

Oracle® Fusion Middleware

Oracle WebCenter Forms Recognition Installation Guide

11g Release 1 (11.1.1)

E50183-02

November 2015

Documentation for Oracle WebCenter Forms Recognition administrators that describes how to install Forms Recognition.

Copyright © 2009, 2015, 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 is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware 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 that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about 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 unless otherwise set forth in an applicable agreement between you and Oracle. 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, except as set forth in an applicable agreement between you and Oracle.

Contents

1	About WebCenter Forms Recognition	6
2	System Requirements	7
2.1	Operating Systems	7
2.2	WebCenter Forms Recognition Database	7
2.3	Internet Information Services (IIS)	7
2.4	Scripting Components	7
3	Hardware Requirements	9
3.1	Network Infrastructure	9
3.2	Hardware and Software Factors	9
3.3	Hardware Estimates: 500 Pages per Day	10
3.3.1	Introduction: AP Packaged Project	10
3.3.2	Hardware Estimate: Batches Stored on File System (Excl. Web Verifier)	10
3.3.3	Hardware Estimate: Batches Stored in Database (Excl. Web Verifier)	11
3.3.4	Hardware Estimate: Entire Application Suite with Batches Stored in the Database	11
3.3.5	Hardware Estimate: Typical Development Environment	12
3.4	Hardware Estimates: 4,000 Pages per Day	12
3.4.1	Introduction: AP Packaged Project	12
3.4.2	Hardware Estimate: Batches Stored on File System (Excl. Web Verifier)	13
3.4.3	Hardware Estimate: Batches Stored in Database (Excl. Web Verifier)	13
3.4.4	Hardware Estimate: Entire Application Suite with Batches Stored in the Database	14
3.4.5	Web Verifier Server Hardware Sizing: Additional Information	14
3.5	Infrastructure Constraints	15
3.5.1	OCR Performance	15
3.5.2	Firewalls	15
3.6	Runtime Server Remote Administration MMC	15
4	Pre-Installation of WebCenter Forms Recognition	16
4.1	Before Installing WebCenter Forms Recognition	16
4.1.1	Installation Checklist	16
4.2	WebCenter Forms Recognition Database	17
4.2.1	Oracle Database Checklist	17
4.2.2	SQL Server Checklist	17
4.3	Upgrading from Previous Versions	18
4.3.1	Upgrading from WebCenter Forms Recognition Version 11.1.1.8.0	18
4.3.2	Upgrading from Oracle Forms Recognition Version 10.1.3.5.0	18
4.4	Removing Oracle Forms Recognition Version 10.1.3.5.0	18
5	Installing WebCenter Forms Recognition	19
5.1	Software Installation	19
5.1.1	Selecting Custom Installation	20
5.1.2	WebCenter Forms Recognition Database Setup	20
5.1.3	Completing the WebCenter Forms Recognition Setup	21
5.2	Manually Creating Database Objects (Post Install)	21
5.3	Configuring the WebCenter Forms Recognition Database (Post Install)	24
5.4	Installing WebCenter Forms Recognition in a Workgroup	24
5.5	Installing WebCenter Forms Recognition in Standalone Mode	25

5.6	Checking the Installation	26
5.7	Migrating Existing Project Files	26
5.8	Removing WebCenter Forms Recognition Version 11.1.1	27
5.9	Repairing a WebCenter Forms Recognition 11.1.1 Installation	27
5.10	Adding or Removing Forms Recognition Components	27
5.11	Password Encryption in Core Configuration Files	28
5.12	AP Packaged Project INI File Encryption.....	30
5.12.1	Project INI File Encryption for the Project Developer.....	30
5.12.2	Project INI File Encryption for the Integrator	31
5.13	Silent Installations	32
5.13.1	Silent Install.ini	32
5.14	Preparing Internet Information Server (Post Install)	34
5.14.1	Configuring IIS 7 or IIS 8	34
5.15	Running Multiple Web Verifier and RTS Instances	35
5.15.1.1	Problem Description	35
5.15.1.2	Problem Cause.....	35
5.15.1.3	Recommended Configuration Changes.....	36
6	Configuring WebCenter Forms Recognition	37
6.1	Configuring Applications.....	37
6.1.1	Configuring the Oracle WebCenter Forms Recognition Database Connection String.....	37
6.1.2	Enabling HTTP Compression for IIS 7.....	37
6.2	Server Security Configuration	37
6.2.1	Registering COM Components.....	37
6.2.2	Preparing the User Context (SQL Server).....	38
6.2.3	Setting Permissions for Forms Recognition Projects Execution.....	38
6.2.4	Encrypting Sections with the aspnet_regiis Tool.....	39
6.2.4.1	Pre-configuring:.....	39
6.2.4.2	Encryption of the web.config file:	39
6.2.4.3	Decryption of the web.config file:	39
6.3	Client Security Configuration	39
6.4	Configuring Single Sign-On Authentication for Web Verifier	40
6.4.1	Enabling Single Sign-On Authentication.....	40
6.4.2	SSO Session vs. Web Verifier Client Session.....	40
6.5	Configuring Windows Authentication for Web Verifier.....	41
6.5.1	Prerequisites.....	41
6.5.2	Configuration for IIS 7.....	41
6.5.3	Changes to the Web.config File	41
6.5.4	Reverting Back to Standard Authentication.....	42
6.6	Configuring SSL for Web Verifier	42
6.7	Configuring Additional Languages	42
6.8	Virus Check	42
6.9	Enabling New Columns for Batch View	43
6.10	Changing Custom Column Names	44
6.10.1	Custom Column Names for Web Verifier	44
6.10.2	Custom Column Names for the Verifier Application	44
6.11	Disabling (and Enabling) Batch Deletion	45
7	Security 47	
7.1	WebCenter Forms Recognition Security.....	47
7.1.1	Project Security	47

7.1.2	Project and Windows Authentication Security	47
7.2	File System Security	48
7.3	Access to Project Data	49
7.4	Accounts and File Access Security	50
7.5	Configuring the Service Account for WebCenter Forms Recognition.....	51
7.5.1	Running WebCenter Forms Recognition on a Domain Network.....	51
7.5.2	System Monitoring.....	51
8	Configuring Runtime Components.....	52
8.1	Configuring the Runtime Service Manager	52
8.2	Configuring the RTS RemoteAdmin MMC Snap-in	52
8.3	Advanced Logging.....	53
8.3.1	System Resource Logging.....	53
8.3.2	Crash/Failure Logging.....	54
9	Enabling Additional OCR Engine Languages.....	55
9.1	Enabling a Language for an OCR Engine.....	55
9.2	Adding an Input Language for Windows 7 or Windows Server 2008	55
Appendix A	Web.config Options and Associated Resource File Parameters	56
Appendix B	File Permission Matrix	61
Appendix C	Registry Options.....	63

1 About WebCenter Forms Recognition

Oracle WebCenter Forms Recognition is a learning-based intelligent document recognition (IDR) solution that can recognize, categorize and extract information from any type of document. WebCenter Forms Recognition uses intelligence - not templates - to effectively locate, extract, and link data to back-end systems and processes, to provide the industry's highest level of document recognition and data extraction. Together with Oracle WebCenter Enterprise Capture and Oracle WebCenter Content: Imaging, WebCenter Forms Recognition offers a pre-packaged, end-to-end invoice processing solution for Oracle's Financial Management applications to facilitate processing large volumes of invoices with high upfront data extraction accuracy to minimize the need for human intervention.

Within the WebCenter Forms Recognition suite, the Designer application enables you to define and customize the automatic processing of incoming documents, for example, which document classes are relevant in your enterprise as well as which information is to be extracted from the classified documents. All custom settings are saved in a Forms Recognition project file.

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

The 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 Forms Recognition Runtime Server can be started simultaneously in a network or on a single machine. These instances cooperate and allow for optimal load distribution, and high availability.

Batches that cannot be automatically processed in their entirety by the Forms Recognition Runtime Server are forwarded to the quality assurance application, Verifier, or Web Verifier, for manual correction.

The Web Verifier application module allows users to verify documents with no software installed on the client side. Web Verifier can be used via a supported browser on any client machine to verify documents. This requires installation and configuration of the project and batches on the database platform. From version 11.1.1 Oracle WebCenter Forms Recognition features a database platform for Forms Recognition applications. It is possible to store project and authentication information in the Forms Recognition database. This solution allows for central management of storage and backup and thus provides for easier security, better connectivity of your applications, and higher flexibility of your personnel.

2 System Requirements

2.1 Operating Systems

Using WebCenter Forms Recognition requires a complete and successful installation of the software on a Windows server or workstation running one of the following operating systems:

- Microsoft Windows Server 2008 R2 (IPv4 and IPv6)
- Microsoft Windows Server 2012 R2
- Microsoft Windows 7
- VMware ESX 4.1 is certified

Note: WebCenter Forms Recognition requires that .NET Framework 3.5 SP1 be installed on the Forms Recognition server or workstation. This remains true even where a later version of the .NET Framework is installed.

2.2 WebCenter Forms Recognition Database

The WebCenter Forms Recognition database has been certified to run on the following database platforms:

- Oracle Database 11g R2
- Oracle Database 12c R1
- Microsoft SQL Server 2008 R2
- Microsoft SQL Server 2012

Note: The 32-bit Oracle Database Client is required on all workstations or servers where one or more WebCenter Forms Recognition components are installed, and where the Forms Recognition database resides on an Oracle Database.

2.3 Internet Information Services (IIS)

Using WebCenter Forms Recognition Web Verifier will require the following software applications installed on the Forms Recognition web server:

- .NET Framework 3.5 SP1
- Internet Information Server (IIS)

Note: Ensure the IIS installation includes the **ASP.NET** application development role service, and the **IIS Management Console** role service.

The Web Verifier client requires a supported browser. Refer to the *Oracle WebCenter Forms Recognition Certification Matrix* for full details of supported browsers and versions.

2.4 Scripting Components

WebCenter Forms Recognition has been certified for the following supported Scripting component version:

- WinWrap version 9.0.0.56

3 Hardware Requirements

Before you can implement WebCenter Forms Recognition, the underlying network must meet certain minimum platform and environment requirements.

This section includes information about, and instructions for, configuring basic network components.

3.1 Network Infrastructure

The technology infrastructure underlying WebCenter Forms Recognition consists of a set of scalable applications and services running on Microsoft Windows operating systems. These applications and services are deployed on a set of high-performance Intel-compatible servers and workstations.

Clients are supported on Intel-compatible workstations running Windows 7.

WebCenter Forms Recognition is a distributed, two-tier application that is typically deployed across multiple Windows Server operating systems. These applications provide core system services when connected using an unimpeded high-speed, low-latency network infrastructure, such as Fast Ethernet (100BaseTX), using TCP/IP.

3.2 Hardware and Software Factors

When implementing a WebCenter Forms Recognition project, there are a number of dependencies that influence the hardware requirements for a Forms Recognition solution.

This information is typically gathered during the requirements analysis phase of the project and used as a guideline to aid the project team members responsible for sizing the hardware and software for the implementation.

The hardware requirements provided in this document can be used as a guideline; it is not a recommended or required hardware configuration for an implementation of WebCenter Forms Recognition. The actual hardware and software configuration for an installation should be based on the client's requirements.

Some factors to consider when sizing the hardware for a production environment are:

- Input volume - The number of documents/pages to process on a daily basis
- Completion Time - The required amount of time from when the document is scanned into the system, to when it is exported out of Forms Recognition
- Complexity of input documents (single or multi-page TIFF, scanned resolution, document size, number of pages OCR'ed per document, etc.)
- Output requirements (data extraction, validation, and export, number of documents processed per day, etc.)
- Complexity of workflow customization (scripting)
- Third-party software integration requirements (Oracle Financials, JD Edwards, CRM systems, etc.)
- Backup strategy
- Disaster recovery (backup, fault tolerance, up time, etc.)
- Network operating system platform
- Network environment

- Room for growth (increased in input/output and other system requirements)
- Users – The number of users (Web Verifier versus Verifier)
- Batch Retention Time – The time for a batch of documents to remain in the system after export
- Number of projects – Number of Forms Recognition projects (per country, per solution, etc.)

The following sections of the document are representations of some hardware scenario configuration.

3.3 Hardware Estimates: 500 Pages per Day

3.3.1 Introduction: AP Packaged Project

Based on the standard Oracle AP Packaged Project, the following production system is recommended.

Assumptions:

- 1 - 3 page TIFF documents
- Document resolution of 300 dpi
- Average TIFF size of approximately 40KB
- Average WorkDoc size of approximately 21KB
- Minimum data extraction and validation (as per the project)
- Project size of less than 5 MB and less than 10 classes
- Cleanup of exported batches (1 - 3 days)

Note: Disk space requirement is implementation dependent; variables such as document complexity, scanned resolution, and document volume help determine the amount of disk space for a project. For example, persistent storage of TIFF images requires approximately 70 KB per page. In addition, approximately 100 KB per document are required to temporarily store WorkDocs with OCR results. If PDF file generation is enabled, another 100 KB per page are temporarily required.

3.3.2 Hardware Estimate: Batches Stored on File System (Excl. Web Verifier)

Machine Role		Hardware	Software Needed
WebCenter Forms Recognition Server (Primary) <ul style="list-style-type: none"> ▪ File Repository ▪ OCR / Classify Extract 	Required	Dual Core Xeon Class, 2.8 GHz CPU 4 GB RAM 20 GB - 40 GB Hard Disk Space	Windows Server 2008 R2 with the latest service pack Oracle WebCenter Forms Recognition version 11g Release 1 (11.1.1)
WebCenter Forms Recognition Designer <ul style="list-style-type: none"> ▪ Project Design ▪ Class Training 	Optional	Pentium IV Class, 2.8 GHz CPU 1 GB RAM 20 GB Hard Disk Space	Windows 7 Oracle WebCenter Forms Recognition Designer version 11g Release 1 (11.1.1)
WebCenter Forms Recognition Verifier <ul style="list-style-type: none"> ▪ Data Verification ▪ Learnset Manager 	Required	Pentium IV Class, 2.8 GHz CPU 1 GB RAM 20 GB Hard Disk Space	Windows 7 Oracle WebCenter Forms Recognition Verifier version 11g Release 1 (11.1.1)
WebCenter Forms Recognition Remote Administration	Optional	Pentium IV Class, 2.8 GHz CPU 1 GB RAM	Windows 7 Oracle WebCenter Forms Recognition

▪ RTS Administration			RTS RemoteAdmin MMC Snap-in version 11g Release 1 (11.1.1)
----------------------	--	--	--

3.3.3 Hardware Estimate: Batches Stored in Database (Excl. Web Verifier)

Machine Role		Hardware	Software Needed
WebCenter Forms Recognition Server (Primary) ▪ File Repository ▪ OCR / Classify Extract Database Server	Required	Dual Core Xeon Class, 2.8 GHz CPU 4 GB RAM 100 GB Hard Disk Space	Windows Server 2008 R2 with the latest service pack Supported Oracle or SQL Server Database .Net Framework 3.5 SP1 Oracle WebCenter Forms Recognition version 11g Release 1 (11.1.1)
WebCenter Forms Recognition Designer ▪ Project Design ▪ Class Training	Optional	Pentium IV Class, 2.8 GHz CPU 1 GB RAM 20 GB Hard Disk Space	Windows 7 .Net Framework 3.5 SP1 Oracle WebCenter Forms Recognition Designer version 11g Release 1 (11.1.1)
WebCenter Forms Recognition Verifier ▪ Data Verification ▪ Learnset Manager	Required	Pentium IV Class, 2.8 GHz CPU 1 GB RAM 20 GB Hard Disk Space	Windows 7 .Net Framework 3.5 SP1 Oracle WebCenter Forms Recognition Verifier version 11g Release 1 (11.1.1)
WebCenter Forms Recognition Remote Administration ▪ RTS Administration	Optional	Pentium IV Class, 2.8 GHz CPU 1 GB RAM	Windows 7 .Net Framework 3.5 SP1 Oracle WebCenter Forms Recognition RTS RemoteAdmin MMC Snap-in version 11g Release 1 (11.1.1)

Note: The 32-bit Oracle Database Client is required on all workstations or servers where one or more WebCenter Forms Recognition components are installed, and where the Forms Recognition database resides on an Oracle Database.

