Installation Guide

e-Billing Manager

Microsoft Windows 2003® Operating System
and the BEA WebLogic® Server

V4.5.2
Data Published: 03.16.2005

edocs, Inc., One Apple Hill Dr., Natick, MA 01760

Printed in the USA.

Table of Contents

Preface

About This Guide

This guide is intended for system administrators and other IT professionals and describes how to install and, configure the third-party platforms that support the EBM production environment and deploy EBM J2EE web applications. See "*EBM System Requirements*" on page 12 for details on the platforms this guide is intended for.

It assumes in-depth understanding of and practical experience with system administrator responsibilities, including:

Operating System Administration Requirements

- Start up and shut down the system
- Log in and out of the system
- Determine software patch/pack levels
- Install software & patches/packs
- Navigate the file system
- Manipulate text files
- Create files and directories
- Change permissions of files and directories
- Use basic network commands
- Transfer files with FTP
- Monitor processes & system resource usage
- Perform system backups and recovery
- Implement system security

Database Administration Requirements

- Install and configure your database server
- Start and stop your database server and database instances
- Use administrative tools

- Manage users, privileges, and resources
- Create an operational database
- Manage database files
- Manage tables and indexes
- Back up and restore databases
- Monitor database performance

Application Server Administration Requirements

- Install and configure your application server
- Start and stop your application server
- Use administrative tools
- Manage users, privileges, and resources
- Configure Java resources
- Package and deploy web applications
- Monitor application server performance

This guide does *not* describe general Windows system administration. See the appropriate Windows user documentation.

If you are unfamiliar with any of these tasks, please consult the related documentation for your system requirements.

Related Documentation

A PDF version of this guide is also available.

Online	How to Access
A PDF of this guide	A PDF of this guide is available on the product CD-ROM.

This guide is part of the EBM documentation set. For more information about using EBM, see the following guides:

EBM SDK	How to customize J2EE web applications for deployment with EBM.
e-Billing Manager Data Definition Guide	How to create Data Definition Files (DDFs) for use in indexing your application and extracting data for live presentment.
e-Billing Manager Presentation Design Guide	How to create Application Logic Files (ALFs) to present statement data for dynamic online display.

EBM SDK	How to customize J2EE web applications for deployment with EBM.
EBM Administration Guide	How to set up and run a live EBM application in a J2EE environment.

Obtaining edocs Software and Documentation

You can download edocs software and documentation directly from Customer Central at https://support.edocs.com. After you log in, click the Downloads button on the left. When the next page appears, a table displays all of the available downloads. To search for specific items, select the Version and/or Category and click the Search Downloads button. If you download software, an email from edocs Technical Support will automatically be sent to you (the registered owner) with your license key information.

If you received an edocs product installation CD, load it on your system and navigate from its root directory to the folder where the software installer resides for your operating system. You can run the installer from that location, or you can copy it to your file system and run it from there. The product documentation included with your CD is in the Documentation folder located in the root directory. The license key information for the products on the CD is included with the package materials shipped with the CD.

If You Need Help

Technical Support is available to customers who have an active maintenance and support contract with edocs. Technical Support engineers can help you install, configure, and maintain your edocs application.

This guide contains general troubleshooting guidelines intended to empower you to resolve problems on your own. If you are still unable to identify and correct an issue, contact Technical Support for assistance.

Information to provide

Before contacting edocs Technical Support, try resolving the problem yourself using the information provided in this guide. If you cannot resolve the issue on your own, be sure to gather the following information and have it handy when you contact technical support. This enables your edocs support engineer to more quickly assess your problem and get you back up and running more quickly.

Please be prepared to provide Technical Support the following information:

Contact information:

- Your name and role in your organization.
- Your company’s name
- Your phone number and best times to call you

- Your e-mail address

Product and platform:

- In which edocs product did the problem occur?
- What version of the product do you have?
- What is your operating system version? RDBMS? Other platform information?

Specific details about your problem:

- Did your system crash or hang?
- What system activity was taking place when the problem occurred?
- Did the system generate a screen error message? If so, please send us that message. (Type the error text or press the Print Screen button and paste the screen into your email.)
- Did the system write information to a log? If so, please send us that file. For more information, see the *EBM Troubleshooting Guide*.
- How did the system respond to the error?
- What steps have you taken to attempt to resolve the problem?
- What other information would we need to have (supporting data files, steps we'd need to take) to replicate the problem or error?
- **Problem severity:**
- Clearly communicate the impact of the case (Severity I, II, III, IV) as well as the Priority (Urgent, High, Medium, Low, No Rush).
- Specify whether the problem occurred in a production or test environment.

