Common Field Dictionary is used for the field in the imported document.</p> <p>No - The font ID of the field in the Documaker document is used for the field in the imported document.</p>										
Default Field Length	<p>Numeric Value - The conversion checks fields contained on the Documaker document against fields in the Common Field Dictionary. The length of the matching field in the Common Field Dictionary is used for the field in the imported document. If there is no matching record in the Common Field Dictionary, the value specified in the default field length is used.</p>										
Use Library Graphic	<p>Always, not embedded - The conversion checks graphics contained on the Documaker document against graphics in the workspace's resource library. If a matching name is found, this option directs the system to use the Studio library graphic including its size. Any placeholder graphics will be replaced with the graphic from the library. The graphic is not embedded in the imported file.</p> <p>If same size, not embedded - The conversion checks graphics contained on the Documaker document against graphics in the workspace's resource library. This option tells the conversion to use the Studio library graphic if it is the same size as the one in the Documaker document, but not to embed it. If the graphics are not the same size, use the graphic inserted in the Documaker Word document but do not embed it.</p> <p>Always, embedded - The conversion checks graphics contained on the Documaker document against graphics in the workspace's resource library. If a matching name is found, this option directs the system to use the Studio library graphic including its size. Any placeholder graphics will be replaced with the graphic from the library. The graphic is embedded in the imported file.</p>										
Name Separator	<p>Enter the character you want Studio to insert between elements of the file name. The separator can be nothing, a blank, an underscore, or any other valid Windows file name character.</p> <p>For example, if the name of the input document is Declaration and you choose to create separate sections you would get these results for page 1:</p> <table border="0" data-bbox="760 1207 1218 1375"> <thead> <tr> <th data-bbox="760 1207 941 1239">Separator</th> <th data-bbox="941 1207 1218 1239">Result</th> </tr> </thead> <tbody> <tr> <td data-bbox="760 1270 941 1302"><i>(nothing)</i></td> <td data-bbox="941 1270 1218 1302">Declaration1.FAP</td> </tr> <tr> <td data-bbox="760 1302 941 1333">Blank</td> <td data-bbox="941 1302 1218 1333">Declaration1 .FAP</td> </tr> <tr> <td data-bbox="760 1333 941 1365">_</td> <td data-bbox="941 1333 1218 1365">Declaration_1.FAP</td> </tr> <tr> <td data-bbox="760 1365 941 1396">#</td> <td data-bbox="941 1365 1218 1396">Declaration#1.FAP</td> </tr> </tbody> </table>	Separator	Result	<i>(nothing)</i>	Declaration1.FAP	Blank	Declaration1 .FAP	_	Declaration_1.FAP	#	Declaration#1.FAP
Separator	Result										
<i>(nothing)</i>	Declaration1.FAP										
Blank	Declaration1 .FAP										
_	Declaration_1.FAP										
#	Declaration#1.FAP										

To continue, click Next. The Convert Files and Finish window appears.



5. Use the following fields to further define the conversion, then click Finish.

Option	Description
Check Files into Library	(Recommended) Check this field if you want Studio to check into the library imported Documaker files. Leave this field unchecked if you want Studio to save imported Documaker files to the location you specify in the Save files to Disk at Location field:
Effective Date	(Optional) Accept the default library effective date or change the date, if necessary. The date you enter in this field is only used if no effective date has been entered for a document in Documaker Add-In for Word. Whether the effective date being used comes from this field or the field in the Documaker Word document, keep in mind that: The effective date used on a imported form applies to any section contained in that form if you check the files into the library. The effective date used on a paragraph list applies to any paragraph contained in that paragraph list file if you check the files into the library. Effective Date is an optional field on the Document Properties, General tab in the Documaker Add-In for Word.
Description	(Optional) If a Documaker document has no description, what you enter in this field gets used as the library description for the imported resource. Description is an optional field on the Document Properties, General tab in Documaker Add-In for Word.
Mode	(Optional) Enter a mode if you are using this library feature and you want it to apply to all imported resources. Mode may not be used if Projects (see System Settings) is activated in this workspace. You cannot set up modes in the Documaker Add-In for Word.

Option	Description
Status	(Optional) Enter a status if you are using this library feature and you want it to apply to all imported resources. Status may not be used if Projects (see System Settings) is activated in this workspace. You cannot set up statuses in the Documaker Add-In for Word
Class	If a Documaker document has no value for Class, then what you enter in this field is used as the library class value. If the Projects option (see System Settings) is turned on for this workspace, you should select a class or blank from the list. Class is an optional field on the Document Properties, Advanced tab in the Documaker Add-In for Word.
Project	If a Documaker document has no value for Project, then what you enter in this field gets used as the library project value. If the Projects option (see System Settings) is turned on for this workspace, you should select a project or blank from the list. Project is an optional field on the Document Properties, Advanced tab in the Documaker Add-In for Word.
Save files to disk at location:	This option is available only when Check File into Library is not selected. Files from a conversion that could potentially be saved to this location are FAP files (sections), PAR files (paragraphs), and LOG files (graphics).
Form (FOR)	This option is available only when Check File into Library is not selected. Files from a conversion that could potentially be saved to this location are FOR files (forms) and PSL files (paragraph lists).
Graphic (LOG)	This option is available only when Check File into Library is not selected. Files from a conversion that could potentially be saved to this location are LOG files (graphics). Use this option to specify the pathing location for LOG files when LOG files do not reside in the same folder as FAP files (sections) and PAR files (paragraphs).
Save the FXR	This option is activated only when a normalized AFP or Metacode file, where the fonts are embedded in a print stream, is being converted.

Note Files imported and saved to disk do not retain the values entered into the following fields: Effective Date, Description, Mode, Status, Class, or Project. If you check one of the imported resources into the library at a later time, you will need to specify any applicable information again.

