



# BLIND REHABILITATION CENTRALIZED SERVER INSTALLATION/IMPLEMENTATION GUIDE



Version 5.0.29  
August 2011

Department of Veterans Affairs  
VistA Health System Design & Development



## Revision History

<i>Date</i>	<i>Description</i>	<i>Author/Project Manager</i>
03/16/2005	Draft I	REDACTED
07/18/2005	Draft II	REDACTED
08/16/2005	Revised	REDACTED
11/29/2005	Revised for Build 26 and reviewer comments	REDACTED
01/18/2006	Revised for Build 5.0.26.4	REDACTED
03/03/2006	Revised for Build 5.0.26.7	REDACTED
05/24/2006	Revised for Build 5.0.26.8 and EVS Feedback	REDACTED
11/13/2006	Updates for version 5.0.27.5	REDACTED
12/06/2006	Updates for version 5.0.27.6	REDACTED
01/25/2007	Added Appendix F - Installation Steps to Upgrade Blind Rehabilitation v5.0.26.8 to v5.0.27.6	REDACTED
3/7/2007	<p>The following changes have occurred since the 5.0.26.8 version of this document:</p> <p><b>Page 3:</b></p> <ul style="list-style-type: none"> <li>• Changed version number of Blind Rehabilitation from 5.0.26.8 to 5.0.27.6</li> <li>• Changed version of Person Service Lookup from 4.0.4.3 to 4.0.4.4</li> <li>• Changed version of Standard Data Service from 7.0 to 10.0</li> </ul> <p><b>Page 9:</b></p> <ul style="list-style-type: none"> <li>• Changed version number of Blind Rehabilitation from 5.0.26.8 to 5.0.27.6 in two places</li> <li>• Changed the version of Standard Data Services from 7.0 to 10.0</li> </ul> <p><b>Page 10:</b></p> <ul style="list-style-type: none"> <li>• Changed the version of Standard Data Services from 7.0 to 10.0</li> </ul> <p><b>Page 11:</b></p> <ul style="list-style-type: none"> <li>• Changed version number of Blind Rehabilitation from 5.0.26.8 to 5.0.27.6</li> </ul> <p><b>Page 12:</b></p> <ul style="list-style-type: none"> <li>• Changed version number of Standard Data Services from 7.0 to 10.0</li> <li>• Changed version number of Person Service Lookup from 4.0.4.3 to 4.0.4.4</li> </ul> <p><b>Page 28:</b></p> <ul style="list-style-type: none"> <li>• Changed version number of PSL from 4.0.4.3 to 4.0.4.4 in two places</li> </ul>	REDACTED

<i>Date</i>	<i>Description</i>	<i>Author/Project Manager</i>
	<p><b>Page 34:</b></p> <ul style="list-style-type: none"> <li>• Changed version number of Blind Rehabilitation from 5.0.26.8 to 5.0.27.6</li> </ul> <p><b>Page 42-44:</b></p> <ul style="list-style-type: none"> <li>• Changed Software_Version from 5.0.26.8 to 5.0.27.6</li> <li>• Added the SessionTimeoutWarningMinutes property</li> <li>• Added UserNotificationsLink and UserNotificationsKeyParameterName properties</li> <li>• Added ReferralNotificationText and ReferralNotificationLink properties</li> <li>• Changed CrystalEnterpriseServerUser, CrystalEnterpriseServerPassword and CrystalEnterpriseServerName properties to lowercase</li> <li>• Changed OvernightDemographicUpdatesEnabled property from true to false</li> <li>• Added PSDUpdaterNumberOfThreads, PSDUpdaterWorkQueueSize, PSC_UPDATER_START_TIME_HOUR and PSC_UPDATER_START_TIME_MINUTE properties</li> <li>• Changed MPI_WEBLOGIC_USERNAME property from "weblogic" to "weblogic_user"</li> <li>• Changed MPI_WEBLOGIC_PASSWORD property from "holycow1" to "weblogic_password"</li> <li>• Added BACK_BOTTON_PREVENTION_MESSAGE property</li> </ul> <p><b>Page 45:</b></p> <ul style="list-style-type: none"> <li>• Removed socketLogger from "log4j.rootCategory=info, socketLogger, rolling"</li> </ul> <p><b>Page 51:</b></p> <ul style="list-style-type: none"> <li>• Changed version of Person Service Lookup to 4.0.4.4</li> </ul> <p><b>Page 52:</b></p> <ul style="list-style-type: none"> <li>• Added Appendix F</li> </ul>	
08/17/2010	<p>The following changes have occurred for the version 5.0.29.4</p> <p><b>Page 3:</b></p> <ul style="list-style-type: none"> <li>• Changed version number of Blind Rehabilitation from 5.0.27.6 to 5.0.29.4</li> <li>• Changed version number of Vista Link from 1.5.0.026 to 1.5.2.004</li> <li>• Changed version number of Kaajee from 1.0.0.019 to 1.0.1.003</li> <li>• Changed version of Standard Data Service from 10.0 to 18.0</li> </ul> <p><b>Page 9:</b></p> <ul style="list-style-type: none"> <li>• Changed the version from br_deployment_5.0.27.6 to BR-PKG-5.0.29.6</li> </ul>	REDACTED

<i>Date</i>	<i>Description</i>	<i>Author/Project Manager</i>
	<p><b>Page 11:</b></p> <ul style="list-style-type: none"> <li>• Changed the version from br_deployment_5.0.27.6 to BR-PKG-5.0.29.6</li> </ul> <p><b>Page 33:</b></p> <ul style="list-style-type: none"> <li>• Changed version number of Blind Rehabilitation from BR_EAR_5.0.27.6 to BR_APP_5.0.29.4</li> </ul> <p><b>Page 54:</b> Added Appendix G</p> <p><b>Page 55:</b> Added Appendix H</p>	



## Table of Contents

Introduction.....	1
Benefits .....	1
Enhanced Technology.....	2
HealthVet-VistA Software Requirements .....	3
Orientation .....	4
Recommended Users .....	4
Related Manuals.....	4
Software Retrieval .....	4
Documentation Retrieval .....	5
VistA Intranet.....	5
Pre-Installation Information.....	6
Blind Rehabilitation Central Server Administration Staff.....	6
Test Sites.....	6
Hardware and Operating Systems Requirements.....	6
System Performance Capacity .....	6
Software Installation Time.....	6
Users on the System.....	7
Backup Routines .....	7
Name Space .....	7
VistA Blind Rehabilitation Hardware and Software Requirements.....	7
Workstation Software Requirements .....	7
Workstation Hardware Requirements and Guidelines .....	7
Production Centralized Server Hardware Requirements and Guidelines .....	8
Installation Instructions.....	9
Step 1: Installation of Oracle Database Server .....	9
Step 2: Configuration of Oracle Database .....	9
Prepare the Oracle Client Software and Extract the Deployment Files .....	9
Create the Standard Data Services (SDS) Reference Tables .....	9
Create the Blind Rehabilitation Oracle Environment .....	10
Step 3: Installation of BEA WebLogic Server Version 8.1.4 .....	11
Step 4: Configuration of Weblogic Server.....	11
Create the Blind Rehabilitation WebLogic Environment .....	11
Step 5: Installation of Crystal Enterprise .....	35
Step 6: Configuration of Crystal Enterprise.....	36
Appendix A – Sample Blind Rehabilitation application.properties file.....	41
Appendix B – Sample Blind Rehabilitation log4j.properties file .....	44
Appendix C – Sample Blind Rehabilitation MPIListener.properties file .....	47
Appendix D – Sample Blind Rehabilitation Patient Service (PSC/PSD) PatSvcPkg.properties file.....	49
Appendix E – Sample Blind Rehabilitation Person Service Lookup (PSL) PatientLookup.properties file (contained in the pslConfig_4.0.4.4.jar file) .....	50
Appendix F – Installation Steps to Upgrade Blind Rehabilitation v5.0.26.8 to v5.0.27.6.....	51
Appendix G – Installation Steps to Upgrade Blind Rehabilitation v5.0.27.6 to v5.0.28.4 .....	53
Appendix H – Installation Steps to Upgrade Blind Rehabilitation v5.0.28.4 to v5.0.29.4 .....	54
Glossary/Acronym List.....	55





# Introduction

The Blind Rehabilitation (BR) application provides enhanced tracking, and reporting, of the blind rehabilitation services provided to veterans by:

- Visual Impairment Service Teams (VIST) Coordinators
- Blind Rehabilitation Centers (BRCs)
- Blind Rehabilitation Outpatient Specialists (BROS)
- Visual Impairment Services Outpatient Rehabilitation (VISOR) Programs
- Visual Impairment Center to Optimize Remaining Sight (VICTORS)

Currently, there is no VistA software that meets the needs of the Blind Rehabilitation Centers or BROS and the VIST 4.0 package only monitors, tracks, and reports on a limited amount of data for the VIST.

The site-based VIST 4.0 package is being replaced with the re-hosted Blind Rehabilitation (BR) 5.0 application supporting the HealthVet-VistA enterprise architecture. In addition to providing the base functionality of the BR 4.0 system, BR 5.0 provides a web-enabled GUI through which users can access enhanced capabilities intended for VIST Coordinators, new functionality for BROS, BRC personnel and waiting times and waiting list.

The Blind Rehabilitation 5.0 application provides entirely new functionality that encompasses and integrates all five segments of the Blind Rehabilitation Services including waiting times and waiting list.

## Benefits

- Complies with HealthVet-VistA Architecture
- Complies with 508 regulations, using W3C standards
- Accessible web based application, via a web browser
- Supports the OI Single Sign-on initiative
- User authentication via role based permissions
- User friendly
- Seamless continuum of care
- Minimum user disruption
- Simplified data entry
- Better identification and treatment of veterans
- Consolidates data
- Enables system driven waiting times and waiting list tracking and reporting capabilities
- Enables users to receive comprehensive views of a patient's BR Services across institutions
- Facilitates data tracking and auditing capabilities
- Improves accountability
- Enhanced reporting features
- Provides Data Standardization which improves and provides consolidated data reporting
- Improved blind services tracking
- Enables Research and Provides Outcomes tracking and reporting capabilities
- Improves VHA organizational communication
- Transmits to the Health Data Repository

## Enhanced Technology

- A single consolidated database and application will replace the current site-specific VIST 4.0 package
- Fulfills the congressional mandate on waiting times and waiting list calculations
- Electronic referral process to track patient applications for service
- Notifications feature to alert users of pending referrals
- Encounters/Progress Notes will be automatically created for assessments and field visits (PCE interface) in a future version.
- Nationwide centralization of Blind Rehabilitation services data to allow nationwide reporting
- Ad-hoc reporting capabilities
- Secure Web Access (128 Bit SSL) from any authorized VA workstation
- Improved technology using web browser access and improved data security, via the VHA intranet
- Uses modern system architecture which allows for faster system enhancements
- Enhancements will be rolled out to all users at the same time ensuring consistent data
- Allows ability to track BR patient care access across institutions
- Patients can be referred or transferred to other institutions if they move without having to recreate patient data
- Patient lookup using the Health\_eVet Person Lookup Service (PSL) and Person Service Demographics (PSC)
- Standardized lookup tables using the Health\_eVet Standard Data Service (SDS)
- Improved data integrity
- Minimize the maintenance and support required by IT support staff

## HealthVet-VistA Software Requirements

During the installation of <sup>1</sup>Blind Rehabilitation 5.0.29.4, the following java packages must be installed.

<i>Software</i>	<i>Version</i>
<sup>2</sup> VistALink	V 1.5.2.004
<sup>3</sup> Kaajee	V 1.0.1.003
Person Service Lookup (PSL)	V. 4.0.4.4
Patient Service Construct (formerly Person Service Demographics. Referred to as PSC or PSD)	V. 2.0.0.8
<sup>4</sup> Standard Data Service	V. 18.0

---

<sup>1</sup> Changed version number of Blind Rehabilitation from 5.0.27.6 to 5.0.29.4

<sup>2</sup> Changed version number of Vista Link from 1.5.0.026 to 1.5.2.004

<sup>3</sup> Changed version number of Kaajee from 1.0.0.019 to 1.0.1.003

<sup>4</sup> Changed version of Standard Data Service from 10.0 to 18.0

# Orientation

## Recommended Users

The intended audience for this document is the staff responsible for installing or administering the centralized components of the Blind Rehabilitation application. This document is not intended for field Information Resources Management (IRM) staff, as there are no centralized components installed in the field. This document is technical in nature and assumes the reader is familiar with Oracle, WebLogic, and Crystal Reports. This document anticipates that separate staff members may be responsible for installing various components and this guide is separated into appropriate sections for this purpose.

## Related Manuals

- Blind Rehabilitation V. 5.0 VistA Installation/Implementation Guide
- Blind Rehabilitation V. 5.0 Technical Manual and Security Guide
- Blind Rehabilitation V. 5.0 User Manual
- Blind Rehabilitation V. 5.0 Release Notes

## Software Retrieval

The Centralized Blind Rehabilitation Application server software is not available for field download. The central server software is intended to be installed only in the production domain or testing domains by centralized system administrators. The software will be provided to the appropriate installation personnel by the Blind Rehabilitation project team.

The complete Blind Rehabilitation 5.0 system requires various VistA packages to be installed on the VistA servers, which will be connected to the central server. The field VistA 'M' packages will be installed by field personnel and are available at the standard VistA [ANONYMOUS.SOFTWARE] download locations. Please refer to the companion document to this guide: **Blind Rehabilitation VistA Installation/Implementation Guide** for details regarding the VistA installation steps.

## Documentation Retrieval

You can find the documentation files for Blind Rehabilitation on the OI Field Office [ANONYMOUS.SOFTWARE] directories.

<i>File Name</i>	<i>Description</i>	<i>Retrieval Format</i>
ANRV5_0CIG.PDF	* Blind Rehabilitation Centralized Server Installation/Implementation Guide	Binary
ANRV5_0VIG.PDF	** Blind Rehabilitation VistA Installation/Implementation Guide	Binary
ANRV5_0RN.PDF	Blind Rehabilitation Release Notes	Binary
ANRV5_0TM.PDF	Blind Rehabilitation Technical Manual/Security Guide	Binary
ANRV5_0UM.PDF	Blind Rehabilitation User Manual	Binary

\* This Installation Guide is only for Centralized Servers, not to be used at the field VistA site.

\*\* This Installation/Implementation Guide is for field VistA sites.

## VistA Intranet

Documentation for this product is available on the intranet at the following address:

<http://www.va.gov/vdl/>.

This address takes you to the VistA Documentation Library (VDL), which has a listing of all the clinical software manuals. Click on the Visit Impairment Service Team (VIST) link and it will take you to the Blind Rehabilitation documentation.

The link below allows access to the Blind Rehabilitation home page:

<http://vista.med.va.gov/ClinicalSpecialties/vist/index.htm>

# Pre-Installation Information

## Blind Rehabilitation Central Server Administration Staff

A centralized system administrator is recommended for installing and supporting Blind Rehabilitation/VIST 5.0 centralized server.

## Test Sites

The BR software was field tested at the following sites:

<i>Test Sites</i>	<i>Beta Cycle I</i>	<i>Beta Cycle II (Go Live)</i>
Augusta	X	X
VA Puget Sound Health Care System		X
Southern Arizona VA Health Care System (Tucson)	X	X
Hines VA Medical Center Chicago	X	X

## Hardware and Operating Systems Requirements

### System Performance Capacity

The Blind Rehabilitation application is designed to accommodate up to 1000 concurrent users. The typical peak is expected to be approximately 20 users per server configuration. It is essential that users are able to perform job duties within a reasonable amount of time.

### Software Installation Time

Software installation time for the centralized components of Blind Rehabilitation can vary from several hours to several days depending on the installer's familiarity with the required components and if the installation is 'from scratch' or an upgrade to a previously installed system. Installation of VistA components at the field is required and the local IRM needs to perform this portion. Please refer to the **Blind Rehabilitation VistA Installation/Implementation Guide** for details regarding the VistA installation steps

Updates to VA developed components or the core BR application may only require several minutes of downtime depending on the nature of what new features are part of the installation. Developers of the software upgrades will provide release notes and installation steps to help estimate the downtime duration.

## Users on the System

Most components cannot be installed while users are logged into or using the system. It is a requirement to inform users of the scheduled downtime and estimated duration of the installation before proceeding. Since there is only one nationally centralized production deployment of the system, installations, and updates will disable the software for all Blind Rehabilitation users throughout the VA.

## Backup Routines

Appropriate data backups should be made prior to installing the software.

## Name Space

The Mumps VistA Blind Rehabilitation software name space is **ANRV**. The top-level Java package name is **gov.va.med.br**.

## VistA Blind Rehabilitation Hardware and Software Requirements

### Workstation Software Requirements

- Web Browser: Microsoft Internet Explorer version 5.0 or higher with High Encryption (128 bit) and JavaScript enabled.
- Adobe Acrobat Reader version 5.1 or higher with browser plug-in enabled.

### Workstation Hardware Requirements and Guidelines

- Refer to VA workstation hardware requirements.

## **Production Centralized Server Hardware Requirements and Guidelines**

### **Application Servers (2 load balanced):**

- Dual CPU 3.0+ GHZ Processors
- 4-GB RAM
- RAID Enabled local storage
- Red Hat Linux 3.0 AS Operating System

### **Database Servers (2 clustered):**

- Dual CPU 3.0+ GHZ Processors
- 4-GB RAM
- SAN Enabled storage for database files
- Red Hat Linux 3.0 AS Operating System

### **Crystal Enterprise Server (1):**

- Dual CPU 3.0+ GHZ Processors
- 4-GB RAM
- SAN Enabled storage for Crystal Enterprise repository
- Windows 2003 Server Operating System

### **Load balancer:**

- Capable of HTTP load balancing and SSL encryption offloading



# Installation Instructions

## Step 1: Installation of Oracle Database Server

1. Blind Rehabilitation was designed to run with Oracle Standard Edition Version 10g.
2. Follow the Oracle documentation and VA installation requirements for Oracle.
3. The BR application is designed to run the Database and Application servers on separate physical machines. Oracle should be installed on database-only servers.
4. The production schema of the BR Application is required to run on Oracle Real Application Cluster (RAC) on Red Hat Linux OS 3.0.
5. Create an Oracle database instance to contain the BR application tables and Standard Data Service (SDS) reference tables. A single instance can be used to house multiple application domains for production, staging, support, and training. Each application domain will require separate database user accounts/schemas in order to maintain data integrity.

## Step 2: Configuration of Oracle Database

(Only for new installations: These steps will overwrite previous BR Oracle configurations!)

### Prepare the Oracle Client Software and Extract the Deployment Files

1. The BR database creation scripts are designed to run from a Windows client workstation against the remotely created Oracle instance. Install the Oracle 10G Client software on a workstation for this purpose.
2. Create an entry in the Oracle Client's *tnsnames.ora* file (or use the Oracle Net Configuration Assistant) to connect this workstation client to the Oracle instance. Refer to the Oracle documentation for further information on configuring the *tnsnames.ora* file. Test to make sure that connectivity to the database through sqlplus is functional.
3. Extract the *<sup>5</sup>BR-PKG-5.0.29.6.zip* file to a temporary working directory on the client workstation. This directory is referred to as *{extract\_root}* in following instructions:

### Create the Standard Data Services (SDS) Reference Tables

**NOTE:** The following SDS steps should be used only for non-production systems. The production domain uses a remotely replicated SDS schema. The production Blind Rehabilitation systems will need an SDS schema that is installed through a coordinated effort with the Standard Data Service team.

1. Follow the Instructions contained in the SDS Database Installation Guide to create the required tablespaces, users, and tables
2. Record the SDS database user information for use in following steps.

**NOTE:** The SDS database scripts can be copied to the Oracle server and run in a Linux shell directly; this will significantly shorten the SDS database creation time

---

<sup>5</sup> Changed the version from br\_deployment\_5.0.27.6 to BR-PKG-5.0.29.6

## Create the Blind Rehabilitation Oracle Environment

**NOTE:** These steps should only be followed during the initial creation of the Blind Rehabilitation database schemas. They will destroy and recreate all the database tables in the schema. To upgrade an existing installation to a later version, refer to the installation steps in the release notes for that release.

