

# Oracle® WebCenter Forms Recognition

Designer User's Guide  
11g Release 1 (11.1.1.8.0)

E50187-01

November 2013

## WebCenter Forms Recognition

11g Release 1 (11.1.1.8.0)

Copyright © 2009, 2013 Oracle and/or its affiliates. All rights reserved.

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

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

If this software or related documentation is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

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

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

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

# Contents

<b>1</b>	<b>About WebCenter Forms Recognition Designer .....</b>	<b>10</b>
<b>2</b>	<b>About This Manual .....</b>	<b>11</b>
	2.1 Intended Audience .....	11
	2.2 Reading Suggestions .....	11
	2.3 Related Documentation .....	11
<b>3</b>	<b>Project Migration .....</b>	<b>13</b>
	3.1.1. Migrating Projects to WebCenter Forms Recognition 11.1.1.8.0.....	13
	3.1.2. Project with Table Analysis Engine to the Brainware Table Extraction Engine .....	13
	3.1.3. Project with Template Classification engine to the Brainware Layout Classification engine.....	14
	3.1.4. Project with Sax Basic to the WinWrap Scripting Engine .....	15
	3.1.5. Project with Validation Script Code to the New Validation Templates and Output Formatting .....	15
	3.1.6. Migration to the Supervised Learning Workflow .....	16
<b>4</b>	<b>Basic Concepts and Techniques .....</b>	<b>21</b>
	4.1 Starting Designer .....	21
	4.2 About Roles, Users, and Authentication .....	21
	4.3 Logging In .....	23
	4.3.1. Logging In to Designer .....	23
	4.3.2. Logging In to Verifier .....	23
	4.4 Creating Users and Groups .....	23
	4.4.1. User Interface.....	23
	4.4.2. Establishing and Changing Passwords .....	23
	4.4.3. Creating the Administrator Password .....	24
	4.4.4. Establishing User Groups.....	24
	4.4.5. Creating Groups.....	25
	4.4.6. Creating and Managing User Accounts.....	27
	4.4.7. Exporting of Users, Roles and Groups to the WebCenter Forms Recognition Database.....	29
	4.5 Windows Based User Authentication .....	29
	4.5.1. Enabling Windows Authentication .....	29
	4.5.2. Importing Windows accounts .....	30
	4.5.3. Windows Authentication Settings .....	32
	4.5.4. Windows Authentication for Web Verifier .....	33
	4.5.5. Usage.....	34
	4.6 Configuring Licensing Properties .....	34
	4.6.1. Description .....	34
	4.7 Exiting Designer .....	35
	4.8 Working with Modes.....	35
	4.8.1. What Are Modes? .....	35
	4.8.2. Document Selection Mode .....	35
	4.8.3. Definition Mode .....	48
	4.8.4. Normal Train Mode .....	53
	4.8.5. Verifier Train Mode.....	56
	4.8.6. Runtime Mode.....	60
	4.8.7. Verifier Design Mode.....	64
	4.8.8. Verifier Test Mode .....	67
	4.9 Working with Projects.....	70
	4.9.1. What Are Project Files?.....	70
	4.9.2. Creating Projects.....	71
	4.9.3. Saving Projects .....	71
	4.9.4. Opening Projects.....	72
	4.9.5. Using Version Control .....	73
	4.9.6. Setting Global Project Variables.....	74
	4.9.7. Making Projects Portable .....	76
	4.10 Working with Documents .....	77

4.10.1.	What are Workdocs? .....	77
4.10.2.	Viewing Documents .....	77
4.10.3.	Processing Documents .....	80
4.10.4.	Learning .....	81
4.10.5.	Highlighting Processing Results .....	81
<b>5</b>	<b>Setting Up the Classification .....</b>	<b>83</b>
5.1	Advanced Classification .....	83
5.2	Preparing Sample and Test Documents .....	84
5.3	Creating the Project .....	84
5.4	Creating Classes .....	84
5.4.1.	Custom Class Names .....	86
5.5	Editing Classes .....	89
5.6	Selecting Classification Methods .....	90
5.7	Configuring Classification .....	91
5.7.1.	Creating Learnsets .....	91
5.7.2.	Encrypting a Learnset .....	93
5.7.3.	Security Extensions for Learnset Encryption .....	93
5.7.4.	Editing Learnsets .....	97
5.7.5.	Learning .....	99
5.7.6.	Checking the Learn Status of a Class .....	100
5.8	Configuring Brainware Layout Classification Engine .....	100
5.8.1.	Overview & Purpose .....	101
5.8.2.	How it Works .....	101
5.9	Configuring Language Classification Engine .....	101
5.9.1.	Overview & Purpose .....	101
5.9.2.	How it Works .....	102
5.9.3.	Setup the Language Classification Engine .....	103
5.9.4.	Extending and Adjusting Standard Learnset with New Languages .....	103
5.10	Configuring Phrase Classification .....	105
5.11	Configuring Image Size Classification .....	107
5.12	Configuring Forms Classification .....	108
5.12.1.	Creating Reading Zones .....	109
5.12.2.	Global OCR Zones in Forms Classification .....	112
5.13	Choosing Classification Method .....	113
5.13.1.	The ASSA Classification Engine .....	113
5.13.2.	ASSA Configuration .....	113
5.14	Testing the Classification .....	114
5.14.1.	In Definition Mode .....	115
5.14.2.	In Train Mode .....	116
5.14.3.	In Runtime Mode .....	116
5.15	Configuring the ASSA Engine for Classification in Automatic Supervised Learning .....	119
5.16	Optimizing the Classification .....	119
5.16.1.	Resolving Problems with the OCR .....	119
5.16.2.	Resolving Problems with the Classification Methods .....	120
5.16.3.	Resolving Problems with the Learnset .....	120
<b>6</b>	<b>Planning Applications .....</b>	<b>121</b>
6.1	Identifying the Document Import Formats .....	121
6.2	Identifying the Document Classes .....	121
6.3	Planning the Classification Methods .....	122
6.3.1.	Content Classification with Brainware .....	122
6.3.2.	Phrase Classification .....	122
6.3.3.	Template Classification .....	123
6.3.4.	Image Size Classification .....	123
6.3.5.	Forms Classification .....	123

- 6.3.6. Brainware Layout Classification ..... 123
- 6.4 Identifying the Fields for Data Extraction ..... 123
- 6.5 Planning the Extraction Methods ..... 123
  - 6.5.1. Brainware Table Extraction ..... 124
  - 6.5.2. Table Analysis ..... 124
  - 6.5.3. Format Analysis ..... 124
  - 6.5.4. Zone Analysis ..... 124
  - 6.5.5. Address Analysis ..... 125
  - 6.5.6. Associative Search Engine ..... 125
- 6.6 Planning the Verification ..... 125
- 6.7 Planning Supervised Learning ..... 125
- 6.8 Planning the Document Export ..... 126
- 6.9 Planning the Page Separation ..... 126
  - 6.9.1. Batch Properties ..... 126
  - 6.9.2. Multipage Detection ..... 126
  - 6.9.3. Page Separation Learnset ..... 128
  - 6.9.4. How to Train the Engine ..... 129
  - 6.9.5. Project Properties for Page Separation ..... 130
- 7 Setting Up the Validation ..... 132**
  - 7.1 Validation Engine ..... 132
  - 7.2 Validation Concepts ..... 132
    - 7.2.1. Levels of Validation ..... 133
    - 7.2.2. Terms and Commands You Should Know ..... 133
    - 7.2.3. Available Validation Settings ..... 135
    - 7.2.4. Types of Field Validations ..... 135
  - 7.3 Working with Validation Levels ..... 140
    - 7.3.1. Working with Project-Level Settings ..... 140
    - 7.3.2. Working with Document- Level or Class-Level Validation ..... 141
    - 7.3.3. Working with Field-Level Validation for Text ..... 143
  - 7.4 Working with Validation Templates ..... 145
    - 7.4.1. Creating Templates ..... 146
    - 7.4.2. Working with Validation Templates at the Field Level ..... 146
    - 7.4.3. Working with Validation Fields at the Class Level ..... 147
    - 7.4.4. Working with Validation Templates at the Project Level ..... 147
  - 7.5 Introduction to Validation Scripts ..... 147
  - 7.6 Stabilization & Enhancements of the Standard Validation Engine ..... 148
    - 7.6.1. Description ..... 148
    - 7.6.2. Usage ..... 156
- 8 Setting Up the Data Extraction ..... 157**
  - 8.1 Brainware Lines Extraction Method ..... 157
    - 8.1.1. Description ..... 157
    - 8.1.2. Usage ..... 158
  - 8.2 Setting up the Fields ..... 159
    - 8.2.1. Creating Fields ..... 159
    - 8.2.2. Editing Fields ..... 162
  - 8.3 Selecting the Analysis Method ..... 163
  - 8.4 Setting Up Brainware Table Extraction ..... 163
    - 8.4.1. About Brainware Table Extraction ..... 164
    - 8.4.2. Configuring Generic Table Extraction ..... 167
    - 8.4.3. Custom Column Names ..... 169
  - 8.5 Setting up Format Analysis ..... 172
    - 8.5.1. Defining the Format Strings ..... 174
    - 8.5.2. Defining the Rules for String Construction from Words ..... 177
    - 8.5.3. Restricting the Analysis to Certain Areas Within the Document ..... 178

8.5.4.	Support of Character Encoding Tables ISO/IEC 8859-2, -5, and -9.....	179
8.5.5.	Ability to Create Associate Search Engine Pool Imported from an ODBC Source with Entries in “Non-Western” Languages.....	179
8.6	Setting up Zone Analysis .....	179
8.6.1.	Creating Reading Zones .....	180
8.6.2.	Editing Reading Zones .....	181
8.6.3.	Fixing the Coordinate System for the Reading Zones .....	182
8.6.4.	Mapping Reading Zones to Document Fields .....	184
8.7	Setting up Address Analysis .....	184
8.7.1.	Creating the Address Pool .....	185
8.7.2.	Configuring Address Analysis .....	186
8.8	Setting Up Table Analysis.....	188
8.8.1.	Defining Table Columns.....	190
8.8.2.	Defining Column Labels and Formats .....	192
8.8.3.	Defining Header and Footer Lines .....	193
8.8.4.	Determining Table Tops.....	193
8.8.5.	Defining Column Layouts .....	195
8.8.6.	Determining Table Bottoms.....	196
8.8.7.	Managing Comment Lines .....	197
8.8.8.	Using Field Inheritance.....	198
8.8.9.	Column Mapping.....	198
8.9	Configuring Associative Search Analysis .....	199
8.9.1.	Usage of Associative Database Search Engine .....	200
8.9.2.	Creating Data Sources .....	200
8.9.3.	Importing Reference data for the Associative Search .....	208
8.9.4.	Setting up the Analysis.....	210
8.9.5.	Input of Documents .....	213
8.10	Setting Up the Field Evaluation .....	213
8.11	Brainware Field Extraction Engine for Generic Fields Extraction.....	214
8.11.1.	Description .....	214
8.11.2.	Restrictions .....	216
8.11.3.	Scripting .....	216
8.11.4.	Usage & Notes .....	216
8.12	Training of Header Fields in Normal Train Mode without Configuring Field Formats .....	217
8.12.1.	Description .....	217
8.13	Learning the Extraction.....	220
8.13.1.	Creating Learnsets.....	220
8.13.2.	Editing Learnsets .....	223
8.13.3.	Learning .....	223
8.13.4.	Checking the Learn Status of a Field .....	223
8.14	Testing the Extraction.....	224
8.15	Optimizing the Extraction.....	227
8.15.1.	Resolving Problems with Incomplete Configuration .....	227
8.15.2.	Resolving Problems with the Learnset .....	228
8.15.3.	Resolving Problems with the OCR .....	228
8.16	Applying Extraction Retaining Previously Available Extraction Results .....	228
<b>9</b>	<b>Setting Up Supervised Learning.....</b>	<b>231</b>
9.1	Training the Base Class .....	231
9.2	Training Other Fields.....	232
9.3	Creating Derived Document Classes .....	232
9.3.1.	Options for Creating Derived Classes in Designer .....	232
9.3.2.	Options for Creating Derived Classes in Verifier .....	234
<b>10</b>	<b>Advanced Recognition Settings .....</b>	<b>235</b>
10.1	Scope of Recognition Settings .....	235

10.1.1.	Project-Level Settings .....	235
10.1.2.	Page-Level Settings .....	237
10.1.3.	Zone-Level Settings .....	239
10.1.4.	Field-Level Settings.....	240
10.2	Engine-Independent Settings .....	241
10.2.1.	General Tab .....	241
10.2.2.	Preprocessing Tab .....	242
10.2.3.	Anchors Tab.....	243
10.2.4.	Recognition Tab .....	244
10.2.5.	Test Tab .....	245
10.3	FineReader OCR Engine.....	245
10.3.1.	FineReader 8.1 and 10.....	246
10.3.2.	FineReader 10 .....	248
10.4	Recognita OCR Engine .....	250
10.4.1.	General Tab .....	250
10.4.2.	Preprocessing Tab .....	251
10.5	Kadmos 5 Engine .....	252
10.5.1.	General Tab .....	252
10.5.2.	Characters Tab .....	253
10.5.3.	Digits Tab .....	254
10.5.4.	Preprocessing Tab .....	254
10.5.5.	Position Tab .....	255
10.6	Recognita Barcode Engine.....	256
10.6.1.	General Tab .....	256
10.6.2.	Preprocessing Tab .....	256
10.7	Cleqs Barcode Engine.....	257
10.7.1.	General Tab .....	257
10.7.2.	Restrictions Tab .....	259
10.7.3.	Preprocessing Tab .....	260
10.8	Cairo OMR Engine .....	261
10.8.1.	OMR Zones.....	261
10.8.2.	General Tab .....	261
10.9	OCR Field Level Digit booster.....	263
<b>11</b>	<b>Regular Expressions .....</b>	<b>265</b>
11.1	What are Regular Expressions?.....	265
11.2	Literal Characters in Regular Expressions .....	265
11.3	Operators in Regular Expressions.....	265
11.3.1.	Example: Find an Invoice Number .....	266
11.3.2.	Example: Find a Date.....	267
11.3.3.	Example: Find an E-mail Address .....	268
<b>12</b>	<b>Advanced Evaluation Settings.....</b>	<b>269</b>
12.1	Project-Level Settings.....	269
12.1.1.	Classification Interpretation – How Does It Work? .....	269
12.1.2.	Project-Level Standard Classification – How Does It Work? .....	270
12.1.3.	Project-Level Parent Classification – How Does It Work? .....	271
12.1.4.	Project-Level Default Classes – How Do They Work? .....	272
12.1.5.	Modifying Project-Level Settings .....	272
12.2	Method-Level Classification Settings .....	273
12.2.1.	Method-Level Absolute Results – How Does They Work? .....	273
12.2.2.	Method-Level Multiple Views – How Do They Work?.....	274
12.2.3.	Modifying Settings for Brainware Classification Methods .....	276
12.2.4.	Modifying Brainware Layout Classification Settings at the Method Level.....	278
12.2.5.	Modifying Phrase Classification Settings at the Method Level .....	279

12.2.6.	Modifying Image Size Classification Settings at the Method Level.....	281
12.2.7.	Modifying Forms Classification Settings at the Method Level .....	281
12.3	Class-Level Settings .....	282
12.3.1.	Class-Level Subtree Classification – How Does it Work? .....	282
12.3.2.	Class-Level Redirection – How Does it Work? .....	283
12.3.3.	Modifying Class-Level Settings .....	283
12.3.4.	Modifying Settings for Brainware Layout Classification Methods .....	284
12.4	Field-Level Settings .....	287
12.4.1.	Field-Level Text Field Candidate Evaluation – How Does it Work? .....	287
12.4.2.	Modifying Text Field Settings .....	288
12.4.3.	Table Field Candidate Evaluation – How Does it Work? .....	289
12.4.4.	Modifying Table Field Settings .....	289
12.4.5.	Field-Level Digit booster Candidate Evaluation.....	289
<b>13</b>	<b>Setting Up the Verification .....</b>	<b>290</b>
13.1	Creating Verification Forms .....	290
13.1.1.	Managing Verification Projects.....	290
13.1.2.	Configuring Project Validation Properties.....	291
13.1.3.	Setting Project Validation Properties .....	292
13.1.4.	Verification Forms .....	292
13.1.5.	Configuring Form Validation Properties .....	296
13.1.6.	Configuring the Verification Form Layout.....	297
13.1.7.	Configuring Form Grids.....	298
13.1.8.	Creating Form Elements .....	298
13.1.9.	Creating and Modifying Form Fields .....	300
13.1.10.	Editing Form Elements.....	302
13.1.11.	Setting Field Validation Properties .....	303
13.1.12.	Editing Text Fields.....	304
13.1.13.	Configuring the Viewer Properties.....	307
13.1.14.	Configuring Table Properties.....	309
13.1.15.	Configuring Smart Indexing.....	311
13.1.16.	Database Support for Smart Indexing .....	315
13.1.17.	Correcting Table Fields .....	316
13.1.18.	Shortcut Menu Configuring Actions.....	317
13.1.19.	Changing of Colors & Fonts for Elements of Verification Forms .....	318
13.2	Testing the Verification .....	319
13.2.1.	Verifier Test Mode User Interface .....	319
13.2.2.	Verifier Test Mode Color Coding .....	319
13.2.3.	Verifier Test Mode Icons .....	320
13.2.4.	Verifier Test Mode Toolbar .....	320
13.2.5.	Verifier Test Mode Keyboard Operation .....	321
13.2.6.	Testing the Visible Classes .....	322
13.2.7.	Testing the Verification Form Layout.....	322
13.2.8.	Testing Validation Rules.....	323
13.2.9.	Testing Smart Indexing .....	324
13.2.10.	Testing Table Analysis and Correction.....	326
<b>14</b>	<b>Printing .....</b>	<b>328</b>
<b>15</b>	<b>Reusing Project Settings with Templates .....</b>	<b>329</b>
15.1	Creating Templates .....	329
15.2	Using Evaluation and Analysis Templates within Projects.....	329
15.3	Editing Analysis and Evaluation Templates .....	330
15.4	Exchanging Analysis Templates Between Projects .....	331
15.4.1.	Exporting Templates to Another Project.....	332
15.4.2.	Importing Templates from another Project .....	332
<b>16</b>	<b>Setting Up the Document Export.....</b>	<b>333</b>
<b>Appendix A</b>	<b>Auxiliary Tools .....</b>	<b>334</b>
A1	Auto-Generation of Template Local Projects in SLW.....	334
A2	Auto-Merging of SLW Learnsets .....	335
A3	New Fields in WebCenter Forms Recognition Workdoc Browser Tool.....	336



---

<b>Appendix B</b>	<b>Designer – Quick reference .....</b>	<b>338</b>
<b>Appendix C</b>	<b>Regional and Currency Settings .....</b>	<b>339</b>
<b>Appendix D</b>	<b>Project Structure and Files Extension .....</b>	<b>340</b>
<b>Glossary</b>	<b>.....</b>	<b>342</b>

# 1 About WebCenter Forms Recognition Designer

WebCenter Forms Recognition is a product suite by Oracle Corporation for automatically processing incoming documents.

In principle, WebCenter Forms Recognition deals with any document that is electronically available. This includes scanned images, faxes, e-mails, and files. WebCenter Forms Recognition automatically classifies these documents and extracts meaningful information from them.

As part of the WebCenter Forms Recognition suite, Designer enables you to customize the automatic processing of incoming documents: which document classes are relevant in your enterprise, which information is to be extracted from the classified documents, how the processing results are to be verified.

To process large volumes of documents, WebCenter Forms Recognition organizes documents into batches, which are defined in the WebCenter Forms Recognition project file. The project files and stored settings are automatically passed to the WFR Runtime Server for production processing.

In addition, Designer enables you to test your custom application. All custom settings are saved as a WebCenter Forms Recognition project file. The finished project file is forwarded to the WFR Runtime Server for processing.

WebCenter Forms Recognition Runtime Server runs unattended as a server process in the background. Several mechanisms ensure that the system is stable — meaning that it can automatically recover from most error situations. Multiple instances of WFR Runtime Server can be started simultaneously in a network or on a single machine. These instances cooperate and allow for optimal load distribution.

Quality assurance and correction are done in WebCenter Forms Recognition Verifier. QA is also controlled by the project file. Batches that cannot be automatically processed in their entirety by WFR Runtime Server are forwarded to Verifier, for manual correction by operators who are subject matter experts in the type of document being processed.

Before WebCenter Forms Recognition can be implemented, the underlying network environment must meet certain minimum platform and environmental requirements. This chapter serves as a vehicle to set expectations, on the part of the destined organization, of the minimum infrastructure components that must be in place to ensure a successful implementation of Forms Recognition.

## 2 About This Manual

### 2.1 Intended Audience

This documentation is for users who intend to create custom applications with WebCenter Forms Recognition Designer. We assume that you are an experienced, knowledgeable user of the Microsoft Windows operating system and of Microsoft Windows applications.

### 2.2 Reading Suggestions

This manual is organized as follows:

- [CHAPTER 4 BASIC CONCEPTS AND TECHNIQUES](#) describes the user interface and some basic concepts you need to understand to work with Designer. Furthermore you will learn here about setting up of user accounts and about exporting them to the WebCenter Forms Recognition database. This chapter also contains a functional description of controls you will need to work with Designer. You should be familiar with this chapter before you start working with other chapters.
- [CHAPTER 5 SETTING UP THE CLASSIFICATION](#) describes how to set up document classification in WebCenter Forms Recognition.
- [CHAPTER 6 PLANNING APPLICATIONS](#) outlines the basic steps of application planning.
- [7 SETTING UP THE VALIDATION](#) addresses configuration of validation.
- [CHAPTER 8 SETTING UP THE DATA EXTRACTION](#) describes how to set up data extraction in WebCenter Forms Recognition.
- [CHAPTER 9 SETTING UP SUPERVISED LEARNING](#) discusses Supervised Learning.
- [CHAPTER 10 ADVANCED RECOGNITION SETTINGS](#) contains information about recognition techniques. You will need this information to improve classification and extraction results.
- [CHAPTER 11 REGULAR EXPRESSIONS](#) contains information about regular expressions. You may need this information to improve extraction results obtained with format analysis.
- [CHAPTER 12 ADVANCED EVALUATION SETTINGS](#) contains information about the evaluation and validation mechanism in WebCenter Forms Recognition. You will need this information to improve classification and extraction results.
- [CHAPTER 13 SETTING UP THE VERIFICATION](#) describes how to set up forms in WebCenter Forms Recognition's quality assurance tool, Verifier, and how to test these forms.

### 2.3 Related Documentation

This manual is part of a set of guides and reference materials. Other materials in the documentation suite are:

- **Installation Guide**  
Explains how to install WebCenter Forms Recognition.
- **Runtime Server User's Guide**  
Explains how to administer the WebCenter Forms Recognition batch processing program.

- ***Verifier User's Guide***  
Explains how to use the WebCenter Forms Recognition verification and manual indexing program.
- ***Web Verifier User's Guide***  
Explains how to use the web based extension of Verifier.
- ***Scripting Reference***  
Explains how to develop with the WebCenter Forms Recognition Designer WinWrap Basic.

## 3 Project Migration

WebCenter Forms Recognition with Supervised Learning is fully backward compatible with version 10.1.3.5.0 of Oracle Forms Recognition.

The first time you open a project created in an earlier version of WebCenter Forms Recognition, all the settings will be set to default values. When a project has been migrated to the new version, changes in the script code will not have to be done. However, the settings to validate a field using the standard validation procedures could be configured.

### 3.1.1. Migrating Projects to WebCenter Forms Recognition 11.1.1.8.0

In general, all project files are portable, so existing applications can easily be reused. WebCenter Forms Recognition version 11.1.1.8.0 contains new features that require special attention when an older project file is going to be migrated:

- Project Authentication. Authentication needs to be established for users and roles. For further details, see sections [ABOUT ROLES, USERS, AND AUTHENTICATION](#) to [CREATING USER ACCOUNTS AND GROUPS](#).
- Project with Table Analysis Engine to the Brainware Table Extraction Engine
- Project with Template Classification engine to the Brainware Layout Classification engine
- Project with Validation script code to the new Validation templates and output formatting
- Migration for the purpose of the Supervised Learning Workflow
- Special considerations should be made with regards to the new database and Web Verifier features.
- Projects using the Sax Basic scripting engine are not supported. They need to be migrated to use WinWrap.

### 3.1.2. Project with Table Analysis Engine to the Brainware Table Extraction Engine

In general, there are no known incompatibility issues between the Table Analysis Engine and the Brainware Table Extraction.

If it is necessary to migrate from the Table Analysis Engine to the Brainware Table Extraction, the following points shall be considered:

- You can convert the Table Analysis Engine to Brainware Table Extraction, and you can convert Brainware Table Extraction to the Table Analysis Engine. During the conversion from one engine to the other, all column names and definitions will be copied. To convert a table from one engine to the other:
  - 1) In Definition Mode, click on the Fields tab on the left side of the screen.
  - 2) Select the Table field to be converted.
  - 3) On the Analysis Editor, select the new table analysis engine (either Brainware Table Extraction or Table Analysis Engine.) The common table settings – column names and definitions – are copied from the original engine to the new engine. .
- If necessary, remove redundant extraction-related script code for the converted table field.

- All scripts for correcting validation and extraction should consider a new case in which a table cell has multiple extracted lines. To set this up, system administrators must use a new “RowNumber” property WebCenter Forms Recognition table object that returns the actual row number for every table line extracted with the Brainware Table Extraction engine.

For more details about batch processing, see the **Scripting Documentation**.

### 3.1.3. Project with Template Classification engine to the Brainware Layout Classification engine

The Template Classification engine is no longer supported and is now fully replaced by the Brainware Layout Classification engine. This is an essential change for Supervised Learning classes.

If you open a project that was created in a prior version and that uses the Template Classification engine for the first time, the following notification will appear:

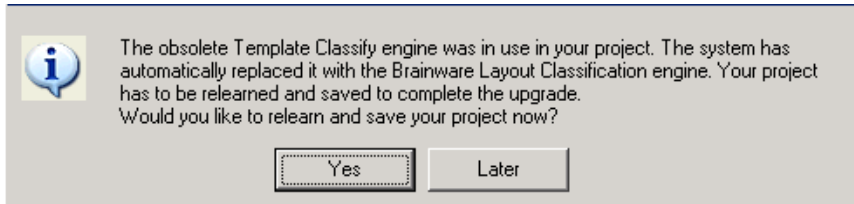


Figure 3-1: Opening a project that uses Template Classification engine

Pressing **Yes** will automatically relearn and save your project. The Brainware Layout Classification engine will be automatically assigned to all of the classes. You can check this in Learnset Input Mode:

	Invoices	Brainw...	Layout ...	ASSA ...	Brainware Layout Classification
1	Doc1	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
2	Doc2	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
3	Doc3	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
4	Doc4	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
5	Doc5	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Figure 3-2: Checking the engines assigned in Learnset Input Mode

Pressing **Later** will not trigger the migration. The project stays opened, and you will be able to check your configuration. However, you will notice that the Brainware Layout Classification engine is already preconfigured for your classes.

**You have two options:**

- Close your project without saving if you do not want to use it with the new version.
- Migrate your project to Brainware Layout Classification engine performing the following steps:
  1. Select *Save Compress* from the *File* menu.
  2. Relearn the project.
  3. Save the project.

*Note: The Brainware Layout Classification engine requires more sample documents in the Learnset than the Template Classification engine. This is especially the case where only one document was used to train the Template Classification engine. However, in order to achieve best classification weightings, a sufficient number of documents in the Learnset is very important.*

*Note: The Brainware Layout Classification engine requires at least two classes defined within a project. If you have a project with only one class, you will be presented with an error message when opening it in Designer for the first time. Please refer to [Chapter SETTING UP THE CLASSIFICATION](#).*

### 3.1.4. Project with Sax Basic to the WinWrap Scripting Engine

The Sax Basic scripting engine is no longer supported. Any existing projects which are using scripts created with Sax Basic need to be updated to use the WinWrap scripting engine.

If you open a project that uses Sax Basic, you will be presented with a reminder message:

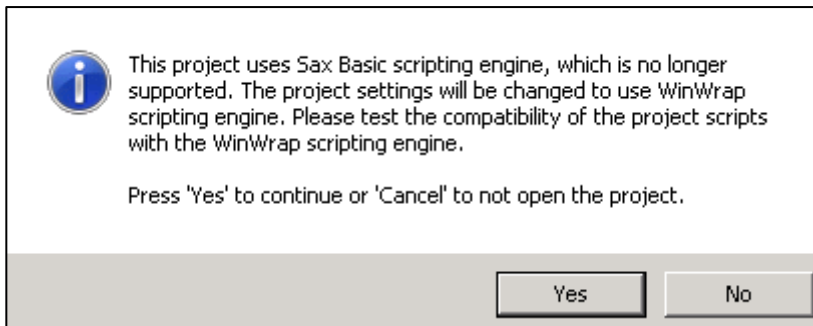


Figure 3-3: Opening a project that uses Sax Basic scripting engine

Press on **Yes** to change the scripting engine to WinWrap.

The project will **NOT** be saved automatically.

Check your script for discrepancies. If you find any, you still have the option to skip the conversion to WinWrap and to open your project in a previous version of the product.

If you press **No**, the project will not be opened. You can open it with an older version.

### 3.1.5. Project with Validation Script Code to the New Validation Templates and Output Formatting

Basically, any validation code created in an older version of WebCenter Forms Recognition is compatible with version 11.1.1.8.0. However, version 11.1.1.8.0's new validation engine and its associated editor make it easier to set up these tasks by providing many new, robust ways to automate and simplify validation routines.

If you do plan to use code created in an older version, keep the following points in mind:

- If additional DLLs are used, test them after migration.
- Use the Always Valid setting on the Validation Editor to skip validation for any field.

- If you use the new validation methods for field-level or document-level validation, remove any redundant scripting that performed similar functions in the older version.
- If a field's standard validation is set to Invalid, corresponding script events will not be fired.

The new automated field formatting occurs after a Field Validate event is fired if a field is completely valid. Formatting results can be then accessed by scripting the Export event using a new Formatted Text property for the field object. The old Text property will still store unformatted text, which could be accessed as well.

For more details and a code example, see [BASIC CONCEPTS AND TECHNIQUES](#), Reference of Script Events, in the *Scripting Documentation*.

### 3.1.6. Migration to the Supervised Learning Workflow

The Supervised Learning Workflow can be applied to part of any WebCenter Forms Recognition project or added to an existing project. However, the meaning and configuration of existing classes might coincide – and therefore conflict – with prospective classes.

#### Considerations and Constraints

- The Supervised Learning Workflow automatically creates and learns new classes as sub-classes of a base document class. (A base document class is one that is not inherited from any other class.)
- These automatically created and trained classes are always placed directly under the base document class to which the document used for auto-training is currently classified.
- The Classification Field property that creates a document class cannot be empty.
- The Associative Search Engine must be assigned as the analysis engine for this field.
- The Associative Search Engine settings for the Classification Field must include definition of Document Class Format. This format must be defined in such a way that it ensures a unique name for every document class that could possibly be created. One way to ensure that the names are unique is to use the primary key (or any other unique field) from the database for the vendor pool as part of the class name.
- The Supervised Learning Workflow must be enabled in Designer and Verifier.
- For the new automatically learned classes, the Supervised Learning workflow uses the following configuration, which cannot be modified:
  - Brainware Layout Classification to automatically train Level 1 classification (usually called vendor-classification in the context of WebCenter Forms Recognition projects used for invoice processing.)
  - The Associative Search Engine to automatically detect a new class name, and, optionally, to automatically train Level 1 classification.
  - Brainware Extraction to train automatic extraction of header fields for fields that have Brainware Extraction and Format Analysis engines assigned for the parent base document class.
  - Brainware Table Extraction to train automatic extraction of header fields for fields that have Brainware Table Extraction engine assigned for the parent, base document class.



- Classification for base document classes must be configured manually. Manual configuration can consist of automatic redirection to a base class, or through any available classification engine.

### Setting up the Migration

To migrate an existing project to the Supervised Learning Workflow, do the following preparatory steps:

- Make sure RTS is configured and running.
- Load the project into Designer.
- Establish logins and passwords.
- Save project and reopen it.
- Login to the project.
- Make sure all DLLs called from script are available.
- Switch to Definition Mode. Classify and analyze the current document.
- Switch to Verifier Test Mode and test project-related validation.

### A Scenario

For example, consider the following scenario:

An existing vendor class, Oracle Invoice that process invoices from supplier Oracle Corporation

A document class with the same purpose, but with a different name, (for example, BW1234567) could be created via the Supervised Learning Workflow, which may cause conflicts.

To avoid conflicts, it is necessary to take into account the following cases in order to combine non-SLW and SLW processed class nodes:

- Case A: The existing set of classes does not coincide with the set of classes that the Supervised Learning Workflow might create.
- Case B: The existing set of classes does coincide with the set of classes that the Supervised Learning Workflow might create. Automatic learning for these classes is not required.
- Case C: The existing set of classes does coincide with the set of classes that the Supervised Learning Workflow might create. Automatic learning for these classes is required. In addition, these classes are Level 1 classes, meaning that they are derived from a base document class.
- Case D: The existing classes do coincide with the set of classes that might be created by the Supervised Learning Workflow. Automatic learning for these classes is required. These classes are not on Level 1 in WebCenter Forms Recognition project classes' hierarchy or their base classes do not satisfy constraints for Supervised Learning – meaning that Classification Field settings are not correctly configured.

Criteria	Case A	Case B	Case C	Case D
----------	--------	--------	--------	--------

Criteria	Case A	Case B	Case C	Case D
Existing classes conflict with potential classes	No	Yes	Yes	Yes
Automatic Learning required	NA	No	Yes	Yes
Level 1 class (Derived from Base DocClass)	NA	No	Yes	No
Base class do not satisfy requirements for SLW execution or classification field improperly configured	NA	NA	NA	Yes

Table 3-1: Criteria for potential class conflicts

**Note:** Classes created by the Supervised Learning Workflow must not have the same names as existing document classes. To ensure that this never happens, you must set up the Classname format in the Associative Search Engine settings in a unique way, so that existing class names and new class names never overlap. For more details about Supervised Learning Workflow constraints, see Section [PLANNING](#).

**Case A**

This case is the easiest to handle. All you have to do is create new base classes or use existing base classes required for Supervised Learning processing, and then run the Supervised Learning Workflow in conjunction with existing classification / extraction project configurations without any additional adjustments.

**Case B**

Case B occurs when you want to keep an existing document class with its Learnset, scripting, etc., as is, and all documents should still be classified to this class. It’s conceivable that Supervised Learning could create a new document class by automatic learning for this document. In that case, you would have to switch off automatic learning for this document class.

To disable “automatic class creation” for a particular class, set the vendor type field in “\*.csv” file to 2 for the corresponding class search entry. For details, see section [CONFIGURING ASSOCIATIVE SEARCH ANALYSIS](#).

If the Vendor\_Type column is not part of the vendor pool, you must create or designate a new column in the vendor pool \*.csv file to be used for Associative Search Engine search database import. This field can have one of the following three values: 0, 1, or 2. By default, when the field is not available in the vendor pool, the system acts as if the Vendor\_Type value is 0. A Vendor\_Type value of 0 means that all documents with more than N% invalid fields are added to the Learnset. For more details, refer to the WebCenter Forms Recognition **Verifier User’s Guide**.

You must re-launch the pool import in the Associative Search settings to be able to establish Classification Field column as the Vendor\_Type field. Do not use this field as a search field.

Returning to our scenario, the illustration at right shows how a migrated project might look. This illustration shows a sample hierarchy consisting of the old document class Companies with its derived document classes and a new base class conceived for



## SLW, Invoices.

In our scenario, there is an entry Oracle in the vendor pool, but learning is prohibited for this entry. Although no new document class for Oracle will be created, you still must ensure that the system classifies the documents to the “old” document class, Oracle Invoices, and not to Invoices. For the example, those document classes are situated within different tree branches. To ensure correct classification, the following project configurations are possible. There are other possible configurations, depending on the specific project being migrated.

- Move part of the document class tree to the new Invoices class. In the illustration at right, automatic learning for Oracle Invoices is disabled and all the configurations (such as for Learnset) are retained. Because this part of the class hierarchy does not have to be trained, the derived document class does not have to at Level 1.
- Do not change the document class hierarchy. For the document class Invoices do not select a classification, but do select one for the Companies branch. Additionally, set a default classification to Invoices. This means that if a document cannot be classified to the document class for which classification engines are selected, it will be classified to Invoices.
- Use WinWrap Basic scripting to ensure that the document is classified to the correct document class



## Case C

If Supervised Learning must be enabled for an existing class that is a derived document class at level 1, and its base class satisfies constraints for Supervised Learning execution, Vendor\_Type in the \*.csv file must be 0 for the corresponding class entry in this file.

In addition, the existing class name and the class name automatically generated by SLW and specified on the ASE settings for Classname Format must coincide with each other. To do this, either:

- Rename the existing class so that its name coincides with the name that would be generated via Classname Format for the corresponding entry in \*.csv file.
- Change Classname Format so that the names are the same. This can only be applied if it can be done for all classes that “Case C” covers. Also, uniqueness of all new document class names must be guaranteed.
- Create a new (non-search) field in the \*.csv file and give this field a descriptive name. (An example might be My Document Class Name.) In the \*.csv file generation procedure for all classes described by Case C, use their class names to fill out the field. For all other classes, create a name according to field naming conventions (For example, vendor name, space, unique vendor ID.) Then, in the settings for the Associative Search Engine, use [My Document Class Name] as Classname Format.

The base document classes must satisfy the constraints for execution of the Supervised Learning Workflow: The Classification Field settings are configured correctly, the generic document class is at level 0 of the classification hierarchy and the Associative Search Engine is configured. For details, see Section [PLANNING](#).

## Case D

If the Supervised Learning Workflow must be enabled for a particular existing class that is not a Level 1 class or its base class does not satisfy constraints for SLW execution, the vendor type field must be 0 for the corresponding class entry in this file.

In this case, follow the instructions covered for Case C.

In addition, the Case D class must be moved under another base class to ensure that the document class is now at Level 1. The easiest way to do this in Designer is to switch to Definition mode, Classes view, and use drag & drop to move it to the required class. You can also implement a scripted procedure, use Project.MoveDocClass, and save the project.

**FIGURE 3-4** illustrates the steps that the system administrator must consider for each document class when migrating a project.

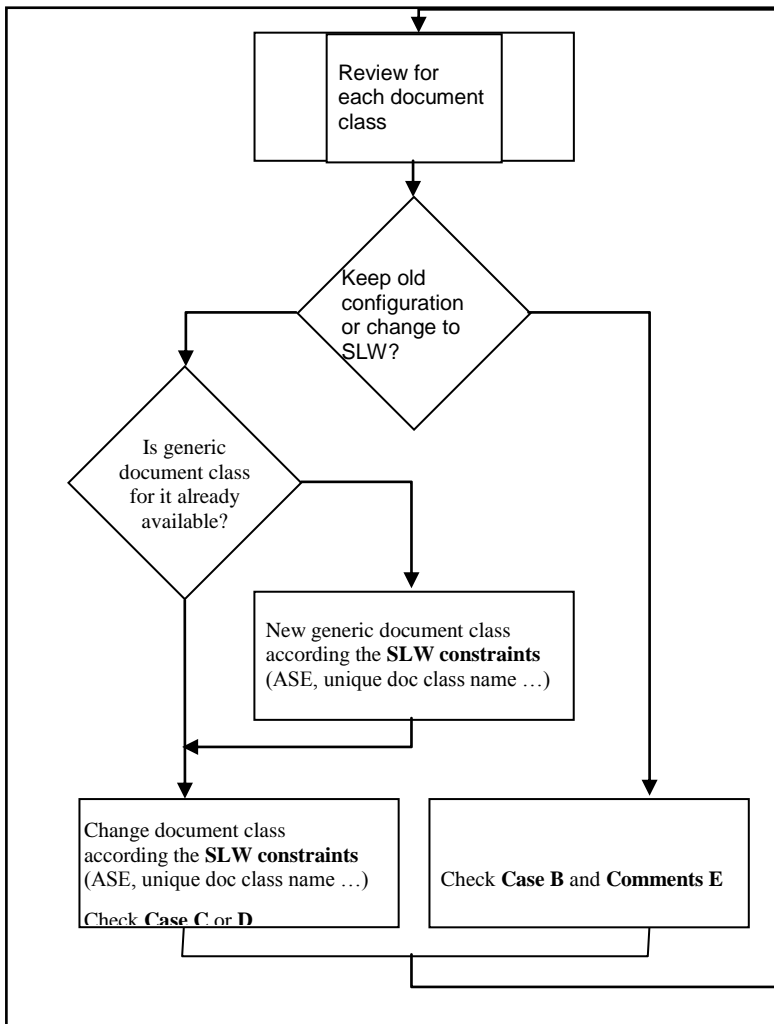


Figure 3-4: Document class migration to combine non-SLW and SLW processed class nodes

## 4 Basic Concepts and Techniques

### 4.1 Starting Designer

There are two ways to start WebCenter Forms Recognition Designer:

#### To start the program with a new project:

- If the program was installed normally, you can start it from the Windows Start menu by using the command sequence Start > Programs > Oracle > WebCenter Forms Recognition > WebCenter Forms Recognition Designer. This starts the program with a new project. You can create a new project (Section [CREATING PROJECTS](#)) or load an existing project (Section [OPENING PROJECTS](#)).
- You can also start the Designer application by clicking the desktop icon.

*You can run multiple instances of Designer simultaneously.*

For more information on projects, please refer to section [WORKING WITH PROJECTS](#).

### 4.2 About Roles, Users, and Authentication

In WebCenter Forms Recognition, users are assigned to groups, and groups are assigned to roles.

Administrators and users must login to Designer and Verifier to load existing projects. The system authenticates the user name/password combination; it either denies authorization to the project, or allows user to perform the functions which is set by their permissions for the project (assuming the functionality is enabled.)

Users can be assigned to one or more groups, and established groups can have one or more roles. A user must be a member of at least one group, and a group must have at least one role assigned to it.

There are six roles: Administrator, Verifier, Verifier Settings, Learnset Manager, Supervised Learning Verifier, and Verifier Filtering.

The roles of these groups follow:

- **Administrator:** The Administrator (ADM) role is to manage users, groups, and user-to-group assignments. Administrators install the system, configure applications, and manage data. They also design and maintain projects. This role is the most powerful of the four roles, because it encompasses the permissions for all other groups.
- **Supervised Learning Manager:** The Supervised Learning Manager (SLM) role is to define, modify, and maintain the Learnset. This functionality is accessible only through Verifier.
- **Supervised Learning Verifier:** The Supervised Learning Verifier (SLV) role is to collect and manage local training data. Supervised Learning Verifiers are subject-matter experts who can propose Learnset candidates to improve system performance. This functionality is accessible only through Verifier.
- **Verifier:** The role of the Verifier group (VER) is to verify documents that could not be automatically processed. Typically, members of the Verifier group are clerks. This functionality is accessible only through Verifier.

- **Verifier Settings:** The Verifier Settings (SET) role is to change the WebCenter Forms Recognition Verifier configuration. This role is given to users who are considered to have enough knowledge of the application to make changes that will be beneficial to all Verifier users.
- **Verifier Filtering:** The Filtering (FLT) role allows for more flexibility and security when working with the Verifier client. The filtering tab in Verifier client is unmapped from the settings dialog window, so that FLT role users are able to filter batches even if they do not have the SET role.

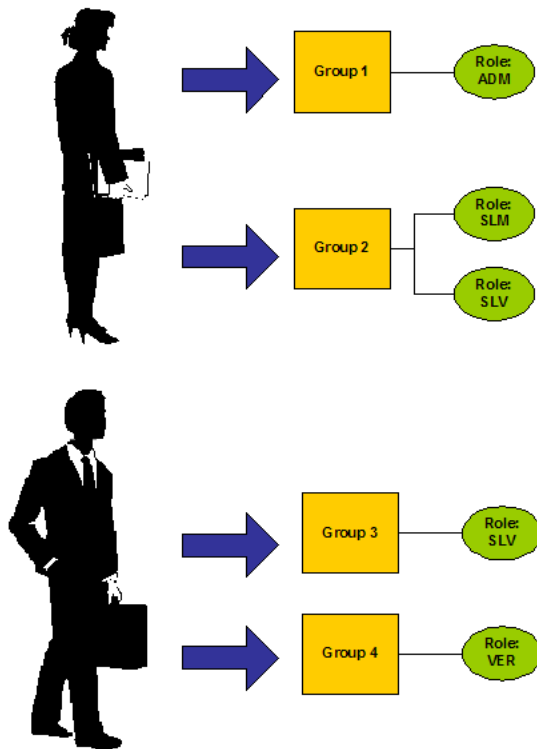


Figure 4-1: Designer Users with group and role assignments

Two criteria determine what users can do in Verifier, both of which are established in Designer:

- Which groups the users are assigned to.
- The functionality enabled for the project.

Users inherit the roles of the groups they are members of. These roles dictate the components that can be used by members of each group. However, even if a user is authorized to perform certain tasks, the functionality is inaccessible unless it has been enabled in the project settings. For example, members of the Learnset Manager group will not be able to use Verifier Train Mode unless Verifier Train Mode is enabled in Settings. (See section [VERIFIER TRAIN MODE](#))

Verifier can use a WebCenter Forms Recognition database focused user management. (Refer to [EXPORTING OF USERS, ROLES AND GROUPS TO THE WEBCENTER FORMS RECOGNITION Database](#))

## 4.3 Logging In

### 4.3.1. Logging In to Designer

To use Designer, you must be assigned to a group that has the Admin role. Users without this authorization will not be able to use Designer.

As an administrator, you must log in to any Designer project you load.

You do not have to log in when you create a project, but you are essentially logged in as an administrator when you save the new project.

That is, when you save a new project, you are logged in as Admin. No password is required the first time you log in to the new project. Your first action should be to change the Admin password. Not immediately changing the password leaves a security hole open.

### 4.3.2. Logging In to Verifier

Also, a user is asked to supply a user ID and password when loading a project in Verifier. Users who forget their passwords must contact their project administrator to have their password reset.

## 4.4 Creating Users and Groups

### 4.4.1. User Interface

User accounts and group assignments are done from *Options* menu.

The user interface consists of three tabs: *Users*, *Groups*, and *Change Password*.

### 4.4.2. Establishing and Changing Passwords

Passwords are created and modified from the *Options* menu by selecting *Users*, *Groups* and *Roles* and then choosing the *Change Password* tab. For details see section [4.4.6.2 CHANGING A USER'S PASSWORD](#).

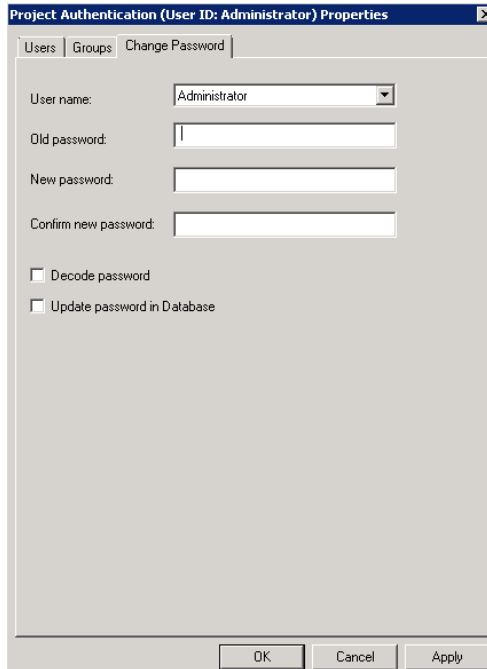


Figure 4-2: The Change Password tab is used to create or reset passwords

#### 4.4.3. Creating the Administrator Password

As an administrator, your first task is to change your password for each existing project. Otherwise, anyone will be able to login to your project and make changes that could potentially affect the entire workflow. To change your password:

Load an existing project. Use Administrator (not case-sensitive) for user name, and leave the password blank.

- On the *Options* menu, click *Users, Groups and Roles*.
- On the *Project Authentication properties* box, click *Change Password*. Create a new password, click *Apply* and then *OK*.

#### 4.4.4. Establishing User Groups

User Groups are created and modified from the *Options* menu by selecting *Users, Groups and Roles* and then choosing the *Groups* tab.



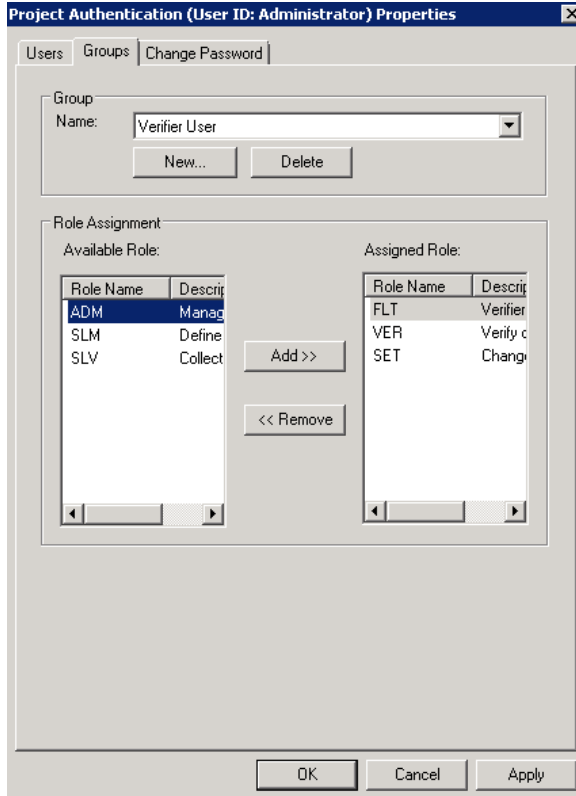


Figure 4-3: On the Groups tab, you can create or delete groups, and assign or remove group roles.

### 4.4.5. Creating Groups

By default, the first time the administrator logs in, the only existing user group is Administrator (ADM), and the only member of the ADM group is the Administrator user.

**To create new groups:**

- 1) Click *Groups*.
- 2) Click *New...*
- 3) Type a name for the new group. The *Groups* field is restricted to 20 alphanumeric characters.
- 4) For *Description*, select one of the four group roles (Administrator, Verifier, Supervised-Learning Manager, or Supervised-Learning Verifier.)

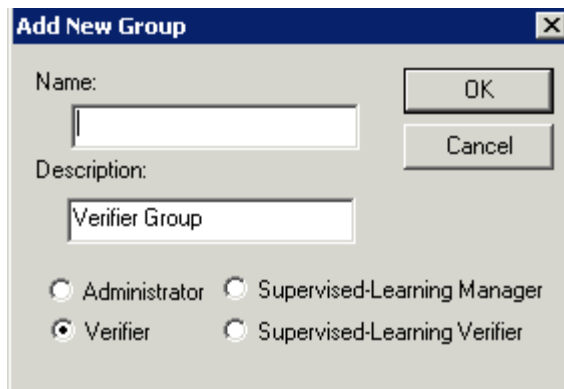


Figure 4-4: Creating a new group

5) Click *Apply* and then *OK*.

#### 4.4.5.1. Assigning Roles to Groups

When you first create a group, you can only associate one of the four profiles to it. However, you can assign or remove roles from an already-established group.

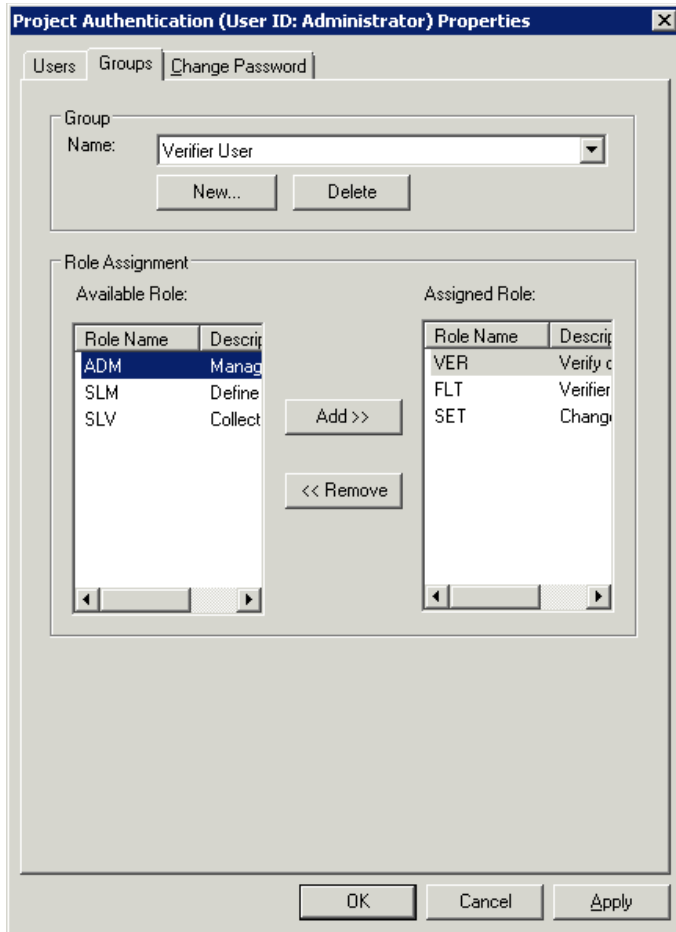


Figure 4-5: Groups Tab – Adding and Removing Roles

##### To add a role to a group:

- 6) Select the role in the *Available Role* and click *Add>>*.
- 7) Click *Apply* and then *OK*.

##### To remove a role:

- 1) Select the role in the *Assigned Role* and click *<<Remove*.
- 2) Click *Apply* and then *OK*.

**Note:** You cannot delete the admin user or the ADM role. You can delete additional users or groups.

After you've established user groups, you are ready to assign users to them.

### 4.4.6. Creating and Managing User Accounts

User accounts are created and modified from the *Options* menu by selecting *Users, Groups and Roles* and then choosing the *Users* tab.

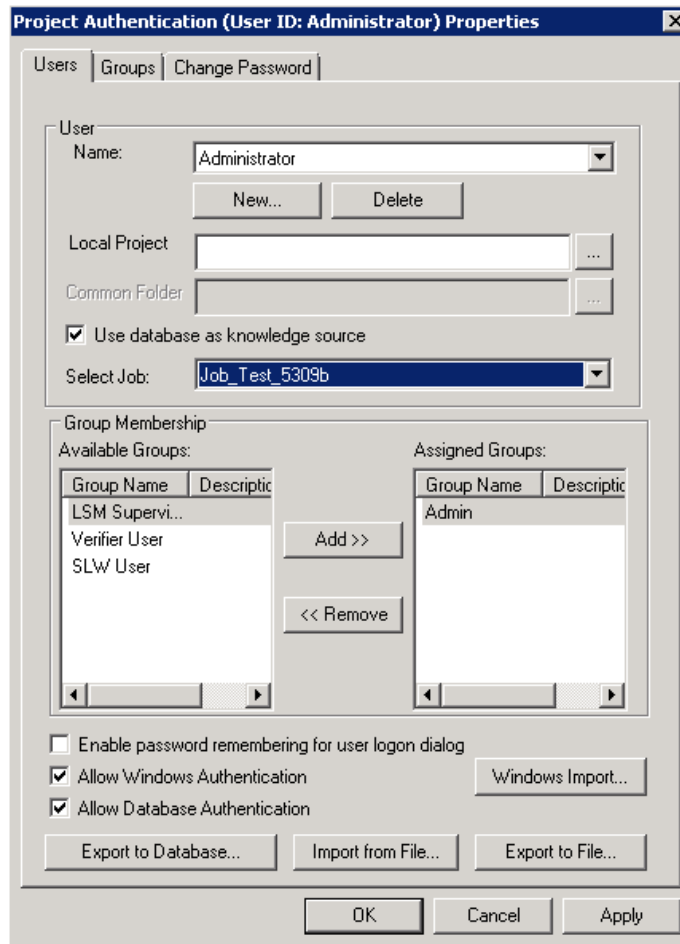


Figure 4-6: Users Tab

On the *User's* tab, you can create or delete individual users, assign or remove users from groups, and import multiple users from an external file.

You can store Common Database Learnset against user settings.

*Note:* From WebCenter Forms Recognition version 11.1.1.8.0 on, all projects should be enabled with 'Allow Database Authentication', and the users should exist in the database in order to successfully login to the system.

#### 4.4.6.1. Creating New User Accounts

**To create new user accounts:**

- 1) On the *User* tab, click *New*.
- 2) Type a name for the user and assign a default password. User names cannot exceed 17 and passwords cannot exceed 20 alphanumeric characters.
- 3) Click *OK*.

After you've created a user account, you must assign it to at least one group. Users who are not assigned to a group will not be able to log in to the project.

To give the user the option of remembering their user name and password between WebCenter Forms Recognition Verifier logons, select *Enable password remembering for user logon dialog*.

#### 4.4.6.2. Changing a User's Password

**To change a user's password:**

- 1) On the *Change Password* tab, check the *Decode Password* checkbox.
- 2) Type the new password in the *New Password* and *Verify Password* text boxes.
- 3) Click *OK*.

*Note: This only works for users that were created within the current project file. Users created via Security Update are invisible for the WebCenter Forms Recognition Designer.*

#### 4.4.6.3. Assigning Users to Groups

**To assign a user to a group:**

- 1) Select a user name from the pull-down list.
- 2) In the *Available Groups* selection box, select a group and click *Add*.
- 3) To remove groups, select a group in the *Assigned Groups* selection box and click *Remove*.

#### 4.4.6.4. Importing and Exporting Users and Groups

The definitions of all users' roles for a project can be exported in bulk to a binary file and then imported in bulk into another project. This is done within the *Import* and *Export* buttons on the *Users* tab of the *Project Authentication* dialog box.

**To export a list of users:**

- 3) On the *Users* Tab, select the group to export and click *Export to File...*
- 4) Name the file and then click *OK*. By default the file is saved as an authentication file (\*.sec) on the project root, you can change the target folder if necessary.

**To import a list of users from another project:**

- 5) On the *Users* tab, click *Import from File...* Select the appropriate file to import. Click *OK*.

*You should not manually edit authentication files.*

**To import user via Security Update:**

- 6) Enable "Allow Database authentication".
- 7) Add the appropriate script snippet. Please refer to the ***Scripting Reference Guide***, section [1.2.1.8 UPDATESYSTEMSECURITY](#).
- 8) Configure RTS to execute "Update System Security" with this project.
- 9) Run RTS and wait for the update event.

### 4.4.7. Exporting of Users, Roles and Groups to the WebCenter Forms Recognition Database

In Verifier, you have the possibility to make the users, roles and groups available in the WebCenter Forms Recognition database. This is necessary if a database platform is used by WFR applications, like Web Verifier.

This can be easily done via the button *Export to Database...* in the WebCenter Forms Recognition project authentication dialog which is available under the *Options* menu – *Users, Groups and Roles*.

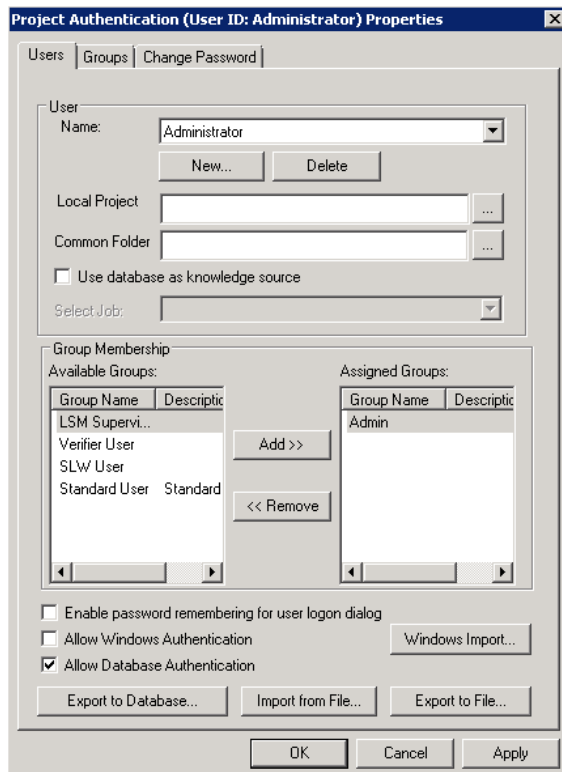


Figure 4-7: Settings for Project Authentication dialog

*Note: Export will only affect users and groups in the project being exported; any other existing DB users or groups will not be affected.*

## 4.5 Windows Based User Authentication

### 4.5.1. Enabling Windows Authentication

To enable Windows based authentication in WebCenter Forms Recognition for a desired project, open the corresponding project file in Designer, select *Options, Users, Groups and Roles...* menu item and enable *Allow Windows Authentication* checkbox on *Users* tab of the *Project Authentication Settings* dialog:

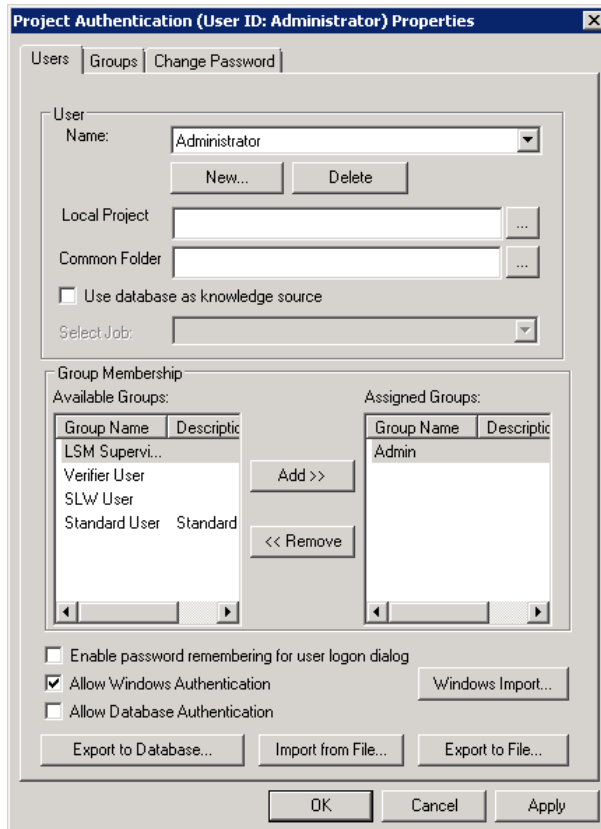


Figure 4-8: Activate Windows Authentication

Windows Authentication and Database Authentication may be enabled at the same time for different users.

#### 4.5.2. Importing Windows accounts

There are two methods to import the users into the database:

- The GUI based *Windows Import* button.
- The Script based *SecurityUpdate* method.

To import required Windows users via the graphical interface, click *Windows Import...*:

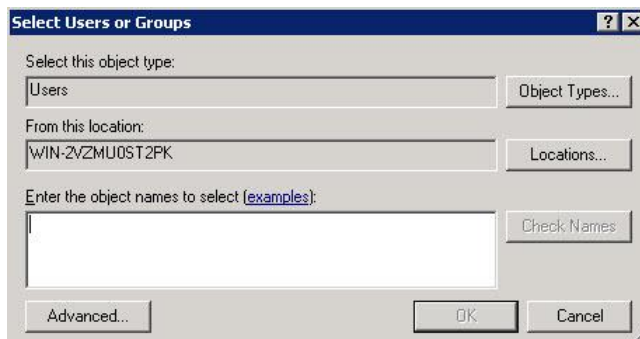


Figure 4-9: Activate Windows Authentication dialog box

Click *Locations...* to select required Windows domain to import the user accounts from:

Then click *Advanced...* to expand the extended search settings and click *Find Now* to locate the users in the selected domain. As soon as the users appear in the list view, you can either choose the desired ones (press and hold the “Ctrl” key to apply multiple selection) or import all available users at once (select the first item in the list and then keep clicking *Page Down* until all items in the list have been selected):

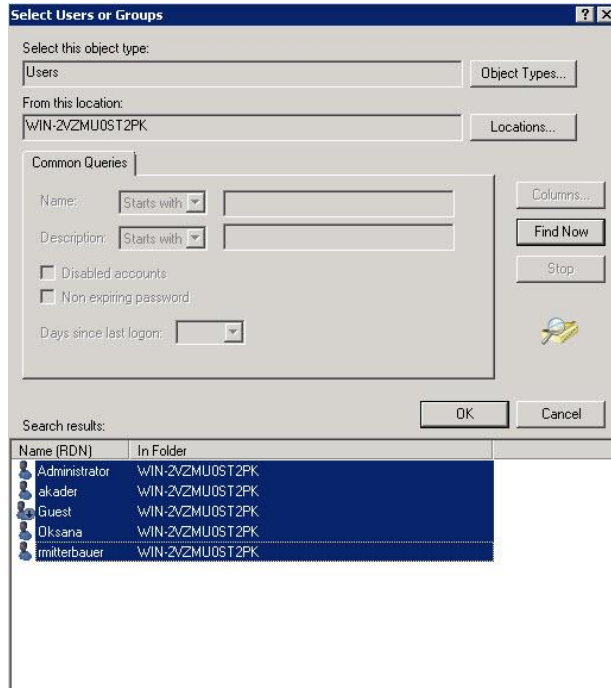


Figure 4-10: Select Users, Computers, or Groups dialog box

Click *OK* to confirm the selection. Now review the selected users:

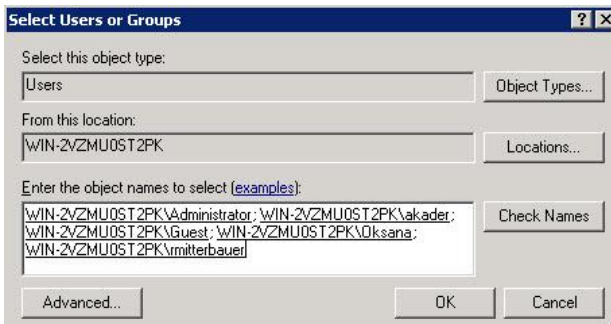


Figure 4-11: View of the selected users

Click *OK* to finish importing the Windows users into the WebCenter Forms Recognition project.

To import users with script use the SecurityUpdate method as described in section [4.4.6.4 IMPORTING AND EXPORTING USERS AND GROUPS](#).

*Note:* All newly imported users are, by default, assigned to a special "Standard User" group. This group, by default, has "SLV", "VER" and "FLT" WebCenter Forms Recognition rights, i.e., rights to run Verifier in Advanced mode. The standard user rights assignments for this group can be adjusted at any time according to the specific project requirements.

### 4.5.3. Windows Authentication Settings

Once the project is saved, the Windows based authentication has been activated, and the next time this project file is opened in Verifier or Designer (or in any other WebCenter Forms Recognition application that requires project level authentication), WebCenter Forms Recognition's authentication sub-system will always try to login automatically using the currently logged in Windows user.

In order to activate the automatic logon feature, both checkboxes "Allow Database Authentication" and "Allow Windows Authentication" have to be activated in Designer's Authentication Settings for the WebCenter Forms Recognition project used in Verifier clients.

*Note: In order to activate the option 'Update passwords in Database' in the Change Password tab of the Verifier, 'Allow Database Authentication' checkbox has to be selected.*

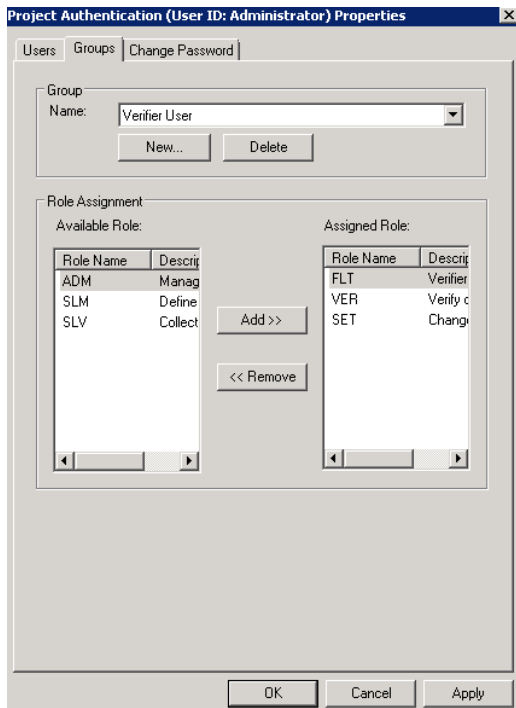


Figure 4-12: Groups tab with assigned Roles

For example, if the Verifier application is run on a workstation where "ADM" user is logged in, the Verifier Logon is going to proceed automatically.

Since by default the "Standard User" group does not include administrative WebCenter Forms Recognition rights or the imported users, trying to login with the same sample user in Designer application is going to lead to a warning message like the one below, because Designer requires administrative privileges:

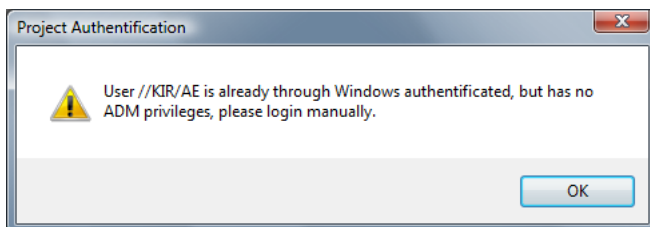


Figure 4-13: Warning message for logons without administrative privileges



The warning message should be followed by a normal log on screen, so that the user can log on using, for example, one of the available internal WebCenter Forms Recognition project's accounts:

Figure 4-14: Logon dialog box

**Note:** By default newly imported users are imported with empty passwords and therefore the system does not allow them to login as "another" Windows user with the normal Log on dialog box (see the screenshot above) unless the password for the Windows user has been explicitly set in WebCenter Forms Recognition.

Individual user's settings can also be modified, for example, in order to assign administrative WebCenter Forms Recognition privileges to some specific users, e.g., to the same user "AE" located in the domain "KIR", like it is shown on the screenshot below:

Figure 4-15: Modification of individual user's settings

Now the user can log on to both the Designer and Verifier applications automatically. The Verifier application is going to display the user as an administrator.

**Note:** In case the currently logged in user account is not a member of the pre-imported Windows users list, WebCenter Forms Recognition applications are supposed to show the normal Log on dialog where the user has to enter a user name and password. In such a case, all previously available Log on features, like remembering the password, still remain applicable.

#### 4.5.4. Windows Authentication for Web Verifier

To use Windows Authentication for the Web Verifier Login:

- Import the Windows accounts of the designated Web Verifier user to the project database via the Windows Import button or the Security Update method as described above.
- Enable the Web server IIS Windows Authentication.

*Note: Only Domain users will be able to log in to the Web Verifier.*

### 4.5.5. Usage

The Windows based user authentication feature can be used for:

- Simplification of the user log on procedure for WebCenter Forms Recognition Verifier and - modified - Web Verifier users – Users log in just once in Windows and then log in to the Verifier application with the same log on information automatically and for Verifier without any visual interaction.
- Synchronization of Windows authentication and internal WebCenter Forms Recognition authentication processes – This makes initial setup of WebCenter Forms Recognition users easier and faster to apply (all required Windows users can be imported into the WebCenter Forms Recognition subsystem automatically at once) and simplifies administration of user accounts in the overall customer environment.

## 4.6 Configuring Licensing Properties

### 4.6.1. Description

Selecting *Options*→*License...* from the menu in WebCenter Forms Recognition Designer application:

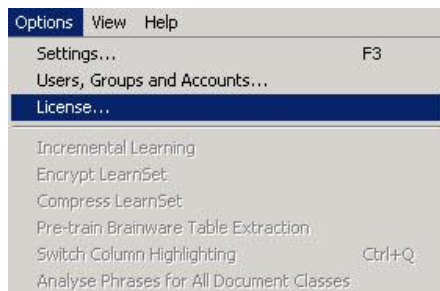


Figure 4-16: Options menu

The application shows the licensing dialog allowing the Designer user to adjust the primary shared licensing location:

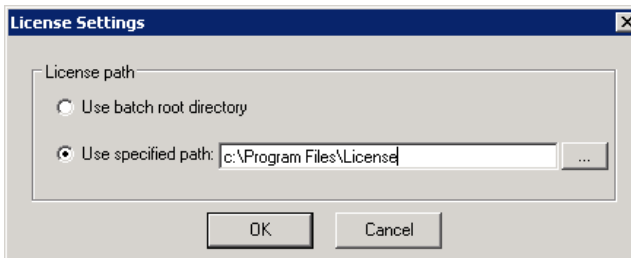


Figure 4-17: Licensing dialog

Note: When using the Database, batch root directory licensing cannot be used.

## 4.7 Exiting Designer

To exit WebCenter Forms Recognition Designer:

- 1) On the *File* menu, click *Exit*.
- 2) On the Confirmation message box, click OK.

## 4.8 Working with Modes

### 4.8.1. What Are Modes?

WebCenter Forms Recognition Designer supports seven modes of operation:

- Document Selection (Section [DOCUMENT SELECTION MODE](#))
- Definition (Section [DEFINITION MODE](#))
- Train (Section [TRAIN MODE](#))
- Runtime (Section [RUNTIME MODE](#))
- Verifier Design (Section [VERIFIER DESIGN MODE](#))
- Verifier Test (Section [VERIFIER TEST MODE](#))
- Verifier Train (Section [VERIFIER TRAIN MODE](#))

Each mode supports a particular set of tasks used during the design of a WebCenter Forms Recognition application. When you switch from one mode to another, the toolbar and the window below it also change to provide the tools and controls used in the new mode.

### 4.8.2. Document Selection Mode

#### 4.8.2.1. Purpose

Use this mode to select a document set that will be used in learning and testing.

#### 4.8.2.2. Selection

To switch to Document Selection mode:



- On the View menu, click *Document Selection Mode*.
- Or
- On the toolbar, click *Switch to Document Input Selection*. Depending on the current input type (Directory, Learnset, or batch) the appearance of this button varies.

If the paths for Batch and Image Roots are not set, these icons are disabled.

- If you change the input type, Document Selection Mode is activated automatically. Please see the Settings Input Mode tab (batch and image path) and the Train Mode tab (Learnset path.)

### 4.8.2.3. Settings


#### The current document set can be:

- A directory of files: Use directories as input type only if you already have the corresponding Workdocs. You can also use files from directories to optimize your Learnset.  
If Workdocs are not available, WebCenter Forms Recognition will create new Workdocs for the .tif files. This can be time consuming, because OCR has to take place.
- A batch: WFR Runtime Server creates batches and works on them. Use batches as the primary input type for classification design. Processing results can only be saved persistently as Workdocs with batches. Use reference batches with documents from given classes to verify the quality of the classification and extraction.
- A part of the Learnset: Once you have defined a classification scheme, you need a Learnset to train the classification and extraction. Use the Learnset as document input to review your sample documents.

#### To specify the location of the Learnset:

- 1) On the *Options* menu, select *Settings*.
- 2) Select the *Train Mode* tab.
- 3) For Learnset Manager, type or browse to a directory. To create a new directory, click *Browse* , and then click *New directory*. Type in the path name.

#### To select an input type:

- 1) On the *Options* menu, select *Settings*. Or, in the toolbar, click  *Show settings*.
- 2) On the *Settings* dialog box, select the *Input Mode* tab.
- 3) Specify the file format under *Import Options*.

You can select *Image & Electronic Documents* and enter the extension of your image file format (\*.tif, \*.jpg, \*.bmp, \*.gif or similar).

You can also use Workdocs without image files. In this case, select *Workdoc Files* (\*.wdc).

**Note:** You cannot process non-image file types such as Word documents this way. Use batches to process non-image file types. Please refer to the *WebCenter Forms Recognition Runtime Server User's Guide* for details.

**Use database as documents and statistics source:** WebCenter Forms Recognition core information can be stored in the database. If you have selected the Database as your data source, you can select the desired job from the *Select Job* dropdown list.

Furthermore, you are able to create a new job by pressing the *Create Job* button.

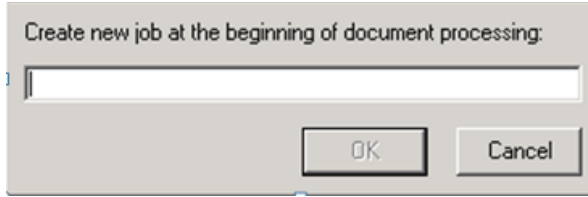



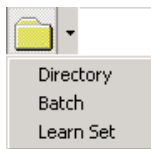
Figure 4-18: “Create Job” dialog window

**To specify the location of batches:**

- 1) On the *Options* menu, select *Settings*. Alternatively, in the toolbar, click  *Show settings*.
- 2) In the *Settings* dialog box, select the *Input Mode* tab.
- 3) Specify the location of your batch files under *Batch-Root Directory*. For each batch file, there is a subdirectory containing the image files that will be batch-processed. Specify the root of these subdirectories under *Image-Root Directory*. In general, both root directories should be the same.

*Note:* Batch location must be defined before you can select batches as input type.

**To select the input type:**



- 1) Click on the drop-down arrow associated with *Switch to Document Input Selection*.
- 2) From the list, select the input type. Depending on your selection, the appearance of the button varies.




Button	Description
	Indicates document input from directories.
	Indicates document input from the Learnset.
	Indicates document input from batches.

Table 4-1: Controls of Document Selection Mode

- 3) The Document Selection Mode is activated automatically.

**4.8.2.4. User Interface**

The user interface of Document Selection Mode enables you to further refine your document selection.

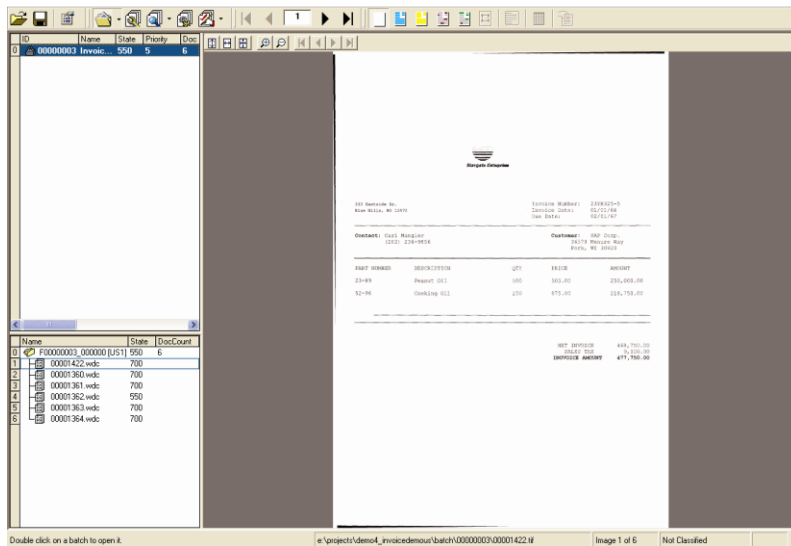


Figure 4-19: Refining settings through the Document Selection Mode

## 1. Document Set Selection

Depending on your settings (See section [SETTINGS](#)), this panel shows:

- A tree view of directories from the file system
- A list view of batches from the batch root directory
- A list view of classes a Learnset has been created for.

If you double click one of the entries, the panel below shows the contents of the current selection.

## 2. Document Selection

This panel shows the documents from your document set. This can be

- A list view of files from the selected directory
- A tree view of folders and document from the selected batch
- A list view of files from the selected class.

If you click one of the entries, the selected document is displayed in the panel on the right side.

## 3. Document Display

This panel displays the currently selected document.

### 4.8.2.5. Input Type Directory

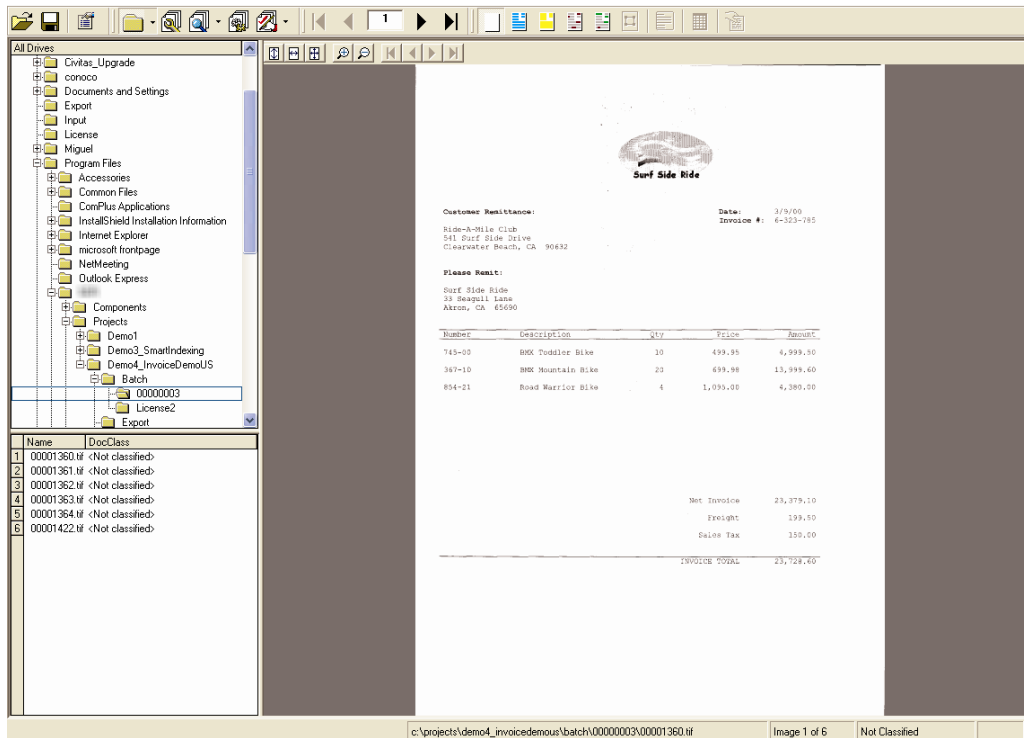


Figure 4-20: Importing files from a directory

#### 1. Directory List

“Tree view” directory from the file system. If you double click one of the entries, the panel below shows the content of the current selection.

#### 2. Workdoc / Other Files List

A “list view” of files from the selected directory. If you click one of the entries, the selected document is displayed in the panel on the right side.

#### 3. Document Display

This panel displays the currently selected document.

#### For drives or directories, you can:

- Open a directory
- Create batches for this project from the directory
- Refresh the selected directory

To select a directory option, select the directory from the drive/directory list and then click *Edit*.

To create batches from a directory, select a directory from the directory list and select *Edit/Import New Batch from Directory*. WebCenter Forms Recognition inserts the selected batch in the Batch directory. Please see *Project Settings – Input Mode Tab*. A progress bar is displayed while the insertion is in progress.

### 4.8.2.6. Input Mode Batch

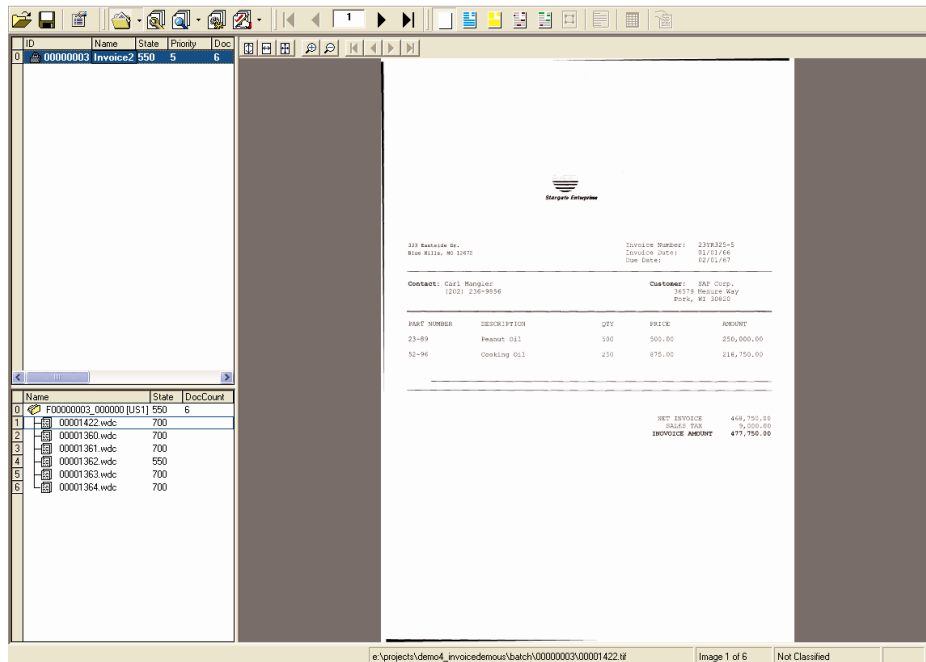


Figure 4-21: Importing files from a batch

#### 1. Batch List.

It contains batches in list view from the batch root directory. If you double click one of the entries, the panel below shows the content of the current selection.

#### 2. Document/Folder List.

It contains folders and documents in tree view from the selected batch. If you click on one of the entries, the selected document will be displayed in the panel on the right side.

#### 3. Document Display.

This panel displays the currently selected document.

You have several options for working with batches and documents in Input Mode.

#### For batches, you can:

- Close a batch.
- Create a batch.
- Rename a batch.
- Change a batch state - Select from a drop-down box to change the state of a batch.
- Change a batch priority - Select from the drop-down box to change the status of a batch in a batch group.
- Add a folder - Enter in a name to add a new folder.
- Delete a batch - Deletes the selected batch.

#### For folders, you can:



- Rename a folder. Enter a new name to rename the folder.
- Delete a folder and documents from the list.
- Add document. This adds a Workdoc to a folder.

**For documents, you can:**

- Split a multipage document into separate Workdocs (for a multipage document).
- Append to proceeding. Appends a Workdoc to the preceding Workdoc to make a multipage document.
- Change state. Changes the status of a document to a new value that you assign.
- Delete. Deletes the selected Workdoc from the batch.

To select a batch option, select the batch from the batch list and select *Edit* → *Batch* from the menu or right click the desired batch and select the desired option.

To select a folder option, select the folder from the Document/Folder list and select *Edit* → *Folder* from the menu or right click on the desired folder and select the desired option.

To select a document option, select the Workdoc from the document/folder list and select *Edit* → *Folder* or right click on the desired document.

You can also drag and drop a folder to another batch or move a selected document to another folder within the same batch.

**4.8.2.7. Input Type Learnset**

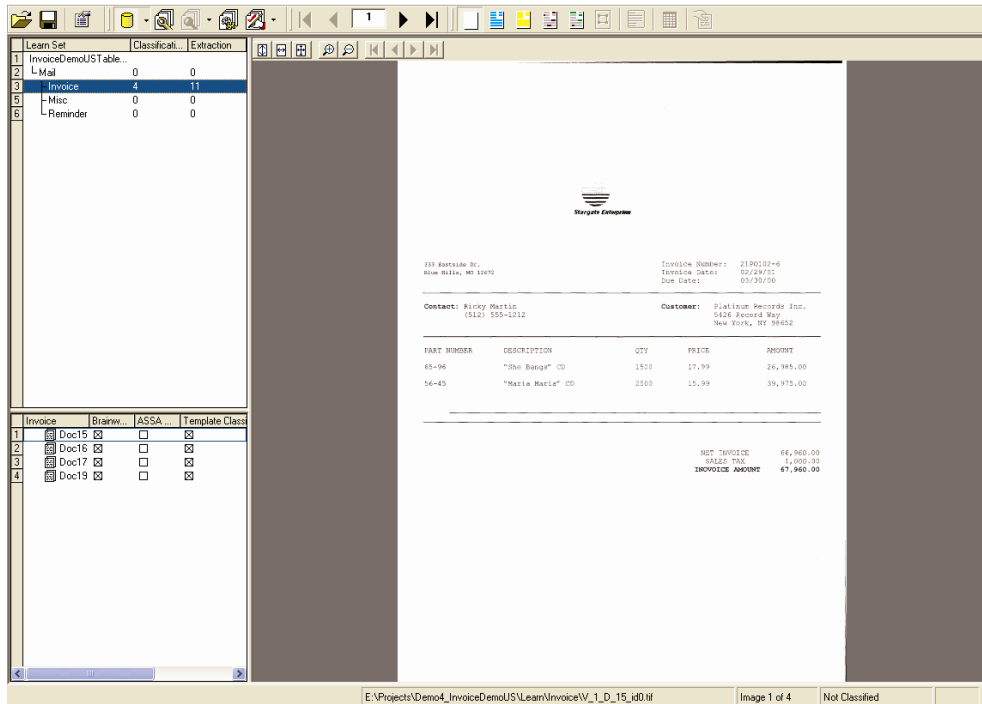


Figure 4-22: Inputting from a Learnset.

**1. Document Class View.**

It contains classes in list view that Learnset has been created for. If you double click one of the entries, the panel below shows the content of the current selection.

## 2. Classification Extraction/Learnset (learned documents.)

It contains files in list view from the selected set.

## 3. Document Display.

This panel displays the currently selected document.

### For document classes, you can:

- View Classification Learnset. A list of documents and their classification Learnset are shown.
- View Extraction Learnset. A list of documents and their extraction Learnset are shown.

### For Learnsets, you can:

- Remove from Learnset. Delete the selected document from the Learnset. Documents in the class will have to be learned again.
- Click the checkboxes for the classification/extraction engines. If a checkbox is selected, the document will be added to the Learnset of the document class for this engine. If the checkbox is cleared, the document will be removed from the engine for this Learnset. Documents in the class will have to be learned again.

### To select a document class option:

- 1) Select a document class from the Document Class View list.
- 2) Select *Edit* → *Learnset DocClass*.

### To select a Learnset option:

- 1) Select a document from the Classification/Extraction Learnset.
- 2) Select *Edit* → *Learnset Document*.

## 4.8.2.8. Creating Test Documents

After you establish your document settings, you can create test documents that will automatically select files from a chosen directory and place them in your batch directory.

### To import a new batch:

- 1) Click the arrow next to *Switch to Document Input Selection*.
- 2) From the list, select the *Directory* as input type.
- 3) Select a directory from the directory list for the origin of your files.
- 4) Select *Edit, Import New Batch from Directory*. WebCenter Forms Recognition inserts files into the batch directory. A progress bar appears during the insertion.