Studio’s output area provides feedback during the conversion. Here is an example.

```

[08:50:49PM] --- Begin Conversion ---
[08:52:57PM] Starting Microsoft Word...
[11:49:06PM]
[11:49:06PM] Started Conversion at 11:49:06 PM
[11:49:06PM]
[11:49:06PM] Conversion, type: MS Word Documaker Documents
[11:49:06PM] C:\Documents and Settings\word add -in documents\Welcome Letter - General.docx
[11:49:06PM] Document type: Form
[11:49:06PM] Recipient <Loss Payee> is a placeholder : Loss Payee
[11:49:06PM] Staple: <Staple with the form set>
[11:49:07PM] Processing Field <INSURED NAME>
[11:49:07PM] Processing Field <INSURED ADDRESS1>
[11:49:07PM] Processing Field <INSURED ADDRESS2>
[11:49:07PM] Processing Field <INSURED CITY>
[11:49:07PM] Processing Field <INSURED STATE>
[11:49:07PM] Processing Field <INSURED ZIP>
[11:49:07PM] Processing Field <POLICY NBR>
[11:49:07PM] Processing Field <INSURED NAME>
[11:49:07PM] Processing Field <INSURED NAME2>
[11:49:07PM] Processing Field <PRODUCT>
[11:49:07PM] Field <PRODUCT> is a placeholder : Alphanumeric (30) Local
[11:49:07PM] Warning: Field <PRODUCT> was not found in Common Fields dictionary
[11:49:08PM] Processing Field <AGENT TYPE>
[11:49:08PM] Processing Field <AGENT NAME>
[11:49:08PM] Processing Field <AGENT PHONE>
[11:49:08PM] Processing Graphic <SIG>
[11:49:25PM] Conversion, checked into library: <Welcome Letter - General> <FOR>.
[11:49:25PM] Finished Conversion at 11:49:25 PM
[11:49:25PM] Elapsed time 00:00:19
[11:49:25PM]
[11:49:25PM] Number of files converted: 1
[11:49:25PM] Number of files in error: 0
[11:49:25PM]
[11:49:25PM] --- Conversion Complete ---

```

Figure 9: Example Output Area Messages when Importing a Documaker Document

The messages in this output area example identify three specific items that need attention (emphasis added):

- A recipient named Loss Payee has been identified as a placeholder.
- A staple message is detected.
- A field named PRODUCT has been identified as a placeholder.

Detection of placeholder information (recipient Loss Payee and field PRODUCT, in this example) is seen in the output area. This placeholder information also appears on the Finishing Report and in the Task List for the Form Welcome Letter – General. There are certain types of information about the imported documents, that only appear in the output area during the conversion. This includes detecting if there is...

- Data in the Staple field — This is entered in the Add-In on the Document Properties, Advanced tab and is only applicable to form documents
- Data in the Paper Stock field — This is entered in the Add-In on the Document Properties, Advanced tab and only applies to form documents

When information is written to the output area concerning the Staple field or the Paper Stock field, someone on the Studio side of the implementation needs to make the necessary adjustments. Staple and paper stock options appear in the list of form options in the Forms manager.

Note Studio clears the output area when you exit or right-click and select the Clear Contents option. To retain this information, right-click in the output area and choose from the options to print the contents, save the contents to a file, or email the contents to yourself or someone else.

Any undefined element or resource identified during the conversion process is also referenced in the output area. This could be an undefined recipient, an undefined field, an undefined trigger or an undefined graphic. Undefined elements or resources are also identified in task lists and on Finishing Reports. (See *Completing Imported Documents* on page 578 for more information.)

Note Undefined resources can also originate from within Documaker Studio.

Comments entered via the Add-In are converted to Task Comments during the conversion process. Comments do not appear in the output area during the conversion process. They are shown, however, in Studio on task lists and on Finishing Reports. (See *Completing Imported Documents* on page 578 for more information.)

Note Task comments can also originate from within Documaker Studio.

UNDERSTANDING THE IMPORT PROCESS

When you import a Word Documaker document into Studio, some formatting, font, and other changes can occur during the import process.

This item in the Add-In	Is converted this way
Fonts	When importing Word Documaker document files, Studio compares the fonts used in the source document to the fonts described in the FXR file. If there is no exact match, Studio calculates a score based on several font attributes. This score reflects how well the attributes match. It then selects the font with the highest score. For more information, see <i>Understanding Font Mapping</i> on page 419.
Underlines	Underlines are converted.
Superscript	Superscripted text is converted to superscripted text, provided there is a corresponding font in the appropriate point size.
Subscript	Subscripted text is converted to subscripted text, provided there is a corresponding font in the appropriate point size.
Strikethrough	Stricken text is converted to stricken text.
All caps	Capital letters are converted, provided there is a corresponding font in the appropriate font size.

This item in the Add-In	Is converted this way
Small caps	Converted as upper and lowercase letters, provided there is a corresponding font in the appropriate font size.
Hidden text	Not converted.
Hyphenation	Hyphenation settings are converted.
Drop caps	If within the margin, the system retains the font attributes as long as there is a corresponding font available. The text that follows the drop cap is placed in a new paragraph. If outside of the margin, the drop cap is not converted. The text that follows the drop cap is converted into a paragraph.
Tables	Each cell is converted into a box and the cell borders become the box borders. The system places the text in the cell into a text area.
Footers	Converted as a text area in the same position on the page.
Headers	Converted as a text area in the same position on the page.
Pictures and drawings	You can import polyline or vector drawings. You can also convert bitmaps in the original document into external Documaker graphic (.LOG) references.
Margins	Margins are converted. The system tries to compensate for borders defined outside paragraph margins.
Line spacing	Converted, but may change if the system substitutes fonts.
Columns	Converted as columns within a text area.
Footnotes	Not supported.
Endnotes	Not supported.
Wrap around formatting	Not supported.
Text boxes	Not supported.
Objects anchored to a paragraph	Positioned correctly.
Captions	Converted to a text areas.
Tables of contents	Converted as a text area in the same position on the page.
Merged table cells	Not supported.
Nested columns (columns within a column such as a table embedded in a column)	Not supported.
Z-ordering or the layering of objects one over the other	Not supported.
Charscalex and Charscaley control words	Not supported.
Section elements	Converted to sections (FAP files) in Studio.

This item in the Add-In Is converted this way

Paragraph elements	Converted to paragraphs (PAR files) in Studio.
--------------------	--

If you experience these types of issues during import process, you must decide whether to make changes in the Word document using the Add-In or modify the imported file in Studio.

If you plan to maintain this document in Then

Studio	Make the adjustments in Studio.
The Add-In	Make the adjustments in the Add-In and re-import the result.

