

Oracle® Documaker

**Documaker Studio User
Guide**

version 11.4

Part number: E14902-01

September 2009

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

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

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

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

U.S. GOVERNMENT RIGHTS

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

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

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

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

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/version 0.96](http://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

Chapter 1, Using Documaker Studio

- 3 Introduction
- 5 Studio Options
- 8 Creating a Multi-user Development Environment
- 11 Installing Studio
- 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
68	Adding Users
70	Changing User Information
70	Deleting Users
70	Setting Rights Using the Security Wizard
77	Cloning User Rights
77	Logging In
78	Storing User Information in Another Database
79	Adding Security at the Resource Level
81	Using the LDAP Security Model

Chapter 3, Working with Application Definition Files

90	Overview
91	Defining a Business Unit
96	Storing Recipient Information
97	Defining Categories
98	Defining Transaction Types
99	Using Regional Date Processing
99	Setting up Regional Date Processing
102	Setting Up Regional Date Processing Rules
108	Generating Reports

Chapter 4, Working with Form Lists

112	Creating a Group
115	Opening a Group
116	Setting Up Triggers
118	SetRecip Options

Chapter 5, Managing Forms

- 122 Overview
- 123 Using the Screen
 - 124 Using the Menu Bar
 - 124 Using the Section Menu
 - 125 Using the Trigger Menu
 - 126 Using the Workspace Menu
 - 127 Using the Tool Bar
- 129 Creating a Form
- 132 Modifying Forms
 - 136 Generating Readability Statistics
- 139 Using Form Templates
 - 139 Using Template Manager
 - 141 Using Form Manager
 - 142 Saving a Form as a Template
 - 143 Controlling Access to Templates
 - 144 Library Manager and Reports
- 145 Setting Up Triggers
 - 147 Editing a Trigger
 - 147 Using the DAL Trigger Report
- 149 Handling Overflow
 - 152 Overflow Examples
 - 153 User Functions
 - 155 Overflow Variables
- 157 SetRecip Options
 - 158 Understanding Recipient Counts
 - 160 Using Variables to Set Copy Counts

Chapter 6, Working with Sections

- 164 Overview
- 165 Using the Screen
 - 167 Using the Menu Bar

	168	Using the Edit Menu
	170	Using the View Menu
	171	Using the Insert Menu
	173	Using the Format Menu
	174	Using the Tools Menu
175		Using the Right Click Menus
177		Using the Tool Bar
180		Creating a Section
182		Opening a Section
185		Adding Objects
188		Using Data Matrix 2-D Bar Codes
193		Using Printcommander to Create FAP Files
	193	Downloading and Installing Printcommander
	194	Creating a Virtual Printer
	195	Configuring a Docucorp Virtual Printer
	195	Installing Tagcommander
	195	Installing Docucreate
	196	Configuring NetMove
	199	Creating a FAP File
	200	Possible Errors
	200	Uninstalling Control Panel, Common Objects, and Printcommander
	200	Updating Control Panel, Common Objects, and Printcommander

Chapter 7, Working with Graphics

202		Using the Screen
203		Using the Menu Bar
	203	Using the Graphic Menu
205		Using the Right-Click Menu
208		Using the Tool Bar
210		Managing Graphics
212		Opening a Graphic
214		Editing a Graphic
215		Setting Rotation Names
	216	Changing the Resolution

217 Setting Ruler and Grid Options

Chapter 8, Creating Scripts

- 220 Using the Screen
 - 221 Using the Menu Bar
 - 221 Using the Edit Menu
 - 222 Using the Script Menu
 - 223 Using the Right-Click Menu
 - 224 Using the Tool Bar
- 226 Overview
- 228 Checking Syntax

Chapter 9, Managing Resources

- 234 Overview
- 235 Terminology
- 237 Concepts
 - 237 Understanding Libraries
 - 239 Processing with Effective Dates
 - 240 Retrieving Resources with Version and Revision Numbers
 - 241 Stringently Checking Resources
 - 243 Understanding Run Dates
- 245 How It All Works
 - 245 In Documaker Workstation
 - 248 In Documaker Server
 - 249 In Docupresentation (IDS)
- 250 Managing Workflow
- 251 Creating Libraries
 - 252 Using xBase and CARFiles
 - 252 Creating the CARFile and Index Files
 - 254 Using the DB2 Native Driver
 - 254 Creating the Database and Tables
 - 256 Sample INI Options
 - 259 Using the DB2 ODBC Driver
 - 259 Creating the Database and Tables

	261	Sample INI Options
265	Using the SQL Server ODBC Driver	
	265	Creating the Database and Tables
	265	Sample INI Options
269	Using the Oracle ODBC Driver	
	269	Data Format Definition (DFD) Requirements
	270	Creating the Database and Tables
	270	Sample INI Options
274	Using Documange	
281	Working with Libraries	
282	Opening a Library	
284	Adding Resources to a Library	
287	Importing Files	
288	Importing Libraries	
289	Migrating a Workspace Library	
291	Checking Out Resources	
293	Checking In Resources	
295	Unlocking Resources	
296	Promoting Resources	
	299	Performing the Promotion
300	Filtering Resources	
302	Editing Resource Information	
304	Deleting Resources	
305	Searching the Library	
	308	Performing the Search
309	Extracting Resources	
	312	Performing the Extraction
313	Expiring Resources	
314	Reviewing a History of Resource Changes	
316	Running Response Files	
317	Defining Mode, Status, Class, and Project Options	
319	Using the LBYPROC Utility	
322	Troubleshooting	
	322	Turning on Tracing
	323	Handling Error Messages

Chapter 10, Using Dictionaries

- 326 Working with Extract Files
- 327 Opening an Extract Dictionary
- 329 Working with Fields
- 330 Importing TGA Files

Chapter 11, Creating Tables

- 332 Overview
 - 333 Transaction File DFD
 - 334 Recipient Batch File DFD
 - 334 Application Index File DFD
 - 335 Field Section
 - 336 Key Section

Chapter 12, Converting Files

- 338 Overview
- 340 Converting Files into FAP Files
- 341 Compiling FAP Files into Print Files
- 342 Converting Graphic Files
- 343 Making Changes to Multiple FAP Files
 - 344 Section Conversions
 - 346 Field Conversions
 - 348 Font Conversions
 - 349 General Conversions
- 351 Converting Bitmap Files into LOG Files
- 352 Finishing a Conversion

Chapter 13, Printing Reports

354 Generating a Report

Chapter 14, Handling Fonts

360 Overview

360 Changing the Point Size

Chapter 15, Testing Your Work

364 Overview

366 Using the Screen

366 Using the Menu Bar

367 Using the View Menu

367 Using the Test Menu

369 Using the Right-Click Menus

371 Using the Tool Bar

373 Creating a Test Profile

376 Running a Test

379 Running a Trigger-Run Only Test

379 Manually Triggering

381 Changing Test Properties

382 Modifying the AFGJOB File

383 Deleting a Test

384 Using the Printstream Analyzer

386 Using the Record View and Display Panes

387 Using the Find Option

388 Finding a Specific Page or Record

389 Displaying Information about the Record

390 Printing What is in the Panes

Chapter 16, Deploying a Library

392 Overview

393	Creating or Running a Deployment
395	Creating a Library Deployment
397	Creating a Flat-File Deployment
398	Additional Resources
399	INI Settings
400	Processing the Deployment

401	Index
-----	-------

Chapter 1

Using Documaker Studio

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 the compliance group and create the dynamic document applications that are provided to the production operations teams.

This document includes information on these topics:

- [Introduction on page 3](#)
- [Creating a Multi-user Development Environment on page 8](#)
- [Starting Studio on page 12](#)
- [Using System Menus on page 16](#)
- [Creating Workspaces on page 27](#)
- [Joining a Workspace on page 34](#)
- [Working with Settings, Users, and Security on page 37](#)
- [Working with Application Definition Files on page 89](#)
- [Working with Form Lists on page 111](#)
- [Managing Forms on page 121](#)
- [Working with Sections on page 163](#)
- [Working with Graphics on page 201](#)
- [Creating Scripts on page 219](#)
- [Managing Resources on page 233](#)

- [Using Dictionaries on page 325](#)
- [Creating Tables on page 331](#)
- [Converting Files on page 337](#)
- [Printing Reports on page 353](#)
- [Handling Fonts on page 359](#)
- [Testing Your Work on page 363](#)
- [Deploying a Library on page 391](#)

INTRODUCTION

Documaker Studio provides a new and integrated way to create and maintain the sections, graphics, fields, and other objects you use in your form sets. Studio also provides tools for managing these objects in libraries and for testing and deploying form sets.

NOTE: Licensed users of Documaker Server or Documaker Workstation, and Docucreate (RP) are eligible to receive Documaker Studio as part of their maintenance agreement. The Documaker product has a high degree of compatibility with the DAP (Document Automation Platform) Solution Suite. DAP users should check with their account representative to explore the advantages of upgrading to Documaker.

The main features and benefits of Studio include:

- *Seamless workbench.* While its predecessor provided a menuing system to access all of the graphical tools and many of the conversion tools, Studio greatly enhances the way the tools work together. For example, you can now drag and drop a field database entry onto a section. Furthermore, you can have several tools, and even multiple copies of those tools, open at the same time.
- *Editing Forms.* Studio makes it easy to edit forms by providing easy access to both the tree view and graphical view of the selected form components, as well as a Trigger wizard for specifying trigger logic. SetOrigin specification of the absolute or relative placement of sections (FAP files) is now done visually. Header, body, and footer sections are automatically organized into a easier to manage tree view. Section and form attributes are easier to access and modify. Recipients are easier to specify. Multi-user development is easier since each developer can work on one or more forms at a time without affecting other developers.
- *Configuring INI options.* You can now load and maintain the INI options which control your system from inside Studio. You can maintain these options at the global (system), local (unique to a developer), or deployment (specifying unique platforms, data sources, locations, and devices) level.
- *Creating DAL Scripts.* DAL (Document Automation Language) offers a high degree of power and flexibility for form and section triggering logic and also for data manipulation and formatting. While other means of triggering sections and manipulating data are still supported, you'll enjoy the benefits and ease of use DAL offers. Studio includes a powerful script editor for maintaining DAL libraries, including effective date version control and check-in and check-out.
- *Visual Debugger.* Dynamic documents cannot truly be viewed without running the rules against test data. Studio includes a new visual testing tool that makes it easy to see how documents will look when run against real data. The ability to set break points and view internal variables while watching the document assemble visually makes it easy to debug problems in the form design.
- *Import Wizards.* The version 11.x runtime works with older configurations, however, taking advantage of the new Studio and its new files requires a forward migration. Studio includes powerful wizards to make it quick and easy to move an existing implementation forward to version 11.x library members.

- *Form-level effective dates.* Effective date based version control (including check-in, check-out, and expiration) can now be applied to forms. Previously this capability applied only to sections of forms, such as FAP files, and their related DDT files. Many implementations creatively combined a large number of physical sections into a more complex logical form, and the need for effective date versioning at the true form level became apparent. Adding this capability requires a new library member: the *form* (FOR) file and a new tool to maintain the objects and attributes of that file.
- *Multi-user form development.* Versions prior to 11.0 stored form level attributes in a number of places: the FORM.DAT file, the SETRCPTB.DAT file, other supporting tables, and DDT files. By consolidating logical form attributes into a single file, not only is it possible to do effective date version control on a form, but it is also easier to facilitate multi-user form development.
- *New files.* The elimination of the FORM.DAT and SETRCPTB.DAT files and the introduction of FOR files also resulted in the addition of two new types of resource members: the *base definition* (BDF) file which defines the combinations of keys that represent the business units into which the forms are assembled, and the *group* (GRP) file which provides the default ordering sequence for the form candidates that are available to each business unit.
- *Eliminating DDT files.* DDT files were introduced as a way to isolate variable data tag information that would be applied to static form overlays. By separating this information from the FAP files, it was possible to achieve a higher degree of performance in simpler implementations. Over time, as advanced formatting capabilities were devised to meet customer needs, it became necessary for the FAP files to be available at runtime because more and more documents began to be subject to dynamic composition at runtime. The DDT file became less of an advantage, and even a stumbling block within implementations.

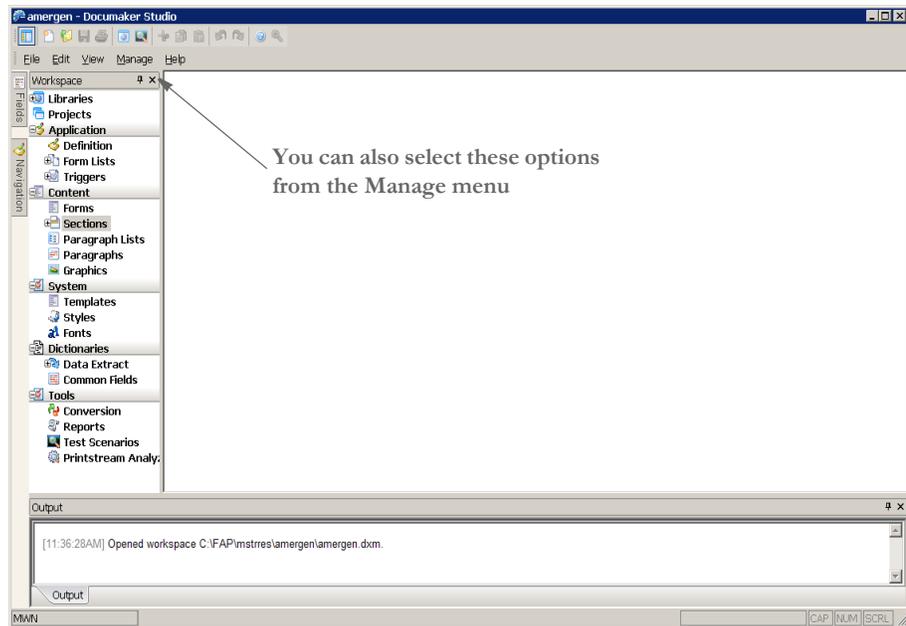
In spite of synchronization facilities in the tools, some implementations struggled to keep DDT and FAP files in sync. With version 11.0 and the introduction of the FOR file, DDT files are eliminated. Section and field-level DDT rules previously stored in the DDT file are now either unnecessary or are stored in the FAP file. Having image level rules (such as SetOrigin) in the FOR file makes it easier to do visual form design. Having field level rules in the FAP file eliminates synchronization worries.

The following pages discuss some of these enhancements in more detail.

STUDIO OPTIONS

The options you can use are listed here

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:



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

Option	Used to	Used during form		
		Entry	Processing	Required?
Content				
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 (IPL 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

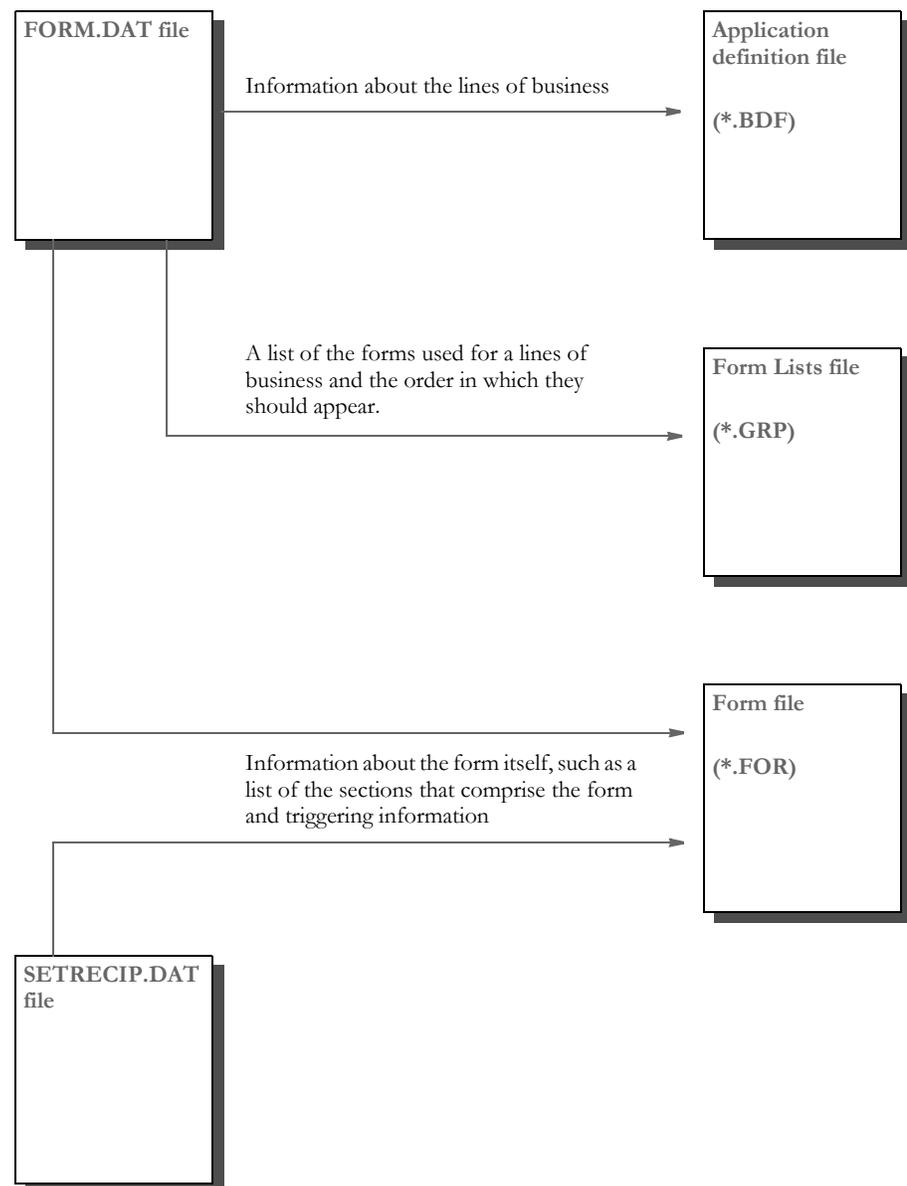
Option	Used to	Used during form		
		Entry	Processing	Required?
Reports	Create reports	No	No	No
Test Scenarios	Define and process test situations that simulate your production environment.	No	No	No
Printstream Analyzer	Analyze print files.	No	No	No

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.

The FORM.DAT file defined the basic lines of business and the available list of forms for each of those lines. As such, there was too much information in this one file for it to be useful in a multi-user development environment. Therefore the FORM.DAT file, along with the SETRECIP.DAT (the trigger file), have been split into multiple components.



NOTE: File names, types, and extensions, as well as menu names, options, and screen names may 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, within 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 111](#) 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 121](#) for more information.

INSTALLING STUDIO

If your system supervisor has not installed the system on your computer, you can do so easily, once you have made sure your computer has the correct hardware and software. Simply insert the CD-ROM into the appropriate drive and follow the instructions that appear on your screen.

NOTE: In most cases, the system supervisor will set up Studio on your computer for you. If you install the system yourself, check with your system supervisor to make sure any custom settings are configured correctly for your computer.

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).

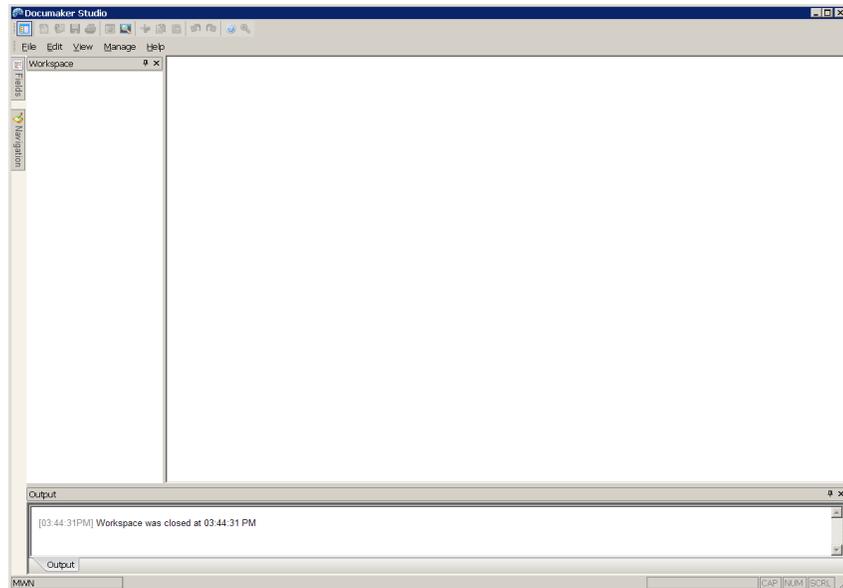
Keep in mind...

- You must have Windows 2000 or later installed on your computer to run Studio.
- Installed under the \DLL directory are files needed to display INI configuration and help information. These files will continue to be updated with subsequent patch releases.
- There is a separate installation for Documaker Server and Documaker Studio.
- You can find the latest installation information on this site:

<http://www.oracle.com/skywiresoftware/index.html>

STARTING STUDIO

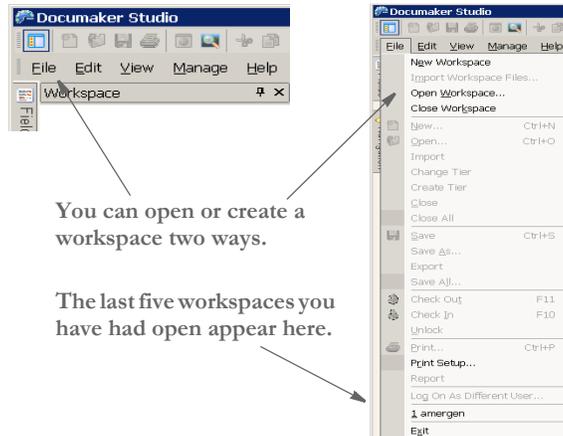
When you first start Studio, the following window appears.



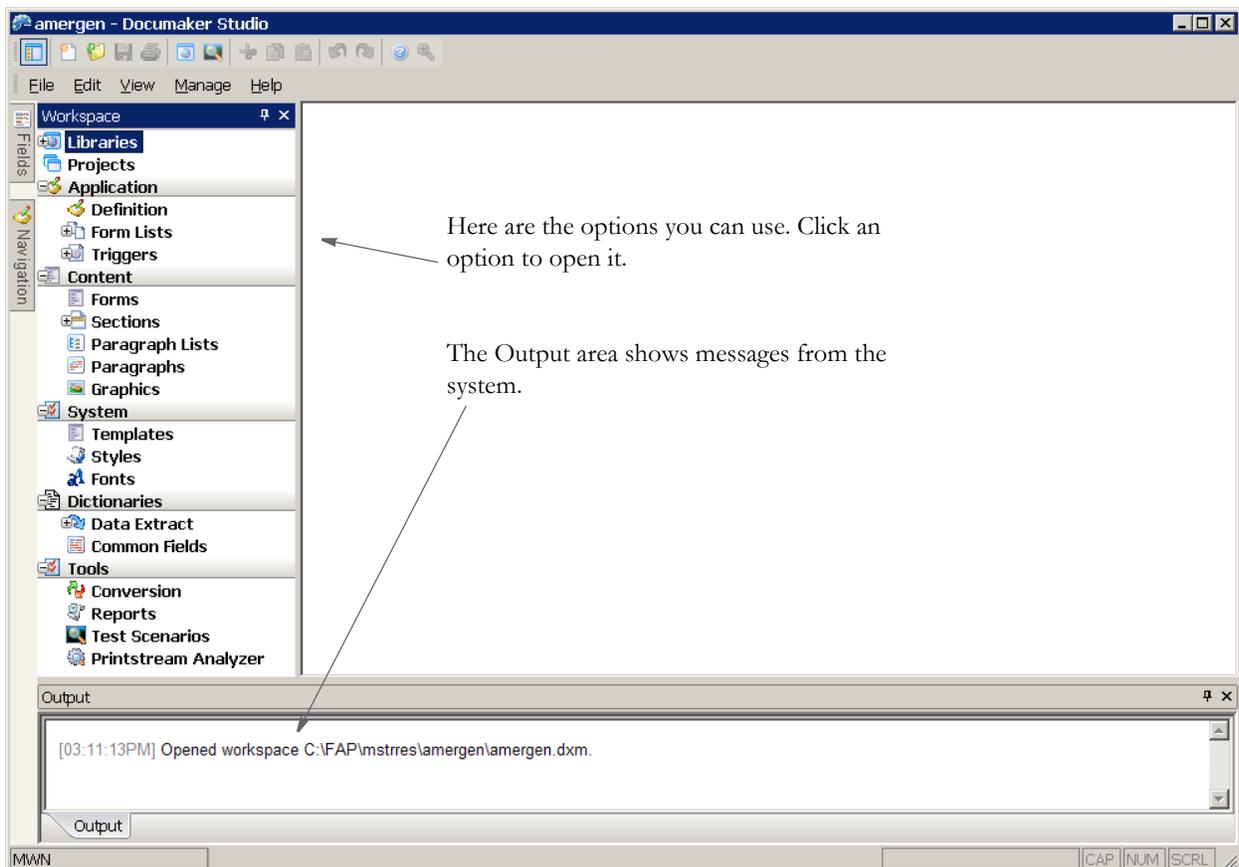
The first step is to open an workspace or create a new workspace.

NOTE: 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](#).

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



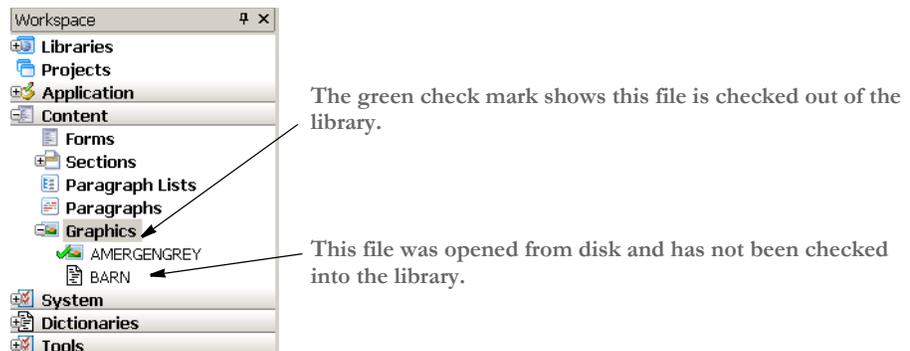
Once open, Studio shows the options you can use to work in that workspace:



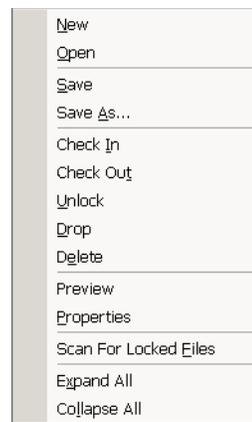
Having the tools displayed in a *tree* arrangement provides an easy way to see the files you checked out from the library. It also serves as a most recently opened file list for all files, including those not in the library.

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.

Option	Description
Unlock	Use to unlock an item that is open when you want to release it back to the library without saving changes.
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:

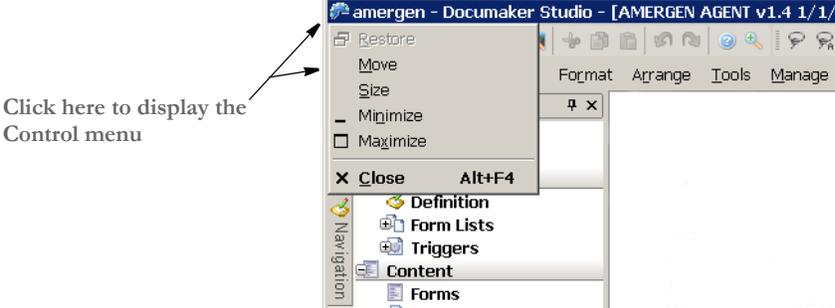
Select	To
File	Create, open, close, save, check in, check out, unlock, create tiers, change tiers, print files and reports, and log on as a different user. You can also use this menu to exit the system.
Edit	Perform normal editing functions such as undo, redo, cut, copy, paste, and delete.
View	Set to turn on or off the display of the tool bar, 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 additional 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:

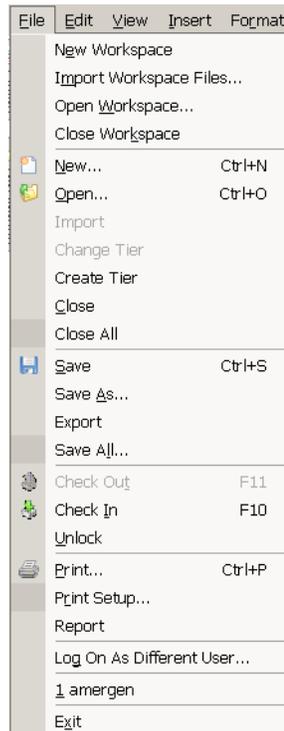


This menu provides these options:

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.

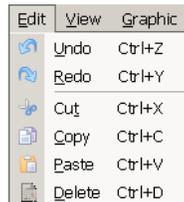
Select	To
New Workspace	Create a new workspace (DXM file).
Import Workspace Files	Import a workspace file.
Open Workspace	Open a workspace.
Close Workspace	Close the current workspace. The system prompts you to save your work.
New	Create a new object, such as a section.
Open	Open as object, such as a 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.

Select	To
Save	Save the object you are working with.
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:



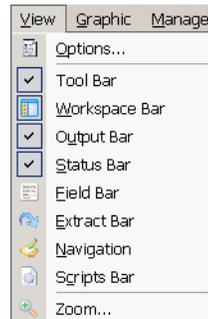
The options on the Edit menu can vary, depending on which tool you are using. This example shows you the standard Edit menu options. Here is a brief summary of these options:

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.
Tool Bar	View the tool bar.
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 may 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.

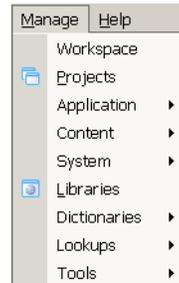
Click the auto hide icon again and Studio hides the bar and adds a tab on which you can click to maximize the bar when you want to work with it again.



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.

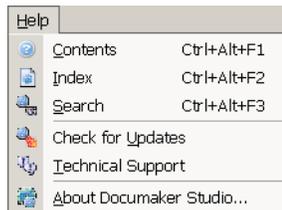
Select...	To...
Definitions	Define tables used by both the batch and entry processing.
Conversion	Convert files from one type to another, such as RTF to FAP and to change characteristics, such as a font ID) on multiple FAP files.
Reports	Print base definition, group, form, or section reports.
Fonts	Organize 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.

Select...	To
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	
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:



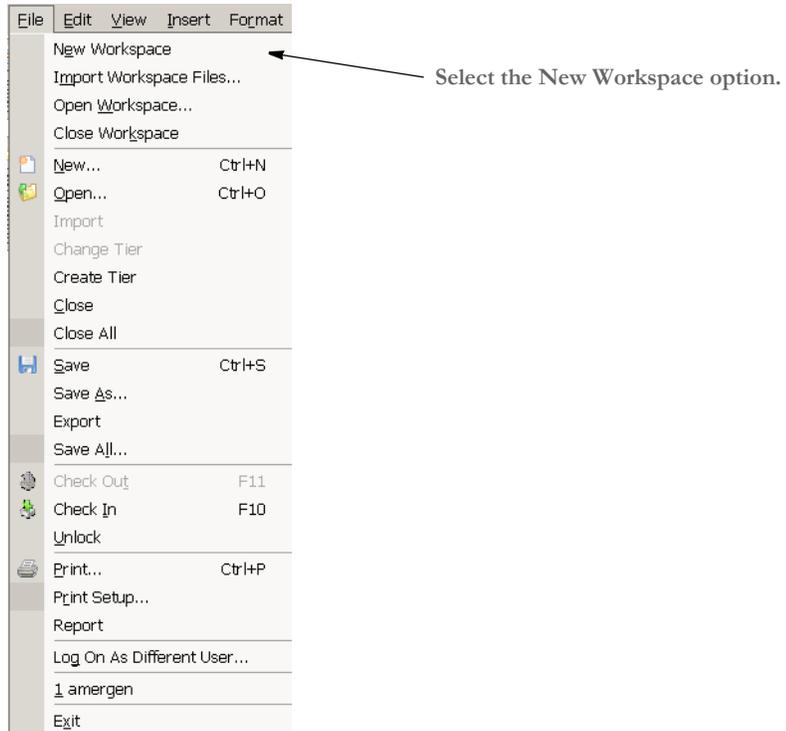
A brief summary of each option appears below.

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.

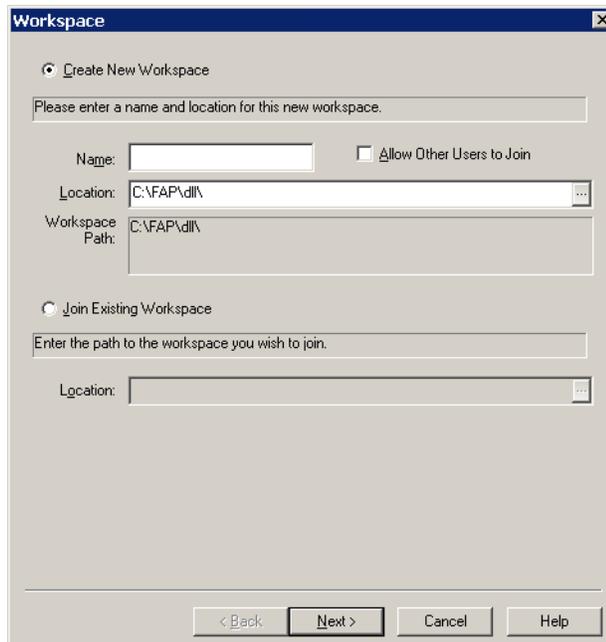
CREATING WORKSPACES

Follow these steps to create a workspace for multiple users:

- 1 Choose the File, New Workspace option.



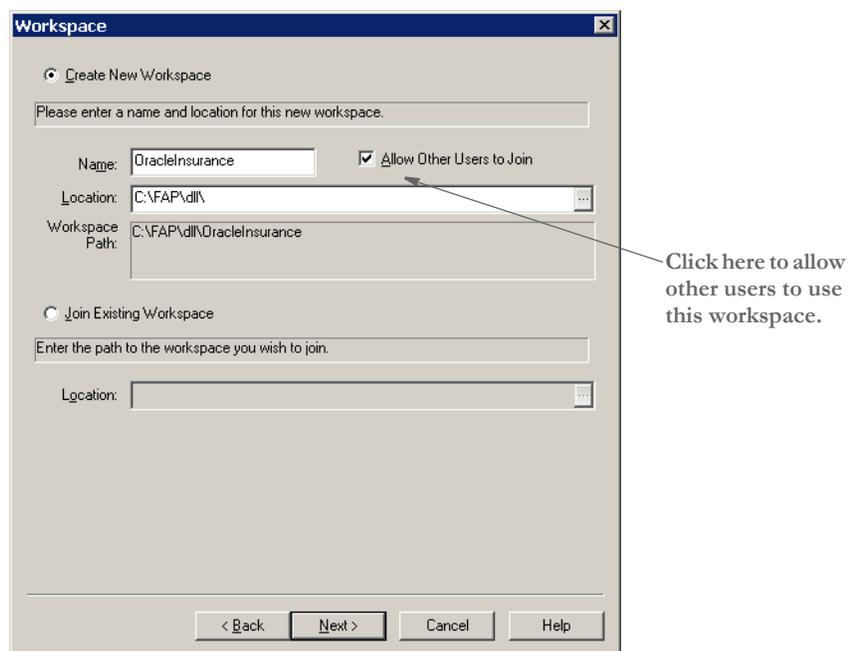
This turns on the Workspace wizard. The first page of the wizard asks whether you are creating a new workspace or joining an existing workspace.



- 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.

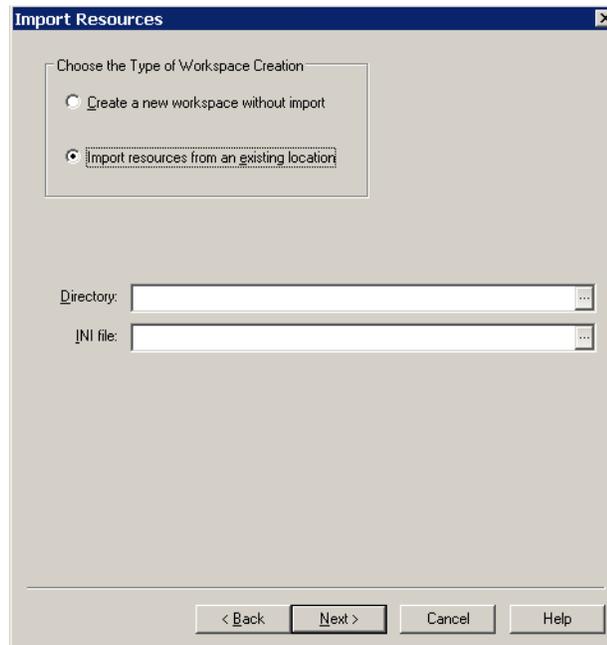
NOTE: The name you enter for the workspace cannot begin with a space or any of these characters: \ / : * ? " < > |

- 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 PSIUSER.INI) that identifies the components of the resources you want to import.



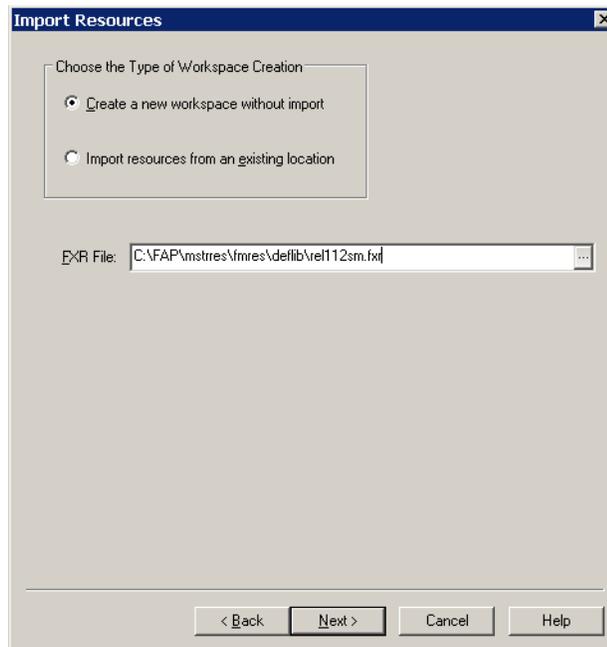
NOTE: Remember, in some setups, the working directory and the location of the INI file for a resource library setup are different. If the INI file uses relative paths, the working directory must be set correctly, or the components of the library may not be found.

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.

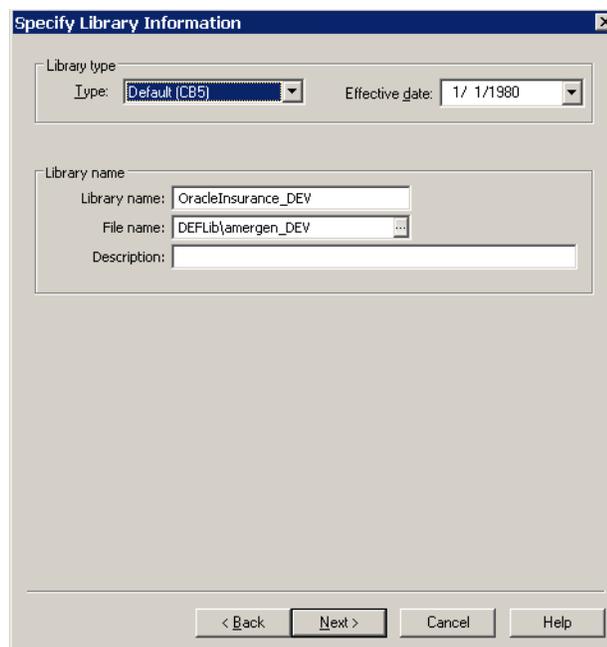
If you choose to import from a resource library, the wizard asks additional questions. If you select the Create a New Workspace Without Import option, the following field appears:



- 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 upon your library management choice, additional questions may be asked.

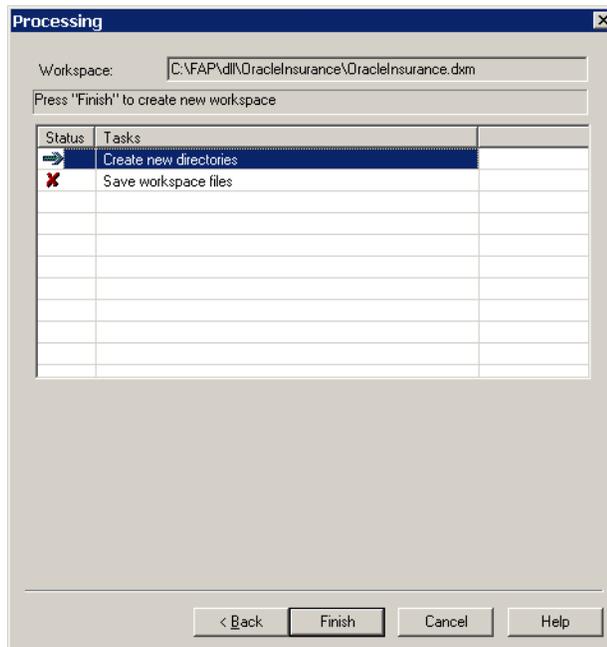
- 7 Use the Effective Date field to specify the default date for resources that may be imported or created for this new workspace. Remember that transactions are assumed to have a date-of-record — sometimes referred to as the *run date*, *effective date*, or *policy date*. Library management uses this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

Click here to display the calendar.

Use the arrows to scroll through the months. Then click the date you want.

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:



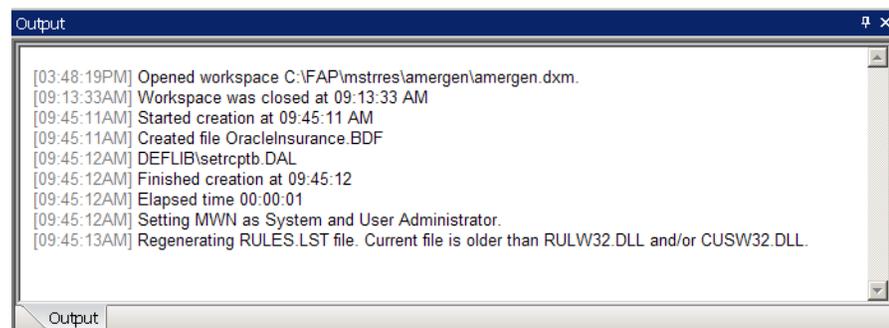
This page shows what steps the creation process has registered to activate. In this example, there are only two entries:

- 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 may appear:



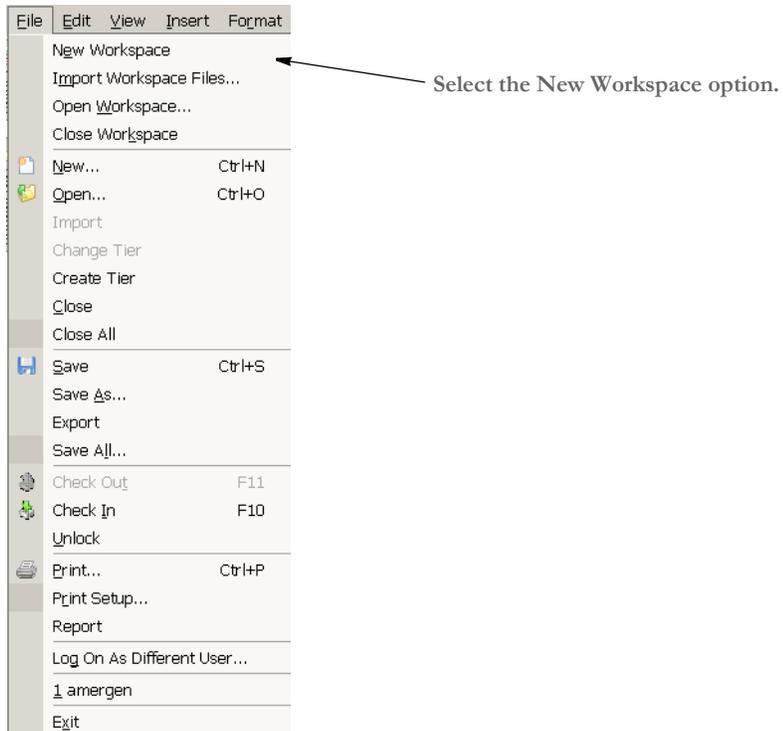
When finished, Studio opens your new workspace.

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

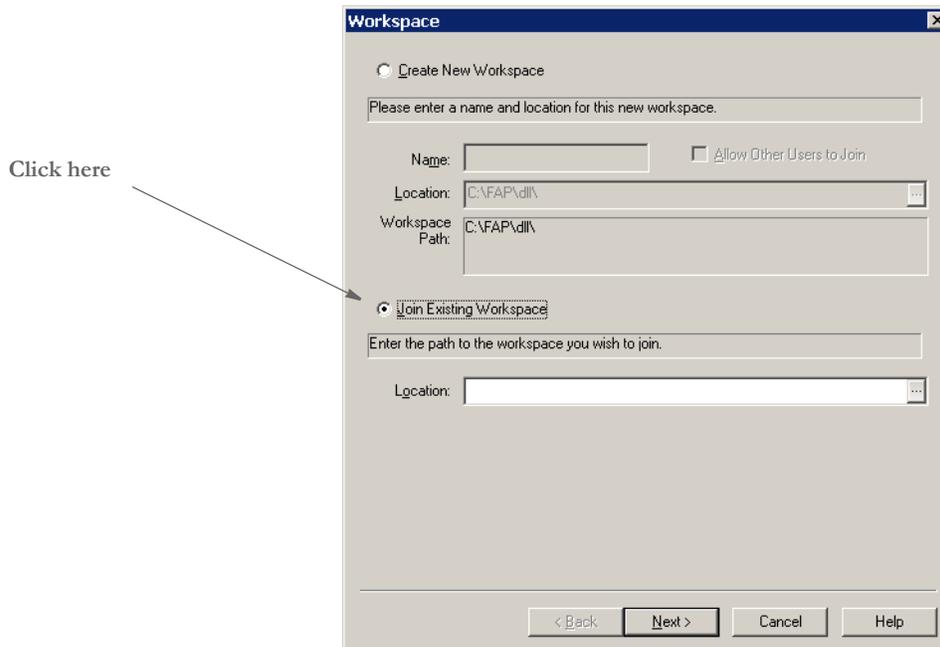
To join a workspace created for multi-user development, follow these steps:

- 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.

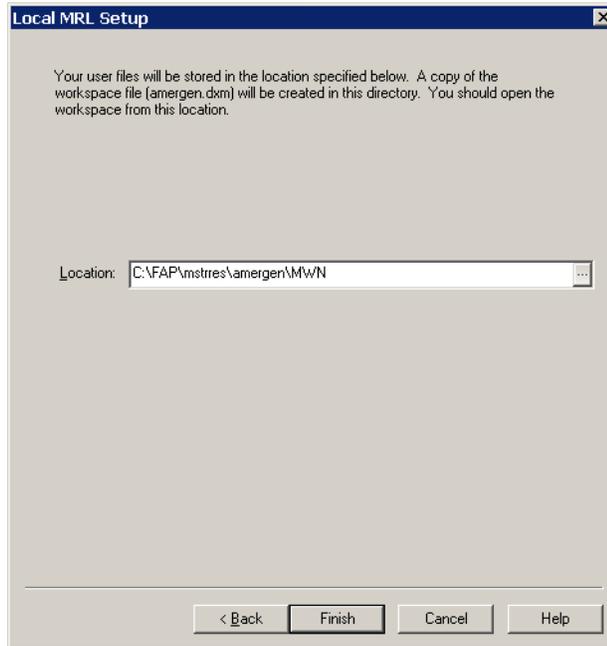
- 2 The first page of the wizard asks whether you want to create a new workspace or to join an existing workspace. Think of joining an existing workspace as you do creating a shortcut on your Windows desktop. You are creating a local reference to a shared workspace. Select the Join Existing Workspace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



- 3 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:



- 4 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...

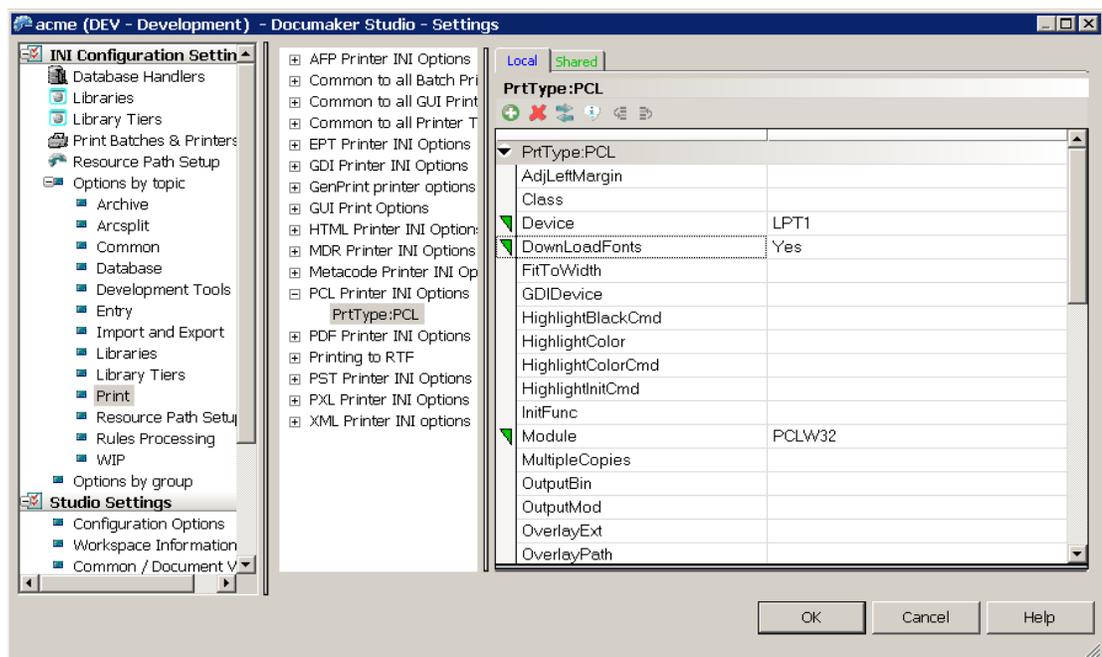
- [Working With INI Options on page 38](#)
- [Defining Studio Settings on page 46](#)
- [Managing Users on page 66](#)
- [Storing User Information in Another Database on page 78](#)
- [Adding Security at the Resource Level on page 79](#)
- [Using the LDAP Security Model on page 81](#)

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.

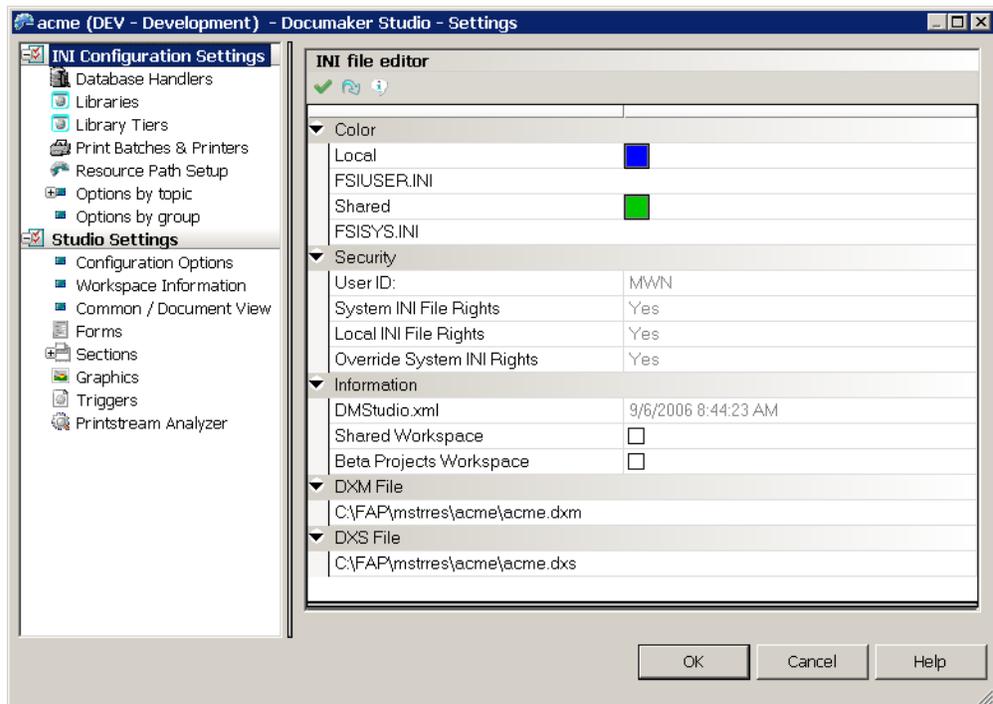


Studio loads the FSIUSER first, then finds the name and location of the FSISYS file from the ENVIRONMENT group, and loads it. Since the FSIUSER is loaded first, the options within this group usually override the FSISYS options.

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.

Click the plus sign (+) to expand the category.

Here you see the options grouped by topic.