1. Open a command window (DOS) on the client workstation. Change directories to *{extract\_root}\DB\* and extract the **BR-DB-5.0.27.6.zip** file to a temporary working directory. The directory referred as *{dbextract\_root}* and change directory to Full DB Install.
2. In this directory, there is a script named **br\_setup.sql**. This file creates the BR tablespaces and initial user account. Open this file with a text editor and make appropriate changes for your Oracle installation. These changes include the directories where the datafiles will be located for the BR tablespaces and the names/passwords for the Oracle users being created. Add users for additional application domains, as needed. Record the created user details for use in following steps. Save your changes.
3. Run **br\_setup.sql** as the Oracle SYSTEM account through the DOS version of sqlplus.

Example:

```
sqlplus system/system_password@tnsname @{extract_root}\DB\{dbextract_root} \Full DB Install\br_setup.sql
```

View the *@{extract\_root}\DB\{dbextract\_root} \Full DB Install\br\_setup.log* file to verify that no errors have occurred. Correct the script and run again if errors were encountered.

4. In the same directory is a batch file named **createAll.bat**. This batch file will call multiple sub scripts to create the BR tables, indices, views, and load reference data. Open this file with a text editor and make appropriate changes for your Oracle installation (if necessary).
5. Run **createAll.bat** with the appropriate parameters.

Example:

```
createAll.bat br_user br_password DB_TNSNAME system_password sdsuser
```

View the log files named: **BR.log**, **BR\_GRANTS.log**, **BR\_INDEXES.LOG**, **BR\_VIEWS.LOG** in the *@{extract\_root}\DB\{dbextract\_root} \Full DB Install\* directory to verify that no errors have occurred. Correct the script(s) and run again if errors were encountered.

6. Run the **createAuditTrail.bat** batch file with the appropriate parameters. If this is the first installation of the BR software on this Oracle server, run the createAuditTrail.bat script as the BRVS Oracle user to create the audit trail tables.

**DO NOT** run this script if reinstalling or updating versions of the software – this script destroys the audit trail tables and removes the records.

Example:

```
createAuditTrail.bat BRVS br_user br_password DB_TNSNAME
```

## Step 3: Installation of BEA WebLogic Server Version 8.1.4

1. Follow the WebLogic documentation and VA installation requirements for WebLogic.
2. The BR application was designed to run the Database and Application servers on separate physical machines. WebLogic should be installed on application only servers.
3. The production domain of the BR Application is recommended to run on Red Hat Linux OS 3.0.
4. The BR Application is designed to run in a WebLogic Managed Server environment on two separate physical servers. A Load Balancing device with SSL acceleration should be placed in front of the WebLogic servers to direct HTTP traffic to two servers and offload the SSL encryption.
5. Create an administration server and domain as directed by the WebLogic documentation and VA WebLogic installation guideline documents.
6. Create a domain for each application environment needed. The standard domains will include Production, Staging, Test, and Training. Each domain should have two managed servers, one on each physical server. Record the managed server DNS names, IP Addresses and http ports for use in configuring the Load Balancer

## Step 4: Configuration of Weblogic Server

### Create the Blind Rehabilitation WebLogic Environment

1. Create a base directory on each physical server to hold the deployment files.  
*Ex: /u01/applications/brdeployments/*

For the following instructions, this directory will be referred to as *{deploy\_root}*.

2. Under the *{deploy\_root}* directory, create a subdirectory to hold the deployment files for each domain.

Ex:

```
/u01/applications/brdeployments/production/  
/u01/applications/brdeployments/test/  
/u01/applications/brdeployments/staging/  
/u01/applications/brdeployments/training/
```

For the following instructions, these directories will be referred to as *{domain\_\_deploy\_root}*.

3. Copy the <sup>6</sup>*BR-PKG-5.0.29.6.zip* file to each of these subdirectories and extract it there with WinZip or other zip file tool.

---

<sup>6</sup> Changed the version from br\_deployment\_5.0.27.6 to BR-PKG-5.0.29.6

4. Open the WebLogic admin console, select the managed server, and click the remote start tab. Edit the classpath and java options fields to match the lists on the following pages. Repeat this for each managed server in each domain. It is easiest to edit these entries in a text editor and paste them into the WebLogic server console fields.

Set the classpath for each managed server to:

**NOTE:** Each path is delimited by a colon (:) and remove any line feeds between entries.

```

${CLASSPATH}:
{domain__deploy_root}:
{domain__deploy_root}/conf/MPIListener.properties:
{domain__deploy_root}/healthvet/sds:
{domain__deploy_root}/healthvet/4sds/vha-stddata-client-18.0/lib/:
{domain__deploy_root}/healthvet/kaajee_security_provider:
{domain__deploy_root}/healthvet/kaajee_security_provider/common_pool_jars/commons-pool-
1.2.jar:
{domain__deploy_root}/healthvet/kaajee_security_provider/ common_pool_jars/commons-dbcp-
1.2.1.jar:
{domain__deploy_root}/healthvet/kaajee_security_provider/ common_pool_jars/commons-
collections-3.1.jar:
{domain__deploy_root}/healthvet/kaajee_security_provider/KaajeeDatabase.properties:
{domain__deploy_root}/healthvet/vlj:
{domain__deploy_root}/healthvet/vlj/testconnector/vljConnector-1.5.2.003.jar:
{domain__deploy_root}/healthvet/vlj/testconnector/vljFoundationsLib-1.5.1.002.jar:
{domain__deploy_root}/healthvet/vlj/testconnector/lib/jaxen-core.jar:
{domain__deploy_root}/healthvet/vlj/testconnector/lib/jaxen-dom.jar:
{domain__deploy_root}/healthvet/vlj/testconnector/lib/log4j-1.2.8.jar:
{domain__deploy_root}/healthvet/vlj/testconnector/lib/saxpath.jar:
{domain__deploy_root}/healthvet/vlj/testconnector/lib/xbean.jar:
{domain__deploy_root}/healthvet/Error! Bookmark not defined.psl/pslConfig_4.0.4.4.jar:
{domain__deploy_root}/healthvet/psd/PatSvcPkg.properties:

```

Set the java options (arguments) for each managed server to:

```

-DAPPLICATION_PROPERTY_FILE={domain__deploy_root}/conf/application.properties
-DLOG4J_PROPERTY_FILE={domain__deploy_root}/conf/log4j.properties
-Dweblogic.alternateTypesDirectory=/ {domain__deploy_root}/healthvet/kaajee_security_provider
-Dgov.va.med.environment.servertype=weblogic
-Dgov.va.med.environment.production=true (set to false in test domains)

```

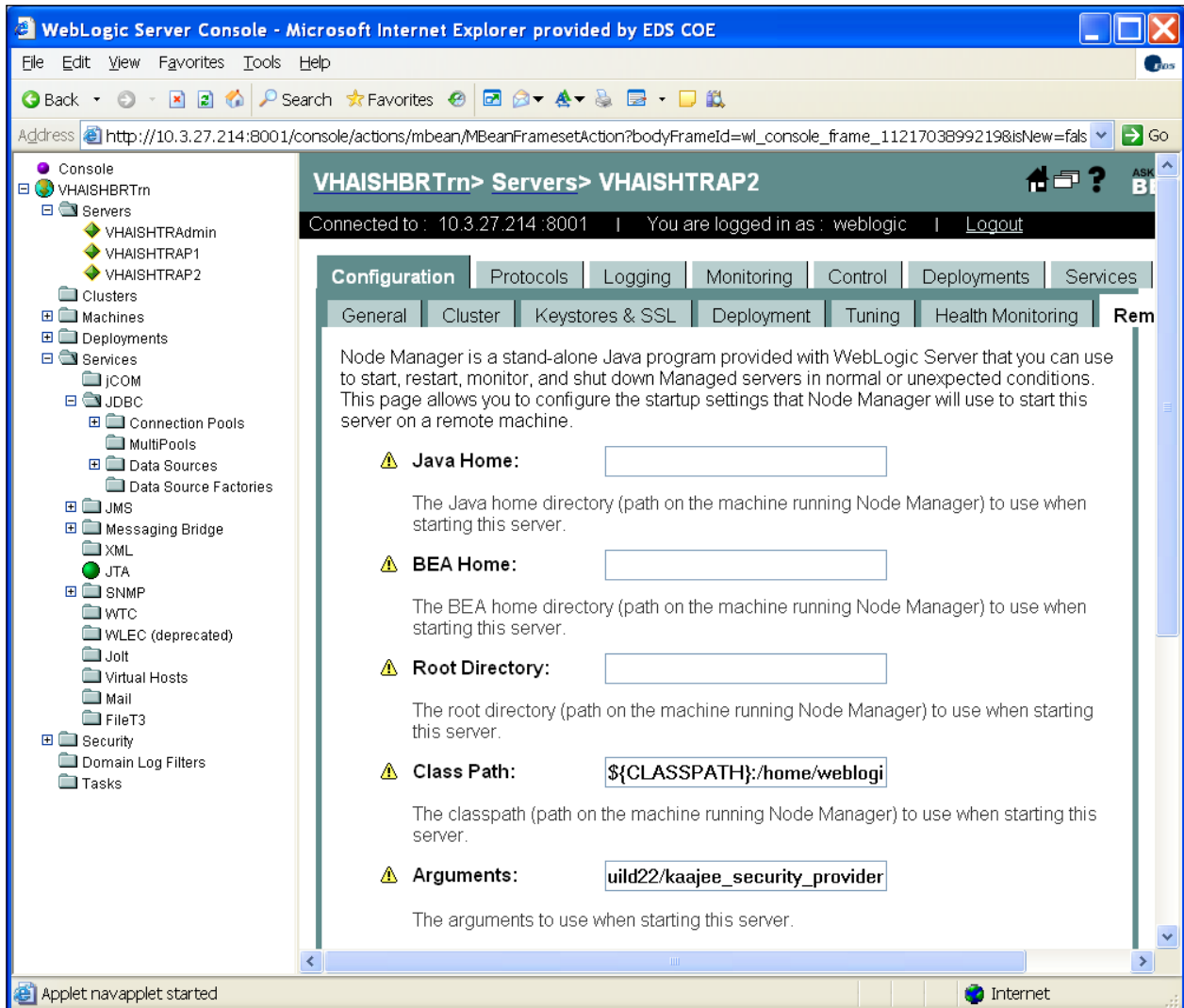
**NOTE:** When setting java arguments in the text box on the console, each argument is delimited by a space. For example:

```

-DjavaArgument=argument -DjavaArgument=argument

```

WebLogic managed server Remote Start Tab sample screen shot:



5. Create a JDBC Connection Pool for the SDS package in each domain and deploy it to each managed server in the domain. Since SDS is a read-only database, the Oracle User account for the SDS package may be shared between domains that use the same version of SDS. Name the pool:

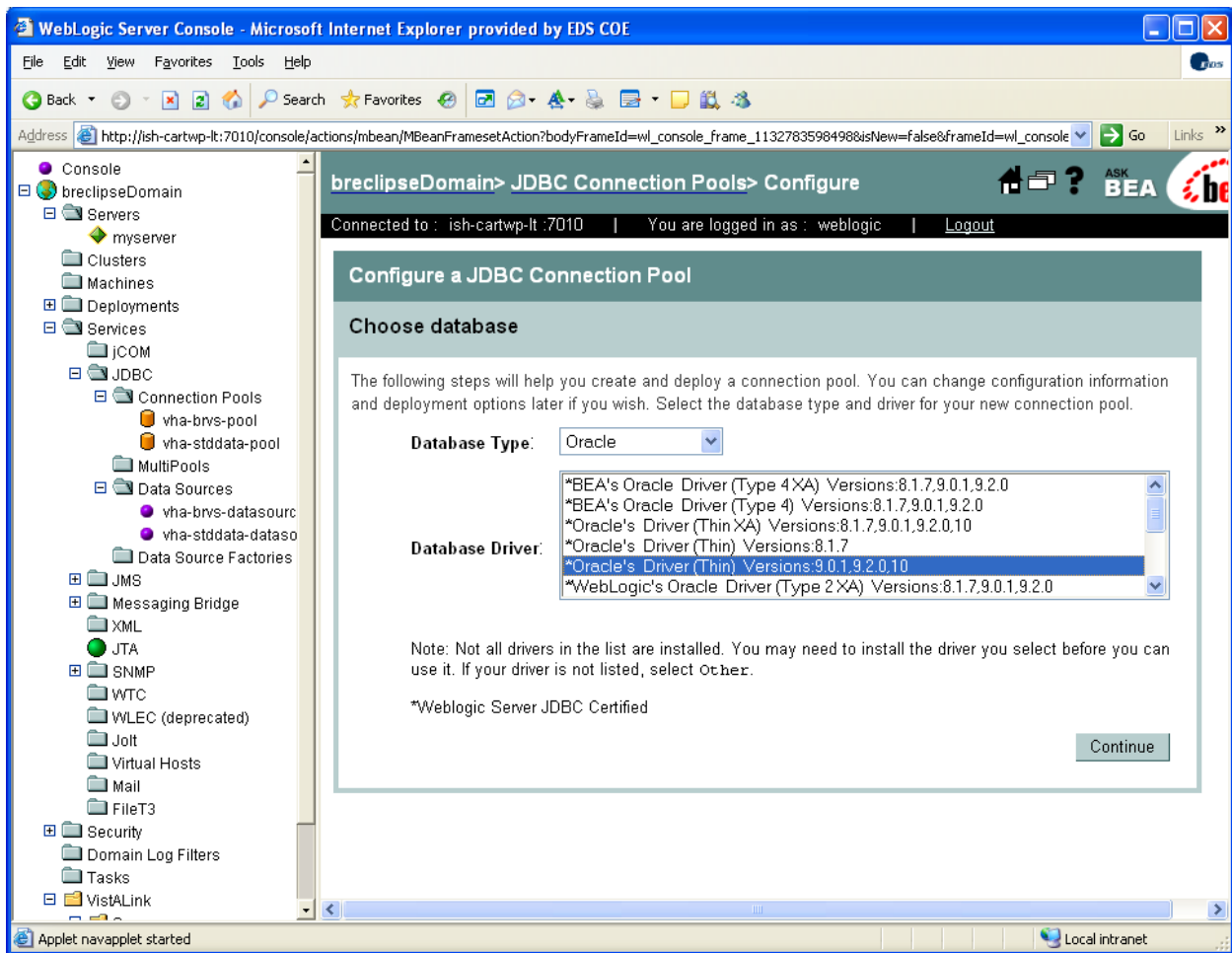
**vha-stddata-pool**

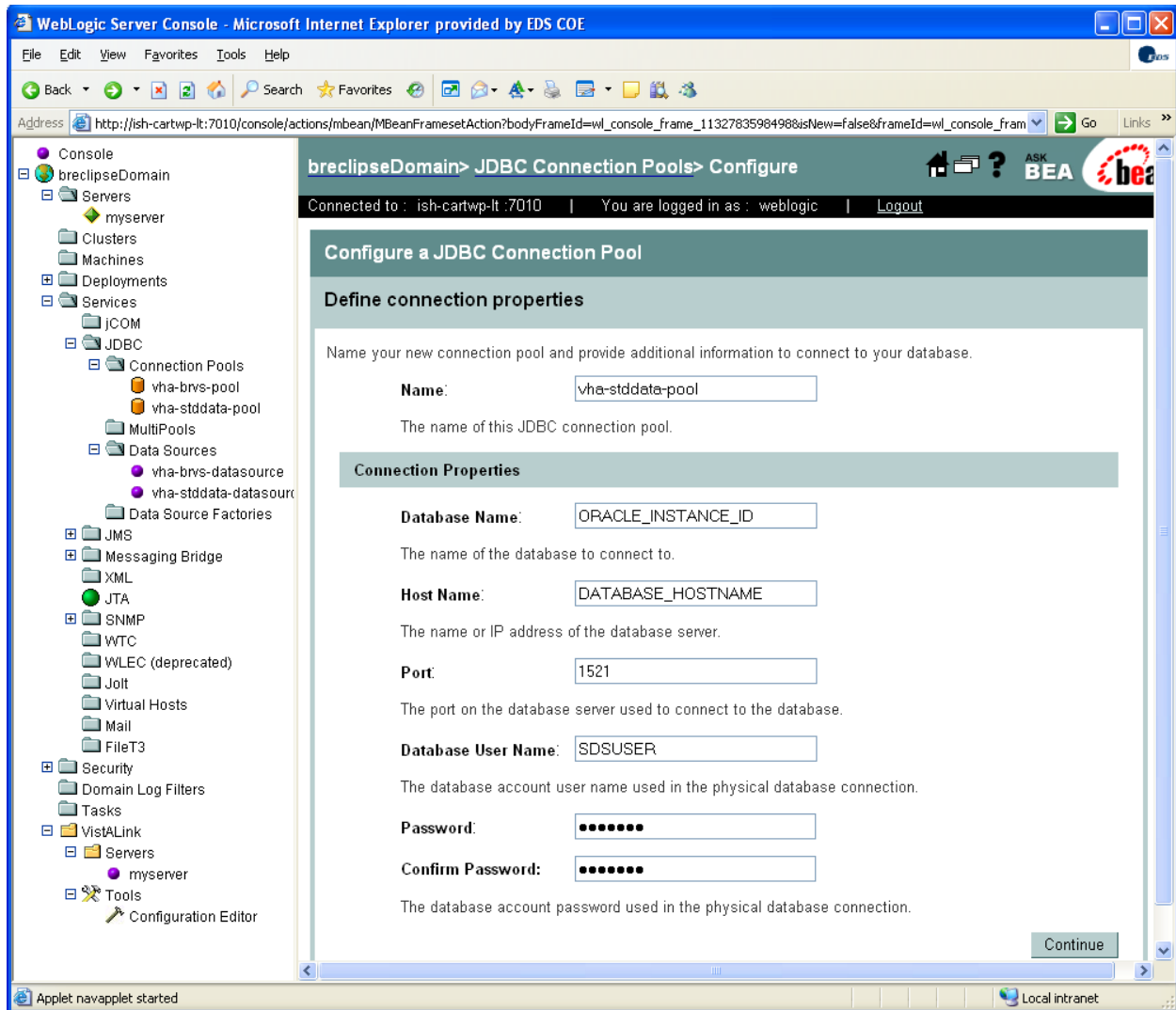
The URL will be constructed as you follow through the series of creation screens, an example is:

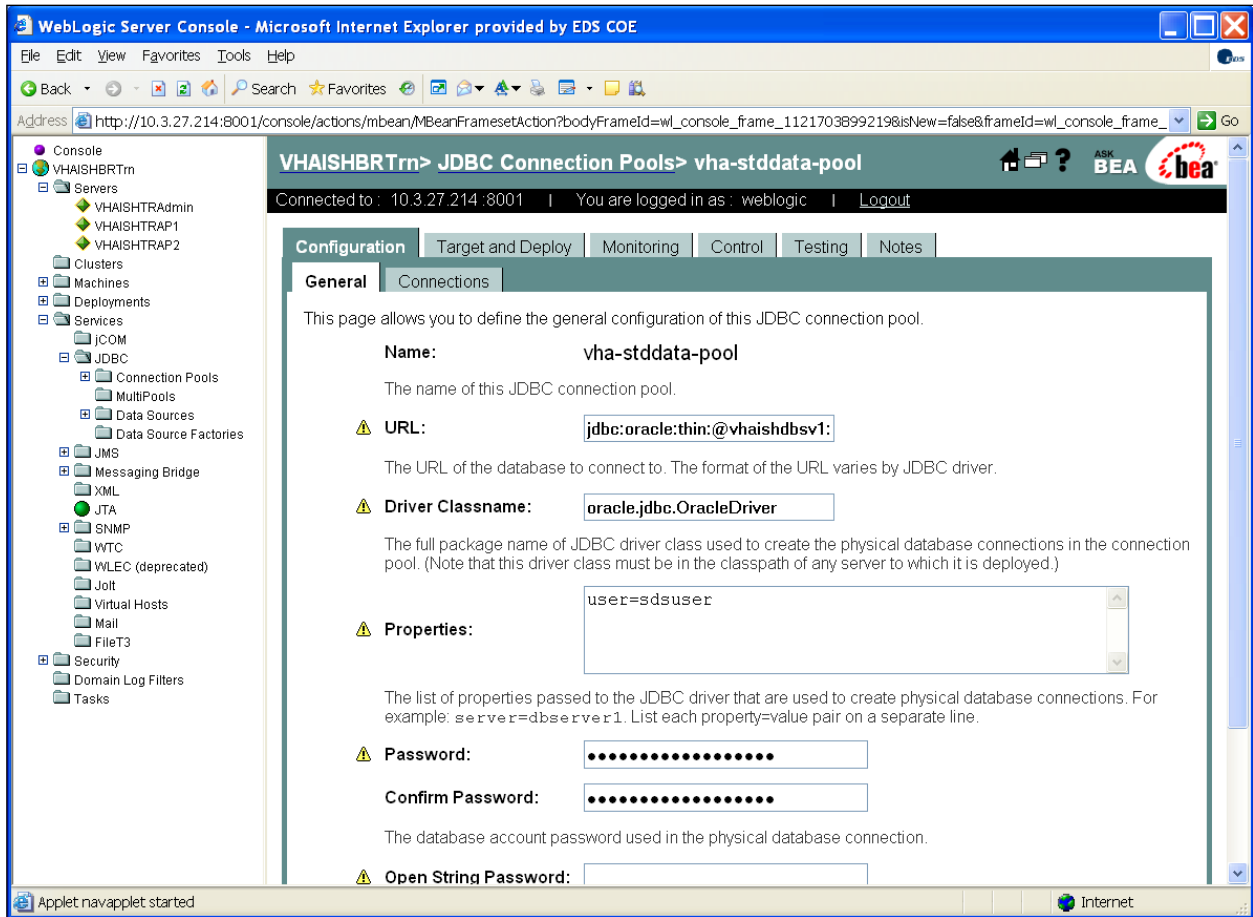
**jdbc:oracle:thin:@DATABASE\_HOSTNAME:1521:ORACLE\_INSTANCE\_ID**

When prompted, enter the database user, password, hostname, and other fields that correspond to the Oracle schema set-up with the SDS tables. Test the connection to verify proper operation.

