

Local Data Update (DATUP) Installation Guide

Pharmacy Reengineering



Version 2.0

July 2014

**Department of Veterans Affairs
Office of Information and Technology (OIT)
Product Development (PD)**

REVISION HISTORY

Each time this manual is updated, the Title Page lists the new revised date and this page describes the changes. No Change Pages document is created for this manual. Replace any previous copy with this updated version.

| Date | Revised Pages | Patch Number | Description |
|------------------|---------------|---------------------|---|
| July 18, 2014 | All | N/A (First Release) | Changed date (month) to reflect real release date Marella Colyvas |
| June 3, 2014 | All | N/A (First Release) | Fixed pagination Marella Colyvas |
| May 30, 2014 | All | N/A (First Release) | Added footnote describing relationship between FDB MedKnowledge Framework and FDB-DIF, updated text appropriately. Updated TOC. Brian Holihan |
| May 28, 2014 | All | N/A (First Release) | Updated images for 508 compliance; changed FDB MedKnowledge Framework back to FDB-DIF. Updated TOC Brian Holihan |
| May 27, 2014 | All | N/A (First Release) | Updated Revision History format; did a partial search and replace on FDB-DIF (and other similar iterations) to FDB MedKnowledge Framework, though not in the instructions as the tool may not be updated yet. Removed extraneous definitions of FDB. Brian Holihan |
| May 21, 2014 | i | N/A (First Release) | Updated Revision History Brian Holihan |
| May 21, 2014 | i, 3, B-1 | N/A (First Release) | Updated Title Page, Corrected Version number in Overview section, minor text edit, added B-1 to Table of Figures. Brian Holihan |
| May 19, 2014 | All | N/A (First Release) | DBA Corrected Information, minor formatting updates Alicia Ray, Brian Holihan |
| May 13, 2014 | All | N/A (First Release) | Edits to headings, TOC, references, and footers Marella Colyvas |
| February 7, 2013 | All | N/A (First Release) | Technical Writer Edits Marella Colyvas |
| February 5, 2013 | All | N/A (First Release) | Updates to various section for DATUP 2.0 Sachin Sharma |

| Date | Revised Pages | Patch Number | Description |
|-------------------|----------------------|---------------------|--|
| February 1, 2013 | All | N/A (First Release) | Incorporated database configuration updates for DATUP 2.0, FDB installation instructions moved to FDB install guide. Alicia Ray |
| August 27, 2012 | All | N/A (First Release) | Changed formatting and performed edits Marella Colyvas |
| August 14, 2012 | All | N/A (First Release) | Updates to various section to address minor configuration changes Sachin Sharma |
| May 29, 2012 | All | N/A (First Release) | Updated the document to address change request #CR5172 (Image Processing for PPS) Sachin Sharma |
| December 3, 2010 | All | N/A (First Release) | No changes since last delivery. Updated the version number to reflect the latest release of DATUP. SwRI |
| November 12, 2010 | All | N/A (First Release) | Updated the document to address change request #CR2942. SwRI |
| October 8, 2010 | All | N/A (First Release) | Renamed all instances of “PEDTUP” to “DATUP.” SwRI |
| September 3, 2010 | All | N/A (First Release) | Local PEDTUP Installation Guide: Initial version. SwRI |

TABLE OF CONTENTS

| | | |
|--|---|-----------|
| 1 | PROJECT SCOPE | 1 |
| 1.1 | Project Description | 1 |
| 1.2 | PRE Project Goals and Objectives | 1 |
| 1.3 | DATUP Background | 1 |
| 1.4 | Related Documents..... | 2 |
| 2 | DOCUMENT OVERVIEW | 3 |
| 2.1 | Document Background | 3 |
| 2.2 | Overview | 3 |
| 3 | INSTALLATION INSTRUCTIONS | 5 |
| 3.1 | Terminology | 6 |
| 3.2 | Assumptions | 6 |
| 3.3 | Database Installation and Configuration | 7 |
| 3.4 | WebLogic Installation Instructions..... | 7 |
| 3.4.1 | Class Path | 7 |
| 3.4.2 | WebLogic Server Startup Configuration | 8 |
| 3.4.3 | Local JDBC Data Source Configuration | 12 |
| 3.4.4 | Log4j..... | 20 |
| 3.4.5 | Local JMS Configuration | 22 |
| 3.4.6 | Site Configuration Properties | 43 |
| 3.4.7 | DATUP Configuration Properties | 44 |
| 3.4.8 | DATUP Cleanup Script | 44 |
| 3.4.9 | Deployment | 44 |
| 4 | UPGRADE INSTALLATION INSTRUCTIONS..... | 57 |
| 4.1 | Uninstall Previous Release | 57 |
| 4.2 | Deploy New Release | 61 |
| 5 | SYSTEM VERIFICATION..... | 63 |
| 5.1 | Verification..... | 63 |
| APPENDIX A | A-1 | |
| Local DATUP Configuration..... | A-1 | |
| APPENDIX B | B-1 | |
| Combined DATUP / PECS Architecture | B-1 | |

LIST OF FIGURES

| | |
|--|----|
| Figure 3-1. WebLogic Console..... | 5 |
| Figure 3-2. Domain Structure | 8 |
| Figure 3-3. Change Center..... | 9 |
| Figure 3-4. Summary of Servers | 9 |
| Figure 3-5. Settings for Deployment Server | 10 |
| Figure 3-6. Server Start Tab..... | 11 |
| Figure 3-7. Activate Changes | 12 |
| Figure 3-8. Domain Structure | 13 |
| Figure 3-9. Change Center..... | 13 |
| Figure 3-10. Summary of JDBC Data Sources | 14 |
| Figure 3-11. JDBC Data Source Properties | 15 |
| Figure 3-12. Transaction Options | 16 |
| Figure 3-13. Connection Properties | 17 |
| Figure 3-14. Test Database Connection | 18 |
| Figure 3-15. Select Targets | 19 |
| Figure 3-16. Summary of JDBC Data Sources | 19 |
| Figure 3-17. Statement Cache Size Parameter..... | 20 |
| Figure 3-18. Activate Changes..... | 20 |
| Figure 3-19. Lock & Edit..... | 23 |
| Figure 3-20. JMS Modules | 23 |
| Figure 3-21. JMS Modules | 24 |
| Figure 3-22. JMS System Module Properties | 24 |
| Figure 3-23. JMS System Module Targets | 25 |
| Figure 3-24. Add Resources to JMS System Module..... | 25 |
| Figure 3-25. Activate Changes..... | 26 |
| Figure 3-26. Lock & Edit..... | 26 |
| Figure 3-27. JMS Modules | 27 |
| Figure 3-28. JMS Modules | 27 |
| Figure 3-29. Summary of Resources..... | 28 |
| Figure 3-30. Choose Type of Resource to Create | 29 |
| Figure 3-31. Foreign Server Properties..... | 29 |
| Figure 3-32. Foreign JMS Server Target | 30 |
| Figure 3-33. Summary of Resources..... | 31 |
| Figure 3-34. Foreign JMS Server General Configuration..... | 32 |
| Figure 3-35. Activate Changes..... | 33 |
| Figure 3-36. Lock & Edit..... | 33 |
| Figure 3-37. JMS Modules | 34 |
| Figure 3-38. JMS Modules | 34 |
| Figure 3-39. Summary of Resources..... | 35 |
| Figure 3-40. Foreign JMS Server General Configuration..... | 36 |
| Figure 3-41. Destinations..... | 37 |
| Figure 3-42. Foreign Connection Factory Properties..... | 37 |
| Figure 3-43. Activate Changes..... | 38 |
| Figure 3-44. Lock & Edit..... | 38 |
| Figure 3-45. JMS Modules | 39 |
| Figure 3-46. JMS Modules | 39 |
| Figure 3-47. Summary of Resources..... | 40 |
| Figure 3-48. Foreign JMS Server General Configuration..... | 41 |
| Figure 3-49. Foreign JMS Server Connection Factories..... | 42 |

| | |
|---|----|
| Figure 3-50. Foreign Connection Factory Properties..... | 42 |
| Figure 3-51. Activate Changes..... | 43 |
| Figure 3-52. Domain Structure | 44 |
| Figure 3-53. Change Center..... | 45 |
| Figure 3-54. Deployments | 45 |
| Figure 3-55. Install Application Assistant | 46 |
| Figure 3-56. Locate Deployment to Install and Prepare for Deployment..... | 47 |
| Figure 3-57. Upload a Deployment to the Admin Server | 48 |
| Figure 3-58. Choose Targeting Style | 49 |
| Figure 3-59. Select Deployment Targets | 49 |
| Figure 3-60. Optional Settings..... | 50 |
| Figure 3-61. Review Your Choices and Click Finish | 51 |
| Figure 3-62. Settings for DATUP..... | 52 |
| Figure 3-63. Activate Changes..... | 53 |
| Figure 3-64. Domain Structure | 53 |
| Figure 3-65. Summary of Deployments..... | 54 |
| Figure 3-66. Start Application Assistant..... | 54 |
| Figure 3-67. Summary of Deployments – DATUP Deployment Active | 55 |
| Figure 4-1. Domain Structure | 57 |
| Figure 4-2. Change Center..... | 58 |
| Figure 4-3. Summary of Deployments – Stopping DATUP | 58 |
| Figure 4-4. Force Stop Application Assistant..... | 59 |
| Figure 4-5. Summary of Deployments – DATUP Deployment Prepared..... | 59 |
| Figure 4-6. Delete Application Assistant..... | 60 |
| Figure 4-7. Summary of Deployments – DATUP Deployment Deleted | 60 |
| Figure 4-8. Activate Changes | 61 |
| Figure B-1: Combined DATUP/PECS Architecture Diagram..... | 2 |

LIST OF TABLES

| | |
|---|----|
| Table 3-1. Terminology | 6 |
| Table 3-2. Optional Site Configuration Properties | 43 |

(This page included for two-sided copying.)

1 Project Scope

1.1 Project Description

The goal of the VHA Pharmacy Reengineering (PRE) project is to design and develop a re-engineered pharmacy system, incorporating changes that have been made to the Enterprise Architecture and changes in pharmacy business processes. The intent of the PRE program is to ensure that no current system functionality is lost, but that it is either replicated in the new system or replaced by improved process and functionality. While the overall plan is still based on designing and implementing a complete pharmacy system, the scope of the effort has been defined to address a focused subset of the PRE functionality confined to the Data Update (DATUP) process.

1.2 PRE Project Goals and Objectives

The objective of the PRE project is to facilitate the improvement of pharmacy operations, customer service, and patient safety for the VHA. The PRE project will help address the identified goals and vision for the VHA Pharmacy System.

The goal for the PRE project is a seamless and integrated nationally-supported system that is an integral part of the HealthVet-Veterans Health Information Systems & Technology Architecture (VistA) environment. To meet this goal, the PRE project will enhance pharmacy data exchange, as well as clinical documentation capabilities, in a truly integrated fashion to improve operating efficiency and patient safety. Additionally, it will provide a flexible technical environment to adjust to future business conditions and to meet patient needs in the clinical environment. Achieving this goal will enable resolution of current pharmacy issues, improve patient safety, and facilitate long-term process stability.

1.3 DATUP Background

DATUP supports the Medication Order Check Healthcare Application (MOCHA) by performing data source updates. MOCHA conducts order checks using First Databank (FDB) MedKnowledge Framework¹ within the existing VistA pharmacy application. FDB MedKnowledge Framework is a data product that provides the latest identification and safety information on medications. Additionally, FDB provides the latest algorithms used to perform order checks. DATUP processes the data updates associated with FDB MedKnowledge Framework. The order checks performed by MOCHA include:

- Drug-Drug Order Check – Check interactions between two or more drugs, including interaction monographs.
- Duplicate Therapy Order Check – Check for duplicated drug classifications between two or more drugs.
- Drug-Dose Order Check – Check minimum and maximum single doses, verify the dosing schedule, and provide the normal dosing range.

¹ At the time of development, this product was known as FDB Drug Information Framework (commonly abbreviated as FDB-DIF). The references to FDB-DIF in this manual are necessary due to previously completed code and instructions that could not be changed to match the new product name.

1.4 Related Documents

A complete list of documents relating to the PRE project and the DATUP development effort can be found in the Glossary and Acronym List (Version 5.0, dated September 26, 2008).

2 Document Overview

The information contained in this Local Data Update (DATUP) Installation Guide is specific to DATUP development, which supports the MOCHA component. This section defines the layout of this document and provides an outline of the document structure.

2.1 Document Background

This document details the steps required to install the DATUP software at a local site, the terminology used for the configuration and deployment of the software, and the assumptions for installing the software. Additionally, this document details how to install and configure the database environment. This document accompanies the delivery of the DATUP.v2.0.00.001CM software release. The DATUP Version Description Document (Version 2.0.00.001CM) is delivered as a companion document to this Installation Guide. Refer to the Version Description Document for more information on the software inventory and versions used in the DATUP.v2.0.00.001CM software release.

2.2 Overview

The following list provides a brief description of the sections included in this document:

- Section 1: Provides introductory material delineating the purpose of the PRE project and the scope of the MOCHA effort
- Section 2: Presents an overview of the layout of the document
- Section 3: Presents the installation instructions for the DATUP v2.0.00.001CM software release
- Section 4: Presents verification steps to verify that the installation was successful

Text in a `Courier New` font indicates WebLogic Console panels or text, commands, and settings that must be typed, executed, or configured to complete the installation.

(This page included for two-sided copying.)

3 Installation Instructions

The following instructions detail the steps required to perform a *fresh* installation of the DATUP software at a local site. For *upgrade* installation instructions see Section 4. Section 3.1 details the terminology used for the configuration and deployment of the DATUP software. Section 3.2 outlines the assumptions for installing the DATUP software. While the system may be configured to run outside the given assumptions, doing so requires modifications that are not detailed in this document. Section 3.3 describes how to install and configure the DATUP software properly. Finally, Section 3.3 provides an overview of the Database Server component.

In order to understand the installation and verification process, the reader should be familiar with the WebLogic console shown in Figure 3-1. The WebLogic console is a Web page viewable from any Internet browser; however, Internet Explorer, Version 7, is recommended. The WebLogic console is generally divided into two sections. The left section contains the Change Center, Domain Structure, and other informational panels. The right section displays panels containing additional options or configuration details. Note: With the exception of the Change Center and Domain Structure references, further references to WebLogic console panels refer to panels in the right section of the WebLogic console.

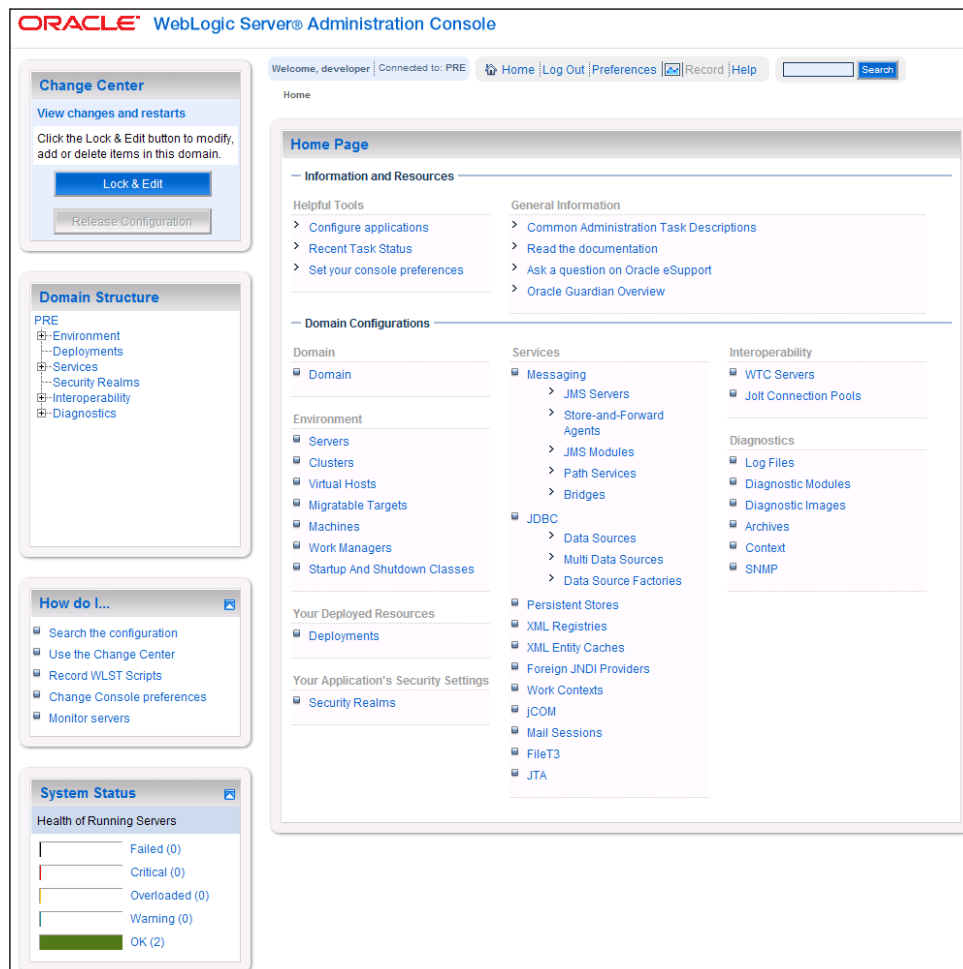


Figure 3-1. WebLogic Console

3.1 Terminology

In an effort to make these installation instructions as general as possible for installation at any site, a few terms are used throughout the instructions with the intent that they be replaced with site-specific values.

Table 3-1 contains a list of those terms used only within this document as well as sample site-specific values for each term. Additionally, references to the DATUP-L-1 server may be replaced with the site-specific name of the destination server at the installation site.

Table 3-1. Terminology

| Term | Definition | Sample |
|--|--|--|
| Database Server | Machine on which Caché is installed and runs | DATUP-L-1-DB |
| Deployment Machine | Site-specific machine on which WebLogic is installed and runs | DATUP-L-1 |
| Deployment Server | WebLogic managed server where DATUP is deployed | LocalPharmacyServer |
| Deployment Server Port | Port on which the Deployment Server is listening | 8010 |
| Deployment Server's class path directory | Folder location on the Deployment Server where libraries on the class path are located (see WebLogic documentation for instructions on setting a WebLogic managed server's class path) | /opt/bea/domains/PRE/lib |
| Java Database Connectivity (JDBC) Universal Resource Locator (URL) | URL to connect to Caché database | jdbc:Cache://DATUP-l-1-db:1972/FDB_DIF |

3.2 Assumptions

The installation instructions found within this guide are intended to be performed on a clean installation of WebLogic 10.3, with a separate managed server to act as the Deployment Server. For details on completing the installation of the following items, please refer to each item's installation and configuration documentation supplied by Oracle.