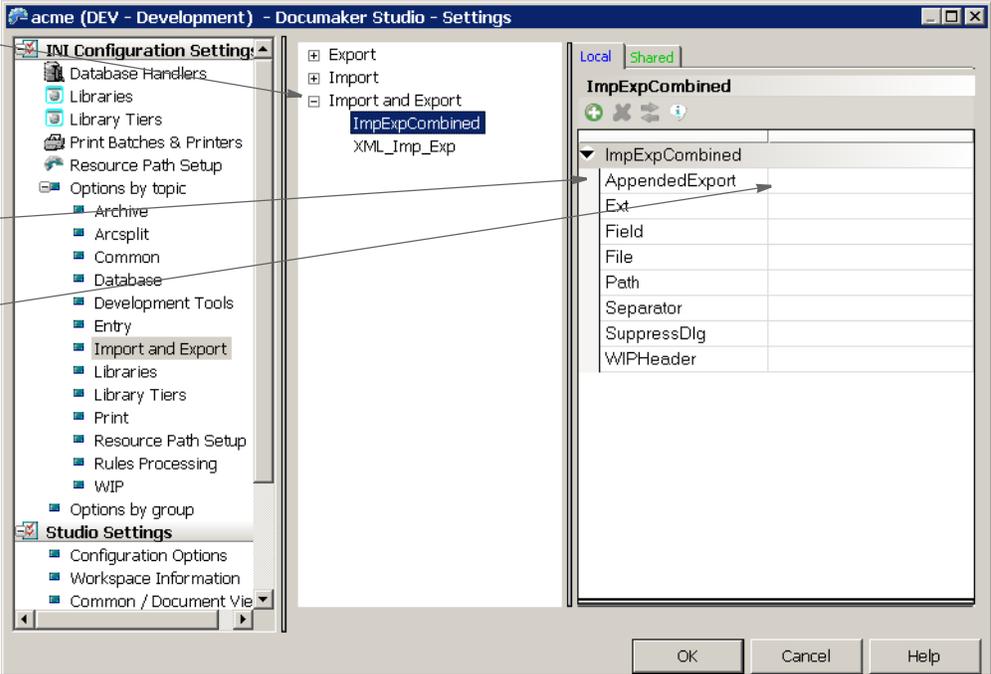
Click here to see a list of the various control groups.

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

Click the +/- symbol to show or hide the control group in this category

The name of the option appears in the left column.

Your entry goes in the right column.

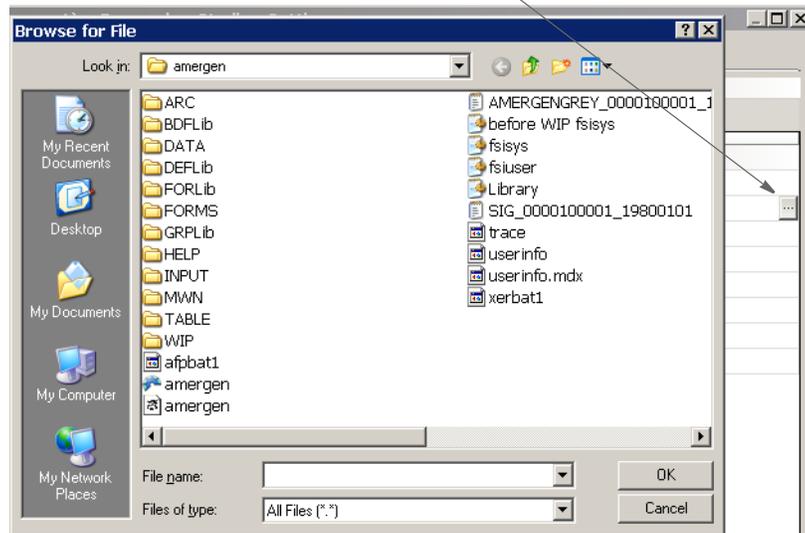


To make an entry for an INI option, click in the entry area beside the option. For some options, you simply enter text. For others, an icon appears to let you browse for a file, choose from a list of pre-set options, or to specify a parameter comprised of several elements.

Here are some examples:

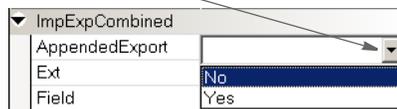
Browsing for a file

Click here to display the Browse for File window.



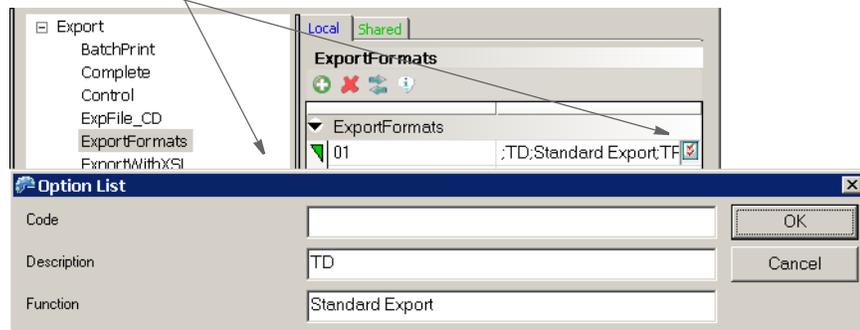
Choosing from a list of options

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:

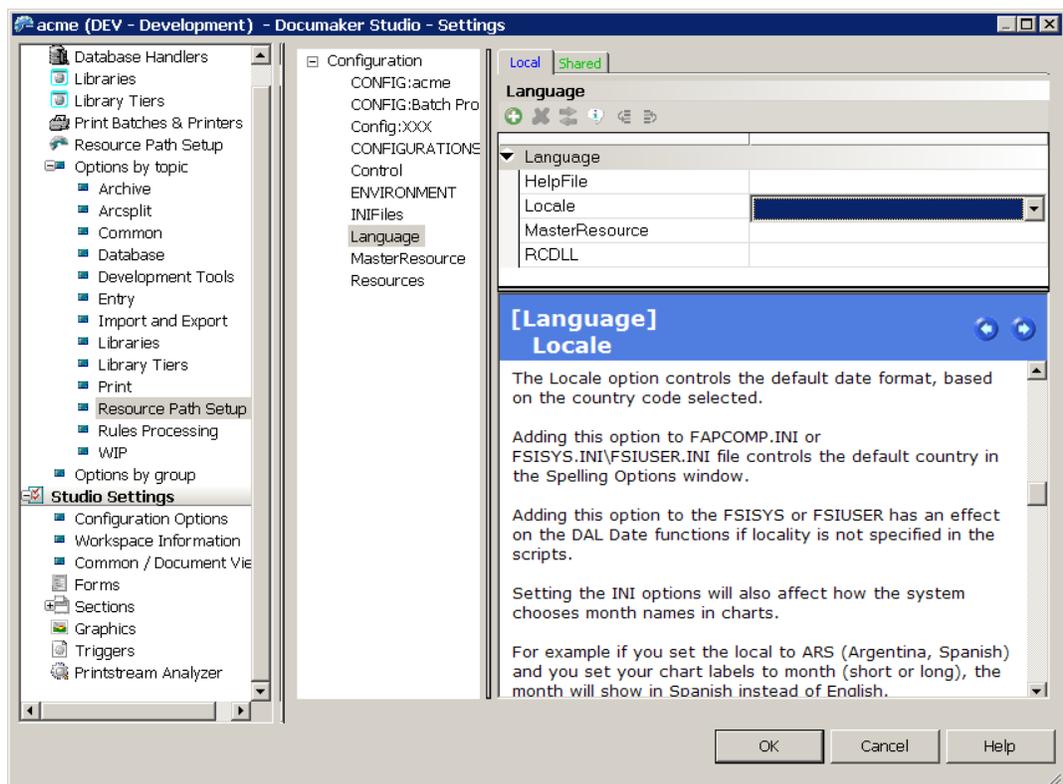


This table describes the topics:

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.

Topic	See
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.
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 GenTrn 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.
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.

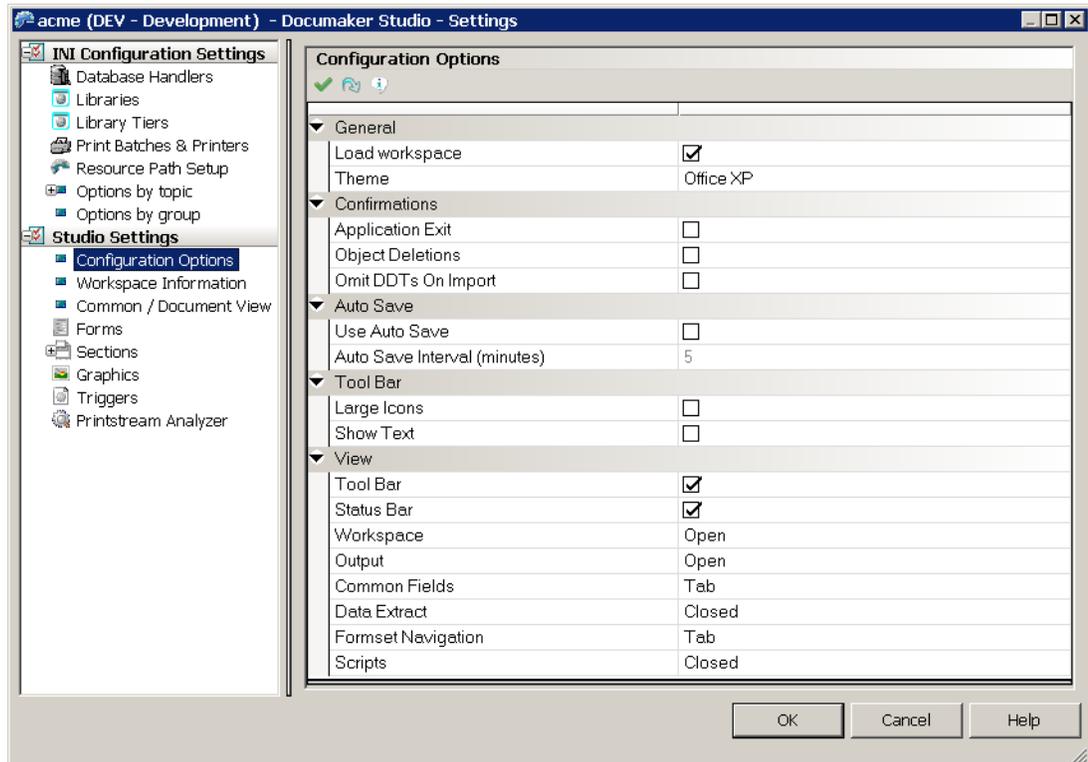
From Studio Settings you can also work with options that control the following:

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

:

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
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>
Tool Bar	
Large Icons	Check this box if you want Studio to display large icons on the tool bars.
Show Text	Check this box if you want Studio to display the names of the icons.
View	
Tool Bar	Check this box if you want Studio to display the tool bar.
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>

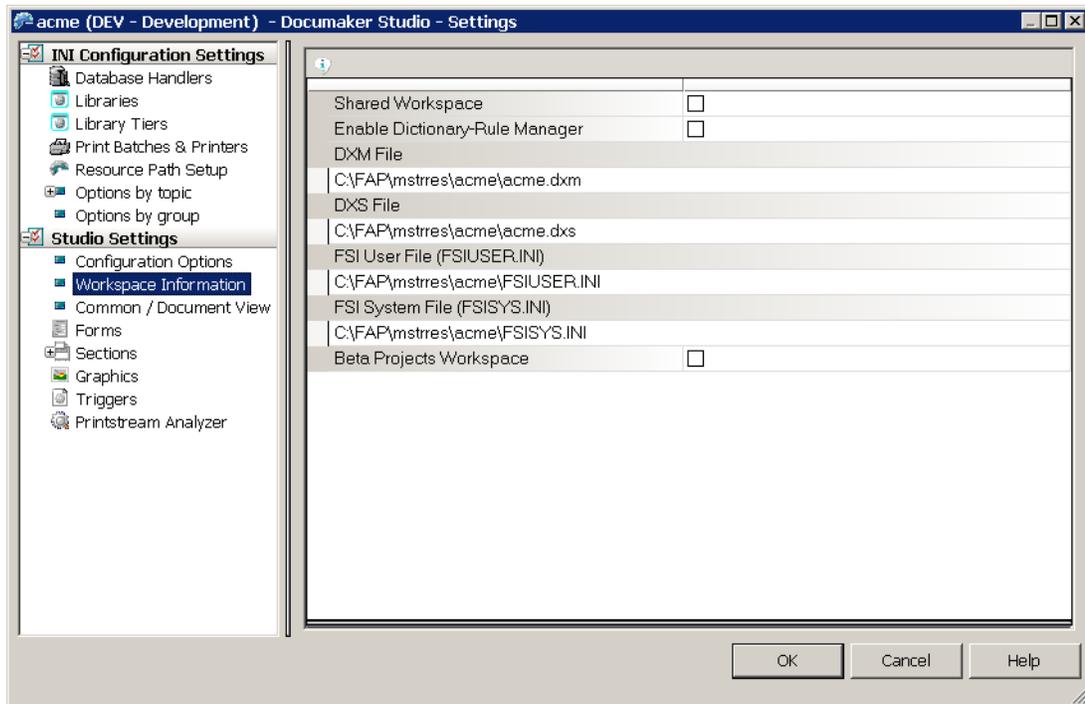
Option	Description
Common Fields	Choose the window mode. You can choose from these options: <ul style="list-style-type: none"> • Closed • Open • Tab
Data Extract	Choose the window mode. You can choose from these options: <ul style="list-style-type: none"> • Closed • Open • Tab
Formset Navigation	Choose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

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:

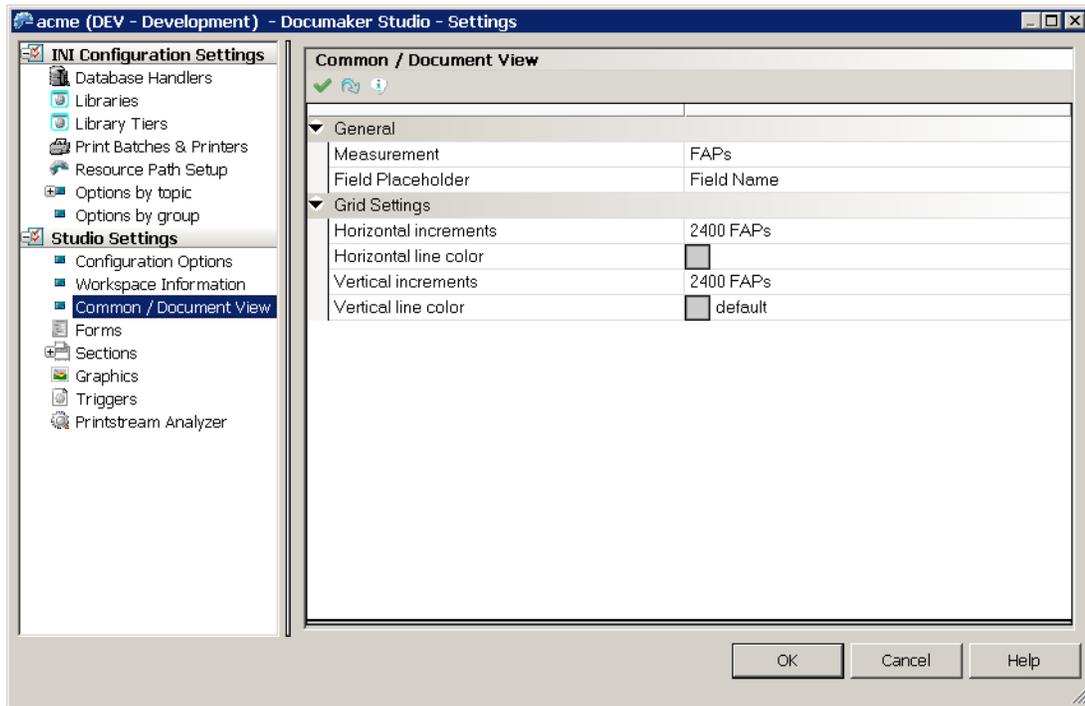
Option	Description
Shared Workspace	Studio checks this box if the current workspace is a shared workspace.
Enable Dictionary-Rule Manager	Studio checks this box if this workspace uses the Dictionary Rule manager. If you use a master.ddt file, be sure to check this option.
DXM File	Here Studio shows you 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.
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.

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:

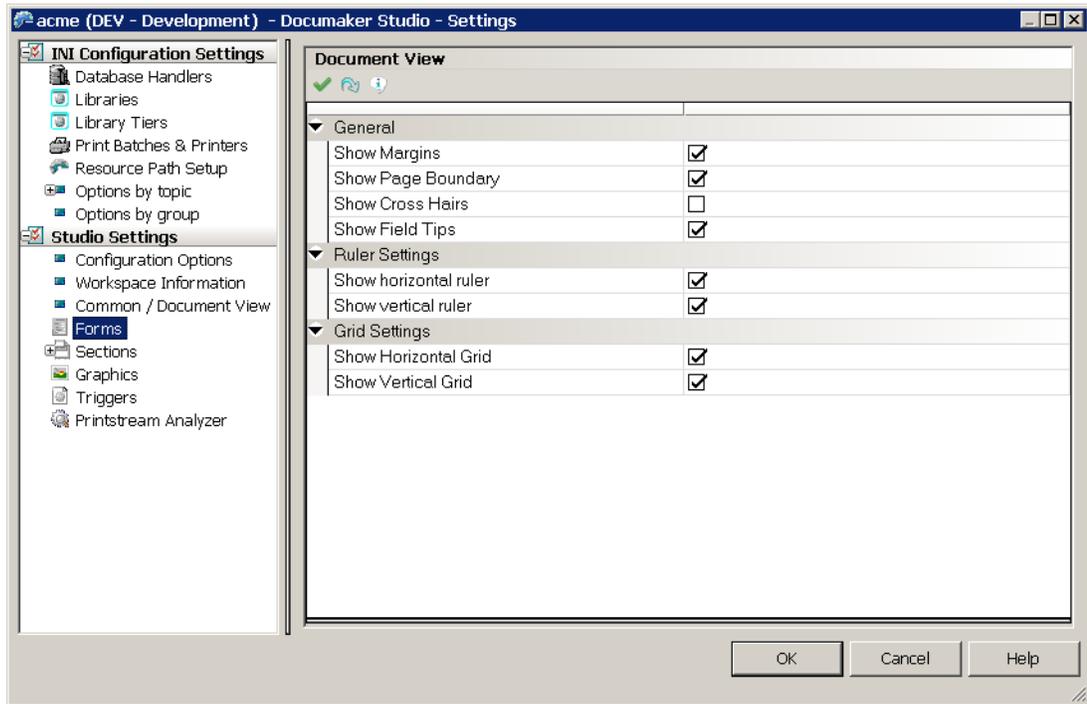
Option	Description
General	
Measurement	Select unit of measurement in which display will be made. You can choose from centimeters, FAP units (2400 per inch), inches, picas, or points.
Field Placeholder	Choose the style of placeholder you want to use to indicate entry fields. You can choose from Template 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.

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 Margins	Check this box to have Studio draw a red line on the page to indicate the page margins.
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	

Option	Description
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.

WORKING WITH SECTION OPTIONS

The section options fall into two categories:

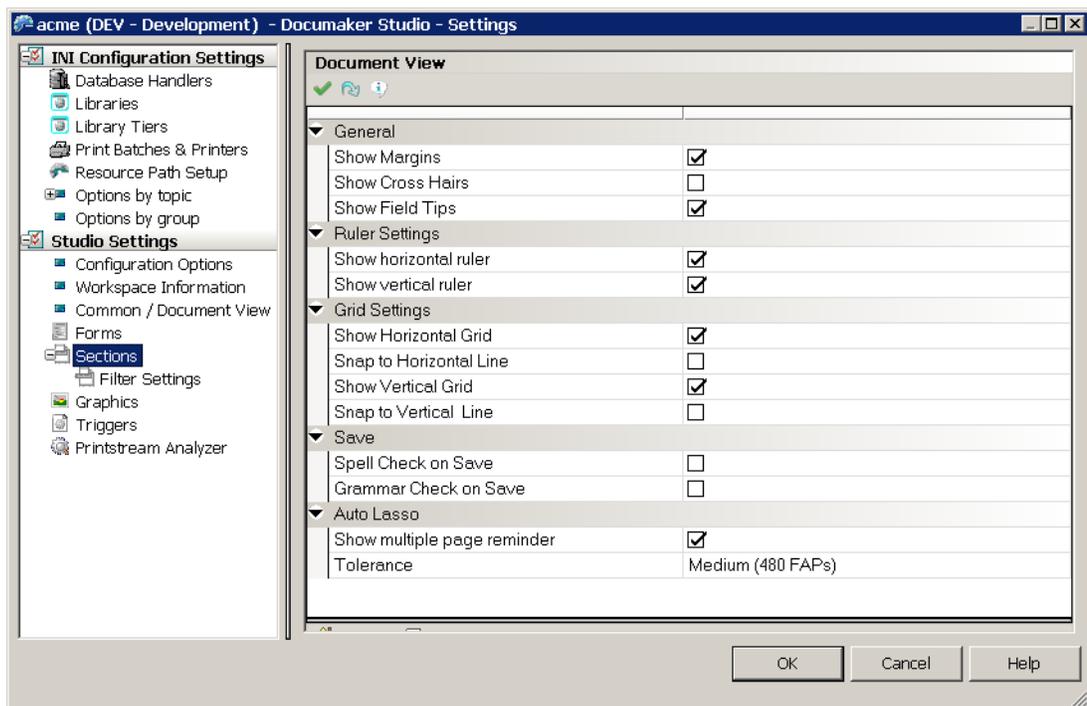
- Document view options
- Filter settings

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:



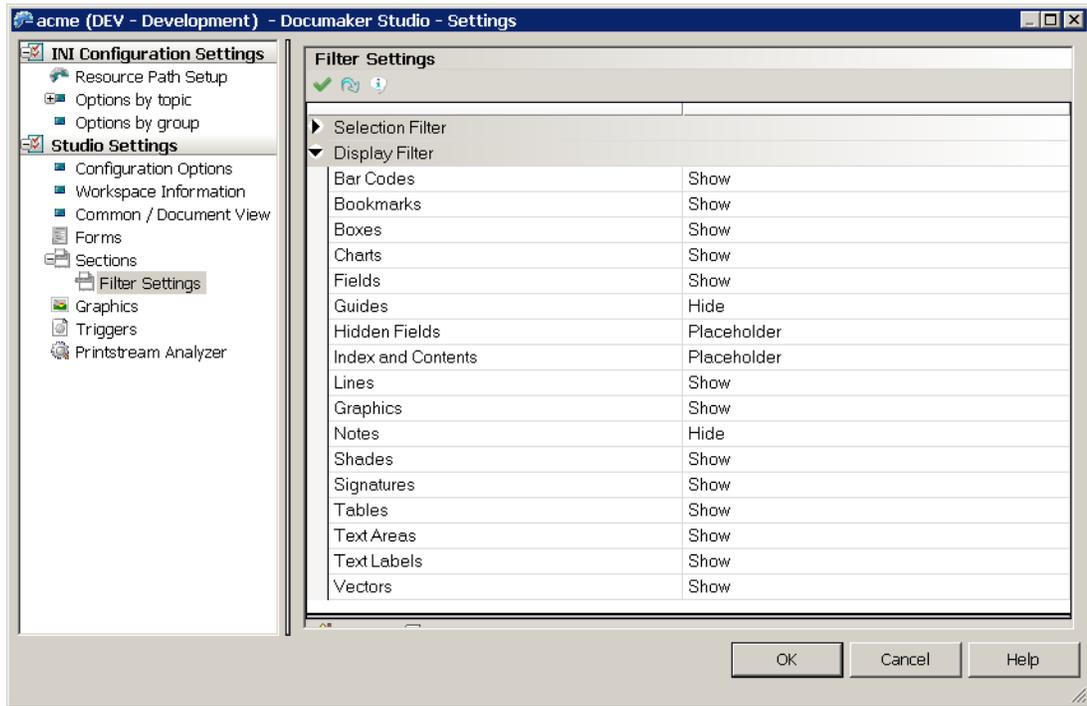
This table explains your options:

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.

Option	Description
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 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.

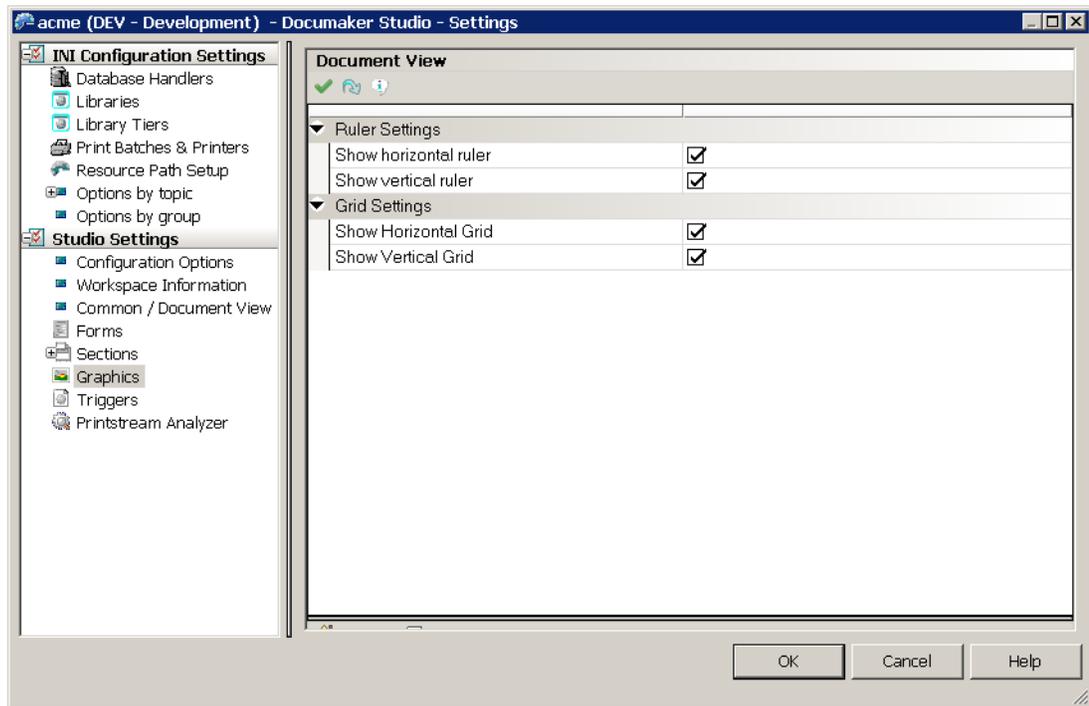
In the example shown previously, Hidden Fields, Indexes, and Contents would be shown with a placeholder while Guides and Notes would be hidden. Studio would show the other objects.

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.

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.

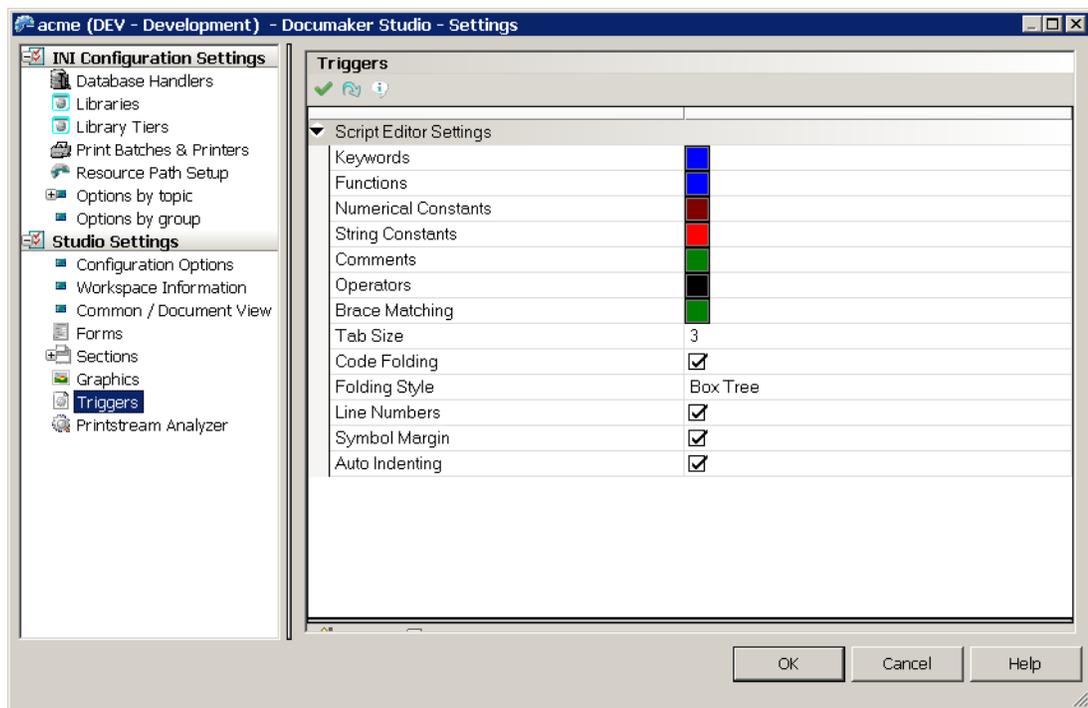
Option	Description
Apply Changes	Lets you apply your changes.
Default Settings	Restores the default settings.
Toggle Information Pane	Shows or hides the information pane. This pane provides information about the INI option currently highlighted.

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.

Option	Description
Comments	Click the icon in this field to display the Color Selection screen and select the color you want to use.
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.

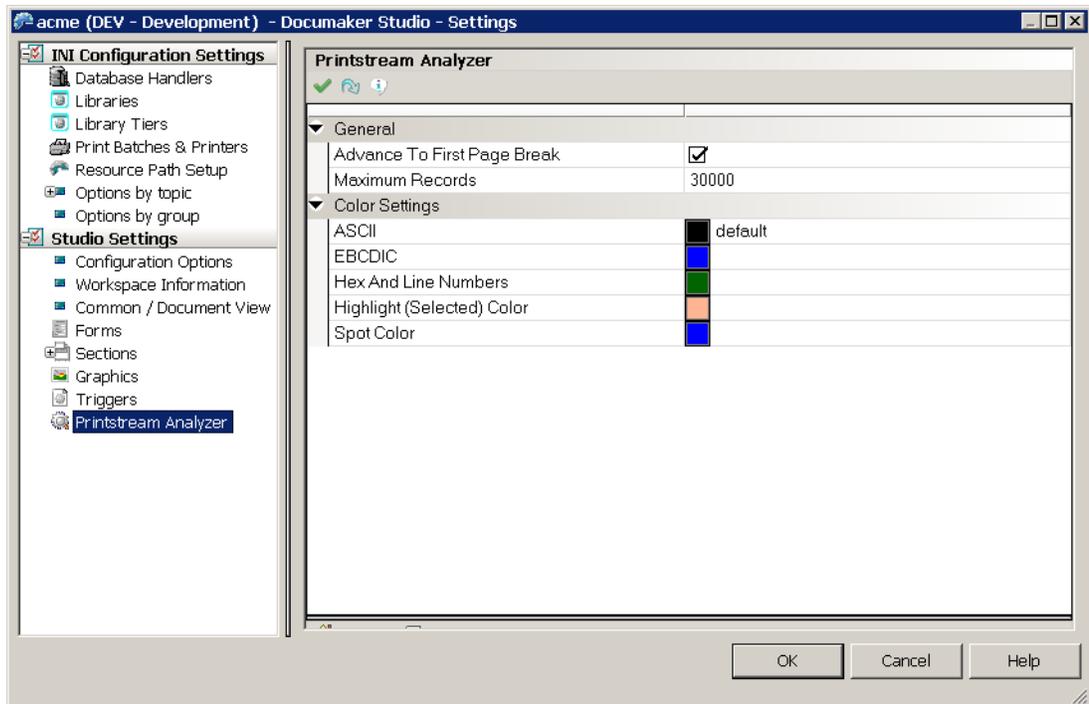
When you right-click in this window you can choose from these options:

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.

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.

Option	Description
Spot Color	The color used for spot color text and lines in the Display pane

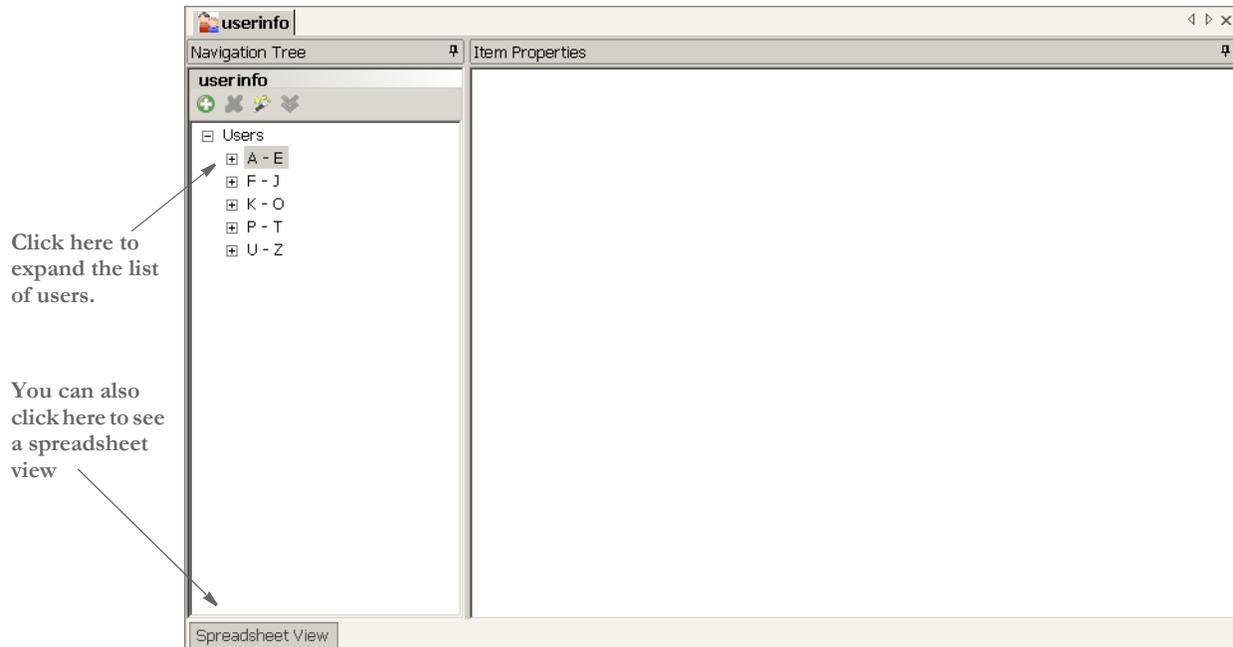
When you right-click in this window you can choose from these options:

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.

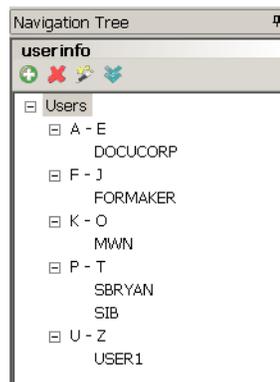
MANAGING USERS

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		Change Password	0	Yes		Security	
FORMAKER		Change Password	0	No		Security	
MWN		Change Password	9	No		Security	
SBRYAN		Change Password	9	No		Security	
SIB		Change Password	0	No		Security	
USER1		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.

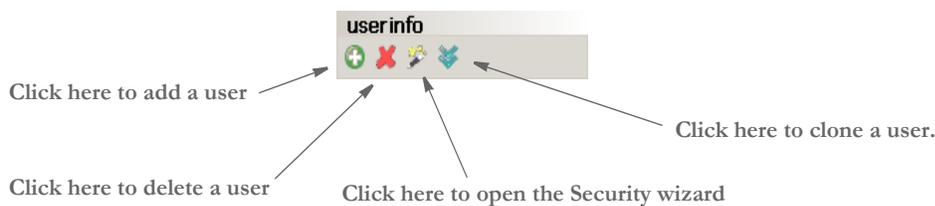
Once you select a user, Studio shows you the basic information for that user.

Item Properties	
User ID:	DOCUCORP
User Name:	
Message:	
Report To:	
Workstation Rights:	0
In Use:	Y
Password:	*****
Verify Password:	*****
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.



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.

Field	Description
Security	<p>Click the Configure button to open the Security wizard. From the Security wizard you can...</p> <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. <p>For more information, see Setting Rights Using the Security Wizard on page 70.</p>

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.

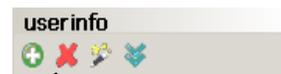
For example, performing conversions from normalized Metacode or AFP to a section (FAP) can alter some INI settings. Those settings are not saved, however, unless the user has the appropriate security rights.

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.

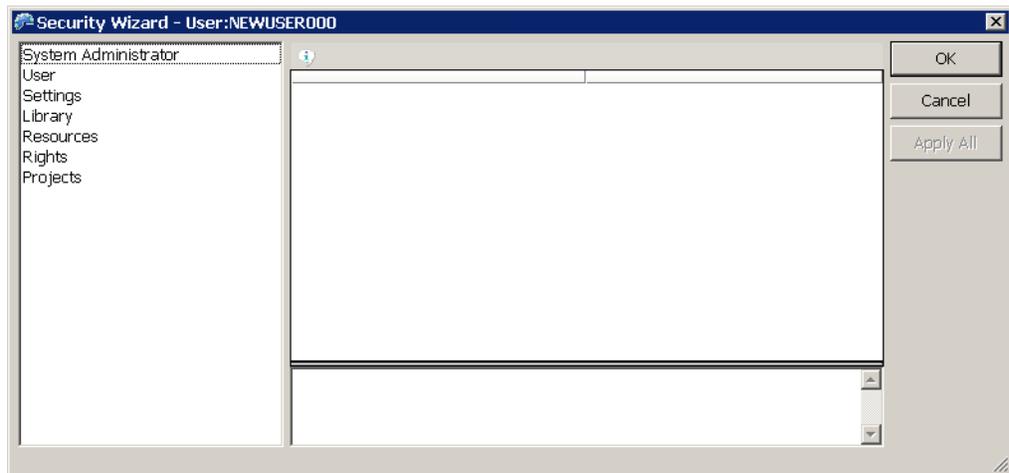
SETTING RIGHTS USING THE SECURITY WIZARD

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 System Administrator rights on page 71 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 User rights on page 71 for information about these settings.

Area	Description
Settings	Grants access to INI options. See Settings rights on page 72 for information about these settings.
Library	Controls library access and the actions the user can perform within the library. See Library rights on page 72 for information about these settings.
Resources	Limits access to the various resource managers, such as Forms, Section, and Trigger manager. See Resource rights on page 73 for information about these settings.
Rights	Defines creation rights for fields, triggers, templates, or all resources. See Rights on page 75 for information about these settings.
Projects	Assigns a project role, such as an administrator, developer, tester, reviewer, and so on. See Project rights on page 76 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.

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

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	<p>Check this box if you want to make this user a library administrator. Library administrators have access to all library functionality.</p>
Limited Property Modifications	<p>Check this box if you want this user to modify the description, mode, status, class and project attributes of library resources.</p>
Perform Promotions	<p>Check this box if you want this user to promote resources within the library.</p>
Delete Library Entries	<p>Check this box if you want this user to delete library entries.</p>

Field	Description
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 Adding Security at the Resource Level on page 79 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.
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"/>

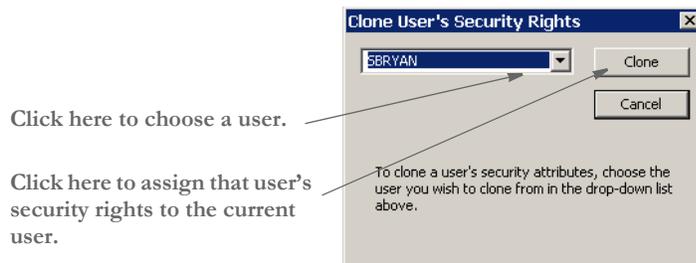
Field	Description
Project Administrator	Check this box to indicate that this user is a project administrator and can create, modify, and delete projects and perform other project management roles.
Project Developer	Check this box to indicate that this user is a project developer.
Project Tester	Check 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:

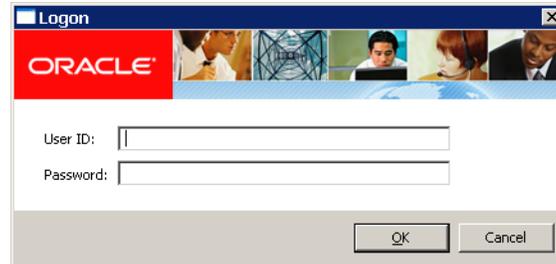


When you click the clone tool, this window appears:



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 81](#).

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.

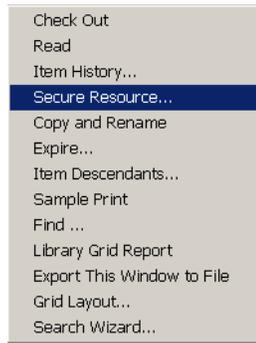
Here is another example within the LDAP security model. Suppose you have two groups of developers who share the same library. You want them to share the library because all of the business runs from a central location. You may, however, want to make sure one group does not accidentally check out or change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

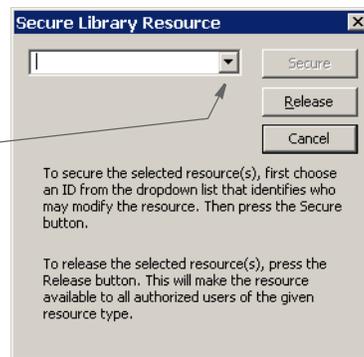
After you highlight the resources you want to secure, right click and choose this option.



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: Administrator@Oracle.com
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>

Option	Description
LDAP.DEREF.LINK	(Optional) Enter Yes or No to indicate whether or not to remove reference links to other nodes during a search. The default is No.
LDAP.VERSION	(Optional) An integer value that indicates the LDAP protocol version to use. You can choose from: <ul style="list-style-type: none"> • 2 - Version 2 • 3 - Version 3 The default is three (3).
LDAP.SEARCH.LEVEL	(Optional) An integer value that indicates the search level. You can choose from: <ul style="list-style-type: none"> • 1 - User type objects • 2 - Group type objects • 3 - Any objects The default is one (1), user type objects.
LDAP.DN.IDENTIFIER	(Optional) The value for this option is used these ways: <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 data-bbox="894 1444 1406 1486">(&((objectClass=groupOfNames)(member=*)))</pre> If a value of <pre data-bbox="894 1541 1357 1562">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 data-bbox="894 1644 1406 1717">(&((objectClass=groupOfNames)(member=CN=Administrator, CN=Users,DC=PDDC,DC=DOCUCORP,DC=COM))))</pre> The default is <i>cn</i>.

Option	Description
LDAP.RDNS	<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:</p> <pre data-bbox="899 699 1403 743">(&((objectClass=groupOfNames)(member=*)))</pre> <p>In a case were a value of</p> <pre data-bbox="899 800 1344 821">CN=Users,DC=PDDC,DC=DOCUCORP,DC=COM</pre> <p>is specified for this option and LDAP.DN.IDENTIFIER contains a value of CN, the search filter generated would look like this:</p> <pre data-bbox="899 926 1403 999">(&((objectClass=groupOfNames)(member=CN= Administrator, CN=Users,DC=PDDC,DC=DOCUCORP,DC=COM)))</pre>
LDAP.USE.SSL	<p>(Optional) Enter Yes to enable encrypted communication through an SSL channel.</p> <p>For SSL connections to work, the LDAP server must be configured for SSL with a certificate from a trusted certification authority.</p> <p>This configuration is vendor specific — please consult your vendor documentation.</p>
LDAP.DEBUG	<p>(Optional) Enter Yes to log debugging information to a trace file.</p>

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 = Y
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 a Business Unit on page 91](#)
- [Storing Recipient Information on page 96](#)
- [Defining Categories on page 97](#)
- [Defining Transaction Types on page 98](#)
- [Using Regional Date Processing on page 99](#)
- [Generating Reports on page 108](#)

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).

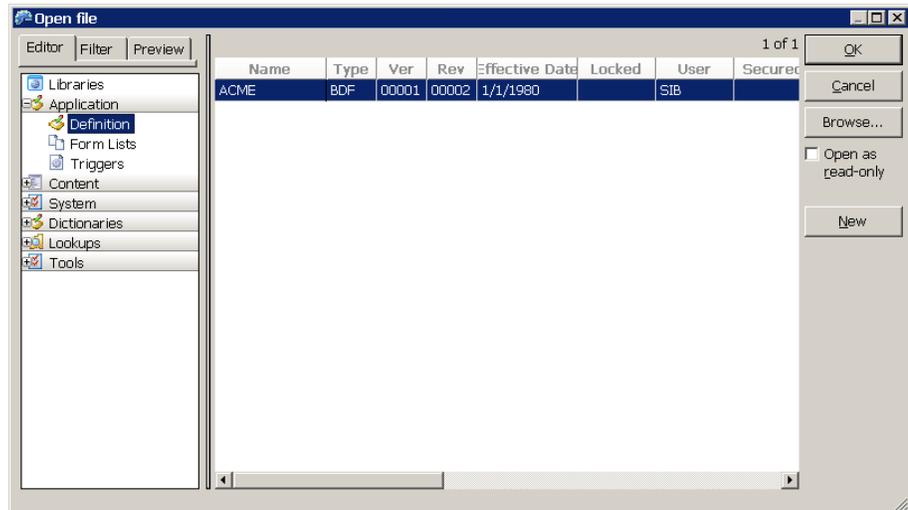
Other information stored in the BDF file includes the list of recipients, form categories (if used), and transaction type information. In addition, the BDF file identifies the primary extract dictionary (XDD) file (if used), as well as the font cross-reference (FXR) file preferred by this setup. Here is an example of a BDF file.

```
<DOCUMENT TYPE="RPWIP" VERSION="11.0">
<DOCSET NAME="SAMPLE_0000100001_19800101">
<FXRFILE NAME="rel1103sm" />
<XDDFILE NAME="Symbol" />
<RECIPIENT NAME="Client" COPYCOUNT="0" CODE="001" SEQUENCE="1">
<DESCRIPTION>Client Copy</DESCRIPTION>
</RECIPIENT>
<RECIPIENT NAME="Office" COPYCOUNT="0" CODE="002" SEQUENCE="2">
<DESCRIPTION>Office Copy</DESCRIPTION>
</RECIPIENT>
<GROUP NAME="Sample_Business" NAME1="Sample" NAME2="Business" />
<TRANSACTION TRNCODE="NB" DESCRIPTION="New Business"
DLLFUNC="TRNW32-&gt;TRNNEW" />
<TRANSACTION TRNCODE="EN" DESCRIPTION="Endorsement"
DLLFUNC="TRNW32-&gt;TRNEndorse" />
<TRANSACTION TRNCODE="RN" DESCRIPTION="Renewal" DLLFUNC="TRNW32-
&gt;TRNRenew" />
<TRANSACTION TRNCODE="QU" DESCRIPTION="Quote" DLLFUNC="TRNW32-
&gt;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. Also, there is no category information in this example.

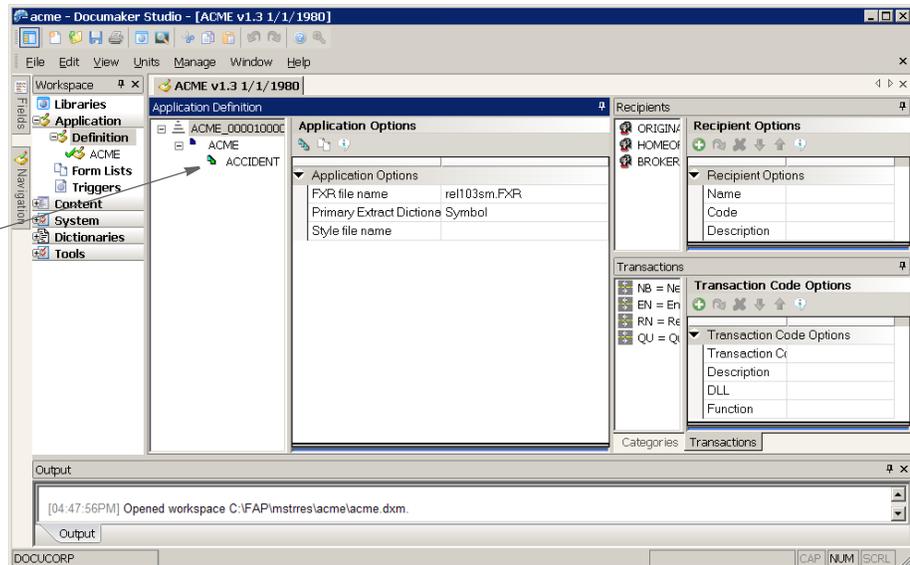
DEFINING A BUSINESS UNIT

To define a line of business for a set of resources, you first check out the BDF file and then make the necessary changes 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 business unit, the Application Definition view appears:

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

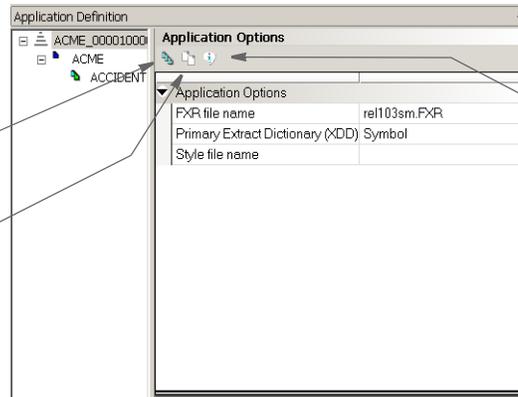


Notice the toolbar above the application options. You will see similar toolbars throughout Studio.

Here you can add unit names, look up a unit file, or toggle the information pane.

Click here to add a unit name.

Click here to look up a unit file.

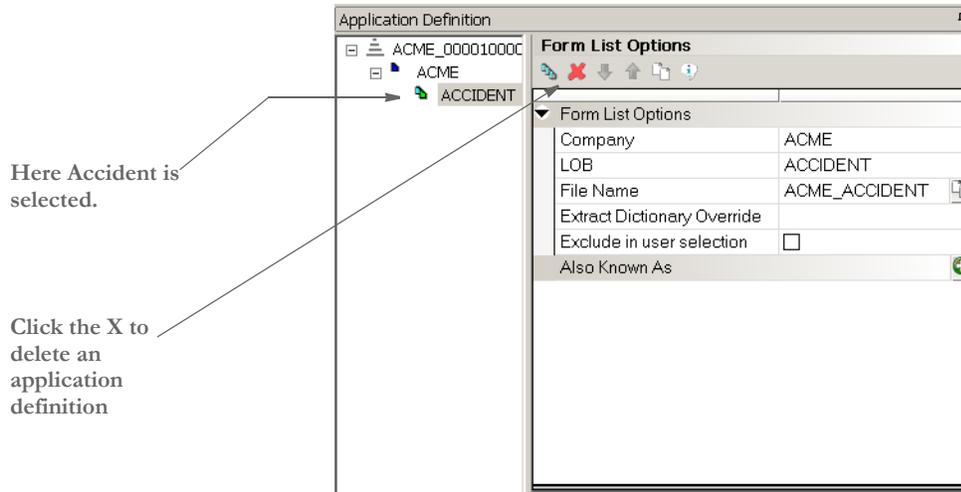


Click here to display or hide the information pane.

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 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 weight (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 will use 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>
Style File Name	<p>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, line bar codes, graphics, signatures, bookmarks, notes, indexes, and charts and associate those settings with named styles.</p> <p>You can click the Browse icon to search for the file you want.</p>

If you click on a specific application definition, the view changes to show you the appropriate fields for that definition. Here is an example:



The toolbar changes to provide some additional options.

To	Click
Delete a Form List definition	
Change the order of an item (if the item can be ordered). 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 groups defined within the BDF file is the same order used when form selection is shown in 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 groups that are most often used are defined closer to the top of the tree.	

To set up a line of business, you must make define the Key values (Company and Line of business), the name of the GRP files that contains the list of forms, and any applicable extract dicutionary overrides.

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.

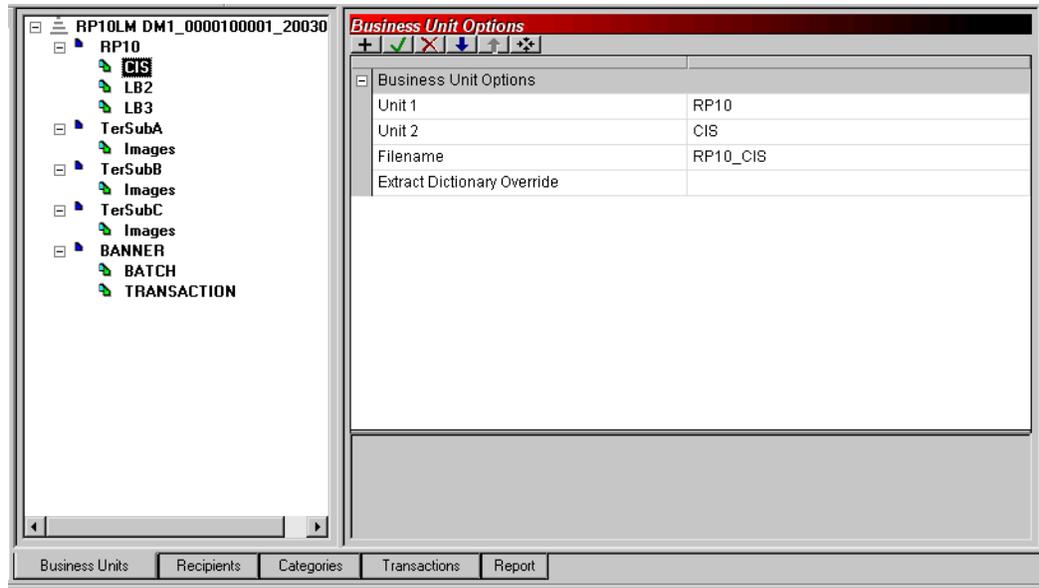
Field	Description
Extract Dictionary Override	<p>Enter the name of the XDD file you want to use for this line of business instead of the primary XDD file defined at the BDF or root level. Leave this field blank if you want to use the primary XDD file. You can browse through the library to select an XDD file.</p> <p>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:</p> <pre>< MasterResource > DictionaryFile =</pre>
Exclude in User Selection	<p>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.</p> <p>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.</p> <p>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.</p> <p>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.</p>
Also Known As	<p>Click in this field to set up the automatic mapping of group alias names into the Key1Table and Key2Table control groups.</p> <p>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.</p>

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:

```
< 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.

If you click on an item in the business unit group definition, Studio adds the Extract Dictionary Override field.