SDS Connection Pool screens:



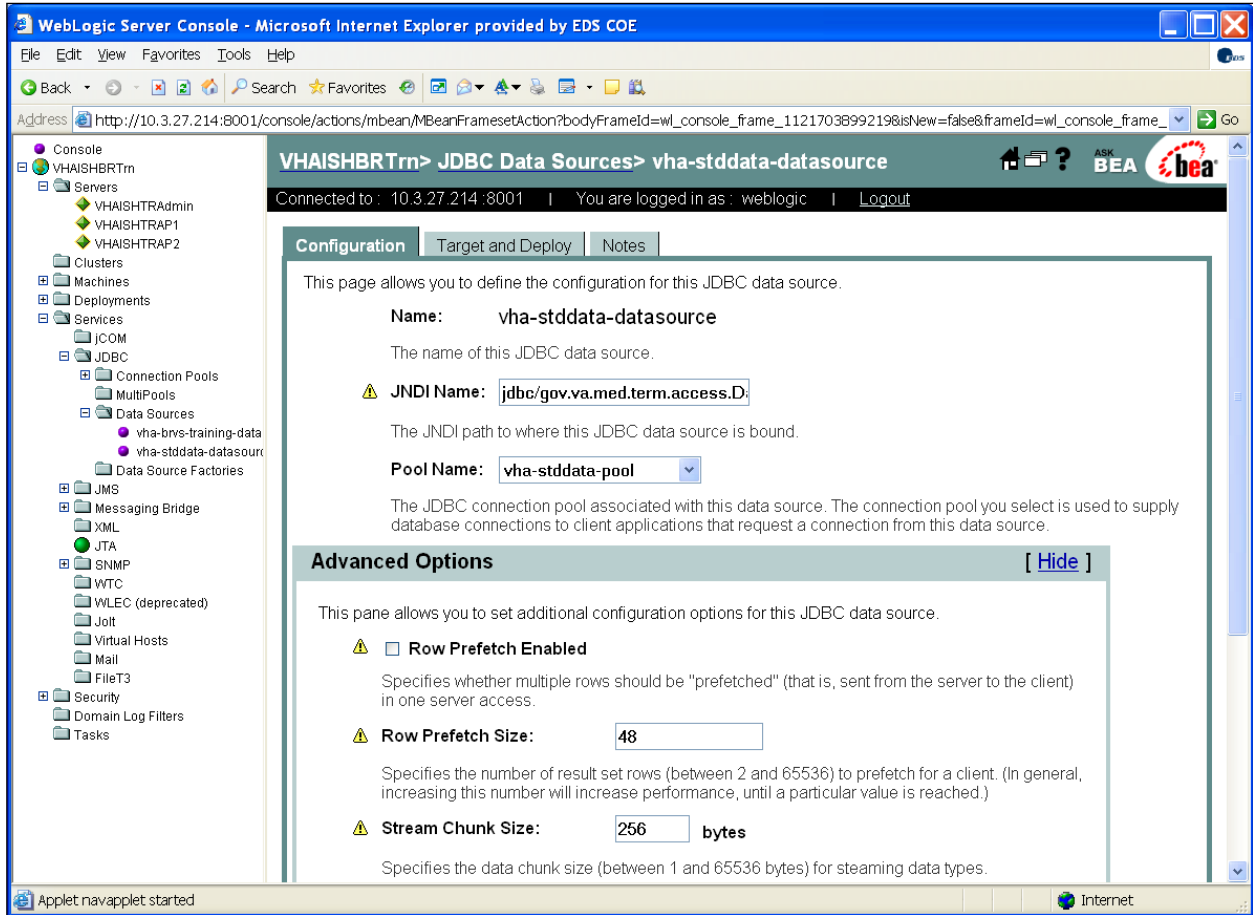






6. Create a JDBC Datasource for the SDS package in each domain, associate with the connection pool that was created in step #5 (**vha-stddata-pool**) and deploy it to each managed server in the domain. Name the datasource **vha-stddata-datasource**. Use the JNDI name: **jdbc/gov.va.med.term.access.Database**.

SDS Datasource screen:



7. Create a JDBC Connection Pool for the SDS package in each domain and deploy it to each managed server in the domain. Since SDS is a read-only database, the Oracle User account for the SDS package may be shared between domains that use the same version of SDS. Name the pool:

**vha-brvs-pool**

The URL is constructed as you follow through the series of creation screens, an example is:

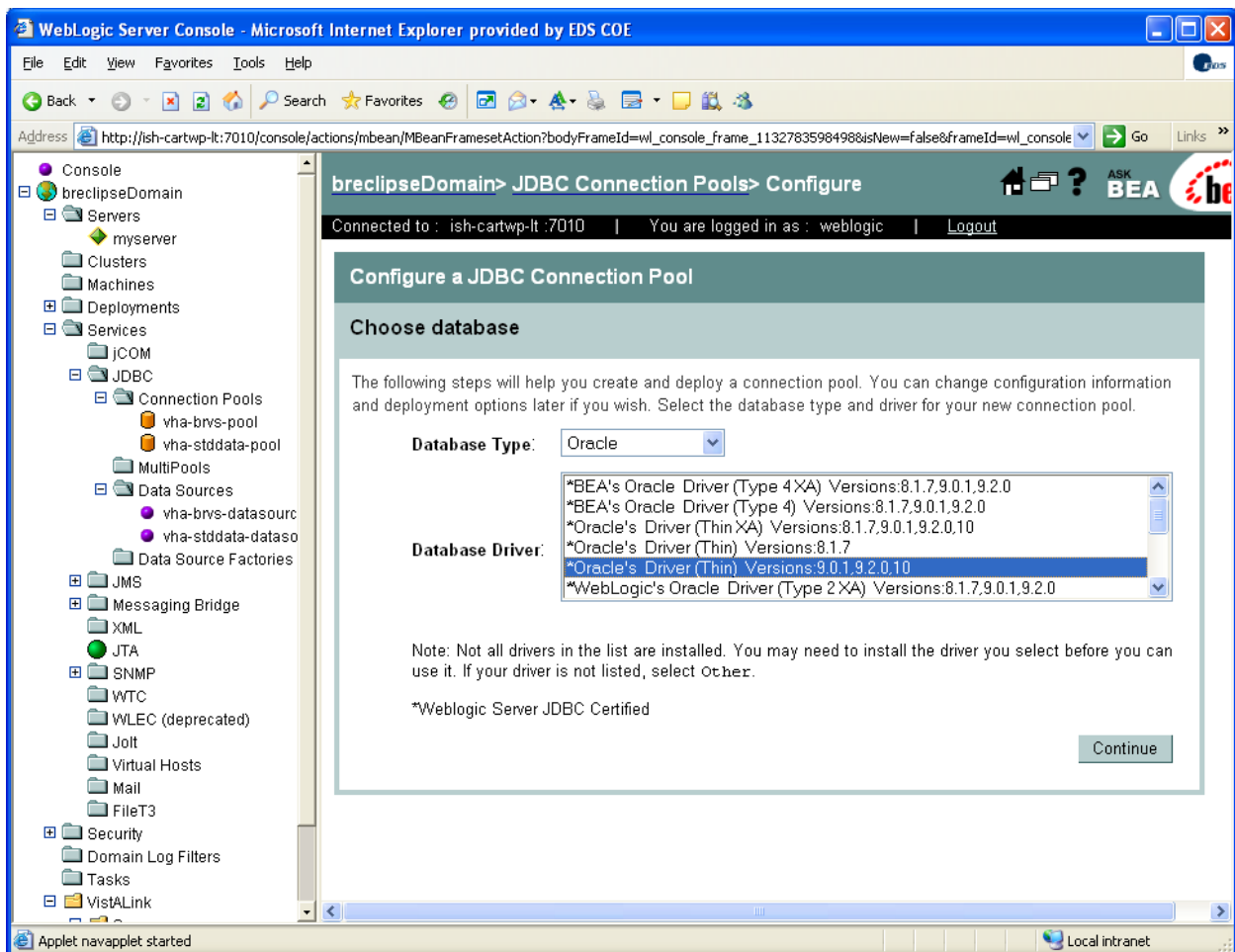
**jdbc:oracle:thin:@DATABASE\_HOSTNAME:1521:ORACLE\_INSTANCE\_ID**

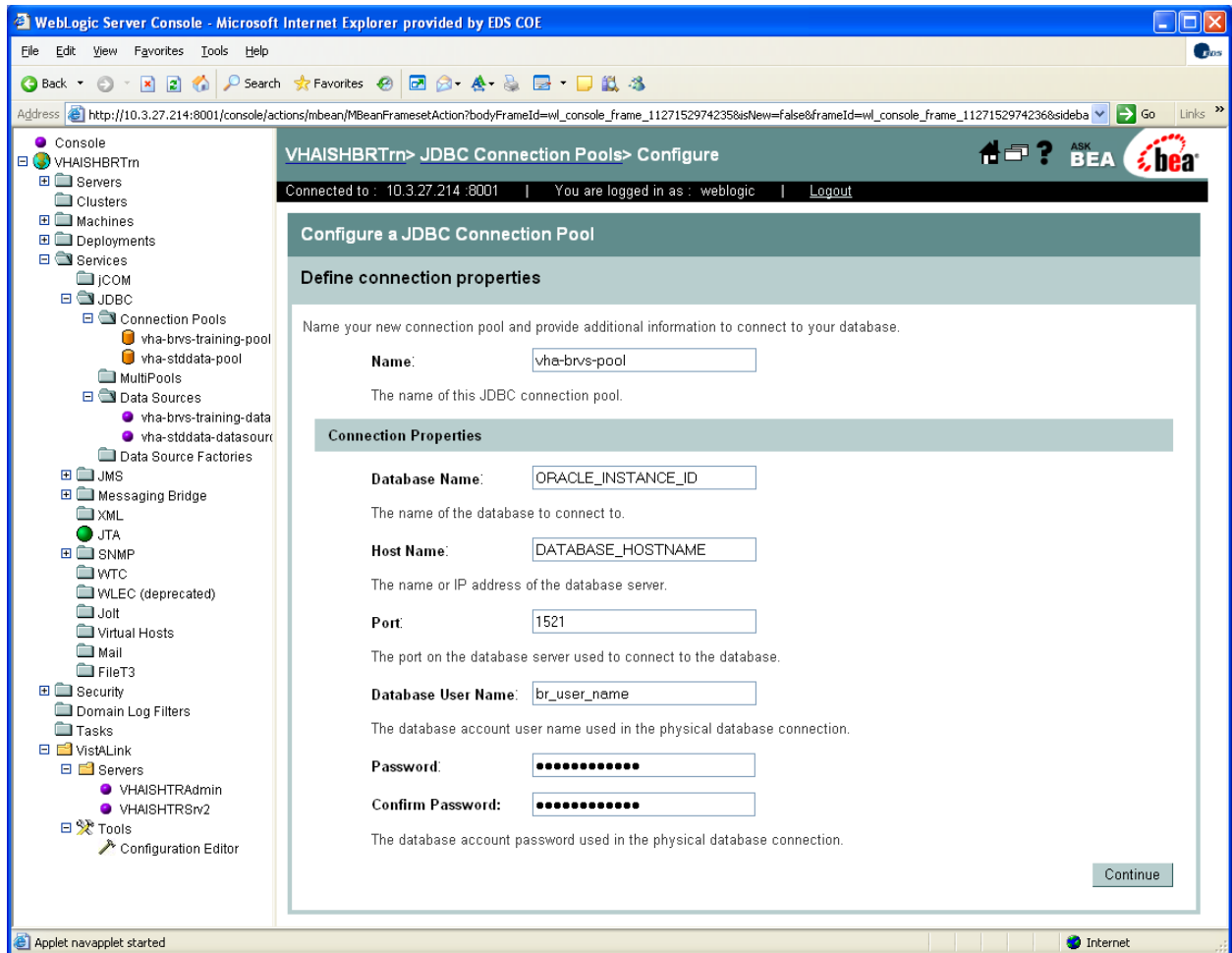
Test the connection to verify proper operation.

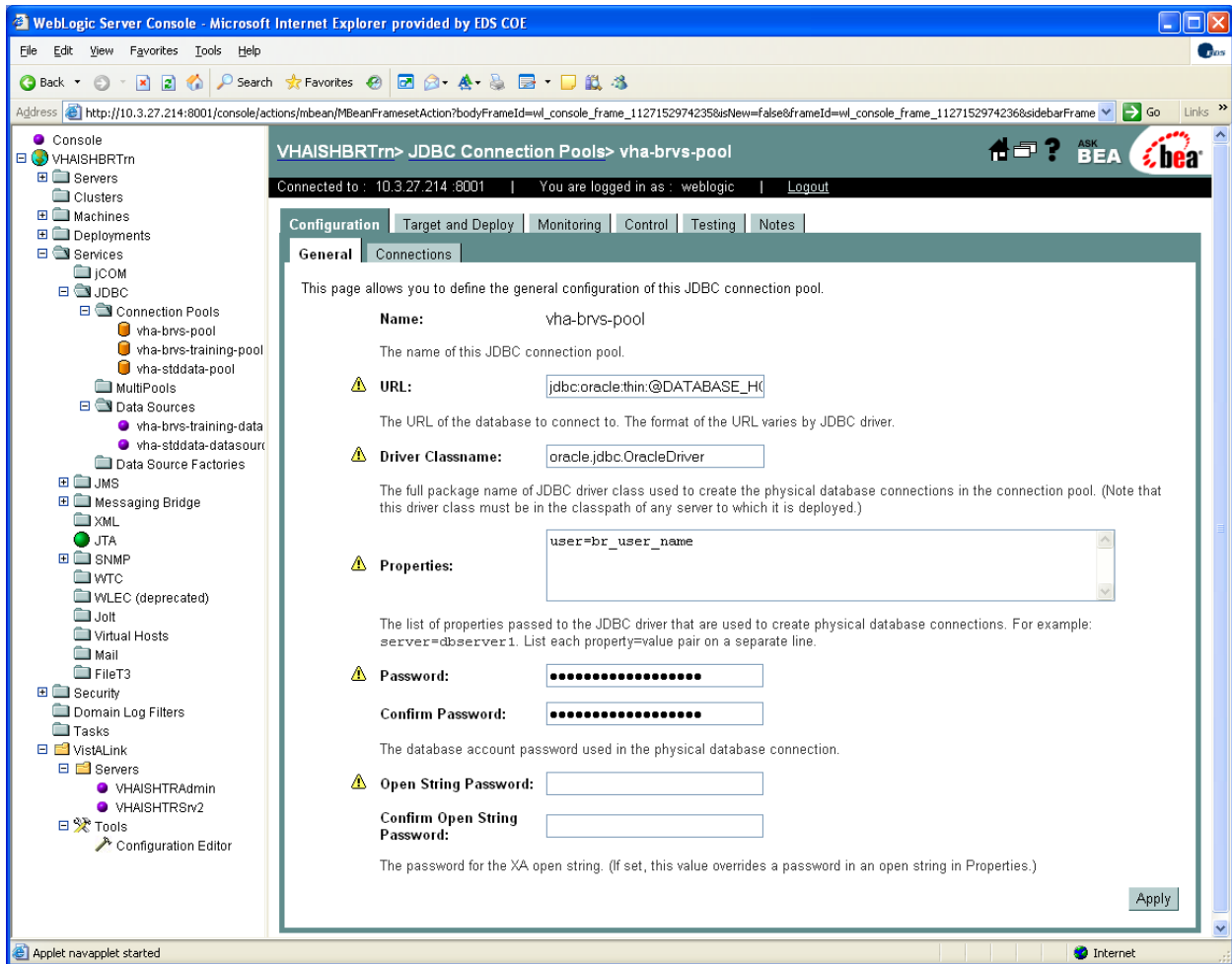
When prompted, enter the database user, password, hostname and other fields that correspond to the Oracle schema set-up for the Blind Rehabilitation domain in the ‘Create the Blind Rehabilitation Oracle Environment’ section of this document.

Advanced options should be set to appropriate values for the anticipated load on each domain.