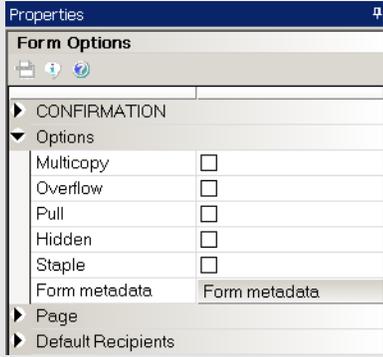
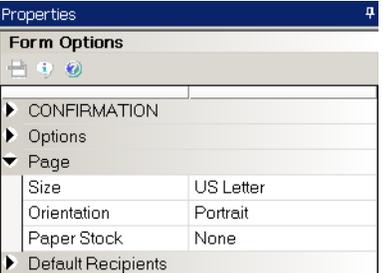
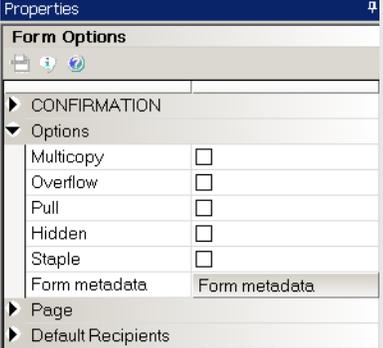
COMPLETING IMPORTED DOCUMENTS

See the table below for information about finishing activities that may appear in the output area during document conversion, on a Task List and the Finishing Report, or when viewing a resource in Studio. This table provides the necessary action to take for each type of activity.

Note For information on running a Finishing Report, see *Printing a Finishing Report* on page 461. For information on using a task list, see *Using the Task List* on page 249.

Add-In	Studio	Your response
Forms	The Documaker document is imported. Any content in the form that is not inside a named section becomes part of the form as a non-named, embedded section. You will see this when you open the imported form in Studio.	No action required, but you can... <ul style="list-style-type: none"> • Unembed the section in Form manager • Replace it with one from the library or disk • Save the section • Unembed the section only
A field is in the Documaker document, but not the Common Fields Dictionary in the workspace	The Documaker document is imported. The messages in the output area tell you that the field does not exist in the Common Field Dictionary. The Task List on the section (and a form that uses the section) or paragraph (and a paragraph list that contains the paragraph) notes the field as Undefined, meaning it does not exist in the Common Fields Dictionary. The Finishing Report notes that the field does not exist in the Common Fields Dictionary.	No action required, but you can choose from these options: <ul style="list-style-type: none"> • Add the field to the Common Fields Dictionary. Check for comments originating in Word concerning field type information as Word only uses alphanumeric field types. <p>Note Regenerate the Workspace Definition file if you want the additional field to be available in the Add-In.</p> <p>Or,</p> <ul style="list-style-type: none"> • Replace the field on the form, section, or paragraph with one from the Common Fields Dictionary.
Bar code and multiline text fields are in the Documaker document	Studio does not allow bar code or multiline text fields to be embedded in text areas or paragraphs. These fields are converted to stand-alone fields on the section. If Studio finds a multiline or bar code field on a paragraph list or in a paragraph file, those fields are not created and this message appears in the output area: Warning: Multiline text and bar code fields are not supported in paragraph elements. Note: The Add-In does not let you insert multiline text fields into a paragraph list or paragraph file.	No action required.
Graphics are in the Documaker document but not in the workspaces library.	The graphic is included in the imported document as an embedded LOG file.	No action required, but if this LOG file would be useful in other Studio documents, unembed the graphic by clicking the Embedded field in the Attributes area of the Graphic Options tab. This starts the Embedded Graphic wizard which lets you unembed the graphic and check it into the library.

Add-In	Studio	Your response
Triggers are in the Documaker document but not in the workspaces library.	<p>The undefined trigger is listed in the output area during the conversion process.</p> <p>The undefined Trigger is listed in the Task List of the affected resource (section or paragraph) as well as its parent (form or paragraph list).</p> <p>The undefined Trigger is listed on the Finishing Report of the affected resource (section or paragraph) as well as its parent (form or paragraph list).</p>	<p>Choose from these options:</p> <ul style="list-style-type: none"> Choose Manage, Application, Triggers to check the SETRCPTB.DAL file out of the library and add the undefined trigger. Check the updated SETRCPTB.DAL file back into the library. <p>Or,</p> <ul style="list-style-type: none"> Change the name of the trigger to one that already exists in the SETRCPTB.DAL.
Recipients are in the Documaker document but not in the workspaces library.	<p>The undefined recipient is listed in the output area during the conversion process.</p> <p>The undefined recipient appears on the Task List of the applicable resource in Studio.</p> <p>The undefined recipient is listed on the Finishing Report of the applicable resource.</p>	<p>Choose from these options:</p> <ul style="list-style-type: none"> Choose Manage, Application, Definition to add the recipient to the BDF file. Check the updated BDF file back into the library. You could also use the Change Recipients option in the Conversion manger to perform this task. <p>Or,</p> <ul style="list-style-type: none"> Change the name of the recipient to one that already exists in the BDF.
Fonts are used in the Documaker document that are not listed in the FXR file assigned to the Studio workspace.	Studio substitutes fonts during the conversion process.	<p>If the font you want was not used, change the font in Section manager or Conversion manager (Change Multiple Sections).</p> <p>If you are importing a Paragraph List, you may need to modify fonts in the affected paragraphs using Paragraphs manager.</p>
Comments are added to Documaker document and various objects, such as fields or graphics on the document	<p>Comments from the Add-In are added to the applicable Studio file or object as task comment.s</p> <p>Studio displays task comments on Task Lists and the Finishing Report.</p>	<p>Run the Finishing Report against the imported files and perform the steps noted in the task comments.</p> <p>Then delete the task comments.</p>
Project and class information is in the Documaker document but not in a workspace INI file	Studio applies project and class information to the imported document if the flag to check converted files into the library is checked on.	No action required, but you can add the class and project to the library. To do so, choose Manage, System, Settings. Then under Options by Topic, choose Libraries. Expand the Library topic and click on LibraryManager.
An effective date is added to Documaker document.	<p>Studio applies effective date information to the imported document if the flag to check converted files into the library is checked on.</p> <p>The form effective date is applied to all sections contained in that form.</p> <p>A paragraph lists' effective date is applied to all paragraphs contained in that paragraph list file.</p>	No action required.