3.3.4 Hardware Estimate: Entire Application Suite with Batches Stored in the Database

The entire application suite comprises WebCenter Forms Recognition Runtime Server, Web Server and database.

Machine Role		Hardware	Software Needed
WebCenter Forms Recognition Server (Primary) ▪ File Repository ▪ OCR / Classify Extract Database Server Web Server	Required	Dual Core Xeon Class, 2.8 GHz CPU 8 GB RAM 100 GB Hard Disk Space	Windows Server 2008 R2 with the latest service pack IIS 7 Supported Oracle or SQL Server Database .Net Framework 3.5 SP1 Oracle WebCenter Forms Recognition version 11g Release 1 (11.1.1)
WebCenter Forms Recognition Designer ▪ Project Design ▪ Class Training	Optional	Pentium IV Class, 2.8 GHz CPU 1 GB RAM 20 GB Hard Disk Space	Windows 7 .Net Framework 3.5 SP1 Oracle WebCenter Forms Recognition Designer version 11g Release 1 (11.1.1)

WebCenter Forms Recognition Verifier ▪ Data Verification ▪ Learnset Manager	Optional	Pentium IV Class, 2.8 GHz CPU 1 GB RAM 20 GB Hard Disk Space	Windows 7 .Net Framework 3.5 SP1 Oracle WebCenter Forms Recognition Verifier version 11g Release 1 (11.1.1)
WebCenter Forms Recognition Web Verifier	Optional	Pentium IV Class, 2.4 GHz CPU 2 GB RAM	Windows 7 Supported Browser
WebCenter Forms Recognition Remote Administration ▪ RTS Administration	Optional	Pentium IV Class, 2.8 GHz CPU 1 GB RAM	Windows 7 .Net Framework 3.5 SP1 Oracle WebCenter Forms Recognition RTS RemoteAdmin MMC Snap-in version 11g Release 1 (11.1.1)

3.3.5 Hardware Estimate: Typical Development Environment

A typical development environment consists of one server that processes less than 500 pages per day. The specification below should be sufficient to run the WebCenter Forms Recognition Runtime Server, Web Server and database in a development environment.

Machine Role		Hardware	Software Needed
WebCenter Forms Recognition Server (Primary) • File Repository • OCR / Classify Extract Database Server Web Server WebCenter Forms Recognition Verifier WebCenter Forms Recognition Designer WebCenter Forms Recognition Remote Administration	Required	Quad Core Xeon Class, 2.8 GHz CPU 8 GB RAM 100 GB Hard Disk Space	Windows Server 2008 R2 with the latest service pack IIS 7 Supported Oracle or SQL Server Database .Net Framework 3.5 SP1 Oracle WebCenter Forms Recognition version 11g Release 1 (11.1.1)

3.4 Hardware Estimates: 4,000 Pages per Day

3.4.1 Introduction: AP Packaged Project

Based on the standard Oracle AP Packaged Project, the following production system is recommended.

Assumptions:

- 1 - 3 page TIFF documents
- Document resolution of 300 dpi
- Average TIFF size of approximately 40KB
- Average WorkDoc size of approximately 21KB
- Minimum data extraction and validation (as per the project)
- Project size of less than 5 MB and less than 10 classes
- Cleanup of exported batches (1 - 3 days)
- All documents are provided at the start of the day

Note: Disk space requirement is implementation dependent; variables such as document complexity, scanned resolution, and document volume help determine the amount of disk space for a project. For example, persistent storage of TIFF images requires approximately 70 KB per page. In addition, approximately 100 KB per document are required to temporarily store WorkDocs with OCR results. If PDF file generation is enabled, another 100 KB per page are temporarily required.

3.4.2 Hardware Estimate: Batches Stored on File System (Excl. Web Verifier)

Machine Role		Hardware	Software Needed
WebCenter Forms Recognition Server (Primary) ▪ File Repository ▪ OCR / Classify Extract	Required	Quad Core Xeon Class, 2.8 GHz CPU 8 GB RAM 200 GB Hard Disk Space	Windows Server 2008 R2 with the latest service pack Oracle WebCenter Forms Recognition version 11g Release 1 (11.1.1)
WebCenter Forms Recognition Designer ▪ Project Design ▪ Class Training	Optional	Pentium IV Class, 2.8 GHz CPU 1 GB RAM 20 GB Hard Disk Space	Windows 7 Oracle WebCenter Forms Recognition Designer version 11g Release 1 (11.1.1)
WebCenter Forms Recognition Verifier ▪ Data Verification ▪ Learnset Manager	Required	Pentium IV Class, 2.8 GHz CPU 1 GB RAM 20 GB Hard Disk Space	Windows 7 Oracle WebCenter Forms Recognition Verifier version 11g Release 1 (11.1.1)
WebCenter Forms Recognition Remote Administration ▪ RTS Administration	Optional	Pentium IV Class, 2.8 GHz CPU 1 GB RAM	Windows 7 Oracle WebCenter Forms Recognition RTS RemoteAdmin MMC Snap-in version 11g Release 1 (11.1.1)

3.4.3 Hardware Estimate: Batches Stored in Database (Excl. Web Verifier)

Machine Role		Hardware	Software Needed
WebCenter Forms Recognition Server (Primary) ▪ File Repository ▪ OCR / Classify Extract Database Server	Required	Quad Core Xeon Class, 2.8 GHz CPU 8 GB RAM 100 GB Hard Disk Space	Windows Server 2008 R2 with the latest service pack Supported Oracle or SQL Server Database .Net Framework 3.5 SP1 Oracle WebCenter Forms Recognition version 11g Release 1 (11.1.1)
WebCenter Forms Recognition Designer ▪ Project Design ▪ Class Training	Optional	Pentium IV Class, 2.8 GHz CPU 1 GB RAM 20 GB Hard Disk Space	Windows 7 .Net Framework 3.5 SP1 Oracle WebCenter Forms Recognition Designer version 11g Release 1 (11.1.1)
WebCenter Forms Recognition Verifier ▪ Data Verification ▪ Learnset Manager	Required	Pentium IV Class, 2.8 GHz CPU 1 GB RAM 20 GB Hard Disk Space	Windows 7 .Net Framework 3.5 SP1 Oracle WebCenter Forms Recognition Verifier version 11g Release 1 (11.1.1)
WebCenter Forms Recognition Remote Administration ▪ RTS Administration	Optional	Pentium IV Class, 2.8 GHz CPU 1 GB RAM	Windows 7 .Net Framework 3.5 SP1 Oracle WebCenter Forms Recognition RTS RemoteAdmin MMC Snap-in version 11g Release 1 (11.1.1)

Note: The 32-bit Oracle Database Client is required on all workstations or servers where one or more WebCenter Forms Recognition components are installed, and where the Forms Recognition database resides on an Oracle Database.

3.4.4 Hardware Estimate: Entire Application Suite with Batches Stored in the Database

The entire application suite comprises WebCenter Forms Recognition Runtime Server, Web Server and database.

Machine Role		Hardware	Software Needed
WebCenter Forms Recognition Server (Primary) ▪ File Repository ▪ OCR / Classify Extract Database Server	Required	Quad Core Xeon Class, 2.8 GHz CPU 8 GB RAM 100 GB Hard Disk Space	Windows Server 2008 R2 with the latest service pack Supported Oracle or SQL Server Database .Net Framework 3.5 SP1 Oracle WebCenter Forms Recognition version 11g Release 1 (11.1.1)
WebCenter Forms Recognition Web Server	Required	Dual Core Xeon Class, 2.8 GHz CPU 4 GB RAM 50 GB Hard Disk Space 30 GB OS Hard Disk Space	Windows Server 2008 R2 with the latest service pack IIS 7 .Net Framework 3.5 SP1 Oracle WebCenter Forms Recognition version 11g Release 1 (11.1.1)
WebCenter Forms Recognition Designer ▪ Project Design ▪ Class Training	Optional	Pentium IV Class, 2.8 GHz CPU 1 GB RAM 20 GB Hard Disk Space	Windows 7 .Net Framework 3.5 SP1 Oracle WebCenter Forms Recognition Designer version 11g Release 1 (11.1.1)
WebCenter Forms Recognition Verifier ▪ Data Verification ▪ Learnset Manager	Optional	Pentium IV Class, 2.8 GHz CPU 1 GB RAM 20 GB Hard Disk Space	Windows 7 .Net Framework 3.5 SP1 Oracle WebCenter Forms Recognition Verifier version 11g Release 1 (11.1.1)
WebCenter Forms Recognition Web Verifier	Optional	Pentium IV Class, 2.4 GHz CPU 2 GB RAM	Windows 7 Supported Browser
WebCenter Forms Recognition Remote Administration ▪ RTS Administration	Optional	Pentium IV Class, 2.8 GHz CPU 1 GB RAM	Windows 7 .Net Framework 3.5 SP1 Oracle WebCenter Forms Recognition RTS RemoteAdmin MMC Snap-in version 11g Release 1 (11.1.1)

3.4.5 Web Verifier Server Hardware Sizing: Additional Information

Additional Web Server considerations depend on the number of Web Verifier users.

Number of Users	Memory	Processor
Per User	150 MB	Peak load 100% of a 3 GHz CPU
0 - 10 Users	4 GB	Dual Core CPU
11 - 20 Users	8 GB	Quad Core CPU

3.5 Infrastructure Constraints

WebCenter Forms Recognition has been tested and is certified to work in most LAN environments. By adhering to the following infrastructure constraints, you can ensure a smooth implementation of the product suite.

3.5.1 OCR Performance

OCR is a processor-intensive task. To maximize OCR performance, Oracle recommends that only one RTS OCR instance is active per CPU on a production server. Although it is possible to run multiple RTS OCR instances on a single CPU, doing so may impair the performance of OCR and the overall system.

3.5.2 Firewalls

WebCenter Forms Recognition is designed to work in a non-encapsulated LAN environment. A non-encapsulated LAN, in this context, is a LAN segment free of impediments such as firewalls and other traffic-filtering devices.

With multi-site network environments, it is the responsibility of the customer's IT personnel to ensure that an unobstructed communication path exists between the user community and host system.

3.6 Runtime Server Remote Administration MMC

You can remotely administer WebCenter Forms Recognition Runtime Server, meaning that you can centrally manage multiple Runtime Servers from a single workstation on the network using a management console snap-in called the RTS Remote Administration MMC.

However, to use the RTS Remote Administration MMC snap-in, the administrator workstation must either reside on the same LAN segment as the RTS server services to be centrally administered or, in the case of a sub-netted network, a name resolution system must be in place to allow clients on one subnet to locate resources on another subnet.

Remote Administration by MMC requires one free configurable port number. The default port is 50607.

The Windows service WebCenter Forms Recognition Runtime Service Manager must be running in order to be able to connect by MMC to the Runtime Server service. Once the service is running, it is possible to start and stop each Runtime Server instance separately.

As long as the configured port is available in any TCP/IP network (or Internet across firewalls) and the main service is running, the MMC can be used to configure and maintain the Runtime Server instances.

4 Pre-Installation of WebCenter Forms Recognition

When you are ready to install WebCenter Forms Recognition, there are several steps you should take to ensure that the installation goes smoothly.

Note: Oracle recommends engaging an appropriately trained implementation partner to ensure the successful installation of the WebCenter Forms Recognition software.

4.1 Before Installing WebCenter Forms Recognition

Before starting the installation, make sure that you have local administrator rights on the target machine. During the installation, a number of DLLs will be copied to the Windows system directory and registered with the operating system. The WebCenter Forms Recognition database will be created on the Oracle or SQL Server database servers. The install process requires administrative privileges and access to the Windows registry.

4.1.1 Installation Checklist

The checklist below is designed to help you install and configure WebCenter Forms Recognition in your environment.

- If you are installing Forms Recognition in a standalone mode (a non-network test or demo installation,) do only the steps outlined in [Section 5: Installing WebCenter Forms Recognition](#) below and skip the rest of the installation checklist.
- If you are upgrading from a previous version of Forms Recognition, read [Section 4.3: Upgrading from Previous Versions](#) below and complete the steps outlined in that section before continuing with the installation checklist.
- Read the hardware requirements in [Section 3: Hardware Requirements](#) above.
- Configure the Oracle or SQL Server database software.
- Install WebCenter Forms Recognition. See [Section 5: Installing WebCenter Forms Recognition](#) below.
- Configure the runtime components. See [Section 8: Configuring Runtime Components](#) below.
- Configure the Runtime Service Manager. See [Section 8.1: Configuring the Runtime Service Manager](#) below.
- Start the Runtime Service Manager.
- Configure the RTS RemoteAdmin MMC snap-in. See [Section 8.2: Configuring the RTS RemoteAdmin MMC Snap-in](#) below.
- Test the installation. Refer to the *WebCenter Forms Recognition Runtime Server User Guide* for further details about configuring and using the Forms Recognition Runtime Server.
- Process a batch through the Runtime Server. As a minimum, process the batch through the OCR, Classification, and Extraction workflow steps.
- If using Web Verifier, configure the IIS and .NET along with application security. See [Section 6: Configuring WebCenter Forms Recognition](#) below.

4.2 WebCenter Forms Recognition Database

WebCenter Forms Recognition version 11g Release 1 (11.1.1) is able to store the following core information directly in the Forms Recognition database, instead of the file system:

- Documents
- Batches (jobs)
- Images and document files in e-format
- Project references
- Users, groups, roles and relationships
- Web Verifier configuration
- Batch/document lock handling
- Application level user licensing

Note: File system functionality is no longer supported. Customers with file system based batches that want to upgrade to the new version will be required to upgrade to database based batches.

Prior to installation of WebCenter Forms Recognition, some care must be taken to make sure that the appropriate database preparation steps have been taken.

4.2.1 Oracle Database Checklist

WebCenter Forms Recognition will need the following items taken care of prior to the installation of the software:

1. The 32-bit Oracle Database Client libraries must be installed on all computers where Forms Recognition will be run. This includes all Runtime Server instances, and all Verifier and Designer workstations.
2. Create a new Oracle instance for Forms Recognition.
3. Create a new user with a password.
4. Assign sufficient rights to the above user:
 - a. Allow for increased growth of data.
 - b. Allow for insertion, modification, and deletion of data.
 - c. Allow for table, views, etc. creation.
5. Administrative database accounts with rights to create, modify, and delete tables. Windows Authentication can be used if the user performing the installation has administrative rights to the database server.
6. A designated user database account that will be used by Forms Recognition to access the database, add, modify, and delete data. Windows Authentication can be used if the user performing the installation has the appropriate rights to the database server.

4.2.2 SQL Server Checklist

WebCenter Forms Recognition will need to create the following items prior to the installation of the software:

1. An administrative database account with rights to create, modify, and delete tables. Windows Authentication can be used if the user performing the installation has administrative rights to the database server.
2. A designated user database account that will be used by Forms Recognition to access the database, add, modify, and delete data. Windows Authentication can be used if the user performing the installation has the appropriate rights to the database server.

4.3 Upgrading from Previous Versions

4.3.1 Upgrading from WebCenter Forms Recognition Version 11.1.1.8.0

The installation process provides a Repair option to upgrade version 11.1.1.8.0 to 11.1.1.9.0. For details refer to the *WebCenter Forms Recognition Product Migration Guide*.

4.3.2 Upgrading from Oracle Forms Recognition Version 10.1.3.5.0

Due to the transition from the file system to the WebCenter Forms Recognition database, that has to be established during the setup process, Oracle Forms Recognition version 10.1.3.5.0 must be uninstalled prior to installation of version 11g Release 1 (11.1.1).

4.4 Removing Oracle Forms Recognition Version 10.1.3.5.0

It is necessary to uninstall version 10.1.3.5.0 of Oracle Forms Recognition prior to installing WebCenter Forms Recognition 11g Release 1 (11.1.1).

The uninstaller may not remove several registry entries and subdirectories adequately. For this reason, they must be removed manually. Please follow the procedure below to properly remove the older version of Forms Recognition before installing version 11g Release 1 (11.1.1).

To remove version 10.1.3.5.0 of Oracle Forms Recognition:

1. Select **Start > Settings > Control Panel**
2. Launch the **Add/Remove Programs** wizard
3. On the **Currently Installed Programs** list, select **Oracle Forms Recognition 10g Release 3 (10.1.3.5.0)**
4. Click **Change/Remove**
5. Follow the on-screen instructions to remove the product
6. Click **Finish**
7. Remove the **C:\Program Files\Oracle\Forms Recognition** subdirectory
8. Restart the machine

Note: Forms Recognition Runtime Server Settings and Verifier Settings files can be reused in the new version. It is recommended to save these prior to un-installation and reuse them when configuring a new Runtime Server or Verifier Workstation.