## 4.8.2.9. Automatic Conversion of Documents during Import Phase

### Description

On the Processing panel of project settings in the Designer application, it is possible to configure a couple of options that allows automatic conversion of documents during import into the WebCenter Forms Recognition system:

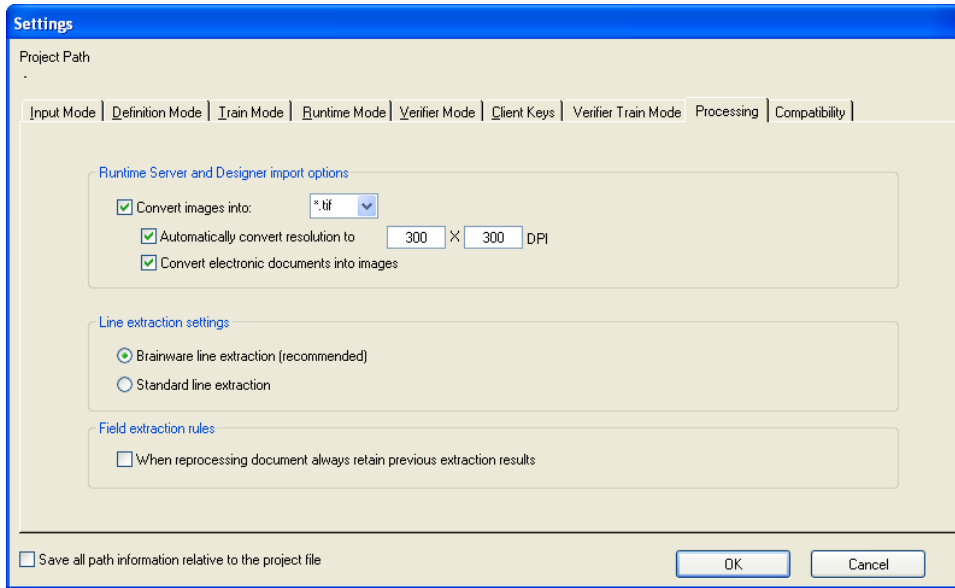


Figure 4-23: Processing tab – options for automatic document conversion

These options are available in the “Runtime Server and Designer import options” group box on the property page. The root option is *Convert images into*: allows selecting one of 39 currently supported image formats to convert the incoming document into.

Below is the detailed list of the currently supported image formats the converted documents can be saved on the disk with:

ID	Brief Format Description	File Extension	Supports Multiple Pages
1	Brooktrout file, Image Type: Bitmap.	BRK	No
2	Brooktrout file.	BRK	No
3	Native bitmap file format of the Microsoft Windows environment.	BMP	No
4	Native bitmap file format of the Microsoft Windows environment, RLE-compressed.	BMP	No
5	Monochrome bitmap, to provide a standardized graphics interchange for electronic graphics and image processing applications.	CAL	No
6	Windows Clipboard, RLE or uncompressed color bitmap.	CLP	No
7	Color table, RLE Monochrome, 24-bit RGB Bitmap. To allow the storage of multiple PCX files in one file.	DCX	Yes
8	Encapsulated PostScript, monochrome Metafile Storage and interchange of single images.	EPS	No
9	Image Object Content Architecture 4, 8, 16-bit palette gray-scale and color, G3 compression.	ICA	No

ID	Brief Format Description	File Extension	Supports Multiple Pages
10	Image Object Content Architecture 4, 8, 16-bit palette gray-scale and color, G4 compression.	ICA	No
11	Icon resource file, Storage and interchange of iconic bitmaps (icons), across many Windows applications, regardless of the pixel resolution and color scheme of the display hardware.	ICO	Yes
12	IMNET graphics.	IMP	No
13	Electronic Art's Interchange File Format, RGB palette up to 24-bit, binary Bitmap.	IFF	No
14	JPEG File Interchange Format, Interchange of .JPG files between applications on different platforms.	JPG	No
15	Mixed Object Document Content Architecture, To allow multiple IOCA images to be stored in one file, G3 compression.	MOD	Yes
16	Mixed Object Document Content Architecture, To allow multiple IOCA images to be stored in one file, G4 compression.	MOD	Yes
17	NCR Image.	NCR	No
18	NCR Image, compression G4.	NCR	No
19	Macintosh PICT, to capture a picture as a set of Macintosh QuickDraw library function calls.	PCT	No
20	PC Paintbrush File Format, Storing images for ZSoft Corporation's paint program, such as PC Paintbrush.	PCX	No
21	Portable Network Graphics, True Color up to 48 bpp, Effective lossless compression of True Color images.	PNG	No
22	Adobe Photoshop, all colors including RGB, CMYK, multi-channel Bitmap.	PSD	No
23	Sun Raster Data Format, RGB color map or True Color Bitmap, Bitmap format for Sun UNIX platforms.	RAS	No
24	SGI Image File Format, Display of images from the SGI image library.	SGI	No
25	True vision Targa, Alpha data 1 or 8-bits Bitmap (2D raster).	TAG	No
26	Tagged Image File Format, Standardized data storage and interchange of images obtained from scanners and incorporated into desktop publishing.	TIF	Yes
27	Tagged Image File Format, Standardized data storage and interchange of images obtained from scanners and incorporated into desktop publishing, G3 compression.	TIF	Yes

ID	Brief Format Description	File Extension	Supports Multiple Pages
28	Tagged Image File Format, Standardized data storage and interchange of images obtained from scanners and incorporated into desktop publishing, G3 compression, 2-dimensional.	TIF	Yes
29	Tagged Image File Format, Standardized data storage and interchange of images obtained from scanners and incorporated into desktop publishing, G4 compression.	TIF	Yes
30	Tagged Image File Format, Standardized data storage and interchange of images obtained from scanners and incorporated into desktop publishing, Huffmann compression.	TIF	Yes
31	Tagged Image File Format, Standardized data storage and interchange of images obtained from scanners and incorporated into desktop publishing, JPEG compression.	TIF	Yes
32	Tagged Image File Format, Standardized data storage and interchange of images obtained from scanners and incorporated into desktop publishing, LZW compression.	TIF	Yes
33	Tagged Image File Format, Standardized data storage and interchange of images obtained from scanners and incorporated into desktop publishing, packed.	TIF	No
34	X Bitmap, none compression, primarily for storing cursor and icon bitmaps for the X System graphical user interface.	XBM	No
35	X PixMap, none compression, to store X Window Pixmap information to disk.	XPM	No
36	X Window Dump, 16, 24 or 32 True Color Bitmap, no compression, and storing screen dumps from the X Window System.	XWD	No
37	Microsoft Windows Metafile, no compression. Convenient storage and interchange for applications run under MS Windows.	WMF	No
38	Lura Document format.	LDF	No
39	JP2 JP2000 Format.	JP2	No

Table 4-2: Supported image formats

The other two options *Automatically convert resolution to* and *Convert electronic documents into images* can activate automatic conversion of image files' resolution to user specified values, in DPI units (recommended values are 300 x 300 DPI) and automatic conversion of electronic documents (e.g. Excel files, Word documents, PowerPoint presentation, PDF files, etc.) to the same image format. Both sub-options only have an effect when the root one *Convert images into ...* is turned on.

The default values for all the options are:

- “Convert images into” – Unchecked in order to satisfy general backwards compatibility requirement. Recommended value is checked. The default value for the image format is supposed to be the one with ID 29 from the formats' table above.

- “Automatically convert resolution to” – Checked. As noted above, the checked state affects the importing in WebCenter Forms Recognition only if “Convert images into” option is on. The default values for both X- and Y-resolution is 300 DPI (dots per inch).
- “Convert electronic documents into images” – By default Unchecked.

When expanding the dropdown list, the currently configured image format is selected:

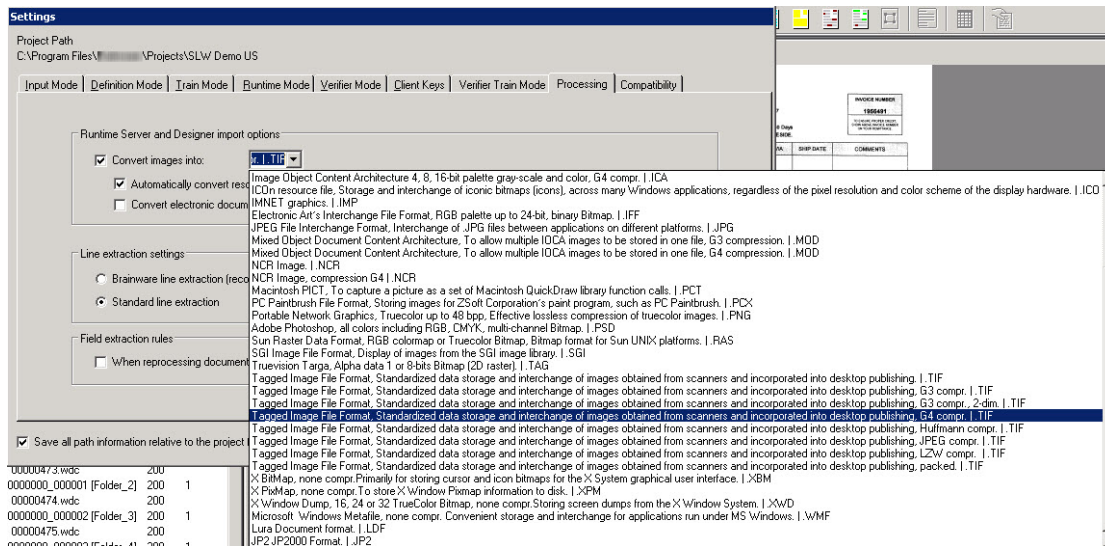


Figure 4-24: Available image format dropdown list

The above mentioned options affect the following parts of processing in WebCenter Forms Recognition:

1. Importing of documents in Document Browsing mode of the Designer application.
2. Importing of documents in the Runtime Server application.

During conversion, the original imported documents remain unchanged, while the documents attached to the WebCenter Forms Recognition’s WorkDocs associated with the imported files are converted to the desired image file format. If the desired format supports multipage document files (see “Supports Multiple Pages” column in the table of formats above), the imported pages of an image file or an electronic document are all going to be stored in a single document file referenced by a single WorkDoc. Otherwise, the imported pages are going to be split into individual document files – all attached to the same single WorkDoc.

**Example**

When importing one 10-pages plain text file and one 2-pages TIF file (with mixed resolution for different pages: first page with 200x200 DPI (common FAX resolution) and the second page with 300 x 300 DPI (desired for WebCenter Forms Recognition)):

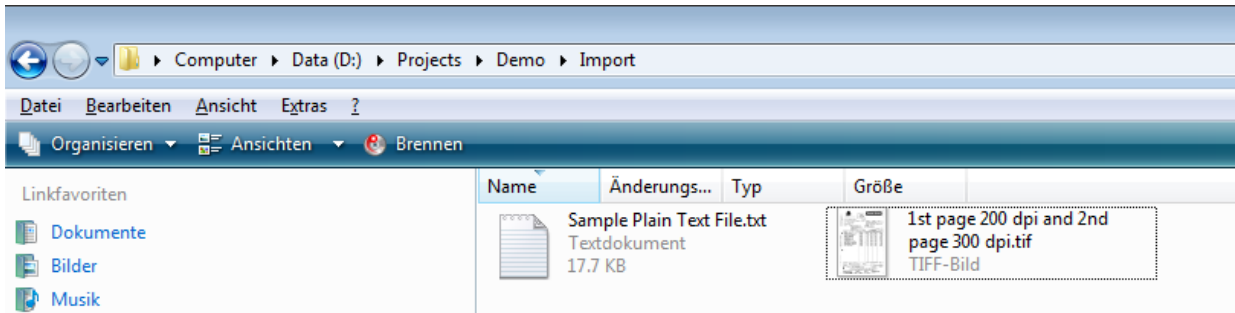


Figure 4-25: Sample for text file import

In Document Browsing mode of Designer:

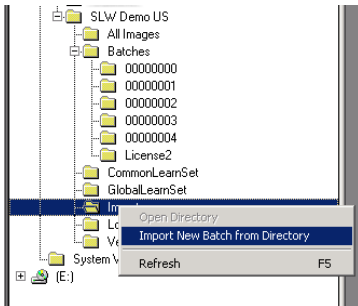
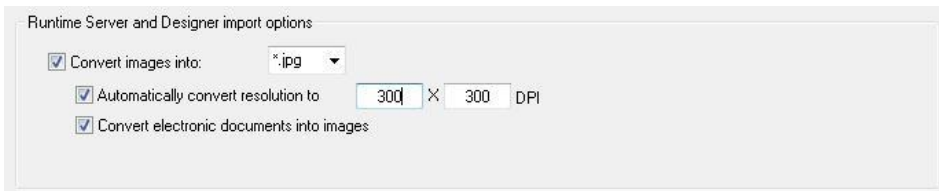
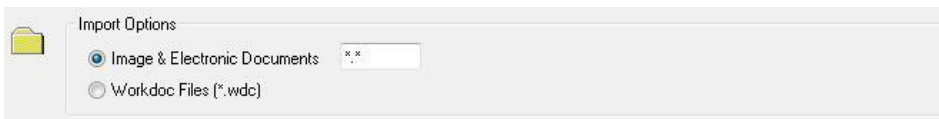


Figure 4-26: Document browsing mode

With the following project level conversion settings:



Also the following Designer's import settings:



WebCenter Forms Recognition's importing sub-system is going to produce the following outcome:

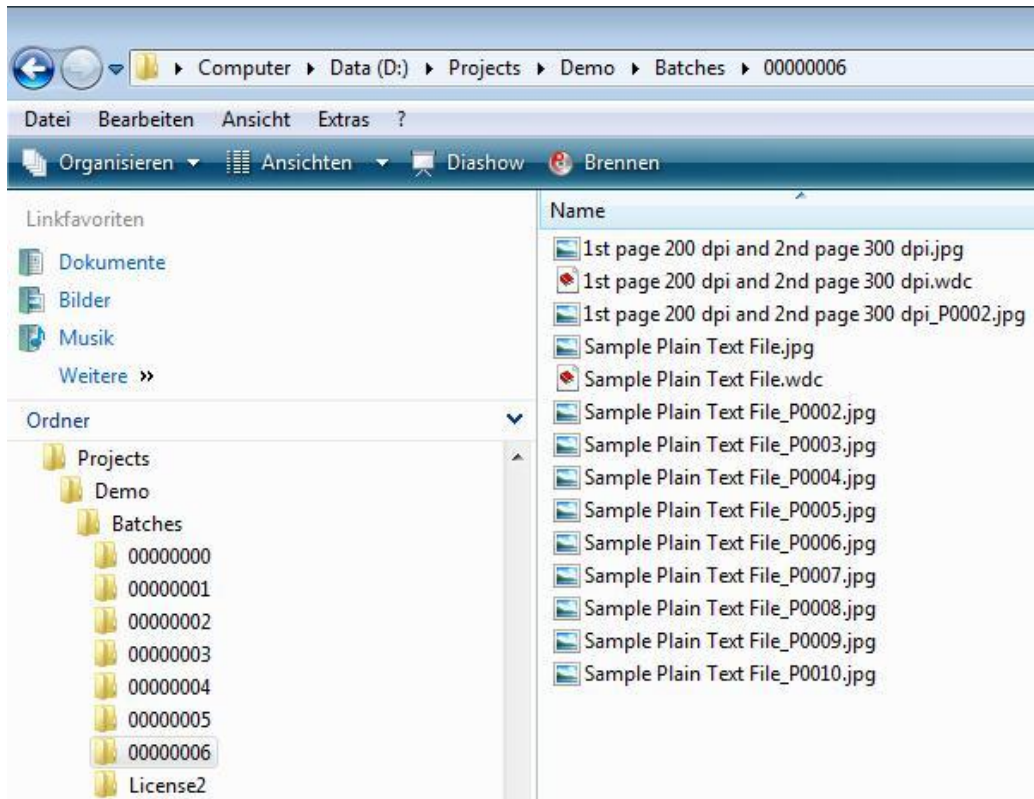


Figure 4-27: Outcome view

For example, two WebCenter Forms Recognition documents (two WorkDocs) with one individual JPEG file per each imported page of the 2-page TIF file and 10-page Text file. In this connection, all 12 JPEG files have the desired 300 x 300 DPI resolutions.

## Usage

The importing options can be primarily used for the following purposes:

- Stabilization of document processing in WebCenter Forms Recognition in relation to unification of the documents' resolution (usually it is supposed to be 300 x 300 DPI for all Learnset and extracted images) and, in some rare case, to prevent from degradation of extraction results when processing certain types of electronic documents where the corresponding subsystem in WebCenter Forms Recognition (third-party INSO library) is not capable of delivering good quality results for some properties of the processed electronic files (i.e., in cases when image based processing with OCR post-processing appears to be better in terms of extraction quality).
- Unification of processed documents, i.e., when different incoming formats are converted into a single, most optimal in terms of temporary data storage and data processing.
- Further archiving, when original documents are supposed to be archived in an external database along with the extracted data entries in one unified format.

### 4.8.3. Definition Mode

Definition Mode is the main working mode of Designer.



**To switch to Definition Mode:** 

- On the *View* menu, select *Definition Mode*.
- Alternatively, on the toolbar, click *Switch to Definition Mode*.

**4.8.3.1. Purpose – Definition Mode**


Working in this mode allows you to:

- design the settings for the project
- establish classification and data extraction
- generate a default export file
- execute an export during classification/extraction.
- Test single documents or an entire document set

**4.8.3.2. Settings – Definition Mode**

In Definition Mode, you can automatically process an entire document set in a single run. To do this, you will need enough time to view the results for one document before the next one is displayed.

**To set the processing delay:**

- 1) On the *Options* menu, select *Settings* or, in the toolbar, click *Show settings*. 
- 2) In the *Settings* dialog box, select the *Definition Mode* tab.
- 3) Under *Definition Mode*:
  - type the delay in seconds into the *sec* text box, or
  - use the slider to set a value.

**4.8.3.3. Support of non-western languages**

WebCenter Forms Recognition 11.1.1.8.0 provides a generic approach to the support of “non-western” languages, where essential parts of their alphabet are not covered by letters of the English alphabet with the same key codes; for example, Greek, Polish, Russian, and Arabic languages.

An automatic phonetic translation is applied from non-western language(s) into English and then an unambiguous conversion back to the original language(s). The conversion is applied for the following workflow steps in WebCenter Forms Recognition:

- Classification
- Extraction
- Validation
- Learning

These processes can be executed on a non-native PC (e.g. on a UK or US workstation or server) to process documents that contain auxiliary information and information to be extracted in supported non-western languages.

The following WebCenter Forms Recognition engines allow phrases, format strings, etc. to be entered with visual input using the characters of the currently supported non-western languages:

- Phrase Classification engine
- Table Analysis engine
- Format Analysis engine
- Associative Search engine (this engine includes the ability to import Unicode CSV files)

Configuring the Phrase Classification, Table Analysis and Format Analysis engines using phrases / strings in the desired non-western language should be applied on the corresponding non-western PC or on any PC where this language is set as the default system's Unicode language.

This is also applicable for all other WebCenter Forms Recognition engines that do not require any visual input in non-native languages like the Brainware Extraction engine, the Brainware and ASSA Classification engines, the Brainware Layout Classification engine, and so on.

Once support for non-western languages has been activated, the project has to be saved and relearned.

While the support option for non-western languages always looks for a one-to-one character transformation, the CJKT support may lead to an increase in text length of the transliterated object. The multi-character representation is phonetically more precise but increases the original text length. The attributes of the SCBCroWorktext objects like `.TextLength` refer to the original character length. The attribute `.TextProperty` and the WinWrap's "Len" method will return the length of the transliterated text. This limitation applies only during the "transliteration phase", which usually corresponds to internal core classification and extraction and also involves some script events, like PostEvaluate, executed during the core extraction.

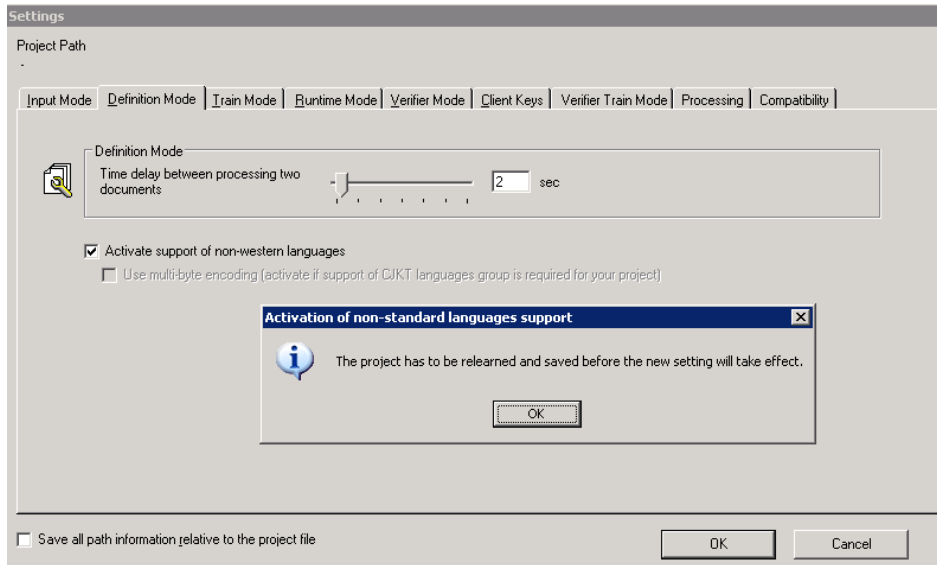


Figure 4-28: Settings for Definition Mode

If you have CJKT languages in your project, you need to activate the multi-byte encoding on the *Definition Mode* tab:

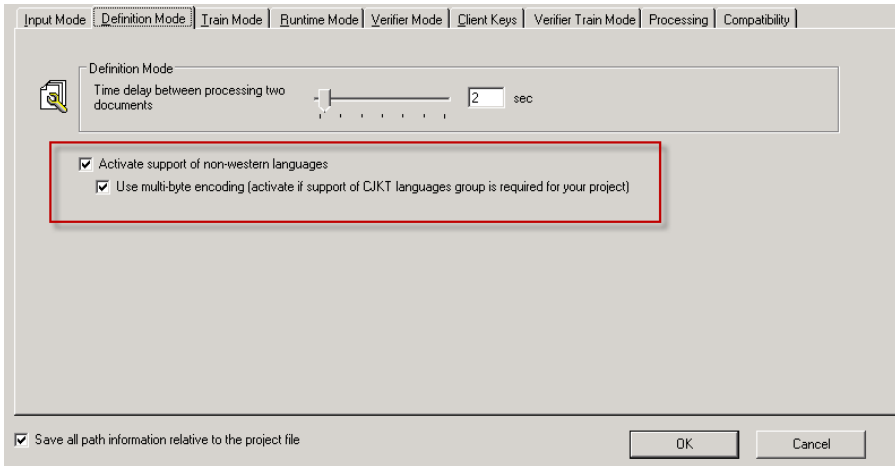


Figure 4-29: Settings for CJKT languages

See section [RUNTIME MODE](#) for export options.

### 4.8.3.4. User Interface – Design Project Level

The user interface of Definition Mode is the core of Designer.

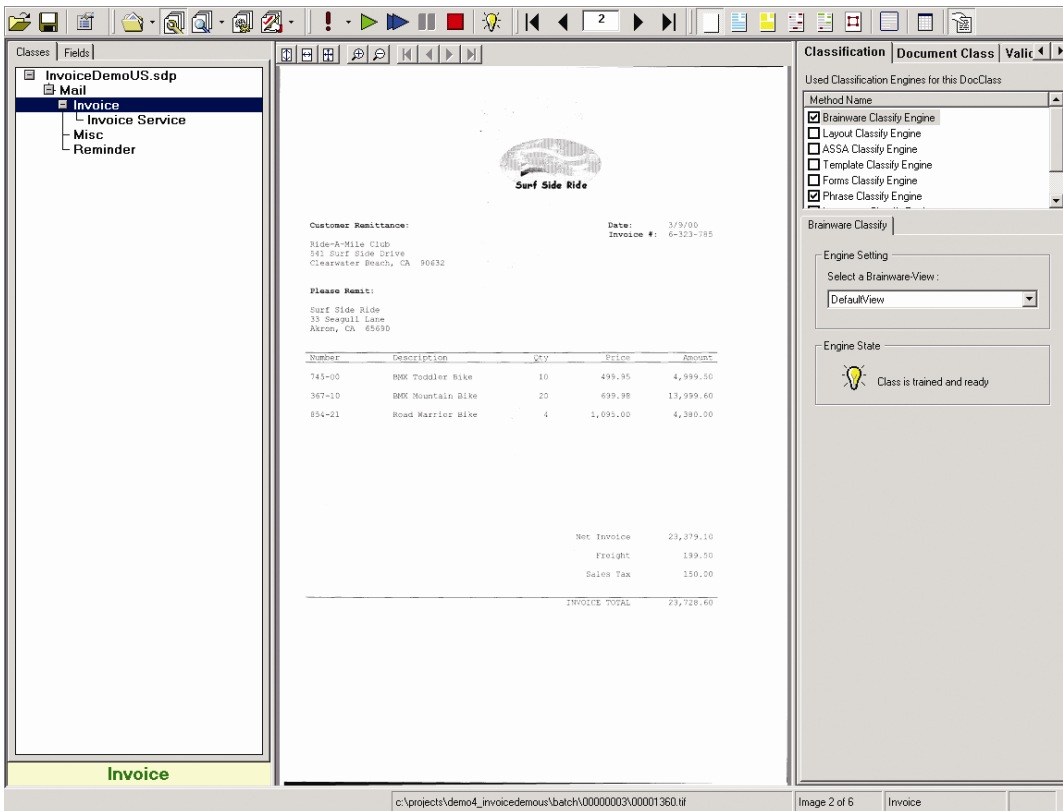


Figure 4-30: User Interface of the Definition Mode

## 1. Project/Class/Field Definition window.

Its two tabs enable you to select a project, class, document, or field to work on.

## 2. Document View.

Display the most recently selected document.

## 3. Property Editors.

This panel displays the currently selected document.

### 4.8.3.5. Project-Level Properties

Four types of project-level properties are available for definition when you select the \*.sdp file on the Class tab on the left side of the screen: Project; Classification, Validation and Supervised Learning. All four property types can be accessed on dedicated tabs on the right side of the screen. These tabs are called editors.

To access these property editors, select *Show Properties* on the *Edit* menu.

#### Project Editor

The Project Editor contains general classification settings such as interpretation, thresholds, and distances for standard classification and parent classification, OCR, word segmentation, and storage.

#### Classification Editor

The Classification Editor enables you to assign a default class to which documents will be assigned if they cannot be automatically classified. It also displays the engines used to classify Document Classes. (See [SETTING UP THE CLASSIFICATION](#) to learn how to use the Classification Editor.)

#### Advance Editor

At the project level, the only validation setting available is whether to permit or forbid Forced Validation. (See [SETTING UP THE VALIDATION](#) to learn how to use the Validation Editor.)

#### Supervised Learning Editor

Supervised Learning is configurable only at the project level. See section [CONFIGURING THE ASSOCIATIVE SEARCH ENGINE FOR CLASSIFICATION IN AUTOMATIC SUPERVISED LEARNING](#) and section [PLANNING](#) to learn about configuring Supervised Learning.

### 4.8.3.6. Class-Level Properties

Three types of project-level properties are available for definition when you select a base class or derived class from the Class tab on the left side of the screen: Classification, Document Class, and Validation.

#### Classification Editor

At this level, the Classification Editor is used to select a classification engine for the document class and to establish settings for the selected engine. (See [SETTING UP THE CLASSIFICATION](#).)

#### Document Class Editor

The Document Class Editor is used to establish settings for the selected document class. (See [SETTING UP THE CLASSIFICATION](#).)

#### Validation Editor

The Validation Editor is used to establish validation settings and to enable the Standard Validation Engine. (See [SETTING UP THE VALIDATION.](#))

#### 4.8.3.7. Field-Level Properties

Four types of field-level properties are available for definition when you select a field on the Field tab on the left side of your screen: Analysis, Evaluation, Field, and Validation.

##### Analysis Editor

This editor is used to select the analysis (extraction) engine. (See [SETTING UP THE DATA EXTRACTION](#)) and the settings from the selected engine

##### Evaluation Editor

This editor is used to select the evaluation engine, if this is necessary for the selected analysis engine

##### Field Editor

This editor is used to establish OCR settings for the selected field.

##### Validation Editor

The Validation Editor is used to establish validation settings and to enable the Standard Validation Engine. (See [SETTING UP THE VALIDATION.](#))


#### 4.8.4. Normal Train Mode

##### 4.8.4.1. Purpose – Classification Learnset

Use this mode to train classification using the properties defined in Definition Mode and the current document set selected in Document Selection Mode. Although learning is also available in Definition Mode, only Train Mode allows the creation of Learnsets.

##### 4.8.4.2. Selection – Classification Learnset


###### To switch to Normal Train Mode:

- From the *View* menu, select *Train Mode*. 
- Alternatively, in the toolbar, click the arrow next to *Switch to Train Mode*. and select Normal *Train Mode* from the drop-down menu

##### 4.8.4.3. Settings – Classification Learnset

Train Mode requires particular settings that enable you to create a Learnset for classification and data extraction.

###### To set the options for Train Mode:

- 1) From the *Options* menu, select *Settings*. Alternatively, in the toolbar, click  *Show settings*.
- 2) In the *Settings* dialog box, select *Train Mode* tab.
- 3) Under Train Mode:
  - To create a Learnset or add documents to it, mark the *Add trained documents to the Learnset...* check box. Clear the check box only if you want to use Train

Mode for demonstration purposes. By default, the check box is checked when you create a new project.

- Mark the *Execute classification for new documents only...* check box if you want Designer to classify each new document that is to be added to the Learnset for data extraction. You can use this option if your sample documents do not belong to the same classes. Clear the option if you work with sorted input.
- If *Review in normal train mode previously extracted values* is activated, during learning in Normal Train Mode the system will use previously extracted values from the document and pre-fill form fields with them.

#### 4) Under Train Manager:

- Select a directory where the documents of the Learnset will be stored. Every time you add a sample document to the Learnset of a class, it will automatically be copied there.

### 4.8.4.4. User Interface – Train Mode (Classification Learnset)

The user interface of Train Mode enables you to train classification and extraction. This section shows the user interface for training classification.

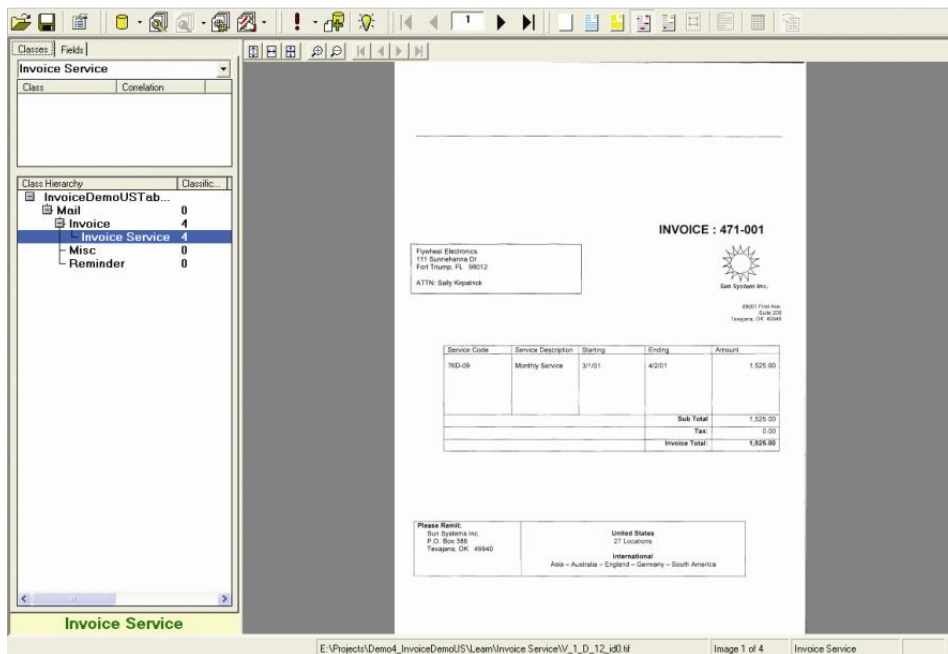


Figure 4-31: Train Mode Interface for Classification Learnsets

#### 1. Class/Field Correction

The left panel contains two tabs, one for classes, and the other for fields. The Classification tab is where you:

- Manually assign a document to a class
- Correct class assignments proposed by the program

When the Class tab is selected, the document will be added to the Classification Learnset.

#### 2. Document Display


The right panel displays the current document.

#### 4.8.4.5. Purpose – Extraction Learnset

Use this mode to train extraction using the properties defined in Definition Mode and the current document set selected in Document Selection Mode.

#### 4.8.4.6. Selection – Extraction Learnset


**To switch to Train Mode:**

- From the *View* menu, select *Train Mode*. 
- Alternatively, in the toolbar, click *Switch to Train Mode*.

#### 4.8.4.7. Settings – Extraction Learnset

Train Mode requires particular settings that enable you to create a Learnset for classification and data extraction. These settings are the same as those described in section [SETTINGS – CLASSIFICATION LEARNSET](#).

**To set the options for Train Mode:**

- 1) From the *Options* menu, select *Settings*. Alternatively, in the toolbar, click  *Show settings*.
- 2) In the *Settings* dialog box, select the *Train Mode* tab.
- 3) Under Train Mode:
  - To create a Learnset or add documents to it, mark the *Add trained documents to the Learnset...* check box. Clear the check box only if you want to use Train Mode for demonstration purposes. By default, the check box is checked when you create a new project.
  - Mark the *Execute classification for new documents only...* check box if you want Designer to classify each new document that is to be added to the Learnset for data extraction. You can use this option if your sample documents do not belong to the same classes. Clear the option if you work with sorted input.
- 4) Under Train Manager:
  - Select a directory where the documents of the Learnset will be stored. Every time you add a sample document to the Learnset of a class, it will automatically be copied there.

### 4.8.4.8. User Interface – Extraction Learnset

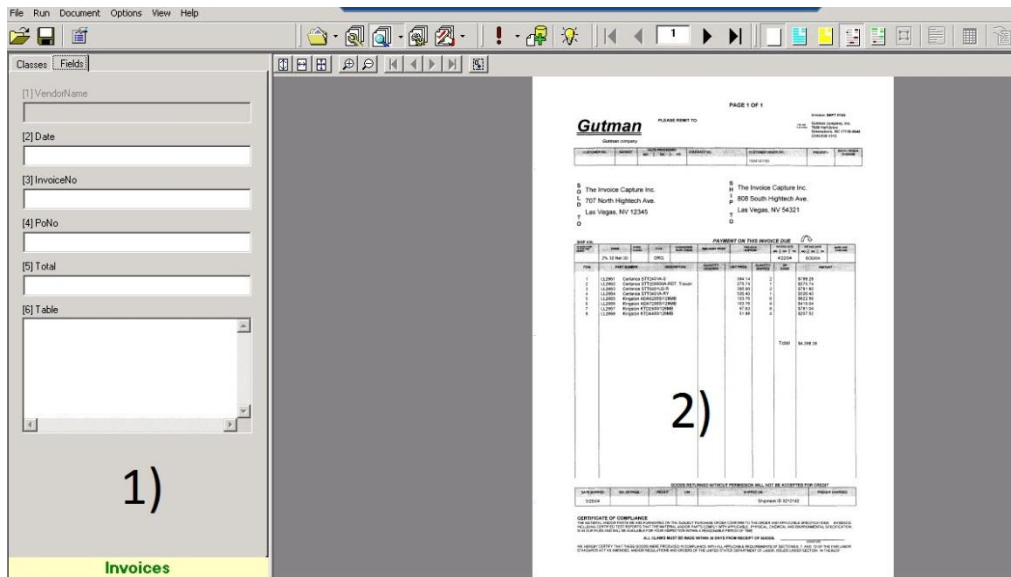


Figure 4-32: Train Mode Interface for Extraction Learnsets

#### 1. Class/Field Correction

The left panel contains two tabs, one for classes, and the other for fields. The Fields Tab is where you:

- Manually select the correct field for data extraction from several candidates
- Correct candidate selections for data extraction proposed by the program.

When a field on the Field tab is selected, the document will be added to the Extraction Learnset.

#### 2. Document Display

The right panel displays the current document.

### 4.8.5. Verifier Train Mode

#### 4.8.5.1. Purpose

Use this mode for Automatic Supervised Learning – an extension of Verifier that uses Associative Learning to extract supplier information.

Automatic Supervised Learning then

- extracts this information to a base Document Class,
- learns the new class,
- and migrate the local project to the global Learnset.

The intent of Automatic Supervised Learning is to improve the global knowledgebase, the quality of documents, and the length of time it takes to design applications.

To use the Verifier Train Mode, you must have already created a Verification Form.

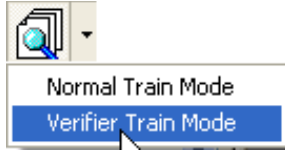
Use the Verifier Train Mode to check the extraction result. When a supplier is extracted correctly and this document is added to the Learnset, a new class is created automatically. To do this, settings must be established for the project level in Definition Mode on the



Supervised Learning editor and on the DocClass level on the Document Class tab (classification field) must be made.

#### 4.8.5.2. Selection

To switch to Verifier Train Mode:



- On the *View* menu, select *Verifier Train Mode*. Alternatively, in the toolbar, click the arrow next to the *Switch to Train Mode* button.
- On the drop-down menu, select *Verifier Train Mode*.

#### 4.8.5.3. Settings

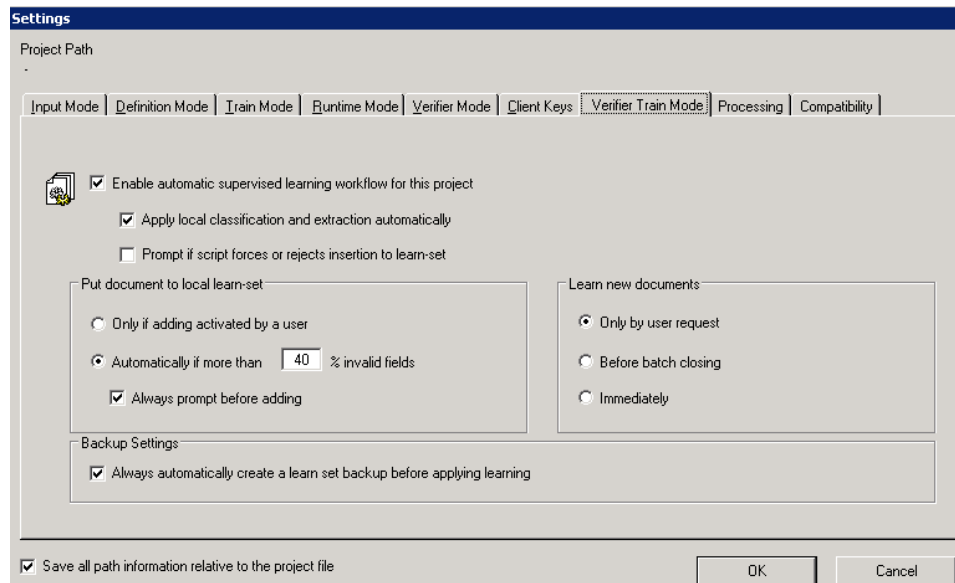



Figure 4-33: Verifier Train Mode Settings

- 1) From the *Options* menu, select *Settings*. Alternatively, in the toolbar, click  *Show settings*.
- 2) In the *Settings* dialog box, select the *Verifier Train Mode* tab.
- 3) Select *Enable automatic supervised learning workflow for this project*. If this checkbox is not enabled, Supervised Learning will not be available in the application.
- 4) Select to save all project information relative to project file.
- 5) Select *Apply local classification and extraction automatically*. New classes will be created using the supplier's name. A Learnset should also be created if you select this setting.
- 6) Select *Prompt if script forces or rejects insertion to the Learnset*.

- 7) Under *Put document to local Learnset*, select either
- *Only if adding activated by a user with Always prompt before Adding selected.* Verifies that a document was added to a Learnset if the document count is updated for a class that was created as the result of supplier extraction.
  - *Only if adding activated by a user with Always prompt before adding cleared.* Verifies that a document was added to the Learnset only if the user enabled activation.
  - *Automatically if more than N% invalid fields with Always prompt before adding selected.* Automatically verifies that a document was added to the Learnset if the number of documents in the class was updated. Learns documents when the specified percentage of fields is invalid before they are manually corrected.
  - *Automatically if more than N% invalid fields with Always prompt before adding cleared.* Learns documents only when they are initially unclassified.
- 8) Under *Learn new documents*, select:
- *Only by user request.* Learning is initiated only by user request.
  - *Before batch closing.* Learning is initiated for every batch in the project each time any batch is closed.
  - *Immediately.* Learning is initiated anytime a document is added to the Learnset.

#### 4.8.5.4. User Interface

The screenshot displays the Verifier Train Mode User Interface. On the left, there is a form area with a table of items and various input fields. On the right, there is a document preview area showing a scanned invoice. Numbered callouts 1) through 4) highlight specific areas of the interface.

Description	Quantity	SinglePrice
UNIFRAC 16 / 30	26,680	96,72
UNIFRAC 16 / 30	25,500	96,72
UNIFRAC 16 / 30	25,530	96,72
UNIFRAC 16 / 30	26,640	96,72
UNIFRAC 16 / 30	25,510	96,72
UNIFRAC 16 / 30	25,150	96,72

VendorName: UNIMIN

InvoiceNo: 1826490

PO #4502182154

Total: 14837.56

Date: 10/02/02

VendorName: UNIMIN

UNIMIN236043

Figure 4-34: Verifier Train Mode User Interface

### 1. Form Area

The form area contains a set of controls for correction and manual indexing.

- Form fields
- Labels

- Viewers

Each control is assigned to a data extraction field. The viewers show the document areas that were extracted to fill the fields.

## 2. Current Input Area

This area shows a magnification of the current form field and the associated viewer. It can be used for particularly convenient viewing and editing.

Note that the position of the form areas may vary according to the selected layout, or only a subset of form areas may be present.

## 3. User Info Area

This area displays the class name of the current document, the name of the current field and – if applicable – a message indicating why the current field is invalid.

## 4. Document Area

The panel on the upper right side displays the current document.

















Button	Description
	Move to previous page.
	Move to next page.
	Highlight all fields.
	Highlight selected fields.
	Highlight all candidates.
	Maximize image height.
	Maximize image width.
	Maximize image size.
	Zoom in.
	Zoom out.
	Keep focus on field.
	Keep zoom on switching to next document.
	Move to next document on validate
	Force document to be validated with selected form.
	Show/hide script.

Table 4-3: Controls in Verifier Train Mode

## 4.8.6. Runtime Mode

### To switch to Runtime Mode:

- On the View menu, select Runtime Mode. 

Alternatively, on the toolbar, click Switch to Runtime Mode.

### 4.8.6.1. Purpose

Working in this mode allows you to:


- simulate classification and extraction using the entire document set,
- test the export scripts,
- perform OCR on a document set that will be used in Definition or Train mode.

*Note: Except for the OCR output and the corresponding images, the results of the Runtime Mode are not saved persistently.*

### 4.8.6.2. Settings

For Runtime Mode, you need to define import and export options. In addition, you can either test classification only or classification plus extraction.

#### To set the options for Runtime Mode:

- 1) On the *Options* menu, select *Settings*. Alternatively, in the toolbar, click  *Show settings* button.
- 2) In the *Settings* dialog box, select the *Runtime Mode* tab.

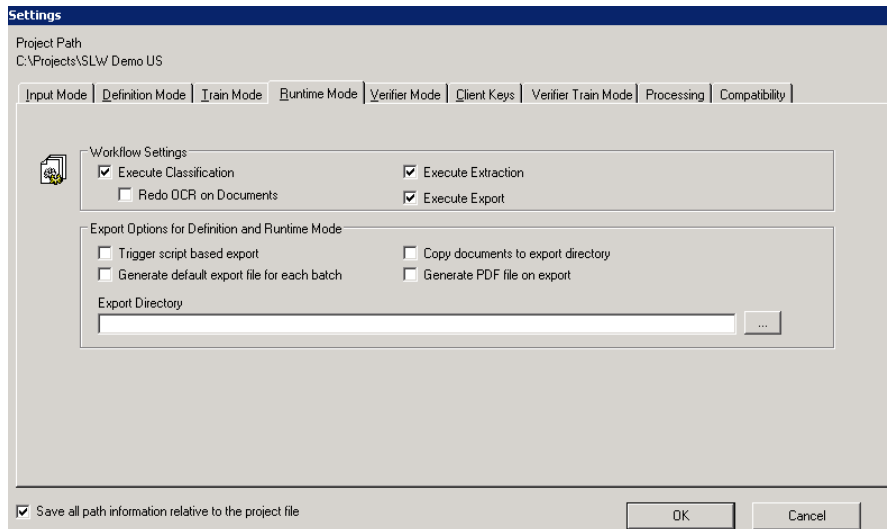


Figure 4-35: Runtime Mode Settings

- 3) Under *Workflow Settings*:
  - Select *Execute Classification* to automatically execute a classification of input.
  - Select *Execute Extraction* to automatically execute extraction of input.
  - Select *Redo OCR on Documents* to rerun an OCR with selected settings.
  - Select *Execute Export* to automatically execute an export of input.
- 4) Under *Export Options for Definition and Runtime Mode*:
  - Select *Trigger script-based export* to start a predefined script that contains export instructions.
  - Select *Copy documents to export directory* to copy processed documents from the import to the export directory instead of moving them.
  - Select *Generate default export file for each batch* to generate an ASCII file with the processing results. The export file will be re-created with each run. It will be written to the export directory. The file name is generated according to the naming convention <Name of the import directory>.exp.
  - Select *Generate PDF file on export* to generate a .pdf file with the processing results. The export file will be re-created with each run and written to the export directory. The file name is generated according to the naming convention <Name of the import directory>.pdf.

- Select *Export Directory* and enter a location where images and OCR output will be stored after processing

## Compatibility Options

On the Settings window, you will find an additional tab associated with the Runtime Mode: the *Compatibility* tab.

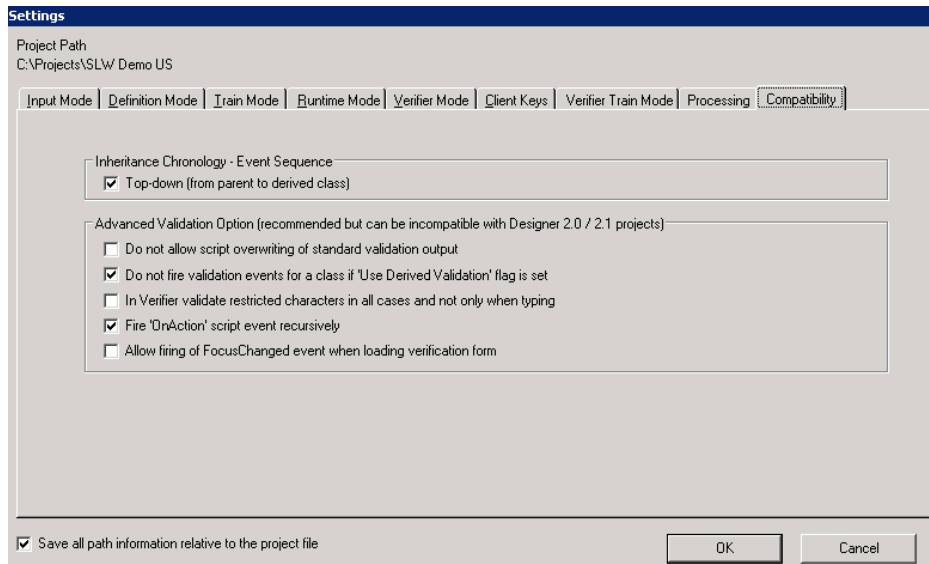


Figure 4-36: *Compatibility* tab

The following settings are possible:

- 1) Under *Inheritance Chronology - Event Sequence*:
  - Select *Top Down* (from parent to derived [child] class)
  - Clear *Top Down* (from derived to parent class – bottom up)

Parent classes sometimes contain multiple derived classes that might also be parent classes themselves. If a DocClass (more formally known as a Document Class) is derived from one or more parent DocClasses, each event for that DocClass will also be executed in all parent DocClasses, starting at the most senior parent class (Top Down).

Both *Bottom Up* and *Top Down* apply to available events. By default, the inheritance succession is set to Top Down.

For most users, the default setting of *Top Down* is the most appropriated. Change the settings for Inheritance Chronology event sequences only if there are scripts created for those events. Unless there is a lot of code, it is probably easier to change the code, rather than change the event chronology.

Event Chronology is a potential issue only for projects that contain script code for the events cited in WebCenter Forms Recognition’s scripting reference guide. It is better to change the event chronology on the settings tab for these projects.

- 2) Under *Advanced Validation Option*:
  - *Do not allow script overwriting of standard validation output*
  - *Do not fire validation events for a class if “Use Derived Validation” flag is set.*

- In Verifier validate restricted characters in all cases and not only when typing
- Fire "OnAction" script event recursively
- Allow firing of FocusChanged event when loading verification form

### 4.8.6.3. User Interface

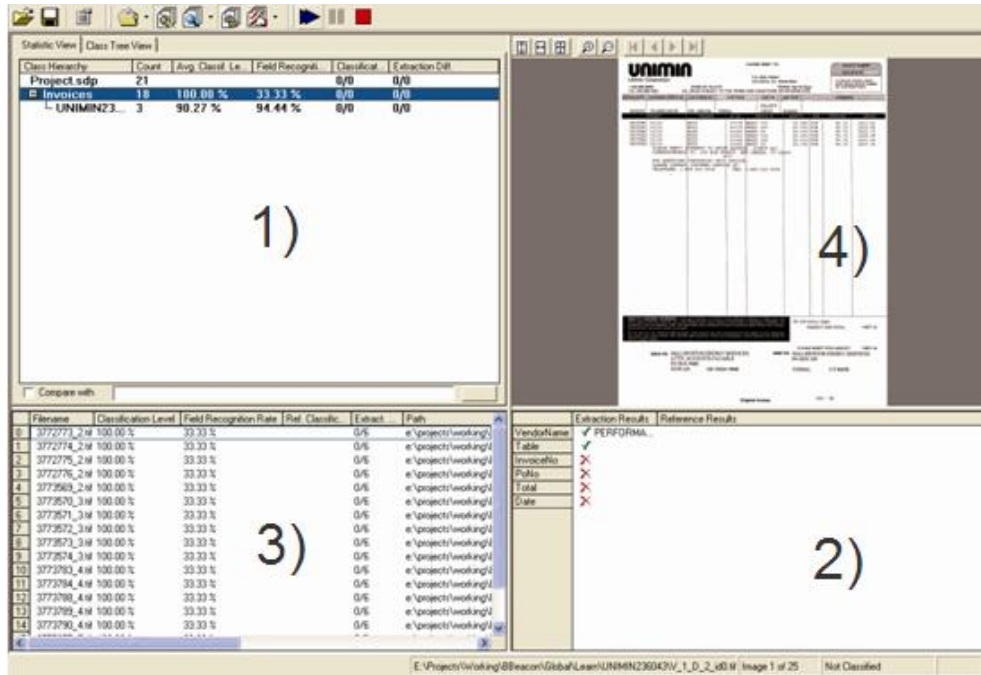


Figure 4-37: User Interface of the Runtime Mode

#### 1. Class Selection

The upper right panel on the left side contains a hierarchical representation of classes that are defined in the current project.

#### 2. Field Selection

The bottom panel on the right side displays a table with the mapping of data extraction fields and verification form fields of the current form. All panels on the left side are visible by default. To hide them, select the menu command View - Verifier Design Mode - Maximize Verification View, or click the toolbar button at right.

#### 3. Form Area

The form area consists of a background and a set of controls that will be used in Verifier for correction and manual indexing.

The following control types are available:

- Form fields
- Labels
- Buttons
- Viewers
- Tables

Each control must be assigned to a data extraction field. The viewers are used to show the document area that was extracted to fill a given field.

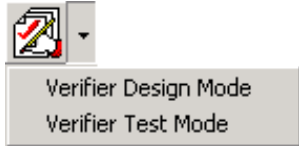
#### 4. Form Document Area

This area shows the document that has been displayed before selecting the Runtime Mode.

#### 4.8.7. Verifier Design Mode

To switch to Verifier Design Mode:

- On the View menu, select *Verifier Design Mode*.



- Alternatively, in the toolbar, click the arrow associated with the *Switch to Verifier Design Mode* button. From the drop-down menu, select *Verifier Design Mode*.

##### 4.8.7.1. Purpose

Working in this mode allows you to:

- create forms that will be used in Verifier to manually correct extracted data and for manual indexing,
- activate or deactivate Force Validation of a field,

set states for classification or extraction for testing a script in the Settings dialog box.

##### 4.8.7.2. User Interface

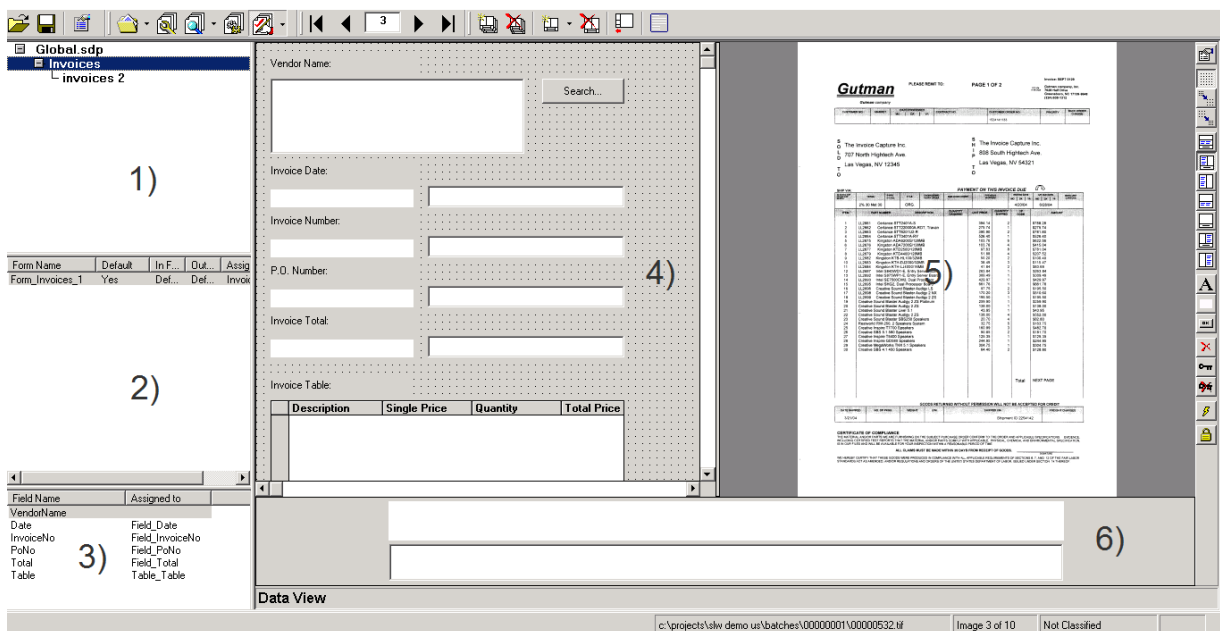


Figure 4-38: User Interface of the Verifier Design Mode




## 1. Class Selection

The upper panel on the left side contains a hierarchical representation of classes that are defined in the current project.

## 2. Form Selection

By default, the middle panel on the left side either displays the name and properties of the form that is assigned to the currently selected class. Alternatively, it displays all the forms that are available in the current project. To display all forms, right click a form name and select Show All from the shortcut menu. To switch again, select a class in the upper panel. Note that the form properties and the remaining commands in the shortcut menu are not yet relevant because at the moment only one form can be defined for each class.

## 3. Field Selection

The bottom panel on the left side displays a table with the mapping of data extraction fields and verification form fields of the current form. All panels on the left side are visible by default. To hide them, select the menu command View - Verifier Design Mode - Maximize Verification View, or click  on the toolbar.

## 4. Form Area

The form area consists of a background and a set of controls that will be used in Verifier for correction and manual indexing.

The following control types are available:

- Form fields
- Labels
- Buttons
- Viewers
- Tables

Each control must be assigned to a data extraction field. The viewers are used to show the document area that was extracted to fill a given field.



## 5. Form Document Area


















This area shows the document that has been displayed before selecting the Verifier Design Mode.

## 6. Form Current Input Area

This area shows a magnification of the current form field and the associated viewer. The area can be used for particularly convenient viewing and editing.

Note that the position of the form areas may vary according to the selected layout, or only a subset of form areas may be present. The layout shown above is the default layout.

Button	Description
	Switch to Document Input Selection (Directory, Batch, Learn Set)
	Switch to Definition Mode

Button	Description
	Switch to Normal Train Mode/ Verifier Train Mode
	Switch to Runtime Mode
	Switch to Verifier Design Mode/ Verifier Test Mode
	Insert default verification forms for all doc class
	Delete all verification forms
	Insert default form/Insert empty form
	Delete verification form
	Maximize Verification form view
	Show/hide script
<b>Toolbar (right hand side)</b>	
	Show selected object properties
	Show/hide Grid
	Enlarge Grid Step
	Shrink Grid Step
	Data View Layout Style 1 The Verifier window is split into three horizontal panes: - Document at the top - Verification form in the middle - Magnification view of the current field at the bottom
	Data View Layout Style 2 The upper pane of the Verifier window is split in two areas: - Verification form on the left - Document view on the right The lower pane displays the magnification view of the current field.
	Data View Layout Style 3 This layout is similar to layout style 2, without the magnification view.
	Data View Layout Style 4 This layout is similar to layout style 1, without the magnification view.












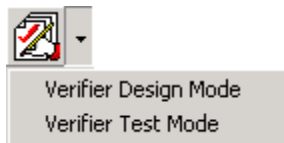
Button	Description
	Data View Layout Style 5 This layout style is recommended for projects with only a few fields (1 – 5) to verify. The Verifier window is split into two horizontal panes: - Document view at the top - Magnification view of the current field at the bottom
	Data View Layout Style 6 The upper pane of the Verifier window is split in two areas: - Document view on the left - Verification form on the right The lower pane displays the magnification view of the current field.
	Data View Layout Style 7 This layout is similar to layout style 6, without the magnification view.
	Add New Label Element
	Add New Viewer Element
	Add New Button Element
	Delete Selected Elements
	Define Smart Indexing
	Delete Smart Index
	Show Action List
	Lock Verifier controls

Table 4-4: Controls in Verifier Design Mode

### 4.8.8. Verifier Test Mode

To switch to Verifier Test Mode:




- From the *View* menu, select *Verifier Test Mode*.
- Alternatively, in the toolbar, click the arrow associated with the *Switch to Verifier Design Mode* button.  
From the drop-down menu, select *Verifier Test Mode*.

### 4.8.8.1. Purpose

Working in this mode allows you to test the verification step with forms that were created in Verifier Design Mode.

### 4.8.8.2. Settings

Verifier Test Mode includes settings for Force Validation and Designer Test Mode Document Routing.

- 1) From the *Options* menu, select *Settings*. Alternatively, in the toolbar, click  *Show Settings*.
- 2) For *Force Validation*, select whether Verifier users can validate a field even if its content is invalid by definition.
  - *Permitted*: The user can force validation by pressing ENTER three times.
  - *Forbidden*: The user cannot force validation.

### Designer Test Mode Document Routing

If you have script code associated with the ScriptModule\_RouteDocument event, you can use the states entered in this area to test your scripting in Designer's Verifier Test Mode. These states will not be used in Verifier.

Enter values for the states you want to test.

- Classification Success State
- Classification Failure State
- Extraction Success State
- Extraction Failure State

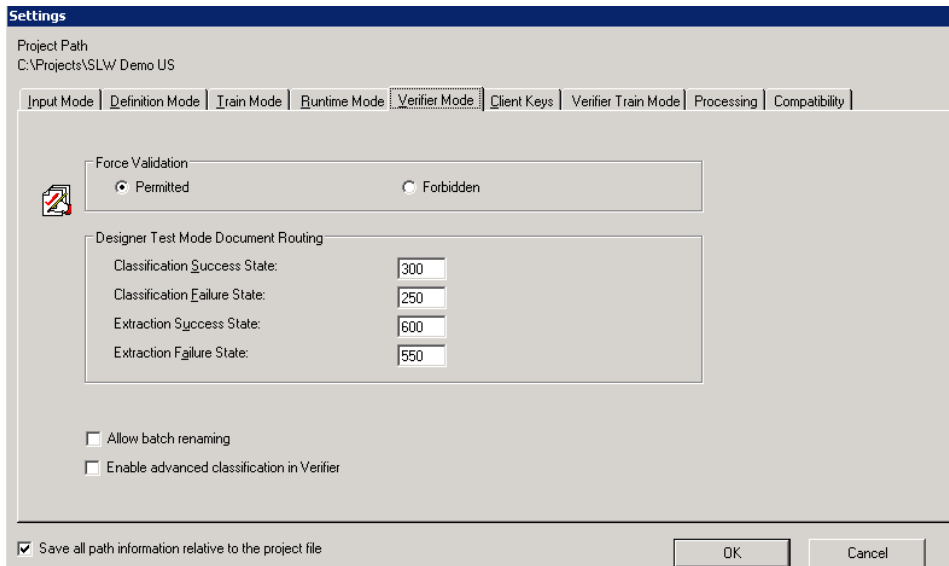


Figure 4-39: Verifier Mode settings box: Settings tab for Verifier Test Mode

### Batch Renaming

This functionality, which is established as part of the options on the Verifier Test Mode (The tab is actually named *Verifier Mode*), enables Verifier users to dynamically rename a batch

when they send a document to a pre-defined exception state. Based on a name supplied by the user, Verifier creates an exception batch with the document and removes the original document from the source batch. The user can rename a batch by right clicking on it on the Verifier summary screen.

Batch renaming gives Verifier users more flexibility in working with exception batch functionality. It also gives developers more freedom to manipulate batches via custom scripting.

To ensure traceability and consistency in combination with the use of statistical tools it is a benefit to store information of the moved documents. Following information is found in the Move Document event of the Event Log: Old Batch ID, New Batch ID, Reason (why the document was moved, Document State (Workflow State).

### 4.8.8.3. User Interface

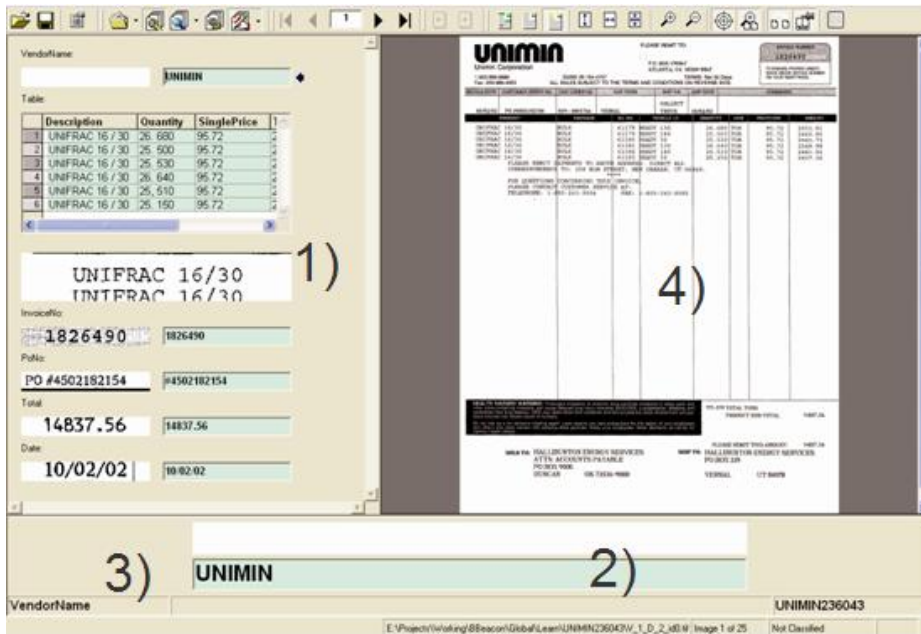


Figure 4-40: User Interface of the Verifier Test Mode

#### 1. Form Area

The form area contains a set of controls for correction and manual indexing.

- Form fields
- Labels
- Viewers

Each control is assigned to a data extraction field. The viewers show the document areas that were extracted to fill the fields.

#### 2. Current Input Area

This area shows a magnification of the current form field and the associated viewer. It can be used for particularly convenient viewing and editing.

*Note:* The position of the form areas may vary according to the selected layout, or only a subset of form areas may be present.

### 3. User Info Area

This area displays the class name of the current document, the name of the current field and – if applicable – a message indicating why the current field is invalid.

### 4. Document Area

The panel on the upper right side displays the current document.

## 4.9 Working with Projects

### 4.9.1. What Are Project Files?

In a WebCenter Forms Recognition Designer application, the automatic analysis and processing of incoming documents relies on a group of settings and on trained knowledge. For each application, these settings are saved persistently in a project file. The project file stores the following information:

- General settings such as the location of import and export directories.
- Document classes and their hierarchical relationship.
- Neural networks for full-text classification.
- References to Learnsets for full-text classification and their status.
- Phrases and rules for rule-based classification.
- Settings for image size classification.
- Field names and field positions for data extraction.
- Analysis settings and standard validation rules for data extraction.
- Field recognition settings for data extraction.
- Table recognition settings for data extraction.
- Neural networks for data extraction and their status.
- Form definitions for data verification in Verifier.
- WinWrap Basic scripts for document classification, data extraction, data validation, data verification, and document export.

For more information on document import, refer to section [IDENTIFYING THE DOCUMENT IMPORT FORMATS](#).

For more information on document classification, refer to [SETTING UP THE CLASSIFICATION](#).

For more information on data extraction, refer to [SETTING UP THE DATA EXTRACTION](#).

For more information on data verification, refer to [SETTING UP THE VERIFICATION](#).

For more information on document export, refer to [SETTING UP THE DOCUMENT EXPORT](#).

### 4.9.2. Creating Projects

In Designer, there is always precisely one active project. There are two ways to create a project, depending on the application's state.

If Designer is active:

- From the *File* menu, select the *New Project* command. Then follow the instructions on screen.


If Designer is not active or to start a new instance:

- Start Designer via the Windows Start menu. This creates a new project automatically.

*Note:* Creating a new project does not mean there is already a project file. To create a project file, you first need to save the project. (See section [SAVING PROJECTS](#)).

### 4.9.3. Saving Projects

When you have modified your current project and attempt to close it, Designer asks you to save changes that could be lost. If you want to save a project without closing it, use one of the following methods:

- On the *File* menu, select *Save Project*. This saves an existing project file. If your project is new, you need to enter a file name. The project file is then saved with the extension \*.sdp.
- Click  *Save Project* on the toolbar.
- You can also save your project under a new name. Select the *File* menu and then the *Save Project As* command. You need to specify a new file name.
- *Save and Compress* is a third option available for saving your project. This functionality prevents the growth of a project file that is saved after a document class is removed.

#### 4.9.3.1. Project Compression to Control File Size Growth

This new feature was needed because saving a project file after removing a document class and its associated Learnset caused the project file to grow (or at least remain the same,) rather than to decrease.

##### Choosing Save and Compress:

- Saves a temporary project file.
- Replaces the original file with this temporary file.
- Removes the temporary project file.

Because *Save and Compress* is slower than *Save* or *Save Project As...*, the new *Save and Compress* should only be used when you release a project file for production, not when you are actively working on the project.


However, using *Save and Compress* will not cause the size of the project file to decrease from the size it was before the document class was deleted. Rather, it will keep the size the same as it was before the deletion. To actually decrease the size of the file, you need to use *Save Project As...*

You can save a project with or without Learn data. To elect either one, in the *Save As* dialog box, select the file type drop-down box. Select either <project name> (\*.sdp), or <project name> without learn data (\*.sdp).

#### 4.9.4. Opening Projects

##### If Designer is active:

There are four ways to open a project:

- On the *File* menu, select the *Load Project* command. Then, select an \*.sdp file from the file system and open it.
- On the *File* menu, select one of the recent projects.
- On the toolbar, click  *Load Project*. And then select a \*.sdp file from the file system and open it.
- Login to Designer using one of the command line calls discussed in section [STARTING DESIGNER](#).

You can load or save a file with or without network data. When loading, click the *File Type* dropdown box and select either <project name> (\*.sdp), or <project name> without learn data (\*.sdp).

##### If Designer is not active or to start a new instance:

- Create a link to the project file on your computer desktop and double click the corresponding icon to open the project.
- Use the command line call /p <project file name> to open the specified project.
- Use the command line call /l to open the previously edited project.

##### Opening a Project that uses Sax Basic

If you open a project that uses Sax Basic, you will be presented with a reminder message:

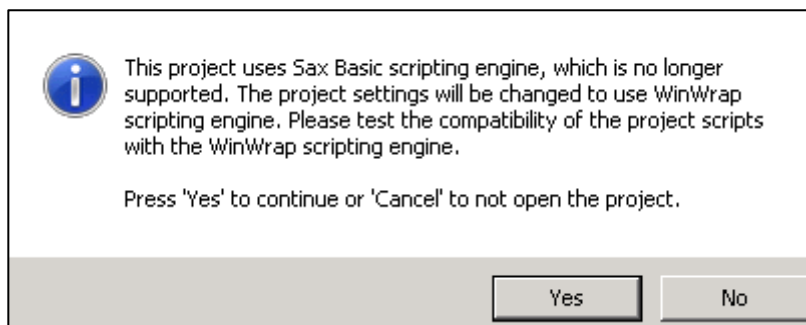


Figure 4-41: Opening a project that uses Sax Basic scripting engine

Please refer to section [PROJECT WITH SAX BASIC TO THE WINWRAP SCRIPTING ENGINE](#) for more information.

##### Opening a Project that uses Template Classification Engine



The Template Classification engine is no longer supported and is now fully replaced by the Brainware Layout Classification engine. If you open a project that uses the Template Classification engine, the following notification will appear:

Project file could not be loaded!

Error Description: SCBCdrProject: Failed to load Cedar project, because invalid file or missing component. - Cannot create collection item with name "Template Classify Engine", because object is not registered.

Designer can try to load the project file once again ignoring all errors. This may result in complete or partial lost of project information.

Do you want to retry loading the project file anyway?

Please refer to section [PROJECT WITH TEMPLATE CLASSIFICATION ENGINE TO THE BRAINWARE LAYOUT CLASSIFICATION ENGINE](#) for further instructions.

### 4.9.5. Using Version Control

WebCenter Forms Recognition Designer features an integrated version-control system. This feature is handy if you are designing complex applications. With version control, you can save your work or back it up in intermediate stages.

For example, you could design your application in stages. In the first stage, define the document classification. Once this part is working, create a version of your project. If anything goes wrong during the design of the extraction part, you can always go back to the version you created earlier.

By default, version control is off. You can activate it when you are about to save an intermediate stage.

#### To use version control:

- From the *File* menu, select the *Versions* command. The *Versions* dialog box is displayed.

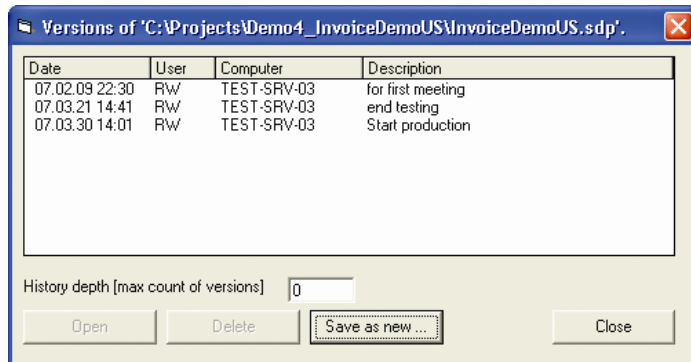


Figure 4-42: Project versions

- To specify the maximum number of versions for the current project, enter an integer in the *History Depth* text box. Once the maximum number of versions is reached, you will not be able to create additional versions. In this case, increase the maximum number of versions or delete versions that are no longer required.
- To create a new version, click *Save as new*. And then enter a description for the new version.
- To access a previously saved version of your project, select it from the list of versions and click *Open*.

- To delete a previously saved version of your project, select it from the list of versions and click *Delete*.

#### 4.9.6. Setting Global Project Variables


The project settings tree is a miniature registry for WebCenter Forms Recognition. With it, you can:

- Create keys that represent global variables.
- Assign values to these keys.
- Organize the keys in a hierarchy.
- Edit key values directly in tree.
- Manage sets of values for multiple clients or customers.

The variables are defined in Designer and employed in WinWrap Basic scripts. In Designer Runtime Mode, in WebCenter Forms Recognition Runtime and Verifier, the appropriate set of key-value pairs becomes available when a client is selected.

##### 4.9.6.1. Displaying the Client Keys Tree

To display the client keys tree:

- On the *Options* menu, select *Settings*. Alternatively, on the toolbar, click  *Show settings* button.
- In the *Settings* dialog box, select the *Client Keys* tab.

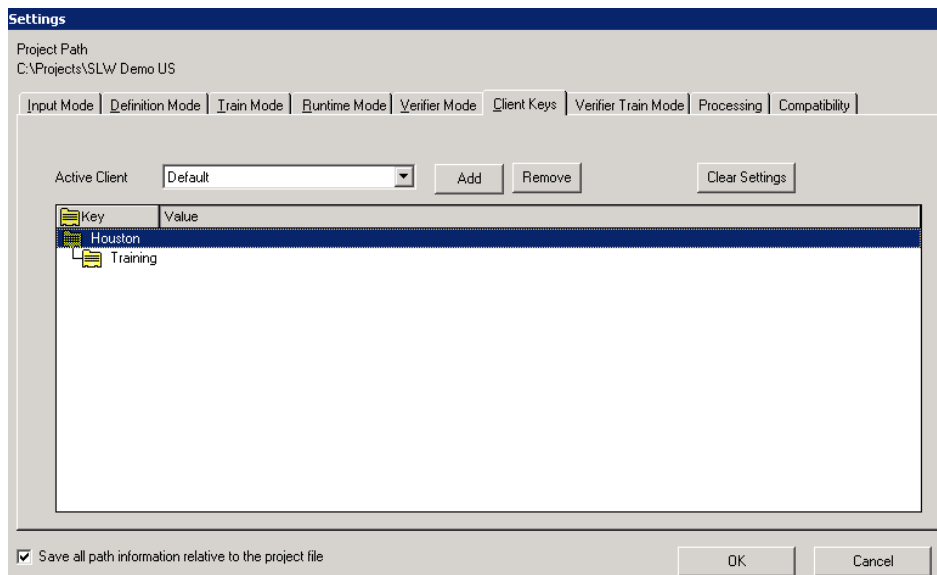


Figure 4-43: Client keys tree.

##### 4.9.6.2. Managing Clients

By default, each WebCenter Forms Recognition project contains a standard client named Default. If your project is to be used for more than one client, you can create additional clients.

**To create a client:**

- 1) In the *Client Keys* tab, click *Add*.

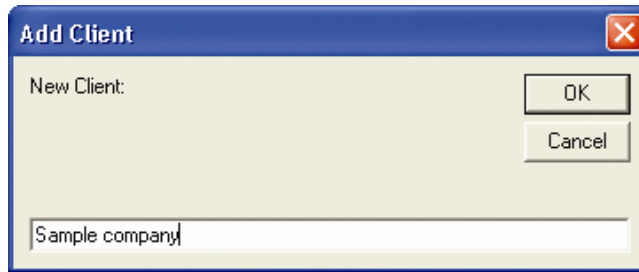


Figure 4-44: Adding a client

- 2) On the *Add Client* dialog box, enter a name for the client and confirm with *OK*.

To select a client:

- In the *Client Keys* tab, from the *Active Client* list box, select an entry.

**To delete a client:**

- In the *Client Keys* tab, click *Remove*. The active client is deleted when you confirm the following message box.

**To delete all clients including the associated keys:**

- In the *Client Keys* tab, click the *Clear Settings* button. The entire settings tree is deleted when you confirm the following message box.

**4.9.6.3. Managing Keys**

You can create a hierarchical structure of keys, consisting of root nodes and child nodes. The same keys are used for all clients.

**To add a root node to the hierarchy of keys:**

- 1) In the *Client Keys* tab, right click within the area in the center of the dialog box.
- 2) From the shortcut menu, select *Add Root Key*. The *Add New Key* dialog box is displayed.

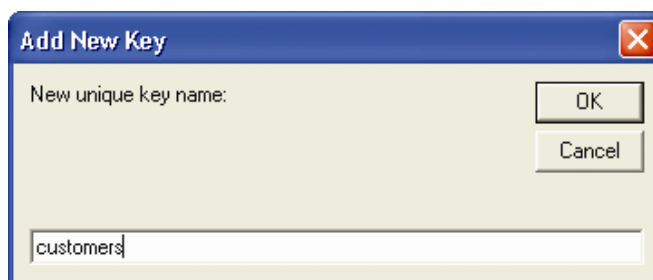


Figure 4-45: Adding a key

- 3) Enter a unique name for the key and confirm with *OK*. The new key is inserted at the top level of the hierarchy.

**To add a child node to the hierarchy of keys:**

- 1) In the *Client Keys* tab, right click an existing key.

- 2) From the shortcut menu, select *Add Child Key*. The *Add New Key* dialog box is displayed.
- 3) Enter a unique name for the key and confirm with *OK*. The new key is inserted below the parent key.

**To remove a key:**

- 1) In the *Client Keys* tab, right click an existing key.
- 2) From the shortcut menu, select *Remove Key*. The key is deleted when you confirm the following message box.

#### 4.9.6.4. Managing Values

An individual value can be specified for each defined key from the tree. This value is going to be used only for the currently selected “Active Client”.

- 1) In the *Client Keys* tab, the *Active Client* drop down list box, select the desired “Active Client” name.
- 2) Click on the key to modify.
- 3) In the “Value” field, type the value that needs to be associated with the key name specified in the “Key” column.

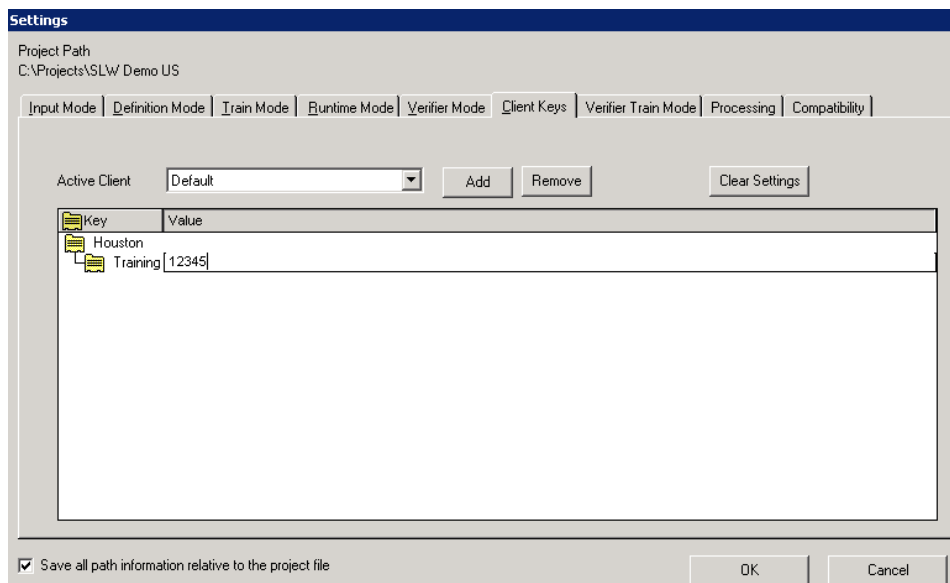


Figure 4-46: Client keys tree with key-value pairs


#### 4.9.7. Making Projects Portable

The project file is portable so that existing applications can easily be reused. Designer can either work with absolute or with relative references to the following directories:

- Export directory
- Learnset directory
- Batch root directory
- Image file directory

By default, absolute references are used. For full portability, you should use relative references.

**To enable relative references:**

- 1) From the *Options* menu, select *Settings*. Alternatively, in the toolbar, click  *Show settings*.
- 2) In the *Settings* dialog box, on the Verifier mode tab, check the *Save all path information...* check box.

To transfer a project, copy the \*.sdp project file and – if applicable – the documents from the Learnset to the target directory. With absolute references, adjust the path settings in the project options. (See section [WORKING WITH MODES](#))

## 4.10 Working with Documents

### 4.10.1. What are Workdocs?

Starting with the image or file that is provided as input, WebCenter Forms Recognition processes the document and writes the processing result to a structure called a Workdoc.

The Workdoc is created during the initial OCR of an image or the initial filtering of a file. The OCR also includes a layout analysis that identifies words and blocks within the document. All subsequent processing steps use the Workdoc and not the original file. While the document is being processed, each step adds its results to the Workdoc. This includes:

- The document's text
- The location of words
- The geometrical relationship between words
- The classification results
- The fields for data extraction
- The candidates for field contents
- The selected correct candidate for each field
- The fields' validation status.

The Workdoc is saved persistently as a \*.wdc file only if it is part of a batch. Otherwise, it only exists temporarily. It is saved in a compressed format, which allows for greater disk space.

The process of analyzing a document is complete when the Workdoc is assigned to a document class and all the fields for extraction are filled and valid.

### 4.10.2. Viewing Documents

When you have selected your document input, the corresponding set of documents is displayed in the Designer viewer. The viewer displays the image or file, and structures in the Workdoc.

Document display is available in:

- Document Selection Mode (Section [DOCUMENT SELECTION MODE](#))

- Definition Mode (Section [DEFINITION MODE](#))
- Train Mode (Section [TRAIN MODE](#))
- Verifier Train Mode (Section [VERIFIER TRAIN MODE](#))
- Runtime Mode (Section [RUNTIME MODE](#))
- Verifier Design Mode (Section [VERIFIER DESIGN MODE](#))
- Verifier Test Mode (Section [VERIFIER TEST MODE](#)).

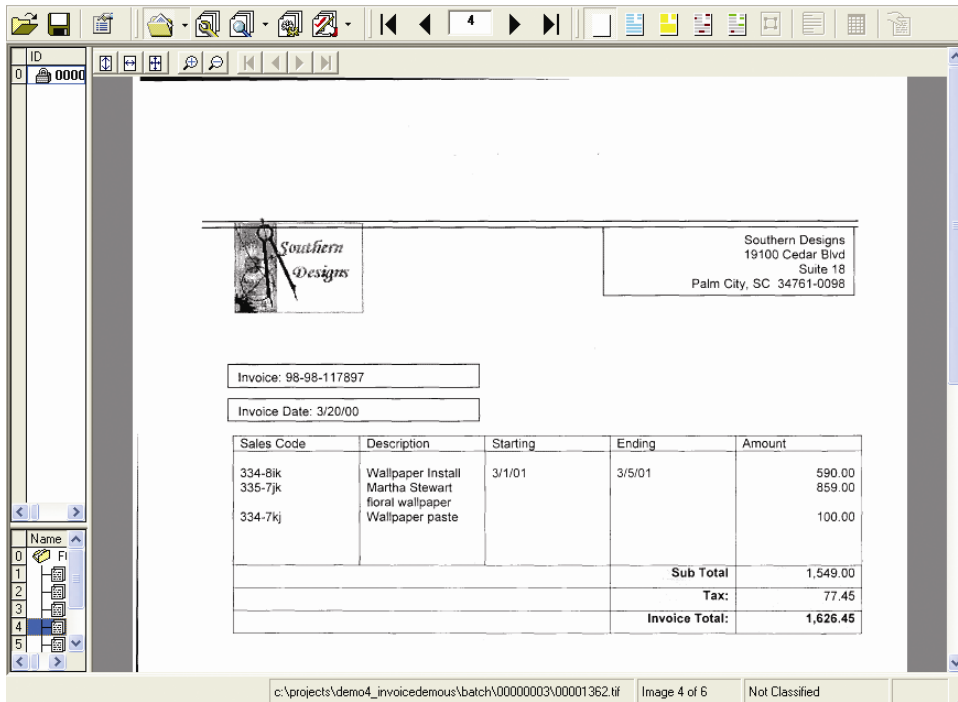


Figure 4-47: The user interface of the Designer viewer

The application’s status bar displays the total number of documents in the current set and the serial number of the current document.

A navigation bar in the toolbar enables you to navigate through the document set.

Button	Description
	Moves to the first document.
	Moves to the previous document.
	Moves to the next document.
	Moves to the last document.
	Displays the serial number of the current document. To move to a particular document, type its number and press ENTER.

Table 4-5: Controls in the navigation toolbar

The navigation bar is available in:

- Document Selection Mode (Section [DOCUMENT SELECTION MODE](#))
- Definition Mode (Section [DEFINITION MODE](#))
- Train Mode (Section [TRAIN MODE](#))
- Verifier Train Mode (Section [VERIFIER TRAIN MODE](#))
- Verifier Design Mode (Section [VERIFIER DESIGN MODE](#))
- Verifier Test Mode (Section [VERIFIER TEST MODE](#)).

The viewer's toolbar provides access to zooming functions and enables you to navigate through multi-page documents.








Button	Description
	Adjusts the document display to the height of the viewer window.
	Adjusts the document display to the width of the viewer window.
	Adjusts the document display to the height or the width of the viewer window so that the entire page is displayed.
	Zooms in.
	Zooms out.
	Moves to the next page of a multi-paged document.
	Moves to the previous page of a multi-paged document.

Table 4-6: Controls in the viewer toolbar

The Viewer toolbar is available in:

- Document Selection Mode (Section [DOCUMENT SELECTION MODE](#))
- Definition Mode (Section [DEFINITION MODE](#))
- Train Mode (Section [TRAIN MODE](#))
- Verifier Train Mode (Section [VERIFIER TRAIN MODE](#))
- Verifier Test Mode (Section [VERIFIER TEST MODE](#)).

In Definition Mode, you can either display the scanned image or the Workdoc. By default, the Workdoc is displayed. To create reading zones on a document, you need to work with the image. Designer automatically displays the image when you start creating zones.

Use the View menu to between image display and Workdoc display:

- Select Show Page to display the image.
- Select Show Workdoc to display the Workdoc.

You may need this to change the recognition settings for the entire document. (See section [PAGE-LEVEL SETTINGS](#))







If Workdocs are visible in the field list of Design Mode, the text is dimmed. When results of the execution are displayed the foreground is black.

### 4.10.3. Processing Documents

The process selection group enables you to process single or multiple documents. Depending on the state of the program, processing may include:

- OCR (if there is no Workdoc for the current document)
- Classification (if there are classes)
- Data extraction (if there are fields; in Runtime or Train Mode, the Fields tab must also be active)
- Document Export (only available in definition and Runtime Mode)
- Verification (only available in Verifier Test Mode).

The processing status is shown in the status bar. Note that the initial OCR may take a while.

Button	Description
	Processes the current document. The button's drop-down menu enables or disables the debug mode. This button is available in Definition Mode (section <a href="#">DEFINITION MODE</a> ) and Train Mode (section <a href="#">TRAIN MODE</a> ).
	Processes the next document. This button is available in Definition Mode. (section <a href="#">DEFINITION MODE</a> ).
	Processes documents in the current set starting with the current one. This button is available in Definition Mode (section <a href="#">DEFINITION MODE</a> ) and Runtime Mode (section <a href="#">RUNTIME MODE</a> ).
	Pauses the processing of documents. This button is available in Definition Mode (section <a href="#">DEFINITION MODE</a> ) and Runtime Mode (section <a href="#">RUNTIME MODE</a> ).
	Stops the processing of documents. Clears any results from previous runs. This button is available in Definition Mode (section <a href="#">DEFINITION MODE</a> ) and Runtime Mode (section <a href="#">RUNTIME MODE</a> ).
	Adds the current document to the Learnset of the selected class. This button is available in Verifier Train Mode (section <a href="#">VERIFIER TRAIN MODE</a> ).




Button	Description
	Starts the learning for the current project. This button is available in Train Mode (section <a href="#">TRAIN MODE</a> ) Verifier Train Mode (section <a href="#">VERIFIER TRAIN MODE</a> ) and Definition Mode (section <a href="#">DEFINITION MODE</a> ).

Table 4-7: Controls to process documents

#### 4.10.4. Learning

Learning is involved in classification and in data extraction. For classification, learning requires that you provide examples that indicate which documents belong to a specified class. For extraction, learning requires that you select the correct data from a set of extraction candidates.

Designer takes the examples you provide and creates Learnsets from them. It then learns to produce the correct output from input that is similar to the Learnset.

Use the Learning toolbar to create Learnsets and trigger the learning.

#### 4.10.5. Highlighting Processing Results

The document highlighting toolbar lets you select highlighting options for the displayed documents. These buttons act as switches.

Highlighting visualizes structures that have been created during document processing. Therefore, this feature is only available when a Workdoc exists. In addition, field and candidate highlighting is only available when fields are defined.






Button	Description	Menu command
	Displays the document without highlighting.	View - Highlight Nothing
	Highlights all words in the document in turquoise. Words are a result of the OCR. Each character is stored together with its position. To view the OCR result, point to a word with the mouse. The recognized word is displayed as a tooltip.	View - Highlight Words
	Highlights all blocks in the document in yellow. Blocks are groups of words identified in Definition Mode by their geometrical relationship.	View - Highlight Blocks
	Highlights all candidates for the currently selected field in maroon. A candidate is a possible value for a field identified in the processing step. To view the weight of the candidate, point to it with the mouse. The weight is displayed as a tooltip.	View - Highlight Candidates
	Highlights all fields in the document. If the field is valid it is highlighted in green, otherwise it is highlighted in red. To view the name of the field, point to it with the mouse. The name is displayed as a tooltip.	View - Highlight Fields
n/a	Only relevant in conjunction with table extraction or programming using the Workdoc API. Highlights all words where Visible = TRUE.	View - Highlight Checked Words

Table 4-8: Buttons and menu commands for document highlighting

If table analysis is used, only the best table candidate can be highlighted. Table highlighting must not be triggered using toolbar buttons but is active by default. The behavior varies with the different modes:

In Definition Mode (Section [DEFINITION MODE](#)) table elements are highlighted during the definition process, and the entire table is highlighted when a document has been processed. Valid elements are highlighted in green. Invalid elements are red.

In Verifier Test Mode (Section [VERIFIER TEST MODE](#)) the extracted table is available and can be used to control which elements of the document table are highlighted. Valid elements are highlighted in green. Invalid elements are red.

## 5 Setting Up the Classification

Setting up the document classification involves:

- Creating a classification scheme
- Defining classification methods
- Setting parameters for these methods
- Testing
- Optimizing

The evaluation of the target class is fairly complex. A document can only be assigned to a class if a number of confidence values exceed predefined thresholds. Although you can influence the threshold values and the evaluation algorithms, the default settings are fine for most cases.

For further instructions on classification evaluation, refer to [12 ADVANCED EVALUATION SETTINGS](#).

For now, the following background knowledge is sufficient:

- The confidence values indicate the degree of similarity between the properties of a document and the properties of a class.
- A document can be classified if its confidence value with respect to a given class exceeds a predefined threshold. By default, this threshold is 70%.
- In general, there is only one target class per document. If several classes have a high confidence with respect to a document, it must be possible to distinguish reliably. Therefore, a certain difference in confidence between the winning class and the second-best competitor is required as well. By default, this so-called distance is 20 percent.
- If multiple classification methods are applied to a given class, WebCenter Forms Recognition will compute a combined result. Normally, classification is based on this combined result.