Add-In	Studio	Your response
<p>Metadata is created for the Documaker document</p>	<p>The metadata is added to the form during the conversion.</p>	<p>Open the form and review the form's metadata. Click the Form Metadata button to make any changes.</p> 
<p>Paper stock information is entered on the Advanced Document Properties tab. (It can only be entered for forms.)</p>	<p>During conversion, the output area displays the paper stock information. Here is an example: [03:45:41PM] Document type: Form [03:45:41PM] Paper Stock: <paper stock = 1></p>	<p>Note the Paper Stock information shown in the output area. You can save to a file, print, email, or copy and paste information in the output area.</p> <p>Open the form from disk or check it out of the library. Then use the Paper Stock option to select the applicable paper stock setting.</p> 
<p>Staple information is entered on the Advanced Document Properties tab. (It can only be entered for forms.)</p>	<p>During conversion, the output area displays the staple information. Here is an example: [03:45:41PM] Document type: Form [03:45:41PM] Staple: <staple - check staple on.></p>	<p>Note the Staple information shown in the output area. You can save to a file, print, email, or copy and paste information in the output area.</p> <p>Open the form from disk or check it out of the library. Then use the Staple option to turn on or off stapling.</p> 

SUMMARY

The Documaker Add-In for Word is a powerful complement to the Documaker suite to help empower business users to create documents in a fast and efficient manner. With the Add-In, users can operate in a familiar environment while leveraging the power of the Documaker publishing engine.

For more information please refer to the [Documaker Add-in for Microsoft Word User Guide](#).

Glossary

The following terms include definitions of system files as well as commonly-used terms.

AFP Advanced Function Printing (AFP), developed by IBM, is a print server language that generates data streams of objects. The data streams merge with print controls and system commands to generate Intelligent Printer Data Stream (IPDS). Your system then sends the IPDS to the AFP printer for printing. The GenPrint program can create spool files for AFP printers.

Application definition file Application definition files defines the key combinations used to locate a specific form set. These key combinations are comprised of a Key1 and Key2 (sometimes referred to as Unit1 and Unit2; or Group1 and Group2). In the insurance world, these keys are typically called: company and line of business (LOB). Other information stored in the BDF file includes the following:

- List of recipients
- Form categories (if used)
- Transaction type information
- Primary extract dictionary (XDD) file (if used)
- Default font cross-reference (FXR) file
- Default style (STY) file

Application definition files have an extension of *.BDF*.

Class Indicates the class of the resource. Forms, sections, paragraph lists, paragraphs and graphics are all library resources which can optionally have classes assigned to them. Classes are defined via INI options in Studio and are optional. Class can be used to group resources by product lines or geographical regions such as GA, TX, or MD.

DAL Document Automation Language (DAL) is the language you use when you tell the system how to calculate variable fields. This calculation is also called a script. When you select calculation options for a variable field, you can choose from:

- DAL Calc. Recalculates the value of all fields each time a user tabs to a new field in the section.
- DAL Script. Recalculates the value of the fields to which you assign the script only when a user tabs out of that field

Note You can find detailed information about the DAL language in the DAL Reference.

- .DAT files** Data table (DAT) files define various information the system uses as it processes information. All DAT are text files which have the extension *DAT*. Some DAT files are comma-delimited text files.
- The NAFILE.DAT file contains the variable data generated by the GenData program. This file, along with the POLFILE.DAT file, tell the GenPrint program what to print. This file also tells the GenWIP and GenArc programs what to place into WIP and what to archive.
- The GenWIP program also creates DAT files for each incomplete transaction it must process. These files are numbered sequentially and for each file there is a corresponding POL file which contains information about the forms to use.
- .DBF files** Database files (DBF) are used in several places in the Documaker system. For each DBF file, there is a corresponding MDX file which serves as its index. Examples of DBF files are FDB.DBF, which is created by Studio's Common Fields manager and WIP.DBF, which is created by the GenWIP program.
- .DFD files** Data field definition (DFD) files define to the system the file formats of the files generated by the system.
- An example of a DFD file is the TRNDFDFL file which the GenTrn program creates. The GenData program uses this file to read the TRNFILE which contains the actual transactions GenTrn creates.
- Duplex** A form printed on both the front and back sides of a sheet of paper is printed in duplex mode.
- See also *Simplex* on page 588.
- Effective date** The date on which you want the resource (forms, sections, paragraph lists, paragraphs and graphics are all resources) to become available for processing or selection in Documaker Server or Documaker Workstation. The effective date is also used in Studio's Library manager to aid in the versioning and revisioning of a resource.
- Embedded** All of the attributes of an embedded item are included in the file in which the item is embedded vs. a reference to that item to an external file. For example, when a graphic is embedded in a section, all of the attributes of that graphic are written or included in the section file. The graphic is neither shared nor loaded from the library.
- Extract files** Extract files are typically text files which contain the data the system processes. Extract files are created by another program, typically a database program, in a format the system can read. The text file format provides a standard interface into the system. For example, your data may be stored in a DB/2 or VSAM database from which you extract the data you want to process in the system in text format.

You can customize the system to read almost any type of file layout. The GenTrn program first reads the extract file and, using that extract data and TRNDFDFL.DFD file, creates transaction files (TRN files) the GenData program can use as it applies the processing rules and creates batch files, the NAFILE.DAT, and the POLFILE.DAT file.

The system includes a base extract file, called EXTRFILE.DAT, which serves as an example of the type of file the base system can read. You can use this file to experiment with the base system and determine how you want to set up your system.