For successful deployment of the DATUP software at a site, the following assumptions must be met:

- The Deployment Server is configured and running.
- WebLogic is configured to run with the Java™ Standard Edition Development Kit, Version 1.6+.
- Access to the WebLogic console is by means of any valid administrative user name and password.
- The proper Caché database driver libraries for the chosen deployment environment are present on the class path for the respective Deployment Servers.
- Red Hat Enterprise Linux 5.2 operating system is properly installed.
- Domain Name Server (DNS) resolution is configured for the DATUP server.
- The installation instructions are followed in the order that the sections are presented within this Installation Guide.
- FDB-DIF v3.3 database is installed on the Database Server. Installation instructions are provided in FDB-DIF Installation/Migration guide. Contact the PRE Configuration Manager who should be identified on the project's Technical Services Project Repository (TSPR) site for a copy of the guide and installations/migration scripts.

3.3 Database Installation and Configuration

The FDB MedKnowledge Framework database used by DATUP requires Caché to be successfully installed. The Caché database has specific installation procedures and files for each operating system. Red Hat Linux must be successfully installed prior to installing the Caché database. A successful installation of a Caché database instance is one in which the installation guide procedures are followed, resulting in an error-free installation.

The installation of the Caché database is described in the Caché Installation Guide, Version 2008.2, Section 4, Installing Caché on UNIX and Linux. The standard installation should be used to install the Caché database server software.

For successful deployment of the DATUP 2.0 software at a site, the FDB-DIF v3.3 database must be installed. Installation instructions are provided in FDB-DIF Installation/Migration guide. Contact the PRE Configuration Manager who should be identified on the project's Technical Services Project Repository (TSPR) site for a copy of the guide and installations/migration scripts.

3.4 WebLogic Installation Instructions

The following sections detail the steps required to configure and deploy DATUP onto WebLogic at a local site.

3.4.1 Class Path

In order for the Caché JDBC data source to be added to the WebLogic configuration, the Caché JDBC driver must first be added to the Deployment Server's class path. Use the JDBC driver provided within the Caché distribution and the WebLogic documentation to add the driver to the class path.

The local DATUP Enterprise Application Archive (EAR) file contains all the required libraries for the proper functioning of the application. If any other applications have been deployed to the Deployment Server, there may be conflicting third-party libraries in the Deployment Server's class path that will cause DATUP to operate differently than expected. If versions on the Deployment Server's class path differ

from those defined in the DATUP Version Description Document (Version 1.0.00.003, dated December 3, 2010), the preferred solution is to remove the library from the Deployment Server's class path. If that is not possible, replace the libraries with the DATUP versions.

3.4.2 WebLogic Server Startup Configuration

DATUP requires additional arguments added to the WebLogic Server's Server Start properties. This section details the steps to add the arguments to the server

1. Open and log into the WebLogic console, using an administrative user name and password. The WebLogic console is located at: `http://<Deployment Machine>:7001/console`.
2. Within the Domain Structure panel found in the left column of the WebLogic console, click on the `Services > JDBC > Data Sources` node. For reference, see Figure 3-2.

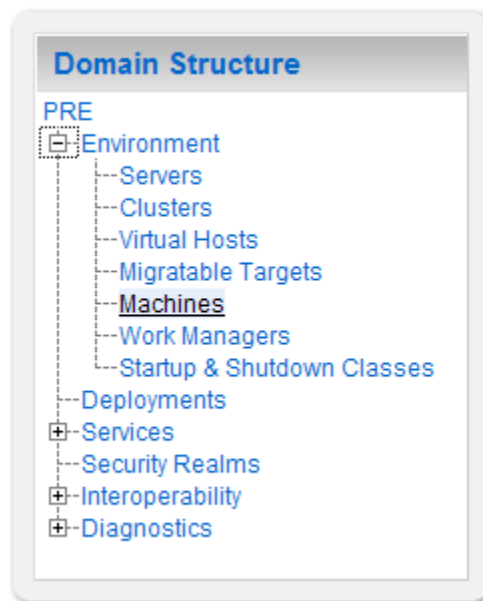


Figure 3-2. Domain Structure

3. Within the Change Center panel found in the left column of the WebLogic console, click Lock & Edit. For reference, see Figure 3-3.

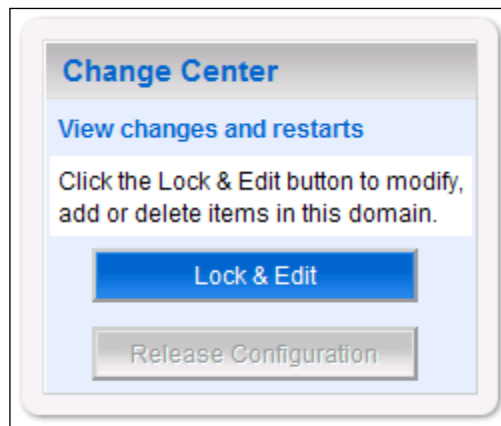


Figure 3-3. Change Center

4. Click on the server name corresponding to the deployment server in the Summary of Servers panel found in the right column of the WebLogic console. For reference, see Figure 3-4.

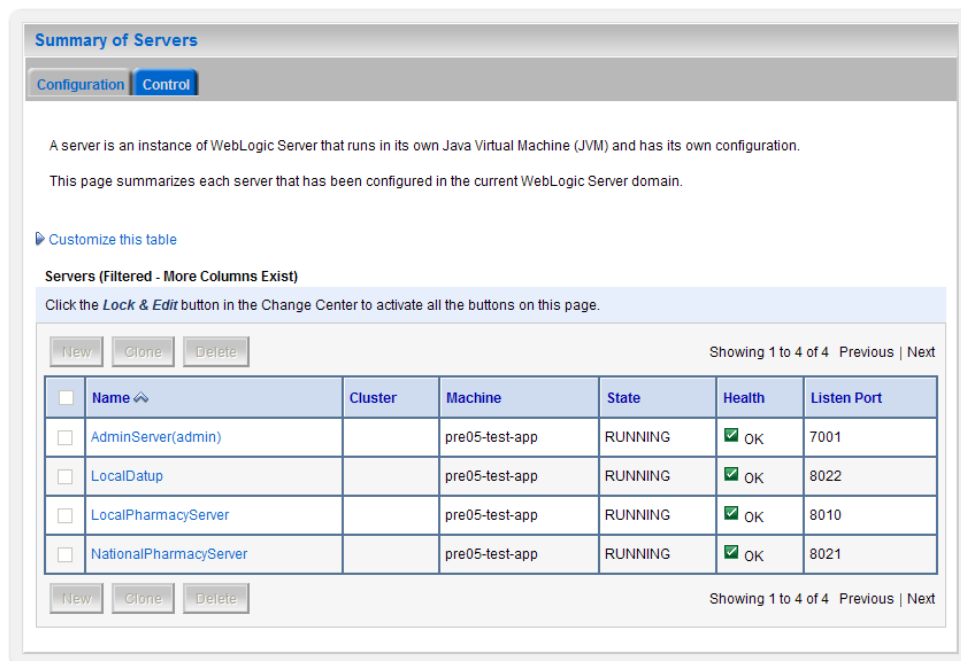


Figure 3-4. Summary of Servers

- WebLogic will now display the panel **Settings** for **Deployment Server** in the right column of the console, where configuration of the Deployment Server are set. For reference, see Figure 3-5.

The screenshot shows the 'Settings for LocalDatup' page in the WebLogic console. The page has a top navigation bar with tabs: Configuration, Protocols, Logging, Debug, Monitoring, Control, Deployments, Services, Security, and Notes. Below this is a sub-navigation bar with tabs: General, Cluster, Services, Keystores, SSL, Federation Services, Deployment, Migration, Tuning, Overload, Health Monitoring, and Server Start. The 'General' tab is selected. A 'Save' button is at the top left. The main content area contains a description: 'Use this page to configure general features of this server such as default network communications.' and a link 'View JNDI Tree'. Below this is a table of settings:

| | | |
|---|----------------|--|
| Name: | LocalDatup | An alphanumeric name for this server instance. More Info... |
| Machine: | pre05-test-app | The WebLogic Server host computer (machine) on which this server is meant to run. More Info... |
| Cluster: | (Stand-Alone) | The cluster, or group of WebLogic Server instances, to which this server belongs. More Info... |
| Listen Address: | 129.162.101.74 | The IP address or DNS name this server uses to listen for incoming connections. More Info... |
| <input checked="" type="checkbox"/> Listen Port Enabled | | Specifies whether this server can be reached through the default plain-text (non-SSL) listen port. More Info... |
| Listen Port: | 8022 | The default TCP port that this server uses to listen for regular (non-SSL) incoming connections. More Info... |
| <input type="checkbox"/> SSL Listen Port Enabled | | Indicates whether the server can be reached through the default SSL listen port. More Info... |
| SSL Listen Port: | 7002 | The TCP/IP port at which this server listens for SSL connection requests. More Info... |
| <input type="checkbox"/> Client Cert Proxy Enabled | | Specifies whether the HttpClusterServlet proxies the client certificate in a special header. More Info... |
| Java Compiler: | javac | The Java compiler to use for all applications hosted on this server that need to compile Java code. More Info... |

At the bottom, there is an 'Advanced' section with a 'Save' button.

Figure 3-5. Settings for Deployment Server

- Click on the **Server Start** tab.

- WebLogic will now display the panel **Server Start** tab in the **Settings** for **Deployment Server** in the right column of the console, where configuration of the **Deployment Server** are set. For reference, see Figure 3-6.

Settings for LocalDatup

Configuration Protocols Logging Debug Monitoring Control Deployments Services Security Notes

General Cluster Services Keystores SSL Federation Services Deployment Migration Tuning Overload Health Monitoring **Server Start**

Save

Node Manager is a WebLogic Server utility that you can use to start, suspend, shut down, and restart servers in normal or unexpected conditions. Use this page to configure the startup settings that Node Manager will use to start this server on a remote machine.

Java Home: The Java home directory (path on the machine running Node Manager) to use when starting this server. [More Info...](#)

Java Vendor: The Java Vendor value to use when starting this server. For example, BEA, Sun, HP etc. [More Info...](#)

BEA Home: The BEA home directory (path on the machine running Node Manager) to use when starting this server. [More Info...](#)

Root Directory: The directory that this server uses as its root directory. This directory must be on the computer that hosts the Node Manager. If you do not specify a Root Directory value, the domain directory is used by default. [More Info...](#)

Class Path: The classpath (path on the machine running Node Manager) to use when starting this server. [More Info...](#)

`/opt/bea/domains/PRE/lib:/opt/jdk/lib/tools.jar:/opt/bea/wls
server_10.3/server/lib/weblogic.jar:/opt/bea/domains/PRE/lib/
jdbc/JDK16-CacheDB-
2008.2.1.902.jar:/opt/bea/domains/PRE/lib/log/log4j-
1.2.14.jar`

Arguments: The arguments to use when starting this server. [More Info...](#)

`-Xms1024m -Xmx1024m -XX:PermSize=1024m -
XX:MaxPermSize=1024m -
Dlog4j.logs.dir=servers/LocalDatup/logs -
Dweblogic.JobScheduler=true -
Dpeps.datup.configuration=/opt/fdb_local_datup_configuration
.properties`

Security Policy File: The security policy file (directory and filename on the machine running Node Manager) to use when starting this server. [More Info...](#)

User Name: The user name to use when booting this server. [More Info...](#)

Password: The password of the username used to boot the server and perform server health monitoring. [More Info...](#)

Confirm Password:

Save

Figure 3-6. Server Start Tab

- Insert the following text in the **Arguments** box:

```
-Xms1024m
-Xmx1024m
-XX:PermSize=1024m
-XX:MaxPermSize=1024m
-Dweblogic.JobScheduler=true
```

Also add a arguments for Log4j file and other Log files. (see for reference below, modify path per your server configuration) :-

```
-Dlog4j.configuration=file:/u01/user_proj/domains/PECS/log4j.xml
```

```
-Dlog4j.logs.dir=servers/NationalPharmacyServer/logs  
-Dpeps.datup.configuration=/opt/fdb_datup_configuration.properties
```

9. Click the Save Button
10. Within the Change Center panel in the left column of the WebLogic console, click Activate Changes. For reference, see Figure 3-7.

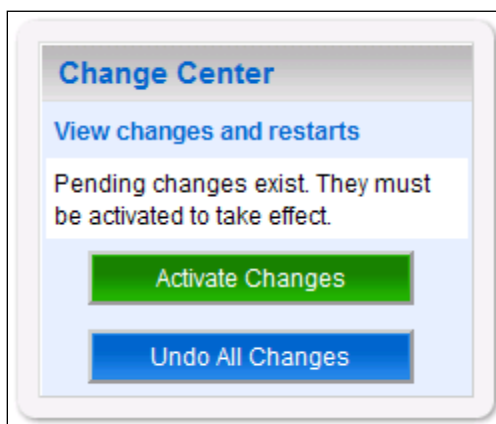


Figure 3-7. Activate Changes

3.4.3 Local JDBC Data Source Configuration

DATUP uses a database connection by means of a data source to DIF in order to perform order checks. Complete the following steps to create a new connection pool and data source for DIF.

1. Open and log into the WebLogic console, using an administrative user name and password. The WebLogic console is located at: <http://<Deployment Machine>:7001/console>.

2. Within the Domain Structure panel found in the left column of the WebLogic console, click on the Services > JDBC > Data Sources node. For reference, see Figure 3-8.

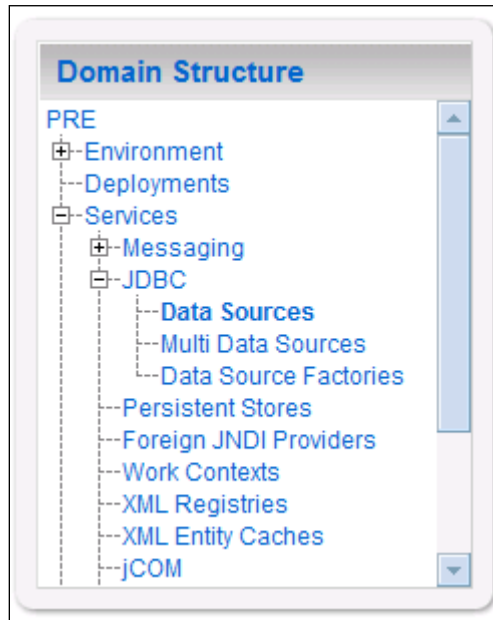


Figure 3-8. Domain Structure

3. Within the Change Center panel found in the left column of the WebLogic console, click Lock & Edit. For reference, see Figure 3-9.

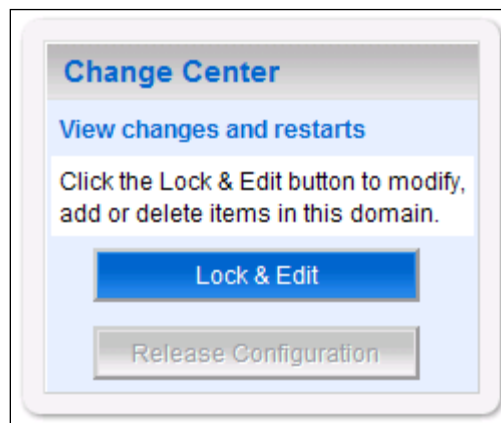


Figure 3-9. Change Center

4. Click New found in the Summary of JDBC Data Sources panel found in the right column of the WebLogic console. For reference, see Figure 3-10.

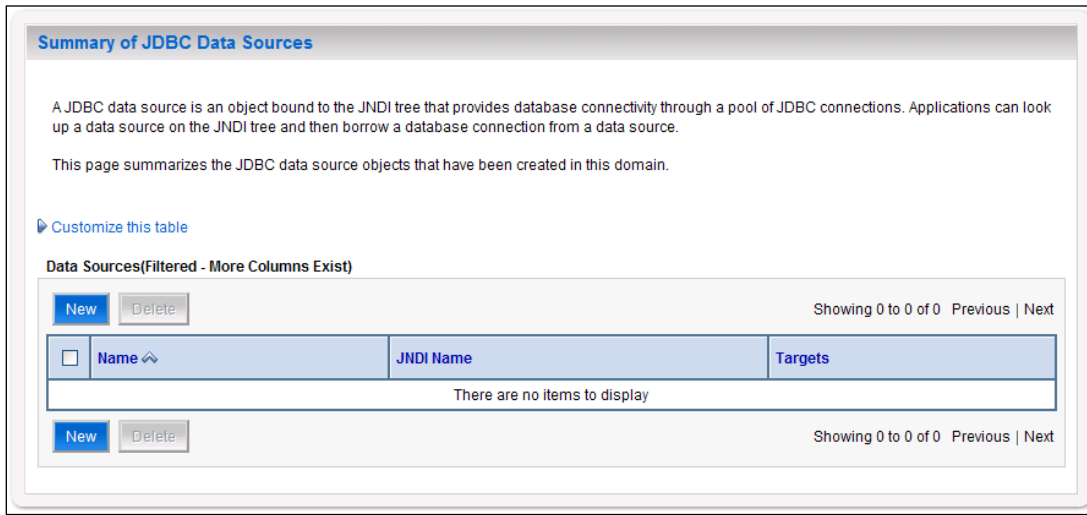


Figure 3-10. Summary of JDBC Data Sources

5. WebLogic will now display the panel **Create a New JDBC Data Source** in the right column of the console, where details of the new data source are set. For reference, see Figure 3-11.

Create a New JDBC Data Source

Back Next Finish Cancel

JDBC Data Source Properties

The following properties will be used to identify your new JDBC data source.

* Indicates required fields

What would you like to name your new JDBC data source?

Name: FDB-DIF

What JNDI name would you like to assign to your new JDBC Data Source?

JNDI Name: datasource/FDB-DIF

What database type would you like to select?

Database Type: Cache

What database driver would you like to use to create database connections?