It is also necessary to ensure that all documents to be processed within the same project have been scanned to the same resolution. If Brainware Table Extraction is to be used the resolution of all documents must be 300 dpi.

### 5.1 Advanced Classification

The Classification Matrix functionality can be configured in Designer for Verifier users. This feature will suggest several probable classes if the class result could not be determined by 100 %. If more than one class is in the list, the class entry is supplemented by a value representing the determination probability. You will find the class with the highest probability on the top of the list.

You can make this functionality available in Verifier by checking *Enable advanced classification in Verifier* on the *Verifier Mode* tab in the *Settings* window under the *Options* menu.

## 5.2 Preparing Sample and Test Documents

Before you begin designing your application, you need to ensure that you have enough documents to set up your project and to test it.

- Create separate directories for sample documents and test documents.
- For each class, create a subdirectory within the sample directory. Each subdirectory should only contain documents that belong to the corresponding class.
- Create separate batches for each subdirectory. The batches will be required as document input. When creating the batches, only the import and the OCR step should be carried out. For further instructions, please refer to the WebCenter Forms Recognition Runtime Server User's Guide.

## 5.3 Creating the Project

To set up a new project:

- 1) Start Designer.
- 2) Set the project options as follows:
  - Make sure that manually trained documents can be added to the Learnset.
  - Specify or create a Definition Mode or Runtime Mode export directory. The software needs this directory to save processing results.
  - Specify or create a Train Mode base directory. The software will write copies of your sample documents to this directory.
  - Specify the batch root directory. The software will take the document input from there.
  - Set all other options as desired.
- 3) Save the project file.

## 5.4 Creating Classes

Designer enables you to create multi-level hierarchical classification schemes by means of base classes and derived classes. Base classes constitute the highest level of a classification tree. Derived classes are specializations of base classes.

When you create a class hierarchy, consider that:

- Derived classes should be a true specialization of their parent class.
- Classes that compete with each other should be disjoint.
- Classes that compete with each other should be based on a similar level of abstraction.

A weird classification tree such as the one shown in [FIGURE 5-1](#) will not yield satisfactory results. This classification tree is insufficient because the Invoice class includes two derived classes that are not parallel: Invoices from Travel Agencies and Invoices March 2000. It's conceivable that Invoices March 2000 would also include invoices that came from travel agencies and therefore would also be part of the Travel Agencies class.

A better classification tree might consist of Invoices from Travel Agencies and Invoices from Mechanics, or Invoices March 2000 and Invoices April 2000.

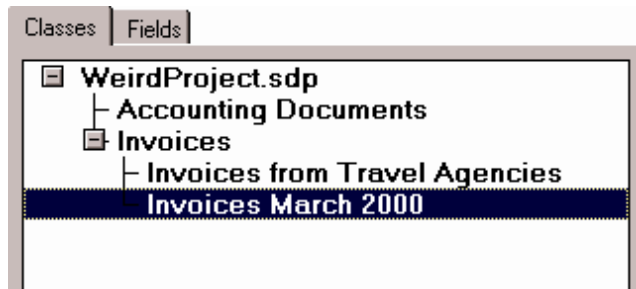


Figure 5-1: Example for a weird classification tree

A working classification tree must contain at least two classes, but four or more classes are better.

### Task Prerequisites

- The program is in Definition Mode.

The panel on the left side of the window displays the Classes tab in the foreground.

### Creating a Classification Tree

To create a classification tree:

- 1) To add a base class to your project, right click any entry in the *Classes* tab. If your project does not yet contain classes, the only available entry is the root node (either labeled as <new project> or with the project file name.)

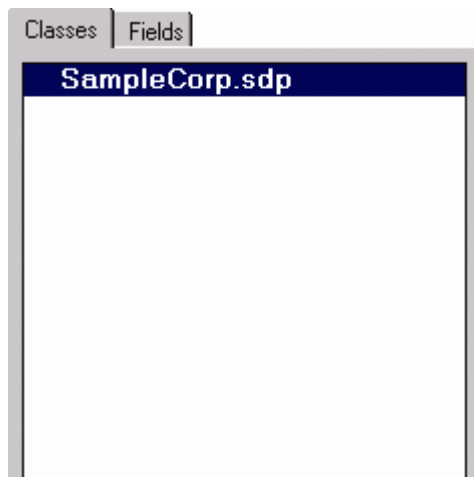


Figure 5-2: Initial state of the Classes tab

- 2) On the shortcut menu, select *Insert Base DocClass*.
- 3) Enter a class name. Valid names consist of alphanumeric characters without spaces or special characters. Later, you can define a different display name in the Document Class properties. (See section [CUSTOM CLASS NAMES](#))
- 4) Click *OK* to confirm. The new class is inserted below the root class in alphabetical order.
- 5) Repeat step 1 to step 4 until all base classes are created.

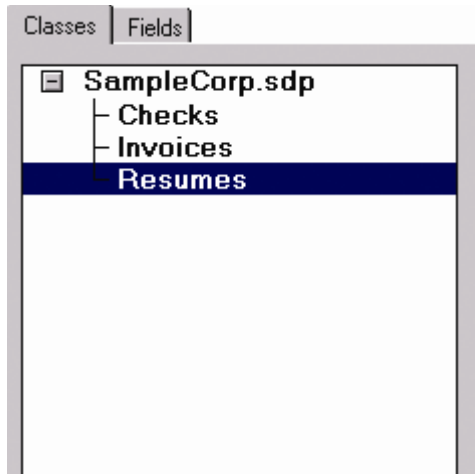


Figure 5-3: Classes tab with base classes

- 6) To add a derived class, right click the prospective parent class. The parent may either be a base class or a derived class itself.
- 7) On the shortcut menu, select *Insert Derived DocClass*.
- 8) Enter a class name and confirm. The new class is inserted below the parent class in alphabetical order.
- 9) Repeat step 5 to step 8 until all derived classes are created.

### 5.4.1. Custom Class Names

The custom class names can be defined for each WebCenter Forms Recognition document class in Designer application using *Display Name* edit box available on *Document Class* property page of selected document class's settings:

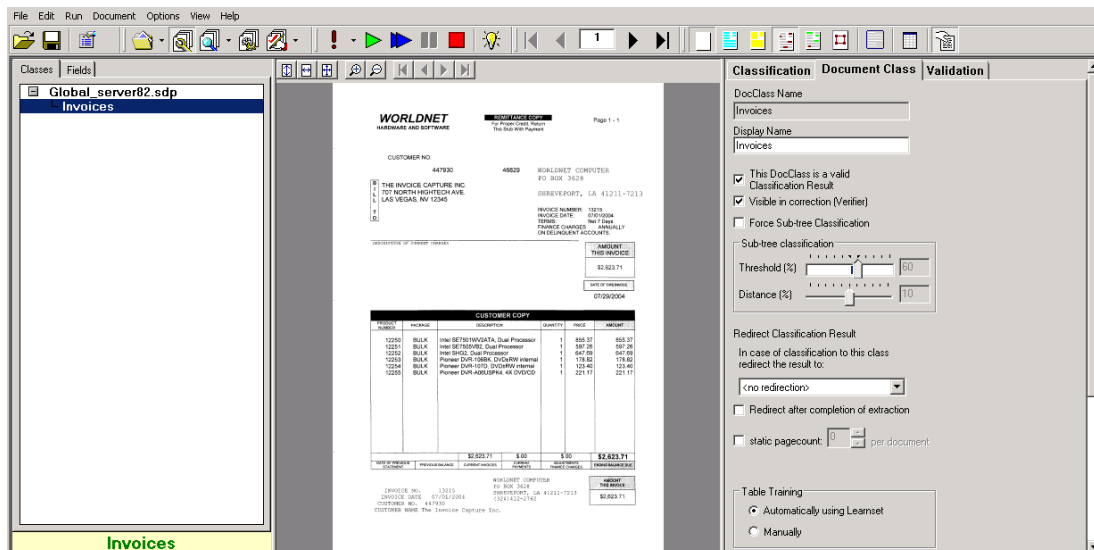


Figure 5-4: Document class property page

Usage of custom class names can be always switched off and on using the project node's settings of the Designer application, *Advanced* tab, *Show custom class names in class tree views* check-box option. By default this option is enabled but the Designer administrator can always switch off (save the project afterwards):

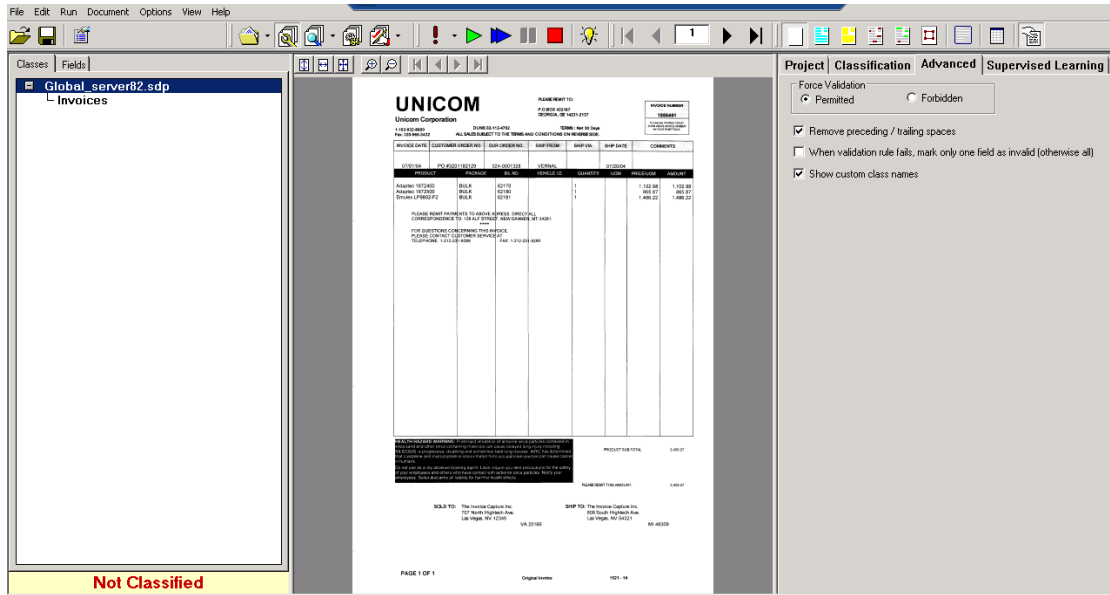


Figure 5-5: Settings for usage of custom class names

Once the new custom class name(s) are defined, this is going to affect both, Verifier and Designer applications, including the list view items and the status bar indicator of the currently selected class:

1. “Fields” view of the Definition mode of the Designer application; including both status bar indicators of the document class of the opened document it is currently classified to.

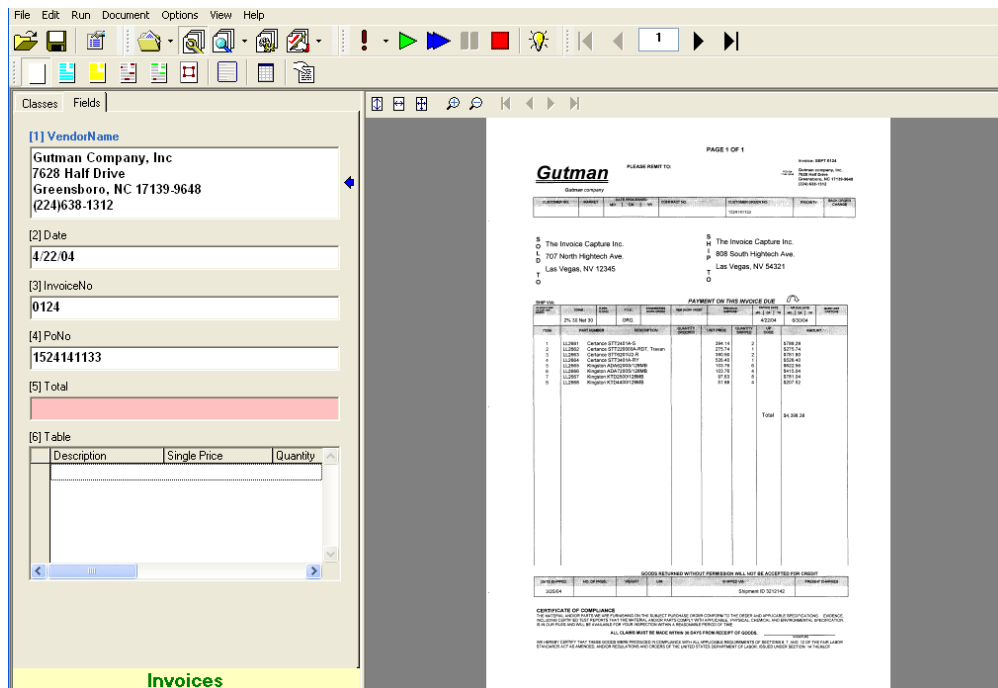


Figure 5-6: Definition Mode – Fields view

- Normal “Train” and “Verifier Test” modes of the Designer application, the status bar indicator (at the right) of the document class of the opened document it is currently classified to.

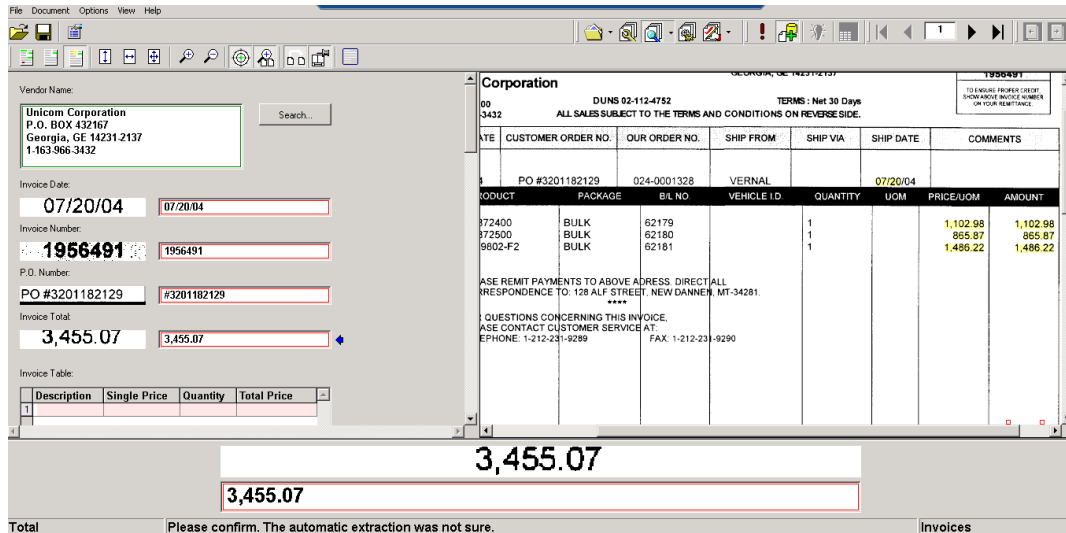


Figure 5-7: Train and Verifier Train Modes – status bar indicator

- Document verification mode of the Verifier application, the status bar indicator (at the right) of the document class of the currently validated document it is classified to.

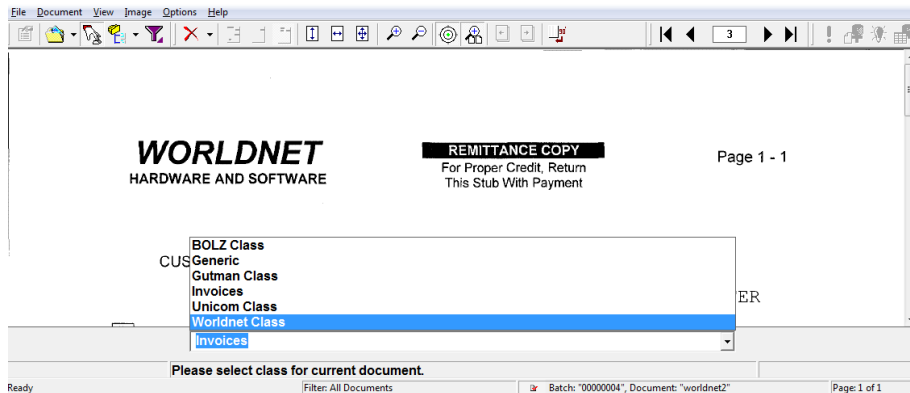


Figure 5-8: Document verification mode

- Manual Classification view of the Verifier application, including the list items of the available classes' list and the currently selected class (see the screenshot below).
- Global Learnset Browsing mode of Learnset Manager Tool (can be launched from Verifier application; see the screenshot below).



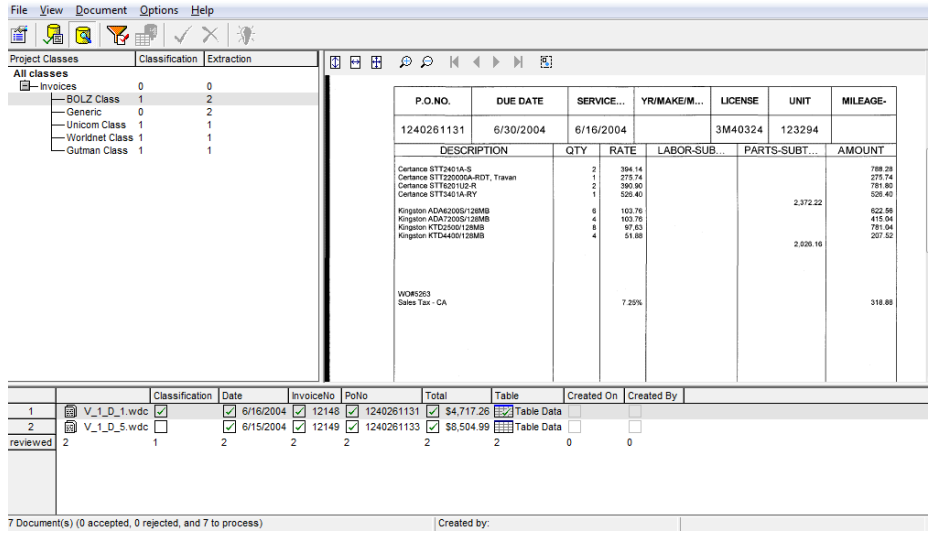


Figure 5-9: Learnset Manager tool – Global Learnset Browsing mode

The other class view modes of WebCenter Forms Recognition applications remain unaffected though. For example:

- Learnset mode of Designer
- Forms Design mode of Designer
- Normal Train mode of Designer
- Runtime mode of Designer
- Accumulated Documents Processing mode of the Learnset Manager tool

### Usage

This provides the end-user with user-friendly object names as a replacement to sometimes non-readable system names for classes in WebCenter Forms Recognition.

The feature can be especially important for some supervised learning configurations and usage of the Learnset Manager tool, in case the reviewed system class names are not readable at all (for example, equal to the numeric ID field from a customer vendor database).

## 5.5 Editing Classes

WebCenter Forms Recognition Designer provides commands for editing classes. Be careful when using them since you might lose some of the settings you have already made.

### Task Prerequisites

The prerequisites for this task are:

- The program is in Definition Mode.
- The panel on the left side of the window displays the Classes tab in the foreground.
- There must be classes to edit.

### Supported Operations

The following operations are supported:

- The *Delete DocClass* command from the shortcut menu of a class deletes the class including all associated settings. In this case, you lose all trained knowledge for the project. Repeat the training.

*Warning: Do not reuse the name of a deleted, learned Document Class in the same project. Doing so may cause inconsistencies within the project. When you create a Document Class, a directory with the name of the Document Class is created that contains the Learnset documents for this Document Class. This directory will remain when you delete the Document Class from the class hierarchy. If you create a new Document Class with this name, it appears as if Learnset documents are available when they are not.*

- The *Rename DocClass* command from the shortcut menu of a class renames the class.  
In this case you lose an existing Learnset for the class and all trained knowledge for the project. Create a new Learnset for the class and repeat the training.
- Within a classification tree, you can move classes using Drag & Drop. In this case, you lose all trained knowledge for the project. Repeat the training. You may also have used inherited field properties. Check whether your settings are still consistent and meaningful.

## 5.6 Selecting Classification Methods

Once you have created classes, specify the classification methods for each class.

You will probably try a single classification method first. Brainware is the recommended primary method. If you need to improve the results, you can use complementary methods. Classification methods can be combined freely.

### Task Prerequisites

The prerequisites for this task are:

- The program is in Definition Mode.
- The panel on the left side of the window displays the Classes tab in the foreground.
- On the right side of the window, the Classification Editor is selected.

### Selecting Classification Methods

To select one or more classification methods:

- 1) In the *Classes* tab on the left side of the window, select a class.
- 2) In the Classification Editor on the right side of the window, check the methods that you want to use for the selected class.  
Clear the corresponding check mark to exclude a previously selected classification method.
- 3) Repeat Step 1 and Step 2 until all classes have classification methods assigned.

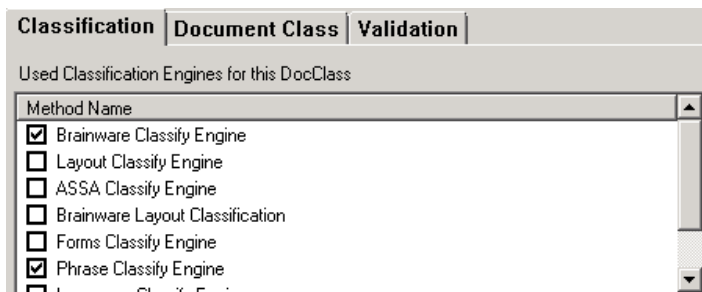


Figure 5-10: Classification tab with classification methods per class.

## 5.7 Configuring Classification

For Brainware classification, you do not have to define a complicated set of rules. All you have to do is to take a couple of sample documents and to manually assign them to the respective target classes. WebCenter Forms Recognition is then able to learn the way you classify the documents.

### 5.7.1. Creating Learnsets

The set of sample documents you need to provide for classification is called the Classification Learnset. Carefully select the samples for each class, because the quality of your Learnset is crucial for the success of the classification.

When you select the sample documents for the Learnsets, consider the following:

- You need a separate Learnset for each class.
- You need samples that truly represent the topic you want to cover.
- Avoid documents containing multiple topics, since these documents cannot be classified exclusively to one class.
- In general, do not use the same document in more than one Learnset.
- Avoid using similar documents for different classes.
- Use only documents with good OCR results for the Learnset. When in doubt, highlight the OCR results to review the documents. Never use handwritten documents.
- You need at least five sample documents per class. The maximum number of samples per class is limited by memory.
- The number of samples needed to create a good Learnset depends on the classification task. The broader a field a class is supposed to cover, the more samples are required. For example, you will need more samples to cover the topic “politics” than you will need for invoices of a company.
- Increase the number of samples if classes are very similar to each other.
- Ideally, you should have an equal number of samples for each class. A factor of two is not a problem, a factor of ten certainly is.

#### The prerequisite for this task is:

- Your project settings must allow manual addition of documents to the Learnset.

#### To create Learnsets:

- 1) Switch to Document Selection Mode.
- 2) Select the first document from the batch that contains the documents for your Learnset.
- 3) Switch to Train Mode. On the left side of the window, the *Classes* tab must be in the foreground.

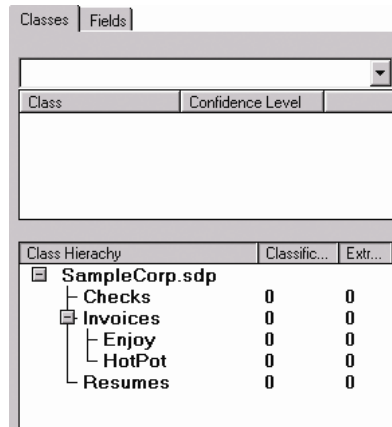



Figure 5-11: Empty Learnsets

The list in the lower section of the *Classes* tab indicates the number of documents in the Learnsets.

- 4) If required, use the navigation bar in the toolbar to browse to a document that you want to add to a Learnset.
- 5) In the *Classes* tab, select a target class from the list box in the upper section.
- 6) In the toolbar, click  *Add to Learnset* for adding the current document to the classification Learnset of the selected class. WebCenter Forms Recognition writes a copy of the document to a protected directory structure that is reserved for Learnsets. The entry in the Classification Learnset column in the list in the lower section of the *Classes* tab indicates that a document has been added.

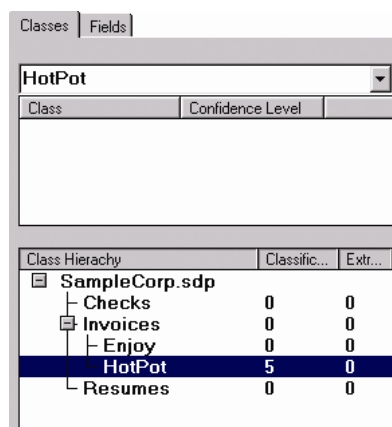


Figure 5-12: Learnset with documents

- 7) Repeat Step 4 through Step 6 until you have assigned enough samples from the current document set.
- 8) To add more documents from other document sets, start again with Step 1.

9) Save the project. This saves all changes to the Learnset and updates the project.

### 5.7.2. Encrypting a Learnset

If you wish to use confidential documents in a Learnset, you can hide them from view by Encrypting the Learnset. You can use an encrypted Learnset to train a project; however, you cannot read the workdoc files or the associated documents. When you use Encrypt Learnset, WebCenter Forms Recognition encrypts an existing Learnset and deletes all document files (images and CI documents).

*Note:* Be sure to make a copy of the Learnset and use it for the encryption before you proceed. You cannot undo this task.

Task Prerequisite

**The prerequisite for this task is:**

- The program is in Definition Mode or Train Mode.

Encrypting a Learnset

**To encrypt a Learnset:**

- 1) Select *Options, Encrypt Learnset*. A warning message appears.
- 2) Click **Yes** to encrypt the Learnset.

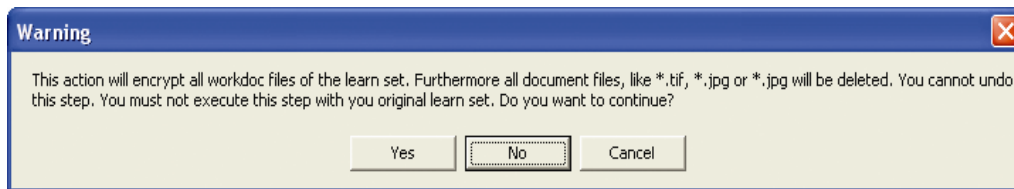


Figure 5-13: Encrypt Learnset warning box

All documents are accessible in the Learnset. After encrypting a Learnset, you can still add documents to the Learnset. You can also repeat the encryption step to encrypt the new documents.

### 5.7.3. Security Extensions for Learnset Encryption

#### 5.7.3.1. Description

The encryption of WebCenter Forms Recognition Learnsets can be activated for any project in the Designer application. As an example, let us apply the encryption feature for the following demo project with a few documents in its Learnset:

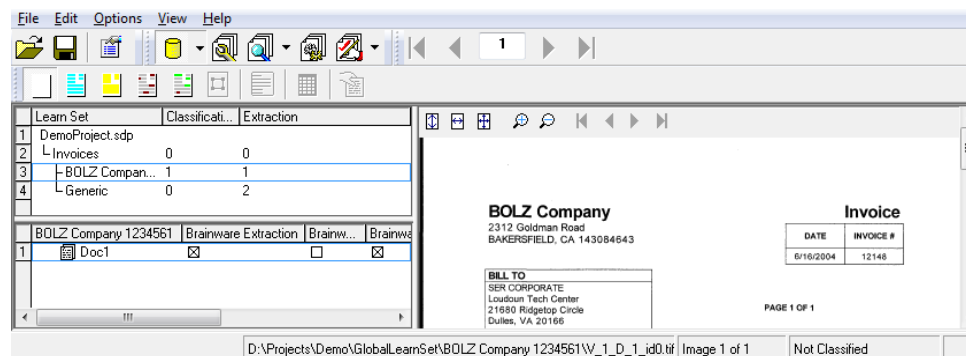


Figure 5-14: Demo project for encryption

In order to activate the encryption, select *Options* menu, and then click on *Encrypt Learnset* menu item. The system will require you to confirm twice:

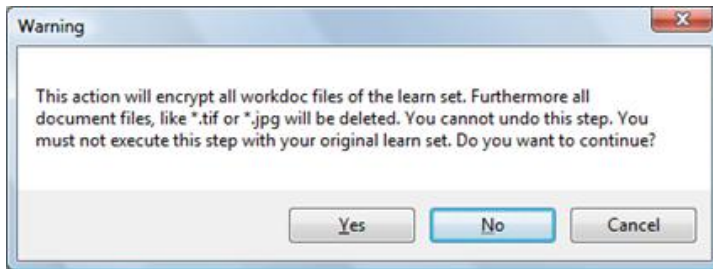


Figure 5-15: Confirmation dialog box for encryption

Click on *Yes* to start the encryption process. As soon as the encryption process has been finished the system is supposed to show a message box like this:



Figure 5-16: Message box for successful encryption

After that, reviewing of the Learnset in Designer should no longer show the images associated with Learnset documents:

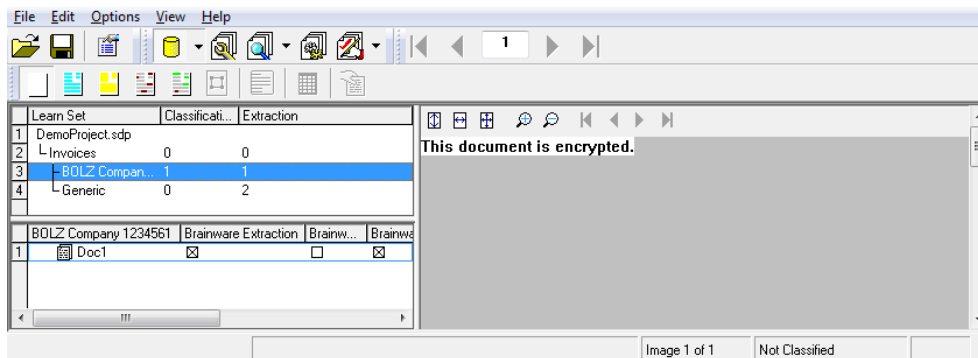


Figure 5-17: New view for encrypted document

The result of the encryption process affects only the content of the Learnset directories, where all WorkDoc files are going to be encrypted, as well as, all the image files – removed from the disk. Make sure to create a backup of the Learnset before applying the encryption process.

In this connection, the encrypted documents can be still used by the WebCenter Forms Recognition applications in terms of re-training and classification/extraction while they cannot be reviewed by the user.

Before the encryption process is applied the Learnset directory of a document class looks like as shown before and includes original image files and the WorkDoc document in a normal

non-encrypted form (internally the WorkDoc contains a couple of ZIP streams that, in principle, can be unpacked and reviewed):

Name	Date modified	Type	Size
bfe.ptb	2/27/2008 12:52 AM	PTB File	33 KB
bfe.xtr	2/27/2008 12:52 AM	XTR File	17,064 KB
bte.ptb	2/25/2008 10:46 PM	PTB File	0 KB
bte.xtr	2/25/2008 10:46 PM	XTR File	1 KB
V_1_D_2.wdc	2/27/2008 12:52 AM	WDC File	32 KB
V_1_D_2_id0.tif	7/30/2004 3:17 PM	TIFF Image	84 KB
V_1_D_3.wdc	2/27/2008 12:52 AM	WDC File	37 KB
V_1_D_3_id0.tif	7/30/2004 3:30 PM	TIFF Image	89 KB

Figure 5-18: Learnset directory

Screenshot above shows Learnset directory content sample.

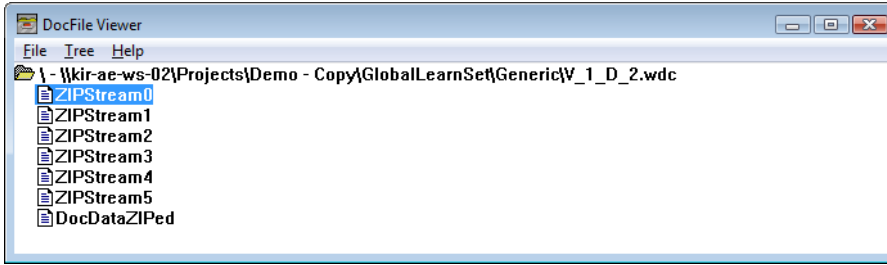


Figure 5-19: Non-encrypted WorkDoc structure

Screenshot above shows non-encrypted WorkDoc structure.

As soon as the encryption has been applied, the Learnset directories do not contain any images and the internal WorkDoc structure cannot be unzipped any longer, being modified via an encryption mechanism:

Name	Date modified	Type	Size
bfe.ptb	2/27/2008 12:52 AM	PTB File	33 KB
bfe.xtr	2/27/2008 12:52 AM	XTR File	17,064 KB
bte.ptb	2/25/2008 10:46 PM	PTB File	0 KB
bte.xtr	2/25/2008 10:46 PM	XTR File	1 KB
V_1_D_2.wdc	4/28/2008 12:21 PM	WDC File	28 KB
V_1_D_3.wdc	4/28/2008 12:21 PM	WDC File	33 KB

Figure 5-20: Learnset directory after encryption

Screenshot above shows Learnset directory content sample after the encryption process has taken place.

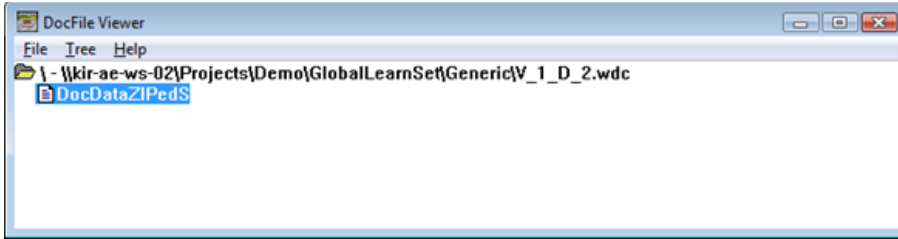


Figure 5-21: Encrypted WorkDoc – Internal structure

Screenshot above shows the internal structure of an encrypted WorkDoc document.

*Note: An encrypted Learnset and even particular encrypted class of the Learnset can be further extended by new documents and relearned cumulatively. However, the OCR process, of course, cannot be reapplied any longer, because the original image data is unavailable.*

The example below shows that after adding one more document to the Learnset encrypted previously, the system represents the new documents as usual:

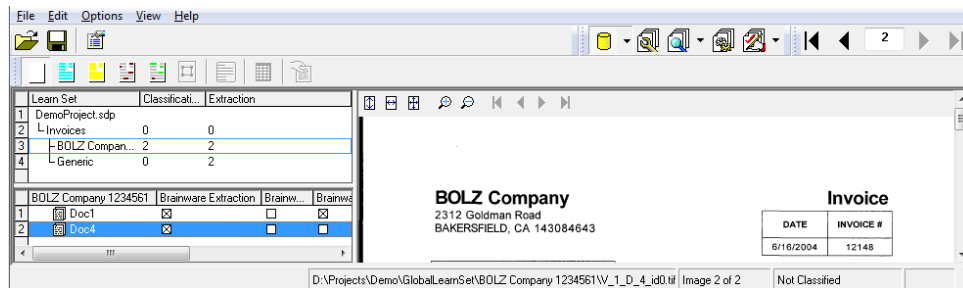


Figure 5-22: New document in encrypted Learnset

Also, all the previously encrypted documents remain hidden:

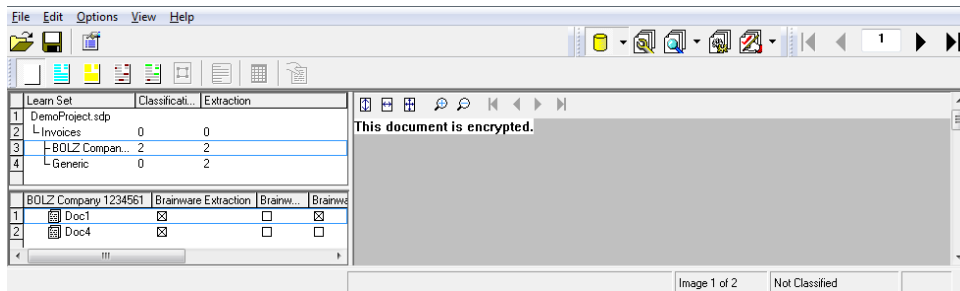


Figure 5-23: View of a previously encrypted document

When desired, the encryption of the Learnset documents can be applied with a redistributor-specific encryption key. The custom encryption key can be defined in Windows Registry as a string value of “EncryptLearnSetKey” variable under “HKEY\_LOCAL\_MACHINE \ Software \ Oracle \ Cedar”.

*Note: On a 64 bit OS the location for the registry key is the following: “HKEY\_LOCAL\_MACHINE \ Software \Wow6432Node \ Oracle \ Cedar”.*

The string content of the variable should not exceed 255 characters. If the custom encryption key is undefined when the encryption process is being started, the system is going to use an internal default key. The custom encryption key can be used in case it is required to get access to the encrypted documents later, e.g., from within the WebCenter Forms



Recognition’s custom script. For this purpose, the developer can call “EnableAccess” method of the SCBCdrWorkdoc interface using the custom encryption key as a parameter.

*Important Note: In case a custom encryption key is used to create the encrypted Learnsets, setting up the "HKEY\_LOCAL\_MACHINE \ Software \ (Wow6432Node ) \ Oracle \ Cedar \ EncryptLearnSetKey" on the customer site where the encrypted Learnset is used for extraction is NOT required (if re-learning of the entire project is going to be applied).*

**5.7.3.2. Usage**

Encryption of Learnsets can be used for secure redistribution of WebCenter Forms Recognition Learnsets that contain critical customer data that cannot be given from one customer to another in open form.

The feature can be used for secure encryption of any WebCenter Forms Recognition Learnsets, but it is basically supposed to be applied to so-called generic Learnsets. When used for vendor Learnsets, Professional Services should take into account that the class names are not encrypted and remain as they are. In other words, if the class names (or anything else, like Associative Database Search (ADS) fields, etc.) contain any customer information that has to be hidden, this has to be done manually (for example, via renaming of the redistributed classes or via neutralizing the database content of the ADS fields).

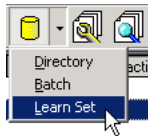
**5.7.4. Editing Learnsets**

You can add documents to existing Learnsets at any time using the method described in section [CREATING LEARNSETS](#).

Occasionally, a document is added to a Learnset by mistake, or it is assigned to the wrong class.

**To review the Learnset:**

- 1) Select the Learnset as document input.



The program automatically switches to Document Selection Mode. A list view of classes is displayed on the upper left.

- 2) Double click one of the classes to display the documents in its Learnset in a second list view below.

Learn Set	Classificati...	Extraction
1	Global.sdp	
2	└ Invoices	5 0

Invoices	Brainw...	Layout ...	ASSA ...	Brainware Layout Classification
1	Doc1	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
2	Doc2	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
3	Doc3	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
4	Doc4	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
5	Doc5	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

Figure 5-24: The classification Learnset for the Checks class

- 3) Using the list view of documents, you can now:
- Browse through the documents – just click one of the documents to display it,
  - Delete a document from the Learnset: right click the document and select Remove from Learnset from the shortcut menu,
  - Move a document to the Learnset of a different class using drag & drop
  - Exclude a document from the learning for specified classification methods – just clear the check mark in the corresponding column. This method is also handy for temporarily removing a document from the Learnset.

#### 5.7.4.1. Manually Adding New Documents to a Learnset

Although you can add new documents to a Learnset at any time, you may find that the results are inconsistent when you add new documents to a Learnset and then save the project without learning. This only happens when you use a project that has a history of learning, as the classification engine has a knowledge base from which to extract information.

Without proper learning, the information from the new documents will not be included in the classification. To remedy this, you can change a Learnset by manually adding a document to it and then changing the working mode. The documents will automatically go through the learning process once the mode changes.

#### Task Prerequisites

The prerequisite for this task is:

- Your project settings must allow manual addition of documents to the Learnset.

Manually Adding a Document to a Learnset

**To manually add a document to a Learnset:**

- 1) Switch to *Document Selection Mode* and select a document from the batch.
- 2) Switch to *Train Mode*. On the left side of the window, the *Classes* tab must be in the foreground.

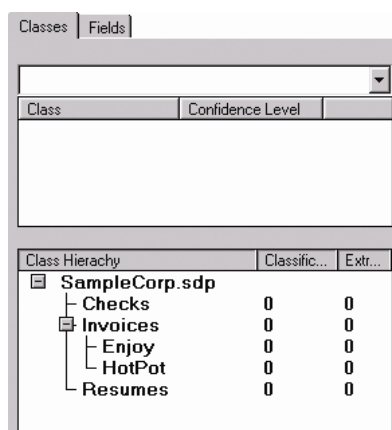



Figure 5-25: Empty Learnsets

The list in the lower section of the *Classes* tab indicates the number of documents in the Learnsets. If required, use the navigation bar in the toolbar to browse to a document to be added to a Learnset.

- 3) In the *Classes* tab, select a target class from the list box in the upper section.
- 4) In the toolbar, click  *Add to Learnset* for adding the current document to the classification Learnset of the selected class. WebCenter Forms Recognition writes a copy of the document to a protected directory structure that is reserved for Learnsets. The entry in the Classification Learnset column in the list in the lower section of the *Classes* tab indicates that a document has been added.
- 5) Repeat until you have finished adding all desired documents.
- 6) Select a working mode. The system will automatically learn the new documents.
- 7) Save the project.

*Note:* You would normally use this method for small projects and demonstrations when you are most likely to change the mode. For projects large projects in which learning would be a lengthy task, it's best to follow the normal learning procedure (See section [LEARNING.](#))

### 5.7.5. Learning

During Learning, Forms Recognition uses your Learnsets and the manually assigned target classes to identify the correlation between document input and target class. WebCenter Forms Recognition learns to classify documents automatically if they are similar to the documents in one of the Learnsets.

Learning is required:

- Once you have set up your classification scheme and created Learnsets.
- When you have changed the classification tree by adding, deleting, moving or renaming classes.
- When you have added or removed a classification method that relies on learning.
- When you have changed any parameters affecting these classification methods.
- When you have changed a Learnset.


Task Prerequisites

**The prerequisites for this task are:**

- You must have set up a classification scheme with the minimum number of classes.
- You must have specified classification methods that rely on learning.
- You must have created appropriate Learnsets with the minimum number of samples.
- The program runs in Train Mode or in Definition Mode.

Learning

**To learn:**

- 1) If required, select the *Options* menu and then *Incremental Learning* to change the scope of learning. If this option is enabled, learning only covers changes to the Learnset. If this option is disabled, the entire project will be learned.
- In the toolbar, click .

Learning may take several minutes. The class that is currently learned is indicated in the status bar. Once finished, a success message is displayed.

### 5.7.6. Checking the Learn Status of a Class

Learning always attempts to cover the entire classification tree. Only classes with insufficient Learnsets are omitted. You can check at any time whether a class has already been learned successfully or whether it requires relearning.

Task Prerequisites

**The prerequisites for this task are:**

- The program is in Definition Mode.
- The panel on the left side of the window displays the *Classes* tab in the foreground.
- On the right side of the window, the tabs with class/field properties are visible.

Check the Learn Status of a Class

**To check the learn status of a class:**

- 1) In the *Classes* tab on the left side of the window, select a class.
- 2) In the *Classification* tab on the right side of the window, under *Used Engines...*, select the Brainware Classify Engine method name.
- 3) Check the Classification engine properties. It displays one of the following states:

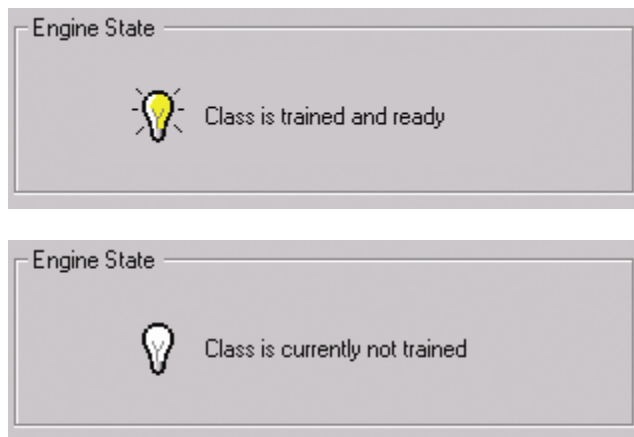


Figure 5-26: Classification Engine Properties

## 5.8 Configuring Brainware Layout Classification Engine

Setting up Brainware Layout Classification is similar to setting up Brainware Classification. You just need to provide sample documents for learning.

If there is already a Learnset for Brainware Classification, there is usually no need to create another one for Brainware Layout Classification.

To create a separate Learnset for Brainware Layout Classification, proceed as described in section [CREATING LEARNSETS](#). To carry out the learning, proceed as described in Section [LEARNING](#).

### 5.8.1. Overview & Purpose

The Brainware Layout Classification engine is an engine for the purpose of so-called “layout” classification using the powerful Brainware Classifier technology. While the Brainware Classification engine available in WebCenter Forms Recognition is used for “content” or “type” classification, the purpose of Brainware Layout Classification (BLC) engine is to provide with more precise classification between documents with similar templates, like, for example, invoices delivered from different vendors, where it is possible to reach more exact result by taking into account the positional information of the documents’ content and not only the textual content of the documents like in ASSA or Brainware “content” classification engines.

### 5.8.2. How it Works

Basically, the BLC engine simply applies normal Brainware Classification. Though, each word of both learned and extracted documents is not just used as it is but extended with special character sequences that uniquely identify which zone of the document the word belongs to. Visually this idea can be represented as a document split to a couple of zones (yellow area is the first page of the document and “VAT” is the word we are looking for):

	A <sub>1</sub>	B <sub>1</sub>	C <sub>1</sub>
A <sub>2</sub>		VAT	
B <sub>2</sub>			
C <sub>2</sub>			

Now the “VAT” word is going to be extended as “VAT\_BBBB\_AAAA” identifying that we look for the word “VAT” only the {B<sub>1</sub>; A<sub>2</sub>} region of the document. This then simply means that if for the learned layout class X the word “VAT” was mostly located in region {C<sub>1</sub>; C<sub>2</sub>} and for the other class Y it was usually {B<sub>1</sub>; A<sub>2</sub>} the system will prefer Y while for normal Brainware Classification such a difference between X and Y would have been irrelevant. Exactly this very simple approach allows to the content classification engine (in this case Brainware Classifier) as more precise layout one.

## 5.9 Configuring Language Classification Engine

### 5.9.1. Overview & Purpose

The Language Classification Engine was introduced to automate the classification of processed documents by language.

The engine is supplied with predefined and pre-trained Learnsets for nine languages:

- Danish
- English
- Finnish
- French
- German
- Greek

- Russian
- Spanish
- Swedish

The English, French, German, Greek and Russian languages are specifically optimized for invoice processing.

New languages can be added easily by the customer, and the predefined ones are adjustable.

### 5.9.2. How it Works

The Language Classification Engine uses the ASSA content classification technology to distinguish between processed documents written in different languages. The engine imports the data in Unicode format and converts them to non-western language Learnsets with a special phonetic English representation suitable for the core ASSA engine. After this conversion, the engine creates an ASSA classification pool that is then used for classification purposes. The pre-trained version of this pool is distributed along with WebCenter Forms Recognition setup and is available without having to learn anything although it is adjustable if the customer would like to extend the predefined Learnset.

When classifying the next incoming document, the system will use the available ASSA classification pool for the classification. During the classification, the engine ignores all numeric words found in the document since they are usually irrelevant in terms of language classification.

*Note that in order to use the engine for the non-western languages, "Activate support of non-western languages" **must be turned on** for the project in the Designer settings, "Definition" tab. If you have documents with CJKT languages, also the multi-byte encoding must be enabled.*

During classification, the system will search for class language assignments in order to determine into which class the document is going to be classified. To create a new language assignment, select the desired class under *Classes* view in Definition mode in Designer, open the *Classification* tab, choose "Language Classify Engine" item in the "Languages available for this DocClass" list:

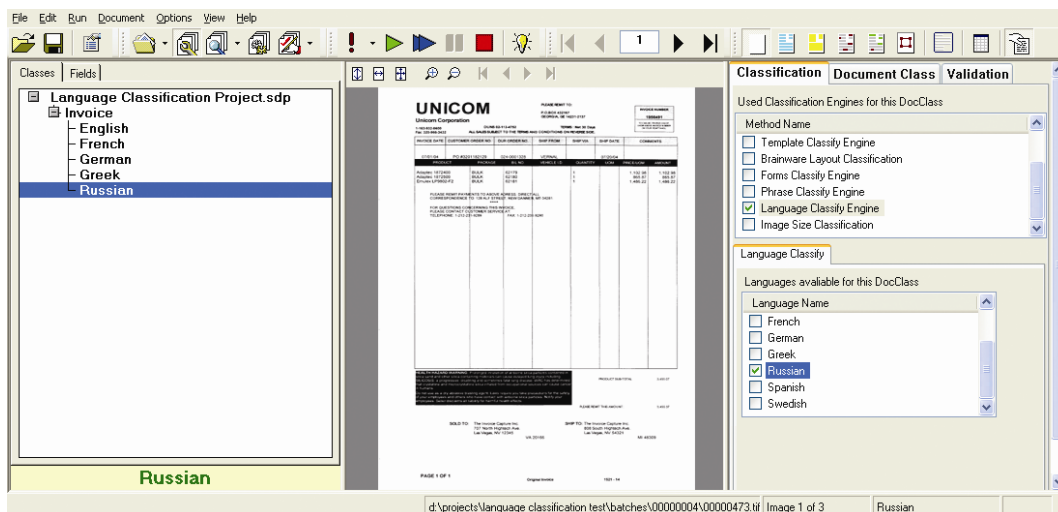


Figure 5-27: Creating a class language assignment for Language Classification.

If required, you can also assign multiple languages to one document class e.g., assigning “German” and “Russian” languages to the *Similar Amount Format* class.

### 5.9.3. Setup the Language Classification Engine

To configure the Language Classification Engine, open the *Class and Field* properties for the project node in Designer and then select the *Classification* tab and choose the Language Classify Engine in the *Used Classification Engines for this project* list:

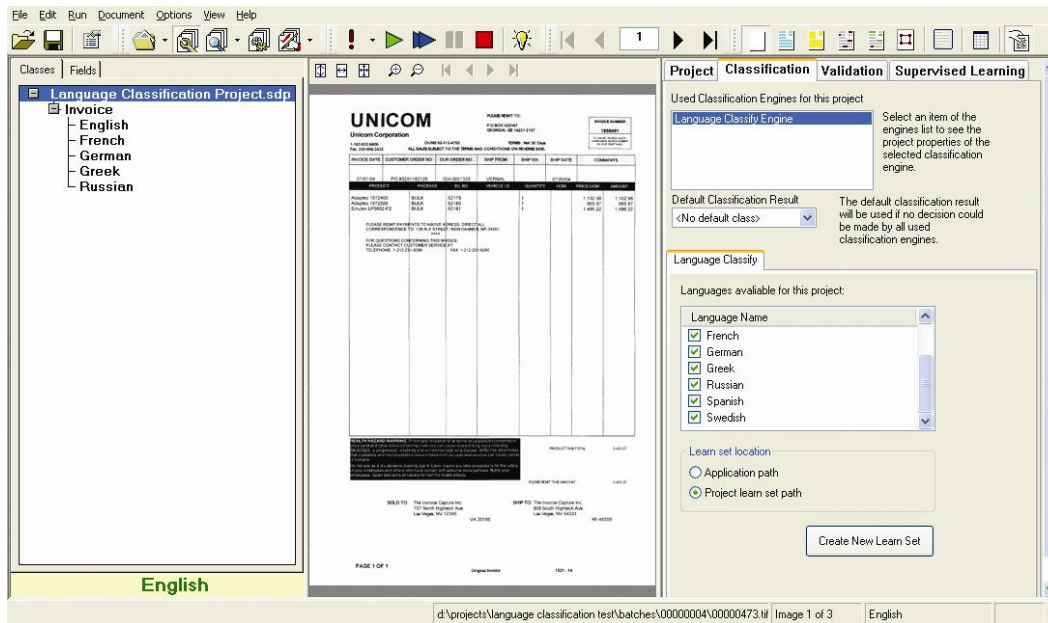


Figure 5-28: Configuring the Language Classification Engine.

The following settings are available on the single *Language Classify* tab:

- *List of languages available for this project* list view, where you can select the languages that are used in your project.
- *Learnset location* setting that defines whether the language classification Learnset and its source files are to be stored in the *Application path* (i.e. in the directory, where WebCenter Forms Recognition applications are launched from) or in the project Learnset (i.e. in the main project Learnset directory, which is configured via the *Base Directory* setting available on the *Train* tab of project settings in Designer).
- *Create New Learnset* button. This is used to regenerate the entire ASSA classification Learnset, for example, to update the language classification configuration with a couple of new languages to be used for classification.

**Note:** The “*threshold*” and “*distance*” settings used while evaluating the classification are taken from the “*Standard Classification*” settings defined on the “*Project*” tab of the project node settings.

### 5.9.4. Extending and Adjusting Standard Learnset with New Languages

When generating a new Learnset, the engine automatically searches for all available \*.LNG files in the “\LangPool\Source” folder. During the generation of the pool, the newly imported languages get their names from the corresponding \*.LNG files.

#### 5.9.4.1. How to Create Unicode Language Input Files

A new Unicode input source language file can be created, for example, in Microsoft Word 2003:

- Open your new input text file with Microsoft Word 2003.
- Select *Save as* from the *File* menu. Choose *Text-only* in the *Data type* drop down list box and click on *Save*.
- Microsoft Word should then display a file conversion dialog. Select *Other encoding* from the *Text encoding* group of radio buttons.
- Now select *Unicode* in the list of available encoding formats and, finally, click *OK*.
- Give an appropriate name to the saved .TXT file and change its extension to .LNG, for example, "Portuguese.Ing" if you are creating a new Learnset for the Portuguese language or "Medical Greek.Ing" if you are creating special language Learnset of Greek medical terms.
- Now copy the .LNG file to the ".\LangPool\Source" sub-folder of the currently configured "Learnset location" (See above for more details on possible language classification Learnset locations). Your new language file is now ready to be imported into the WebCenter Forms Recognition system.

#### 5.9.4.2. Adding New Languages to the Learnset

Once the new .LNG file is copied to the source Learnset location, click *Create New Learnset* on the general Language Classification engine configuration tab to create a new Learnset that supports the desired language.

#### 5.9.4.3. Removing a Language from the Learnset

To remove a language from the language classification Learnset you can either:

- 1) Uncheck the desired language in the *Languages available for this project* list to exclude the language from the list of possible classification results. The language will not be removed from the ASSA classification Learnset. If you would like to make sure that the ASSA pool does not contain any references to the language, click *Create New Learnset* in order to re-generate the entire language classification pool.
- 2) Remove the desired source language file (with .LNG extension) from the ".\LangPool\Source" sub-folder of the Learnset location directory (consider making a back up copy first), and then click *Create New Learnset*. The language will disappear from the list of available languages.

#### 5.9.4.4. Modifying Existing Language Learnsets

In order to modify the Learnset for one (or more) of the existing languages, open the desired language .LNG file in, e.g., Microsoft Word, apply your modification and then use the procedure described in the sections "How to Create Unicode Language Input File" and "Adding New Languages to the Learnset" to replace the existing language source file with the adjusted custom one.



### 5.9.4.5. Classifying Documents with Language Classification Engine

WebCenter Forms Recognition uses the standard classification rules when applying the classification process with the Language Classification Engine. For example, in order to check the exact classification results and weights on a per language basis, use the standard *Class Result Matrix* function available in Designer.

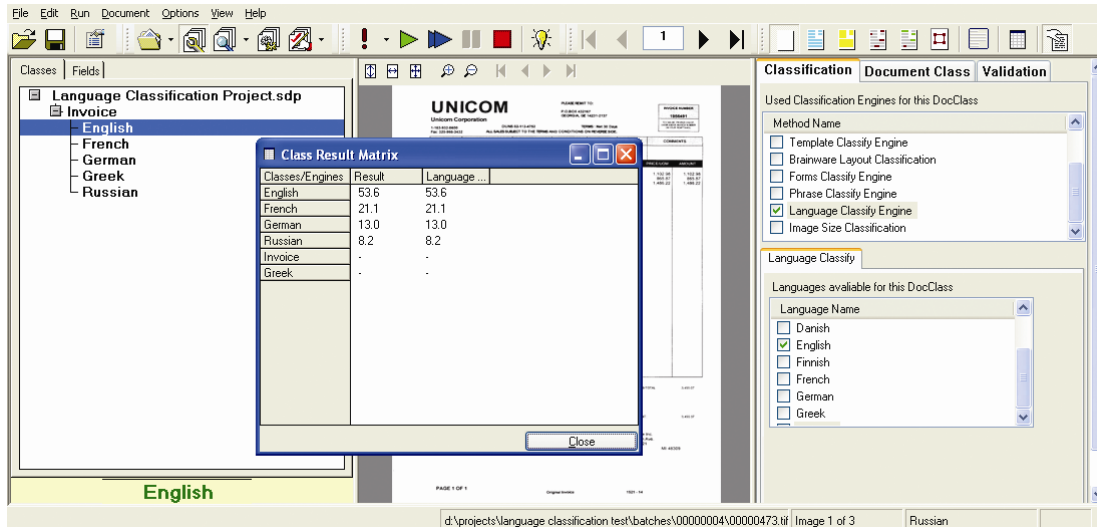


Figure 5-29: Using the classification matrix to review the result of language classification in Designer.

## 5.10 Configuring Phrase Classification

To use phrase classification, you must specify certain keywords or phrases that lead to a particular class or exclude a document from a class.

### Task Prerequisites

The prerequisites for this task are:

- You must have set up a classification scheme.
- Phrase classification is selected for at least one of the classes.
- The program is in Definition Mode.
- The panel on the left side of the window displays the *Classes* tab in the foreground.
- On the right side of the window, the tabs with class/field properties are visible.

### Entering Phrases

To enter phrases for phrase classification:

- 1) In the *Classes* tab on the left side of the window, select a class.
- 2) In the *Classification* tab on the right side of the window, under *Used Engines...* select the *Phrase Classify Engine* method name.
- 3) In the *Phrase Classify Engine* property sheet, under *Classification engine properties*, type a keyword or a phrase into the *New phrase* text box. Alternatively, highlight words in the document and click a word to copy it into the *New phrase* text box. To build phrases, click the next word. The new word is added to the existing entry with a space as separator. Click *Ins* to add the new phrase to the list of active phrases that is displayed below. New phrases are added with medium significance.

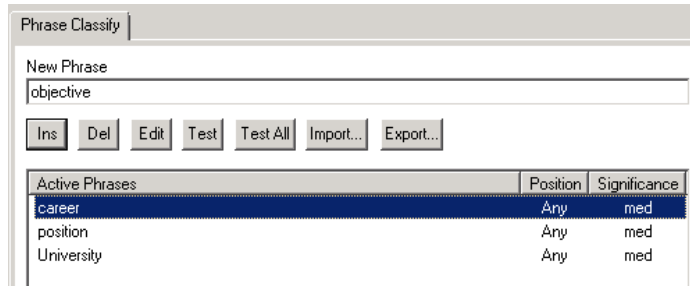


Figure 5-30: Phrase Classification Property Sheet

- To edit the significance, select an active phrase from the list and click *Edit*. The *Phrase Settings* dialog box is displayed. This dialog box contains a separate tab for each word in the phrase.

## Parameters

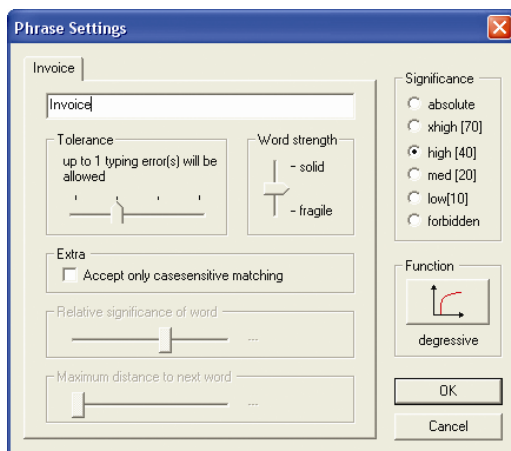


Figure 5-31: The Phrase Settings dialog box

The parameters you can adjust are:

- Tolerance:** Maximum number of typing errors allowed for this word. If the phrase contains the word “invoice,” for example, and the number of error characters has been specified at two for this word, the words “invoke,” “involve,” and “invite” would also meet the requirements.
- Word strength:** The stronger the word, the more typing errors can occur until the word loses its meaning. A maximum strength stands for a word that still retains full significance with the maximum number of permissible typing errors. In general, you should keep the default setting for the word strength.
- Accept only...:** Select this option to enforce case-sensitivity.
- Relative significance...:** Use this option to set the proportional significance of the current word within the phrase.  
For example, if your phrase is “with kind regards,” you could reduce the relative significance of the word “with.” As you move the associated slider, the weight will be shown to the right in plain text.
- Maximum distance...:** Use this setting to specify whether your phrase will still be recognized if there are additional words between the current one and the next one specified.

For example, you could use a single definition for “please pay by” and “please pay this amount by” with a maximum distance of 3. For consecutive words, the distance is 1.

- **Significance:** This setting affects the entire phrase. The significance may vary between absolute (i.e., a document belongs for sure in this class if the phrase is found) and forbidden (i.e., a document cannot belong in this class if the phrase is found). Be careful when using the extreme values. Use them only on very strong phrases and try to avoid them on single words.
- **Function:** Specifies the function that applies if a phrase is found more than once in the same document. This setting affects the entire phrase. To change the function, just click the button. You can choose from among:




Button	Description
	Degrressive: The significance increases degressively for every further occurrence, i.e. the more often a phrase is found, the less effect a new discovery has. This is the default behavior.
	Linear: The full significance of the phrase is scored with every further occurrence.
	Binary: A multiple occurrence is not scored higher than a single occurrence.

Table 5-1: Function Controls for Phrase Classification

- 5) To delete a phrase, select it from the list of active phrases and click *Del*.
- 6) To write phrases and associated settings to an export file, click *Export* and specify a file name. This generates an \*.exp file. To import an existing \*.exp file, click *Import* and select it from the file system.
- 7) To test a phrase, select it from the list of active phrases and click *Test*. Or click *Test All* to test all active phrases.
- 8) To highlight the occurrences of a phrase in the active document, make sure that all other highlighting options are turned off. Then select the phrase from the list of active phrases. Each occurrence is highlighted in green. To view the number of typing errors and the significance, point to the highlighted phrase.

## 5.11 Configuring Image Size Classification

To use image size specification, you must specify the document size for the corresponding class. It is also possible to specify a size range.

### Task Prerequisites

The prerequisites for this task are:

- You must have set up a classification scheme.
- Image size classification is selected for at least one of the classes.
- The program runs in Definition Mode.

- The panel on the left side of the window displays the *Classes* tab in the foreground.
- On the right side of the window, the tabs with class/field properties are visible.

### Specifying Document Size

To specify the document size:

- 1) In the *Classes* tab on the left side of the window, select a class.
- 2) In the *Classification* tab on the right side of the window, under *Used Engines...* select the Image Size Classification method name.

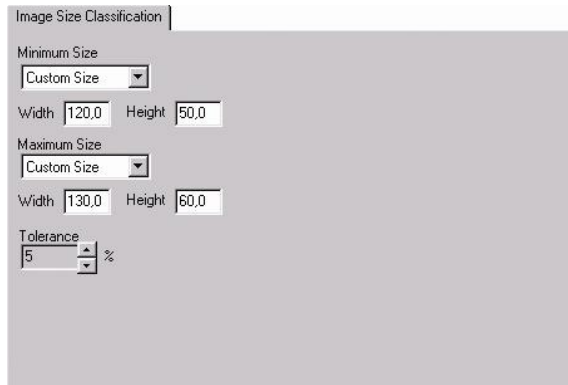


Figure 5-32: Image Size Classification Property Sheet

- 3) In the *Image Size Classification* property sheet, under *Minimum Size*, select one of the predefined sizes, or select *Custom Size* and enter the Height and Width in millimeters into the corresponding text boxes.
- 4) Repeat this for the *Maximum Size*.
- 5) Under *Tolerance*, use the spin box to specify the error tolerance in percent that is applied when comparing sizes. This value is particularly important if your Maximum Size and Minimum Size are equal.

## 5.12 Configuring Forms Classification

To use forms classification, you must create reading zones on documents and compare the recognition results from these zones with known identifiers of a class of forms. You can also configure a forms classification on the project level if more than one class shares zones.

Reading zones are fixed areas on documents that contain information that is to be recognized. In WebCenter Forms Recognition context, three different types of reading zones exist – OCR, OMR and barcodes. However, in forms classification context, only OCR and barcode zones can be used:

- **OCR:** The contents of the reading zone are determined using optical character recognition. The textual content of the zone is the result of this process.
- **Barcode:** The contents of the reading zone are determined using barcode recognition. The number or strings represented by the barcode is the result of this process.

To learn about OMR zones, see section [OMR ZONES](#).

### 5.12.1. Creating Reading Zones

#### Task Prerequisites

The prerequisites for this task are:

- You must have set up a classification scheme.
- Forms classification is selected for at least one of the classes.
- The program is in Definition Mode.
- The panel on the left side of the window displays the *Classes* tab in the foreground.
- On the right side of the window, the tabs with class/field properties are visible.
- Reading Zones must be visible. (On the *View* menu, select *Show Page*.)

You may use more than one reading zone on each form.

#### Creating Reading Zones

To create reading zones:

- 1) In the *Classes* tab on the left side of the window, select a class.
- 2) Load a document into the viewer that belongs to the selected class.
- 3) From the menu, select *View - Show Page*. The viewer toolbar displays additional buttons for zone creation:






Button	Description
	Activates the selection tool
	Creates an OCR reading zone
	Creates an OMR reading zone
	Creates a barcode reading zone
	Creates an anchor

Table 5-2: Controls for creating reading zones

- 4) In the viewer panel, click the relevant toolbar button to create an OCR zone or a barcode zone.
- 5) Click on the document and drag to create a rectangular reading zone around the form identifier. Obviously, the rectangle needs to be large enough to hold the identifier. It should not be much larger, though. The reading zone is displayed as transparent rectangles with red borders. Each zone has a label that displays the name of the zone. The initial name is created from the word *Zone* and a consecutive number.

For more information about creating and editing reading zones, see section [SETTING UP ZONE ANALYSIS](#) and [ADVANCED RECOGNITION SETTINGS](#).

## Configuring Form Identifiers

### To configure the form identifiers (applicable only for OCR zones):

- 1) In the *Classification* tab on the right side of the window, under *Used Engines...* select the Forms Classify Engine method name.

	Zone Name	Compare Text	Page Number
0	Zone	NET INVOICE	1
1			

advanced settings

Match all zones

Required number of  of 1 zones

Figure 5-33: Forms Classification Property Sheet

- 2) Click on a cell of the *Zone Name* column and select a zone from the list of available reading zones.
- 3) In the *Compare Text* column, type the string that identifies your form. This string will be compared to the recognition result from the reading zone.
- 4) In the *Page Number* column, click to specify whether the reading zone is on the first or on the last page of the document.
- 5) Repeat Step 2 through Step 4 for all reading zones that should be taken into account.

By default, recognition result and form identifier are compared using an error-tolerant algorithm.

### To specify the comparison method in more detail:

- 1) On the *Forms Classify* tab, click the row number to select a row.
- 2) Click *Advanced*.

Advanced Settings

Compare Text  
NET INVOICE

Compare Method  
Levenshtein

ignore first  characters

ignore last  characters

max. compare distance (0-100)

OK Cancel

Figure 5-34: Advanced Comparison Settings

- 3) In the *Compare Text* field, you can edit the comparison string. This may be required, depending on the comparison method you select.

- 4) In the *Compare Method* list, you can select the following methods:
- String Compare: A very simple method that returns a match for a literal occurrence of the comparison string.
  - Trigram: An error-tolerant method that returns a match for a literal occurrence of the comparison string, but also for strings that can be derived from the specified one by fragmenting the text into groups of three characters called trigrams. The number of identical groups determines whether there is still a match.
  - Levenshtein: Another error-tolerant method that returns a match for a literal occurrence of the comparison string, but also for strings that can be derived from the specified one by inserting, interchanging or deleting single characters. This is the default method.
  - Regular Expression: A method that returns a match if the recognition result corresponds to a possibly complex format pattern that needs to be entered into the Compare Text field.
  - Simple Expression: A method that returns a match if the recognition result corresponds to a simple format pattern that needs to be entered into the Compare Text field.
- 5) You may be willing to assign several forms with similar identifiers or several versions of the same form to the same class. In this case, it may be useful to ignore leading or trailing characters of the recognition string. You can set the number of characters that are to be neglected using the *ignore first...* and *ignore last...* spin boxes.
- 6) The Max. Compare Distance allows you to perform the comparison, neglecting leading or trailing characters of the comparison string. The compare distance is computed as follows:
- $$\text{Compare Distance} = 1.00 - \text{Number of Characters (Recognition String)} / \text{Number of Characters (Comparison String)}.$$
- A match requires that the actual compare distance is less or equal the maximum compare distance.

For more information about the comparison methods and the syntax of simple expressions, refer to section [SETTING UP](#). The syntax of regular expressions is described in [REGULAR EXPRESSIONS](#).

In the *Forms Classify* tab, color coding is applied to the comparison string to indicate the selected method.






Swatch	Color	Method
	purple	String Compare
	turquoise	Trigram
	olive	Levenshtein
	gray	Regular Expression
	black	Simple Expression

Table 5-3: Color coding in *Forms Classify* tab

In some cases, you may need multiple reading zones to obtain a reliable classification. In this case, you need to specify how the classification result is evaluated.

**To set the number of zones that must match, do one of the following:**

- In the *Forms Classify* tab, check *Match all zones*. This is the default.
- In the *Forms Classify* tab, clear *Match all zones* and set the number of required zones in the *Required number...* spin box.

**5.12.2. Global OCR Zones in Forms Classification**

It is possible to share OCR zones from the project page between different document classes. If there are several document classes that have to read the forms ID from the same place in the document, you can set it up in the project page to read the zone globally.

Before you share OCR zones, make sure to have at least one class set up with forms analysis.

**To set up a shared OCR zone, do the following:**

- 1) Select the project name.
- 2) Set up a zone by following instructions for *Creating a Reading Zone*, section [CREATING READING ZONES](#). When setting up the zone, minimize the *All* zone first.

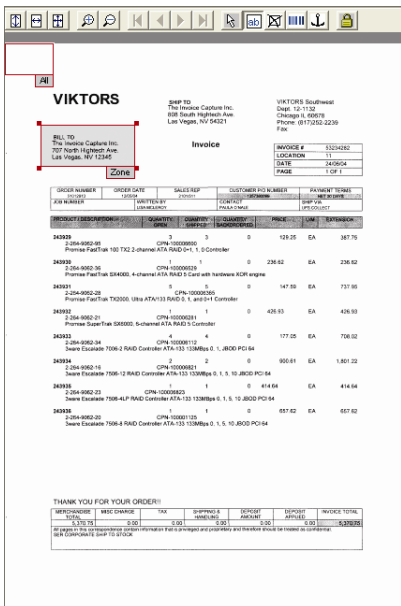


Figure 5-35: A project zone set up at the project level. Establishing OCR Zones

The global project zone will appear in the *Zone Name* dropdown box for all classes that have forms classification.



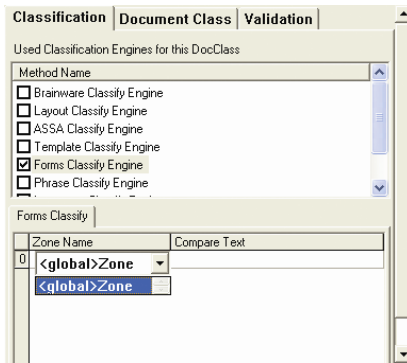


Figure 5-36: Dropdown box showing shared OCR zone from the project level

## 5.13 Choosing Classification Method

### 5.13.1. The ASSA Classification Engine

Like the Brainware Classify Engine, the ASSA Classify Engine needs a Learnset of documents for classification. The difference between the Brainware Engine and the ASSA Engine lies in the different search pools that are generated out of the documents of the Learnset when the documents are learned.

For its search pool, Brainware generates a dictionary of words. The documents for the search pool are coded with words from this dictionary.

The ASSA engine uses the trigram method to code the search pool. The classification of documents is based on the block text search, which involves retrieving the text from the document block-by-block and generating a query string out of the block text. The ASSA engine then searches for that query within the search pool.

#### The prerequisites for this task are:

- You must have a loaded project.
- The program is in Definition Mode.
- ASSA classification is selected for at least one of the classes.
- The panel on the left side of the window displays the *Classes* tab in the foreground.
- On the right side of the window, the tabs with class/field properties are visible.

### 5.13.2. ASSA Configuration

#### The ASSA engine has three tabs for configuring project-level settings:

- *ASSA classify*: Sets the confidence parameters.
- *Regions*: Restricts the area of document to be classified.
- *Tuning*: Defines stretching and calibration that is relevant when the engine is combined with other classification engines, as well as aspects concerning the search speed versus quality.
- *Stretching*: Stretching improves the results of Brainware Classification when combined with other Classification Engines. Stretching means to increase the distance between results given by the classification engine.

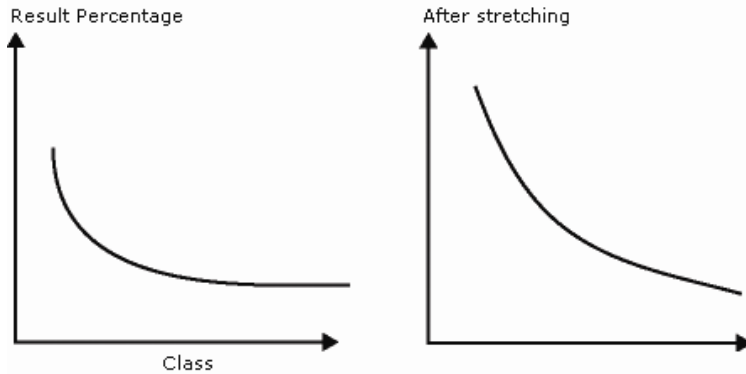


Figure 5-37: Stretching the results in ASSA

You can change the Stretching factor with the slide bar. The stretching factor, which increases the distance, can vary from 1.0 to 5.0.

For example if the stretching factor is 2.0 and Brainware classification result is:

	Before Stretching	After Stretching
<b>CLASSID</b>	Percentage	Percentage
<b>Class1</b>	70	70
<b>Class2</b>	66	62
<b>Class3</b>	56	46
<b>Class4</b>	32	8

Table 5-4: ASSA Stretching results

- **Calibration:**  
Calibration fine-tunes the various classification engines. Calibration is based on the Calibration factor, which can vary from 0.1 to 1.5..
- **Max Quality / Min Speed ...:**  
Max Quality/Min Speed offers the option to select a rapid classification, which sacrifices quality, or a slower classification, which offers greater accuracy. To elect rapid classification, you can set a percentage of the full block text. Or you can elect greater accuracy by creating a query based on the full block text, which takes additional time to compute.

## 5.14 Testing the Classification

When a classification scheme has been set up and the classification methods have been defined and configured, the classification can be tested.

### Task Prerequisites

The prerequisites for this task are:

- There is an active document set.
- There is a trained classification scheme.
- To test the classification only, your project settings should exclude data extraction from runtime functionality.

There are several ways to test the classification.

### 5.14.1. In Definition Mode

Use one of the following buttons to determine target classes and confidence values:





Button	Description
	Processes the current document. The button's drop-down menu enables/disables the debug mode.
	Processes the next document.
	Processes all documents in the current set starting with the current one.

Table 5-5: Buttons for document analysis available in Definition Mode

For unprocessed documents, the software displays Not Classified in the bottom-left edge of the window. After processing, the proposed class is displayed. If WebCenter Forms Recognition cannot assign the document, it displays Uncertain Classification.

To display an array of classification results for the previously processed document, click  **Show/Hide Classification Results Matrix**.

Classes/Engines	Result	Brainware ...
Invoices	100.0	100.0
Misc	62.5	62.5
Agreements of Sale	35.9	35.9
License Agreements	35.5	35.5
Maintenance Agreements	16.1	16.1
Contracts	-	-
SalesDocuments	-	-

Figure 5-38: Classification results for a given document

For more information about how the confidence levels are obtained, please refer to [ADVANCED EVALUATION SETTINGS](#).

If debug mode is enabled, a dialog box is displayed every time a document is being processed. It shows a tree view of possible target classes and confidence levels. In addition, the parameters for classification evaluation are displayed. For more information about evaluation parameters, please refer to [ADVANCED EVALUATION SETTINGS](#).

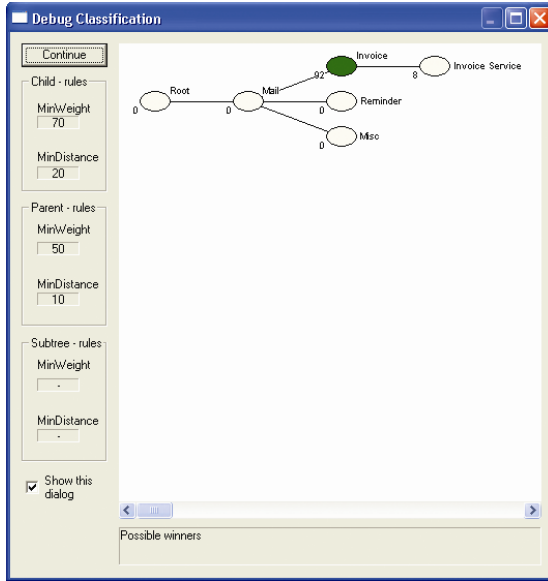


Figure 5-39: Testing classification in debug mode

- To close the dialog box, click *Continue*.
- To quit debug mode, clear the *Show this dialog* check box.

### 5.14.2. In Train Mode

Use the following button to determine target classes:


Button	Description
	Processes the current document. The button's drop-down menu enables/disables the debug mode.



Table 5-6: Button for classification available in Train Mode

Just as in Definition Mode, for unprocessed documents, the software displays Not Classified in the bottom-left edge of the window. After processing, the proposed class is displayed. If WebCenter Forms Recognition cannot assign the document, it displays Uncertain Classification.

Debug mode options in Train Mode are the same as in Definition Mode.

### 5.14.3. In Runtime Mode

Use one of the following buttons to determine target classes and confidence values:

Button	Description
	Processes the next document.
	Processes all documents in the current set starting with the current one.


Button	Description
	Stops the processing of documents. Clears any results from previous runs.

Table 5-7: Buttons for document processing in Runtime Mode

The results are presented as follows: The panel on the upper left side displays summary results for all classes within the current project. It contains two separate tabs, one for a statistic view of mean classification and extraction results, the other one for a tree view of the classification distribution. Both views are continuously updated at runtime.

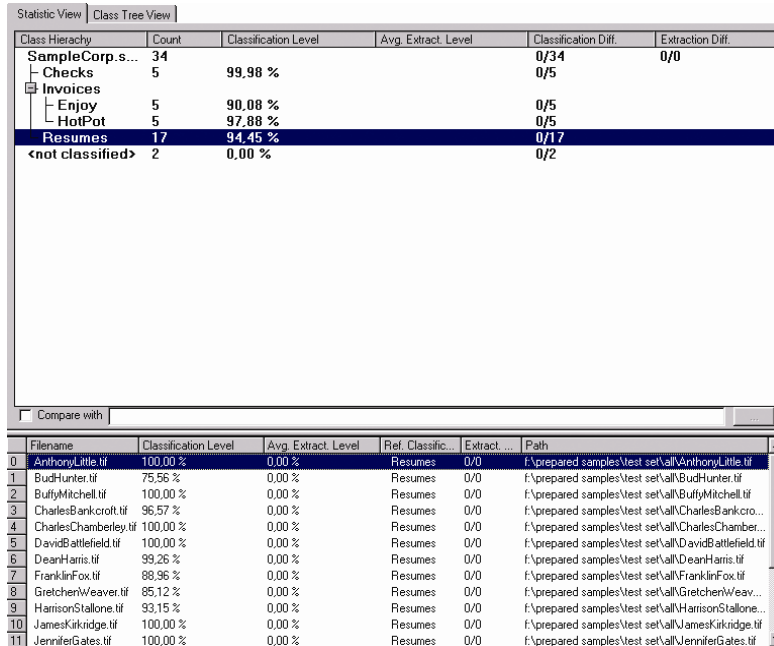


Figure 5-40: Runtime mode classification results

The classification-related columns in the *Statistic View* tab have the following meaning:

Column	Description
Class Hierarchy	Class name and position within the classification scheme
Count	Number of documents assigned to a class or total number of processed documents. Numbers are accumulated until you stop processing.
Classification Level	Mean confidence for assignments to a class.
Classification Difference	Comparison of actual classification results with results from a reference file. The first number represents the number of different classifications in this run; the second one represents the total number of classifications in this run. If there is a difference between test results and reference results, this is indicated by the red color.

Table 5-8: Columns of Classification results in Runtime Mode

The *Statistic View* tab lets you compare your classification results with results stored as reference files. This feature is valuable for example for benchmark tests.

**To create a reference file:**

- Enable creation of the default export file in the project settings.
- Right click any class entry within the *Statistic View* tab and select *Save As Reference* to write the current results to a reference file.
- You can edit reference files as well as the default export file with a text editor, or you can use text editors to create a reference files from scratch. Each line in the file represents the results of one document. The line structure is as follows:

```
<file name><TAB><class><TAB><field 1><TAB>...<TAB><field n>
```

Example:

HotPot2.tif → HotPot → \$5.489,98 → 10324

#### To select a reference file:

- Right click any class entry within the *Statistic View* tab and select *Load Reference*, then select a \*.ref file or the default export file.
- Click on  at the bottom of the panel, and then select a reference file.

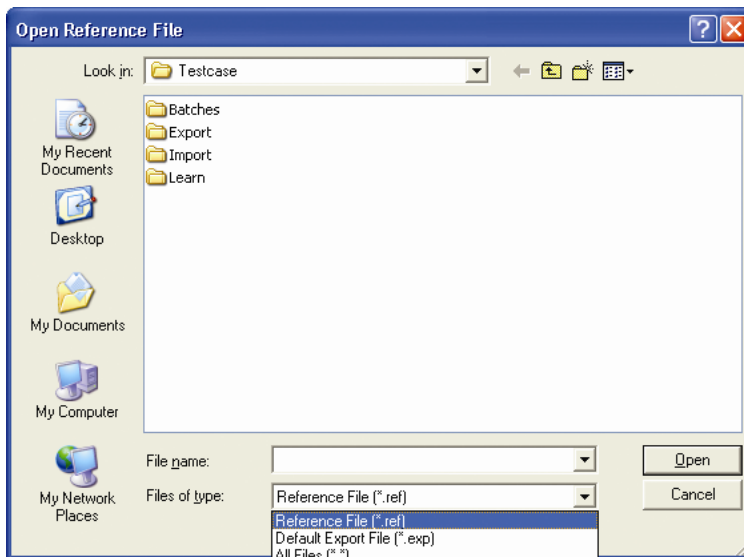


Figure 5-41: Loading reference files

#### To use a selected reference file for comparison with the subsequent runs:

- Right click any class entry within the *Statistic View* tab and select *Compare with Reference*, then confirm.
- Check the *Compare with* check box at the bottom of the panel.

#### The Class Tree View tab displays the distribution of class assignments:

- To hide a branch, double click the corresponding root node.
- To show a hidden branch, double click again.

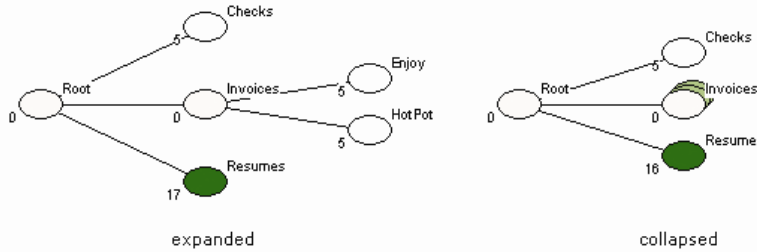


Figure 5-42: Classification distribution with expanded and with partially collapsed branches

If you click a class in either the *Statistic View* or the *Class Tree View*, the panel below will show which documents have been assigned to the current class. The classification-related columns in this panel have the following meaning:

Column	Description
Classification Level	Confidence for assignments to a class.
Ref. Classification	Target class for this document as given in the reference file

Table 5-9: Document-Related Classification Results

## 5.15 Configuring the ASSA Engine for Classification in Automatic Supervised Learning

To configure classification with the ASSA Engine:

- 1) In Definition mode, click on the *Classes* tab on the left side of your screen.
- 2) On the *Classes* tab, select a class.
- 3) On the *Classification* Editor on the right side of your screen, under *Method Name*, select ASSA Classify Engine.

## 5.16 Optimizing the Classification

Most classification problems are due to either:

- 1) Insufficient OCR quality.
- 2) Insufficient Learnset quality.
- 3) Inappropriate selection of classification methods.

### 5.16.1. Resolving Problems with the OCR

Use the highlighting options to identify whether there are problems with the OCR. If this is the case, proceed as follows:

- The problem may be caused by the quality of your images. Make sure that your document input is properly prepared. For example, check scanner settings or fax export settings.
- The default OCR settings may not be suitable to process your document input. For instructions on how to change these settings, see section [ADVANCED RECOGNITION SETTINGS](#).

Changing the OCR settings at the project level is a severe intervention. Make sure that other reasons are excluded before you do this.

### 5.16.2. Resolving Problems with the Classification Methods

In Definition Mode, check the classification results of problematic documents. Identify the method that causes the problem. Determine whether classification fails because the confidence is too low or whether the distance is too low. Also, watch for large fluctuations.

- If documents are classified to the wrong class, you need to change the Learnset or the parameters for the method that returns the wrong result. Result fluctuations often indicate an inappropriate set of rules.
- If too many documents are not classified at all, consider adding another classification method, or providing more samples for the Learnset.
- If this does not help either, you may have to change the settings for classification evaluation. You may refer to [ADVANCED EVALUATION SETTINGS](#).

### 5.16.3. Resolving Problems with the Learnset

Check the contents of unclassified or wrongly classified documents and compare them with the documents in your Learnset. Often, you just need to add unclassified documents to the Learnset of the target class to improve your results. When changing the Learnset, consider the following general rules:

- Add documents with a confidence below the threshold if their content fits into the selected class.
- Add documents that were previously not classified correctly due to the initially insufficient Learnset.
- Use documents with an almost identical confidence for two or more classes. These documents are suitable to differentiate the classes.
- Avoid documents with a high confidence. These documents do not improve the Learnset, because they would already be classified correctly.



## 6 Planning Applications

In WebCenter Forms Recognition, every document is processed in a sequence of operations:

- 1) Import
- 2) OCR
- 3) Layout analysis
- 4) Classification with optional verification
- 5) Data extraction with optional verification
- 6) Planning Supervised Learning (optional)
- 7) Export

Generally, each step requires a careful analysis of the current document input, the associated business processes, and some planning.

### 6.1 Identifying the Document Import Formats

You can process scanned paper documents, faxes, e-mails, files, or documents from mixed sources.

Paper documents are forwarded as \*.tif images to WebCenter Forms Recognition. You need to ensure that the image quality is good enough to obtain reasonable OCR results. Black-and-white images with a resolution of 300 dpi work best. You may need to optimize your scanner's settings to attain the resolution.

Although OCR optimization can be done in WebCenter Forms Recognition, there is no way to regain information that was lost during scanning.

To process files, a suitable filter for text extraction must be available. WebCenter Forms Recognition's built-in filter can process:

- ASCII documents
- Word documents
- Excel documents
- PowerPoint documents
- HTML pages
- AmiPro documents, up to Version 3.1
- WordPerfect documents, up to Version 8

Other file formats require individual testing. Before importing the document into WebCenter Forms Recognition, first convert it to a supported format.

To process e-mails, first save the messages to the file system.

### 6.2 Identifying the Document Classes

Review your documents and look for:

- Similar textual content

- Reoccurring phrases
- Reoccurring forms or letterhead paper
- Characteristic document dimensions

These document properties can be employed in classification. Use them to define an initial classification hierarchy.

You cannot define a classification hierarchy based on document dimensions alone.

## 6.3 Planning the Classification Methods

WebCenter Forms Recognition offers several classification methods that can be used alone or in combination with each other.

Classification methods are specified at the class level. You can apply one or several classification methods to each class, and an evaluation algorithm will combine the results and decide whether a given document belongs to a particular class.

At this stage, you need only a rough idea of classification methods. To improve the initial classification results, you can easily add classification methods later.

### 6.3.1. Content Classification with Brainware

Content classification is based on Brainware neural network technology. To use this method, you yourself must be able to distinguish the documents by their content. The formal structure of the document is unimportant. The same is true for syntax or phrases, since only words are used as input.

For example, the following classification problems can be solved with Brainware classification:

- E-mails classified by project
- Business correspondence by type (such as orders, invoices, or resumes)
- Essays classified by author
- Customer requests classified by topic
- Movie descriptions classified by genre.

Brainware classification can be used as the only classification method.

### 6.3.2. Phrase Classification

Phrase classification uses words and phrases to identify the document class. Within certain error limits, the phrases or words must literally occur within the documents.

Phrase classification can be used as a very powerful complement to content classification since you can define certain keywords that must lead to a specific class. You do not have to define every possible phrase, since Oracle does that job.

Phrase classification can be used as the only classification method. However, this is not generally recommended, because it may require setting up and maintaining a potentially complex set of rules. Typically, phrase classification is used in conjunction with Brainware Classification. Phrase classification should not be used to train base classes.

### 6.3.3. Template Classification

The Template Classification has been replaced by the Brainware Layout Classification. Please refer to section [6.3.6](#).

### 6.3.4. Image Size Classification

Image size classification uses the physical size of documents to distinguish classes. This method will not be able to positively prove that a document is in a specific class. Instead, image size classification is used to rule out classes and whole branches by excluding documents that do not match. This method is very fast. Use it to reduce a classification problem and to shorten the computing time all at once.

### 6.3.5. Forms Classification

Forms classification can be used to identify forms or other structured documents that have an identifier of the document class printed on them. If the identifier is placed at a fixed position on the document, reading this zone can quickly provide a classification result. Forms classification is well suited to assigning structured documents to a class and combining several forms in one class. However, the document input is usually too heterogeneous to use forms classification as a stand-alone method.