5 Installing WebCenter Forms Recognition

5.1 Software Installation

Important: You must install .Net Framework 3.5 SP1 prior to installing WebCenter Forms Recognition. This remains true even if a later version of .Net Framework is already installed.

Important: Where the Forms Recognition database resides in an Oracle Database, the 32-bit Oracle Database Client must be installed prior to installing WebCenter Forms Recognition. The 32-bit Oracle Database Client is required, even if the installation is being performed on a 64-bit version of Windows.

To install WebCenter Forms Recognition:

1. Browse to the installation folder and run **setup.exe**

Note: English and German are the supported installation languages. The installer gets its language settings from the regional settings of the operating system. The installation defaults to English if a language other than English or German is detected

2. Click **Next** to continue
3. Select the installation type: **Complete** or **Custom**
 - Complete:** Installs the complete WebCenter Forms Recognition application suite including all components and all OCR engines (FineReader 10, FineReader 11, Kadmos 5 and Cleq Barcode). The default install location (..\Program Files (x86)\Oracle\WebCenter Forms Recognition) and the default program group are used
 - Custom:** Enables you to install only the components you will use, and to specify the install location and program group
4. For a complete installation choose **Complete** and click **Next** then continue to step 5 below
For a custom installation choose **Custom** and click **Next** then continue to [Section 5.1.1: Selecting Custom Installation](#).
5. Specify the desired **Program Folder**. This defines where the Forms Recognition menu shortcuts will be located within the Windows Start menu. The default location is **Start > Programs > Oracle > WebCenter Forms Recognition**
6. Click **Next**
7. Review the Current Settings. You can change any settings, if required, by using the **Back** button
8. Click **Next** to begin the complete installation and install all Forms Recognition applications and all optional components
9. Continue to [Section 5.1.2: WebCenter Forms Recognition Database Setup](#)

5.1.1 Selecting Custom Installation

If you selected **Custom** on the **Setup Type** screen, you can specify the installation directory to use and select the features and other components to be installed.

For a custom installation:

1. First, specify the installation directory. Click the **Change...** button to choose the folder where Forms Recognition will be installed
2. Click **Next**
3. In **Feature Selection** dialog box, select the desired **Applications** by checking the box beside those to be installed
4. In the **OCR Engines** list, you can select the OCR engines to be installed

Only components selected during the installation will be available. However, you can install additional components later. See [Section 5.10: Adding or Removing Forms Recognition Components](#).

Optional Components: OCR Engines	
FineReader 10	Supporting Chinese/Korean/Japanese characters in addition to English, German, Italian, French, and Spanish. Converts paper-based or scanned images into editable text
FineReader 11	The FineReader11 engine is fully integrated and supports OCR of several additional languages. The FineReader 11 engine features a number of general improvements in the quality of OCR output relative to the FineReader 10 engine: <ul style="list-style-type: none">▪ Receipt Mode,▪ Improved auto-orientation,▪ Improved OCR of amounts with leading/trailing asterisks,▪ Improved OCR of amount with leading dollar sign \$.
Cleqs Barcode	Reads handwritten and machine-printed data and barcode information. It reads 18 types of barcodes
Kadmos 5.0	Used for handwriting recognition

5. Click **Next**
6. Specify the desired **Program Folder**. This defines where the Forms Recognition menu shortcuts will be located within the Windows Start menu. The default location is **Start > Programs > Oracle > WebCenter Forms Recognition**
7. Click **Next**
8. Review the Current Settings. You can change any settings, if required, by using the **Back** button
9. Click **Next** to begin the installation and install the selected Forms Recognition applications and components
10. Continue to [Section 5.1.2: WebCenter Forms Recognition Database Setup](#)

5.1.2 WebCenter Forms Recognition Database Setup

After the selected applications and components have been installed, the **Database Setup Options** are displayed in the installation wizard.

If you want to complete the installation without performing the database setup (e.g. on an additional Runtime Server machine or a Verifier workstation), select the **Do Not Install Database** option and proceed to [Section 5.1.3: Completing the WebCenter Forms Recognition Setup](#), otherwise:

1. Select the desired database server, **Oracle** or **SQL Server**, then click **Next**
2. Enter the database username and password created in [Section 4.2: WebCenter Forms Recognition Database](#) above, then click **Next**
3. Enter the **Database Server Name** in the appropriate format for the selected database server:

Oracle Database: <database server name>:<database port>/<instance name>
e.g. databaseServer.us.oracle.com:1521/orcl

SQL Server: <database server name>

4. Click **Next**

The installation wizard will search for the specified database server, connect to it and initialize the WebCenter Forms Recognition database.

If you already have a WebCenter Forms Recognition database installed, this database will be overwritten by this installation process. In that case, a notification would be displayed to warn you of deletion:

A WebCenter Forms Recognition Database has been detected. If you continue the Database will be overwritten. It is strongly recommended that the existing Database be backed up before continuing.

If you want to save your existing database, back it up before continuing. This applies for the installation of a new version on the same machine.

If you want to keep your current installation but want to install Forms Recognition you can keep your existing database by copying all of the core configuration files from the existing installation folder to the new setup folder.

5.1.3 Completing the WebCenter Forms Recognition Setup

The final part of the installation shows the **Performed Tasks** and confirms the components that have been installed.

1. Click **Next** to continue
2. Optionally select the **Create Desktop Shortcuts for Applications** checkbox
3. Click **Finish** to complete the installation

Note: WebCenter Forms Recognition 11g Release 1 (11.1.1) supports the French language for the Verifier and Learnset Manager applications. To enable the French language, select the language on the **Formats** tab of your system's Regional and Language Options.

5.2 Manually Creating Database Objects (Post Install)

It is also possible to install the database manually. This can be due to corporate policies. In such an instance, perform the following steps to install and configure the database manually:

1. Launch Windows Explorer and navigate to the installation folder

Navigate to <installerFolderLocation>\FirstPart\Database\CreationScripts

There are two folders, **Oracle** and **SQL Server**. Each folder contains database scripts to execute that will create the tables, views, indexes, and default data values.

2. Execute the following steps appropriate to your database platform

For Oracle Database

- Follow the steps outlined in [Section 4.2.1: Oracle Database Checklist](#) above
- Using SQL*Plus or SQL Developer, log into the database schema where the Forms Recognition tables will be located
- Execute the **BrwCreateDatabase.sql** script

For SQL Server

- Follow the steps outlined in [Section 4.2.2: SQL Server Checklist](#) above
- Log into the database with Administrator rights
- Create a new database
- Execute the **BrwCreateDatabase.sql** script

3. Navigate to `<installerFolderLocation>\FirstPart\Database\UpdateScripts`

Again, you will find folders for the **Oracle** and **SQL Server** scripts.

4. Execute the following steps appropriate to your database platform

Note: If you refrain from executing the steps outlined below, an error message is generated when executing the Update Scripts.

For Oracle Database

- Edit the **BRW_Upgrade_Database.sql** script and replace all instances of **TargetDBSchemaName** with the name of the database schema created in [Section 4.2.1: Oracle Database Checklist](#) above
- Execute the **BRW_Upgrade_Database.sql** script against the Forms Recognition database schema

For SQL Server

- Edit the **BRW_Upgrade_Database.sql** script and replace all instances of **TargetDatabaseName** with the name of the database user created in [Section 4.2.2: SQL Server Checklist](#) above
- Execute the **BRW_Upgrade_Database.sql** script

5. Check that the database tables have been created correctly and no errors were reported on execution of the database scripts
6. There are several configuration components that require modification. Navigate to the WebCenter Forms Recognition installation folder. By default this is located in **C:\Program Files (x86)\Oracle\WebCenter Forms Recognition**
7. Navigate to the **WebCenter Forms Recognition Web Server** folder and open the **Web.config** file in a text editor, such as Notepad
8. Search for the connection string in the file:
`<connectionStrings></connectionStrings>`
9. Modify the connection string to connect to the database:

For Oracle Database

```

<connectionStrings>
<add name="Entities"
connectionString="metadata=res://*/Entity.ORAEntities.csdl|res://*/
/Entity.ORAEntities.ssd|res://*/Entity.ORAEntities.msl;
provider=EFOracleProvider; Provider Connection String='Data
Source=OracleServerName;User Id=Oracle;Password=Oracle'"
providerName="System.Data.EntityClient" />
</connectionStrings>

```

For SQL Server

```

<connectionStrings>
<add name="Entities"
connectionString="metadata=res://*/Entity.Entities.csdl|res://*/En
tity.Entities.ssd|res://*/Entity.Entities.msl;
provider=System.Data.SqlClient; provider connection
string=&quot;Data Source=DBINSTANCE\SQLEXPRESS; Initial
Catalog=SQLServerDatabaseCatalog;Integrated Security=false;User
ID=Oracle;Password=Oracle;MultipleActiveResultSets=True&quot;;"
providerName="System.Data.EntityClient" />
</connectionStrings>

```

10. Navigate to the **WebCenter Forms Recognition\Bin\bin** folder
11. There are 6 other configuration files that require changing as with the web.config. These are:
 - DstDsr.exe.config
 - DstHost.exe.config
 - DstSIm.exe.config
 - Brainware.System.Project.exe.config
 - DstVer.exe.config
 - DstWkBrw.exe.config

Open each one in Notepad to make the appropriate changes below.

12. Search for the connection string in the file and modify the connection string to connect to the database. Refer to Step 9 above for examples.

For Oracle Database Only

13. For Oracle Database, it is required to make one more addition to the .Net Framework installation for the Oracle connection string above to work:
 - a. Navigate to the Windows folder using Windows Explorer
 - b. Navigate to **Windows\Microsoft.NET\Framework\v2.0.50727\CONFIG**
 - c. Open the **machine.config** file for editing and locate the **DbProviderFactories** tag
 - d. Add the line indicated below on a single line in the .config file. **Do not delete any existing data:**

```

<system.data>
  <DbProviderFactories>

```

```

        <add name="EF Oracle Data Provider"
            invariant="EFOracleProvider" description="EF Provider
            for Oracle testing"
            type="EFOracleProvider.EFOracleProviderFactory,
            EFOracleProvider, Version=1.0.0.0, Culture=neutral,
            PublicKeyToken=def642f226e0e59b"/>
    </DbProviderFactories>
</system.data>

```

5.3 Configuring the WebCenter Forms Recognition Database (Post Install)

After the installation of WebCenter Forms Recognition with database, there are additional configuration steps that are required.

1. Check the project file names. The project file name will be used to display the available project lists in Web Verifier
2. Review the list of users in the projects
 - a. All usernames and passwords must be consistent throughout all project files
 - b. Each user must have their own username and password – user IDs cannot be shared
3. Export the users from the project file into the Forms Recognition database. This will now make users available to access projects via the Web Verifier.
4. Log into the User table (known in Oracle Database as **USER_**) and for each user, add a Forename and Surname into the Forms Recognition database
5. Create a Runtime Server instance for the project, or import existing Runtime Server settings, and configure again the Forms Recognition database (creating a job and linking to the Forms Recognition database).

You are now ready to use WebCenter Forms Recognition with database.

For information on how to configure a project for a WebCenter Forms Recognition instance with database, or to migrate file system batches to the Forms Recognition database, see the *WebCenter Forms Recognition Runtime Server User's Guide*.

5.4 Installing WebCenter Forms Recognition in a Workgroup

A Windows workgroup, also referred to as peer-to-peer networking, is a network implementation of Windows-based operating systems (clients and servers) in which there is no central security authority (domain controller) responsible for user authentication or a central repository (such as Active Directory) for locating network resources.

In such a network, each machine is responsible for securing its resources, and users needing access to a resource located on a machine must have an account defined on that machine. Windows workgroup networking should only be used when you want to set up communication between a limited number of computers (less than 10) and the machines are not members of a Windows domain.

For an implementation of WebCenter Forms Recognition in corporate network environments that are standardized on network operating systems (servers) other than Windows – Novell Netware – a Windows Workgroup setup may be the only choice. To install Forms Recognition in Windows Workgroup, perform the following steps (you must be logged on as an Administrator):

1. Create a user with the same name on each Forms Recognition machine. Oracle recommends **WFR RTSSvc** as the user name. (This does not apply to machines that will be used only as a Designer or Verifier workstation.)
2. Add the above user to the local Administrators group on each Forms Recognition server
3. Install WebCenter Forms Recognition on each machine by following the steps in [Section 5.1: Software Installation](#) above
4. Configure and start the Runtime Service Manager by performing the steps in [Section 8.1: Configuring the Runtime Service Manager](#) on each machine, with the following exception: Wherever a domain user is requested, add the user created in step 1 above
5. Configure the RTS Remote Administration MMC snap-in by performing the steps in [Section 8.2: Configuring the RTS RemoteAdmin MMC Snap-in](#) on each machine
6. Configure project settings on each machine and test the instance created in Step 5 above. You should (at a minimum) test the OCR, Classification, and Extraction workflow steps. For more information on how to configure project settings for a Forms Recognition RTS instance, see the *WebCenter Forms Recognition Runtime Server User's Guide*
7. Once you have successfully tested each machine, test remote communication by adding a remote machine to the local MMC snap-in of one your servers. You can accomplish this by performing a subset of the steps (Steps 4 through 7) in [Section 8.2: Configuring the RTS RemoteAdmin MMC Snap-in](#)
8. You should be able to administer the RTS instances (start, stop, change batch states, etc.) of the remote machine from the MMC snap-in of the local machine on which it was added.

5.5 Installing WebCenter Forms Recognition in Standalone Mode

WebCenter Forms Recognition can be deployed in a standalone mode. However, this implementation method is intended for test and demonstration purposes only. It should not be used in a production environment. Deployment of WebCenter Forms Recognition in Standalone mode for production purposes would be deemed unsupported by Oracle.

In this setup, all Forms Recognition components (applications, RTS, RemoteAdmin MMC snap-in, etc.) are installed and intended to be used on a single machine. Additionally, the integrated machine is not part of a Windows domain or is not intended to communicate with Forms Recognition RTS instances running on other machines in a network environment.

Using this setup, almost all of the configuration constraints (Windows domain, users and groups, etc.) can be ignored.

To install WebCenter Forms Recognition in a standalone mode:

1. Install WebCenter Forms Recognition by following the steps in [Section 5.4: Installing WebCenter Forms Recognition in a Workgroup](#)
2. Configure the RTS Remote Administration MMC snap-in by following the steps in [Section 8.2: Configuring the RTS RemoteAdmin MMC Snap-in](#) with one exception: In Step 6 of the procedure outlined, type **localhost** instead of the server name
3. Configure project settings and test the instance created in step 2 above. You should (at a minimum) test the OCR, Classification, and Extraction workflow steps

For more information on how to configure project settings for a Forms Recognition RTS instance, see the *WebCenter Forms Recognition Runtime Server User's Guide*.

5.6 Checking the Installation

The installation was successful if WebCenter Forms Recognition runs without errors.

To check for the correct installation of components:

1. Using Windows Explorer, navigate to the installation directory
2. Navigate to the **Components\Tools** subfolder
3. Run **SCBLibVersion.exe**
4. From the menu, select **View > Components General Info**. This displays a list of installed components:
5. Check the list for:
 - Completeness of components
 - Homogeneity of build numbers
 - Installation paths
6. All components (Cro*.dll, Cdr*.dll and Bwe*.dll) should have been registered automatically during the installation. If some of them seem to be missing, try to register them manually by executing the following Windows batch files:
 - **RegCro.bat** located in <Installation Folder>\Components\Cairo
 - **RegCdr.bat** located in <Installation Folder>\Components\Cedar
 - **BweReg.bat** located in <Installation Folder>\Components\Bwe

If the automatic registration does not work, try to register manually using the program **regsvr32.exe** from the Windows system directory.

If this does not help, create a copy of the components list using the command **File > Save to File** in the **Cairo and Cedar Component Version Info** utility. Submit an error report, the components list, and the log files located in the <Installation Folder>\Bin\bin\Log folder to Oracle Customer Support.

5.7 Migrating Existing Project Files

Project files designed in the earlier version of Forms Recognition must be converted to the current version format before they can be used in the current version.