BR Connection Pool screens:

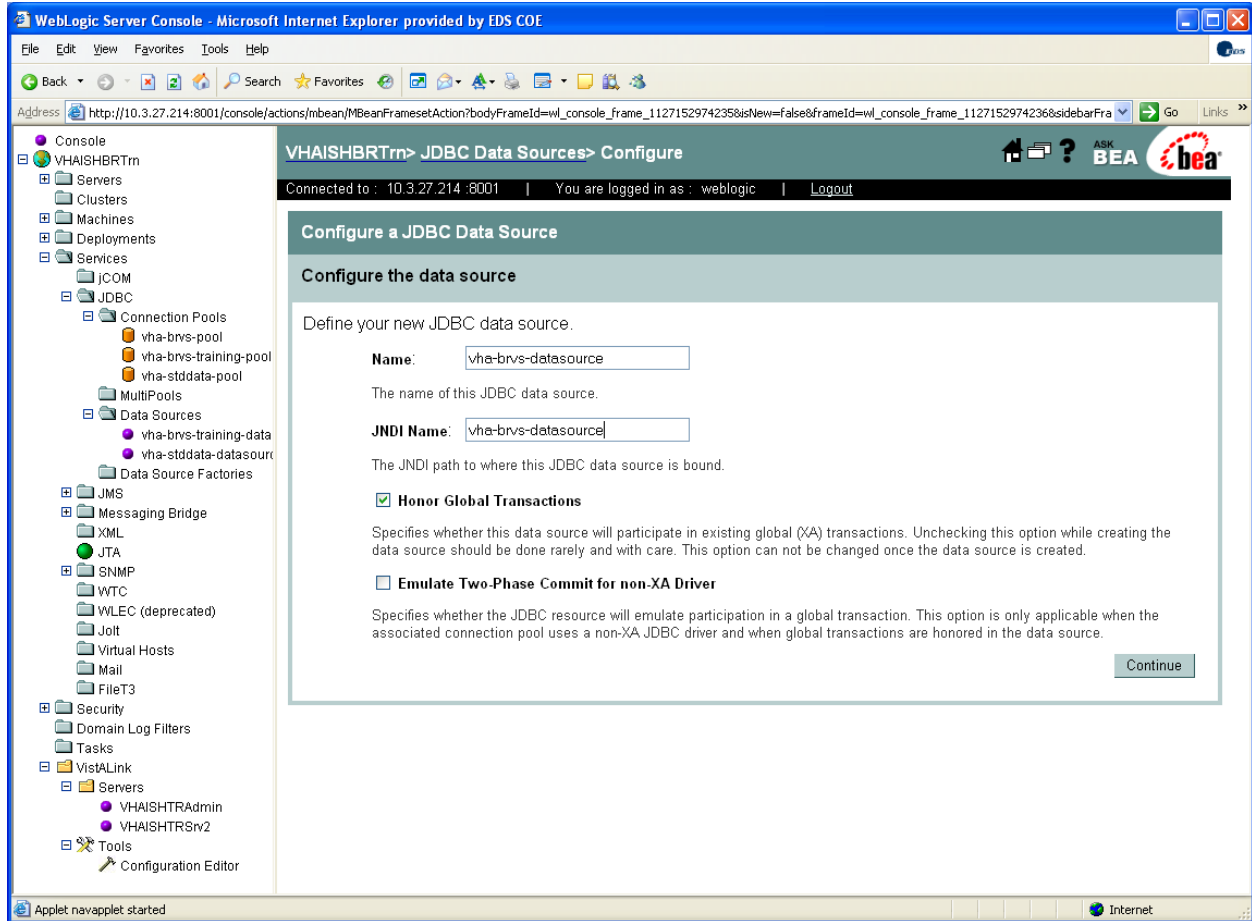


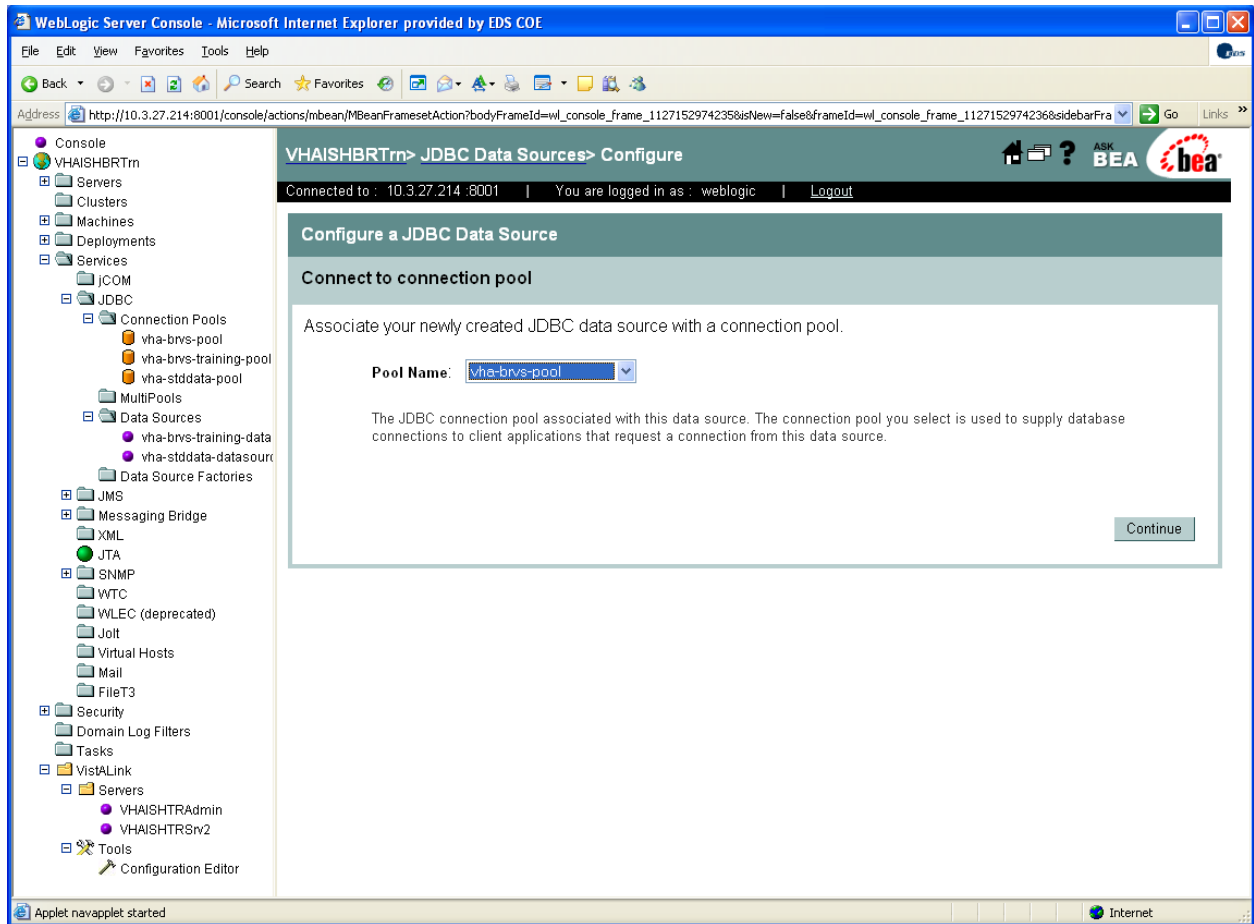


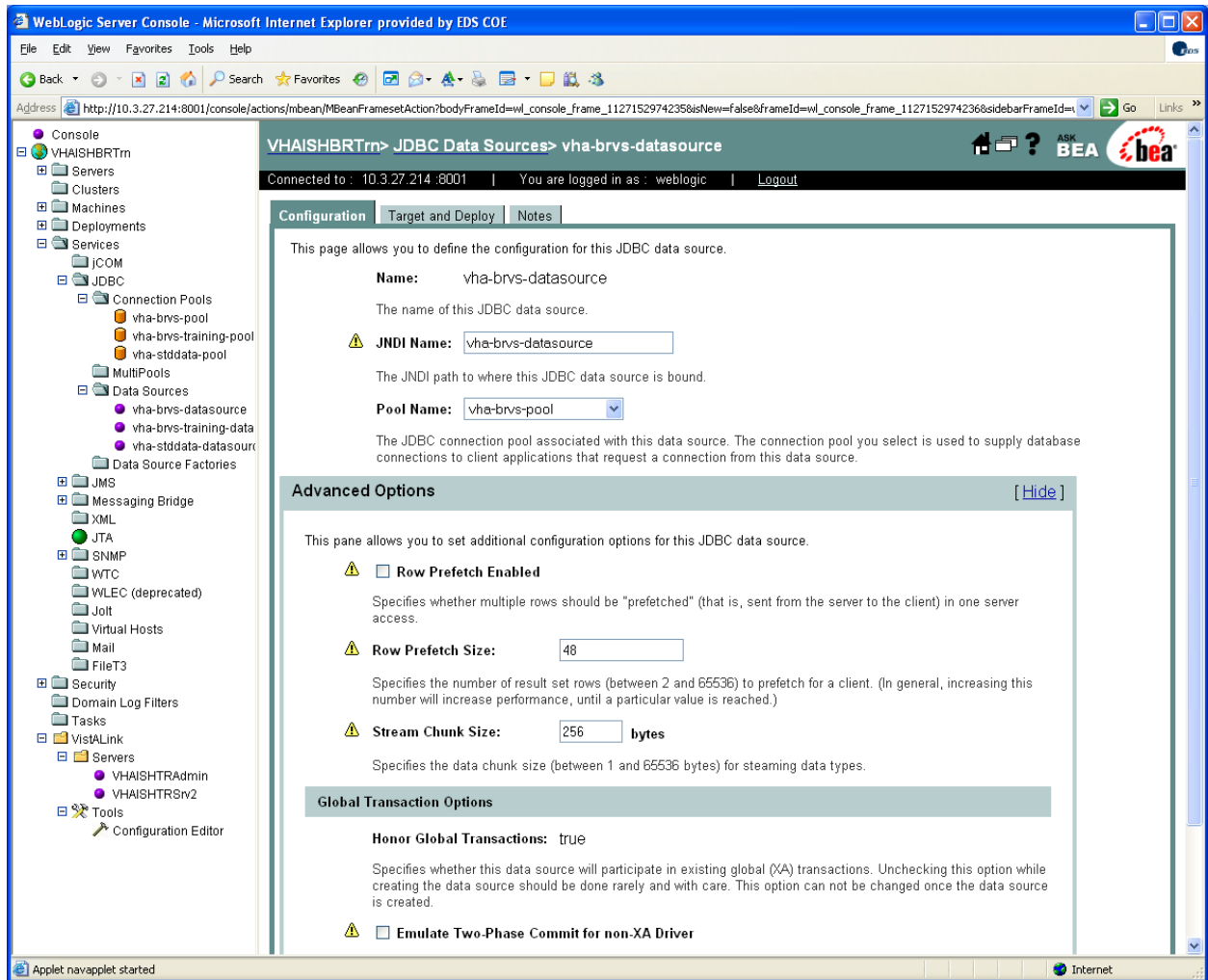


8. Create a JDBC Datasource for the BR package in each domain, associate with the connection pool that was created in step #7 (**vha-brvs-pool**) and deploy it to each managed server in the domain. Name the datasource: **vha-brvs-datasource**.  
Use the JNDI name: **vha-brvs-datasource**.

BR Datasource screens:



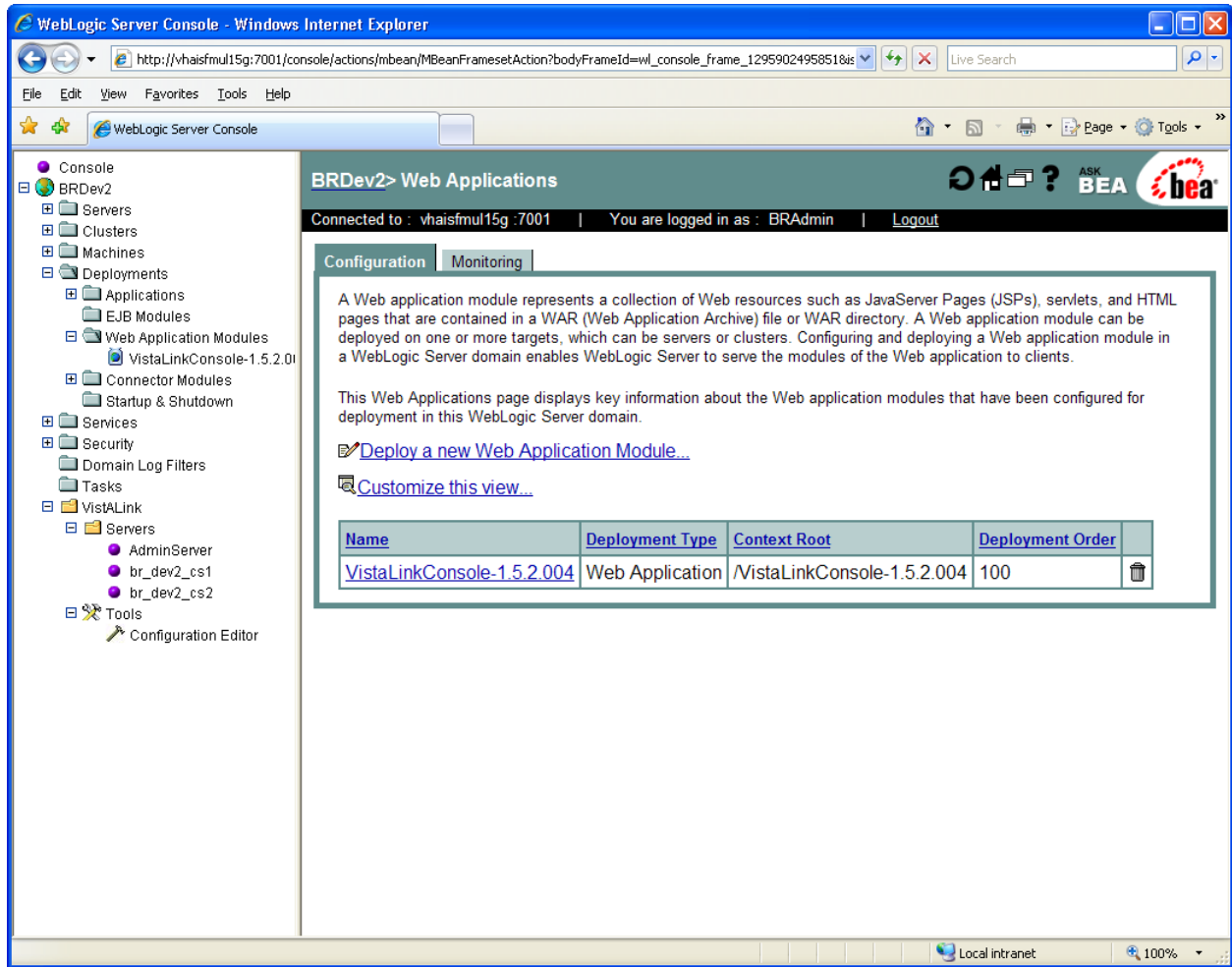




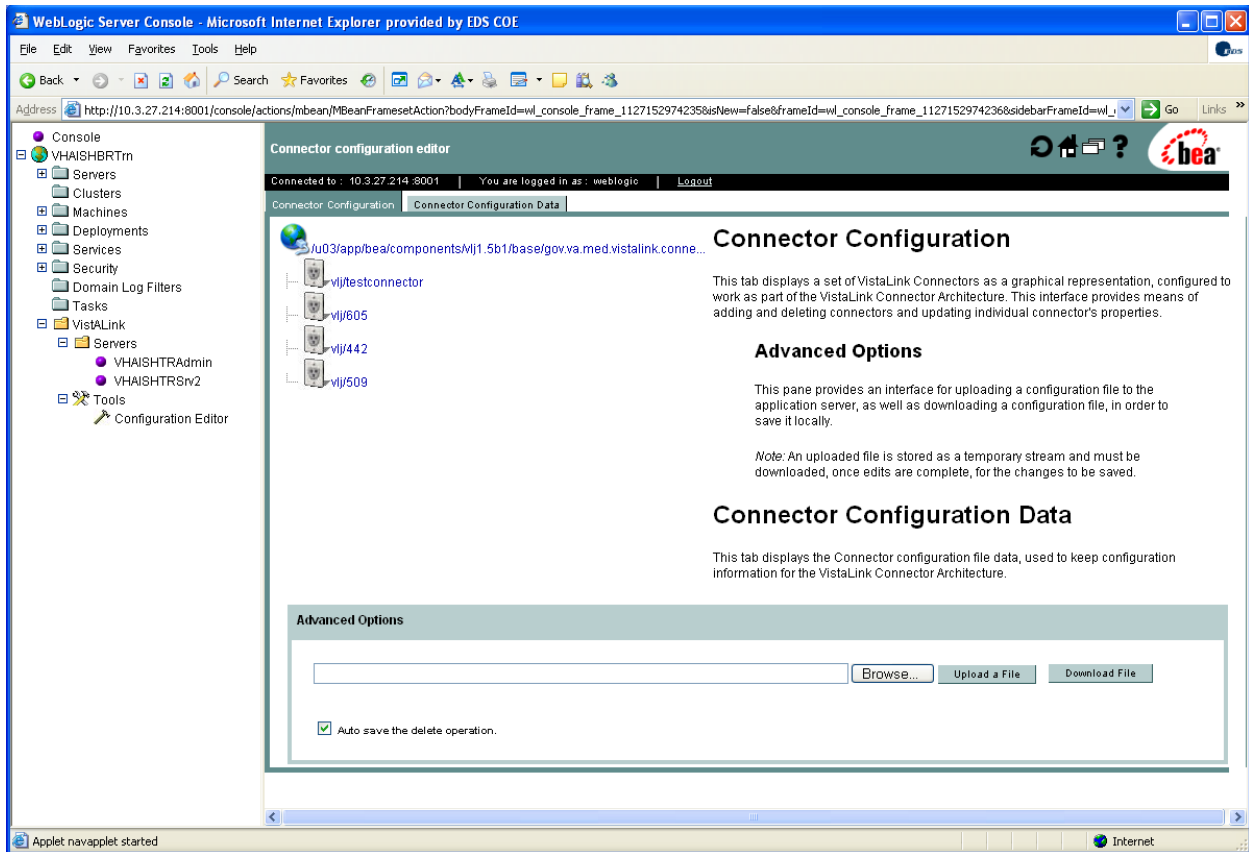
9. Deploy VistALink Console file:  
*VistaLinkConsole-1.5.2.004.war.*

Follow the VistALink installation instructions for this step in:  
**XOB 1.5.2.004 Install Guide.doc.**

VistALink Console Screens:







10. Deploy VistALink with a testconnector and a connector for each VistA server with which the domain will communicate. Follow the VistALink installation instructions for this step in: **XOB 1.5.2.004 Install Guide.doc.**

VistALink Connector Screen:

The screenshot shows the WebLogic Server Console interface. The left-hand navigation pane is expanded to show the 'Connector Modules' section under the 'VHAISHBRTm' domain. The main content area is titled 'VHAISHBRTm > Resource Connectors' and shows the 'Configuration' tab selected. Below the title bar, there is a status bar indicating the connection details and user information. The main content area contains a descriptive paragraph about connectors, followed by a table listing the configured connector modules. The table has four columns: Name, URI, Deployment Order, and a delete icon. The 'testconnector' module is highlighted in blue.

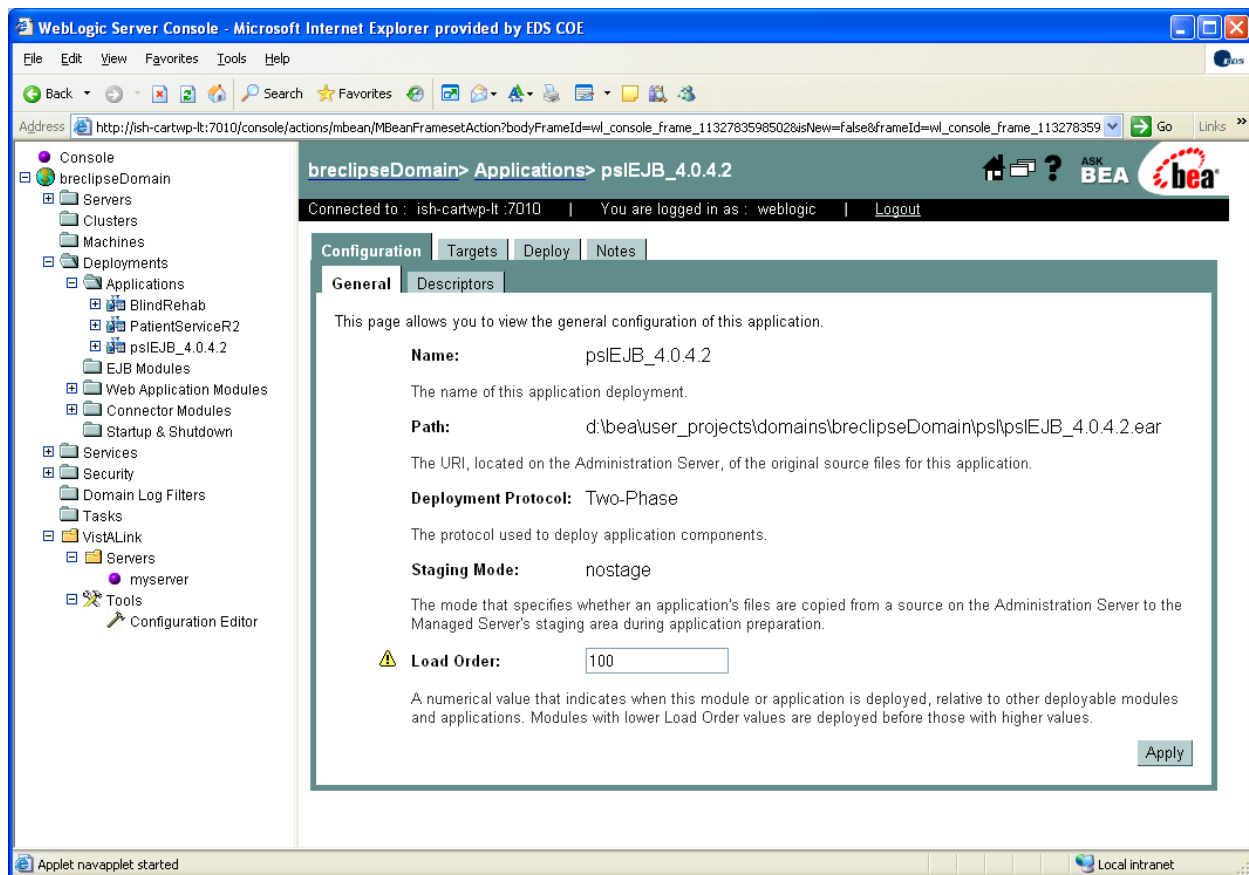
Name	URI	Deployment Order	
<a href="#">testconnector</a>	testconnector	100	
<a href="#">vlj442</a>	vlj442	100	
<a href="#">vlj509</a>	vlj509	100	
<a href="#">vlj605</a>	vlj605	100	
<a href="#">vlj663</a>	vlj663	100	
<a href="#">vlj663A4</a>	vlj663A4	100	
<a href="#">vlj678</a>	vlj678	100	

11. Edit the Person Service Lookup (PSL) configuration file(s) to match your server settings. Follow the PSL installation instructions for this step in the installation document: *Sys Admin\_Dev Guide.doc*.