### 6.3.6. Brainware Layout Classification

The Brainware Classification engine is used for content or type classification. The Brainware Layout Classification (BLC) engine provides a more precise classification between documents with similar templates. For example, considering invoices from different vendors it would be possible to reach a better result by taking into account the positional information of the documents' content as well as the textual content of the documents.

The document is divided into a number of zones. Each piece of text is tagged to indicate which zone it appears in. This means that if for the learned layout class X the word "VAT" was mostly located in region the top left-hand corner and for the another class Y it was usually in the bottom right-hand corner, the system would prefer one classification over the other while for normal Brainware Classification such a difference between X and Y would have been irrelevant.

## 6.4 Identifying the Fields for Data Extraction

Extraction means that selected data from a document is automatically written to an extraction file. In general, classification is a precondition for extraction because the fields that need to be extracted are usually different for each class. If it is not necessary to distinguish different document classes and only extraction has to be performed, you can carry out a dummy classification with only one class that needs to be defined as the default. (See section [PROJECT-LEVEL DEFAULT CLASSES – HOW DO THEY WORK?](#)).

For each class, identify the business processes that use the documents. Identify the data that is required by subsequent systems. Then define the set of fields that are to be filled per class.

## 6.5 Planning the Extraction Methods

Extraction of information from the document into a field occurs in two distinct steps:

- Analysis: In this step, the document content is analyzed and a set of possible values for a field is generated. These values are called candidates.

- Evaluation: In this step, the correct candidate is selected from the set of candidates.

WebCenter Forms Recognition supports several analysis methods. The best choice for the evaluation method depends on the selected analysis method. Analysis and evaluation methods are defined at the field level.

Although it is possible to combine several classification methods for each class, you can use only one analysis method and one evaluation method per field.

Evaluation is normally only necessary for the Format Analysis engine (Section [FORMAT ANALYSIS](#)) It can also be used for Zone Analysis (Section [ZONE ANALYSIS](#)) or Address Analysis (Section [ADDRESS ANALYSIS](#)) if various zones are defined. Other engines do not need an evaluation engine.

### 6.5.1. Brainware Table Extraction

Use this method to interactively train table extraction line-by-line (row-by-row.) This engine can learn which lines to extract, which lines not to extract, and which lines to use for learning — and which lines not to use. It can also learn line types and assign color coding to them. This engine also “knows” when the Learnset has changed. Further, the engine can extract columns and cell data. See section [SETTING UP](#) for detail.

Included within the Brainware Table Extraction is the extension for Generic Table Extraction that provides easy access and simple configuration of the extraction. For details see section [8.4.2 CONFIGURING GENERIC TABLE EXTRACTION](#).

### 6.5.2. Table Analysis

Prior to Version 10.1.3.5.0, this method was used to extract a series of records that were organized in columns. As with Brainware Table Extraction, traditional Table analysis yielded several candidates, but the one with the highest confidence is used automatically. Therefore, no evaluation step is required. To access the remaining candidates, you used custom evaluation methods implemented as WinWrap Basic scripts. For details, see section [SETTING UP TABLE ANALYSIS](#).

### 6.5.3. Format Analysis

Use this method to extract data that might be located at arbitrary positions in the documents. Format analysis uses a formal description of the character string you are looking for. For each field, a set of search patterns can be defined. All strings within the document text that match at least one of the specified patterns are candidates. Typically, there are several candidates.

Format analysis is usually combined with Brainware evaluation. That is, the user can take a couple of examples and select the correct candidate from the possible ones. The software can then learn to select automatically. Alternatively, format analysis can be combined with custom evaluation methods implemented as WinWrap Basic scripts. For details, see section [SETTING UP](#).

### 6.5.4. Zone Analysis

Use this method to extract data located at fixed positions in a document. This requirement is usually met with classical forms. For each field, a reading zone or a set of reading zones on the document is defined. The recognition methods available for zone reading include OCR engines, barcode recognition, and optical mark detection (OMR). Typically, zone analysis

yields a single candidate, but with multiple reading zones per field, a set of candidates can be obtained.

Zone analysis only requires the evaluation step if there are multiple candidates. You can then either use Brainware evaluation or custom evaluation methods implemented as WinWrap Basic scripts.

For details, see section [SETTING UP ZONE ANALYSIS](#).

### 6.5.5. Address Analysis

Use the Address Analysis2 Engine to identify addresses on documents and to validate the analysis result against entries in a database. Addresses can be located at arbitrary positions within the documents. Address analysis yields one or several candidates.

Address analysis always requires the evaluation step, even if there is only one candidate. You can either use Brainware evaluation or custom evaluation methods implemented as WinWrap scripts.

*Address analysis is currently supported for German, Austrian, Swiss, and U.S. address formats only. For details, see section [SETTING UP ADDRESS ANALYSIS](#).*

### 6.5.6. Associative Search Engine

Use this method to extract addresses, or as a classification engine in combination with Brainware Layout Classification. (See sections [PLANNING BRAINWARE LAYOUT CLASSIFICATION](#) and [CONFIGURING BRAINWARE LAYOUT CLASSIFICATION ENGINE](#) for more details).

## 6.6 Planning the Verification

WebCenter Forms Recognition features a dedicated application for quality assurance: WebCenter Forms Recognition Verifier. This application provides the means to check and correct uncertain or invalid results of automatic document classification and data extraction. In addition, it permits manual classification and indexing of documents.

For each class, decide which steps are to be carried out automatically, which steps are to be carried out manually, and which verification steps will be involved (Classification and extraction verification separately, or both procedures in one step.) If manual indexing is involved, decide whether this should be done with or without database support (Smart indexing.)

*Verification can also be carried out in stages, and exception handling mechanisms can be implemented to account for special error situations.*

Plan the verification forms that will be required and the validation rules that will be imposed on user input.

## 6.7 Planning Supervised Learning

The Supervised Learning Workflow (SLW) is used to automatically create and learn new document classes. This automatic process creates the new derived documents classes for a generic – base – document class (Level 0.) This document class is not inherited from any other document class. The derived document classes (Level 1) cannot have further derived document classes.

Additionally, the SLW must be enabled in both Designer and Verifier.

## 6.8 Planning the Document Export

By default, WebCenter Forms Recognition uses a specified export directory where documents and their associated Workdocs are saved after processing. To change this behavior, use one of the available custom export methods implemented as WinWrap scripts.

## 6.9 Planning the Page Separation

### 6.9.1. Batch Properties

To create a batch for the “page separation” feature, configure a RTS instance for Import and OCR.

- The documents to import must consist of single-sided documents.
- The import creates batches with one folder per document.
- The workflow states should be different to the default workflow states. Otherwise, there will be conflicts with the other instances using RTS that have their own state numbers.

### 6.9.2. Multipage Detection

Multipage detection uses phrases to find documents that belong together and makes multiple page TIFF documents from them. The documents are checked during the classification and extraction steps of batch processing. Multipage detection works only within each folder of the batch. You must scan the relevant documents in the correct order to the same folder.

*If pages of a document are spread to different folders, multipage detection will not work even when the folders belong to the same batch.*

#### 6.9.2.1. Set Up Multipage Detection

Multipage detection is activated in Designer.

Task Prerequisites

**The prerequisites for this task are:**

- The program is in Definition Mode.
- On the left side of the screen, the *Classes* tab is activated
- On the input tab of the settings the option *Save Execution Results to Workdoc* is enabled. Otherwise the changes cannot be stored.

Activating Multipage Detection

**To activate multipage detection:**

- 1) In the *Classes* tab, right click the entry representing your project to display a shortcut menu.

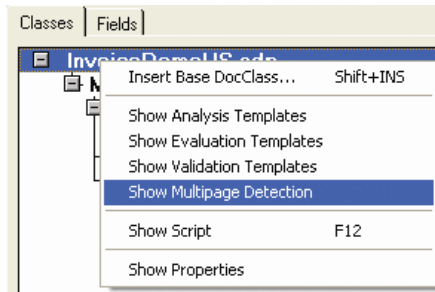


Figure 6-1: Project shortcut menus:

- 2) Select *Show Multipage Detection*. The Multipage Detection dialog box appears.
- 3) Check on the Activate Multipage Detection.
- 4) Select a default for the start of detection, either First Page, or the following page.
- 5) If you want to keep the page sizes similar, select *Similar Page Size Required* and select the tolerance for the page size.

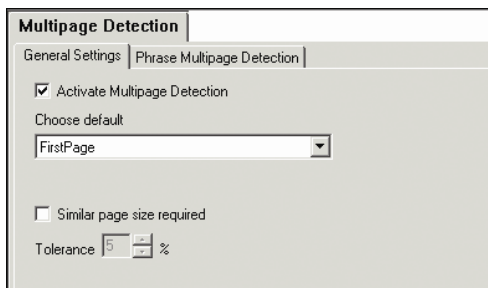


Figure 6-2: The Multipage Detection dialog box, General Settings

- 6) Select the *Phrase Multipage Detection* tab.

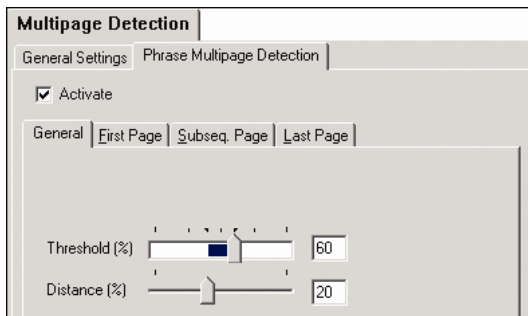


Figure 6-3: The Phrase Multipage Detection tab

- 7) Set thresholds and distances, if desired, for Multipage Detection. The default is 60 percent for Threshold, and 20 percent for Distance.
- 8) Select *First Page*, *Subsequent Page*, and *Last Page* to define phrases for the different pages.
  - To enter a phrase, edit the *New Phrase* text field. By clicking *Ins*, you can insert the phrase with a default significance of med and default value of any.
  - To modify settings for a phrase, either double click a phrase or press *Edit*. For more information about editing phrases, see section [CONFIGURING PHRASE CLASSIFICATION](#).

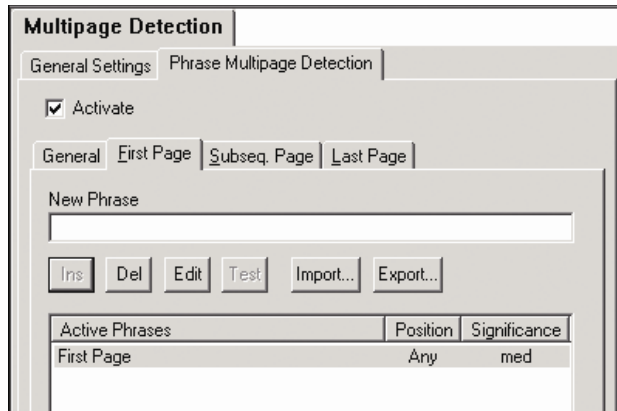


Figure 6-4: First Page in Multipage Detection, defining phrases

### 6.9.2.2. Other Multipage Detection Settings

To run Multipage Detection, you must configure the following settings in Runtime Server:

Select these settings on the Workflow tab in Runtime Server:

- The classification and extraction step must be performed together.
- Perform folder-based classification and extractions

*Note: Do not use the Import option "Document Grouping - 1 folder per document."*

For more information about Multipage Detection and Runtime Server settings, see the ***Runtime Server User's Guide***.

### 6.9.3. Page Separation Learnset

The "page separation" needs its own Learnset.

To prepare a Learnset for "page separation", a directory, which contains at least 50 documents fully extracted and validated, single-sided images with work document files (\*.wdc), is needed. To prepare these documents use a verifier project containing a single base class and five defined fields given exactly in the following order:

- InvoiceNumber
- Date
- Page
- PageNumber
- SenderID.



### 6.9.3.1. Background of “Page Separation Learnset”

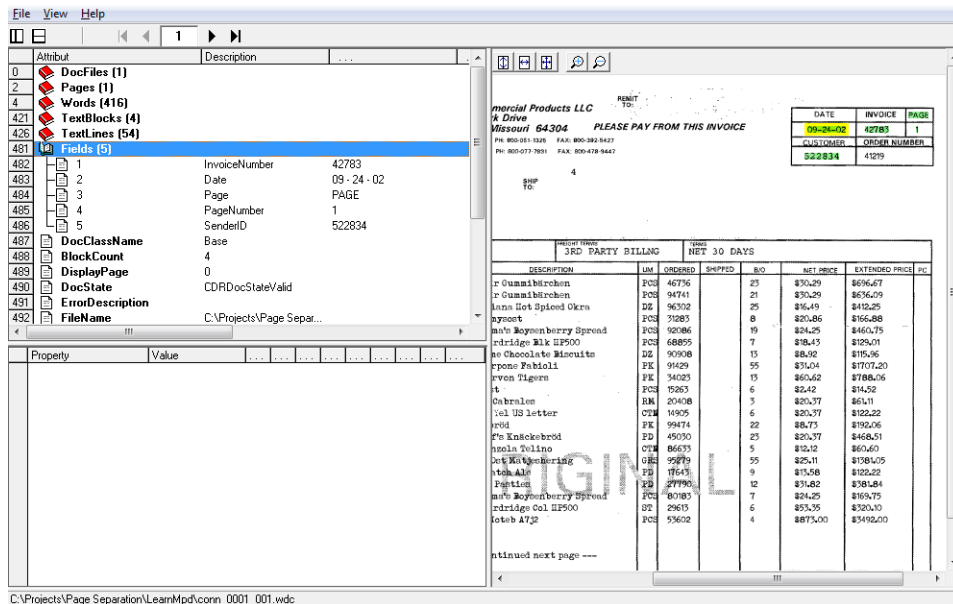


Figure 6-5: Page Separation Learnset

**InvoiceNumber** – A principle identifier of a multi-page document. If the Invoice number changes between two pages, a new document will start.

**Date** – Another criterion that can strengthen the determination of a multi-page document.

**Page** – This field has to contain the page keyword to strengthen the page number extraction. For example, the Page keyword can be “Page” or “Pg.Nr.” or something similar – whatever is present on the document.

**PageNumber** – If the document has pagination, a new document can be recognized when the pagination starts over.

**SenderID** – The identifier of an invoice type. For example, the tax number of the vendor can be used to identify a unique invoice of multiple pages from the same vendor. If the SenderID changes between two pages, this signals that a new document has started.

*If some fields are not present on the trained document, simply leave these fields empty.*

### 6.9.4. How to Train the Engine

Here is an example as to how the Page Separation engine can be trained. The example shows how one document that consists of two pages is supposed to be prepared to be added to the Learnset using the Verifier application. Each page of the document has to be represented as a separate document. Open the first document (First page) in Verifier and enter the fields using selection tool feature (Position of the entered fields is essential in terms of further learning):

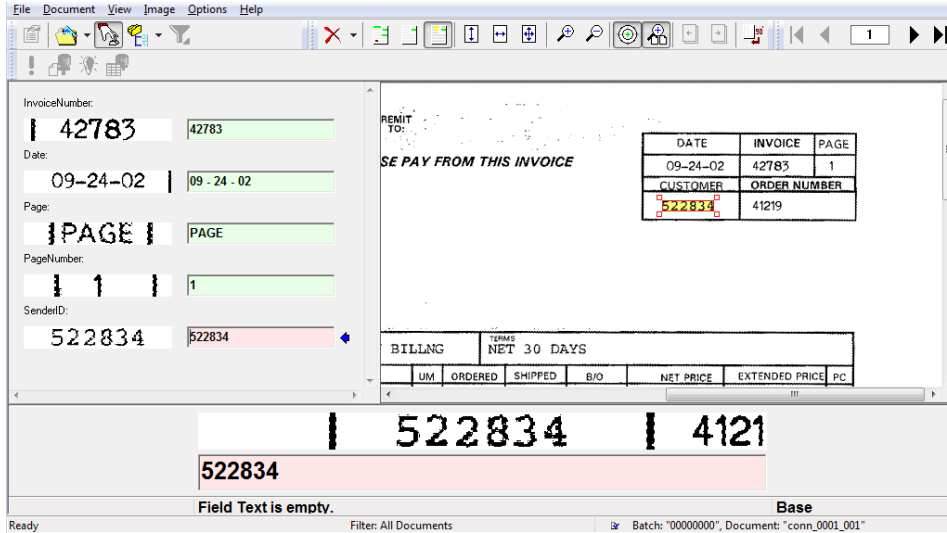


Figure 6-6: Preparation of a document for a Learnset

The next page has to be trained in the same way, as a separate document:

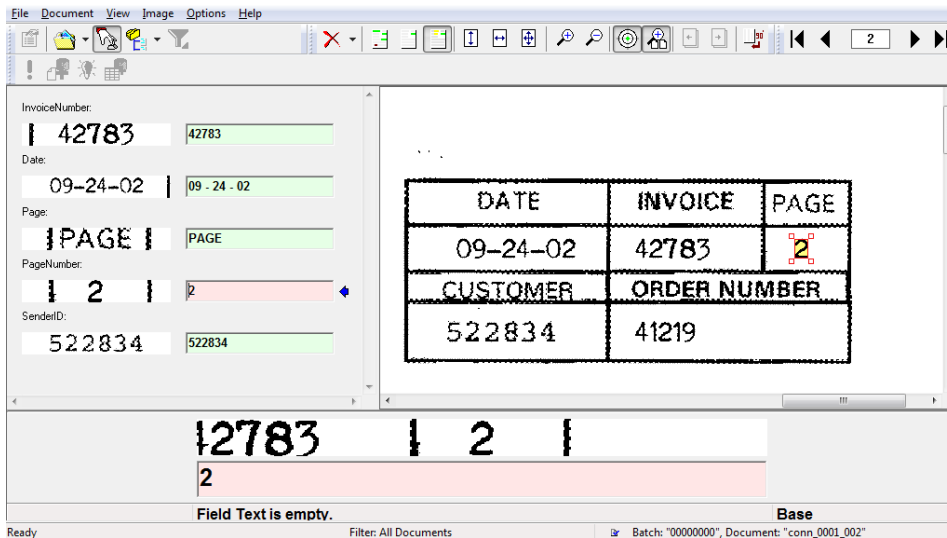


Figure 6-7: Preparation of a document for a Learnset – next page

### 6.9.5. Project Properties for Page Separation

To prepare a new project for the “Page Separation” feature, only a few steps are necessary.

#### 6.9.5.1. Step 1: Insert Base class

In Definition Mode (Tab classes), insert a single base class. This base class needs no definitions for classification. It will be defined as the default class in the project properties:

- Right click on the project name in classes tab
- Select *Show Properties*
- Select *Classification* tab on the right side of the Designer window.
- Select the name of the new base class as “default classification result”.

This default setting is necessary, because the “page separation” feature will be executed in the extraction step. No special extraction settings are necessary.

### 6.9.5.2. Step 2: Activate Multi-Page Detection

For the activation of the multi-page detection, right click on the project name in the classes list of the definition mode. Select *Show Multipage Detection* from the context menu. On the right side of the window you will now see the *Multipage Detection* properties. In the *General* tab, select this feature by checking the *Activate Multipage Detection* checkbox.

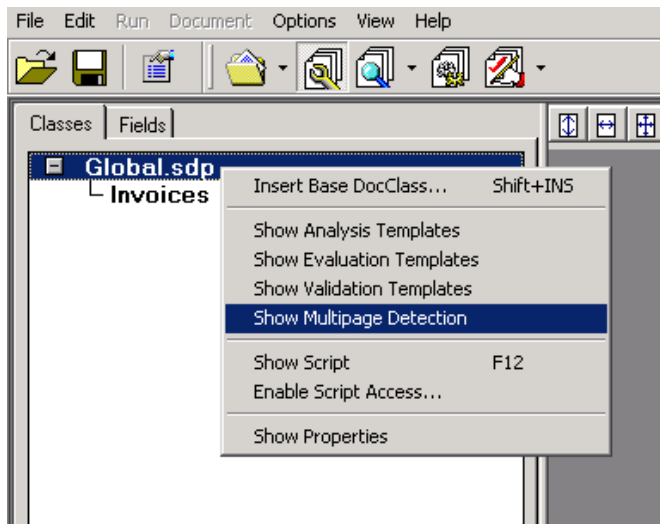


Figure 6-8: Activation of multi-page detection

Ignore the tab *Phrase Multipage Detection*. For purposes herein, the “Triton ADS engine” is being used. Activate the “Triton ADS engine” by checking the *Activate* checkbox on “Triton ADS engine” tab.

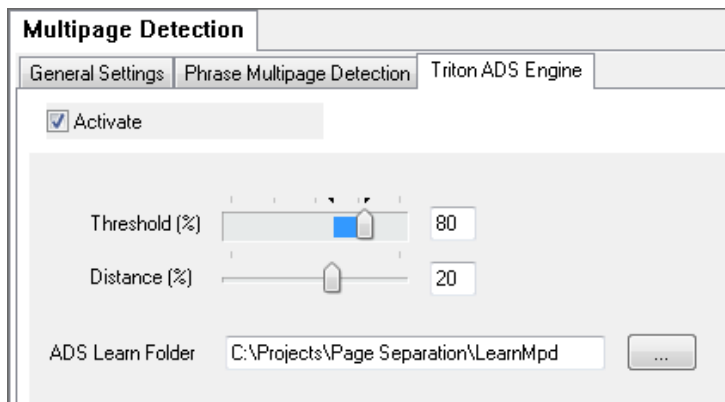


Figure 6-9: Activation of Triton ADS engine

### 6.9.5.3. Step 3: Define Path to ADS Learnset

To complete the projects configuration for automatic document separation, enter the path to the ADS Learnset (See section [6.9.3 PAGE SEPARATION LEARNSET](#)). Save the project file.

## 7 Setting Up the Validation

### 7.1 Validation Engine

Older versions of WebCenter Forms Recognition offered basic validation capabilities. This out-of-the-box basic validation was supported by powerful scripting capabilities. But because these scripts were written and implemented for specific customer sites, scripting efforts were often duplicative, and the quality was sometimes inconsistent – varying based on the script authors' coding experience. Maintaining scripts on an implementation-by-implementation basis was also difficult. Thus, validation was subject to significant enhancements.

WebCenter Forms Recognition's robust standard validation and output formatting is intended to standardize the development and maintenance cycles for each implementation of WebCenter Forms Recognition. However, standard validation has not replaced scripted validation. Validation scripts are still supported. For a brief introduction to script validation, please refer to section [INTRODUCTION TO VALIDATION SCRIPTS](#). For detailed information about scripting, please refer to the ***Scripting Reference Guide***.

### 7.2 Validation Concepts

Validation is a quality assurance task that involves confirming whether a field extraction is correct or incorrect. Validation is configured in Designer's Validation Editor, but the actual work is done in Runtime Server, Verifier, and Advanced Verifier.

Runtime Server loads the project at runtime and executes validation after extraction but before exporting the project. Documents that fail validation in Runtime Server are forwarded to Verifier for quality assurance. Verifier loads the settings and uses property sheets to display the validation settings. Validation is done with the Standard Validation Engine – currently the only engine available for validation.

*All Validation settings are established with respect to the Validation engine. This means that the Standard Validation Engine must be enabled for validation settings to be made or used.*

Validation uses WebCenter Forms Recognition's hierarchical organization of relationships between parent and child elements. These hierarchies make WebCenter Forms Recognition customizable and useful, but they also make the software more complex.

While Runtime Server is processing a batch or list of batches, each containing several documents, it first tries to classify the processed document. After classification, the extraction is performed for each field defined for this document class. The validation rules for a field can either be defined for this document class but can also be inherited from the parent class.

A document will be valid when all fields on the document are valid. If one field is invalid according to standard validation or script validation rules, the whole document is invalid. The script validation for a field is performed only when the standard validation is valid. If one document of a batch is invalid, the whole batch is passed to the Verifier for QA.

If you are familiar with previous versions of WebCenter Forms Recognition, you probably know about how templates can be used for Analysis and Verification. Templates can now also be used for Validation. This chapter shows you how to reuse settings for Validation that were established in your current project or in another project within your enterprise.

To learn more about Analysis and Evaluation templates, see [REUSING PROJECTS SETTINGS WITH TEMPLATES](#).

## Reusing projects Settings with Templates

The standard validation and output formatting is intended to standardize the development and maintenance cycles for each implementation of WebCenter Forms Recognition. However, standard validation has not replaced scripted validation. Validation scripts are still supported. For a brief introduction to script validation, please refer to section [INTRODUCTION TO VALIDATION SCRIPTS](#). For detailed information about scripting, please refer to the **Scripting Reference Guide**.

Script validation can override standard validation settings.

*This section introduces you the concepts to understand, and to use, the Standard Validation Engine and the Validation Editor effectively. Section [WORKING WITH VALIDATION LEVELS](#) and section [WORKING WITH VALIDATION TEMPLATES](#) show you how to use validation. Section [INTRODUCTION TO VALIDATION SCRIPTS](#) is a brief introduction to WebCenter Forms Recognition's scripting language WinWrap Basic.*

### 7.2.1. Levels of Validation

Generally, validation properties are inherited along the parent-child hierarchy. Children automatically inherit their parent's validation settings. You can accept inherited settings or override them. Inherited settings in the project, class, or document are homogenous, whereas overrides are useful for heterogeneous constructs.

There are three levels of validation:

- Project (For projects, the only validation setting you can make is whether or not to permit forced validation. You also store global settings using validation templates at this level.)
- Class and Document.
- Field (Includes text field and table field)

### 7.2.2. Terms and Commands You Should Know

#### Standard Validation Engine:

The Standard Validation Engine enables you to use a GUI to establish robust validation rules for documents or classes, and fields. The engine must be enabled in order for any validation to occur, or for you to establish validation settings. See section [STABILIZATION & ENHANCEMENTS OF THE STANDARD VALIDATION ENGINE](#).

#### Force Validation – permitted, forbidden, default:

Force validation allows you to force a field to be valid, even if it is invalid by definition. If force validation is permitted, the entity being validated is always valid, regardless of its content or specific settings. Default force validation means that the validation settings are taken from the parent. See section [TYPES OF FIELD VALIDATIONS](#).

#### Derived Validation:

Tell a child field, document, or class to inherit all validation settings from its parent. Derived validation can be used only on trained classes. See section [TYPES OF FIELD VALIDATIONS](#).

**General (tab):**

The General tab is available for all validation levels, although its contents differ depending on the entity to be validated.

**Character Filtering:**

The Character Filtering tab is available for all types of validation. This tab enables you to indicate which characters are valid and which are not, which characters should be removed during validation, and which should be replaced during validation. This tab also enables you to establish OCR settings and case conversion.

By default, all special characters are indicated for removal for date and amount fields, but you can modify character filtering to add other characters to the list. If the *Remove Characters* field is blank, every character is acceptable.

Characters that can be replaced are those that are usually misinterpreted during OCR. The default values are shown in below:

Character to be replaced	Replacement character
O	0
B	8
i (lower case i)	1
l (lower case L)	1
l (upper case l)	1

Table 7-1: Default characters to be replaced during OCR

You can add new characters to the list, or change the way the characters are interpreted.

The final character filtering settings govern how the characters are handled by the OCR engine.

**Output formatting.**

Output formatting enables you to standardize how data output from WebCenter Forms Recognition will appear when it is exported. This is useful for adding homogeneity to the appearance of output results, or conforms to your enterprise’s requirements for data storage. Output formatting is available only for date and amount fields. Output formatting must be specifically enabled at the field level. See [TYPES OF FIELD VALIDATIONS](#).

**Output formatting for dates:**

After you enable output formatting, select a region. This selection enables a list of standard date formats used in that region. You can add to the list by typing a value in the *Format* combo box.

<b>Input Conversion</b>	<i>No Conversion:</i> Characters are not changed. <i>Upper Case:</i> All characters in the field are converted to upper case. <i>Lower Case:</i> All characters in the field are converted to lower case.
<b>No Rejects</b>	Accepts all OCR results, regardless of quality.
<b>OCR confidence level</b>	Determines how good the OCR must be in order for the content of the document to be validated. A low confidence level mandates excellent OCR quality.
<b>Valid</b>	Enables you to set which characters will be accepted as valid. By clicking the

<b>characters</b>	All Numbers button, you can accept all numbers as valid. By clicking the All Characters button, you can accept all non-numeric characters; by clicking All, you accept all characters as valid. You can also select or unselect individual characters or groups of characters by clicking on them.
-------------------	--

Table 7-2: Available settings

**Output formatting for amounts:**

After you enable output formatting, select a region. You can then set values for currency, positive and negative formats, decimal symbols, number of digits after the decimal, digit grouping symbol and digit grouping.

**7.2.3. Available Validation Settings**

Available validation settings vary based on whether you are establishing settings at the project level, the class level or document level, or the field level. Basically, the more specific the level, the more specific and complex the validation rules are.

**7.2.4. Types of Field Validations**

Five types of fields can be validated: Text, Amount, Date, Checkbox, and List.

Each of these types of fields can be validated for certain general characteristics, although the type of validation varies from field to field.

Character filtering is available for all types of fields.

Output formats can be set for amount and date fields, but not for others.

Setting	Description	Default value
Force Validation	Default: Accepts the parent-level settings. Permitted: Allows force validation, regardless of the validation settings on the parent. Forbidden: Prevents force validation, regardless of the settings on the parent.	At project level: Forbidden. At child level: Default.
Available Templates	This selection box allows you to choose a template to validate the field with.	At project level: Blank if no templates exist for project; otherwise all templates available for the specific field type are shown.
Available Validation Engines	Only Standard Validation can be selected.	
Copy	Click this button to copy global settings to create local settings based on global settings.	NA
Save	Click this button to save global settings to create local settings from global settings.	NA
Use derived validation	Select this checkbox to use validation settings from the parent.	At project level: NA At child level: Not checked; only available for trained classes
Always Valid	Select this checkbox to make the field always valid regardless of its	Not checked.

Setting	Description	Default value
	contents or validation settings. No validation takes place.	

The types of validation you can perform vary with the type of field validation you selected.

Validation Type	General Validation	Character Filtering	Output Formatting
Text	Yes	Yes	No
Table	Yes	No	No
Amount	Yes	Yes	Yes
Date	Yes	Yes	Yes
Checkbox	Yes	Yes	No
List	Yes	Yes	No

Table 7-3: Validation Types and Available Validation Settings

#### 7.2.4.1. Text

##### General Settings for Text

General settings specific to amounts are shown below:

Setting	Description	Default value
Multi-line	Converts multi-line data to a single, non-delimited line.	Not checked.
Allow empty field	Field text can be empty.	Not checked.

Table 7-4: Settings for Text fields

##### Character Filtering for Text

The only setting enabled by default is *No Rejects*.

Output Formatting for Text

Output formatting is not applicable for text.

#### 7.2.4.2. Amount

##### General Settings for Amount

General settings specific to amounts are shown below:

Setting	Description	Default value
Amount Range	Establishes setting for validating a range of values.	Unchecked.
Minimum Value	Fields with values below this range will not be validated.	0.00



Setting	Description	Default value
Maximum Value	Fields with values below this range will not be validated.	0.00
Region	WebCenter Forms Recognition supports 27 regional settings, derived from default written languages.	U.S. English
Currency Symbol	Selections vary based on the region setting. For example, if you chose U.S. English for region, the symbols for U.S. currency are available. If you chose Canadian English, the symbols for Canadian currency are used. All currency formats for the region are accepted as valid.	Depends on regional settings.
Positive Format	Uses the currency settings for the region you selected and standard currency formats for depicting positive numbers.	(currency symbol)n.n Example: 1.1
Negative Format	Uses the currency settings for the region you selected and standard currency formats for depicting negative numbers.	[(currency symbol)n.n] Example: (1.1)
Decimal Symbol	By default, the decimal symbol is a point, but you can type in another symbol and have it available in the list for later use. The character used for decimal symbol cannot be the same character used for digit grouping symbol.	
Number of Digits after decimal	You can specify the number of digits that may appear after the decimal symbol. This is useful for truncating decimal characters.	two
Digit grouping symbol	The default setting is a comma, but you can also type in your own setting. This symbol cannot be the same as the Decimal Symbol.	,
Digit grouping	You can elect to have two or three characters per group or not to group digits at all.	none

Table 7-5: General Settings for Amount fields

### Character Filtering for Amounts

The following characters will be removed by default:

: ; ' & ? ~ ! @ # % ^ \* .

You can add to this list or delete from it.

### Output Formatting for Amounts

Setting	Description	Default value
Region	WebCenter Forms Recognition supports 27 regional settings, derived from default written languages.	U.S. English
Currency Symbol	Selections vary based on the region setting. For example, if you chose U.S. English for region, the symbols for U.S. currency are available. If you chose Canadian English, the symbols for Canadian currency are used. All currency formats for the region are accepted as valid.	Depends on regional settings.
Positive Format	Uses the currency settings for the region you selected and standard	(currency symbol)n.n

Setting	Description	Default value
	currency formats for depicting positive numbers.	Example: 1.1
Negative Format	Uses the currency settings for the region you selected and standard currency formats for depicting negative numbers.	[(currency symbol)n.n] Example: (1.1)
Decimal Symbol	By default, the decimal symbol is a point, but you can type in another symbol and have it available in the list for later use. The character used for decimal symbol cannot be the same character used for digit grouping symbol.	

Table 7-6: Output Formatting for Dates

### 7.2.4.3. Checkboxes

Setting	Description	Default value
Checkbox Caption	Enables you to indicate whether a checkbox should be captioned Yes/No, Accept/ Decline, True/False, or some other value.	Yes/No
Checked Value	Enables you to indicate whether a checkbox equals Yes, True, Accept, or some other value when it is selected. This setting should not conflict with Checkbox Caption.	Yes
Unchecked Value	Enables you to indicate whether a checkbox equals No, False, Decline, or some other value when it is cleared. This setting should not conflict with Checkbox Caption.	No
Default (Checked/ Unchecked)	Enables you to indicate whether a checkbox should be selected or cleared by default.	Checked

Table 7-7: Settings Available for Validating Checkboxes

### Character Filtering for Checkboxes

Character filtering is available for checkboxes. The only setting that is enabled by default is for No Rejects.

### Output Formatting for Checkboxes

Output formatting is not applicable for checkboxes.

### 7.2.4.4. Lists

Setting	Description	Default value
Min Len	Establishes a minimum valid field length	Not enabled/0
Max Len	Establishes a maximum valid field length	Not enabled.
Insert/delete list items	Create or remove list rows.	NA
Allow list values only	Only values that appear on the list will be interpreted as valid.	Enabled

Setting	Description	Default value
Fill list of suppliers		Not enabled.
Sort list items	Arranges items on the list in alphanumeric order.	Not enabled.

Table 7-8: Validation Settings Available for Lists

### Character Filtering For Lists

The only setting enabled by default is for *No Rejects*.

### Output Formatting for Lists

Output formatting is not applicable for lists.

#### 7.2.4.5. Dates

### General Settings for Dates

The table below shows the general validation settings available for dates:

Setting	Description	Default value
Enable Date Range	Dates within the range are interpreted as valid.	Enabled
From Date	Earliest valid date	Today's date
To Date	Latest valid date	Today's date
No future date permitted	Dates in the future are not valid.	Enabled
Sample	Uneditable field showing how the date will be displayed based on regional and format settings.	Depends on regional and format settings
Region	WebCenter Forms Recognition supports 27 regional settings, derived from default written languages.	Derived from system settings
Formats	Formats dates using a list of valid formats that are acceptable for that field.	Most common format for selected region.

Table 7-9: Validation Settings Available for Dates

### Output Formatting for Dates

Below table shows the output formatting settings that are available for dates.

Setting	Description	Default value
Enable Output Formatting	Select this checkbox to customize the field text using output formatting.	Not Checked.
Region	One of the 27 regional settings derived from the languages that WebCenter Forms Recognition supports.	U.S. English

Setting	Description	Default value
Format	Sets output formats for date strings.	Most common format for the selected region.
Sample	Uneditable field showing how the settings for format will actually appear.	NA

Table 7-10: Output formatting for dates

### Character Formatting for Dates

For dates, the following characters are removed by default:

: ; " ' & ? ~ ! @ # % ^ \*

You can add or delete characters for this setting.

#### 7.2.4.6. Table

### General Settings for Tables

Table validation is another type of field validation. Below table shows the Validation settings available for table fields.

Setting	Description	Default value
Type	Indicates whether the table column should be validated as text, amount, checkbox, list, or date.	Text
Template	Assigns a template to the field.	Not Set
Read Only	Determines whether data in the table row can be modified.	Not Set

Table 7-11: Validation Settings Available for Tables

### Character Filtering for Tables

Character filtering is not applicable for tables.

### Output Formatting for Tables

Output formatting is not applicable for tables.

## 7.3 Working with Validation Levels

### 7.3.1. Working with Project-Level Settings

The only validation settings available at the project level are whether to permit or forbid forced validation.

#### 7.3.1.1. Settings Available at the Project Level

At the project level, you can:

- Indicate whether force validation should be permitted or forbidden as a default for documents and fields. You can also assign Validation templates (See section [WORKING WITH VALIDATION TEMPLATES](#)).

## Prerequisites for Project-Level Validation

The prerequisites for this task are:

- The program is in Definition Mode.
- In the panel on the left side of the window, the *Classes* tab is active.
- On the right side of the window, the Project/Classification/Validation property sheet is visible.
- The *Validation* tab (the Validation Editor) is active.

### To set project-level validation

- On the Validation Editor, select whether to permit or forbid force validation for the project. This setting can be overridden for specific entities at lower levels of the parent/child hierarchy.

## 7.3.2. Working with Document- Level or Class-Level Validation

### 7.3.2.1. Settings Available at the Document or Class Level

At the document or class level, you can:

- Override or accept the parent-level settings for force validation.
- Turn on the Standard Validation Engine. Validation will not be done on the document or class unless the Standard Validation Engine is on.
- Accept all validation settings derived from a parent or from the child class
- Establish general validation rules for the document.

*At this level, you can establish validation rules using amount fields, date fields, and table fields. You cannot establish validation settings for any other type of fields.*

### 7.3.2.2. Prerequisites for Document- Level or Class-Level Validation

#### Task Prerequisites

The prerequisites for this task are:

- A document is selected and visible in the viewer in the middle of the window. (Note: The document does not have to be visible for the functionality to work, but having document in the viewer will make it easier to visualize this task.)
- The program is in Definition Mode.
- In the panel on the left side of the window, the *Classes* tab is active.
- On the right side of the window, the Document/ClassValidation property sheet is visible.
- The Validation Editor is active.

You can also save your settings as a template by clicking *Save As Template*.

### 7.3.2.3. Managing Document-Level or Class-Level Validation

To set document-level or class-level validation:

- 1) On the left side of the screen, select a document or class.
- 2) On the Validation Editor, select Standard Validation Engine. If you do not select Standard Validation, no validation will be done on the document or class.
- 3) For Force Validation, select *Default*, *Permitted*, or *Forbidden*.
  - If you select *Default*, the setting for Force Validation for document is inherited from the parent or project.
  - If you select *Permitted*, Force Validation is permitted for the document or class as a whole, regardless of whether or not you permitted it on the parent.
  - If you select *Forbidden*, Force Validation is prevented for the document or class, regardless of whether you permitted or forbade it on the parent.
- 4) Select a document-level template. You can also copy a template or create a template by saving your settings as a template. For more information about working with Validation templates, see section [WORKING WITH VALIDATION TEMPLATES](#).
- 5) If you selected Standard Validation Engine in Step 1, the General Validation Rules tab was enabled.

#### 7.3.2.4. Establishing Validation Rules at the Document Level or Class Level

*This functionality works only on Amount fields, Date fields, and Table fields.*

On the *Classes* tab on the left side of the screen, select a class.

On the Validation Editor, make sure the *Standard Validation Engine* is selected. When it is selected, the General secondary tab is enabled. This is the tab you will use to create validation rules at the Document/Class level. These rules enable you to compare one field to another, establish data criteria, and perform certain mathematical and logical functions.

The correct syntax for an expression is illustrated in [FIGURE 7-1](#).

To build an expression, select a field, click one of the Function buttons (Sum, Max, Min, Count, Avg, or ABS), click an operator, and then parameters. Click *Add* to move the expression to the list of Validation Rules.

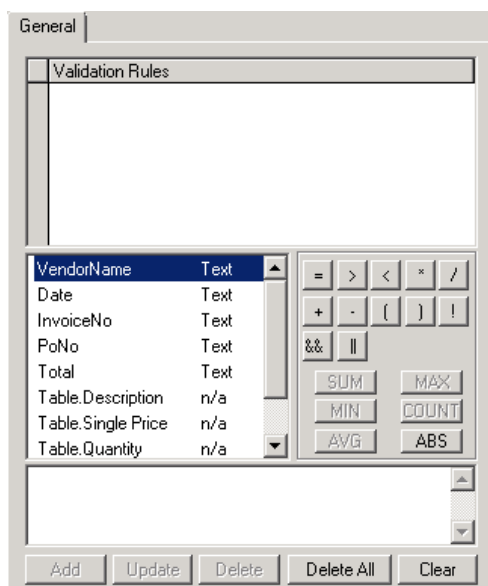


Figure 7-1: Establishing General Validation Rules






Button	Description
	Equals
	Greater than
	Less than
	Multiplication
	Division
	Plus
	Minus
	Open grouping for expression
	Close grouping for expression
	Not equal (must always be followed by an equal sign)
	AND
	OR
	Sums all table rules for a column
	Sets the maximum value for a column
	Sets the minimum value for a column
	Counts the number of rows in a column
	Averages table rows
	ABS function

Table 7-12: Controls for validation rules

### 7.3.3. Working with Field-Level Validation for Text

Field-Level Validation allows you to accept the parent-level settings for forced validation, turn on the Standard Validation Engine, and set general validation rules for the selected field.

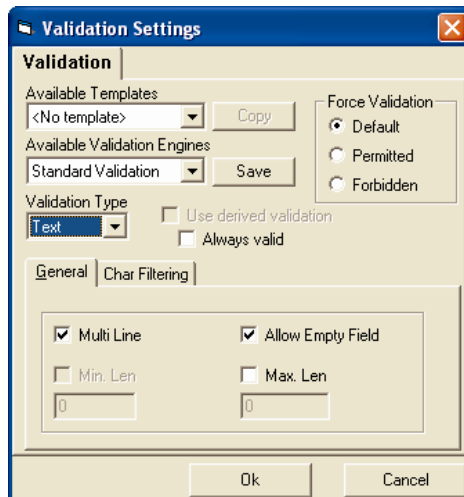


Figure 7-2: Text Field Properties dialog box

### 7.3.3.1. Settings Available at the Field Level for Text Validation

At the field level for text fields, you can:

- Override or accept the document-level settings for forced validation
- Turn on the Standard Validation Engine. Validation will not be done on the selected field unless the Standard Validation Engine is on.
- Establish general validation settings and character filtering for the selected field.

At the field level for table fields, you can:

- Accept the parent-level settings for forced validation
- Turn on the Standard Validation Engine. Validation will not be done on the selected field unless the Standard Validation Engine is on.
- Set general validation rules for the selected field.

#### Task Prerequisites

The prerequisites for this task are:

- The program is in Definition Mode.
- In the panel on the left side of the window, the *Fields* tab is active.
- On the right side of the window, the *Analysis/Evaluation/Field/Validation* editors are visible.
- The *Validation* tab is active.

#### Setting Validation for Text Fields

To set field-level validation on text fields:

- 1) On the left side of the screen, make sure the *Field* tab is active.
- 2) Select a text box.
- 3) On the Validation Editor, select either the *General* tab or the *Character Filtering* tab.



### 7.3.3.2. Setting Validation for Table Fields

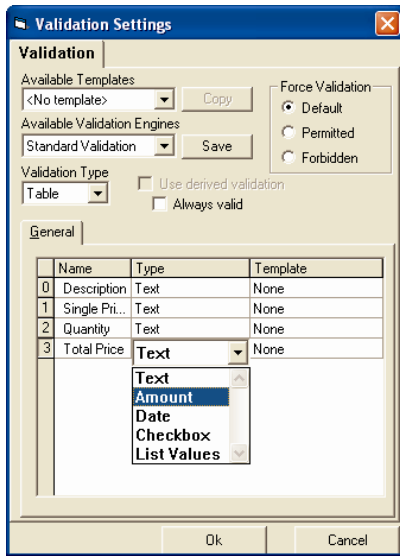


Figure 7-3: Table Field Properties

To set field-level validation on table fields:

- 1) On the *Fields* tab on the left side of the screen, right click on a field.
- 2) Click *Table* to change the field from a text field to a table field.
- 3) On the Validation Editor, establish the validation rules for the field.
  - Make sure that the Validation Type is set to Table.
  - For Force Validation, select *Default*, *Permitted*, or *Forbidden*.
  - Decide whether to check *Use Derived Validation* (available only for trained classes.)
  - Decide whether to check *Always Valid*. This forces WebCenter Forms Recognition to interpret all columns in the table as valid, regardless of their contents or any inherited settings. It also prevents you from establishing specific settings for each row.
  - On the *General* tab, notice that the columns of your table are available. Select a validation type for each row, and if available, select a template.

## 7.4 Working with Validation Templates

If you are familiar with earlier versions of WebCenter Forms Recognition, you know how templates can be used for analysis and evaluation. Validation templates are based on a similar concept.

*As with analysis and evaluation templates, you can and should reuse validation templates within a project or globally across projects.*

Before you can use validation templates, they must be available in your project. If none is available, you will need to either import them or create them.

### 7.4.1. Creating Templates

You create templates for specific kinds of validation at the field level and then apply them at various parent or child levels throughout the project. You can apply templates at the field level, the document/class level, or the project level.

#### Task Prerequisites

The prerequisites for creating templates are:

- A document is selected and visible in the Viewer. (The document does not have to be visible for the functionality to work, but being able to see the document will help you visualize how the validation templates should be created.)
- The program is in Definition Mode.
- On the left side of the screen, the *Fields* tab is active.
- On the right side of the screen, the Validation Editor is active.

#### Creating Validation Templates for Fields

To create validation templates for fields:

- On the *Fields* tab on the left side of the screen, select a field.
- On the *Validation* tab, establish validation settings for the field.
- Click *Save*.
- Give the template a descriptive name and click *OK*.

#### Creating Validation Templates for Classes

*Template names are case-sensitive.*

To create validation templates for fields:

- On the *Classes* tab on the left side of the screen, select a class.
- On the *Validation* tab, establish validation settings for the class.
- Click *Save*.
- Give the template a descriptive name and then click *OK*.

### 7.4.2. Working with Validation Templates at the Field Level

You can apply a field validation template created in another document in your project to any field in any document within the same project.

To do this, first select a document and then select a field on the document. Make sure you have the Validation Type set correctly for the field; the available templates depend on the Validation Type. For example, Amount templates are not available for Text fields.

Select a Validation Type for that field and click *Copy*. This turns the template into validation settings for the specific field you are working on.

Any changes you make to the validation settings at this point will apply only to the field on hand, and not to the template. However, you can save the settings into a new template or overwrite the existing template.

Your template is now saved in the project file and can be used for validation of similar classes throughout the project, but only at the field level.

### 7.4.3. Working with Validation Fields at the Class Level

The templates you create at the class level are stored as custom templates.

### 7.4.4. Working with Validation Templates at the Project Level

To manage Validation templates at the project level, either:

- Right click on the project and select *Show Validation Templates* from the shortcut menu, or,
- On the *Edit* menu, select *Project*, and then select *Show Validation Templates*.

On the Validation Templates property sheet on the right side of the window, you can:

- Import templates from other projects
- Export templates for use in other projects
- Delete templates
- Rename templates.

#### Assigning Templates

To assign a template, first select a validation type. Next, select the template you want to assign for that field type. Any changes you make apply only to the project and not to the template itself. You must use Template Manager to actually change a template itself.

#### Importing Templates

To import a template, select a validation type and click *Import*. Browse to a template and click *OK*. Clear the checkboxes for the templates you do not want to import and then click *OK*.

*Any changes you make to an imported template apply only to the project.*

#### Exporting Templates

To export a template, click *Export*. Clear the checkboxes for the templates you do not want to export and then click *OK*. Browse to the root folder of the target project and click *Save*. All the selected templates are exported in bulk and saved as an \*.exp file.

#### Deleting Templates

To delete a template, select a Validation Type and a template to delete. Click *Delete*.

*You cannot delete a template that is in use.*

#### Renaming Templates

To rename a template, select a Validation Type and a template to rename. Click *Rename*.

*You cannot rename a template that is in use.*

## 7.5 Introduction to Validation Scripts

You may need to customize Validation beyond what you can do with the Standard Validation Engine. Designer includes the Visual Basic scripting language WinWrap Basic that can be used for validation, analysis, and evaluation tasks. To use WinWrap Basic effectively, you

must have a programming background. Experience in VB Script or VBA is helpful as WinWrap is VBA-compatible.

This section serves only as a brief introduction to script validation and its availability. For a complete discussion of this topic, including code samples, see the ***Scripting Reference Guide***.

*Note that script validation overrides standard, GUI-based validation.*

To open the script editor, switch to Definition Mode. Click the button for script editing in the toolbar or use the menu *Edit* → *Show Script*.

Script validation can be accomplished for projects, documents, fields, table cells, table rows, and entire tables.

*Scripting can be used to write warnings and informational messages to the Runtime Server log files.*

You might have older projects with scripts made using the Sax Basic engine. The two scripting engines Sax Basic and WinWrap Basic are generally compatible. However, WinWrap Basic is optimized, quicker and more robust. It supports Unicode strings in both internal and external functions.

*Note: From version 11.1.1.8.0 on, WebCenter Forms Recognition does not support the Sax Basic scripting engine.*

*Please refer to section [PROJECT WITH SAX BASIC TO THE WINWRAP SCRIPTING ENGINE](#) for information on handling older projects containing scripts created with Sax Basic.*

WebCenter Forms Recognition has been certified on the following Scripting component versions:

- WinWrap Version 9.0.0.56

## **7.6 Stabilization & Enhancements of the Standard Validation Engine**

### **7.6.1. Description**

This section describes the features of Standard Validation Engine available in the present version of WebCenter Forms Recognition.

#### **7.6.1.1. Feature to Control Amount of Invalidated Fields**

A project level option configurable in the Designer application, project node's settings, *Advanced* tab and called "When validation rule fails, mark only one field as invalid (otherwise all)" allows control of the amount of fields that are getting set to invalid in case a document level validation expression fails.

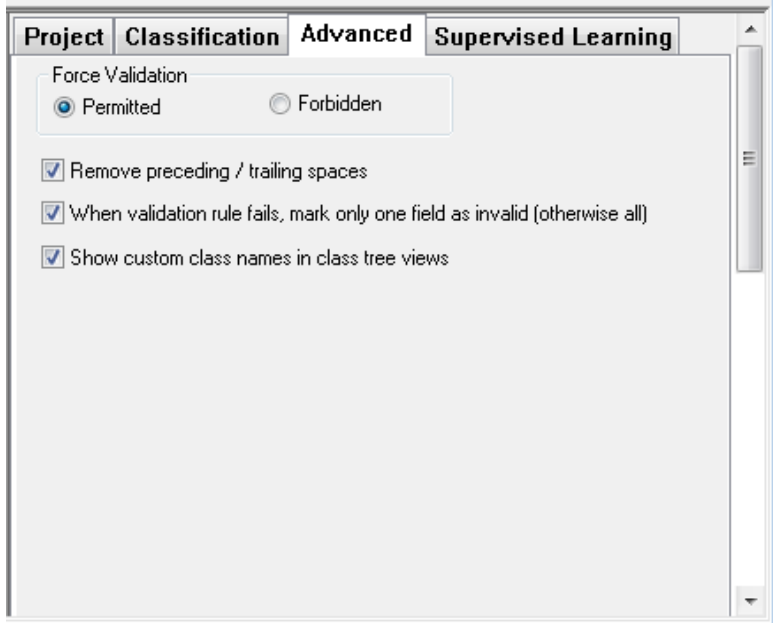


Figure 7-4: Advanced tab – options for control of the amount of invalidated fields

By default this option is switched off, i.e., a validation rule like “ $NetAmount + VAT = GrossAmount$ ” is going to fail (i.e., the expression is not getting equal for the currently processed document), the validation subsystem will make all three fields invalid. At the same time, if, e.g., from statistical stand point it is desired to make just one of these three fields invalid, this can be achieved by setting the above mentioned check box option:

**Note:** If the option is turned on, and a rule with both table columns and header fields fails (example: “ $SUM(Table.TotalPrice) = GrossAmount$ ”), then the validation subsystem is supposed to attempt to invalidate one of the available header fields in the expression and not the table cells.

**Note:** When making a table column invalid according to a document level validation rule (expression), the validation engine is supposed to mark only the first cell of the column as invalid. Exactly like it is shown on the screenshot below:

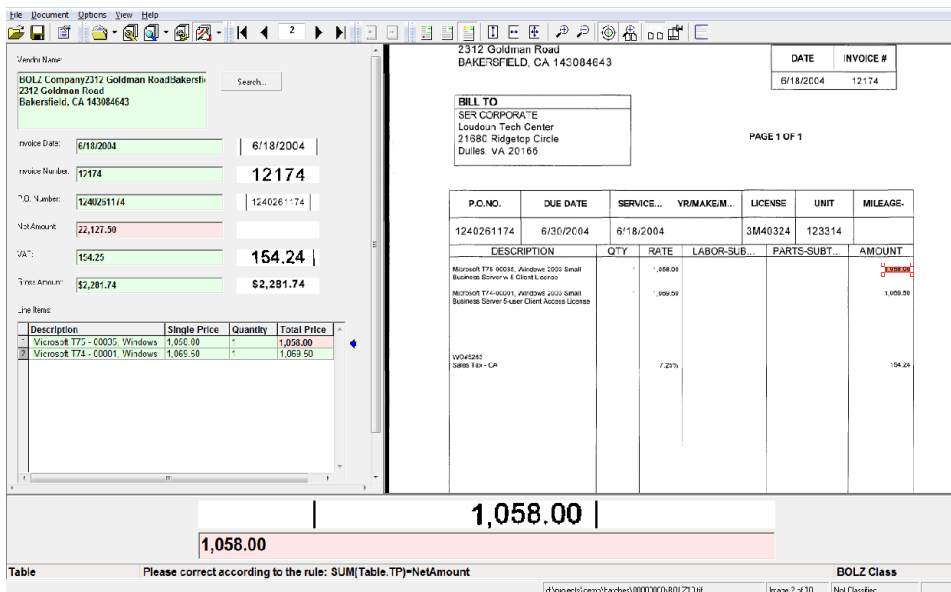


Figure 7-5: Document level validation rule – table column invalid

### 7.6.1.2. Error Descriptions in Case of Validation Failures

The validation descriptions point to the failed validation rule allowing the Verifier user to understand what went wrong quicker than before. For example, a rule like

“ABS (NetAmount+VAT-GrossAmount) <0.02”:

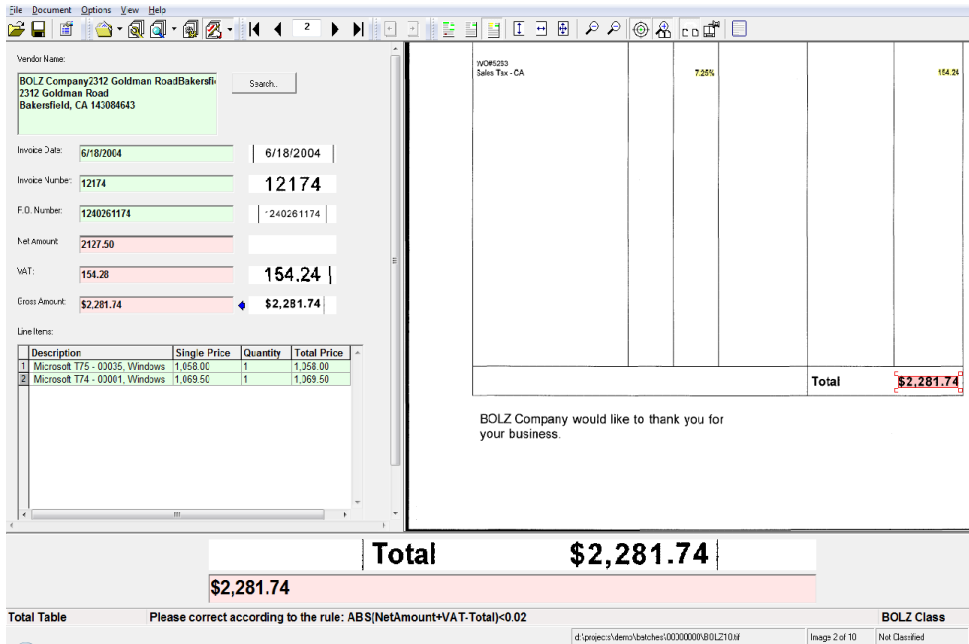


Figure 7-6: Validation Description

The validation engine provides a clear description like on the screenshot above: “Please correct according to the rule: ABS (NetAmount+VAT-Total) <0.02”.

Or, like it is shown on the screenshot in the section “Feature to Control Amount of Invalidated Fields” above, failure of the expression “SUM(Table.TotalPrice)=NetAmount” leads to an error message “Please correct according to the rule: SUM(Table.TP)=NetAmount”.

Another example is invalidation of the fields’ formats that proceeds to the document level validation:

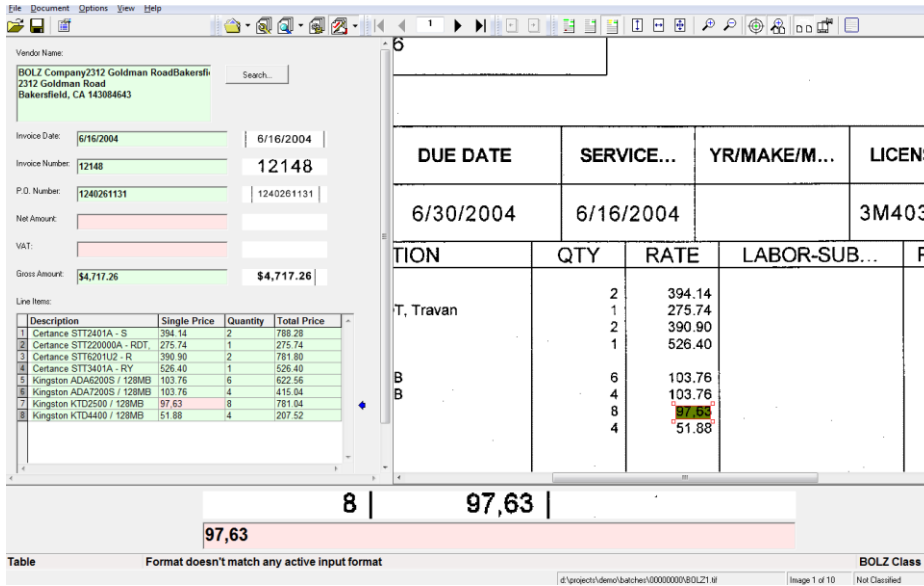


Figure 7-7: Error message for invalid fields

The cell's content "97,63" in the example above is invalid, because according to the defined amount template (in this project) only US format for decimal separator sign is allowed (i.e., "." and never - ",",). Now the system displays the appropriate error message "Format does not match any active input format" pointing to the incorrect content of this table cell that needs to be replaced with "97.63".

### 7.6.1.3. Commands for Document Level Validation Expressions (Rules)

An ABS command is available when configuring document level validation expressions (rules). This command evaluates absolute value of the enclosed expression and simplifies entering very common commands like (See "ABS" button on the screenshot below):

`" ( (NetAmount+VAT-Total) < 0.02) && ( (NetAmount+VAT-Total) > (-0.02) )"`

Up to a simple representation like:

`"ABS (NetAmount+VAT-Total) < 0.02"`.

Another important extension is about "!=" sign that can be used as "not equal" sign and about the "!" sign, which can now be also utilized as NOT operator.

*Note: The former "DATE" command has been removed as redundant.*

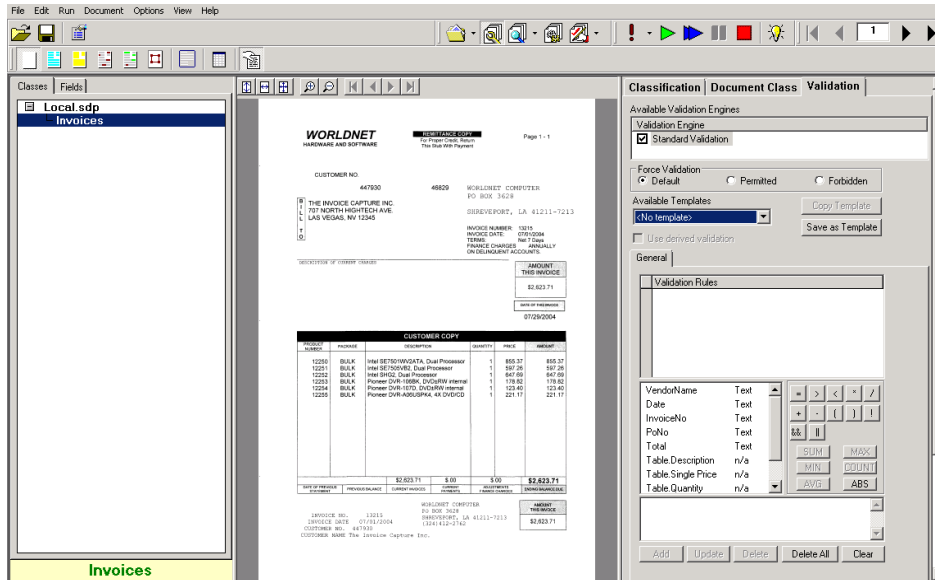
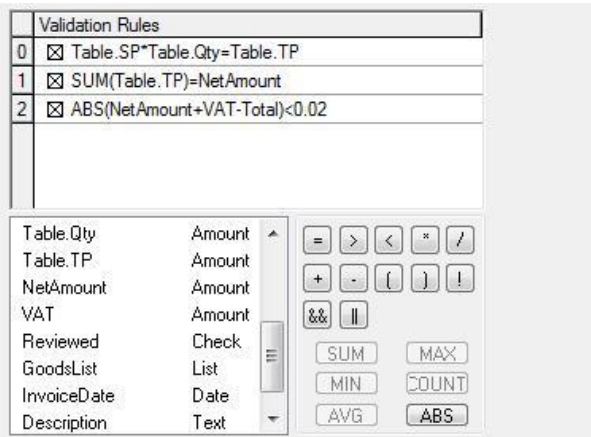


Figure 7-8: Validation settings for class level

*Note: When specifying constant values in the document level validation expressions (For example, "0.02" the one "ABS (NetAmount+VAT-Total) <0.02") always use English format for the decimal part of the real number. The integral part of the real number should be given without any delimiters in between. As well as the decimal part, which can include any reasonable amount of signs after the ".", for example: "20300.5467".*

### 7.6.1.4. New Features & Errors Message Available for Document Level Validation Rules Definition

When defining document level validation expressions (Rules) the Designer application's GUI provides a column supplying the list of available class fields with the type of the field, which can be either Text (for text field), Amount (for an amount field), Date (for a date field), List (for a drop down list field) or Check (for a check-box field) preventing the user from configuration errors (when, e.g., a Text field is to be used as part of a mathematical expression, which is not going to work):





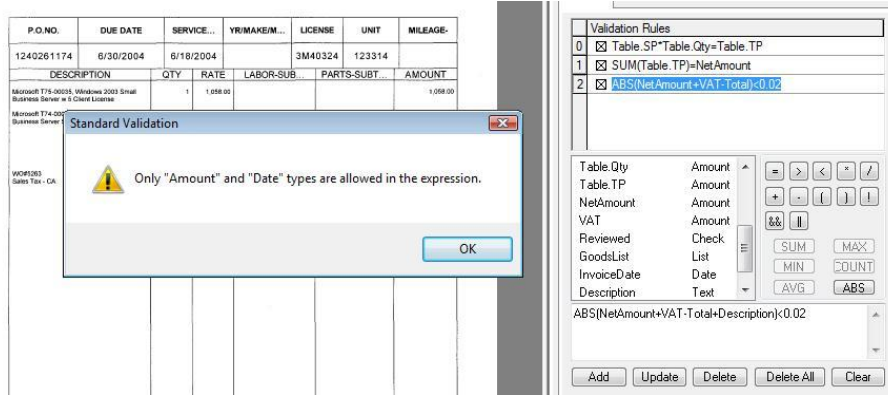


Figure 7-9: New GUI for definition of document level validation expressions

Also, when using a field in a validation expression, the validation subsystem provides appropriate warning messages. For example, a user attempts to use a “Text” field in an mathematical expression, the system will show a warning message “Only “Amount” and “Date” types are allowed in the expression”, that will not allow the user to update the expression by clicking *Update*:

In the past versions of WebCenter Forms Recognition usage of space characters (“#\_\*, +”, etc.) in names of table columns (when configuring Table Analysis or Brainware Table Extraction Engines’ setting) would lead to a serious implicit configuration error. This is no longer the case and all the above mentioned special characters, as well as many others are allowed.

### 7.6.1.5. Usage of Document Level Validation for the Inherited Classes

Document level validation rules are applied cumulatively from parent to child and not only for the affected class node.

If it is desired to apply document level validation rules for inherited classes, the “Use derived validation” option (see [FIGURE 7-10: “USE DERIVED VALIDATION” OPTION FOR INHERITED CLASSES](#)) has to be turned on for the required derived class(es). By default, for all newly created classes this option is supposed to be turned on automatically. For previous projects, where this option was OFF before, this remains unchanged after the migration to the new version in order to support backwards compatibility.

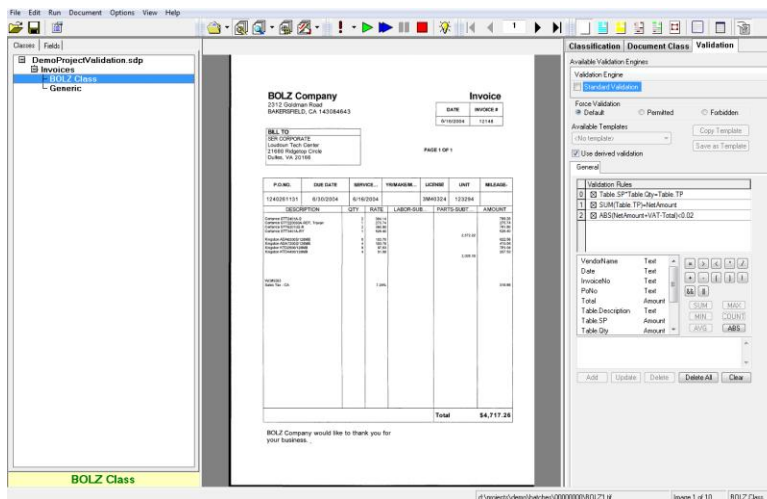


Figure 7-10: “Use derived validation” option for inherited classes

### 7.6.1.6. Definition of Complex Validation Formats

One or more complex validation formats can be defined via a new GUI implementation for the format definition of the Amount fields:

*Note: Clicking on the language selection drop down box allows to choose the desired basic validation format, which can be then extended with multiple further definitions, like it was for "1000 separator" (accepts both "," and "." characters) and other fields on the screenshot above. The basic format settings can be copied all at once by clicking on "<< All <<" button and then extended with the required custom ones.*

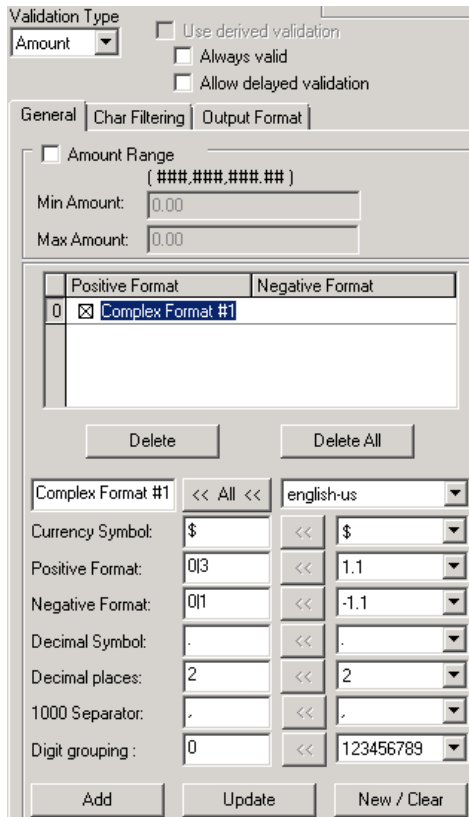


Figure 7-11: Format settings

**Notes:**

The edit box field at left of the "<< All <<" button represents the name of the currently edited format definition. Modify it if you would like to add an extra format (click Add).

"|" sign is used to separate the formatting variants.

When "no" expression is used as a formatting option, this means absence of any sign. For example, for the field "Currency Symbol" this means that currency value like "2,000.00" without currency symbol "\$" is going to be accepted.

With the buttons "<<" you can add custom (e.g., manually typed) formatting options to the corresponding cumulative format definitions.

Absence of any symbols in a cumulative format definition edit box (the ones at the left side) means that all options available in the corresponding drop down list at the right side are supposed to be supported.

### 7.6.1.7. Robust Unified Output Formatting

A robust unified output formatting can be configured on the “Output Format” property page for the Amount fields:

The screenshot shows the 'Output Format' property page for an 'Amount' field. At the top, there are checkboxes for 'Use derived validation', 'Always valid', and 'Allow delayed validation'. Below these are three tabs: 'General', 'Char Filtering', and 'Output Format'. The 'Output Format' tab is active. It contains an 'Appearance Sample' section with 'Positive' and 'Negative' fields showing formatted values. Below that are several dropdown menus for 'Region', 'Currency Symbol', 'Positive Format', 'Negative Format', 'Decimal Symbol', 'No. of digits after decimal', 'Digit grouping symbol', and 'Digit grouping'. At the bottom, there is a checked checkbox for 'Enable Output Format' and an 'Apply' button.

Figure 7-12: “Output format” property page

The resolved formatted value is stored in a special “FormattedText” string property of the SCBCdrField object. The same property is also available for table cells and can be accessed via new property:

*pTable.CellFormattedText (Column, Row)*

The properties can be accessed from within WebCenter Forms Recognition’s custom script from, e.g., “Document\_Export” or “Validate\_Document” events.

### 7.6.1.8. “Value” Property for Simplified Scripting of Extended Validation Routines

A “Value” property is available for both table cells and header fields and can be used to significantly simplify access to the automatically extracted data from within the WebCenter Forms Recognition custom script.

WebCenter Forms Recognition provides with new fields’ and table cells’ property “Value” that contains field’s value in the variable type depending on the field validation type. The “Value” property is “VARIANT” and returns date (for Date fields), currency (for Amount fields), Boolean (for Check-box fields), index (long value) of the list item (for the List fields) or text for the other field types.

The “Value” property for the header fields is accessible as “pField.Value” and for the table cells as “pTable.CellValue (Column, Row)”.

Below is a script sample that demonstrates usage of the new properties in the custom script:

```
Private Sub Document_Validate(pWorkdoc As SCBCdrPROJLib.SCBCdrWorkdoc, pValid As Boolean)
```

```
    Dim dblValue As Double
```

```
Dim strFormattedValue As String

dblValue = pWorkdoc.Fields.ItemByName("Total").Value
strFormattedValue = pWorkdoc.Fields.ItemByName("Total").FormattedText

MsgBox "Value = " & CStr(dblValue) & vbCrLf & "Formatted Text = " & strFormattedValue

End Sub
```

### 7.6.2. Usage

All these features are, first of all, important in terms of minimization of manual scripting required to code custom project's validation rules and formatting, which is currently a valuable part of any WebCenter Forms Recognition installation. Minimization of the scripts (including both GUI for math expressions and partial coverage of the validation business logic) is extremely important in terms of minimization of Professional Services efforts and, as a result, minimization of time and costs required to implement a WebCenter Forms Recognition project for an individual customer.

## 8 Setting Up the Data Extraction

Setting up the data extraction involves:

- Creating data fields
- Defining analysis methods to obtain candidates
- Defining evaluation methods to select the correct candidate
- Setting parameters for these methods
- Testing
- Optimizing

A candidate can only be assigned to a field if its weight exceeds a certain predefined threshold. You can influence the threshold values and the evaluation algorithms, but in most cases, the default settings should be fine.

For now, the following background knowledge is sufficient:

- The weight indicates the degree of similarity between the properties of a candidate and the properties of user-selected candidates.
- Data can be extracted if the weight of the best candidate with respect to a given field exceeds a predefined threshold. By default, this threshold is 50 percent.
- In general, there is only one successful candidate per field and document. If several candidates have a high weight with respect to a field and document, it must be possible to distinguish reliably. Therefore, a certain difference in weight between the winning candidate and the second-best competitor is required as well. By default, this so-called distance is 10 percent.

For more information about candidate evaluation, see section [FIELD-LEVEL SETTINGS](#).

### 8.1 Brainware Lines Extraction Method

#### 8.1.1. Description

The method can be switched on in the Designer application's setting dialog. In the Processing tab, select Brainware *line extraction* under *Line extraction settings* group box:

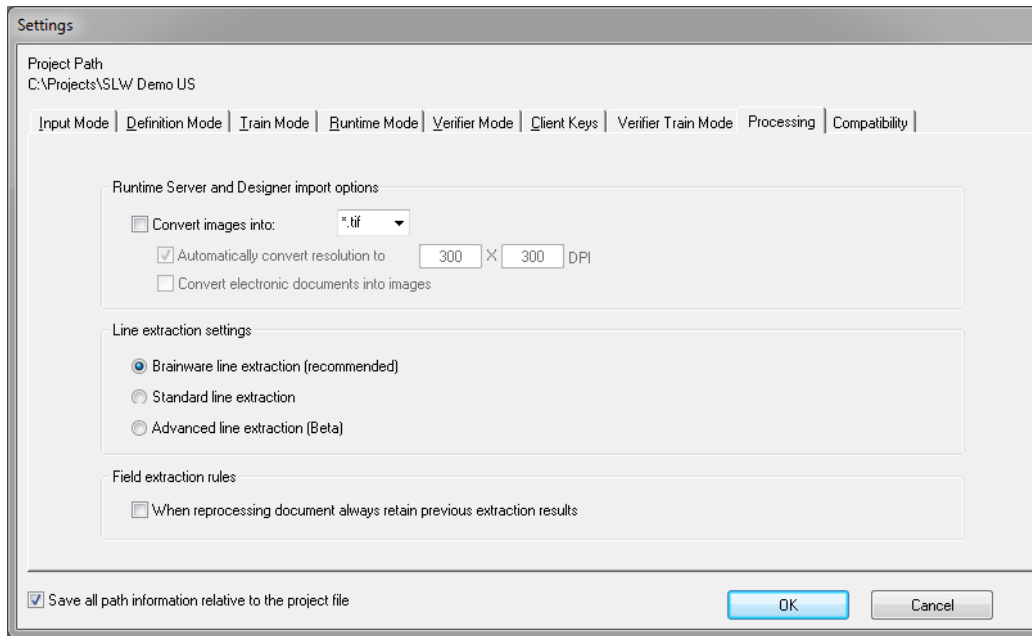


Figure 8-1: Processing tab – Brainware line extraction

The following line extraction engines are available:

- Brainware line extraction
- Standard line extraction
- Advanced line extraction (Beta)

*Note: This option is not recommended for use yet.*

When selected, the option affects all WebCenter Forms Recognition applications that invoke OCR processing or re-launch internal post-processing via the script statement:

```
pWorkdoc.RebuildBasicObjects
```

Same as the previously available “Standard line extraction” method, the one is invoked automatically right after applying OCR recognition and detection of word objects. If the method is activated, the standard (the old one) is not going to be applied at all as the method completely replaces it and delivers the results in the same way as the previous method of extracting the results as a collection of SCBCdrTextBlock objects accessible via WorkDoc’s interface from within the custom script as:

```
pWorkdoc.Textline lLineIndex
```

The method is activated by default for all newly created WebCenter Forms Recognition projects but is switched off when migrating projects from the old versions to ensure the backwards compatibility is unaffected in any way.

### 8.1.2. Usage

The method takes an advantage of the powerful Brainware Table Extraction Engine and provides significantly better quality of lines recognition as compared with the standard (old) lines extraction method. It is generally recommended to use the method instead of the old one. Moreover, in some cases, this may appear to be a mandatory setting (i.e., if for some project’s document samples the defined Format Analysis extraction delivers incorrect results

due to low-quality lines extraction). Note that the Format Analysis engine uses internal line objects for formats recognition.

The old (standard) extraction method is left in the system to ensure the backwards compatibility is not negatively affected in any way.

## 8.2 Setting up the Fields

### 8.2.1. Creating Fields

Fields are structures that hold the data extracted from the documents. They are set up on the class level. Fields and their properties are inherited along branches in the classification tree; that is, all classes derived from a parent will use the parent's field definitions.

At the child level, you can override field properties or add more fields, but you cannot delete inherited fields. Take this into account when defining your fields.

There are two types of fields:

- Text fields: They can be used with Format Analysis, Zone Analysis, and Address Analysis.
- Table fields: They can only be used with Table Analysis.

For each class, you can define one table field at most.

#### Task Prerequisites

The prerequisites for this task are:

- The program is in Definition Mode.
- The panel on the left side of the window displays the *Classes* tab in the foreground.
- The class you want to create fields for already exists.

#### Creating Fields for a Class

To create fields for a class:

- 1) In the *Classes* tab, double click the class you want to create a field for. The *Fields* tab is displayed in the foreground. If you create your first field, the tab is empty. Otherwise, previously defined fields will be displayed.
- 2) Right click within the tab's background.
- 3) From the shortcut menu, select *Insert Field Definition*. The *Add Field* dialog box is displayed.
- 4) Enter a field name. Valid names consist of alphanumeric characters without spaces or special characters. Later you can define a different display name in the Field properties (see section [CUSTOM FIELD NAMES](#)).
- 5) Click *OK* to confirm. New fields are inserted below existing ones in the order in which they were created.
- 6) To create additional fields, repeat Step 2 to Step 5.

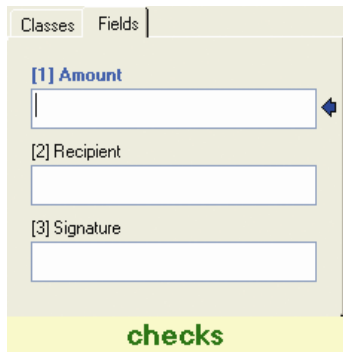


Figure 8-2: Fields created for the Checks class

## Table Fields

By default, the created fields are text fields.

To add a table to a form:

- 1) In the *Fields* tab on the left side of the window, right click to bring up the shortcut menu.
- 2) Select *Insert field*. When a dialog box appears to name the field, enter a field name that signifies a table. A new field appears in the *Fields* tab.
- 3) Right click the new field and select *Table* from the shortcut menu. The field now turns into a button with the table icon in the center.

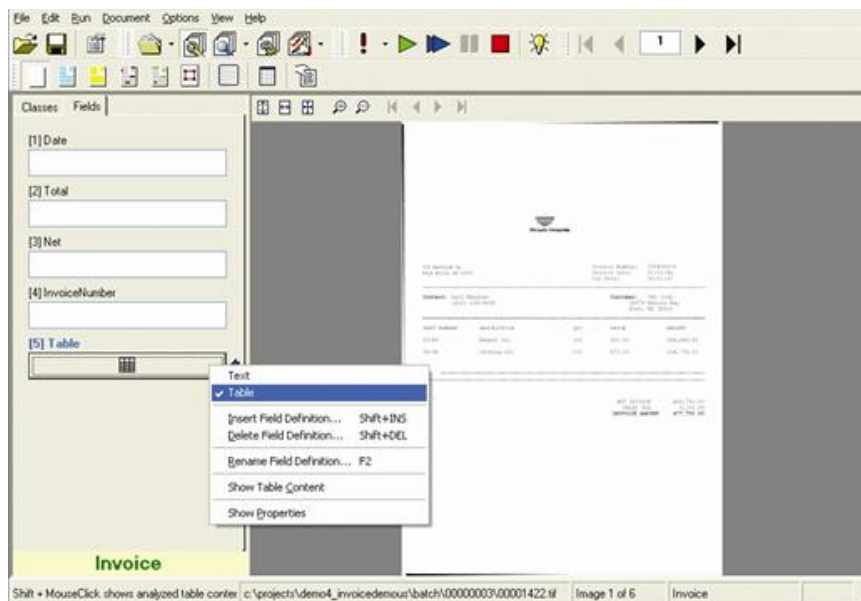


Figure 8-3: Selecting a Table Field.

- 4) Click *Table*.
- 5) In *Analysis Engines*, select the Table Analysis Engine.
- 6) See section [SETTING UP TABLE ANALYSIS](#) for information about defining table settings.



### 8.2.1.1. Custom Field Names

The custom field names can be defined for each individual field definition of any desired WebCenter Forms Recognition document classes in Designer application using “Display Name” edit box available on “Field” property page of selected field definition’s settings:

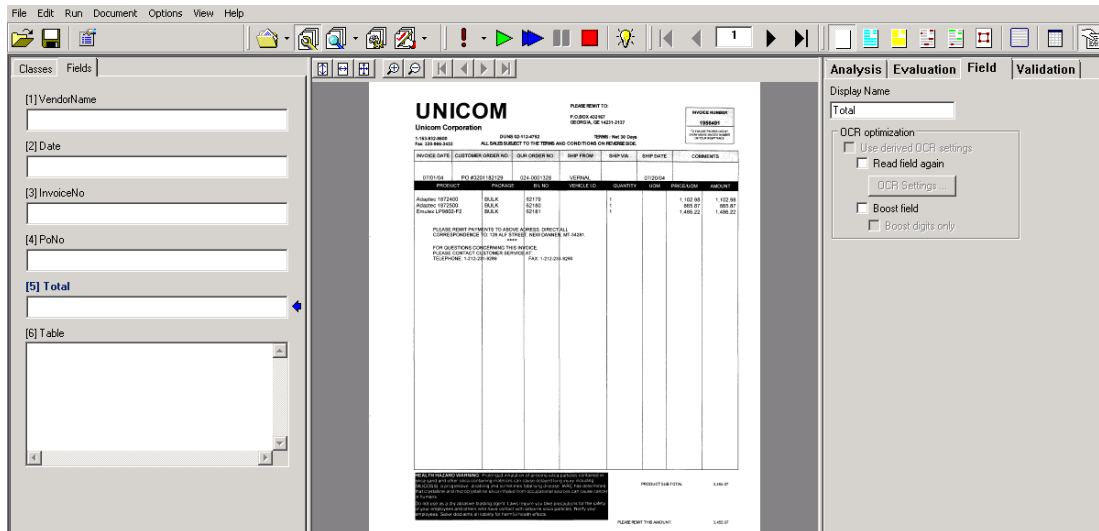


Figure 8-4: Field property page – definition of custom field names

The defined “Display Names” primarily affect verification mode of the Verifier application, showing the configured custom name (for the currently selected field element) in the left bottom area of the status bar instead of the original system name (on the screenshot below the Verifier application shows configured “Gross Amount” name instead of the default system “Total” – see also the screenshot below, field “[5] Total”):

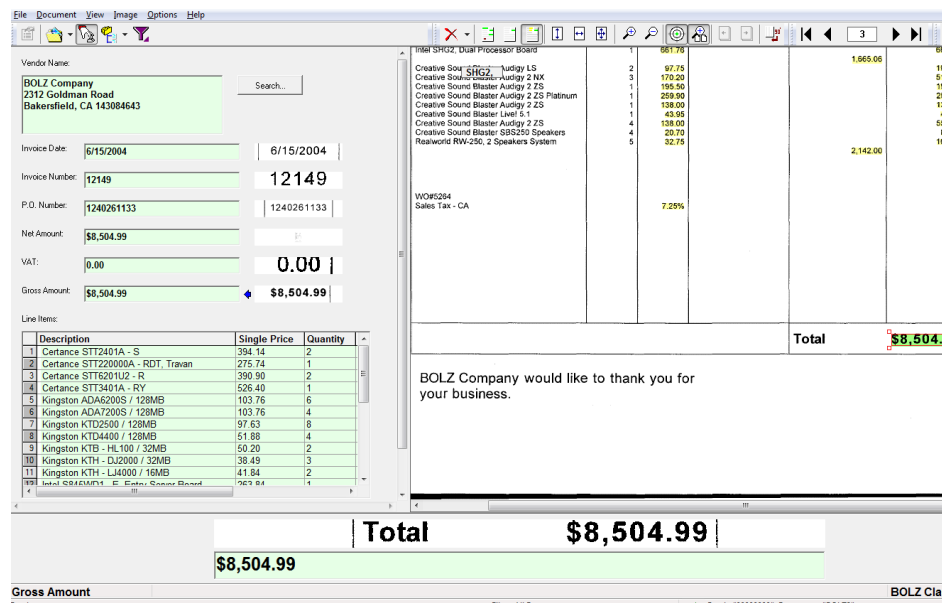


Figure 8-5: Configured field names

Additionally, this feature affects some minor features of WebCenter Forms Recognition, for example, the forms’ printing function. For an instance, in the case of the document sample above, the Form Print function would generate the following printing output:

File name: d:\projects\demo\batches\00000000\BOLZ2.wdc  
Document class name: BOLZ Company 1234561

**DOCUMENT FIELDS:**

Vendor Name: BOLZ Company  
2312 Goldman Road  
Bakersfield, CA 143084643

Invoice Date: 6/15/2004  
Invoice Number: 12149  
P.O. Number: 1240261133  
Net Amount: \$8,504.99  
VAT: 0.00

Line Items:

Description	SP	Qty	TP
1 Certance STT2401A - S	394.14	2	788.28
2 Certance STT220000A - RDT, Travan	275.74	1	275.74
3 Certance STT6201U2 - R	390.90	2	781.80
4 Certance STT3401A - RY	526.40	1	526.40
5 Kingston ADA6200S / 128MB	103.76	6	622.56
6 Kingston ADA7200S / 128MB	103.76	4	415.04
7 Kingston KTD2500 / 128MB	97.63	8	781.04
8 Kingston KTD4400 / 128MB	51.88	4	207.52
9 Kingston KTB - HL100 / 32MB	50.20	2	100.40
10 Kingston KTH - DJ2000 / 32MB	38.49	3	115.47
11 Kingston KTH - LJ4000 / 16MB	41.84	2	83.68
12 Intel S845WD1 - E, Entry Server Board	263.84	1	263.84
13 Intel S875WP1 - E, Entry Server Board	309.49	1	309.49
14 Intel SE7500CW2, Dual Processor	429.97	1	429.97
15 Intel SHG2, Dual Processor Board	661.76	1	661.76
16 Creative Sound Blaster Audigy LS	97.75	2	195.50
17 Creative Sound Blaster Audigy 2 NX	170.20	3	510.60
18 Creative Sound Blaster Audigy 2 ZS	195.50	1	195.50
19 Creative Sound Blaster Audigy 2 ZS Platinum	259.90	1	259.90
20 Creative Sound Blaster Audigy 2 ZS	138.00	1	138.00
21 Creative Sound Blaster Live! 5.1	43.95	1	43.95
22 Creative Sound Blaster Audigy 2 ZS	138.00	4	552.00
23 Creative Sound Blaster SBS250 Speakers	20.70	4	82.80
24 Realworld RW - 250,2 Speakers System	32.75	5	163.75

Figure 8-6: Printing output

In other words, use the configured “display names” instead of the system names (Compare the bold names on this screenshot with the system names configured in).

*Note:* The “Display Names” settings defined for a parent class are supposed to be inherited for all its sub-classes.

## 8.2.2. Editing Fields

WebCenter Forms Recognition Designer provides commands for editing fields. Be careful when using them since you might lose some of the settings you have already made.

### Task Prerequisites

The prerequisites for this task are:

- The program is in Definition Mode.
- The panel on the left side of the window displays the *Fields* tab in the foreground.
- There must be fields to edit.

### Selecting a Field to Edit

To select the field you want to edit:

- Double-click the field or the field name. Now the field name is highlighted and the cursor is positioned in the selected field.

To edit the selected field, click the Edit menu and go to Field definition or right-click directly on the field or field name.

For the selected field, the following operations are supported:

- The *Delete Field Definition* command from the shortcut menu deletes the field including all associated settings. In this case, you lose all trained knowledge for the data extraction. Repeat the training.
- The *Rename Field Definition* command from the shortcut menu renames the field. In this case, you lose an existing Extraction Learnset for the field and all trained

knowledge for the project. If required, create a new Learnset for the field and repeat the training.

- The *Text and Table* commands from the shortcut menu allow you to change the field type.
- To check for analyzed table content, press *Shift* and click the mouse button on the table icon in the field.

### 8.3 Selecting the Analysis Method

Once you have created fields and assigned a language, specify the analysis method for each of them.

#### Task Prerequisites

The prerequisites for this task are:

- The program is in Definition Mode.
- The panel on the left side of the window displays the *Fields* tab in the foreground.
- On the right side of the window, the tabs with class/field properties are visible.
- The *Analysis* tab is displayed in the foreground.

#### Selecting an Analysis Method

To select the analysis method for all fields of a class:

- 1) In the *Fields* tab on the left side of the window, select a field.
- 2) In the *Analysis* tab on the right side of the window, select the method that you want to use from the Available Analysis Engines list box.

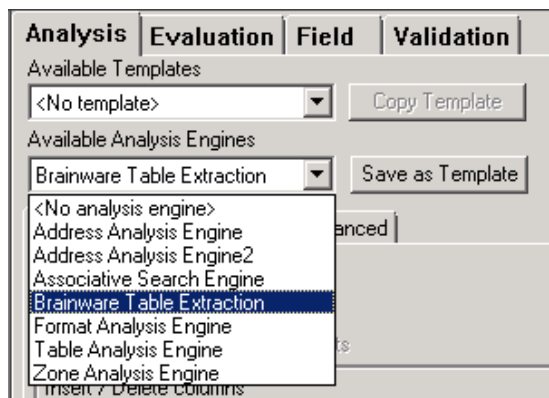


Figure 8-7: Analysis tab with analysis methods per field

- 3) Repeat Step 1 and Step 2 until all fields have analysis methods assigned.

### 8.4 Setting Up Brainware Table Extraction

The Brainware Table Extraction contains the fast and easy-to-use Generic Table Extraction. This allows the user to configure generic table extraction for any project type in an easy and comfortable manner.

The Brainware Table Extraction Engine is a powerful table extraction engine that is intended to:

- Reduce the amount of time required to configure data extraction from tables.
- Reduce the amount of time required for table verification rates
- Improve table data recognition rates.
- Indirectly reuse table data verified by customer services for incremental performance and quality improvements, thereby reducing support and installation time and increasing overall quality.
- Perform table extractions on projects that were previously too large for table extraction tasks.