- .FAP files The information which defines each section is stored in a FAP file. FAP files are text files with the extension *FAP*. You can edit FAP files using Studio.
- FDB.DBF file The FDB.DBF file is the default name of the database file created by the Common Fields manager which contains a record for each unique variable field you create. You can use any name you prefer as long as the extension is *DBF*. You can add records (variable fields) using the Common Fields manager or as you create sections in Studio.
- Fields A field, sometimes referred to as a variable field, is a blank field in your section or paragraph into which data is entered. The variable data can be entered manually during entry (Documaker Workstation) or merged during runtime (Documaker Server). Fields have attributes assigned to them including type, font, length, and so on.
- Form lists Form lists define a list of the forms that apply to each business unit and to maintain form-level triggering information. You can also define the order in which these forms should be provided.
- Information about each group of forms is stored in a file with a *GRP* extension. For each group (Key1/Key2), you have a separate GRP file. Group name (Key1/Key2) information is stored in the application definition (BDF) file, whereas specific information for each of the forms that make up the group is stored in a GRP file.
- Forms A form is a single document containing one or more pages or sections. Most forms contain multiple pages that are usually printed on both sides of a single sheet (duplex). Some forms are printed only on one side (simplex). Typical forms include insurance policies, tax returns, and mortgage documents.
- Forms include two types of data: fixed and variable.
- Fixed data is the same on every copy of the form. This includes items such as graphics, headers, and titles. This information remains constant regardless of the data entry.
 - Variable data may differ from form to form. This includes items such as individuals' names, addresses, and policy numbers. This information relates to the specific data processed on each form.
- Form files have an extension of *FOR*.

- Form sets** A form set is a group of logically related forms required to process a single transaction. A form set may contain one or many forms. You can group forms any way you want as you create form sets.
- FSISYS.INI file** The FSISYS.INI file is a one of the initialization (INI) files used by the system to set system parameters and to enable or disable system features.
- FSIUSER.INI file** The FSIUSER.INI file is one of the initialization (INI) files used by the system to set system parameters. For example, the FSIUSER.INI file contains information specific to each user, such as the location of files.
- .FXR files** Font cross-reference (FXR) files are used by the system so you can make sure your documents print the same way, regardless of which printer you choose. These files contain information about the various fonts you use and their equivalents on various printers.
- Graphics** A picture or image. Graphics can include a company logo, a signature, pictures, photos, and so on. Graphic files are typically stored in the workspace's library and have a file extension of *.LOG*.
- .INI files** Initialization (INI) files are used by the system to set system parameters and to enable or disable system features. Some examples of system INI files are: FAPCOMP.INI, FSISYS.INI, and FSIUSER.INI. For example, the FSISYS.INI file contains information the GenTrn program uses to determine when a new record starts and other information about the extract files the GenTrn program processes. The FSIUSER.INI file contains information specific to each user, such as the location of files and so on.
- Libraries** Studio lets you maintain multiple versions of various resources in a library. Libraries let you set up your master resource libraries and their associated files and resources, and then use those resources when you need them.
- Libraries can record descriptions when files are updated to a new version and provide file-lock capability to ensure that multiple users do not modify the same files. Libraries control resources much in the same way a traditional book library operates.
- You check out files to use them and check in files when you are done. You can also create versions that are for future use and date stamp them so they become active or are activated on the effective date. This lets you create a resource such as a form before it goes into effect. Libraries work with your master resource libraries to manage versions.
- Within any given resource library, you can have multiple versions of the same resource. The system identifies form versions by the effective date. The effective date is the date used in the processing environment (Documaker Workstation and/or Documaker Server) to determine which version of a resource to use.
- .LOG files** Logos and other graphics, such as scanned signatures, are stored as LOG files in the system. You use the Graphics manager to manage and manipulate LOG files.

Master resource library	The master resource library provides a central repository into which you can place all reusable resources such as sections, fonts, graphic files, data definitions, processing rules, and processing procedures.
Metacode	A printer definition language developed by Xerox. Metacode is the native language of Xerox's Centralized Printing Systems. The GenPrint program can create spool files for Metacode printers.
Metadata	Metadata is data that provides information about or documentation of other data that is managed in an application or environment. The purpose and use of these values is left to the user. A catalogue might be considered metadata because it describes books, clothes, and so on.
Objects	Objects are the individual items which comprise your image. Examples of objects are boxes, bar codes, lines, graphics, and text. All objects have unique attributes within the image. Attributes include items such as position, size, font type, and color.
Pages	Pages are the printed result of a section or a group of sections. You can have one section per page, several sections per page, or even a section that spans several pages. You determine the size of a page based on the size of your printed output. You can design forms for any size page your printer can print.
Paper stock	Paper stock allows users who have more than nine types of paper stocks to specify what paper stock the form should print on. Paper stock is used for Metacode printers only. In Studio, paper stock is defined at the form level. Data entered on a Documaker documents paper stock field is put onto a forms task comments when the Documaker document is converted.
Paragraph lists	Studio lets you create a list of paragraphs which lists all of the paragraphs from which either a Documaker Workstation user or Documaker Server can select at processing time. You insert this paragraph list into a multiline text field on a section. paragraph lists are stored in <i>.PSL</i> files.
Paragraphs	Studio lets you create canned paragraphs of text that can be inserted into a form. The system then uses the selected paragraph to fill a multiline text field. The way paragraphs are selected differs between Documaker Workstation and Documaker Server. With Documaker Workstation, a user selects which paragraph to use. With Documaker Server, trigger processing makes that determination. Paragraphs are stored in <i>.PAR</i> files.
PCL	PCL (Printer Control Language) is a printer definition language developed by the Hewlett-Packard company. The GenPrint program can create spool files for PCL printers.
PostScript	PostScript is a printer definition language developed by Adobe Systems which you can use on various printers. The GenPrint program can create spool files for PostScript printers. ⁴

- Projects** Indicates the project code assigned to a resource. Forms, sections, paragraph lists, paragraphs and graphics are all examples of library resources which can optionally have project codes assigned to them. Project codes are defined via INI options in Studio and are optional.
- Recipients** A recipient is any person, company, or other entity who receives a copy or copies of a form set, or any part of a form set such as a single form, or image. Examples of recipients are insurance policy owners (insured), agents, lien holders, and mortgage companies. Recipients are stored in the Application Definition file (BDF) of the workspace.
- Sections** A section is a group of text or graphics or both that make up a part of a form. You create sections using Studio. Each section is stored in a separate file (FAP file), so you can reuse sections in several forms and form sets. Multiple sections can comprise a single form. For instance, a three-page form with text and graphics, printed on both sides of each page, could contain a total of six sections. Some examples of sections include an insurance policy declaration page, the return portion of a bill, and page one of a 1040 Federal tax return form.
- You can choose to create a single page containing multiple sections, especially if you develop a page with graphics.
- SETRCPTB.DAT file** This file, also known as the *form set trigger table*, contains information which tells the GenData program which recipients receive which forms or sections.
- This file also contains the information the GenData program needs to determine whether or not to include or exclude a form.
- Simplex** A form printed on only one side of a sheet of paper is printed in simplex mode. See also *Duplex* on page 584.
- Triggers** Triggers define the criteria that must exist for content to be included in the form set. Triggers may be assigned to include or exclude:
- An entire form
 - Sections within a form
 - Text areas within a section
 - Optional paragraphs in a paragraph list
- Triggers are typically stored in the workspace's library and have an extension of *.DAL*.
- Variable data** Variable data may differ from form to form. This includes items such as individuals' names, addresses, and policy numbers. This information relates to the specific data processed on each form.