The *PatientLookup.properties* configuration file is located within the **pslConfig\_4.0.4.4.jar** archive file. Open this archive with WinZip or other zip file utility program. Edit the *PatientLookup.properties* file and save the changes back to the archive file. You may need to extract the property file, edit it, and then add it back to the archive file depending on the zip file utility that you are using.

12. Deploy the PSL EJB ear file: **pslEJB\_4.0.4.4.ear**.

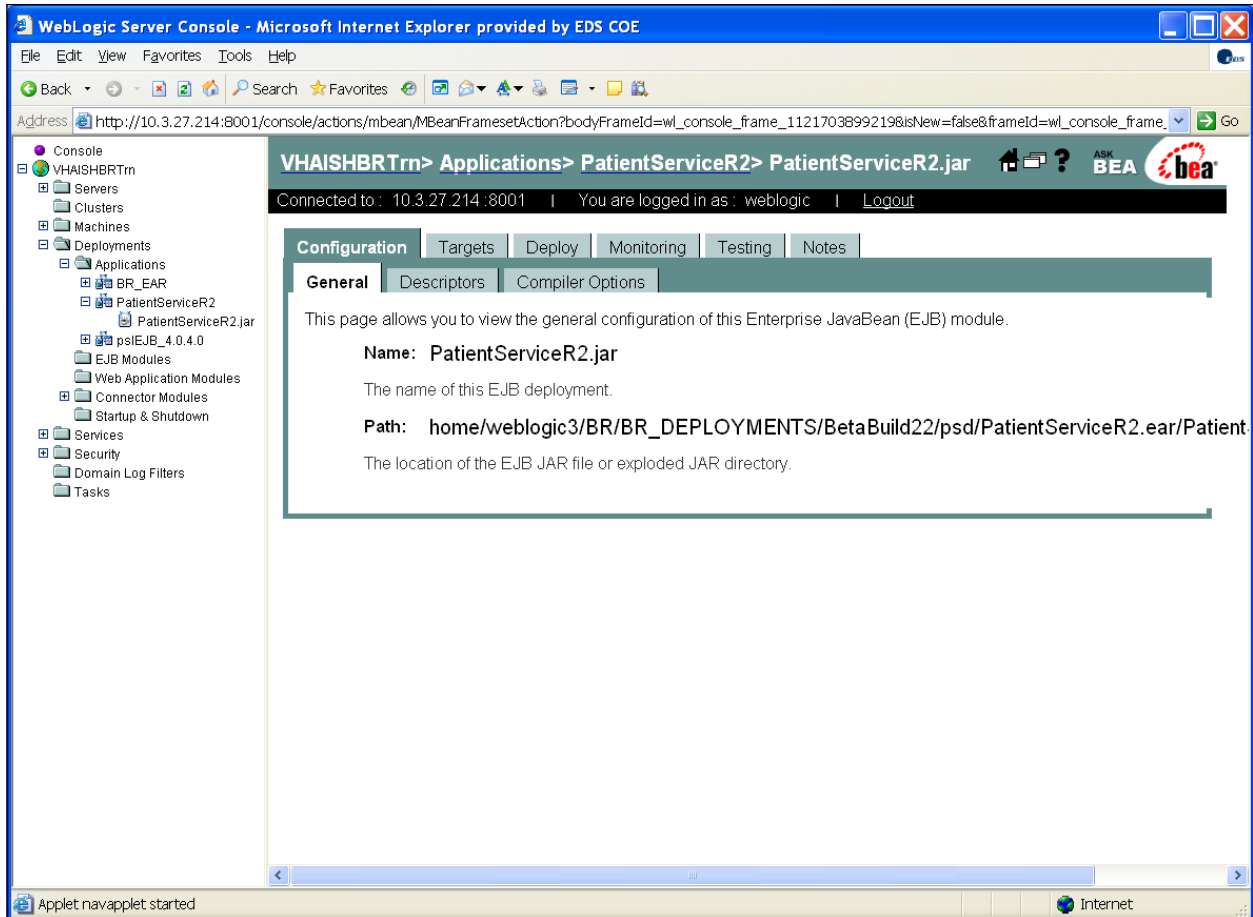
PSL Deployment screen:



13. Edit the Patient Service Construct (PSC/PSD) configuration file(s) to match your server settings. Follow the PSC installation instructions for this step in the installation document: *PatientServiceR2\_Installation\_Guide.doc*. The configuration file is *PatSvcPkg.properties*. It is not enclosed in an archive file like the PSL configuration file.

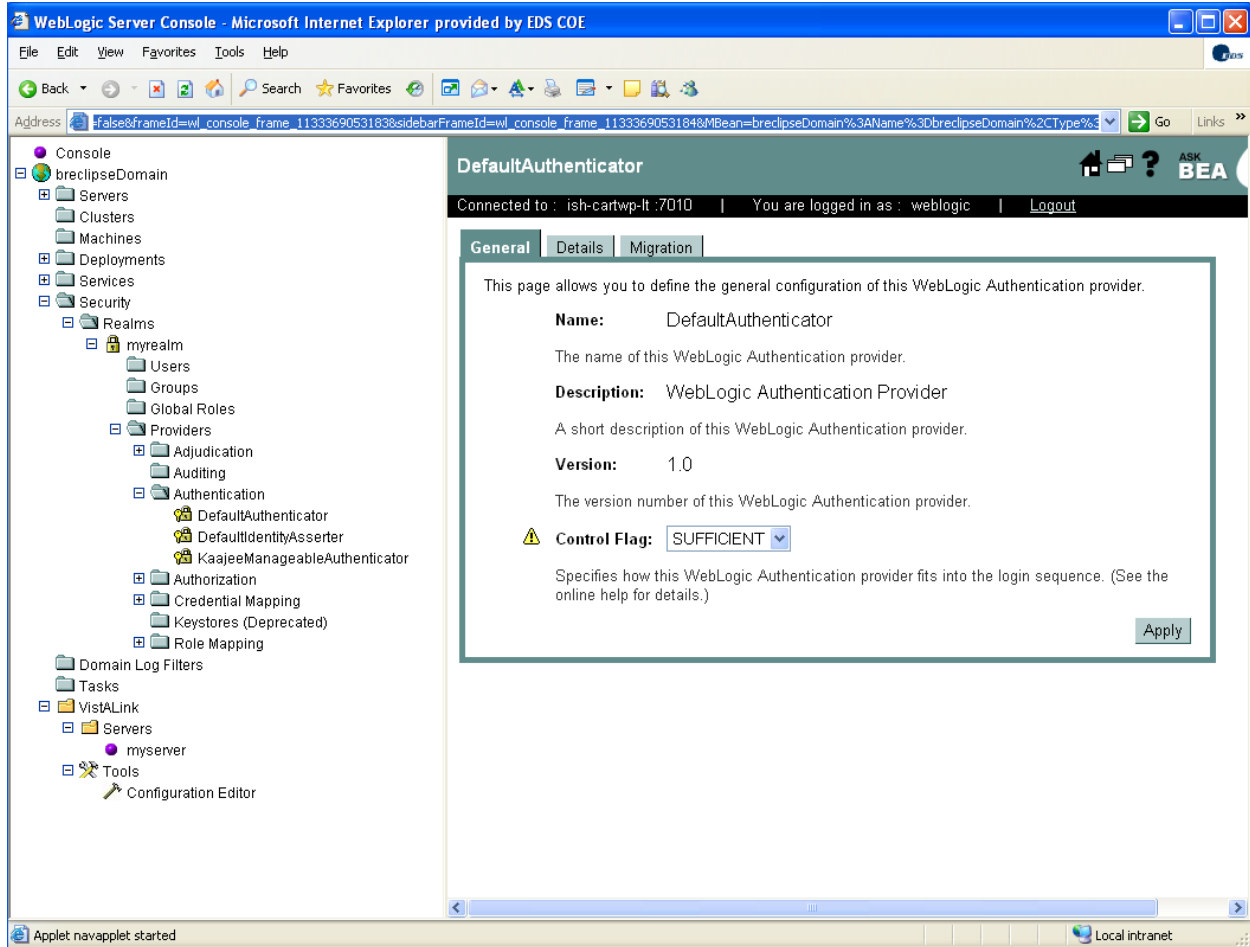
14. Deploy the **PatientServiceR2.ear** file.

PSC Deployment screen:



15. Configure the WebLogic Default Authenticator. In the WebLogic console go to Security-Realms-myrealm-Providers-Authentication. Click the DefaultAuthenticator, set the control flag option to 'Sufficient,' and apply the change.

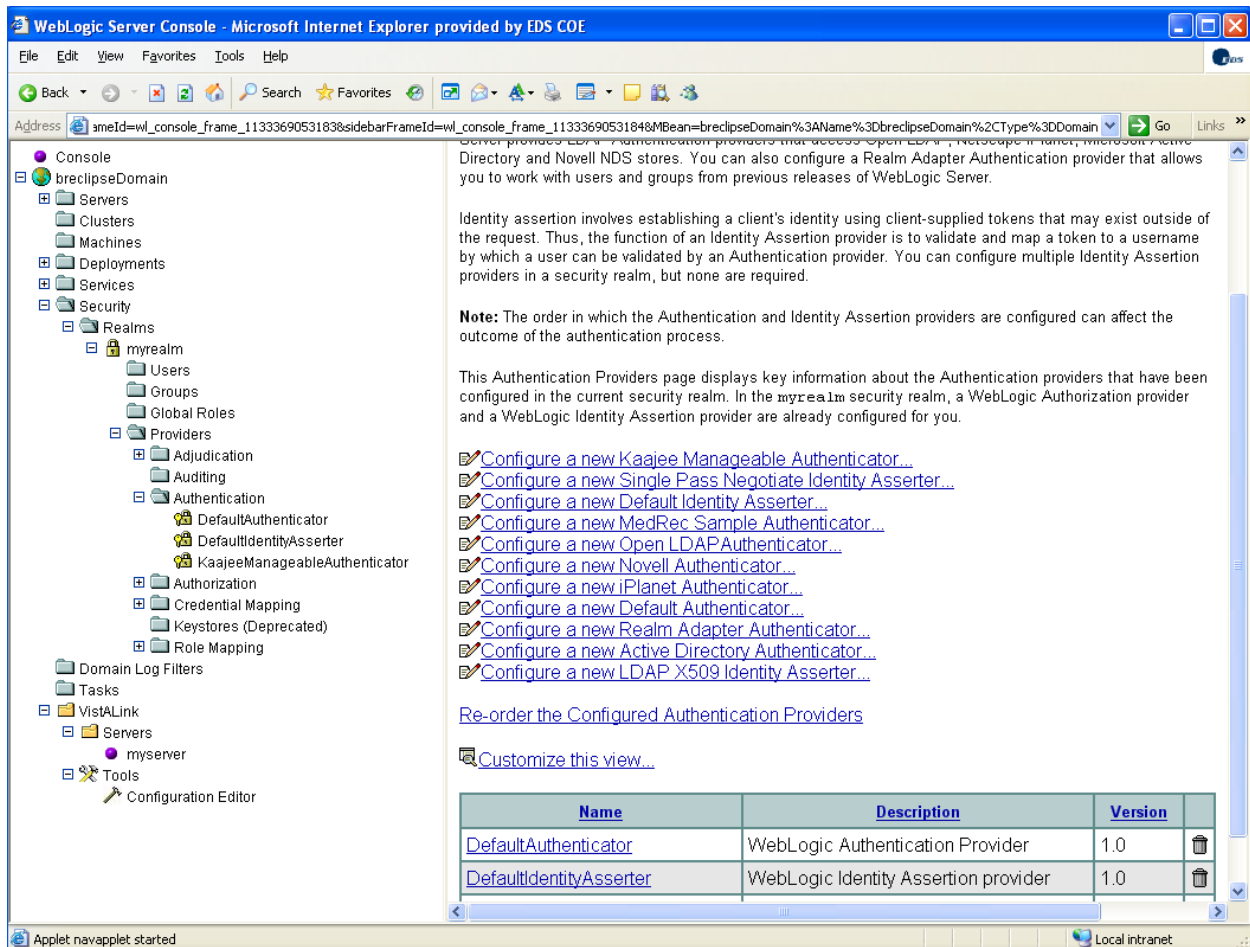
WebLogic Default Authenticator screen:



- Configure the KAAJEE Manageable Authenticator. In the WebLogic console go to Security-Realms-myrealm-Providers-Authentication. Click on ‘Configure a new Kaajee Manageable Authenticator...,’ set the control flag option to ‘Optional’ and apply the change.

**NOTE:** Configure a new Kaajee Manageable Authenticator’ will only appear in the WebLogic console if the WebLogic Administrator server classpath has the {domain\_\_deploy\_root}/kaajee\_security\_provider entry appended to it. See the KAAJEE Documentation for more information found in: *kaajee\_security\_provider \doc.*

Configure a new Kaajee Manageable Authenticator screen:



## Kaajee Manageable Authenticator screen:

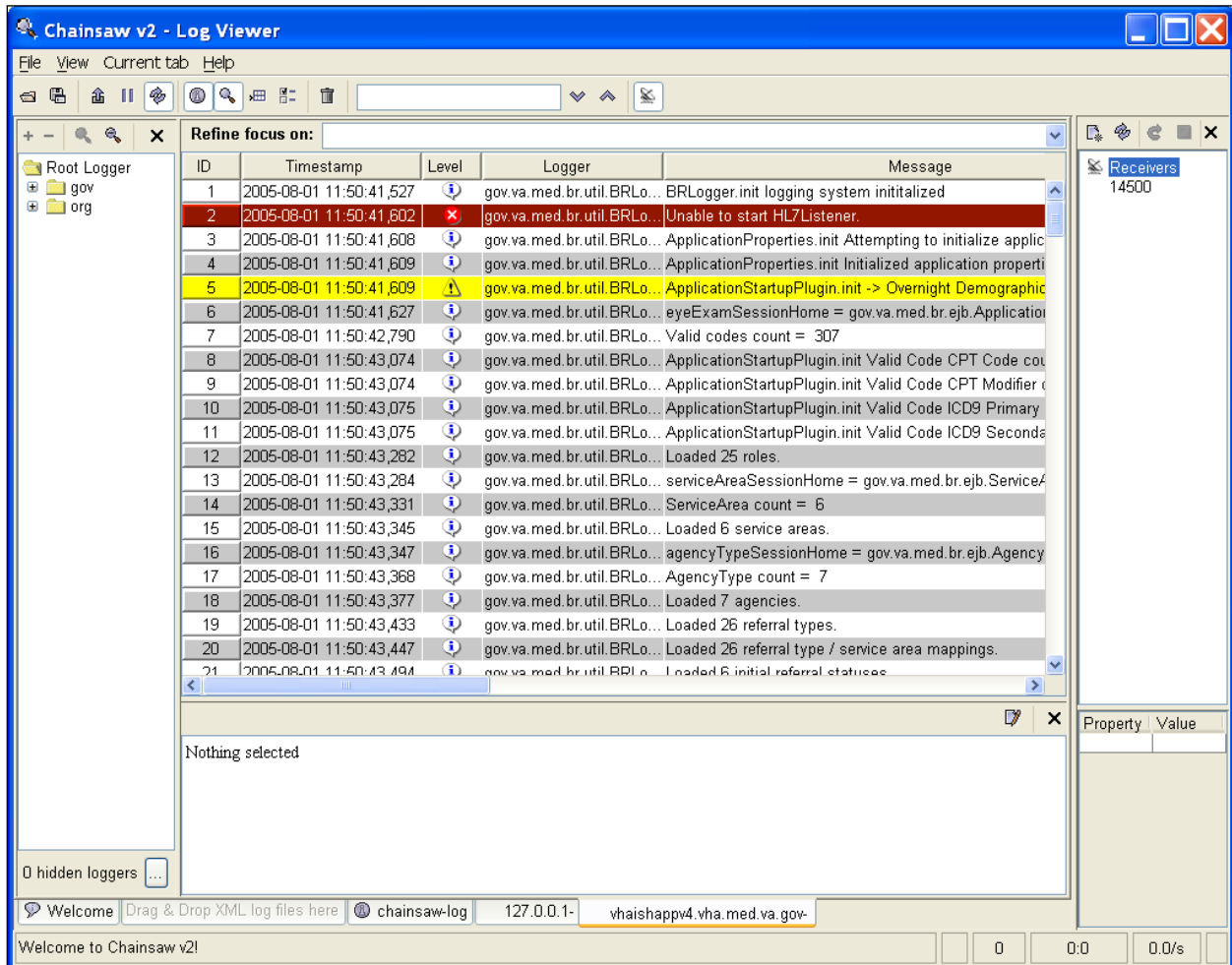
The screenshot displays the WebLogic Server Console interface in a Windows Internet Explorer browser. The browser's address bar shows the URL: `http://vhaisfmul15g:7001/console/actions/mbean/MBeanFramesetAction?bodyFrameId=wl_console_frame_12959024958518&...`. The console's left-hand navigation pane is expanded to show the 'Authentication' section under 'Providers', with 'KaajeeManageableAuthenticator' selected. The main content area is titled 'KaajeeManageableAuthenticator' and includes a status bar indicating the user is logged in as 'BRAdmin'. Below the title, there are two tabs: 'General' (selected) and 'Details'. The 'General' tab contains the following configuration details:

- Name:** KaajeeManageableAuthenticator  
The name of this custom Authentication provider.
- Description:** Weblogic Kaajee Manageable Authentication Provider  
A short description of this custom Authentication provider.
- Version:** 1.0.0.010  
The version number of this custom Authentication provider.
- Control Flag:** OPTIONAL (selected in a dropdown menu)  
Specifies how this custom Authentication provider fits into the login sequence.

An 'Apply' button is located at the bottom right of the configuration area. The browser's status bar at the bottom indicates 'Local intranet' and a zoom level of 100%.

- (Optional) Configure the LOG4J Chainsaw V2 application to receive logging events over a socket receiver. The Blind Rehabilitation application will send events to port 14500 by default. Each WebLogic domain and BR deployment should send its LOG4J events to a different socket to avoid confusing the source of events. Start up the Chainsaw application. See the Chainsaw documentation at <http://logging.apache.org/log4j/docs/chainsaw.html> for more information on log4j and chainsaw.