Database Driver: Intersystems's Cache Driver (Type 4) Versions: Any

Back Next Finish Cancel

Figure 3-11. JDBC Data Source Properties

6. For the Name, type FDB-DIF.
7. For the JNDI Name, type datasource/FDB-DIF.
8. For the Database Type, select Cache.
9. For the Database Driver, verify that Intersystems's Cache Driver (Type 4) Versions: Any is selected.
10. Click Next.

11. WebLogic will now display the panel `Transaction Options` in the right column of the console, where the transaction attributes for this data source are set. For reference, see Figure 3-12.

The screenshot shows a wizard window titled "Create a New JDBC Data Source". At the top, there are four buttons: "Back", "Next", "Finish", and "Cancel". The "Next" button is highlighted. Below the buttons, the section is titled "Transaction Options". The text reads: "You have selected non-XA JDBC driver to create database connection in your new data source." followed by "Does this data source support global transactions? If yes, please choose the transaction protocol for this data source." There are four options, each with a radio button and a description:

- ☒ **Supports Global Transactions**
Select this option if you want to enable non-XA JDBC connections from the data source to participate in global transactions using the *Logging Last Resource* (LLR) transaction optimization. Recommended in place of Emulate Two-Phase Commit.
- ☐ **Logging Last Resource**
Select this option if you want to enable non-XA JDBC connections from the data source to emulate participation in global transactions using JTA. Select this option only if your application can tolerate heuristic conditions.
- ☒ **Emulate Two-Phase Commit**
Select this option if you want to enable non-XA JDBC connections from the data source to participate in global transactions using the one-phase commit transaction processing. With this option, no other resources can participate in the global transaction.
- ☐ **One-Phase Commit**

At the bottom of the panel, there are four buttons: "Back", "Next", "Finish", and "Cancel". The "Next" button is highlighted.

Figure 3-12. Transaction Options

12. Select the `Emulate Two-Phase Commit` radio button.
13. Click `Next`.

14. WebLogic will now display the panel **Connection Properties** in the right column of the console, where the connection pool attributes are set. For reference, see Figure 3-13.

The screenshot shows a web-based wizard titled "Create a New JDBC Data Source". At the top, there are four buttons: "Back", "Next", "Finish", and "Cancel". Below this is a section titled "Connection Properties" with the instruction "Define Connection Properties." The form contains several fields with labels and prompts:

- Prompt: "What is the name of database you would like to connect to?"
Field: "Database Name:" with value "FDB_DIF"
- Prompt: "What is the name or IP address of the database server?"
Field: "Host Name:" with value "datup-l-1"
- Prompt: "What is the port on the database server used to connect to the database?"
Field: "Port:" with value "1971"
- Prompt: "What database account user name do you want to use to create database connections?"
Field: "Database User Name:" with value "fdb_dif"
- Prompt: "What is the database account password to use to create database connections?"
Field: "Password:" with masked characters "••••••"
- Field: "Confirm Password:" with masked characters "••••••"

At the bottom of the form, there are four buttons: "Back", "Next", "Finish", and "Cancel".

Figure 3-13. Connection Properties

15. For **Database Name**, type the name of the Caché database to which DATUP will connect. For example, FDB_DIF.
16. For **Host Name**, type the name of the machine on which Caché is running. For example, DATUP-L-1-DB.
17. For **Port**, type the port on which Caché is listening. For example, 1972.
18. For **Database User Name**, type the user's name to connect to the FDB database. For example, developer. The user entered should be the same as configured during the FDB database setup using the FDB-DIF Installation/Migration guide. Verify username and password with the DBA.
19. For **Password** and **Confirm Password**, type the password for the user given previously. For example, pharmacy.
20. Click **Next**.

21. WebLogic will now display the panel **Test Database Connection** in the right column of the console, where the new data source can be tested. For reference, see Figure 3-14.

Create a New JDBC Data Source

Test Configuration Back Next Finish Cancel

Test Database Connection

Test the database availability and the connection properties you provided.

What is the full package name of JDBC driver class used to create database connections in the connection pool?
(Note that this driver class must be in the classpath of any server to which it is deployed.)

Driver Class Name:

What is the URL of the database to connect to? The format of the URL varies by JDBC driver.

URL:

What database account user name do you want to use to create database connections?

Database User Name:

What is the database account password to use to create database connections?
(Note: for secure password management, enter the password in the Password field instead of the Properties field below)

Password:

Confirm Password:

What are the properties to pass to the JDBC driver when creating database connections?

Properties:

What table name or SQL statement would you like to use to test database connections?

Test Table Name:

Test Configuration Back Next Finish Cancel

Figure 3-14. Test Database Connection

22. Leave all values as set by default, with the exception of **Test Table Name**. For this attribute, type `fdb_version`.
23. Click **Next**.

24. WebLogic will now display the panel **Select Targets** in the right column of the console, where the target server is selected for the new data source. For reference, see Figure 3-15.

The screenshot shows a web interface titled "Create a New JDBC Data Source". At the top, there are four buttons: "Back", "Next", "Finish", and "Cancel". Below this is a section titled "Select Targets" with a descriptive paragraph: "You can select one or more targets to deploy your new JDBC data source. If you don't select a target, the data source will be created but not deployed. You will need to deploy the data source at a later time." Below the text is a table with the heading "Servers". It contains two rows: "AdminServer" with an unchecked checkbox and "LocalPharmacyServer" with a checked checkbox. At the bottom, there are again four buttons: "Back", "Next", "Finish", and "Cancel".

Figure 3-15. Select Targets

25. Select the Deployment Server as the target. For example, LocalPharmacyServer.
26. Click **Finish**.
27. WebLogic will now display the panel **Summary of JDBC Data Sources** in the right column of the console, where the newly created data source is displayed. For reference, see Figure 3-16.

The screenshot shows a web interface titled "Summary of JDBC Data Sources". It contains a descriptive paragraph about JDBC data sources and a link "Customize this table". Below this is a section titled "Data Sources(Filtered - More Columns Exist)" with "New" and "Delete" buttons and a "Showing 1 to 1 of 1 Previous | Next" indicator. A table follows with three columns: "Name", "JNDI Name", and "Targets". The table has one row with the values "FDB-DIF", "datasource/FDB-DIF", and "LocalPharmacyServer". Below the table, there are "New" and "Delete" buttons and another "Showing 1 to 1 of 1 Previous | Next" indicator.

| Name | JNDI Name | Targets |
|---------|--------------------|---------------------|
| FDB-DIF | datasource/FDB-DIF | LocalPharmacyServer |

Figure 3-16. Summary of JDBC Data Sources

28. Prepared statement caching will need to be turned off to work around a defect in Cache. To do so, select the newly created data source, FDB-DIF, and navigate to the Connection Pool tab. Change the Statement Cache Size parameter to 0 then click save. For reference, see Figure 3-17.

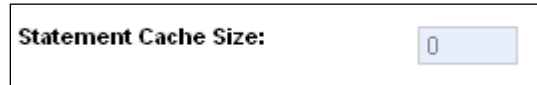


Figure 3-17. Statement Cache Size Parameter

29. Within the Change Center panel in the left column of the WebLogic console, click Activate Changes. For reference, see Figure 3-18.

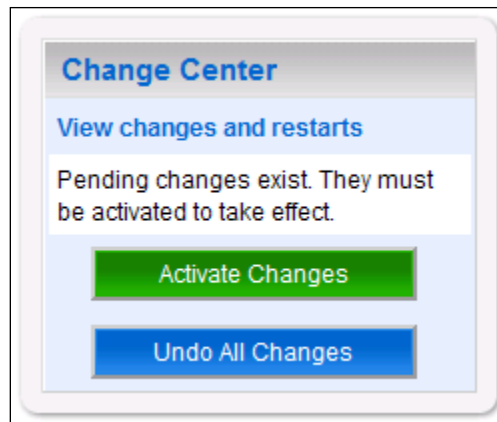


Figure 3-18. Activate Changes

3.4.4 Log4j

DATUP uses Log4j to provide debug and error logs. Although the application will function without Log4j installed, using it can be helpful to troubleshoot potential issues. Because DATUP can operate without Log4j configured, all instructions within this section are only required if debugging deployed code.

If the installation of Log4j is desired, the Java Archive (JAR) can be found within the local DATUP EAR, or it can be downloaded from the Internet. Please refer to the DATUP Version Description Document (Version 1.0.00.003, dated December 3, 2010) for the version required.

To install Log4j, the Log4j JAR must be placed on the Deployment Server's class path and the `log4j.xml` must be edited to include the DATUP appenders and loggers. Complete the following instructions to place the Log4j library on the Deployment Server's class path. If Log4j is already installed on the Deployment Server, these steps do not need to be completed.

1. Locate the Deployment Server's Class Path Directory.
2. Copy the `log4j-1.2.15.jar` file into a folder within the class path.
3. Configure WebLogic to include the Log4j library in the Deployment Server's class path. Please refer to the WebLogic documentation provided by BEA for completing this step.
4. Restart the Deployment Server to load Log4j.

With Log4j installed on the Deployment Server, the `log4j.xml` file must be modified to include the DATUP configuration. Note that the appenders place the logs under a `log` folder. This folder must be created at the same directory level at which the Deployment Server is running. For example, `/opt/boa/domains/PRE/log`. Without this folder, Log4j will not be able to create the log files specified in the DATUP configuration. Alternatively, the file locations could be altered to be placed in a different location. Follow the steps below to complete this process:

1. If Log4j has already been installed, locate the `log4j.xml` file used for the Deployment Server. Otherwise, create a new `log4j.xml` file that is either located in a folder on the Deployment Server class path, or use the `log4j.configuration` Java system property to set the location of the file. Please refer to the WebLogic provided by BEA and Log4j documentation provided by Apache to complete any of these operations.
2. Add the following configuration to the `log4j.xml` file:

```
<appender name="PepsAppender"
class="org.apache.log4j.RollingFileAppender">
    <param name="File" value="log/peps.log"/>
    <param name="Append" value="false"/>
    <param name="MaxBackupIndex" value="10"/>
    <layout class="org.apache.log4j.PatternLayout">
        <param name="ConversionPattern" value="%d{dd MMM
        yyyy hh:mm:ss a} %-5p [%c:%M] %m%n"/>
    </layout>
</appender>

<appender name="SpringAppender"
class="org.apache.log4j.RollingFileAppender">
    <param name="File" value="log/spring.log"/>
    <param name="Append" value="false"/>
    <param name="MaxBackupIndex" value="10"/>
    <layout class="org.apache.log4j.PatternLayout">
        <param name="ConversionPattern" value="%d{dd MMM
        yyyy hh:mm:ss a} %-5p [%c:%M] %m%n"/>
    </layout>
</appender>

<logger name="org.springframework" additivity="false">
    <level value="error" />
    <appender-ref ref="SpringAppender"/>
</logger>

<logger name="gov.va.med.pharmacy.peps" additivity="false">
    <level value="error" />
    <appender-ref ref="PepsAppender"/>
```

```
</logger>
```

3. If profiling is turned on and should be recorded, add the following configuration to the `log4j.xml` file:

```
<appender name="ProfileAppender"
class="org.apache.log4j.RollingFileAppender">
    <param name="File" value="log/profile.log" />
    <param name="Append" value="false" />
    <param name="MaxBackupIndex" value="10" />
    <layout class="org.apache.log4j.PatternLayout">
        <param name="ConversionPattern" value="%d{dd MMM
        yyyy hh:mm:ss a} %-5p [%c%M] %m%n" />
    </layout>
</appender>

<logger
name="gov.va.med.pharmacy.peps.common.utility.profile"
additivity="false">
    <level value="info" />
    <appender-ref ref="ProfileAppender" />
</logger>
```

4. Restart the Deployment Server to load the Log4j configuration.

The given Log4j configuration assumes that an existing `log4j.xml` file is being modified, as the configurations above are only a fragment of a complete Log4j configuration. In particular, the given configuration will only log messages for classes in the `org.springframework` and `gov.va.med.pharmacy.peps` packages and sub-packages. No other classes are covered. If additional logging is desired, other logger elements or the root element must be configured. In addition, the given Log4j configuration only logs error-level messages and optionally the info-level profiling messages. For further information, reference <http://wiki.apache.org/logging-log4j/Log4jXmlFormat>.

Note: Due to policy constraints, this document cannot support live links. Copy and paste the above link into your browser.

3.4.5 Local JMS Configuration

A DATUP local instance is comprised of a JMS module, including the remote JMS server to the DATUP national instance with destinations pointing to the local receive topic, the national receive queue, as well as, a connection factory. Complete the following instructions, in order by section, for each element of the local JMS configuration. These installation instructions must be repeated for each DATUP local site installation.

JMS Module

1. Open and log into the WebLogic console, using an administrative user name and password. The WebLogic console is located at: `http://<Deployment Machine>:7001/console`.
2. Within the Change Center panel in the left column of the WebLogic console, click Lock & Edit. For reference, see Figure 3-19.

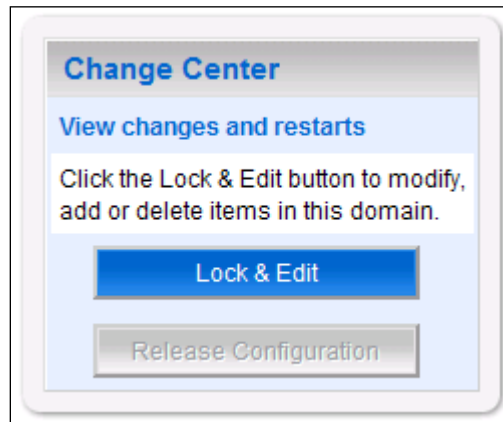


Figure 3-19. Lock & Edit

3. Within the Domain Structure panel in the left column of the WebLogic console, click the Services > Messaging > JMS Modules node. For reference, see Figure 3-20.



Figure 3-20. JMS Modules

- WebLogic will now display the panel JMS Modules in the right column of the console, where the currently configured JMS servers will be found. For reference, see Figure 3-21.

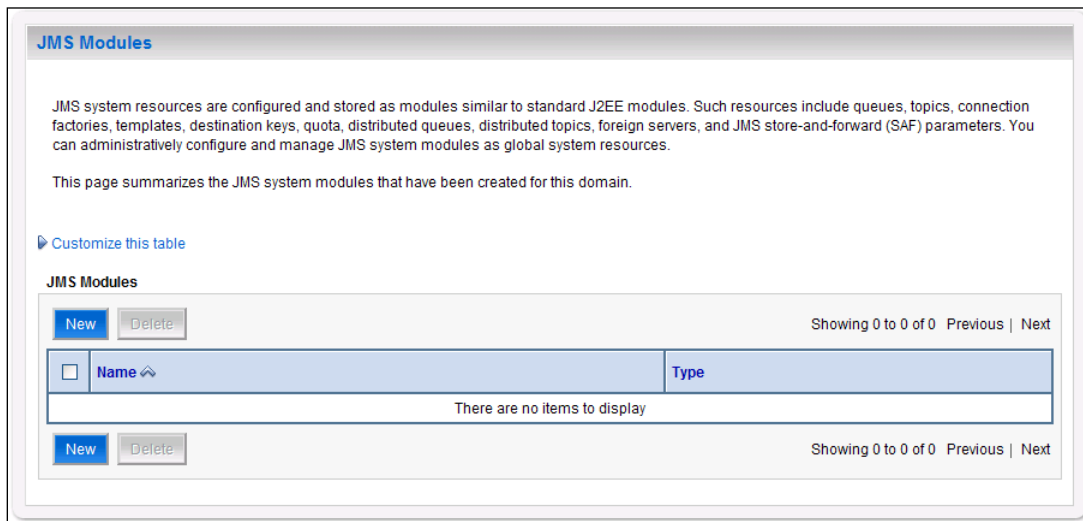


Figure 3-21. JMS Modules

- Click New.
- WebLogic will now display the panel Create JMS System Module in the right column of the console. Within the panel the following properties will be used to identify your new module, where the new JMS module will be configured. For reference, see Figure 3-22.

The screenshot shows the 'Create JMS System Module' panel in the WebLogic console. At the top, there is a title bar 'Create JMS System Module' and a set of navigation buttons: 'Back', 'Next', 'Finish', and 'Cancel'. Below this is a section titled 'The following properties will be used to identify your new module.' followed by a descriptive paragraph: 'JMS system resources are configured and stored as modules similar to standard J2EE modules. Such resources include queues, topics, connection factories, templates, destination keys, quota, distributed queues, distributed topics, foreign servers, and JMS store-and-forward (SAF) parameters. You can administratively configure and manage JMS system modules as global system resources.' Below this is a note '* Indicates required fields'. The main area contains three sections for input: 1. 'What would you like to name your System Module?' with a required field 'Name' containing the text 'LocalPharmacyJmsModule'. 2. 'What would you like to name the descriptor file name? If you do not provide a name, a default will be assigned.' with a field 'Descriptor File Name'. 3. 'Where would like to place the descriptor for this System Module, relative to the jms configuration sub-directory of your domain?' with a field 'Location In Domain'. At the bottom, there are navigation buttons: 'Back', 'Next', 'Finish', and 'Cancel'.

Figure 3-22. JMS System Module Properties

7. For Name, enter a unique name for the new JMS system module. For example, LocalPharmacyJmsModule.
8. Leave Descriptor File Name and Location In Domain blank.
9. Click Next.
10. WebLogic will now display the panel Create JMS System Module in the right column of the console. Within the panel is Targets, where the new JMS module will be configured. For reference, see Figure 3-23.

The screenshot shows a web-based configuration window titled "Create JMS System Module". At the top, there are four buttons: "Back", "Next", "Finish", and "Cancel". Below the buttons, a text block states: "The following properties will be used to target your new JMS system module. Use this page to select the server or cluster on which you would like to deploy this JMS system module. You can reconfigure targets later if you wish." Below this is a section labeled "Targets :". It contains a table with the heading "Servers". The table has two rows: the first row is "AdminServer" with an unchecked checkbox, and the second row is "LocalPharmacyServer-1" with a checked checkbox. At the bottom of the dialog, there are again four buttons: "Back", "Next", "Finish", and "Cancel".

Figure 3-23. JMS System Module Targets

11. For Targets, select the Deployment Server for the DATUP local instance. For example, LocalPharmacyServer-1.
12. Click Next.
13. WebLogic will now display the panel Create JMS System Module in the right column of the console. Within the panel is Add resources to this JMS system module, where the new JMS module will be configured. For reference, see Figure 3-24.