Contacting edocs Technical Support

You can contact Technical Support online, by email, or by telephone.

edocs provides global Technical Support services from the following Support Centers:

US Support Center
Natick, MA
Mon-Fri 8:30am – 8:00pm US EST
Telephone: 508-652-8400

Europe Support Center
London, United Kingdom
Mon-Fri 9:00am – 5:00 GMT
Telephone: +44 20 8956 2673

Asia Pac Rim Support Center
Melbourne, Australia
Mon-Fri 9:00am – 5:00pm AU
Telephone: +61 3 9909 7301

Customer Central
https://support.edocs.com

Email Support
mailto:support@edocs.com

Escalation process

edocs managerial escalation ensures that critical problems are properly managed through resolution including aligning proper resources and providing notification and frequent status reports to the client.

edocs escalation process has two tiers:

1. **Technical Escalation -** edocs technical escalation chain ensures access to the right technical resources to determine the best course of action.
2. **Managerial Escalation -** All severity 1 cases are immediately brought to the attention of the Technical Support Manager, who can align the necessary resources for resolution. Our escalation process ensures that critical problems are properly managed to resolution, and that clients as well as edocs executive management receive notification and frequent status reports.

By separating their tasks, the technical resources remain 100% focused on resolving the problem while the Support Manager handles communication and status.

To escalate your case, ask the Technical Support Engineer to:

1. Raise the severity level classification
2. Put you in contact with the Technical Support Escalation Manager
3. Request that the Director of Technical Support arrange a conference call with the Vice President of Services
4. Contact VP of Services directly if you are still in need of more immediate assistance.

1

Getting Started

Before You Install; Preparing Your Platform

Before installing EBM, verify that your platform is ready:

- Install and test required hardware and software for your platform.
- Define required user and group permissions for your database server and application server.
- Start and test your database server. For details, see your server documentation.
- Start and test your application server. For details, see your server documentation.
- For distributed environments, make sure you have any required database client software installed on your application server and any other client machines of your database server.

Overview of the Installation Process

The process of installing and setting up e-Billing Manager includes the following steps:

1. Installing edocs Platform Services and EBM Consumer on your database and application servers using InstallAnywhere. You can install these individually, running InstallAnywhere twice (once to install each feature), or together using the Custom install feature.

2. Installing edocs Tools on a Windows 2000 machine. Run InstallAnywhere here and exclusively install Tools.

3. Configuring the database server.

4. Configuring the application server.

Follow the chapters in this guide in sequence, consulting your third-party documentation as needed.

Once you successfully install EBM and configure your database and application servers, you can customize and deploy your J2EE application.

Configuring Your Database Server

Configuring your database server requires you to:

1. Define database server environment variables.
2. Create and configure the EBM database with *DBConfigTool.jar*.
3. Connect to your EBM database before configuring your application server.

Configuring Your Application Server

Configuring your application server requires you to:

1. Define application server environment variables.
2. Configure JDBC resources for EBM on your application server.
3. Configure JMS resources for EBM on your application server.
4. Install Windows Services for your application server and the EBM Scheduler.

Customizing and Deploying J2EE Applications

After installing EBM and configuring your database and application servers, you can:

1. Customize your J2EE web application(s) for EBM.
2. Deploy J2EE web applications for EBM.
3. Deploy your custom J2EE web application.

EBM System Requirements

edocs' Platform Services and EBM Consumer (Windows/SQL Server/WebLogic)

This guide assumes you are installing EBM on a Windows 2003 Server operating system, SQL Server 2000 database, and WebLogic application server.

OPERATING SYSTEM

- Microsoft Windows 2003 Server

HARDWARE

- CD-ROM
- Disk space (database) 5 GB
- Disk space (for EBM software) 500 MB (in addition to the space required for the application server)
- Intel Pentium or compatible processor at 850 MHz or higher
- Swap space 512 MB per CPU (1 GB recommended)
- RAM 512 MB per CPU (1 GB recommended)

JAVA/C++

- Sun Java 2 SDK Standard Edition 1.4.1 (version shipped with WebLogic 8)

SUPPORTED DATABASE SERVER

- Microsoft SQL Server 2000 SP3

SUPPORTED APPLICATION SERVERS

- BEA WebLogic Server 8.1 SP3

SUPPORTED BROWSERS

- Netscape Navigator 6.2
- Microsoft Internet Explorer 5.5 SP2 or 6 (on networked PC)
- Firefox 1.0

edocs’ Tools (Windows)

OPERATING SYSTEM

- Microsoft Windows 2000/Server SP4 or a later version of Windows

2

Installing EBM for Windows

This chapter provides a step-by-step guide to installing EBM with InstallAnywhere. It assumes that you have an in-depth understanding of and practical experience with administrating your operating system. Consult your system documentation as necessary.

Installing EBM

InstallAnywhere is a graphical cross-platform wizard that lets you install EBM in a distributed environment:

- **edocs Platform Services** – Install on all database and application servers.
- **EBM Consumer** – Install on all application servers.
- **edocs Tools** – Install on a Windows machine accessible to the servers on your network.

edocs recommends that you install and configure EBM in the same top-level directory structure, first on the **database server**, then on the **application servers**.

edocs Platform Services and EBM Consumer

You must install both edocs Platform Services and EBM Consumer using InstallAnywhere. You can install them individually or create a custom install to install at once.