Workspace	A workspace, sometimes referred to as an MRL (master resource library) which contains all of the resources (such as forms, sections, paragraphs) and definition files (such as INI files) needed to compose and maintain the resources and files used in products such as Documaker Workstation or Documaker. Documaker Workstation and Documaker provide functionality to create policies (Insurance), letters and forms (Correspondence), and so on. A workspace is created using Studio.
Workspace definition files	<p>A workspace definition file contains information specific to a workspace and is a snapshot of several types of resources in a workspace at the time the file is generated. This file contains information about the following:</p> <ul style="list-style-type: none">• Common fields• DAL triggers• Recipients• Fonts• Project lists• Class lists• Graphics• Metadata <p>The Documaker Add-In for Word uses the information in this file to present its users with choices when they are creating Documaker documents in Word. Workspace definition files have an extension of <i>.WDF</i>.</p>
xBase	A generic term for industry-standard dBase IV file format.

INDEX

Numerics

4-State Customer bar code, 204

A

Acrobat Reader

included fonts, 500

Action field, 377

adding

printer fonts to the FXR file, 482

AFM files, 470

AFP

converting page segments, 276

fonts, 481

using custom fonts, 488

ANSI code page

for PC platforms, 493

application definition file

overview, 90

Application manager

granting rights, 73

ARCEFFECTIVEDATE attachment variable, 315

archive

and libraries, 312

INI options, 43

ArcSplit

INI options, 43

ASCII

code pages, 475

ASCII encodation, 201

Auto Lasso

options, 57

Auto Save, 48

AutoPagination option, 189

B

BDFFile option, 317, 321, 326, 329, 334

BDFLib option, 355

bitmap files

converting, 276

effective dates, 303

bitmap fonts

defined, 471

black and white

converting graphics, 279

brace matching

defining a color, 63

breakpoints, 520

business units, 23

C

CARFile, 317

CARFILE.DFD file, 332

CARFileDFD option, 333

CARFiles

creating libraries, 317

character sets

defined, 478

check in

defined, 299

check out

defined, 299

Class field, 349, 357, 360, 361, 363, 366, 369, 373, 378, 381, 393, 449

Class option, 322, 327, 331, 335

cloning users, 75

code pages

ASCII code pages, 475

code page 1004, 474, 476

code page 37, 477

code page 437, 474

code page 850, 474, 475

code page names, 478

converting text files from one code page to another, 496

EBCDIC code pages, 477

for EBCDIC platforms, 494

using the ANSI code page for PC platforms, 493

CODEPAGE.INI file

and PostScript fonts, 482

and the CPCNV utility, 488

defined, 499

comments

defining a color, 62

Comments field, 249

Common Fields manager

granting rights, 74

common INI options, 43

constants

defining a color, 62

converting

text files from one code page to another, 496

copy counts

using variables, 168

create dates

overview, 307

Create New Workspace option, 28

CreateIndex option, 323, 328, 331, 335, 385

CreateTable option, 319, 323, 325, 328, 331, 335

CreateTime field, 307, 309, 311, 312

- cross hairs, 54, 56
- CSV files
 - importing, 401
- custom fonts, 487
- D**
- DAL scripts
 - defining editor settings, 62
- DAL variables, 169
- DALFile option, 314, 317, 321, 326, 329, 334
- data code word, 199
- Data Entry check, 180
- Data Extract manager
 - granting rights, 74
- Data Matrix
 - bar codes, 199
 - symbol sizes, 199
- database
 - INI options, 43
- Database option, 323
- databases
 - storing user information, 77
- DB2, 43
 - using the native driver, 319
 - using the ODBC driver, 324
- dBase, 589
- DBHandler option, 322, 327, 330, 331, 334, 335
- DBTable
 - USERINFOSQL control group, 77
- DBTable option, 322, 327, 330, 334
- DCD files
 - mapping fonts, 503
- DDT files
 - omitting, 48
 - stringent checking, 305
- DDTFile option, 318, 321, 326, 330, 334
- DDTLib option, 355
- Debug option, 323, 328, 331, 335, 384
- Debug_FontSubs option, 420
- Definition Lookup manager
 - granting rights, 74
- DEFLib option, 355
- Deployment manager
 - granting rights, 74
- Development Tools
 - INI options, 43
- dictionaries, 23
- Dictionary-Rule manager
 - granting rights, 74
- Documaker Add-In for Microsoft Word, 555
- Documaker Bridge
 - and Library manager, 298
- Documaker Server
 - and libraries, 312
 - and Library manager, 298
- Documaker Workstation
 - and libraries, 309
 - and Library manager, 298
- Documanager, 31
- Documerge Field report, 465
- Docupresentment
 - and Library manager, 298, 314
- DPRInitLby rule, 314
- DPRRetrieveFormset rule, 315
- dprtrc.log file, 384
- DXM files, 50
- DXS files, 50
- E**
- EBCDIC platforms
 - and code pages, 477
 - using Code Page 37, 494
- Edit menu, 20
- effective date, 31
- Effective Date field, 31, 348, 356, 363, 365
- effective dates
 - defined, 299
 - overview, 303
 - printer resource files, 303
 - setting rights, 72
 - ShowEffectiveDate option, 309
 - versions and revisions, 304
- embedded
 - sections, 127
- embedding fonts, 500
- Enable_Debug_Options option, 384, 420
- entry
 - INI options, 43
- ErrFile option, 305
- ErrorOnMissingFile option, 305
- expiring resources, 72
- export
 - INI options, 43
- Extended Binary Coded Decimal Interchange Code, 477
- extract files
 - and code pages, 496
 - Primary Extract Dictionary field, 92
- F**
- FAP files
 - mapping fonts, 503
- FAPCOMP.INI file
 - mapping fonts, 503
- fields
 - showing tips, 54
 - tips, 56
 - undefined, 249