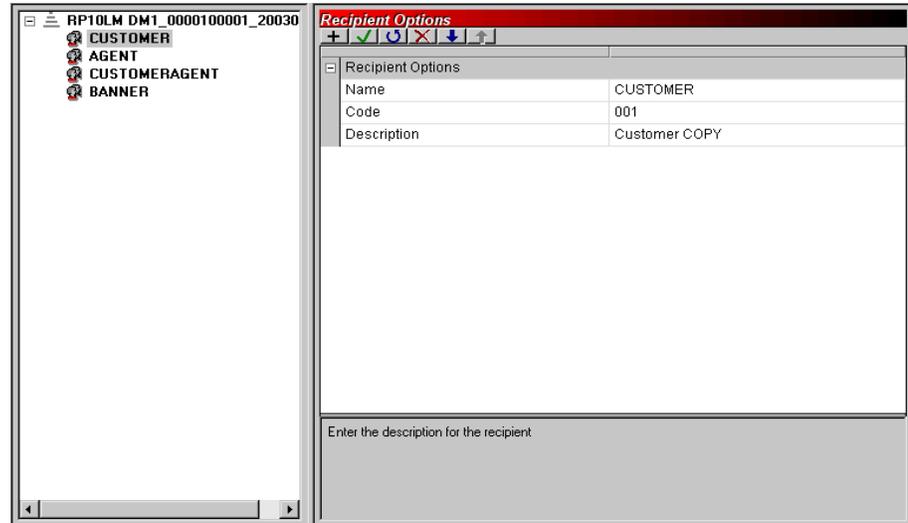
Use the Extract Dictionary Override field to define a different XDD than the primary XDD defined at the BDF (root) level. You can leave the name blank, if you want this business unit to use the primary XDD definition. Otherwise, you should enter the appropriate name or browse the library for the appropriate XDD file to use.

During batch processing, when the system maps fields it checks the associated group level for a defined symbol (XDD) file. If one is not defined, it queries the BDF (root) level for the primary XDD. If one is not defined there, it chooses the one specified in this INI option:

```
< MasterResource >
DictionaryFile =
```

STORING RECIPIENT INFORMATION

Click the Recipients tab to manage the list of recipients for your documents. Here is an example:



Here you define the recipient type name, such as Insured or Customer; the sorting code, which is used by batch implementations; and a brief description that will display in Send Copy To fields defined in your document when you print that document for that recipient.

NOTE: There are two parts to the recipient definition in the Recip_Names control group. The first part is a code and the second is a description. The code is only used in batch processing (the GenData program) and is a unique value written to the batch files to the system will know which recipient to queue at print time.

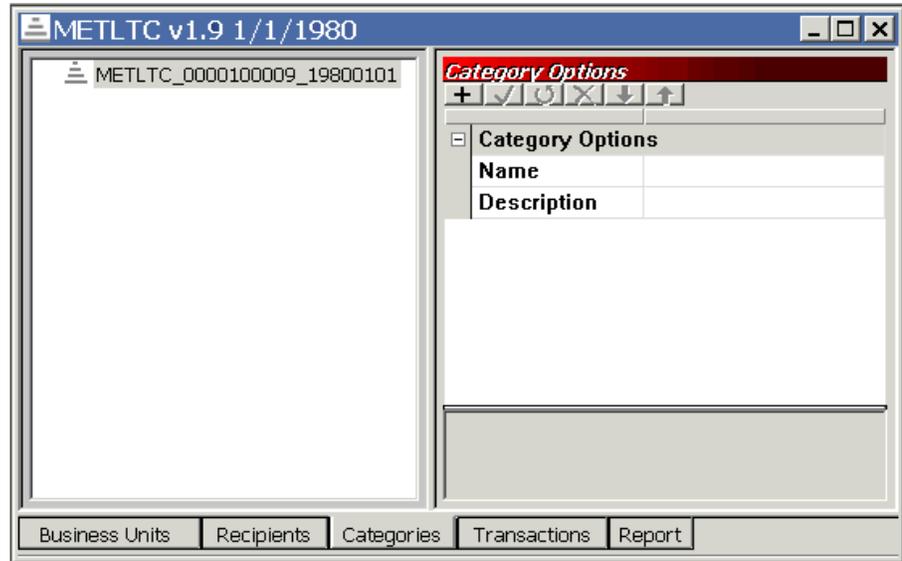
The second part is the description field. This is used any time a SendCopyTo field is printed. This is where the text like *Insured's Copy* comes from when you print. This is used by both entry systems like Documaker Workstation or iDocumaker and the GenPrint program.

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

Field	Description
Name	Enter the name of the recipient you want to add.
Code	Assign a code to the recipient.
Description	Enter a description of the recipient.

DEFINING CATEGORIES

Click the Categories tab to define form categories.



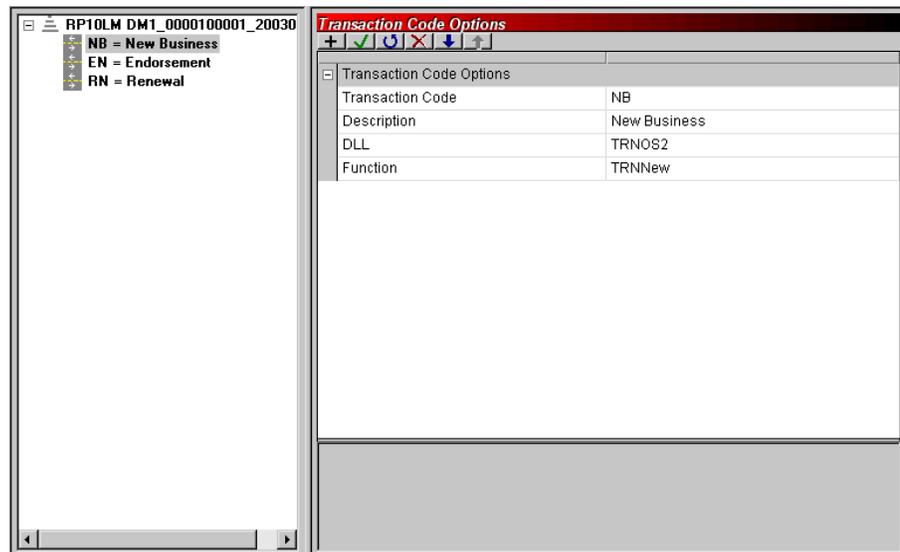
Form categories are optional. If you define them, the form categories appear when you edit a group (GRP) file. The category definitions can help you organize large sets of forms by letting you associate similar forms to make the list more manageable. Categories are not the same as the higher level key groups (Key1 and Key2) and are not used during form triggering or selection.

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

DEFINING TRANSACTION TYPES

Click the Transactions tab to define transaction types and optionally assign an Entry Workstation-affiliated hook function associated with form selection.

NOTE: Batch implementations do not use the associated hook function.



Transaction codes are sometimes used in batch processing as one level of form filtering. Transaction codes are also used in Entry/WIP situations to identify the type of transaction being created.

Also in Entry, the DLL and Function declarations define internal or external hook functionality that must be executed on that transaction type to validate the list of forms selected.

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.

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.

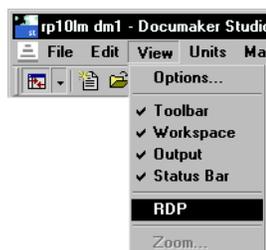
This necessity to only activate the use of a form in a given region after a specific date complicates the creation of trigger conditions. Not only do you have to consider the typical transaction information that would cause you to include the form, you also have to calculate the various details to comply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

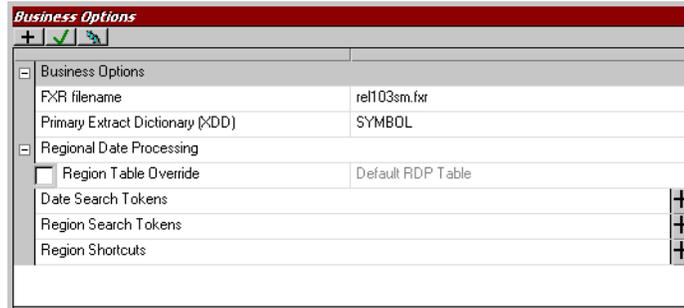
SETTING UP REGIONAL DATE PROCESSING

To begin using regional date processing (RDP) rules you must activate them in the Business Definition Manager for your Business Definition File (BDF). To activate, you first check out the appropriate BDF and then select the View, RDP option from the BDF Manager menu.



Once activated, new items are added to the Business Options Properties window. These options let you define your default Region Table, Date Search Tokens, Region Search Tokens, and Region Shortcuts.

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



Studio provides a basic Region Table that contains the fifty U.S. states and the District of Columbia. This table is not required for run-time processing. It is included to help you select regions when setting up your RDP rules. If you need to override the default and use an alternative region table, do this by clicking the Region Table Override field and entering the name and location of your region table. You can use the Browse button to look up a region table:



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*.

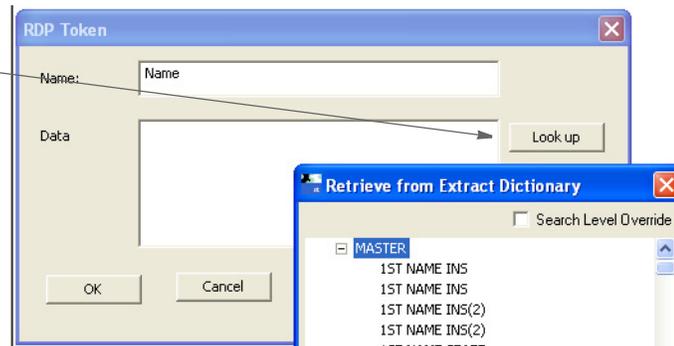
You can edit the content of a region table using Studio’s Batch Table Manager.

The Date Search Tokens section is where you identify the token names that represent the date searches you will use within an RDP rule.



To add a date search token (DST), click the Plus (+) button. The RDP Token window appears so you can name the DST and enter its search information.

Click here to look up a date search token



Click the Look Up button to display the Retrieve from Extract Dictionary window from which you can select a symbolic lookup from the Extract Dictionary (XDD). You can also use alternate search methods like =GVM or =DAL and specify the appropriate values where necessary.

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.

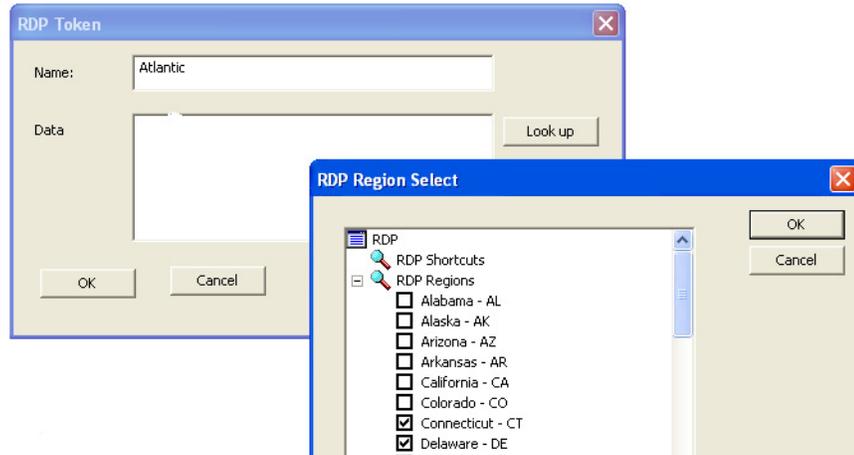
When adding a new Region Search Token, you can use the Look Up button to select from your available fields defined in the Extract Dictionary (XDD) for your setup.

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.

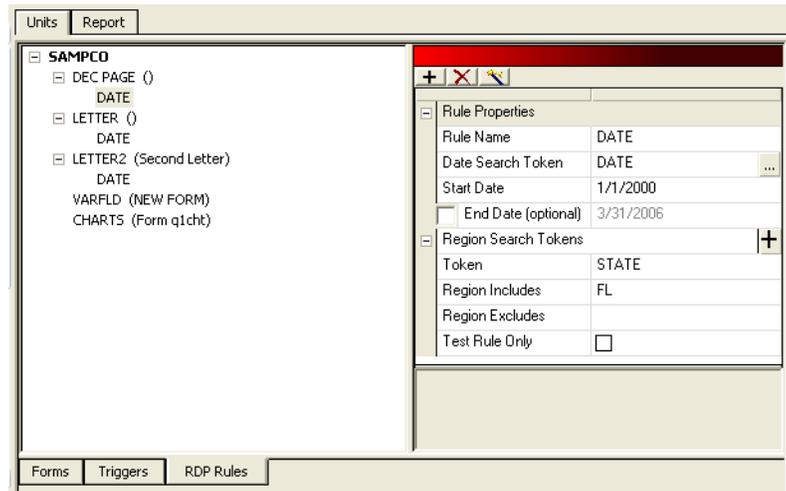
When using the Lookup selection window, simply check, or uncheck, the items you want to include or exclude from this Shortcut definition.

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.

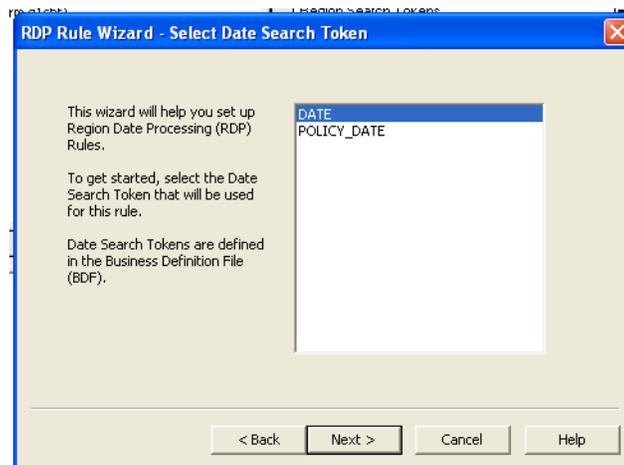
Regional date processing rules are separate from normal trigger rules you might assign to forms. RDP rules operate as a *filter* that occurs before normal form triggering commences. Thus, RDP rules are specified separate to any necessary trigger condition you also would use to identify when you want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



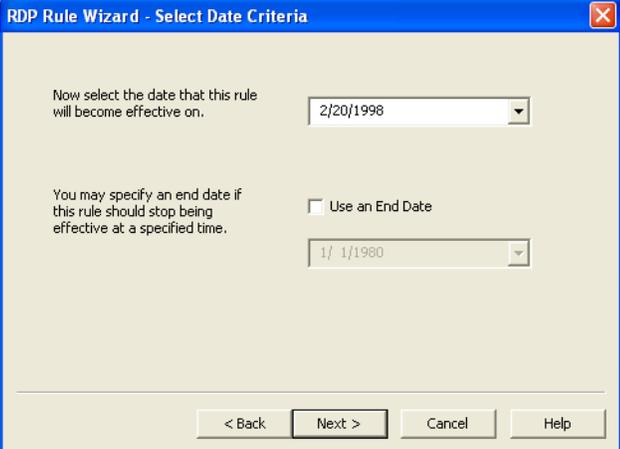
Select the RDP Rules tab to manage your RDP rules assigned to forms. RDP rules are optional. Any form that does not specify an RDP rule is not subject to filtering before the triggering process begins. Also, a form can have multiple RDP rules if necessary. During runtime, each rule will execute in sequence 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.



RDP Rule Wizard - Select Date Criteria

Now select the date that this rule will become effective on.

2/20/1998

You may specify an end date if this rule should stop being effective at a specified time.

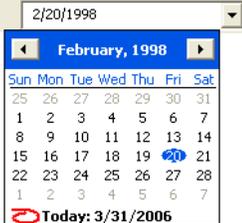
Use an End Date

1/ 1/1980

< Back Next > Cancel Help

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.



2/20/1998

February, 1998

Sun	Mon	Tue	Wed	Thu	Fri	Sat
25	26	27	28	29	30	31
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
1	2	3	4	5	6	7

Today: 3/31/2006

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.



RDP Rule Wizard - Select Region Search Tokens

Select the Region Search Tokens that will locate regions within your extract file. You may select multiple tokens by holding the Ctrl key and clicking on the token name in the list.

Region Search Tokens are defined in the Business Definition File (BDF).

CONTROL_STATE
LOC_SLOC
STATE

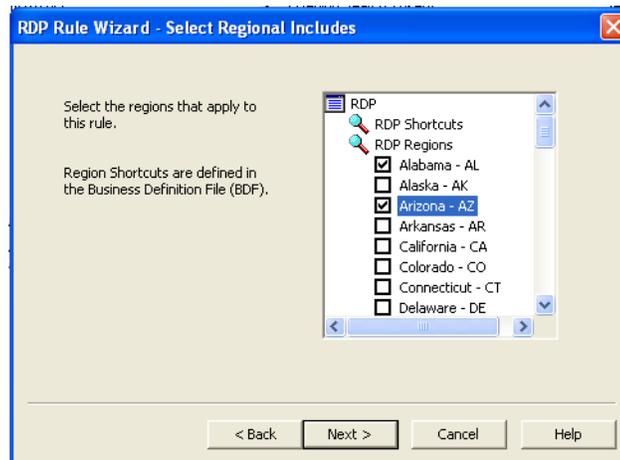
< Back Next > Cancel Help

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.

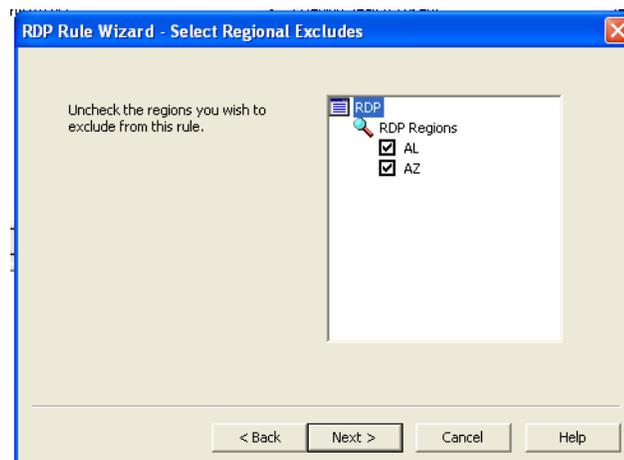
This page lets you identify all the states or Region Shortcut names you want to include with this rule evaluation.



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.



The screenshot shows a dialog box titled "RDP Rule Wizard - Summary". It contains the following fields and lists:

- Rule Name: DATE
- Date Search Token: DATE
- Start Date: February 20, 1998
- Region Search Tokens: LOC_SLOC, STATE
- Region Summary: AL, AZ

At the bottom of the dialog, there are four buttons: "< Back", "Finish", "Cancel", and "Help".

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
End Date (optional)	
Region Search Tokens	
Token	LOC_SLOC
Token	STATE
Region Includes	ALAZ
Region Excludes	
Test Rule Only	<input type="checkbox"/>

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 REPORTS

To generate a report which shows the options defined for this business unit, click the Report tab and follow these steps:

- 1 This window appears when you click the Report tab. Click Generate Report.

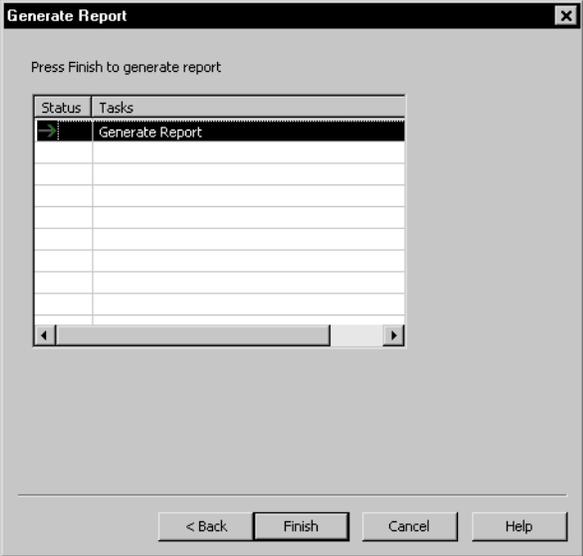
Click here to generate a report



The Reports window appears

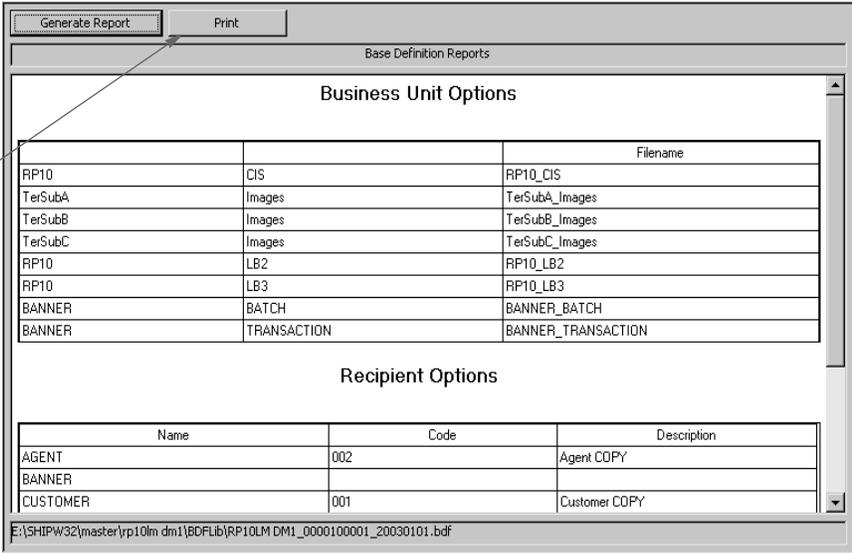


- 2 Highlight the report you want to generate, in this case a Base Definition Report, and Click Next. The Generate Report window appears.



3 Click Finish to generate the report. A sample report appears below:

Click here to print the report.



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 Group on page 112](#)
- [Opening a Group on page 115](#)
- [Setting Up Triggers on page 116](#)
- [SetRecip Options on page 118](#)

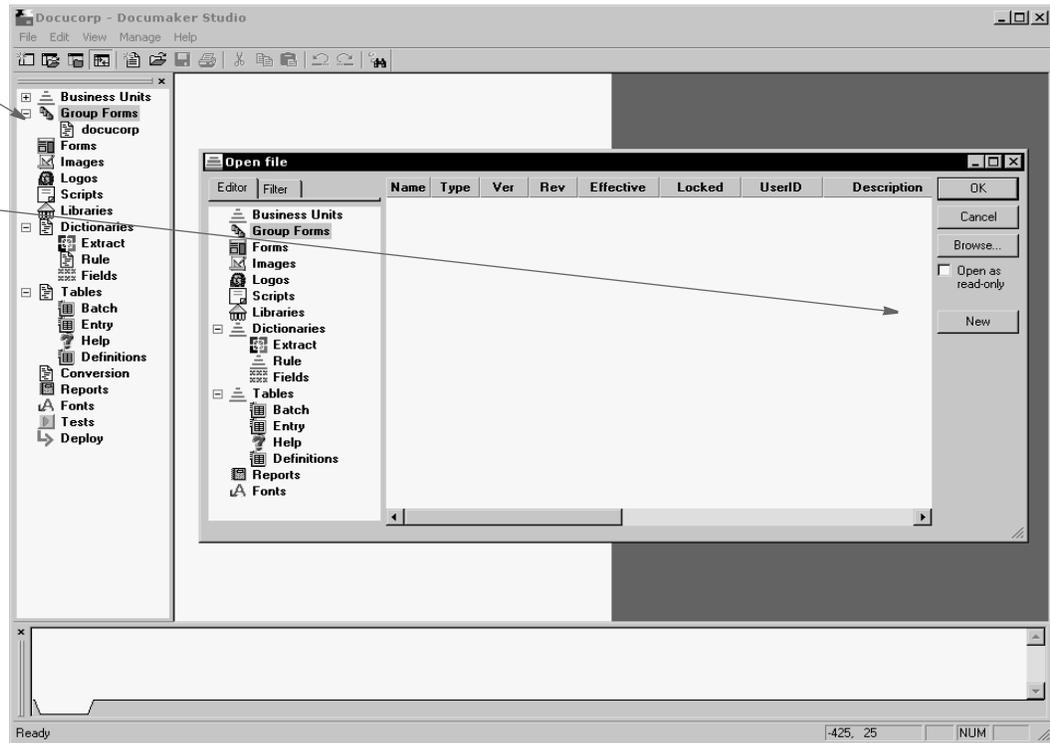
CREATING A GROUP

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

- 1 To create the group, double click on Form Lists. Then click New on the Open File screen.

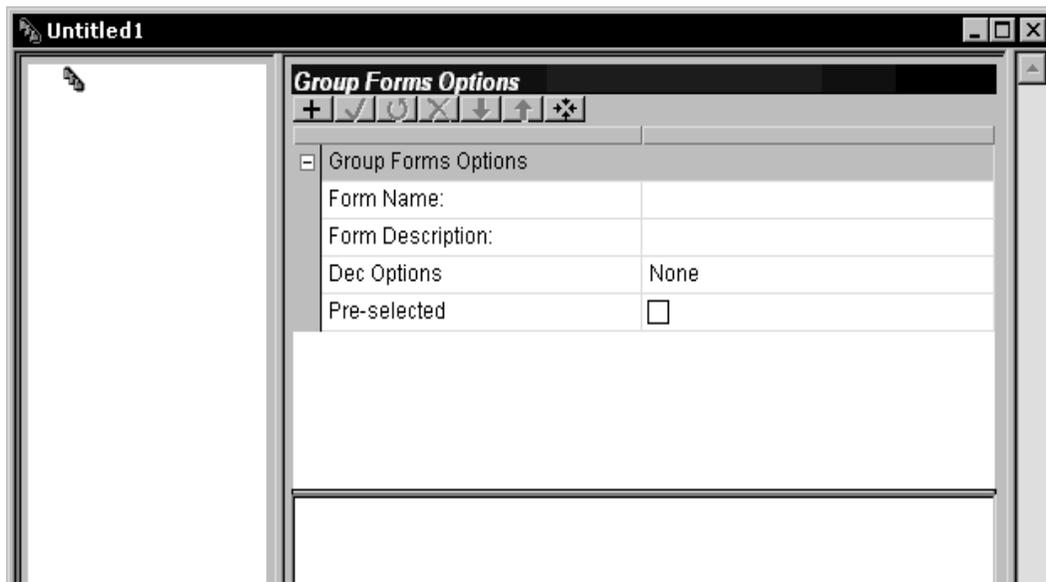
Double click on Group Forms.

Then click New.



You can also right click on the Form Lists option and select New, leave Form Lists highlighted on the next screen and click Ok.

The screen below appears.

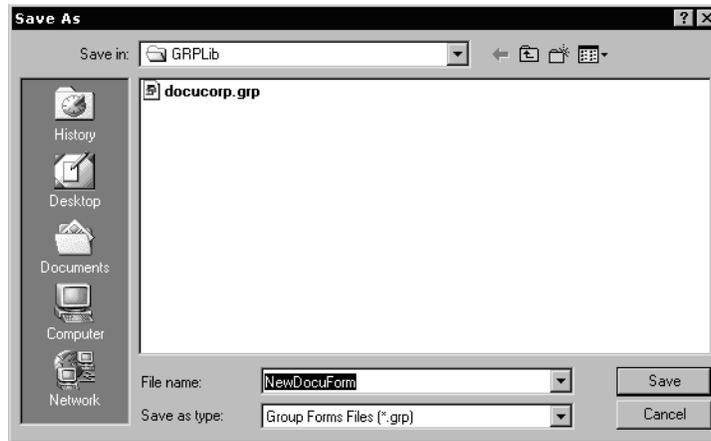


- 2 Enter the following information to set up the forms you want included in the group in the following fields on the Forms tab. Keep in mind that the forms do not have to exist at this point, you are merely creating a list of form names and defining some of the options that apply to those forms.

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: 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.
Filename	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 Filename field.

NOTE: Another way to create a new form list is to select the Edit File option from the Key2 (LOB) level while working with business units. This option will create a new file if one does not exist.

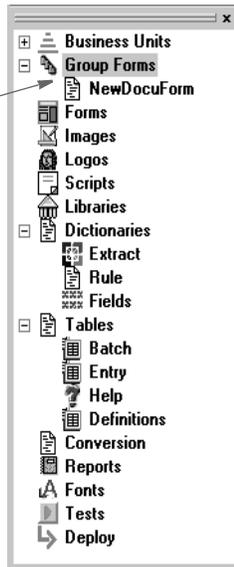
OPENING A GROUP

To modify a group file, follow these steps:

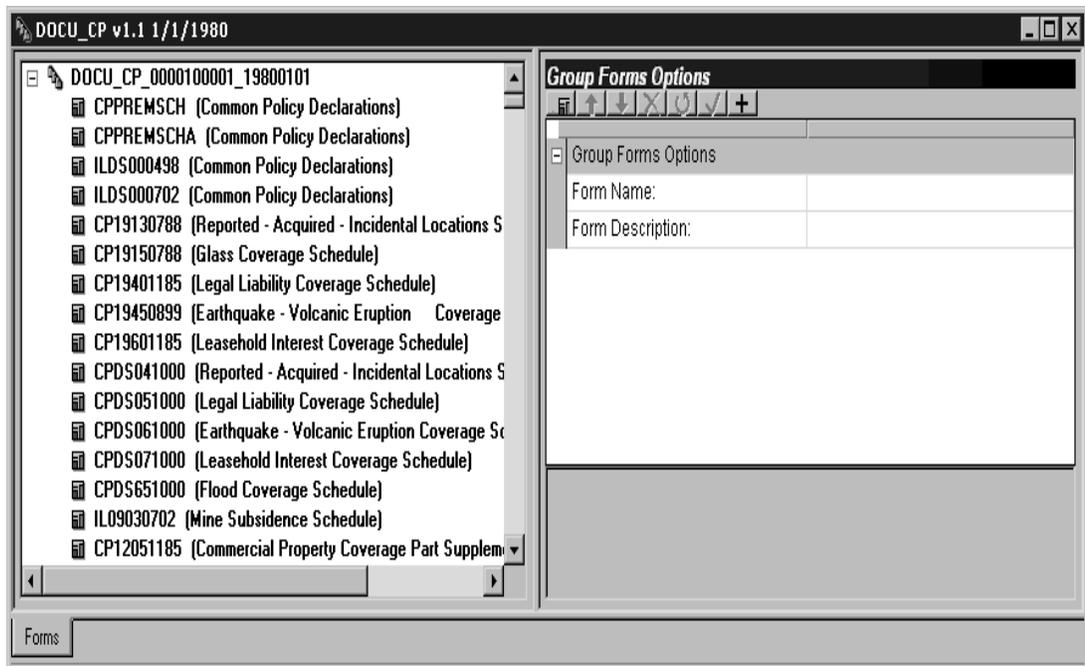
- 1 Click the plus sign (+) to the left of Form Lists and then select the group you want to work with.

Click here.

Then select the group list you want to work with.



The following window appears.



- 2 Make any necessary changes.

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.

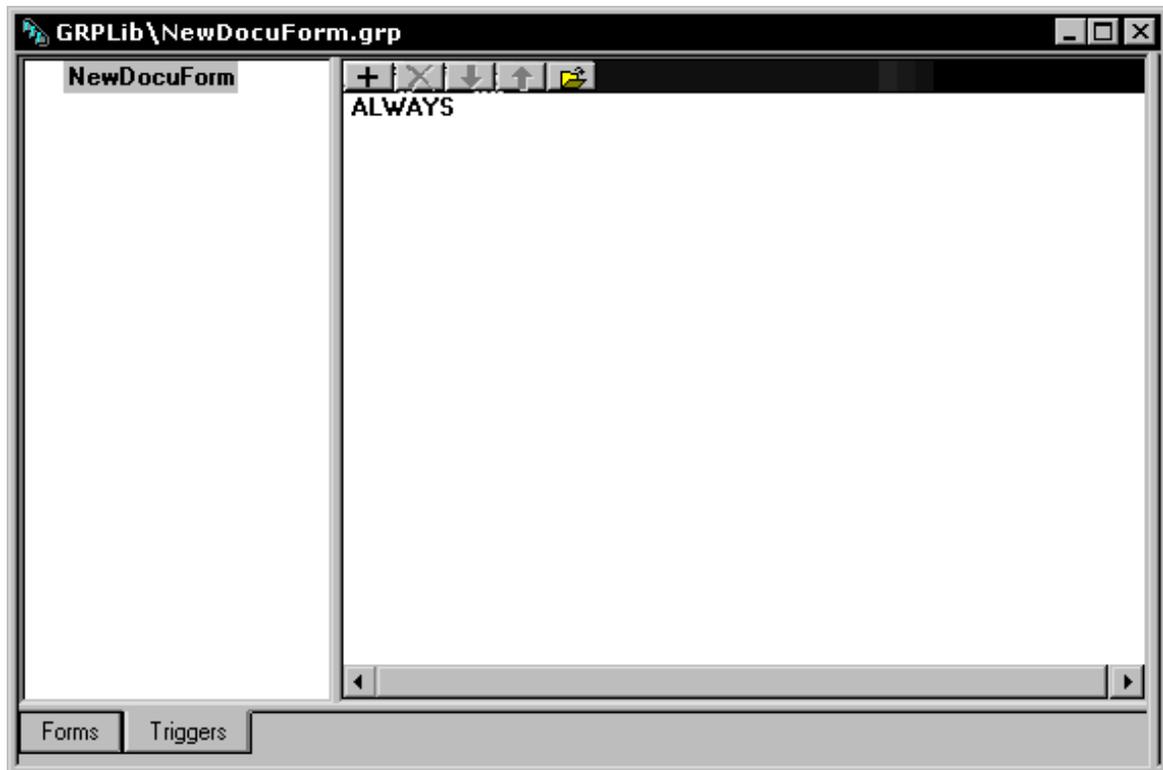
NOTE: Section level triggers are stored at the form level and specify the condition that would cause the section to be included on the form. Only set up section level triggers if the section's presence on the form is conditional.

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.

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 Open.
- 3 Click the Triggers tab.



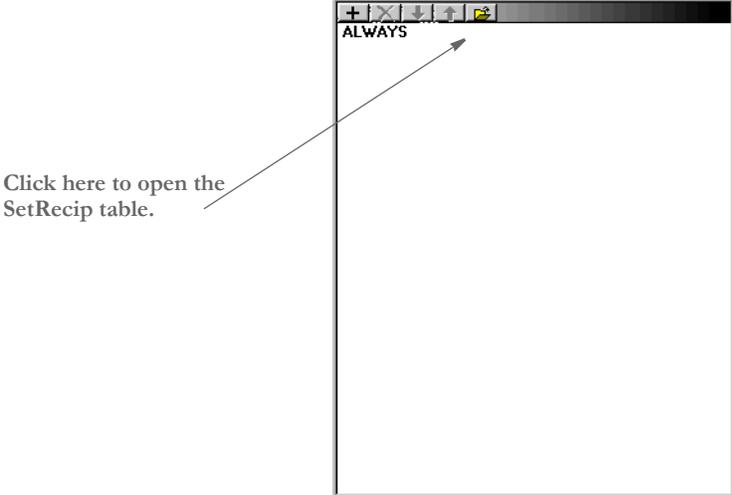
- 4 Highlight the form to which you wish to add a trigger and right click. Select Add. The SetRecip table appears.



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

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:



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 <RECIPCPYCT>	

Section	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.

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 122](#)
- [Using the Screen on page 123](#)
- [Creating a Form on page 129](#)
- [Modifying Forms on page 132](#)
- [Using Form Templates on page 139](#)
- [Setting Up Triggers on page 145](#)
- [Handling Overflow on page 149](#)
- [SetRecip Options on page 157](#)

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.

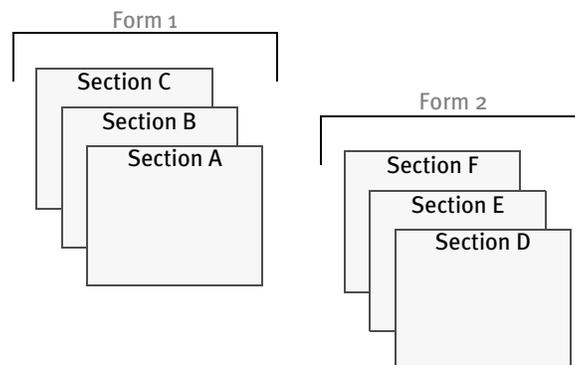
The information for a form is stored in a file with an FOR extension. For each form you create, you have a separate FOR file. A form is listed in a GRP file, whereas the specific information for each form is stored in a separate FOR file.

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



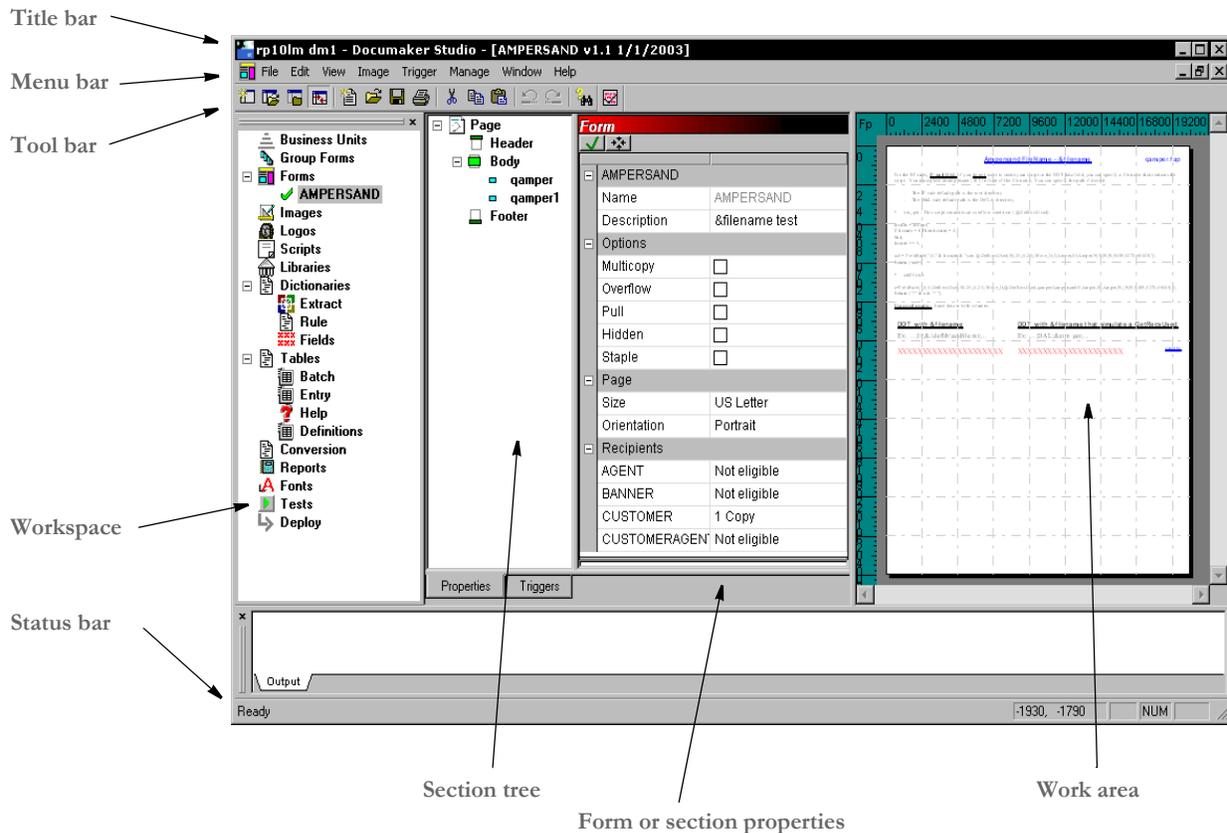
A form may consist of multiple sections, and consequently you can view the varying sections within a form by displaying different pages of the form.

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 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.
Tool bar	The tool bar 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.

Item	Description
Section tree	Shows the sections that comprise the form.
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.

Option	Description
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.
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.

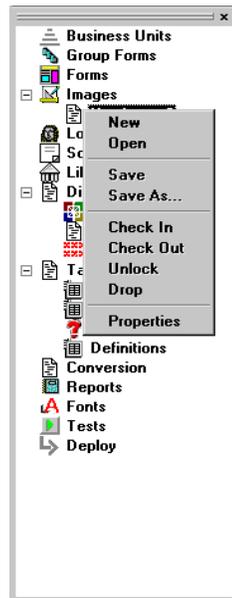


Option	Description
Add	Adds a trigger to the highlighted section.
Delete	Deletes the highlighted trigger.
Move up	Moves the highlighted trigger up.
Move down	Moves the highlighted trigger down.

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...

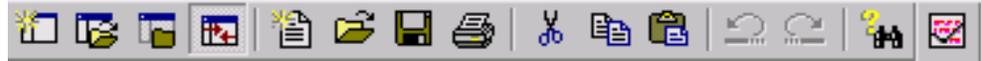
- Checked out (green check mark) by you
- Checked out by another user and you would only have read-only access to (red check mark)
- Open in read-only mode or have never been checked into the library



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 TOOL BAR

The tool bar 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 tool bar shown when you are working with forms:



Standard tool bar icons

Shown below are the tool bar 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

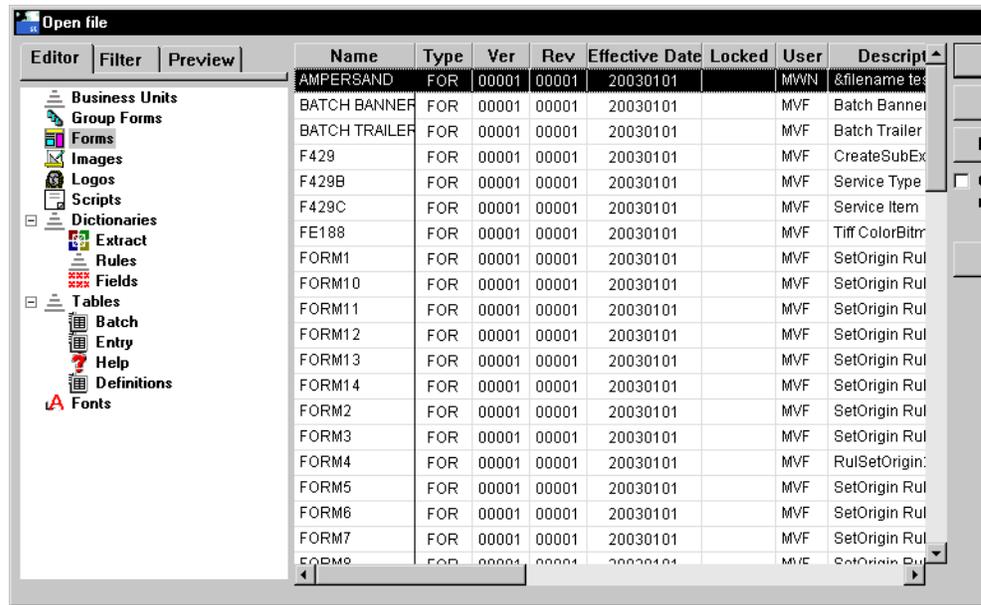
Form tool bar icons Shown below are the tool bar icons that appear when you are working with forms.

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

CREATING A FORM

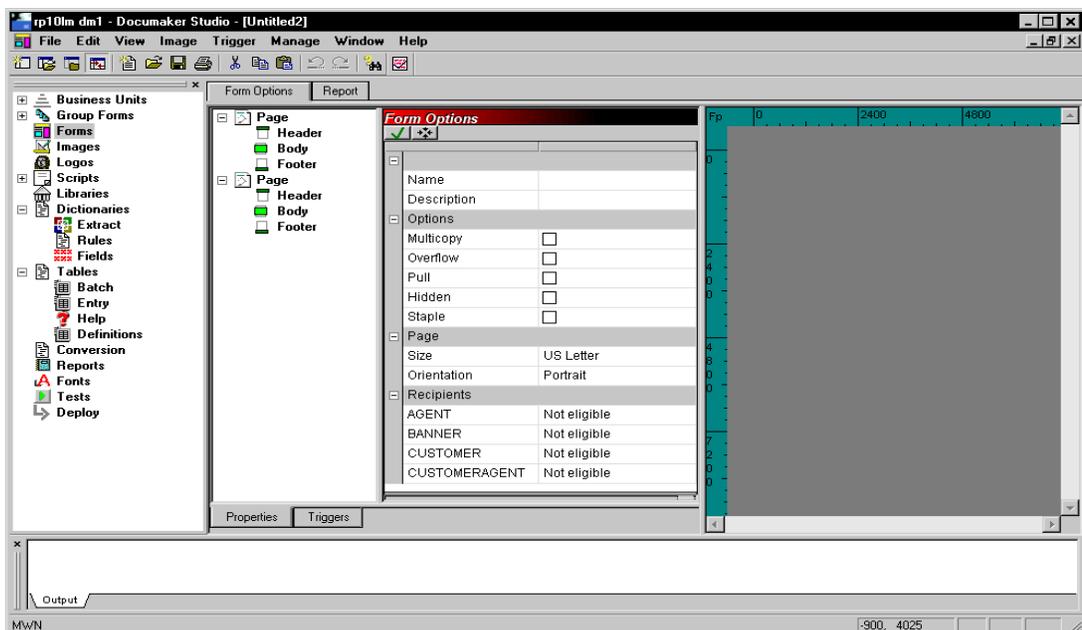
To create a form, follow these steps:

- 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: - 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.

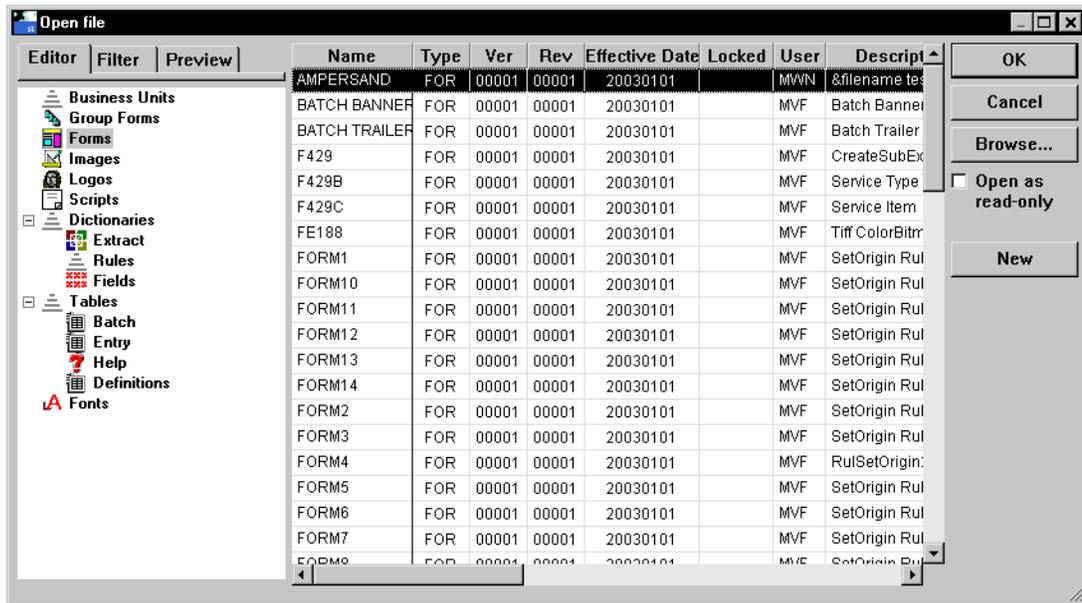
NOTE: You can use the icons at the top of the option area to save the changes or insert a section from the library or disk.

To define section-level triggers, see [Setting Up Triggers on page 145](#) and [SetRecip Options on page 157](#).

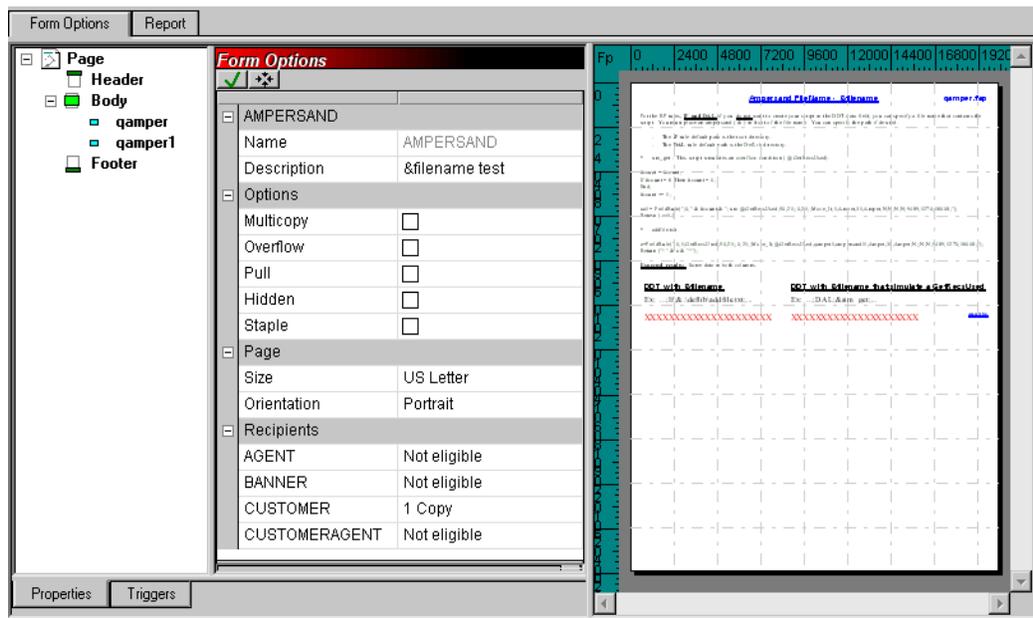
MODIFYING FORMS

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.

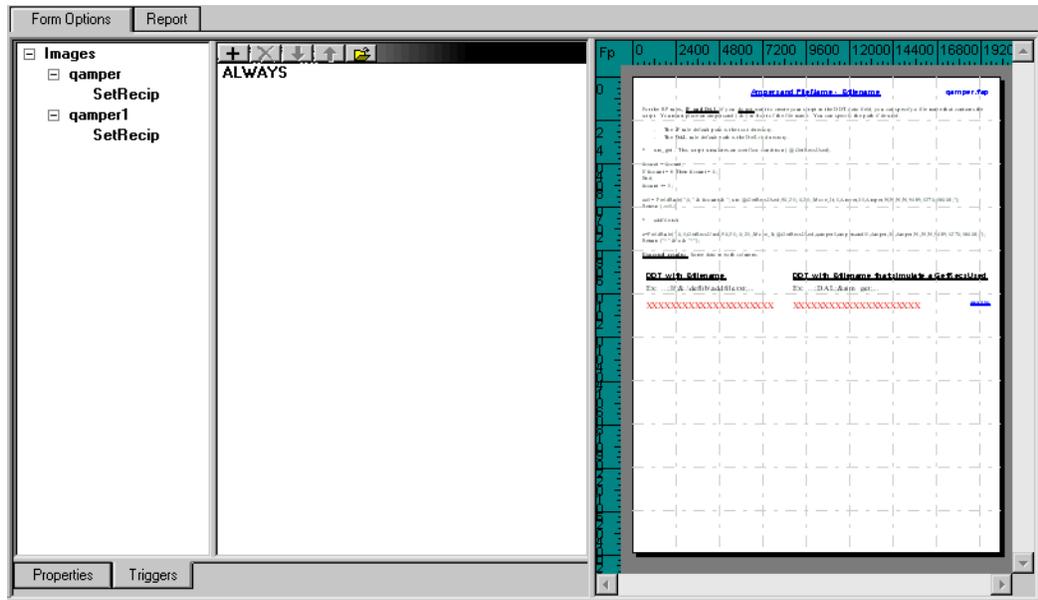


The first panel shows a tree view of the sections that comprise the form. The sections are segmented by page and whether they are designated as a *header* or *footer* sections.

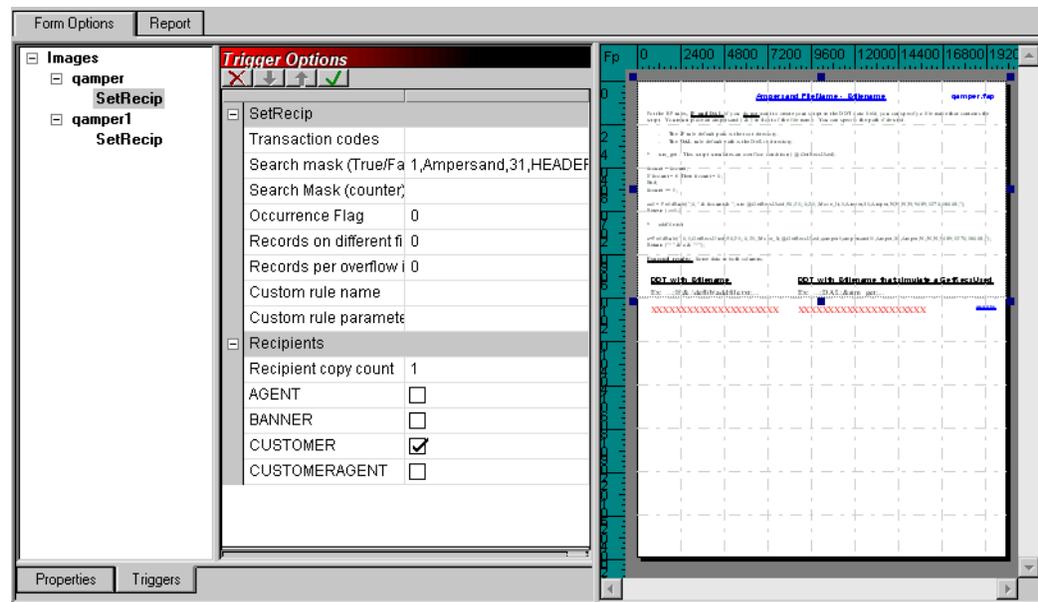
The next panel shows you the property settings for the active object. This is where you add or modify the property settings.

The final panel includes a preview of the form that shows the default layout of the sections. You can scroll through this panel to see all the pages defined for the form.

If you click on the Triggers tab. The window changes as shown here:



The tree view shows you the form and section triggers associated with each. As you click on a trigger in the tree, the Property window shows you the various attributes associated with that trigger.