The conversion process is fully automated and is done by Forms Recognition Designer. To convert project files created with earlier versions of Forms Recognition, do the following:

1. Launch the Designer application
2. Select **Load Project...** from the **File** menu
3. On the **Load Project** dialog box, browse to the project file location and double click on the project file that you want to convert
4. Login to the project as an administrative user for the project
5. Click **OK** to launch the automatic project conversion process. The conversion takes from a few seconds to a few minutes, depending on the size of the project
6. Once the conversion is completed, enter Designer's normal train mode by selecting the **Train Mode** option from the **Options** menu
7. Click the **Learn Documents** (lightbulb) toolbar button to relearn the project

8. Save the project. The project is ready for use in the current version of WebCenter Forms Recognition

Important: A project and learnset backup should always be taken before migrating project files.

For information on how to configure a project for a WebCenter Forms Recognition database, see the *WebCenter Forms Recognition Designer User's Guide*.

5.8 Removing WebCenter Forms Recognition Version 11.1.1

WebCenter Forms Recognition can be uninstalled by using the Windows Control Panel. It is important to stop all running services using the Task Manager before uninstalling the application. To remove Forms Recognition:

1. Launch the Windows Control Panel
2. Navigate to **Control Panel > Programs > Programs and Features**
3. On the **Uninstall or Change a Program** list, select **Oracle WebCenter Forms Recognition 11.1.1.9.0**
4. Click **Uninstall**
5. Follow the on-screen instructions
6. After un-installation, reboot your computer

5.9 Repairing a WebCenter Forms Recognition 11.1.1 Installation

The WebCenter Forms Recognition installer may be used to repair a copy of Forms Recognition that has stopped working properly. Factors that could cause an installation to malfunction include:

- Accidental deletion of application files
- Missing registry entries
- Corrupted application files
- Malicious attacks on a machine housing Forms Recognition

To repair WebCenter Forms Recognition:

1. Launch the Windows Control Panel
2. Navigate to **Control Panel > Programs > Programs and Features**
3. On the **Uninstall or Change a Program** list, select **Oracle WebCenter Forms Recognition 11.1.1.9.0**
4. Click **Change**
5. On the Setup dialog box, select **Repair** then click **Next**. This will reinstall all program components that were installed by the previous setup
6. Click **Finish** when setup is completed

5.10 Adding or Removing Forms Recognition Components

WebCenter Forms Recognition is a product suite consisting of the following applications:

- WebCenter Forms Recognition Runtime Server

- WebCenter Forms Recognition Designer
- WebCenter Forms Recognition Verifier
- WebCenter Forms Recognition Web Verifier

The Forms Recognition installer uses a modular approach that enables you to add or remove applications from a machine.

To modify an existing WebCenter Forms Recognition installation:

1. Launch the Windows Control Panel
2. Navigate to **Control Panel > Programs > Programs and Features**
3. On the **Uninstall or Change a Program** list, select **Oracle WebCenter Forms Recognition 11.1.1.9.0**
4. Click **Change**
5. On the Setup dialog box, select **Modify** and click **Next**
6. In the **Select Components** dialog box, select or clear the desired components
7. Click **Next**. Setup adds (if checked) or removes (if unchecked) the components
8. Click **Finish** when setup completes

5.11 Password Encryption in Core Configuration Files

The application architecture of WebCenter Forms Recognition makes it very important to be able to hide sensitive security information, such as database access passwords, stored in Forms Recognition core configuration files or custom project configuration files.

Password encryption is optional and former configuration files with unencrypted passwords will still work with no issues.

Note: The maximum number of characters allowed to encrypt is 30. Passwords longer than 30 characters would not be encrypted.

Below are the steps to encrypt the database connection password for the core Forms Recognition configuration files:

1. Open one of the Forms Recognition configuration files, for example, *<Installation Folder>\Bin\bin\DstDsr.exe.config*, in a text editor.
2. Locate the connection string and the password part of the string, example:

```
<connectionStrings>
  <add name="Entities"
    connectionString="metadata=res://*/Entity.Entities.csdl|res://*/Entity.Entities.ssdl|res://*/Entity.Entities.msl;provider=System.Data.SqlClient;provider connection string="Data Source=MYSQLSRV;Initial Catalog=DatabaseName;Integrated Security=false;User ID=alexey; Password=MyPassword;MultipleActiveResultSets=True";"
    providerName="System.Data.EntityClient" />
</connectionStrings>
```

3. Modify the password, replacing it with any number of asterisk (*) characters, example:

```
<connectionStrings>
  <add name="Entities"
    connectionString="metadata=res://*/Entity.Entities.csdl|res://*/Entity.Entities.ssdl|res://*/Entity.Entities.msl;provider=System.Data.SqlClient;provider connection string="Data Source=MYSQLSRV;Initial Catalog=DatabaseName;Integrated Security=false;User ID=alexey; Password=*****;MultipleActiveResultSets=True";"
    providerName="System.Data.EntityClient" />
```

```

        ties.ssd|res://*/Entity.Entities.msl;provider=System.Data.SqlClient;provid
er connection string=&quot;Data Source=MYSQLSRV;Initial
Catalog=DatabaseName;Integrated Security=false;User ID=alexey;
Password=*****;MultipleActiveResultSets=True&quot;;"
        providerName="System.Data.EntityClient" />
    </connectionStrings>

```

Note: The number of asterisks used is not important.

- Using the Command Line, run the `<Installation Folder>\Bin\bin\DstCrypt.exe` tool with the following arguments:

```
DstCrypt.exe /text "<password>" >> my_encrypted_password.txt
```

Note: You could add the line above to a new .bat file created in the `<Installation Folder>\Bin\bin` directory and double click on it. This should produce a new file with the name "my_encrypted_password.txt" in the same directory where the executable is located.

- Open the resulting text file. It will contain text like in the example below. Copy the section indicated in bold type in the following example that represents the encrypted password:

```

Text MyPassword encoded to
Y652CeXVdMtdNtbnBD2itCEmFFyHf9IGsN2psi6svy/MsKp8nKUgv2P7M37uu5rNo3V7wkH5795A5z6WG
ox/KEm6016AG9f1X+B5miOQg7iOgJCBqoHrsAbICHzm2EJbCkaMp1oUcvtP+8hpeJVM1BpD+QkfL1ithUX
INhWacM=

```

- Locate the **appSettings** section of the `DstDsr.exe.config` file and add the new **EncrPwd** key to this section, assigning the encrypted sequence above to the value of the key:

```

<appSettings>
    <add key="EncrPwd"
        Y652CeXVdMtdNtbnBD2itCEmFFyHf9IGsN2psi6svy/MsKp8nKUgv2P7M37uu5rNo3V7wkH579
        5A5z6WGoX/KEm6016AG9f1X+B5miOQg7iOgJCBqoHrsAbICHzm2EJbCkaMp1oUcvtP+8hpeJVM1
        BpD+QkfL1ithUXINhWacM=
    </appSettings>

```

- Save the `DstDsr.exe.config` file.
- When required, apply steps 1 - 7 to the other core configuration files, which represent different applications. These are:

- `<Installation Folder>\Bin\bin\DstHost.exe.config`
- `<Installation Folder>\Bin\bin\DstSIm.exe.config`
- `<Installation Folder>\Bin\bin\DstDsr.exe.config`
- `<Installation Folder>\Bin\bin\DstVer.exe.config`
- `<Installation Folder>\Bin\bin\Brainware.System.Project.exe.config`
- `<Installation Folder>\Oracle WebCenter Forms Recognition Web Server\web.config`

Note: Corrupted or incorrect encryption key or an incorrect password in the `web.config` file will entail a 'Login failed' error message when trying to open the Web Verifier.

5.12 AP Packaged Project INI File Encryption

WebCenter Forms Recognition allows the user to encrypt a password (or any other value) within the AP Packaged Project's INI file. RSA encryption is used, which contains a public key and a private key.

The public key can be distributed to anybody that needs to encrypt strings and store them in the project INI file, for example, a WebCenter Forms Recognition administrator. Refer to [Section 5.12.2: Project INI File Encryption for the Integrator](#) below for information about how to encrypt a password using the public key.

Public key example:

```
<RSAKeyValue><Modulus>vJ+W7SuXuvOrWVoy4tPrbflCuoHElo750cpTuEzLPk6iz6bHAodPVgLFaOEK+XMS2G5z+6961vuQsDGUt+OlAg1PiTXCa6rrAaeCaaDO4HI8Mmpw0OkUZEfCZpTTYCYQPfZlgokwomF6VDSB9dlUS430IT0gc tQY1b5iM4MqT0=</Modulus><Exponent>AQAB</Exponent></RSAKeyValue>
```

Only the project developer should know the private key. It will be coded into the project script to enable the decryption of the encrypted INI settings at runtime. Refer to [Section 5.12.1: Project INI File Encryption for the Project Developer](#) below for information about how to use the private key in project script to decrypt and use an encrypted password from the project INI file.

Private key example:

```
<RSAKeyValue><Modulus>vJ+W7SuXuvOrWVoy4tPrbflCuoHElo750cpTuEzLPk6iz6bHAodPVgLFaOEK+XMS2G5z+6961vuQsDGUt+OlAg1PiTXCa6rrAaeCaaDO4HI8Mmpw0OkUZEfCZpTTYCYQPfZlgokwomF6VDSB9dlUS430IT0gc tQY1b5iM4MqT0=</Modulus><Exponent>AQAB</Exponent><P>8SRHEvT5Bn2paRHSDR9yCQb7WGYE9PbeHzuqW6iWa0LNYJrSrhhUeCEpwlPLQWQq10KmMZgG0+Br4nuBMmMHQ==</P><Q>yD719fjB/MJWYaV3LcEzY286Q+Xvo74i6 THvHkKqB1NKYGcN9xF9d8XbiUQNgBZ/4F02T6mFeYDO32KFVRXHoQ==</Q><DP>nRDTFn7nwRmSgfRwi8minky5DQ 3IF035EIZ+x3Ao4Z52ZWkStwDz6/c12vR3XJVg7irkU0NB1zoDK1bklSw5Q==</DP><DQ>B3xieGmORva05/2ZkPpS A3ubAALojJ6FC5a0S7tOQ+vXmfdoTD45JIsfA+ipYIp2yVpyt10tC7fHBA7Y0S95QQ==</DQ><InverseQ>4S1xqlX K9f1rawGCbFwOVp6lzlfcOq8RfyDE87/G/pUilHRJV2acBAcngY3c/MRMKrXQb81x99k7dENUYc8ywQ==</Inverse Q><D>KAL6cwkCQKgbuvKFRNSLZmF0qV2JpB5kI/p1U+0GWA6QI4wnPqy+5303naOa2faPctXLSKJqvlvS21VDMUC syphvOSxBtclcZHJp4ueQPA7u+qrIJaDY1Rh1AVoqNfCJFX6+McVJ+I/X+mZOCtdUaCuAoNn014UYOaMujYDQE=</D ></RSAKeyValue>
```

Note: WebCenter Forms Recognition does not include tools for generating RSA encryption keys. The examples provided above can be used for demonstrating/testing, but it is the responsibility of the implementer to generate and provide the appropriate public and private keys in the required XML format.

5.12.1 Project INI File Encryption for the Project Developer

Where a database password is encrypted in the project INI file it is necessary to decrypt it at runtime, and then use the decrypted password in the database connection string to make the connection to the database. An example of how this would typically appear in the project INI file is:

```
SQL_VL_01_ConnectionString=Provider=OraOLEDB.Oracle.1; Persist Security Info=True;User ID=WFR;Data Source=ORCL

SQL_VL_01_ConnectionPassword=puejB5SQNCFGgwe6MRoWc1G1y7qX8xSAhgUZjhN6JolhYdKIXla7vLMU4bYmG 9V3Ayxualp/ObgXRqnSAmGsGF1FPZxktRmf58SXbnCDXmYrYgp8eS3IaqiLUPrhTiRCvfr8ZsMtK+3usmahfxpESUO Q7MZf36suWV4V3sBf9Xw=
```

Note: The **ConnectionString** setting does not contain the password. Instead, the database password is stored in its encrypted form in the **ConnectionPassword** setting.

The following script example shows how the encrypted password is retrieved from the INI file, decrypted, and then added to the connection string, resulting in a fully formed connection string that can be used to make a connection to the database through ADO.

Note: You must add a reference to the **CdrCrypt** COM object in the project script page.

```
Dim theCedarCryptographyHelper As New CdrCrypt.RSACodecInt
Dim strEncryptedPassword As String
Dim strOpenPassword As String
Dim strPrivateKey As String

strPrivateKey =
"<RSAKeyValue><Modulus>vJ+W7SuXuvOrWVoy4tPrbflCuoHElo750cpTuEzLPk6iz6bHAodPVgLFaOEK+XMMS2G
5z+6961vuQsDGUt+O1Ag1PiTXCa6rrAaeCaaDO4HI8Mmpw0OkUZEfCZpTTYCYQPfZlGokwomF6VDSB9dlUS430IT0g
ctQY1b5iM4MqT0=</Modulus><Exponent>AQAB</Exponent><P>8SRHEvT5Bn2paRHSDR9yCQb7WGYE9PbeHzuqW
H6iWa0LNYJrSrhhUeCEpwlPLQWQq10KmMZgG0+Br4nuBmMHQ==</P><Q>yD7l9fjB/MJWYaV3LcEzY286Q+Xvo74i
6THvHkKqB1NKYGcN9xF9d8XbiUQNgBZ/4F02T6mFeYD032KFVRXHoQ==</Q><DP>nRDTFn7nwRmSgfRwi8minkyk5D
Q3IF035EIZ+x3Ao4Z52ZWkStwDz6/c12vR3XJVg7irkU0NB1zodK1bklSw5Q==</DP><DQ>B3xieGmORva05/2ZkPp
SA3ubAAL0jJ6FC5a0S7tOQ+vXMfdoTD45JIsfA+ipYIp2yVpyt1OtC7fHBA7Y0S95QQ==</DQ><InverseQ>4Slxql
XK9f1rawGCbFW0Vp6lzlfcQ08RfyDE87/G/pUilHRJV2acBAcngY3c/MRMKrXQb81x99k7dENUYc8ywQ==</Invers
eQ><D>KAL6cwkCQKgbuvKFRNSLZmFOqV2JpB5kI/plU+0GWas6Qi4wnPgy+53O3naOa2faPctXLSKJqvlvSz21VDMU
CsyphvOSxBtc1cZHJp4ueQPA7u+qrIJaDY1RhIAVoqNfCJFX6+McVJ+I/X+mZOCtdUaCuAoNn014UYOaMujYDQE=</
D></RSAKeyValue>"

strEncryptedPassword = DicVal("01" & "ConnectionPassword", "SQL")

If Len(strEncryptedPassword) > 0 Then
    strOpenPassword = theCedarCryptographyHelper.Decode(strEncryptedPassword, strPrivateKey)
End If

If Len(strOpenPassword) > 0 Then
    strConnection = strConnection + ";Password=" + strOpenPassword
End If
```

5.12.2 Project INI File Encryption for the Integrator

As an implementer or WebCenter Forms Recognition administrator, you simply need encrypt (for example) the database password and add the encrypted value to the project INI file. To encrypt a value, such as the database password:

1. Open the Windows Command Line
2. Navigate to the *<Installation Folder>\Bin\bin* directory in the Command Line
3. Execute the following command, replacing the *<myPassword>* and *<publicKey>* placeholders with the actual values for your environment:

```
DstCrypt.exe /text <myPassword> /key "<publicKey>" >> <output text file name>
```

For example:

```
DstCrypt.exe /text MyPassword /key
"<RSAKeyValue><Modulus>vJ+W7SuXuvOrWVoy4tPrbflCuoHElo750cpTuEzLPk6iz6bHAodPVgLFaOE
K+XMMS2G5z+6961vuQsDGUt+O1Ag1PiTXCa6rrAaeCaaDO4HI8Mmpw0OkUZEfCZpTTYCYQPfZlGokwomF6
VDSB9dlUS430IT0gctQY1b5iM4MqT0=</Modulus><Exponent>AQAB</Exponent></RSAKeyValue>"
>> my_encrypted_password.txt
```

Note: Notice that the public key value is contained in quotes but the password is not.