The screenshot shows the same "Create JMS System Module" dialog box, but now on the "Add resources to this JMS system module" section. The top buttons are "Back", "Next", "Finish", and "Cancel". Below them, a text block says: "Add resources to this JMS system module. Use this page to indicate whether you want to immediately add resources to this JMS system module after it is created. JMS resources include queues, topics, connection factories, etc." Below this is a checkbox labeled "Would you like to add resources to this JMS system module?". The checkbox is currently unchecked. At the bottom, there are four buttons: "Back", "Next", "Finish", and "Cancel".

Figure 3-24. Add Resources to JMS System Module

14. Leave the Would you like to add resources to this JMS system module? check box unchecked.
15. Click Finish.
16. Within the Change Center panel in the left column of the WebLogic console, click Activate Changes. For reference, see Figure 3-25.

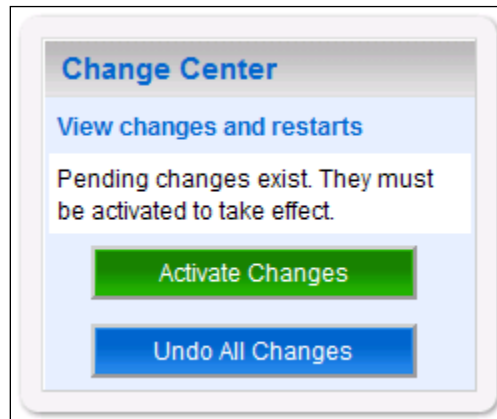


Figure 3-25. Activate Changes

Foreign JMS Server

1. Open and log on to the WebLogic console, using an administrative user name and password. The WebLogic console is located at: <http://<Deployment Machine>:7001/console>.
2. Within the Change Center panel in the left column of the WebLogic console, click Lock & Edit. For reference, see Figure 3-26.

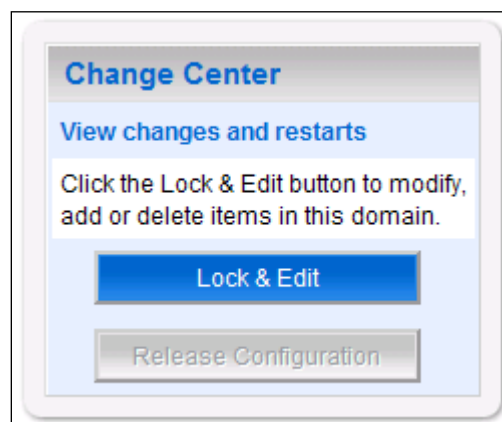


Figure 3-26. Lock & Edit

3. Within the Domain Structure panel in the left column of the WebLogic console, click the Services > Messaging > JMS Modules node. For reference, see Figure 3-27.



Figure 3-27. JMS Modules

4. WebLogic will now display the panel JMS Modules in the right column of the console, where the currently configured JMS servers will be found. For reference, see Figure 3-28.

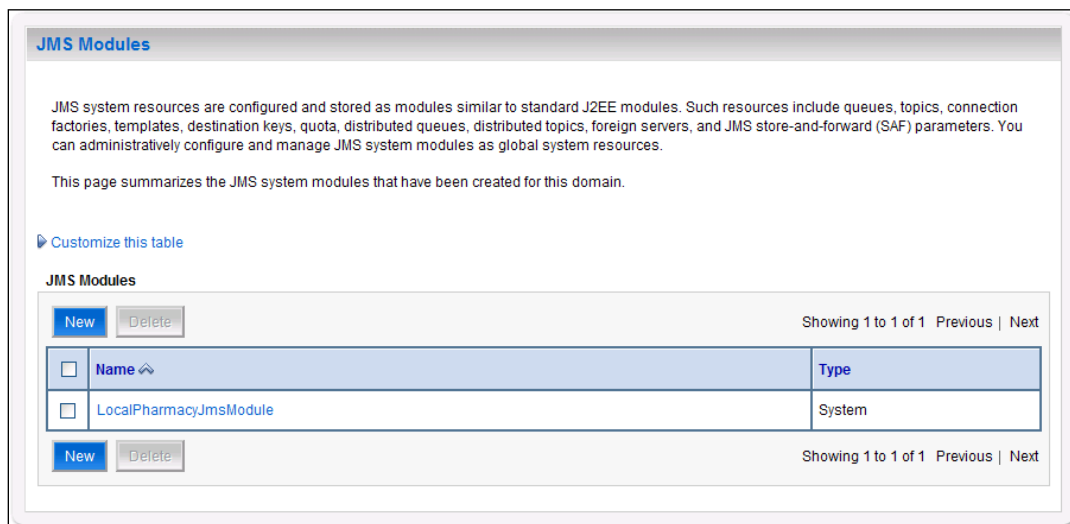


Figure 3-28. JMS Modules

5. Click on the link to the JMS system module created in the JMS Module section. For example, LocalPharmacyJmsModule.

- WebLogic will now display the panel **Settings** for **LocalPharmacyJmsModule** in the right column of the console. Within the panel is **Summary of Resources**, where the JMS module will be further configured. For reference, see Figure 3-29.

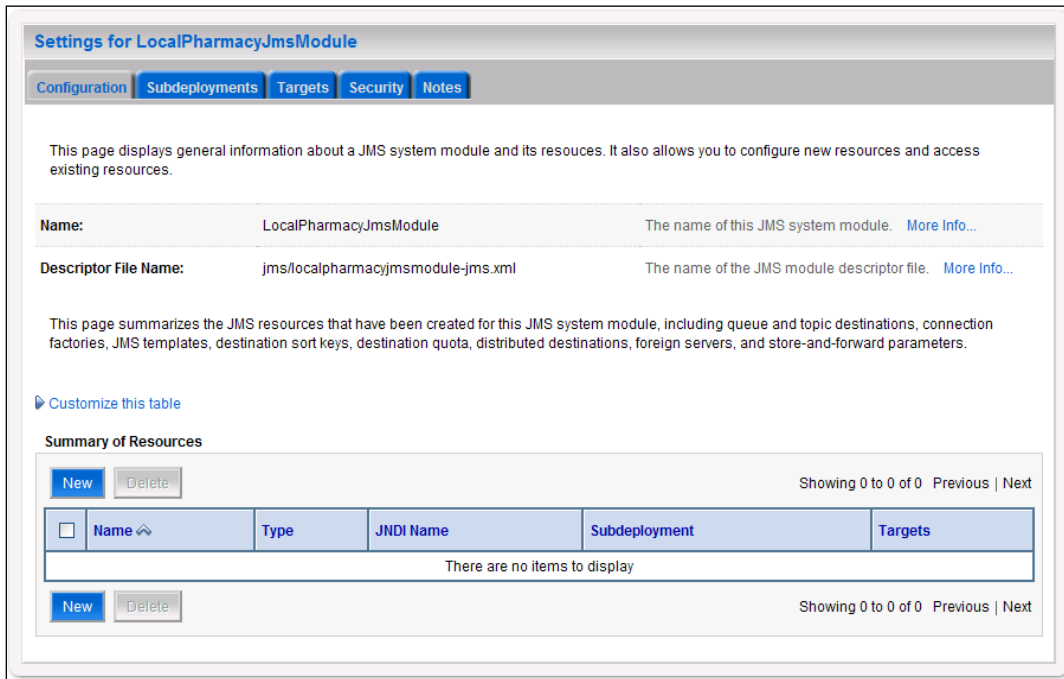


Figure 3-29. Summary of Resources

- Click **New**.

8. WebLogic will now display the panel Create a New JMS System Module Resource in the right column of the console. Within the panel is Choose the type of resource you want to create, where the JMS module will be further configured. For reference, see Figure 3-30.

Create a New JMS System Module Resource

Back Next Finish Cancel

Choose the type of resource you want to create.

Use these pages to create resources in a JMS system module, such as queues, topics, templates, and connection factories.

Depending on the type of resource you select, you are prompted to enter basic information for creating the resource. For targetable resources, like stand-alone queues and topics, connection factories, distributed queues and topics, foreign servers, and JMS SAF destinations, you can also proceed to targeting pages for selecting appropriate server targets. You can also associate targetable resources with subdeployments, which is an advanced mechanism for grouping JMS module resources and the members to server resources.

| | |
|---|---|
| <input type="radio"/> Connection Factory | Defines a set of connection configuration parameters that are used to create connections for JMS clients. More Info... |
| <input type="radio"/> Queue | Defines a point-to-point destination type, which are used for asynchronous peer communications. A message delivered to a queue is distributed to only one consumer. More Info... |
| <input type="radio"/> Topic | Defines a publish/subscribe destination type, which are used for asynchronous peer communications. A message delivered to a topic is distributed to all topic consumers. More Info... |
| <input type="radio"/> Distributed Queue | Defines a set of queues that are distributed on multiple JMS servers, but which are accessible as a single, logical queue to JMS clients. More Info... |
| <input type="radio"/> Distributed Topic | Defines a set of topics that are distributed on multiple JMS servers, but which are accessible as a single, logical topic to JMS clients. More Info... |
| <input checked="" type="radio"/> Foreign Server | Defines foreign messaging providers or remote WebLogic Server instances that are not part of the current domain. More Info... |

Figure 3-30. Choose Type of Resource to Create

9. Select Foreign Server.
10. Click Next.
11. WebLogic will now display the panel Create a New JMS System Module Resource in the right column of the console. Within the panel is Foreign Server Properties, where the JMS module will be further configured. For reference, see Figure 3-31.

Create a New JMS System Module Resource

Back Next Finish Cancel

Foreign Server Properties

The following properties will be used to identify your new foreign server. The current module is LocalPharmacyJmsModule.

* Indicates required fields

* Name: RemoteNationalPharmacy

Back Next Finish Cancel

Figure 3-31. Foreign Server Properties

12. For Name, enter a unique name for the foreign server. For example, RemoteNationalPharmacyJmsServer.

13. Click Next.
14. WebLogic will now display the panel Create a New JMS System Module Resource in the right column of the console. Within the panel is Foreign Server Properties, where the JMS module will be further configured. For reference, see Figure 3-32.

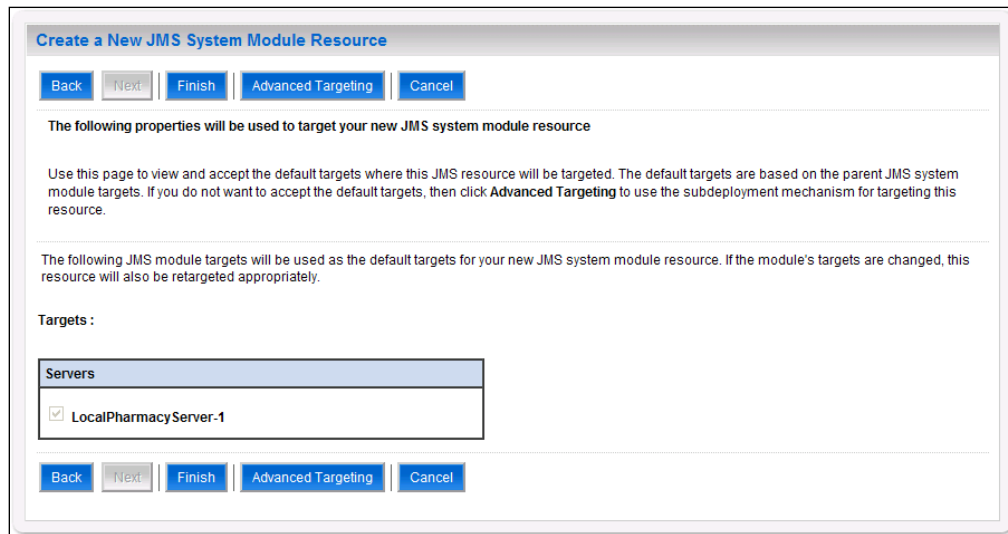


Figure 3-32. Foreign JMS Server Target

15. For Targets, verify that the target chosen is the WebLogic server for this DATUP installation.
16. Click Finish.

17. WebLogic will now display the panel **Settings** for `LocalPharmacyJmsModule` in the right column of the console. Within the panel is **Summary of Resources**, where the JMS module will be further configured. For reference, see Figure 3-33.

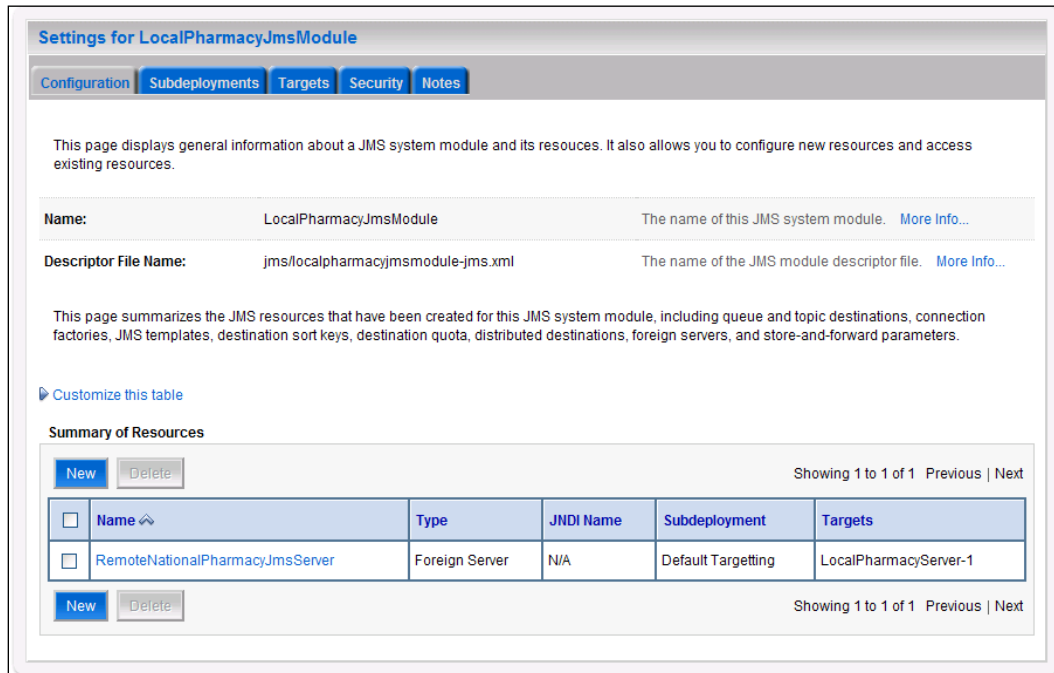


Figure 3-33. Summary of Resources

18. Click on the link for the foreign JMS server just created. For example, `RemoteNationalPharmacyJmsServer`.

19. WebLogic will now display the panel **Settings** for **RemoteNationalPharmacyJmsServer** in the right column of the console. Within the panel is **Configuration - General**, where the JMS module will be further configured. For reference, see Figure 3-34.

Settings for RemoteNationalPharmacyJmsServer

Configuration Subdeployment Notes

General Destinations Connection Factories

Save

A foreign server represents a JNDI provider that resides outside a WebLogic Server. It contains information that allows WebLogic Server to reach the remote JNDI provider. This way, a number of connection factory and destination objects (queues or topics) can be defined on one JNDI directory. Use this page to configure a foreign server.

Name: RemoteNationalPharmacyJmsServer The name of this foreign server. [More Info...](#)

JNDI Initial Context Factory: weblogic.jndi.WLInitialContextFactory The name of the class that must be instantiated to access the JNDI provider. This class name depends on the JNDI provider and the vendor that are being used. [More Info...](#)

JNDI Connection URL: t3://test-datup-n:8021 The URL that WebLogic Server will use to contact the JNDI provider. The syntax of this URL depends on which JNDI provider is being used. For WebLogic JMS, leave this field blank if you are referencing WebLogic JMS objects within the same cluster. [More Info...](#)

JNDI Properties Credential: Any Credentials that must be set for the JNDI provider. These Credentials will be part of the properties will be passed directly to the constructor for the JNDI provider's InitialContext class. Note: For secure credential management, use the Credential field. Using the Properties field results in the credential being stored and displayed as originally entered [More Info...](#)

Confirm JNDI Properties Credential:

JNDI Properties: Any additional properties that must be set for the JNDI provider. These properties will be passed directly to the constructor for the JNDI provider's InitialContext class. [More Info...](#)

☒ **Default Targeting Enabled** Specifies whether this JMS resource defaults to the parent module's targeting or uses the the subdeployment targeting mechanism. [More Info...](#)

Save

Figure 3-34. Foreign JMS Server General Configuration

20. For **JNDI Connection URL**, enter the URL to the National Deployment Server. For example, `t3://test-datup-n:8021`.

21. Leave the default values for the remaining settings JNDI Properties Credential, Confirm JNDI Properties Credential, JNDI Properties, and Default Targeting Enabled.
22. Click Save.
23. Within the Change Center panel in the left column of the WebLogic console, click Activate Changes. For reference, see Figure 3-35.

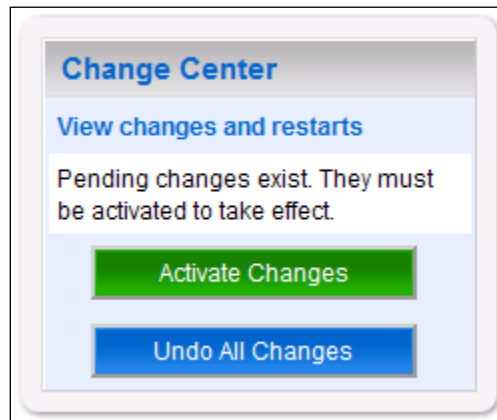


Figure 3-35. Activate Changes

Destination

1. Open and log on to the WebLogic console, using an administrative user name and password. The WebLogic console is located at: <http://<Deployment Machine>:7001/console>.
2. Within the Change Center panel in the left column of the WebLogic console, click Lock & Edit. For reference, see Figure 3-36.