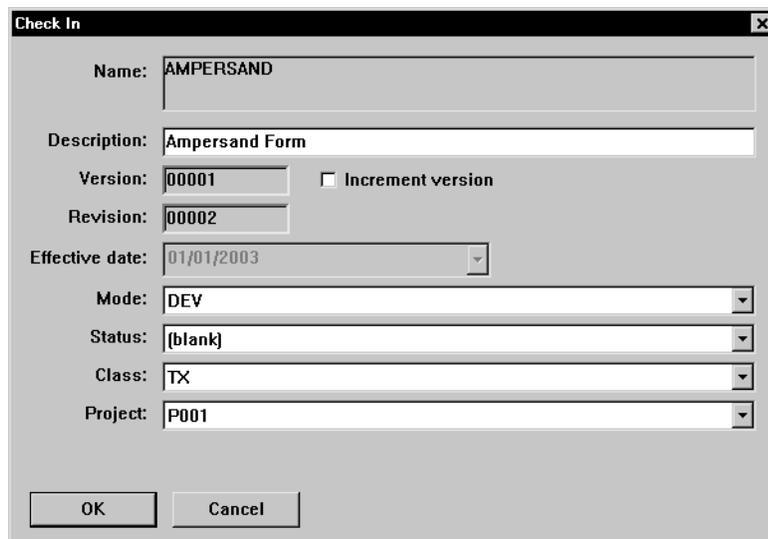


NOTE: There are two types of triggers. The ones shown in the examples are the standard triggers which have been in use through prior releases. Newer triggers are actually given names that appear in the tree. The attributes displayed in the Property window will differ.

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 145](#) and [SetRecip Options on page 157](#).

- 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.



- 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 233](#).

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.

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).

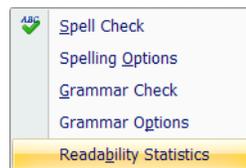
NOTE: This test is designed for English. The scores may not be valid when you run the test on non-English text.

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.

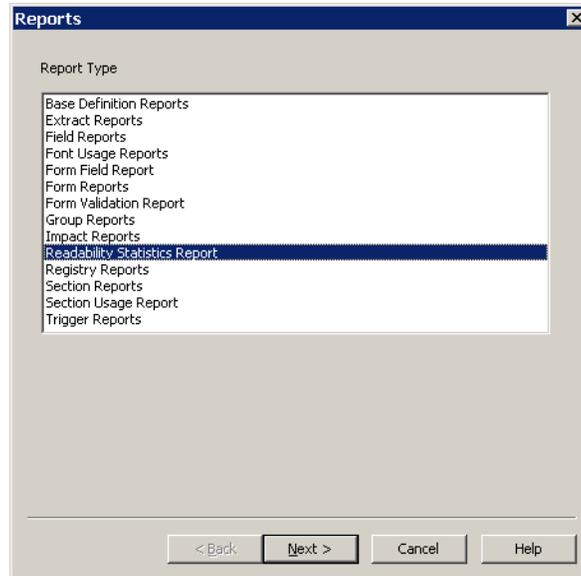
When running a test scenario, this option is only available when sections and forms are displayed as part of the test run. When generated during a test run, the readability statistics are based on all of the forms and sections in the form set.

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

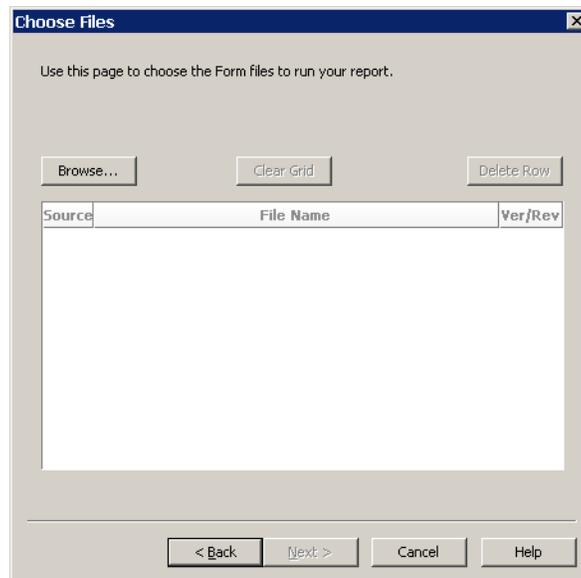


Printing the Readability Statistics Report

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

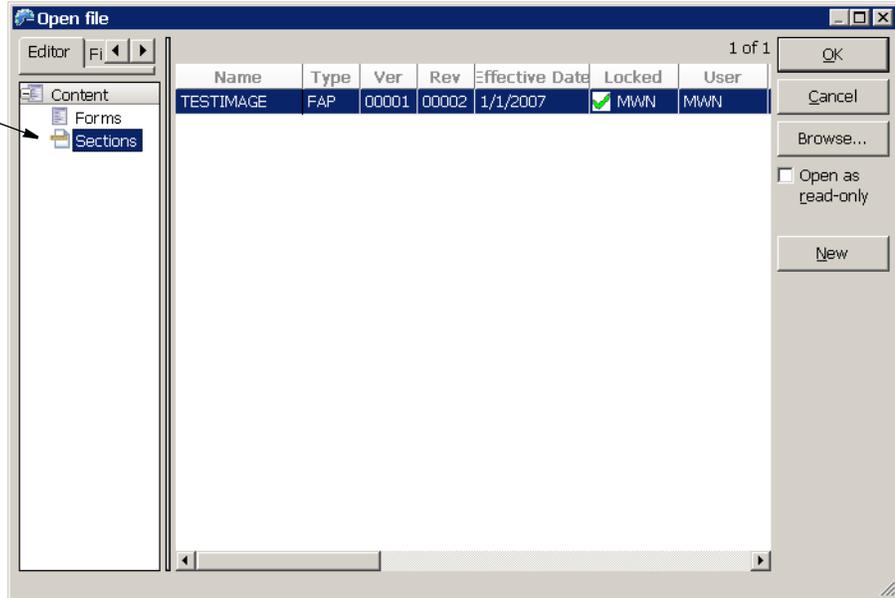


When you click Next, the Choose Files window appears:

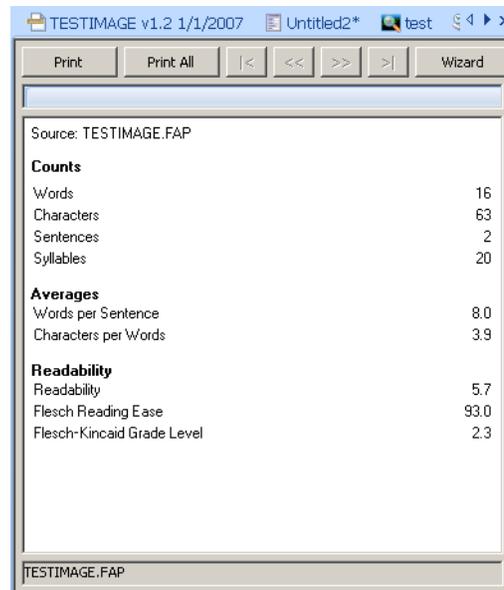


Click Browse to display the Open File window and select the forms or sections for which you want readability statistics generated:

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

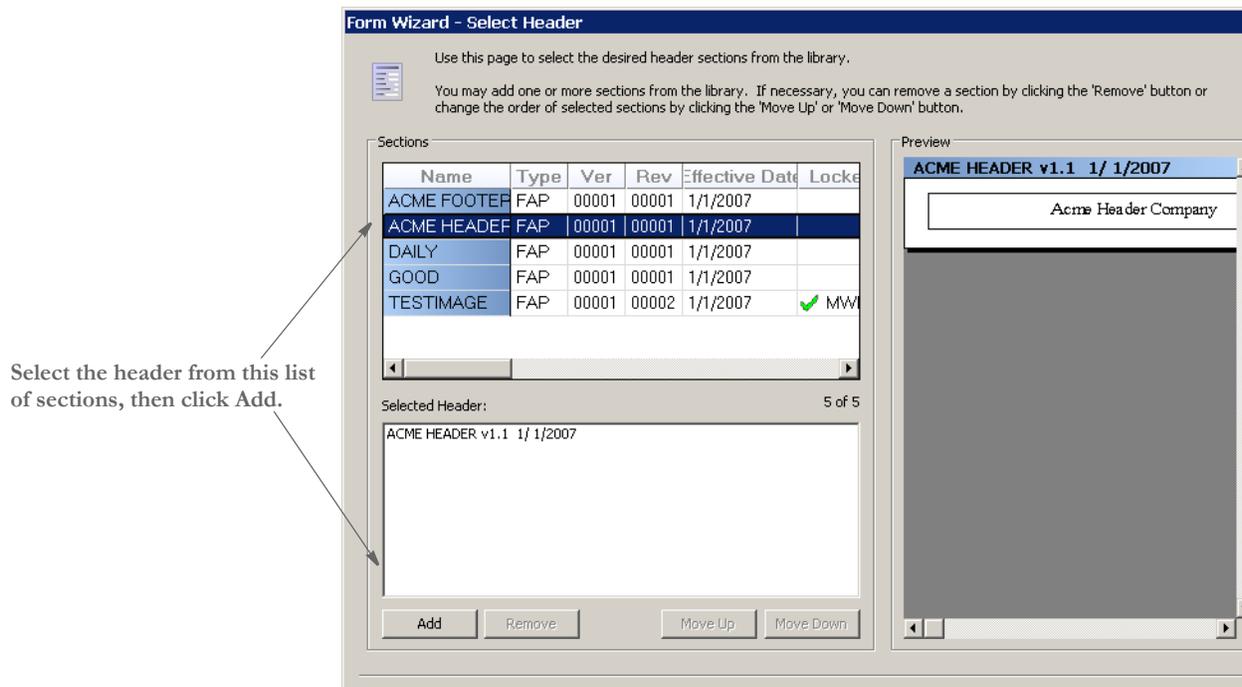
Studio lets you create templates for the forms you create with Studio. You can use templates to more easily implement and maintain document standards while also speeding the process of creating new forms. For instance, you can use templates to make changing your company's name or logo easier.

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.

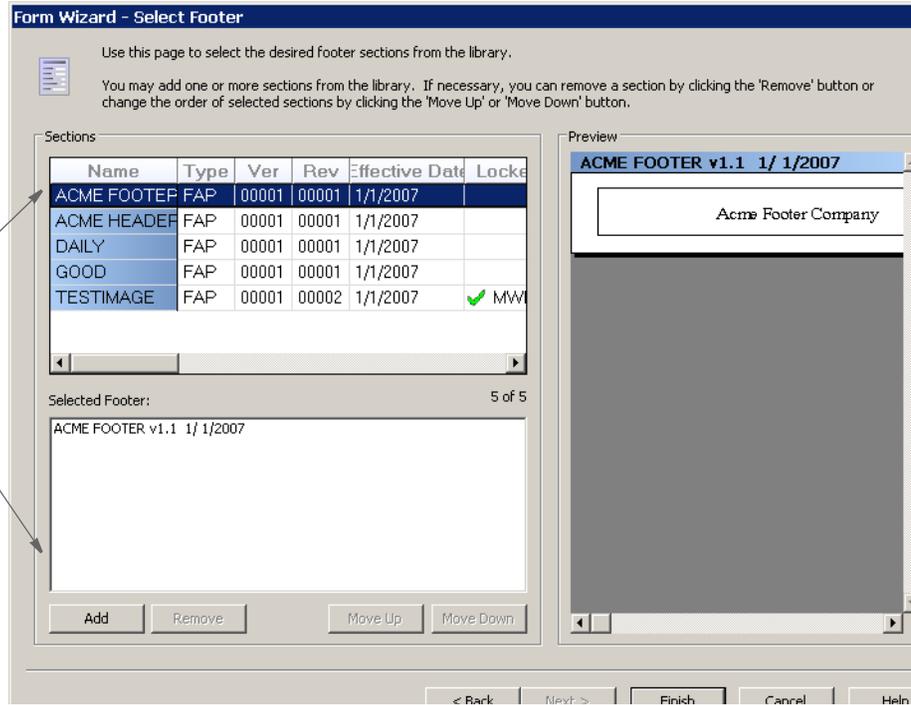


You can see a preview of the section by clicking on it.

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.

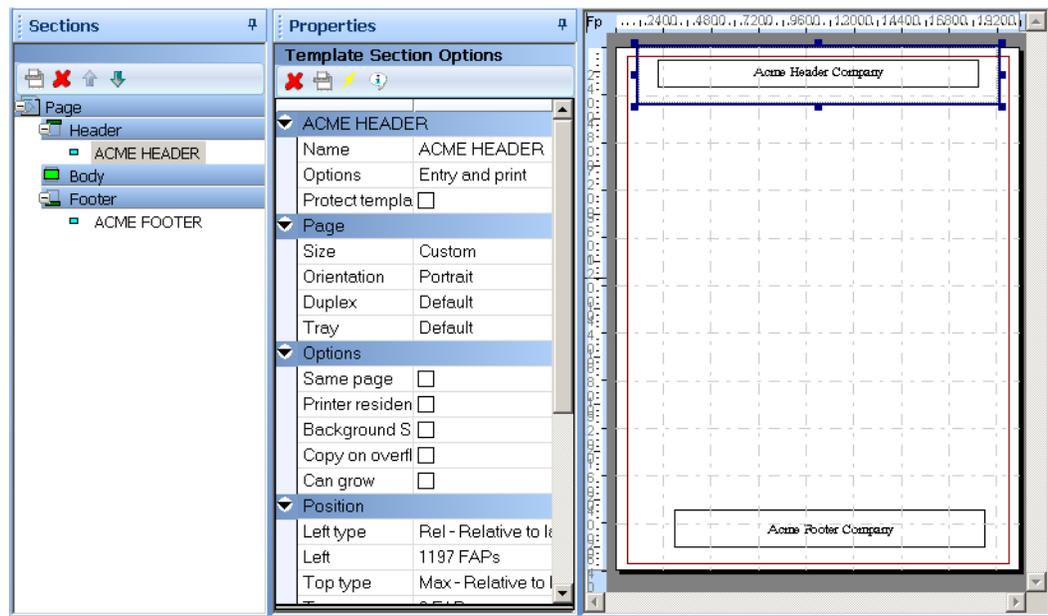
Click Next to display the Footer window and select the sections for the template's footer.

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 (IPL 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.

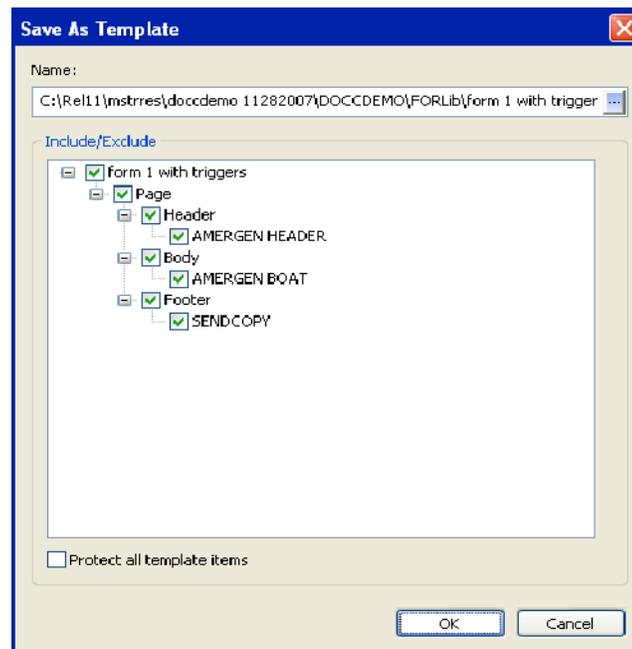
Note that when you reapply a template, this can rearrange, add, remove, and alter the previous template items. In addition, if the previous template definition did not protect the header or footer sections and the new template does, any sections in those areas that are not part of the template are removed.

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.

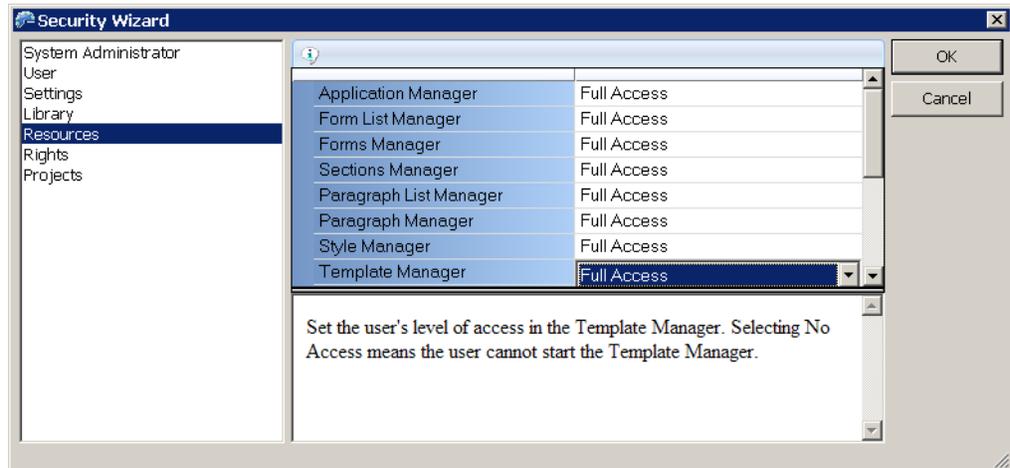


You can select or deselect items by clicking on the top item in the tree (or any other parent item).

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

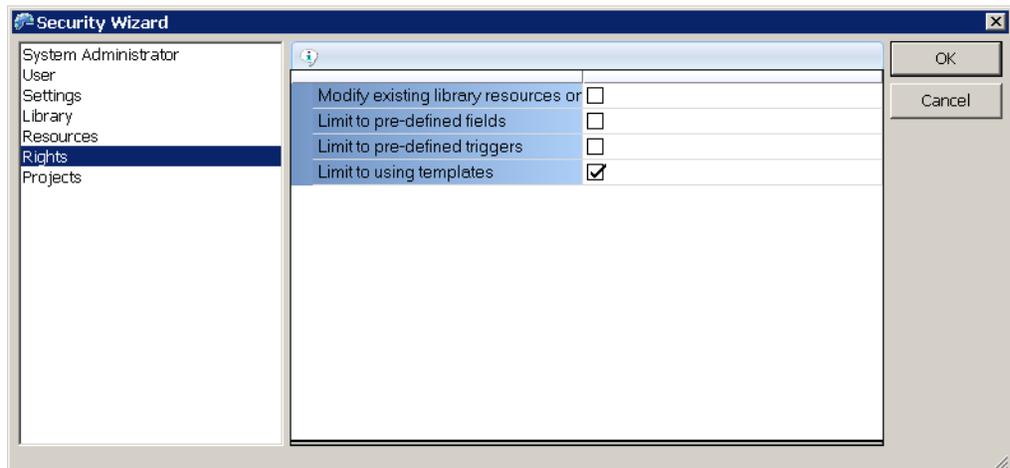
Controlling Access to Templates

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.



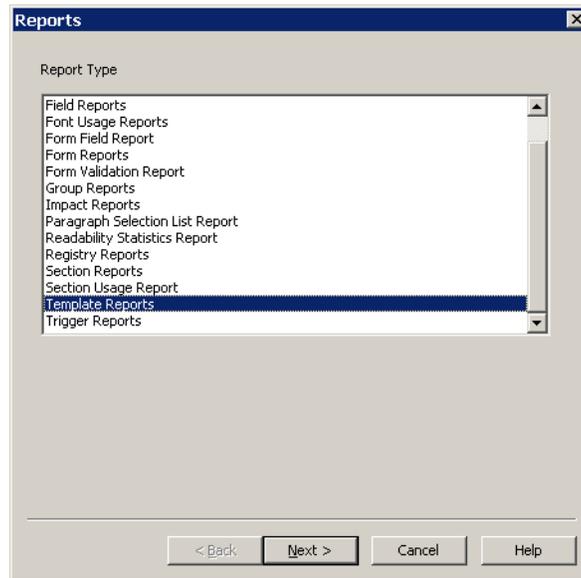
Option	Description
Full Access	Choose this option if you want the user to check out templates and modify them.
No Access	Choose this option if you want to prevent the user from checking out templates. If you choose this option, the user cannot open Template manager.
View Only Access	Choose this option 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.

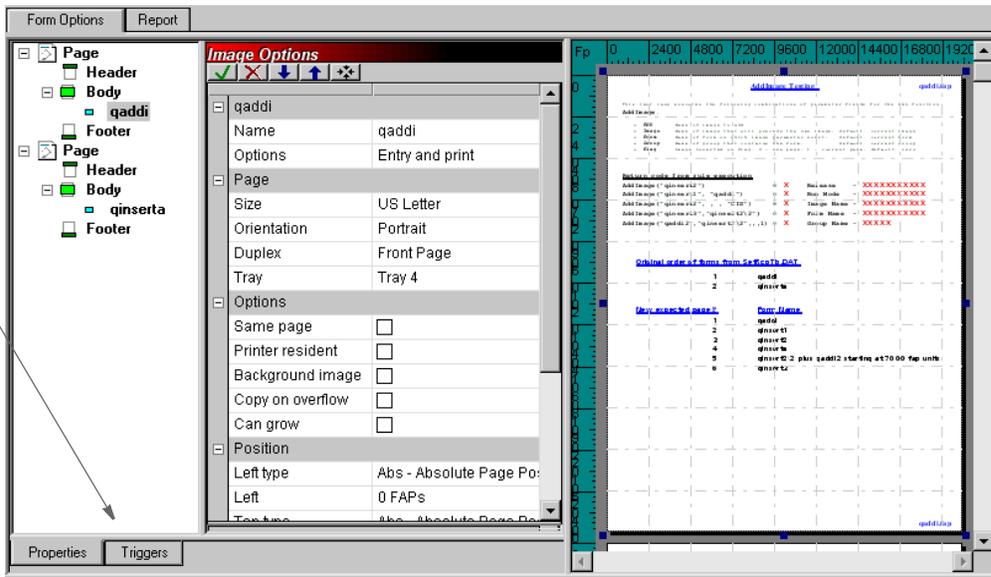
Studio assumes you do not want to trigger any of the sections that comprise a form if the form-level trigger is false. Therefore, section level triggers are automatically ignored when the form level trigger for the transaction being processed is false.

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.

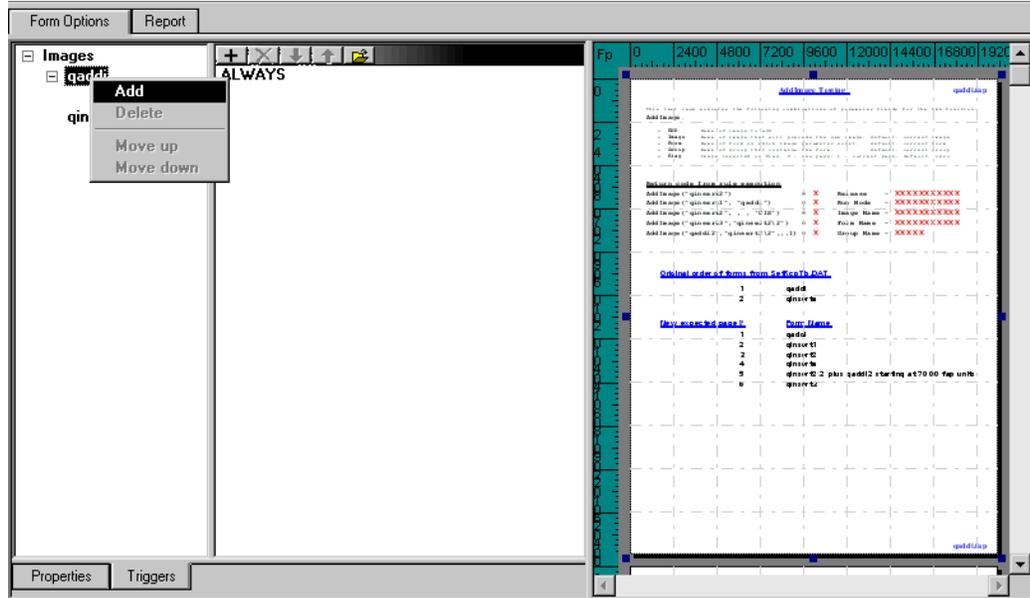
Follow these steps to add section-level triggers.

- 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.

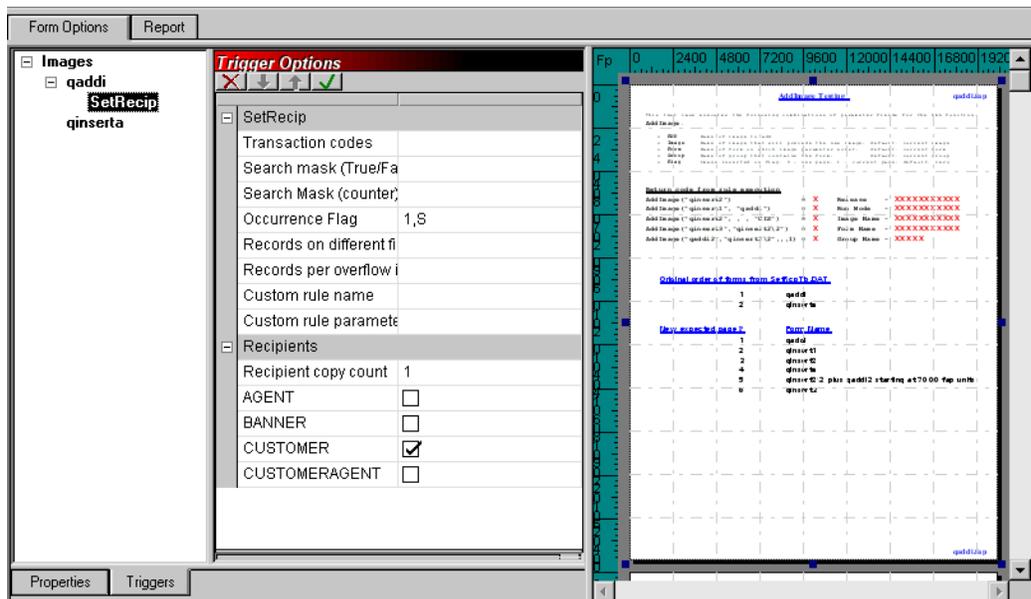
Click the Triggers tab.



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



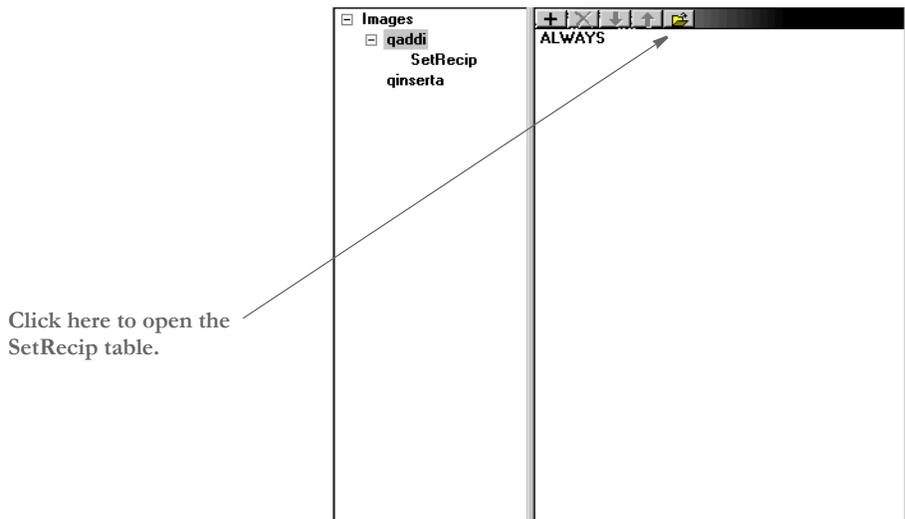
- The SetRecip table appears, on which you can edit or set up the triggers for the section.



EDITING A TRIGGER

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

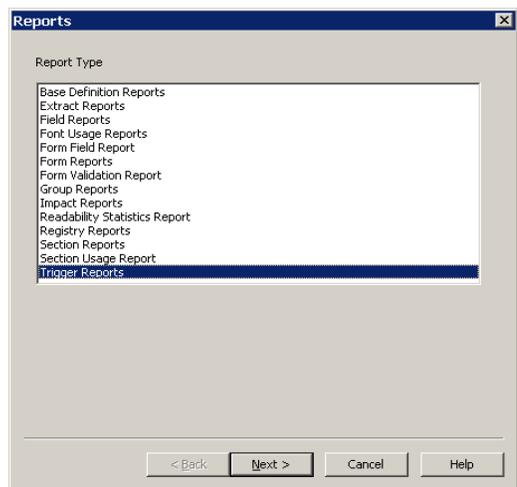
You can also highlight the trigger and click on this icon:



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:

The screenshot shows a report window with the following structure:

Business Definition: MYNEWWORKSPACE	
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> <table border="1"> <tr><td>Record</td><td>0</td></tr> <tr><td>Required</td><td>Not</td></tr> <tr><td>Overflow Multiplier</td><td>0</td></tr> <tr><td>Overflow</td><td>Section</td></tr> <tr><td>Mask</td><td>Default</td></tr> <tr><td>Data</td><td>Form</td></tr> <tr><td></td><td>Formset (Section)</td></tr> <tr><td></td><td>None</td></tr> <tr><td></td><td>Section</td></tr> </table> <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>	Record	0	Required	Not	Overflow Multiplier	0	Overflow	Section	Mask	Default	Data	Form		Formset (Section)		None		Section
Record	0																		
Required	Not																		
Overflow Multiplier	0																		
Overflow	Section																		
Mask	Default																		
Data	Form																		
	Formset (Section)																		
	None																		
	Section																		

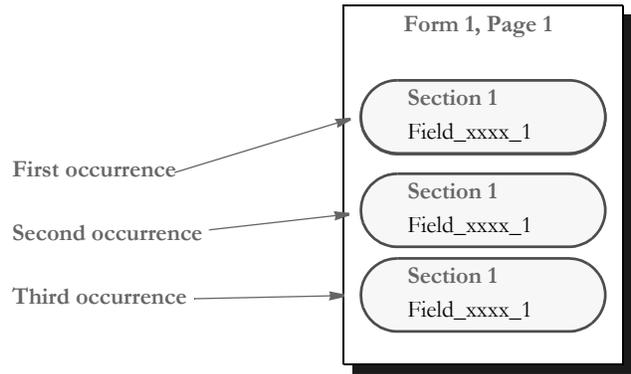
If you use the defaults for these fields at the field level, these fields inherit what is defined at the section level. If you choose different settings at the field level, the system uses your choices instead of the section level settings. So, unless you are mixing overflow and non-overflow variables 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.

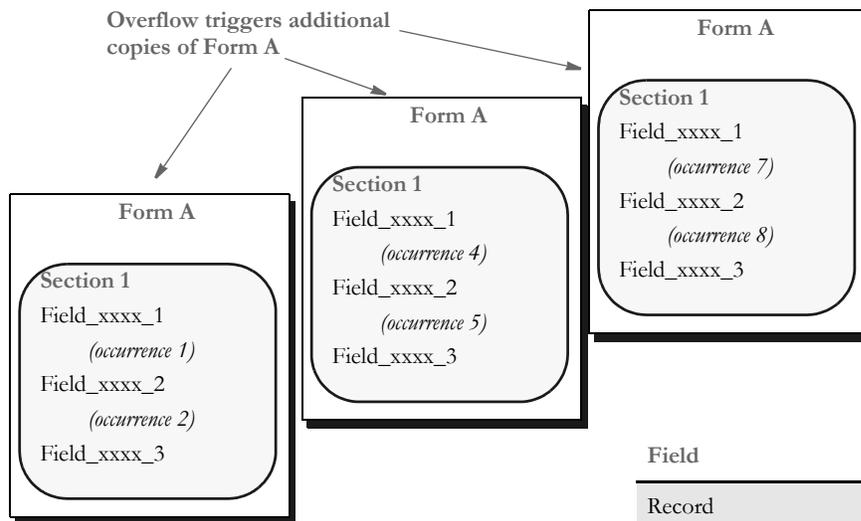
Section-level overflow In this example of section-level overflow, Field_xxx_1 repeats on Section_1.



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.

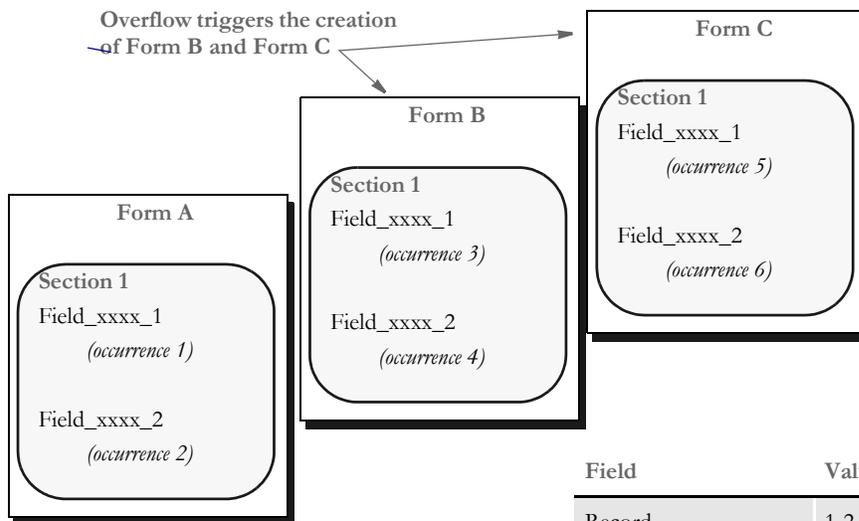


Field	Value
Record	1-3
Overflow	Form
Overflow Multiplier	3

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.



Form set(section)

Field	Value
Record	1-2
Overflow	Formset(Section)
Overflow Multiplier	2

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.

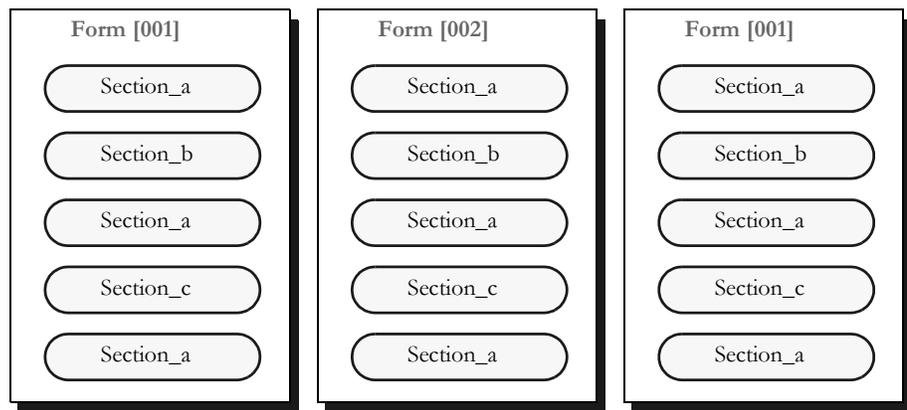
NOTE: These functions correspond to the options you can choose for the Overflow field: Section, Form, Formset(Section).

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

Form	Section	GetImage	GetFormsetImage	GetForm
	a	2	5	1
	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:

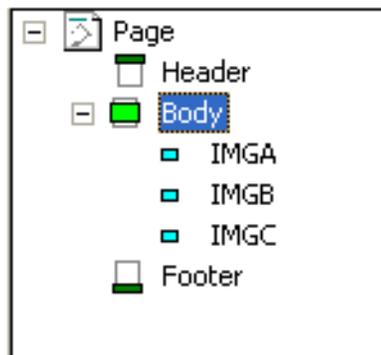
Field	Description
SetRecip Information	
Transaction Codes <TRANSCDS>	By including one or more Transaction codes in this field, you trigger a particular form (or section) only if the extract file record includes that particular Transaction Code.
<RECIPS>	???
Search Mask (Counter) <MASK1>	Lets you set 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 <RECIPCPYCT> 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.

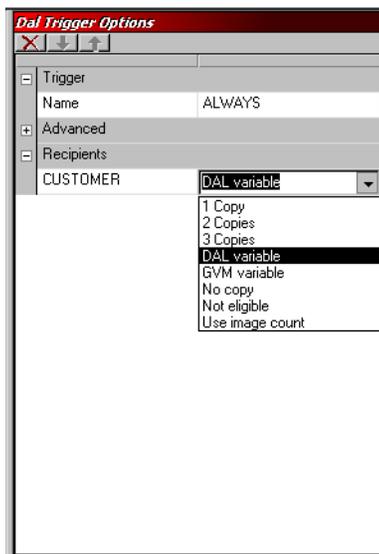
Using Variables to Set Copy Counts

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. Two examples of sections include an insurance policy declaration page and a 1040 Federal tax return form.

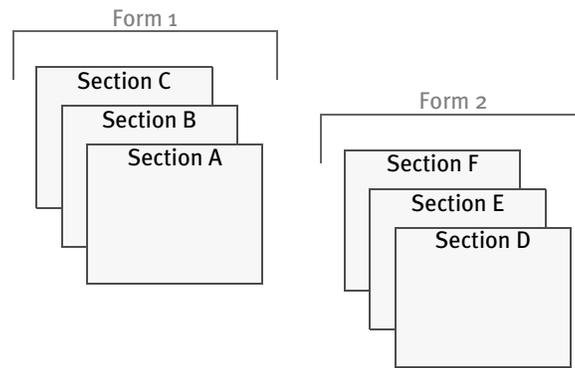
This topic discusses:

- [Overview on page 164](#)
- [Using the Screen on page 165](#)
- [Creating a Section on page 180](#)
- [Opening a Section on page 182](#)
- [Adding Objects on page 185](#)
- [Using Printcommander to Create FAP Files on page 193](#)

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



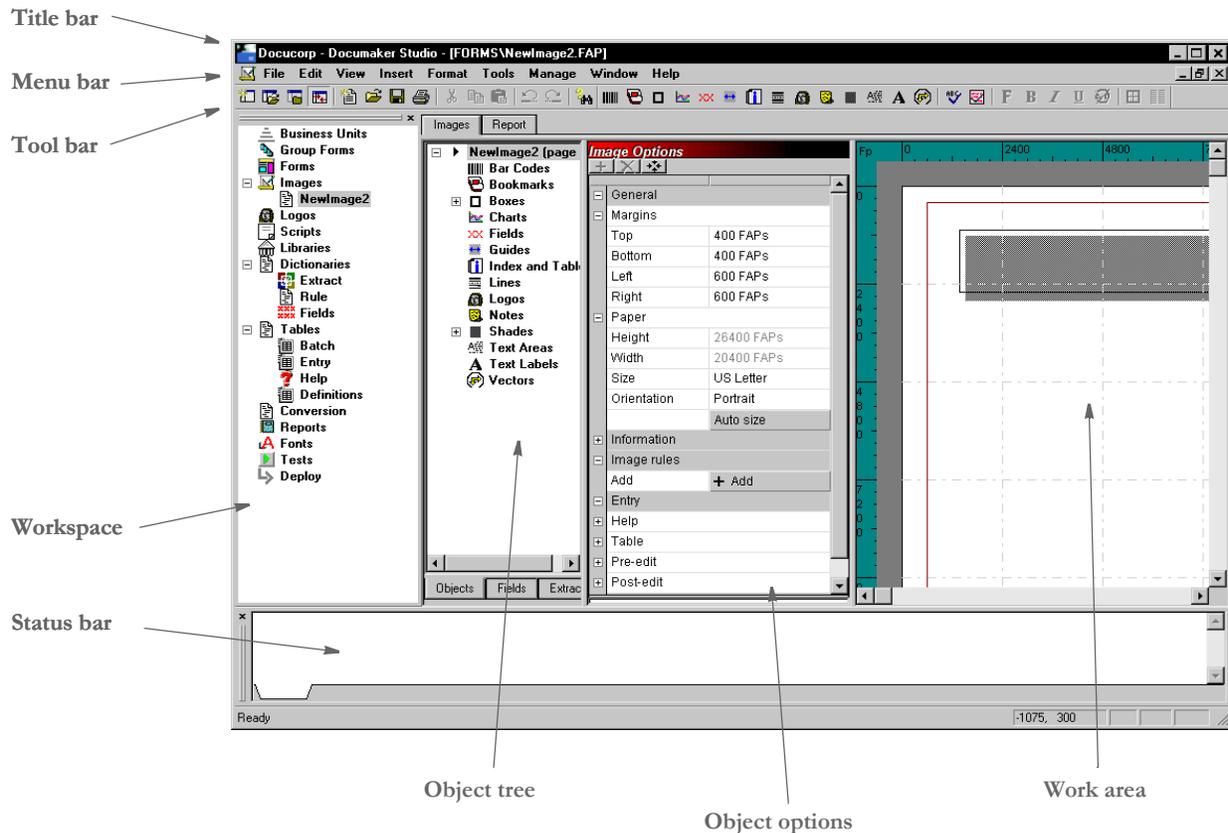
A form may consist of multiple sections, and consequently you can view the varying sections within a form by displaying different pages of the form.

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.

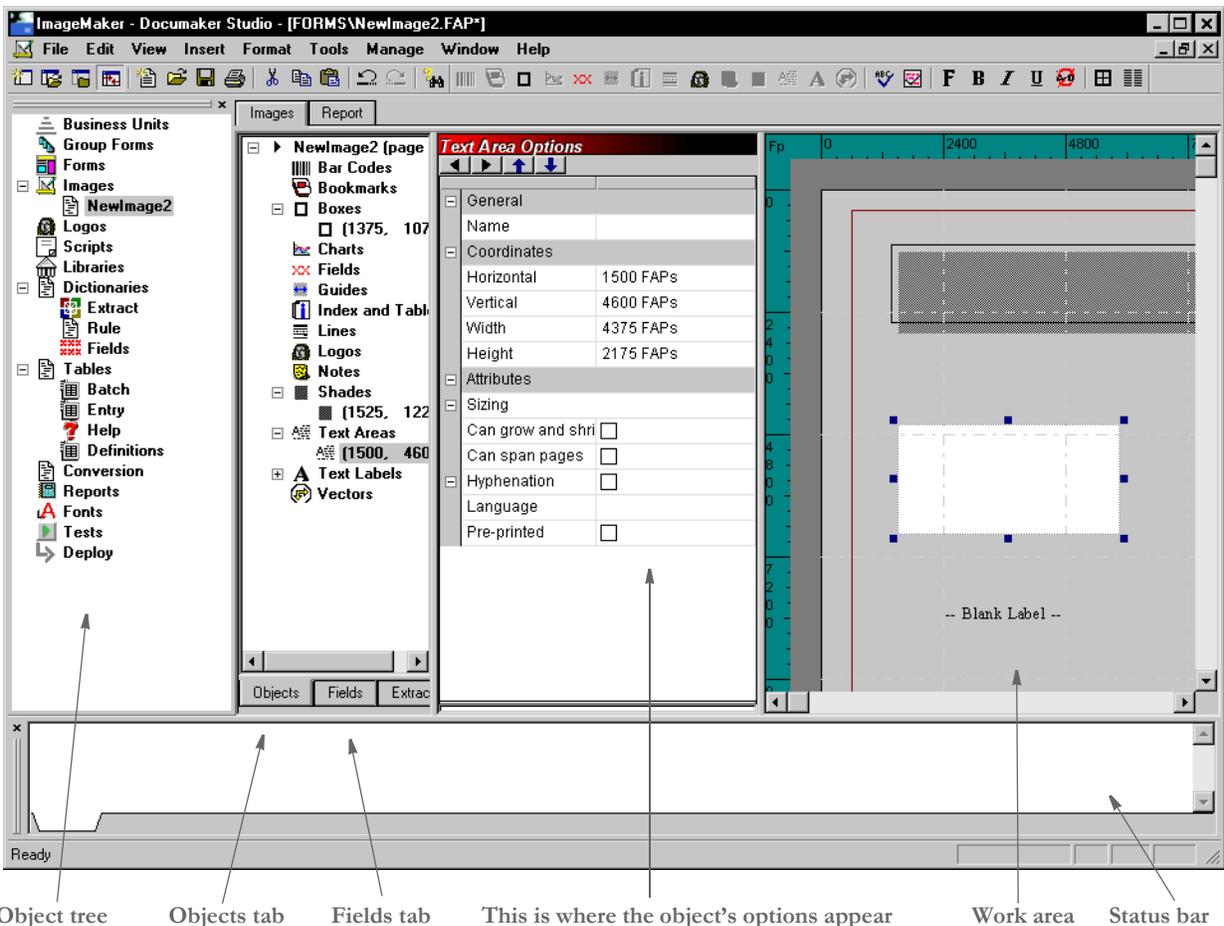


Item	Description
Title bar	The title bar displays the name of the workspace you have open, followed by Documaker Studio, and then the name of the section you have open.
Menu bar	The menu bar provides the list of available pull-down menus.
Tool bar	The tool bar contains a row of icons that provide quick 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.

Item	Description
Object tree	Shows the objects, by object type, that comprise the section.
Object options	Depending upon which object is selected in the work area, specific object properties will be shown. If no object is selected, the section properties appear. This area is grayed out until you begin working in the work area.
Work area	This is where you create the section. Objects are placed in this area as the section is built.

The object tree gives you a list of all objects on the section on which you are currently working. You can access these objects from this area, but you cannot add objects from here.

NOTE: A plus sign by the object type denotes that there are objects of that type existing on the section.



Area	Description
Object tree	The Object tree gives you a list of all objects on the section on which you are currently working. You can access these objects from this area, but you cannot add objects from here.
Objects tab	The Objects tab shows all types of objects, along with the specific objects under each type. Within type, they are listed in the order in which they were created.
Fields tab	The Fields tab shows all fields in the field database (FDB), broken into the group (numeric or alpha) under which it falls.
Options area	The options area gives you property information for the section or for individual objects you can create
Work area	The work area is the area where you create the section. You can enlarge this area by closing or minimizing some of the other areas on the screen. You can also customize the ruler and grid.
Status bar	The status bar gives you useful information while you are working with sections.

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.
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 tool bar.
Format	The Format menu provides options that let you format objects and text and set up the specific formatting properties.
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:

Undo	Ctrl+Z
Redo	Ctrl+Y
Cut	Ctrl+X
Copy	Ctrl+C
Paste	Ctrl+V
Delete	Ctrl+D
Duplicate...	
Select	
Color...	
Font...	
Bold	Ctrl+B
Italic	Ctrl+I
Underline	Ctrl+U
Strike-out	
Do not Hyphenate	
Do not Break	
Language...	
Find...	Ctrl+F
Replace...	Ctrl+H

The grayed-out options are standard Edit menu options.

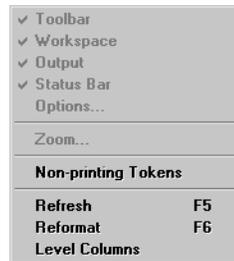
Option	Description	On right click?
Undo	Cancels or reverses your last action or choice.	Yes
Redo	Reverses your last undo.	Yes
Cut	Removes a section of a section and places it on the clipboard.	Yes
Copy	Duplicates a selected section of a section and places it on the clipboard.	Yes
Paste	Places a section stored on the clipboard into an open section.	Yes
Delete	Deletes the selected object.	Yes
Duplicate	Creates multiple copies of an object directly adjacent to the selected object.	Yes
Select		
Character	Selects a character.	
Word	Selects a word.	
Line	Selects an entire line.	

Option	Description	On right click?
Paragraph	Selects a paragraph.	
Column	Selects a column.	
All	Selects everything.	
Deselect	Deselects everything selected	
Color	Change the color of the selected object. Highlight the object, then choose this option.	
Font	Select or change fonts.	
Bold	Bold selected text.	
Underline	Underline selected text.	
Strike-out	Strike-out selected text	
Do not Hyphenate	Prevents hyphenation of text.	
Do not Break	Prevents paragraphs, text labels, and embedded variable fields within text areas, and multi-line variable fields, from being broken by page breaks.	
Language	Select or change the language.	
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:



The grayed-out options are standard View menu options.

Option	Description	On right click?
Toolbar	Toggles on/off the Toolbar	
Workspace	Toggles on/off the Workspace area	
Output	Toggles on/off the Output area	
Status Bar	Toggles on/off the Status bar	
Zoom...	Lets you zoom in or out.	Yes
Non-printing Tokens	Display or hide formatting symbols such as paragraph markers and tab markers.	
Refresh	Lets you update and redisplay the section.	
Reformat	Reformat the text. Use after you have made significant formatting changes	
Level Columns	Balance the text in columns.	

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 pull-down menu or you select to create an object from the tool bar.

When you select Insert, this menu appears:



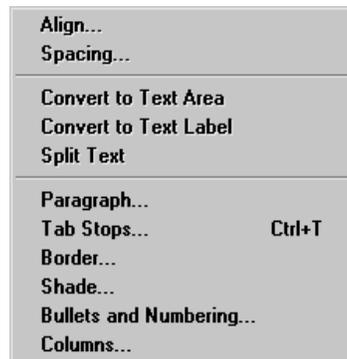
Option	Description
Autocreate	Lets you create objects repeatedly without entering the object's options.
Bar Code	Activates bar code create mode.
Bookmark	Used to add bookmarks to FAP files. A bookmark defines the text you want to print in the table of contents. A level number (TOC1, TOC2, and so on) 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	Activates box create mode.
Break	Text area feature – allows for inserting a break into a text area
Chart	Activates chart create mode.
Field	Activates variable field mode.
File	Text area feature – allows for inserting a file into a text area
Guide	Activates guide box create mode. Used for alignment of objects.
Index and Tables	Used to add an index marker or a table of figures. Only applicable for Documaker Workstation.
Line	Activates line create mode.

Option	Description
Graphic	Activates graphic insert mode.
Note	Activates note create mode.
Page	Activates page insertion screen.
Shade	Activates shading or pattern area create mode.
Text Area	Activates text area create mode.
Text Area Special	Activates text area draw to insert from a file or the clipboard.
Text Label	Activates text label create mode.
Vector	Activates vector create mode.

Using the Format Menu

The Format menu controls formatting functions associated with the section and 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
Align	Aligns according to section margins or a specific object.
Spacing	Places selected objects at specified intervals either vertically or horizontally in relation to each other.
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.
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

Using the Tools Menu

The Tools menu controls features often used when your section is complete or near completion. Spell check and Data Entry check are some of the features available.

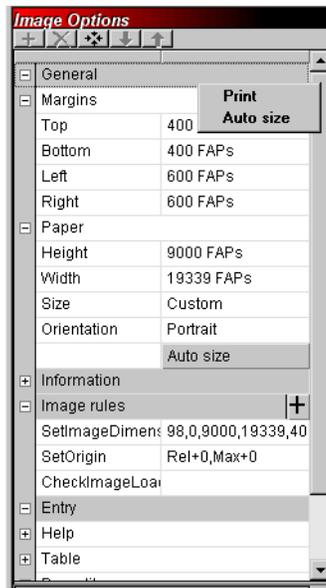
When you select Tools, this menu appears:



Option	Description
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.
Data Entry Check	Activates the section check feature. Lets you test the data entry and navigation rules you assigned to variable fields.
Validate	Lets you validate your section.
Section Report	Generate a report about the current section. You can view and print the report.
Compile	Lets you compile your section file for a particular printer. Compile creates a section file for a PCL, AFP, PostScript, or Metacode printer.
Convert	Lets you convert a FAP file into a normalized Metacode or AFP file.

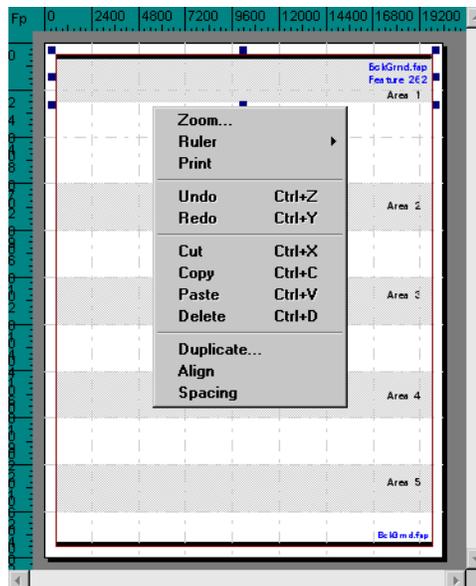
USING THE RIGHT CLICK MENUS

When working with section properties, you can right click to see the following menu:



Option	Description
Print	Use to print the section.
Auto size	Use to automatically size the section.

When working with a section, you can right click to see the following menu.



Option	Description
Zoom	Lets you see a larger or smaller version of the section.
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 section.
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.
Paste	Duplicates part of a section and places it on the clipboard.
Delete	Places what's in the clipboard into an open section.
Duplicate	Deletes the selected object.
Align	Aligns according to section margins or a specific object.
Spacing	Places selected objects at specified intervals either vertically or horizontally in relation to each other.

USING THE TOOL BAR

The tool bar 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 tool bar shown when you are working with sections:



Standard tool bar icons

Shown below are the tool bar 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

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

Section tool bar icons Shown below are the tool bar icons that only appear when you are working with sections.



Icon	Name	Description
	Insert bar code	Lets you insert a bar code
	Insert bookmark	Lets you insert a bookmark
	Insert box	Lets you insert a box
	Insert chart	Lets you insert a chart
	Insert field	Lets you insert a variable field
	Insert guide*	Lets you insert a guide box
	Insert index or table	Lets you insert an index or table
	Insert line	Lets you insert a line
	Insert graphic	Lets you insert a graphic file
	Insert note*	Lets you insert a note
	Insert shade	Lets you insert a shaded area
	Insert text area	Lets you insert a text area

Icon	Name	Description
	Insert text label	Lets you insert a text label
	Insert vector	Lets you add a vector drawing
	Spell check*	Lets you check spelling
	Data entry check*	Lets you run a data entry check
	Fonts	Opens the font cross-reference file.
	Bold	Changes the font for the selected text to bold.
	Italic	Changes the font for the selected text to italic.
	Underline	Underlines the selected text.
	Do not hyphenate	Tells Studio not to hyphenate the selected text.
	Border	Opens the Paragraph Properties window so you can add a border.
	Columns	Opens the Columns Properties window so you can define columns.

NOTE: An asterisk (*) denotes that icon is only available when you are composing forms. A note, however may be used during entry by including the following option:

```
< Control >
  ShowNotes = Yes
```

The default is No.