The text file specified in the command (e.g. *my_encrypted_password.txt*) will now contain the encrypted text string for the password.

4. Add the encrypted password to the **ConnectionPassword** setting in the project INI file. For example:

SQL_VL_01_ConnectionPassword=puejB5SQNCFGgwe6MRoWc1G1y7qX8xSAhgUZjhN6JolhYdKIx1a7v
 LMU4bYmG9V3Ayxualp/ObgXRqnSAmGsGF1FPZXktRmf58SXbnCDXmYrYgp8eS3IaqiLUPrhTiRCvfr8ZsM
 tK+3usmahfxpESUOQ7Mzf36suWV4V3sBf9Xw=

Note: Remember to remove the **Password=<database password>**; component from the setting in the corresponding **ConnectionString** setting.

5.13 Silent Installations

A silent install mode is provided for situations where the same configuration of WebCenter Forms Recognition is to be installed on several machines, for example, Verifier workstations. The use of a configuration file removes the necessity to go through the installation dialog on each machine.

5.13.1 Silent Install.ini

The configuration settings for the silent installation are read from the **Silent Install.ini** file in the WebCenter Forms Recognition installation directory. The directory contains an example file, which must be edited before performing a silent installation.

The file contains six sections:

- General
- Applications
- OCR Engines
- Additional
- Database Configuration
- DB Credentials

It is allowed to delete single entries or complete sections.

However, it is not allowed to use options without the section name. If any information is deleted from the **Silent Install.ini** file, the Setup uses the default values as described.

Name	Description
[General]	Determines how and where WebCenter Forms Recognition is to be installed.
EULA	The end-user agreement needs to be accepted on each machine where the software is installed. If this section is not modified in the silent install.ini file, the silent installation would be canceled. 0: Accepted 1: Not accepted
Path	Indicates where the application should be installed. The pathname should not have a final backslash. Example: Path = "C:\Program Files (x86)\Oracle"
MoveComponentsIfRequired	If an older version of the application is installed, this indicates whether to use the existing component folder or whether to move the old components into the new directory prior to installation. 0: Use existing component folder. 1: Move components to the new path.
CreateDesktopIcons	0: Don't create desktop shortcuts. 1: Create desktop shortcuts.
InstallWibuKey	0: Skip Wibukey driver installation.

	1: Install Wibukey drivers.
StopIfDotNetIsNotFound	0: If .Net Framework 3.5 SP1 is not found on the system the installation proceeds. The Features (WebVerifier, Database Connection...) will not be installed. 1 (DEFAULT): If .Net Framework 3.5 SP1 is not found on the system the installation will be aborted.
[Applications]	Defines which applications are to be installed. Note that it is permissible to skip all applications if, for example, only the extraction components are to be installed.
Designer	0: Skip installation of the Designer application. 1: Install the Designer application.
Verifier	0: Skip installation of the Verifier application. 1: Install the Verifier application.
Runtime Service	0: Skip installation of the Runtime Server application. 1: Install the Runtime Server application.
Web Verifier	0: To skip installation of Web Verifier application. 1 (DEFAULT): Install Web Verifier application. See also option StopIfDotNetIsNotFound
[OCR Engines]	Defines which OCR engines are to be installed. It is permissible to skip all the engines.
FineReader10	0: Skip installation of ABBYY FineReader 10. 1: Install ABBYY FineReader 10.
FineReader11	0: Skip installation of ABBYY FineReader 10. 1: Install ABBYY FineReader 10.
Kadmos5	0: Skip installation of Kadmos 5 engine. 1: Install Kadmos 5 engine.
Cleqs	0: Skip installation of Cleqs engine. 1: Install Cleqs engine.
[Additional]	Additional files to install.
Demo Files	0: Skip installation of the demo project files. 1: Install demo project files.
[Database Configuration]	Configures an existing database server. See also option StopIfDotNetIsNotFound .
DBServerType	1: SQL Server database will be configured. 2: Oracle Server database will be configured. 3 (DEFAULT): No database will be configured.
If there is any wrong information for the following options, DBServerType will be set to 3.	
UseDBConfnfiFile	Text file name that contains Database Connection String. If this option is empty, credentials will be taken from [DB Credentials] section. If there is neither a configuration file nor a [DB Credentials] section, DBServerType will be set to 3 (no database) internally. (DEFAULT empty string).

[DB Credentials]	This section can be used instead of a configuration file. (DEFAULT database configuration will be skipped, if option UseDBConfIniFile is not set)
Only for SQL Server Usage (See Option DBServerType and UseDBConfIniFile)	
SQLServerWindowsAuthent	0 (DEFAULT): No Windows Authentication will be used for DBA. 1: Windows Authentication will be used for DBA.
SQLServerAdminUser	DBA account name (See also option SQLServerWindowsAuthent). (DEFAULT empty string).
SQLServerAdminPassword	DBA account password (See also option SQLServerWindowsAuthent) (DEFAULT empty string).
For both Database Server Types (See Option DBServerType and UseDBConfIniFile)	
DBUserWindowsAuthent	0 (DEFAULT): No Windows Authentication will be used for DB user. 1: Windows Authentication will be used for DB user.
DBUserName	DB user account name (See also option DBUserWindowsAuthent) (DEFAULT empty string).
DBUserPassword	DB user account password (See also option DBUserWindowsAuthent) (DEFAULT empty string).
DatabaseServerPath	Name of the database. Usually it is specified as: For Oracle Database: <database server>:<database port>/<instance name> For SQL Server: <MachineName>\<InstanceName> (DEFAULT empty string).

5.14 Preparing Internet Information Server (Post Install)

One of the preconditions for working with WebCenter Forms Recognition Web Verifier is the installation of the Internet Information Server. Windows Server 2008 R2 works with IIS 7. Windows Server 2012 R2 works with IIS 8.5.

Perform the installation of the IIS application. Use the following links for more information:

For Windows Server 2008 R2

<http://learn.iis.net/page.aspx/29/installing-iis-7-on-windows-server-2008-or-windows-server-2008-r2/>

For Windows Server 2012 R2

<http://www.iis.net/learn/install/installing-iis-85/installing-iis-85-on-windows-server-2012-r2>

5.14.1 Configuring IIS 7 or IIS 8

The Internet Information Server (IIS) is use for executing the WebCenter Forms Recognition Web Verifier. To configure II 7 or IIS 8 for as the Forms Recognition Web Server:

1. Launch the IIS Manager (**Start** > **Administrative Tools** > **Internet Information Services (IIS) Manager**)
2. Expand the <machine name> node, then expand the **Sites** node in the navigation tree
3. Right-click **Default Web Site**

4. Select the **Add Application...** menu item
5. In the dialog window, enter the **Alias** you want to use to gain access to this Web virtual directory. Oracle recommends setting the alias as **WebVerifier**.
6. Set the **Physical path** to `<Installation Folder>\WebCenter Forms Recognition Web Server`
7. Click **OK**
8. Double click the **Default Document** icon for the WebVerifier application
9. Click **Add...** in the **Actions** menu
10. Enter **Login.aspx** in the resulting dialog, then click **OK**
11. Right-click on the **WebVerifier** node and choose **Add Virtual Directory...**
12. In the dialog box, set the **Alias** to **WL**
13. Set the **Physical Path** to `<Installation Folder>\Components\Cedar\WL`
14. Click **OK** to close the dialog
15. Select the **Application Pools** node in the navigation tree
16. Select **DefaultAppPool** from the list of application pools, then click **Advanced Settings...** in the **Actions** menu
17. Ensure the all of the following settings are as follows:
 - Enable 32-Bit Applications: **True**
 - Managed Pipeline Mode: **Integrated**
 - Identity: **NetworkService**
18. Click **OK**
19. Using Windows Explorer, create the following three subfolders:
 - `..\Windows\Temp\Bwe`
 - `..\Windows\Temp\CdrDBCACHE`
 - `..\Windows\Temp\DIST`
20. For each of the subfolders created in the previous step, ensure that the **NETWORK SERVICE** user specified in the application pool is assigned the **Full Control** permission.

To complete the configuration for Web Verifier, Data Execution Prevention (DEP) should be disabled. To disable DEP, execute the following command in the Command Line:

```
bcdedit.exe /set {current} nx AlwaysOff
```

Finally, restart the server. After the server has restarted, the WebCenter Forms Recognition Web Verifier application will be accessible by the address <http://localhost/WebVerifier/login.aspx>

5.15 Running Multiple Web Verifier and RTS Instances

5.15.1.1 Problem Description

When running more than approximately 12 concurrent Web Verifier user sessions, or more than approximately 14 Runtime Service instances, the system may start experiencing lack of Windows desktop heap resources and the extra user sessions or RTS instances can fail with different internal memory allocation errors.

5.15.1.2 Problem Cause

The default Windows OS setting of desktop heap size for the non-interactive Windows station often appears to be too low to host multiple simultaneously running Web Verifier or Runtime Service instances with extensive script engine utilization.

5.15.1.3 Recommended Configuration Changes

1. Open Windows Registry Editor

2. Browse to the key:

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Session  
Manager\SubSystems\Windows]
```

3. Check the third argument of the **SharedSection** parameter:

```
SharedSection=1024, 3072, 512
```

512 (KB) is the default value that causes the issue as described in the problem description above.

Note that for some OS versions, this default setting can be different. The table below gives the recommended values for this setting:

Number of RTS and/or Web Verifier Instances*	Desktop Heap Size in KB
1 - 10	512
11 - 24	1024
25 - 36	1536
37 - 48	2048
49 - 60	2560
61 - 72	3072

* This represents the total number of simultaneously running Runtime Server instances **plus** the number of concurrent Web Verifier users running on the same physical server. For example, a server running 4 RTS instances with 8 concurrent Web Verifier users connected would total 12, so the "11 - 24" setting would be observed.

4. After modifying this parameter restart the server

6 Configuring WebCenter Forms Recognition

6.1 Configuring Applications

There are some main configuration parameters to be accounted for. See [Appendix A: Web.config Options and Associated Resource File Parameters](#) for more information.

6.1.1 Configuring the Oracle WebCenter Forms Recognition Database Connection String

1. Open the application configuration file `<Installation Folder>\WebCenter Forms Recognition Web Server\web.config` in a text editor, such as Notepad
2. Find the following string:

```
<connectionStrings>

<add name="Entities"
connectionString="metadata=res://*/Entity.Entites.csdl|res://*/Entity.Entites.ssdl
|res://*/Entity.Entites.msl;provider=System.Data.SqlClient;provider connection
string=&quot;Data Source=NEO\SQLSERVER2005;Initial
Catalog=Oracle_verifier_work;Integrated Security=false;User ID=developer;
Password=123456;MultipleActiveResultSets=True&quot;;"
providerName="System.Data.EntityClient" />
```

3. Modify the connection string in accordance with your database settings
4. Replace the connection string in the `Brainware.System.Project.exe` config file with the one configured within the `web.config` file

Note: These two connection string entries must be identical in order to assure the availability of all Web Verifier functionalities associated with the Knowledge base.

6.1.2 Enabling HTTP Compression for IIS 7

In order to enhance your application performance and to save server costs, it is recommended you enable HTTP compression in the context of the Internet Information Services.

Refer to the following link for detailed information:

<http://technet.microsoft.com/en-us/library/cc771003.aspx>

6.2 Server Security Configuration

6.2.1 Registering COM Components

After applying a patch, locate and run the **Register Web Server.bat** as an administrator. It is located in the `<Installation Folder>\WebCenter Forms Recognition Web Server\bin` folder. For registering this component:

1. Right-click on the **Register Web Server.bat** file
2. Select **Open** from the context menu

6.2.2 Preparing the User Context (SQL Server)

It is necessary for the user of the user context in which the Web Verifier is running in IIS to have the proper rights to access the SQL Server database. By default, the Web Verifier runs under the NETWORK SERVICE user context, hence the same should be allowed to access the database.

If you selected **Windows Authentication** during the installation of WebCenter Forms Recognition, you will need to add the domain username to the SQL Server database additionally to the NT AUTHORITY\NETWORK SERVICE.

Steps to add NETWORK SERVICE to SQL server:

1. Open Microsoft SQL Server Management Studio
2. Expand the local computer name, select **Security** > **Logins**
3. Right click **Logins**, select **New Login**
4. On **Login Properties**, under **General**, click **Search**
5. Enter **NETWORK SERVICE** and then click **Check Names**
6. Click **OK**
7. Select **sysadmin** (*public* is selected by default) for **Server Roles**
8. Click **OK**
9. The **NT AUTHORITY \NETWORK SERVICE** has been added to SQL server.

After adding Network Service to SQL server, make sure that the IIS is running under NT AUTHORITY \NETWORK SERVICE by opening the IIS Manager:

1. Click **Start**, and then select **Control Panel**
2. Select **Administrative Tools**, and then click **Internet Information Services (IIS) Manager**
3. In the **Connections** panel, expand the server node and click **Application Pools**
4. On **Application Pools**, select the application pool that you want to specify an identity, and then click **Advanced Settings** in the **Actions** panel
5. For the identity property, the built-in account should be **NetworkService**
6. If it does not contain NetworkService, click **Set...** to open the **Application Pool Identity** dialog box
7. Select the **Built-in account** option and select **NetworkService** account from the list

6.2.3 Setting Permissions for Forms Recognition Projects Execution

All WebCenter Forms Recognition projects are located in a file system folder. The Web Verifier sources this path from the Forms Recognition database. Forms Recognition projects are loaded by the **Brainware.System.Project.exe** process. This process cannot load the projects until it has the appropriate permissions for the projects folder. In this case, it is necessary to grant permission to the **Network Service** Windows user for this folder by performing the following steps:

1. Using Windows Explorer, navigate to the projects folder
2. Right-click on the folder and select **Properties**
3. In the dialog window, select the **Security** tab
4. Add the **Network Service** user to the list

6.2.4 Encrypting Sections with the aspnet_regiis Tool

If you want to protect the data stored in the configuration file, perform the following steps:

6.2.4.1 Pre-configuring:

1. Find the **Brainware.System.AppConfiguration.dll** file in the *<Installation Folder>\Bin\bin* directory
2. Register this assembly in the GAC using the command:
gacutil -I Brainware.System.AppConfiguration.dll

6.2.4.2 Encryption of the web.config file:

1. Use the aspnet_regiis command-line tool. This tool is located at:
%WinDir%\Microsoft.NET\Framework\v2.0.50727\aspnet_regiis.exe
2. For encrypting a particular section of the configuration file, you can use the **-pe** option when executing the aspnet_regiis tool

For example, for encryption of the **connectionStrings** section use:

```
aspnet_regiis -pe connectionStrings -app/MyApp
```

Note: The **-app** option is used to specify the application's virtual path.

6.2.4.3 Decryption of the web.config file:

1. For decryption of a configuration section, use the following command:
aspnet_regiis -pd connectionStrings -app/MyApp

6.3 Client Security Configuration

This section describes security configuration for the Web Verifier client.

1. In Internet Explorer, select **Tools > Internet Options**
2. Select the **Security** tab, and then click **Custom Level...**
3. Locate **ActiveX controls and plug-ins** in the list and ensure the following options are enabled:
 - Binary and script behaviors
 - Run ActiveX controls and plug-ins
4. Locate **Scripting** in the list and ensure the following options are enabled:
 - Active scripting
 - Allow status bar updates via script (*Optional. See note below*)
5. Click **OK** to save the browser settings

Note: The **Allow status bar updates via script** option is required to display information about batches, documents, current filters and the page number in the Internet Explorer status bar. The Web Verifier client will function normally without this option enabled, but batch, document, filter and page information will not be visible in the Internet Explorer status bar.

6.4 Configuring Single Sign-On Authentication for Web Verifier

Web Verifier has been enhanced to support Single Sign-On (SSO) user authentication. The SSO implementation will intercept the user's login request and either gather the user credentials and authenticate the user, or accept the user as already authenticated.

The SSO functionality is implemented as a generic solution to work with any SSO implementation and configuration that provides the user credential information via an HTTP header.

Note: Configuration of the SSO service provider is out of scope of this document. Refer to the provider's product documentation.

6.4.1 Enabling Single Sign-On Authentication

To enable external SSO authentication for Web Verifier, the following changes are required in the `web.config` file:

1. Check the `<verifier.webclient>` section for the `<httpHeaderBasedSso>` parameter.
2. Set the flag for the `enabled` setting to `true`:

```
<httpHeaderBasedSso loginHeader="idUserName" enabled="true"
  sessionHeader="ShibSessionID" />
```

3. Set the `loginHeader` value to the HTTP header attribute name which is returned by the SSO service. For example:

```
loginHeader="remoteuser"
```

4. Set the `sessionHeader` parameter to the default session ID header used by the SSO provider.

6.4.2 SSO Session vs. Web Verifier Client Session

There are two different sessions to consider when SSO is used; the SSO session and the Web Verifier Client (WVC) session. The SSO session generally has longer life as it might be used for other applications and not only for WVC.

The WVC and SSO sessions are both renewed with every server request, for example, field validation or opening the next batch. They are not renewed with client-side actions such as zooming in on the image or typing a value into form field without validating it.

Note: When SSO authentication mode is used, the WVC session timeout must be set to a smaller value than the SSO session timeout. Otherwise, if the SSO session times out while the WVC session is still active, the user will be re-directed to the SSO login page, and any changes made in Web Verifier will be lost. To achieve this, the following parameter must be set appropriately:

```
<sessionState timeout="20"/>
```

Refer to the product documentation of your SSO provider for details how to configure the SSO session duration.

6.5 Configuring Windows Authentication for Web Verifier

The Web Verifier application allows you to login with your Windows user account. In this case, the password that is shared with Windows will be used to login into Web Verifier.

To use this option, you first need to configure the server.

Note: Only Windows Authentication access will be possible after this option is configured. However, when logged in to Web Verifier via Windows Authentication, it will be possible to use the re-login menu option to login e.g. as an administrator in order to perform certain administrative tasks.

6.5.1 Prerequisites

- Before starting to configure IIS, make sure that the Web Verifier application is working properly using an existing project user account
- Back up the web.config file

6.5.2 Configuration for IIS 7

To configure Windows Authentication access to Web Verifier with IIS 7:

1. Open **Authentication** settings in IIS group for the WebVerifier application
2. Enable **Windows Authentication** and disable all other authentication methods
3. Close all of the running browser sessions prior to access the WebVerifier application
4. Add the Windows user to the database. Refer to the *WebCenter Forms Recognition Designer User Guide* for information on how to do this

6.5.3 Changes to the Web.config File

It is highly recommended to have two versions of the web.config file; one for standard authentication and one for Windows Authentication. This will simplify switching between modes.

The following list shows required steps to convert standard web.config to a web.config with Windows Authentication enabled:

1. Make a backup copy of the existing web.config file
2. Change **<authentication>** section (located in the <configuration><system.web>) to the following:

```
<authentication mode="Windows">
```

3. Remove the following line:

```
<forms loginUrl="Login.aspx" returnUrl="BatchView.aspx" />
```

This is a child of the <authentication> section, and is only needed for standard authentication.

4. Change **<authorization>** section (located in the <configuration><system.web>) from **deny** to **allow**:

```
<authorization>  
  <allow users="?" />  
</authorization>
```

5. Add the **enableSessionState** attribute to **<pages>** section (located in the **<configuration>** **<system.web>**):

```
<pages enableSessionState="true">
```

6. Remove all **<location>** sections (located in the **<configuration>** section right before **<appSettings>**). Those sections look like the following:

```
<location path="WL">
  <system.web>
    <authorization>
      <allow users="*" />
    </authorization>
  </system.web>
</location>
```

6.5.4 Reverting Back to Standard Authentication

To switch from Windows Authentication mode back to standard authentication mode, the following adjustments to IIS are required:

1. In Internet Information Services (IIS) Manager, open **Authentication** settings in IIS group for the WebVerifier application
2. Disable **Windows Authentication** and enable both, **Anonymous Authentication** and **Forms Authentication**
3. Replace the web.config file with the backup made prior to enabling Windows authentication for Web Verfier

6.6 Configuring SSL for Web Verifier

For information how to set up SSL on your Internet Information Server (IIS) machine, refer to the Microsoft Support article *How to Implement SSL in IIS* (<http://support.microsoft.com/kb/299875>).

6.7 Configuring Additional Languages

Web Verifier supports an extended list of languages:

Chinese Simplified, Chinese Traditional, Danish, Dutch, English, French, Finnish, German, Italian, Japanese, Korean, Norwegian, Polish, Portuguese, Romanian, Russian, Spanish, Swedish, Turkish.

Note: By default, the Chinese language selection points to Simplified Chinese. In case Traditional Chinese language has to be used, the content of the Bin\Resources\cmn folder can be copied into the "zho" folder, which contains the Simplified Chinese. Prior to overwriting the "zho" folder content, please **back it up**. In order to display the correct language on the drop down list of the language selection menu, a web.config change for the parameter LanguageDisplayName is needed. See [Appendix A](#).

6.8 Virus Check

Please note that the settings for the virus checker on the Web Server exclude the [Local Temp Folder]/CdrDbCache directory (Batch and the Common Learnset folders) from the locations which are checked for viruses. This is due to performance considerations.

6.9 Enabling New Columns for Batch View

Four additional columns are available to hold additional information on batches:

- **Batch.ExternalGroupId** - default display name: "User Group"
data type: The Group ID which has been assigned to a batch is relating to security. Batches can be assigned to user group via a unique ID. For example, German invoices belong to Group 1 and English invoices belong to Group 2. When in a shared service center, you could hide all German invoice batches from English Verifiers.
- **Batch.ExternalBatchId** - default display name: "Batch Group"
data type: It allows the developer to uniquely identify the batch. For example, external system ID, storage box ID, etc.
- **Batch.TransactionId** - default display name: "Transaction"
data type: It allows the developer to synchronize a newly created batch of documents with another external system. It can be used to identify originators of batch of documents.
- **Batch.TransactionType** - default display name: "Transaction Type"
data type: It allows the developer to synchronize a newly created batch of documents with another external system. It can be used to identify the types of documents (Invoices, Claim forms etc.) in batches or source of document (Email, Scanned etc.)

These table columns are not project or application specific and therefore cannot be configured in Designer or Verifier or RTS applications.

By default, these columns will be invisible. To configure the columns' visibility for Web Verifier, adjust the batch columns' attributes in the `batchColumnVisibility` section of the `web.config` file appropriately as described in [Appendix A](#).

The values of the columns can only be set via the Project Script (PostimportBatch). Check the SQL scripts in the installation folder to activate the displaying of those columns. After enabling one or all of the additional columns in database, it applies to all application modules.

The additional columns can be enabled with columns customized.

For Oracle Database

Syntax:

```
exec sp_SetGlobalApplicationSetting('<column setting name>', '<column name to display>', <enabled>)
```

Examples:

```
exec sp_SetGlobalApplicationSetting('SysAppBatchColumnExternalGroupId', 'User Group', 1)
exec sp_SetGlobalApplicationSetting('SysAppBatchColumnExternalBatchId', 'Batch Group', 1)
exec sp_SetGlobalApplicationSetting('SysAppBatchColumnTransactionId', 'Transaction', 1)
exec sp_SetGlobalApplicationSetting('SysAppBatchColumnTransactionType', 'Transaction Type', 1)
```

For SQL Server

Syntax:

```
exec sp_SetGlobalApplicationSetting '<column setting name>', '<column name to display>', <enabled>
```

Examples:

```

exec sp_SetGlobalApplicationSetting 'SysAppBatchColumnExternalGroupId', 'User Group', True
exec sp_SetGlobalApplicationSetting 'SysAppBatchColumnExternalBatchId', 'Batch Group',
True
exec sp_SetGlobalApplicationSetting 'SysAppBatchColumnTransactionId', 'Transaction', True
exec sp_SetGlobalApplicationSetting 'SysAppBatchColumnTransactionType', 'Transaction
Type', True

```

Note: For setting up the Group ID column, due to the security control, make sure the group ID is matching with the ID created for the users.

6.10 Changing Custom Column Names

After you have enabled new custom columns following the instructions in [Section 6.9 Enabling New Columns for Batch View](#), you may want to give them more meaningful names.

6.10.1 Custom Column Names for Web Verifier

To change the custom column names for the Web Verifier application:

1. Using Windows Explorer, navigate to *<Installation Folder>\WebCenter Forms Recognition Web Server\Bin\Resources\eng*
2. Open the file **Brainware.Verifier.WebClient.resx** in a text editor such as Notepad
3. Change the name of the four items below by adjusting the `<value>` parameter as indicated in bold type:

```

<data name="TEXT_EXTERNALBATCH_NAME" xml:space="preserve">
  <value>External Batch ID</value>
</data>

<data name="TEXT_EXTERNAL_GROUP_ID" xml:space="preserve">
  <value>User Group</value>
</data>

<data name="TEXT_TRANSACTION_ID" xml:space="preserve">
  <value>Transaction ID</value>
</data>

<data name="TEXT_TRANSACTION_TYPE" xml:space="preserve">
  <value>Transaction Type</value>
</data>

```

For the other application languages, repeat the steps 1 - 3 above using the **Brainware.Verifier.WebClient.resx** file from the appropriate folder under *<Installation Folder>\WebCenter Forms Recognition Web Server\Bin\Resources*

6.10.2 Custom Column Names for the Verifier Application

For the Verifier client application, custom column names can be changed via SQL Script. Execute the below mentioned script for each column name you want to change:

For Oracle Database

Syntax:

```

exec sp_SetGlobalApplicationSetting('<column setting name>', '<column name to display>', <enabled>)

```

Examples:

```
exec sp_SetGlobalApplicationSetting('SysAppBatchColumnExternalGroupId', 'User Group', 1)
exec sp_SetGlobalApplicationSetting('SysAppBatchColumnExternalBatchId', 'Batch Group', 1)
exec sp_SetGlobalApplicationSetting('SysAppBatchColumnTransactionId', 'Transaction', 1)
exec sp_SetGlobalApplicationSetting('SysAppBatchColumnTransactionType', 'Transaction
Type', 1)
```

For SQL Server

Syntax:

```
exec sp_SetGlobalApplicationSetting '<column setting name>', '<column name to display>', <enabled>
```

Examples:

```
exec sp_SetGlobalApplicationSetting 'SysAppBatchColumnExternalGroupId', 'User Group', True
exec sp_SetGlobalApplicationSetting 'SysAppBatchColumnExternalBatchId', 'Batch Group',
True
exec sp_SetGlobalApplicationSetting 'SysAppBatchColumnTransactionId', 'Transaction', True
exec sp_SetGlobalApplicationSetting 'SysAppBatchColumnTransactionType', 'Transaction
Type', True
```

6.11 Disabling (and Enabling) Batch Deletion

By default, it is possible to delete batches using the Management Console and through the Designer application. You can disable batch deletion through either or both of these applications by executing the following SQL command against the Forms Recognition database:

For Oracle Database

To disable batch deletion in Designer:

```
exec sp_SetGlobalApplicationSetting('SysAppBatchDeletionDisabledInDesigner', 'True', 1)
```

To enable batch deletion in Designer:

```
exec sp_SetGlobalApplicationSetting('SysAppBatchDeletionDisabledInDesigner', 'True', 0)
```

To disable batch deletion in the Management Console:

```
exec sp_SetGlobalApplicationSetting('SysAppBatchDeletionDisabledInRTS', 'True', 1)
```

To enable batch deletion in the Management Console:

```
exec sp_SetGlobalApplicationSetting('SysAppBatchDeletionDisabledInRTS', 'True', 0)
```

For SQL Server

To disable batch deletion in Designer:

```
exec sp_SetGlobalApplicationSetting 'SysAppBatchDeletionDisabledInDesigner', 'True', True
```

To enable batch deletion in Designer:

```
exec sp_SetGlobalApplicationSetting 'SysAppBatchDeletionDisabledInDesigner', 'True', False
```

To disable batch deletion in the Management Console:

```
exec sp_SetGlobalApplicationSetting 'SysAppBatchDeletionDisabledInRTS', 'True', True
```

To enable batch deletion in the Management Console:

```
exec sp_SetGlobalApplicationSetting 'SysAppBatchDeletionDisabledInRTS', 'True', False
```

Note: These settings will affect all users. You must restart Designer or the Management Console (as appropriate) for the settings to take effect.

7 Security

7.1 WebCenter Forms Recognition Security

7.1.1 Project Security

WebCenter Forms Recognition contains an internal application security model with 6 project security roles:

- **Administrator (ADM):** The Administrator's 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 roles because it is the highest role within Forms Recognition.
- **Learnset Manager (LSM):** The Learnset Manager role is used to define, modify, and maintain the global project learnset. Reference WebCenter Forms Recognition Designer and Verifier documentation for additional information.
- **Supervised Learning Verifier (SLV):** The Supervised Learning Verifier role is to collect and manage the local training data. These verifiers are subject-matter experts who can propose learnset improvements.
- **Verifier User (VER):** The Verifier role is to make document correction that could not be automatically processed. Typically, members of this group are data correction users.
- **Setting Role (SET):** The Verifier Settings role is used to give permission to the Verifier (VER) to alter or access the verifier configuration settings.
- **Filtering Role (FLT):** The filtering role is to allow Verifier user to configure custom filtering of batches. By application design, FLT users would be able to use the filtering feature even if they do not have the SET role. This solution provides more flexibility and security.

Additional information on adding users and groups can be found in the Designer documentation.

7.1.2 Project and Windows Authentication Security

Local workstations and network Windows users can be imported into the Forms Recognition project authentication sub-system, allowing automatic project authentication with the currently logged in Windows user account for all WebCenter Forms Recognition applications.

To enable Windows based authentication in Forms Recognition for a desired project:

1. Open the project file in the Forms Recognition Designer application
2. Select the **Options > Users, Groups and Accounts...** menu item
3. Check the **Allow Windows Authentication** check-box on the **Users** tab of the **Project Authentication Settings** dialog

When using database authentication in Verifier, access rights have to be granted to the verifier user. By default, this should be:

- Read and write to all database tables
- Execute rights to stored procedures and functions

7.2 File System Security

Although WebCenter Forms Recognition does provide application-level security, the product relies on integrated Windows file system security built into the underlying operating system for file system access.

Forms Recognition uses operating system files (.sdp, .dat, .wdc, .sdb, etc.) to store all application and project data. A combination of shared and NTFS permissions are used to protect application data.

NTFS file and folder permissions are used to control the type of access that a user, group, or application has to folders and files. This includes everything from reading the contents of a folder or a file to modifying a folder's contents and/or executing individual files. There are five basic NTFS file permissions and six folder permissions:

File Permission	Access Granted
Read	Allows the user or group to read the file and view its attributes, ownership, and the permissions set.
Write	Allows the user or group to overwrite the file, change its attributes, view its ownership, and view the permissions set.
Read and Execute	Allows the user or group to run and execute the application. In addition, the user can perform all duties allowed by the Read permission.
Modify	Allows the user or group to modify and delete a file including performing all of the actions permitted by the Read, Write, and Read and Execute NTFS file permissions.
Full Control	Allows the user or group to change the permission set on a file, take ownership of the file, and perform actions permitted by all of the other NTFS file permissions.
Folder Permission	Access Granted
Read	Allows the user or group to view the files, folders, and subfolders of the parent folder. It also allows the viewing of the folder attributes, ownership, and permissions.
Write	Allows the user or group to create new files and folders within the parent folder, view folder ownership and permissions, and change folder attributes.
List Folder Content	Allows the user or group to view the files and subfolders contained within the folder.
Read and Execute	Allows the user or group to navigate through all files and subfolders, and to perform all actions allowed by the Read and List Folder Contents permissions.
Modify	Allows the user to delete the folder and perform all activities included in the Write and Read & Execute NTFS folder permissions.
Full Control	Allows the user or group to change permissions on the folder, take ownership of it, and perform all activities included in all other permissions.

The difference between NTFS file and folder permissions is the **List Folder Contents** folder permission. NTFS folder permissions enable system administrators to limit a user's ability to browse through a tree of folders and files. This is useful for securing a specific directory such as an application directory. A user must know the name and location of a file to read or execute it when this permission is applied to its parent folder. However, in a WebCenter Forms Recognition environment, client applications in the product suite, instead of Windows Explorer, are used to process project data. The intent of file and folder permissions is to minimize the probability of accidental or malicious data destruction.

Shared permissions serve for purposes similar to NTFS permissions: They help protect files from unauthorized access. If you are a member of the Administrators or Power Users group, you can share folders on a local computer so that users on other computers can access those folders over the network. By assigning shared folder permissions to any shared folder, you can restrict or allow access to those folders over the network. Use NTFS folder permissions if the shared folder

is located on a NTFS drive. NTFS permissions are effective on the local computer and over the network.

For more information regarding folder permissions, refer to [Appendix B: File Permissions Matrix](#).

7.3 Access to Project Data

WebCenter Forms Recognition uses a hierarchical file structure to store project-related data. The project directory is at the highest level of this structure.

All Forms Recognition components (including services, applications, license engine, and users) need appropriate access rights to the project directory and all of its subfolders.

See [Section 7.2: File System Security](#) above for details on how to enable access to project data.

Once Forms Recognition has been installed, configured, and prepared for production, appropriate file access security should be applied to the project directory before releasing the implementation to the general user community. A correct application of file access security can prevent unauthorized access to project data while granting access to authorized users.

To apply file access security to the Forms Recognition project directory:

1. Launch Windows Explorer on the server containing the project directory
2. Locate the project folder, right-click the folder name, and select **Properties**
3. In the **Properties** dialog box, go to the **Sharing** tab
4. Click **Share this folder**
5. In the **Share** name field, type a name for the share
6. Click **Permissions**. In the **Share Permissions** dialog box, do the following tasks, and then click **OK**:
 - a. Add the local Forms Recognition group with **Full Control** permission
 - b. Add the local Forms Recognition Users group with **Change** permission
 - c. Add the local Administrators group with **Full Control** permission
 - d. Remove the **Everyone** group
7. Go to the **Security** tab
8. Do the following tasks and click **OK** when finished:
 - a. Add the local Forms Recognition group with **Full Control** permission
 - b. Add the local Forms Recognition Users group with **Change** permission
 - c. Add the local Administrators group with **Full Control** permission
 - d. Remove the **Everyone** group

Note: The Forms Recognition and Forms Recognition Users groups are local groups. The Forms Recognition local group should be created on all WebCenter Forms Recognition servers and RemoteAdmin machines; the Forms Recognition Users local group is only required on the server storing the project data. For an explanation of these groups, see [Section 7.4: Accounts and File Access Security](#).

7.4 Accounts and File Access Security

Access to project data in a WebCenter Forms Recognition implementation should be granted using a combination of Discretionary Access Control (DAC) and Role-based Access Control (RBAC).

The Discretionary Access Control model allows the owner of objects or resources (in this context, a System Administrator) to control who accesses them and what operations they can perform. For example, a System Administrator who creates a share called **Projects** to hold data pertaining to a particular WebCenter Forms Recognition project can control and dictate (per the organization's security policy and business rules) who can access the items within the share.

The Role-Based Access Control model, also referred to as a non-discretionary model, makes access decisions based on the rights and permissions granted to a role or groups, instead of an individual. In this model, System Administrators create roles (or groups), and then assign rights and permissions to the role (or group) instead of directly to a user; users are then placed into a role (or group) and inherit the rights and permissions assigned to the role (or group).

The following table lists the recommended groups and accounts that should be created for each implementation of WebCenter Forms Recognition:

Group/ Account Name	Purpose
Forms Recognition Project Users	Global group containing all users designated as a Forms Recognition project designer and/or data verifier within an organization.
Forms Recognition Admin	Global group containing all users designated as a Forms Recognition System Administrator within an organization. This group should be added to the local Forms Recognition group on all RTS servers and RTS Remote Admin workstations.
Forms Recognition	Local group used to grant access to local Forms Recognition resources; the Forms Recognition Admin global group should be added to its membership. Create this group on all Forms Recognition Server and RemoteAdmin machines
Forms Recognition Users	Local group used to grant access to the project directory. Add the global group Forms Recognition Project Users to its membership. Create this group on the Forms Recognition server housing the project directory.
Forms Recognition RTSvc	Service account used to start the WebCenter Forms Recognition Service Manager. This user should be a member of the Forms Recognition Admin global group and the local Administrators group on all Forms Recognition servers and Remote administration machines.

The following table lists the groups and accounts, assigned permissions, and the folders/objects on which the permissions should be applied for each implementation of WebCenter Forms Recognition:

Group/ Account Name	Permission Type: Shared	Permission Type: NTFS	Folder/Objects Assigned On
Forms Recognition	Full Control	Full Control	<Project Path>
Forms Recognition Users	Change	Modify	<Project Path>

For a comprehensive list of security settings and options, see [Appendix B: File Permission Matrix](#).

7.5 Configuring the Service Account for WebCenter Forms Recognition

7.5.1 Running WebCenter Forms Recognition on a Domain Network

WebCenter Forms Recognition Runtime Server service utilizes a Windows service that runs in the server background. This service manages the operation of Runtime Server instances, and processing of documents automatically.

When running Forms Recognition on multiple servers located on a domain network, it is recommended that the WebCenter Forms Recognition Runtime Server service be assigned a domain user against the Windows service. This will allow Forms Recognition to communicate with all servers running the service across the domain.

The service account used in WebCenter Forms Recognition is also given permission to any file/folder shares across the servers to allow the Runtime Server service access to all project related files.

7.5.2 System Monitoring

The System Monitoring service is used to send email notification to selected users to notify of any errors, or warnings, that any Runtime Server instance may raise during its operation.

The service user account used for System Monitoring should have sufficient rights to be able to send emails on the server and domain.

8 Configuring Runtime Components

8.1 Configuring the Runtime Service Manager

Below are the steps required for configuring the Runtime Service Manager. Administrator rights are needed to perform these steps:

1. Select **Start** > **Administrative Tools** > **Services**
2. Locate and double-click the **WebCenter Forms Recognition Runtime Service Manager**
3. On the **General** tab, under **Startup type**, select **Automatic** from the drop down list
4. Go to the **Log On** tab
5. Under **Log on as**, select **This account**
6. Click **Browse...**
7. Find and add the domain user with appropriate and sufficient permissions for Forms Recognition processing network access rights (e.g. WFR RTSsvc), and then click **OK**
8. Type the domain password for the user in the fields provided
9. Click **OK**, and then close the Services window

8.2 Configuring the RTS RemoteAdmin MMC Snap-in

The installation of WebCenter Forms Recognition creates a default console, called WebCenter Forms Recognition Service Manager, that you can use to configure the Forms Recognition RTS RemoteAdmin MMC snap-in.

Important: Before configuring the RemoteAdmin MMC snap-in, make sure that the steps outlined in [Section 8.1: Configuring the Runtime Service Manager](#) above have been performed and the WebCenter Forms Recognition Runtime Service Manager is started. Unless the service has been started, the MMC will not connect to the machine.

1. Launch the WebCenter Forms Recognition Service Manager MMC snap-in by selecting **Start** > **Programs** > **Oracle** > **WebCenter Forms Recognition** > **Runtime Service** > **Management Console** on the desktop of the target machine
2. Right-click the **Runtime Server Administration** node and select **New RTS Group** from the context menu
3. On the **New Group** dialog, type a group name and click **OK**
4. Expand the **Runtime Server Administration** node
5. Right-click the group you created, and then select **New Machine...**

Note: There may be a short pause before the Group Management dialog is displayed. This is because the RemoteAdmin MMC Snap-in is searching the domain for machines where the WebCenter Forms Recognition Runtime Service Manager service is running.

6. In the **Domains** dropdown, select the domain where the machine being configured is located
7. Select or enter the name of the WebCenter Forms Recognition server to be added and click **OK**
8. Right-click on the machine name and select **New > RTS Instance...**
9. On the **New RTS Instance** dialog, type the instance name and then click **OK**

For information on how to configure project settings for a Forms Recognition instance, see the *WebCenter Forms Recognition Runtime Server User's Guide*.

8.3 Advanced Logging

The standard Runtime Server log includes system-level resource information and, in the event of a system crash or failure, special error logs.

8.3.1 System Resource Logging

In the `<Installation Folder>\Bin\bin\Log` folder, the log files for the different WebCenter Forms Recognition components can be found as follows:

- **V_ Log File:** For Verifier messages, e.g. any custom script errors
- **H_ Log File:** For Runtime Server messages
- **VA_ Log File:** For Advanced Verifier messages
- **L_ Log File:** For Learnset Manager messages, e.g. when the user triggers document learning, or when a backup of the Learnset is taken, etc.
- **D_ Log File:** For Designer messages (including scripting errors)
- **U_ Log File:** For Unknown/External application messages
- **S_ Log File:** For Standard Service Manager messages
- **I_ Log Files:** Component log files for all applications and are written to by the application during the normal running.

For example, the file name syntax for the Runtime Server log file, where the Runtime Service instance is called **Oracle AP**, would be:

`<Installation Folder>\Bin\bin\Log\H_Oracle AP_<yyyymmdd>.log`

Information is written to the log files using a single, delimited line per entry, in the following format:

Position	Description
1	Type of message (info, warning, error, etc)
2	Severity of message
3	Time logged
4	Process ID (PID)
5	Overall used/available physical memory
6	Overall used/available virtual memory
7	Used physical/virtual memory by this Runtime Manager
8	Process handles used by this Runtime Manager

9	GDI resources/UserObjects used by this Runtime Manager
10	Message Description

8.3.2 Crash/Failure Logging

In the case of a system or application crash or failure, an additional error log file will be created with a format `C_<Process ID>_yyyymmdd.log`

This will log crashes under the following circumstances:

- **Import Crashes:** The log file will have stack information specific to the status of the system when the crash/failure occurred.
- **OCR Engine Crashes:** The log file will have a **ReadZone** entry for the specific OCR engine for which the crash occurred and stack information specific to the status of the system when the crash/failure occurred.
- **Classification Engine Crashes:** The log file will have a **Classify** entry for the specific Classification engine for which the crash occurred, the specific Class name where the crash occurs, and stack information specific to the status of the system when the crash/failure occurred.
- **Extraction Engine Crashes:** The log file will have an **EvalZone** entry for the specific Extraction engine for which the crash occurred, the field where the crash occurs, and stack information specific to the status of the system when the crash/failure occurred.
- **Export Crashes:** The log file will have a **StepExport** and stack information specific to the status of the system when the crash/failure occurred.
- **Clean-up Crashes:** The log file will have a **ProcessDocumentsCleanUp** and stack information specific to the status of the system when the crash/ failure occurred.
- **Script Events:** The name of the script event, which could be project level or a certain class. For script events on field level, the field name will be added to the crash/failure log.

9 Enabling Additional OCR Engine Languages

WebCenter Forms Recognition supports many OCR engine languages. The FineReader 10 engine supports Asian languages such as Japanese, Chinese and Korean.

9.1 Enabling a Language for an OCR Engine

Note: If not already done, you first have to enable the support of double byte and extended ASCII character sets, (Greek, CJK, Russian, Hebrew) for your system, as described in [Section 9.2: 1.1 Adding an Input Language for Windows 7 or Windows Server 2008](#) below.

A language can only be processed by WebCenter Forms Recognition if it is installed on the server machine, and if it is present in the FineReader directory:

1. Exit all Forms Recognition applications
2. On the Forms Recognition servers, stop the WebCenter Forms Recognition Runtime Server services
3. Ensure the appropriate FineReader language file(s) are present in the folder:
`<Installation Folder>\Components\Cairo\FineReaderX` on the Forms Recognition server for FineReader 10, or `<Installation Folder>\Components\Cairo\FineReaderX1\Data` for FineReader 11

If necessary, the FineReader language files can be copied from the WebCenter Forms Recognition installation media, from the folder:

- `<Installer Root>\Bin\WLPostInstall\LangFile\FineReaderX` for FineReader 10, or
`<Installer Root>\Bin\WLPostInstall\LangFile\FineReaderX1\Data` for FineReader 11
4. Restart the WebCenter Forms Recognition Runtime Server services
 5. Restart the client application

9.2 Adding an Input Language for Windows 7 or Windows Server 2008

1. Select **Start** > **Control Panel** > **Clock, Language, and Region** > **Region and Language**
2. Click the **Keyboards and Languages** tab
3. Click **Change keyboards...**
4. Under Installed services, click **Add...**
5. Expand the language item you want to add, then expand **Keyboard**
6. Select the input languages you want to add
7. Click **OK** to confirm

Appendix A Web.config Options and Associated Resource File Parameters

The table below contains some items that can be modified in the *<Installation Folder>\WebCenter Forms Recognition Web Server\web.config* file with regards to enabling, disabling or otherwise customizing certain features.

Option	Description
ADOCommandExecutionTimeout	Optional attribute. Timeout in seconds for database stored procedures execution. If not specified timeout from database connection string is used. <client ADOCommandExecutionTimeout=10></client>