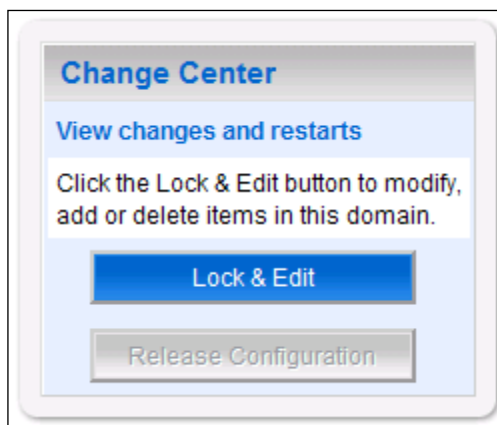


Figure 3-36. Lock & Edit

3. Within the Domain Structure panel in the left column of the WebLogic console, click the Services > Messaging > JMS Modules node. For reference, see Figure 3-37.



Figure 3-37. JMS Modules

4. WebLogic will now display the panel JMS Modules in the right column of the console, where the currently configured JMS servers will be found. For reference, see Figure 3-38.

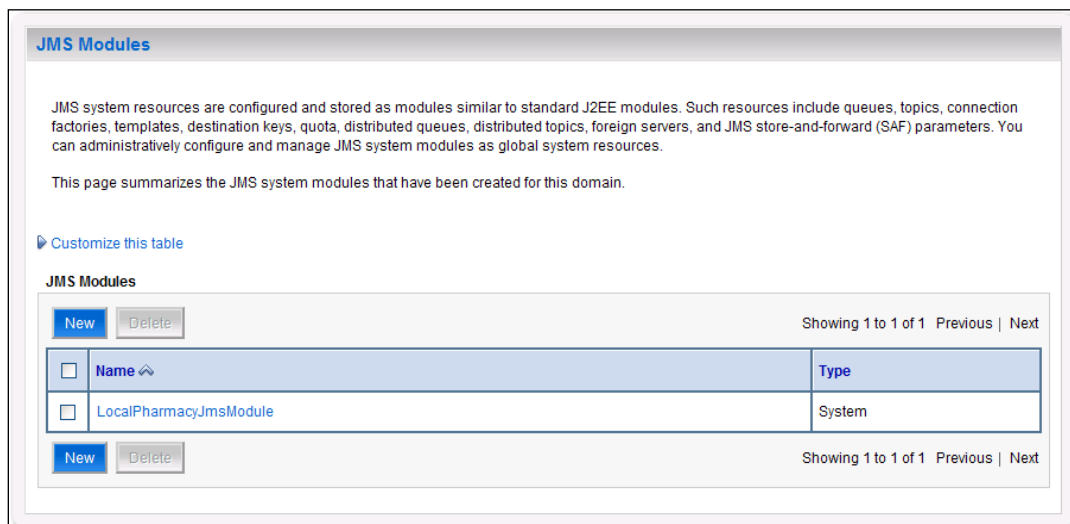


Figure 3-38. JMS Modules

5. Click on the link to the JMS system module created in the JMS Module section. For example, LocalPharmacyJmsModule.

- WebLogic will now display the panel **Settings** for `LocalPharmacyJmsModule` in the right column of the console. Within the panel is **Summary of Resources**, where the JMS module will be further configured. For reference, see Figure 3-39.

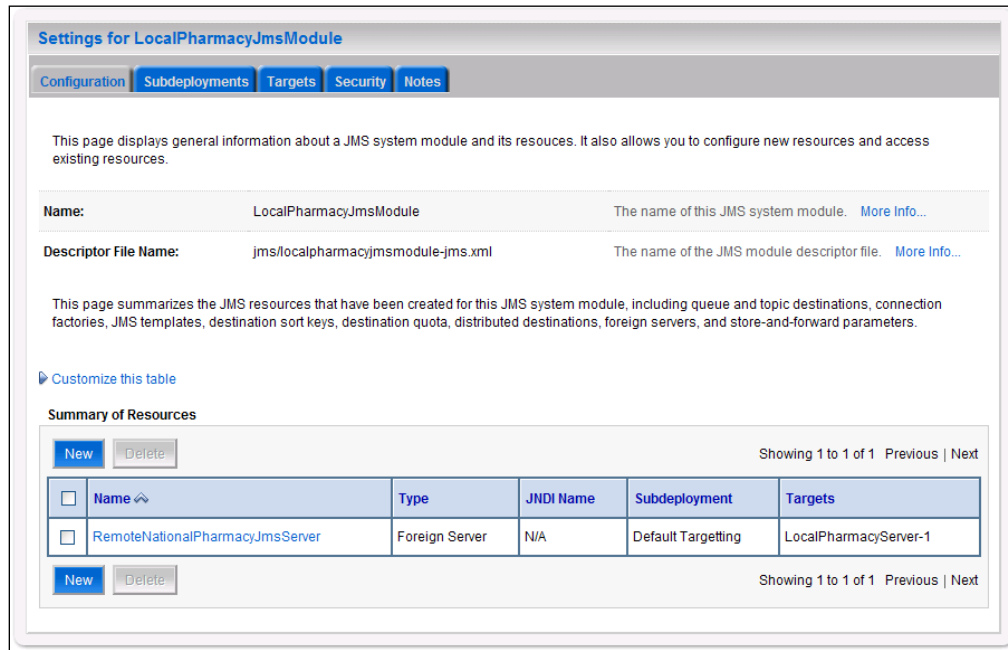


Figure 3-39. Summary of Resources

- Click on the link for the foreign JMS server created in the Foreign JMS Server section. For example, `RemoteNationalPharmacyJmsServer`.

- WebLogic will now display the panel Settings for RemoteNationalPharmacyJms Server in the right column of the console. Within the panel is Configuration - General, where the JMS module will be further configured. For reference, see Figure 3-40.

Settings for RemoteNationalPharmacyJms Server

Configuration Subdeployment Notes

General Destinations Connection Factories

Save

A foreign server represents a JNDI provider that resides outside a WebLogic Server. It contains information that allows WebLogic Server to reach the remote JNDI provider. This way, a number of connection factory and destination objects (queues or topics) can be defined on one JNDI directory. Use this page to configure a foreign server.

Name: RemoteNationalPharmacyJmsServer The name of this foreign server. [More Info...](#)

JNDI Initial Context Factory: weblogic.jndi.WLInitialCon The name of the class that must be instantiated to access the JNDI provider. This class name depends on the JNDI provider and the vendor that are being used. [More Info...](#)

JNDI Connection URL: t3://test-datup-n:8021 The URL that WebLogic Server will use to contact the JNDI provider. The syntax of this URL depends on which JNDI provider is being used. For WebLogic JMS, leave this field blank if you are referencing WebLogic JMS objects within the same cluster. [More Info...](#)

JNDI Properties Credential: Any Credentials that must be set for the JNDI provider. These Credentials will be part of the properties will be passed directly to the constructor for the JNDI provider's InitialContext class. Note: For secure credential management, use the Credential field. Using the Properties field results in the credential being stored and displayed as originally entered [More Info...](#)

Confirm JNDI Properties Credential:

JNDI Properties: Any additional properties that must be set for the JNDI provider. These properties will be passed directly to the constructor for the JNDI provider's InitialContext class. [More Info...](#)

☒ **Default Targeting Enabled** Specifies whether this JMS resource defaults to the parent module's targeting or uses the the subdeployment targeting mechanism. [More Info...](#)

Save

Figure 3-40. Foreign JMS Server General Configuration

- Select the Destinations tab.

10. WebLogic will now display the panel Settings for RemoteNationalPharmacyJmsServer in the right column of the console. Within the panel is Foreign Destinations Configuration, where the JMS module will be further configured. For reference, see Figure 3-41.

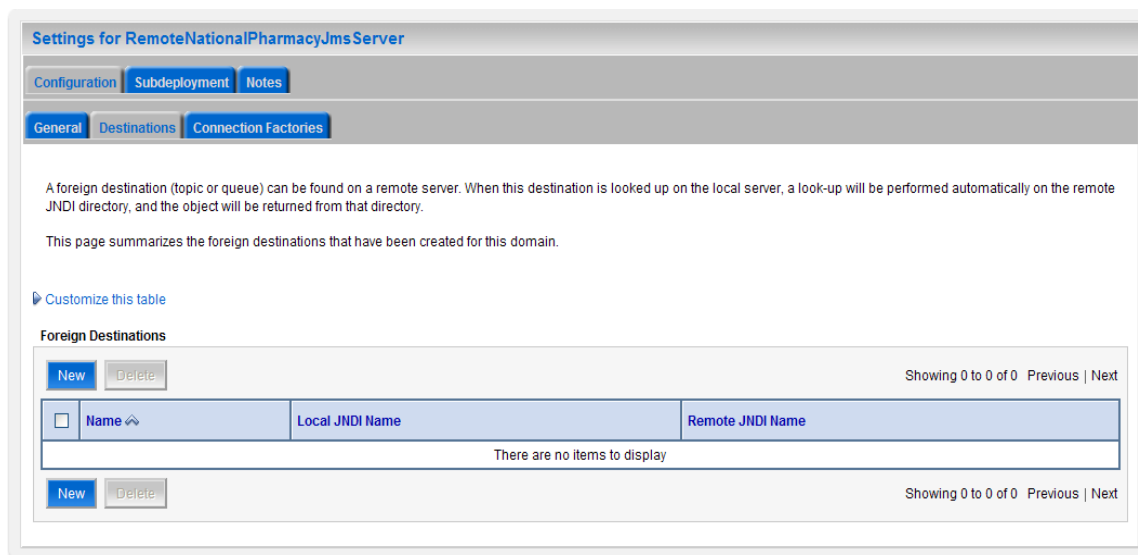


Figure 3-41. Destinations

11. Click New.
12. WebLogic will now display the panel Settings for RemoteNationalPharmacyJmsServer in the right column of the console. Within the panel is Foreign Connection Factory Properties, where the JMS module will be further configured. For reference, see Figure 3-42.

Create a New Foreign JMS Destination

OK Cancel

Foreign Destination Properties

The following properties will be used to identify your new foreign destination.

* Indicates required fields

* Name: DatupDestination

Local JNDI Name: ms/gov/va/med/pharmacy/peps/messaging-service/queue/

Remote JNDI Name: macy/peps/messaging-service/queue/national/datup/receiv

OK Cancel

Figure 3-42. Foreign Connection Factory Properties

13. For Name, enter a unique name for the foreign JMS Destination. For example, DatupDestination

14. For Local JNDI Name, enter:
jms/gov/va/med/pharmacy/peps/messagingservice/queue/national/
datup/receive.
15. For Remote JNDI Name, enter:
jms/gov/va/med/pharmacy/peps/messagingservice/queue/national/
datup/receive.
16. Click OK.
17. Within the Change Center panel in the left column of the WebLogic console, click Activate Changes. For reference, see Figure 3-43.

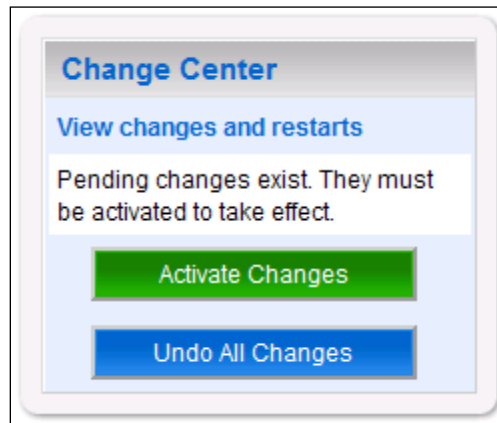


Figure 3-43. Activate Changes

Connection Factory

1. Open and log on to the WebLogic console, using an administrative user name and password. The WebLogic console is located at: <http://<Deployment Machine>:7001/console>.
2. Within the Change Center panel in the left column of the WebLogic console, click Lock & Edit. For reference, see Figure 3-44.

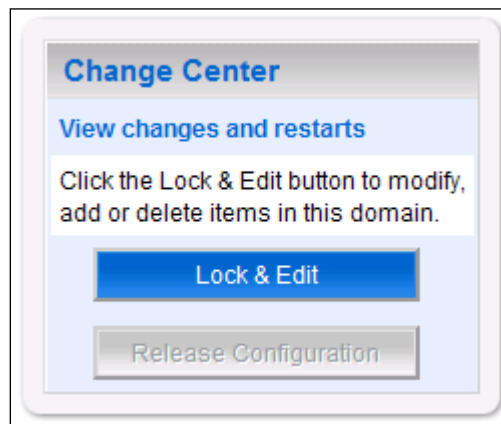


Figure 3-44. Lock & Edit

3. Within the Domain Structure panel in the left column of the WebLogic console, click the Services > Messaging > JMS Modules node. For reference, see Figure 3-45.



Figure 3-45. JMS Modules

4. WebLogic will now display the panel JMS Modules in the right column of the console, where the currently configured JMS servers will be found. For reference, see Figure 3-46.

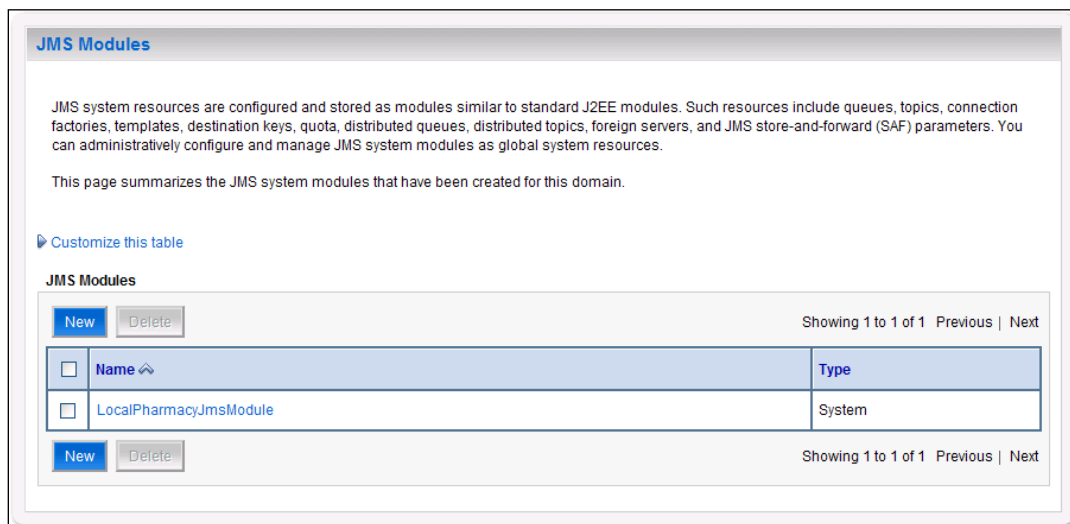


Figure 3-46. JMS Modules

5. Click on the link to the JMS system module created in the JMS Module section. For example, LocalPharmacyJmsModule.

- WebLogic will now display the panel **Settings** for `LocalPharmacyJmsModule` in the right column of the console. Within the panel is **Summary of Resources**, where the JMS module will be further configured. For reference, see Figure 3-47.

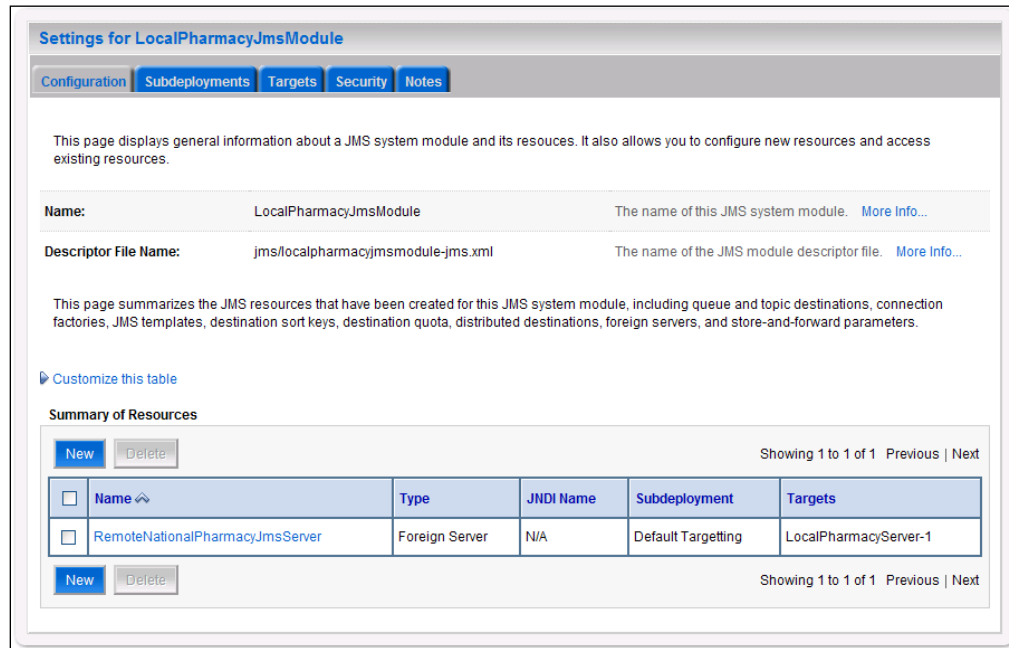


Figure 3-47. Summary of Resources

- Click on the link for the foreign JMS server created in the Foreign JMS Server section. For example, `RemoteNationalPharmacyJmsServer`.

- WebLogic will now display the panel Settings for RemoteNationalPharmacyJms Server in the right column of the console. Within the panel is Configuration - General, where the JMS module will be further configured. For reference, see Figure 3-48.

Settings for RemoteNationalPharmacyJms Server

Configuration Subdeployment Notes

General Destinations Connection Factories

Save

A foreign server represents a JNDI provider that resides outside a WebLogic Server. It contains information that allows WebLogic Server to reach the remote JNDI provider. This way, a number of connection factory and destination objects (queues or topics) can be defined on one JNDI directory. Use this page to configure a foreign server.

Name: RemoteNationalPharmacyJmsServer The name of this foreign server. [More Info...](#)

JNDI Initial Context Factory: weblogic.jndi.WLInitialCon The name of the class that must be instantiated to access the JNDI provider. This class name depends on the JNDI provider and the vendor that are being used. [More Info...](#)

JNDI Connection URL: t3://test-datup-n:8021 The URL that WebLogic Server will use to contact the JNDI provider. The syntax of this URL depends on which JNDI provider is being used. For WebLogic JMS, leave this field blank if you are referencing WebLogic JMS objects within the same cluster. [More Info...](#)

JNDI Properties Credential: Any Credentials that must be set for the JNDI provider. These Credentials will be part of the properties will be passed directly to the constructor for the JNDI provider's InitialContext class. Note: For secure credential management, use the Credential field. Using the Properties field results in the credential being stored and displayed as originally entered [More Info...](#)

Confirm JNDI Properties Credential:

JNDI Properties: Any additional properties that must be set for the JNDI provider. These properties will be passed directly to the constructor for the JNDI provider's InitialContext class. [More Info...](#)

☒ **Default Targeting Enabled** Specifies whether this JMS resource defaults to the parent module's targeting or uses the the subdeployment targeting mechanism. [More Info...](#)

Save

Figure 3-48. Foreign JMS Server General Configuration

- Select the Connection Factories tab.