Chainsaw V2 sample screen:





### Sample Chainsaw V2 startup.xml file:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE log4j:configuration SYSTEM "log4j.dtd">
<log4j:configuration xmlns:log4j="http://jakarta.apache.org/log4j/" debug="true">

  <plugin name="LocalHost"      class="org.apache.log4j.net.SocketReceiver">
    <param name="Port" value="14500"/>
    <param name="Host" value="localhost"/>
    <param name="ReconnectionDelay" value="5000"/>
  </plugin>

  <root>
    <level value="debug"/>
  </root>
</log4j:configuration>
```

### Sample Chainsaw V2 startup script (Windows):

```
start /b javaw -classpath log4j-1.3alpha-7.jar;log4j-chainsaw-1.3alpha-7.jar;ugli-simple.jar;log4j-xml.jar;log4j-optional.jar org.apache.log4j.chainsaw.LogUI
```

### Sample Chainsaw V2 startup script (Linux):

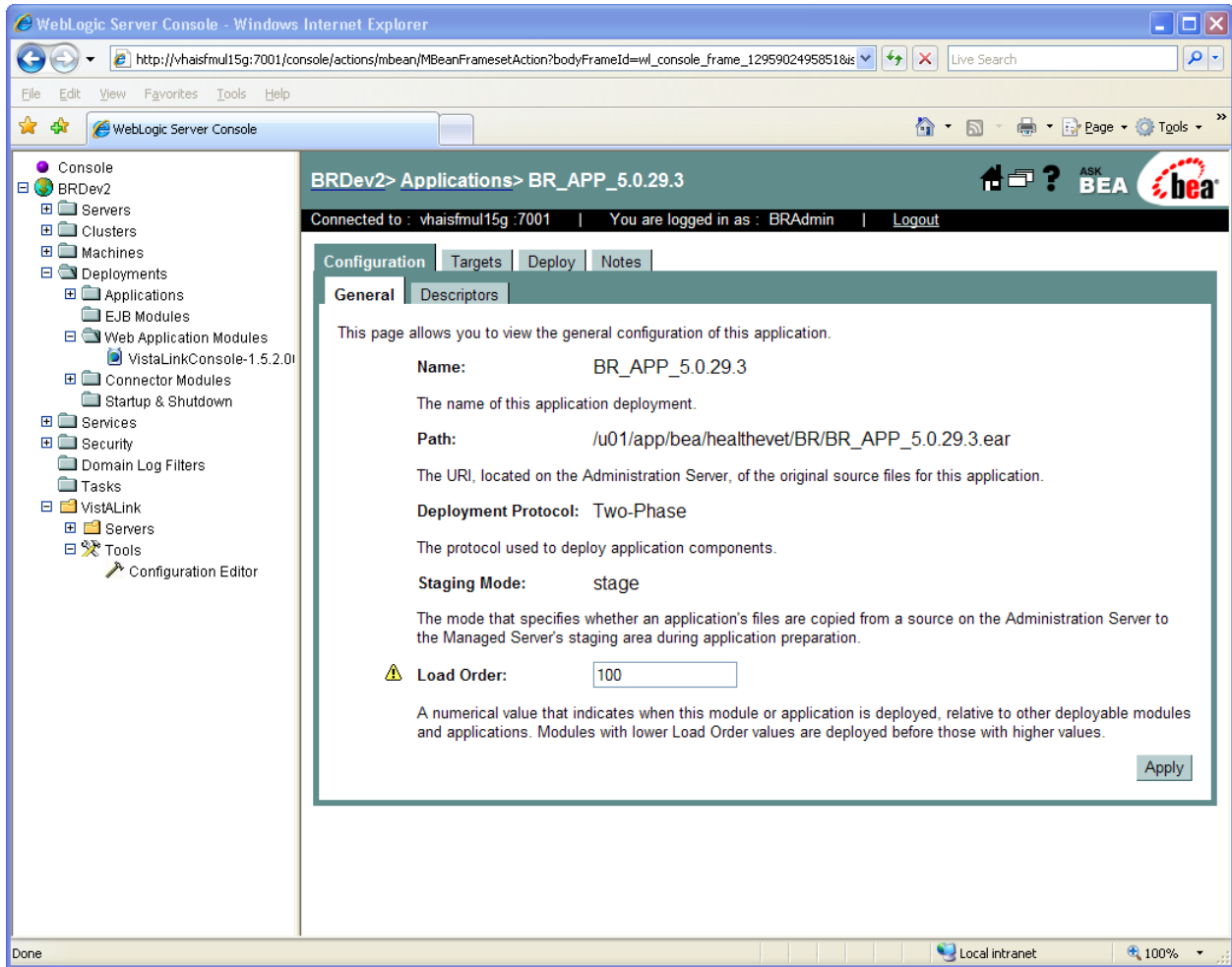
```
#!/bin/sh
java -classpath log4j-1.3alpha-7.jar;log4j-chainsaw-1.3alpha-7.jar;ugli-simple.jar;log4j-xml.jar;log4j-optional.jar org.apache.log4j.chainsaw.LogUI
```

18. Edit the *{domain\_\_deploy\_root}/conf/ MPIListener.properties* configuration file to match the MPI parameters for each domain. During startup, the Blind Rehabilitation application will start an MPI listener according to the properties in this file. Example *MPIListener.properties* file contained in appendix.
19. Edit the Blind Rehabilitation configuration files. BR uses two separate configuration files found in the *{domain\_\_deploy\_root}/conf/* directory: *application.properties* and *log4j.properties*. Change the settings in these files to match the configuration for each domain. Example property files are contained in the appendix.
20. Deploy the <sup>7</sup>**BR\_APP\_5.0.29.4.ear** file. Several dozen LOG4J messages should be logged during deployment and startup of the Blind Rehabilitation application. This is normal. Review these messages to determine if any failures occurred. Correct any issues found and restart the managed servers, if necessary, until no startup errors are generated.

---

<sup>7</sup> Changed version number of Blind Rehabilitation from BR\_EAR\_5.0.27.6 to BR\_APP\_5.0.29.4

BR Ear file deployment screen:



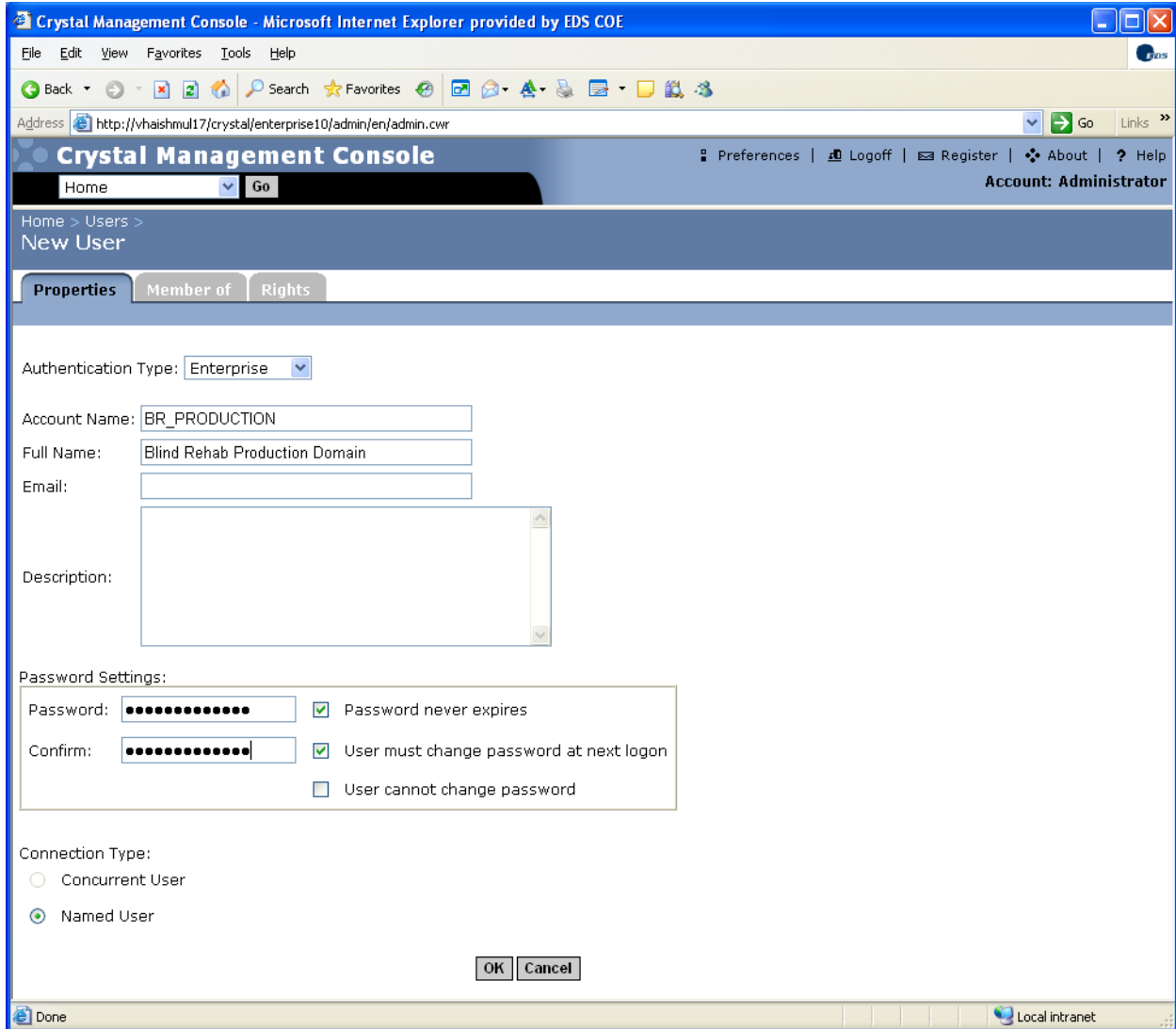
## Step 5: Installation of Crystal Enterprise

The Crystal Enterprise product is used to generate and display reports from Blind Rehabilitation data. Most reports are simple 'list' style reports and several are more complex 'summary' style reports with multiple sections. The Crystal Reports design tool is used to author these reports and the produced report files with '.rpt' extension are then published to the Crystal Enterprise server. The '.rpt' files are contained within the Blind Rehab deployment zip file and extracted into the *{domain\_\_deploy\_root}/reports* directory when unzipped. During runtime, the Blind Rehabilitation application gathers parameter information from the user through a criteria page for each report. After the criteria page is submitted, the application passes the parameters to Crystal Enterprise. Upon successful processing, Crystal Enterprise then displays the report within a report viewer page.

1. Install the Oracle Client software on the intended Crystal Enterprise server. Create an entry in the Oracle Client's *tnsnames.ora* file (or use the Oracle Net Configuration Assistant) to connect to the Oracle instance that contains the database. Consult Oracle documentation for assistance with configuring TNS Names entry.
2. Follow the WebLogic documentation and VA installation requirements for Crystal Enterprise Server.

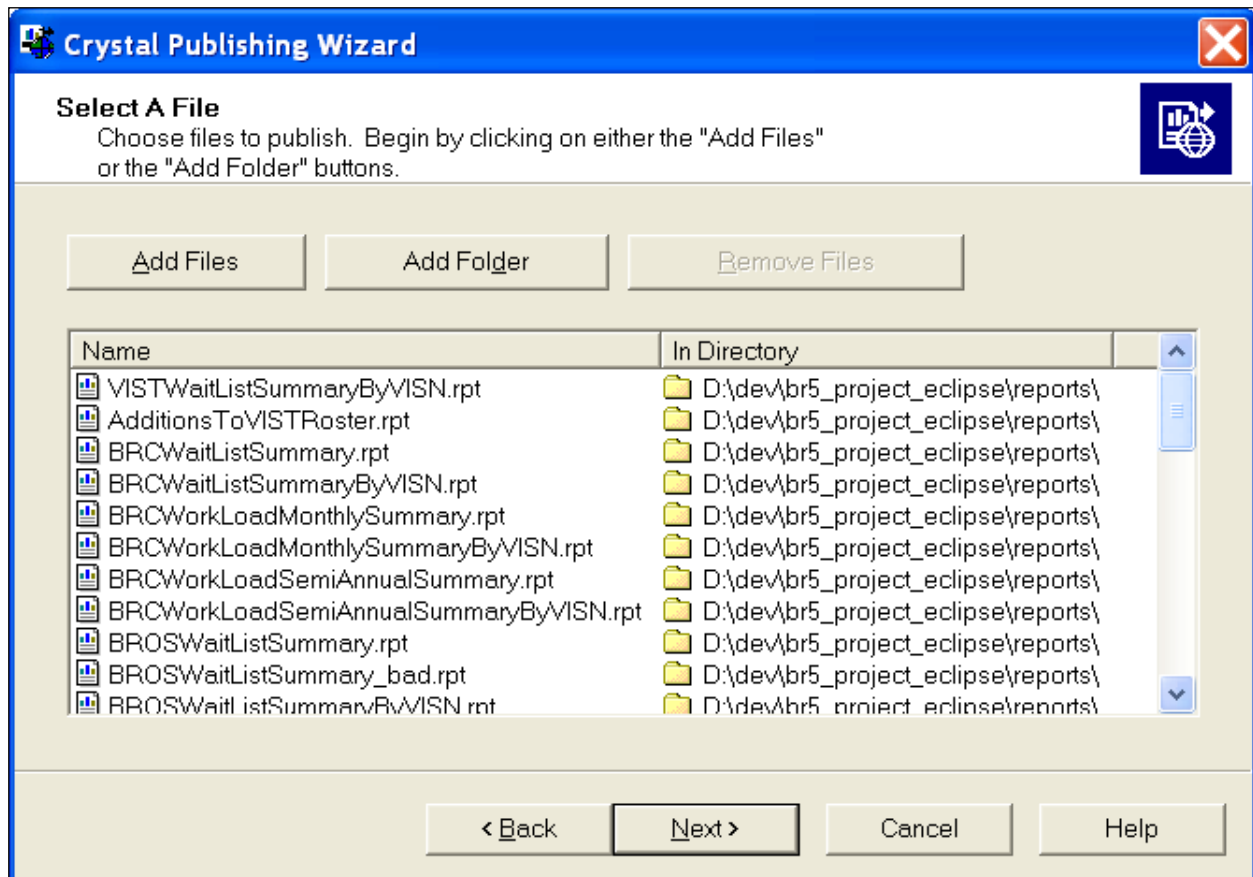
## Step 6: Configuration of Crystal Enterprise

1. Launch the Crystal Management Console application. Go to the User Administrator section. Create a Crystal Enterprise user account for each domain of Blind Rehab. Example: BR\_PROD, BR\_STAGING, BR\_TRAINING users.



2. Using the Crystal publishing wizard (preferred method) or the Crystal Management Console, deploy the reports files in the *{domain\_\_deploy\_root}/reports* directory for each domain. The Crystal Publishing wizard is a full client application that is installed on the Crystal Enterprise Server during installation and can be optionally installed on System Administrator client workstations.

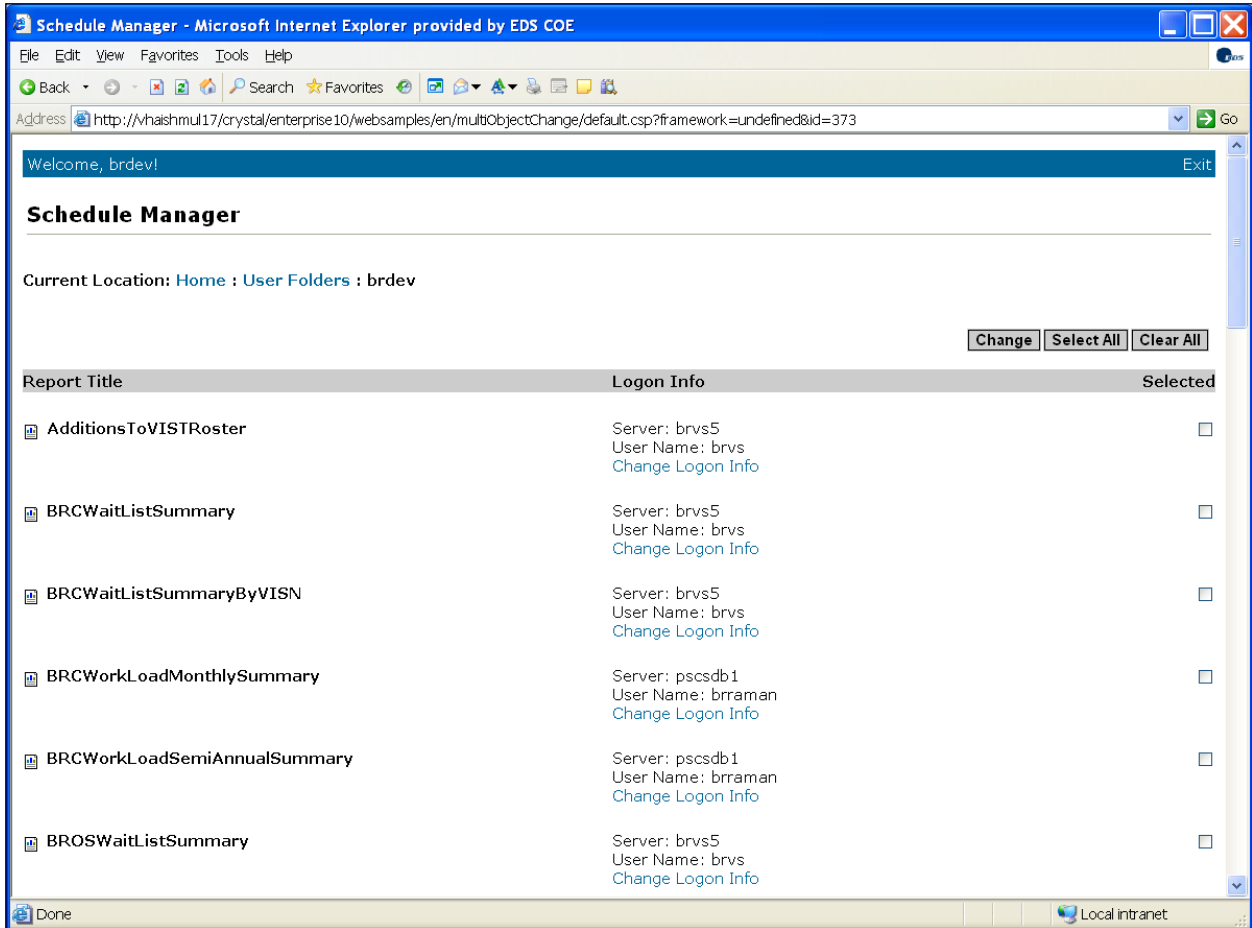
Crystal publishing wizard screen:



3. Edit the report database login parameters.

- Start the Crystal Enterprise Admin Launchpad application.
- Login as the Crystal User for the domain that you are editing.
- Click the Schedule Manager link to show a list of the deployed reports.
- Click the 'Change Login Info' link for **each different** login found.
- Edit the Database parameters to match the Oracle Schema for that domain.

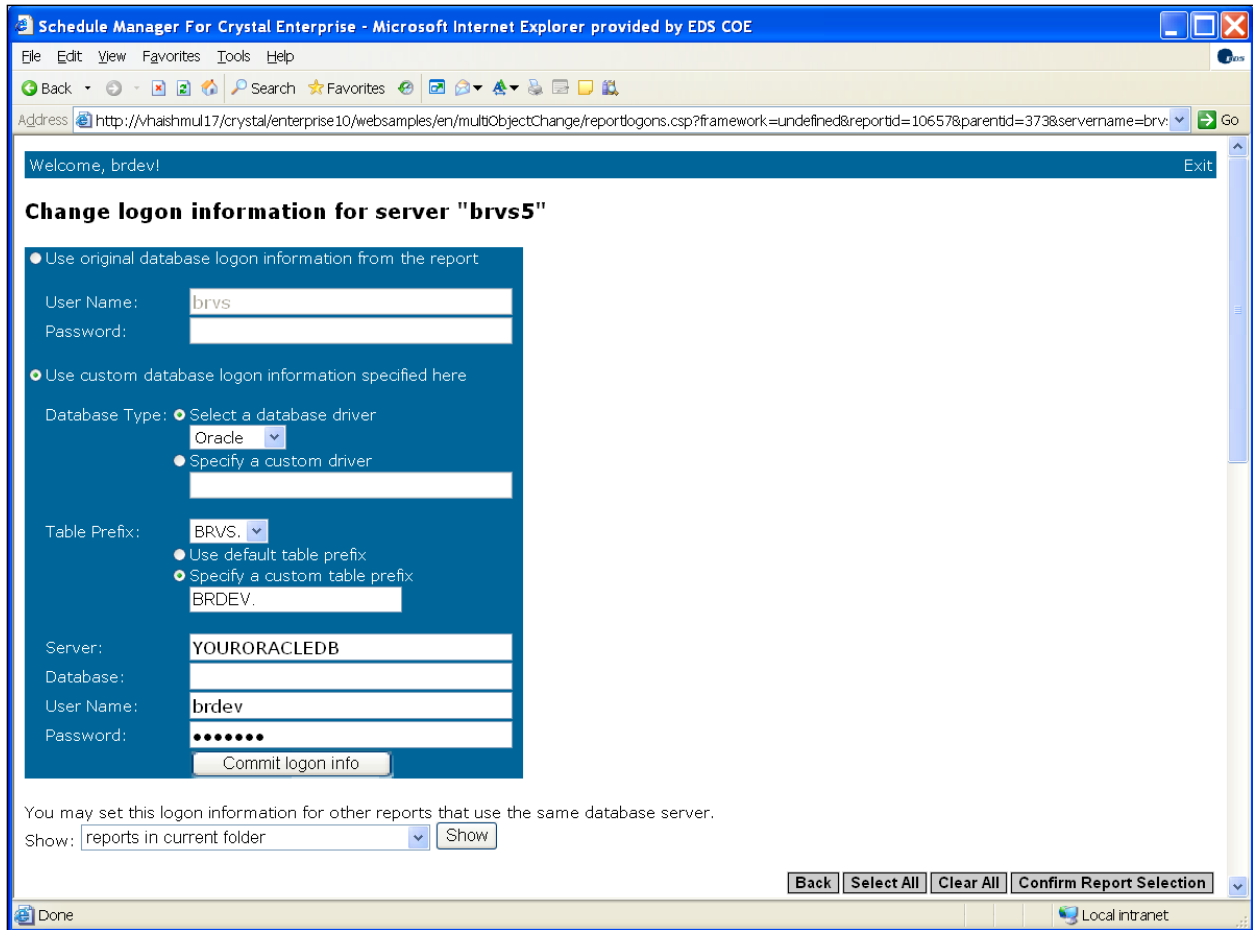
Crystal schedule manager screen showing published reports:



Crystal schedule manager screen for editing database parameters:

Steps in this screen:

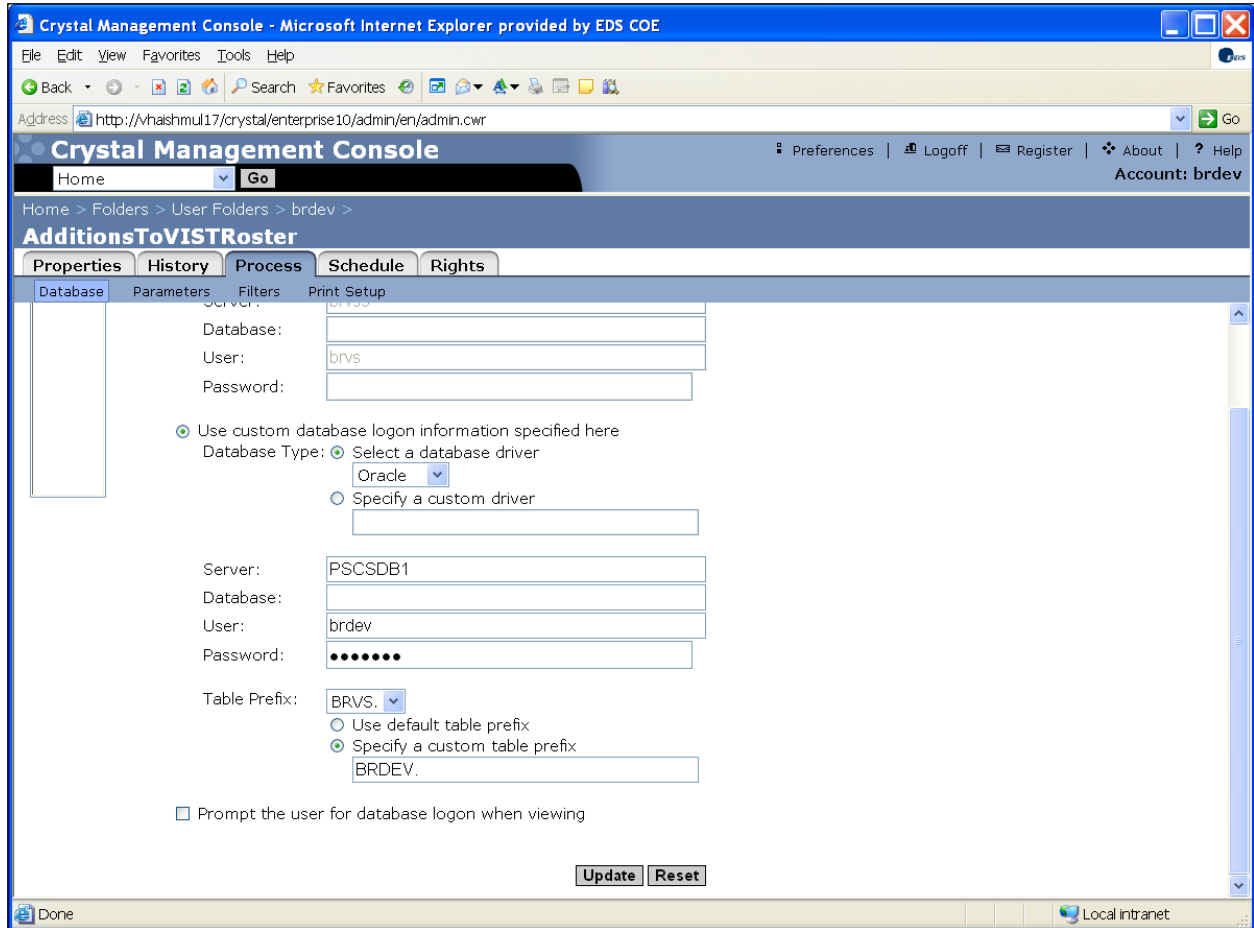
- Verify the 'Use custom database login' radio button is selected.
- Verify the 'Select a database driver' radio button is selected and Oracle is selected in the accompanying list box.
- Verify the 'Specify a custom table prefix' radio button is selected and the schema name is entered in the accompanying text field is **CAPITALIZED** and ends with a period.
- Verify the database user name and password is correct for the schema.
- When all fields are entered, click the 'Select All' button. This sets all reports with this connection type to accept the new connection information.
- Click the 'Commit logon info' button.
- Repeat this procedure for each different connection listed in the previous screen. (see previous screen shot)



4. Start the Crystal Management Console application.
  - Login as the Crystal User for the domain that you are editing. (Authentication type should be 'Enterprise.')
  - Click the 'Objects' link for that user.
  - Click on the first report listed.
  - Click the 'Process' tab, and then click the 'Database' link.
  - Verify that the database fields are properly set for the correct schema.
  - Verify that the '**Prompt the user for database logon when viewing**' checkbox is unchecked. **Leaving this checked will cause significant performance issues!**
  - Click the Update button.

**Repeat this for each deployed report.**

Crystal Management console screen for Login Prompt checkbox:



5. Edit the Blind Rehabilitation application.properties file for each domain to match the Crystal Enterprise login and server information.



## Appendix A – Sample Blind Rehabilitation application.properties file