### 8.4.1. About Brainware Table Extraction

The learning process for the Brainware Table Extraction (BTE) engine consists of two phases:

- Learning lines
- Learning mappings of columns

*Note:* In order to use Brainware Table Extraction all documents must have been scanned to a resolution of 300 dpi.

*Note:* The Brainware Table Extraction (BTE) and Brainware Field Extraction (BFE) engines' external Learnsets are now stored in a secure encrypted form (this effects "bte.ptb" / "bte.xtr" and "bfe.ptb" / "bfe.xtr" Learnset files stored per class's Learnset directory, in case an external BTE / BFE Learnset is available for a class).

#### 8.4.1.1. Learning Lines

The engine considers the following main types of the lines:

- **Primary line:** A line that defines table structure. The BTE engine applies advanced and precise similarity analysis for all primary lines. All primary lines must be well-structured and similar to each other in many of the rows to be extracted. However, the engine easily supports an unlimited number of different types of primary lines for one table definition. The primary line must be the first line in the table row and must contain at least four words.
- **Secondary line:** A line between primary lines. The engine applies smooth similarity analysis for these types of lines, which is possible because BTE searches only between two neighboring primary lines. This allows BTE to extract data that varies widely, which often happens with multi-line descriptions. There is no limitation on the number of words in secondary lines, and no limitation on the number of secondary lines. However, a document's page must have at least one primary line; otherwise, secondary lines on this page will not be extracted.
- **Wrong line:** A primary line that is learned as a negative line sample. In other words, all lines classified by the engine as a member of one particular "wrong" line class will not be extracted. In principle, it is possible to learn an unlimited number of the wrong lines, though this will take effect only during in-document learning. Cross-document learning (Learning the whole document after all the fields are completely valid) may not automatically train the wrong lines.

After learning any type of line, the BTE engine automatically creates and manages a new line class (cluster). Afterwards, all lines in the document considered by the engine as members of the line class (as similar to the learned line sample) will be extracted, or not extracted in the case of “wrong” lines.

It is possible to learn an unlimited number of different line classes. However, the overall quality of the extraction may suffer if too many lines are learned.

Learning lines can be applied in lines learning (or lines highlighting) mode. Likewise, mapping of the column data in the lines can be done in column mapping learning (or columns highlighting) mode. The user can switch between learning (highlighting) modes via the Switch Table Highlighting (CTRL + Q) menu option in Verifier or via pop-up menu options Show Lines and Show Columns of the document viewer in Verifier or Designer.

To learn a line the engine requires the line to contain a minimal number of words. The *Minimal number of words* value defines that minimal number of words that makes a line eligible for learning. Set this value according to the demands of the project documents.

The minimal primary lines threshold can be adjusted, if for example the correct primary line is not being extracted, by changing the Minimal Threshold on the Advanced property page of the BTE engine's setting. The default value is 40%.

The screenshot shows a settings window with three tabs: 'Columns', 'Labels & Formats', and 'Advanced'. The 'Advanced' tab is active. It contains three sections:

- Primary lines extraction:**
  - Minimal number of words: 4
  - Minimal threshold (0 - 100%): 40
- Secondary lines extraction:**
  - Minimal threshold (0 - 100%): 11
- Appliance of numeric correlation of columns:**
  - Correlation (enable/disable):

Figure 8-8: Primary and Secondary Lines Extraction

#### 8.4.1.2. Learning Mappings of Columns

When learning column mapping, the user trains the engine on how the data from the extracted lines must be mapped to the user's table data. For primary lines, this mapping can be defined differently for different line classes. For example, if a user learned two different line samples that went to two different lines classes internally in one document, the user can then map “Unit Price” in the document to the “Unit Price” data column and the “Total Price” to the “Total Price” for the first line sample. For all lines of the second line type, the user can map “Unit Price” to “Total Price” and “Total Price” to “Unit Price” and the engine will perfectly solve this task. Next time the BTE engine will always use mapping rules #1 for the lines classified to the first line type and mapping rules #2 for the lines classified as the second line type.

If you have several BTE tables in one class, the Learnset is shared between these tables. In other words, if you used interactive learning for one BTE table, the cross-document learning (which happens if the system added the document to the Learnset after document validation) will be applied for all BTE tables in the document.


### 8.4.1.3. Pre-train Brainware Table Extraction

To extract a table before training:

- 1) Switch to Definition Mode
- 2) Under *Options* select Pre-train Brainware Table Extraction.

### 8.4.1.4. Configuring Brainware Table Extraction

To setup a table field and define Column Names for Brainware Table Extraction follow these steps (Step 10 only for the build-in Generic Table Extraction):

- 1) Switch to Definition Mode
- 2) On the *Classes* tab on the left side of the screen, select a parent class.
- 3) On the left side of the screen, click the *Fields* tab.
- 4) Insert a new field. Right click anywhere on the tab, select *Add Field Definition* and give the new field a name. Right click on the field and change it from a text field to a table field.
- 5) Right-click in the table field and open *Properties*. On the right side of the screen the properties pane opens.
- 6) On *Analysis* tab under *Available Analysis Engines* select Brainware Table Extraction.
- 7) On the *Columns* tab, insert the columns to include them in the extracted table. To do this, click the *Insert columns* button .

Type a name for the column. Select *Column Required* if the column cannot contain null values. Select *Multiline Cells* if the engine should not convert multiline text to a single line. Clear the selection for *Column Visible* if you want to hide the column.

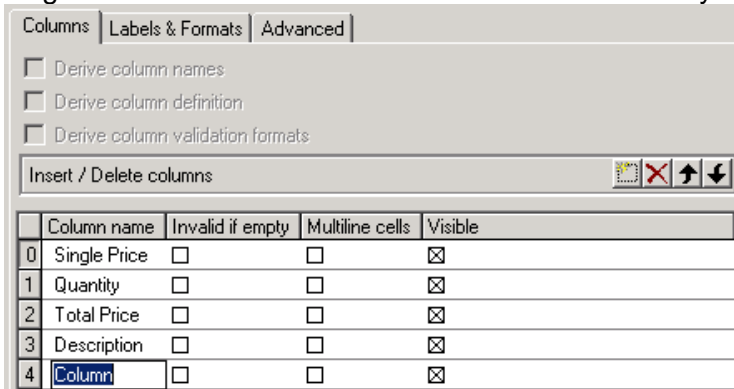


Figure 8-9: Columns tab of the Brainware Table extraction

- 8) Click this button again to add more columns. Use the button with the red X to delete columns and the twisted arrow buttons to move columns up or down.
- 9) Switch to the secondary tab, *Labels & Formats*. Next to *Column Name*, select a column. Use the arrow buttons to scroll through the lists of columns you have created.
- 10) Add column labels in the *Label property* box only for generic extraction. For details see section [8.4.2](#) on the Generic Table Extraction.

- 11) Under *Column Formats*, insert your formats (see section [SETTING UP FORMAT ANALYSIS](#) and [REGULAR EXPRESSIONS](#) to learn how to write expressions) and select *Format Comparison* from the second column.)

## 8.4.2. Configuring Generic Table Extraction

The powerful generic mode of the Brainware Table Extraction performs without prior scripting or training. The extraction is based on the matching of column labels and additionally works with simple mathematical correlation between columns. Furthermore, it increases the speed of the extraction process.

To use the Generic Table Extraction:

1. Configure the table columns as described above in chapter [8.4.1.4](#), step 7.
2. Enable Generic Table Extraction by assigning Column Labels (see section [8.4.2.1](#)). The generic mode is activated automatically as soon as a table field gets the first column label configured.
3. Apply Numeric Correlation optionally (see section [8.4.2.2](#)).

### 8.4.2.1. Column Labels

The Generic Table extraction identifies the table header line with the help of the Column Labels. The column headers that match with Column Labels will be mapped. Matches will be placed in the data table under the Column Name that is defined in the Column list.


To ensure that the engine finds adequate matches, define all column headers that are expected in the documents as Column Labels of the corresponding Column Name. E.g. for the Column Name *Quantity* define the Column Labels *Amount*, *Qty*, *Quantity*, *Quantity shipped*).

The search for Column Labels is a fuzzy search providing some results even for misspelled or poorly OCR'd words.

#### Note:

- *A Label needs at least three characters otherwise it will be ignored. The ideal length of a one word label line expression is four characters or more.*
- *Do not use the same Label word under different Column Names.*
- *For multi-line headers only the line that is judged as most appropriate is used for matching.*

To set Column Labels:

1. Go to the *Labels & Formats* tab.
2. Select a Column Name with the left/right arrows.
3. Click the *Insert labels* button  and enter the Column Label.
4. Repeat step 3 until all Column Labels for the selected Column Name appear on the list.
5. Repeat steps 2-4 for all other Column Names that should appear in the extracted table.

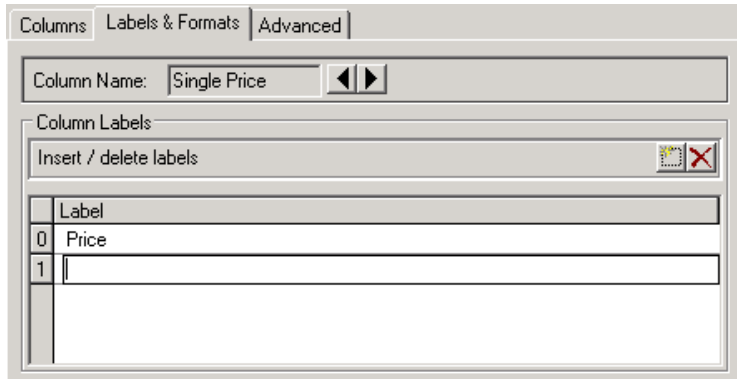


Figure 8-10: Labels & Formats tab of the Brainware Table extraction

Delete all Column Labels to turn off the Generic Table extraction.

#### 8.4.2.2. Numeric Correlation of Columns

The Numeric Correlation of the Generic Table Extraction uses the mathematical interdependency of values to assign the correct values to the corresponding columns. The first three columns with the numbers 0, 1 and 2 in the column list on the *Column* tab will be checked for correlation. The correlation follows the scheme  $0 * 1 = 2$ . A common example for an *Invoices* project is  $\text{Single Price} * \text{Amount} = \text{Total Price}$ . The columns need to be ordered in the same sequence: 0 - Single Price, 1 - Amount, 2 - Total Price.

Only use the Correlation feature when the columns are expected to correlate in most documents. It is recommended to turn the Correlation option off in case the majority of the processed documents does not have a numeric dependency between table columns.

The numeric Correlation is disabled by default.

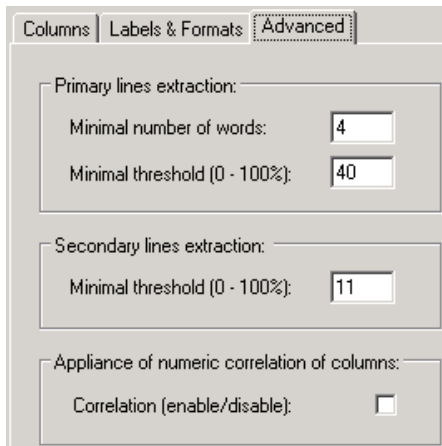


Figure 8-11: Advanced tab of the Brainware Table extraction

To enable the Correlation:

1. Go to the *Advanced* tab of the Brainware *Table* extraction properties.
2. Check the *Correlation (enable/disable)* checkbox.
3. Go to the *Column* tab.
4. Sort the rows by *Column name* in the desired order. To move the rows up or down select one by clicking on it and press the Up/Down arrow on the top right of the table.



Make sure that the 3 columns selected for correlation are on the top of the list followed by all other columns.

	Column name	Invalid if empty	Multiline cells	Visible
0	Single Price	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
1	Quantity	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
2	Total Price	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
3	Description	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

Figure 8-12: Column list in correct order for numeric correlation

### 8.4.2.3. Limitations

- All documents have to be scanned with 300 DPI for generic extraction.
- Generic Table Extraction ignores Labels with less than 3 characters.
- The user cannot control the identification of a label line (table header line) nor the selection of the columns that are going to be mapped to the data table via a threshold approach. The confidence of their detection cannot be controlled by the threshold settings.
- To learn a line the engine requires the line to contain a minimal number of words. The *Minimal number of words* value defines that minimal number of words that makes a line eligible for learning. Set this value according to the demands of the project documents.
- The label line expressions can consist of multiple words, but there is no difference between specifying two expressions, e.g. "line" + "number", and one single expression "line number".

### 8.4.3. Custom Column Names

The WebCenter Forms Recognition Designer application provides GUI settings to setup custom column names for verification table objects.

To set the desired table column display names, open your project in the Designer application, select required class in Definition mode, switch to the Verifier Design mode and select the desired table object. Now show the table object's pop-up menu by right clicking the box and select the *Properties* menu item:

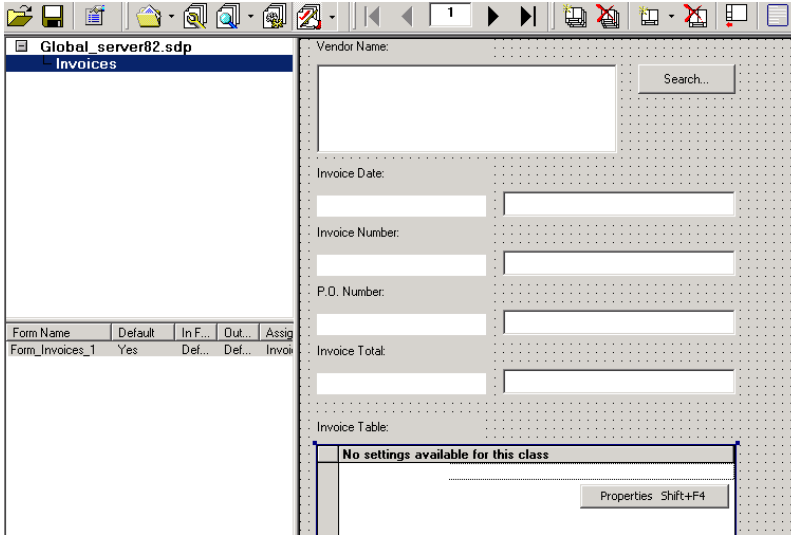


Figure 8-13: Design mode - Properties

The Forms Designer should display the table object’s properties. Click *Column Settings...*:

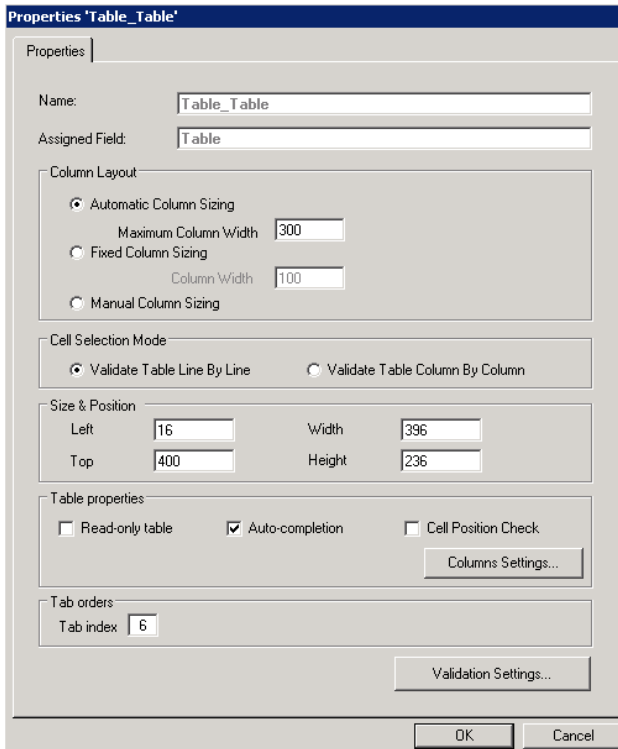


Figure 8-14: Table object’s properties – Column Settings

Modify the desired column names by selecting the column row, click on the “Display Name” area of the row, and then type in the required text:

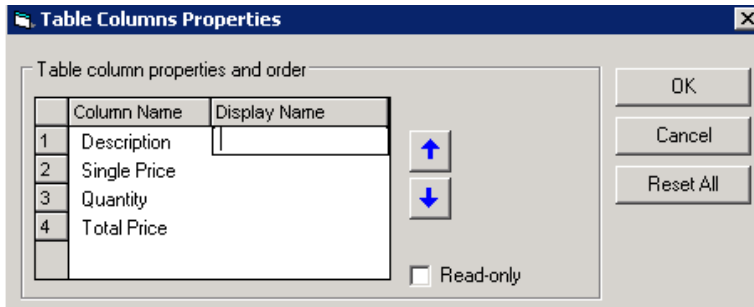


Figure 8-15: Modification of column names

Click OK when you finished configuring.

*Note:* The display column name will remain the same as the standard "system" name in case you leave the "Display Name" field empty.

When the new "Display Name" fields have been configured, the Forms Designer of the Designer application will update the controls' names immediately:

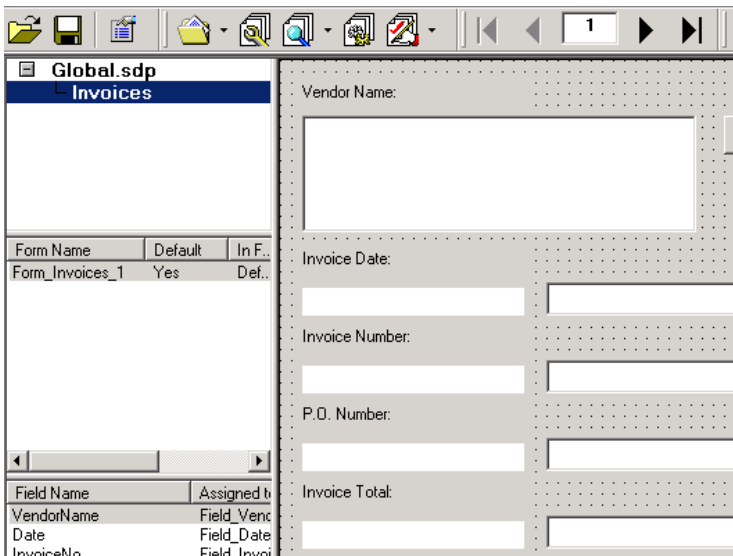


Figure 8-16: Form with modified controls' names

The configured display column names affect the Verifier application and Verifier Test / Verifier Train modes of the Designer application, when representing the table's column header

It also shows "display" names when using column mapping control in context of Brainware Table Extraction interactive learning:



And provides with "display" names when using normal column mapping and swapping operations:

07/09/04	PO #3201182129	024-0001358	VERNAL	08/02/04				
PR	Map Column	B/L NO	VEHICLE I.D.	QUANTITY	UOM	PRICE/UOM	AMOUNT	
Connect3D 6	Swap Column	Description		4		84.72	338.88	
3Diabs 01-00	Unmap Column	Single Price		1		372.44	372.44	
Canopus 770	Table Candidates	Quantity		2		397.81	795.62	
Colorgraphic	Show Lines	Total Price		2		269.01	538.02	
Colorgraphic				2		674.24	1,348.48	
Colorgraphic 620122				2		62.32	124.64	
Colorgraphic CRDVGACFH131				2		190.25	380.50	
Intel S845WD1-E		RETAIL	20100	1		258.19	258.19	
Intel S875WP1-E		RETAIL	20101	1		306.11	306.11	
Intel SE7500CW2		RETAIL	20102	1		420.81	420.81	
Intel SE7501BR2		RETAIL	20103	1		734.37	734.37	
Intel SE7501CW2		RETAIL	20104	1		436.44	436.44	
Intel SE7501HG2		RETAIL	20105	1		839.86	839.86	

Figure 8-17: New “display” names in menus

### Usage

The table column names configured in settings of Brainware Table Extraction and Table Analysis engine have internal meaning (as a property of the engines’ interfaces) and cannot be easily adjusted or modified dynamically from within the custom script. The GUI settings in Designer allow assigning of user-readable column names for the corresponding system column names to be shown to a Verifier user.

Particularly, this feature can be used for dynamic translation of verification forms into different languages, when it is desired to be supplied with a multilingual Verifier installation in context of a single shared project file.

## 8.5 Setting up Format Analysis

Format analysis is suitable for extracting data that is located at arbitrary positions on the documents. It is the primary analysis method and can be used with unstructured documents and with structured documents such as forms.

Format analysis is a search within the document text for strings that match certain pre-defined patterns. Configuring this method requires that you define the patterns, i.e. the format strings. The following methods to find strings are supported:

- **String Compare:** String Compare is a very simple search method that finds each literal occurrence of the specified format string.
- **Levenshtein:** Levenshtein search is an error-tolerant search method that finds each literal occurrence of the specified format string, but also strings that can be derived from the specified one by inserting, interchanging or deleting single characters. The number of key operations required to derive the erroneous string determines whether there is still a match.

Example:

0 errors	1 error	2 errors
invoice	invoike	invoke
	involve	involve

Table 8-1: Example for Levenshtein error-tolerant search

Use this method to account for typical typographical errors like character interchange.

- **Trigram:** Trigram search is another error-tolerant search method. To compare two words they are fragmented into groups of three characters called trigrams. The number of identical groups determines whether there is still a match.

Example:

	1st trigram	2nd trigram	3rd trigram
brain	bra	rai	ain
train	tra	rai	ain

Table 8-2: example for Trigram error-tolerant search

Use this method to account for OCR errors in your document.

- **Simple Expression:** Simple expressions can be used to specify simple format patterns. This is the default method. In simple expressions, some characters have a special meaning (See [TABLE 8-3: SPECIAL CHARACTERS IN SIMPLE EXPRESSIONS](#)). All other characters have no special meaning and represent themselves.

Character	Description
#	Represents any number. Example: ### matches 123.
@	Represents any upper-case or lower-case letter. Examples: @@@@ matches Love. @# matches U2.
?	Represents any alphanumeric character. Example: ?rain matches brain and train.
' '	Indicates a word start or end. Example: 'ABC' matches ABC, but not ABCD
[ ]	Indicates a number or a range of repetitions of the previous character. Examples: #[3] matches any three numbers like 123. a[1,2] matches a and aa. me[1,2]t matches met and meet.

Table 8-3: Special characters in simple expressions

- **Regular expressions:** Regular expressions can be used to precisely specify complex format patterns. Designer supports a subset of regular expressions described in [REGULAR EXPRESSIONS](#).

The string you are looking for may consist of only one word only or several words. Whether the string is finally found in the document also depends on rules as to how it is constructed from words. Default rules are set for this and should work in most cases. If your format string looks fine, but you still cannot find candidates, you may have to adjust them. (See section [DEFINING THE RULES FOR STRING CONSTRUCTION FROM WORDS](#))

### 8.5.1. Defining the Format Strings

For each field, a set of format strings can be defined. Before you begin, consider the following:

- You should review your documents and, for each field, create a list of the strings that are to be extracted. This way, it is easier to identify the patterns to search for.
- Do not try to find precisely one candidate; try to generate a set. Learning works better if there are both correct and false examples.
- Try to anticipate other likely formats. If the correct candidate has no chance to be identified, the need for manual correction will increase.

#### Task Prerequisites

The prerequisites for this task are:

- The program is in Definition Mode.
- The panel on the left side of the window displays the *Fields* tab in the foreground.
- A field is selected for which format analysis has been defined.
- The viewer displays a document from the class.
- On the right side of the window, the tabs with class/field properties are visible.
- The *Analysis* tab is displayed in the foreground.

#### Defining Format Strings

To define format strings:

- 1) On the *Analysis* tab, select the *Format Analysis* tab at the bottom.
- 2) Initially, most of the controls in this tab are disabled. You need to specify the format string before you can adjust the search method.
- 3) In the *Format Strings* table, the last row is always empty. Click on the row number and type a search string. Press *ENTER*. This adds a new row. The string remains selected. You can now make further adjustments.

The screenshot shows the 'Format Analysis' dialog box with the 'General' tab selected. The 'Analysis Method' section has 'Find Format' selected. The 'Designator Type' is set to 'Next Word'. The 'Compare Method' is set to 'Simple Expression'. The 'Format Strings' table at the bottom has one row with the number '0' in the first column and an empty cell in the second column.

?	Format Strings
0	

Figure 8-18: Initial state of the Format Analysis tab

- 4) If required, select a different search method from the *Compare Method* list box. By default, Simple Expression is set.
- 5) If required, select the Find Designator option. In this case, the format analysis will look for your search string, but the candidate will be a different string that has a well-defined geometrical relationship to the search string.  
 For example, you can search for the string "Phone" to extract the subsequent information, which typically is a phone number.  
 In the Designator Type list box, you can select from among the following options:

Type	Sample Format Strings	Sample Candidates
Word below	Customer	<p>Customer ID: [REDACTED]</p> <p>U-2345-234 email: budhunter@yahoo.com Fax 868-327-1279</p> <p>DeWitt University London UK</p> <p><b>Billing</b> Hungry Owl All-Night Grocers <b>Address:</b> 8 Johnstown Road Cork Co. Cork Ireland</p>
Word Above	UK	<p>Customer ID: [REDACTED]</p> <p>U-2345-234 email: budhunter@yahoo.com Fax 868-327-1279</p> <p>DeWitt University London UK</p> <p><b>Billing</b> Hungry Owl All-Night Grocers <b>Address:</b> 8 Johnstown Road Cork Co. Cork Ireland</p>
This Block	Billing	<p>Customer ID: [REDACTED]</p> <p>U-2345-234 email: budhunter@yahoo.com Fax 868-327-1279</p> <p>DeWitt University London UK</p> <p><b>Billing</b> Hungry Owl All-Night Grocers <b>Address:</b> 8 Johnstown Road Cork Co. Cork Ireland</p>
Previous Word	University	<p>Customer ID: [REDACTED]</p> <p>U-2345-234 email: budhunter@yahoo.com Fax 868-327-1279</p> <p>DeWitt University London UK</p> <p><b>Billing</b> Hungry Owl All-Night Grocers <b>Address:</b> 8 Johnstown Road Cork Co. Cork Ireland</p>
Next Word	email	<p>Customer ID: [REDACTED]</p> <p>U-2345-234 email: budhunter@yahoo.com Fax 868-327-1279</p> <p>DeWitt University London UK</p> <p><b>Billing</b> Hungry Owl All-Night Grocers <b>Address:</b> 8 Johnstown Road Cork Co. Cork Ireland</p>

Type	Sample Format Strings	Sample Candidates
End of Line	Fax	
Next Line	Format	<p><b>Format Strings</b></p> <p>On the Analysis tab, select the Format Analysis tab at the bottom. Initially, most of the controls in this tab are disabled.</p> <p>You need to specify the format string before you can adjust the search method. In the Format Strings table, the last row is always empty. Click on the row number and type a search string. Press ENTER. This adds a new row. The string remains selected. You can now make further adjustments.</p>
Next Paragraph	Format	<p><b>Format Strings</b></p> <p>On the Analysis tab, select the Format Analysis tab at the bottom. Initially, most of the controls in this tab are disabled.</p> <p>You need to specify the format string before you can adjust the search method. In the Format Strings table, the last row is always empty. Click on the row number and type a search string. Press ENTER. This adds a new row. The string remains selected. You can now make further adjustments.</p>
Next Block	Format	<p><b>Format Strings</b></p> <p>On the Analysis tab, select the Format Analysis tab at the bottom. Initially, most of the controls in this tab are disabled.</p> <p>You need to specify the format string before you can adjust the search method. In the Format Strings table, the last row is always empty. Click on the row number and type a search string. Press ENTER. This adds a new row. The string remains selected. You can now make further adjustments.</p>

Table 8-4: Examples for Designator Type

- 6) Under *Prefix* and *Suffix*, you can enter characters or syllables that are part of a word, but will be ignored when found at the beginning or the end of the word. For example, you can look for 22,99, and \$22,99 will match.
- 7) Under *Ignore Characters*, you can enter characters that are part of a word, but will be ignored when found anywhere within the word. For example, you can look for 213-4789, and 2134789 will match.
- 8) To define additional strings for the current field, repeat Step3 to Step 7. Note that you can combine search methods freely. However, a word can only be member of one candidate. The format strings will be evaluated in the same order as listed in the Format Strings table. To change the order, click a row number and drag the corresponding string to its new position. To delete a string, click the row number and press the *DELETE* key.

**Note:** Test the analysis step with candidate highlighting to check early whether you have defined the suitable format strings. But be careful if you use field inheritance. The extraction test always involves a classification step. You can easily end up changing the child class settings by mistake. Always check the class name that is displayed in the bottom-left edge of the window. This is the class you are currently editing.



*Note:* Format analysis normally yields several candidates. Therefore, a candidate evaluation must take place.

### 8.5.2. Defining the Rules for String Construction from Words

Mainly due to the OCR, a candidate may be fragmented into several words, or several words may have been concatenated. Therefore the string search also covers combinations and subsets of words. The corresponding rules are valid for all format strings assigned to a given field.

#### Task Prerequisites

The prerequisites for this task are:

- The program is in Definition Mode.
- The panel on the left side of the window displays the *Fields* tab in the foreground.
- A field is selected for which format analysis has been defined.
- The viewer displays a document from the respective class.
- On the right side of the window, the tabs with class/field properties are visible.
- The *Analysis* tab is displayed in the foreground.

#### Defining Rules for Words

To define rules for string construction from words:

- Within the *Analysis* tab, select the *General* tab at the bottom.

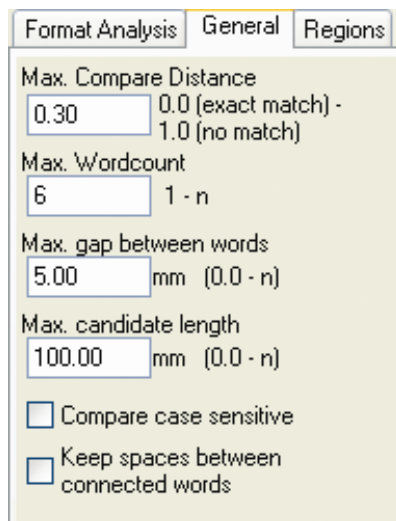


Figure 8-19: String construction rules

- The Max. Compare Distance allows you to extract a search string from a word, neglecting a certain number of characters at the beginning and / or at the end. The compare distance is computed as follows:  

$$\text{Compare Distance} = 1.00 - \frac{\text{Number of Characters (Search String)}}{\text{Number of Characters (Word)}}$$
 A match requires that the actual compare distance is less or equal the maximum compare distance.

*Example:*

*Search String: 1234 (4 characters)*

*Max. Compare Distance: 0.30*

*Word 1: \$1234 (5 characters)*

*Compare Distance 1:  $1.00 - 4/5 = 1.00 - 0.80 = 0.20 < 0.30$  -> match*

*Word 2: \$1234.00 (8 characters)*

*Compare Distance 2:  $1.00 - 4/8 = 1.00 - 0.50 = 0.50 > 0.30$  -> no match*

- The *Max. Word count* allows you to extract strings that consist of multiple words.  
Example:  
By default, the hyphen (-) is a word separator. Therefore, the phone number 123-456-7890 consists of three words. To get a match in the format analysis, a maximum word count of three or more is required.
- The *Max. Gap Between Words* specifies the maximum distance in mm that permits word concatenation during the search. Note that the requirements strongly depend on font size.
- The *Max Candidate Length* specifies the maximum length in mm a candidate is allowed to have. If the candidate is longer it will not be accepted.
- Check *Compare case sensitive* to make the candidate search case-sensitive
- By default, spaces between concatenated words are deleted.  
Example:  
The search string 123 456 7890 will be extracted as 1234567890.  
Check *Keep spaces between connected words* if you want to keep the spaces. In this case, enter the spaces in your format string as well.

### 8.5.3. Restricting the Analysis to Certain Areas Within the Document

You can restrict the text that is analyzed to certain areas of the document. This may increase processing speed and accuracy.

#### Task Prerequisites

The prerequisites for this task are:

- The program is in Definition Mode.
- The panel on the left side of the window displays the *Fields* tab in the foreground.
- A field is selected for which format analysis has been defined.
- The viewer displays a document from the respective class.
- On the right side of the window, the tabs with class/field properties are visible.
- The *Analysis* tab is displayed in the foreground.

#### Restricting Analysis Areas

To restrict the analysis to certain areas within the documents:

- 1) On the *Analysis* tab, select the *Regions* tab at the bottom.

	First Page	Subseq.	Last Page
Top (%)	0	0	0
Bottom (%)	100	100	100
Left (%)	0	0	0
Right (%)	100	100	100

Figure 8-20: Regions tab for Oracle view

- 2) Check the *Restrict engine to region* check box.
- 3) Select the pages that are affected by the restriction.
- 4) For each selected page, enter the regions that are to be taken into account.

#### 8.5.4. Support of Character Encoding Tables ISO/IEC 8859-2, -5, and -9

WebCenter Forms Recognition document processing extends the previously introduced non-western languages processing approach with selective support of character encoding tables ISO/IEC 8859-2, -5 and -9, particularly implementing full phonetic translation support of Polish, Czech and Turkish languages.

##### 8.5.4.1. Usage

The feature is clearly very important to expend WebCenter Forms Recognition installation on the projects running in the countries like Poland, Turkey and Czech Republic. The support of non-western languages in general is, by the way, an important differentiator when we compare our product with the ones from our competitors, which very often cannot work with documents in, e.g., Greek or Russian language, while WebCenter Forms Recognition can process them perfectly.

#### 8.5.5. Ability to Create Associate Search Engine Pool Imported from an ODBC Source with Entries in “Non-Western” Languages

The Associative Search engine now supports non-western languages extension also when importing from an ODBC data source.

##### 8.5.5.1. Usage

This extension to the non-western languages support makes it possible to import the ADS pools for non-western languages directly from a database while in previous versions it was only possible to import the native data via an intermediate CSV file.

## 8.6 Setting up Zone Analysis

Zone analysis is suitable for extracting data located at fixed positions within the documents, such as with classical forms.

### 8.6.1. Creating Reading Zones

Reading zones are fixed areas on documents that contain information that is to be extracted. Three types of reading zones can be used:

- **OCR:** The contents of the reading zone are determined using optical character recognition. The textual content of the zone is the result of this process.
- **Barcode:** The contents of the reading zone are determined using barcode recognition. The number or strings represented by the barcode is the result of this process.
- **OMR:** The contents of the reading zone are determined using optical mark recognition. The degree of blackness within the zone is used as indicator. A Boolean value is the result of this process.

#### Task Prerequisites

The prerequisites for this task are:

- The program is in Definition Mode.
- The panel on the left side of the window displays the *Fields* tab in the foreground.
- A field is selected for which zone analysis has been defined.
- The viewer displays a document from the respective class.

If zone analysis has been defined for the active field, the viewer displays the page representation of the active document rather than the Workdoc representation. In this case, the viewer toolbar contains additional buttons for zone creation:






Button	Description
	Activates the selection tool
	Creates an OCR reading zone
	Creates an OMR reading zone
	Creates a barcode reading zone
	Creates an anchor

Table 8-5: Additional buttons in the viewer toolbar for zone creation

#### Creating Reading Zones

To create reading zones:

- 1) In the viewer panel, click the respective toolbar button to create an OCR zone, an OMR zone or a barcode zone.
- 2) Click on the document and drag to create a rectangular reading zone. Obviously, the rectangle needs to be large enough to hold the prospective contents. It should not be much larger, though. The reading zones are displayed as transparent rectangles with

red borders. Each zone has a label that displays the name of the zone. The initial name is created from the word *Zone* and a consecutive number. The currently selected reading zone is displayed with a gray background and solid red handles at each corner.

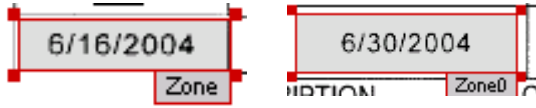


Figure 8-21: Appearance of reading zones

## 8.6.2. Editing Reading Zones

WebCenter Forms Recognition Designer provides some commands for editing reading zones.

### Task Prerequisites

The prerequisites for this task are:

- The panel on the left side of the window displays the *Fields* tab in the foreground.
- A field is selected for which zone analysis has been defined.
- The viewer displays a document from the respective class.
- Obviously, there must be zones on the document to edit.

### Supported Operations

The following operations are supported:

- You can move the currently selected zone using the mouse.
- You can resize the currently selected zone using one of the following methods:
  - You can drag the handles at each corner.
  - You can adjust the size of the reading zone to the size of the entire page using the shortcut menu and the *Fit to image size* command.
  - You can call the zone's property sheet using the shortcut menu and the *Properties* command. On the *General* tab, the following parameters are available for resizing:

Group	Parameter	Description
General	Units	Specifies the units used for geometric parameters (pixel, inch, or mm).
Size	Fit to image height	Adjusts the height of the zone to the height of the page.
	Fit to image width	Adjusts the width of the zone to the width of the page.
Position/Size	Left	Sets the distance of the reading zone to the left edge of the page.

Group	Parameter	Description
	Top	Sets the distance of the reading zone to the top edge of the page.
	Width	Sets the width of the reading zone
	Height	Sets the height of the reading zone

Table 8-6: Parameters to resize reading zones

- You can rename the currently selected zone via the zone's property sheet. Use the shortcut menu and the *Properties* command. On the *General* tab, type a new name into the Name field.
- You can delete the currently selected zone by pressing the *DELETE* key.

**Note:** If multiple reading zones are mapped to a single data field, a candidate evaluation must take place. This usually involves learning. If you change the properties of your reading zones after the learning, you lose all trained knowledge for the extraction. Repeat the training.

### 8.6.3. Fixing the Coordinate System for the Reading Zones

It is quite common that paper documents are scanned at an angle. From page to page, there will be fluctuations of the page position. However, zone analysis can only be successful if the relative position of reading zones on documents is constant over the entire document set. So-called anchors can be used to look for prominent geometric features on the document, such as the page corners. Once they are found, anchors can provide a fixed coordinate system for the reading zones, i.e. reading zone positions are determined relatively to anchor positions. This can balance fluctuations of several millimeters.

#### Task Prerequisites

The prerequisites for this task are:

- The program is in Definition Mode.
- The panel on the left side of the window displays the *Fields* tab in the foreground.
- A field is selected for which zone analysis has been defined.
- The viewer displays a document from the respective class.

#### Creating and Using Anchors

To create and use anchors:

- 1) In the Viewer panel, click the toolbar button that creates anchors.
- 2) Click into the image and drag to create a rectangular zone. This zone should have a size proportional to the geometric feature that you want to look for. For example, if all the documents in your class contain a 50 mm line in the footer area, your anchor should not have a side length of 5 or 250 mm. The anchor position should be close to the geometric feature you want to use. Anchors are displayed as transparent rectangles with blue borders. Each anchor has a label that displays its name. The initial name is created from the word Anchor and a consecutive number.

The currently selected anchor is displayed with a gray background and solid blue handles at each corner.

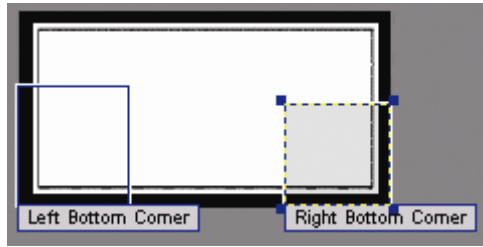


Figure 8-22: Appearance of anchors

- 3) To specify the anchor type, right click the anchor to display its shortcut menu and select Properties. The anchor's property sheet is displayed. The following parameters can be adjusted:

Group	Parameter	Description
General	Name	Set the anchors label
Position/Size	Left	Sets the distance of the reading zone to the left edge of the page.
	Top	Sets the distance anchor to the top edge of the page.
	Width	Sets the width of the anchor
	Height	Sets the height of the anchor
Anchor Type	(List box)	Sets the basic anchor type, i. e. the shape the anchor is looking for. Available are - image corners, - lines, - box corners, - T shapes, - crosses.
	Properties	Specifies the relevant edges for the selected anchor type. Each basic anchor type has its own set of properties. For example, if the basic shape is a line, it can be a horizontal or a vertical line. If the basic shape is a corner, there are four to choose from.

Table 8-7: Parameters to customize anchors

- 4) When everything is set, click *Learn* and close the property sheet.
- 5) To create more anchors, repeat Step 2 to Step 4.
- 6) Call the property sheet of a reading zone using the shortcut menu and the Properties command.
- 7) Select the *Anchor* tab.
- 8) Move one or more anchors from the *Available Anchors* list box to the *Used Anchors* list box using the buttons between the lists. Only explicitly used anchors provide a geometrical reference for the current reading zone.

- 9) To fix more reading zones, repeat Step 6 to Step 8.

### 8.6.4. Mapping Reading Zones to Document Fields

To actually extract data from your reading zones, you need to assign them to data fields.

#### Task Prerequisites

The prerequisites for this task are:

- The program is in Definition Mode.
- The panel on the left side of the window displays the *Fields* tab in the foreground.
- For the field, zone analysis has been selected.
- The viewer displays a document from the respective class.
- On the document, the reading zones have been defined.
- On the right side of the window, the tabs with class/field properties are visible.
- The *Analysis* tab is displayed in the foreground.

#### Mapping Data Fields and Reading Zones

To map data fields and reading zones:

- 1) In the Fields tab on the left side of the window, select the field.
- 2) In the Zone Analysis tab on the right side of the window, click a cell of the Zone Name column.
- 3) From the list of available reading zones, select a zone.

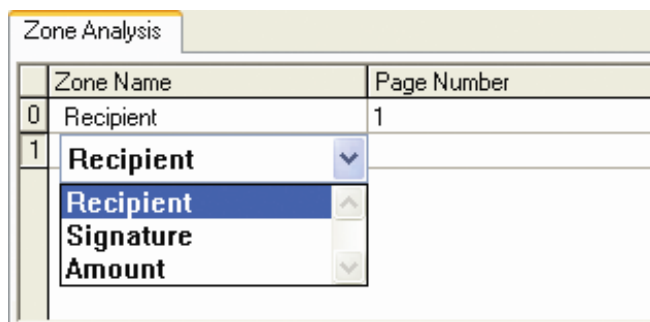


Figure 8-23: Mapping of data fields and reading zones

- 4) In the *Page Number* column, click to specify whether the reading zone is on the first or on the last page of the document.
- 5) To create multiple candidates from zone analysis, repeat step 2 to step 4 using the next table rows.

*Note:* If multiple reading zones are mapped to a single data field, a candidate evaluation must take place.

## 8.7 Setting up Address Analysis

*Important note:* The "Address Analysis Engine" and "Address Analysis Engine 2" are no longer supported but kept in version 11.1.1.8.0 of the software for backwards



compatibility reasons. If you would like to define addresses' extraction in 11.1.1.8.0 version the general recommendation is to use Address Extraction feature of the Associative Search engine instead of the engines listed above.

**Note:** Address analysis is a specialized method for searching addresses on a document. To use the address analysis, an address pool has to be created first. You can build your own address pool starting with a plain text file that lists names and addresses. An address pool can be used for different document classes.

Figure 8-24: The Address Analysis Engine

### 8.7.1. Creating the Address Pool

If you already have an address pool, you can skip this procedure.

To create an address pool:

- 1) Select the *General* tab of the Address Analysis Engine.
- 2) Create new directories for the text file and the pool.
- 3) Generate an address text file containing the addresses, as follows:  
 Lastname; Firstname; Street; Zipcode; Town; Customer ID (ID-Number can be customer number or something similar)

*Example:*

*Mustermann;Max;Musterstr. 1;12345;Musterstadt;0001*

*Quantum;Gesellschaft;Hauert 1;44227;Dortmund;123456*

- 4) Insert the path information in the Address Analysis Engine2 General tab.
- 5) Select *Generate/Update Pool* to create or update the search pool.
- 6) Select *Preview* to view the data of the text file in the General tab.
- 7) Select the checkbox *Replace with Database Entries* on the *General* tab to replace the address by the information saved in the address text file. This information may be somewhat different from the information in the zone.

Address Analysis General

Pool Path:  
C:\Projects\SLW Demo US\Pool

Pool Name:  
ASearch

Text file path for Address-Import:  
C:\Projects\SLW Demo US\address.txt

Generate/Update Pool

Pool Generation

Create Pool Update Pool

Update Preview  Replace with Database Entries

Preview Addresses

	Name	Firstname	Street	Pin Code	City	Customer ID
0	Mustermann	Max	Musterstr. 1	12345	Musterstadt	0001
1	Quantum	Gesellschaft	Hauert 1	44227	Dortmund	123456
2						
3						

Figure 8-25: General tab for Address Analysis

## 8.7.2. Configuring Address Analysis

### Task Prerequisites

The prerequisites for this task are:

- An address pool has been created and stored on your system's hard drive, preferably in your projects directory.
- The program is in Definition Mode.
- The panel on the left side of the window displays the *Fields* tab in the foreground.
- A field is selected for which address analysis (Address Analysis Engine2) has been defined.
- The viewer displays a document from the respective class. Note that the viewer displays the document page with recognition zones instead of the Workdoc.
- On the right side of the window, the tabs with class/field properties are visible.
- The *Analysis* tab is displayed in the foreground.

### Configuring Address Analysis

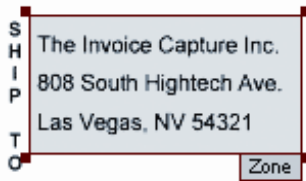
To configure address analysis:

- 1) Select the *Address Analysis* tab of the Address Analysis Engine.
- 2) Select the address type from the combo box. Your choices are Austria, Germany, Switzerland, or USA.

By default, the entire document is searched for addresses.

### 8.7.2.1. Designing Address Reading Zones

You can also create reading zones on the document and restrict the address search area to these zones, or exclude certain areas from the search. Create an OCR reading zone as described in Section [CREATING READING ZONES](#). It is a good idea to give the zone a descriptive name.



To select a zone for address analysis, click an empty cell in the first column of the Included/Excluded Zones table. From the list box, select the reading zone. In the second column, select the first or the last page. In the third column, double click the cell to include or exclude the zone.

Zone Name	Page Number	Included/Excluded
0 AddressZon		Included
1 AddressZone		Included

Figure 8-26: Defining the search zone.

If you then classify a document, the search engine tries to find the address within the zone, using the ASSA engine. When the search engine finds an address, it extracts the information from the zone.

When the checkbox *Replace With Database Entries* on the *General* tab is selected, the address will be replaced by the information saved in the address text file. If you disable the *Replace With Database Entries* selection, the information appears as in the zone.

Normally, you will have one address pool with all your addresses and this address pool can be used for several document classes. You can set up templates once you configure an address pool to use with other document classes. (See section [REUSING PROJECT SETTINGS WITH TEMPLATES](#).)

If you already have an address pool, you do not need to make any changes on the General tab, but you might want to change a setting in the Address Analysis tab. For example, you can change the zones that you want to include if you have multiple address zones. This might

be useful when your document contains two similar addresses and you are only interested in one of them. In this case, you would exclude the zone of the unwanted address. For example, if you are scanning invoices for addresses, you may have zones for both a customer address and a billing address. In this case, you could exclude the billing addresses and only extract the customer addresses.

*Note:* Address analysis may yield only one or several candidates. In any case, a candidate evaluation must take place.

## 8.8 Setting Up Table Analysis

*Note:* All new projects should be created with Brainware Table Extraction.

This section provides information about table analysis for project administrators working with legacy projects.

Prior to Version 10.1.3.5.0, table analysis was used to extract data from document tables. In principal, there are two approaches to traditional table analysis:

- Fixed layout: If the position of table elements is constant, you can use their coordinates to locate the table and its elements. This resembles the zone analysis of text fields.
- Variable Layout: If the position of table elements varies, or to analyze different table layouts with a single set of settings, you can use column labels and formats to locate a set of table candidates. This resembles the format analysis of text fields.

It is also possible to mix both approaches.

WebCenter Forms Recognition allows you to extract a single table from each document. This table can span multiple pages, provided that the table layout does not change over the entire range. In order to be accessible by table analysis, tables must consist of the following elements:

- Columns,
- Rows,
- and Cells

A table presents data in a structured layout. The basic elements of tables are columns and rows of data. Each row and column is made up of one or more cells. The cell is the most basic table element.

Most tables represent an enumeration of records, with each record listing a number of properties. The most common table layout organizes the listing of different properties in columns, while individual records are represented by table rows.

In some cases, one line is not sufficient to fit all the text that belongs into a table cell. In that case, a line feed occurs within the cell. This happens, however, only in particular columns. It is, for example, frequently encountered in columns containing the description of an item.

*Note:* Some tables have too many columns to place them all on one page side by side. In this case, some manufacturers create layouts where they actually interleave the columns in the table rows. Reading tables with interleaved columns is currently not supported.

*Note:* Some tables have empty rows. Reading tables with empty rows is currently only supported, if the table bottom is clearly indicated by a footer line.

Label 1	Label 2	Label 3	Label 4	Label line
---------	---------	---------	---------	------------

Cell	Cell	Cell	Cell
Cell	Cell	Cell	Cell
Cell	Cell	Cell	Cell
Cell	Cell	Cell	Cell
Cell	Cell	Cell	Cell
Cell	Cell	Cell	Cell

Column 1 (under first column), Cell 3,6 (under third cell of third row), Row 4 (to the right of the fourth row)

Figure 8-27: Main elements of tables

**Labels**

It is good practice to equip each table column with a label to indicate the corresponding property. In most cases, all column labels are positioned at about the same horizontal position, i.e. they belong to one text line. This line is called label line. In some cases, the labels extend over more than one text line. This is, for example, frequently encountered in industry invoices.

The label line plays a central role for the automatic table analysis, providing

- the start position of the table,
- the preliminary table layout, and
- the preliminary mapping of the table columns.

WebCenter Forms Recognition can process single-line as well as multi-line labels.

In order to properly assign a cell to a column, some degree of overlap with the label must exist. If a label overlaps with cells of different columns, the column is created using the cells with more overlap.

Since the label detection is vital for table recognition, the OCR quality is very important to obtain good results. Normally, light labels on a dark background are more difficult to read than vice versa.

Label 1	Label 2	Label 3
Cell	Cell	Cell
Cell	Cell	Cell
Cell	Cell	Cell

Column 1 (under first column), Column 2 (under second column), Column 3 (under third column)

Figure 8-28: Overlapping labels - in this example, one column will not be recognized

**Header and Footer Lines**

Frequently, tables contain a header line above the label line. If available, the header line may be used for detection of the table start position (Alternatively or in addition to the label line).

Footer lines are located below a table. An example of a footer line is a row that contains the sum of all column cell entries. If available, footer lines can be used to detect the end of a table (instead of or in addition to further criteria).

Header			
Label 1	Label 2	Label 3	Label 4
Cell	Cell	Cell	Cell
Cell	Cell	Cell	Cell

Footer (under the table)

Figure 8-29: Table header and footer

**Comments**

Comments, such as free text in tables, are automatically ignored if two requirements are met:

- There are at least two lines with cells above the comment.
- There is at least one line with cells below the comment.

Otherwise, comment lines will interfere with the table analysis.

Label 1	Label 2	Label 3	Label 4
Cell	Cell	Cell	Cell
Cell	Cell	Cell	Cell
This is a comment			
Cell	Cell	Cell	Cell

Figure 8-30: Comments in tables

## Graphical Lines

In some cases tables contain vertical or horizontal graphical lines. Horizontal lines can be used to detect the end of a table (instead of or in addition to further criteria). Vertical lines can indicate the segmentation of columns, but are currently not evaluated.

To configure the table analysis, you need to define a settings table: a table containing the columns you want to retrieve from the document along with a number of properties for each column. The tables within the document are called the document tables.

## Task Prerequisites

The prerequisites for this task are:

- The program is in Definition Mode.
- On the left side of the window, the *Fields* tab is activated.
- A field is selected for which table analysis has been defined.
- The viewer displays a document from the respective class.
- On the right side of the window, the tabs with class/field properties are visible, and the *Analysis* tab is in the foreground.

During table analysis, WebCenter Forms Recognition:

- 1) Identifies the document table that best matches the settings table,
- 2) Analyzes the rows and columns of the document table,
- 3) Maps the columns of the document table to those of the settings table, and
- 4) Extracts the content of the document table.

Each part of the analysis can create its own set of candidates. In the user interface, only the table candidate with maximum confidence is returned so that learning is not required. However, the Workdoc contains a list of all table candidates, sorted by confidence. These can be accessed from scripts.

### 8.8.1. Defining Table Columns

The first step in defining new table settings is to specify the columns to be searched for. Each column has to be assigned a name, which can be used to access the column data from script (alternatively to use the column index), and a set of properties. Defining tables for a derived class enables the checkboxes.

*Column names and column labels can be different from each other.*

## Defining Columns

To define the table columns:

- 1) In the *Analysis* tab, select *Table Analysis Engine* under Available Analysis Engines, go to the *Columns* tab.

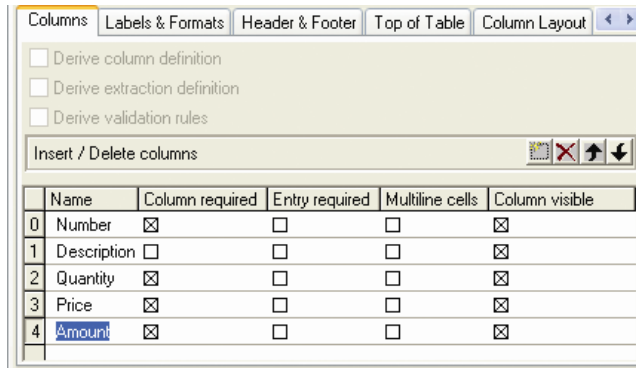


Figure 8-31: Tab for column definition

- 2) In the *Insert / Delete columns* list, create and sort columns using the following buttons:

Button	Description
	Adds a new entry at the bottom of the list. At the insertion point, type a name for the entry.
	Deletes the selected entry without displaying a confirmation message.
	Moves the selected entry one position up in the list.
	Moves the selected entry one position down in the list.

Table 8-8: Controls for list entries

- 3) For each column, specify the properties by marking the corresponding checkbox. There are three principal properties of each column:
  - *Column Required*: The column is required in all document tables, in contrast to optional.
  - *Entry required*: Entries are required for each cell of the column.
  - *Multiline cells*: The cell content can extend over more than one text line.
  - *Column Visible*: Column visible means that the column will be displayed on the Verifier form when this check box is selected. If not, you will not see this table column. You can set the visible property also in script in order to display this table column only when it contains a special content.

**Note:** During verification of tables, the user navigates from invalid cell to invalid cell by pressing ENTER. If no entries are required in a column, empty cells are valid, even though the document table contains corresponding input. To avoid losing information while keeping the ability to accept empty cells, you can make input mandatory for a column, but permit forced validation in the verification step.

This is a kind of pre-validation option. If it is selected for a table cell, it returns an invalid table cell when no entry is found. The result of the pre-validation will be discarded when the table cell is set to valid within the script.

### 8.8.2. Defining Column Labels and Formats

For each column, you can specify:

- A set of labels that will be used to identify the column. Labels are required.
- A set of format strings that will be used to determine whether the column cells have a valid content. Format strings are optional.

#### Defining Labels and Formats

To define column labels and formats:

- 1) In the *Analysis* tab, select the *Labels & Formats* tab.

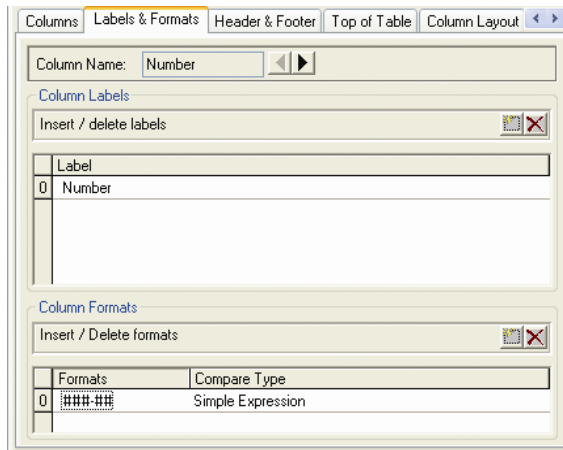


Figure 8-32: Tab for column labels and format definition

- 2) Select a column using the arrow buttons next to the *Column Name* field.



- 3) In the *Insert / Delete labels* list, define a set of labels using the following buttons:

Button	Description
	Adds a new entry at the bottom of the list. At the insertion point, type a name for the entry.
	Deletes the selected entry without displaying a confirmation message.

Table 8-9: Insert and delete list entries

For multi-line labels, no special settings are required as they are automatically combined to one label during table analysis.

- 4) In the *Insert / Delete format* list, you can define a set of format strings. Enter the format string in the *Format* column. When the format is inserted you can change the



compare type by clicking with the mouse into the corresponding compare type field and selecting the type from the combo box list.

*Note:* Formats have to be inserted first before you change the compare type. Otherwise, the field will be deleted.

- 5) Select a comparison type in the *Compare Type* column.

Syntax and comparison algorithms have already been described in Section [SETTING UP](#). If you use regular expressions, refer [REGULAR EXPRESSIONS](#) for details.

- 6) Repeat Step 2 – Step 4 until labels and format strings are defined for all table columns.

### 8.8.3. Defining Header and Footer Lines

You can specify a set of headers and footers for the table. Headers and footers can be used to determine table top and bottom. Normally, defining them is optional, but might make the task of table recognition easier. If your tables contain empty rows, you need footer. Otherwise, there is no way to determine the table bottom.

To define table headers and footers:

- 1) In the *Analysis* tab, select the *Header & Footer* tab.

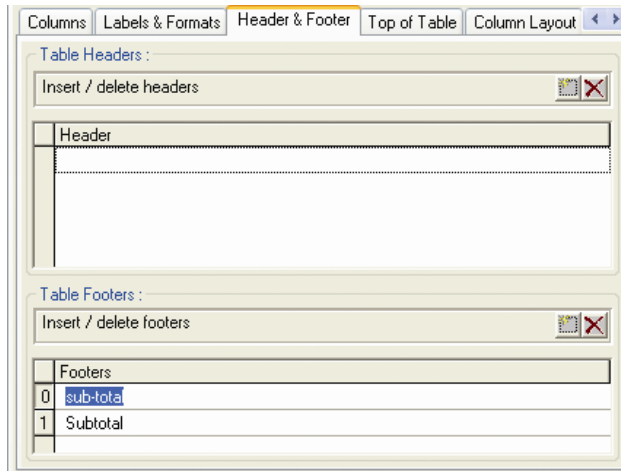


Table 8-10: Tab for the definition of header and footer lines

- 2) In the *Insert / Delete headers* list, define a set of labels using the buttons from [TABLE 8-9: INSERT AND DELETE LIST ENTRIES](#).
- 3) In the *Insert / Delete footers* list, define a set of labels using the same buttons.

### 8.8.4. Determining Table Tops

For each table, you need to specify the algorithms used to search the top. There are three options:

- Searching the label line (based on the labels defined for each column, see section [DEFINING COLUMN LABELS AND FORMATS](#)).
- Searching the header (based on the list of header strings defined, see section [DEFINING HEADER AND FOOTER LINES](#)).
- Specifying a fixed position as distance from the top of page.

These methods can also be combined.

### Determining the Table Top

To determine the table top:

- 1) In the *Analysis* tab, select the *Top of Table* tab.

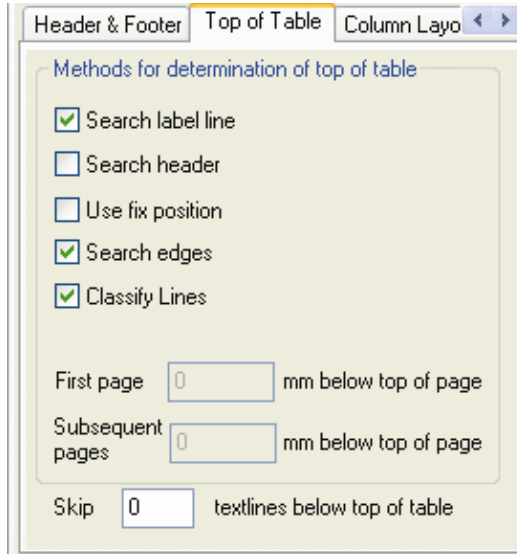


Figure 8-33: Tab for determining table tops

- 2) Select one or more of the following options:

- Check *Search label line*.
- Check *Search header*. This setting is only recommended in combination with a fixed column layout. Use with the *Header & Footer* tab to determine table top and bottom.
- Check *Use fix position* and specify the distance from the top of page in millimeters both for the first and for all subsequent pages. If you analyze the document, the specified position will be highlighted in the document area. This setting is only recommended in combination with a fixed column layout.
- Check *Search edges*. This setting when checked enables Table Analysis to find table columns by searching for edges. As it finds columns, you would select the columns that you defined for the table on the *Column* tab. To make this mapping, use either script, or for invoices, the tab *Column Mapping* that has mapping with fields that are predefined. Within the script, you can navigate through the cells of the columns to check the content. The combo box on this tab contains the column names that are defined on the *column* tab. You can select the column name that corresponds to the Description, Price, and so on. You can define the table columns that you want to write to the workdoc in the *Column* tab. (See section [DEFINING TABLE COLUMNS](#)). With multiple options, the analysis stops when the first criterion is met.

- 3) Optionally, you can specify to skip a number of text lines below the table top by typing a number into the *Skip...* text box. This should only be applied to tables with a fixed layout.

### 8.8.5. Defining Column Layouts

For each table, specify the algorithms used to determine the column layout. There are two options:

- Layout analysis, based on the labels of the label line and the entries of the individual rows.
- Fixed layout, using positions for the left and right border of each column.

#### Defining the Layout

To define the column layout:

- 1) In the *Analysis* tab, select the *Column Layout* tab.

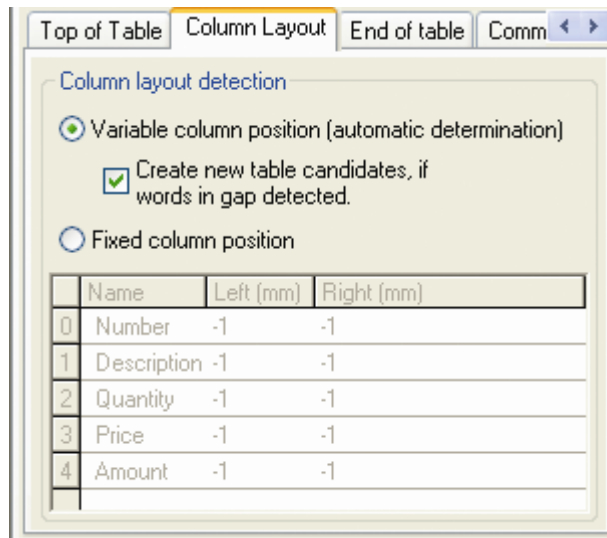


Figure 8-34: Tab for the definition of column layouts (initial state)

- 2) Do one of the following:

- For variable layout: Select the *Variable column position* option. Optionally, you can check *Create new table candidates*. This is useful in cases where a column is not represented by a label. However, it implies the creation of a large number of candidates, thus effecting the time required for the table analysis.
- For fixed layout: Select the *Fixed column position* option. In the table at the bottom of the tab, specify the distance of the left and right column borders from the left border of the page in millimeters. These ranges will be highlighted in the document area.

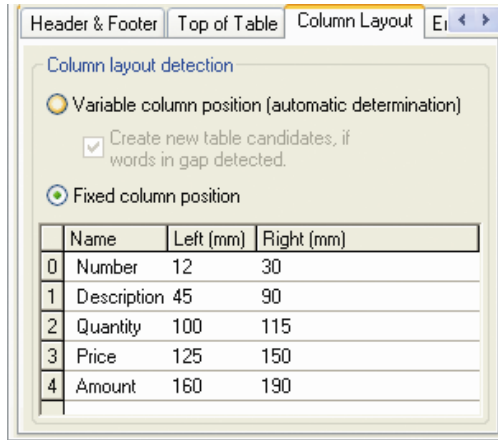


Figure 8-35: Tab for the definition of column layouts (sample fixed layout)

### 8.8.6. Determining Table Bottoms

For each table, you need to specify the algorithms used to search the bottom. The available options are:

- Searching for variations in the text line distance.
- Searching a graphical line.
- Searching the footer (based on the list of footer strings defined, see section [DEFINING HEADER AND FOOTER LINES](#)).
- Searching for empty cells in a control column.

These methods can also be combined.

#### Determining the Table Bottom

To determine the table bottom:

- 1) In the *Analysis* tab, select the *End of Table* tab.

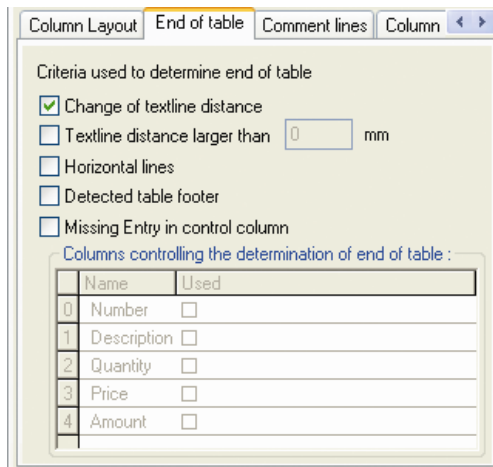


Figure 8-36: Tab for determining table bottoms

- 2) Select one or more of the following options:

- Check *Change of textline distance* if the table ends when the text line distance changes by more than 50%.
- Check *Textline distance larger...* and enter a value into the associated text box if the table ends when the text line distance exceeds a predefined threshold.
- Check *Horizontal lines* if a horizontal line indicates the end of the table.
- Check *Detected table footer* if one of the footers defined previously indicates the end of the table. This option must be checked if your tables contain empty rows.
- Check *Missing entry in control column* if an empty cell in specified columns indicates the end of the table. In the table at the bottom of the tab, mark the *Used* checkbox for at least one column.

With multiple options, the analysis stops when the first criterion is met.

### 8.8.7. Managing Comment Lines

Comment lines are automatically ignored if two requirements are met.

- There are at least two lines with cells above the comment.
- There is at least one line with cells below the comment.

#### Detecting Comment Lines

To detect comment lines:

- 1) In the *Analysis* tab, select the *Comment lines* tab.

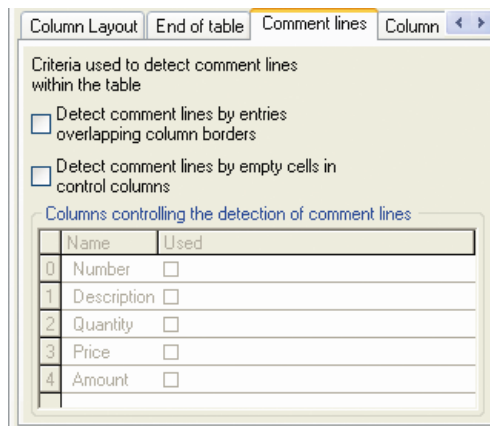


Figure 8-37: Tab for handling comment lines

- 2) Select one or both of the following options:
  - Check.... *overlapping column borders* if entries spanning multiple columns indicate a comment. In the table at the bottom of the tab, mark the *Used* checkbox for at least one column.
  - Check ... *empty cells in control columns* if empty cells in specified columns indicate a comment. In the table at the bottom of the tab, mark the *Used* checkbox for at least one column.

With multiple options, the analysis stops when the first criterion is met.

### 8.8.8. Using Field Inheritance

To allow a coupling between parent and child settings, the concept of inheritance has been introduced. Simply stated, child classes inherit all fields from their parents. If the Table Analysis engine is selected for the table field in the child class, a copy of the parent settings is created, which may be further modified. For this purpose, the table settings have been subdivided into three groups:

- **Column settings** – the definition of the number of columns and their names.
- **Extraction settings** – all parameters controlling the extraction algorithms.
- **Validation settings** – the definitions of formats for the individual columns.

Depending on the inheritance settings, the corresponding controls in the table settings tabs will be disabled.

*Note: If a child class is created manually, without passing the SLW process, it is necessary to activate the Brainware Table Extraction engine for the 'Table' field at the child class level.*

#### Using Field Inheritance with Tables

To use field inheritance with tables:

- 1) In the *Analysis* tab, select the *Columns* tab.

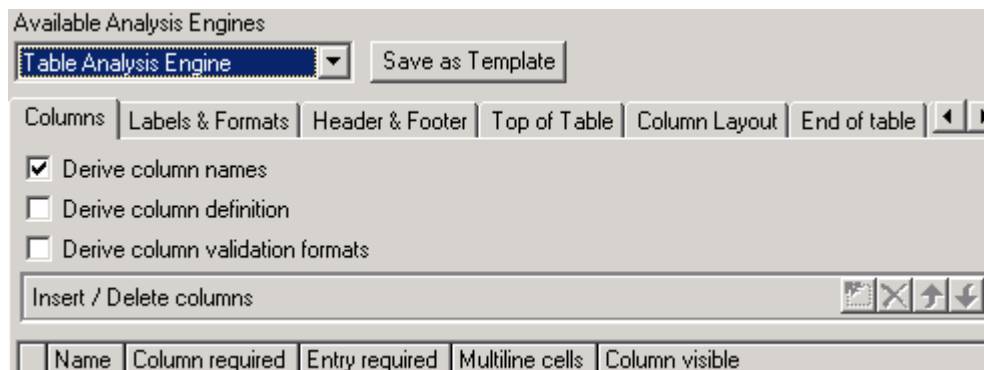


Figure 8-38: Tab for inheritance settings

- 2) Check *Derive column names* to use the same set of columns as the parent class. All controls related to adding, deleting, renaming and sorting columns will now be disabled. The *Derive column definition* and *Derive column validation formats* checkboxes will be enabled.
- 3) Optionally, check *Derive column definition* to inherit column settings..
- 4) Optionally, check *Derive column validation formats* to inherit validation settings..

### 8.8.9. Column Mapping

After you select the columns from which you would extract information for the Workdoc in the column tab, you can search for certain table columns in a batch. This is most commonly used if you are working with invoices. To search for columns, you would select *Search Edges* in the *Top of Table* tab (See section [DETERMINING TABLE TOPS](#)) and then use the *Column Mapping* feature in Table Analysis.

## Mapping Table Columns

To map table columns:

- 1) On the *Analysis* tab, select the *Column Mapping* tab.

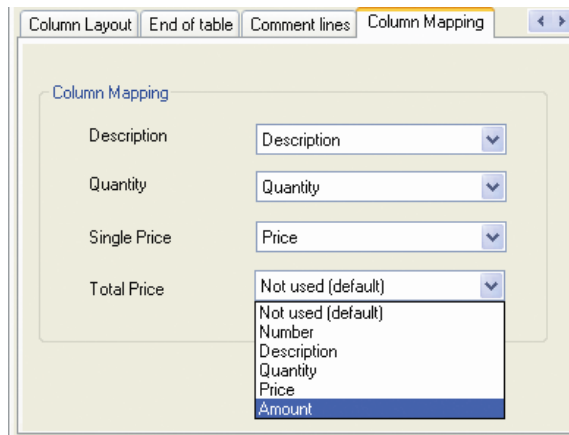


Figure 8-39: Tab for table column mapping

- 2) Select the drop-down arrows to choose the item that best describes your column. When mapping columns for invoices, usually the table columns “description,” “quantity,” “single price,” and “total price” are the most relevant. In this case, you can use the default mapping methods that are defined on the tab “column mapping.” Special mapping methods are used to identify the correct table columns (See section [DEFINING TABLE COLUMNS](#)).
- 3) Click *OK*.

Another possibility for defining table columns is to define the mapping within the script.

If the mapping is incorrect, it must be corrected for the verification (See section [CORRECTING TABLE FIELDS](#)).

## 8.9 Configuring Associative Search Analysis

The Associative Search Engine searches a data source – typically either a semi-colon delimited spreadsheet or a database. This data source, which is also sometimes referred to as a search pool, is used to match data in the search pool to Workdoc data used in classification. In many applications, this will be a vendor name. The Classname Format and Field Content Format are defined after the data from the data source has been imported.

The data source can be created in any spreadsheet application, word processing program, or text editor like WordPad or Notepad.

The table in the *General* tab shows the attributes that are found in the data source.

In the ID column, set the Supplier’s ID to establish this field as the primary key for your associative search.

The Associative Search engine performs a full search by default. To filter certain attributes configure the Multi-column Attribute search extension by checking the appropriate columns in the *Filter* column. For details see section [8.9.4 SETTING UP THE ANALYSIS](#).

Leave the default settings for the fields to search on (square checkboxes).

### 8.9.1. Usage of Associative Database Search Engine

To set the Associative Search engine properties, switch to Definition Mode, Show Class and Field Properties. Go to *Analysis* tab, select *Associative Search Engine* from the drop down list of *Available Analysis Engines*. The *Used core engine* drop down list has two values: “Brainware V2” and “Brainware V3”:

Used core engine: [Brainware V2 |v| ]

By default *Brainware V3* is selected. If loaded ADS settings are “old” (i.e. the version does not include the option yet), then the default value will be *Brainware V2*. This is to support backwards compatibility. In Designer GUI the setting will be positioned at *Analysis* .

There is also a Registry key that can be used to force all Associative Search engine fields in all projects on a Windows system to use a specific version of the core Brainware engine. The key is a DWORD variable “ASEngineVersion” under Brainware\ Cedar key. Set it to “3” for “Brainware V3” and to “2” for “Brainware V2”.

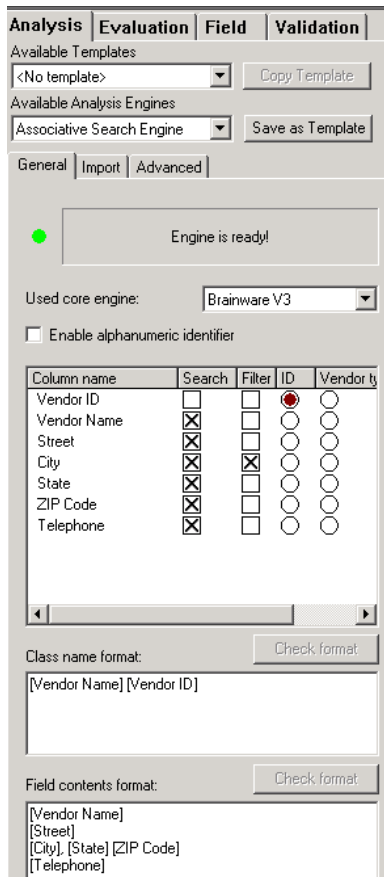


Figure 8-40 General Section of Associative Search Engine

### 8.9.2. Creating Data Sources

Configuring data sources for Automatic Supervised Learning is a process that requires steps in both Windows Explorer and Designer.

#### 8.9.2.1. Windows Explorer Steps

##### Creating a Project Structure



- 1) In Windows Explorer, create a project folder
- 2) Under that, create the following folders:
  - Batches
  - Common Learnset
  - Export
  - Global Project
  - Input
  - Miscellaneous
  - Vendor Pool
- 3) These folder names are suggestions; you can use whatever schema fits your project best.
- 4) Under Global Project and Common Learnset, create folders for each of the derived classes that will be created by WebCenter Forms Recognition based on the name of the first field.

## Create a Reference File from a Spreadsheet or Text Editor

- 1) In Excel, WordPad, or Notepad, create a semi-colon delimited file with the following fields:

Vendor\_ID;Internal\_ID;Vendor\_Name;Vendor;Vendor\_Type;Vendor\_Address;Vendor\_City;Vendor\_State\_Vendor\_ZIP.

The following fields are mandatory:

**Vendor ID:** It must be a unique numerical designation for the vendor name.

**Internal ID:** It is a unique sequential designation for each record in the file.

**Vendor Name:** Is a short form of the Vendor's full name.

**Vendor:** Is the full name of the vendor.

**Vendor Type** is an optional field. Vendor Type is used to influence a Supervised Learning setting that indicates the threshold for invalid fields, that, if exceeded, will force a document, or, by extension, members of a DocClass, to be learned. In most cases, this value is set to 40 percent. The value for Vendor\_Type can either be 0, 1 or 2, with 0 being the default value and the value that is assumed if there is no Vendor\_Type field in the \*.csv file. A value of 0 for Vendor\_Type forces all documents classified by that vendor name to be added to the local Learnset. A value of 2 indicates that no document in the class can be added to the Learnset. A value of 1 indicates that only the first document in the class can be added to the Learnset.

You can always designate another field as Vendor Type, as long as the values are correct.

The rest of the fields are conditional – it is only required if you plan to use Address Analysis. See [ADDRESS ANALYSIS](#).

*Note: The actual name of the columns in the header of the \*.csv file can be different, but the content must match the data that would be expected for columns that have the names shown in the example.*

*Note: If you use Excel, place all fields in the first column (Do not place your data individually in Column A, Column B, Column C, etc.) and use semi-colons to delimit your data, as shown in Step 1. Also, ensure that field names do not contain periods or other special characters.*

You can choose to have header rows or not; this will affect a WebCenter Forms Recognition setting later (see [IMPORT SETTINGS](#)).

- 2) Save the file with a \*.csv extension in the Project Root folder. This file is the lookup table that WebCenter Forms Recognition will use for Associative Learning.

*An address must be on the document that matches the address in the \*.csv file.*

## Creating a Reference File Using a Database Connection

An alternative to creating the \*.csv file is to use data from a similarly constructed database and establishing an ODBC connection to it.

### 8.9.2.2. WebCenter Forms Recognition Designer Steps

In Designer, create and save a new project as you normally would. Use the following settings:

#### Project Input Settings

- 1) Set *File Options* as *Other*, with the default file extension of \*.tif.

- 2) Enable *Save Execution Results to Workdoc*.
- 3) Set the batch root directory and the image root directory to the folder you created for batches.

**Project Definition Mode Settings**

Accept all defaults.

**Train Mode Settings**

- 1) Select *Add documents to Learnset*.
- 2) Set the *Learnset Base Directory* to the folder you created for Local Project

**Runtime Mode Settings**

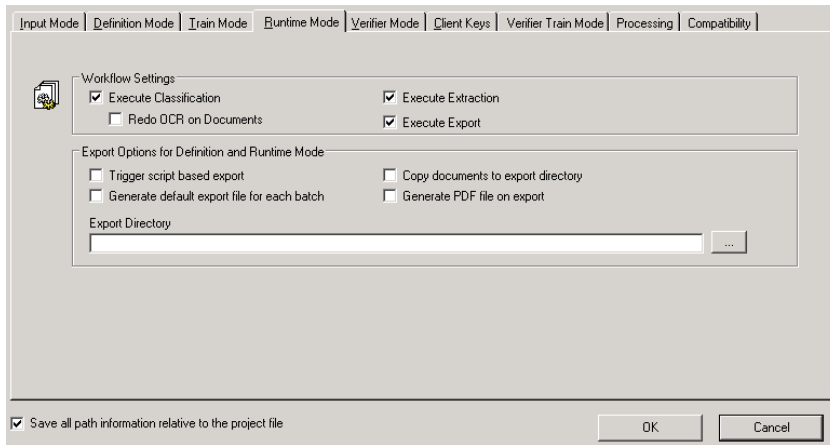


Figure 8-41: Runtime Mode Settings

Enable *Execute Classification* and *Execute Extraction*, and keep all other defaults.

**Verifier Mode Settings**

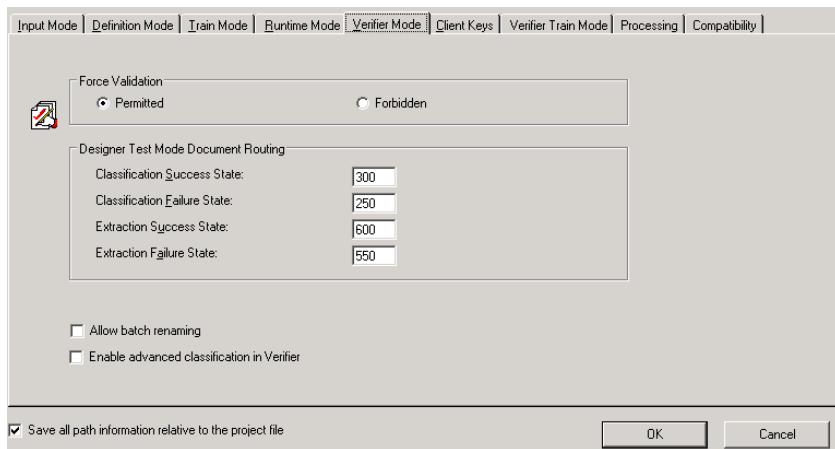


Figure 8-42: Verifier Mode Settings

Keep all defaults.

**Project Tree Settings**

Keep as is.

**Verify Train Mode Settings**

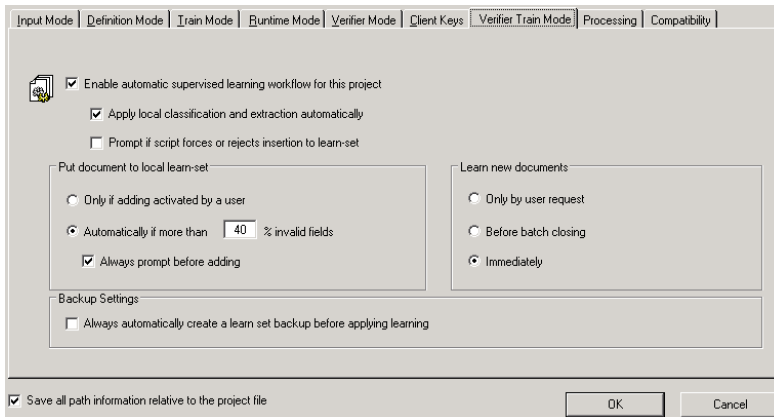


Figure 8-43: Verifier Train Mode Settings

- 1) Switch to the *Verify Train Mode Settings* tab.
- 2) Select *Enable Automatic Supervised Learning*.
- 3) Select *Apply local Classification and Extraction automatically*.
- 4) Select *Save all path information relative to project file*.

### 8.9.2.3. Defining the Project

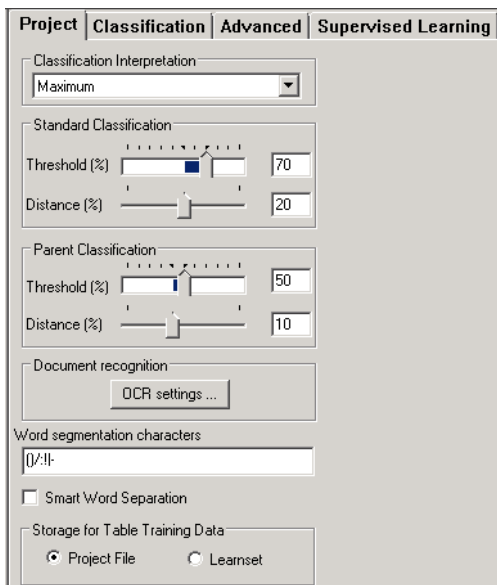


Figure 8-44: Defining the Project

- 1) Switch to Definition Mode.
- 2) Right click on the project name and select *Show Properties* from the shortcut menu.
- 3) On the *Project* tab, leave all default settings for Classification Interpretation, Parent Classification, and Word Segmentation Characters.

### 8.9.2.4. Setting OCR Options

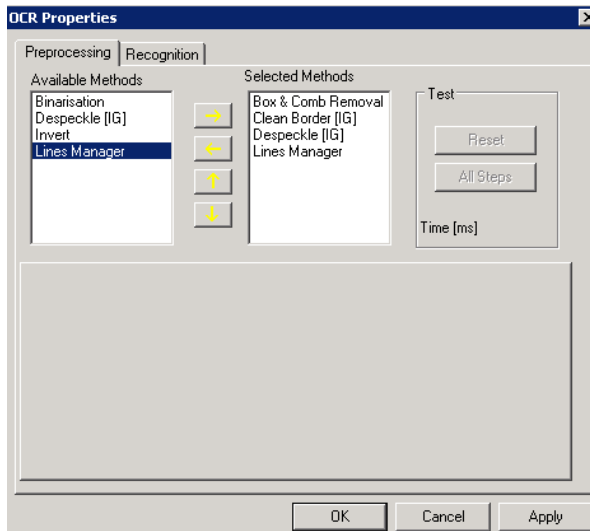


Figure 8-45: Setting OCR Options

- 1) Still on the *Project* tab, click *OCR Settings* in the middle of the tab.
- 2) On the Pre-processing Options tab, select the following recommended settings:
  - Box and Comb Removal
  - Clean Border
  - Despeckled
  - Lines Manager

To make the selections, select the items above from the *Available Methods* column and click the arrow pointing to the right for each.

*A word on despeckling: This process, which removes dots and visual dust from your documents, sometimes detects decimal points and removes them. This, of course, can be undesirable. If your documents are generally light, do your first preprocessing on them without despeckling to test the results. If the quality is still poor, enable Despeckling and preprocess them again.*

If applicable to your environment, you can also select **binarisation** (to convert grayscale images to black-and-white) and **invert** (for reversing white-on-black pages to black-on-white pages.)

### 8.9.2.5. Setting Specific OCR Tolerances

Still on the *Pre-processing Options* tab, now set the specific tolerances for each of the OCR settings you chose. To do this, click on a method in the *Selected Methods* column:

- For *Box and Comb removal*, select Box Removal and Automatic.

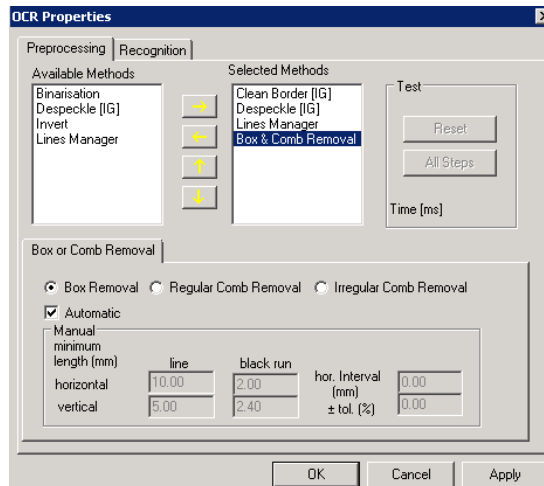


Figure 8-46: Box and Comb Removal Setting

- For *Clean Border*, do nothing. No additional options are available.
- For *Despeckle*, do nothing. No additional options are available.
- For *Lines Manager*, select Remove Horizontal Lines, Remove Vertical Lines, and Repair Characters. Accept all defaults.

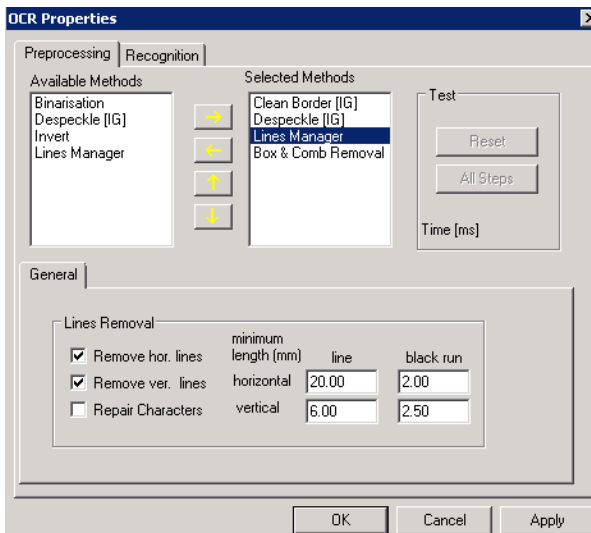


Figure 8-47: Lines Manager Setting

### 8.9.2.6. Establishing Recognition Settings

- 1) Click on the *Recognition* tab.
- 2) For *Recognition Type*, select OCR.
- 3) Select a Fine Reader engine in Available Engines (Select FR8 in this case).

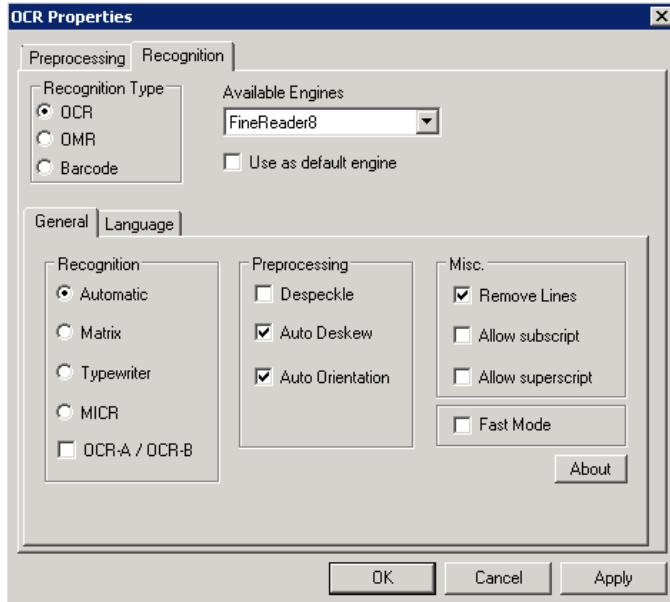


Figure 8-48: Recognition Settings

On the *Recognition* tab, there are two tabs: *General* and *Language*

On the *General* Tab:

- 1) For *Recognition*, choose *Automatic*, unless all your documents come from typewriters or dot-matrix printers.
- 2) For *Preprocessing*, accept the defaults of auto-deskew and auto-orientation and select *Despeckle* if you have previously tested this option and were satisfied with its affect on your documents.
- 3) For *Miscellaneous*, accept the default of *Remove Lines*.

On the *Language* tab, select Digits and English.

### 8.9.2.7. Setting Your Base Document Class

- 1) Returning to the tabs on the left side of your screen, click on the *Classes* tab.
- 2) Right click on your project name. From the shortcut menu, select *Insert Base DocClass*.
- 3) Type a name for the class and then click *OK*.

### 8.9.2.8. Defining Your Reference Field

- 1) Click the *Field* tab.
- 2) Right click anywhere on the *Field* tab. On the shortcut menu, select *Insert Field Definition*. Assign a field definition (field name) to the first field and click *OK*. Right click in the field and select *Show Properties*.
- 3) On the Analysis Editor on the right side of your screen, select *Associative Search Engine* from the Available Analysis Engines selection box.

## 8.9.3. Importing Reference data for the Associative Search

### 8.9.3.1. Import settings

Figure 8-49: Import Settings

The Import tab gathers settings for the *Import source* and the *Import Destination*.

For *Import Source* select whether data from the supplier pool (reference file) should be imported manually or automatically. For manual import specify either the CSV file path or the ODBC source.

*Import Destination* determines where the reference file will be saved. The recommended setting is *Save Supplier Pool inside Project*.

### 8.9.3.2. Importing Your Reference Data from a \*.CSV File

- 1) On the *Analysis Editor*, click the tab *Import*.