10. WebLogic will now display the panel Settings for RemoteNationalPharmacyJmsServer in the right column of the console. Within the panel is Connection Factories Configuration, where the JMS module will be further configured. For reference, see Figure 3-49.

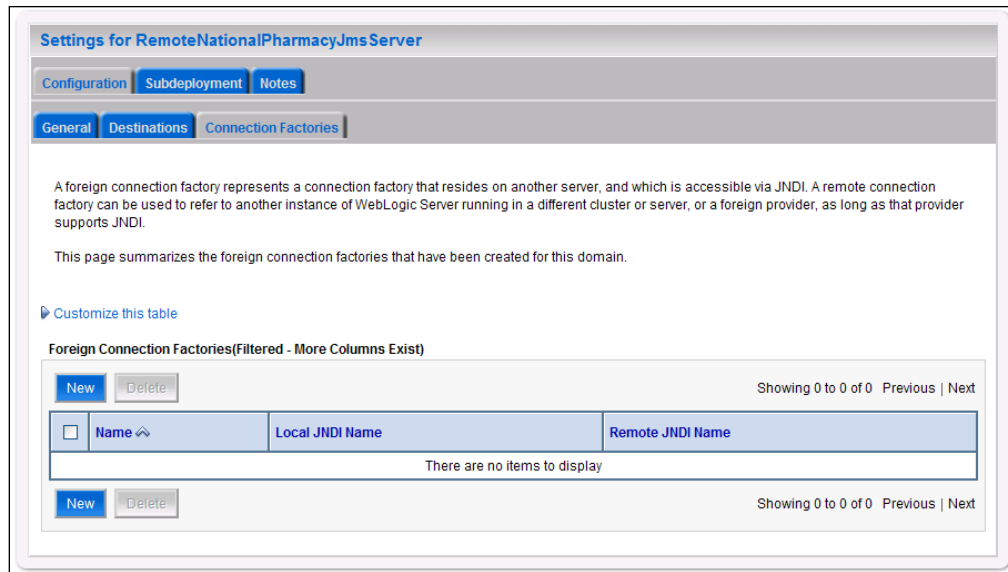


Figure 3-49. Foreign JMS Server Connection Factories

11. Click New.
12. WebLogic will now display the panel Settings for RemoteNationalPharmacyJmsServer in the right column of the console. Within the panel is Foreign Connection Factory Properties, where the JMS module will be further configured. For reference, see Figure 3-50.

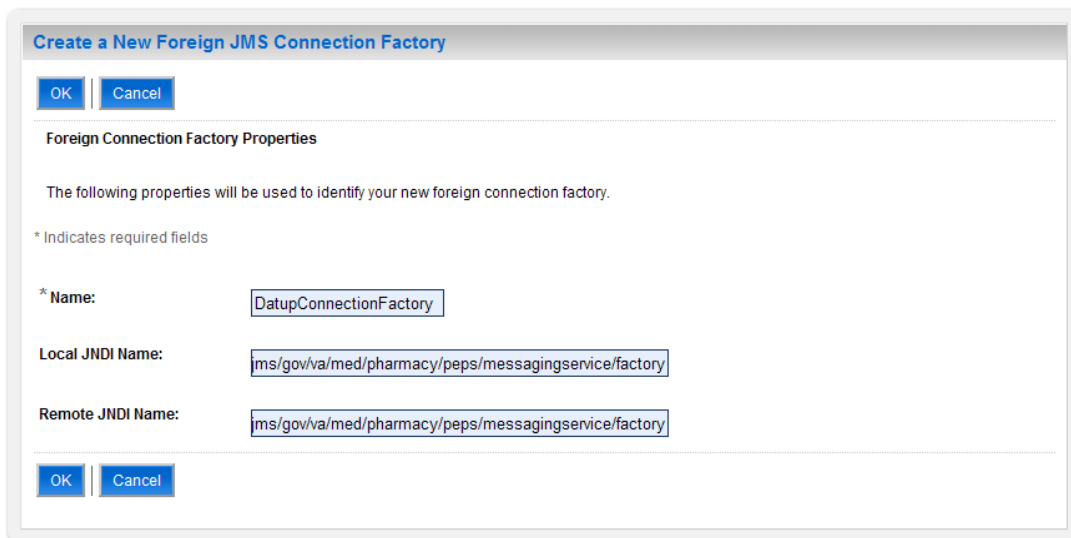


Figure 3-50. Foreign Connection Factory Properties

13. For Name, enter a unique name for the foreign connection factory. For example, `DatupConnectionFactory`.
14. For Local JNDI Name, enter:
`jms/gov/va/med/pharmacy/peps/messagingervice/factory`.
15. For Remote JNDI Name, enter:
`jms/gov/va/med/pharmacy/peps/messagingervice/factory`.
16. Click OK.
17. Within the Change Center panel in the left column of the WebLogic console, click **Activate Changes**. For reference, see Figure 3-51.

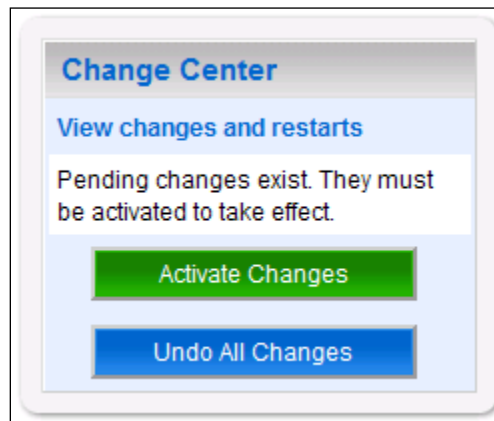


Figure 3-51. Activate Changes

3.4.6 Site Configuration Properties

In order to filter FDB drug-drug interactions replaced by custom VA drug-drug interactions, the `fdb_custom_ddimstrings` FDB-DIF table must be populated with a mapping between the FDB-DIF interaction to be replaced and the custom VA drug-drug interaction. One attribute of this mapping is a configurable category code, with a default of `FDB_ID`. A file, `gov.va.med.pharmacy.peps.siteConfig.properties`, can be placed within a folder on the Deployment Server's class path in order to override this default. Follow the BEA WebLogic documentation for adding folders to a server's class path. Each property is set via a key/value pair. For example, `fdb.id.category=FDB_ID`, where `fdb.id.category` is the key and `FDB_ID` is the value. Table 3-2 defines the optional property.

Table 3-2. Optional Site Configuration Properties

| Key | Definition | Sample |
|------------------------------|--|---------------------|
| <code>fdb.id.category</code> | Category code used within the <code>fdb_custom_ddimstrings</code> table for mapping FDB-DIF drug-drug interactions replaced by custom VA drug-drug interactions. | <code>FDB_ID</code> |

3.4.7 DATUP Configuration Properties

In order to use the DATUP component, a configuration file must be configured for each WebLogic deployment. The location of this file was configured in Section 3.4.2 and is by default `fdb_datup_configuration.properties`. This file is self documenting and contains the list of configurable properties for DATUP. See Appendix A for a sample version.

3.4.8 DATUP Cleanup Script

DATUP creates temporary zip files during the update process. A script has been provided in the `/scripts/datupcleanup.sh` file. This file provides a template to remove any files that DATUP creates during the update process. If the bash interpreter is not located at `/bin/bash` or the system's default temporary directory is not located at `/tmp`, the script file must be updated, comments in the example file show which lines to change.

To automate this process using the CRON scheduler, copy the file to the `/etc/cron.weekly/` directory for weekly execution. If you wish this script to run more often, it can be copied to the `/etc/cron.daily/` directory for daily execution. The script must be given execution permissions, so the command `chmod 755 datupcleanup.sh` must also be run on the command line.

3.4.9 Deployment

The following steps detail the deployment of the DATUP application at a local site. Prior to completing these steps, the WebLogic class path, the WebLogic database configurations, and the Deployment Server must be restarted to load the changed configuration. Please refer to Sections 3.4.1 and 0 for instructions concerning these configuration items. Complete the following steps to deploy DATUP:

1. Open and log into the WebLogic console. This is located at: `http://<Deployment Machine>:7001/console`.
2. Within the Domain Structure panel in the left column of the WebLogic console, click the Deployments node. For reference, see Figure 3-52.



Figure 3-52. Domain Structure

3. Within the Change Center panel in the left column of the WebLogic console, click Lock & Edit. For reference, see Figure 3-53.

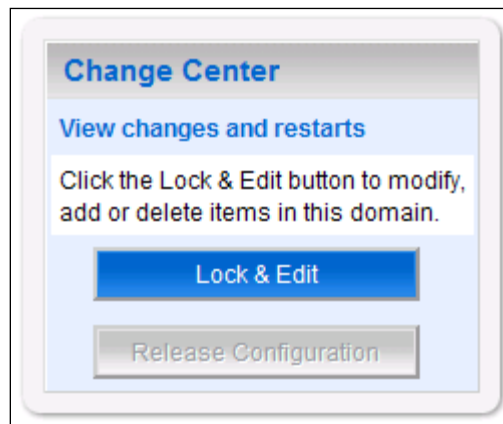


Figure 3-53. Change Center

4. Click Install found in the Deployments panel in the right column of the WebLogic console. For reference, see Figure 3-54.

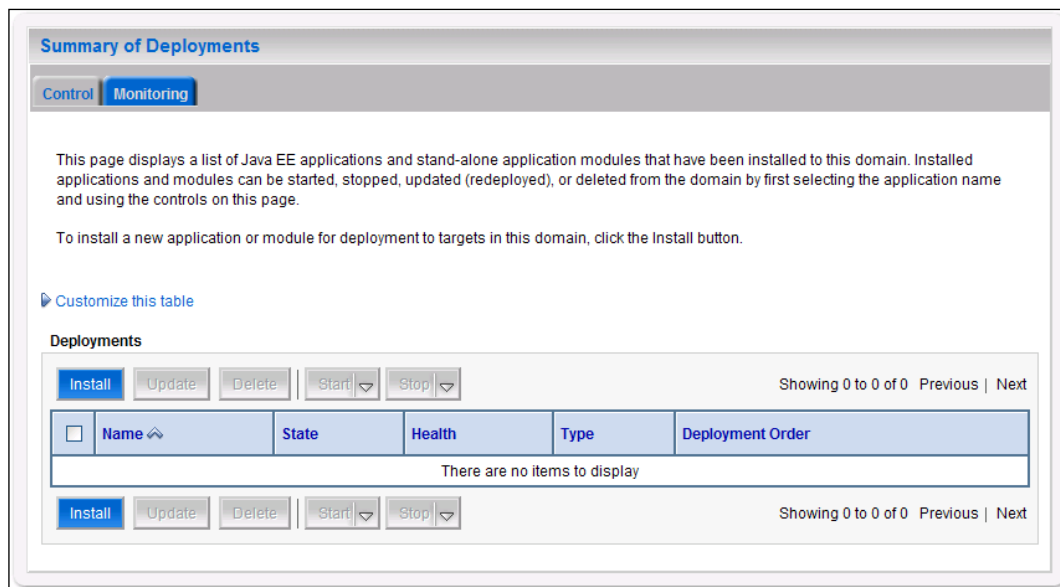


Figure 3-54. Deployments

5. WebLogic will now display the panel `Install Application Assistant` in the right column of the console, where the location of the DATUP deployment will be found. For reference, see Figure 3-55.

Install Application Assistant

Back Next Finish Cancel

Locate deployment to install and prepare for deployment

Select the file path that represents the application root directory, archive file, exploded archive directory, or application module descriptor that you want to install. You can also enter the path of the application directory or file in the Path field.

Note: Only valid file paths are displayed below. If you cannot find your deployment files, [upload your file\(s\)](#) and/or confirm that your application contains the required deployment descriptors.

Path: /opt/bea/domains/PRE/servers/AdminServer/upload

Recently Used Paths: /opt/bea/domains/PRE/servers/AdminServer/upload
/opt/bea/domains/PRE/servers/AdminServer/upload/PREv05

Current Location: pre05-test-app / opt / bea / domains / PRE / servers / AdminServer / upload

There are no files at the current location which are selectable. Choose a parent folder from the location links above or enter a new path.

Back Next Finish Cancel

Figure 3-55. Install Application Assistant

6. Select the DATUP deployment, select the `DATUP.local.2.0.00.001CM.ear` file. Replace the release number for the current release.
 - a) If the DATUP deployment has already been transferred to the Deployment Machine, navigate to the deployment file location using the links and file structure displayed within the Location panel within the Install Application Assistant in the right column of the console. For reference, see Figure 3-56.



Figure 3-56. Locate Deployment to Install and Prepare for Deployment

- b) If the DATUP deployment has not been transferred to the Deployment Machine:
 - (1) Click on the `upload your file(s)` link in the Install Application Assistant panel in the right section of the console. For reference, see Figure 3-56.
 - (2) Click the `Deployment Archive Browse` to see the `Choose file` dialogue used to select the `Deployment Archive`.

- (3) Click Next in the Upload a Deployment to the admin server panel in the right column of the WebLogic console to return to the Locate deployment to install and prepare for deployment panel within the Install Application Assistant. For reference, see Figure 3-57.

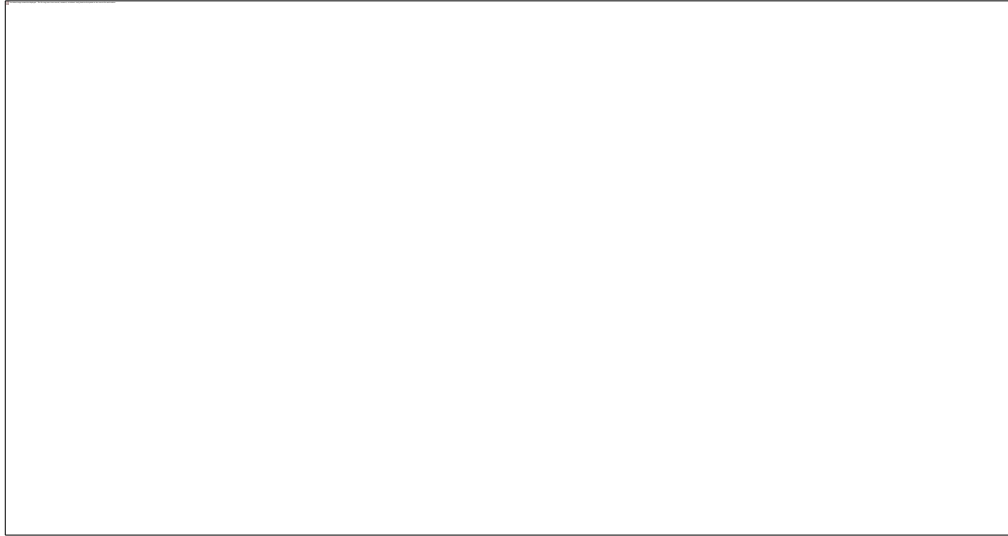


Figure 3-57. Upload a Deployment to the Admin Server

7. Once the DATUP deployment is located and selected, click Next.

8. WebLogic will now display the panel Choose targeting style within the Install Application Assistant in the right column of the console. Leave the default value selected, Install this deployment as an application, and click Next. For reference, see Figure 3-58.

The screenshot shows the 'Install Application Assistant' dialog box with the 'Choose targeting style' panel selected. The panel has a title bar 'Install Application Assistant' and navigation buttons 'Back', 'Next', 'Finish', and 'Cancel'. The main content area is titled 'Choose targeting style' and contains the following text: 'Targets are the servers, clusters, and virtual hosts on which this deployment will run. There are several ways you can target an application.' Below this, there are three radio button options: 'Install this deployment as an application' (selected), 'Install this deployment as a library', and 'Install this deployment as an application, but target the components individually'. Each option has a brief description. At the bottom, there are navigation buttons 'Back', 'Next', 'Finish', and 'Cancel'.

Figure 3-58. Choose Targeting Style

9. Within the Install Application Assistant in the right column of the console, WebLogic will now display the panel Select deployment targets, where the Deployment Server will be selected as the target in the next step. For reference, see Figure 3-59.

The screenshot shows the 'Install Application Assistant' dialog box with the 'Select deployment targets' panel selected. The panel has a title bar 'Install Application Assistant' and navigation buttons 'Back', 'Next', 'Finish', and 'Cancel'. The main content area is titled 'Select deployment targets' and contains the following text: 'Select the servers and/or clusters to which you want to deploy this application. (You can reconfigure deployment targets later).' Below this, there is a section titled 'Available targets for PREv05 :'. This section contains a table with two rows: 'AdminServer' and 'LocalPharmacyServer'. The 'LocalPharmacyServer' row has a checked checkbox. At the bottom, there are navigation buttons 'Back', 'Next', 'Finish', and 'Cancel'.

Figure 3-59. Select Deployment Targets

10. For the Target, select the Deployment Server. For example, LocalPharmacyServer.

11. Click Next.
12. Within the Install Application Assistant, WebLogic will now display the panel Optional Settings in the right column of the console, where the name of the deployment and the copy behavior are chosen. For reference, see Figure 3-60.

Install Application Assistant

Back Next Finish Cancel

Optional Settings

You can modify these settings or accept the defaults

General

What do you want to name this deployment?

Name:

Security

What security model do you want to use with this application?

☒ DD Only: Use only roles and policies that are defined in the deployment descriptors.

☐ Custom Roles: Use roles that are defined in the Administration Console; use policies that are defined in the deployment descriptor.

☐ Custom Roles and Policies: Use only roles and policies that are defined in the Administration Console.

☐ Advanced: Use a custom model that you have configured on the realm's configuration page.

Source accessibility

How should the source files be made accessible?

☒ Use the defaults defined by the deployment's targets

Recommended selection.

☒ Copy this application onto every target for me

During deployment, the files will be copied automatically to the managed servers to which the application is targeted.