```
# This file contains application wide properties in key=value fashion
# to get a value for one of the keys, use a java call like this:
# ApplicationProperties.getProperty("key")
# ex. ApplicationProperties.getProperty("Mailing_Address_Type")
#
#####
# System Build/Version
Software_Version=5.0.29 - Domain Name (Server Name) Support Phone Number: 1-
888-596-4357

#####
# Session Timeout Warning - in minutes should be 5 minutes less than app
server timeout
# Used to display an alert to user indicating a pending session timeout
# Default is 25 minute warning (for a 30 minute session timeout)
#SessionTimeoutWarningMinutes=.25
SessionTimeoutWarningMinutes=25

# Currently supported Vista configuration
CurrentKidsVistaVersion=5.0.1.4
#CurrentKidsVistaVersion=BYPASS

#####
# jdbc JNDI datasource name
jdbc_jndi_name=jdbc_brvs

#####
# Address Types
Mailing_Address_Type=M
Business_Address_Type=B

#####
# User Preference Types
Current_Patient_List=P

#####
# Notifications properties
UserNotificationsLink=/userNotifications.do
UserNotificationsKeyParameterName=notificationKey

#####
# Referral Auto-Search Properties
PerformReferralAutoSearchDuringLogin=true
#PerformReferralAutoSearchDuringLogin=false

#This must be a comma separated list with no spaces or formatting characters
# EXAMPLE: ReferralAutoSearchStatusTypes=Offered,Accepted,Scheduled,Pending
#ReferralAutoSearchStatusTypes=Offered,Accepted,Scheduled,Pending
ReferralAutoSearchStatusTypes=Pending

#How many days in the past to use for the "start date"
ReferralAutoSearchDaysBackToInclude=40
```

```

#The next 2 properties are not available until version 5.0.27.1
ReferralNotificationText=You have NEW referrals. Click this link to view.
ReferralNotificationLink=/modifyReferral.do?modifyReferralButtonPressed=Auto
Search

# Blind Rehab Instructors Role
BlindRehabInstructorsRoleCode=114

#####
# Record Limiter - Set the maximum records for a search to return
# Only used in some transactions: referral search, ...
MaxRecordsReturnedLimit=2000

#####
# VistaLink Login Settings
# CAUTION!!! If byPassVista is set, NO login and password is actually used to
# access the system. These settings are only used for development while
# disconnected from a VistaLink server!!!
#Set this to 'Y' to bypass the VistaLink Login
Bypass_Vista_Login=N
#The duz code must be set here if "Bypass_Vista_Login=Y"
User_DUZ_CODE_If_Vista_Bypass_Enabled=4618

#####
# CCOW constants
ccowDisabled=true

ccowApplicationName=SampleApp1
ccowApplicationPasscode=IAKMDpN1ZsOF10cqX4zd4nFD1hlhBUaFG0jIl-
ruw11BGkm9STQi_m-mcoVoljDDvm9p9krAYKx5fO-PF-
XwZUXEpTGst_jFxtXu4MCNw_WNOqIZOGS8s78CUHwQL_9tZf2a-
BWDfrM80aIO71YHzZpXwzVkjDg5pQpOl0AJ6HsK8esomtlMzHYZPmXWxJrYCVrBYsU17nhplnd4iO
yKK3b9AvsEnloCsoAzstcf_VO0yJRPylpSh_2yrUBOLkB9

ccowActionPrefix=ccow.do?ccowAction=
ccowActionSetup=setup
ccowActionParticipate=participate
ccowActionListener=respondToContextChange

ccowLocatorAppletAction=ccowLocatorApplet.do
ccowLocatorTimeout=60

ccowListenerRefreshTarget=main

# Make these blank to turn off debugging
ccowLocatorDebug=true
ccowListenerDebug=true
ccowNotifierDebug=true

ccowIcnLookup=Patient.ID.MRN.CCOW
ccowDfnLookupPrefix=Patient.ID.MRN.DFN_

#####
# Crystal Reports Settings
CrystalEnterpriseServerUser=crystal_user
CrystalEnterpriseServerPassword=crystal_password

```

```

CrystalEnterpriseServerName=crystal_server_name
CrystalEnterpriseAuthType=secEnterprise

#####
# PSD Update Settings
#
# Note: If OvernightDemographicUpdatesEnabled is set to false
# OvernightADTUpdatesEnabled is ignored, and ADT updates will not run.
#
PSDUpdateMaxBatchICNs=500
ProxyUserName=ANRVAPPLICATION,PROXY USER
OvernightDemographicUpdatesEnabled=false
OvernightADTUpdatesEnabled=false
PSDUpdaterNumberOfThreads=8
PSDUpdaterWorkQueueSize=128
# The HOUR below must be from 0-23 (0 being midnight), and the MINUTE
# below must be 0-59.
PSC_UPDATER_START_TIME_HOUR=0
PSC_UPDATER_START_TIME_MINUTE=0

#####
# Email Notification Settings
#
IcnChangeNotifySmtpServer=vhaxxxxxx
IcnChangeNotifyToEmailAddress=VHABlindRehabHL7Data@va.gov
IcnChangeNotifyFromEmailAddress=VHABlindRehabHL7Data@va.gov

#####
# MPI Settings
#
MPI_WEBLOGIC_FACTORY=weblogic.jndi.WLInitialContextFactory
MPI_WEBLOGIC_URL=t3://localhost:xxxx
MPI_EJB=IcnUpdateSession
MPI_WEBLOGIC_USERNAME=weblogic_user
MPI_WEBLOGIC_PASSWORD=weblogic_password

#####
# Back Button/Double Save prevention message
#
BACK_BOTTON_PREVENTION_MESSAGE=You may have attempted to use the back button
to update a record to a previous state or clicked the save button twice.
These actions are not allowed. Please lookup your record again and update it
without using the back button or clicking save twice.

```

## Appendix B – Sample Blind Rehabilitation log4j.properties file

**NOTE:** lines of significance are in **bold**. See log4j documentation on <http://logging.apache.org/log4j/docs/index.html> for more information.

### **log4j.rootCategory=info, rolling**

```
# Example of adding a specific package/class at a different
# logging level...
# --log everything in the com.johnmunsch package at debug level
# ..even better, send it to a different appender. Note, however, that
# this doesn't mean that any loggers from a lower level won't be used:
# everything still inherits, so this new logger is used in addition to
# the loggers it would have otherwise already used.
#log4j.category.com.johnmunsch=debug, socketLogger

# --on the other hand, everything in the
# com.johnmunsch.stuff class shouldn't log
# unless the log message is at 'warn' level or worse.
# (It just so happens that stuff generates a lot of
# logging when it's used)
#log4j.category.com.johnmunsch.stuff=warn

# --also, it just so happens that we have a different
# appender that we're using that we want to have
# log information from a specific location, and we
# don't want to send that information anywhere else.
#log4j.category.com.johnmunsch.otherstuff=warn, xml
#log4j.additivity.com.johnmunsch.otherstuff=false

# BEGIN APPENDER: CONSOLE APPENDER (stdout)
# first: type of appender (fully qualified class name)
log4j.appender.stdout=org.apache.log4j.ConsoleAppender

# second: Any configuration information needed for that appender.
# Many appenders require a layout.
# log4j.appender.stdout.layout=org.apache.log4j.TTCCLayout
log4j.appender.stdout.layout=org.apache.log4j.SimpleLayout

# Possible information overload?
# log4j.appender.stdout.layout=org.apache.log4j.PatternLayout
# additionally, some layouts can take additional information --
# like the ConversionPattern for the PatternLayout.
# log4j.appender.stdout.layout.ConversionPattern=%d %-5p %-17c{2} (%30F:%L) %3x
# - %m%n
# END APPENDER: CONSOLE APPENDER (stdout)

# BEGIN APPENDER: ROLLING FILE APPENDER (rolling)
# first: type of appender (fully qualified class name)
```

```

log4j.appender.rolling=org.apache.log4j.RollingFileAppender

# second: Any configuration information needed for that appender.
# Many appenders require a layout.
log4j.appender.rolling.File=BlindRehab_LOG4J.log

log4j.appender.rolling.MaxFileSize=1000KB
# Keep 10 backup files
log4j.appender.rolling.MaxBackupIndex=10

log4j.appender.rolling.layout=org.apache.log4j.PatternLayout
log4j.appender.rolling.layout.ConversionPattern=%p %d{yyyy/MM/dd HH:mm:ss,SSS} %t %c -
%m%n

# Possible information overload?
# log4j.appender.stdout.layout=org.apache.log4j.PatternLayout
# additionally, some layouts can take additional information --
# like the ConversionPattern for the PatternLayout.
# log4j.appender.stdout.layout.ConversionPattern=%d %-5p %-17c{2} (%30F:%L) %3x
- %m%n
# END APPENDER: ROLLING FILE APPENDER (rolling)

# BEGIN APPENDER: SOCKET APPENDER (socketLogger)
# Note: if you don't have anything configured to accept the events
# from the socketLogger appender, you'll see an exception on program
# startup (to console), and occasional status messages (to console)
# on if the log4j system has managed to connect to the specified
# socket..
log4j.appender.socketLogger=org.apache.log4j.net.SocketAppender
log4j.appender.socketLogger.RemoteHost=CHAINSAW_SERVER_NAME
log4j.appender.socketLogger.Port=14500
log4j.appender.socketLogger.LocationInfo=false
# END APPENDER: SOCKET APPENDER (socketLogger)

# BEGIN APPENDER: LogFactor5 APPENDER (lf5)
# LogFactor5 is a Swing window that directly receives logging messages and
# displays them. It offers filtering, searching etc. similar to Chainsaw or
# Lumbermill but you don't have to use a socket appender so it should be faster
# when the logging display is on the same machine as the program issuing
# messages.
log4j.appender.lf5=org.apache.log4j.lf5.LF5Appender
log4j.appender.lf5.MaxNumberOfRecords=1000
# END APPENDER: LogFactor5 APPENDER (lf5)

# BEGIN APPENDER: XML APPENDER (xml)
# A standard file appender where we have put an XML layout onto the output
# event records. A file put out using this technique can be loaded after
# the fact into Chainsaw for viewing, filtering, searching, etc.
log4j.appender.xml=org.apache.log4j.FileAppender
log4j.appender.xml.file=example_xml.log
log4j.appender.xml.append=false

```

```
log4j.appender.xml.layout=org.apache.log4j.xml.XMLLayout
# END APPENDER: XML APPENDER (xml)

# BEGIN APPENDER: LogFactor5 Rolling APPENDER (lf5Rolling)
# Like the XML appender above, this is a specialized format designed to be read
# from a tool. In this case, LogFactor5 can load up files in this format for
# after the fact review.
log4j.appender.lf5Rolling=org.apache.log4j.RollingFileAppender
log4j.appender.lf5Rolling.File=example_lf5.log
log4j.appender.lf5Rolling.layout=org.apache.log4j.PatternLayout

log4j.appender.lf5Rolling.layout.ConversionPattern=[slf5s.start]%d{DATE}[slf5s.DATE]%n
%p[slf5s.PRIORITY]%n%x[slf5s.NDC]%n%t[slf5s.THREAD]%n%c[slf5s.CATEGORY]%n
%l[slf5s.LOCATION]%n%m[slf5s.MESSAGE]%n%n
# END APPENDER: LogFactor5 Rolling APPENDER (lf5Rolling)

# Start up PSL logging:
log4j.logger.gov.va.med.person.lookup=INFO
```

## Appendix C – Sample Blind Rehabilitation MPIListener.properties file

```
# The local port we will run our MPI Listener on
MPI_LISTENER_PORT=xxxx

# The remote IP and port of the MPI system. This is the MPI system
# that will send messages to our MPI Listener.
# MPI_IP = Remote IP Address
# MPI_PORT = Remote Port Number
#MPI_IP=00.00.00.00
#MPI_PORT=xxxx

# MPI Test Environment (Leave this commented out unless you are
# testing. It is only here so we can keep track of the test IP and
# port numbers.)
MPI_IP=xx.x.xxx.xx
MPI_PORT=xxxx

# Listener Mode should be one of the following:
# D = Debugging
# P = Production
# T = Training
LISTENER_PROCESSING_MODE=T
LISTENER_MESSAGE_VERSION=2.4

# The name of the logger to use
LOGGER_NAME=MPILogger

# Sending Application and Facility values
SENDING_APP_NAMESPACE_ID=BLIND REHAB
SENDING_FACILITY_NAMESPACE_ID=XXXBR
# Production Namespace ID:
#SENDING_FACILITY_NAMESPACE_ID=XXXBR
SENDING_FACILITY_UNIVERSAL_ID=BLINDREHAB.MED.VA.GOV
SENDING_FACILITY_UNIVERSAL_ID_TYPE=DNS

# Properties for outgoing QBP messages
FIELD_SEPARATOR=^
COMPONENT_SEPARATOR=~
REPETITION_SEPARATOR=|
SUB_COMPONENT_SEPARATOR=&
SENDING_APPLICATION=MPI
SENDING_FACILITY_NAMESPACE_ID=XXXBR
# Production Namespace ID:
#SENDING_FACILITY_NAMESPACE_ID=XXXBR
SENDING_FACILITY_UNIV_ID=XXXXXXXXX.VHA.MED.VA.GOV
SENDING_FACILITY_UNIV_ID_TYPE=DNS
RECEIVING_APPLICATION=MPI
```

RECEIVING\_FACILITY\_NAMESPACE\_ID=XXXM  
RECEIVING\_FACILITY\_UNIV\_ID=XXXX.XX-XXXXXXXX.MED.VA.GOV  
RECEIVING\_FACILITY\_UNIV\_ID\_TYPE=DNS



## Appendix D – Sample Blind Rehabilitation Patient Service (PSC/PSD) PatSvcPkg.properties file

**NOTE:** sections of significance are in **bold**.