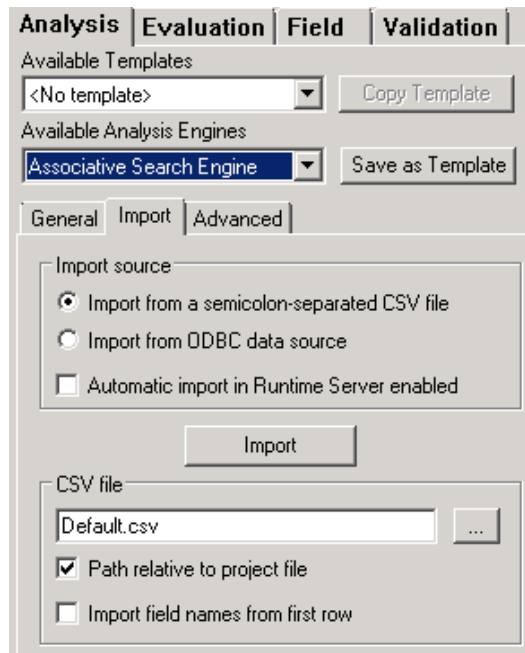


Figure 8-50: File import tab

- 2) Choose *Import from a semicolon-separated CSV file*
- 3) For *Import Source*, browse to the \*.csv file you created.
- 4) Deactivate *Path Relative to Project File*.
- 5) If your \*.csv file has header rows, select *Import field names from first row*. If your file does not have header rows, leave this checkbox empty.
- 6) Click *Import*.

### 8.9.3.3. Importing Your Reference Data via ODBC

To establish an ODBC connection for your reference data:

- Still in the Analysis Editor, click the *Import* tab .
- Choose *Import from ODBC data source*
- Supply the source, login information and SQL statement to access the data for reference.

### 8.9.4. Setting up the Analysis

- 1) Still in the Analysis Editor, click the *General* tab.

The screenshot shows the 'General' tab of the Analysis Editor. At the top, there are tabs for 'General', 'Import', and 'Advanced'. A green dot indicates the engine is ready. Below this, the 'Used core engine' is set to 'Brainware V3'. There is a checkbox for 'Enable alphanumeric identifier' which is currently unchecked. A table with columns 'Column name', 'Search', 'Filter', 'ID', and 'Vendor ty' is present. The 'Search' column has checkboxes for all fields. The 'Filter' column has checkboxes for 'City', 'State', 'ZIP Code', and 'Telephone'. The 'ID' column has a radio button selected for 'Vendor ID'. Below the table, there are two text boxes for 'Class name format' and 'Field contents format', each with a 'Check format' button. The 'Class name format' box contains '[Vendor Name] [Vendor ID]' and the 'Field contents format' box contains '[Vendor Name]', '[Street]', '[City], [State] [ZIP Code]', and '[Telephone]'.

Column name	Search	Filter	ID	Vendor ty
Vendor ID	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="radio"/>	<input type="radio"/>
Vendor Name	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="radio"/>	<input type="radio"/>
Street	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="radio"/>	<input type="radio"/>
City	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="radio"/>	<input type="radio"/>
State	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="radio"/>	<input type="radio"/>
ZIP Code	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="radio"/>	<input type="radio"/>
Telephone	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="radio"/>	<input type="radio"/>

Figure 8-51: Establishing Analysis criteria

- 2) Notice the table with the fields from your \*.csv file.
- 3) On this table, set ID for Vendor ID to establish this field as the primary key for your associative search by selecting the round radio button. Leave the default settings for the fields to search on.
- 4) Enable the Multi-column Attribute search extension of the Associative Search engine by checking the boxes of appropriate fields in the Filter column (see section [8.9.4.1 MULTI-COLUMN ATTRIBUTE SEARCH](#)). A data source reimport is needed when settings are changed.
- 5) Type a format for ClassName Format.
- 6) Type a format for Field Contents Format.
- 7) If desired, enable Address Analysis and determine which address fields should be included or excluded.

#### 8.9.4.1. Multi-column Attribute search

This search extension limits the vendor search to certain attribute values. The search filter shows only results with the same value for an attribute, e.g. value "Washington" for city. Select the attributes for the search filter as described above in [8.9.4 SETTING UP THE ANALYSIS](#)

under 4). Attribute values are always case-sensitive. Perform a data source reimport after changing settings. It is not advised to select all rows.

The Attribute values are set in custom script with following parameters:

1. Call `.ClearFilterAttributes()`
2. Then call: `.AddFilterAttribute("Attribute Name 1", "Attribute Value 1")`  
`.AddFilterAttribute("Attribute Name 1", "Attribute Value 2")`  
`.AddFilterAttribute("Attribute Name 2", "Attribute Value 1")`
3. Continue for all selected attributes.

Filtering multiple values of one attribute("OR" operation) is supported for the first attribute only. The different attributes are connected with the AND operator.

The extension can be used for the Verifier Vendor search or with the RTS preextract event.

To use the extension with the Verifier search button, configure the `.SearchField` or the `.AnalyzeField` call as shown in the following script example:

```
Dim theSupplierSettings As Object
Set theSupplierSettings = FieldAnalysisSettings
Dim theAdsSettings As CDRADSLib.SCBCdrSupExSettings
Set theAdsSettings = theSupplierSettings

theAdsSettings.ClearFilterAttributes
theAdsSettings.AddFilterAttribute "SupplierName", "VAN"
theAdsSettings.AddFilterAttribute "SupplierName", "VAN3"
```

For Associative Search filtering with RTS, configure the extension in the VendorName (or VendorASSA) field preExtract event. This enables the filtering before the Attribute search is implemented in the extraction:

```
Private Sub VendorName_PreExtract(pField As SCBCdrPROJLib.SCBCdrField,
pWorkdoc As SCBCdrPROJLib.SCBCdrWorkdoc)

    Dim theSupplierSettings As CDRADSLib.SCBCdrSupExSettings
    Dim theDocClass As SCBCdrDocClass
    Dim theAnalysisSettings As ISCBCdrAnalysisSettings
    Dim theObject As Object
    Set
theDocClass=Project.AllClasses.ItemByName(pWorkdoc.DocClassName)

    theDocClass.GetFieldAnalysisSettings "VendorName","German",
theAnalysisSettings
    Set theObject = theAnalysisSettings
    Set theSupplierSettings = theObject

    theSupplierSettings.ClearFilterAttributes()
    theSupplierSettings.AddFilterAttribute "SupplierName", "VAN"
    theSupplierSettings.AddFilterAttribute "SupplierName", "VAN3"

End Sub
```

The Multi-column Attribute search is supported by the *Brainware V3* core engine.

### 8.9.4.2. Advanced Settings

Advanced Settings enable you to establish settings for Candidate Detection and Validation Threshold.

To establish these settings:

- 1) Click the *Advanced* tab.

Left	Top	Width	Height	Values in %
0	0	100	25	→ [ ]
0	90	100	10	→ [ ]

Phrase	Status

Figure 8-52: Establishing Advanced Settings

- 2) Change the default settings for Height and Width and Threshold and Distance if the address and vendor name are not in the default Viewer area. If they are not, the following settings are recommended:
  - For *Candidate Detection*, set Height and Width to 100 percent
  - For *Conditions for Validation*, set threshold to 30 percent and distance to 10 percent.

### 8.9.4.3. Establish Settings for ClassName Format and Field Contents Format

- 1) Return to the *Fields* tab on the left side of your screen and insert any additional fields that need to be learned.
- 2) Still on the left side of your screen, click on the project name. On the *Classification* tab on the right side of your screen, select your Base DocClass as the default classification result. This setting tells WebCenter Forms Recognition to assign all documents to this class if Associative Learning fails to classify them.

- 3) For now, leave the settings on the *Advance* tab as is. You will establish Validation Settings at the document and field levels later.
- 4) On the right side of the screen, click the *Supervised Learning* tab. Select *Do Smart Decision*.

### 8.9.5. Input of Documents

- 1) View Document Input Mode.
- 2) Select the documents from their directory.
- 3) Set validations at the document and field level

## 8.10 Setting Up the Field Evaluation

WebCenter Forms Recognition supports two methods for candidate evaluation:

- Custom methods implemented as Visual Basic scripts
- Evaluation using Brainware

Setting up the evaluation method is not required, if:

- You use custom methods or
- You use zone analysis with a 1:1 zone/field relationship.
- In all other cases, the evaluation method must be specified. During evaluation, WebCenter Forms Recognition analyzes the areas around each candidate for characteristic properties that can be learned.

### Task Prerequisites

The prerequisites for this task are:

- The program is in Definition Mode.
- The panel on the left side of the window displays the *Fields* tab in the foreground.
- On the right side of the window, the tabs with class/field properties are visible.

### Setting the Evaluation Method

To set the evaluation method:

- 1) In the *Fields* tab on the left side of the window, select a field.
- 2) On the right side of the window, select the *Evaluation* tab.
- 3) Under Available Evaluation Engines, select Brainware Extraction.
- 4) To specify the geometry of the area around the candidate that is to be analyzed, select the *Microlayout* tab at the bottom of the window. Note that microlayout settings only have a small influence on extraction results. They need a lot of samples to be distinguished. In most cases, the default setting will work fine.
- 5) Select one of the following options:

Option	Description
Table-based	The context of the candidate is taken from a top-asymmetric region around the candidate, the line to the left and the column above. This configuration is the best for candidates in tables. This is the default microlayout.
Left-Top oriented	The context of the candidate is taken from a left-asymmetric region around the candidate and the line to the left. This configuration is the best for candidates in captions or after colons.
Symmetric	The context of the candidate is taken from a symmetric region around the candidate. This configuration is the best for candidates in running text.
Line oriented	The context of the candidate is taken from the region to the left and to the right of the candidate. This configuration is the best for candidates in lines.

Table 8-11: Microlayout options

## 8.11 Brainware Field Extraction Engine for Generic Fields Extraction

### 8.11.1. Description

Brainware Field Extraction engine introduces a dimension in automatic indexing (extraction) of header field data on a generic level (i.e., when multiple logical document categories (classes) are handled with a single internal knowledgebase). The engine is based on an advanced N-gram Triton technology and provides high quality field extraction based on statistical evaluation of:

- Surrounding (keywords) around the learned field
- Format of the learned field
- Typical location of the learned field
- Correlation between different fields learned in context of one document class.

All the information above is learned automatically. The engine also autonomously extracts candidates while time consuming specification of regular expressions via the Format Analysis Engine is no longer required for the processing.

The engine fully supports supervised learning workflow.

Brainware Field Extraction (BFE) engine can be configured and trained in the same way as the Brainware Extraction engine. See sections [FIELD-LEVEL SETTINGS](#) or [SETTING UP THE FIELD EVALUATION](#) for more details.

The engine can be assigned to any header field of a document class using the *Evaluation* tab of the field definition settings and selecting *Brainware Field Extraction* from the list of available engines:

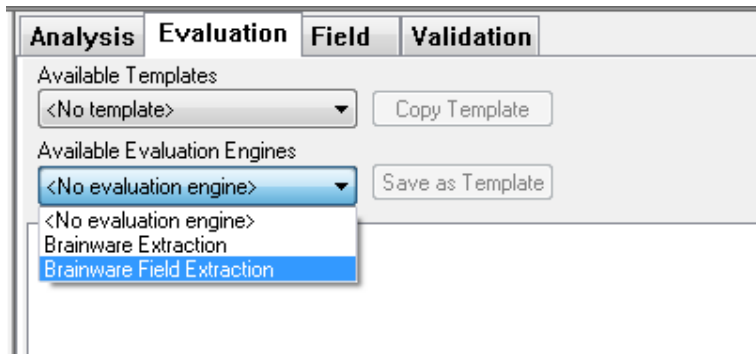


Figure 8-53: Evaluation tab – Brainware Field Extraction

When the engine is assigned it can be configured similar to Brainware Extraction engine. As for the Brainware Extraction engine the BFE settings view also shows the brief “learned” status for the selected field:

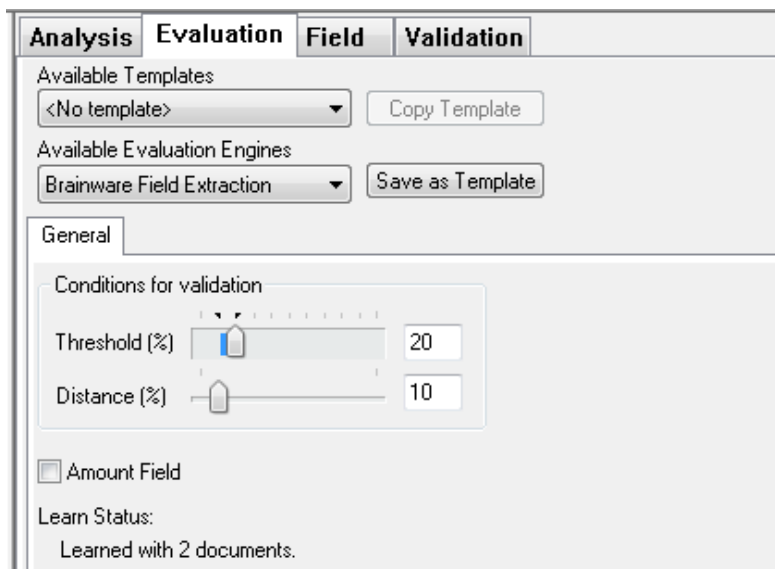


Figure 8-54: BFE settings – “learned” status for selected field

There are a couple of differences related to configuring of the BFE engine as opposed to the Brainware Extraction engine:

- Format Analysis engine pre-processing and time consuming configuration is NO longer required for the BFE engine function. See “Restrictions” and “Scripting” subsections below for more details, particularly in regards to the case, when the candidates extracted by the Format Analysis are nevertheless supposed to be kept as a part of the extraction’s outcome.
- Microlayout settings are not supported by this engine.
- There is one extra check box setting called “Amount”. This setting is supposed to be checked for those fields that have to be treated as amounts. In this case, as one of the internal extraction conditions the engine will consider all “amount” fields as the ones with potential numeric correlation.

### 8.11.2. Restrictions

Unless specifically handled, the Brainware Field Extraction engine deletes all previously available candidate objects (e.g., the ones created by the preceding execution of the Format Analysis engine) and creates and evaluates self-extracted candidates.

### 8.11.3. Scripting

The following script sample can be used if for any reason it is desired that the extracted BFE candidates are to be replaced with the ones extracted by the Format Analysis engine or if both kinds of candidates are supposed to be kept in the candidates array.

```
Private Sub Date_PostEvaluate(pField As SCBCdrPROJLib.SCBCdrField, pWorkdoc As SCBCdrPROJLib.SCBCdrWorkdoc)

    Dim i As Long
    For i = pWorkdoc.Fields.ItemByName("Date").CandidateCount - 1 To 0 Step -1
        ' Comment the code line below if you'd like to keep both types of candidates (from FAE and BFE)
        pWorkdoc.Fields.ItemByName("Date").RemoveCandidate(i)
    Next i

    pWorkdoc.DocState = CDRDocStateClassified
    pWorkdoc.Fields.ItemByName("Date").FieldState = CDRFieldStateReset
    Project.AllClasses.ItemByName("Invoices").AnalyzeField pWorkdoc, "Date"

End Sub
```

### 8.11.4. Usage & Notes

Due to quality, performance, and capacity of the disk space required per one class' BFE Learnset, it is not recommended to use the Brainware Field Extraction engine for the "vendor" level training, i.e., when a class of similar documents with very similar layout is supposed to be trained with just a few (1-5) documents to achieve perfect extraction results for particular high-volume vendor class.

General recommendation is to use the engine for generic training only, i.e., for data extraction from no categorized documents. In this connection, the engine has to be trained with at least of 10 documents per extracted fields, while the ideal amount of documents per field is **50-100** samples. Good OCR quality for the trained documents is preferred but not a mandatory requirement.

Also, exceptional Learnset samples in terms of both document quality and layout uniqueness are generally acceptable.

Even though the engine is integrated in supervised learning workflow (SLW), due to above mentioned reasons, it makes sense to keep using the former Brainware Extraction engine for automatic SLW vendor level training. For this purpose, it is recommended to assign the old "Brainware Extraction" engine to the header fields of the SLW's root classes using a special dedicated class for generic extraction via Brainware Field Extraction engine. For example, the following classes' hierarchy:



```

Invoices
    Generic
    VendorClass1
    ...
    VendorClassN
Void

```

“Invoices” class is the root template class for SLW training, while “Generic” class defines generic extraction via BFE engine. In this connection, the standard classification result for “invoices” has to be set to “Generic”, so that in case a document has not been classified to one of available vendor class with precise layout extraction “VendorClassX”, it goes to “Generic” class with generic extraction pattern defined. After the generic processing has been applied, the classification result has to be set back to “Invoices” in case further SLW processing is to be maintained. This can be achieved via the following script code placed into “PostExtract” event’s handler of the “Generic” document class:

```

' Cedar Document Class Script for Class "Generic"
Private Sub Document_PostExtract (pWorkdoc As SCBCdrPROJLib.SCBCdrWorkdoc)
    pWorkdoc.DocClassName = "Invoices"
End Sub

```

In some project configurations, when vendor classification succeeds but does not deliver any results for some selected fields, it may also make sense to take an advantage of the “Retain previous extraction results” option and define double extraction processing to combine generic extraction with vendor level extraction.

*Note:* The Brainware Table Extraction (BTE) and Brainware Field Extraction (BFE) engines’ external Learnsets are now stored in a secure encrypted form (this effects “bte.ptb” / “bte.xtr” and “bfe.ptb” / “bfe.xtr” Learnset files stored per class’s Learnset directory, in case an external BTE / BFE Learnset is available for a class).

## 8.12 Training of Header Fields in Normal Train Mode without Configuring Field Formats

### 8.12.1. Description

Training of header fields in Normal Train mode of the Designer application can now be applied without any Format Analysis pre-processing.

Open the required (already OCR-ed) document in Normal Train mode with desired class name assigned, select a header field (in any highlighting mode, i.e., “Highlight All Words”) with Brainware Extraction or Brainware Field Extraction engine assigned and use mouse rubber banding to select the desired words from the document:

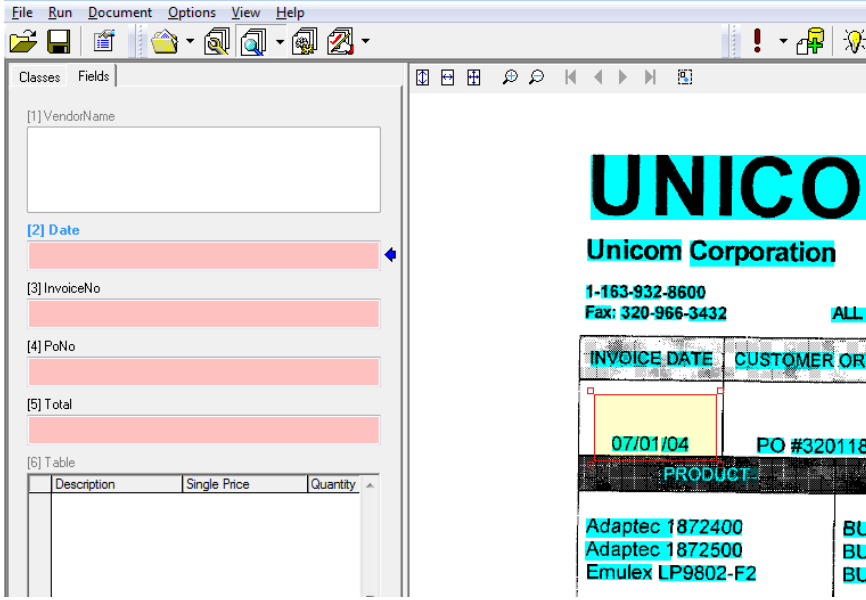


Figure 8-55: Normal Train mode – word selection

The system will then select the available words with their best matching position:

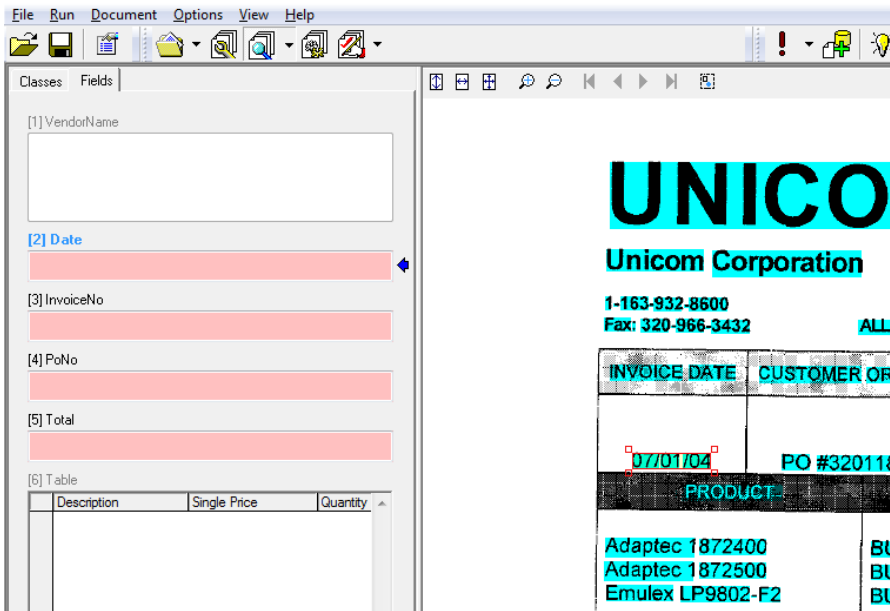


Figure 8-56: Selection of available words

Now double click on the selected area. Internally the system is going to create the candidate object and apply the field assignment as required for further training.

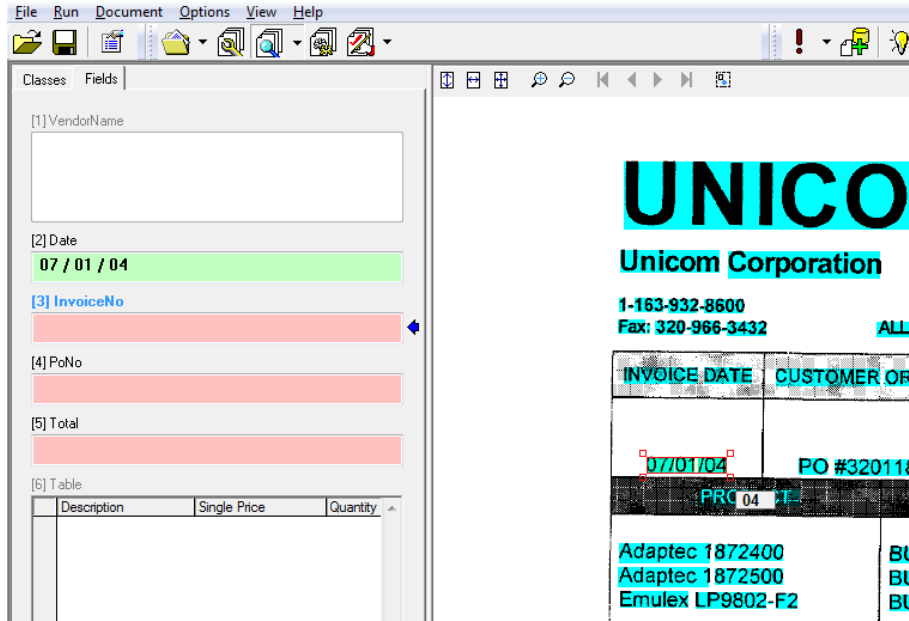


Figure 8-57: Field assignment

### 8.12.1.1. Usage

This feature was primarily designed to allow quick learning of Brainware Field Extraction engine without necessity to define header field formats via Format Analysis engine's regular expressions and/or without a need to configure SLW to use Verifier Train mode for the purpose of Brainware Field Extraction learning.

Note that this feature can be used for training of the former Brainware Extraction engine, as well. On the other hand, unless specifically desired, it is not recommended to utilize it for the Brainware Extraction engine for the following reasons:

- Further extraction with the Brainware Extraction engine requires availability of the expected extraction result among the pre-extracted (via Format Analysis engine) candidates. The engine only evaluates the existing candidates assigning them individual weights. These weights are then used by the general extraction subsystem to assign the final extraction result. Therefore, if extraction result was not among the auto-extracted candidates initially and was just created "run-time" during learning, the probability that the same case will occur for a hypothetical similar document to extract is very high. This will then lead to complete unavailability of the correct extraction result.
- When a document is being trained with Brainware Extraction, the engine processes the learned field's content using it as the single member of internal "Correct Results" Brainware Classifier's class and using all its other available candidates as multiple members of "Wrong Results" class. The further extraction process is close to smart classification between these two classes. Thus, it is clear that overall quality of Brainware Extraction engine's outcome also depends on quality of "Wrong Results" class' content. This content, in case no candidates were available prior to start of the document learning procedure, will be reduced up to just one single member for the "Wrong Results" class, which is going to be the first available word in the document. While such content is sufficient for training / extraction goal in general, it is obviously not optimal in terms of quality.

## 8.13 Learning the Extraction

For Brainware evaluation, you have to take a couple of sample documents and to manually assign some candidates to the respective field. WebCenter Forms Recognition is then able to learn the way the data should be extracted.

### 8.13.1. Creating Learnsets

The set of sample documents that you need to provide for data extraction is called the extraction Learnset. Carefully select the samples as the quality of your Learnset is crucial for the success of the extraction.

When you select the sample documents for the Learnsets, take the following into account:

- Use only documents with good OCR results for the Learnset. When in doubt, highlight the OCR results to review the documents. Never use handwritten documents.
- Use only documents that generate candidates for all relevant fields.
- Where possible, prefer documents that generate a couple of candidates instead of just one for each field. The system analyzes the neighboring words of candidates. It not only learns positive indicators for successful candidates, but also negative indicators.
- The number of samples needed to create a good extraction Learnset depends on the task. In general, you need more samples to cover extraction than to cover classification.

#### Task Prerequisites

The prerequisites for this task are:

- Your project settings must allow for manual addition of documents to the Learnset.
- Your project settings must permit data extraction as runtime functionality.
- Learning data extraction is done class by class. Therefore you should prepare homogeneous sets of sample documents where all documents within a set belong to the same class.
- You can either learn all fields defined for a class at once, or one field at a time. In the first case, your project settings should support selection of the first field when switching to the next document, in the second case your project settings should make sure that the field selection is not changed when switching to the next document.

#### Creating Learnsets for all Fields in a Class

To create Learnsets for all fields in a class:

- 1) Switch to Document Selection Mode.
- 2) Select the first document from the batch that contains the documents for your Learnset.
- 3) Switch to Train Mode. On the left side of the window, the *Classes* tab is in the foreground.
- 4) In the list in the lower section of the *Classes* tab, double click the class you want to train. If you use field inheritance, select the parent class. The *Fields* tab displays the fields of the selected class. The fields are empty, and the first field is selected. Candidate highlighting is turned on automatically.

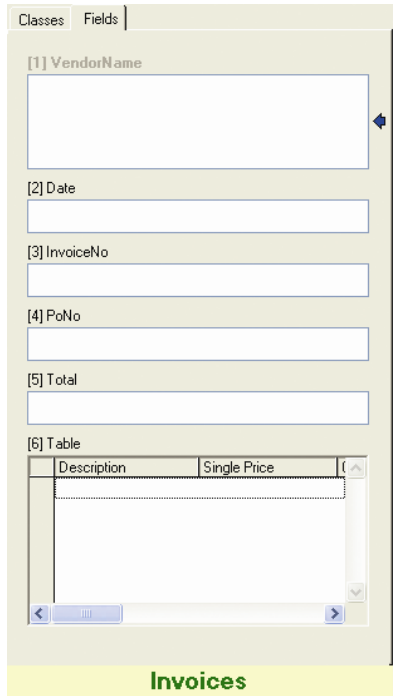


Figure 8-58: Fields tab in train mode



- 5) In the toolbar, click *Classify/Analyze current document* to highlight all candidates for the first field on the document.

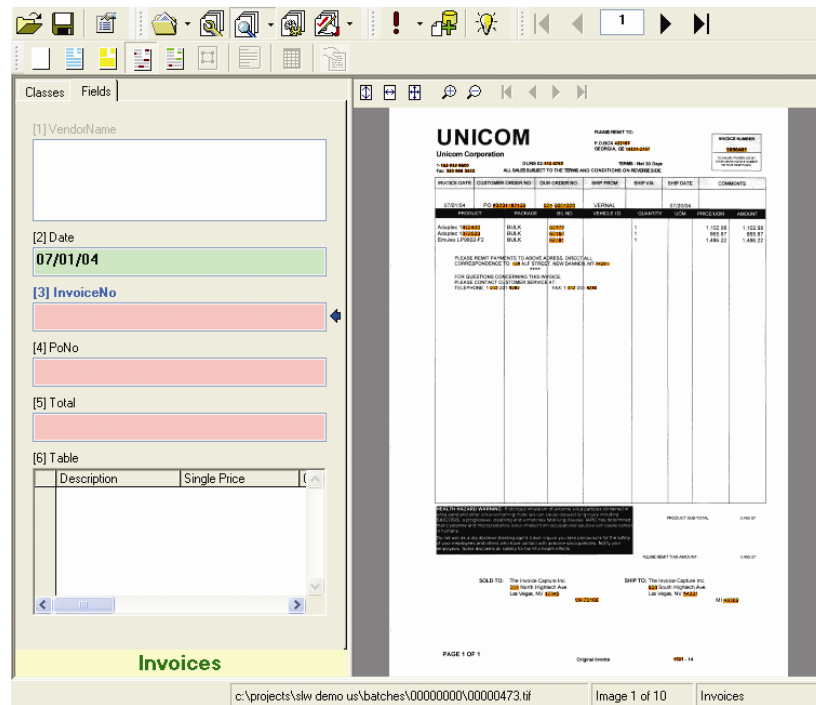


Figure 8-59: Candidate selection in train mode

- 6) To select the correct candidate, just click it. The candidate's text is written to the field, and the field's background color changes to green. The next field is selected automatically, and its candidates are highlighted automatically. When all document fields have a value, the document is automatically added to the extraction Learnset. The program continues automatically with the next document and the first field. All you have to do is to continue choosing candidates until you have enough samples.
- 7) You can check the number of documents in the extraction Learnset in the *Classes* tab.

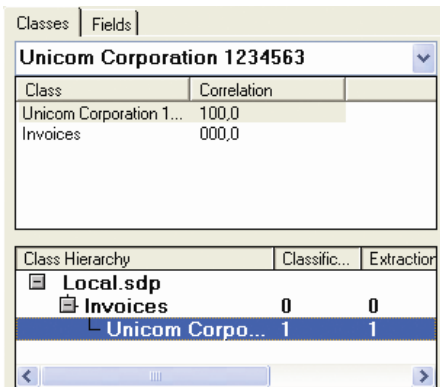




Figure 8-60: Documents in the extraction Learnset

### Creating Learnsets for a Single Field



To create Learnsets for a single field in a class:

- 1) Switch to Document Selection Mode.
- 2) Select the first document from the batch that contains the documents for your Learnset.
- 3) Switch to Train Mode. On the left side of the window, the *Classes* tab is in the foreground.
- 4) In the list in the lower section of the *Classes* tab, double click the class you want to train. The *Fields* tab displays the fields of the selected class. The fields are empty, and the first field is selected. *Candidate highlighting* is turned on automatically.
- 5) On the *Fields* tab, select the field.
- 6) On the toolbar, click  *Classify/Analyze current document* to highlight all candidates for the selected field on the document.
- 7) To select the correct candidate, just click it. The candidate's text is written to the field, and the field's background color changes to green.

In the toolbar, click  *Add document to Learnset* to add the current document to the Extraction Learnset. The program continues automatically with the next document and the selected field.

- 8) Repeat [Step 6](#) to [Step 8](#) until you have enough samples.

Your sample documents may not always be perfect. To handle this, the following options are available:

- You can skip fields or redo training of a field by manually selecting a new field. In this case, click  for manual analysis and click  to manually add the current document to the Extraction Learnset. You may have to confirm that you want to use the document with some fields still being empty.
- You can use words to fill the fields that were not found as candidates. Highlight all words, click the word you want to add and confirm the following message box. Note that in this case you still have to adjust your analysis settings to make sure the candidate will be found in the future.

### 8.13.2. Editing Learnsets

You can edit the Extraction Learnset just like the Classification Learnset (Section [FORMAT ANALYSIS](#)). Just select the Learnset as document input, then right click a class and select *View Extraction Learnset*.

### 8.13.3. Learning

During learning, WebCenter Forms Recognition takes your Learnsets and the manually assigned candidates. It identifies the correlation between successful candidates, unwanted candidates and their environment. It thus learns to extract data automatically if they are similar to the ones that you have selected.

Learning is required:

- Once you have set up your extraction scheme and created Learnsets,
- When you have changed field definitions by adding, deleting, moving or renaming fields,
- When you have changed any parameters affecting the analysis and evaluation methods,
- When you have changed a Learnset.

*Classification and extraction are usually learned together. To learn extraction, proceed as described for classification (See [LEARNING](#)).*

### 8.13.4. Checking the Learn Status of a Field

You can check at any time whether a field has already been learned successfully or whether it requires relearning.

#### Task Prerequisites

The prerequisites for this task are:

- The program runs in Definition Mode.
- The panel on the left side of the window displays the *Classes* tab in the foreground.
- On the right side of the window, the tabs with class/field properties are visible.

#### Checking a Field's Learn Status

To check the learn status of a field:

- 1) In the *Classes* tab on the left side of the window, double click a class. The corresponding *Fields* tab is displayed.

- 2) Select a field.
- 3) Select the *Evaluation* tab on the right side of the window.
- 4) Check the *Brainware Extraction* tab. It displays the learn status.

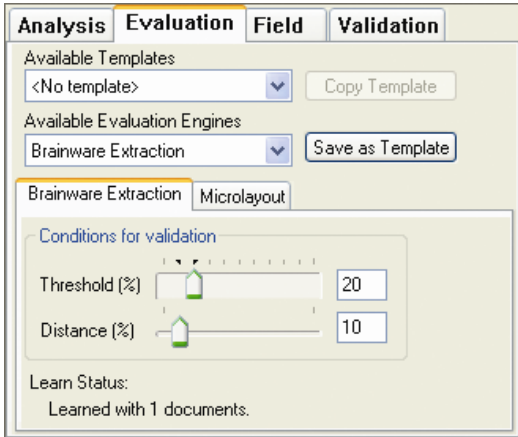


Figure 8-61: Learn status of a field

## 8.14 Testing the Extraction

When the fields have been set up and the analysis and evaluation methods have been defined and configured, the extraction can be tested.

### Task Prerequisites

The prerequisites for this task are:


- There is an active document set.
- The classification must work.
- If Brainware evaluation is required, the evaluation must have been trained.
- Your project settings must allow data extraction as runtime functionality.

There are several methods to test the extraction.

### In Definition Mode

- If the *Classes* tab is in the foreground, only classification will be tested.
- If the *Fields* tab is in the foreground, classification and extraction will be tested.

Use one of the following highlighting options to determine candidates, weights and validity:

Button	Description	Menu command
	<p>Highlights all candidates for the currently selected field in maroon.</p> <p>A candidate is a possible value for a field identified in the processing step. To view the weight of the candidate, point to it with the mouse. The weight is displayed as a tooltip.</p>	View - Highlight Candidates




Button	Description	Menu command
	Highlights all fields in the document. If the field is valid it is highlighted in green, otherwise it is highlighted in red. To view the name of the field, point to it with the mouse. The name is displayed as a tooltip.	View - Highlight Fields

Table 8-12: Highlighting options for extraction analysis

Use one of the following buttons to analyze the documents:

The test results are displayed in the *Fields* tab on the left side of the window.




Button	Description
	Processes the current document. The button's drop-down menu enables/disables the debug mode.
	Processes the next document.
	Processes all documents in the current set starting with the current one.

Table 8-13: Buttons for document analysis available in Definition Mode

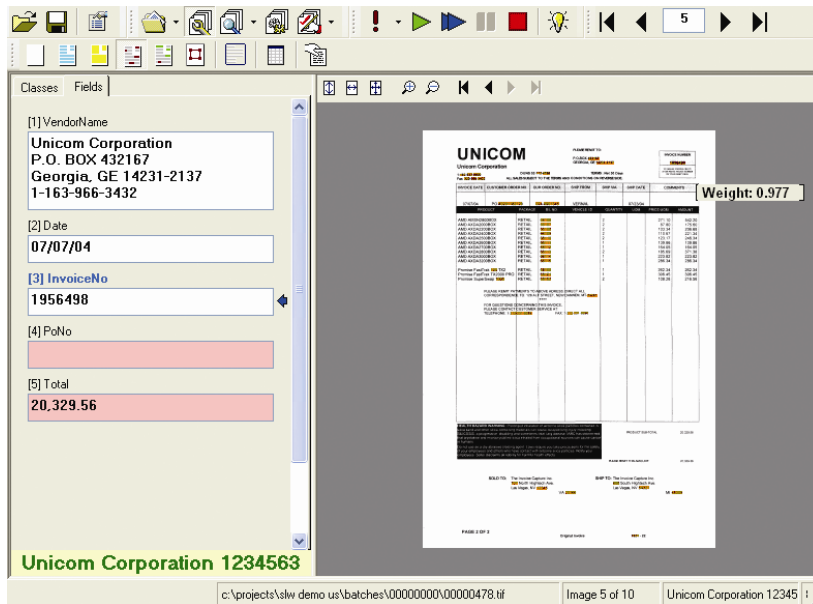


Figure 8-62: Data extraction test in definition mode

Possible results on the field level are:

- The field is empty. This happens if
  - no candidate was found or
  - no candidate had the required weight.
- The field is invalid. This is indicated by the red background color and happens if

- the best candidate had the required weight, but not the required distance or
- the best candidate did not meet standard or script-based validation requirements.
- The field is valid. This is indicated by the white background color.

**Not In Train Mode**

Data extraction is not carried out in Train Mode; the candidate selection is always done by the user. Therefore you cannot test the extraction in Train Mode.

**In Runtime Mode**

Use one of the following buttons to analyze the documents:




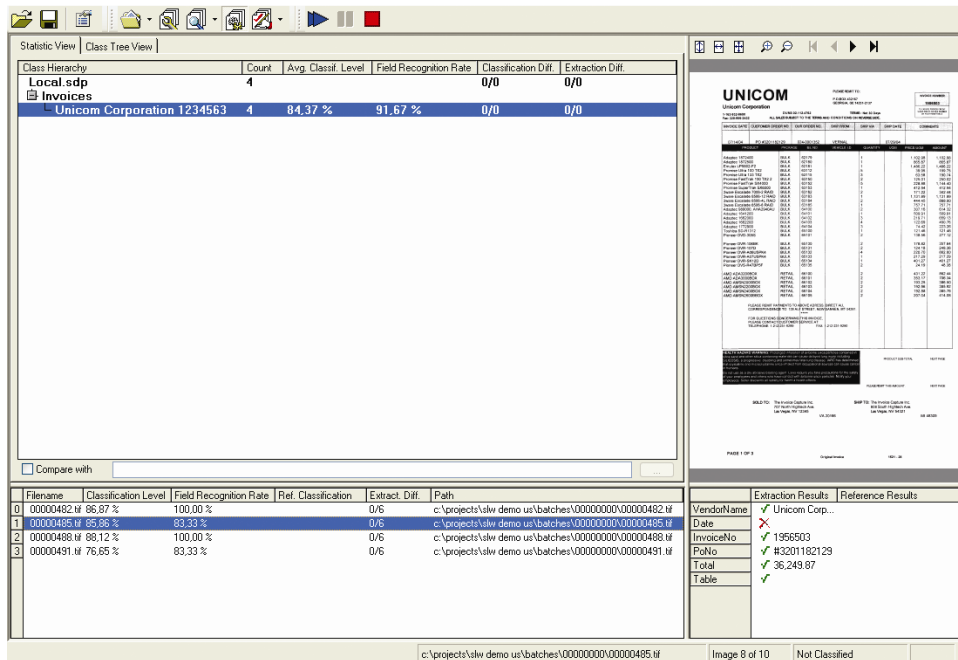
Button	Description
	Processes documents in the current set starting with the current one.
	Pauses the processing.
	Stops the processing of documents. Clears any results from previous runs.

Table 8-14: Buttons for document analysis available in Runtime Mode



The screenshot shows the 'Statistic View' tab with a table of document statistics:

Class Hierarchy	Count	Avg. Classif. Level	Field Recognition Rate	Classification Diff.	Extraction Diff.
Local.sdp	4			0/0	0/0
Invoices					
Unicom Corporation 1234563	4	84.37 %	91.67 %	0/0	0/0

Below this is a detailed table of extracted data for a specific document:

Filename	Classification Level	Field Recognition Rate	Ref. Classification	Extract. Diff.	Path
0 00000482.tif	86.87 %	100.00 %	0/6		c:\projects\slvw demo us\batches\00000000\00000482.tif
1 00000485.tif	85.85 %	83.33 %	0/6		c:\projects\slvw demo us\batches\00000000\00000485.tif
2 00000488.tif	88.12 %	100.00 %	0/6		c:\projects\slvw demo us\batches\00000000\00000488.tif
3 00000491.tif	76.85 %	83.33 %	0/6		c:\projects\slvw demo us\batches\00000000\00000491.tif

On the right, a preview of a 'UNICOM' invoice is shown. Below the preview is a table of 'Extraction Results' and 'Reference Results':

Field	Extraction Results	Reference Results
VendorName	✓ Unicom Corp...	
Date	✗	
InvoiceNo	✓ 1956503	
PolNo	✓ #3201182129	
Total	✓ 36,249.87	
T table	✓	

Figure 8-63: Runtime mode extraction results

The extraction-related columns in the *Statistic View* tab have the following meaning:

Column	Description
Avg. Classif. Level	Actual number of classifications vs. maximum number of classifications for the selected class in percent.
Classification Diff.	Class-level comparison of actual extraction results with results from a reference file. The first number represents the number of differences; the second one represents the total number of extractions. If there is a difference between test results and reference results, this is indicated by the red color.

Table 8-15: Class-Related Extraction Results

The extraction-related columns in the panel to the lower left have the following meaning:

Column	Description
Field Recognition Rate	Actual number of extractions vs. maximum number of extractions for the selected document in percent.
Extraction Diff.	Document-level comparison of actual extraction results with results from a reference file. The first number represents the number of differences; the second one represents the total number of extractions. If there is a difference between test results and reference results, this is indicated by the red color.

Table 8-16: Document-Related Extraction Results

The panel to the lower right displays the field values determined for the current document and compares them with the results for the reference documents.

- Valid results have a check mark.
- Invalid results have a cross.
- Differences between the test result and the reference result are indicated by the red color of the print.

### In Verifier Test Mode

If you use table analysis and want to check the extraction results, do this in Verifier Test Mode. The test options described above only tell you whether a table was found at all, and whether the extracted result was valid. Which data was actually written to the table cells is only visible in Verifier Test Mode.

## 8.15 Optimizing the Extraction

Most extraction problems are due to either:

- Incomplete configuration
- Insufficient quality of the Learnset
- Insufficient quality of the OCR

### 8.15.1. Resolving Problems with Incomplete Configuration

Incomplete configuration will most likely result in empty fields for all documents in a test set. The problem may be restricted to certain fields.

- For a given field, use candidate highlighting to check whether your analysis settings deliver enough candidates.
- For a given field, check the candidates' weights.
  - If they are all 0, check whether the evaluation method has been specified at all. (See section [SELECTING THE ANALYSIS METHOD](#)).
  - Check whether the evaluation has been learned (See section [LEARNING THE EXTRACTION](#)).
  - If you used field inheritance, there might be a problem with parent and child configuration. Check both configurations.

### 8.15.2. Resolving Problems with the Learnset

Problems with the Learnset will most likely result in low average extraction levels with empty and invalid fields. For some documents in the test set, the extraction will work, for others not. The problem will most likely affect all fields that use the Learnset.

- Check the candidates' weights and the distances. If your weights and distances are too low, you need more samples.
- Add candidates which were previously not classified correctly due to the initially insufficient Learnset.
- Use candidates from the same document with an almost identical weight. These candidates are suitable to differentiate.
- Avoid candidates with a high weight. These candidates do not improve the Learnset, because they would already be selected correctly.

### 8.15.3. Resolving Problems with the OCR

Problems occurring in the OCR will make the extracted data unusable. If you use a standard validation (No Rejects), you will obtain low average extraction levels with many invalid fields.

- The problem may be caused by the quality of your images. Make sure that your document input is properly prepared. For example, check scanner settings or fax export settings.
- The default OCR settings may not be suitable to process your document input. If candidates are found with many unrecognized characters, change the OCR settings at the zone level or at the field level. If the OCR is so bad that you cannot find candidates, you may have to change the OCR settings at the project level. For instructions on how to change OCR settings, please refer to section [ADVANCED RECOGNITION SETTINGS](#).

## 8.16 Applying Extraction Retaining Previously Available Extraction Results

### 8.16.1.1. Description

The feature can be activated on a project level and, as soon as activated, affects all applications that apply extraction processing in WebCenter Forms Recognition. It can be

turned by checking “When processing document always retains previous extraction results” option in “Processing” tab of Project Settings dialog of the Designer application:

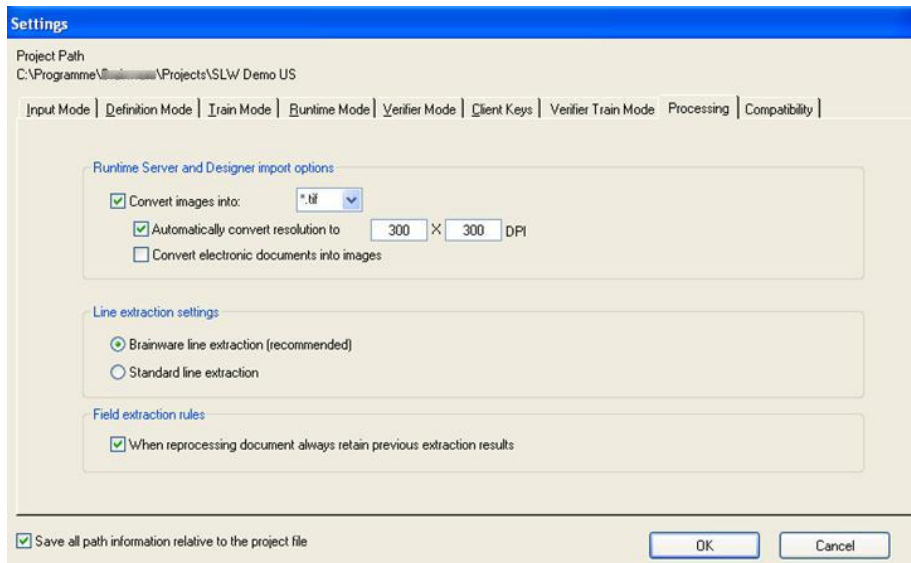


Figure 8-64: Processing tab – Field extraction rules

When activated, the system always combines new extraction results with those previously available in the working document. For example, if a WorkDoc contained fields A, B, and C and then extraction is applied to extract fields C, D, and E, the system will attach missing fields A and B to the updated WorkDoc, so that it finally contains C, D, E, A, and B. In this connection, a smart rollback is going to be applied for the content of field C, in case the new extraction result appeared to be empty. The roll-back, as well as attaching the old fields is going to be applied retaining all the secondary properties of the field objects (like attached candidate objects or enclosed tables) and not only the extraction result itself.

### 8.16.1.2. Usage & Scripting

The “Retain previous extraction results” feature can be activated if double (or triple) document processing is required. This can be configured either via two separate projects or via secondary extraction initiated directly from the first project’s custom script.

#### Example:

Extraction for header fields “A”, “B”, and “C” is supposed to be applied with one of the sub-classes of a parent class “Header Fields” that defines the extraction scheme for all three above mentioned fields. The secondary extraction is applied for another table field “T” and it has to be separated, e.g., applied via a totally different class “Table Fields” that includes definition and extraction scheme for the field “T” only.

#### 1<sup>st</sup> approach – double project processing:

Project “P1” contains definition of classes’ hierarchy with root “Header Fields” class. “Retain previous extraction results” feature is turned off. A Runtime Server instance applies an extraction workflow step with “P1” redirecting the documents to state 551 in both failure and full success cases. Another Runtime Server instance applies extraction with the second project “P2” that contains the definition of “Table Fields” class. “Retain previous extraction results” feature is turned ON for “P2”. This Runtime Server instance has 551 as input

extraction state and “normal” 552 / 700 output workflow states for extraction failure / success respectively. The resulting WorkDocs contain all desired fields “A”, “B”, “C”, and “T”.

## **2<sup>nd</sup> approach – double processing within one project via custom script:**

Same as in the first approach but there is just one project “P1” that contains both classes’ hierarchies: the one with “Header Fields”, and the one with “Table Fields”. By default it always applies extraction via the first hierarchy but, if required, can also launch the second level processing via a script like the one below:

```
Private Sub Document_PostExtract (pWorkdoc As SCBCdrFROULib.SCBCdrWorkdoc)
    On Error GoTo ERROR_HANDLER
    Project.AllClasses.ItemByName("Table Fields").Extract pWorkdoc
ERROR_HANDLER:
    Project.LogScriptMessageEx CDRTypeError, CDRSeveritySystemMonitoring, "Second level extraction has failed."
End Sub
```

## 9 Setting Up Supervised Learning

The Supervised Learning Workflow (SLW) is used to automatically create and learn new document classes. This automatic process creates the new derived Document Classes for a generic – base – document class (Level 0.) This Document Class is not inherited from any other document class.

*The derived Document Classes (Level 1) cannot have further derived Document Classes.*

In addition, the SLW has to be enabled in both Designer and Verifier.

Remember that you must learn any document you add to the local Learnset before it is released. Otherwise, the subclass will not be created in the project. In addition, the local project must be saved in order for the sub-class to be created in the global project.

### 9.1 Training the Base Class

A generic or base Document Class – which is also called a base class or base DocClass – is a generic category into which all the documents you are processing will fit. An example would be “Invoices”.

The classification for this generic class is not trained automatically and therefore must be pre-configured manually. To do this:

- 1) Configure the classification step by either assigning ASSA, Brainware, or any other available classification engine, or using this generic DocClass as default classification result.
  - To select a classification engine, click on the generic Document Class name in Definition Mode and select *Show Properties* from the *Edit* menu. On the Classification Editor on the right side of the screen, select a classification engine.
  - To define a document class as default classification result, click on the project name in Definition Mode and select *Show Properties* from the *Edit* menu. On the Classification Editor on the right side of the screen, notice the dropdown box for *Default Classification Result*. Select your generic document class for this setting. The Supervised Learning Workflow can only be applied for base classes and their direct sub-classes. Only administrator-selected base classes can become SLW nodes, where SLW node is a class node that defines how its sub-nodes will be automatically created, learned, etc. The simplest way to define a project’s SLW node classification is through its default classification result. The method works if there is only one SLW node.
  - Establishing multiple classification results can be done via script. In Designer, establish a default classification result. Then use the "Document\_PostExtract" event in the scripting module to set "pWorkdoc.DocClassName" to a different class. This technique enables you to keep the generic extraction pointed toward the default class, while moving the validation script to a different class. This can be configured and tuned manually via the various classification engines.
  - Insert at least one header field (usually called a text field) for which the Associative Search Engine is selected as the analysis engine.
  - Set the Classname Format on the Analysis Editor. The header field is the one that will be used to find the document class name used to create a new derived

document class (if the class does not already exist), and then to automatically classify the document.

- 2) Additionally, the Classification field on document class-level must be defined. Select the *Document Class* tab on the right side of the screen, and notice the dropdown box for *Classification field*. Select your generic document class for this setting from the drop-down list.

## 9.2 Training Other Fields

- 1) For all other text fields that are defined for a generic document class besides the one for which the Associative Search Engine is selected, you must use the Format Analysis Engine for analysis and the Brainware Extraction Engine for the evaluation engine.
- 2) If you have table fields, you must use Brainware Table Extraction for those fields.

*In WebCenter Forms Recognition, field names cannot contain periods or other special characters.*

## 9.3 Creating Derived Document Classes

### 9.3.1. Options for Creating Derived Classes in Designer

#### 9.3.1.1. How the Derived Classes are created

You do not have to manually configure the derived Document Classes; they will be created automatically when the documents are added to the Learnset.

All you need to do in order for the derived classes to be created are select Verifier Train Mode from the Settings menu, check the extraction results, and add the documents to the Learnset.

When the documents are added to the Learnset, a new derived Document Class is automatically created. The new DocClass name corresponds to the settings established for the document-level settings of the field Classname Format on the Analysis Editor. Therefore, this must be defined in a way that ensures uniqueness of every document class name that could be created, that means that the ID, or any other unique field, should be part of the Classname Format.

Though not obligatory, it can be useful to train some additional fields for the generic class now. This would mean that there would already be extraction results for the new class, and the Verification form would therefore already be partially populated.

To train the generic Document Class now, switch to Train Mode. If there is only one generic Document Class, you can set it as the default classification result. When this option is set, all documents are classified to the generic Document Class and only the extraction results have to be checked.

#### 9.3.1.2. How WebCenter Forms Recognition does the Classification

For the new automatically learned classes the system uses fixed and pre-determined engines that cannot be adjusted. Namely, the SLW uses:

- **Brainware Layout Classification** to automatically train Level 1 classification.



- **The Associative Search Engine** to automatically detect a new class name, and, optionally, to automatically train Level 1 classification.
- **Brainware Extraction** to train automatic extraction of header fields for fields that have Brainware Extraction and Format Analysis engines assigned for the parent base document class.
- **Brainware Table Extraction** to train automatic extraction of header fields for fields that have Brainware Table Extraction engine assigned for the parent base document class.

Even though a new learned DocClass has pre-defined engines that cannot be adjusted, the classes used in Supervised Learning can have any other engines assigned. When new classes are created automatically, they initially have the configurations of the engines mentioned above. However, the configuration can be changed, adjusted, or tuned manually.

### 9.3.1.3. Adding Supervised Learning to Brainware Layout Classification

The Brainware Layout Classification engine is used to automatically train the derived (Level 1) document classes. However, the user can also choose that the results of the Associative Search engine will be taken into account for the classification. These settings are established on the Supervised Learning Editor.

The screenshot shows the 'Supervised Learning' tab with the following settings:

- Make supplier field invalid
- Do smart decision
- Assign current document class name to supplier field
- Assign supplier field content to document current class name
- Do nothing
- Allow interactive table extraction for plain Verifier user
- In Verifier apply local extraction only for invalid fields

Figure 9-1: Supervised Learning Tab

In Definition Mode:

- 1) Select your project.
- 2) Select *Show Properties* from the *Edit* menu.
- 3) Click on the *Supervised Learning* tab.
- 4) On this tab, you can choose from *Make supplier field invalid* and/or:
  - *Do smart decision*: The system will decide which one is the right DocClass based on an algorithm that compare the results of the associative search and the Brainware Layout Classification.
  - *Assign current document class name to supplier field*: The system decides which document class will be used to assign the document to the class name in the field you created, based on the result of the Brainware Layout Classification.

- *Assign supplier field content to current document class name:* The system will use the result of the Associative Search Engine if the DocClass is available; otherwise the result of the Brainware Layout Classification will be used.)
- *Do Nothing:* The system uses only the result of the Brainware Layout Classification and not the result of the Associative Search Engine.
- *Allow interactive table extraction for Verifier user:* Activates the *Correct Tables* button in Verifier.
- *In Verifier apply local extraction only for invalid fields:* Local classification and extraction will be carried out only for invalid fields (either manually via *Analyze document* button or automatically via settings).

### 9.3.2. Options for Creating Derived Classes in Verifier

In Verifier, users can set an option on the Self-Learning tab for the properties that a document will be automatically added to the Learnset when a certain percent of the fields are invalid. Users can also choose to be prompted whenever a document is added to the Learnset.

## 10 Advanced Recognition Settings

In WebCenter Forms Recognition, you can use the following recognition techniques:

- OCR (Optical Character Recognition)
- Barcode Recognition
- OMR (Optical Mark Recognition)

OCR is applied to provide text input for the classification step. In the extraction step, either of the recognition techniques can be used.

Depending on the recognition method, WebCenter Forms Recognition provides several recognition engines you can select from:

- For OCR:
  - FineReader 10
  - FineReader 8.1
  - Recognita OCR Engine
    - Kadmos5 OCR/ICR Engine
- For OMR
  - Cairo OMR Engine
- For Barcode Recognition
  - Recognita Barcode Engine
  - Cleqs Barcode Engine
  - QualitySoft Barcode

Recognition is performed on demand during the classification and extraction steps. On demand means recognition can be performed as requested, any number of times. The result is stored in the Workdoc.

Each engine has its default parameter set that should work well in most cases. However, to enable you to select the optimum parameters for your problem if this should be required, you can customize the recognition settings independently on several levels for each processing step involves recognition.

### 10.1 Scope of Recognition Settings

#### 10.1.1. Project-Level Settings

Project-level settings affect:

- The classification step
- Extraction steps using format analysis
- Extraction steps using address analysis

- Extraction steps using zone analysis, provided that no different settings have been specified at the zone level.

Project-level settings affect every document processed with this project. They influence both quality and performance. Therefore, be very careful when changing them.

However, you should edit them if your classification does not work properly due to OCR problems, or if you have OCR-related problems getting candidates from format analysis or address analysis. You also should edit at the project-level to change the default engines for each recognition type.

## Task Prerequisites

The prerequisites for this task are:

- The program is in Definition Mode.
- In the *Classes* tab on the left side of the window, the entry representing your project is selected.
- On the right side of the window, the tabs with class/field properties are visible.
- The *Project* tab is displayed in the foreground.

The screenshot shows the 'Project' tab in the 'Advanced Recognition Settings' dialog. The dialog has four tabs: 'Project', 'Classification', 'Advanced', and 'Supervis'. The 'Project' tab is active. It contains several sections:

- Classification Interpretation:** A dropdown menu set to 'Maximum'.
- Standard Classification:** Two sliders. 'Threshold (%)' is set to 70 and 'Distance (%)' is set to 20.
- Parent Classification:** Two sliders. 'Threshold (%)' is set to 50 and 'Distance (%)' is set to 10.
- Document recognition:** A button labeled 'OCR settings ...'.
- Word segmentation characters:** A text box containing '()/?!|'.
- Smart Word Separation:** An unchecked checkbox.
- Storage for Table Training Data:** Two radio buttons, 'Project File' (selected) and 'Learnset'.

Figure 10-1: Project tab

## Available Options

The following options are available:

- To change preprocessing and recognition setting, click *OCR settings*. This displays the General OCR Settings Properties.



Figure 10-2: Project-Level Recognition Settings

- For detailed instructions on customizing the preprocessing options refer to section [THE PREPROCESSING TAB](#).
- For detailed instructions on customizing the recognition options, refer to section [THE RECOGNITION TAB](#) and section [THE FINEREADER OCR ENGINE](#).
- To change the rules for word segmentation, edit the enumeration of characters in the Word segmentation characters text box.

*Be extremely careful with this. You should have a couple of sample documents on hand and use the highlighting options to immediately check the consequences of your changes. Keep in mind that classification and extraction methods that rely on learning use a dictionary created from the words in your documents. If learning has already taken place, it needs to be repeated.*

### 10.1.2. Page-Level Settings

The project-level recognition settings are actually taken from and written to a special reading zone, the *All* zone: This zone represents the entire page and is normally not visible.

To display the *All* zone:

- 1) Switch to Definition Mode.
- 2) In the *Classes* tab, select the entry representing your project.
- 3) From the *View* menu, select *Show Page*. In the viewer, the *All* zone is displayed as a red bordered rectangle.

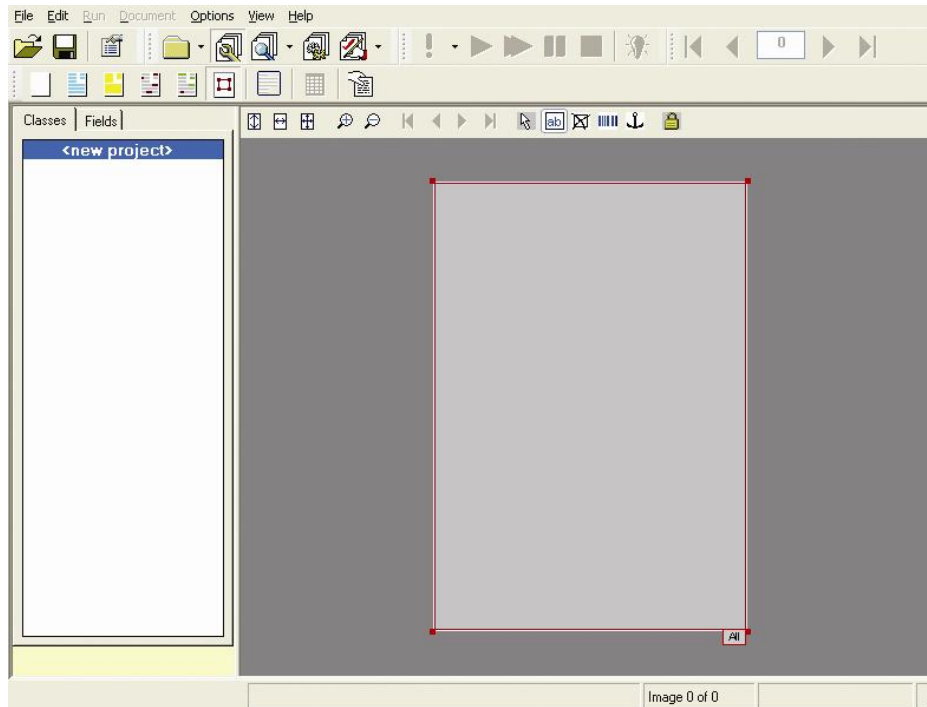


Figure 10-3: The All zone that represents the entire page

- 4) Select the *All* zone by clicking on it. It is now grey shadowed.
- 5) Right click the *All* zone and select *Properties* from the shortcut menu. This displays the Zone Settings Properties for the *All* zone.

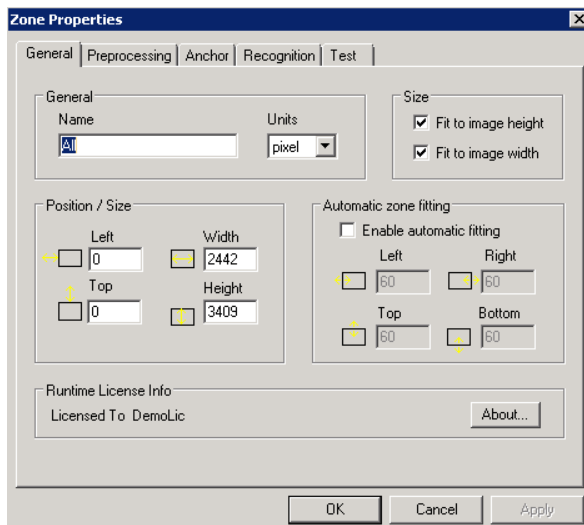


Figure 10-4: Recognition properties of the All zone

You should not edit these properties, since some of the displayed options do not make sense when applied to the entire document set. Edit at the project-level. (Section [PROJECT-LEVEL SETTINGS](#)) Synchronization between project-level settings and page-level settings is done automatically.

### 10.1.3. Zone-Level Settings

Zone-level settings affect extraction steps using zone analysis. Locally, they override project-level settings. You can use a different set of parameters for each reading zone.

You should edit zone-level settings if you are experiencing recognition-related problems to extract the correct data.

At the zone level, you can normally use all kinds of preprocessing and recognition options that improve your results. Due to the small reading area, the loss in performance is comparably small.

#### Task Prerequisites

The prerequisites for this task are:

- The program is in Definition Mode.
- The panel on the left side of the window displays the *Fields* tab in the foreground.
- A field is selected for which zone analysis has been defined.
- The viewer displays a document from the respective class.
- Obviously, there must be zones on the document to edit.

#### Editing Zone-Level Settings

To edit zone-level settings:

- 1) Select the reading zone.
- 2) Right click the zone and select *Properties* from the shortcut menu. This displays the Zone Settings Properties.

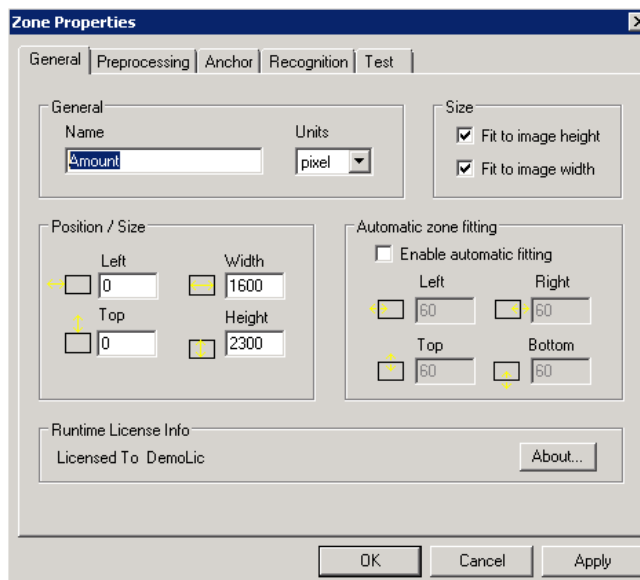


Figure 10-5: Zone level Settings

- The tabs are described in detail in section 10.2 Engine-Independent Settings.
- For detailed instructions on customizing the general options, refer to Section [THE GENERAL TAB](#).

- For detailed instructions on customizing the preprocessing options, see section [THE PREPROCESSING TAB](#).
- For detailed instructions on using anchors, see sections [FIXING THE COORDINATE SYSTEM FOR THE READING ZONES](#) and [THE PREPROCESSING TAB](#).
- For detailed instructions on customizing the recognition options, see sections [THE RECOGNITION TAB](#) and [THE FINEREADER OCR ENGINE](#).
- For detailed instructions on testing your settings, see section [THE TEST TAB](#).

#### 10.1.4. Field-Level Settings

Field-level settings affect extraction steps using format analysis or address analysis. It can be used to read the field again after the best candidate has been selected. By this time, the format and the geometry of the field is well known. Therefore, the OCR quality can be improved dramatically by using a different OCR engine and/or by imposing certain restrictions via classifiers.

#### Task Prerequisites

The prerequisites for this task are:

- The program is in Definition Mode.
- The panel on the left side of the window displays the *Fields* tab in the foreground.
- The respective field is selected.
- On the right side of the window, the tabs with class/field properties are visible.
- The *Fields* tab is displayed in the foreground.

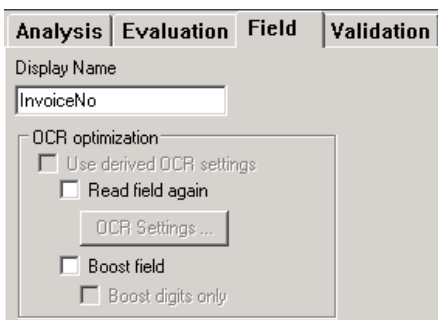


Figure 10-6: Field tab

#### Editing Field-Level Recognition Settings

Following field-level recognition settings are available:

- If the current field is a derived field, you can use inherited OCR settings of the parent class. Inheritance is the default behavior. To override the parent class settings, clear *Use derived OCR settings*. This enables the remaining OCR parameters.
- Check *Read field again* to activate OCR optimization.
- Click *OCR settings*. The Field OCR Settings Properties are displayed:



- Click *Boost field* to activate the Digit booster. The Digit booster provides optimized OCR results for defective digits. For details see section [10.9 OCR FIELD LEVEL DIGIT BOOSTER](#).
- Check the results of field boosting by looking on the highlighting of candidates. For details on the extended candidate highlighting mode see section [12.4.5 FIELD-LEVEL DIGIT BOOSTER CANDIDATE EVALUATION](#).

For further instructions on customizing the recognition options, see sections [THE RECOGNITION TAB](#), [THE FINEREADER OCR ENGINE](#), [THE RECOGNITA OCR ENGINE](#) and [KADMOS 5 ENGINE](#).

## 10.2 Engine-Independent Settings

### 10.2.1. General Tab

To edit size and position of the reading zone, select the *General* tab.

Figure 10-7: OCR General tab

The following options are available:

- Under *General*:
  - *Name*: Use this option to specify the name of the reading zone. By default, *All* is the reading zone for the entire page.
  - *Units*: Use this option to specify the units used for geometric parameters (pixel, inch, or mm).
- Under *Size*:
  - *Fit to image height*: Use this option to adjust the height of the zone to the height of the page.
  - *Fit to image width*: Use this option to adjust the width of the zone to the width of the page.
- Under *Position / Size*:

- *Left*: Use this option to specify the distance of the reading zone to the left edge of the page.
- *Top*: Use this option to specify the distance of the reading zone to the top edge of the page.
- *Width*: Use this option to specify the width of the reading zone.
- *Height*: Use this option to specify the height of the reading zone.

### 10.2.2. Preprocessing Tab

To edit preprocessing options, select the *Preprocessing* tab.

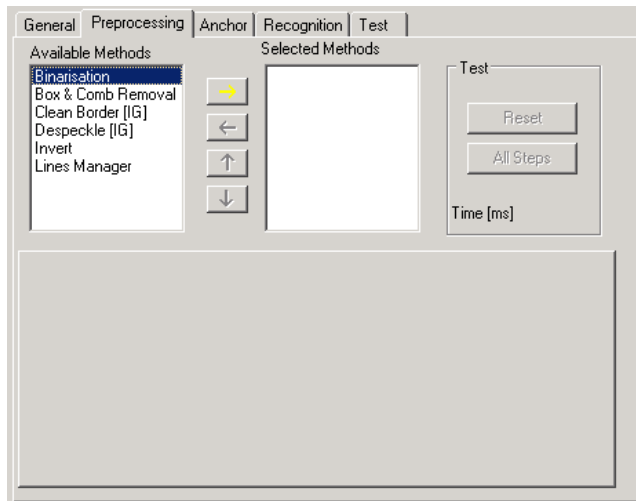


Figure 10-8: OCR Preprocessing tab

#### Enabling Preprocessing Methods


To enable preprocessing methods:

- 1) Under *Available Methods*, select one or multiple entries.
  - Select *Despeckle* to remove speckles from your documents. Speckles are made up of groups of black pixels surrounded by white pixels or vice versa.
  - Select *Lines Manager* to remove horizontal or vertical lines from your document.
  - Select *Clean Border* to improve the quality of images that have a dark border as, for example, often generated in photo copies.

- 2) Click  to move the selected entries to *Selected Methods*.

#### Disabling Preprocessed Methods

To disable preprocessing methods:

- 1) Under *Selected Methods*, select one or multiple entries.
- 2) Click  to remove the selected entries from *Selected Methods*.

#### Ordering Preprocessing Methods

Preprocessing methods are applied in the same order as listed under *Selected Methods*.

To order preprocessing methods:

- 1) Under *Selected Methods*, select an entry.



- 2) Click on the following button to move up or click the following button to move down.

### Configuring the Line Manager

To configure the line manager:

- 1) Under *Selected Methods*, select *Line Manager*.
- 2) On the *General* tab, check *Remove hor. lines* to remove horizontal lines. Enter a minimum line length in mm in the *Min. hor....* text box.
- 3) On the *General* tab, check *Remove ver. lines* to remove vertical lines. Enter a minimum line length in mm in the *Min. ver....* text box.

### Testing the Effect of Preprocessing Methods

To test the effect of preprocessing methods:

- 1) Make sure the current document is visible and not hidden behind the dialog box.
- 2) Do one of the following:
  - To test a single method: Under *Selected Methods* select an entry and click *Single Step*.
  - To test all selected methods: Click *All Steps*. Check the preprocessing effects by looking at the document. The time required for preprocessing is displayed under *Test*.
- 3) For further tests, click *Reset* before you proceed.

*If performance is an issue, check the sum of recognition time and preprocessing time. Even if preprocessing takes a while, recognition may be faster after preprocessing.*

### 10.2.3. Anchors Tab

To assign anchors to the current reading zone, select the *Anchors* tab.

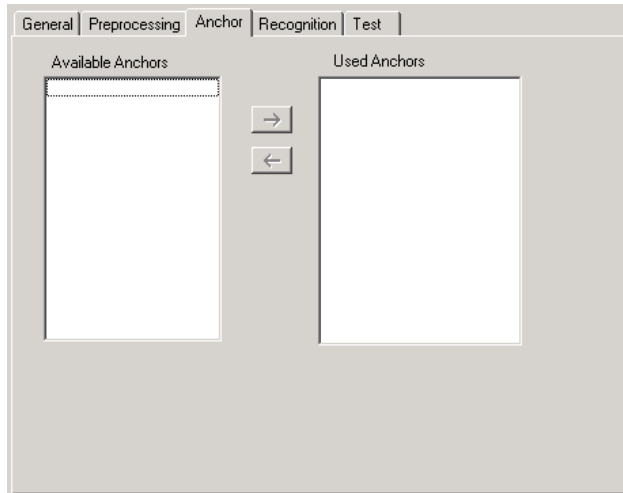




Figure 10-9: OCR Anchors tab

### Available Operations

The following operations are possible:

- To assign anchors, select them from the *Available Anchors* list box and click 
- To delete assignments, select anchors from the *Selected Anchors* list box and then click 

### 10.2.4. Recognition Tab

To edit recognition settings, select the *Recognition* tab.

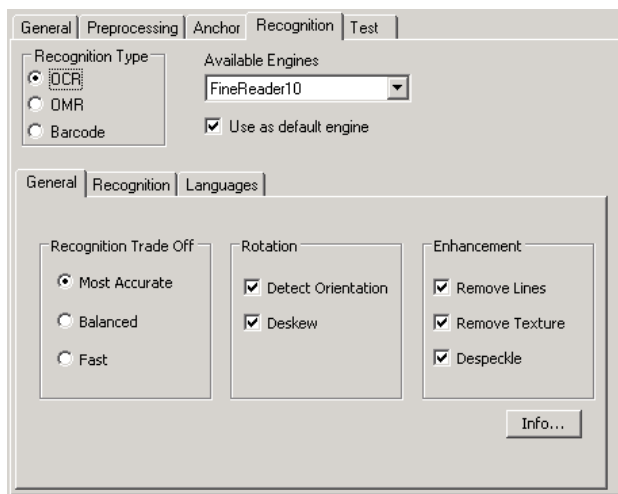


Figure 10-10: OCR Recognition tab

To select a recognition engine:

- 1) Under *Recognition Type*, select OCR, OMR, or Barcode.
- 2) Under *Available Engines*, select the recognition engine.
- 3) To use the selected engine as default engine for the current recognition type, check *Use as default engine*.

For more information about customizing the selected engine:

- FineReader 8.1 OCR Engine (Section [FINEREADER 8.1](#))
- FineReader 10 Engine (Section [FINEREADER 10](#))
- Recognita OCR Engine (Section [RECOGNITA OCR ENGINE](#))
- Kadmos5 OCR Engine (Section [KADMOS 5 ENGINE](#))
- Recognita Barcode Engine (Section [THE RECOGNITA BARCODE ENGINE](#))
- Cleqs Barcode Engine (Section [THE CLEQS BARCODE ENGINE](#))
- Cairo OMR Engine (Section [THE CAIRO OMR ENGINE](#))

### 10.2.5. Test Tab

To test the recognition results with the current document, select the *Test* tab.

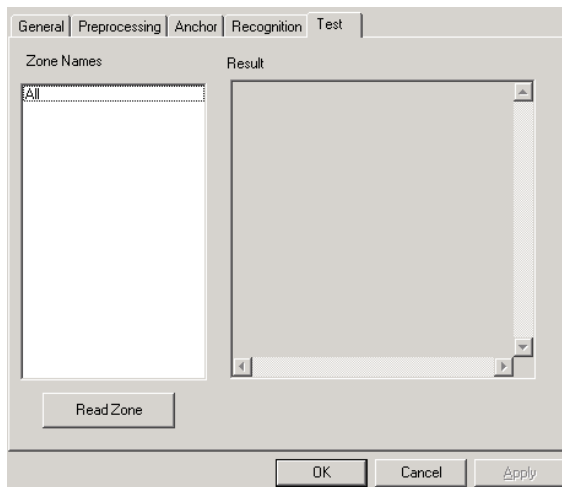


Figure 10-11: OCR Test tab

To perform the test, do one of the following:

- If no zone is selected: In the *Zone Names* list box, click the zone name.
- If there is a zone selected: Click *Read Zone*.

The recognition result is displayed in the *Reco Result* text box. Below, the execution time and -if applicable- the degree of blackness is displayed.

*Avoid recognition tests with entire pages.*

## 10.3 FineReader OCR Engine

The FineReader OCR Engine includes FineReader 8.1 and FineReader 10. They optimize the enterprise documents with poor or marginal printer quality, such as documents printed with dot matrix or chain printers.

*Note:* It is not recommended that two different FineReader Engine versions participate in one process. Potential process failure might occur.

### 10.3.1. FineReader 8.1 and 10

FineReader 8.1 is basically an update for the previously integrated FineReader engines.

One important difference, as related to the support of FR 8.x engine in WebCenter Forms Recognition, is about processing of one document with multiple zones, where the OCR is applied multiple times for small areas within one single page of an image file. For such a case, the FineReader engine increments its internal license counter only once per all zones' recognition attempts applied for the currently loaded document's page.

#### 10.3.1.1. The General Tab

To edit general settings, select the *General* tab:

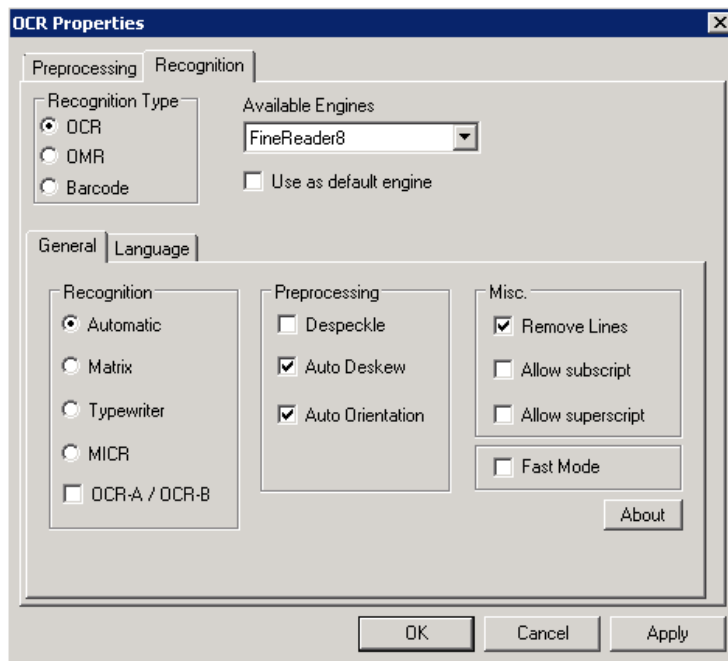


Figure 10-12: Editing settings for text, preprocessing and others FineReader 8 General Tab

- **Recognition:** The options are Automatic, Matrix, Typewriter and OCR-A / OCR-B:
  - *Automatic:* Select this option to automatically detect how the documents have been printed. As a rule, use this option unless your document input is very homogeneous and from a dot-matrix printer or a typewriter
  - *Matrix:* Select this option for material printed with dot-matrix printers.
  - *Typewriter:* Select this option for material created with typewriters.
  - *MICR:* A character recognition technology to facilitate the procedure of instance check.
  - *OCR-A / OCR-B:* Select this when the documents use OCR-A and OCR-B fonts. This is typically where high OCR character recognition rates are needed. The fonts are optimized for machine character recognition. For example, they are used on passports and other security documents.
- **Preprocessing:** The options are Despeckle, Auto Deskew, and Auto Orientation:

- *Despeckle*: Select this option to remove speckles from your documents. Speckles are made up of a group of black pixels surrounded by white pixels or vice versa.
- *Auto Deskew*: Select this to automatically fix the alignment of badly scanned pages.
- *Auto Orientation*: If checked, the engine will automatically detect the page orientation and rotate the image to the correct position if necessary.
- *Miscellaneous*
  - *Remove Lines*: Select to remove all internal lines, either horizontal or vertical.
  - *Allow Subscript*: Select this option to allow a character that is printed on a level lower than the rest of the characters on the line, for example, the “2” in the chemical formula “H2O”
  - *Allow Superscript*: Select this option to allow for letters, characters, or symbols that are written above, or above and to the right or left of, another character. For example, “Forms Recognition™”
  - *Fast Mode*. If checked recognition is quicker but less accurate.
- *About Button*: Information about the Runtime license.

### 10.3.1.2. The Languages Tab

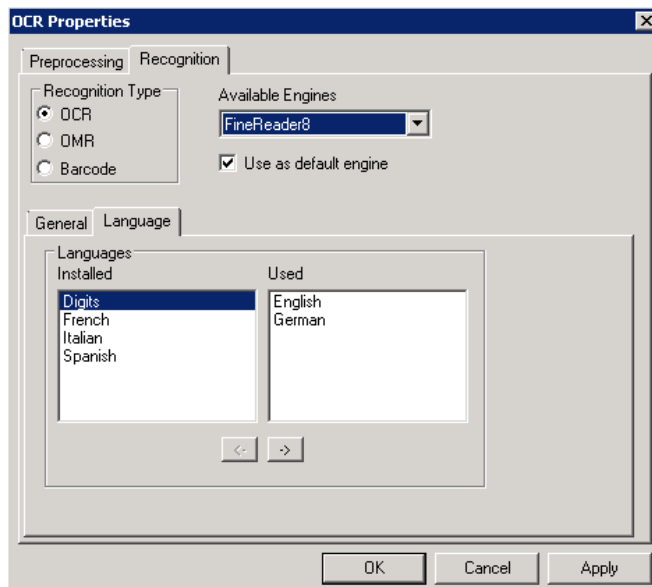


Figure 10-13: Setting languages on the Languages tab for FineReader 8.1.

**Languages:** The options are as follows:

- *Installed*: Set of languages installed on the system.
- *In Use*: Select your text language. Recognition will be processed with the corresponding language data files (language database). Select multiple languages only if you are processing multilingual documents.

## 10.3.2. FineReader 10

### 10.3.2.1. The General Tab

To edit general settings, select the *General* tab

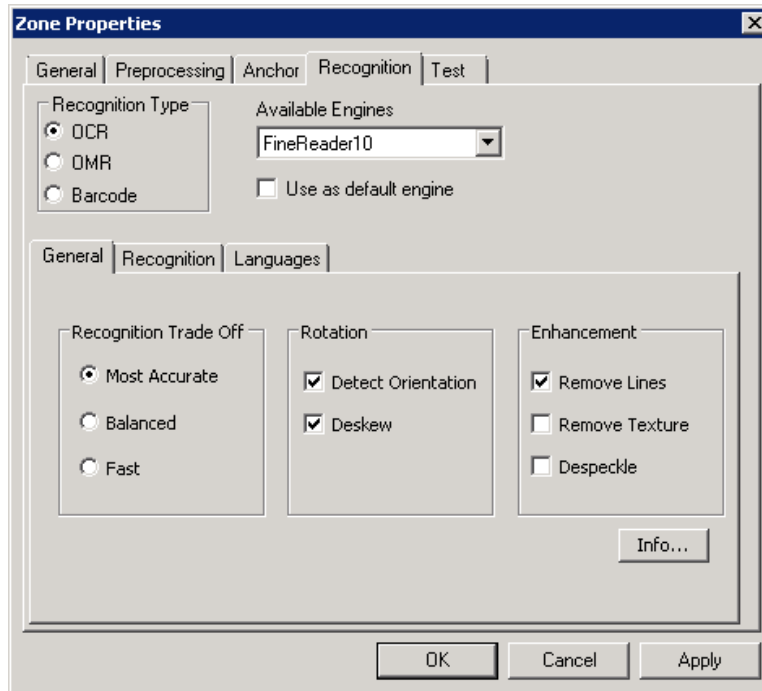


Figure 10-14: FineReader 10 General Tab

- *Recognition Trade Off* is a trade off between performance and OCR accuracy.
  - *Most Accurate*: If selected recognition will have less error rate but higher in processing time. This is the default mode.
  - *Balanced*: If selected recognition is in between *Most Accurate* and *Fast*.
  - *Fast*: If selected recognition is quicker but less accurate.
- *Rotation*
  - *Detect Orientation*: Select this to automatically fix the alignment of badly scanned pages
  - *Deskew*: Select this to automatically fix the alignment of badly scanned pages.
- *Enhancement*
  - *Remove Lines*: Select to remove all internal lines, either horizontal or vertical.
  - *Remove Texture*: Select to remove background noise from the image for recognition.
  - *Despeckle*: Select this option to remove speckles from your documents. Speckles are made up of a group of black pixels surrounded by white pixels or vice versa.



### 10.3.2.2. The Recognition Tab

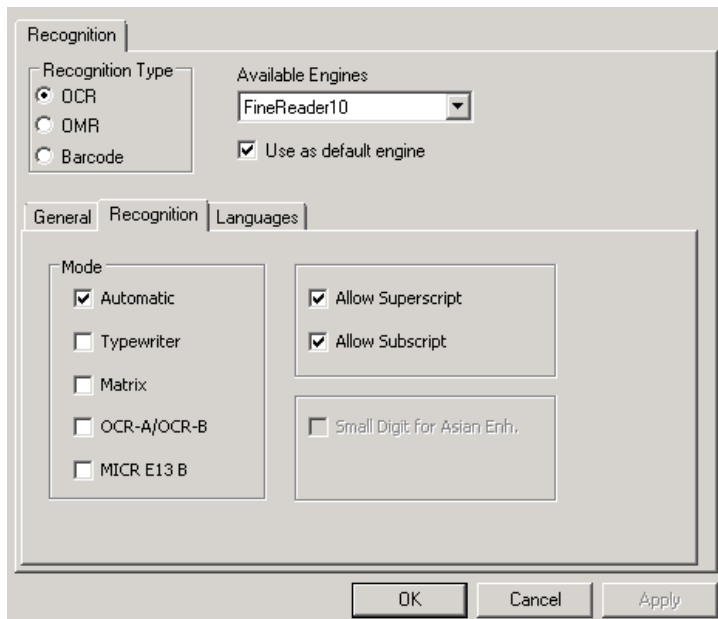


Figure 10-15: Recognition Tab of FineReader10

- **Recognition:** The options are Automatic, Matrix, Typewriter and OCR-A / OCR-B:
  - **Automatic:** Select this option to automatically detect how the documents have been printed. As a rule, use this option unless your document input is very homogeneous and from a dot-matrix printer or a typewriter
  - **Matrix:** Select this option for material printed with dot-matrix printers.
  - **Typewriter:** Select this option for material created with typewriters.
  - **OCR-A / OCR-B:** Select this when the documents use OCR-A and OCR-B fonts. This is typically where high OCR character recognition rates are needed. The fonts are optimized for machine character recognition. For example, they are used on passports and other security documents.
  - **MICR E13 B:** A character recognition technology to facilitate the procedure of instance check.
  - **Allow Superscript:** Select this option to allow for letters, characters, or symbols that are written above, or above and to the right or left of, another character. For example, "WebCenter Forms Recognition™"
  - **Allow Subscript:** Select this option to allow a character that is printed on a level lower than the rest of the characters on the line, for example, the "2" in the chemical formula "H<sub>2</sub>O"
  - **Small digit for Asian enhancement box:** The partial fix for single digit columns can be controlled by this checkbox. This is used with one of the CJKT languages in the language list. This box is only accessible if at least one of this languages is selected in the list, .e.g Chinese and English. To use the enhancement select the box (see section [4.8.3.3 SUPPORT OF NON-WESTERN LANGUAGES](#) for more details).

### 10.3.2.3. The Language Tab

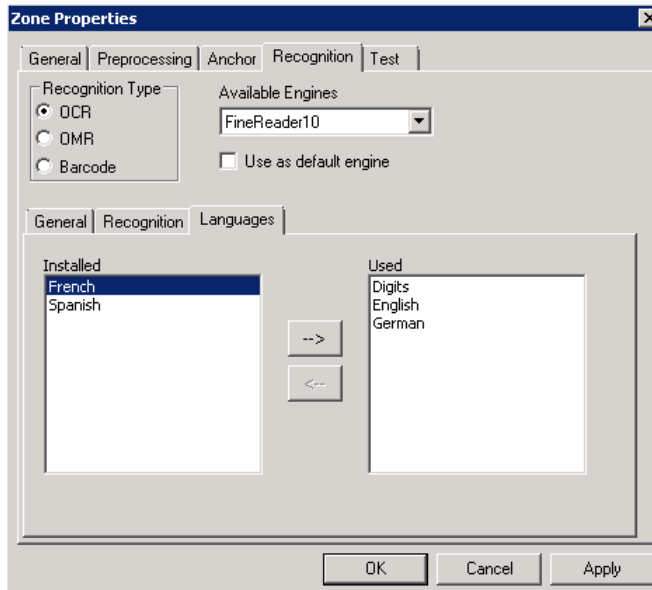


Figure 10-16: The Language Tab

#### Editing Language Settings

To edit language settings, select the *Language* tab.

- *Languages*: The options are as follows:
  - *Installed*: Set of languages installed on the system.
  - *Used*: Select your text language. Recognition will be processed with the corresponding language data files (language database). Select multiple languages only if you are processing multilingual documents.

## 10.4 Recognita OCR Engine

### 10.4.1. General Tab

To edit general settings, select the *General* tab.

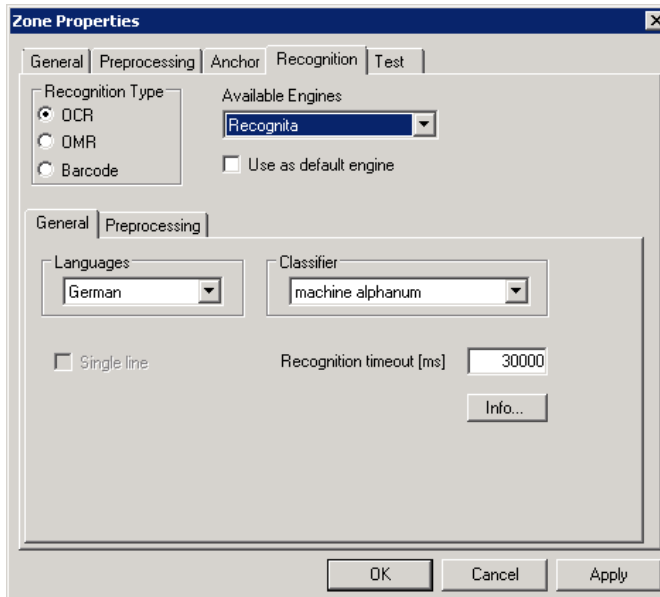


Figure 10-17: General tab of the Recognita OCR Engine

The following options are available: Language, Classifiers, and Recognition timeout.

- **Languages:** Select your text language. The recognition results will be compared with entries in the corresponding dictionary.
- **Classifiers:** Select one of the following options:
  - Machine alphanum: for machine-printed material containing alpha characters and numbers
  - 24-pin draft dot-matrix: for material printed with dot-matrix printers with 24 pins.
  - 9-pin dot-matrix: for material printed with dot-matrix printers with 9 pins.
  - hand, numeric: for hand-written numbers.
- **Recognition timeout [ms]:** Specify the time in milliseconds after which the recognition will be cancelled even if there is no result.