☐ I will make the deployment accessible from the following location

Location:

Provide the location from where all targets will access this application's files. This is often a shared directory. You must ensure the application files exist in this location and that each target can reach the location.

Back Next Finish Cancel

Figure 3-60. Optional Settings

13. Enter the Name for the deployment. For example, Local DATUP.

14. Verify that the following default option for Security is selected:
DD Only: Use only roles and policies that are defined in the deployment descriptors.
15. Verify that the following default option for Source accessibility is selected:
Use the defaults defined by the deployment's targets.
16. Click Next.
17. Within the Install Application Assistant in the right column of the console WebLogic will now display the panel Review your choices and click Finish, which summarizes the steps completed above. For reference, see Figure 3-61.



Figure 3-61. Review Your Choices and Click Finish

18. Verify that the values match those entered in Steps 1 through 17 and click Finish.

19. WebLogic will now display the panel `Settings` for `Local DATUP`, in the right column of the console, where the values previously entered are available as well as a setting to change the deployment order. For reference, see Figure 3-62.

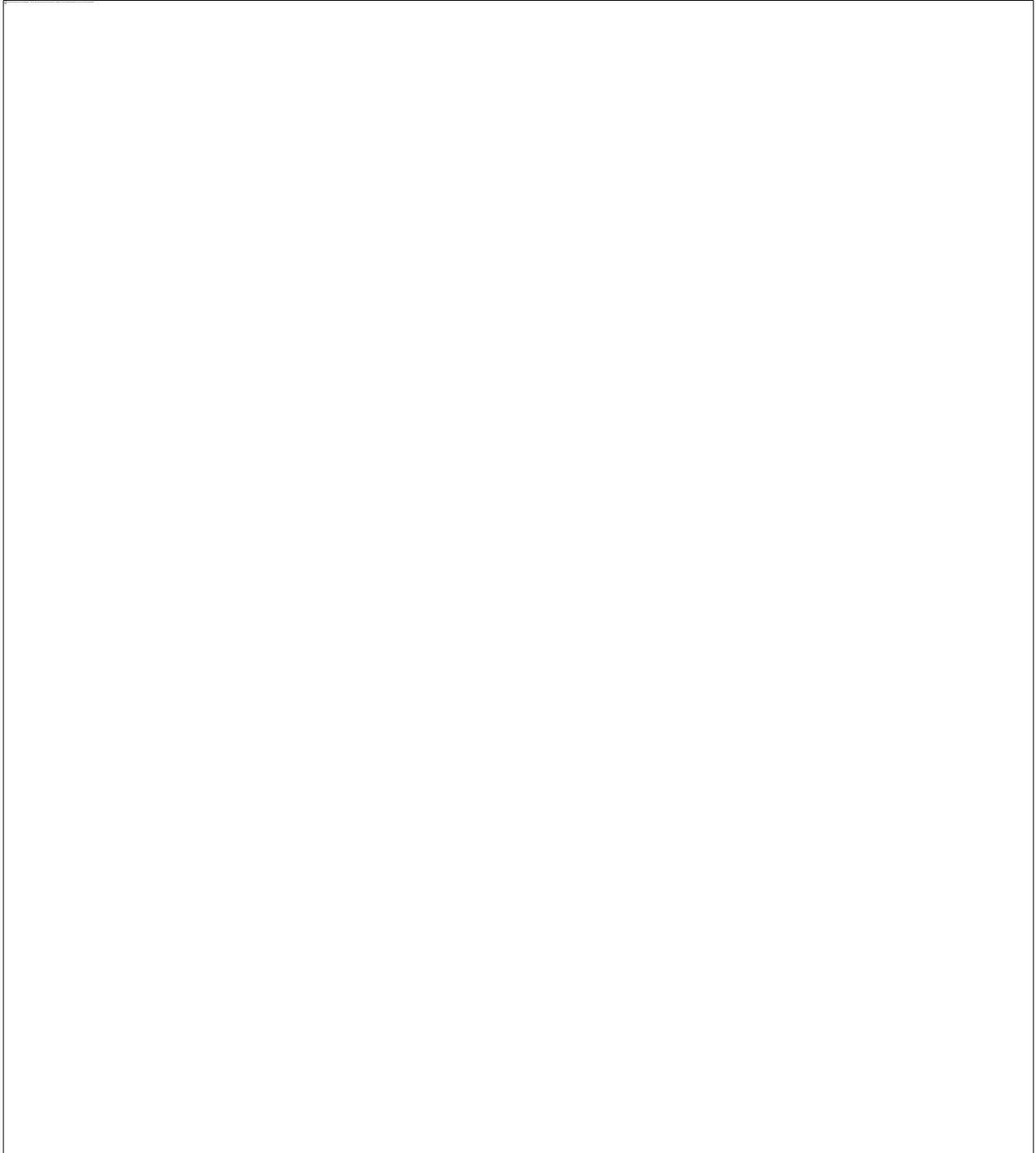


Figure 3-62. Settings for DATUP

20. Leave all the values as defaulted by WebLogic and click `Save`.

21. Within the Change Center panel in the left column of the WebLogic console, click `Activate Changes`. For reference, see Figure 3-63.

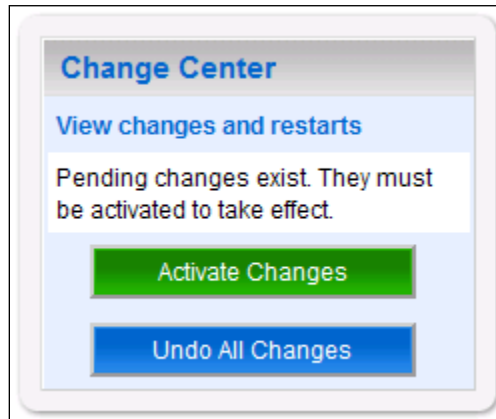


Figure 3-63. Activate Changes

22. Within the Domain Structure panel in the left column of the WebLogic console, click the `Deployments` node. For reference, see Figure 3-64.

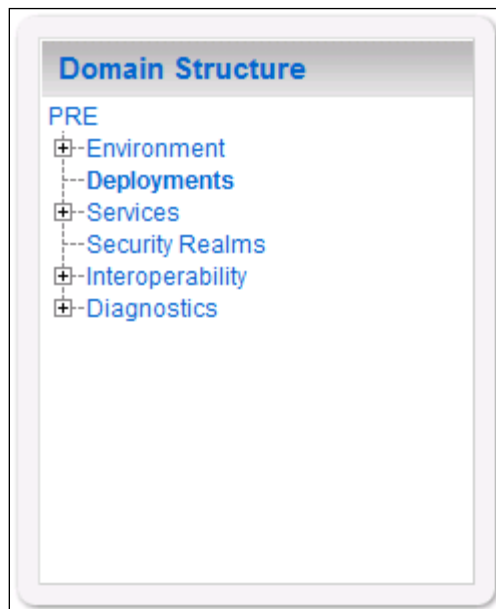


Figure 3-64. Domain Structure

23. WebLogic will now display the panel **Summary of Deployments** in the right column of the console, where all deployments for the WebLogic domain are listed. For reference, see Figure 3-65.

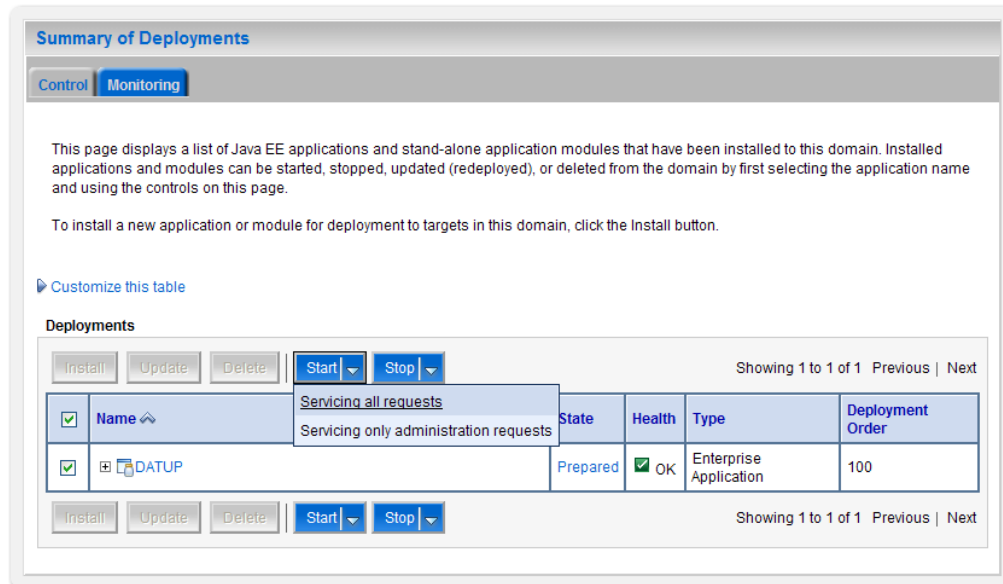


Figure 3-65. Summary of Deployments

24. Select the previously deployed DATUP deployment, click **Start**, and then select **Servicing all requests** from the drop-down list box.
25. WebLogic will now display the panel **Start Application Assistant** in the right column of the console for confirmation to start servicing requests. For reference, see Figure 3-66.

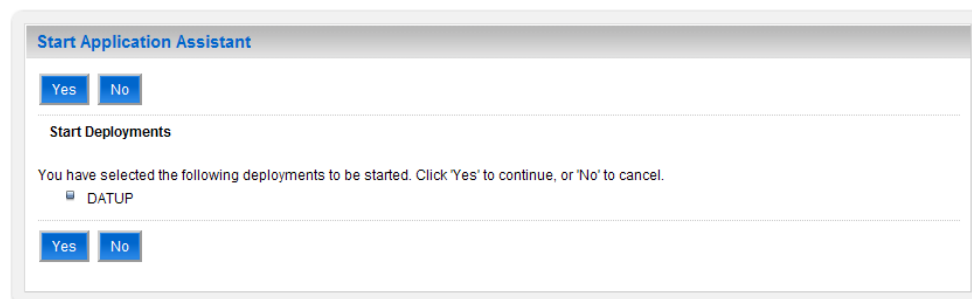


Figure 3-66. Start Application Assistant

26. Click **Yes** in the **Start Application Assistant** panel in the right column of the WebLogic console.

27. WebLogic now returns to the Summary of Deployments panel in the right column of the console. For reference, see Figure 3-67.

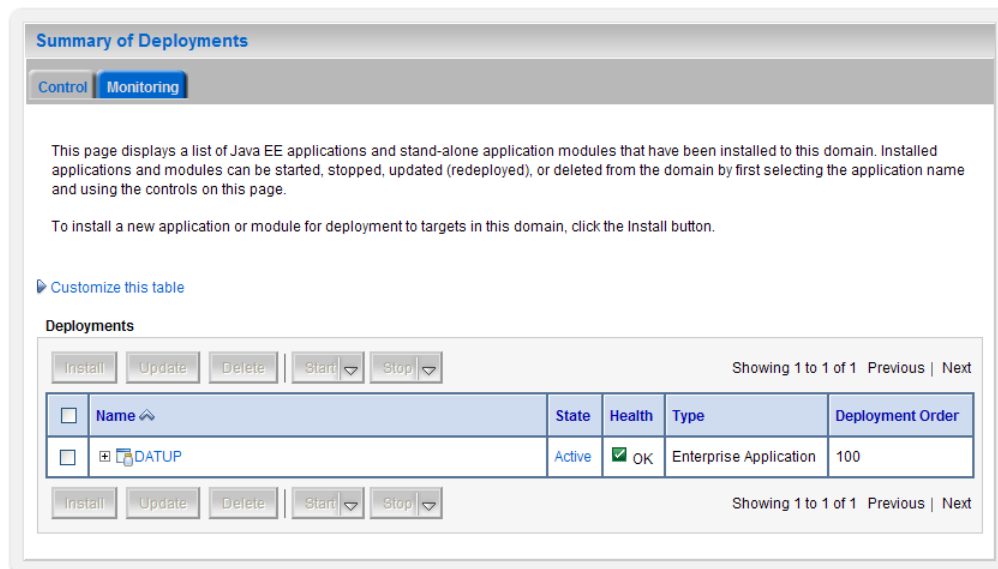


Figure 3-67. Summary of Deployments – DATUP Deployment Active

28. Verify that the State of the DATUP deployment is Active.

(This page included for two-sided copying.)

4 Upgrade Installation Instructions

The following instructions detail the steps required to perform an installation of a release for the DATUP software, when an existing release is already deployed at a local site. These steps assume a fresh installation has been completed, following the steps in Section 3.

4.1 Uninstall Previous Release

The following steps detail the un-installation of the DATUP application. Prior to completing these steps, the DATUP application must have been deployed following the steps in Section 3. Complete the following steps to un-deploy DATUP at a local site:

1. Open and log into the WebLogic console. This is located at: `http://<Deployment Machine>:7001/console`.
2. Within the Domain Structure panel in the left column of the WebLogic console, click the Deployments node. For reference, see Figure 4-1.

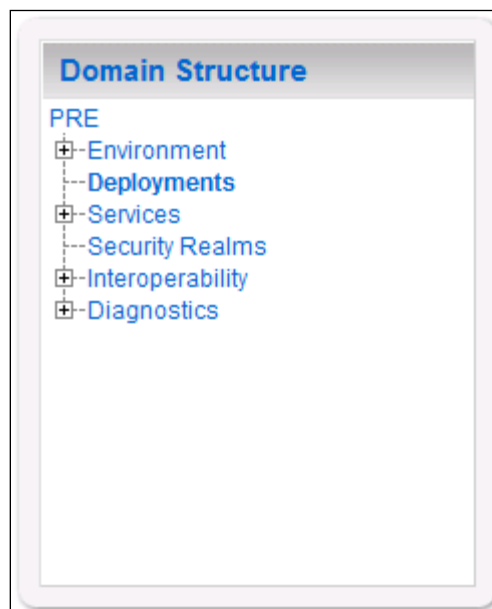


Figure 4-1. Domain Structure

3. Within the Change Center panel in the left column of the WebLogic console, click Lock & Edit. For reference, see Figure 4-2.

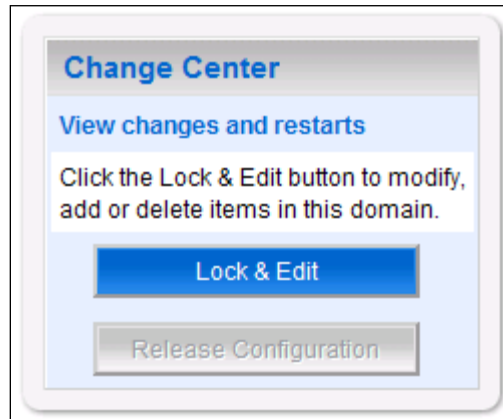


Figure 4-2. Change Center

4. WebLogic will now display the panel Summary of Deployments in the right column of the console, where all deployments for the WebLogic domain are listed. For reference, see Figure 4-3.

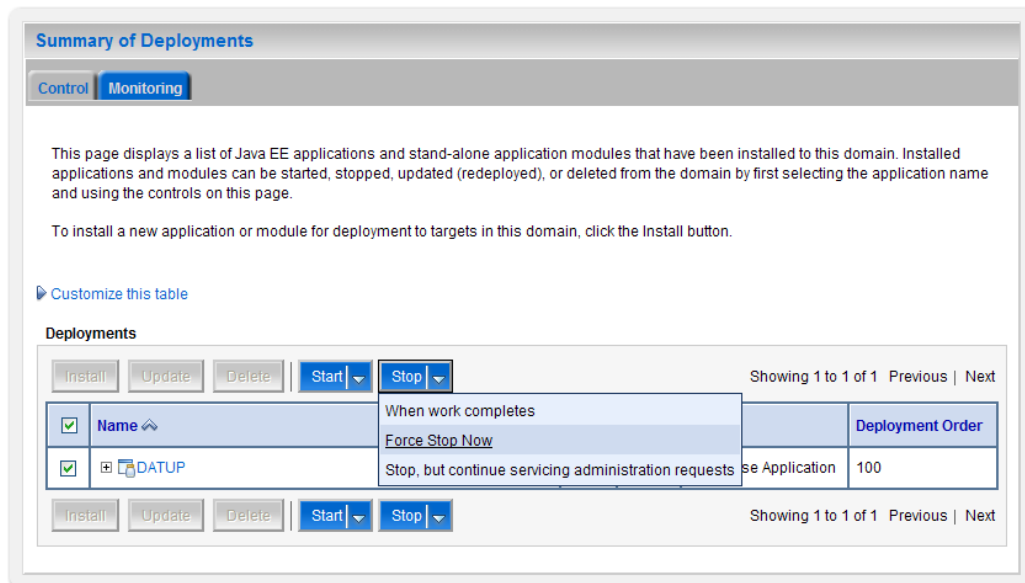


Figure 4-3. Summary of Deployments – Stopping DATUP

5. Select the previously deployed DATUP deployment, click Stop, and then select Force Stop Now from the drop-down list box.

- WebLogic will now display the panel **Force Stop Application Assistant** in the right column of the console for confirmation to start servicing requests. For reference, see Figure 4-4.

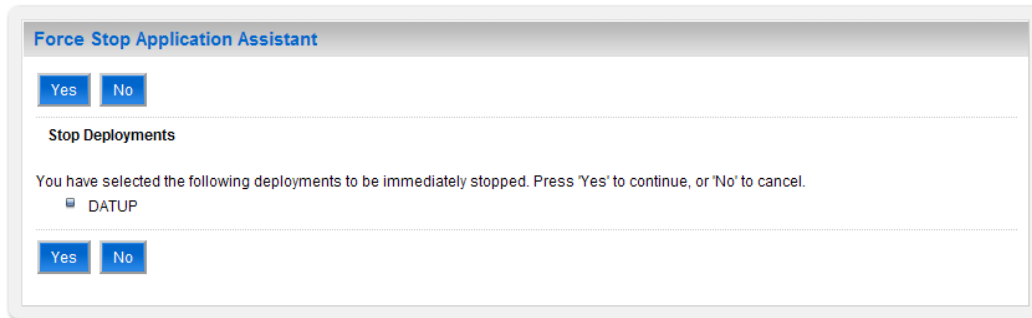


Figure 4-4. Force Stop Application Assistant

- Click **Yes** in the **Force Stop Application Assistant** panel in the right column of the WebLogic console.
- WebLogic now returns to the **Summary of Deployments** panel in the right column of the console. For reference, see Figure 4-5.