```
#
# Patient Service package properties file.
# -----
#
# Values for JNDI Initial Context

# weblogic jndi factory - local host
initialContextFactory = weblogic.jndi.WLInitialContextFactory
providerURL = t3://xxx.x.x.x:7xxx/PatientServiceR2
securityPrincipal = weblogic
securityCredentials = weblogic
dedicatedrmicontext =

# The EJBs in use
QueryPatient = gov.va.med.patientadmin.ejb.QueryPatientHome

# Use NDS or not
# false means NDS will be used to perform JNDI lookup
# true means property values will be used to perform JNDI lookup
deCaipitated = true
```

# Appendix E – Sample Blind Rehabilitation Person Service Lookup (PSL) PatientLookup.properties file (contained in the psIConfig\_4.0.4.4.jar file)

NOTE: sections of significance are in **bold**.

```
*****
# PatientLookup properties file.
*****

#dataAccessType = cache OR vistalink
# NOTE: THIS NEEDS TO BE REMOVED FROM OUR CODE
# RAJ HAS ADDED THIS FOR CACHE....
#=====
dataAccessType = vistalink

#WEBLOGIC Parameters
#=====
initialContextFactory = weblogic.jndi.WLInitialContextFactory
providerURL = t3://xxxxxxxx.vha.med.va.gov:xxxx
securityPrincipal = weblogic
securityCredentials = weblogic
dedicatedrmicontext = true

#Veteran Image Server Parameters
#=====
enableVeteranImage = true
veteranImageURL = https://xxxx.xxxx.xxx.va.gov/vic/NCMD_Result_Picture_Streaming2.aspx
veteranImageServer = xxxx.etech.med.va.gov

#Data source JNDI name for IDENTITY MANAGEMENT (IM) database
#=====
imDatasourceName = IMDS

#CAIP Configuration Parameters
#=====
caip_nds_major_version=4
caip_nds_minor_version=0
caip_nds_revision_number=0
caip_nds_build_number=0
ndsurl=t3://vhaispora04:8511
ndsuser=weblogic
ndspassword=getITwrite

#PSL version
PSL_VERSION=4.0.4.4

plu_client_help_url=http://xxxxxxxx.vha.med.va.gov:7xxx/PSLClientHelp/help/pluhelp.html
caip_enabled=false
```

# Appendix F – Installation Steps to Upgrade Blind Rehabilitation v5.0.26.8 to v5.0.27.6

## General steps:

1. Unzip the application package file **br\_deployment\_5.0.27.6.zip** into a working directory.
2. Review the following documents (found in the doc subdirectory created while unzipping the application package file) and have available during upgrade for detailed information:  
**/doc/Installation/BR\_Installation\_Implementation\_Guide\_C.doc**  
**/doc/ReleaseNotes/v5.0.27.3.txt**  
**/doc/ReleaseNotes/v5.0.27.4.txt**  
**/doc/ReleaseNotes/v5.0.27.5.txt**  
**/doc/ReleaseNotes/v5.0.27.6.txt**
3. Ensure that the Standard Data Services (SDS) database is running v10.0.

## Application Server Steps:

1. Backup the deployment directories on the managed servers.
2. Verify that the VistaLink JAR files are on the CLASSPATH per install guide.
3. These have been removed from the Blind Rehab EAR file as directed by VistALink Team.
4. Undeploy the Blind Rehab application from WebLogic.
5. Undeploy the existing PSL EAR.
6. Stop the Weblogic servers running the Blind Rehab application.
7. Restart all WebLogic servers that will run the application.
8. Copy the Blind Rehab configuration files from the **/conf** directory expanded from the zip file into the servers config directory. See the install guide for detailed instructions on the configuration files.
9. Configure the **application.properties** files for all servers being deployed to.
10. There are new entries in this file from previous releases.
11. The PSD Updater has thread settings and start time schedule.
12. The Session Timeout warning has a SessionTimeoutWarningMinutes setting.
13. Remove old PSL Config files from CLASSPATH if present.
14. Add **pslConfig\_4.0.4.4.jar** to CLASSPATH of managed server(s).

15. Edit **pslConfig\_4.0.4.4.jar** and adjust settings in the contained **PatientLookup.properties** file to match your environment.
16. Deploy the **pslEJB\_4.0.4.4.ear** EAR File.
17. Deploy the new BR Application Ear file: **BR\_EAR\_5.0.27.6.ear**.
18. Reboot the Application servers.

### **Database Steps:**

1. Backup the BR Oracle data.  
Upgrade the SDS Database Schema to v10.0 See BR Central Install Guide and SDS Install Package. Both are included in the **br\_deployment\_5.0.27.6.zip** file.
2. Run the **/sql/BR\_Build27Additions.SQL** script against the Oracle schema. Check for errors.  
Example: sqlplus BR\_ORACLE\_USER/BR\_ORACLE\_PWD@database  
@sql/BR\_Build27Additions.SQL
3. Run the **/sql/BR\_VIEWS.SQL** script against the Oracle schema. Check for errors.  
Example: sqlplus BR\_ORACLE\_USER/BR\_ORACLE\_PWD@database  
@sql/dataLoad/BR\_VIEWS.SQL SDS\_ORACLE\_USER

### **Crystal Enterprise Steps:**

1. Create a backup of the existing Crystal User environment with the published reports.
2. Delete all the existing Crystal reports for the domain you are upgrading and publish all the Crystal Reports from **/reports** directory created while unzipping the **br\_deployment\_5.0.27.6.zip** file.
3. Set the database parameters for all crystal reports in Crystal Management Console (See install guide).
4. Reboot the Crystal enterprise servers.

# Appendix G – Installation Steps to Upgrade Blind Rehabilitation v5.0.27.6 to v5.0.28.4

## General steps:

1. Unzip the application package file **br\_deployment\_5.0.28.4.zip** into a working directory.
2. Review the following documents (found in the doc subdirectory created while unzipping the application package file) and have available during upgrade for detailed information:  
**/doc/Installation/BR\_Installation\_Implementation\_Guide\_C.doc**  
**/doc/ReleaseNotes/v5.0.27.3.txt**  
**/doc/ReleaseNotes/v5.0.27.4.txt**  
**/doc/ReleaseNotes/v5.0.27.5.txt**  
**/doc/ReleaseNotes/v5.0.27.6.txt**  
**/doc/ReleaseNotes/v5.0.28.4.txt**
3. Ensure that the Standard Data Services (SDS) database is running v10.0.

## Application Server Steps:

1. Backup the deployment directories on the managed servers.
2. Undeploy the Blind Rehab application from Web Logic.
3. Deploy the new BR Application Ear file: **BR\_EAR\_5.0.28.4.ear**.

## Database Steps:

1. Backup the BR Oracle data.  
Upgrade the SDS Database Schema to v10.0 See BR Central Install Guide and SDS Install Package. Both are included in the **br\_deployment\_5.0.28.4.zip** file.
2. Run BR-DB-PATCH-5.0.28.4 (maintenance\_patch)

# Appendix H – Installation Steps to Upgrade Blind Rehabilitation v5.0.28.4 to v5.0.29.4

## General steps:

1. Unzip the application package file **BR-PKG-5.0.29.6.zip** into a working directory.
2. Review the following documents and have available during upgrade for detailed information:  
**BR\_Installation\_Implementation\_Guide\_C.doc**  
**ReleaseNotes/v5.0.29.txt**
3. Ensure that the Standard Data Services (SDS) database is running v18.0

## Application Server Steps:

4. Backup the deployment directories on the managed servers.
5. Undeploy the Blind Rehab application from Web Logic.
6. Deploy the new BR Application Ear file: **BR\_APP\_5.0.29.4.ear**.

## Database Steps:

1. Backup the BR Oracle data.  
Upgrade the SDS Database Schema to v18.0
2. Run BR-DB-PATCH-5.0.27.6 (conditional\_patch) to restore BR FK references and grants after SDSADM schema has been dropped and restored to an upgraded version.  
Run after the SDSADM schema was re-created and re-populated with SDS data
3. Need to run 2 Conditional DB patches and 1 Maintenance DB patch for this release
  - i) BR-DB-PATCH-5.0.29.1 (conditional\_patch)
  - ii) BR-DB-PATCH-5.0.29.2 (conditional\_patch)
  - iii) BR-DB-PATCH-5.0.29.3 (maintenance\_patch)

## Glossary/Acronym List

<i>Term/Acronym</i>	<i>Description</i>
AAA	(Veteran Health Administration) Authentication, Authorization and Accountability Standards
AAIP	Authentication and Authorization Infrastructure Program
ADPAC	Automated Data Processing Application Coordinator
AMIS	Automated Management Information System
API	Application Program Interface
Audit Trail	A history of the changes made to a record including old data, new data, and the name of the user who made the change. Record of access and modifications
BCMA	Bar Code Medication Administration. A VistA software application that validates medications against active orders before the medication is given to the patient.
BR	Blind Rehabilitation Project
Blind Rehabilitation Center (BRC)	A residential inpatient program that provides comprehensive adjustment to blindness training and serves as a resource to a catchments area usually comprised of multiple Veterans Integrated Service Networks (VISN).
BRC Application Letter	This is a cover letter for a Blind Rehabilitation Center (BRC) Application packet. This letter requires editing and is used to print for individual veterans.
BRC Follow-up Letter	This is a questionnaire sent to the veteran following blind rehabilitation training. It is used to assist the center or clinic in following-up on the veteran.
Blind Rehabilitation Outpatient Specialist (BROS)	Blind Rehabilitation instructors possessing advanced technical knowledge and competencies in at least two Blind Rehabilitation disciplines at the journeyman level.[2]
CAT	Computer Access Training
CARF	Commission on the Accreditation of Rehabilitative Facilities
CCOW	Clinical Context Object Work Group
CCOW Term Telnet	An application (written in Delphi) which is RPCBroker aware and capable of CCOW with CCOW, which can be used to access the Roll and Scroll environment, such as List Manager, in VistA.
CCOW Timing Program	A program, written in Delphi that tests the amount of time for Remote Procedure Calls to be processed by the server.
CHISS	Common Health Information Security Services
C&P	Compensation & Pension

<b><i>Term/Acronym</i></b>	<b><i>Description</i></b>
Claim Letter	This is a cover letter to a Veterans Administration Regional Office (VARO) when filing a claim on behalf of a VIST veteran. This letter is used to print for individual veterans.
Common Procedure Terminology (CPT)	A method for coding procedures performed on a patient, for billing purposes.
CPRS	A VistA software application that provides an integrated patient record system for use by clinicians, managers, quality assurance staff, and researchers
CPRS/CCR	Computerized Patient Record System/Computerized Clinical Reminder Module
CPRS/VSM	Computerized Patient Record System/Vital Signs Module
Computerized Patient Record System (CPRS)	A clinical record system that integrates many VistA packages to provide a common entry and data retrieval point for clinicians and other hospital personnel. (CPRS). CPRS is a Veterans Health Information Systems and Technology Architecture (VistA) software application that enables clinicians, nurses, clerks, and others to enter, review, and continuously update all information connected with patients.
Context Vault	Data store that houses user sign-on credentials in a CCOW user context.
DaIS	Development and Infrastructure Support
DBIA	Data Base Integration Agreement
DELPHI	A Rapid Application Development (RAD) system/application developed by Borland International, Inc. Delphi is similar to Visual Basic from Microsoft, but whereas Visual Basic is based on the BASIC programming language, Delphi is based on Pascal.
Division	The subunit under institute has 5-6 digits/letter division ID and less than a 35 character name
EJB	Enterprise Java Bean
Encounter	A contact between a patient and a provider who has the primary responsibility of assessing and treating the patient. A patient may have multiple encounters per visit. Outpatient encounters include scheduled appointments and walk-in unscheduled visits. A clinician's telephone communications with a patient may be represented by a separate visit entry. If the patient is seen in an outpatient clinic while an inpatient, this is treated as a separate encounter.
Episode of Care	An interval of care by a health care facility or provider for a specific medical problem or condition. It may be continuous or it may consist of a series of intervals marked by one or more brief separations from care, and can also identify the sequence of care (e.g., emergency, inpatient, outpatient), thus serving as one measure of health care provided.
FSOD	Functional Status Outcomes Database



<b><i>Term/Acronym</i></b>	<b><i>Description</i></b>
Graphical User Interface (GUI)	A type of display format that enables users to choose commands, initiate programs, and other options by selecting pictorial representations (icons) via a mouse or a keyboard.
HCFA	Health Care Financing Administration
HCPCS	HCFA Common Procedure Coding System
HFS	Host File Server is a system (WinNT/Dec Alpha) file access mechanism that enables the M software (server software) to access the system-level files.
Health <sub>e</sub> Vet-VistA	The Health <sub>e</sub> Vet-VistA architecture will be a services-based architecture. Applications will be constructed in tiers with distinct user interface, middle and data tiers. Two types of services will exist, core services (infrastructure and data) and application services (a single logical authoritative source of data).
HIPAA	Health Insurance Portability and Accountability Act of 1996. Also referred to as, HIPAA.
HL7	Health Level Seven
HSD&D	Health Systems Design & Development
HSM	Hospital-Supplied Self Medication
HTTP	Hyper Text Transfer Protocol
HTTPS	Secured HTTP Protocol
ICD9	International Classification of Diseases 9 <sup>th</sup> Edition
ICN	Identification Control Number
IDL	Iterative Development Lifecycle
IE	Internet Explorer
IEN	Internal Entry Number
Independent Verification and Validation (IV&V)	The IV&V team supports the HSD&D mission by promoting standardization, improving software release quality and effectiveness of healthcare delivery through planned and controlled evaluation, testing, and integration of healthcare information systems. . Visit the <a href="http://vista.med.va.gov/ivv/">http://vista.med.va.gov/ivv/</a> site for additional information.
Inpatient Visit	The admission of a patient to a VAMC and any clinically significant change related to treatment of that patient. For example, a treating specialty change is clinically significant, whereas a bed switch is not. The clinically significant visits created throughout the inpatient stay would be related to the inpatient admission visit. If the patient is seen in an outpatient clinic while an inpatient, this is treated as a separate encounter.
Institution	A major hospital with subdivisions, usually has a name < 30 letters and a three-digit division ID

<b><i>Term/Acronym</i></b>	<b><i>Description</i></b>
Invitation for VIST Review	This is an invitation to blinded veterans from VIST, offering a health evaluation. Veterans may accept or deny the invitation. This letter satisfies the requirements of M-2, Part XXIII and is meant to be printed as a mass mailing.
IP	Meds Inpatient Medications
IRM	Information Resources Management
IRS Exemption Letter	This letter advises the Internal Revenue Service of legally blind status of veterans. This letter requires editing and is to be printed for individual veterans.
ISO	Information Security Officer
ISSRA	Interim Security Services for Rehosted Applications
Iterative Development	The technique used to deliver the functionality of a system in a successive series of releases of increasing completeness. Each iteration is focused on defining, analyzing, designing, building, and testing a set of requirements.
IV	Intravenous
J2EE	The Java 2 Platform, Enterprise Edition (J2EE) is an environment for developing and deploying enterprise applications. The J2EE platform consists of a set of services, APIs, and protocols that provide the functionality for developing multi-tiered, Web-based applications.
JAAS	Java Authentication and Authorization Service. For more information refer to the JAAS Web site at the following address: <a href="http://java.sun.com/products/jaas/index-14.html">http://java.sun.com/products/jaas/index-14.html</a>
JAVA	Java is a programming language. It can be used to complete applications that may run on a single computer or be distributed among servers and clients in a network.
JCAHO	Joint Commission on the Accreditation of Health Care Organizations
JDBC	Java Database Connection
JSP	Java Server Page
Kernel	Set of VistA software routines that function as an intermediary between the host operating system/application and the VistA application packages such as Laboratory, Pharmacy, IFCAP, etc. The Kernel provides a standard and consistent user and programmer interface between application packages and the underlying M implementation.
Kiosk	Public workstations shared by multiple users.
List Manager	A VistA software product that creates a framework for user actions. List Manager is part of the VistA software infrastructure.
LOINC	Logical Observation Identifier Names and Codes
MAH	Medication Administration History

<b><i>Term/Acronym</i></b>	<b><i>Description</i></b>
MAS	Medical Administration Service
MH Assistant	Mental Health Assistant
MPI	Master Patient Index
MST	Military Sexual Trauma
MTAS	Middle Tier WebLogic Application Server
MVC	Model View Controller
New Person (#200) File	A VistA file that contains data on employees, users, practitioners, etc. of the VA.
NOIS	National Online Information System
NVS	National VistA Support
O-R	Object-Relational
OCS	VA Office of Cyber Security
OID	Oracle Internet Directory
ORACLE	Oracle is a relational database that supports the Structured Query Language (SQL), now an industry standard.
ORACLE 9iAS	Oracle 9i Application Server
PATS	Patient Advocate Tracking System/application. When completed, the Patient Advocate Tracking System/application will replace the current, site-based Patient Representative package with a national level application.
PCE	Patient Care Encounter
PIMS	Patient Information Management System
PIR	Patient Incident Review File
PLU	Patient Lookup
PSC	Patient Service Construct
PSD	Patient Service Demographics
PSL	Person Service Lookup
PRN	Pro Re Nata, Latin meaning 'as needed'
Prototype	An initial working model as proof of concept of a product or new version of an existing product.
Provider	The entity that furnishes health care to consumers. An individual or defined group of individuals who provide a defined unit of health care services (defined = codable) to one or more individuals at a single session.
PTF	Patient Treatment File (PTF) at AAC

<b><i>Term/Acronym</i></b>	<b><i>Description</i></b>
RDBMS	Relational Database Management System
Registration	Registration File
RN	Registered Nurse
ROES	Remote Order Entry System
SAS	SAS is a company that provides data analysis, data mining, and data storage
ScreenMan	VA FileMan utility that provides a screen-oriented interface for editing and displaying data
SDD	Software Design Document
SQA	Software Quality Assurance
SDS	Standard Data Service
SRS	Software Requirements Specifications
SSL	Secure Socket Layer
SSO	Single Sign On
TCP/IP	Transmission Control Protocol/Internet Protocol
Thin-client	A simple client program, which relies on most of the function of the system being in the server, usually the Web browser in a Web domain
TIU	Text Integration Utility
User	An Administrator, a Clinician, or a Researcher
VA	Veterans Affairs
VA FileMan	VistA database management system.
VAMC	Veterans Affairs Medical Centers
VARO	Veterans Administration Regional Office
VHA	Veterans Health Administration
VISN	Veterans Integrated Service Network
VIST	Visual Impairment Service Team
VistA	Veterans Health Information Systems and Technology Architecture
VistA MailMan	VistA electronic mail system