### 10.4.2. Preprocessing Tab

To edit preprocessing settings, select the *Preprocessing* tab.



Figure 10-18: Preprocessing tab of the Recognita OCR Engine

The following options are available: Automatic image inversion, Skew Correction, and Fax Image Enhancement.

- *Automatic Image Inversion*: Select this to automatically detect black characters on a white background and white characters on a black background.
- *Skew Correction*: Select this to automatically adjust pages that were scanned at an angle.
- *Fax Image Enhancement*: Select this to activate specialized preprocessing options for faxes.

## 10.5 Kadmos 5 Engine

The Kadmos 5 engine is optimized for recognition of handwritten document forms and material containing handwritten zones. It offers support and optimization for different language.

### 10.5.1. General Tab

To edit general settings, select the *General* tab.

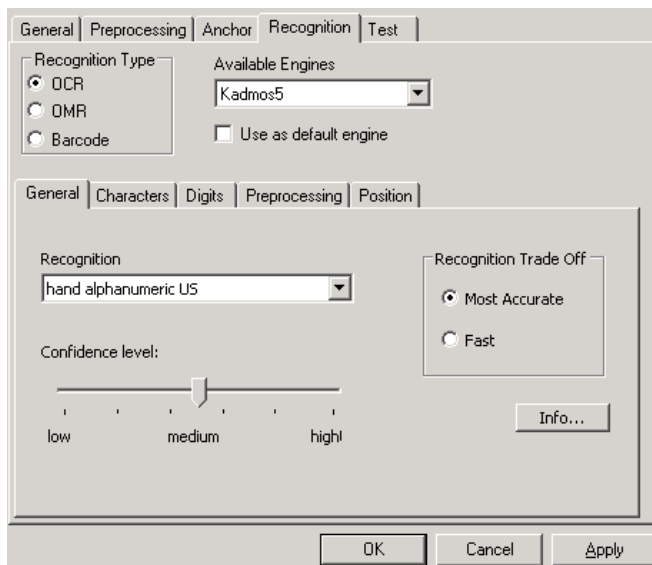


Figure 10-19: The General tab of the Kadmos5 Engine

Following options are available: Recognition classifier, Confidence Level and Recognition Trade Off.

- *Recognition*: Select the classifiers in the dropdown list. The selection determines the range of valid characters. These characters are displayed on the Characters tab and can be further restricted there:
  - **Hand alphanumeric German, Europe, UK, US**: for German, European, UK or US hand-written material containing alphanumeric characters
  - **CMC7, E13B, Farrington F7B Normfont**: for material written with CMC7, E13B (US checks), F7B (ID cards) fonts.
  - **OCRA, OCRB**: for material written with dedicated OCR fonts.
  - **Hand + machine print, numeric**: for material containing handwritten or machine-printed numbers and currency symbols, but no other alpha characters

- **Hand + machine print, Europe, US numeric:** for Europe/US documents
- **Machine, alphanum, German, UK, US:** for UK, US, and German machine-printed material containing alphanumeric characters.
- **Confidence Level:** Use the slider to specify the threshold for accepting characters. Characters recognized below the required confidence level are termed as rejects, characters recognized above the confidence level are termed as valid. High confidence level causes more reject, but there will be fewer errors in the accepted results. High confidence level also affects performances. Generally, it takes about 10 times longer to read with the highest confidence level than with the lowest.
- **Recognition Trade Off:** Select the type of optimization assigned to the ICR process. The *Most accurate* parameter provides higher accuracy with lower speed.

### 10.5.2. Characters Tab

Restrict the range of valid characters. The offered Character set depends on the classifier that was selected before.

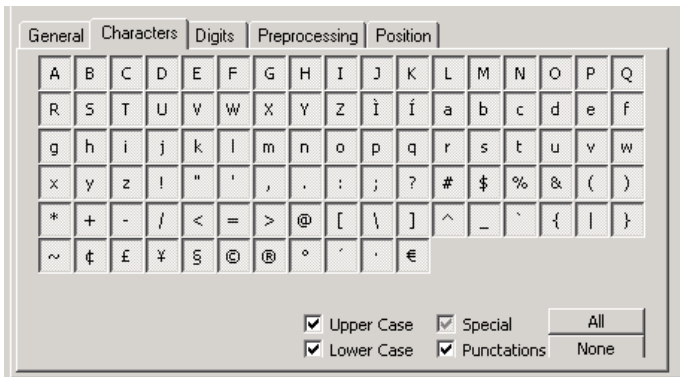


Figure 10-20: The Characters tab of the Kadmos5 Engine

- In the upper section, the *Valid Characters* tab displays a button for each possible character of the selected classifier (See section [GENERAL TAB](#)). A pressed button corresponds to a valid character.
- In the bottom section, there are check boxes that enable you to edit entire character groups in one step. If a check box is selected, the corresponding character group is valid.
- Use the *All* and *None* buttons to edit the entire character set in one step. By clicking *All*, all characters are marked as valid. By clicking *None*, all characters are marked as invalid.

### 10.5.3. Digits Tab

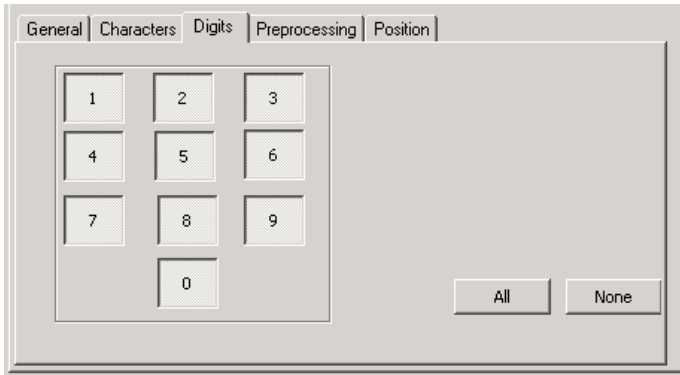


Figure 10-21: The Digits tab of the Kadmos5 Engine

Define the digits that are eligible to be found on the document. Use the *All/None* buttons to select/deselect all digits.

### 10.5.4. Preprocessing Tab

**Note:** The Kadmos Preprocessing features are implemented for special usage only. In most cases it is recommended to use the Brainware Preprocessing features as described in section 10.2.2 Preprocessing Tab.

To edit preprocessing settings of the Kadmos engine, select the *Preprocessing* tab.

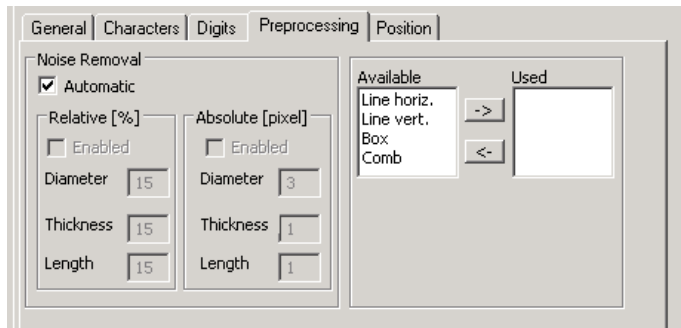


Figure 10-22: Preprocessing tab of the Kadmos5 Engine

Edit the following *Noise Removal* settings:

- *Automatic:* The *Automatic* setting for noise removal is based on an algorithm that will produce best results for many scenarios depending on the characters position, size and segmentation. For cheques use the *Automatic* setting. Only adjust the settings manually if specific values are needed for well known cases.
- *Relative [%], Absolute [pixel]:* Select one of these options to reduce background noise on your documents manually. You need to select the *Enabled* check box and to specify the maximum size of spots that are removed, either using the pixel scale (*Noise removal [pixel]*), or as a fraction of the line thickness of the found characters (*Noise removal [%]*). The size parameters are:
  - Diameter: maximum diameter of a spot
  - Thickness: maximum horizontal length of a spot
  - Length: maximum vertical length of a spot

Edit the following *Object Removal* settings:

- *Objects Removal*: Use this extension to remove lines or other geometrical shapes that could interfere with the OCR from the documents. You can remove:
  - Lines, horizontal or vertical
  - Boxes
  - Combs

For specific documents it can be recommended to use Brainware Lines Manager or Despeckling as Preprocessing first.

### 10.5.5. Position Tab

To specify settings with respect to word and line surroundings, select the *Position* tab.

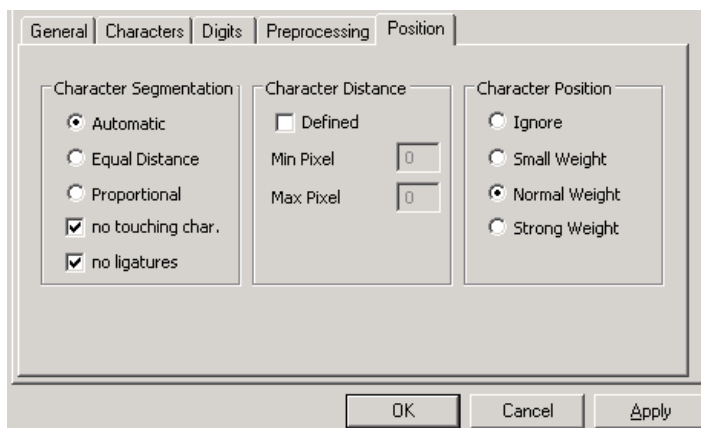


Figure 10-23: Context tab of the Kadmos5 Engine

Three options are available: Character Segmentation, Character Distance and Character Position:

- *Character Segmentation*: The *Automatic* setting is used by default.
  - If the characters in the zone are within equal or proportional distance, select *Equal Distance* or *Proportional*.
  - If the characters do not touch, select *no touching char.* With no ligatures in the zone select *no ligatures*. If ligatures are expected in the zone, deselect *no ligatures*.
- *Character Distance*: Speed up processing by specifying the horizontal character distance in quantity of image pixel. Keep in mind that the character distance is resolution dependent. To activate the option, check *Defined*. Specify the minimum and the maximum distance in image pixel using the following parameters:
  - Min. Pixel:
  - Max. Pixel:
- *Character Position*: Select one of the options to take the position of a character within a line into account. The scale goes from irrelevant (*Ignore*) up to very relevant (*Strong weight*).

## 10.6 Recognita Barcode Engine

The Recognita Barcode engine is the default engine for Barcode recognition.

### 10.6.1. General Tab

To edit general settings, select the *General* tab.

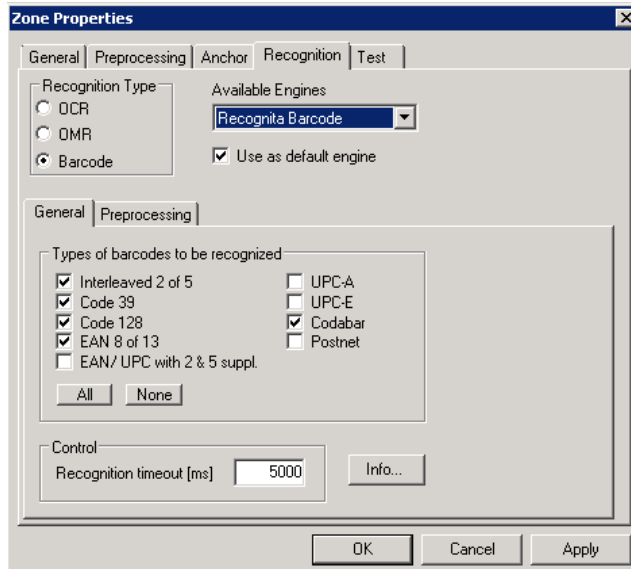


Figure 10-24: General tab of the Recognita barcode Engine

Two options are available: Types of Barcodes and Recognition Timeout.

- *Types of Barcodes...*
  - Use the check boxes to select barcode types that you expect to find. Do not check more options than required as this will slow down processing and affect accuracy.
  - Click on All to select all barcode types.
  - Click on None to clear all selections.
- *Recognition timeout [ms]:* Specify the time in milliseconds after which the recognition will be cancelled even if there is no result.

### 10.6.2. Preprocessing Tab

To edit preprocessing settings, select the *Preprocessing* tab.



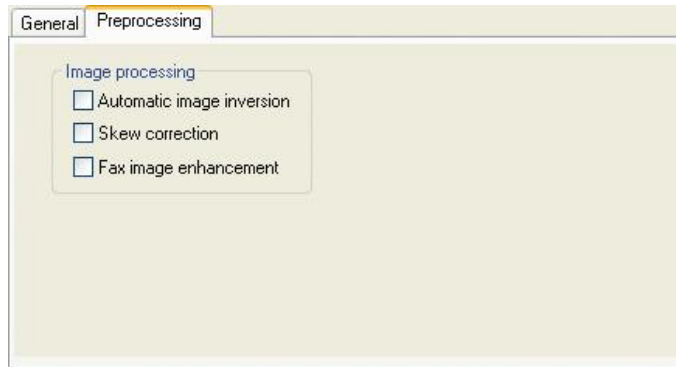


Figure 10-25: Preprocessing tab of the Recognita barcode Engine

Three options are available: Automatic Image Inversion, Skew Correction, and Fax Image Enhancement.

- *Automatic Image Inversion*: Select this to automatically detect black characters on a white background and white characters on a black background.
- *Skew Correction*: Select this to automatically adjust pages that were scanned at an angle.
- *Fax Image Enhancement*: Select this to activate specialized preprocessing options for faxes.

## 10.7 Cleq Barcode Engine

The Cleq Barcode Engine is a barcode engine compatible with WebCenter Forms Recognition.

### 10.7.1. General Tab

To edit general settings, select the *General* tab.

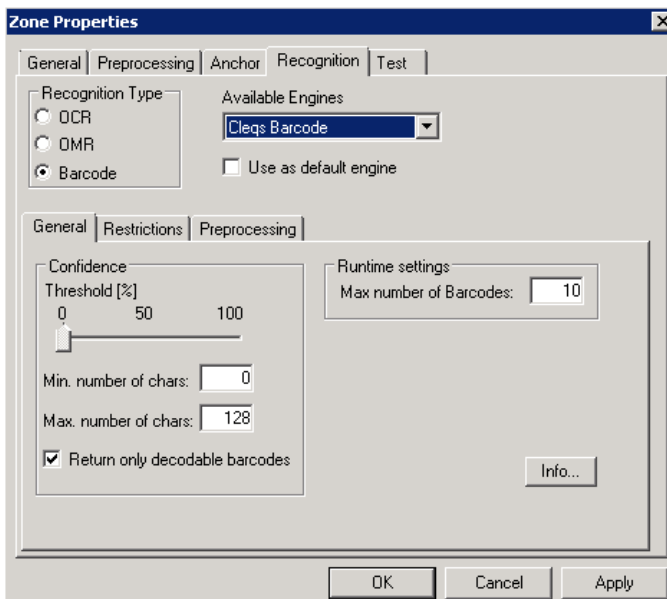


Figure 10-26 General tab of the Cleq barcode engine

The following options are available:

- *Confidence*: Currently not available
- *Min. / Max. number of chars*: Enter the minimum or maximum length of the barcode in characters.
- *Return only...:* Check this option to accept barcodes only if all characters are recognized, i.e. no rejects.
- *Max. number of Barcodes*: Enter the maximum number of barcodes to be recognized within the reading zone. The default is 10, which can usually be reduced to speed up processing.

## 10.7.2. Restrictions Tab

To edit barcode types, select the *Restrictions* tab.

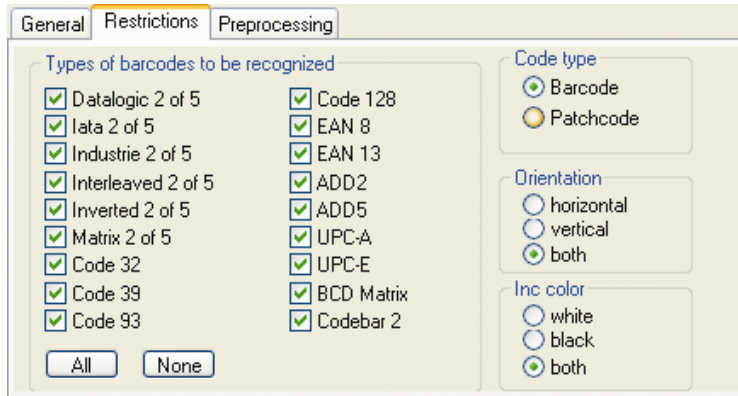


Figure 10-27: Restrictions tab of the CleqS barcode engine

The following options are available:

- **Code type:** Select whether you want to recognize barcodes or patchcodes. A patchcode is a pattern of horizontal black bars separated by spaces and typically placed near the leading edge of a paper document. Patchcodes can be used to separate documents.

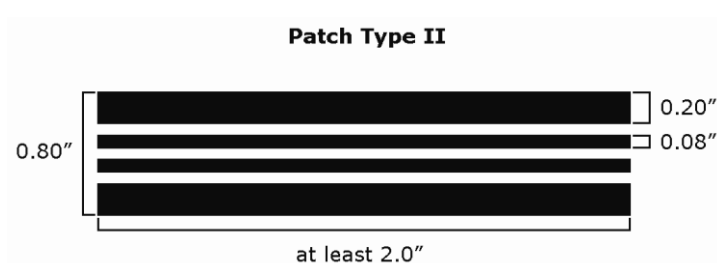


Figure 10-28: Sample patchcode

- **Types of barcodes...**
  - Use the check boxes to select barcode types that you expect to find. Do not check more options than required as this will slow down processing and affect accuracy.
  - Click on All to select all barcode types.
  - Click on None to clear all selections.
- **Orientation:**
  - Horizontal: Select this option only if all your barcodes or patchcodes are oriented horizontally.
  - Vertical: Select this option only if all your barcodes or patchcodes are oriented vertically.
  - Both: Select these options to process heterogeneous material.
- **Ink Color:**
  - White: Select this option to detect white barcodes on a black background.

- Black: Select this option to detect black barcodes on a white background.
- Both: Select these options to detect white barcodes on a black background and black barcodes on a white background.

### 10.7.3. Preprocessing Tab

To edit preprocessing settings, select the *Preprocessing* tab.

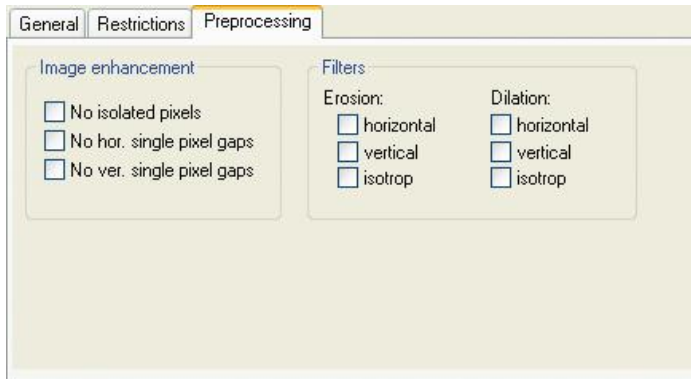


Figure 10-29: Preprocessing tab of the Cleq barcode engine

The following options are available:

- Image enhancement:
  - *No isolated pixels*: Select this option to automatically remove small speckles from the reading zone.
  - *No hor. single pixel gaps*: Select this option to automatically fill small gaps in horizontal lines. This option is recommend processing barcodes with a corresponding orientation.
  - *No ver. single pixel gaps*: Select this option to automatically fill small gaps in vertical lines. This option is recommend processing barcodes with a corresponding orientation.
- *Filters / Erosion*: The erosion filter thins down lines by setting any black pixel to white if it is located at a specified direction to at least one white pixel in the source image. To specify the direction, select from the following options:
  - horizontal
  - vertical
  - isotropic (i.e., all directions)
- *Filters / Dilation*: The dilation filter thickens lines by setting any white pixel to black if it is located at a specified direction to at least one black pixel in the source image. To specify the direction, select from the following options:
  - horizontal
  - vertical
  - isotropic (i.e., all directions)

## 10.8 Cairo OMR Engine

### 10.8.1. OMR Zones

Using the Cairo OMR Engine, you can determine the contents of OMR reading zone using optical mark recognition. The purpose of this process is to detect whether a check box on a document is marked or not. This is done by analyzing the black to white ratio of the defined zones. Therefore, an OMR reading zone can contain only one check box. If you have more than one check box on the document, you have to create an OMR reading zone for each of them.

#### Task Prerequisites

The prerequisites for this task are:

- The program is in Definition Mode.
- On the *Classes* tab on the left panel of the window, select a class.

#### Creating OMR Reading Zones

To create reading zones:

- 1) Load a document into the viewer that belongs to the selected class.
- 2) From the menu, select *View - Show Page*. The viewer toolbar displays additional buttons for zone creation:






Button	Description
	Activates the selection tool
	Creates an OCR reading zone
	Creates an OMR reading zone
	Creates a barcode reading zone
	Creates an anchor

Table 10-1: Controls for creating reading zones

- 3) In the viewer panel, click the respective toolbar button to create an OMR zone.
- 4) Click on the document and drag to create a rectangular reading zone around the check box. The reading zone is displayed as transparent rectangles with red borders. Each zone has a label that displays the name of the zone. The initial name is created from the word *Zone* and a consecutive number.

### 10.8.2. General Tab

You need to specify threshold values to distinguish Unchecked from Checked areas.

#### Task Prerequisites

- The application is in Definition Mode.
- On the *Classes* tab on the left panel of the window, select a class.
- Select then the *Fields* tab on the left panel of the window so that it is in the foreground.
- On the right side of the window, select the *Fields* tab.
- Click *OCR Settings...*
- Select *OMR* as Recognition type.
- On the *General* tab you can see the settings for the Cairo OMR engine:

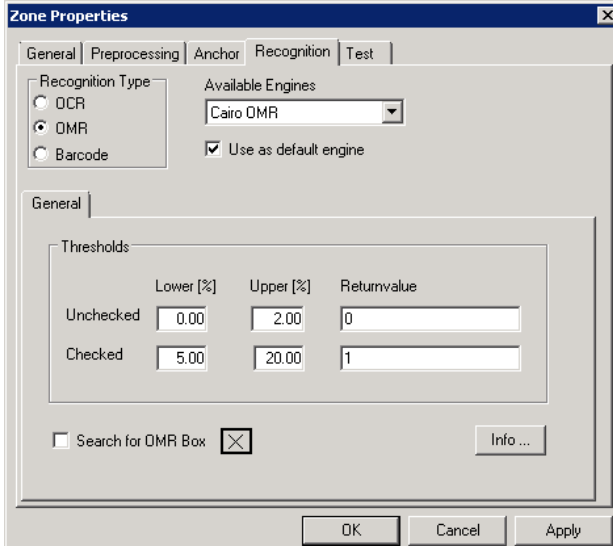


Figure 10-30: General tab of the Cairo OMR engine

The following options are available:

- *Unchecked:*
  - Lower [%]: Lower limit for degree of blackness in percent.
  - Upper [%]: Upper limit for degree of blackness in percent.
  - Returnvalue: A number or a character string. This value is returned if the measured degree of blackness is between the lower and the upper limit for Unchecked. In this case, it can reliably be assumed that the OMR zone is unchecked.
- *Checked:*
  - Lower [%]: Lower limit for degree of blackness in percent. The value should of course exceed the upper limit for Unchecked.
  - Upper [%]: Upper limit for degree of blackness in percent.
  - Returnvalue: A number or a character string. This value is returned if the measured degree of blackness is between the lower and the upper limit for Checked. In this case, it can reliably be assumed that the OMR zone is checked.

If it cannot be decided whether the OMR zone is checked or not, i.e. for all degrees of blackness outside the ranges specified, a question mark is returned, similar to the behavior of

OCR engines with unrecognized characters. The OMR zone should not be smaller than the specified zone, and there should not be more than one rectangle in the specified zone.

In particular, if your OMR zones are filled in manually, do not choose too high values for the upper limit of checked OMR zones. A very high degree of blackness may rather indicate that a manual entry has been corrected. If this can be safely excluded, you can also configure the OMR recognition to never return question marks.

- *Search For OMR Box*: When the *Search For OMR Box* checkbox is selected, an algorithm will search for a black square box within the defined zone. For better recognition results, the recognition is then applied for this square box *zone*. If nothing was found, search for a check mark will be executed within the originally defined zone.
  - Unchecked: Returned value will be 0 or 1.
  - Checked: The returned value is “?” if no OMR box was found.

## 10.9 OCR Field Level Digit booster

The Digit Booster boosts ‘digit to digit’ for alphanumeric fields and ‘alpha to digit’ for numeric fields. The digit to digit booster provides optimized digit recognition. The alpha to digit booster also repairs digits that were recognized as letters.

**The Boost field** option enables the alpha and digit to digit boosting (see [FIGURE 10-6: FIELD TAB](#)). Use this feature to correct a wrong recognition of numbers as letters (0 OCR'ed as letter 'O', 5 OCR'ed as 's' or 3 OCR'ed as 8 for example). It should be used on numeric fields only like Amounts or PO Number.

**The Boost digits only** option enables digit to digit boosting. Use this feature to correct a wrong recognition of numbers as numbers (3 OCR'ed as 8 for example). It should be used on alpha-numeric fields like Invoice Number or Date".

Boosted objects are worktexts for fields, candidates and table cells. The workdoc itself remains unchanged. Multi-page documents are supported but the engine speed may drop for multi-page documents with a lot of fields and candidates. Too high resolutions (>300 dpi) and too low resolutions (e.g. 96 dpi) lead to boost failure.

Boost Process takes place in the *Document\_PostExtract* event by default. If extra boosting is needed between the Analysis and Evaluation processes of the Extraction add the DWORD key “ExtraBoostingPriorToEvaluationEnabled” with a value of 1 to the windows registry in following location: HKEY\_LOCAL\_MACHINE\ Software\ Brainware\ Cedar.

### Boosting tables

The option *Boost digits only* in the Designer GUI is not working with table fields. This limitation can be circumvented using script. For table fields use the Column level property `pTable.IsColumnNumeric` at the very end of *Document\_PostExtract* event. Set the `isColumnNumeric` property to False if alpha to digit boosting is not allowed. See below example for a correct configuration of a table with regard to character boosting:

```
Private Sub Document_PostExtract(pWorkdoc As SCBCdrPROJLib.SCBCdrWorkdoc)
```

```
    Dim pField As SCBCdrField
    Set pField = pWorkdoc.Fields.ItemByName("LineItems")
```

```
pField.Table(pField.ActiveTableIndex).IsColumnNumeric("Date") = False  
pField.Table(pField.ActiveTableIndex).IsColumnNumeric("Unit Price") = True  
pField.Table(pField.ActiveTableIndex).IsColumnNumeric("Quantity") = True  
pField.Table(pField.ActiveTableIndex).IsColumnNumeric("Total") = True
```

End Sub

"

## Migration

The Format Analysis Engine delivers candidates for Brainware Extraction Engine, so that re-learning is not required for Booster migration. For Digit to digit boosting re-learning is for sure NOT useful. For Alpha to Digit boosting re-learning can be helpful.

## Reviewing results

For information on reviewing booster results see section [12.4.5 FIELD-LEVEL DIGIT BOOSTER CANDIDATE EVALUATION](#).

## Booster evaluation

For evaluation via script the ptrCandidate.Worktext.GetCharAttributes method is available:

```
ptrCandidate.Worktext.GetCharAttributes(...)
```

```
Confidence = (Confidence - 101) : 100.00
```

```
Boosted char vs. Original char
```

**Always** supply level 3 logs when reporting Digit Booster related issues or defects.

**Always** apply statistical evaluation of quality separately for the two boosting methods (Alpha to digit/digit to digit).



## 11 Regular Expressions

### 11.1 What are Regular Expressions?

Regular expressions are a generalized pattern language that WebCenter Forms Recognition uses for format analysis. With regular expressions, strings that match a specified pattern can be identified in a document's text. Although regular expressions look cryptic, they are easy to learn.

Try using regular expressions if you have problems getting the correct candidates for field extraction with simple expressions. Regular expressions precisely describe the candidates you are looking for.

### 11.2 Literal Characters in Regular Expressions

Two types of characters are used in regular expressions: Literal and special.

Literal characters are made up of normal text characters, such as letters or numbers, and of all symbols that are not used as operators.

The second type of characters is special characters. These are symbols that can be used as operators.

\*     -     [     ]     ^     {     }     \     .

If a special character appears in a regular expression as a literal character, it must usually be marked so that it is treated as a literal character and not as an operator. To use a special character as a literal character, place a backslash in front of the symbol, or enclose the symbol with square brackets.

### 11.3 Operators in Regular Expressions

Operators consist of one or more special characters.

Operator	Description
	<p><i>Match-self Operator</i> This operator is represented by any literal character. Its definition is included for completeness only. It matches the character itself. <i>Example:</i> t matches only the string t. It does not match the string tt.</p>
	<p><i>Concatenation Operator</i> This operator is not represented by any character. It concatenates two regular expressions a and b. b is simply placed after a. The result is a regular expression that will match a string if a matches its first part and b matches the rest. <i>Example:</i> xy matches xy.</p>
.	<p><i>Match-Any-Character Operator</i> This operator is represented by a period. It matches any single printing or non-printing character. <i>Example:</i> a.b matches any three-character string beginning with a and ending with b.</p>

Operator	Description
*	<p><i>Match-Zero-or-More Operator</i></p> <p>This operator is represented by an asterisk. The * operator is always used following a literal character or a special character. It repeats the smallest possible preceding regular expression as many times as necessary to match the pattern, including one repetition and zero repetitions.</p> <p><i>Examples:</i></p> <p>o* matches any string made up of zero or more o's.</p> <p>f o* matches f, fo, foo and so on.</p>
\{count\ \{min,\ \{min, max\}	<p><i>Interval Operators</i></p> <p>The Open Interval Operator is represented by \{. The Close Interval Operator is represented by \}. Interval operators repeat the smallest possible preceding regular expression a specified number of times:</p> <p>\{count\} matches exactly &lt;count&gt; occurrences of the preceding regular expression.</p> <p>\{min, \} matches at least &lt;min&gt; occurrences of the preceding regular expression.</p> <p>\{min, max\} matches between &lt;min&gt; and &lt;max&gt; occurrences of the preceding regular expression.</p> <p>The interval is invalid if &lt;min&gt; is greater than &lt;max&gt;, or any of &lt;count&gt;, &lt;min&gt;, or &lt;max&gt; is outside the range 0 - 256.</p>
[ ] [^ ]	<p><i>Character Class Operators</i></p> <p>[ and ] are used in regular expressions as special characters to indicate a character class. A character class matches a single character, regardless of how many characters or character ranges are defined in the character class.</p> <p>Characters contained in the class can be listed individually. Alternatively, character ranges can be specified using the - range operator. A character class can contain as many individual characters as needed, and can contain multiple character ranges.</p> <p>A negated character class can also be specified. A negated character class consists of any character except the characters and/or ranges listed in the character class. A negated character class includes the ^ metacharacter as the first character in list.</p> <p>Most special characters lose any special meaning inside a character class. The following characters still have a special meaning at certain positions within an expression:</p> <ul style="list-style-type: none"> <li>] closes the character class unless it is the first character.</li> <li>\ quotes the next character unless it is the last character before the closing bracket.</li> <li>- represents the range operator unless it is the first character after the opening bracket or the last character before the closing bracket.</li> </ul> <p>Empty character classes are invalid.</p>
	<p><i>Range Operator</i></p> <p>The range operator is represented by a hyphen -. It specifies a range of characters when used inside a character class.</p> <p><i>Examples:</i></p> <p>a-f matches any character between a and f: a, b, c, d, e, or f.</p> <p>A-Z matches any capital letter.</p> <p>a-z matches any lower case letter</p> <p>0-9 matches any number.</p>
	<p><i>Escape Operator</i></p> <p>The escape operator is represented by a backslash \. It is used to quote certain special characters: If you need to use the literal value of a special character within an expression, then a backslash should be inserted before that special character to signal that the following character is to be treated literally.</p>

Table 11-1: Special Characters in Regular Expressions

### 11.3.1. Example: Find an Invoice Number

#### Task

The invoice number is between 6 and 10 characters long.

It consists of digits but may also contain an R or a K.

Sometimes the numbers are separated by a slash /.

**Solution**

`[0-9RK/] \{6-10\}`

*matches 892785/222, 9R095949, or 000000484.*

**How the expression was constructed**

**[0-9RK/]** The first half of the expression indicates the characters that could be included in the invoice number. The characters 0-9 state that the invoice number consists of any character from 0 through nine, including 0 and 9. RK are the letters that might be included.

The forward slash shows that the invoice number might include a forward slash. These characters are enclosed in square brackets to show a character class.

**\{6-10\}** The second half of the expression indicates the existence of an interval of between 6 and 10 characters long, including 6 and 10. The two backslashes and the open and close French brackets are necessary components of an interval expression.

**11.3.2. Example: Find a Date****Task**

The date consists of digits and can have a short or a long format. Various characters are used: period, slash, and hyphen.

**Solution**

`[0-3][0-9][-./][0-3][0-9][-./][0-9]\{2,4\}`

*matches 02-20-2000, 02/20/00, or 02.20.2000.*

**How the expression was constructed**

**[0-3][0-9][-./]**

**[0-3]** indicates that the numerical expression for the month is either one or two characters long.

**[0-9]** The numerical expression for month can include any number from 0 to 9, including 0 to 9.

**[-./]** Either a dash, a period, or a slash separates day from the month (which is defined in the second part of the expression.)

**[0-3][0-9][-./]**

**[0-3]** The numerical expression for the day of the week is either one or two characters long.

**[0-9]** The numerical expression for day of the week can include any number from 0 to 9, including 0 to 9

**[-./]** Either a dash, a period, or a slash separates the month from the year (which is defined in the third part of the expression.)

**[0-9]\{2,4\}**

**[0-9]** The numerical designation for year can include any numeral from 0 to 9, including 0 and 9

**\{2,4\}** The year can be either two characters long or for characters long.

**Task**

Spelled months must also be considered.

### Solution

```
[[[AJFMSOND] [a-ceg-il-prvy]{\2,8}[,[ ]\{1,2} \{2,4}
```

*matches May 13, 2000, for example.*

### How the expression was constructed

```
[[[AJFMSOND] [a-ceg-il-prvy]{\2,8}[,[ ]\{1,2} \{2,4}
```

[[AJFMSOND] Indicates that the words may begin with the listed uppercase characters. The list includes only the letters that begin the names of the months.

[a-ceg-il-prvy] The lowercase letters that are building blocks for the names of the months

[,] The comma after the month

[ ] The square brackets, with a space in the middle, indicate the space between the comma and the year, {\2,8} specifies that the month can be two to nine characters long \{1,2} The day of the month can be one or two characters long \{2,4} The third part of the expression indicates that the year can be either two or four characters long.

### 11.3.3. Example: Find an E-mail Address

#### Task

Find an expression that is more specific than `*@*.*` that matches any word containing the @ character.

#### Solution

```
[a-z.]{2,20}@[a-z.]{2,10}
```

#### How the expression was constructed

[a-z.] Indicates that the email address can include of all letters in the alphabet (from A to Z, including A and Z) and that these letters are followed by a dot.

{2,20} Indicates that the first part of the email address is between 2 and 20 characters long, not including 2 and 20. (The author of the expression used a comma to show a non-inclusive range, instead of a dash to show an inclusive range.)

@ Indicates that the email address definitely includes an ampersand.

[a-z.] The last part of the email address can include all letters in the alphabet (from A to Z, including A and Z) and that these letters are followed by a dot.

{2,10} Indicates that the first part of the email address is between 2 and 10 characters long, not including 2 and 10.

## 12 Advanced Evaluation Settings

### 12.1 Project-Level Settings

#### 12.1.1. Classification Interpretation – How Does It Work?

Let's assume we have given classes, a given document, and several classification methods that return confidence levels. It is unlikely that all methods will return the same confidence level. To get the best bet on what the confidence really is, we need to compute a combined result.

WebCenter Forms Recognition supports three ways to do this:

- The Maximum method
- The Average method
- The Weighted Distance method.

##### 12.1.1.1. The Maximum Method

In this method, only the maximum confidence level of a class is taken into account. This is the default setting.

##### Example

	Result	Method A	Method B
<b>CLASSID</b>	Percentage	Percentage	Percentage
<b>Class A</b>	88	88	68
<b>Class B</b>	67	67	60
<b>Class C</b>	63	62	63

Table 12-1: The Maximum method for classification evaluation

##### 12.1.1.2. The Average Method

The average of all confidence levels for a class is taken into account.

##### Example

	Result	Method A	Method B
<b>CLASSID</b>	Percentage	Percentage	Percentage
<b>Class A</b>	78	88	68
<b>Class B</b>	63.5	67	60
<b>Class C</b>	62.5	62	63

Table 12-2: The Average method for classification evaluation

##### 12.1.1.3. The Weighted Distance Method

The maximum of all results for all classes is taken as a reference. For each result, the distance to this reference is calculated. For each class, the distances are added. This sum is subtracted from the maximum result obtained for that class.

**Example**

	Result	Method A	Method B
<b>CLASSID</b>	Percentage	Percentage	Percentage
<b>Class A</b>	68 = 88 –(0+20)	88 (0)	68 (20)
<b>Class B</b>	18 = 67 –(21+28)	67 (21)	60 (28)
<b>Class C</b>	8 = 63 – (28-27)	62 (26)	63 (25)

Table 12-3: The Weighted Distance method for classification evaluation

In general, the portion of e-mails that will be classified is as follows:

*Maximum > Average > Weighted Distance*

In particular, with the Weighted Distance method, it is highly unlikely that e-mails will be assigned to a class if the class did not get high confidence levels from both methods applied.

**12.1.2. Project-Level Standard Classification – How Does It Work?**

Standard classification uses two values defined at the project level to decide whether a document can be classified or not:

- A minimum confidence level called a threshold.
- A minimum difference in confidence between the best class and the second-best competitor called distance.

Both requirements must be met to classify a document.

**Example**

Let’s take our results from section [CLASSIFICATION INTERPRETATION – HOW DOES IT WORK?](#), where the Maximum method was applied.

	Result
<b>Class A</b>	88
<b>Class B</b>	67
<b>Class C</b>	63

Let’s further assume that the default values for threshold and distance are applied:

Parameter	Value
<b>Threshold</b>	70
<b>Distance</b>	20

**Result**

Class A is the best class. Confidence for class A is above the threshold.

Class B is the second-best class. The distance to Class A is 21. This is just above the required distance.

The document will be assigned to Class A.

**Example**

Let's take our results from section [CLASSIFICATION INTERPRETATION – HOW DOES IT WORK?](#), where the Average method was applied.

	Result
Class A	78
Class B	63.5
Class C	62.5

Let's again assume that the default values for threshold and distance are applied:

Parameter	Value
Threshold	70
Distance	20

### Result

Class A is the best class. Confidence for Class A is above the threshold.

Class B is the second-best class. The distance to Class A is 14.5. This is just below the required distance.

The document cannot be classified using standard classification.

### Example

Let's take our results from section [CLASSIFICATION INTERPRETATION – HOW DOES IT WORK?](#), where the Weighted Distance method was applied, and the default values for threshold and distance.

	Result
Class A	68
Class B	18
Class C	8

Let's again assume that the default values for threshold and distance are applied:

Parameter	Value
Threshold	70
Distance	20

### Result

- Class A is the best class. Confidence for class A is below the threshold.
- Distances to the next classes are very high.
- The document cannot be classified using standard classification

If you use the weighted distance method, you probably have to work with smaller thresholds to obtain a satisfactory fraction of classified documents.

### 12.1.3. Project-Level Parent Classification – How Does It Work?

So far, no hierarchical elements were taken into account for classification. However, if classification is not possible using the flat mechanism, hierarchical elements may resolve the conflict.

One of the available hierarchical methods is parent classification. If parent and child classes compete for the same documents, the documents will rather be assigned to the children. If there are several children with similar confidence levels, the parents can decide which class is the correct one.

Parent classification uses two additional values defined at the project level:

A parent threshold (50 percent by default)

A parent distance (10 percent by default).

In below figure, classification would occur to the medium branch, and the child class with 80 percent confidence would win.

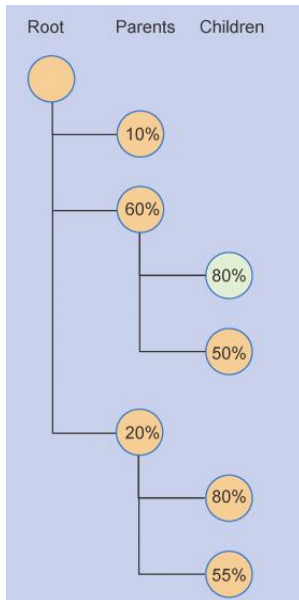


Figure 12-1: Parent Classification: Illustration of parent classification

For more information about hierarchical classification elements, refer to section [CLASS-LEVEL SUBTREE CLASSIFICATION – HOW DOES IT WORK?](#).

#### 12.1.4. Project-Level Default Classes – How Do They Work?

Normally, all documents that cannot be assigned to a particular class are marked as Not Classified in Designer's Runtime Mode.

During production, they are marked by WebCenter Forms Recognition Runtime with a particular status value indicating that classification was not successful. The corresponding documents will not be processed any further.

To prevent this, you can define a default class for each project that will be assumed if classification fails. You can also use this feature to create applications without classification, but with extraction.

#### 12.1.5. Modifying Project-Level Settings

##### Task Prerequisites

The prerequisites for this task are:

- The program is in Definition Mode.



- In the *Classes* tab on the left side of the window, the \*.sdp file for your project is selected.
- On the right side of the window, the tabs with class/field properties are visible.

## Available Options

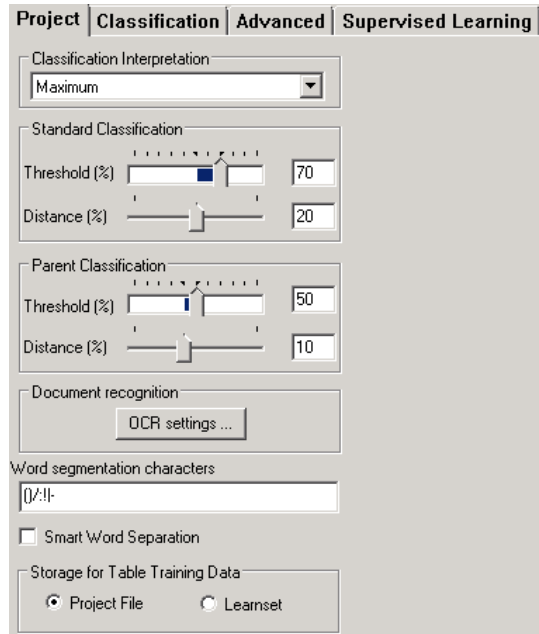


Figure 12-2: Project tab

On the *Classification* tab, the following options are available:

- To set a default class, select an entry from the *Default Classification Result* list box.

On the *Project* tab, the following options are available:

- To change the algorithm to obtain combined results for given classes, select an entry from the *Classification Interpretation* list box.
- To change *Threshold* or *Distance* for *Standard Classification*, use the sliders or type values into the corresponding text boxes.
- To change *Threshold* or *Distance* for *Parent Classification*, use the sliders or type values into the corresponding text boxes.

## 12.2 Method-Level Classification Settings

### 12.2.1. Method-Level Absolute Results – How Does They Work?

For a certain class and document, each classification method may return different result types. Possible result types are:

- *Confidence*: This document belongs to a specified class with a probability that is given in percent.
- *Yes*: This document is definitely in this class.
- *Maybe*: This document might belong to this class.

- *No*: This document does definitely not belong to this class.
- *-*: This method is not applied.

The various classification methods are not equally reliable. For example, it is dangerous to decide that a document is an invoice based only on the fact that it is a certain size. [TABLE 12-4](#) presents the possible results for each classification method. It also shows which result type is returned by default.

Method	Result	Default or Configured
Brainware classification	Confidence Yes No	Default Configured Configured
ASSA	Confidence Yes No	Default Configured Configured
Phrase classification	Confidence Yes No	Default Configured Configured
Image size classification	Maybe No	Default Default
Form classification	Yes Maybe Confidence	Default Default Configured
Layout classification	Confidence Yes No	Default Configured Configured
Brainware Layout Classification	Maybe No Confidence	Configured Configured Default
Language classification	Maybe No Confidence	Configured Configured Default

Table 12-4: Possible automatic results vs. classification methods

Yes and No are called absolute results. If the program is configured accordingly, they are obtained if confidence levels returned by one of the methods are either very high or very low.

If a method returns Yes for a class, the combined result will be Yes. Classification of the document will stop at this stage. All other classes will be excluded automatically.

If a method returns No for a class, the combined result will be No. Within the current branch, classification of the document will stop at this stage: All child classes will be excluded automatically.

You can use absolute results to speed up the entire classification process. Image size classification is a suitable method to exclude entire branches very quickly. Brainware Layout Classification can be used to positively confirm formal classes very quickly. The same is true for forms classification.

### 12.2.2. Method-Level Multiple Views – How Do They Work?

By default, in WebCenter Forms Recognition, each document can only be assigned to one class. However, WFR also supports the concept of multiple views, i.e. a single set of documents can be classified several times, each time according to different criteria. Within a single view, a document cannot belong to more than one class. However, a document may well belong to more than one class if the classes are in different views. You could, for example, first classify news by topic and then classify the same news by geographical region. Multiple views may also be required to process documents with multiple topics.

**Example:**

		Result
<b>Class A</b>	View 1	90
<b>Class B</b>	View 1	69
<b>Class C</b>	View 2	60
<b>Class D</b>	View 2	80

Let's assume that the default values for threshold and distance are applied:

Parameter	Value
<b>Threshold</b>	70
<b>Distance</b>	20

**Result:**

- Class A is the best class from View 1. Confidence for class A is above the threshold.
- Class B is the second-best class in View 1. The distance to Class A is 31. This is above the required distance.
- Class D is the best class from View 2. Confidence for class D is above the threshold.
- Class C is the second-best class in View 2. The distance to Class D is 20. This is just the required distance.
- The document is assigned to Class A and to Class D.

A more complex application of views is the creation of a cascading classification scheme. ([FIGURE 12-3](#)) In this case, you need to assign parent classes and child classes to different views. Document redirection from the parent class to the subordinated view occurs automatically.

For example, you can use the fast formal classification methods (Brainware Layout Classification, image size classification) to roughly sort your documents. Then apply the semantic classification methods (Brainware classification, phrase classification) to pre-sorted documents to determine the target classes. You will need fewer classes and improve the performance at the same time.

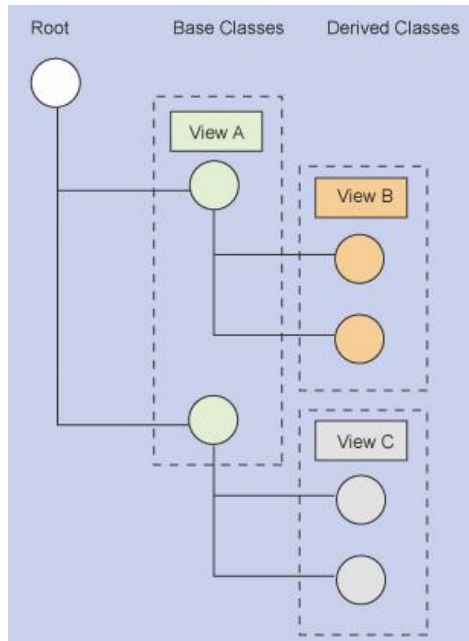


Figure 12-3: Cascading classification scheme

### 12.2.3. Modifying Settings for Brainware Classification Methods

#### Task Prerequisites

The prerequisites for this task are:

- The program is in Definition Mode.
- The panel on the left side of the window displays the *Classes* tab in the foreground.
- The panel on the right side of the window with the class/field properties is visible.

#### Creating a View

To create a view:

- 1) In the *Classes* tab, select the \*.sdp file. The *Classification* tab is displayed on the right side of the window.
- 2) Under *Used Classification Engines...*, select Brainware Classify Engine. A corresponding tab is displayed below the list box.

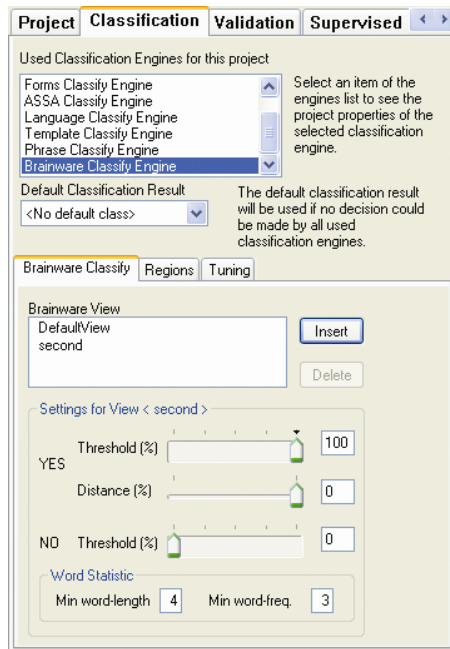


Figure 12-4: Settings for Brainware classification

- 3) On the *Brainware Classify Engine* tab, click *Insert*. The *New Brainware View* dialog box is displayed.
- 4) In the text box, enter a name for the new view. The name must not contain spaces.
- 5) Click *OK*. The new view is displayed in the *Brainware View* list box.

For instructions on how to assign classes to views, refer to section [MODIFYING CLASS-LEVEL SETTINGS](#).

Each WebCenter Forms Recognition project automatically contains a view called *DefaultGroup*. Without user intervention, all classes use this view.

### Changing Evaluation Settings

To change the evaluation settings for a view:

- 1) In the *Classes* tab, click the \*.sdp file. The *Classification* tab is displayed on the right side of the window.
- 2) Under *Used Classification Engines...*, select *Brainware Classify Engine*. A corresponding tab is displayed below the list box.
- 3) Select the view from the *Brainware View* list box.
- 4) Under *Settings for view...*, use the sliders or text boxes to set the following values for absolute classification:
  - *YES Threshold*: Minimum confidence for a class within the view to return Yes.
  - *YES Distance*: Minimum distance to the second-best class within the view to return Yes.
  - *NO Threshold*: Maximum confidence for a class within the view to return No.

- 5) Under *Word Statistic*, you can specify parameters for words that are to be included in the view's dictionary that embodies the base for learning with Oracle's neural network. As a rule, you do not have to change these values.
  - *Min. word length*: Minimum number of characters in words. Shorter words will not be taken into account.
  - *Min. word frequency*: Minimum number of occurrences in the Learnsets of this view. Words that occur less frequently will not be considered.

You can restrict the text base for dictionary creation to certain areas within the document. This may speed up processing as no full-page OCR is carried out. To actually use this advantage, make sure that the WebCenter Forms Recognition Runtime settings do not override your restrictions. For further information on OCR settings in WFR Runtime, please refer to the ***Runtime Server User's Guide***.

### Restricting Dictionary Creation

To restrict the dictionary creation to certain areas within the documents:

- 1) In the *Classes* tab, click the entry representing your project. The *Classification* tab is displayed on the right side of the window.
- 2) Under *Used Classification Engines...*, select Brainware Classify Engine. A corresponding tab is displayed below the list box.
- 3) Select a view from the Brainware *View* list box.
- 4) Select the *Regions* tab.

The screenshot shows the 'Regions' tab in the Brainware Classify interface. It features a 'Region Settings for View < second >' section with a 'Restrict engine to region' checkbox. Below this are three checkboxes for 'First Page', 'Subseq.', and 'Last Page'. A grid of input fields allows setting percentages for 'Top (%)', 'Bottom (%)', 'Left (%)', and 'Right (%)' for each of the three page types. The current values are: Top (0, 0, 0), Bottom (100, 100, 100), Left (0, 0, 0), and Right (100, 100, 100).

Figure 12-5: Regions tab for Brainware view

- 5) Check the *Restrict engine to region* check box.
- 6) Select the pages that are affected by the restriction.
- 7) For each selected page, enter the regions that are to be taken into account.

As you are setting regions, remember that Top 100 %, Bottom 100%, Left 100% and Right 100 % equal an empty area. The entire area would be specified by Top 0 %, Bottom 0%, Left 100%, Right 100%. Top 20, bottom 100 would cover the bottom 80 percent of the document, cutting 20 percent off from the top. Top 100, Bottom 20, would cover the top 80 percent of the document, cutting 20 percent off from the bottom.

## 12.2.4. Modifying Brainware Layout Classification Settings at the Method Level

### Task Prerequisites

The prerequisites for this task are:

- The program is in Definition Mode.
- The panel on the left side of the window displays the *Classes* tab in the foreground.
- The panel on the right side of the window with the class/field properties is visible.

### Changing Settings

To change the settings for this method:

- 1) In the *Classes* tab, click the entry representing your project. The *Classification* tab is displayed on the right side of the window.
- 2) Under *Used Classification Engines...*, select Brainware Layout Classification. The *appropriate* tab is displayed below the list box.

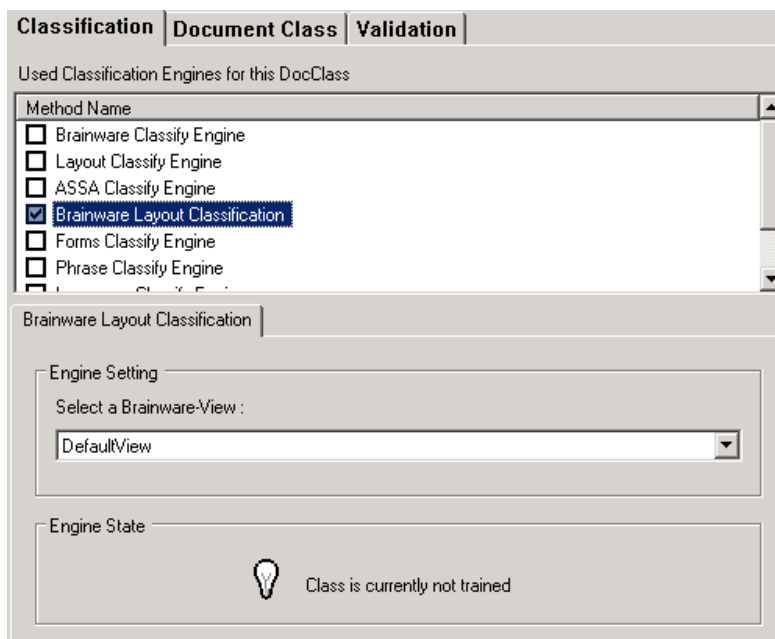


Figure 12-6: Settings for Brainware Layout Classification

For information on the Brainware-View please see section [12.3.3](#).

## 12.2.5. Modifying Phrase Classification Settings at the Method Level

### Task Prerequisites

The prerequisites for this task are:

- The program is in Definition Mode.
- The panel on the left side of the window displays the *Classes* tab in the foreground.
- The panel on the right side of the window shows the *Classification/ Document Class/Validation* tabs.

### Changing Settings

To change the evaluation settings for this method:


- 1) In the *Classes* tab, click the entry representing your project. The *Classification* tab is displayed on the right side of the window.
- 2) Under *Used Classification Engines...*, select *Phrase Classify Engine*. The corresponding tab is displayed below the list box.


### 12.2.5.1. Editing Phrase Classification

On this tab, you can insert, delete, edit, test, import, and export lists of phrases to search for.

#### Inserting and Deleting Phrases

There are two ways to insert phrases: By typing them in the *New Phrase* field, or by selecting a word from a document. For the second way:

- 1) Click *Highlight All Words*  on the toolbar.
- 2) On the document, select a word or phrase. The word now populates the *New Phrase* field. If you click another word, it will be appended to the phrase.

Whether you type your phrases or select them from a document, click *Insert*  to add the phrase to the Active Phrase Grid.

To delete a phrase from the grid, select the phrase and click *Delete* .

#### Editing a Phrase

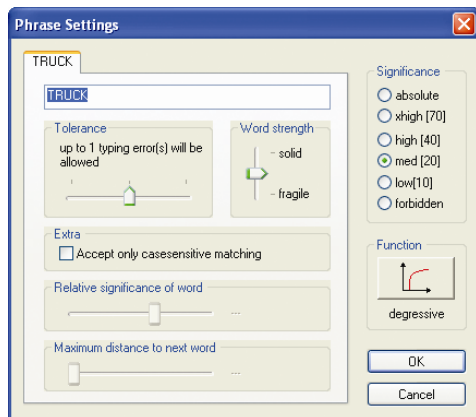


Figure 12-7: Editing Phrase Settings

Editing a phrase enables you to increase the complexity of phrase analysis. You can set Word Strength, tolerance, case sensitivity, enhanced word postulation (under function) and enhanced significance settings – absolute and extra high.

#### Restricting Analysis Areas

You can restrict the text that is analyzed to certain areas within the document, for any of the phrases you've included. This may speed up processing as no full-page OCR is carried out. To actually use this advantage, make sure that the Runtime Server settings do not override your restrictions.

To restrict the method to certain areas within the documents:



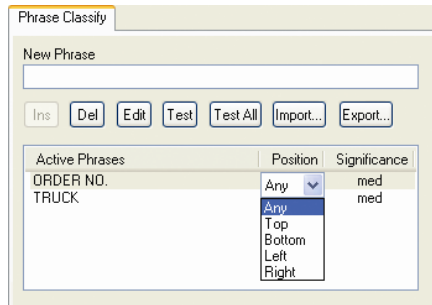


Figure 12-8: Phrase Positioning

- 1) On the *Phrases* Grid, select a phrase to restrict.
- 2) In the *Position* column, click the position of the phrase. The default is *Any*. By changing the selection, you can restrict OCR to the top half, bottom half, left half, or right half of the document. If any part of the phrase extends into the selected half then it will be detected.

### Establishing Significance Settings

- 1) On the *Phrases* grid, select a phrase.
- 2) In the *Significance* column, click the current significance setting for the phrase. The default is *Medium*. By changing the selection, you can vary how much weight the phrase will carry during analysis. The settings are High, Medium, Low, and Forbidden.

### Sharing Phrase Classification Words and Settings with Other projects

The *Import* button enables you to import classification settings from other projects in which the phrases were saved as Export (\*.exp) files. Likewise, via the *Export* button, you can save phrases to share with other projects. *Import* and *Export* is done at the Class level.

### 12.2.6. Modifying Image Size Classification Settings at the Method Level

At this level, no modifications are possible to Image Size Classification.

### 12.2.7. Modifying Forms Classification Settings at the Method Level

#### Task Prerequisites

The prerequisites for this task are:

- The program is in Definition Mode.
- The panel on the left side of the window displays the *Classes* tab in the foreground.

The panel on the right side of the window with the class/field properties is visible.

#### Changing Evaluation Settings

To change the evaluation settings for this method:

- 1) In the *Classes* tab, click the entry representing your project. The *Classification* tab is displayed on the right side of the window.
- 2) Under *Used Classification Engines...*, select *Forms Classify Engine*. The corresponding tab is displayed below the list box.

Forms Classify

Select the type of interpretation

binary classification (YES/Maybe)

weighted classification (0-100%)

Figure 12-9: Settings for forms classification

3) Select one of the following options:

- *Binary classification*: Possible atomic results of forms classification are Yes and Maybe. This is the default.
- *Weighted classification*: Forms classification returns a confidence as atomic result.

## 12.3 Class-Level Settings

### 12.3.1. Class-Level Subtree Classification – How Does it Work?

Subtree classification is a means to push classification from parent to child classes by defining reduced requirements for classification into the subtree.

Subtree classification uses two additional values defined at the class level:

- A subtree threshold (60 percent by default)
- A subtree distance (10 percent by default.)

It is also possible to entirely exclude the parent class as valid classification result.

In the example below, without subtree classification, the parent class with 90 percent confidence would win. With subtree classification, classification would lead to the medium branch, and the child class with 60 percent confidence would win.

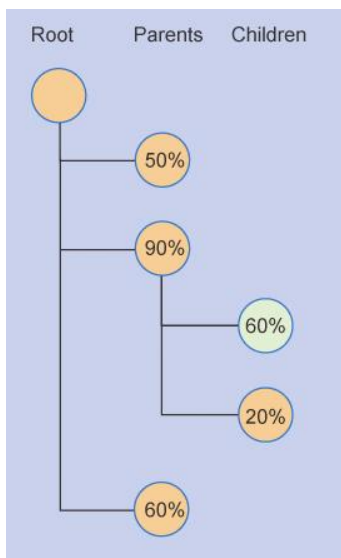


Figure 12-10: Subtree Classification

### 12.3.2. Class-Level Redirection – How Does it Work?

Redirection is something very simple: you just define that in case of classification to a Class A, the documents will in fact be assigned to a different Class B. Use this to combine several classes or in cascading classification schemes.

### 12.3.3. Modifying Class-Level Settings

#### Task Prerequisites

The prerequisites for this task are:

- The program is in Definition Mode.
- The panel on the left side of the window displays the *Classes* tab in the foreground.
- The panel on the right side of the window with the class/field properties is visible.
- A suitable classification tree exists. When working with multiple views, you can, for example, create a base class for each view and insert derived classes below.

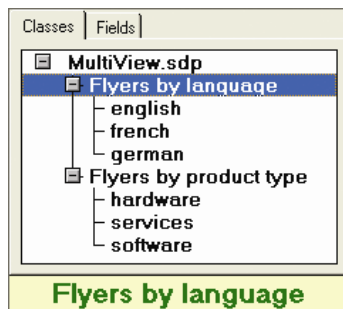


Figure 12-11: Sample classification tree for working with multiple views

#### Changing Class-Level Settings

To change the class-level settings:

- 1) In the *Classes* tab on the left side of the window, select a class.
- 2) On the right side of the window, select the *Classification* tab.
- 3) To assign the class to a view, click the Brainware Classify Engine method name
- 4) In the Brainware *Classify Engine* tab, select a view from the list box.
- 5) On the right side of the window, select the *Document Class* tab.
- 6) To use a different display name for the class, enter it into the corresponding text box.
- 7) To exclude the class as classification result, clear the *DocClass is a...* check box.
- 8) To hide the class in Verifier, clear the *Visible in correction...* check box.
- 9) To push classification to the children of this class, check the *Force Subtree Classification* check box. To change *Threshold* or *Distance* for *Subtree classification*, use the sliders or type values into the corresponding text boxes.
- 10) To redirect classification results, select a target class from the list box at the bottom of the tab.
- 11) To apply a static page count to all documents of this class, check the static pagecount check box and specify the number of pages using the spin box.

Figure 12-12: Document class tab

### 12.3.4. Modifying Settings for Brainware Layout Classification Methods

#### Task Prerequisites

The prerequisites for this task are:

- The program is in Definition Mode.
- The panel on the left side of the window displays the *Classes* tab in the foreground.
- The panel on the right side of the window with the class/field properties is visible.

#### Creating a View

To create a view:

- 1) In the *Classes* tab, select the \*.sdp file. The *Classification* tab is displayed on the right side of the window.
- 2) Under *Used Classification Engines...*, select Brainware Layout Classify Engine. A corresponding tab is displayed below the list box.

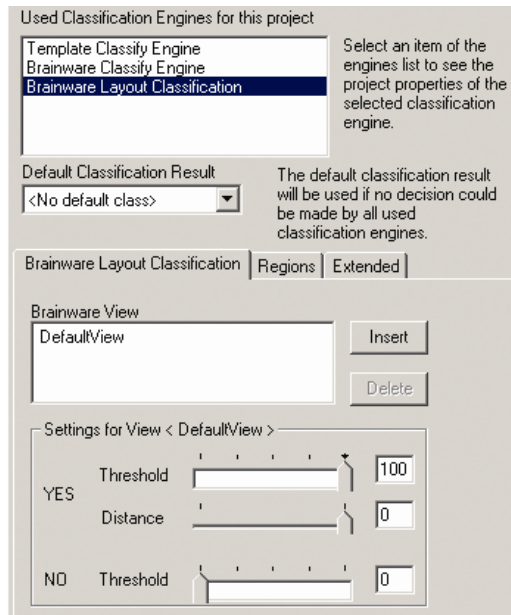


Figure 12-13: Settings Brainware Layout Classification

- 3) On the *Brainware Layout Classify Engine* tab, click *Insert*. The *New Brainware View* dialog box is displayed.
- 4) In the text box, enter a name for the new view. The name must not contain spaces.
- 5) Click *OK*. The new view is displayed in the *Brainware View* list box.

For instructions on how to assign classes to views, please refer to section [MODIFYING CLASS-LEVEL SETTINGS](#).

Each WebCenter Forms Recognition project automatically contains a view called DefaultGroup. Without user intervention, all classes use this view.

### Changing Evaluation Settings

To change the evaluation settings for a view:

- 1) In the *Classes* tab, click the \*.sdp file. The *Classification* tab is displayed on the right side of the window.
- 2) Under *Used Classification Engines...*, select *Brainware Layout Classify Engine*. A corresponding tab is displayed below the list box.
- 3) Select the view from the *Brainware View* list box.
- 4) Under *Settings for view...*, use the sliders or text boxes to set the following values for absolute classification:
  - *YES Threshold*: Minimum confidence for a class within the view to return Yes.
  - *YES Distance*: Minimum distance to the second-best class within the view to return Yes.
  - *NO Threshold*: Maximum confidence for a class within the view to return No.

### Restricting Dictionary Creation

To restrict the dictionary creation to certain areas within the documents:

- 1) In the *Classes* tab, click the entry representing your project. The *Classification* tab is displayed on the right side of the window.
- 2) Under *Used Classification Engines...*, select *Brainware Layout Classify Engine*. A corresponding tab is displayed below the list box.
- 3) Select a view from the *Brainware View* list box.
- 4) Select the *Regions* tab.

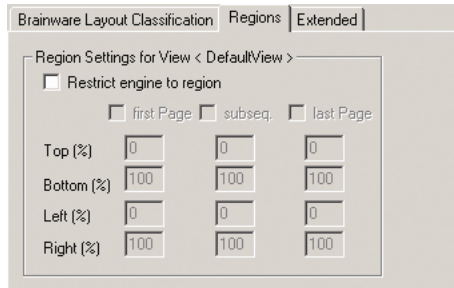


Figure 12-14: *Regions* tab for *Brainware Layout Classification* view

- 5) Check the *Restrict engine to region* check box.
- 6) Select the pages that are affected by the restriction.
- 7) For each selected page, enter the regions that are to be taken into account.

As you are setting regions, remember that Top 100 %, Bottom 100%, Left 100% and Right 100 % equal an empty area. The entire area would be specified by Top 0 %, Bottom 0%, Left 100%, Right 100%. Top 20, bottom 100 would cover the bottom 80 percent of the document, cutting 20 percent off from the top. Top 100, Bottom 20, would cover the top 80 percent of the document, cutting 20 percent off from the bottom.

### Extended BLC Specific Settings

To change the extended settings for the Brainware Layout Classification engine:

- 1) Select the *Extended* tab.
- 2) Under *Extended Settings*, set the following values:
  - *Horizontal covering frequency*: The number of zones each document is divided into across the page.
  - *Vertical covering frequency*: The number of zones each document is divided into down the page.
  - *Do not differentiate between digits*: Treat “123” and “987” as the same sequence of digits of the same length. Use this option when most numbers do not have specific meaning, for example, in the tables of a typical invoice document. Leave this option unchecked if the project’s documents contain specific sequences of digits that can be critical for layout classification e.g., bank account information specific to each vendor class.
  - *Attach delimiters to words*: This option should be checked if, in most cases, delimiters (like “,” “.” “/” “;”, etc.) are essential to distinguish between words. An American format date, for example, may be better treated as a single differentiating word – “00/00/000” - in the classification’s surrounding compared with 3 “noise” words “00”, “00” and “0000”.

*Note that the changes in settings will take effect only after the project has been saved and relearned.*

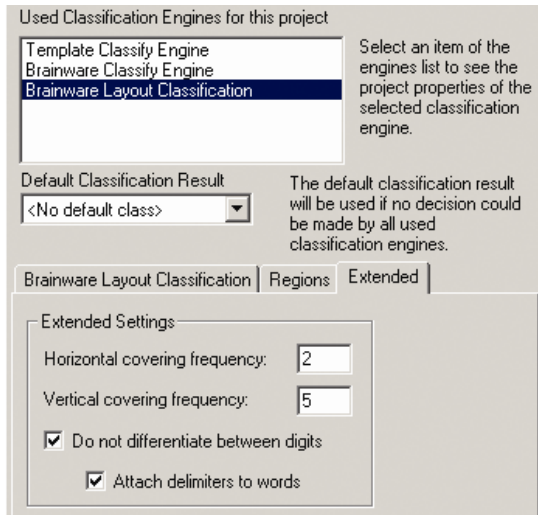


Figure 12-15: Extended Settings of Brainware Layout Classification Engine

## 12.4 Field-Level Settings

### 12.4.1. Field-Level Text Field Candidate Evaluation – How Does it Work?

Text field candidate evaluation works like classification evaluation.

Let’s assume we have a given text field, a given document, and *Brainware* extraction as an evaluation method that returns weights for each candidate.

Candidate evaluation uses two values defined at the field level to decide whether there is a valid candidate or not:

- A minimum weight called threshold (by default 50 %)
- A minimum difference in weight between the best candidate and the second-best competitor called distance (by default 10%.)

Both requirements must be met to extract a valid candidate.

**Example:**

	Weight of Candidate A	Weight of Candidate B	Weight of Candidate C
Field A	0.45	0.36	0.28
Field B	0.30	1.00	0.00
Field C	0.55	0.49	n/a

Table 12-5: Weights for Candidate evaluation

**Result:**

- For Field A, there is no valid candidate. The weight of Candidate A is below the threshold, and the distance to Candidate B is below the minimum distance.

- For Field B, candidate B is a valid candidate. Its weight is above the threshold. And the distance to the second-best competitor, Candidate A, is above the minimum distance.
- For Field C, there is no valid candidate. The weight of Candidate A is above the threshold, but the distance to Candidate B is below the minimum distance.

## 12.4.2. Modifying Text Field Settings

### Task Prerequisites

The prerequisites for this task are:

- The program is in Definition Mode.
- The panel on the left side of the window displays the *Fields* tab in the foreground.
- On the right side of the window, the tabs with class/field properties are visible.

### Modifying Evaluation Settings

To modify the evaluation settings for a field:

- 1) On the left side of the window, go to *Fields*.
- 2) Go to *Evaluation* tab on the right side of the window.
- 3) On the *Brainware Extraction* tab, to change the *Threshold* or *Distance* for *Brainware Extraction*, use the sliders or type values into the corresponding text boxes.

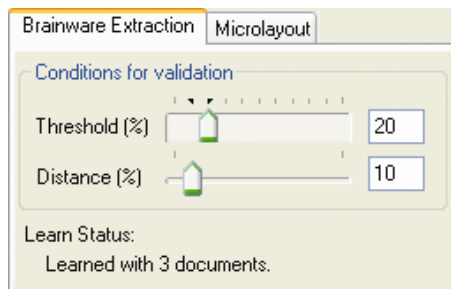


Figure 12-16: Evaluation parameters

- 4) Go to *Field* tab on the right side of the window. The validation parameters are displayed in the Validation group at the bottom of the tab.
- 5) If the current field is a derived field, you can use inherited validation settings of the parent class. To do this, check *Use derived validation*. This disables the remaining validation parameters.
- 6) If any field content should be treated as valid, select *Always valid*. This disables the remaining validation parameters.
- 7) To mark fields as invalid if they contain uncertain OCR results, select *No Rejects*. Use the slider to set the confidence required to accept characters.
- 8) To set a minimum length for valid field content, select *Min. Length* and set the required number of characters using the associated spin box. This option can also be used to make a field mandatory.



- 9) To set a maximum length for valid field content, check *Max. Length* and set the permitted number of characters using the associated spin box.

### 12.4.3. Table Field Candidate Evaluation – How Does it Work?

In table analysis, each candidate is evaluated using the criteria listed below.

Table property	Range of Results
Fraction of default columns identified	0.0 – 1.0
Fraction of table columns mapped	0.0 – 1.0
Fraction of entries found in cells which require entries	0.0 – 1.0
Fraction of neighboring cells without overlap (column view)	0.0 – 1.0
Fraction of neighboring cells without overlap (row view)	0.0 – 1.0
Fraction of correct formats	0.0 – 1.0

*Table 12-6: Table properties investigated during table candidate evaluation*

For each table candidate, the total significance is determined as the average of the results per property. All table candidates which reach a certain threshold for the total significance are transferred to the Workdoc, ordered by the total significance. By default, the candidate with the highest significance is used for data extraction.

### 12.4.4. Modifying Table Field Settings

During table analysis, field-level validation settings defined via the user interface are ignored. However, the evaluation can be influenced using WinWrap Basic scripts.

### 12.4.5. Field-Level Digit booster Candidate Evaluation

The use of the Digit booster (see section [10.1.4 FIELD-LEVEL SETTINGS](#)) in the text field OCR optimization enables an extended Candidate highlighting mode. Three colors indicate the Boost result reliability:

- Green: 'Boosted' and 'boosted secure' - This result will be adopted. The tooltip shows the action results and behaviour.
- Yellow: Engine is not sure - The character is not going to be converted, the decision has to be approved manually.
- Red: Engine is absolutely unsure, character is not going to be converted.

*Note:* Advanced highlighting for booster results review is not supported for table cell boosting.

## 13 Setting Up the Verification

WebCenter Forms Recognition features a dedicated application for quality assurance: WebCenter Forms Recognition Verifier. This application provides a way to check and correct the results of automatic document classification and data extraction. In addition, it enables you to manually index documents.

Verifier users are in charge of validating the documents, which are presented as batches. Whether a batch is valid or not determined using the following rules:

- A batch is valid if all of its documents are valid.
- A document is valid
  - if it has a valid classification result and
  - if it has a valid extraction or indexing result.
- An extraction result is valid if all the fields are filled with valid content.

The Verifier user is presented with all invalid or uncertain results from a given batch and needs to correct and confirm them. Normally, results that are invalid by definition cannot be confirmed by the user, but in the project definition, you can allow the user to validate invalid field results in exceptional cases. This procedure is called forced validation.

To enable you to adjust Verifier to the requirements of your custom application, two modes are available in Designer:

- The Verifier Design Mode, in which you set up forms to use in the verification step
- The Verifier Test Mode, in which you can dynamically simulate verification.

Setting up the verification is easy. The Verifier user interface for classification verification is so simple that the same form can be used all the time – nothing needs to be defined there. What you need to create are the forms that will be used for verification of data extraction or for manual indexing. This step can be simplified by using default forms, but it can also be a complex task if there are special requirements. In addition to the features offered via the user interface, you can implement custom verification routines in WinWrap Basic scripts.

In WebCenter Forms Recognition, verification steps take place when a batch of documents reaches a predefined state that is represented by an integer number. The state changes again when verification is complete. States are used to distinguish valid and invalid results. For more information about input and output states, consult these manuals:

- **Runtime Server User's Guide:** The settings of this module determine the input and output states in the entire Forms Recognition workflow.
- **Verifier User's Guide:** The settings of this module determine the input and output states of verification steps.

### 13.1 Creating Verification Forms

#### 13.1.1. Managing Verification Projects

WebCenter Forms Recognition Designer organizes verification forms in verification projects. It uses the following structure:

- Each Designer project can contain one verification project.

- Each verification project contains a set of forms that are assigned to document classes.
- Each document class has one form that will be used by default. The remaining forms are only used when a document that requires validation is in a certain pre-defined state.
- Each verification has up to four areas:
  - Form field area.
  - Current input area.
  - Document display area
  - User info area.
- A form field area can contain the following elements:
  - Form fields.
  - Labels.
  - Viewers.
  - Tables.
  - Buttons.
- A form field is a text field that can be specified as:
  - Text Field.
  - List Box.
  - Check Box.
  - Amount Field
  - Date Field.
- All elements of a verification project have properties that can be edited.

Designer provides several functions for administering verification projects.

### **Task Prerequisites**

The prerequisites for this task are:

- The program runs in Verifier Design Mode.
- Classes and fields are already defined.

### **Creating a New Verification Project**

Switching to this mode will create a new verification project and make it available.

An empty verification project does not contain any forms. If you create a new project, an existing project including its forms will be deleted when you confirm.

#### **13.1.2. Configuring Project Validation Properties**

In Designer, you can set validation properties

- At the project level

- At the form level
- At the field level.

The most general validation properties can be set at the project level, but you can partially override the project-level settings at the form and field level and make more specific settings there.

### Task Prerequisites

The prerequisites for this task are:

- The program is in Verifier Design Mode.
- Classes and fields are already defined.

### 13.1.3. Setting Project Validation Properties

To set the project validation properties:

- 1) In the class selection panel at the upper left edge of the window, right click the root node that represents the Designer project.
- 2) On the shortcut menu, select *Project Properties*. The *Validation Settings* dialog box is displayed.

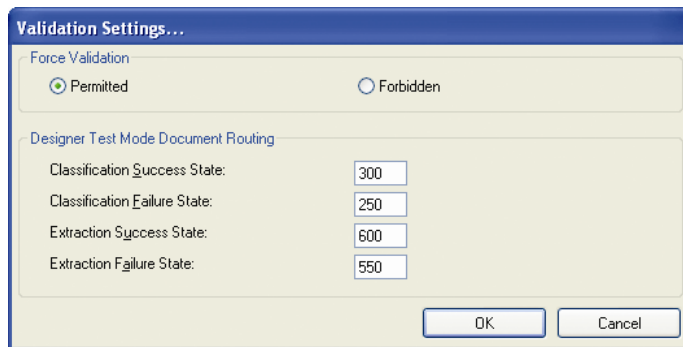


Figure 13-1: Project validation settings

- 3) Under *Force Validation*, select whether Verifier users can validate a field even if its content is invalid by definition.
  - *Permitted*: The user can force validation by pressing ENTER three times.
  - *Forbidden*: The user cannot force validation.
- 4) Under *Designer Test Mode Document Routing*: If you have script code associated with the `ScriptModule_RouteDocument` event, you can use the states entered in this area to test your scripting in Designer's Verifier Test Mode. These states will not be used in Verifier.

### 13.1.4. Verification Forms

There are four ways to create verification forms in Designer:

- You can create empty forms first and add the form elements then.

- You can create default forms that automatically contain a form field, a label, and a viewer for each field that will be visible in Verifier.
- You can create one form at a time.
- You can create forms for all classes in one step.

### Task Prerequisites

The prerequisites for these tasks are:

- The program is in Verifier Design Mode.
- Classes and fields are already defined.

### Creating an Empty Form

To create a single empty form:


- 1) In the class selection panel at the upper-left edge of the window, select a document class.
- 2) Do one of the following:
  - From the menu, select *Edit - DocClass - Insert Empty Form*.
  - From the shortcut menu of the document class, select *Insert Empty Form*.
  - Click on the drop-down arrow associated with the *Insert New Default Verification Form* button and select *Insert Empty Form*.

### Create Default Forms

To create a single default form:

- 1) In the class selection panel at the upper left edge of the window, select a document class.
- 2) Do one of the following:
  - From the menu, select *Edit - DocClass - Insert Default Form*.
  - On the shortcut menu of the document class, select *Insert Default Form*.
  - Click the drop-down arrow next to the *New Default Verification Form* button and select *Insert Default Form*.

To create a new verification project with one default form per document class, do one of the following:

- From the menu, select *Edit - Project - Insert Default Forms*.
- In the class selection panel at the upper-left edge of the window, right click the root node that represents the Designer project. From the shortcut menu, select *New Default Forms*.
- In the toolbar, click the *Insert New Default Verification Forms*  for all DocClass button.

*If you create a new project, an existing project including its forms will be deleted when you confirm.*

### 13.1.4.1. Displaying Forms

To view a table of all forms contained in the current project, either:

- On the menu, select *Edit - Verification Form - Show All*.
- Right click an entry in the form selection panel or on the panel's background if no entries are available. On the shortcut menu, select *Show All*.

To view a table of all forms associated with a certain class:

- In the class selection panel at the upper-left edge of the window, click the class concerned.

Form Name	Default	In Filter	Out Filter	Assigned to
Form_Invoices_1	Yes	-1	-1	Invoices
Form_Invoices_2		601	-1	Invoices
Form_Invoices_3		551	602	Invoices
Form_Invoices_4		602	-1	Invoices

Figure 13-2: Table of forms


The table of forms has the following columns:

- *Form Name*: Displays the name of the form. It is generated automatically and consists of the prefix Form, the name of the associated document class, and a consecutive number.
- *Default*: Displays “Yes” if the form is the default form of a document class, and nothing if this is not the case. Default forms are used for each input state, unless another form has been explicitly assigned to an input state.
- *In Filter*: Displays the input state assigned to a form, or -1 if no special input state has been selected.
- *Out Filter*: Displays the output state assigned to a form, or -1 if no special output state has been selected. Output states are relevant if partial validation of documents is allowed and useful in the context of exception handling.
- *Assigned to*: Displays the name of the associated document class.

To display a form, either:

- Select a document class in the upper-left panel to display its first form.
- Select a form from the table of forms.

To display a form in maximum size, either:

- From the menu, select *View - Verifier Design Mode - Maximize Verification Form* .
- In the toolbar, click *Maximize Verification Form View*.  
When the grid is turned off and the form is maximized, you get a good idea of how your form will look at runtime.

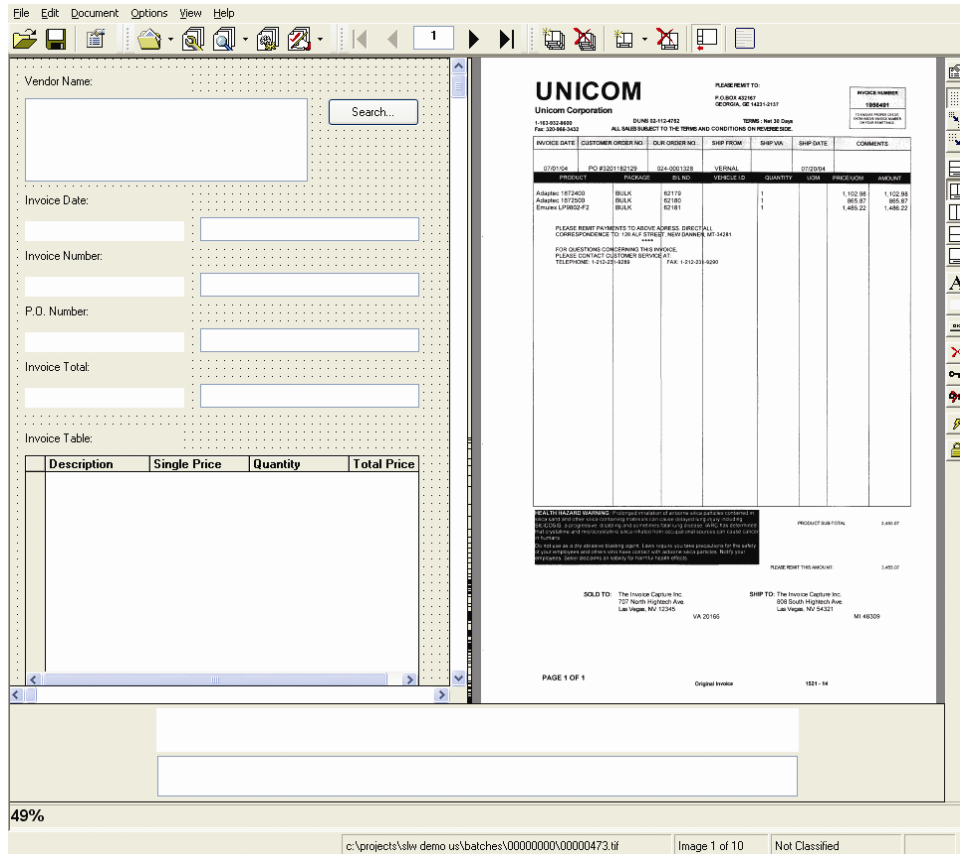




Figure 13-3: Maximized Form

### 13.1.4.2. Deleting Forms

To delete a single form:

- 1) In the form selection panel on the left side of the window, select a form.
- 2) Do one of the following:
  - From the menu, select *Edit - Verification Form - Delete Form*.
  - From the shortcut menu of the form, select *Delete Form*.
  - In the toolbar, click .

To delete all forms associated with a project, either:

- From the menu, select *Edit - Project - Delete All Forms*.
- In the Class Selection panel at the upper-left edge of the window, right click the root node that represents the Designer project. On the shortcut menu, select *Delete All Forms*.
- In the toolbar, click .

### 13.1.5. Configuring Form Validation Properties


#### Task Prerequisites

The prerequisites for this task are:

- The program is in Verifier Design Mode.
- At least one form is created.

#### Setting Form Validation Properties

To set the form validation properties:

- 1) Display a form and make sure that no form elements are selected.
- 2) Do one of the following:
  - In the vertical toolbar at the right edge of the window, click *Show selected object properties* .
  - Right click the form's background to display a shortcut menu and select *Properties*. The *Properties* dialog box is displayed.

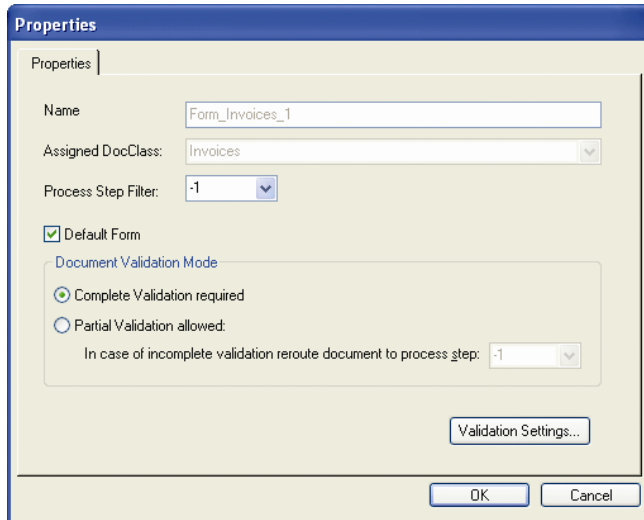


Figure 13-4: Property sheet of a form

- 3) Fill in the property sheet as follows:
  - *Name*: Displays the name of the form. This field is read-only.
  - *Assigned DocClass*: Displays the name of the associated document class. This field is read-only.
  - *Process Step Filter*: To use this form only for documents with a given state, select one of the pre-defined states from the list, or type a value into the field. For default forms, keep the initial value -1.
  - *Default Form*: Check this box to use the form as the default form, i.e. in all verification steps without a special pre-defined input state.
  - *Document Validation Mode*: Select complete validation if only one user verifies an entire batch and partial validation of several users validate a batch in stages. If complete validation is required, the Verifier user must validate all the fields in



the documents of a batch before being able to release the batch. If partial validation is allowed, only the fields on the form must be validated. With partial validation, you can employ multiple forms, i.e. one for each stage. You can set an output state for partially validated documents in Designer which overrides the output state set in Verifier.

4) Click *Validation Settings* to establish the following settings:

- *Validation Engine*: Select Standard Validation Engine to enable validation. Validation will not occur on the form unless the engine is enabled.
- *Force Validation Mode*: Select whether forced validation is forbidden or permitted with this form. To force validation of a field, the Verifier user must press ENTER three times. Default inherits the project-level validation setting.
- *Available Templates*
- *General Validation Rules*

For more information about the Validation Editor, see section [SETTING UP THE VALIDATION](#).

### 13.1.6. Configuring the Verification Form Layout

Each verification form is made up of up to four areas:

- *Form field area*: This area contains all or a subset of fields that have been defined for a given class.
- *Current input area*: This area contains a magnified version of the current field.
- *Document display area*: This area shows the current document, possibly with candidates highlighted
- *User info area*: This area works like a status bar. Custom messages can be displayed here telling the user why a field is invalid and providing hints concerning the correction of invalid results.

Regarding the arrangement of these areas, you can select from among five pre-defined layouts.

#### Selecting a Layout

To select a layout:

- Click the corresponding button in the vertical toolbar on the right side of the window. The new layout is applied to the current form. The form field area and the document display are divided by a splitter. You can move the splitter to adjust the size of these areas. The status bar displays the percentage of the window occupied by the upper or the left part, respectively. Be careful when sizing the areas: The splitter is only available in Verifier Design Mode, but not in Verifier Test Mode and not in the Verifier application.






Button	Description
	Top: Document display area. Center: Form field area. Bottom: Current input area.
	Left: Form field area. Right: Document display area. Bottom: Current input area. This is the default layout.
	Left: Form field area. Right: Document display area.
	Top: Document display area. Bottom: Form field area.
	Top: Document display area. Bottom: Current input area.

Table 13-1: Form layout controls

### 13.1.7. Configuring Form Grids

To enable you to align form elements, the background of a form has a magnetic grid which can be turned on and off. In addition, the grid size can be adjusted. Grid settings are valid for all the forms in a project.

#### Task Prerequisites

The prerequisites for these tasks are:

- The program runs in Verifier Design Mode.
- A form is selected.

#### Enabling or Disabling a Grid

To enable or disable the grid:



- In the vertical toolbar on the right side of the window, click *Grid* in the toolbar.

#### Increasing or Reducing Distance

To increase or reduce the distance between grid lines:



- In the vertical toolbar on the right side of the window, click *shrink grid* or *enlarge grid*.

### 13.1.8. Creating Form Elements

A form has three main elements: A label, a viewer, and a form field. From a form field, you can select a text field or table field. Using a text field or table field, you can create check boxes or combo boxes. You can also add a button to a form to fire actions.