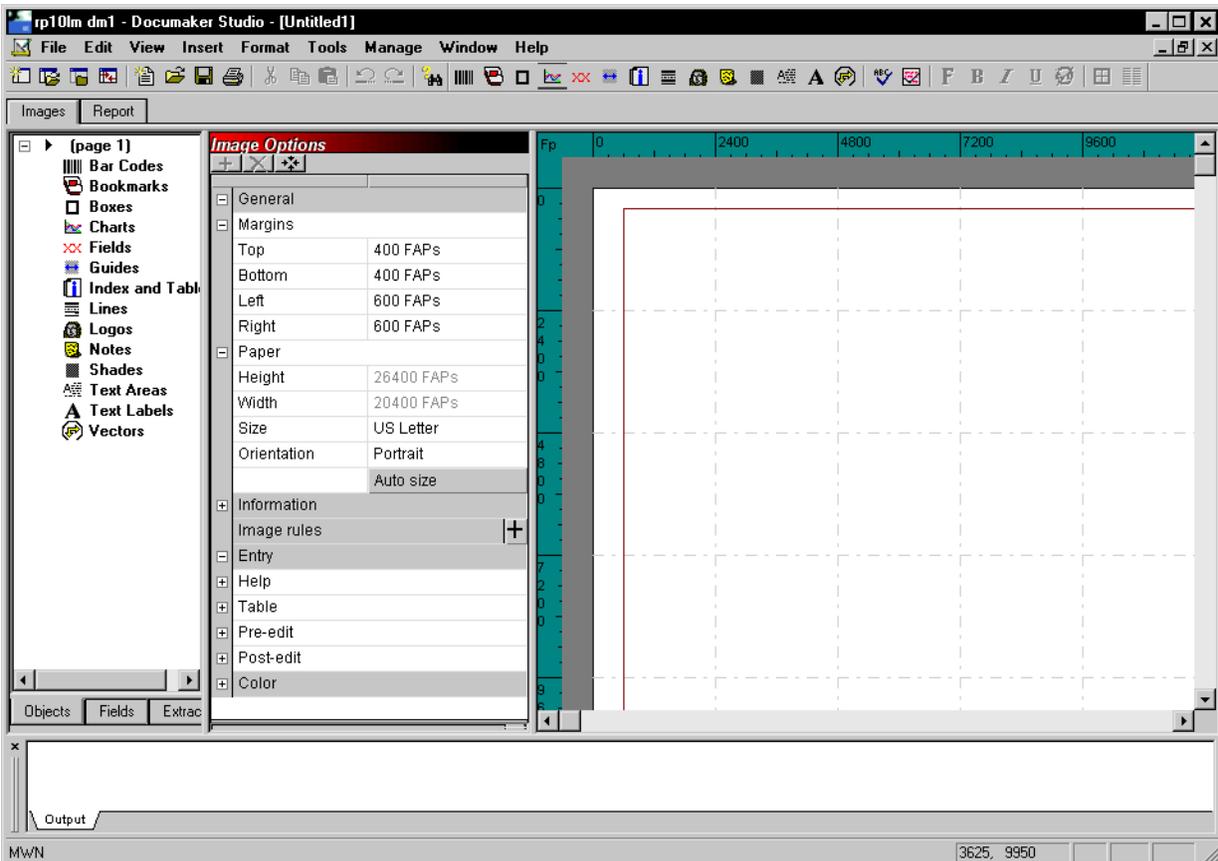
All section objects have default attributes. You can override these attributes by modifying the values assigned to the options in the default value can be overridden. The default values/attributes for the object are established in the Options by Group Section of the INI file.

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 185](#) 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.

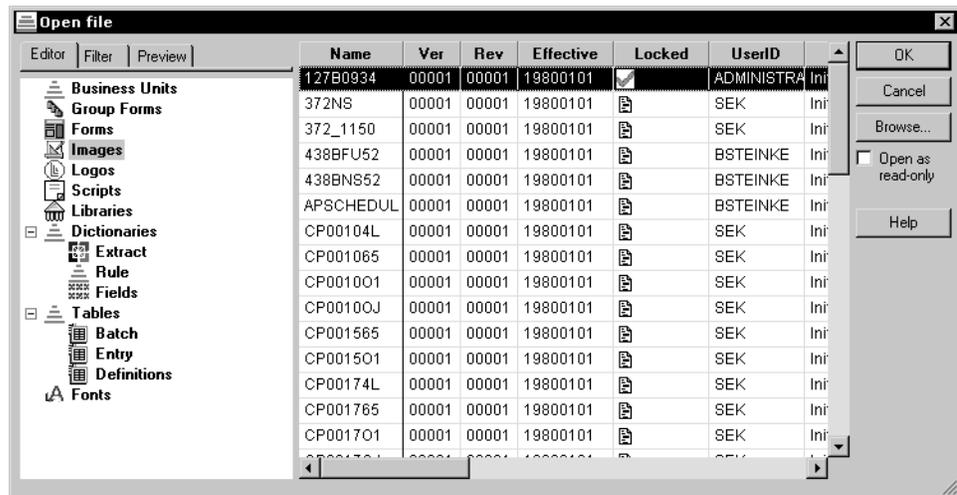
For example, suppose you have a section that is two inches in height, but the SetOrigin rule for that image specifies a relative placement 1/2 inch from the previous section. In this case, 2 1/2 inches of space is required for that section to fit on the page. If there is less than 2 1/2 inches remaining 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

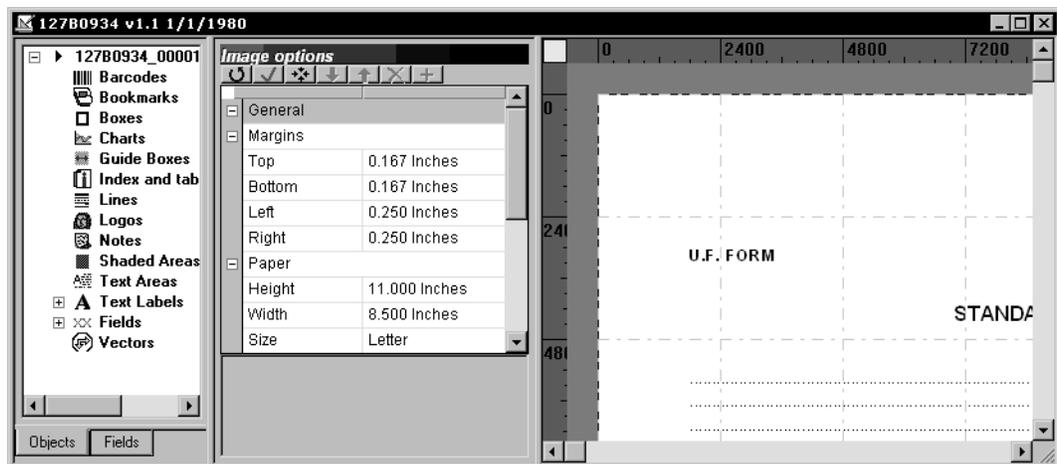
Follow these steps to select and open a section:

- 1 Double-click sections. The Open File window appears. You can also right click on Sections and choose Open or Check out.



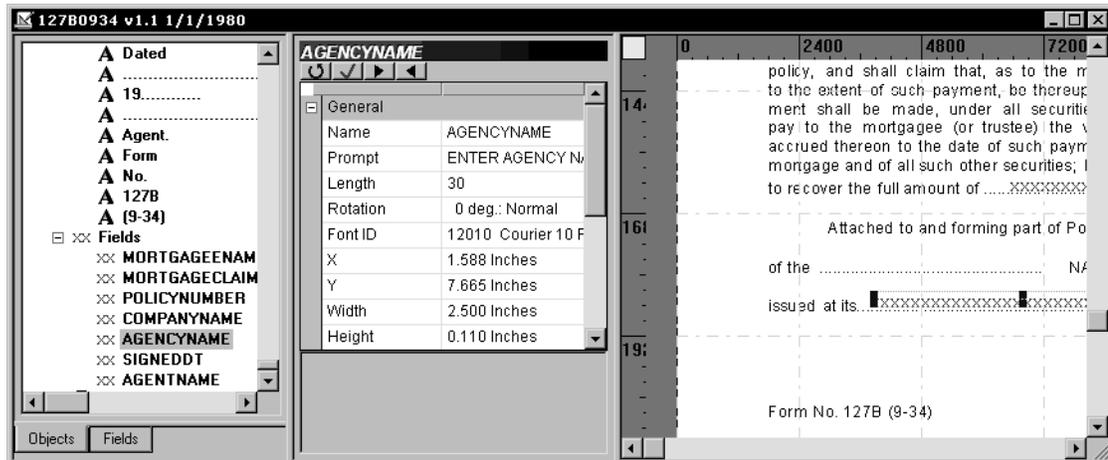
- 2 Highlight the section you want to edit or view and click Ok. The Section View window appears.

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.

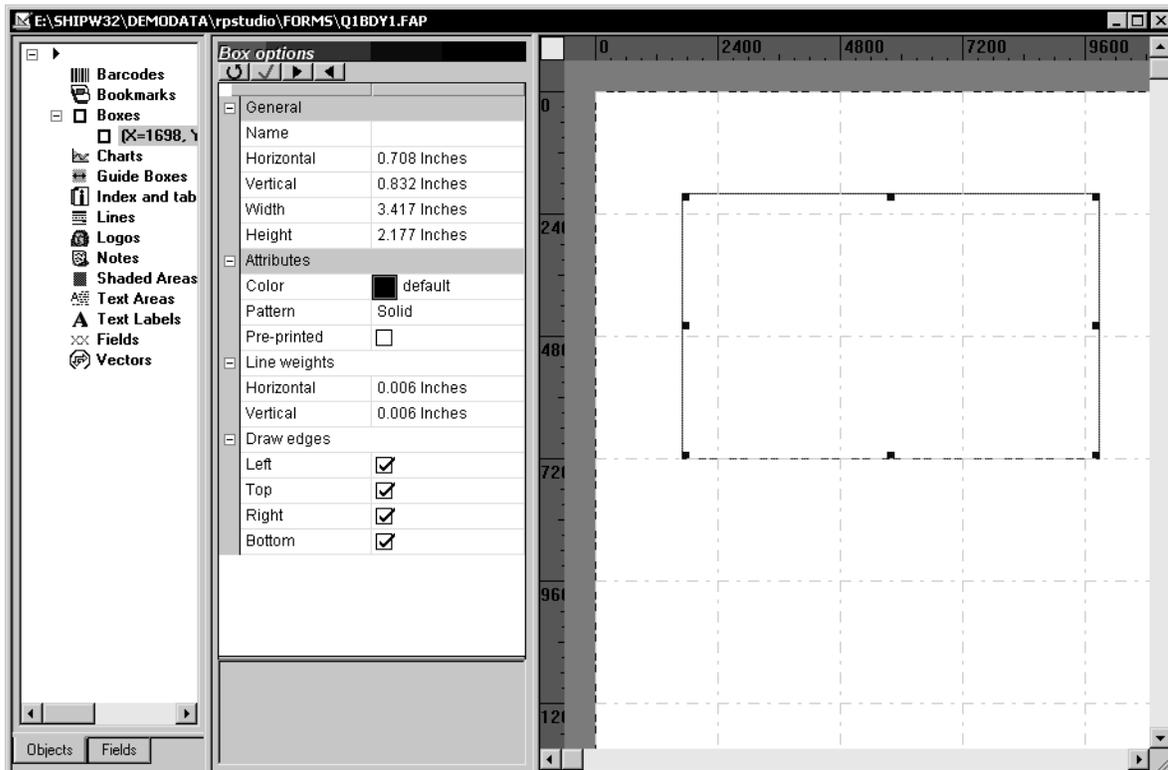


The Section View is similar to the Form View in that it shows you a representation of the section and provides a property panel when attributes about the various section objects can be shown.

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:

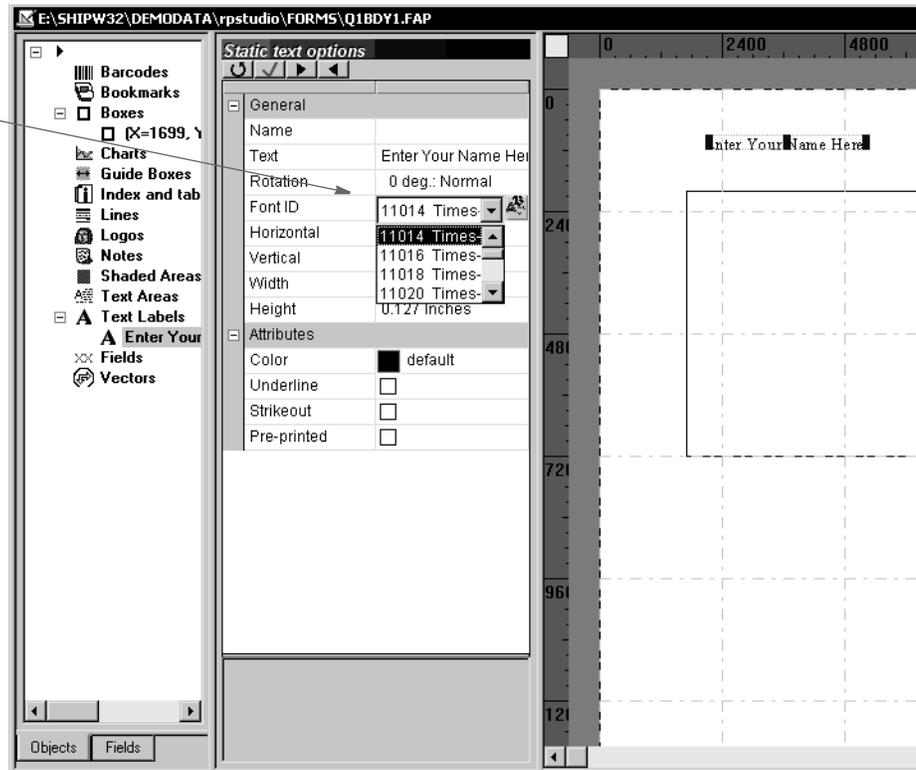


Here is an example of how you edit box properties:



Here is 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.



The property panel also shows you section properties.

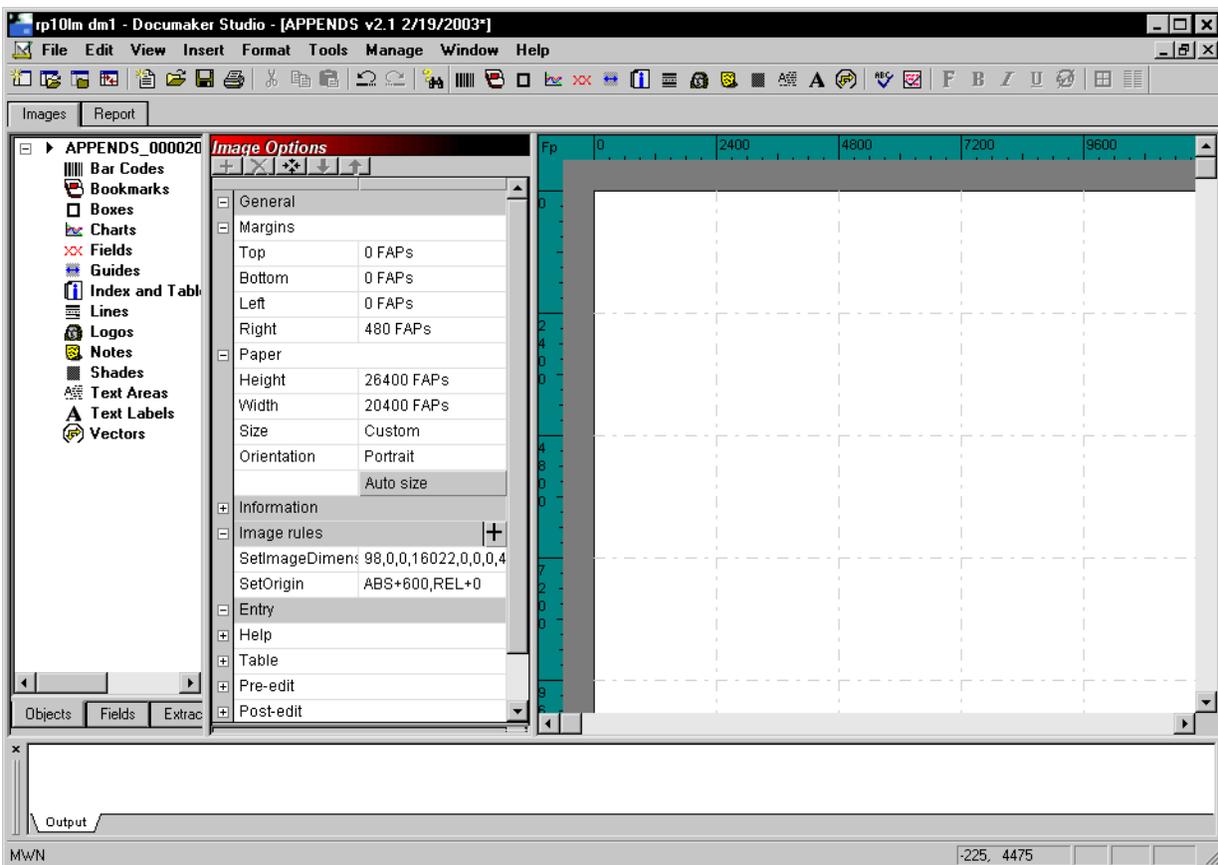
Creating new fields and other objects on your section is as simple as clicking on the menu or toolbar to select what type of item you want to create and then using the mouse to designate the location on the section.

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.

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

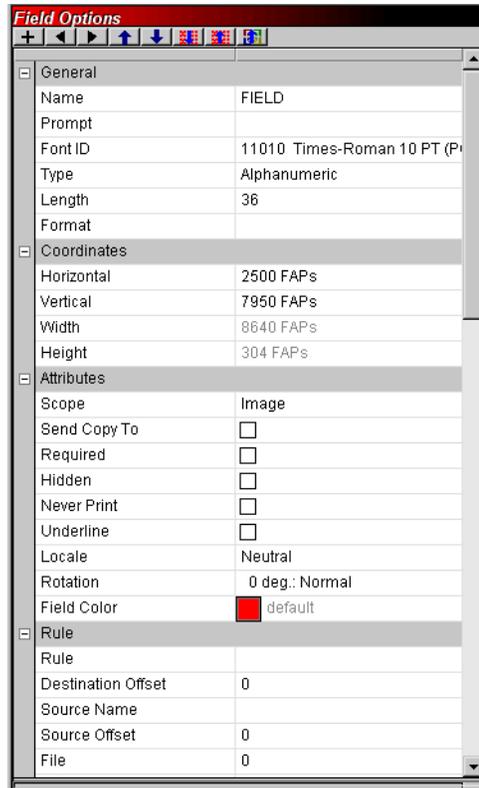
Use these icons:

NOTE: These tool bar icons only appear when you are working with sections.



To	Click
Insert a bar code	
Insert a bookmark	
Insert a box	
Insert a chart	
Insert a field	
Insert a guide*	
Insert an index or table	
Insert a line	
Insert a graphic	
Insert a note*	
Insert a shaded area	
Insert a text area	
Insert a text label	
Insert a vector drawing	

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:



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

Data Matrix Bar Code

Symbol Size	Regions	Max Numeric	Data Code Words
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
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 new fonts are listed below and referenced in the new font cross-reference (FXR) files (rel113.fxr and rel113sm.fxr) included in this release. 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:

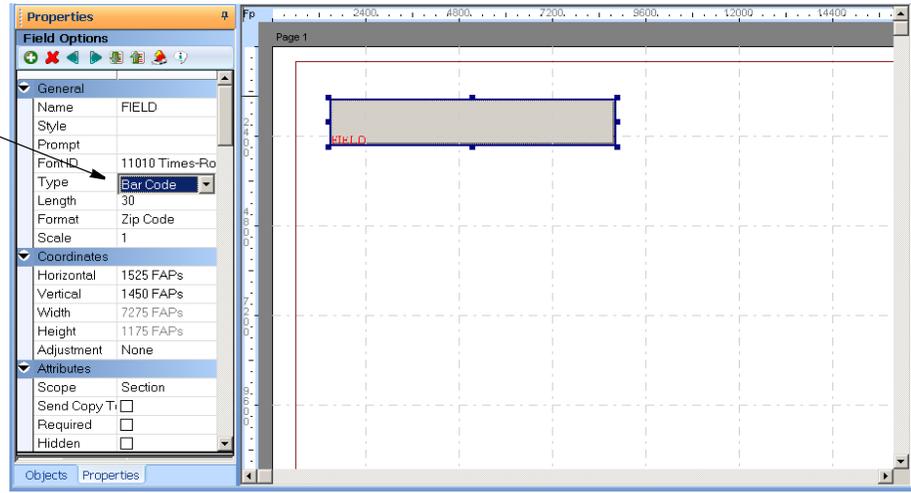
Size	Font name
TrueType	
All sizes	dm____.ttf
Postscript	
All sizes	dm____.pfb
PCL	
4 point	fpdmn4.pcl
5 point	fpdmn5.pcl
6 point	fpdmn6.pcl
AFP 240 DPI	
4 point	x0dadmn4.fnt, c0fadmn4.240 (Coded Font, Character Set)
5 point	x0dadmn5.fnt, c0fadmn5.240
6 point	x0dadmn6.fnt, c0fadmn6.240
AFP 300 DPI	
4 point	x0dadmn4.fnt, c0fadmn4.300 (Coded Font, Character Set)
5 point	x0dadmn5.fnt, c0fadmn5.300
6 point	x0dadmn6.fnt, c0fadmn6.300
Metacode	
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:

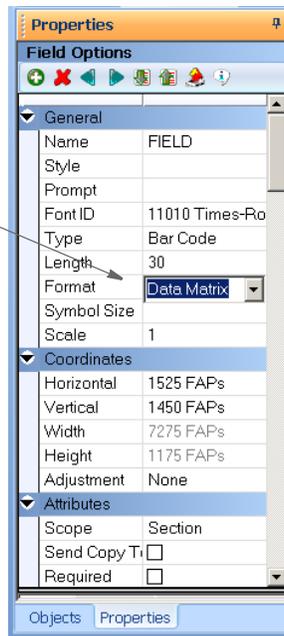
- 1 Insert a field in a section. For the field type, choose Bar Code.

Choose Bar Code as the field type.



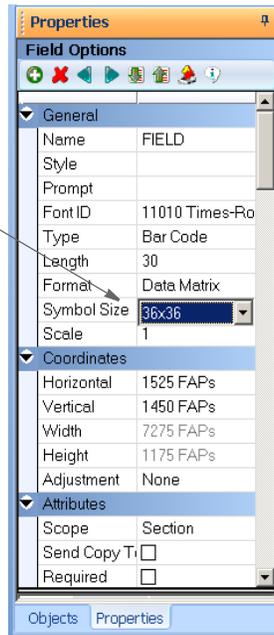
2 Choose Data Matrix as the field format.

Choose Data Matrix as the format.



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 PRINTCOMMANDER TO CREATE FAP FILES

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 **Tip** To quickly find the latest patchsets go to Quicklinks.

Results

Patch	Description	Release	Updated	Size		
*** No search conducted ***						

- 4 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

- 5 Click View Readme to read information about this patch. Click Download to download the patch.
- 6 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`
- 7 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:

- 1 From the Start menu, select Settings, then click on Printers. Next, double-click on the Add Printer icon.
- 2 Select My Computer, then click Next.
- 3 Select DPT1: (DocuCorp Port) from the available ports, then click Next.
- 4 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.
- 5 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.

- 6 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:

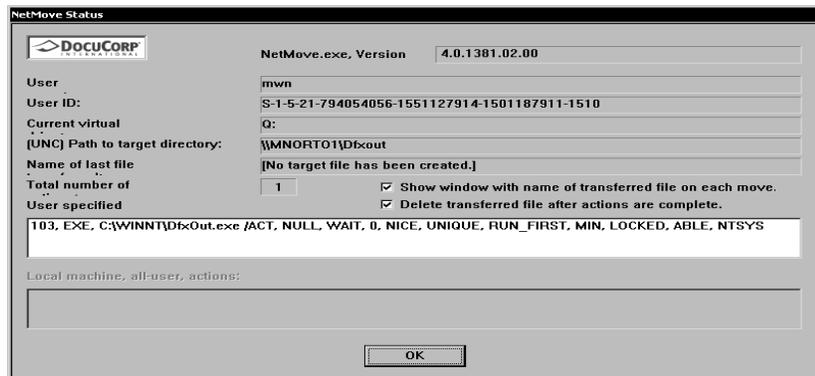
- 1 Hold down the CTRL and SHIFT keys and left click on the green Docucreate Output Control icon in the system tray on the bottom right hand side of your screen.

Click here to start Docucreate's Output Control



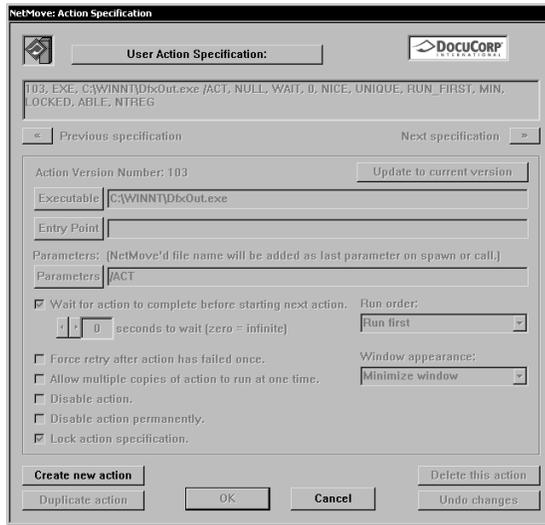
The NetMove Status window appears.

Here is a list of user-specified actions.



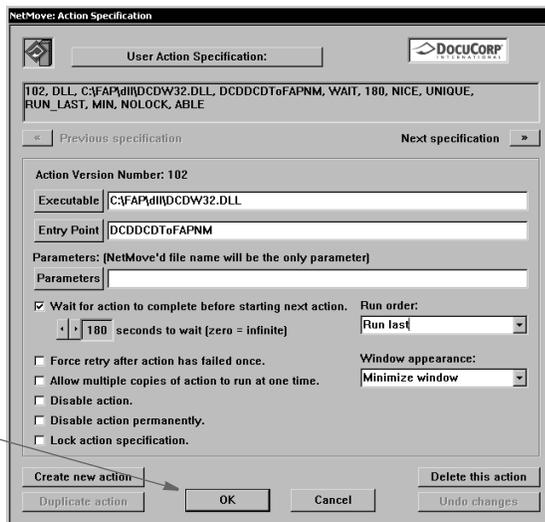
- 2 Double-click in the box which displays the list of user-specified actions. The Action Specification window appears. On the Action Specification window, click the Create New Action button.

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:
DCDDCDTofAPNM
- 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.



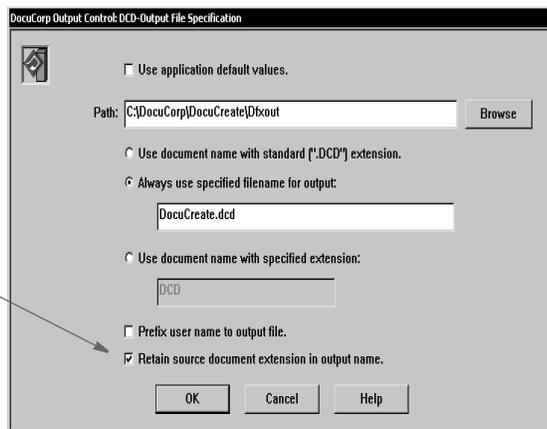
Then click Ok on the Action Specification window and on the NetMove Status window.

- 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.

In Studio or Image Editor, use the Format, Convert to Text Label option to combine the individual text labels into a single label.

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.

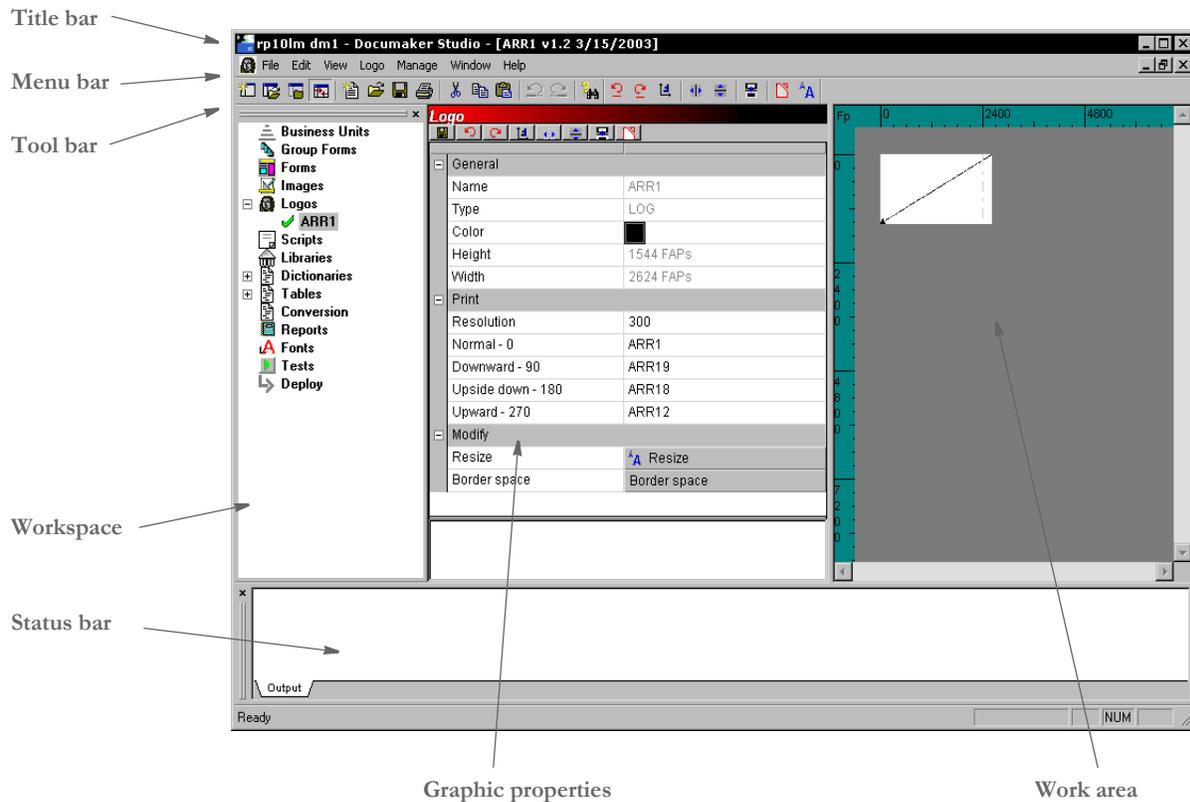
This chapter discusses these topics:

- [Using the Screen on page 202](#)
- [Managing Graphics on page 210](#)

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
Title bar	The title bar displays the name of the workspace you have open, followed by Documaker Studio, and then the name of the graphic you have open.
Menu bar	The menu bar provides the list of available pull-down menus.
Tool bar	The tool bar 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.
Graphic properties	Here Studio shows you the properties for the graphic you are working on.

Item	Description
Work area	This is where you work with the graphic.

USING THE MENU BAR

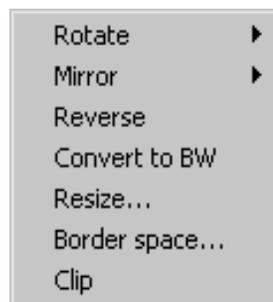
This section introduces you to the pull-down menus which include additional options or are only available when you are working with graphics.

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.

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:

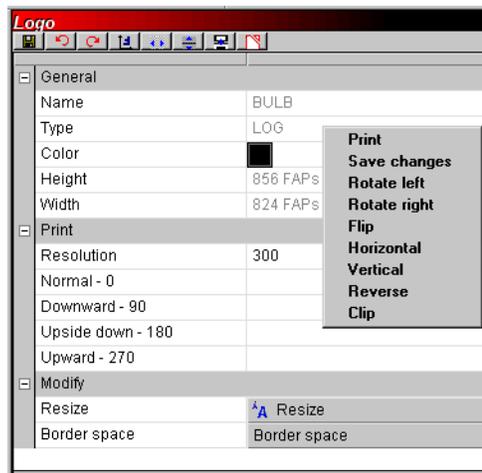


Option	Description	On right click?
Rotate		Yes
Left	Rotates the graphic 90 degrees in a clockwise motion.	Yes
Right	Rotates the graphic 90 degrees in a counter-clockwise motion.	Yes
Flip	Rotates the graphic 180 degrees.	Yes
Mirror		Yes
Horizontal	Lets you create a mirror image of the graphic by flipping it horizontally.	Yes

Option	Description	On right click?
Vertical	Lets you create a mirror image of the graphic by flipping it vertically.	Yes
Reverse	Creates a negative image of the graphic file, where dark becomes light and vice versa.	Yes
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.	Yes
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).	Yes
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).	Yes
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.	Yes

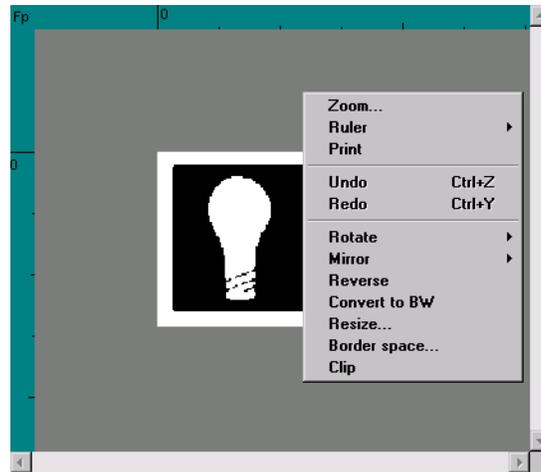
USING THE RIGHT-CLICK MENU

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.
Save changes	Saves the changes you have made
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.

When working with the actual graphic, you can right click to see the following menu.



Option	Description
Zoom	Lets you see a larger or smaller version of the graphic.
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 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.
Mirror	
Horizontal	Lets you create a mirror image of the graphic by flipping it horizontally.

Option	Description
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 TOOL BAR

The tool bar provides a quicker way to select options that may be listed on a drop down menu. Here is an example of the tool bar shown when you are working with graphics:



Standard tool bar icons

Shown below are the tool bar 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

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

Graphic tool bar icons

Shown below are the tool bar 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.

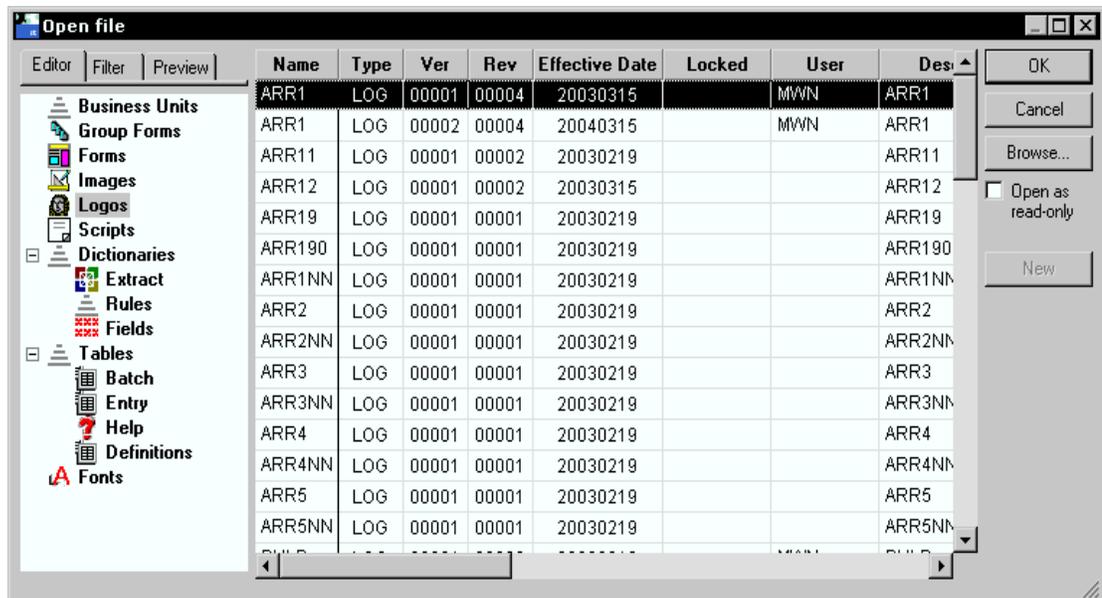
MANAGING GRAPHICS

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 Docucorp 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 open the following types of files:

- Logo files (*.LOG)
- Bitmap files (*.BMP)
- Xerox font files (*.FNT)
- Xerox font data files (*.FNT)
- Xerox image files (*.IMG)
- Xerox logo files (LGO)
- PCX files (*.PCX)
- TIFF (Tagged Image File Format) files (*.TIF)
- Printer overlay files (*.OVL)
- AFP page segment files (*.SEG)

- JPEG (Joint Photographic Experts Group) files (*.JPG)
- IBM MMR IOCA (Image Object Content Architecture) files (*.ICA)
- Portable Network Graphics files (*.PNG)
- Windows Meta files (*.WMF)

NOTE: 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 MVS 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 Edit, Save As, option to convert a graphic to one of the following file types:

- Logo files (*.LOG)
- Comp. Pack (*.LOG)
- Comp. TIFF (*.LOG)
- Bitmap (*.BMP)
- Xerox font (*.FNT)
- Xerox image (*.IMG)
- 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. The Open File window appears.

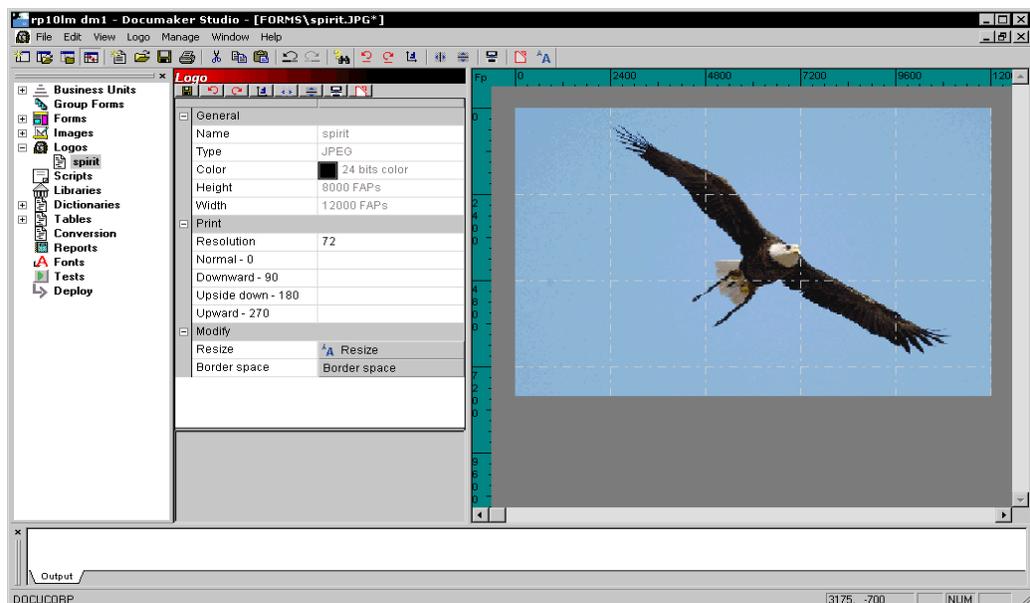


NOTE: This option specifies the directory for storing graphics:

```
< MasterResource >
  LogoLib =
```

If you omit this option, Studio looks for graphics in the path defined for FormLib.

- 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

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.

Reverse The Reverse option creates a *negative* of the graphic by taking the open graphic and switching the black and white colors.

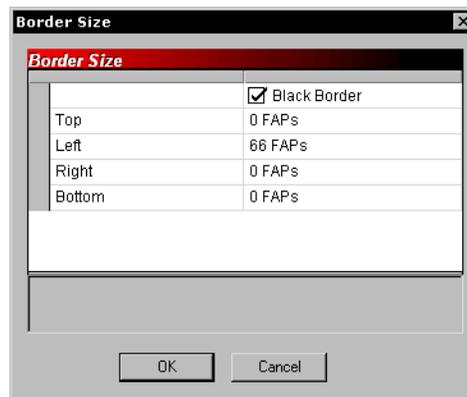
Convert to BW Use the Convert to BW option to convert a graphic to black and white. Reducing the number of colors typically reduces the size of the graphic. The system displays the Color Selection window when you choose this option. If you select Print in Color, the system prints the graphic in the color you selected.

Resize The Resize option gives you two choices for changing the size of the graphic you are working with: absolute and by percentage. The Absolute option lets you set the height and width in inches. If you choose the Absolute option, your graphic will not necessarily retain its current proportions.

The Percentage option lets you increase or decrease the size of the graphic by the percentage you choose. If you choose this option the system retains the current proportions of the graphic.

Border space The Border Space option lets you add and remove white and black space around the graphic. The amount of space can be as precise as needed.

To add white space, simply uncheck the Black Border field and set all the fields to the specific amount of white space you need. If you then want a black border or frame around the graphic, check the Black Border field again and enter the amount or thickness of the border.



Clip The Clip option lets you crop the graphic. After you select the Clip option from the Edit menu, outline the portion of the graphic you want to keep by clicking and dragging the mouse. The selected area remains while everything outside the outline you dragged around the section is discarded.

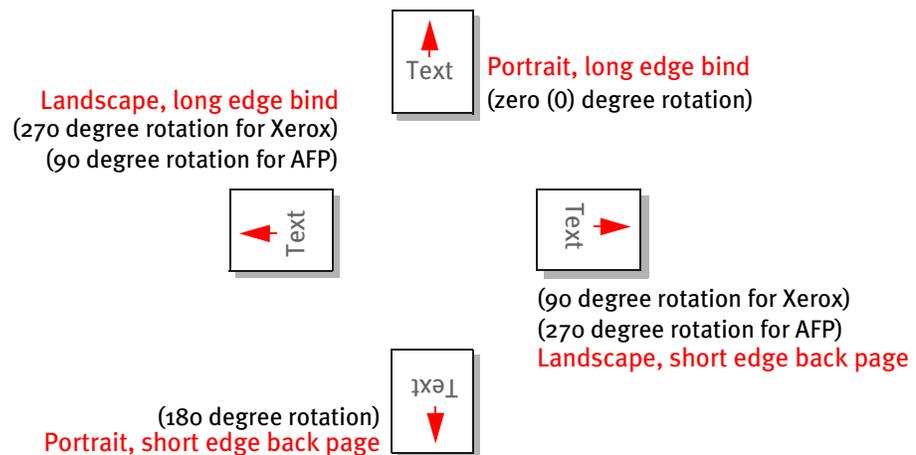
SETTING ROTATION NAMES

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,

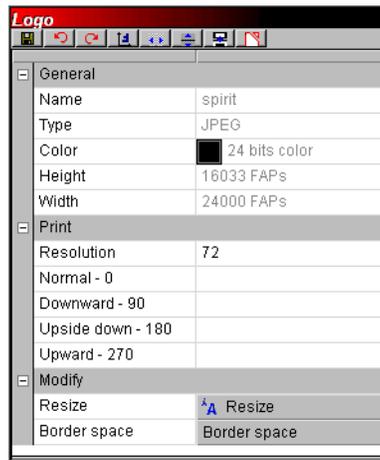
- Portrait, long edge bind requires a zero (0) degree (normal) rotation printer resource.
- Portrait, short edge back page requires a 180 degree (upside down) rotation printer resource.
- Landscape, long edge bind requires a 270 degree (upward) rotation printer resource for Xerox printers and a 90 degree 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.

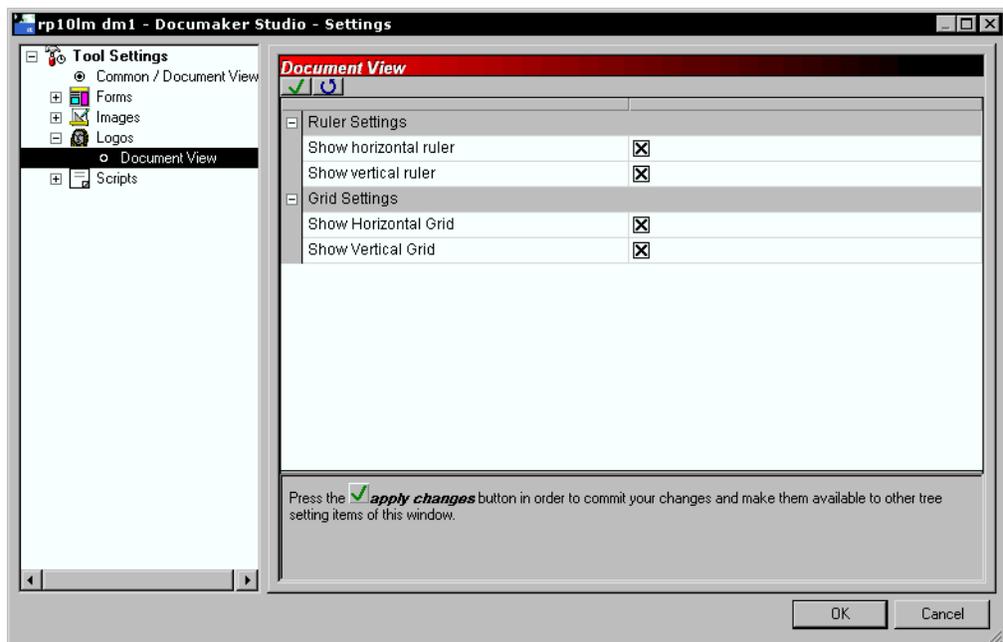
Changing the Resolution

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 select View, 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 the window 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.

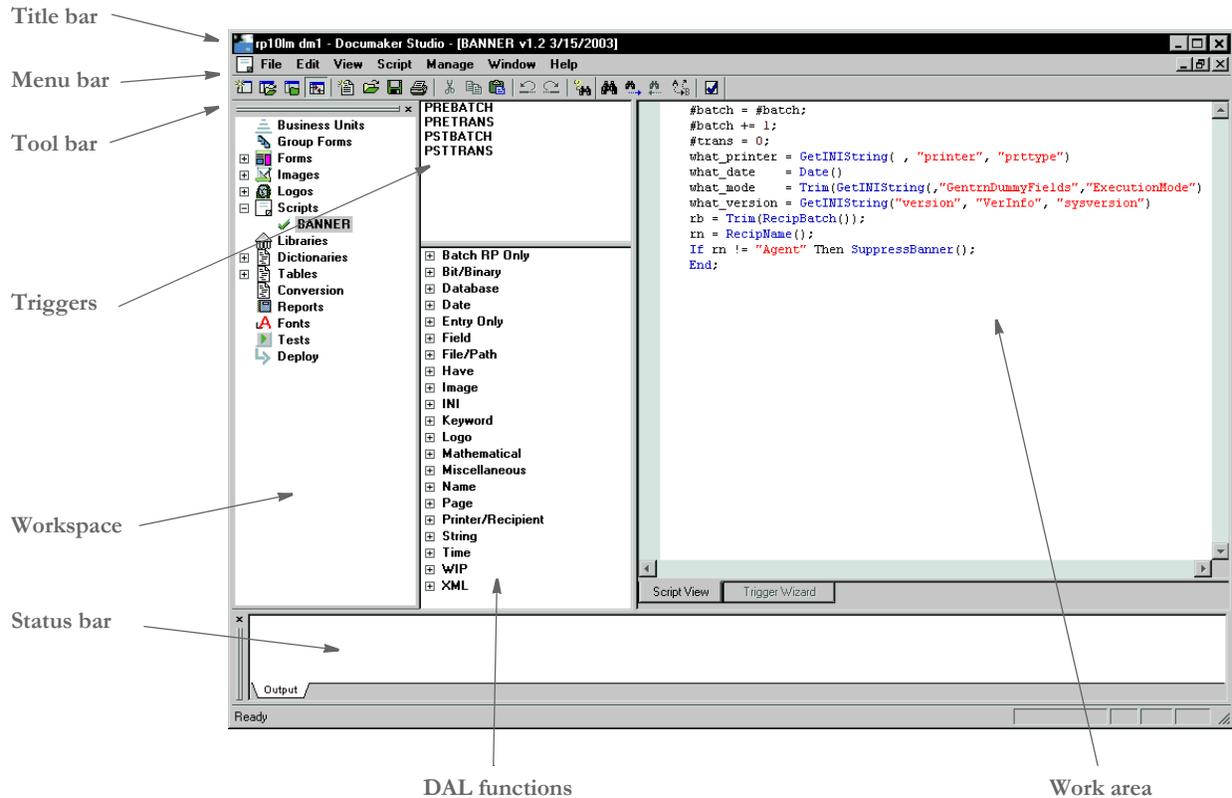
This chapter discusses this topics:

- [Using the Screen on page 220](#)
- [Overview on page 226](#)
- [Checking Syntax on page 228](#)

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.
Tool bar	The tool bar 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

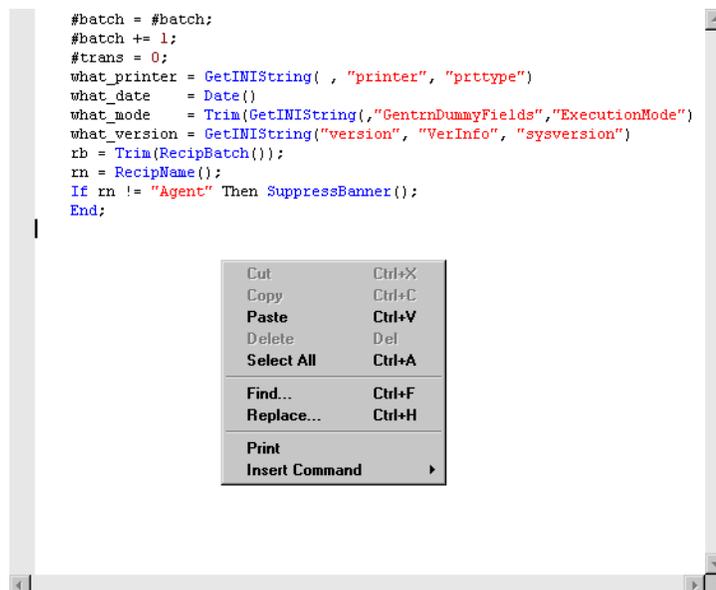
The Script menu provides you with options you can use as you work with DAL scripts. You can insert and delete functions or convert the file into a library, trigger file, or script. When you select Script, this menu appears:



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 TOOL BAR

The tool bar provides a quicker way to select options that may be listed on a drop down menu. Here is an example of the tool bar shown when you are working with scripts:



Standard tool bar icons

Shown below are the tool bar 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

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

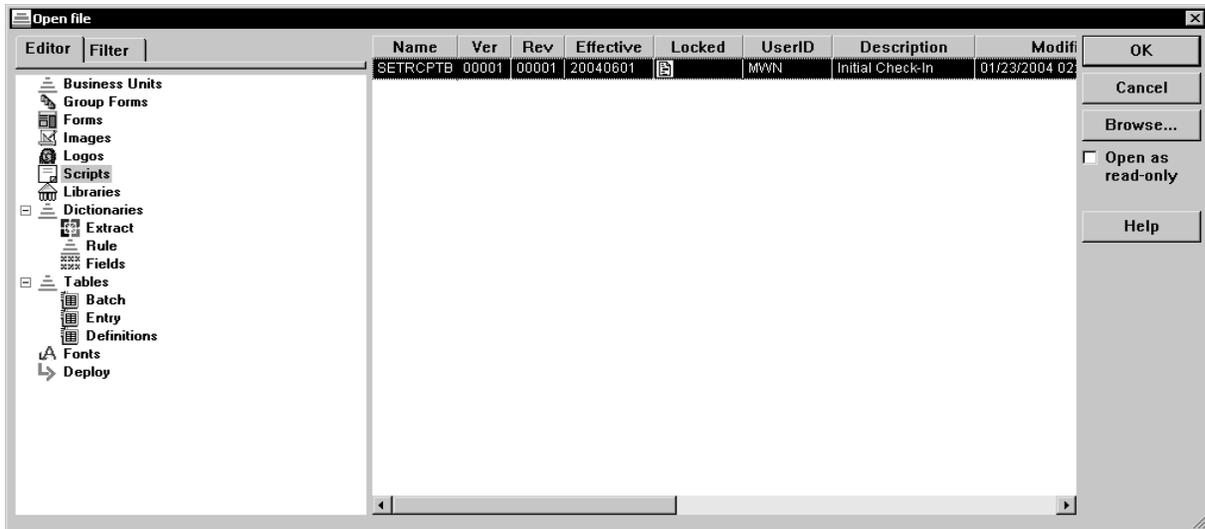
Script tool bar icons

Shown below are the tool bar 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.

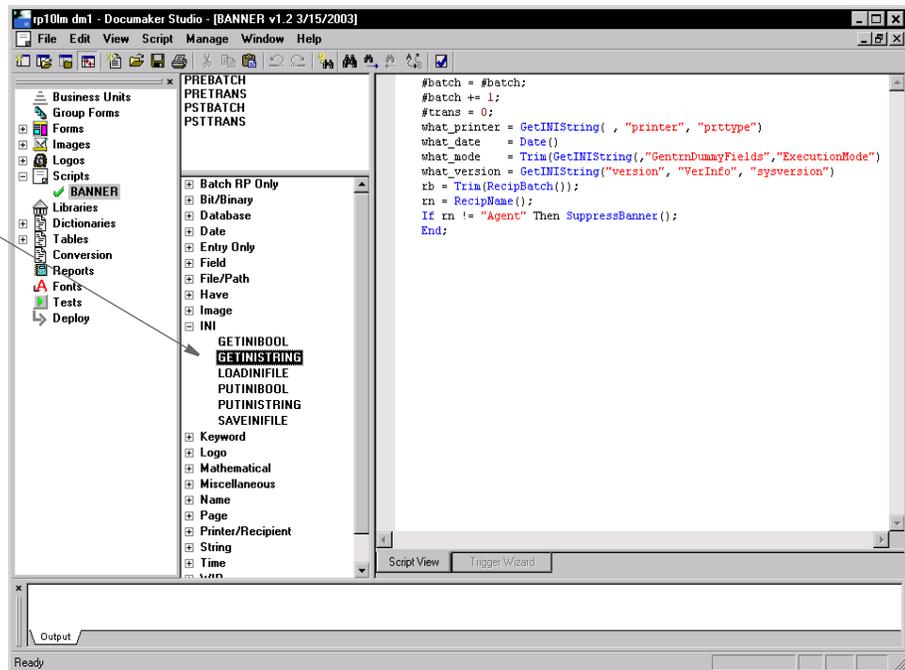
OVERVIEW Here is an example of the window that appears:



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.