- File Information window, 315
- File menu, 18
- filters
 - defining settings, 58
- Font check, 180
- font cross-reference
 - defining, 92
- font cross-reference files, 30
 - adding printer fonts, 482
 - choosing, 491
 - for Monotype fonts, 484
- Font manager
 - granting rights, 74
- FontFamilyMatching control group, 503
- fonts, 24
 - AFP, 481
 - bitmap fonts, 471
 - custom fonts, 487
 - effective dates, 303
 - font substitution in Windows, 480
 - FXR files for Monotype fonts, 484
 - how computers and printers use fonts, 472
 - installing screen fonts in Windows, 481
 - matching, 43
 - Metacode, 482
 - Monotype fonts, 483
 - naming conventions, 502
 - PCL, 482
 - PostScript, 472, 482
 - printer fonts, 481
 - scalable fonts, 471
 - screen fonts, 480
 - synchronizing with the Add-In, 561
 - terminology, 468
 - True Type, 472
- footers, 147
 - print control options, 222
- FORFile option, 318, 321, 326, 330, 334
- FORLib option, 355
- Form List manager
 - granting rights, 73
- FormFile option, 314, 318, 321, 326, 330, 334
- FormLib option, 355
- forms, 23
 - creating, 75
 - effective dates, 303
 - terminology, 126
- Forms manager
 - granting rights, 73
- FSISYS.INI file
 - currently in use, 50
- FSIUSER.INI file
 - creating workspaces, 28
 - currently in use, 50
- functions
 - defining a color, 62
- FXR File Name field, 92
- FXR files
 - affect on display and print quality, 490
 - choosing FXR files, 491
- G**
- GetRunDate rule, 308, 312
- grammar
 - checking on a save, 57
- graphics, 23
 - converting, 276, 279
 - defining settings, 60
 - effective dates, 303
 - for the Add-In, 562
 - resizing, 279
 - resolution, 281
 - reversing, 279
 - rotating, 278
 - undefined, 249
- Graphics manager
 - granting rights, 74
- grid
 - defining, 52
 - showing, 54, 57, 60
- GRPFile option, 318, 321, 326, 330, 334
- GRPLib option, 355
- GVM variables, 169
- H**
- headers, 147
 - print control options, 222
- Help menu, 26
- hooks
 - defining, 101
- HPINTL.FXR file, 484
- HPINTLSM.FXR file, 484
- HTML file
 - stacking objects in order, 275
- I**
- importing
 - a signature or logo, 278
- importt
 - INI options, 43
- In Use status, 68
- Increment Version field, 348, 356
- indenting, 63
- Information pane, 44
- INI options
 - access to, 70
 - setting, 71

- Insert menu, 176
- installing
 - screen fonts in Windows, 481
- Intelligent Mail® bar code, 204
- international language support, 493
- IStream
 - converting RTF files, 423

- J**
- JPEG files
 - converting, 276

- K**
- KeepFiles option, 384
- keywords
 - defining a color, 62

- L**
- language
 - international language support, 493
 - national language terminology, 469
 - using international characters, 495
- LBRYMGR utility
 - and DB2, 319
 - and DB2 ODBC driver, 324
 - Oracle ODBC driver, 333
 - SQL Server ODBC driver, 329
- LBYP option, 323, 328, 331, 336
- LBYP option, 323, 328, 331, 336
- LbyLib option, 314, 318, 384
- LBYP option, 323, 328, 331, 336
- LbyLogFile option, 318, 322, 327, 330, 334
- LBYSQLR job, 320
- LDAP
 - setting up, 81
 - using, 80
- LDAP_Enabled option, 81
- LibAutoRefresh option, 350, 355, 357
- libraries, 23
 - access to, 70
 - creating, 72, 316
 - creating a DB2 library, 319
 - data portion, 302
 - default name, 317
 - defined, 299
 - format, 317
 - how it all works, 309
 - index tables, 301
 - INI options, 43
 - overview, 298
 - purging history, 72
 - setting access rights, 72
 - understanding, 301
- Library field, 359, 368, 372, 377
- Library Manager
 - error messages, 385
- library tiers
 - INI options, 43
- LibraryManager control group, 350, 355, 357
- Lightweight Directory Access Protocol, 80
- line numbers, 63
- line of business
 - overview, 90
- Locale option, 44
- Locked field, 363
- logging in, 76
- LogoFile option, 314, 318, 321, 326, 330, 334
- LogoLib option, 277, 355

- M**
- margins
 - showing, 56
- master.ddt files
 - Dictionary Rule manager, 50
- MASTER.LBY file, 317
- Menu Bar, 132, 133, 173, 174, 266, 267, 284, 285, 520
- menu bar, 132, 173, 266, 284
- menus
 - Edit, 20
 - File, 18
 - Help, 26
 - options, 16, 23
 - Tools, 21
- Metacode
 - fonts, 482
- Mode field, 348, 357, 360, 361, 363, 365, 369, 373, 378, 380, 393, 449
- modification date
 - defined, 299
- Monotype fonts
 - FXR files, 484
 - using system fonts, 483

- N**
- Name field, 359, 363, 368, 369, 372, 377
- NetMove
 - configuring, 259

- O**
- objects
 - hiding, 58
 - stacking order, 275
 - terminology, 126
- ODBC, 31, 43
- operators
 - defining a color, 63