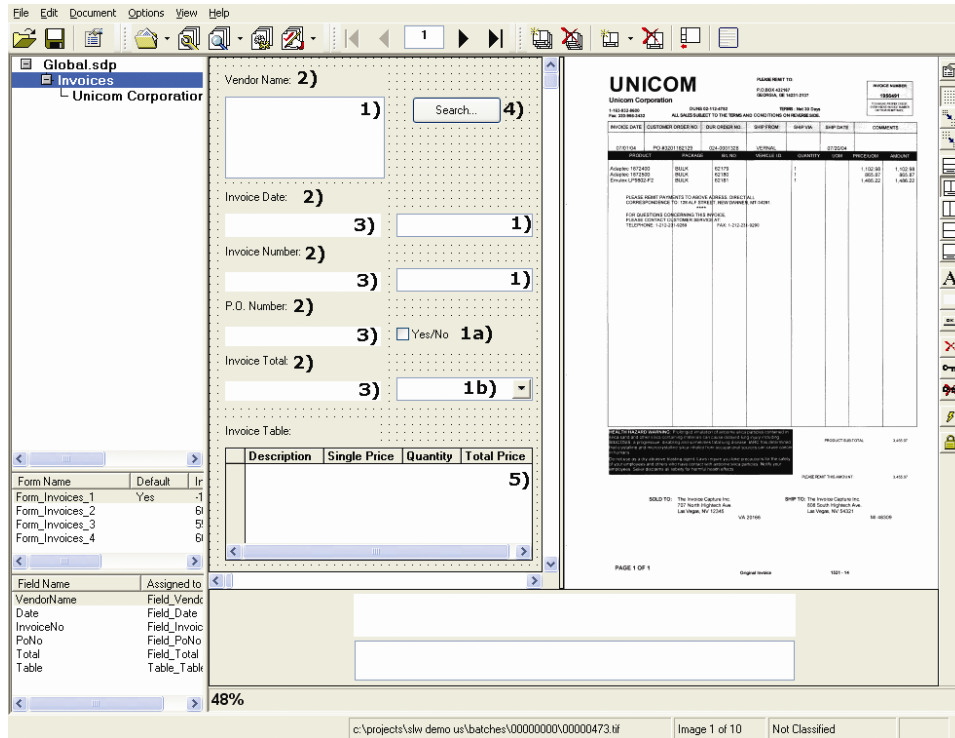


Figure 13-5: Elements of verification forms

Element	Description
1)	Form fields Controls that are used to display and edit extracted data and to enter data during manual indexing. You can use form fields to create Check boxes and Combo boxes.
1a)	Check boxes A toggle selection affecting data input applied when you have one of two values to choose from, such as on/off or yes/no. Check boxes are derived from Form Fields. You can set it up the caption with the text desired and select the default view.
1b)	Combo Boxes Contains a selection list to use when verifying an item on the document. Used during manual verification, this selection works with automatic completion.
2)	Labels Captions that help users to identify form fields and - if desired - also viewers and tables.
3)	Viewer Snippets of document areas, normally those that were extracted to fill fields or tables.
4)	Buttons Buttons that fire Actions for a new script event.
5)	Tables Relevant when table extraction is configured. The Verifier form supports multiple tables. However, even if you defined multiple tables, you can only display the first table on the verification form. You can display different tables on different forms.

Table 13-2: Elements of verification forms

WebCenter Forms Recognition Designer lets you add these elements to custom forms.

### Task Prerequisites

The prerequisites for these tasks are:

- The program is in Verifier Design Mode.
- The form you want to design is selected.

## 13.1.9. Creating and Modifying Form Fields

### 13.1.9.1. Creating Form Fields

To add a form field to a form:

Drag and drop a data field from the field selection panel at the bottom-left edge of the window to the target position on the form. By default, the field type is a text field. To define a special field type such as a combo box or check box, select the properties from the shortcut menu or click the button on the toolbar.

### 13.1.9.2. Modifying Form Fields


To improve Verification results, you will probably want to change the way Verifier users interact with Verification forms. This is done by accessing the Validation Editor discussed in depth in [SETTING UP THE VALIDATION](#).

### Converting a Text Field to Another Field Type

With the Validation Editor, you can change any field you created to a List Box, Amount Field, Date Field, Check Box, or back to a Text Field. You can also set Validation properties for each of these field types.

It's also possible to change a text field to a table field, but the procedure is somewhat different than the one discussed here for the other field types. To learn more about working with table fields, see section [SETTING UP THE FIELDS](#).


To modify field types:

- 1) Create a new field or select an empty field.
- 2) In the vertical toolbar at the right edge of the window, click *Show selected object properties* .
- 3) The *Properties* dialog box displays. Click *Validation settings*. For details about the property dialog box, see sections [SETTING FIELD VALIDATION PROPERTIES](#) and [WORKING WITH VALIDATION LEVELS](#).
- 4) In the *Validation Type* dropdown box, select a field type. The settings available to you will change depending on your selection. (See [FIGURE 13-8](#).)
- 5) Select either *Do not allow values that are not on the list*, or *Search and show nearest value automatically*.
- 6) Add items to the list that would pertain to the selected field. You can add and remove items at any time, or clear the list and start over.
- 7) Select *Sort strings alphabetically* for easy selection when verifying.

- 8) Click *OK*. The combo box appears next to the selected field. Select the arrow button next to the box to view the selections.

## Creating Check Boxes

Check Boxes are a type of form field. To create a check box:

- 1) Select an empty field or create a new empty field (See section [CREATING FORM ELEMENTS.](#))
- 2) In the vertical toolbar at the right edge of the window, click *Show selected object properties* .

On the *Properties* dialog box displays, click *Validation Settings*. This opens the Validation editor. For more details about the property dialog, see section [SETTING FIELD VALIDATION PROPERTIES.](#)

- 3) Under *Field Type*, select *Check Box* and press *Details*. The *Check Box Field Properties* box appears.(See [FIGURE 13-6](#))
- 4) Enter a name for the Check Box Caption. Example: Yes/No.
- 5) Enter the Checked Value, which is the text that appears when the check box is selected. If you want, you can set this as the default.
- 6) Enter the Unchecked Value, which is the text that appears when the check box is not selected. If you want, you can set this as the default.
- 7) Click *OK*. The check box appears next to the selected field.

## Creating Labels

To add a label to a form:



- 1) In the vertical toolbar on the right side of the window, click *Add new label element*. The mouse pointer changes.
- 2) Click inside the form and drag the frame to the desired size.
- 3) On the *Change Captions* dialog box, type a caption for the label.

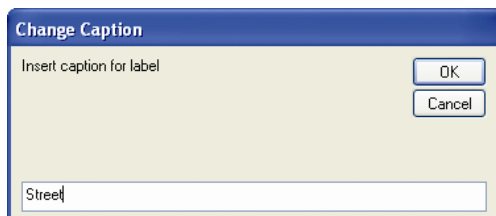


Figure 13-6: The Label's Change Caption dialog box

- 4) Enter a name in the box. Click *OK*.



To change the caption, select *Change Caption* from the shortcut menu. A label can only be deleted from the form by using the *Delete* button from the vertical toolbar.

## Creating Viewers

To add a viewer to a form:



- 1) In the vertical toolbar on the right side of the window, click *Add new viewer element*.

A blue frame with white background appears at the upper edge of the form.

- 2) Drag the viewer to its target position.
- 3) Assign a field to be viewed, by selecting the properties dialog from the shortcut menu. See section [CONFIGURING THE VIEWER PROPERTIES](#).

## Creating Buttons

To add a button to a form:



- 1) In the vertical toolbar on the right side of the window, click *Add New Button*.
- 2) Click inside the form and drag the blue frame to the desired size.
- 3) To name the button, right click the button. The *Change Captions* dialog box appears

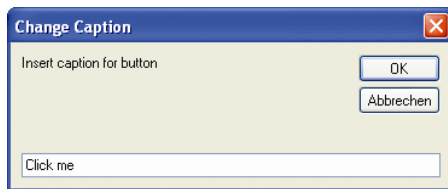


Figure 13-7: The Change Caption dialog box

- 4) Enter a name in the box. Click *OK*.

See section [SHORTCUT MENU CONFIGURING ACTIONS](#) for information about configuring actions that are triggered by buttons.

## Viewing Tables

You can only have tables in a form if table extraction is configured for the corresponding document class. A form can have more than one table. However, even if you defined multiple tables, you can only display the first table on the verification form. You can display different tables on different forms.

If your tables do not need verification, you can make your columns invisible. The invisible columns must be valid in order to validate the entire table.

### 13.1.10. Editing Form Elements

WebCenter Forms Recognition Designer provides some commands for editing form elements.


#### Task Prerequisites

The prerequisites for these tasks are:

- The program is in Verifier Design Mode.
- The form containing the elements is selected.

#### Supported Operations

The following operations are supported for all elements:

- You can move a selected element using the mouse.
- You can resize a selected element by dragging the handles at each corner or by setting the size and position in the property box.
- You can delete a selected element using one of the following methods:
  - Press the *DELETE* key.
  - Click  in the vertical toolbar

### 13.1.11. Setting Field Validation Properties

#### Task Prerequisites

The prerequisites for this task are:

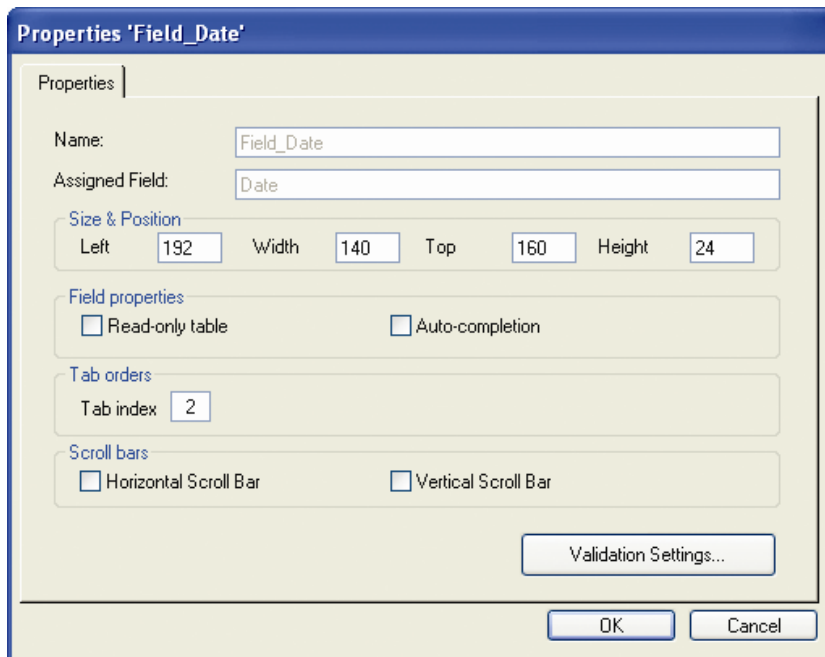
- The program is in Verifier Design Mode.
- At least one form with one form field is created.

#### Setting the Properties

To set the field validation properties:

- 1) Select a form field.
- 2) In the vertical toolbar at the right edge of the window, click *Show selected object properties*.

The *Properties* dialog box is displayed.



The screenshot shows the 'Properties' dialog box for a form field named 'Field\_Date'. The dialog has a title bar 'Properties 'Field\_Date'' and a 'Properties' tab. It contains several sections:

- Name:** Field\_Date
- Assigned Field:** Date
- Size & Position:** Left (192), Width (140), Top (160), Height (24)
- Field properties:**
  - Read-only table
  - Auto-completion
- Tab orders:** Tab index (2)
- Scroll bars:**
  - Horizontal Scroll Bar
  - Vertical Scroll Bar

At the bottom right, there is a 'Validation Settings...' button, and at the very bottom, 'OK' and 'Cancel' buttons.

Figure 13-8: Property sheet of a form field

- 3) Fill in the property sheet as follows:
  - *Name*: Displays the name of the form field. This property is read-only.

- *Assigned Field*: Displays the name of the associated data field. This property is read-only.
- *Size and Position*: Affects form layout. Using numerical values rather than drag and drop enables you to design your form with greater precision. But in most cases, the default values – derived from the location you assigned to the field when you dragged it onto the form – are fine.
- *Read Only*: Select Read-only to show static and dynamic complementary information to aid in manual verification.
- *Auto-completion* helps to speed up typing. When you start to type, auto-text completes the word, suggesting the best matching item(s) among all of the words or candidates available after OCR and format analysis phases.
- *Tab index* controls which Verification Form field gains focus next when the TAB key is pressed. Fields are visited from low to high tab index. If the specified index is already in use by another verification field, its tab index and, when required, subsequent fields, will be incremented by one automatically.
- *Validation Settings* button to make further enhancements. For more information about these settings, see section [SETTING UP THE VALIDATION](#).

### 13.1.12. Editing Text Fields

WebCenter Forms Recognition Designer includes automated features for editing text fields that can help users enter and correct text faster. You can use automatic character entry when the auto-completion option is selected in the form field Properties dialog box to edit text fields and cells. Other options for character changes include Multi-line fields, combo boxes, and check boxes. You can also insert and replace text in cells and fields, either in single words or blocks of text, using drag and drop or by double clicking on selected text.

#### Example of Auto-Completion

Multi-line fields are necessary for address analysis, but can also be useful in other cases. A multi-line field enables line wrap and displays a vertical scroll bar, if required. To add a new line to a multi-line field, press CTRL+ENTER.

A list box includes predefined strings related to the verification document. To aid in verification, you can select from the list of strings.



Figure 13-9: Example of a list box

The Check Box provides an either/or option that toggles table field options on and off. For example, a “Yes/No” check box checked for Yes would bring up field options related to the verification, and unchecked for No would hide them.



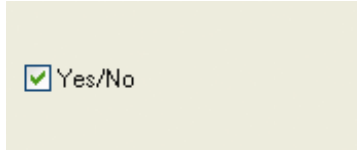


Figure 13-10: Example of a checkbox

Do not enter formatted text for auto-completion. Auto-complete does not work on formatted text and characters incorrectly read by OCR.

### Auto-Completion

Auto-completion helps to speed up data entry. When you start to type, auto-text completes the word, suggesting the best matching items among all of the words or candidates available after OCR and format analysis. For example, you can type the first two characters of a 20-character invoice. The auto-text feature finds the best matched candidate suggested by the Format Analysis engine and places it in that field. The auto-selected text is also highlighted in the original document. Select whether a single-line or a multi-line text field should be displayed. To override auto-completion, continue typing.

Auto-completion feature for a header field is supposed to automatically select the best candidate from the available ones. However, the viewer will be updated only if the candidate appears once in the document; otherwise the viewer will be blank when auto-text completes the word for the field. This function only works with *Highlight Candidates* mode.

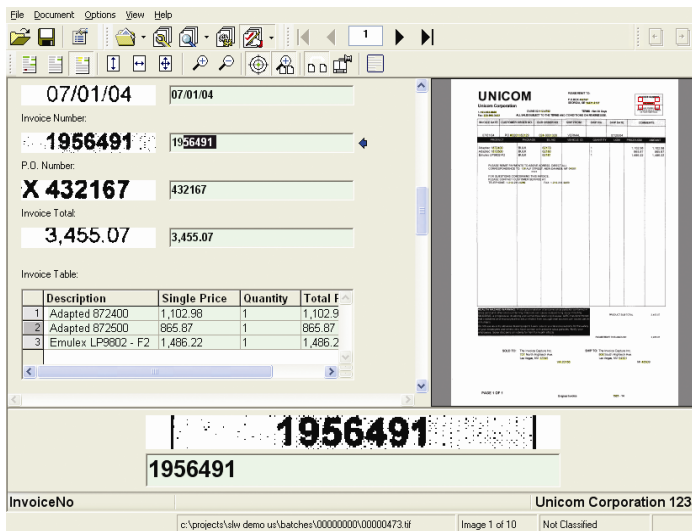


Figure 13-11: Example of auto-completion

### Inserting Words In Fields

To speed up verification, you can insert words to replace or append text. The method for inserting words depends on the availability of candidates. A candidate is one that matches the learned words for that field. It will show up in green when you select it after selecting the field. Non candidates will display in orange when selected. You can insert words in fields or table cells. You can append or insert words and using the mouse to append or replace the field. Make sure to select the *Highlight Candidates* and *Fields* button in the toolbar editing field candidates.

### Words with Candidates

If the word is a candidate for a field, you can append to or replace a word in a field box. A candidate is one that matches the learned selections for the field. It will show up in green when you select it after selecting the field. The append feature takes the current word left of the candidate and appends the field text. It places the text in the best location, either right or left of the word, and places the field by text or location of the word. Or, you can replace text. For example, a blank candidate might be replaced by “285.98.”

- To append text with the new text, click the desired word next to the text that you want to append (this text will appear in green if it is a candidate). A box appears around the word. Double click the box, or right click in the document and select *Append Field Text by Word*.
- To replace a word, click the desired word. A box will appear around the word. Double click the desired candidate, drag and drop the word to the field, or right click in the document and select *Map Candidate* from the shortcut menu to replace it.

*You can insert only one candidate per field per document verification session.*

Check to make sure that this word fits the format analysis rules defined for that particular field. If not, the word is highlighted in orange. In this case, it would not be a good candidate for the field.

### **Words Without Candidates**

Even if the word does not belong to any candidates for the field, you can append or replace a word with a new one. Appending places the text in the best location, either right or left of the word, by text or location of the word. Or, you can replace the field text and location by the text and location of a word. A word that does not belong to any candidates for that field will display in orange when selected. For example, a field named “sales total” might be replaced by “invoice total.”

- To append text with the new text, with the mouse drag a box around the desired word. Double click the desired word in the box, or right click in the document and select *Append Field Text by Word*.
- To replace text, select the desired word with the mouse. A box will appear around the word. Double click with the left mouse button, or select *Replace Field Text by Word* in the shortcut menu to replace it.

*You can insert only one candidate per field, per document verification session.*

Check to make sure that this word fits the format analysis rules defined for that particular field. If not, the word is highlighted in orange to help distinguish it. In this case, it would not be a good candidate for the field.

### **Inserting Blocks of Text**

Inserting large blocks of text with minimal mouse movement is helpful when you have multiple word data verification elements, for fields such as address information or for cell descriptions. Before you can insert blocks of text, you must first select the settings in the Workflow dialog box in Verifier to immediately copy information. See the ***Verifier User’s Guide***.

To insert large blocks of text:

- 1) Click and drag the left mouse button over the desired text in the image viewer.
- 2) Release the mouse button. A rectangle appears surrounding the text. Adjust the rectangle by selecting the nodes at any corner, if necessary.

- 3) Drag and drop the rectangle to the desired field or table cell. A copy of the rectangle will appear over the field (or table cell).

**Or**

Double click the rectangle. The text in the rectangle replaces the text in the field (or table cell).

*Note:* You can move or resize this rectangle by clicking in the area in the image viewer. When the rectangle appears, select the nodes to resize it, or drag it using the drag and drop method described above.

### **13.1.13. Configuring the Viewer Properties**

Viewers can be used flexibly:

By default, each viewer is assigned to exactly one data field.

- Several viewers can be assigned to one data field, providing different views on the same zone of the document.
- One viewer can be used to visualize several fields, depending on the field selection at runtime.
- The viewer assignment and geometry are configured in the viewer's property sheet.


#### **Task Prerequisites**

The prerequisites for this task are:

- The program is in Verifier Design Mode.
- At least one viewer is created.

#### **Setting Viewer Content Properties**

To set the viewer content properties:

- 1) Select the viewer concerned.
- 2) Do one of the following:
  - Right click to display a shortcut menu and select *Properties*.
  - In the vertical toolbar at the right edge of the window, click *Show selected object properties* .

The *Properties* dialog box is displayed.

Figure 13-12: Viewer Property sheet dialog box

Complete the property sheet as follows:

- **Name:** Displays the name of the viewer. This property is read-only.
- **Assigned Field:** Displays the name of the associated data field and lets you change the selection. This field can also be empty if the content displayed should depend on the selected document only or on the field that is selected by the user at runtime.
- **View Mode** -
  - **Whole Image:** If this option is selected, the viewer covers the entire page of the current document. You need to specify the page number in the View Mode Zone - Page field.
  - **Fixed Zone:** If this option is selected, the viewer covers a fixed area of a page of the current document. You need to specify the page number in the View Mode Zone - Page field. In addition, you need to specify the distance of the viewer area from the top and left edge of the page, and the width and height of the area. These parameters are expressed as percentages of the height or width, respectively, of the entire page.
  - **Field:** If this option is selected, the viewer covers the area of the document that contains the words used to fill the associated field. This works no matter whether the field assignment is set at design time or at runtime.
  - **Field in Zone:** If this option is selected, the viewer covers a rectangle around the area of the document that contains the words used to fill the associated field. This works no matter whether the field assignment is set at design time or at runtime. In the View Mode Zone group, the dimensions of the rectangle are expressed as percentages of the height or width, respectively, of the entire page.
  - **Field in Line:** If this option is selected, the viewer covers the lines around the area of the document that contains the words used to fill the associated field. This works no matter whether the field assignment was set at design time or at runtime.
- **View Mode Zone:** Please see View Mode-Fixed Zone

- *Border*: Where this makes sense, you can specify that a white border is displayed around the area covered by the viewer. The dimensions of the border are expressed as percentages of the height or width, respectively, of the entire area.
- *Size and Position*: Set size of the viewer and the position of the viewer on the form.

What you see in the viewer depends not only on the viewer properties, but also on the viewer dimensions. The viewer always covers the area specified in the property sheet, but actually shows a larger section of the page that matches the viewer dimensions. This way, the document areas are possibly shown as minimized way, but they are never distorted. You should obtain the best results if the viewer dimensions closely resemble the geometry of the area that is to be shown.

### 13.1.14. Configuring Table Properties

When table extraction is configured, the table validation properties need to be set. In table validation, the single table cells behave like fields in regular data extraction.

#### Task Prerequisites

The prerequisites for this task are:

- The program is in Verifier Design Mode.
- A form is selected that has a table field.
- The table has been enabled.

#### Setting Table Validation Properties

To set the table validation properties:

- 1) Select the table.
- 2) Do one of the following:
  - Right click to display a shortcut menu and select *Properties*.
  - In the vertical toolbar at the right edge of the window, click *Show selected object properties*.



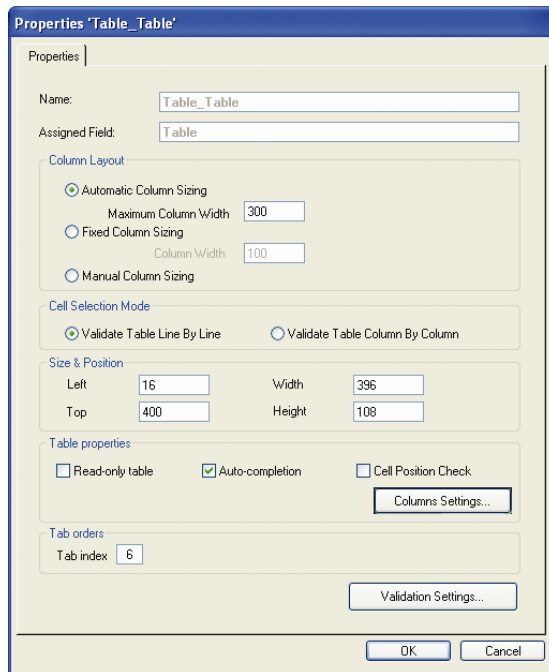


Figure 13-13: Table Property sheet dialog box

Fill in the property sheet as follows:

- **Name:** Displays the name of the table. This field is read-only and currently always empty.
- **Assigned Field:** Displays the name of the associated table field. Currently, only one table per document class can be configured.
- **Column Layout**
  - **Automatic ...:** Column widths are optimized automatically from the column content and restricted by a maximum value which is given in pixels. The user can change the column width at runtime, but after document change, the automatic setting will be used again.
  - **Fixed ...:** Columns have a fixed width given in pixels which is the same for all columns but the last one. The user can change the column width at runtime, but after document change, the fixed setting will be used again.
  - **Manual ...:** The user can set the column width at runtime.
- **Cell Selection Mode:** Determines the order in which invalid cells receive focus when the user navigates through the table at runtime. If the order is line by line, the focus moves from left to right up to the end of a row, and then to the next row. If the order is column by column, the focus moves from top to bottom up to the end of a column, and then to the next column.
- **Table Properties:** Options include Read Only, Auto-completion, Cell position Check, and Columns Settings. *Read Only* sets a lock on the table against edits. *Auto-completion* enables faster verification of information. Auto completion methods might include check boxes or combo boxes, as well as automatic word completion when entering text.

- Column settings include individual column settings, rather than settings for the entire table. For example, you can select a single column to be read only, rather than the whole table.

### 13.1.14.1. Setting Column Properties

To set the table validation properties:

- 1) Select the table.
- 2) Click *Column Settings...*

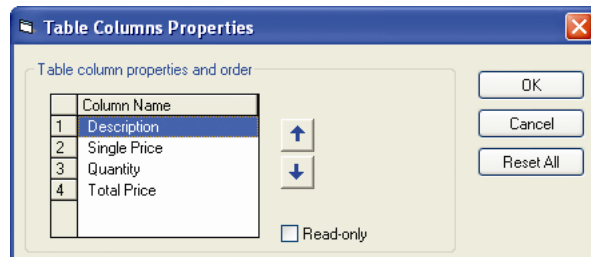


Figure 13-14: Changing of table column display order in Designer application.

- 3) Select the column name to change and use the blue “up” and “down” arrow buttons to set the desired order.

### 13.1.15. Configuring Smart Indexing

Smart indexing is a method to fill fields using entries from a database table. It can be used in the automatic extraction step as well as during verification and manual indexing. In smart indexing, two types of fields are distinguished:

- Smart index field.  
The information related to this field must be available both in the documents that are to be processed, and in the database table. Often, a unique identifier such as “Invoice #” or “Customer #” is used for this purpose.
- Index lookup fields.  
The information related to these fields must only be available in the database table, but not in the documents. This is because these fields are filled using a database lookup with the value of the smart index field as parameter. The lookup returns the table rows that will be used to fill the lookup fields. For more information, see section [DATABASE SUPPORT FOR SMART INDEXING](#).

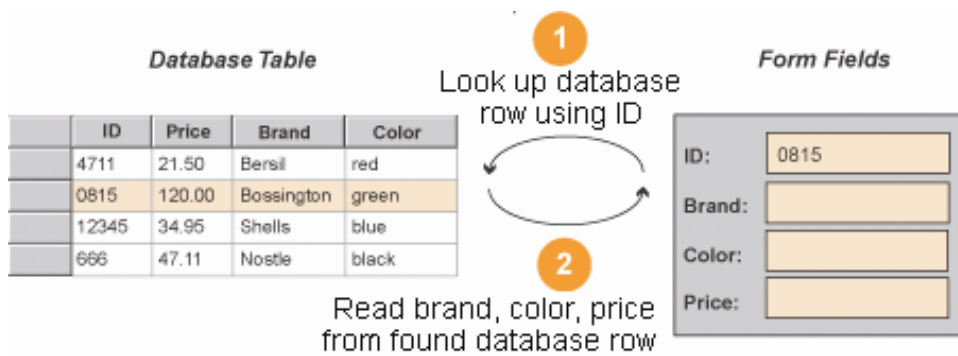


Figure 13-15: Smart indexing method

If used manually, the form for smart indexing should have the smart index field above the lookup fields. Otherwise the form's tab order will disturb the workflow.

### Task Prerequisites

The prerequisites for this task are:

- The program is in Verifier Design Mode.
- A form is selected that has at least two form fields.
- The database exists that contains a table with corresponding columns.
- There is an OLE DB provider for the database system you use.

### Configuring Smart Indexing

To configure smart indexing:

- 1) Select the form field that will be used as a smart index field.

In the vertical toolbar at the right edge of the window, click *Define Smart Indexing*.



The *Smart indexing definition* dialog box is displayed. It shows the available fields. The smart index field is highlighted in yellow and marked with a key icon.



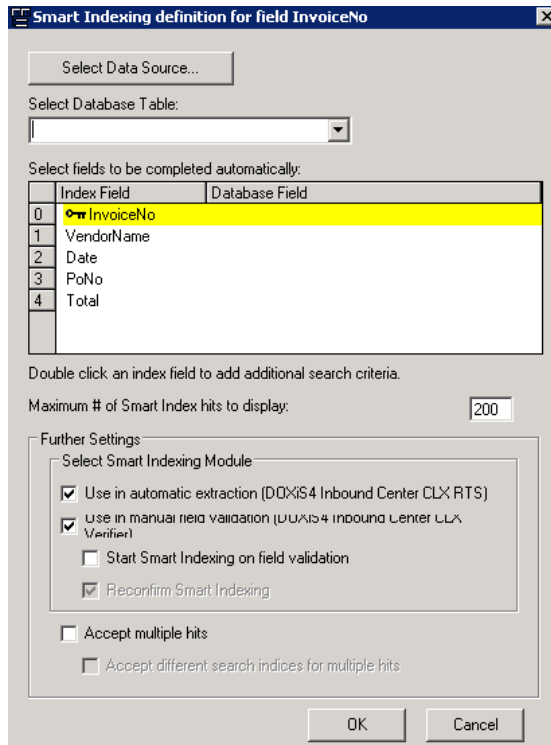


Figure 13-16: Smart indexing dialog box

- 2) Click on *Select Data Source*. The *Data Link Properties* dialog box is displayed.

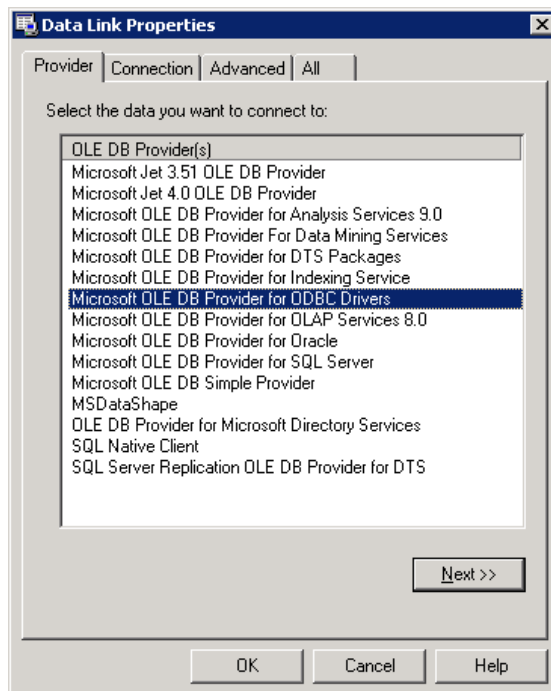


Figure 13-17: Data link properties - Provider

- 3) In the *Provider* tab, select the appropriate OLE DB provider for the type of data you want to access.
- 4) Click *Next*.

- 5) Use the *Connection* tab to specify how to connect to your data. The *Connection* tab is provider-specific and displays only the connection properties required by the OLE DB Provider of your choice. For more information, click *Help*.

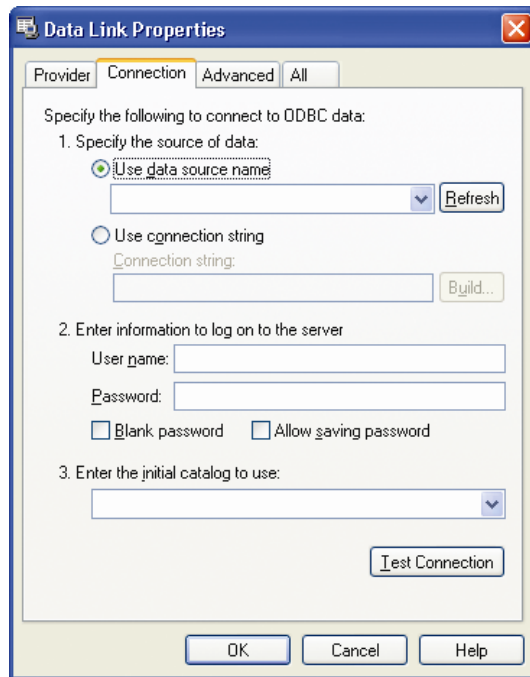


Figure 13-18: Data link properties - Connection

- 6) Click on *OK*. The *Smart indexing definition* dialog box is displayed again.
- 7) Select a database table from the *Select database table* list box.
- 8) Click on a cell in the *Database Field* column. A list is displayed, containing the names of all columns of the selected table. Map columns to
- the smart index field
  - all lookup fields.
- Smart indexing will fill each field that is assigned to a database column.
- 9) Optionally, double click each additional field that you want to use as index field. The corresponding fields are marked with key icons. Note that search criteria are evaluated using AND joins, i.e. there must be a matching row for both index values.
- 10) A database lookup using one index term may return multiple results. In interactive mode, a dialog box opens where the user can select the appropriate record. To limit the number of results, type an entry into the *Maximum # of SmartIndex hits...* text box, or accept the default.
- 11) Under *Further Settings*, specify where and when smart indexing should be performed.
- *Use in automatic extraction*: Smart indexing is performed during the automatic extraction step by WebCenter Forms Recognition Runtime.
  - *Use in manual field validation*: Smart indexing is available during the extraction verification or the manual indexing step in Verifier.

- *Start smart indexing ...*: If this option is checked, the lookup is performed on validation, i.e. when the user presses *ENTER* in the index field. Otherwise the user needs to press ALT+F12 to trigger smart indexing.
- *Reconfirm smart indexing*: If this option is checked, the user has the opportunity to check and correct the values retrieved from the database (recommended). Otherwise the retrieved values are validated automatically.
- *Accept multiple hits*: If this option is checked, a valid lookup result may consist of multiple values.
- *Accept different ...*: If this option is checked, a valid lookup result with multiple values may have been obtained using different index fields.


### 13.1.15.1. Viewing Smart Indexing Fields

When editing the form, the *Smart Index* field will appear in yellow with the word “SmartIndex.”



Figure 13-19: Example of smart indexing of the Invoice number on the form.

To delete the smart indexing definition:

- 1) Select the form field that is used as smart index field.
- 2) In the vertical toolbar at the right edge of the window, click *Delete Smart Index* . The smart index definition will be deleted instantly.

### 13.1.16. Database Support for Smart Indexing

In smart indexing, you have two types of fields:

- Index Field
- Index Lookup Fields

#### 13.1.16.1. Index Field

The information related to this field must be available both in the documents that are to be processed, and in the database table. Often, a unique identifier such as “Invoice #” or “Customer #” is used for this purpose.

#### Index Lookup Fields

The information related to these fields must only be available in the database table, not in the documents. This is because these fields are filled using a database lookup with the value of the smart index field as parameter. The lookup returns the table rows that will be used to fill the lookup fields.

Although Designer and Verifier support the use of an RDBMS with Smart Indexing, a problem arises when using a stand-alone database package such as Microsoft Access. When database access for Smart Indexing is configured, an ODBC connection to the data source must be defined. The definition is stored in a configuration file with a .dsn extension.

However, ODBC used with Microsoft Access requires a static drive mapping (instead of a UNC path) to the data source.

For example, if the designated data source for Smart Indexing is located at \\DataSourceMachine\ShareName\DataSource.mdb, ODBC used with Microsoft Access requires that a drive letter (such as F:) be mapped to the data source. ODBC then stores the drive mapping in a .dsn configuration file. This seems to work, as long as the WebCenter Forms Recognition project containing the Smart Indexing that references the data source is used on the machine on which the project was created.

However, when the project is opened on a machine other than the one on which it was created and an attempt is made to use the Smart Indexing feature, the application generates an “Object is closed” error. The work around for this error is to map the same drive letter (F:), stored in the .dsn configuration file that was created when the ODBC connection was defined, to the data source on the machine on which the project is being used. Although it is possible to synchronize drive mapping in some environments it not conceivable in all network environments. This is especially true for those organizations that rely heavily on legacy applications requiring static drive mapping.

As a result of this ODBC constraint, WebCenter Forms Recognition does not support the use of Smart Indexing with stand-alone databases such as Microsoft Access.

### **13.1.17. Correcting Table Fields**

If the field is a table field, you can correct invalid cells as if they were text fields.

#### **13.1.17.1. Using Auto-complete**

This feature allows to automatically select the best candidate from the available words, which works within *Highlight Candidates* mode. Auto completion works in table cells as well as with text fields. When you type two or more characters, auto complete will suggest a word or phrase for that cell. The candidate appears in green if the field is valid and orange if the field is invalid.

#### **13.1.17.2. Inserting Words in Table Cells**

You can insert single words or append existing text in table cells.

##### **Words That Are Candidates for Cells**

If the word belongs in a cell area, you can append or replace a word in a cell. The append feature takes the current word behind the candidate and appends it to the cell text. It places the text in the best location, either right or left of the word, and cell location by text or location of the word. The word belonging to a cell area will highlight in green when selected. Or, you can replace text.

To append text with the new text, you would double click the desired word, or right click in the image viewer and select Append Cell Text by Word. If you have candidates, you can double click the desired candidate to replace it, or right click in the document, and then select “Select Cell” from the shortcut menu.

In the search region, word candidates are all words that are not covered (by location) by other table cells and that have the same beginnings as the whole text of the cell.

##### **Words That Are Not Candidates for a Cell**

If the word does not belong to cell areas, it will display in orange when selected. Even if it is not a candidate, you can append or replace the word. Appending places the text in the best location, either right or left of the word, by text or location of the word. For example, a cell named “C2658” might be appended by “number.” Or, you can replace the cell text and location by the text and location of a word. To append text with the new text, you would double click the desired word, or right click in the image viewer and select Append Cell Text by Word. To replace text, select the word, hold down CTRL and click the left mouse button, or select Replace Cell Text by Word in the shortcut menu.

### Correcting Table Structure

You may need to correct the table structure as well. Table rows and columns exhibit shortcut menus that give you options to modify the table structure. To invoke them, right click the row or column label. The available commands are summarized below.

Shortcut menu	Command	Description
Column	Unmap	Clears all data for the selected verification column and turns the state of the corresponding column of the recognized table back to “unmapped.” To view an unmapped column, double click the table header in the verification form. All unmapped columns are highlighted in red.
	Map	Adds the column selected from the shortcut menu. Or, you can right click an unmapped column to map it to a column in the verification form.
	Swap	Exchanges the position of the current column and the one selected from the drop-down menu.
Row	Insert	Inserts an empty row above the current one.
	Delete	Deletes the current row.
	Duplicate	Duplicates the current row.
	Append	Appends an empty row at the bottom of the table.



Table 13-3 Table shortcut menus:

### 13.1.18. Shortcut Menu Configuring Actions

Actions are triggers for scripted events that you can create to perform project-specific tasks. For example, you can trigger any allowed modification of WebCenter Forms Recognition-related objects, or you can set an action to open an application or invoke third-party COM interface methods.

You can use buttons or keyboard shortcuts to activate an action.

To configure an action:

- 1) Select Add new *button element*  and create a button on your form.
- 2) Name the button to reflect the desired action by right clicking on the button. The *Change Caption* box appears.
- 3) Click Show *action list* .

The Define Actions dialog box appears.

- 4) Press *Insert* to add an action.
- 5) Press the *Action* label and enter a name for your action.
- 6) Press *Description* and enter a description of your action.
- 7) Press *Button* and select the name of the button that you want to link to the action.
- 8) Press *Accelerator*.

The *Accelerator Properties* box appears.

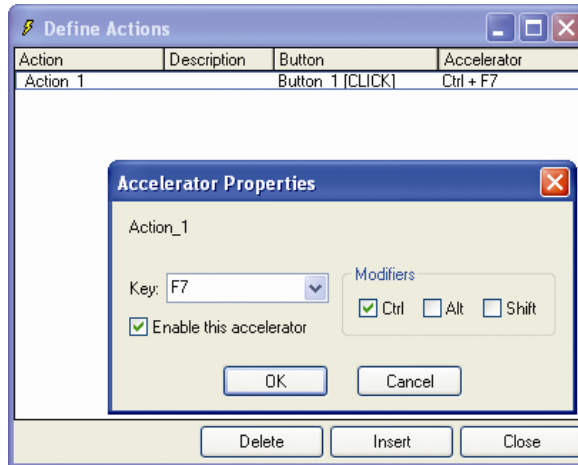


Figure 13-20: Accelerator Properties in Define Actions.

- 9) Select the type of keyboard combination for the accelerator, if desired.
- 10) Select the Script Editor by pressing the script button in the main menu.
- 11) Compose an event handler for a Document On-Action event.

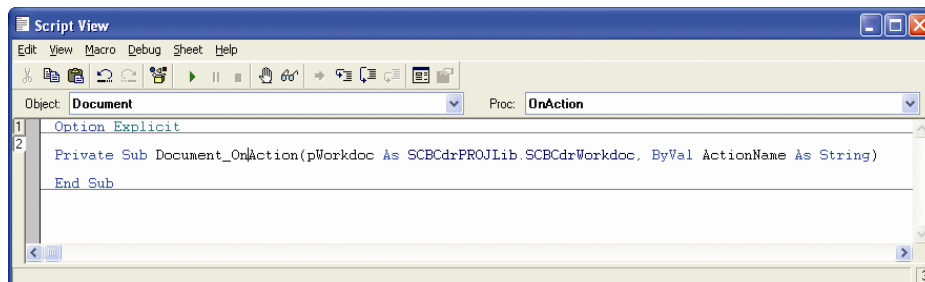


Figure 13-21: Example of a script for an action button

- 12) Switch to Verifier Test Mode and test the action. You should get a message box that confirms the activation.

### 13.1.19. Changing of Colors & Fonts for Elements of Verification Forms

WebCenter Forms Recognition 10.1.3.5.0 or higher supports script methods to dynamically or statically adjust fonts, colors, and background colors for verification forms and their verification elements. Please refer to the Scripting Guide documentation for more details.

## 13.2 Testing the Verification

The Verifier Test Mode provides a complete environment for the simulation of classification and extraction verification. This includes the corresponding state evaluations and transitions. However, changes of the Workdoc are not saved persistently. They are kept in memory only and discarded when the document input changes.

### 13.2.1. Verifier Test Mode User Interface

The user interface that is displayed depends on your current settings:

- It can be determined dynamically, using the document class and state. This leads to one of the following results:
  - The classification user interface is displayed.
  - The extraction/indexing user interface is displayed using the form corresponding to the current document's state.
  - There is no matching form, i.e. an empty user interface will be displayed.
- If *Force document...* is pressed, the selected form is displayed and used for extraction, independent from the document properties.

### 13.2.2. Verifier Test Mode Color Coding

In Verifier Test Mode, color coding is used to support the quality assurance process.

The appearance of the field area tells us something about the validity of the extracted data:

- A green background indicates a valid extraction result.
- A red background indicates an invalid extraction result.
- A question mark indicates illegible characters.
- A red character indicates rejects, i.e. WebCenter Forms Recognition is in doubt whether the character was read correctly.

Similar information can be obtained from the document area:

- A green highlight indicates a valid extraction result.
- A red highlight indicates an invalid extraction result.
- A yellow highlight indicates a candidate.

Depending on the current selection in the field area, document highlighting can be applied to the following items:

- Document areas associated with text fields.
- Document areas associated with entire tables.
- Document areas associated with table columns.
- Document areas associated with table rows.
- Document areas associated with table cells.

### 13.2.3. Verifier Test Mode Icons

In the field area, the following icons are used to indicate the nature of the field:




Icon	Description
	Indicates the currently selected field.
	Indicates a smart index field that can be filled by a database lookup. This icon is visible only when a smart index field is selected.
	Indicates a smart index field that can be used to start a lookup. This icon is visible no matter whether a smart index field is selected or not.

Table 13-4: Icons in Verifier Test Mode

### 13.2.4. Verifier Test Mode Toolbar

The toolbar of the Verifier Test Mode provides some additional commands that help you to test the verification procedure. Use one of the following highlighting options to highlight fields and candidates:







Button	Description
	Highlights all areas on the current document that have been used to fill text fields. If the result is valid, the area is highlighted in green. If the result is invalid, the area is highlighted in red.
	Highlights only the area on the current document that was used to fill the current text field. If the extraction result is valid, the area is highlighted in green. If the extraction result is invalid, the area is highlighted in red.
	Highlights the area on the current document that was used to fill the current text field. If the extraction result is valid, the area is highlighted in green. If the extraction result is invalid, the area is highlighted in red. In addition, all other candidates for the current field are highlighted in yellow.

Table 13-5: Highlighting options in Verifier Test Mode

Further highlighting options are available via the menu. These are the same highlighting options as in document selection, definition and Train Mode.

Use the following buttons to control form and document display during the verification procedure:

Button	Description
	If this button appears pressed, the program always keeps the focus on the document area that is associated with the currently selected field. If required, the document section displayed in the viewer changes accordingly on field change.
	If this button appears pressed, the next documents are displayed with the currently selected magnification. Otherwise the default magnification is used after document change.
	If this button appears pressed, the next document will be displayed when you validate the current one. Otherwise no document change will take place, unless triggered via the navigation bar. Note that, in contrast to the Verifier application, valid documents are not skipped in Verifier Test Mode.




Button	Description
	If this button appears pressed, the next document will be verified using the current form. Otherwise, document state and class assignment are used to determine the appropriate form on document change.

Table 13-6: Control options in Verifier Test Mode

### 13.2.5. Verifier Test Mode Keyboard Operation

In Verifier Test Mode, navigation and validation is mainly done using keyboard operations. Compared to the Verifier application, a reduced set of keyboard operations is available.

The following operations can be used in classification mode:





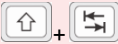



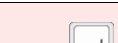

Keyboard Shortcut	Description
n/a	If you know the correct class name, you may type its first characters and wait until the systems automatically displays the full class name.
	Moves one class up in the selection list.
	Moves one class down in the selection list.
	Validates the classification result and displays the current document in extraction / indexing mode.
	Moves to the next field in the current document or to the first field after document change.
	Moves to the previous field in the current document.
	Starts the database lookup, if smart indexing is configured.
	Deletes the end of the line.
	Deletes the entire field content.
	Inserts a new line into a multi-line field.
	Validates the current rejected character or field and moves to the next invalid character field. Depending on your current settings, this may involve a document change. Validation may also trigger smart indexing.

Table 13-7: Keyboard shortcuts in extraction / indexing mode

### 13.2.6. Testing the Visible Classes

Testing the classification verification is simple. All you need to check is whether classes that are not supposed to appear in Verifier are actually hidden.

Prepare as document input:

- Unclassified documents. The quality of the documents is not important, you could for instance just use some \*.tif images from the file system.

Use the following toolbar settings in Verifier Test Mode:

- Default settings.

To run a test:

- 1) Load the documents and switch to Verifier Test Mode. Because the current document is unclassified, the classification user interface is displayed.
- 2) To check the visible classes, open the list box at the bottom of the window.
- 3) Select a class and press *ENTER*. This classifies the document manually. The program will then check whether there is a form available for the selected class. If a form is available, the extraction is performed, and the document and its extraction results are displayed using this form.

### 13.2.7. Testing the Verification Form Layout

Next, you should test whether your verification form layout works under “normal” circumstances -- that is, with documents that extract nicely.

Prepare as document input:

- Classified documents. To work with documents that do not cause particular problems, you can take the documents from the Learnset.
- Create separate batches for each document class to test single forms, and another batch with mixed content to test whether the correct form is displayed for each class.
- If you permit partial validation and have multiple forms per class, where each form is associated with a given state, you can edit the batch control file and modify the document state in order to force verification with a given form. Obviously this only lets you pretend that actually finished processing steps have not yet been performed, but not vice versa. Also, you need to make sure that batch states, folder states and document states remain consistent. For detailed information about the structure of the batch control file, please refer to the ***Runtime Server User's Guide***.

Use the following toolbar settings in Verifier Test Mode:

- Default settings are fine in most cases.
- To test a single form if your document input is heterogeneous concerning the classes, force validation with this form.
- To resolve special problems which only occur with some documents, do not navigate to the next document on validation.

To run a test:

- 1) Load the documents and switch to Verifier Test Mode. Since the current document is classified, the extraction / indexing user interface is displayed.

- 2) Press the *TAB* key to activate the first field. The arrow icon appears next to the field.
- 3) Check whether
  - your field has the required size,
  - you have multiline edits where you need them,
  - the associated viewer shows the appropriate document area,
  - the relevant information in the viewer is readable.
- 4) Use the *TAB* key to move to the next fields and check them as well.
- 5) When the last field is checked, press *ENTER*. This validates the document. The field background appears in yellow, and the mouse pointer assumes the shape of an hourglass. Once the document is validated, the next document is loaded into the viewer area.
- 6) Repeat this procedure for a number of documents to make sure that you cover all relevant cases.

### 13.2.8. Testing Validation Rules

Next you should test whether your validation rules are functional. The list below provides an overview of possible limitations concerning field content:

- Handling of OCR rejects.
- Permitted number of characters.
- Range of valid characters.
- Restriction to uppercase / lowercase characters.
- Handling of forced validation.
- Custom validation rules implemented as WinWrap Basic scripts.

Prepare as document input:

- Classified documents.
- In order to run tests where you deliberately enter invalid input, you can use documents that extract nicely. You can, for instance, reuse the batches created previously to test the form layout.
- In order to test the handling of OCR rejects and of empty fields, create a corresponding set with OCR problems, for instance by scanning with a lower resolution.

Use the following toolbar settings in Verifier Test Mode:

- In order to test as many violations as possible with a limited number of documents, do not navigate to the next document on validation.

To run a test:

- 1) Load the documents and switch to Verifier Test Mode. Since the current document is classified, the extraction / indexing user interface is displayed.

- 2) Press the *TAB* key to activate the first field. The arrow icon appears next to the field.
- 3) Test the field validation rules as follows, and observe the messages that are displayed in the status bar:
  - If the field contains OCR rejects, correct them and use the *ENTER* key to confirm each corrected character. You should not be able to validate the field if rejects are forbidden.
  - If applicable, enter strings that are too long or too short. You should not be able to validate the field unless the number of characters is within the valid range.
  - If applicable, enter some invalid characters. They should be discarded automatically.
  - If applicable, enter some characters in the wrong case. They should be converted automatically.
  - If possible, enter some input that is invalid and try a forced validation by pressing *ENTER* three times. If this is permitted, a message box should be displayed. Select *No* to interrupt the forced validation.
- 4) Without validating the field, use the *TAB* key to move to the next field and test their validation rules as well.

### 13.2.9. Testing Smart Indexing

Miscellaneous preparations:

- You should manually test whether the feature is working as expected. To do this, turn off automatic smart indexing while you are testing.
- You should have access to the database used for the lookup.

Prepare as document input:

- Extracted documents which have been generated with the automatic lookup turned off. In order to work with documents that do not cause particular problems, you can for instance take the documents from the Learnset. Test each class concerned separately.

Use the default toolbar settings in Verifier Test Mode:

To run a test:

- 1) Load the documents and switch to Verifier Test mode. Because the current document is classified with an invalid extraction result, the extraction / indexing user interface is displayed.
- 2) Activate a smart index field that can be used to start a lookup. The arrow icon, the key icon, and the database icon appear next to the field. All other fields that can be filled using database lookup are marked with a database icon.



Figure 13-22: Smart indexing fields (before lookup)

- 3) Run a search with the value that has been extracted, or enter a value that should return a single result. Then press ALT+F12 to trigger the lookup. This should fill the lookup fields.



Figure 13-23: Smart indexing fields (single result returned.)

- 4) Run a search that should return multiple results. This can, for instance, be a wildcard search, using the \* to represent multiple characters, or the ? to represent a single character. Two different results are acceptable in this case.
  - If multiple results are forbidden, a dialog box should be displayed where you can select the appropriate record. The lookup fields are then filled accordingly.

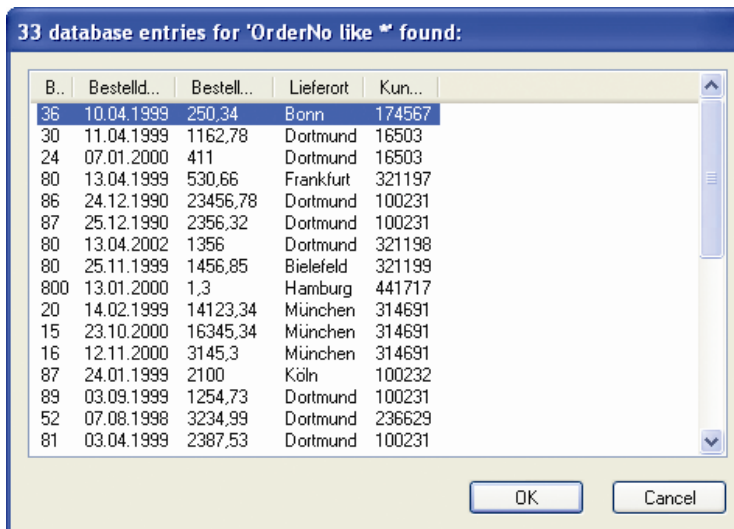


Figure 13-24: Selection list with multiple results

- If multiple results are allowed, the lookup fields will be filled with multiple values. In this case, you need multiline fields to display the results properly.

- 5) Run a search that should yield no results. You should see a corresponding error message.

If you need automatic smart indexing, do not forget to change the settings again when your tests are finished.

### 13.2.10. Testing Table Analysis and Correction

The preparations for table analysis test are essentially the same as for general extraction tests. The list below provides an overview of possible limitations concerning table content:

- Default columns.
- Cells with required entries.
- Valid column formats.
- Handling of forced validation.
- Custom validation rules implemented as WinWrap Basic scripts.

Prepare as document input:

- Classified or extracted documents.
- Create separate batches for all document classes that use common table settings.

Use the following toolbar settings in Verifier Test Mode:

- Default settings.

To run a test:

- 1) Load the documents and switch to Verifier Test Mode. Since the current document is classified, the extraction / indexing user interface is displayed.
- 2) The most basic test checks the presence of the table field. If some table fields are missing, you will have to improve the table analysis settings, or provide a mechanism how these cases are handled. A simple method to make sure that table fields exist (which can then be corrected) is the use of default columns.
- 3) The table field provides additional highlighting options:



	Description	Single Price	Quantity	Total Price
1	Adapted 872400	1,102.98	1	1,102.98
2	Adapted 872500	865.87	1	865.87
3	Emulex LP9802 ...	1,486.22	1	1,486.22

Figure 13-25: Table field example.

- Clicking the button in the upper-left corner of the field highlights the entire document table.
  - Clicking a column label in the field area highlights the document column.
  - Clicking a row label in the field area highlights the document row.
  - Clicking a cell in the field area highlights the document cell.
  - Valid areas appear in green, invalid areas in red.
- 4) Use the highlighting options for the following tests:

- Check whether table tops and bottoms have been detected correctly.
  - Check for default columns. They must be available in valid tables. If a default column is missing, the corresponding table must be invalid.
  - Check for mandatory cells. They must be filled in valid tables. If a mandatory cell is empty, the corresponding table, column, and row must be invalid.
  - Check whether column formats have been applied correctly. If cells have an incorrect format, the corresponding table, column, and row must be invalid.
- 5) Check whether forced validation of invalid tables is possible.
- 6) The column labels' shortcut menu allows you to correct the column structure of tables. It contains the following commands:

Command	Description
Unmap Column	Removes the current column.
Swap Column	Exchanges the position of the current column and the one selected from the drop-down menu.

*Table 13-8: Table column shortcut menu*

- 7) The row labels' shortcut menu allows you to correct the row structure of tables. It contains the following commands:

Command	Description
Insert Row	Inserts an empty row above the current one.
Delete Row	Deletes the current row.
Duplicate Row	Duplicates the current row.
Append Row	Appends an empty row at the bottom of the table.

*Table 13-9: Table row shortcut menu*

- 8) Fill the cells with correct values and validate the table by pressing *ENTER*.

## 14 Printing

You can print documents from Designer, including pages with highlights.

To print a document:

- 1) On the Main Menu, click *File* and then *Print*.
- 2) On the *Print* dialog box, select *Settings* and click *OK*.

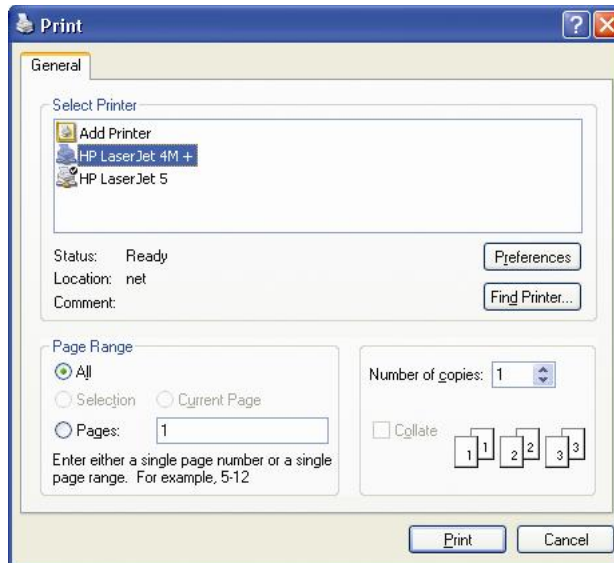


Figure 14-1: Typical print dialog box. Print dialog box



## 15 Reusing Project Settings with Templates

In data extraction, once your analysis and evaluation settings for a field are working, you can reuse them elsewhere within or outside your project. For example, if the format analysis has been configured to find a date, it could be useful to take the same definition at any other place where dates have to be extracted. Therefore, in Designer, analysis and evaluation settings can be saved as templates. You can also create templates for table definitions.

To learn how to work with Validation templates, see section [SETTING UP THE VALIDATION](#).

### 15.1 Creating Templates

#### Task Prerequisites

The prerequisites for this task are:

- The program is in Definition Mode.
- The panel on the left side of the window displays the *Fields* tab in the foreground.
- A field is selected for which settings have been defined that will be reused.
- On the right side of the window, the tabs with class/field properties are visible.

To create analysis templates: The *Analysis* tab is displayed in the foreground.

To create evaluation templates: The *Evaluation* tab is displayed in the foreground.

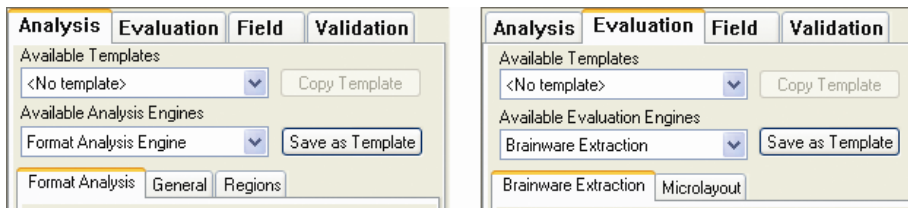


Figure 15-1: Analysis & Evaluation Tab

#### Creating Templates

To create templates:

- 1) Click *Save as Template*. The *New Template* dialog box is displayed.
- 2) Enter a name for the template and click *OK*. The template is saved internally within the project.

### 15.2 Using Evaluation and Analysis Templates within Projects

There are two ways to use a template within a project:

- As defined.
- As a starting point for further adjustments.

#### Task Prerequisites

The prerequisites for both methods are:

- The program is in Definition Mode.
- The panel on the left side of the window displays the *Fields* tab in the foreground.

- A field is selected.
- On the right side of the window, the tabs with class/field properties are visible.
- To use an existing analysis template: The *Analysis* tab is displayed in the foreground.
- To use an existing evaluation template: The *Evaluation* tab is displayed in the foreground.

### Using Existing Templates

To use an existing template as defined:

- Select it from the *Available Templates* list box. This overwrites all previous settings. In addition, the tabs for analysis and evaluation settings will be hidden.

To use an existing template and adjust it:

- 1) Select it from the *Available Templates* list box.
- 2) Click *Copy Template*. This updates the tabs for analysis or evaluation settings below with the settings from the template. You can modify the settings. However, this will not be reflected within the template.

## 15.3 Editing Analysis and Evaluation Templates

In Designer, templates are stored and managed on the project level. To edit them, you need to display the project properties.

### Task Prerequisites

The prerequisites for this task are:

- The program is in Definition Mode.
- The panel on the left side of the window displays the *Classes* tab in the foreground.
- On the right side of the window, the tabs with class/field properties are visible.

### Viewing Templates

To view the templates:

- 1) In the *Classes* tab, right click the entry representing your project to display a shortcut menu.
- 2) Do one of the following:
  - Select *Show Analysis Templates* to display the tab of analysis templates.
  - Select *Show Evaluation Templates* to display the tab of evaluation templates.

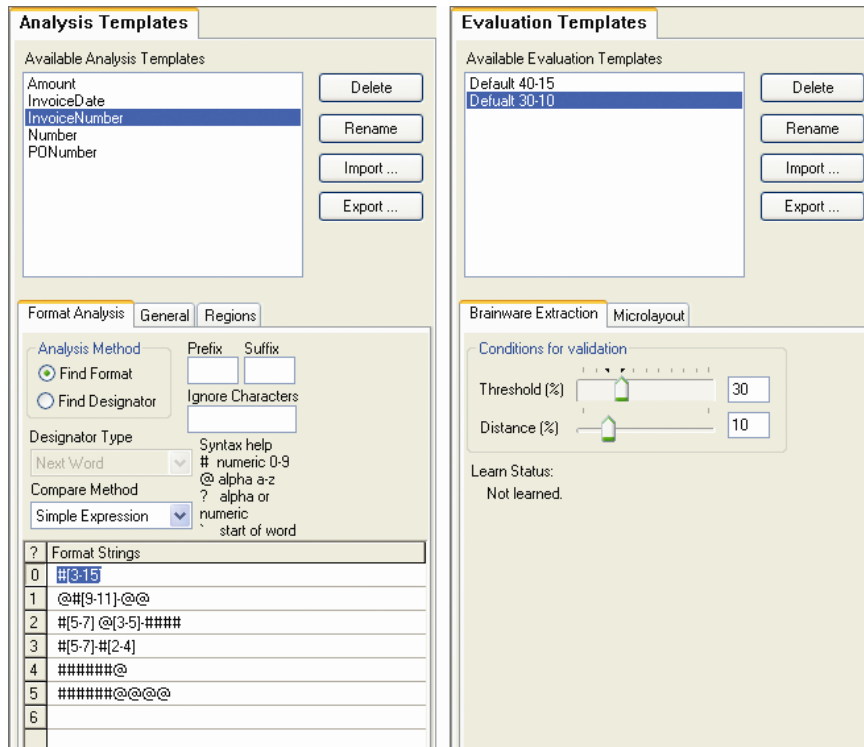


Figure 15-2: Overview of analysis and evaluation templates

### Options for Editing

Three editing options are available within these tabs: Renaming, Deleting, and Changing the Settings.


- **Renaming:** To rename a template, select it from the *Available Templates* list box and click *Rename*. Enter the new name in the following dialog box. You can only rename templates that are currently not in use.
- **Deleting:** To delete a template, select it from the *Available Templates* list box and click *Delete*. Confirm in the following dialog box. You can only delete templates that are currently not in use.
- **Changing the settings:** You can change all settings that are saved with the template. To change the template settings, select it from the *Available Templates* list box and edit in the tabs that are displayed below. All fields using a template will immediately work with the new settings.

## 15.4 Exchanging Analysis Templates Between Projects

You can save analysis templates as files and use them in other projects. This feature is not supported for evaluation templates.

The prerequisites for this task are:

- The program is in Definition Mode.
- The panel on the left side of the window displays the *Classes* tab in the foreground.

- On the right side of the window, *class and field properties* (Click on ) are visible.

### 15.4.1. Exporting Templates to Another Project

To export templates:

- 1) In the *Classes* tab, right click the entry representing your project to display a shortcut menu.
- 2) Select *Show Analysis Templates* to display the tab of analysis templates.
- 3) In the *Analysis Templates* tab on the right side of the window, click *Export*. The *Export Analysis Templates* dialog box is displayed.

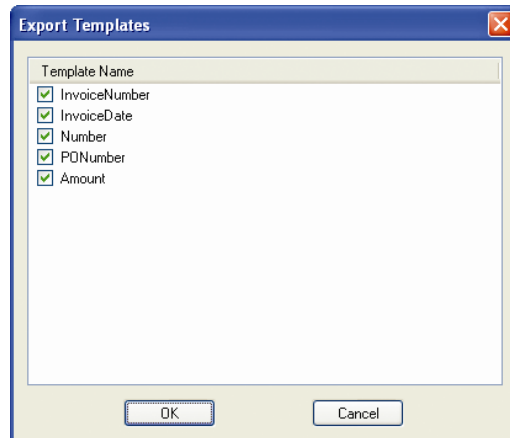


Figure 15-3: Exporting templates

- 4) To exclude templates from the export, clear the corresponding check boxes and click *OK* to start the export. The *Save File As* dialog box is displayed.
- 5) Enter a file name and click *Save*. The template export file is written to the file system with the extension \*.exp.

### 15.4.2. Importing Templates from another Project

To import templates:

- 1) In the *Classes* tab, right click the entry representing your project to display a shortcut menu.
- 2) Select *Show Analysis Templates* to display the tab of analysis templates.
- 3) In the *Analysis Templates* tab on the right side of the window, click *Import*. The *Open* dialog box is displayed.
- 4) Select a template export file with the extension \*.exp and click *Open*. The templates from the file are imported and displayed in the *Available Analysis Templates* list box. The program automatically recognizes the template type contained in the export file. If it is the wrong type, template will not be imported.

## 16 Setting Up the Document Export


Without an export script, the documents are written to the Runtime Export directory after processing.

Individual document export options need to be implemented using WinWrap Basic scripts. Please refer to the WebCenter Forms Recognition ***Scripting Documentation*** for detailed information about the available options.

### Example:

The following export script writes classified images to subdirectories within the export directory. Each subdirectory holds documents from one class only; the subdirectory name corresponds to the class name.

To implement the script:

- 1) If required, switch to Definition Mode.
- 2) In the toolbar, click *Script*  to open the WinWrap Basic IDE.
- 3) From the list box to the left, select *ScriptModule*.
- 4) From the list box to the right, select *ExportDocument*. This generates the outline of a WinWrap Basic subroutine.
- 5) Add the following code:

```
Dim sNewPath As String
Dim MyImage As SCBCroImage
Dim NewFileName As String
sNewPath=ExportPath & "\" & pWorkdoc.DocClassName 'Set directory name
Set MyImage=pWorkdoc.Image(0) 'Access the image file to the current
WorkDoc
On Error GoTo Skip 'Skip next step if directory exists
MkDir sNewPath 'Create directory
Skip:
NewFileName=Mid(MyImage.FileName, InStrRev(MyImage.FileName,"\")) 'Set
file name
MyImage.SaveFile sNewPath & NewFileName 'Save file to directory
```

- 6) Close the dialog box.
- 7) Test the export script in Runtime Mode.

## Appendix A Auxiliary Tools

### A1 Auto-Generation of Template Local Projects in SLW

An overall Learnset Management within a company's mainframe could be essentially simplified through a couple of additional tools as presented below.

You may wish to create a new Local Project out from an existing Global one, keeping either the entire settings of your Base class, your Learnset and your Script, or, maybe, some other reasonable combination of your choice. This task could be easily performed using a "SLW Export Local Project" tool that can be launched via Windows *Start \ Programs \ Oracle \ Forms Recognition 11.1.1.8.0 \ Tools \ SLW Export Local Project Tool* menu item.

The first dialog allows you to search and choose a project, to be used as source:

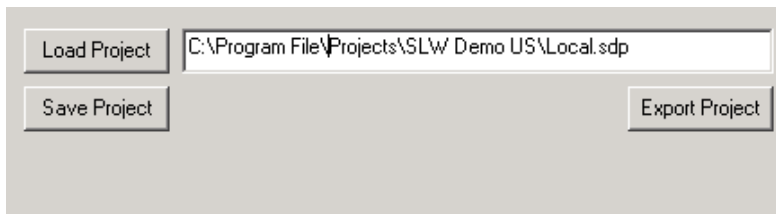


Figure A-1: Export Local Project dialog box

The next dialog is an internal structure (P - as project, B - as base class) of the selected one and provides a possibility to select (check/uncheck) the desirable settings to be kept or withdrawn.

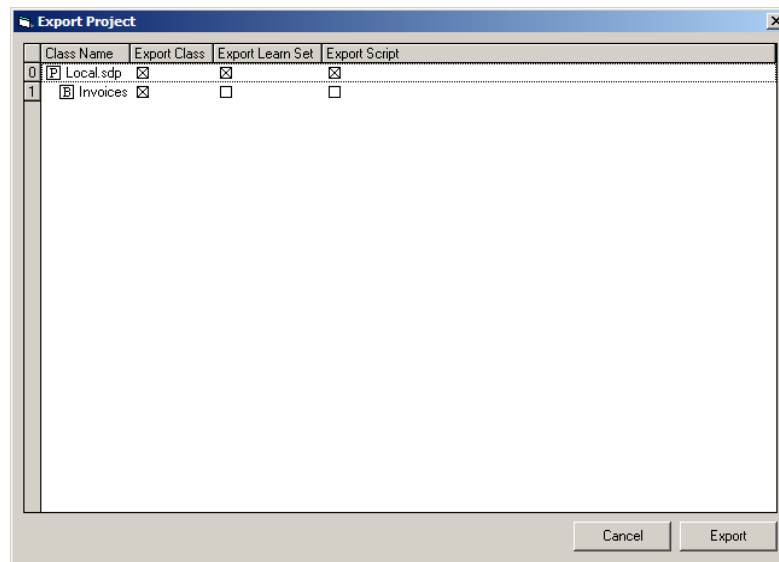


Figure A-2: Settings selection for the selected project

*Cancel* will quit this step and return to dialog 1. *Export* gives, as usual, a possibility to name a Target Project and save it for future use. After saving, return to the dialog 1 to proceed with next source Projects, if any. While exporting, the tool automatically analyzes and creates the corresponding Learnset directories and Learnsets, based on a new name of the Target Project, or prompts for a replacement, if a name of the Target Project already exists.

*Save Project* in dialog 1 saves the entire export project configuration as a part of the global project's stream and exits.

Usage:

This tool can be used for automation of manual creation of template local projects (to be used by Advanced Verifier workstations) out of the primary global one in content of supervised learning workflow. As a result, it will minimize the corresponding PS efforts required each time a new local project is supposed to be created.

## A2 Auto-Merging of SLW Learnsets

An Administrator supporting a highly distributed system of different workstations could be easily confronted with maintenance of a high number of separate knowledgebases, created at different locations, at different times, and so on. Thus in some cases it might be required to be able to merge the Learnsets (created in context of the same production global SLW project) from different project files, being able to provide some additional considerations. This merging method has to be intelligent enough to be able to select some Learnset settings for a particular document within a class to be either merged or withdrawn. This goal can be achieved with an auxiliary tool, named "Merge SLW Learnsets". The tool can be started from Windows *Start / Programs / Oracle / Forms Recognition 11.1.1.8.0 / Tools / Merge SLW Learnsets* menu item.

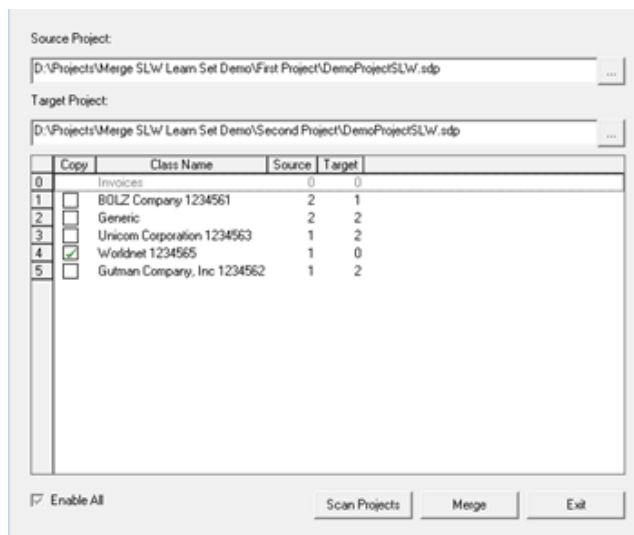


Figure A-3: Merge SLW Learnsets dialog box

First, scan projects to discover and analyze the structures of concern. The button *Merge* stays disabled for the very first, and is enabled after a successful scan, if there are some logical steps to be followed (See the screenshot above).

In the case above, the tool analyzed and found out:

- Base Class "Invoices" with no Documents in Learnsets (neither Source, nor Target)
- Lines 2,3,4,5 the Derived Classes with names and number of already learned Documents (both in Source and Target projects)
- As of Line 4 the "Worldnet 1234565" has one Document in the Source project, but no Documents in the Target one, therefore the tool automatically suggests to make a Copy from Source to Target project ("Copy" check-box checked). You can also

manually select some other nodes; for example, you'd choose two extra nodes "Unicom" and "BOLZ", the class "Unicom" of the Target project is going to be trained with one more document available for this class in the Source project and the class "BOLZ" – with two more documents available in the Source project.

If there is nothing to be merged, the button *Merge* will remain disabled. Select *Exit* to quit this tool.

*Important Note: The "Merge SLW Learnsets" tool is designed to work only with SLW (supervised learning workflow) projects, which are the instances of the same master (global) SLW project. In this connection, only the SLW vendor classes can be populated and/or updated via this tool. In the project contains any classes that are not created automatically by SLW, these classes cannot be upgraded via the present tool.*

Usage:

This can be used for quick merging of multiple SLW sub-projects, overtaking the "SLW knowledge" without usage of Learnset Manager tool. For example, in order to quickly correct corresponding configuration errors related to "disconnected" local Advanced Verifier workstations.

### A3 New Fields in WebCenter Forms Recognition Workdoc Browser Tool

The Workdoc Browser tool for reviewing of WebCenter Forms Recognition's document objects has been extended to support "ADS Numeric ID" and "ADS Alpha-numeric ID" fields that represent actual extraction result provided by the Associative Database Search engine:

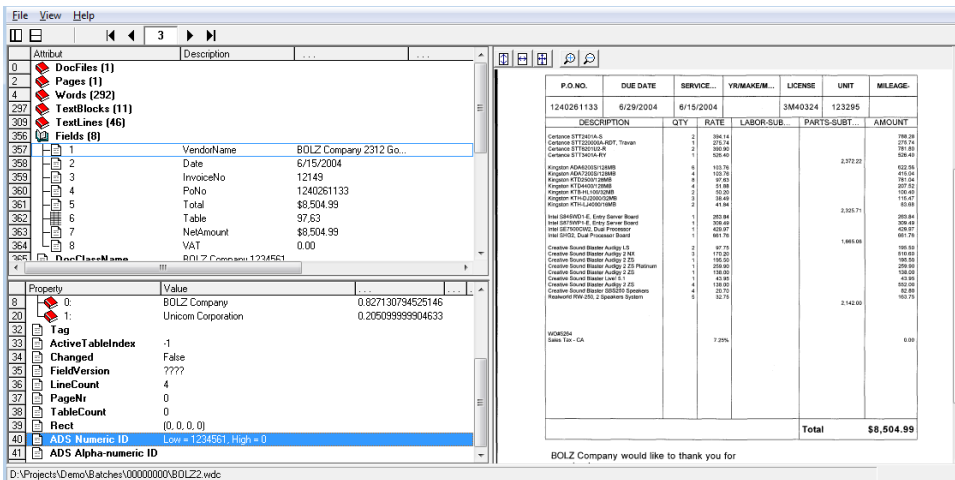


Figure A-4: Workdoc Browser

In context of the Associative Database Search (ADS) engine's extraction, the text content of the extracted result is just the way the system represents it to the end-user. Meanwhile, when the system uses an ADS field in terms of, e.g. automatic SLW learning, WebCenter Forms Recognition does not use the ambiguous textual content of the field, but takes either the "ADS Numeric ID" (using "GetUniqueEntryId" property of the SCBCdrField interface) or "ADS Alpha-numeric ID" (using the "UniqueID" property) instead. Whether the 8-byte numeric ID or alpha-numeric ID is going to be used can be queried by the "IsIDAAlphaNum" property of the SCBCdrField interface.

Usage:



This can be important for a system administrator to be able to review the actual engine's results that is going to affect further functioning of the SLW sub-system.

## Appendix B Designer – Quick reference

<p><b>Project Manipulation</b></p> <ul style="list-style-type: none"> <li> Load Project</li> <li> Save Project</li> <li> Show Settings</li> </ul>	<p><b>Document Processing</b></p> <ul style="list-style-type: none"> <li> Classify/Analyze</li> <li> Execute and debug</li> <li> Execute and Export</li> <li> OCR and Execute</li> <li> Process next document</li> <li> Process multiple documents</li> <li> Wait at current position</li> <li> Stop processing</li> </ul>	<p><b>Miscellaneous</b></p> <ul style="list-style-type: none"> <li> Show/Hide Script</li> <li> Pagedesigner</li> <li> Show/Hide classification Matrix</li> <li> Show Properties</li> </ul>
<p><b>Mode Selection</b></p> <ul style="list-style-type: none"> <li> Document Input (Batch)</li> <li> Document Input (Directory)</li> <li> Document Input (Learnset)</li> <li> Definition Mode</li> <li> Train Mode</li> <li> Runtime Mode</li> <li> Verifier Test/Design Mode</li> </ul>	<p><b>Verifier Design Mode</b></p> <ul style="list-style-type: none"> <li> Create a form for all classes</li> <li> Create a form for current class</li> <li> Delete all forms</li> <li> Delete current form</li> <li> Maximize form view</li> </ul>	<p><b>Document Recognition</b></p> <ul style="list-style-type: none"> <li> Adjust documents display to Height</li> <li> Adjust documents display to Width</li> <li> Fit page into window</li> <li> Zoom in</li> <li> Zoom out</li> <li> Selection tool</li> <li> Insert OCR Zone</li> <li> Insert OMR Zone</li> <li> Insert Barcode Zone</li> <li> Insert Anchor</li> <li> Previous page</li> <li> Next page</li> </ul>
<p><b>Document Highlighting</b></p> <ul style="list-style-type: none"> <li> Highlight Nothing</li> <li> Highlight Words</li> <li> Highlite Blocks</li> <li> Highlite candidates</li> <li> Highlite Fields</li> </ul>	<p><b>Verifier Test Mode</b></p> <ul style="list-style-type: none"> <li> Keep focus on Field</li> <li> Keep zoom on Browsing</li> <li> Next document on validate</li> <li> Validate with selected form</li> </ul>	
<p><b>Document Navigation</b></p> <ul style="list-style-type: none"> <li> First document</li> <li> Previous document</li> <li> Last document</li> <li> Next document</li> </ul>	<p><b>Learning</b></p> <ul style="list-style-type: none"> <li> Learn documents</li> <li> Add document to Learnset</li> </ul>	

## Appendix C Regional and Currency Settings

This appendix lists settings for region and currencies used in Validation and Output Formatting

Region	Currency	Supported Symbols
Dutch	Euro, Guilder	NLG, fl, €
Dutch-Belgium	Euro, Franc	BEF, BF, €
English	Euro, Dollar	USD, \$, €
English-Australia	Euro, Dollar	AUD, \$, €
English-Canada	Euro, Dollar	CAD, \$, €
English-New Zealand	Euro, Dollar	NZD, \$, €
English-United Kingdom	Euro, Pound	GBP, €
English-United States	Dollar	USD, \$, €
French	Euro, Franc	FRF, F, €
French-Belgium	Euro, Franc	BEF, BF, €
French-Canada	Euro, Dollar	CAD, \$, €
French-Switzerland	Euro Franc	CHF, SFR, €
German	Deutsche Mark, Euro	DEM, DM, €
German-Liechtenstein	Deutsche Mark, Euro	DEM, DM, €
German-Luxembourg	Deutsche Mark, Euro	DEM, DM, €
German-Swiss	Euro, Franc	CHF, SFr, €
Italian	Euro Lira	ITL, L, €
Italian-Swiss	Euro, Franc	CHF, SFr, €
Neutral	Euro	€
Norwegian	Euro, Kroner	NOK, k
Polish	Euro Zloty	€
Russian	Euro	€
Spanish	Peseta	ESP, pta
Spanish-Mexico	Peso	MXN, ls
Spanish-Modern	Peseta	\$, ESP, pta
Swedish	Euro, Krona	SEK, KR, €

## Appendix D Project Structure and Files Extension

This appendix illustrates a typical project's folder structure and the types of files and file extensions in use. If you installed WebCenter Forms Recognition as recommended, projects are stored under...\\program files\Oracle\projects. Each project has its own folder structure. The root folder for the project carries the same name as the project name (the \*.sdp file you created for the project.) Folder structures for projects that did not use associative learning differ from those that do use associative learning.

### Non-Associative Projects

In non-associative projects, the project file, the \*.sdp file, templates and comma-delimited reference files (\*.csv), such as vendor pools, are stored at the project root. Below the project root, all projects have Batch folders, Export folders and Learn folders. The primary batch folder holds batch state files, batch control files, lock files, sdb files and sdl files. Batch folders contain secondary folders for each batch in the project. These folders hold the Workdocs and image files for each document in the batch. The learn folder contains all the Learnsets for the project. Each of these Learnset folders contains the images and workdocs in the Learnset. The Export folder contains exported files.

### Associative Learning Projects

The folder structure of Associative Learning Projects is more granular. As with the project root for non-associative projects, the project root contains the project file, the export files, templates, and \*.csv files.

The project root also should consist of the following folders:

- Batches
- Common Learnset
- Export
- Global Project
- Input
- Local Project
- Misc.
- Vendor Pool

### File Extensions

Region	Currency	Supported Symbols
SDP	Project file	project root
EXP	Exported templates	project root
CSV	Comma-delimited reference files	project root
DAT		batch folder
500 (or some other number)	Batch state files with the state of the number assigned	batch folder
SDB		batch folder
SDL		batch folder, learn folder
WDC	WorkDocs	specific batch folder, learn folder

Region	Currency	Supported Symbols
TIF	Tagged Image Format Files	specific batch folder
ID0	Learned image files	Learn folder
STS	Project Status Information	Vendor pool folder
TMP	Temp File	Vendor pool folder
ATT		Vendor pool folder
DCT		Vendor pool folder
DFF		Vendor pool folder
DLY		Vendor pool folder

## Glossary

Address pool	Special database format that stores address entries that can be used during the analysis step to obtain candidates.
Administrator	In Forms Recognition, an administrator is a power user who creates user accounts, passwords, and groups, and assigns users to groups.
Analysis	In this processing step, the document content is analyzed and a set of possible values for a field is generated. These values are called candidates.
Anchor	Anchors are special types of reading zones. They look for prominent geometric features on documents, such as page corners. Once these features are found, anchors can provide a fixed coordinate system for the reading zones. Position for reading zones is determined relative to anchor positions. This can overcome fluctuations in the page position of scanned images.
ASSA Classification Engine	Retrieves text from a document block by block and generates a query string from the results, and then searches for that query.
Associative Search Engine	Uses a reference field to extract results.
Auto-deskew	Automatically corrects distortion of a scanned image.
Auto-orientation	Automatically makes a scanned image less crooked.
Automatic Supervised Learning	Uses the Associative Search Engine to process, classify, and extract vendor information.
Barcode	A pattern of vertical black bars separated by spaces. Bars of varying thickness represent different characters. In document management, barcode recognition can be used to uniquely identify or to separate documents.
Base class	The highest level of a classification
Batch	A logical organizational structure to control a set of documents during a process. A batch is normally created during the scan process from a batch of paper. The status of a batch is used to manage the input flow.
Brainware Classification Engine	The Brainware Classification engine is used for content or type classification.
Calibration	Fine-tuning of classification results
Candidate	Set of possible values for a field.
Character filtering	Allows you to indicate valid and invalid characters and characters that should be removed or replaced during validation.
Child class	A class spawned by a parent class. See also base class and parent class.
Class	A set of documents that are grouped by common content. Each class usually has a mnemonic name that describes its contents from the user's point of view.
Classification	The process of assigning one or more classes and corresponding confidence values to one or more unknown documents.
Classification Learnset	Set of sample documents needed for Brainware classification.
Classifier	In OCR, classifiers are specialized algorithms that cast their votes in favor of one character over another. OCR accuracy and performance can be improved if only a subset of classifiers needs to be involved in character recognition.
Confidence	In classification, the confidence level indicates the degree of similarity between a document to be processed and the documents in the Learnset of a given class.
Definition Mode	Used to design classification, extraction, validation, verification, and export.
Derived Validation	Tells a child field, document, or class to inherit all validation settings from its parent.

Designer Test Mode document routing	A way to test export scripts
Document Selection Mode	Used to select documents to be used in learning and testing
Deskewing	Because of the mechanical feeding action of scanners, documents may get out of alignment during scanning. Deskewing is the process used to remove skew or distortion from scanned images through a small angle rotation.
Despeckle	Process used to remove speckles from scanned images. Speckles are made up of groups of black pixels surrounded by white pixels or vice versa.
Dictionary	The documents from each Learnset must be preprocessed before classification. All documents in the set are filtered and statistically analyzed. Their wording patterns are used to create a dictionary that is required by the neural network for learning.
Distance	In classification evaluation, the distance is the difference between the confidence level of the best and the second-best class. In extraction evaluation, the distance is the difference between the weight of the best and the second-best candidate. It is a measure of how clearly the winner of the evaluation can be distinguished from the runner-up.
DPI	Dots per inch. Affects the size and clarity of an image file.
Document	Any electronic file mainly consisting of ASCII text. If this is not the case in the first place, OCR or filtering must be applied to create the text representation. A document can be classified, a document can have fields used for extraction, and a document can have one or more images attached.
DocClass	A parent Document Class.
Evaluation	The process of determining a class or the contents of a field from confidence levels, weights or distances for classes or candidates.
Export	In Forms Recognition, document export releases the documents so that they are no longer managed by the software.
Extraction	The process of automatically finding specified information within the contents of a document and writing the information to data fields associated with the document. Extraction is used for automatic indexing.
Folder	A logical structure inside a batch for coherent documents. A folder may for example consist of all pages of a correspondence with many folders inside one batch.
Forced validation	Means that a document or field is valid regardless of whether it would normally be invalid based on its contents or assigned validation settings.
Form	(1) A structured, standardized document that is used to support business processes. (2) A custom dialog box in a software application.
Format Analysis	Uses a formal description of a character string to define a set of search strings.
Forms Classification Engine	Can identify forms or other structured documents that have an identifier of the document class printed on them. If the identifier is placed at a fixed position on the document, reading this zone can quickly provide a classification result. Not a first choice for a single classification method.
Global Learnset	A general Learnset that encompasses similar classes or projects. See also Local Learnset
ICR	Intelligent Character Recognition. Special type of OCR that is used to process handwriting.
Image	A digital raster image normally created during scanning. The image is compressed and stored in a specific format. Internally, Forms Recognition uses a structure called Image that represents the raster image. The image contains methods to load, store and manipulate it. The image can be displayed in the viewer.
Image Size Classification Engine	Uses the physical dimensions of a document to classify it. Often used to rule out a document's membership in a certain class, rather than to confirm it.

Importing	Bringing documents into Forms Recognition for management and processing
Indexing	The process of assigning attributes to a document. This can either be done manually semi-automatically (Smart Indexing), or entirely automatically (Extraction).
Inheritance	The order in which validation, analysis, or extraction characteristics are passed between parent and child. Can be top-to-bottom or bottom-to-top.
Chronology	
Learnset	In classification, a Learnset is a set of documents whose class assignments are specified by the user. For each view and each class, the user must provide a sufficient number of representative documents. Similarly, in extraction, a Learnset is a set of documents whose field contents are selected by the user from a set of candidates.
Learning	Given a view with a set of documents in vector representation and their class assignments, a neural network is created, so that the so-defined classes can be reproduced without error. This neural network is then used in all subsequent classification tasks.
Levenshtein search	Error-tolerant search method that finds each literal occurrence of a specified string, but also strings that can be derived from the specified one by inserting, interchanging or deleting single characters. The number of key operations required to derive the erroneous string determines whether there is still a match. Accounts for typing errors.
Literal character	Normal alphanumeric characteristics that are not used as operators.
Local Learnset	Learnset specific to a document class.
Matrix	The ability to read documents created on a dot-matrix printer.
Recognition	
Neural network	An artificial neural network is an application that in some ways works like a human brain. This includes the ability to learn. It consists of artificial neurons that are linked into a network of layers. The neural network can receive signals through an input layer, process it within the internal layers and send signals through the output layer. During learning, a specified input (a so-called teacher signal, such as documents from a Learnset) and the desired output (such as the corresponding classes) are presented to the network together. Processing is then adjusted until the desired output can be produced from the teacher signal.
ODBC	Open Database Connectivity. Enables you to access files from an external database.
OCR	Optical Character Recognition. The reading and recognition of symbols of text from a piece of paper or a scanned image. OCR detects the symbols and converts them into characters and words that can be read electronically.
OMR	Optical Mark Recognition. The sensing of marks on a document. OMR might be used to collate questionnaires or multiple-choice examinations where a student marks certain boxes on an answer sheet. OMR detects the marks and converts them into electronic signals.
On demand	A process that is performed whenever requested, not just once.
Output	Enables you to standardize the appearance of data exported from Forms Recognition.
Formatting	
Parent class	A class with derived classes, called children.
Patchcode	A pattern of horizontal black bars separated by spaces and typically placed near the leading edge of a paper document. Patchcodes can be used to separate documents.
Persistent	Permanent; something that is saved persistently is save permanently, unless the user deletes it.
Phrase	Uses words or phrases to identify a document class.
Classification	
Engine	
Project	Project files are used to persistently save custom settings for Forms Recognition applications. They are created in Forms Recognition Designer



	and handed over to Forms Recognition Runtime for productive operation.
Project Settings Tree	Acts as a registry for Forms Recognition.
Reading zone	Fixed areas on documents that contain information that is to be extracted.
Regular expression	Generalized pattern language used to search strings that match a specified pattern.
Runtime Mode	Used to testing classification, extraction, and export scripts. Can also be used to perform OCR.
Simple expression	Simple pattern language used to search strings that match a specified pattern.
Smart Indexing	Smart indexing uses a database lookup to determine document attributes. It can be used for automatic indexing and to support manual indexing.
Special character	Characters that can be used as operators.
Standard Validation Engine	Automatic interface used to enhance Forms Recognition's tradition script-based validation capabilities.
Stretching	Used with other classification engines, improves the results of Brainware classification by increasing the distance (in confidence levels) between results.
Sub-class	A derivative class.
Supervised-Learning Manager	A user who designs, modifies, and maintains Learnsets.
Supervised-Learning Verifier	Users who collect and maintain local training data.
Table Analysis.	Extracts records organized in columns and rows on a document.
Template	A named set of settings used in data extraction. Via templates, the same set of settings can be reused within projects and exchanged between projects.
Template Classification Engine	This engine has been fully replaced by the Brainware Classification Engine.
Train Mode	Used to train classification and extraction. Can also be used to create Learnsets.
Trigram search	Error-tolerant search method for strings. To compare two words, they are fragmented into groups of three characters called trigrams. The number of identical groups determines whether there is still a match. Accounts for OCR errors.
Typewriter Recognition	Reads output from documents that were created on a typewriter.
Validation	A quality assurance task that involves confirming whether a processing result is correct. This can be done at several levels: for the class or a field associated with a document, for the document as a whole or for an entire batch.
Verification	A quality assurance task that involves checking and correcting processing results.
Verifier	Forms Recognition's QA application.
Verifier Design Mode	Used to create forms that will be used in Verifier for data correction and manual indexing.
Verifier Test Mode	Used for simulation in Verifier Design Mode and to test verification.
Verifier Train Mode	Uses Automatic Supervised Learning to extract meaningful supplier information.
View	A set of documents that represent at least two classes. A view is usually defined using a small set of documents that represent the domain of interest. In a view, classes compete for documents; that is, a document may only be assigned to one class within the view.
Weight	In extraction, indicates the degree of similarity between candidates on a

---

	document that is to be processed and the candidates in the Learnset of a given field.
Word strength	Refers to the number of typos that a word can contain before losing significance. The stronger the word, the more error it can have and still be significant.
Workdoc	An internal structure representing the logical structure of a document. The Workdoc represents the data created during processing of a single document and is stored in a file with the extension *.wdc. Since the Workdoc includes all OCR and analysis results it may exceed the document file by size.