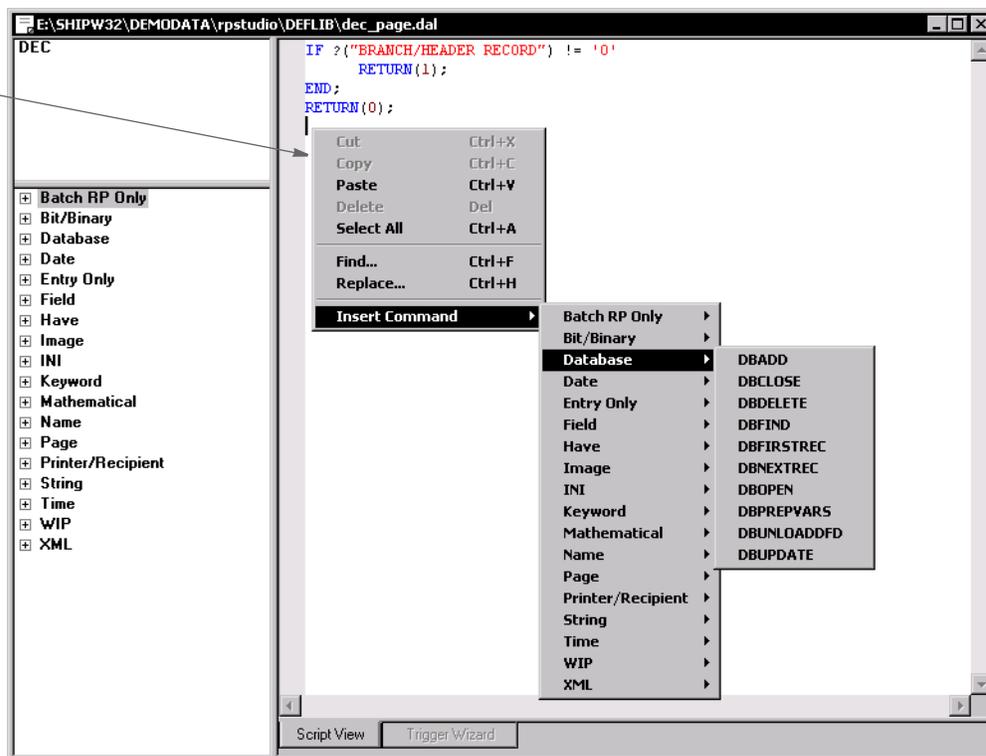
When editing a DAL script, you can right click to display a menu of editing options. This menu provides another way to insert a DAL function or procedure, as shown below:

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

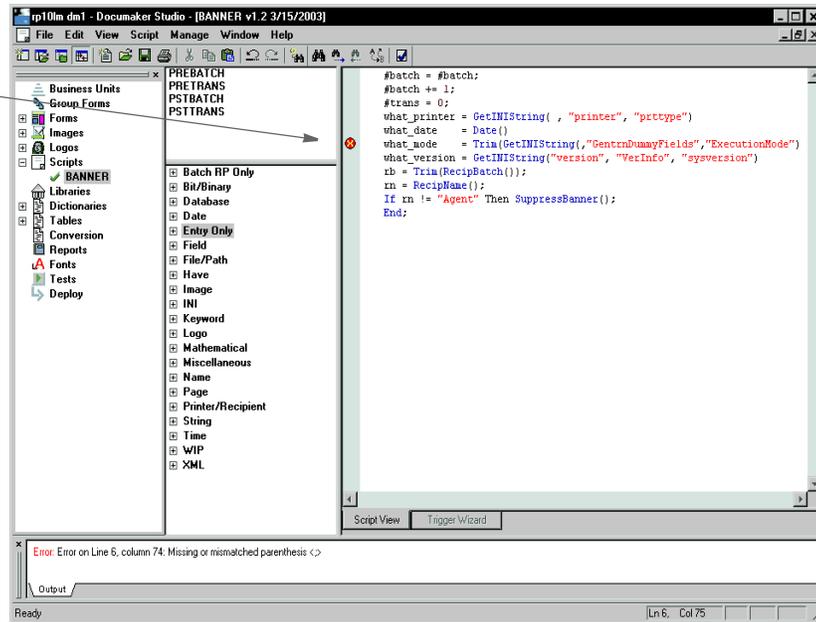
When creating a DAL script, you can easily check for syntax errors by choosing the Edit, Check Syntax option.

NOTE: You can also check for syntax errors by clicking this icon on the tool bar: 

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.

Error	Description
Open failure on script file	<p>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.</p> <p>You would typically not receive this error during a syntax check.</p>
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, <code>AVAR=BVAR</code> is correct syntax; using <code>AVAR==BVAR</code> would generate the invalid assignment statement message because the operator <code>==</code> is unknown. <code>AVAR BVAR</code> (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 <code>1 = AVAR</code>. 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>

Error	Description
Unexpected internal error	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. Typically, this message indicates you have an expression or function parameters that do not resolve to an expected state.
Missing or mismatched parenthesis	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.
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.

Error	Description
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.
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 data-bbox="818 1037 1073 1087">GOTO BOB (incorrect) GOTO BOB: (correct)</pre>
Invalid statement label	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.
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>

Error	Description
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 234](#)
- [Creating Libraries on page 251](#)
- [Working with Libraries on page 281](#)
- [Using the LBYPROC Utility on page 319](#)
- [Troubleshooting on page 322](#)

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 235](#)
- [Concepts on page 237](#)
- [How It All Works on page 245](#)
- [Managing Workflow on page 250](#)

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 <i>00001</i> .
<hr/> NOTE: When a resource is expired, the revision field is set to <i>EXP</i> . <hr/>	
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 237](#)
- [Processing with Effective Dates on page 239](#)
- [Retrieving Resources with Version and Revision Numbers on page 240](#)
- [Stringently Checking Resources on page 241](#)
- [Understanding Run Dates on page 243](#)

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 <i>LOC</i> , otherwise it is blank.

Field	Description
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.
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 Defining Mode, Status, Class, and Project Options on page 317 .
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 Defining Mode, Status, Class, and Project Options on page 317 .
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 Defining Mode, Status, Class, and Project Options on page 317 .
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 Defining Mode, Status, Class, and Project Options on page 317 .

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, Documanager), 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, RCBDFFDFL.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, RCBDFFDFL.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 RCBDFDFL.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 DPRRRetrieveFormset. 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 243](#) 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 252](#)
- [Using the DB2 Native Driver on page 254](#)
- [Using the DB2 ODBC Driver on page 259](#)
- [Using the SQL Server ODBC Driver on page 265](#)
- [Using the Oracle ODBC Driver on page 269](#)
- [Using Documanage on page 274](#)

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:LBXI >
  Catalog      = LBXC
< DBTable:LBXC >
  DBHandler    = DB2
< DB2_FileConvert >
  Catalog      = DAP112_LBXC_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 <i>MASTER.LBY</i> .

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>..\mstrres\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>LBYLOG</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 *LBYD* 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:LBYD >
DBHandler = DB2
DFD = DEFLIB\LBYD_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 OS390 On OS390, 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> .

Option	Description
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> have been chosen to 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 cannot exceed eight characters. See the <i>DB2_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>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 <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 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

Option	Description
DBHandler	This option tells the system to access the table known as <i>CATALOG</i> using the database handler named <i>DB2</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 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 <i>Class</i> option is unnecessary if the name of the handler (<i>DB2</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> .
CreateIndex	This option tells the system not to create database indexes. This option should <i>always</i> be set to <i>No</i> .
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 <i>Debug</i> option or set it to <i>No</i> . In troubleshooting situations, set this option to <i>Yes</i> 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 <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.
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 [Docuoolbox 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:

```
..rpx1\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> .

Option	Description
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> have been chosen to 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> . 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: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> . 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: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 option, the system expects to find a 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 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 Class 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> .
CreateIndex	This option tells the system not to create database indexes. This option should <i>always</i> be set to <i>No</i> .
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 <i>No</i> . In troubleshooting situations, set this option to <i>Yes</i> 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.
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.

Option	Description
LBYLog	This option tells the system the table referenced in INI options as <i>LBYT</i> is really named <i>DAP110_LBYT</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> .
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> .

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 within the table, you receive warning messages indicating no unique tag is defined. Include this option to avoid those warning messages.

DBTable:LBLOG control group

DBHandler	This option tells the system to access the table known as <i>LBLOG</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

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>LBYDBASE</i> . If you omit this option, the database set up as the default database for the <i>LBYSQL</i> ODBC data source is used.

Option	Description
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

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.
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 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 a Docucorp-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.
------------	---

Option **Description**

Library:LBYI control group

DBTable	<p>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.</p> <p>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.</p>
---------	---

DBTable:LBYI control group

DBHandler	<p>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>.</p> <p>Microsoft's SQL Server is an ODBC-compliant database.</p>
-----------	---

DBTable:LBYD control group

DBHandler	<p>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>.</p>
UniqueTag	<p>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.</p> <p>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 id defined. Include this option to avoid those warning messages.</p>

DBTable:LBYLOG control group

DBHandler	<p>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>.</p>
-----------	--

DBTable:CATALOG control group

DBHandler	<p>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>.</p>
UniqueTag	<p>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.</p> <p>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.</p>

DBHandler:ODBC control group

Class	<p>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 <i>DB2</i>, <i>ODBC</i>, <i>ORA</i>, or <i>DMS</i>.</p>
-------	--

Option	Description
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>LBYDATABASE</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

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.
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 DOCUMANAGE

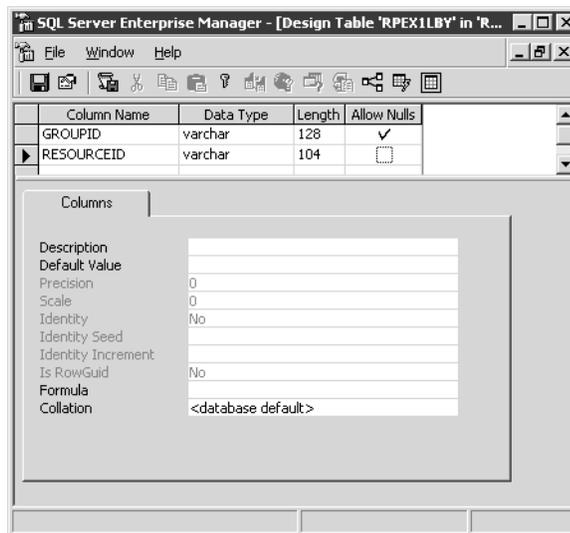
Studio supports installable interfaces to alternate document management systems (DMS). The default DMS is the DAP archive system, but you can also use Documanager. To use Documanager, follow these steps:

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.

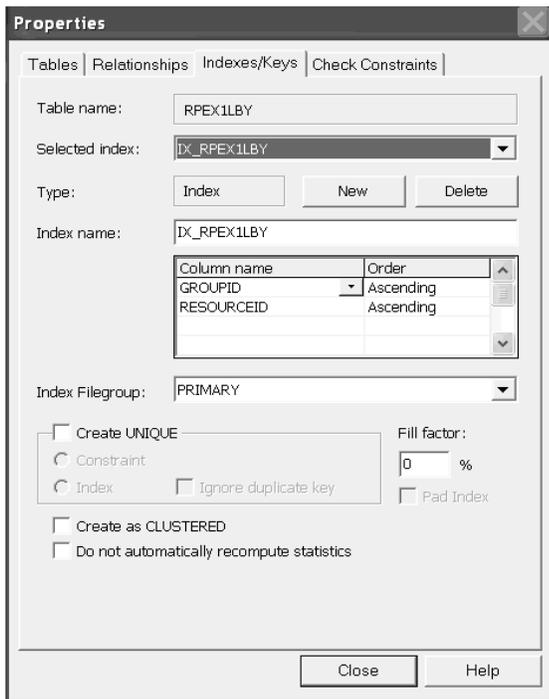
- 1 Create a table in an ODBC database. You can name the table anything you like. This example uses RPEX1LBY. Include in the table the following fields. All fields should be of the VARCHAR type:

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.

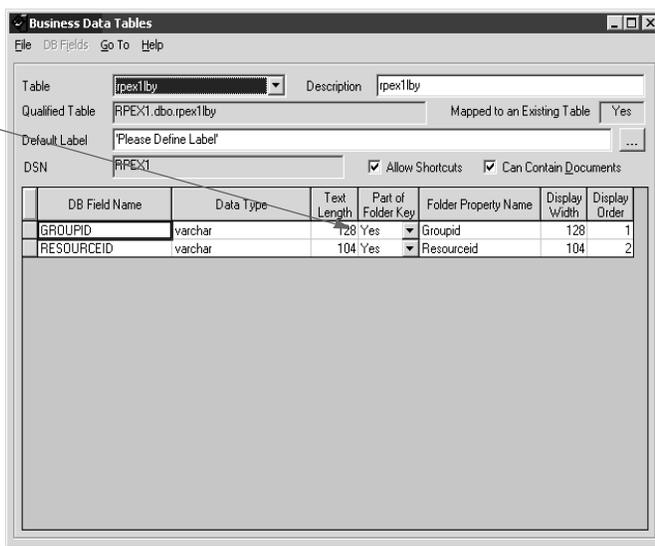


For performance reasons, it is best to add an index on the new table by GROUPID and RESOURCEID, as shown on the following window.



- 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.

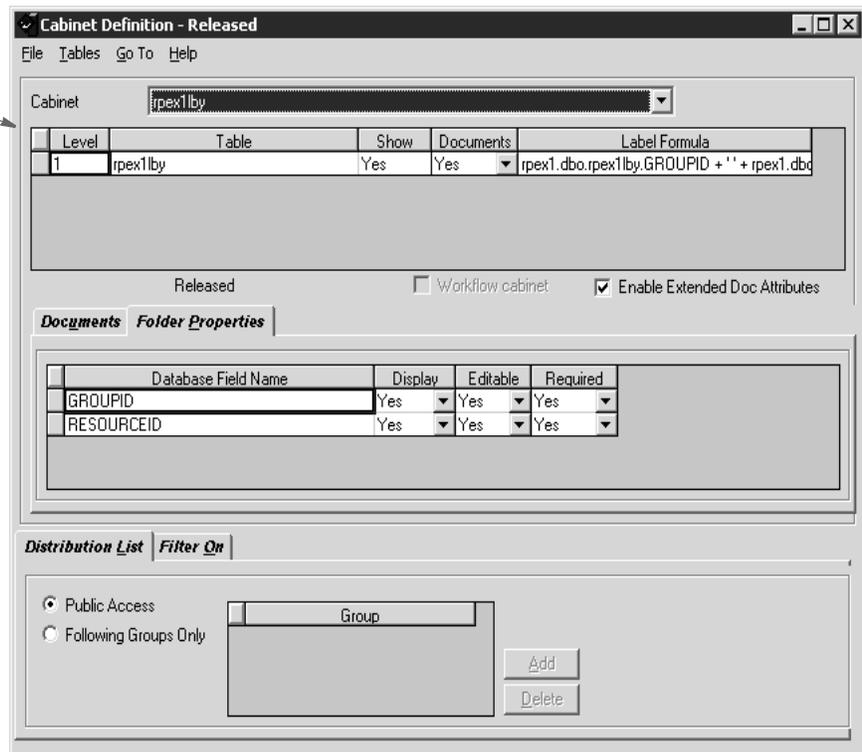
Make sure these values are set to Yes.



- 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

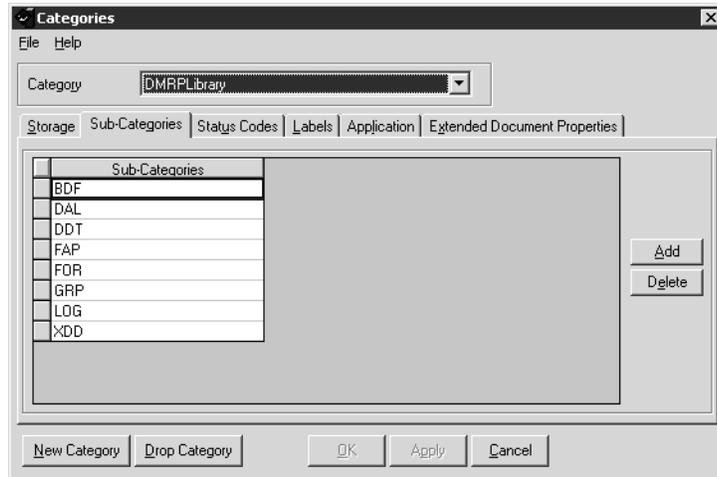


Once you define the cabinet, select Save, Release, and then Refresh Servers from the File menu.

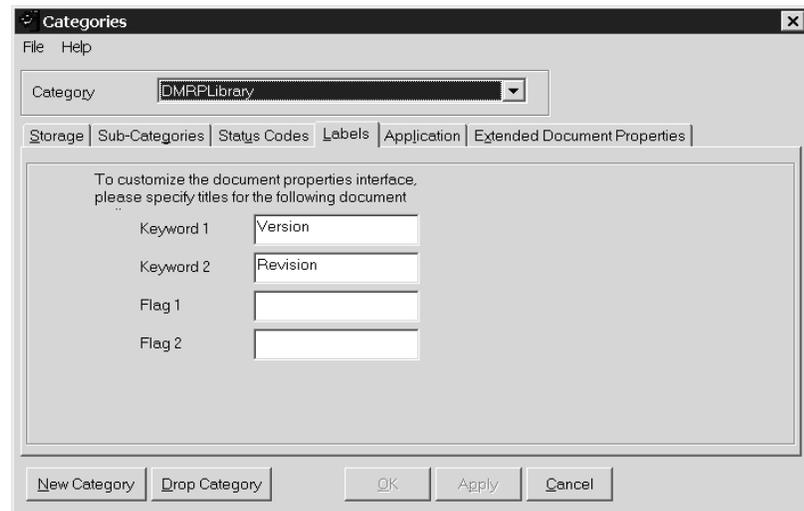
- 6 Define a category called *DMRPLibrary* using the Documanager Administrator. Set up a storage location under the Storage tab. This is usually *DEFAULT/Permanent* or *Temporary*. Under the Sub-Categories tab, add these sub-categories:
 - BDF
 - DAL
 - DDT
 - FAP
 - FOR
 - GRP
 - LOG

- 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.

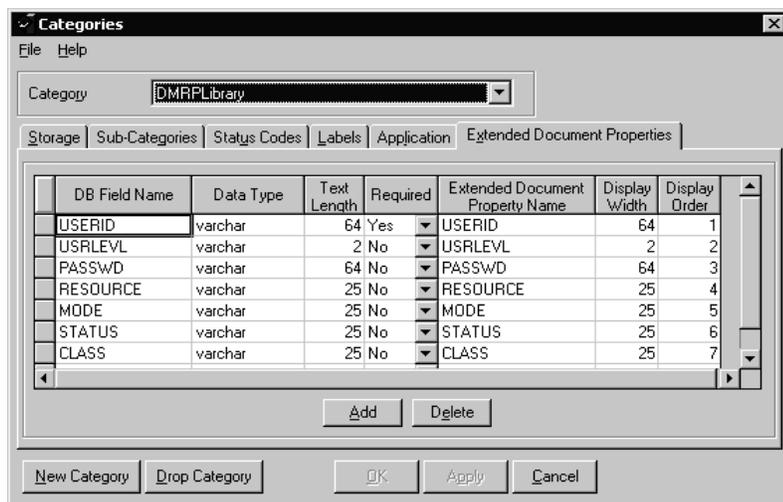


- 8 Next, select the category *DMRPLibrary*, click on the Extended Document Properties tab, and add the following information:

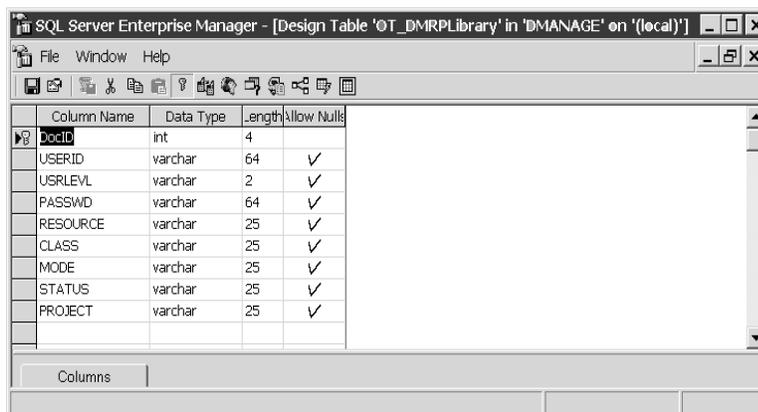
NOTE: The display order is not important.

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

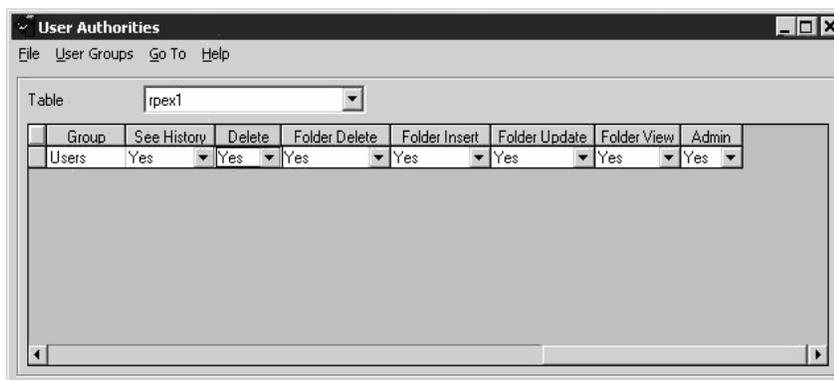
As shown here:



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 Documanager Administrator.



- 9 Load the test master resource libraries (MRLs) into the config RPEX1. Here are the necessary FAPCOMP.INI options for Documaker: (RPEX1LBY is the Documanager 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 316](#).

WORKING WITH LIBRARIES

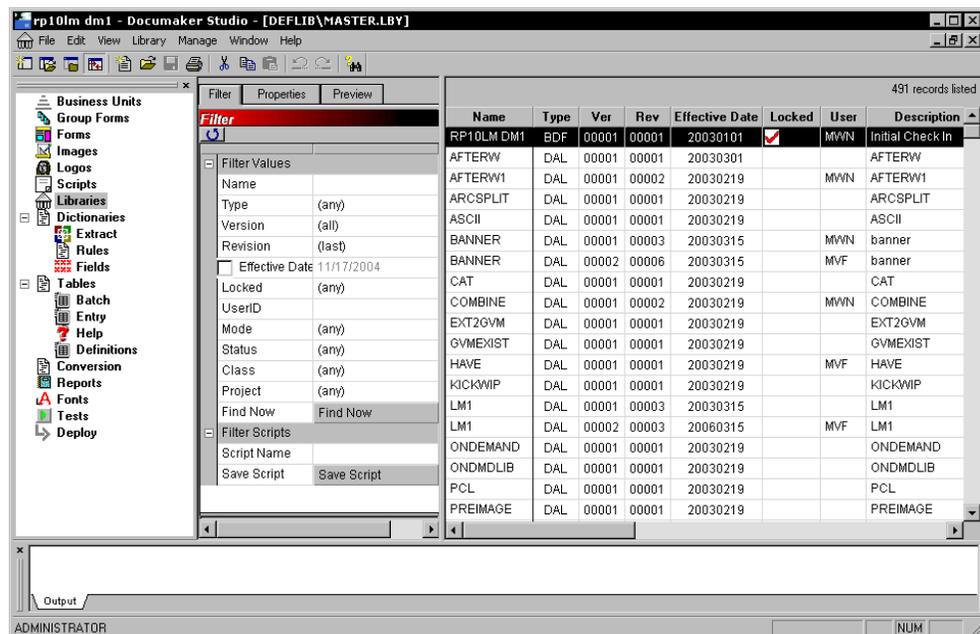
Studio lets you open a library, add resources to a library, delete, or extract resources from a library and edit some of the properties of a resource. You can also run a response file, print a list of resources, and view a history of changes made to a resource. For initialization settings, Studio uses 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 282](#)
- [Adding Resources to a Library on page 284](#)
- [Importing Files on page 287](#)
- [Importing Libraries on page 288](#)
- [Checking Out Resources on page 291](#)
- [Checking In Resources on page 293](#)
- [Unlocking Resources on page 295](#)
- [Promoting Resources on page 296](#)
- [Filtering Resources on page 300](#)
- [Editing Resource Information on page 302](#)
- [Deleting Resources on page 304](#)
- [Searching the Library on page 305](#)
- [Extracting Resources on page 309](#)
- [Expiring Resources on page 313](#)
- [Reviewing a History of Resource Changes on page 314](#)
- [Running Response Files on page 316](#)

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.

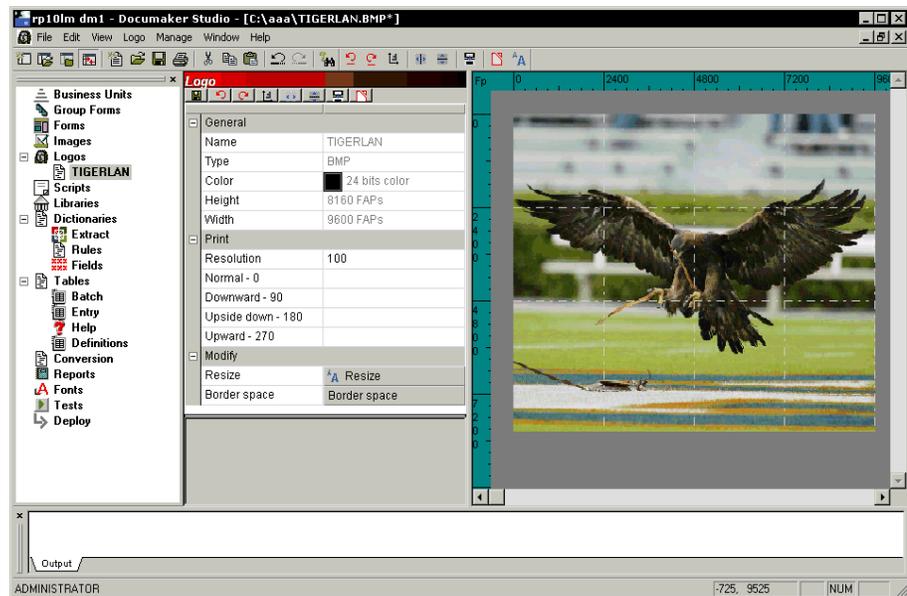
To	Do this
Save a copy of the resource to disk	Highlight the resource and right click. Then choose Extract. The Extract Library Resources window appears. From this window you can specify where to extract the resource and other options.
Print a listing of the resources	Right click in the list of resources 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.

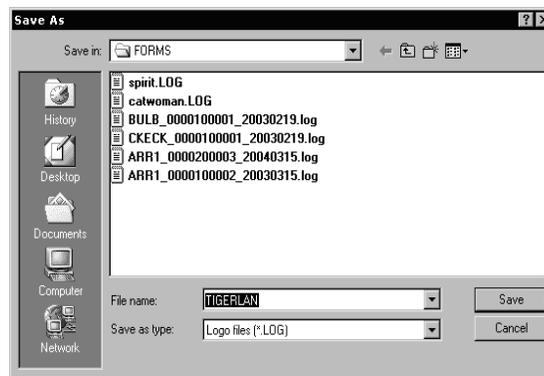
There are several ways to add a resource to your library. For instance, you can use the Library, Import Files options to add several files at once. See [Importing Files on page 287](#) for more information. You can also add a single resource to your library.

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:

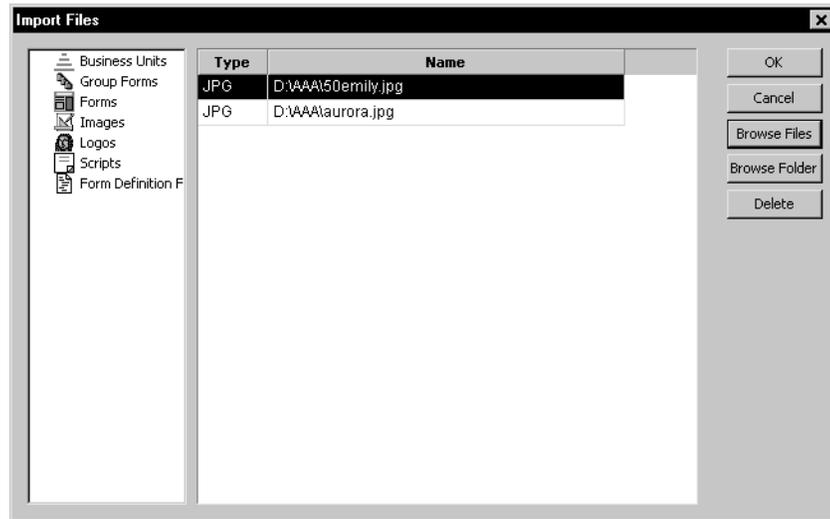
Field	Description
Description	<p>You can enter up to 100 characters to indicate what type of change was made to the resource. This field defaults to the contents of the Description field of the resource checked out, but you can enter a different description if you like.</p> <p>The Description field appears 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>

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. See Defining Mode, Status, Class, and Project Options on page 317 for information on how to set up modes.
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. See Defining Mode, Status, Class, and Project Options on page 317 for information on how to set up statuses.
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 Defining Mode, Status, Class, and Project Options on page 317 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 Defining Mode, Status, Class, and Project Options on page 317 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.

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 Documange.

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

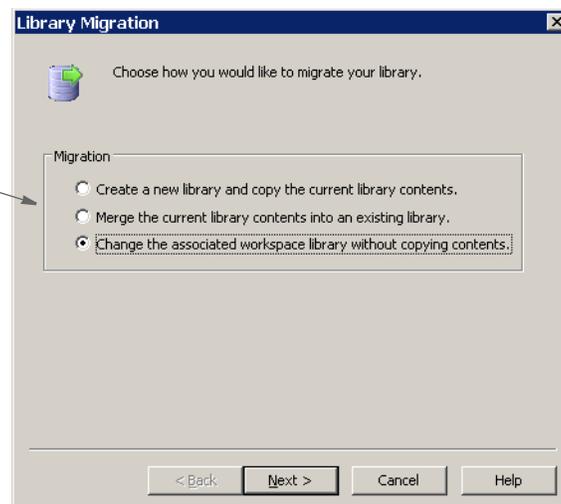
You can easily change the format of a library within a workspace. For instance, if you have administrator level security, you can use the migration wizard to quickly change a library from xBase to SQL or Documanager.

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

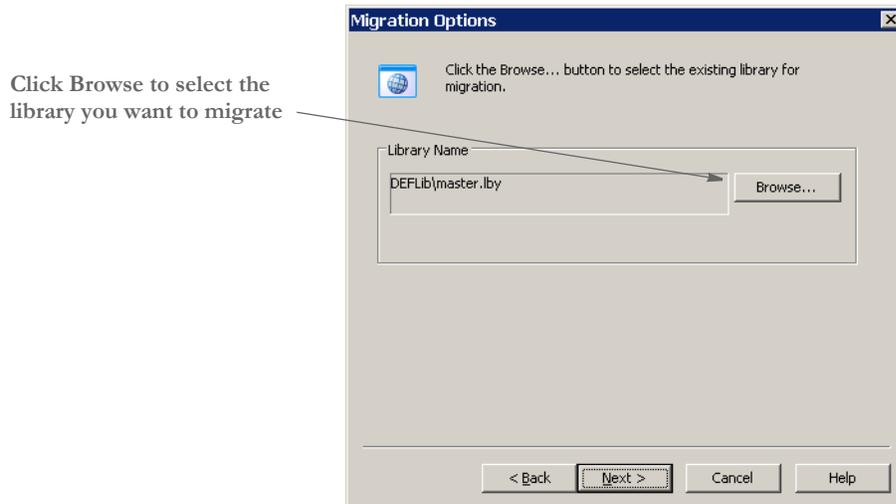


You can choose from these options:

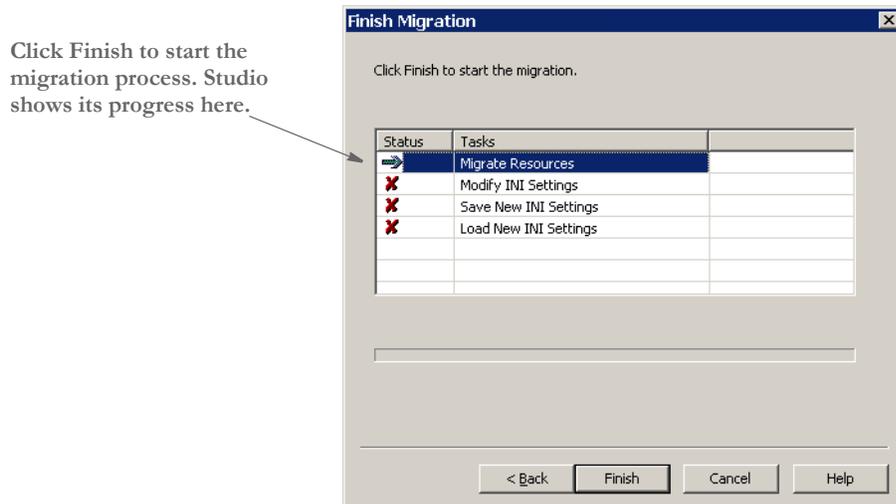
Migration option	Description
Create a new library and copy the current library contents	When creating a new library, you specify the name, location, and description. If you specify an existing library, Studio copies all of the resources into that library. If there are existing resources with the same 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.

Migration option	Description
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	Dct
102SMFNT	FAP	00001	00001	20030219		MVF	102SM
APPENDS	FAP	00001	00002	20030219		MWN	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	✓	MWN	BCKGI
BILLGAR	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.

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. 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>

Field	Description
Effective Date	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. 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. The effective date defaults to MM/DD/YYYY, but may have a different format depending on your locale setting.
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 Defining Mode, Status, Class, and Project Options on page 317 .
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 Defining Mode, Status, Class, and Project Options on page 317 .
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 Defining Mode, Status, Class, and Project Options on page 317 .
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 Defining Mode, Status, Class, and Project Options on page 317 .

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.

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		MVN	APPENDS 0E
ARRWDMY	FAP	00001	00001	20030219		MVF	ARRWDMY 0E
BASE1	FAP	00001	00001	20030219		MVF	BASE1 0E
BASE2	FAP	00001	00001	20030219	✓	ADMIN	BASE2 0E
BASE3	FAP	00001	00002	20030219		ADMIN	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	✓	MVN	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.

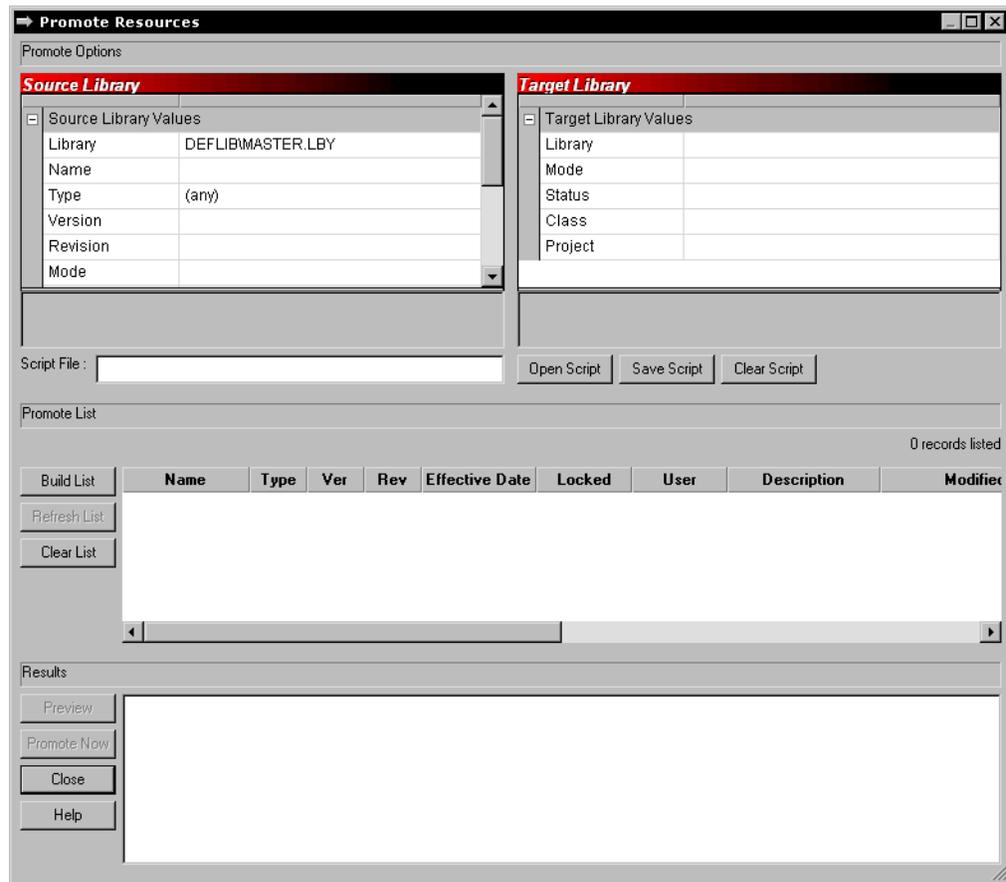
There are several ways to unlock a resource:

- 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>bri*</i> tells Studio to select all resources with names that start with <i>bri</i> .

Field	Description
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.
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 Defining Mode, Status, Class, and Project Options on page 317 .
Status	Select which status you want Studio to look for when selecting resources to promote. For more information, see Defining Mode, Status, Class, and Project Options on page 317 .
Class	Select which class you want Studio to look for when selecting resources to promote. For more information, see Defining Mode, Status, Class, and Project Options on page 317 .
Project	Select which project ID you want Studio to look for when selecting resources to promote. For more information, see Defining Mode, Status, Class, and Project Options on page 317 .

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



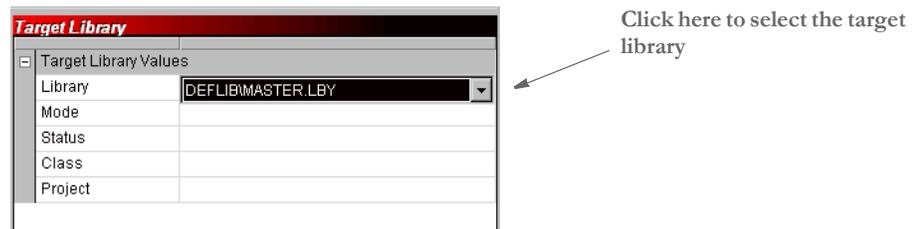
Field	Description
Mode	Select which mode you want Studio to assign after it moves a copy of the resource into the target library. For more information, see Defining Mode, Status, Class, and Project Options on page 317 .
Status	Select which status you want Studio to assign after it moves a copy of the resource into the target library. For more information, see Defining Mode, Status, Class, and Project Options on page 317 .
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 Defining Mode, Status, Class, and Project Options on page 317 .

Field	Description
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 Defining Mode, Status, Class, and Project Options on page 317 .

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 Defining Mode, Status, Class, and Project Options on page 317 .
Status	Select which status you want Studio to assign after it moves the resource into the target library. For more information, see Defining Mode, Status, Class, and Project Options on page 317 .
Class	Select which class you want Studio to assign after it moves the resource into the target library. For more information, see Defining Mode, Status, Class, and Project Options on page 317 .
Project	Select which project ID you want Studio to assign after it moves the resource into the target library. For more information, see Defining Mode, Status, Class, and Project Options on page 317 .

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

Promote List

23 records listed

Name	Type	Ver	Rev	Effective Date	Locked	User	Description	Modi
ARR1	LOG	00001	00001	20030315			ARR1	05/27/2004 C
ARR1	LOG	00001	00002	20030315		MWN	ARR1	05/27/2004 C
ARR1	LOG	00001	00003	20030315		MWN	ARR1	08/03/2004 C
ARR1	LOG	00001	00004	20030315		MWN	ARR1	10/19/2004 C
ARR1	LOG	00002	00001	20040315			ARR1	05/27/2004 C

Results

Preview

Promote Now

Close

Help

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.

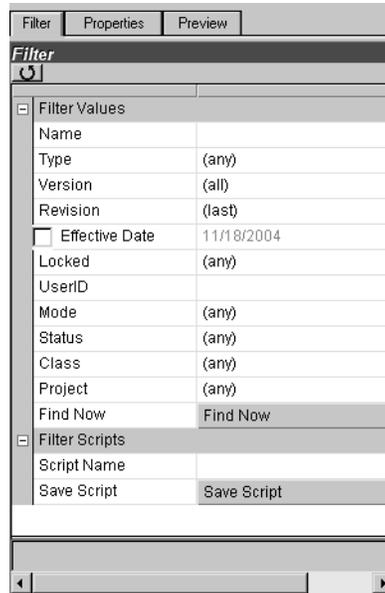
NOTE: At this point, no changes have been made. Preview simply shows you what the result of your promotion script would be if you actually ran it.

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.

Field	Description
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.

These prompts guide you through the fields.

Click Update Resource to save your changes.

Resource Properties	
Name	COMBINE
Type	DAL
Version	00001
Revision	00002
Effective Date	02/19/2003
Description	COMBINE
UserID	MWN
Mode	(blank)
Status	(blank)
Class	(blank)
Project	(blank)

The version of the resource. You cannot change the version.

- 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 Defining Mode, Status, Class, and Project Options on page 317.

Field	Description
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 Defining Mode, Status, Class, and Project Options on page 317 .
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 Defining Mode, Status, Class, and Project Options on page 317 .
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 Defining Mode, Status, Class, and Project Options on page 317 .

- 3 Click the Update Resource icon on the toolbar to record your changes.

DELETING RESOURCES

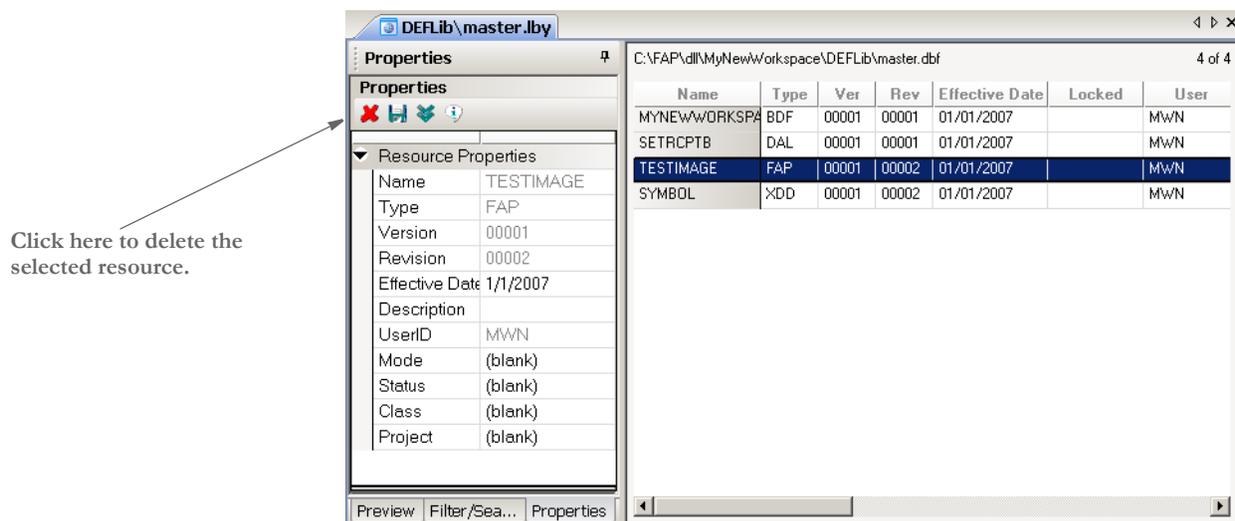
You must be a library or system administrator or have been assigned the specific right to delete library resources. If you do not have the appropriate rights, the red **X** icon shown below will not be enabled.

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.

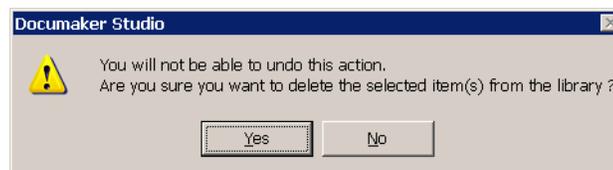
If you delete a resource from the library while it is checked out, you cannot later check in the resource. If you try, Studio tells you the resource has not been checked out. You would then have to close the resource and delete it from disk.

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.



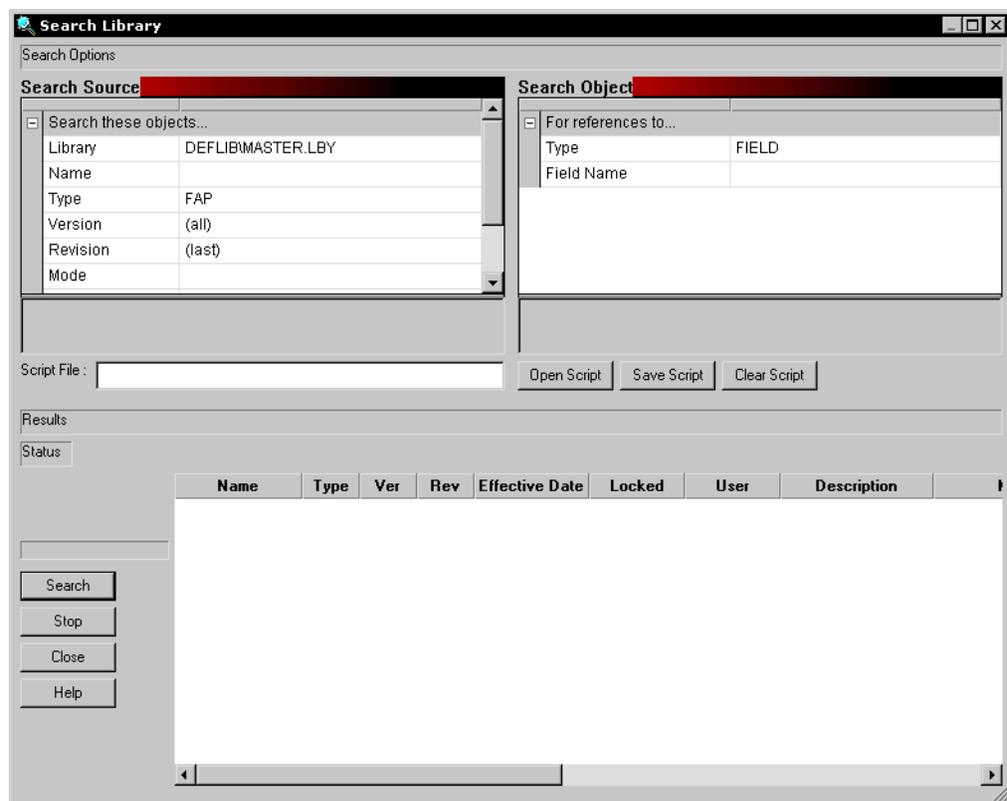
Click Yes to delete the resource you selected. Studio then deletes the resource from the library and from your hard disk.

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.



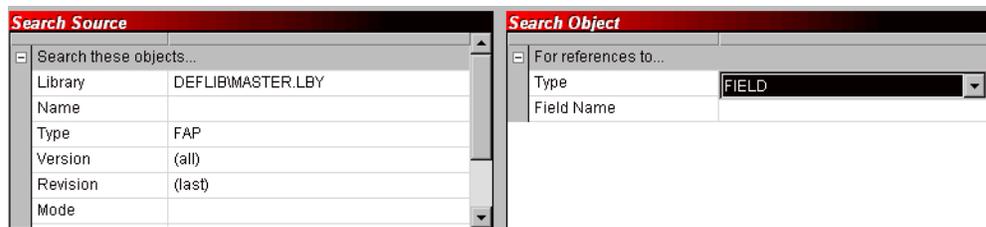
Defining the resources Use the Search Source fields to identify the resources you want to search. You must limit your search to a specific type of resource.

Field	Description
Library	Select the library in which the resources are stored.

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 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
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:

Search Object	
For references to...	
Type	TEXT
Text Element Name	
Text String	
Text Case	Insensitive

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.
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

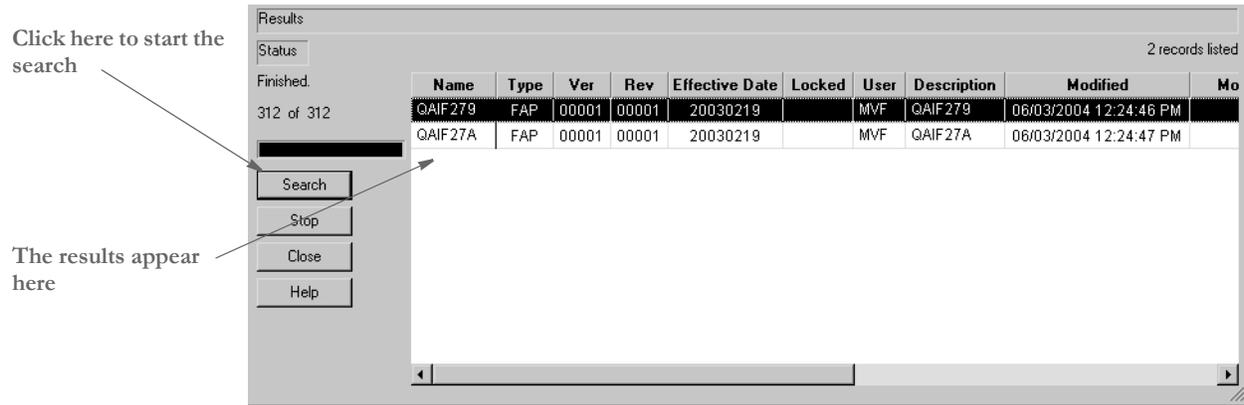
You can save your search settings and reuse them. To save your settings, enter a name for the script in the Script File field and click Save Script. Studio appends an LSC extension onto the name you entered and stores the script in the DEFLIB directory.

Script File : <input type="text"/>	Open Script	Save Script	Clear Script
------------------------------------	-------------	-------------	--------------

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.



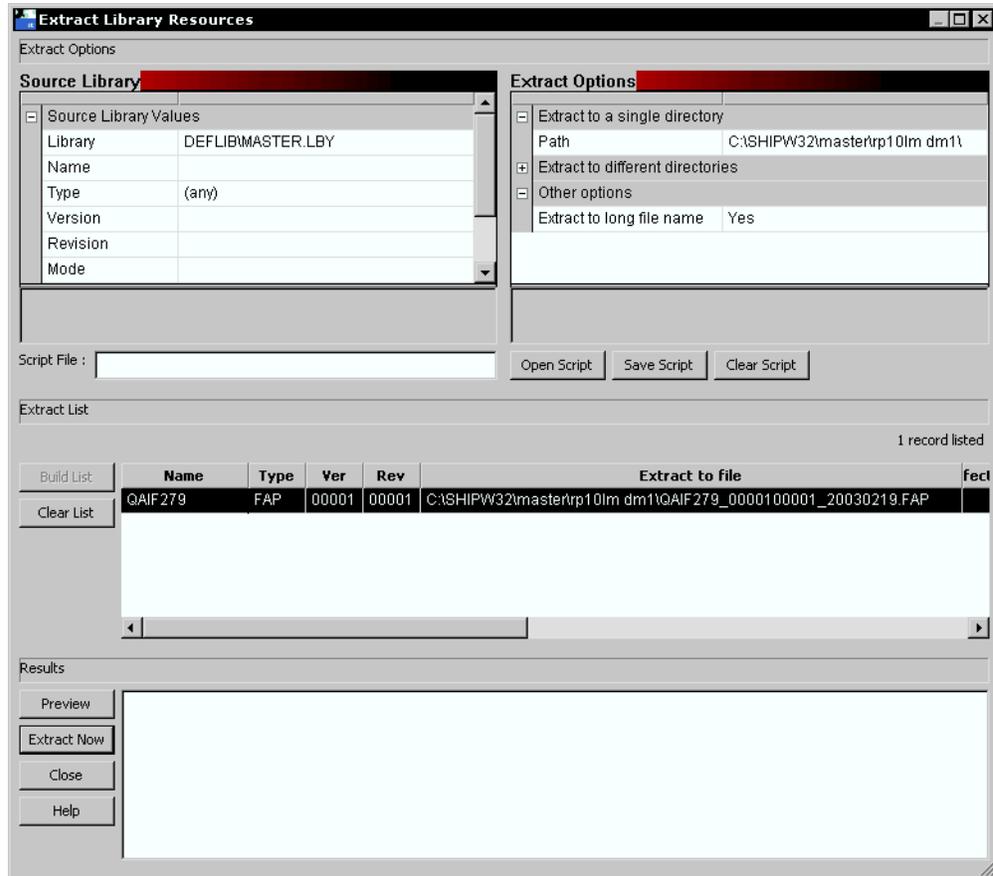
You can highlight on any selected resource, right click, and perform these tasks:

To	Select this option
Check out the resource	Check Out. See Checking Out Resources on page 291 for more information.
Open the resource without checking it out	Read
See the history of changes to the resource	History. See Reviewing a History of Resource Changes on page 314 for more information.
Extract the resource from the library	Extract. See Extracting Resources on page 309 for more information.
Promote the resource	Promote. See Promoting Resources on page 296 for more information.
Expire the resource	Expire. See Expiring Resources on page 313 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.

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 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: <code>FileName_0000100001_20041201.fap</code>

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.

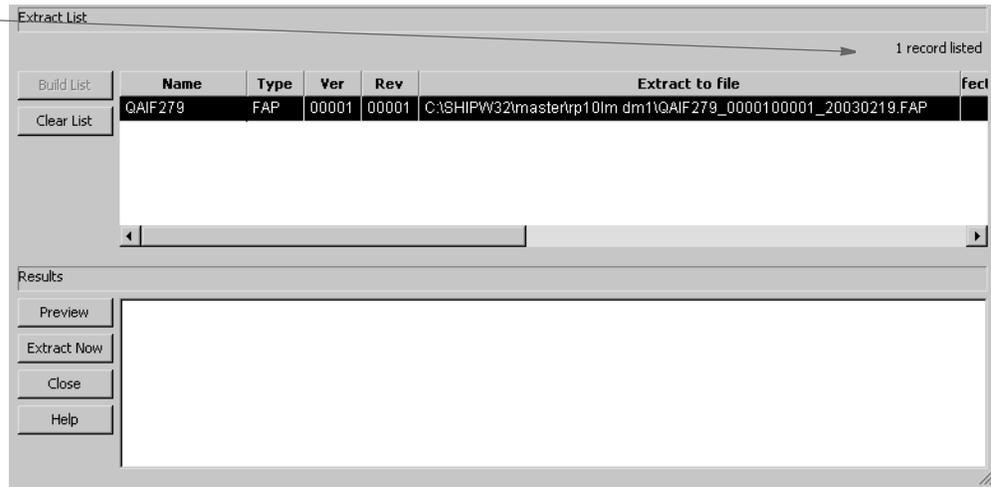


To open a script, click Open Script and choose the script you want to run or edit. You can clear the settings from a script by clicking Clear Script.

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.



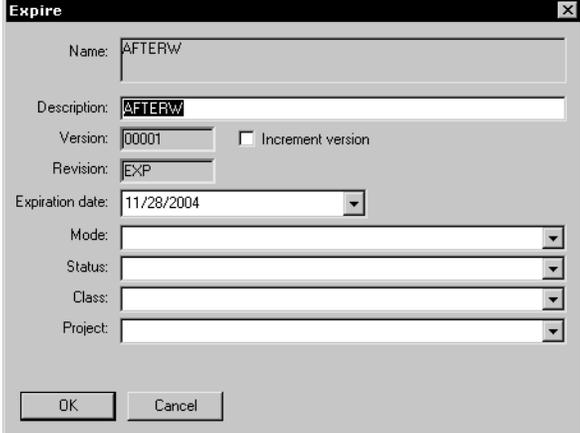
Inspect the resources shown in the list. If necessary, make changes to your selection criteria and build the list again. When you have the list as you want it, click Preview. Studio runs a test extraction and shows you the results.

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 a dialog box titled "Expire" with the following fields and values:

- Name: AFTERW
- Description: AFTERW
- Version: 00001 (with an unchecked "Increment version" checkbox)
- Revision: EXP
- Expiration date: 11/28/2004 (with a dropdown arrow)
- Mode: (empty dropdown)
- Status: (empty dropdown)
- Class: (empty dropdown)
- Project: (empty 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.

Library Log 639 records listed

Filter		Name	Action	Library	Date	Time	Type	Ver	Rev
Name		PHONES	CHECKED IN	MASTER.LBY	20041019	17:00:03	LOG	00001	00002
Action	(any)	CATWOMAN	UNLOCKED	MASTER.LBY	20041019	16:59:25	LOG	00001	00001
Library	DEFLIBMASTER.LBY	JPG	UNLOCKED	MASTER.LBY	20041019	16:59:19	LOG	00001	00001
Type	(any)	JPG	CHECKED OUT	MASTER.LBY	20041019	16:59:00	LOG	00001	00001
UserID		CATWOMAN	CHECKED OUT	MASTER.LBY	20041019	16:58:34	LOG	00001	00001
Mode	(any)	CHECK	CHECKED IN	MASTER.LBY	20041019	16:58:28	LOG	00001	00002
Status	(any)	BULB	CHECKED IN	MASTER.LBY	20041019	16:58:18	LOG	00001	00002
Class	(any)	ARR1	CHECKED IN	MASTER.LBY	20041019	15:01:29	LOG	00002	00004
Project	(any)	ARR1	CHECKED IN	MASTER.LBY	20041019	15:01:21	LOG	00001	00004
Find Now	Find Now	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
		RP101.M.DM1	CHECKED OUT	MASTER.LBY	20041004	15:33:03	RDF	00001	00001

Studio shows you a log of all changes to all resources. You can filter this list by assigning filter values.

Filter	
Filter Values	
Name	
Action	(any)
Library	DEFLIBMASTER.LBY
Type	(any)
UserID	
Mode	(any)
Status	(any)
Class	(any)
Project	(any)
Find Now	Find Now

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> .

Field	Description
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.
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.

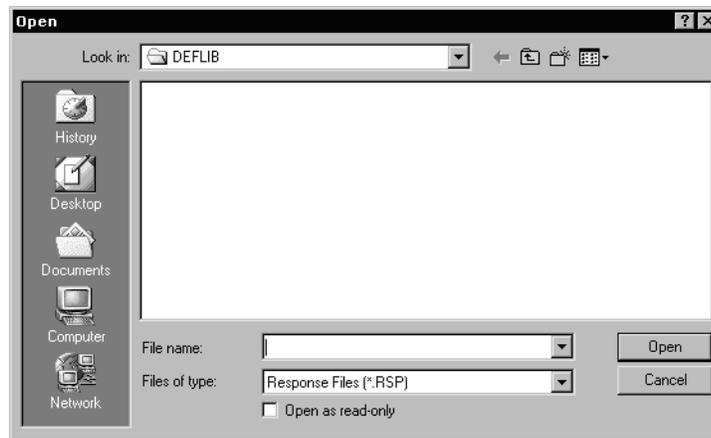
RUNNING RESPONSE FILES

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:

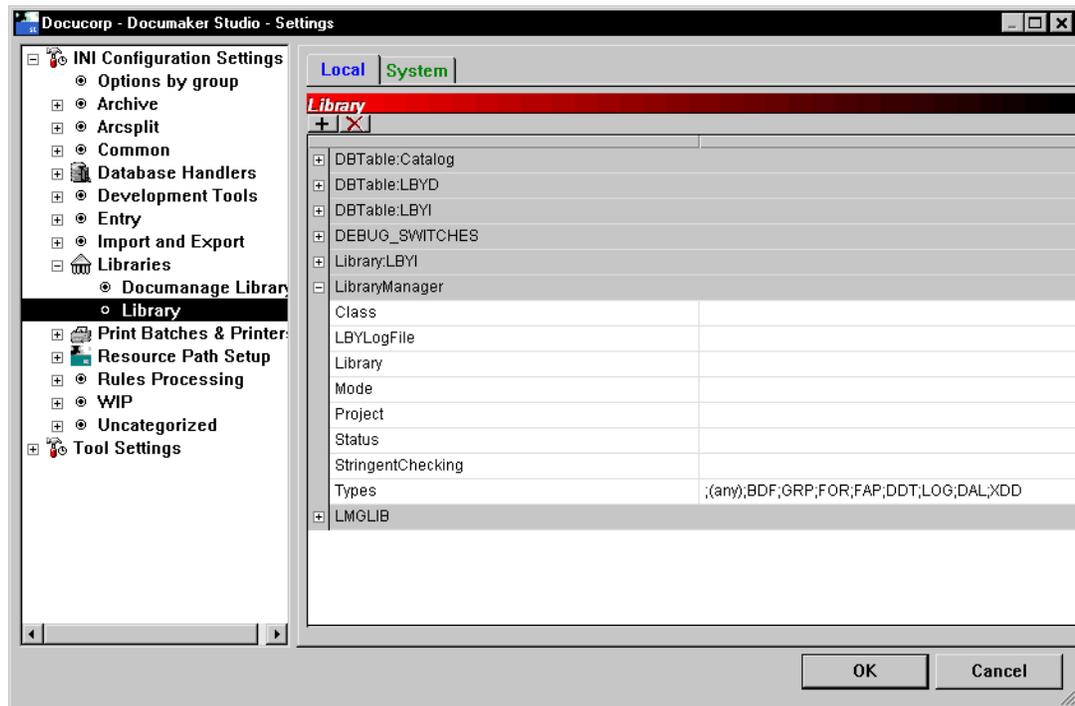
- 1 Choose the Library, Run Response File option. The Open window appears from which you can select the response file you want to run.



- 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 LibraryManager, 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>

Option	Description
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>
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:

```
<LBYSRIPT>
<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
Docupresentation (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 Docupresentation, 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 Docupresentation (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 322](#)), 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 322](#)), 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 Extract Files on page 326](#)
- [Working with Fields on page 329](#)
- [Importing TGA Files on page 330](#)

WORKING WITH EXTRACT FILES

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

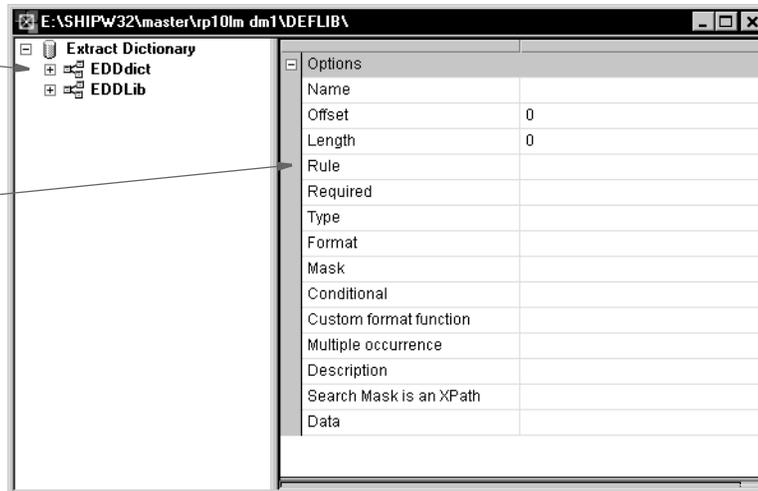
There are two ways to open an extract dictionary:

- Using the Manage, Dictionaries, Extract option
- By clicking on Extract in the workspace

The main window that appears when you are managing an extract dictionary is shown here.

Click here to expand these items

You maintain the property information 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.
Type	Reserved for future use.
Format	Reserved for future use.
Mask	Include the formatting mask for the data returned by the rule.
Conditional	Set to Yes if this entry can be used as a trigger.
Custom format function	Reserved for future use.
Multiple occurrence	Reserved for future use.
Description	Reserved for future use.

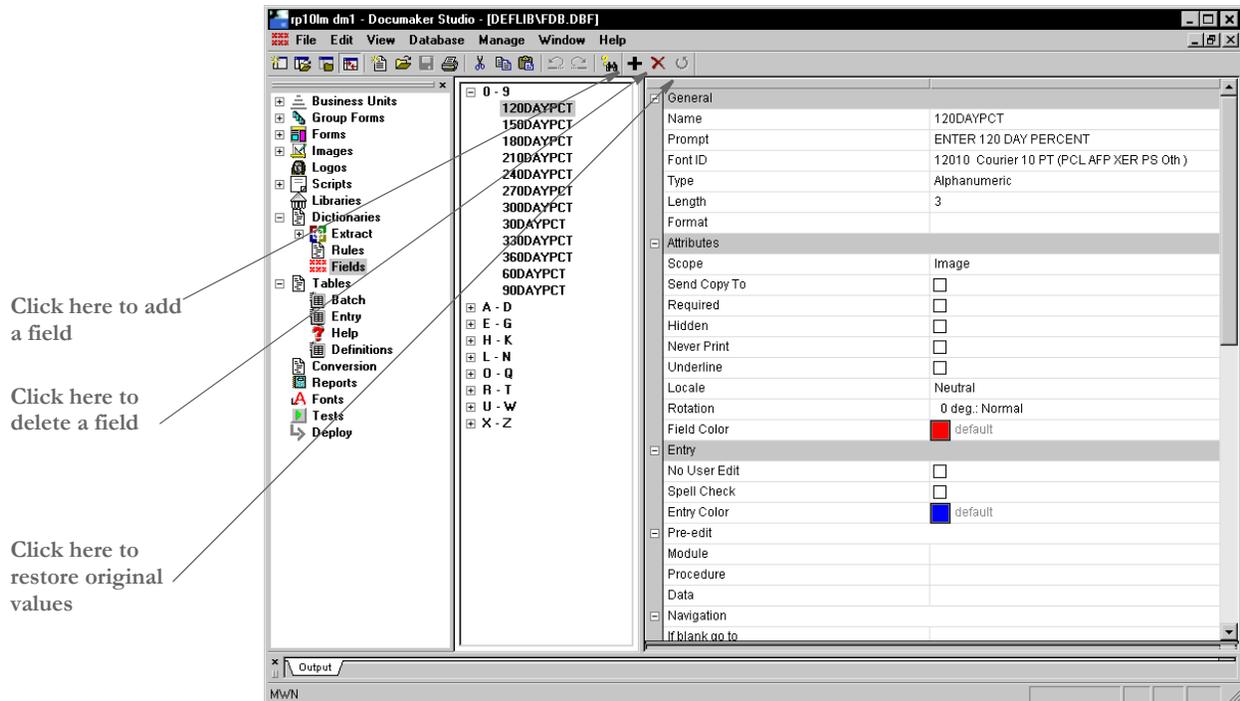
Property	Description
Search mask is an XPath	Set to Yes if the search mask is an XPath. Otherwise, choose No.
Data	Use this field for search masks and rule parameters.

WORKING WITH FIELDS

Click on Fields 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:

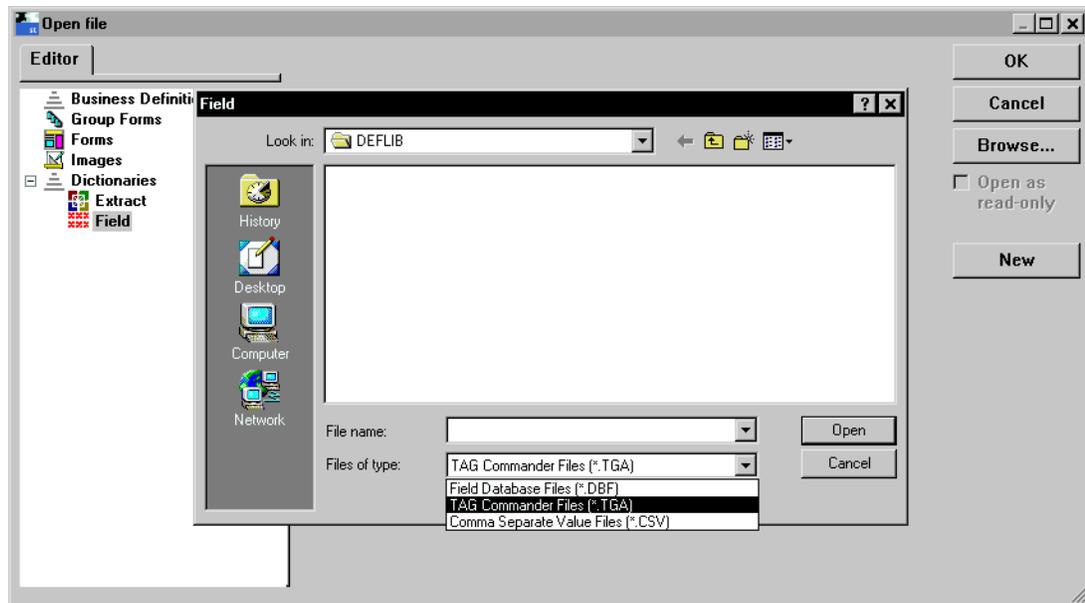


When you highlight a field, its property information appears so you can modify it.

IMPORTING TGA FILES

Follow these steps to import a Tagcommander (TGA) file:

- 1 Choose Manage, Dictionaries, Fields.
- 2 Choose File, Import (not Import Workspace files). On the Open File window, double click on Fields.
- 3 In the Field window, change the Files of Type to TGA, as shown here:



- 4 Browse for the file you want to import and then press 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 Codebase, 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 Codebase, 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:

- a transaction file DFD
- a recipient batch file DFD
- an 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 MVS or OS/400.

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 Codebase, 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 Docucorp Binary Large Object NULL terminated string character array not NULL terminated A Docucorp 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 Docucorp 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		Data format of the field on storage media.
	Same as Internal Type format types.	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.

Field	Format Type	Field / Format Description
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

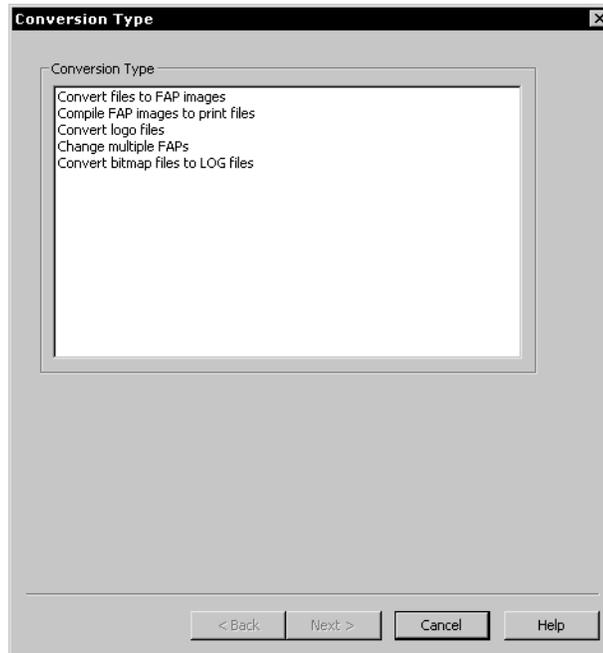
Use the Conversion option to convert, change, or compile one or more files.

This chapter covers these topics:

- [Overview on page 338](#)
- [Converting Files into FAP Files on page 340](#)
- [Compiling FAP Files into Print Files on page 341](#)
- [Converting Graphic Files on page 342](#)
- [Making Changes to Multiple FAP Files on page 343](#)
- [Converting Bitmap Files into LOG Files on page 351](#)
- [Finishing a Conversion on page 352](#)

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 move to the next page. Click Cancel to exit Conversions.

To	See
Import files into a FAP file and, optionally, check them into library.	Converting Files into FAP Files on page 340
Compile FAP files into PCL, AFP, or Metacode print files	Compiling FAP Files into Print Files on page 341
Convert graphics files into another format and, optionally, check them into library (if converting into the LOG format)	Converting Graphic Files on page 342

To	See
<p>Make changes to FAP files. This includes:</p> <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 <p>Many of these options are present in order to adjust situations in legacy systems. After the changes have been made, the files can optionally be checked into the library.</p>	<p>Making Changes to Multiple FAP Files on page 343</p>
<p>Convert graphics files into the LOG file format and check them into library</p>	<p>Converting Bitmap Files into LOG Files on page 351</p>

NOTE: Other conversion options are currently available by running command line utilities. Refer to the [Docutoolbox Reference](#) for more information.

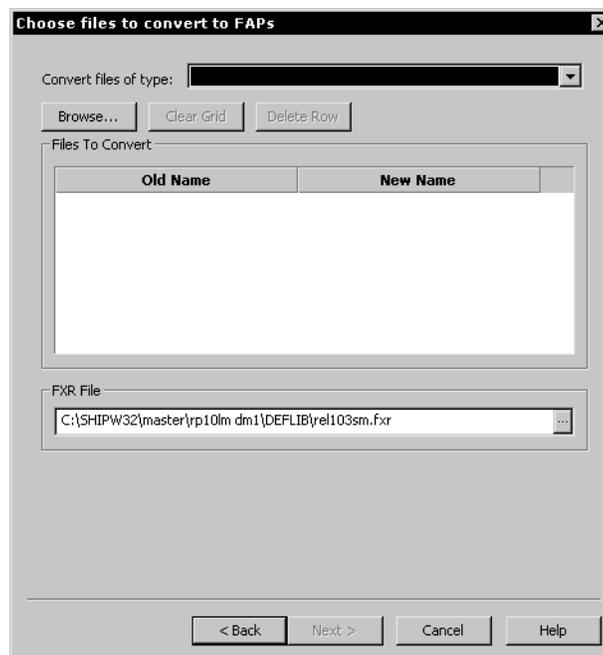
CONVERTING FILES INTO FAP FILES

You can choose to convert to several file types including:

- Rich text files (RTF)
- PCL print files
- Metacode files
- AFP print files
- Docucorp Compound Document (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) and CSET2FAP (CompuSet to FAP). See the [Docutoolbox Reference](#) for more information.

Click Browse to select the files you want to convert.



Once you have selected the files you want to change, click Next to move to the Finish page. See [Finishing a Conversion on page 352](#) for more information.

COMPILING FAP FILES INTO PRINT FILES

You can choose to compile to several file types including:

- AFP print files
- PCL print files
- Metacode files
- Postscript files

Click Browse to select the files you want to convert.



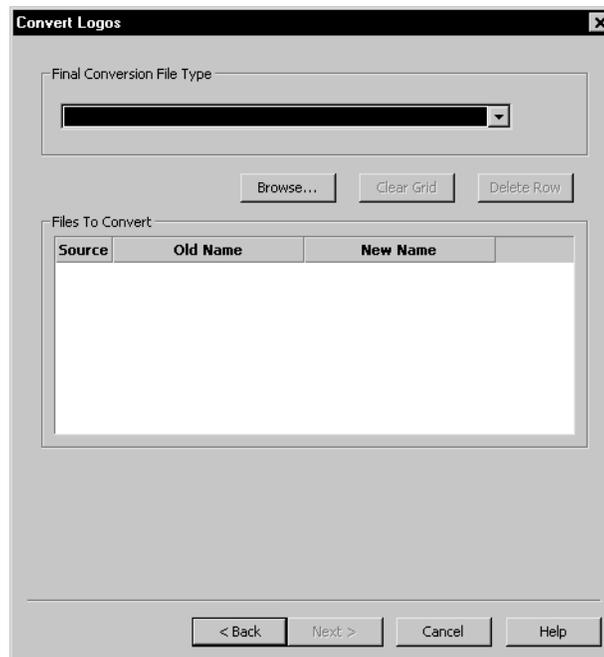
Once you have selected the files you want to change, click Next to move to the Finish page. See [Finishing a Conversion on page 352](#) for more information.

CONVERTING GRAPHIC FILES

You can choose to convert to several file types including:

- Standard graphic (LOG) files
- Compressed Pack files
- Compressed TIFF files
- Xerox font files
- Xerox image files
- Bitmaps (.BMP)
- Overlays (.OVL)
- Segmented graphics for AFP printers (.SEG)
- JPEG files (.JPG)

Click Browse to select the files you want to convert.



Once you have selected the files you want to change, click Next to move to the Finish page. See [Finishing a Conversion on page 352](#) for more information.

MAKING CHANGES TO MULTIPLE FAP FILES

You make numerous changes to a group of section (FAP) files. Typically, you would use this option to convert FAP files created with older versions of 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
- Update DDT files from FAP files and the data dictionary

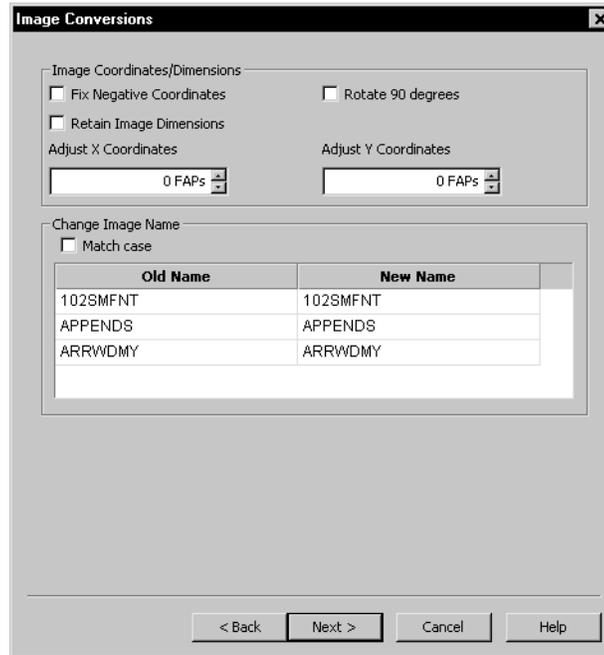
Click Browse to find and select files to run this conversion on. Once the files are selected they appear in the grid.



Once you have selected the files you want to change, click Next to move to the Section Conversions page. See [Finishing a Conversion on page 352](#) for more information.

SECTION CONVERSIONS

The Section Conversions page lets you define changes that apply to the sections you have selected.



From this page you can:

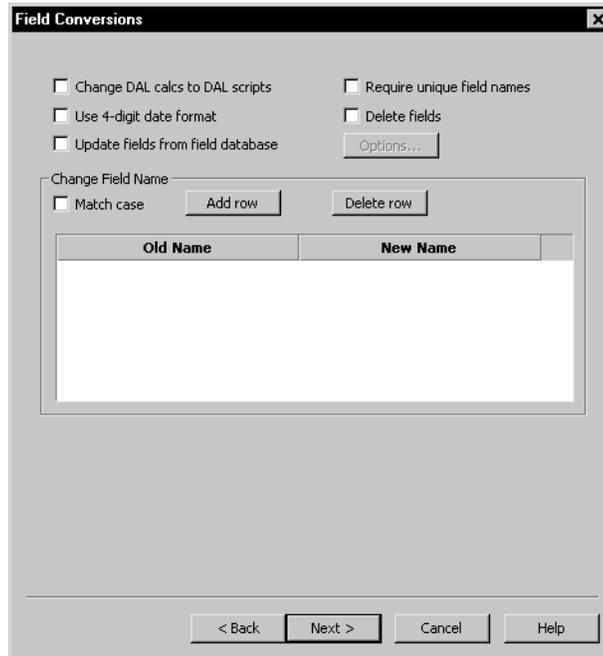
To	Then
Fix negative coordinates	Check the Fix Negative Coordinates field. Negative coordinates are not produced by Studio, but some users have encountered them when converting print streams.
Retain current section dimensions	Studio increases a sections horizontal or vertical dimensions if you place objects outside the section's borders. To prevent this from happening, check the Retain Section Dimensions field.
Rotate sections 90 degrees	Check the Rotate 90 Degrees field.
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.
Change the names of FAP files	Enter the current name of the FAP file 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 FAP files to change.

Once you have selected the options you want, click Next to move to the Field Conversions page.

FIELD CONVERSIONS

The Field Conversions page lets you make changes that apply to several fields at one time.



From this page you can:

To	Then
Change DAL calculations into DAL scripts	Check the Change DAL calculations to DAL scripts field.
Format dates using 4-digit years	Check the Use 4-digit Year Date Format field to change all date fields on the selected sections to use 4-digit years. At the turn of the century, it may become difficult to tell which century a 2-digit year value references. Studio adjusts the fields input length if necessary, but makes no other changes to the field.
Update fields from the Field Database	Check the Update Fields from Database field. Studio retrieves all of the information stored in the field database except for the font ID.
Use unique field names	Check the Require Unique Field Names 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.
Delete fields	Check the Delete Fields 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 turn a section into a template which you can reuse or as another way of creating overlays.

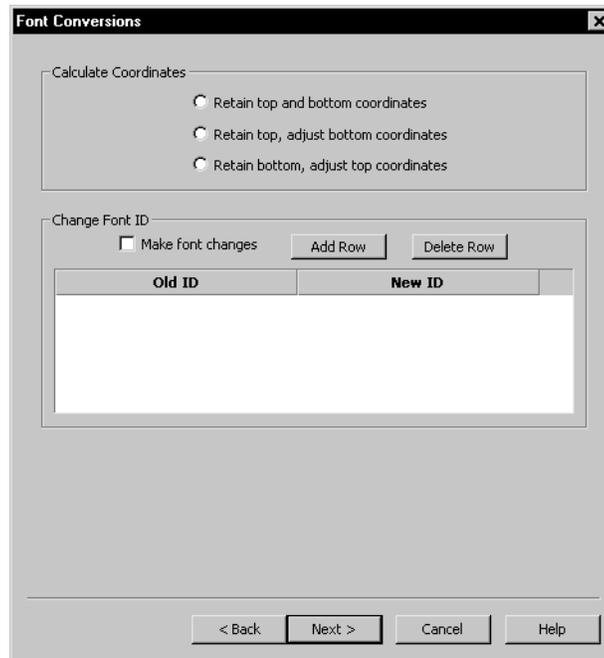
To	Then
Change field names	Enter the current name of the field in the Old Name field and the name you wish to change to in the New Name field.

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 move to the Font Conversions page.

FONT CONVERSIONS

The Font Conversions page lets you make changes that apply to several fonts at one time.



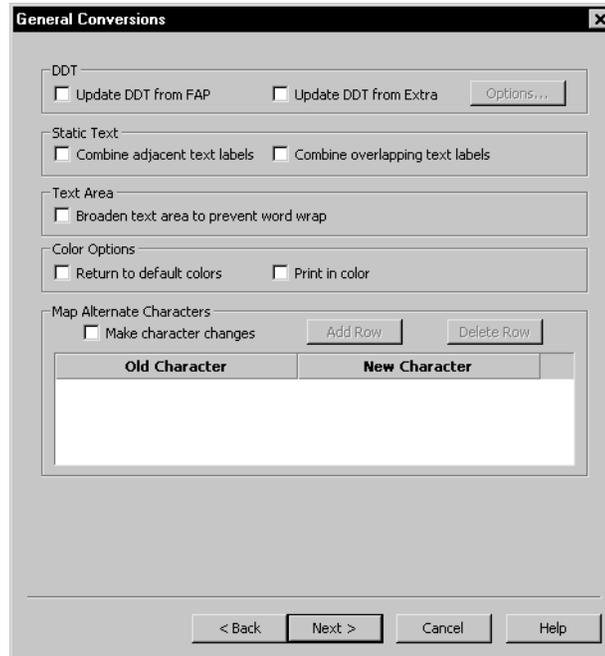
From this page you can:

To	Then
Calculate coordinates	To have Studio recalculate font coordinates, click the appropriate option. Typically, you would use the Retain Bottom, Adjust Top Coordinates option so the text would retain its current baselines. Click Retain Top, Adjust Bottom Coordinates if you want to keep the top coordinates. Click Retain Top and Bottom Coordinates if you do not want Studio to recalculate font coordinates.
Change font IDs	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 move to the General Conversions page.

GENERAL CONVERSIONS

Use the General Conversions page to make general changes to the sections you selected.



From this page you can:

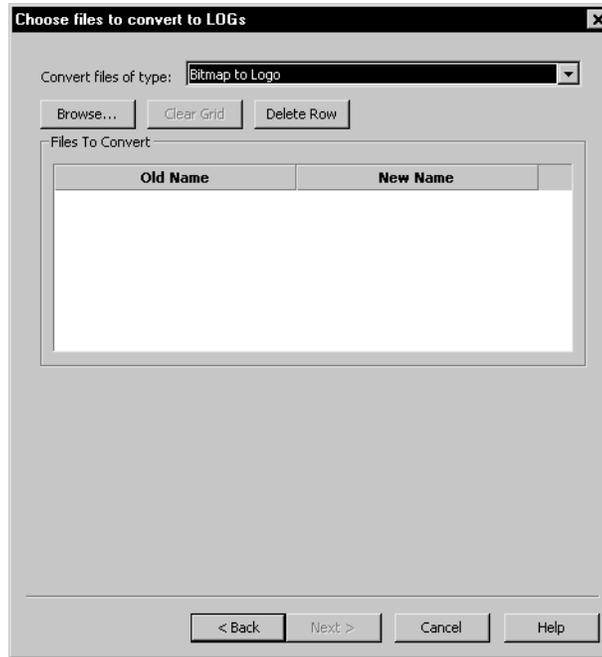
To	Then
Update DDT from FAP and XDB	To update the DDT files for the sections you selected using existing FAP and Data Dictionary information, check this field. Studio updates the font ID and x and y coordinates. If Studio finds a field name which matches one found in the data dictionary, it updates the remaining DDT information.
Combine adjacent text labels	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. 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.
Combine overlapping text labels	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. 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.

To	Then
Broaden text area to prevent wrapping	Check this option to enlarge your text areas to prevent the text from wrapping.
Use the 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 <i>a</i> to <i>á</i>, 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>

Once you have selected the options you want, click Next to move to the Finish page. See [Finishing a Conversion on page 352](#) for more information.

CONVERTING BITMAP FILES INTO LOG FILES

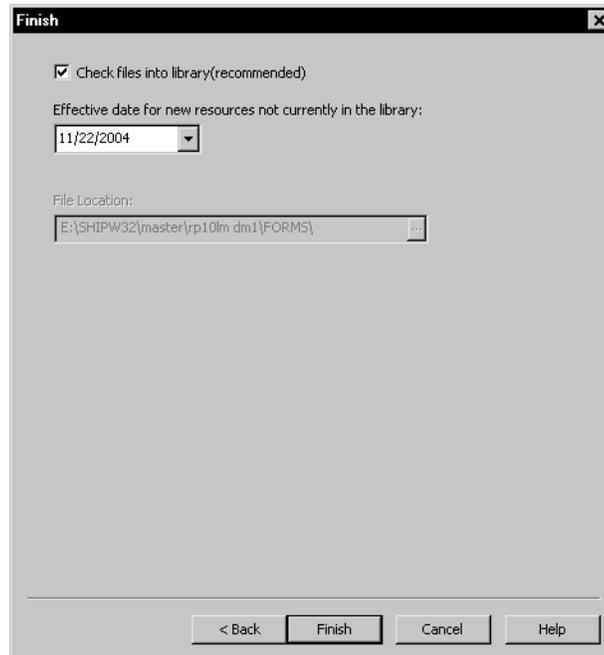
Click Browse to select the files to run the conversion on.



Once you have selected the files you want to change, click Next to move to the Finish page. See [Finishing a Conversion on page 352](#) for more information.

FINISHING A CONVERSION

On the Finish page you can choose to check files into the workspace library or place them into a directory.



Click Finish to complete the conversion.

Chapter 13

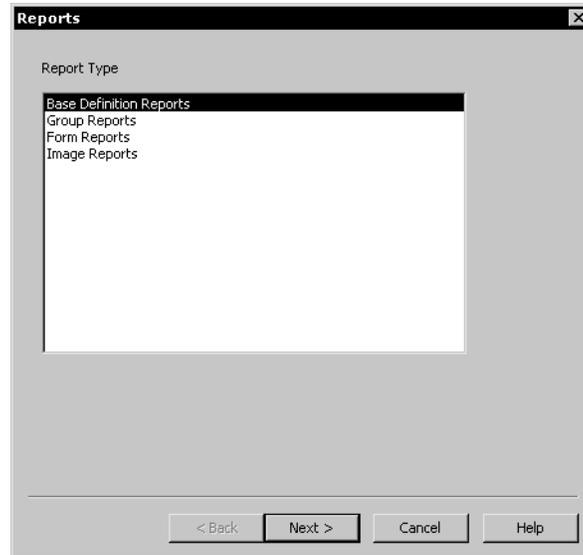
Printing Reports

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-screen or sending it to your printer.

GENERATING A REPORT

When you choose Reports from the Workspace tree, the Reports window appears, as shown here.



You can choose from these kinds of reports:

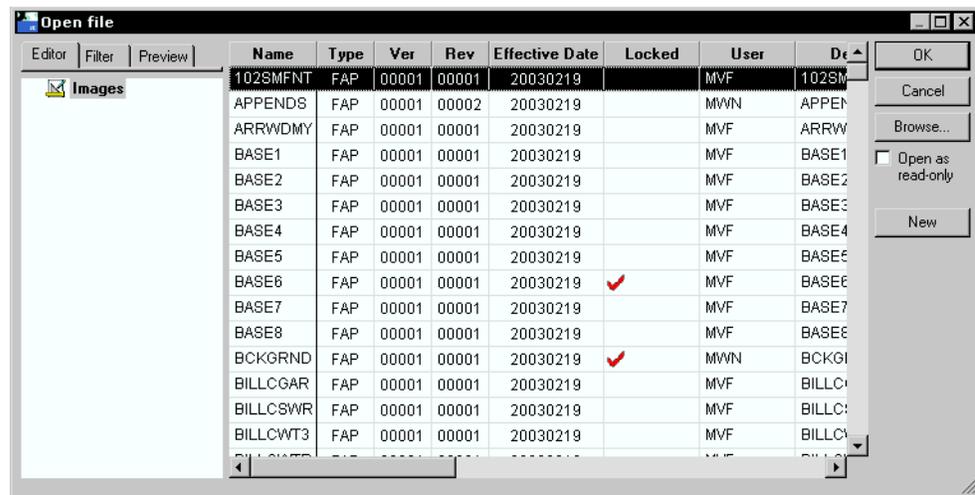
- Base Definition Reports
- Group Reports
- Form Reports
- Section Reports

The following steps take you through the steps for printing reports. Section reports are used as an example.

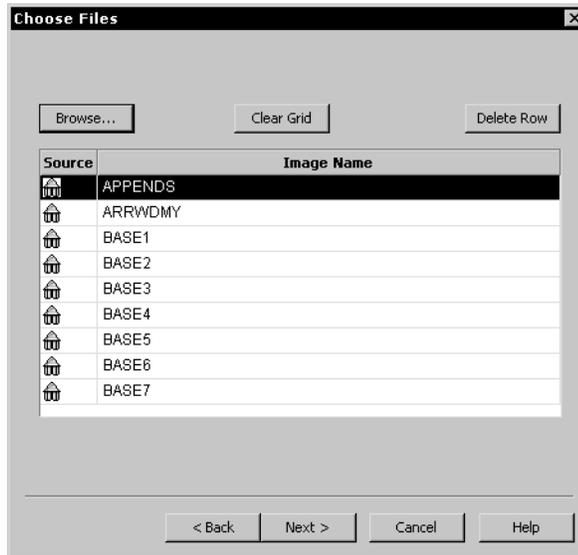
- 1 Highlight the type of report you want to print and click Next. The Choose Files window appears.



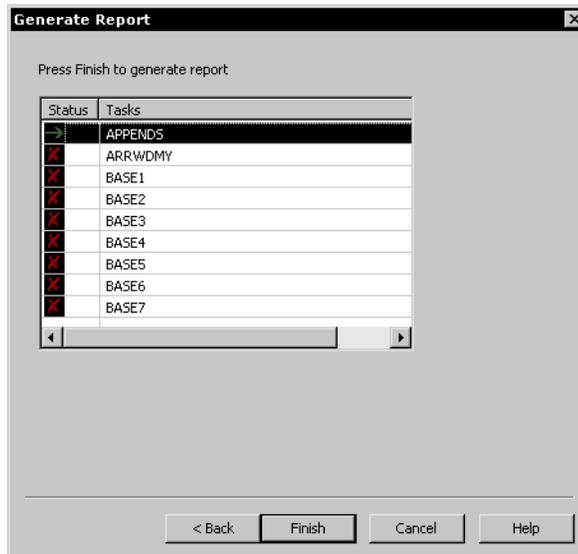
- Click Browse to choose files from the Open File window.



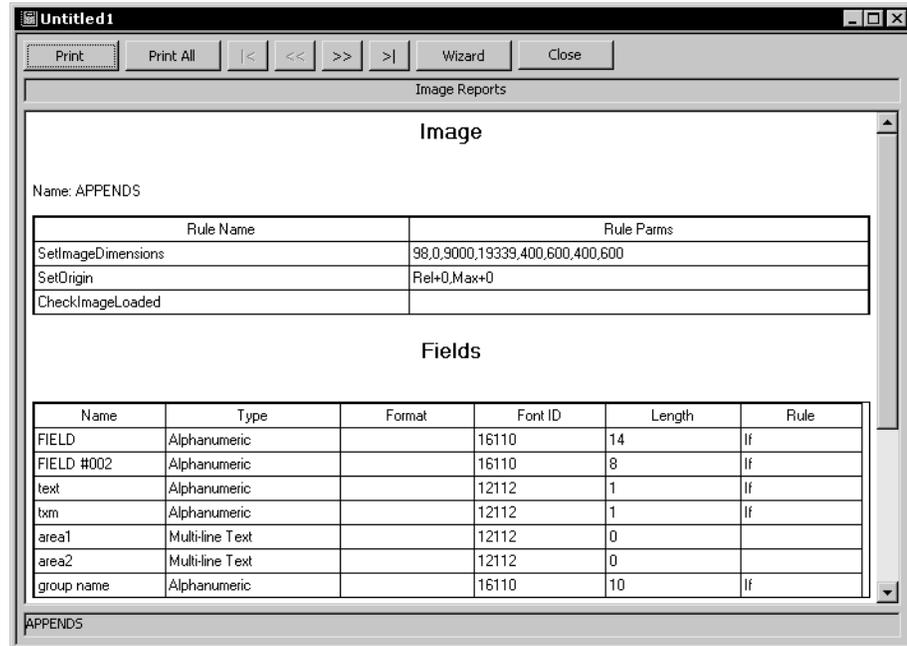
- Highlight the files you want included on the report and click Ok. Press CTRL to select multiple files. You return to the Choose Files window.



- 4 The files you selected appear in the grid. You can use the Delete Row button to remove a file or click Clear Grid to remove all files. Once you have the correct set of files listed, click Next. The Generate Report window appears.



- 5 Click Finish to generate the report. Studio displays the report on your screen. Here is an example.



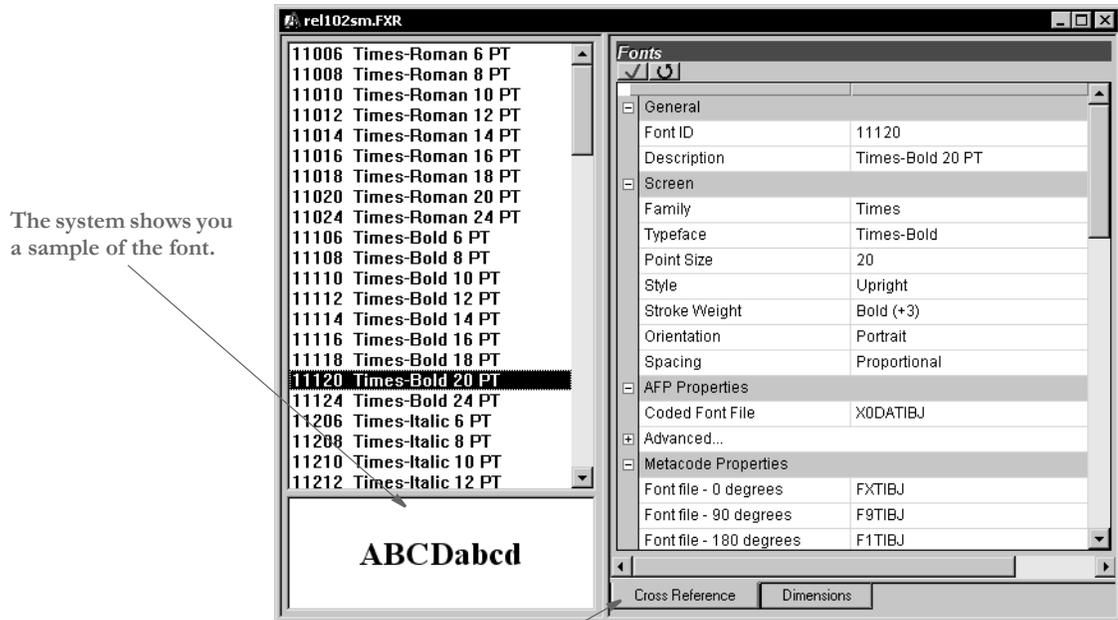
The buttons on this window let you print the current report, print all reports, move through the reports, restart the wizard to generate other reports, or close the window.

Chapter 14

Handling Fonts

Click on Fonts to open and work with the various fonts you use on your forms. A font is a collection of letters, symbols, and numbers which share a particular design. The system lets you organize sets of fonts for section creation, viewing, and printing needs.

OVERVIEW Here is an example of the window that appears:

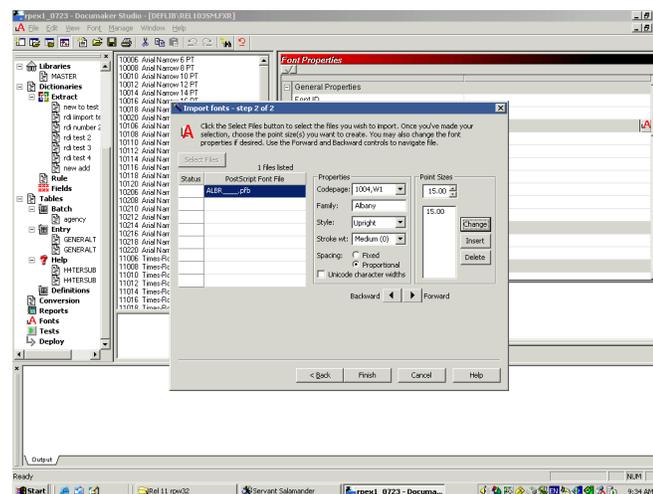


The system shows you a sample of the font.

Click on these tabs to set the dimensions and other properties

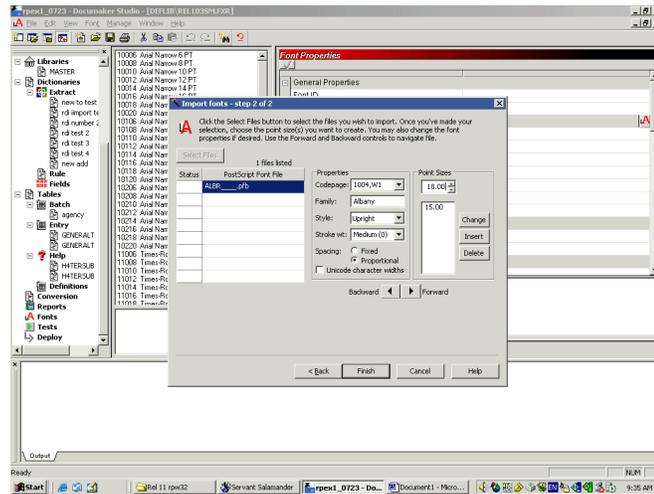
CHANGING THE POINT SIZE

Follow these steps to change the point size when you are inserting a font:

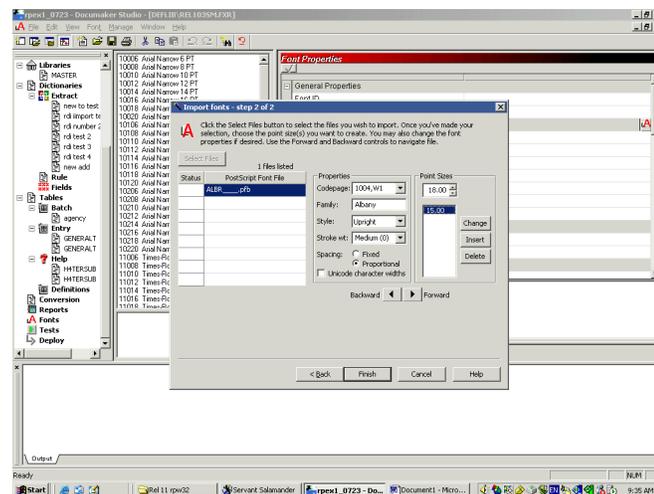


- 1 Insert a PostScript font.
- 2 Enter the point size and click Insert.

To change the point size to something else, enter the new point size in the Point Size field:



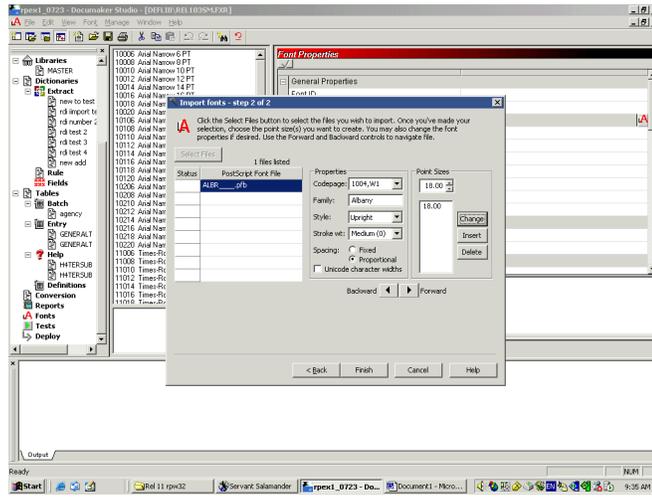
3 Click on the point size of 15.00, then click Change.



Studio changes the 15.00 to match the value in the Point Size field.

Chapter 14

Handling Fonts



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 364](#)
- [Using the Screen on page 366](#)
- [Creating a Test Profile on page 373](#)
- [Running a Test on page 376](#)
- [Changing Test Properties on page 381](#)
- [Modifying the AFGJOB File on page 382](#)
- [Deleting a Test on page 383](#)
- [Using the Printstream Analyzer on page 384](#)

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.

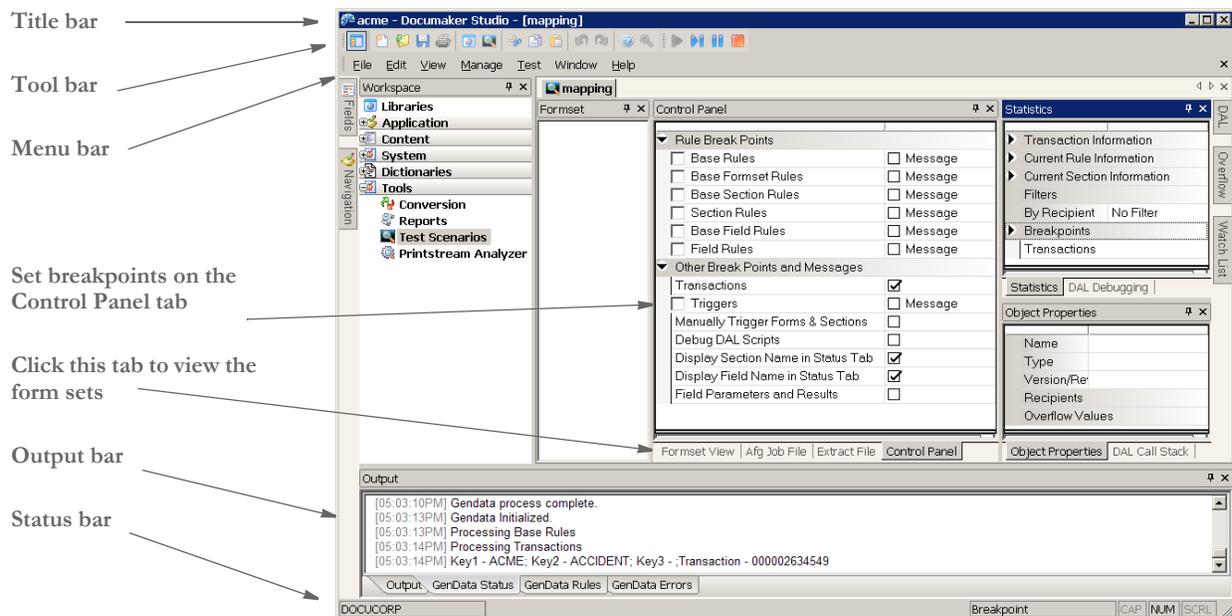
Messaging can give you more feedback of exactly what processed and in what order. When things are going along nicely, you may want to turn messaging off to reduce the clutter you see on the screen. When things are not working as expected, you would probably turn on more messaging to get an idea of what 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.
Tool bar	The tool bar 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

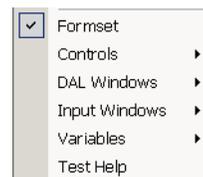
This section introduces you to the pull-down menus which include additional options or are only available when you are working with tests. When you open a test profile, Studio adds the Test menu to your menu bar.

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

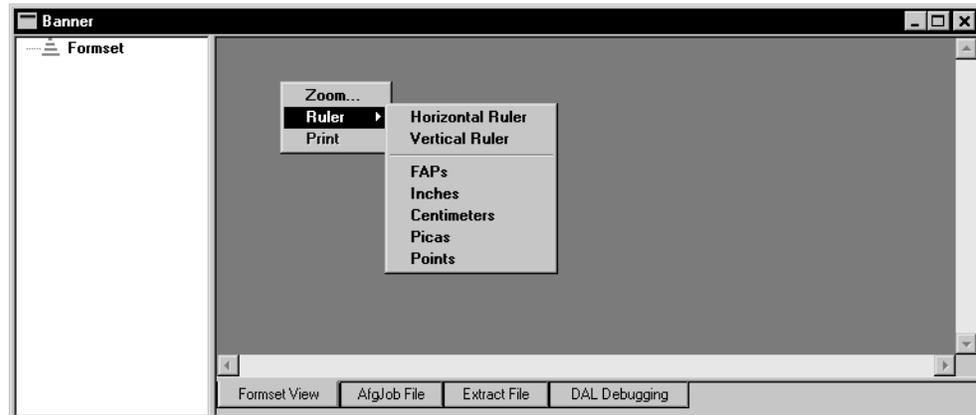
The Test menu provides you with tools to manage the test profiles you create and clear breakpoints. When you select Test, this menu appears:



Option	Description
Run	Start the test.
Step	
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	

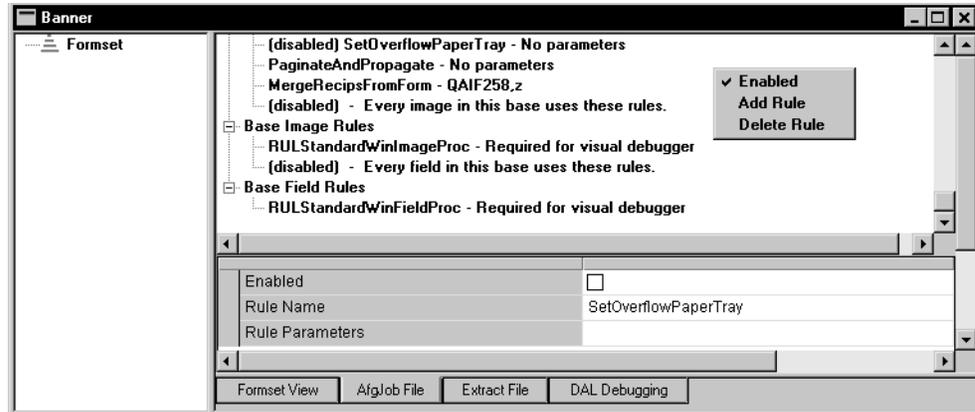
USING THE RIGHT-CLICK MENUS

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 TOOL BAR

The tool bar provides a quicker way to select options that may be listed on a drop down menu. Here is an example of the tool bar shown when you are working with test profiles:



Standard tool bar icons

Shown below are the tool bar 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

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

Test tool bar icons

Shown below are the tool bar 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.