- Options menu, 23
- Oracle, 43
 - ODBC driver, 332
- overlays
 - effective dates, 303
- P**
- pages
 - showing boundaries, 54
- pagination, 189
- Paragraph List manager
 - granting rights, 73
- Paragraph manager
 - granting rights, 73
- Passwd option, 323, 328, 331, 335
- passwords, 68
- PCL
 - custom fonts, 488
 - fonts, 482
- PDF
 - incompatibilities, 487
- PDF files
 - converting, 425
 - fonts, 500
- PDF417 fonts, 486
- placeholders
 - field style, 52
- PLANET Code® bar code, 203
- PNG files
 - importing, 275
- PNG formats, 275
- policy date, 31
- positioning
 - JPG objects, 275
- PostScript
 - custom fonts, 488
 - fonts, 472, 482
- PostScript fonts
 - included with Acrobat Reader, 500
- Poweroffice
 - using with Library Manager, 337
- primary extract dictionary
 - defining, 92
- Primary Extract Dictionary field, 92
- print
 - INI options, 44
- Print Control options, 222
- Printcommander
 - creating FAP files, 256, 262
 - downloading and installing, 256
 - fidelity, 262
 - virtual printers, 257
- printers
 - adding fonts to the FXR file, 482
 - AFP fonts, 481
 - Metacode fonts, 482
 - PCL bitmap fonts, 482
 - PostScript fonts, 482
 - resource files, 303
 - using custom fonts, 487
 - using printer fonts, 481
- Printstream Analyzer
 - setting up, 64
- Project field, 349, 357, 360, 361, 363, 366, 369, 373, 378, 381, 393, 450
- project management
 - library tiers, 43
- projects
 - assigning roles, 70
 - granting rights, 75
- promotion
 - defined, 300
- promotions
 - setting rights, 72
- Q**
- Qualifier option, 328, 331, 335
- R**
- recipients
 - undefined, 249
- RecStatus column, 354
- regional date processing, 103
- RegionalDateProcess rule, 103
- REL112.FXR, 486
- REL112SM.FXR, 486
- relative paths, 29
- RemoveImageMissingDDT option, 305
- resizing
 - logos, 279
- Resolution option
 - Graphics manager, 281
- resources
 - access to, 70
 - checking in, 356
 - checking out, 354
 - creating, 74
 - editing information, 365
 - granting rights, 72
 - path setup, 44
 - securing, 72, 78
 - unlocking, 358
- response files
 - defined, 299
- RetrieveVersionInfo option, 310
- reversing
 - logos, 279

- revision
 - defined, 299
 - Revision field, 360, 363, 368, 372
 - revisions
 - overview, 303
 - run dates, 307
 - stringent checking, 305
 - rotating
 - logos, 278
 - RTF files
 - converting IStream migration files, 423
 - mapping fonts, 503
 - RTFDefaultFont option, 420
 - RTFFontMAP control group, 503
 - RTFFontSubs control group, 419
 - Ruler Units option
 - Logo Manager, 282
 - rulers
 - showing, 54, 57, 60
 - rules processing
 - INI options, 44
 - using international characters, 495
 - run date, 31
 - format, 307
 - run dates
 - overview, 307
 - RunDate field, 307, 309, 311, 312, 315
 - RunDate GVM, 312
 - RunDate option, 307
- S**
- sandbox
 - location, 71
 - scalable fonts, 471
 - scratch pads, 35
 - screen fonts
 - installing in Windows, 481
 - using, 480
 - Script Name field, 363
 - scripts, 23
 - sections, 23
 - embedded, 127
 - Sections manager
 - granting rights, 73
 - security rights
 - LDAP, 80
 - Security wizard, 68, 69
 - Server option, 328, 331, 335
 - SetOrigin rules, 190
 - Setup Data field
 - example, 491
 - ShowEffectiveDate option, 309
 - signatures
 - importing, 278
 - Spell check, 180
 - spelling
 - checking on a save, 57
 - SQL, 31
 - ODBC driver, 329
 - stacking order, 275
 - state compliance, 103
 - Status Bar, 132, 266, 284, 520
 - status bar, 48
 - Status field, 315, 348, 357, 360, 361, 363, 365, 369, 373, 378, 380, 393, 449
 - status line, 132, 266, 284, 520
 - StringentChecking option, 305
 - Style File Name field, 93
 - Style manager
 - granting rights, 73
 - system administrator
 - designating, 70
 - system resource files
 - uploading, 496
- T**
- Tabs
 - defining, 63
 - Tagcommander, 258, 402
 - task comments, 249
 - Task Comments field, 249
 - task list, 249
 - Template manager
 - granting rights, 74
 - templates, 147
 - creating forms without a template, 75
 - terminology
 - fonts, 468
 - forms, 126
 - objects, 126
 - test, 24
 - Text Case field, 370
 - Text Element Name field, 369
 - text files
 - converting from one code page to another, 496
 - Text String field, 370
 - TGA files
 - importing, 402
 - Title, 132, 173, 284, 520
 - title bar, 132, 173, 284, 520
 - tool bar, 132, 173, 266, 284, 520
 - toolbars
 - displaying, 48
 - icons, 48
 - Tools menu, 21
 - trace files, 384
 - TraceFile option, 384
 - transaction types

defining, 101
 Trigger manager
 granting rights, 74
 triggers
 creating, 75
 defining editor settings, 62
 report, 155
 undefined, 249
 TrueType fonts, 472
 description, 482
 Type field, 359, 363, 368, 369, 372, 377

U

unexpiring resources, 72
 UniqueTag option, 322, 327, 330, 331, 335
 units of measurement
 defining, 52
 unlock
 defined, 300
 user database
 managing, 71
 User ID field, 363, 378
 UserID column, 354
 UserID option, 323, 328, 331, 335
 USERINFOSQL table, 77
 users, 24
 adding, 67
 changing information, 69
 cloning, 75
 defining, 33
 deleting, 69
 managing, 66, 71
 Security wizard, 68
 setting rights, 69
 specifying a login message, 68
 storing information, 77
 supervisors, 68
 user IDs, 67, 68
 using LDAP, 80
 using
 ANSI code page for PC platforms, 493
 custom fonts, 487
 printer fonts, 481
 screen fonts, 480

V

variable fields
 creating, 75
 Version field, 359, 363, 368, 372
 versions
 defined, 300
 overview, 303
 run dates, 307

stringent checking, 305
 View menu, 175

W

WarnOnMissingFile option, 305

Windows

font substitution, 480
 installing screen fonts, 481
 using the ANSI code page, 493

WIP

and libraries, 311
 INI options, 44

Word

importing files from the Add-In, 568

workflow, 315

Workspace, 132, 135, 266, 284

workspace definition file (WDF), 555, 561, 564, 565, 566

workspaces

creating, 27
 joining, 34
 managing a shared workspace, 70
 new, 12
 shared, 50

workstation access levels, 68

X

Xbase, 589

Xerox

converting font files, 276
 converting image files, 276
 using custom fonts, 488

Xerox font

importing signatures or logos, 278

Z

Z-Index option, 275