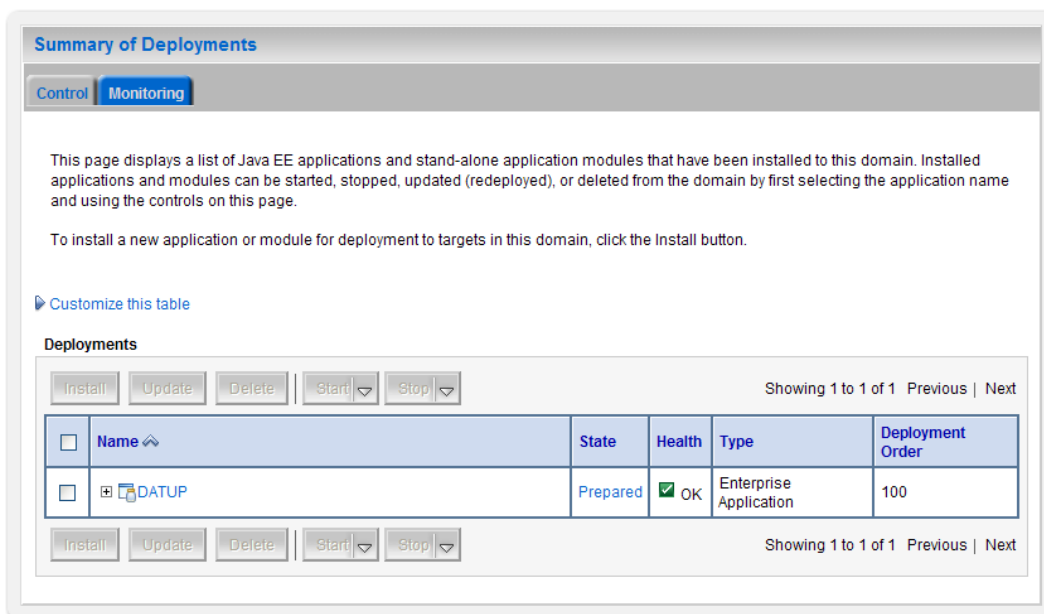


Figure 4-5. Summary of Deployments – DATUP Deployment Prepared

- Verify that the State of the DATUP deployment is **Prepared**.
- Select the previously deployed DATUP deployment, and then click **Delete**.

11. WebLogic will now display the panel Delete Application Assistant in the right column of the console for confirmation to start servicing requests. For reference, see Figure 4-6.

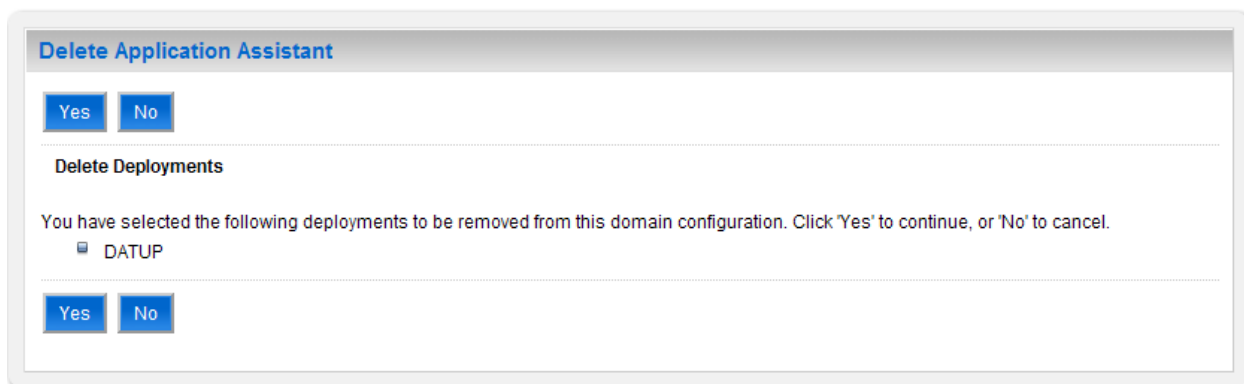


Figure 4-6. Delete Application Assistant

12. Click Yes in the Delete Application Assistant panel in the right column of the WebLogic console.
13. WebLogic now returns to the Summary of Deployments panel in the right column of the console. For reference, see Figure 4-7.

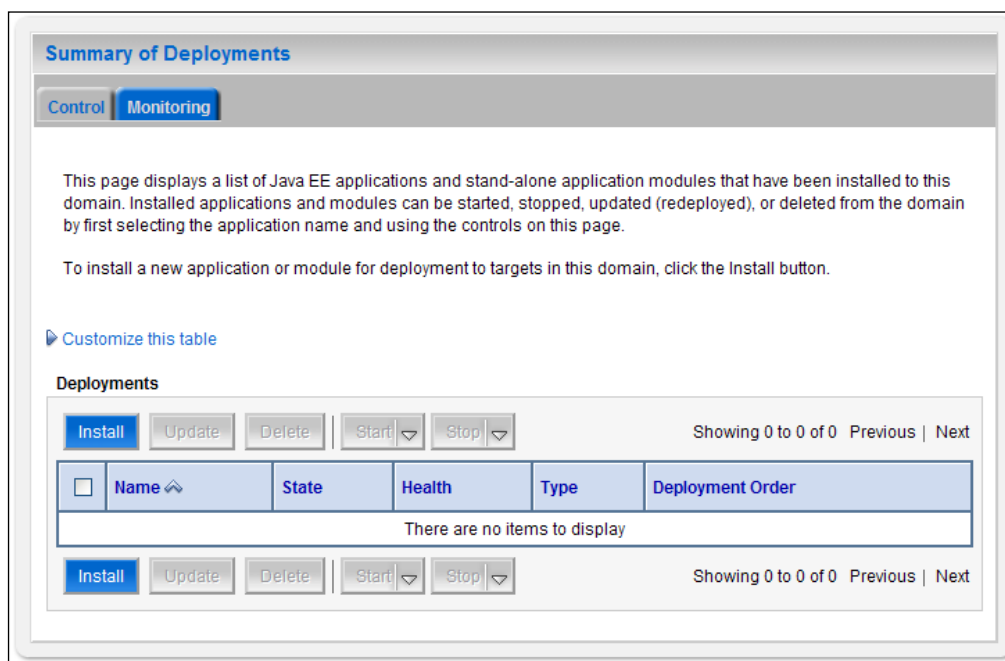


Figure 4-7. Summary of Deployments – DATUP Deployment Deleted

14. Verify that the DATUP deployment is deleted and no longer present.

15. Within the Change Center panel in the left column of the WebLogic console, click **Activate Changes**. For reference, see Figure 4-8.

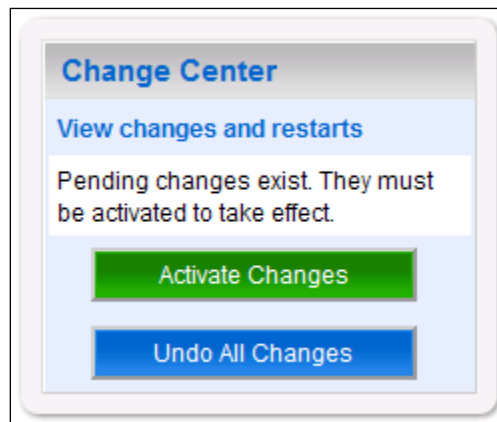


Figure 4-8. Activate Changes

4.2 Deploy New Release

To deploy the new release, follow the same deployment steps found in Section 3.4.9.

(This page included for two-sided copying.)

5 System Verification

The following section will verify that the DATUP system is up and running at a local site.

5.1 Verification

To verify that a DATUP installation is up and running at a local site, navigate a web-browser to the logs directory on the server, for example, <http://datap-1-1/logs/LocalDatupServer/logs>.

Verify that the `server.log` file has an entry indicating the next scheduled run time for the DATUP application. The `server.log` entry looks like:

```
DEBUG
[gov.va.med.pharmacy.peps.updater.common.utility.DifUpdateScheduler:scheduleNextTimer]
Next scheduled DIF update time: Thu, 08/26/2010, 02:45:00 PM, CDT
```

This line indicates that the system is running.

(This page included for two-sided copying.)

Appendix A

Local DATUP Configuration

This appendix provides a configuration file example for a local site based on the baseline `fdb_datup_configuration.properties` file located on the delivery CD in the `/configuration` directory.

Example DATUP Configuration Properties for a Local Site

```
#####
#----- Scheduler -----
#####

#####
# Scheduled nightly update time (military time).
#
# For example, "0230" schedules the nightly update
# for 2:30 am.
#
# *This parameter applies to National and Local.
#####
scheduled.time=1405

#####
#----- FTP Server -----
#####

#####
# FTP server hostname
#
# Specify the anonymous FTP server hostname.
#
# *This parameter applies to National and Local.
#####
ftp.hostname=10.3.29.201

#####
# FTP server port number
#
# Specify the anonymous FTP server port number.
#
# *This parameter applies to National and Local.
#####
ftp.port=21

#####
# FTP server username/password.
#
# Specify the anonymous account username/password.
#
# *These parameters apply to National and Local.
#####
ftp.username=PECS
ftp.password=

#####
# FTP server working directory
#
# Specify the FTP working directory, relative to
# the FTP root directory.
#
# *This parameter applies to National and Local.
```

```

#####
ftp.directory.working=pharmacy_uft

#####
# Pending FDB-DIF update storage directory.
#
# Specify the pending directory, relative to the
# working directory, to the location where FDB-DIF
# full, incremental, and custom ZIP files will be
# placed for processing.
#
# *This parameter applies to National.
#####
ftp.directory.pending=fdb_dif

#####
#----- FDB DIF -----
#####

#####
# Number of random FDB-DIF verification tests
#
# Specify the number of random FDB-DIF verification
# tests to run. 10 is a reasonable number. However,
# do not specify a large number as it will cause an
# unacceptable delay for processing new VistA order
# checks during that time.
#
# *This parameter applies to National and Local.
#####
fdb.verification.test.count=5

#####
#----- Email Server -----
#####

#####
# Email server hostname
#
# *This parameter applies to National and Local.
#####
email.hostname=SMTP.VA.GOV

#####
# Email sender name
#
# For example, "noreply@va.gov".
#
# *This parameter applies to National and Local.
#####
email.sender=noreply_national_DATUP1.1_sqa@med.va.gov

#####
# Email username/password
#
# May be necessary to relay email.
#
# *These parameters apply to National and Local.
#####
email.username=
email.password=

#####
# Email list for success notifications
#
# Include individuals that should be notified about
# successful FDB/FDB-Custom updates.
#
# *This parameter applies to National and Local.
#####
email.list.success=SDDPREArch@med.va.gov

```

```

#####
# Email list for failure notifications
#
# Include individuals that should be notified about
# failed FDB/FDB-Custom updates.
#
# *This parameter applies to National and Local.
#####
#email.list.failure=Sachin.Sharma@va.gov
email.list.failure=SDDPREArch@med.va.gov

#####
# Email list for available update notifications
#
# Include individuals that should be notified about
# available FDB/FDB-Custom updates once they are
# applied and tested and National. This list should
# include the local site managers.
#
# *This parameter applies to National.
#####
email.list.update.available=SDDPREArch@med.va.gov

#####
#----- Locality -----
#####

#####
# Regional Data Center (RDC) name
#
# Specify the name of the RDC or leave blank if
# this installation is not part of a RDC.
#
# *This parameter applies to Local.
#####
locality.rdc.name=

#####
# Site number(s)
#
# Specify the site number(s) for this installation.
# If more than one site is associated with this
# installation, separate the site numbers with a
# comma (e.g., 423,512,211).
#
# *This parameter applies to Local.
#####
locality.site.number=

#####
# Number of retained FDB-DIF incremental archives
#
# Due to potential site outages, it is necessary
# to retain a certain number of FDB-DIF archives.
#
# *This parameter applies to National.
#####
fdb.retention=20

#####
# Number of statements to batch before commit
#
# Specify the number of statements to batch before
# a commit to the database. This value is database
# vendor and JDBC driver dependent. A reasonable
# batch size is 500. However, tests show that Cache
# may throw system errors with a batch size greater
# than 200.
#
# Specify a batch size of 0 to disable batching. A

```

```

# single commit will be issued at the end of the
# incremental update.
#
# *This parameter applies to National and Local.
#####
fdb.batch.commit.size=0

#####
#----- IMAGE PROCESSING-----
#This parameter specifies if Images will be processed
#or not.
#The Parameter is "TRUE" for National and "FALSE" for
#LOCAL by default.
#####
image.processing.national=TRUE
image.processing.local=FALSE
#####
# The location where the image files will be stored
#
#
# *This parameter applies to National only
#####
image.directory.national=/tmp/imaging/
#####
# The location where the image files will be stored
#
#
# *This parameter applies to local only
#####
image.directory.local=/tmp/imaging/
#####
# The email to send to names or group for Image
#Processing.
#This parameter can be blank
#
# *This parameter applies to National only
#####
image.email.sendto.national=
#image.email.sendto.national=SDDPREArch@med.va.gov
#image.email.sendto.national=sachin.sharma@va.gov
#####
# The email to send to names or group for Image
#Processing
#This parameter can be blank
#This parameter applies to local only
#####
image.email.sendto.local=SDDPREArch@med.va.gov

```

Appendix B

Combined DATUP / PECS Architecture

This appendix provides the combined DATUP / PECS architecture diagram for reference. The combined logical system components are:

1. DATUP – Implements the FDB-DIF update business logic.
2. Scheduler – Background process for scheduling DATUP.
3. WebLogic – Application server environment.
4. Configuration File – Defines the DATUP configuration settings.
5. Email Templates – Templated emails for notifications sent to National/Local Managers.
6. Anonymous FTP Server – FTP Server that hosts the FDB-DIF update archives.
7. Email Server – Email relay server.
8. PECS – Implements the FDB-Custom drug business logic.
9. CT Staging Database – Stores PECS FDB-Custom modifications.
10. DATUP Database – Stores DATUP site update history.
11. FDB-DIF Database – Stores the FDB-DIF drug database.
12. Legacy VistA – Existing VistA server.

Figure B-1 illustrates the logical system components for the National and Local environments. The National components are responsible for verifying and publishing FDB-DIF and FDB-Custom updates to the Anonymous FTP Server. The Local components then consume and apply the verified updates in an automated manner.

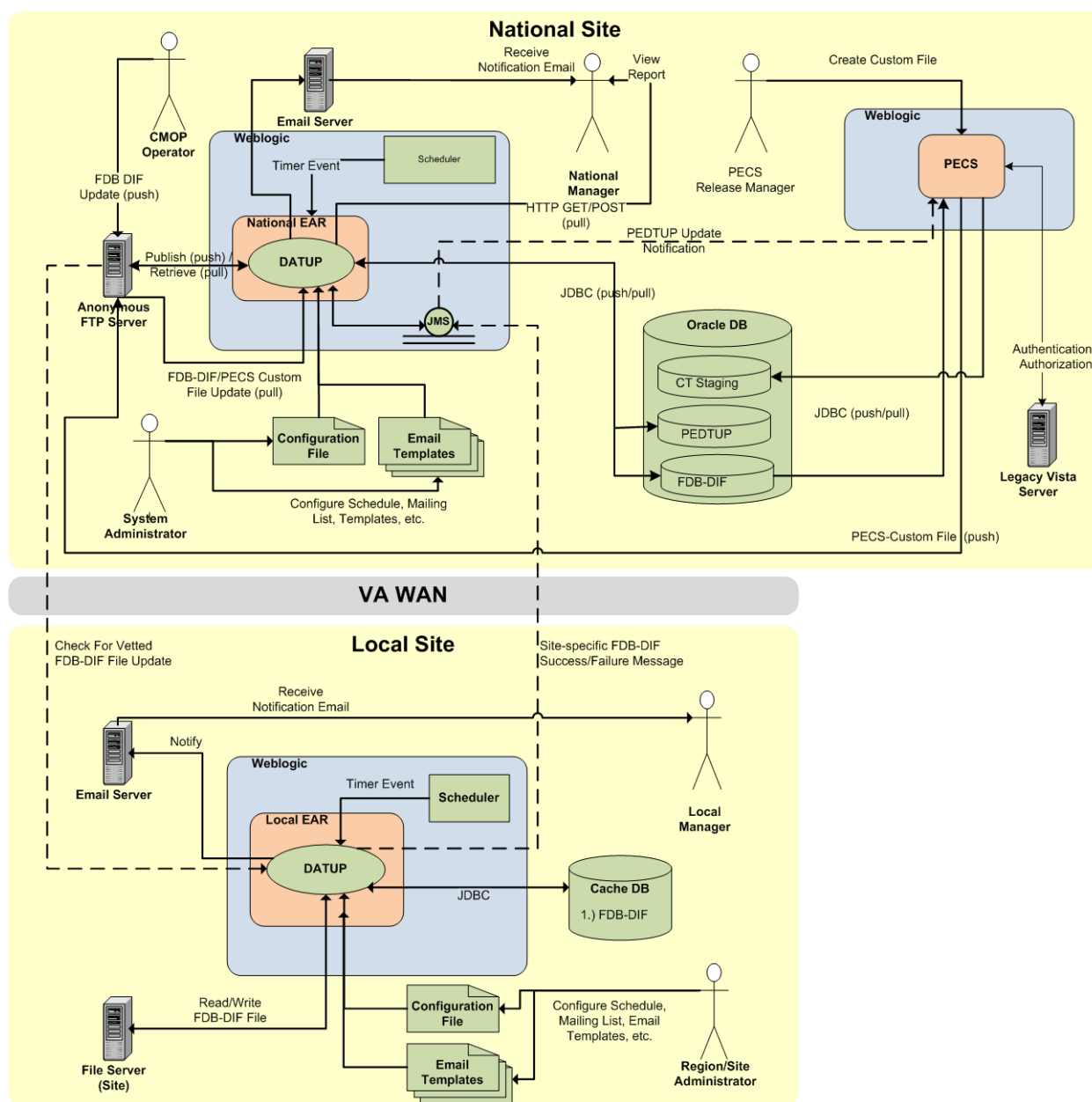


Figure B-1: Combined DATUP/PECS Architecture Diagram