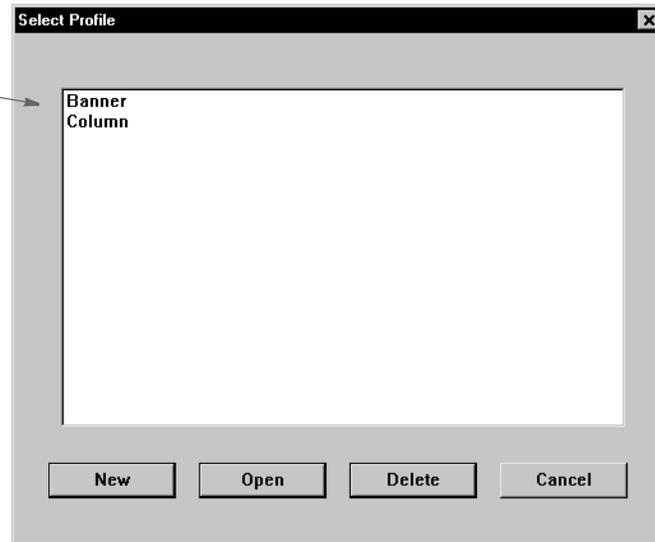
NOTE: You can hide these icons by choosing View, Test Remote Control.

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:

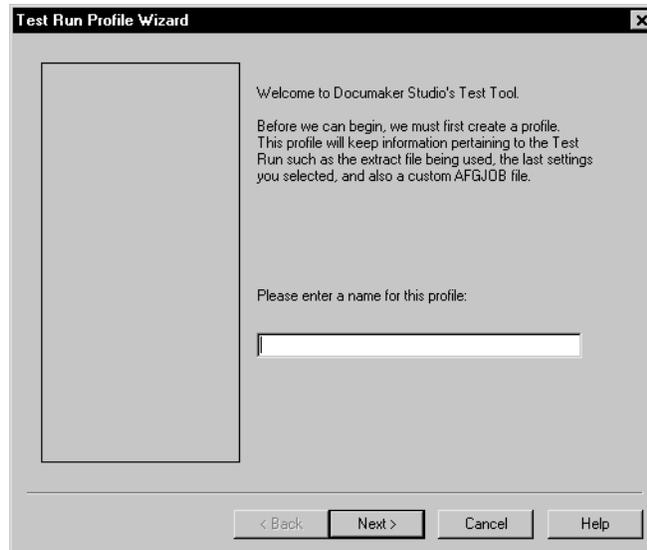
These tests have already been set up.

To choose one, double click on it or highlight it and click Open.



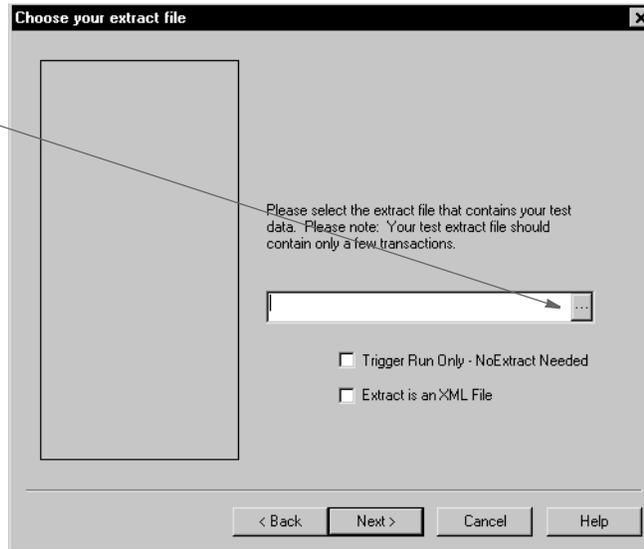
To create a new test profile, click New and follow these steps:

- 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.

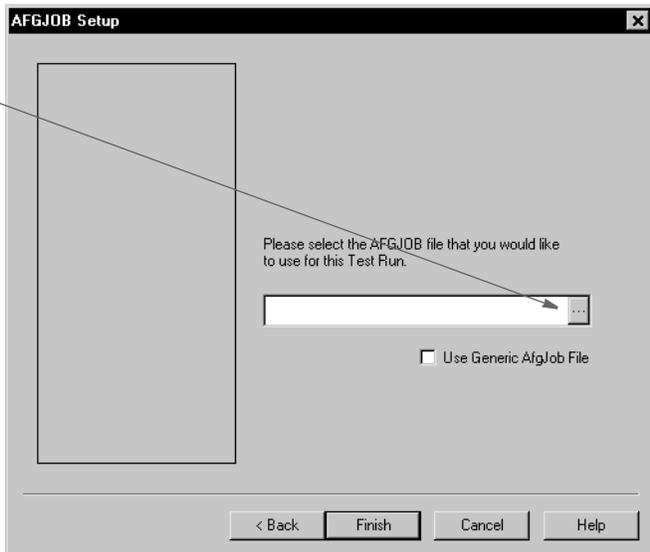
NOTE: See [Running a Trigger-Run Only Test on page 379](#) for more information on running a trigger-only test.

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.



- 3 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.

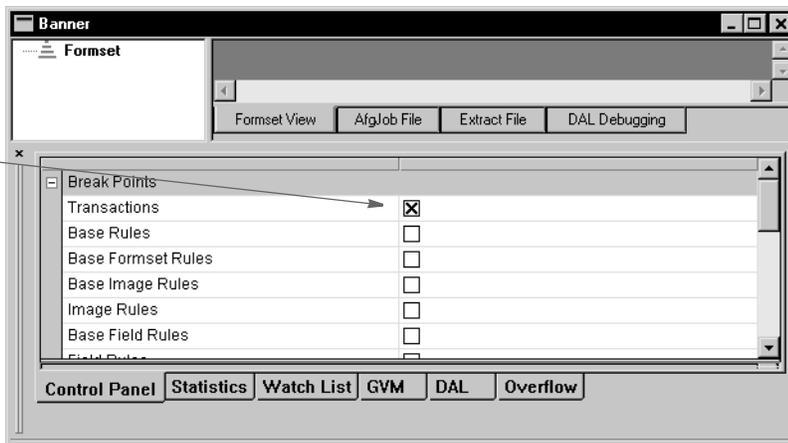
If you do not have a specific JDT file you want to use, click the Use Generic Afgjob File option.

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.



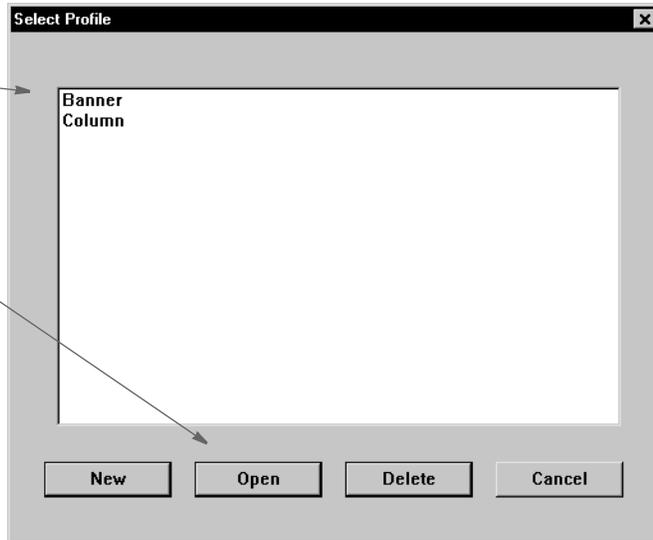
RUNNING A TEST

Follow these steps to run a test profile you have already defined.

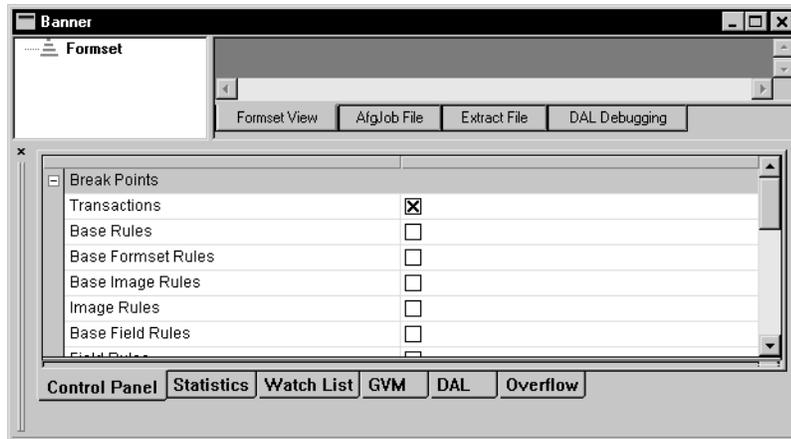
- 1 Open a workspace, then click on Tests. The following window appears.

These tests have already been set up.

To run a test, highlight it and click Open.

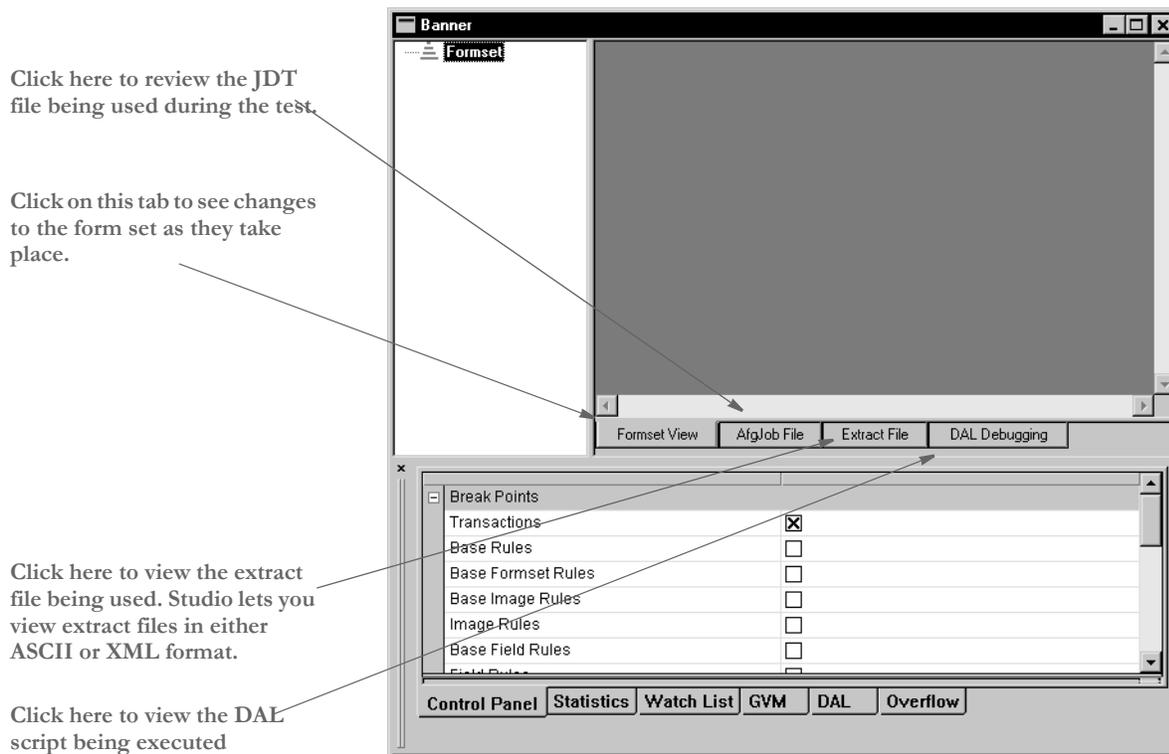


- 2 Highlight the test profile you want to run, then click Open. Studio displays a window similar to this one.



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.



Click here to review the JDIT file being used during the test.

Click on this tab to see changes to the form set as they take place.

Click here to view the extract file being used. Studio lets you view extract files in either ASCII or XML format.

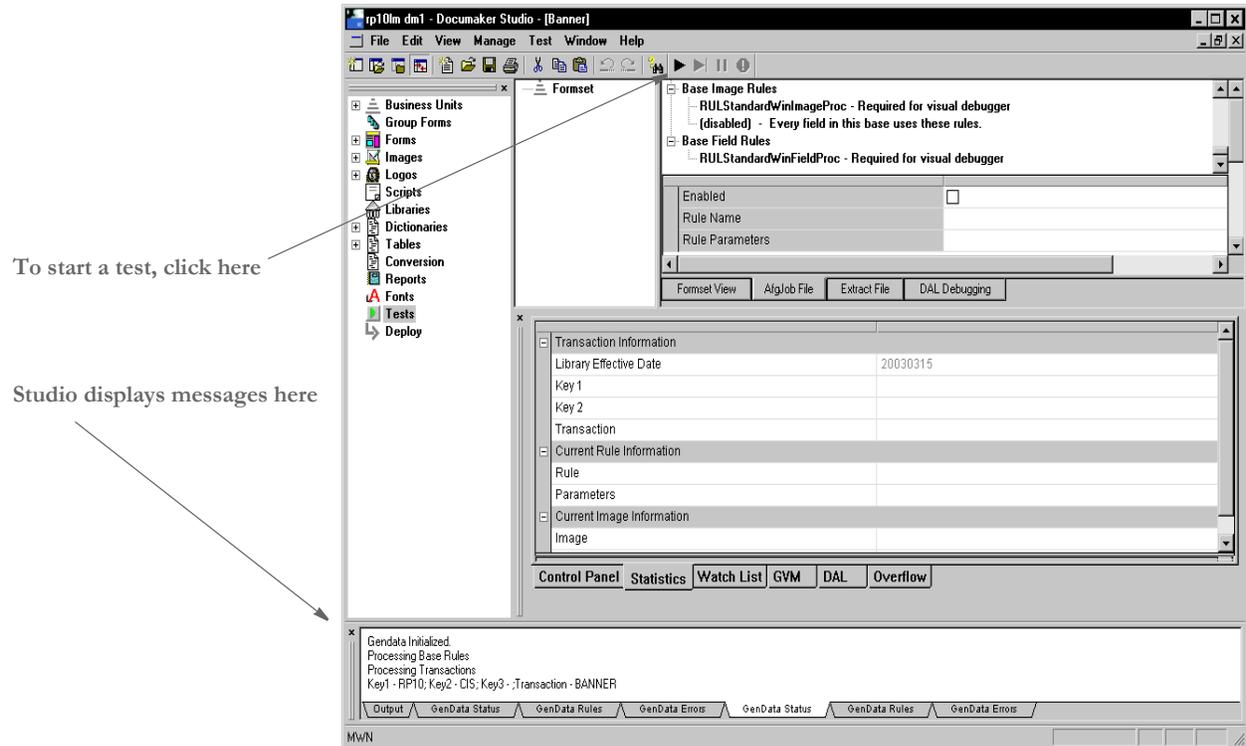
Click here to view the DAL script being executed

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. For instance, you can	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.

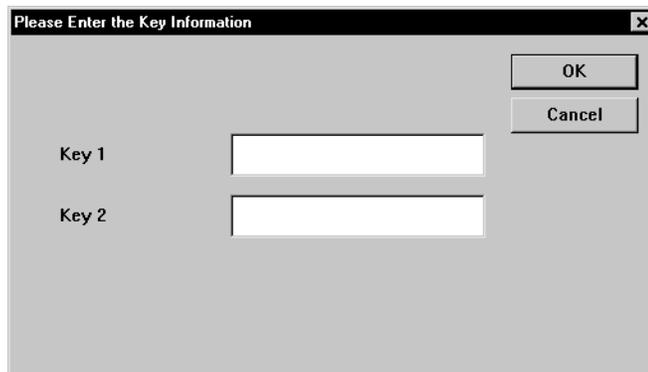
4 To start the test, click Start on the tool bar, as shown here:



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:

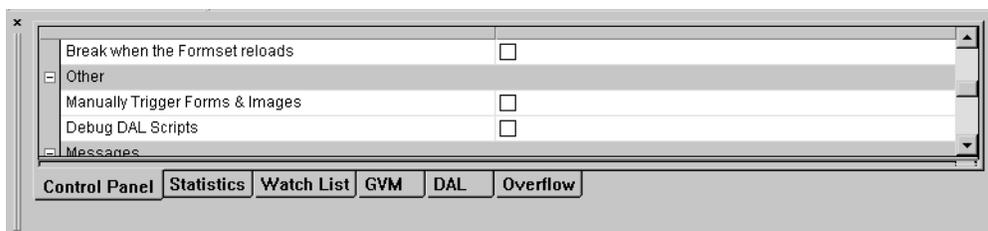


A dialog box titled "Please Enter the Key Information" with a close button (X) in the top right corner. It contains two text input fields labeled "Key 1" and "Key 2". To the right of the input fields are two buttons: "OK" and "Cancel".

Use this window to enter the Key1 and Key2 fields you want to trigger on.

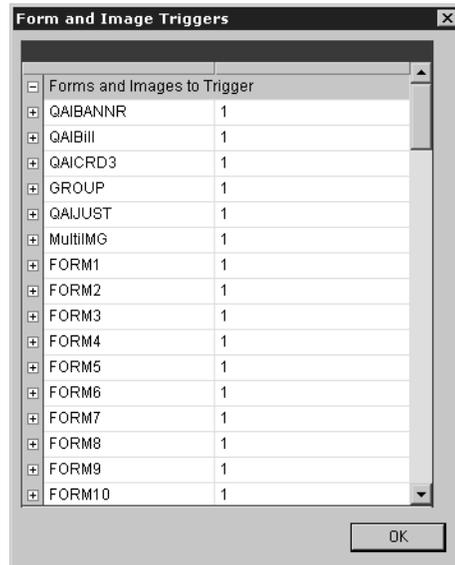
MANUALLY TRIGGERING

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" (checkbox), "Other" (checkbox), "Manually Trigger Forms & Images" (checkbox), "Debug DAL Scripts" (checkbox), and "Messages" (checkbox). Below the list is a row of buttons: "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.



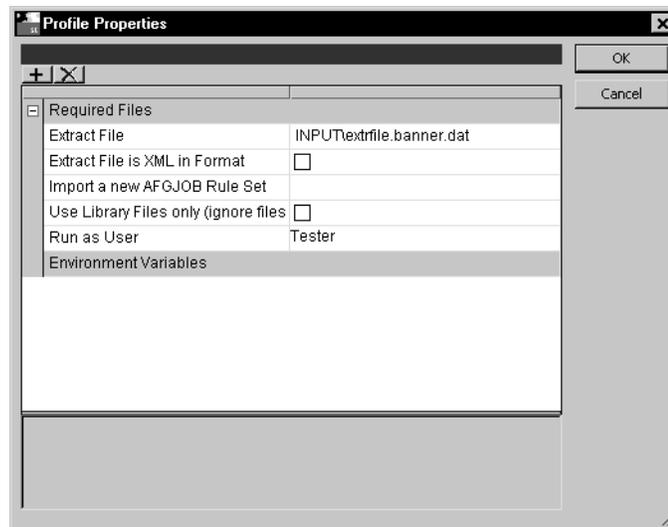
This can be useful when you need to test complicated scenarios without robust data.

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:



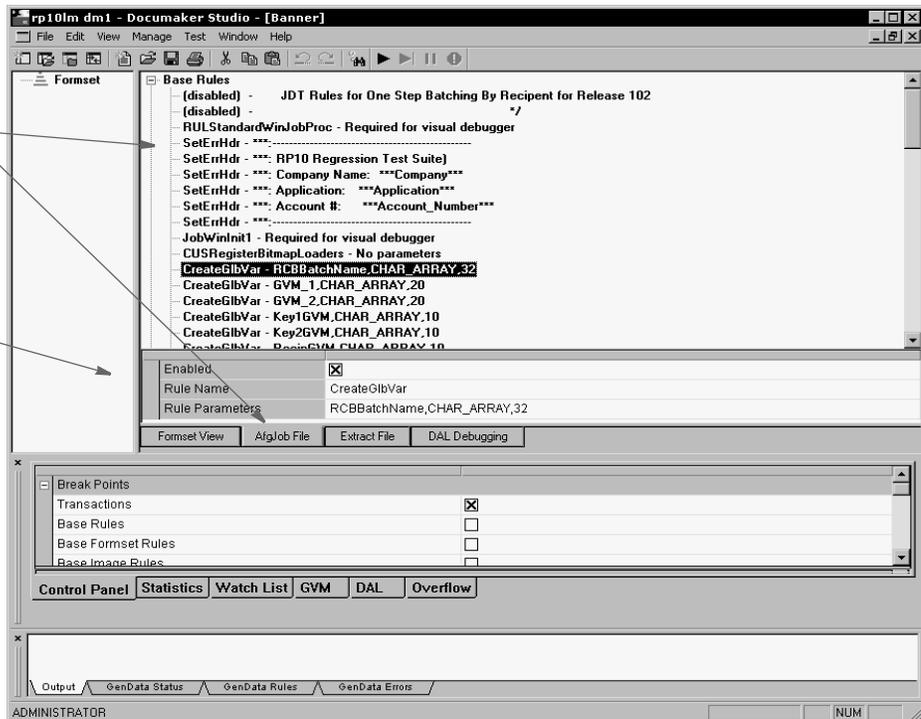
Make the appropriate changes and click Ok to accept them or Cancel to exit without making changes.

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.

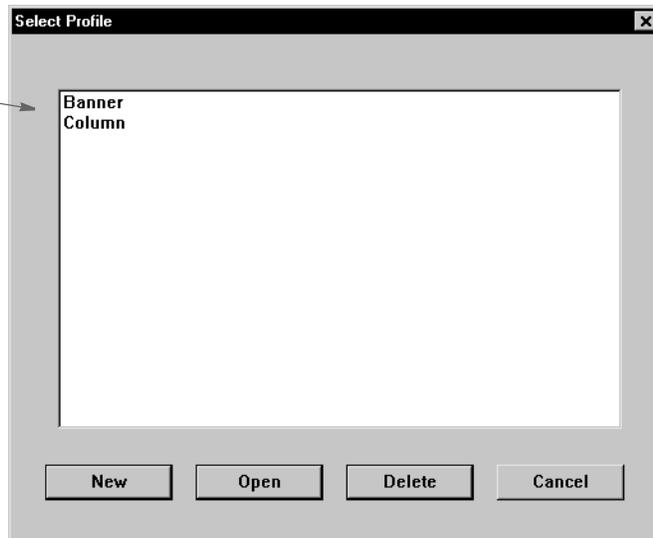
DELETING A TEST

Follow these steps to delete a test.

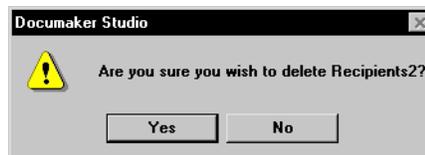
- 1 Click on Tests in the Workspace. The following window appears.

These tests have already been set up.

To delete a test, highlight it and click Delete.



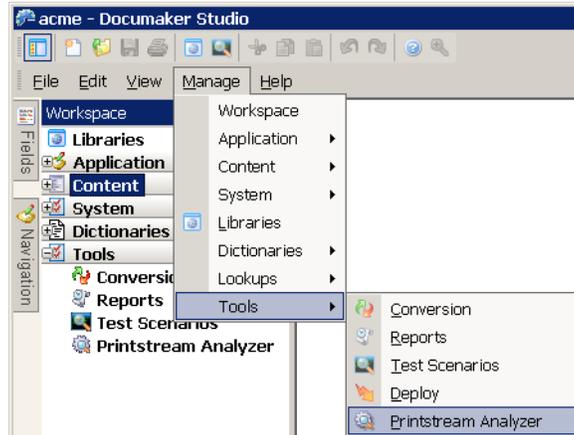
- 2 Highlight the test you want to delete, then click Delete. Studio lets you confirm your decision to delete the test.



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

For this record delimiter type Set the OutMode option to

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.

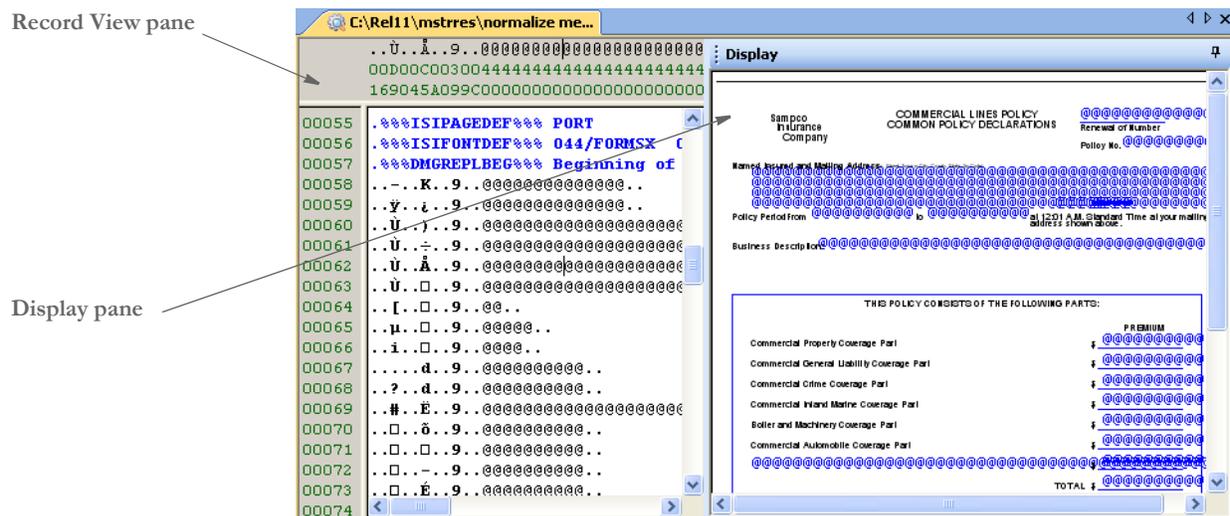
USING THE RECORD VIEW AND DISPLAY PANES

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.

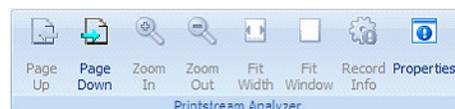
The Display pane, which you can dock and pin, provides a WYSIWIG view of the page that contains the records shown in the Record View pane. The text you select in the Record View pane is highlighted in the Display pane.

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:



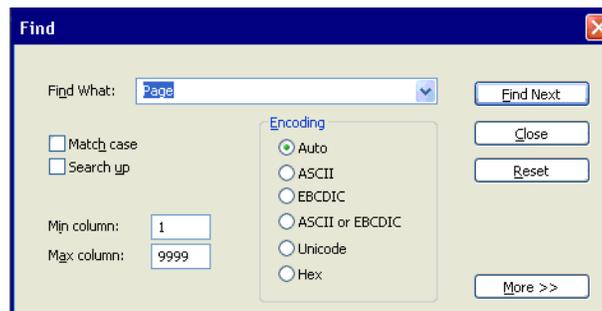
Icon	Description
	Click this icon to go to the next page.
	Click this icon to go to the previous page.
	Click this icon to incrementally zoom in on the page.
	Click this icon to incrementally zoom out on the page.
	Click this icon to resize the page display so you see the entire width of the page.

Icon	Description
Fit Window	Click this icon to resize the page display so you see the entire page in the window.
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 Displaying Information about the Record on page 389 for more information.
Properties	Click this icon to see information about the print steam, 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.

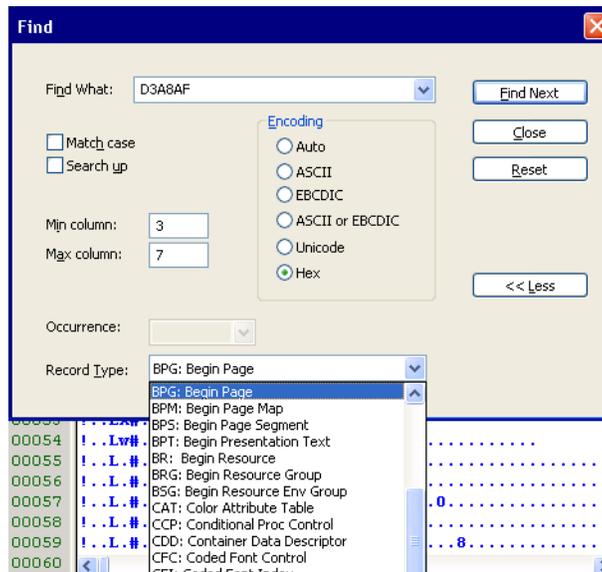
In AFP, text is usually EBCDIC. In Metacode, text is usually ASCII. Comment records in Metacode are EBCDIC so occasionally it is necessary to choose EBCDIC encoding when you are searching in a Metacode print stream.



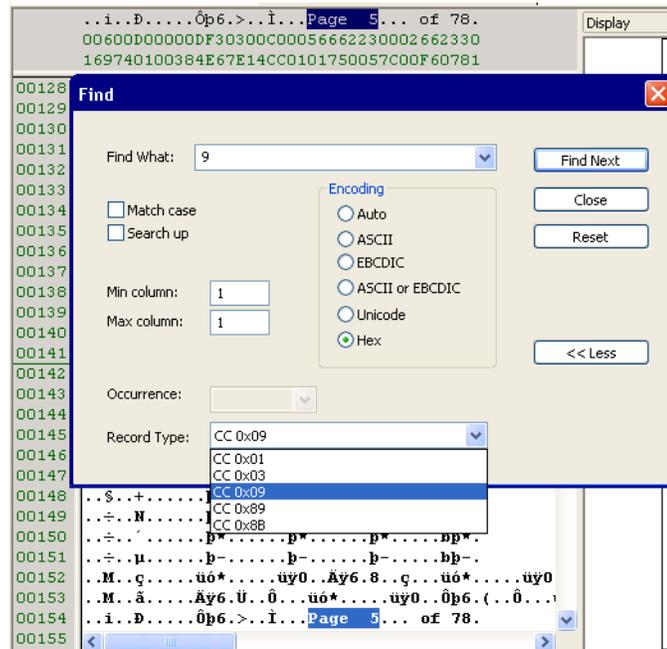
Select the Hex option and enter minimum and maximum column values to find records with a particular identifier type or a carriage control byte. Click More to automatically set the Hex encoding and appropriate minimum and maximum column values for AFP and Metacode.

Here are some examples:

AFP advanced



Metacode advanced



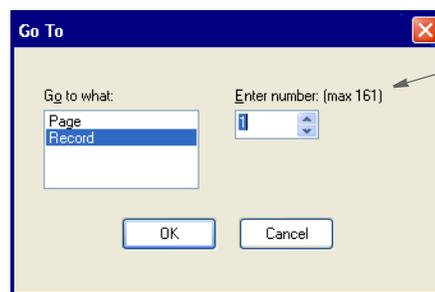
FINDING A SPECIFIC PAGE OR RECORD

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 printstream, 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 printstream.

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 printstream.

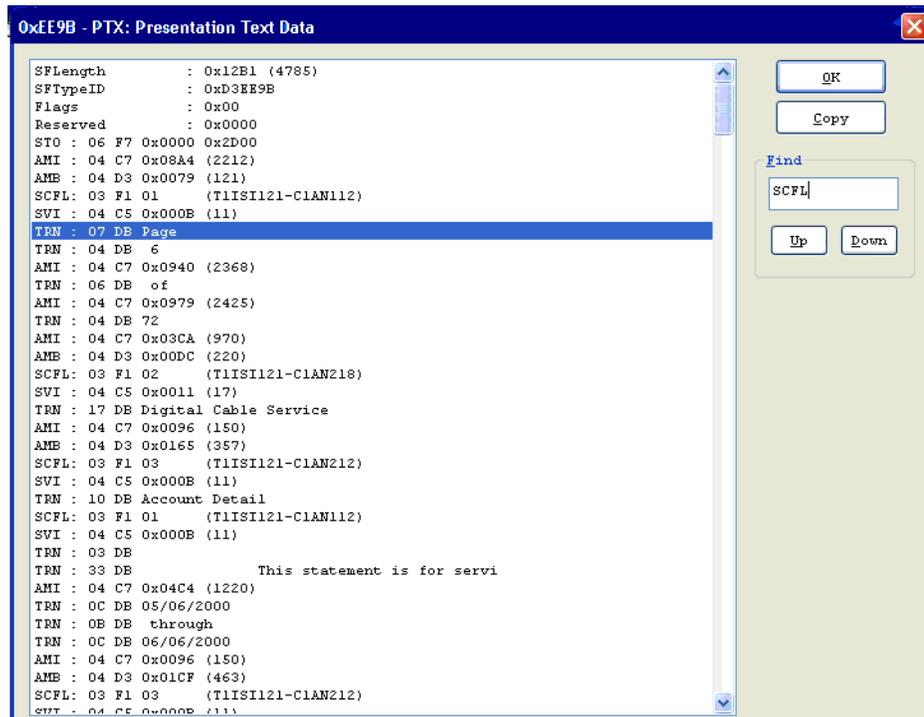
DISPLAYING INFORMATION ABOUT THE RECORD

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.

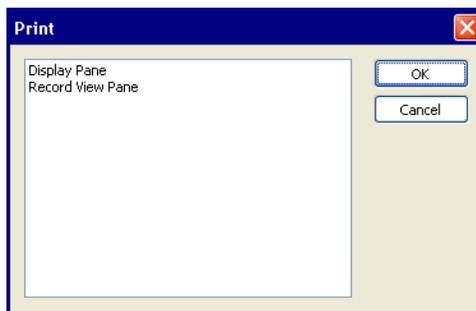
Use the Up and Down buttons in the Find group to search for commands in the record.

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 393](#)
- [Additional Resources on page 398](#)
- [INI Settings on page 399](#)
- [Processing the Deployment on page 400](#)

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.

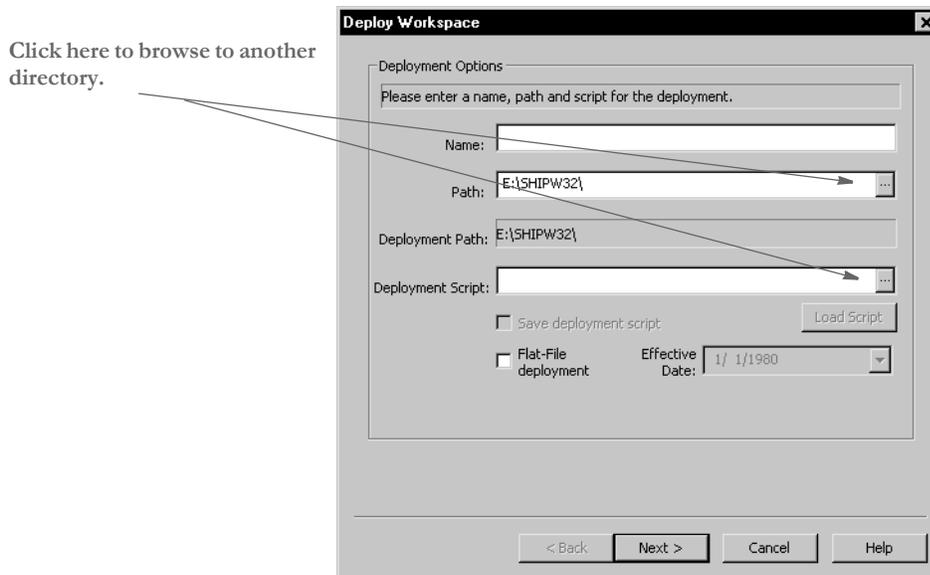
NOTE: A deployment only copies resources that have been checked into the library.

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.

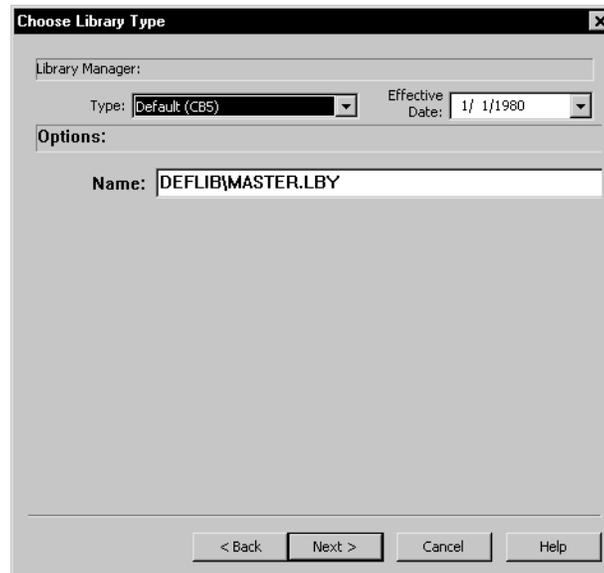
Once you have identified the name and location of the deployment and have either accessed an existing script, entered a new script, and checked whether you want to save the new script, click Next to go to the next step in the wizard. This step differs, depending on whether you chose a flat-file or library deployment.

If you chose a	Go to
library deployment	Creating a Library Deployment on page 395
flat-file deployment	Creating a Flat-File Deployment on page 397

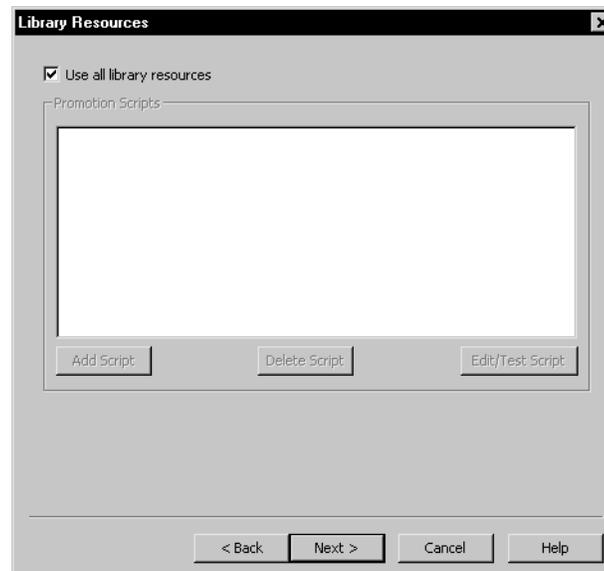
CREATING A LIBRARY DEPLOYMENT

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.

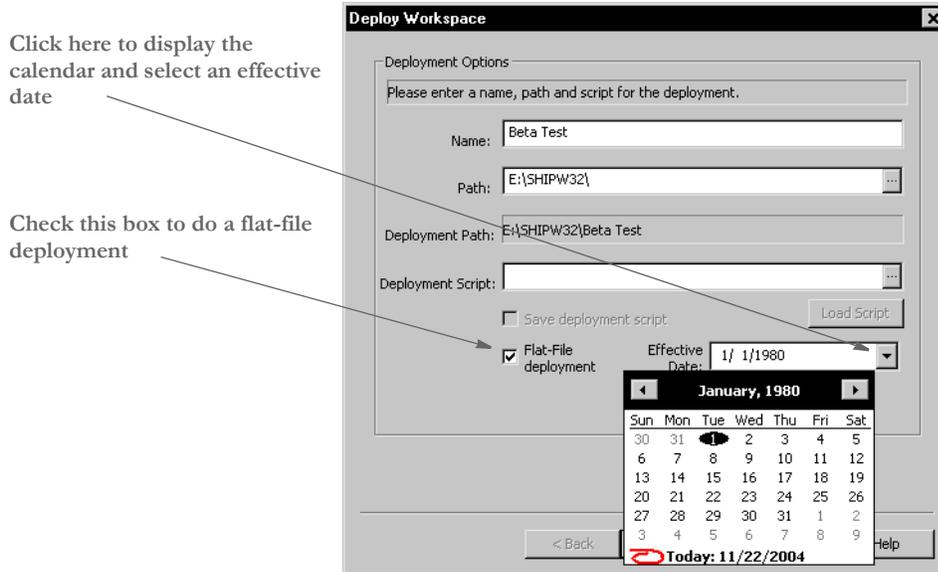
Add Script lets you name one or more promotion scripts you want to use to move resources to the deployment destination. The promotion scripts you choose are shown in a list. You can add or delete them as needed. In addition, you can click Edit Script to edit or create promotion scripts. This takes you 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.



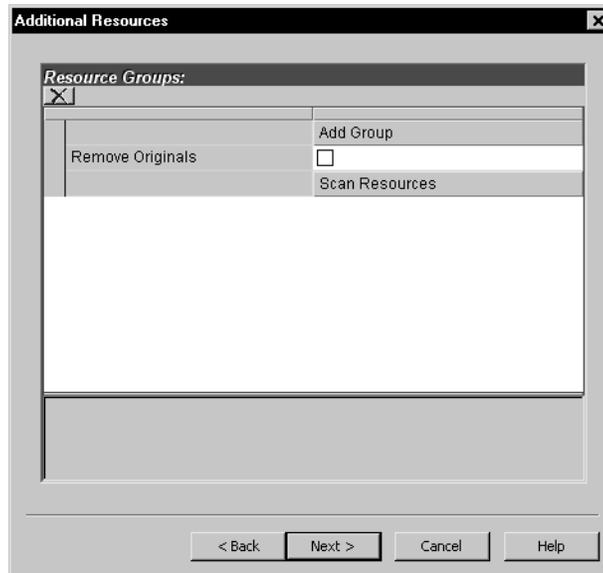
Press ESC to close the calendar.

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.

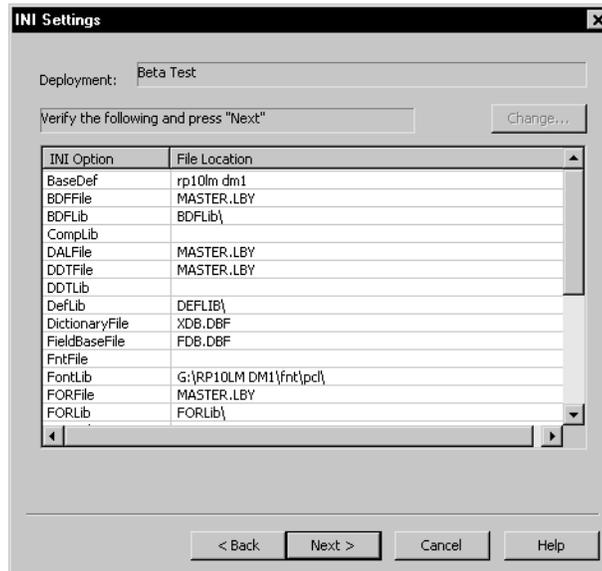
The Other Resources category is where you identify files that are not known to the workspace. Here you can add files including the source location and destination location where the file should be copied.

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.



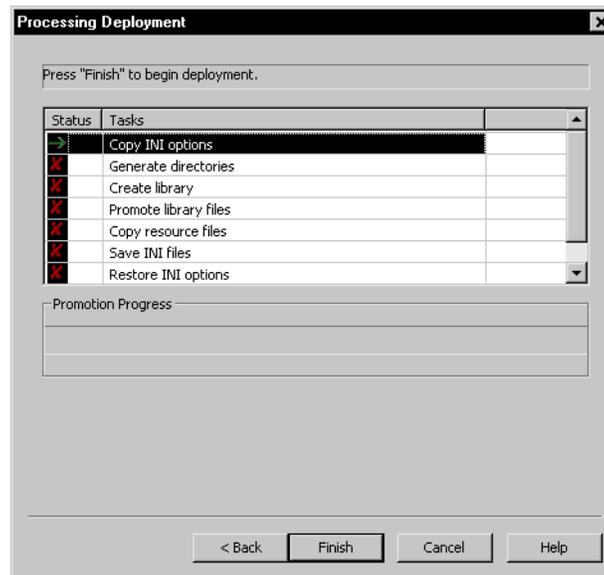
Once you finish, click Next to move to the Processing Deployment window.

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.



Index

A

Action field 315

AFP

- converting page segments 212

application definition file

- overview 90

Application manager

- granting rights 74

ARCEFFECTIVEDATE attachment variable 250

archive

- and libraries 248

- INI options 43

ArcSplit

- INI options 43

ASCII encodation 189

Auto Lasso

- options 57

Auto Save 48

AutoPagination option 180

B

BDFFile option 252, 256, 261, 266, 271

BDFLib option 292

Index

bitmap files
 converting 212
 effective dates 240

black and white
 converting graphics 215

brace matching
 defining a color 63

breakpoints 366

business units 23

C

CARFile 252

CARFILE.DFD file 269

CARFileDFD option 270

CARFiles
 creating libraries 252

categories
 defining 97

check in
 defined 235

check out
 defined 235

Class field 286, 294, 297, 298, 300, 303, 306, 310, 315, 318

Class option 258, 263, 267, 272

clipping
 logos 215

cloning users 77

comments
 defining a color 63

Common Fields manager
 granting rights 75

common INI options 43

constants
 defining a color 62

copy counts
 using variables 160

create dates
 overview 243

Create New Workspace option 28

CreateIndex option 258, 263, 267, 273, 323

CreateTable option 254, 258, 260, 263, 267, 273

CreateTime field 243, 245, 247, 248

cross hairs 54, 56

D

DAL scripts
 defining editor settings 62

DAL variables 161

DALFile option 249, 252, 256, 261, 266, 271

data code word 188

Data Entry check 174

Data Extract manager
 granting rights 75

Data Matrix
 bar codes 188
 symbol sizes 188

database
 INI options 43

Database option 258

databases
 storing user information 78

DB2 43
 using the native driver 254
 using the ODBC driver 259

DBHandler option 257, 258, 262, 263, 266, 267, 272

DBTable
 USERINFOSQL control group 78

DBTable option 257, 262, 266, 272

DDT files
 eliminating 4
 omitting 48
 stringent checking 241

DDTFile option 253, 256, 261, 266, 271

DDTLib option 292

Debug option 258, 263, 267, 273, 323

Definition Lookup manager
 granting rights 75

DEFLib option 292

- Deployment manager
 - granting rights 75
- Development Tools
 - INI options 43
- dictionaries 23
- Dictionary-Rule manager
 - granting rights 75
- Documaker Bridge
 - and Library manager 234
- Documaker Server
 - and libraries 248
 - and Library manager 234
- Documaker Workstation
 - and libraries 245
 - and Library manager 234
- Documanager 31
- Docupresentation
 - and Library manager 234, 249
- DPRInitLby rule 249
- DPRRetrieveFormset rule 250
- dprtrc.log file 322
- DXM files 50
- DXS files 50

E

- Edit menu 20
- effective date 31
- Effective Date field 31, 285, 294, 300, 302
- effective dates
 - defined 235
 - overview 239
 - printer resource files 240
 - setting rights 73
 - ShowEffectiveDate option 245
 - versions and revisions 240
- Enable_Debug_Options option 322
- entry
 - INI options 43
- ErrFile option 241
- ErrorOnMissingFile option 241
- expiring resources 73
- export
 - INI options 43
- extract files
 - Primary Extract Dictionary field 92

F

- FAP files
 - and DDT files 4
- fields
 - showing tips 54
 - tips 57
- File Information window 250
- File menu 18
- filters
 - defining settings 58
- Font check 174
- font cross-reference
 - defining 92
- font cross-reference files 30
- Font manager
 - granting rights 75
- fonts 24
 - effective dates 240
 - matching 43
- footers 139
- FORFile option 253, 256, 261, 266, 271
- FORLib option 292
- Form List manager
 - granting rights 74
- FormFile option 249, 253, 257, 262, 266, 271
- FormLib option 292
- forms 23
 - assigning categories 97
 - creating 76
 - effective dates 240
 - terminology 122

Index

Forms manager
 granting rights 74
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

G

GetRunDate rule 244, 248
grammar
 checking on a save 57
graphics 23
 clipping 215
 converting 212, 215
 defining settings 60
 effective dates 240
 resizing 215
 resolution 216
 reversing 214
 rotating 214
 white space 215
Graphics manager
 adding white space 215
 granting rights 75
grid
 defining 52
 showing 55, 57, 60
GRPFile option 253, 257, 262, 266, 271
GRPLib option 292
GVM variables 161

H

headers 139

Help menu 26
hooks
 defining 98
HTML file
 stacking objects in order 211

I

importing
 a signature or logo 213
importt
 INI options 43
In Use status 68
Increment Version field 285, 293
indenting 63
Information pane 45
INI options
 access to 71
 setting 72
Insert menu 171

J

JPEG files
 converting 212

K

KeepFiles option 322
keywords
 defining a color 62

L

- LBRYMGR utility
 - and DB2 254
 - and DB2 ODBC driver 259
 - Oracle ODBC driver 270
 - SQL Server ODBC driver 265
- LBYPD option 258, 263, 268, 273
- LBYSI option 258, 263, 268, 273
- LbyLib option 249, 253, 322
- LBYSLog option 258, 264, 268, 273
- LbyLogFile option 253, 257, 262, 266, 271
- LBYSQLR job 255
- LDAP
 - setting up 82
 - using 81
- LDAP_Enabled option 82
- libraries 23
 - access to 71
 - creating 73, 251
 - creating a DB2 library 254
 - data portion 239
 - default name 252
 - defined 235
 - format 252
 - how it all works 245
 - index tables 237
 - INI options 44
 - overview 234
 - purging history 73
 - setting access rights 72
 - understanding 237
- Library field 296, 305, 309, 315
- Library Manager
 - error messages 323
- library tiers
 - INI options 44
- Lightweight Directory Access Protocol 81
- line numbers 63
- line of business
 - overview 90
- Locale option 44

- Locked field 300
- logging in 77
- LogoFile option 249, 253, 257, 262, 266, 271
- LogoLib option 213, 292

M

- margins
 - showing 54, 56
- master.ddt files
 - Dictionary Rule manager 50
- MASTER.LBY file 252
- Menu Bar 123, 124, 165, 167, 202, 203, 220, 221, 366
- menu bar 123, 165, 202, 220
- menus
 - Edit 20
 - File 18
 - Help 26
 - options 16, 23
 - Tools 21
- Mode field 286, 294, 297, 298, 300, 302, 306, 310, 315, 317
- modification date
 - defined 235

N

- Name field 296, 300, 306, 309, 314
- NetMove
 - configuring 196

O

- objects
 - hiding 58
 - stacking order 211
 - terminology 122
- ODBC 31, 43

Index

operators
 defining a color 63
Options menu 23
Oracle 43
 ODBC driver 269
overlays
 effective dates 240

P

pages
 showing boundaries 54
pagination 180
Paragraph List manager
 granting rights 74
Paragraph manager
 granting rights 74
Passwd option 258, 263, 267, 273
passwords 68
placeholders
 field style 52
PNG files
 importing 211
PNG formats 211
policy date 31
positioning
 JPG objects 211
Poweroffice
 using with Library Manager 274
primary extract dictionary
 defining 92
Primary Extract Dictionary field 92
print
 INI options 44
Printcommander
 creating FAP files 193, 199
 downloading and installing 193
 fidelity 200
 virtual printers 194

printers
 resource files 240
Printstream Analyzer
 setting up 64
Project field 286, 294, 297, 298, 301, 303, 306, 310, 315,
 318
project management
 library tiers 44
projects
 assigning roles 71
 granting rights 76
promotion
 defined 235
promotions
 setting rights 72

Q

Qualifier option 263, 267, 273

R

Recip_Names control group 96
recipients
 defining 96
RecStatus column 291
regional date processing 99
RegionalDateProcess rule 99
relative paths 29
RemoveImageMissingDDT option 241
reports
 business unit 108
resizing
 logos 215
Resolution option
 Graphics manager 216

resources

- access to 71
 - checking in 293
 - checking out 291
 - creating 76
 - editing information 302
 - granting rights 73
 - path setup 44
 - securing 73, 79
 - unlocking 295
- response files
- defined 235
- RetrieveVersionInfo option 246
- reversing
- logos 214
- revision
- defined 235
- Revision field 297, 300, 306, 310
- revisions
- overview 240
 - run dates 243
 - stringent checking 241
- rotating
- logos 214
- Ruler Units option
- Logo Manager 217
- rulers
- showing 54, 57, 60
- rules processing
- INI options 44
- run date 31
- format 243
- run dates
- overview 243
- RunDate field 243, 245, 247, 248, 250
- RunDate GVM 248
- RunDate option 243

S

- sandbox
- location 72
- scratch pads 36
- Script Name field 301
- scripts 23
- sections 23
- Sections manager
- granting rights 74
- security rights
- LDAP 81
- Security wizard 69, 70
- Send Copy To fields 96
- Server option 263, 268, 273
- SetOrigin rule 3
- SetOrigin rules 181
- ShowEffectiveDate option 245
- signatures
- importing 213
- Spell check 174
- spelling
- checking on a save 57
- SQL 31
- ODBC driver 265
- stacking order 211
- state compliance 99
- Status Bar 123, 165, 202, 220, 366
- status bar 48
- Status field 250, 286, 294, 297, 298, 300, 303, 306, 310, 315, 318
- status line 123, 165, 202, 220, 366
- StringentChecking option 241
- Style File Name field 92
- Style manager
- granting rights 74
- system administrator
- designating 71

T

 Tabs
 defining 63

Tagcommander 195, 330

Template manager
 granting rights 75

templates 139
 creating forms without a template 76

terminology
 forms 122
 objects 122

test 24

Text Case field 307

Text Element Name field 307

Text String field 307

TGA files
 importing 330

Title 123, 165, 202, 220, 366

title bar 123, 165, 202, 220, 366

tool bar 123, 165, 202, 220, 366

tool bars 48

Tools menu 21

trace files 322

TraceFile option 322

transaction types
 defining 98

Trigger manager
 granting rights 75

triggers
 creating 76
 defining editor settings 62
 report 147

Type field 297, 300, 306, 309, 315

U

unexpiring resources 73

UniqueTag option 257, 258, 262, 263, 267, 272

units of measurement
 defining 52

unlock
 defined 235

user database
 managing 71

User ID field 300, 315

UserID column 291

UserID option 258, 263, 268, 273

USERINFOSQL table 78

users 24
 adding 68
 changing information 70
 cloning 77
 defining 33
 deleting 70
 managing 66, 71
 Security wizard 69
 setting rights 70
 specifying a login message 68
 storing information 78
 supervisors 68
 user IDs 67, 68
 using LDAP 81

V

variable fields
 creating 76

Version field 297, 300, 306, 309

versions
 defined 235
 overview 240
 run dates 243
 stringent checking 241

View menu 170

W

WarnOnMissingFile option 241

White Space option

 Logo Manager 215

WIP

 and libraries 247

 INI options 44

workflow 250

Workspace 123, 126, 165, 202, 220

workspaces

 creating 27

 joining 34

 managing a shared workspace 71

 new 12

 shared 50

workstation access levels 68

X

Xerox

 converting font files 212

 converting image files 212

Xerox font

 importing signatures or logos 213

Z

Z-Index option 211