AllowAccessToDocumentsToIndex Only	This option controls whether navigation is enabled only for documents for indexing (those with states from enabled workflow input states). This option only takes effect when "Disable navigation to valid documents" is set to <i>True</i> in settings. When set to <i>False</i> (or not included in web.config) WVC works as before allowing navigation to out-of-workflow documents. Default Value: false
assembly	Note: This attribute is not configurable. Required attribute. Assembly contains custom user provider class. <user.controller> <userProvider.assembly="Brainware.Verifier.WebClient" ... </user.controller> Required attribute. Assembly contains custom logger provider class. <system.logger> <userProvider.assembly="Brainware.System.Logger" ... </system.logger>
batchColumnVisibility	Configuration of additional columns in the Batch View. Setting this attribute to true will display the <i>External Group ID</i> batch column in WVC: externalGroupIdColumn visible="true"/ Setting this attribute to true will display the <i>External Batch Name</i> column in WVC: externalBatchNameColumn visible="true"/ Setting this attribute to true will display the <i>Transaction ID</i> batch column in WVC: transactionIdColumn visible="true"/ Setting this attribute to true will display the <i>Transaction Type</i> batch column in WVC: transactionTypeColumn visible="true"/
BatchViewPageSize	The number of batches to display on Web Verifier in the batch list. Any batches exceeding that count are divided into other navigation pages. The default value is 20 , allowing for up to 20 batches to be shown in the Web Verifier batch list. <add key="BatchViewPageSize" value="20" />
cacheSize	A parameter that allows users to specify the number of documents to cache when working on a batch of documents. Document pre-loads workdoc objects to improve performance when opening the next documents within the batch. Set the value for the cacheSize property within the <system.controllers>/<document.controller>/<document> section:

	<pre><document cacheSize="10" maxPagesToPreload="4"></pre> <p>It is not possible to disable the pre-loading feature, but it is possible to minimize the number of pre-loaded documents to 2, which represents one current document and one document pre-loaded.</p>
class	<p>Required attribute. Custom user provider.</p> <pre><user.controller> <userProvider.class="Brainware.Verifier.WebClient.Core.WebUser Provider" ... </user.controller></pre> <p>Required attribute. Custom logger provider.</p> <pre><system.logger> <userProvider.assembly="Brainware.System.Logger.LoggerFactory" ... </system.logger></pre>
connectionStrings	<p>Configuration connects to database.</p> <pre><connectionStrings> <add name="Entities" connectionString="..." providerName="System.Data.EntityClient" /> </connectionStrings></pre>
DocumentViewPageSize	<p>The number of folders to display in the Document Tree view, when selecting <i>Show Selected Batch</i>. The default value, 4, denotes 4 folders to display in <i>Show Selected Batch</i> view; any additional batches are shown in subsequent navigation panels.</p> <pre><add key="DocumentViewPageSize" value="4"/></pre> <p>Default Value: 4</p>
EnableProfiler	<p>Enables the Web Verifier profiler.</p> <pre><add key="EnableProfiler" value="true false" /></pre> <p>The profiler collects and records time taken by user actions such as commands and their internal sub-operations.</p>
formEvents	<p>Required attribute. Enable/disable focus changed event on fields on the Verification view.</p> <pre><focusChanged enabled="true false" /></pre> <p>Controls firing of FocusChanged event on Enter key press in a field. Defaults to true, if web.config does not have this setting, it is considered to be turned on. This setting has no effect on FocusChanged event in case <mouseClicked> is set to true.</p> <p>Required attribute. Enable/disable mouse click event on fields and table on the Verification view in Indexing mode.</p> <pre><mouseClicked enabled="true" /></pre> <p>Required attribute. Enable/disable tabPressed event on fields and table on the Verification view in Indexing mode.</p> <pre><tabPressed enabled="true" /></pre> <p>Required attribute. Enable/disable itemCopied event.</p> <pre><itemCopied enabled="true" /></pre> <p>Required attribute. Enable/disable table cell select event.</p> <pre><tableCellSelected enabled="true" /></pre>
httpHeaderBasedSso	<p>Enables the Single Sign-On (SSO) user authentication using an HTTP header sent by an SSO service. The header value contains the SSO authenticated user name.</p> <p>This parameter is used by the SSO service to send logged in user name.</p> <p>Setting the <i>enabled</i> attribute to true enables the Web Verifier SSO feature.</p>

	<p>Note: On a system using standard authentication the <i>enabled</i> setting should be set to false.</p> <p>The <i>loginHeader</i> value corresponds to the HTTP header attribute name that contains the SSO authenticated user.</p> <p>Example:</p> <pre><httpHeaderBasedSso loginHeader="remoteUser" enabled="true" /></pre>
inactiveUserTimeout	Required attribute. It is not used to control user session timeout. The user session timeout is controlled by the <sessionState Timeout. parameter.
inspectionServerTimeout	<p>Required attribute. Time of the periodical ping the IIS server by the process in separate mode.</p> <pre><system.project> <project. inspectionServerTimeout ="00:00:20" ... </system.project></pre>
inspectionTimeOut	<p>Required attribute. It is not used to control user session timeout. The user session timeout is controlled by the <sessionState Timeout. parameter..</p> <p>Default Value: 00:05:00</p>
instanceName	<p>The name of the Web module that will be shown to have access the batch list.</p> <pre><client instanceName="Web Verifier"></client></pre>
LanguageDisplayName_[ISO]	<p>This optional set of keys is located in the <appSettings> section of the web.config file.</p> <p>The [ISO] part of the key needs to be replaced by the three letters name of the folder in the Web Verifier resources folder.</p> <p>This key can be used to customize the language display names of the language selection drop down menu.</p> <p>To display the correct string for Simplified Chinese, the following should be added:</p> <pre><add key="LanguageDisplayName_ZHO" value="中文简体"/></pre> <p>Note: As soon as this parameter is defined it will override the system language name.</p>
licensePath	<p>The location of the shared license file, reference documentation regarding configuration. This should point to the License Share file.</p> <pre><project licensePath="C:\Program Files (x86)\Oracle\WebCenter Forms Recognition\License\Runtime.lic" mpdDistance="19" mpdThreshold="60" /></pre>
loadInSeparateProcess	Required attribute. Read only. The value is true only.
maxPagesToPreload	<p>This setting defines the number of pages to be pre-loaded for a document.</p> <p>According to the settings made, the following actions will be triggered on page images in the background when a document is loaded:</p> <ul style="list-style-type: none"> • Pre-loading • Converting to JPEG • Saving to the database <pre><document cacheSize="10" maxPagesToPreload="4"></pre> <p>First and last pages are always pre-loaded, remaining cache slots are filled with pages that have field candidates starting from the lower index.</p>
pathToProjectExe	<p>The location of the Forms Recognition Designer module (DstDsr.exe).</p> <p>Default Value: pathToProjectExe="C:\Program Files (x86)\Oracle\WebCenter Forms Recognition\Bin\Bin\</p>
priority	Set this attribute to identify tracing level. Options are:

	<ul style="list-style-type: none"> • DEBUG: Full tracing of information and errors • ERROR: Errors only <pre><priority value="ERROR"/></pre> <p>Default Value: ERROR</p>
reinitScript	<p>By default, this attribute is always set to true. This will recover the script engine whenever a script error occurs in Web Verifier application.</p> <p>Default Value: true</p>
remoteObjectRenewalTimeout	<p>Optional attribute. Remote object references are renewed at this time period (in seconds). Defaults to 180. Minimum accepted value is 30. The lower the number the faster unused objects free memory but this can lead to errors for long running commands. One can increase this value if some actions (i.e. field validation) take a while to finish with remoting error.</p> <p>Note that this value should be set in both web application config file and Brainware.System.Project.exe config file</p> <pre><client remoteObjectRenewalTimeout=45></client></pre> <p>Default Value: 180</p>
sessionHeader	<p>Name of the HTTP header used by the SSO service to send its user session ID.</p> <p>This is SSO service dependent. For example, the <i>headerName</i> parameter is ShibSessionId for a Shibboleth provider.</p>
sessionState Timeout	<p>The <i>sessionState Timeout</i> parameter controls the timeout for a user session. The value represents the number of minutes that a user is allowed to be inactive before the session is ended.</p> <p>Note: Session timeout should be set to a value less than that of the SSO session. Refer to the product documentation of your SSO provider for details how to configure the SSO session duration.</p> <p>Default Value: 20</p>
ShowExtendedErrorMessages	<p>Set this attribute to true to enable stack trace information in the error messages appearing in Web Verifier. Messages are written to the Trace Log file.</p> <p>Allowable values are true and false.</p> <pre><add key="ShowExtendedErrorMessages" value="true"/></pre> <p>Default Value: true</p>
slogan	<p>A text message that can be displayed on the Web Verifier browser header with corporate messages / announcements / Corporate Slogan.</p>
SYSTEM_LONG_DATE_FORMAT	<p>This special XML key is located in each resources /[3-letters language]/Brainware.Verifier.WebClient.resx file.</p> <p>This key contains the date formatting pattern for the last access date column presentation within the batch list for that language. It is optional, and if set to <i>Empty</i>, then the system default formatting is applied.</p> <p>For Chinese Traditional and Simplified languages, the date format to be used is YYYY-MM-DD, and the time format is 24 based without any Chinese characters.</p> <p>Example for the Chinese:</p> <pre><data name="SYSTEM_LONG_DATE_FORMAT" xml:space="preserve"> <value>yyyy-MM-dd, hh:mm:ss</value> </data></pre>
Trace log/ debug file 1	<p>It keeps the debug/ trace log file under X size. Once the X size is reached, the log is recycled/ deleted and a new log is created.</p> <p>For example, below its set to 100kb, as soon as the trace.log file went over 100kb the log file size changed to 0kb and new log messages were written.</p> <pre><appender name="RollingFile"</pre>

	<pre>type="log4net.Appender.RollingFileAppender, log4net"> <layout type="log4net.Layout.PatternLayout, log4net"></pre>
Trace log/ debug file 2	<p>When the file reaches X size, it is archived as trace.log.1 and a new trace.log will be created. When the trace.log exceeds size once more, trace.log.1 becomes trace.log.2 and the existing trace.log becomes trace.log.1, again a new trace.log will be created.</p> <pre><appender name="RollingFile" type="log4net.Appender.RollingFileAppender, log4net"> <layout type="log4net.Layout.PatternLayout, log4net"></pre>
usePath	<p>Required attribute. Enable/disable using pathToProjectExe parameter. Set this attribute to false to set pathToProjectExe parameter is current directory.</p>
waitLoadTimeOut	<p>Required attribute. Timeout for initial loading of project.exe. This parameter is used with enable option: loadInSeparateProcess = true</p> <p>Default Value: 00:01:00</p>

Appendix B File Permission Matrix

The table below displays the various file permissions that are used within WebCenter Forms Recognition.

Role / Group	Description							
Administrators	Administrator user with full access rights to all application modules and features.							
Developers	The groups of users that develop, maintain, and enhance projects.							
Learnset Manager	Typically one user in the organization responsible for maintaining the project learnsets.							
Advanced Verifiers	Several users responsible for identifying documents for improvements to the project learnset.							
Standard Verifiers	Data entry clerks responsible for document correction.							
RTS Service User	The service account responsible for running the service for automatic document processing. Configured only on the server machines.							
Directory	Groups	NTFS Permissions						
		Full Control	Modify	Read & Execute	List Folder Content	Read	Write	No Access
Root Batch Folder	Administrators							
	Developers							
	Learnset Manager	•	•	•	•	•	•	
	Advanced Verifiers							
	Standard Verifiers RTS Service User							
Common Folder	Administrators							
	Developers							
	Learnset Manager	•	•	•	•	•	•	
	Advanced Verifiers							
Global Project and Global Learnset	Standard Verifiers							•
	RTS Service User							
	Administrators							
	Developers							
	Learnset Manager	•	•	•	•	•	•	
Local Project and Local Learnset	RTS Service User							
	Standard Verifiers							
	Advanced Verifiers			•	•	•		
	Standard Verifiers							
ASE Pool	Administrators							
	Developers	•	•	•	•	•	•	
	RTS Service User							
ASSA CSV File	Learnset Manager							
	Advanced Verifiers			•	•	•		
	Standard Verifiers							
ASSA CSV File	Administrators							
	Developers	•	•	•	•	•	•	
	RTS Service User							

	Learnset Manager Advanced Verifiers Standard Verifiers								•
--	--	--	--	--	--	--	--	--	---

Appendix C Registry Options

The table below contains some items that can be modified in the Registry concerning enabling/disabling/customizing certain features.

Option	Description
DumpProjectScriptCode	<p>This key is available to allow Oracle Support to carry out advanced troubleshooting on any issues with script running on the Web Verifier.</p> <p>To create the registry key to provide script dumps, follow the instructions outlined below:</p> <ol style="list-style-type: none"> 1. Launch Windows Registry Editor 2. Navigate to: HKEY_LOCAL_MACHINE\SOFTWARE \[Wow6432Node \]Oracle\Cedar 3. Create a new REG_DWORD (DWORD Value key) and call it DumpProjectScriptCode 4. Modify the new key created and enter the one of the following values: 3 - to enable this feature for script page export 0 - to disable this feature <p>With the feature enabled the scripting engine files can be used to review any compilation problems. This is an advanced feature and requires advanced knowledge of the product.</p>
ErrorTraceDir	<p>The ErrorTraceDir registry key is available for those customers who wish to place the component tracing logs in a different location than the default <Installation Folder>\Bin\bin\Log folder. The registry key allows the administrator to place the logs in a specific folder location separate from the core product logs.</p> <p>The registry setting is only applicable for the component logs, not for the core product logs.</p> <p>To configure a new location for Component Logs, follow the instructions outlines below:</p> <ol style="list-style-type: none"> 1. Launch Windows Registry Editor 2. Navigate to: HKEY_LOCAL_MACHINE\SOFTWARE \[Wow6432Node \]Oracle>ErrorTrace 3. Create a new REG_SZ (String Value key) and call it ErrorTraceDir 4. Modify the new key created and enter the filepath location for component logs to be entered. Verify that the path entered exists and the service account/ user has sufficient permissions to write to that location. <p>For the change to take place, exit all WebCenter Forms Recognition applications, and stop any services running on the machine related to Forms Recognition, then launch the application and all new component logs will be written in the desired location.</p>
LanguageSupportWorkflowSettingsVisible	<p>This registry key is used within WebCenter Forms Recognition to allow the developer to utilize advanced Language Support configuration and setting. This additional feature can be enabled via the Registry using the steps outlined below:</p> <ol style="list-style-type: none"> 1. On the server/ machine where Designer is installed and used, launch the Windows Registry Editor. 2. Navigate to: HKEY_LOCAL_MACHINE\SOFTWARE \[Wow6432Node \]Oracle\Cedar 3. Create a new REG_DWORD (DWORD Value key) and call it LanguageSupportWorkflowSettingsVisible 4. Modify the new key created and enter the one of the following values:

	<p>1 - to enable this feature 0 - to disable this feature</p> <p>For the change to take place, exit all WebCenter Forms Recognition applications, and stop any services running on the machine related to Forms Recognition, then launch the application and all new component logs will be written in the desired location.</p> <p>To view the advanced options in Forms Recognition Designer:</p> <ol style="list-style-type: none"> 1. Launch Designer application 2. From the Options Menu select Settings 3. Navigate to the Definition tab and new settings will display. <p>With the settings showing, the developer can utilize additional language features which will allow them to convert words/etc to extended ASCII character set.</p>
<p>ASEnginePoolAllowedChar Difference</p>	<p>In certain instances, some duplicates in the ASE/ASSA search may not be returned from the vendor/customer search. In these cases it may be that the ASSA engine perceives these as duplicates of existing entries.</p> <p>There is a configuration step that can be undertaken which can return the suspected duplicates as well. This may slightly increase the results of the ASE pool, but also return potential items that were not returned in the original search.</p> <p>To configure the ASE pool to return additional likely results:</p> <ol style="list-style-type: none"> 1. Launch Registry Editor 2. Navigate to: HKEY_LOCAL_MACHINE\SOFTWARE \[Wow6432Node \]Oracle\Cedar 3. Create a new DWORD registry variable for ASEnginePoolAllowedCharDifference 4. Close the Registry Editor <p>Reanalyze the document once more and any missing entries should now appear.</p>
<p>HideBatchReleaseDialog</p>	<p>This key allows Oracle Support to disable the Batch Release dialog box within the Verifier, where the business does not require prompting users on next task. The registry value can be used to determine the next action carried out by users.</p> <p>The default action of the Batch Release dialog box is to verify the next invalid batch. When the dialog is suppressed, this value is maintained. To change to a different action, use the Batch Release dialog box once, then change the setting accordingly and click OK.</p> <p>To create the registry key to suppress the Batch Release confirmation screen, follow the instructions below:</p> <ol style="list-style-type: none"> 1. Launch Windows Registry Editor 2. Navigate to: HKEY_LOCAL_MACHINE\SOFTWARE \[Wow6432Node \]Oracle\Cedar 3. Create a new REG_DWORD (DWORD Value key) and call it HidebatchReleaseDialog 4. Modify the new key created and enter the one of the following values: 0 - to enable the confirmation screen (default) 1 - to disable/ hide the confirmation screen <p>For the changes to take place, exit all Forms Recognition applications, and then launch the applications again.</p> <p>To view that the change has been implemented:</p> <ol style="list-style-type: none"> 1. Launch Verifier 2. Verify the batch to completion - no dialog box should appear
<p>All</p>	<p>The ErrorTrace registry Key was introduced into core product logs to provide</p>

	<p>additional trace information on any errors or warnings in the system. The default value after installation is to record errors only related details.</p> <p>Modify the registry values to set the value from 0 to either 1, 2, or 3:</p> <ul style="list-style-type: none"> 1 - Only Errors (Default) 2 - Errors & Warnings 3 - Errors & Warnings & Information <p>To configure ErrorTrace All value:</p> <ol style="list-style-type: none"> 1. Launch Registry Editor 2. Navigate to: HKEY_LOCAL_MACHINE\SOFTWARE\[Wow6432Node\]Oracle\ErrorTrace 3. Create a new DWORD registry variable for All, set to the appropriate value of either 0, 1, 2, or 3 4. Close the Registry Editor
MaximumDiskSpaceUsageMB	<p>This registry value controls the amount of disk space allocated for component level logs on this server / workstation in MB. Setting this value to 0 has the same effect as if the value is not created at all, which is “deactivated”.</p> <p>To configure the maximum disk space usage:</p> <ol style="list-style-type: none"> 1. Launch Registry Editor 2. Navigate to: HKEY_LOCAL_MACHINE\SOFTWARE\[Wow6432Node\]Oracle\ErrorTrace 3. Create a new DWORD registry variable for MaximumDiskSpaceUsageMB, set to the appropriate value in MB 4. Close the Registry Editor
TotalDaysToKeepFiles	<p>This registry value maintains the number of days the old component level logs are kept by the Forms Recognition server. Setting this value to 0 has the same effect as if the value is not created at all, which is “deactivated”.</p> <p>To configure the number of days to keep files:</p> <ol style="list-style-type: none"> 1. Launch Registry Editor 2. Navigate to: HKEY_LOCAL_MACHINE\SOFTWARE\[Wow6432Node\]Oracle\ErrorTrace 3. Create a new DWORD registry variable for TotalDaysToKeepFiles, set to the appropriate number of days to maintain logs 4. Close the Registry Editor
LimitBatchCountRequest	<p>By default, Forms Recognition will limit the number of batches displayed to 1,000. If more (or less) than 1,000 batches are required to be displayed this can be configured as described below.</p> <p>To configure the maximum number of batches to be displayed:</p> <ol style="list-style-type: none"> 1. Launch Registry Editor 2. Navigate to: HKEY_LOCAL_MACHINE\SOFTWARE\[Wow6432Node\]Oracle\Database 3. Create a new DWORD registry variable for LimitBatchCountRequest, set to the appropriate number of batches to be displayed 4. Close the Registry Editor