To install edocs Platform Services and/or EBM Consumer with InstallAnywhere:

1. Obtain and locate the InstallAnywhere installer as described in "*Obtaining edocs Software and Documentation*" on page 7.
2. Launch InstallAnywhere by double-clicking the **`TBMins.exe`** icon.
3. INTRODUCTION: InstallAnywhere recommends that you quit all programs before installing.
4. LICENSE AGREEMENT: Review the License Agreement and click "**`I accept the terms of the License Agreement`**" to accept the terms.

5. ENTER SERIAL NUMBER provided when you purchased EBM. If lost, contact edocs Technical Support at http://support.edocs.com/

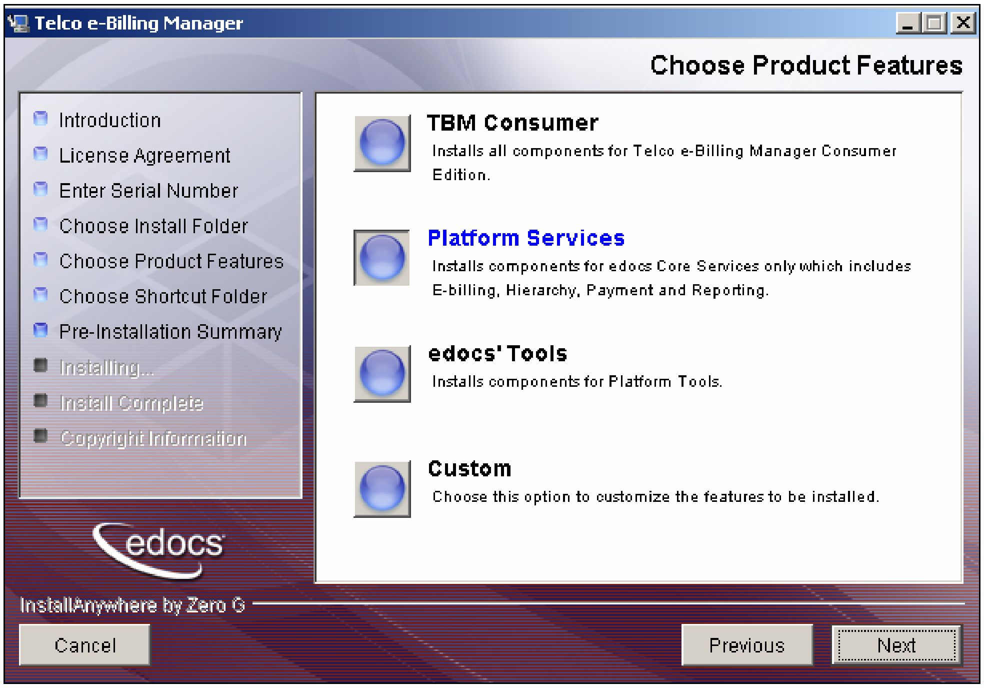
6. CHOOSE INSTALL FOLDER: Click `Next` to accept the default or specify another directory. edocs recommends that you install and configure EBM in the same top-level directory structure, first on the database server, then on the application server.

7. CHOOSE PRODUCT FEATURES: Choose which feature you want to install first, or click **`Custom`** and choose to install EBM Consumer and edocs Platform services at the same time.

   EBM Consumer - Installs all components for e-Billing Manager Consumer Edition.

   Platform Services (default) - Installs the components for edocs Core Services only, including e-Billing, Hierarchy, Payment, and Reporting.

   edocs'Tools - Installs DefTool and Composer (on a Windows machine only)



Choose Product Features

TBM Consumer
Installs all components for Telco e-Billing Manager Consumer Edition.

Platform Services
Installs components for edocs Core Services only which includes E-billing, Hierarchy, Payment and Reporting.

edocs' Tools
Installs components for Platform Tools.

Custom
Choose this option to customize the features to be installed.

8. CHOOSE SHORTCUT FOLDER: Click `Next` to install EBM in the edocs program group.

9. PRE-INSTALLATION SUMMARY: Review the screen to confirm your product and version, install folder, product components, and disk space required and available and click **`Install.`**

   InstallAnywhere sets up a directory hierarchy on each server and copies files to the appropriate directories.

10. INSTALL COMPLETE: If installation is successful, you see a congratulatory message. Click **`Done`**.

11. Repeat the installation for other EBM servers on your network as necessary.

Tools

To install edocs' Tools on a Windows machine on your network:

1. Obtain and locate the InstallAnywhere installer as described in "Obtaining edocs Software and Documentation" on page 7.
2. Launch InstallAnywhere by double-clicking the **`TBMins.exe`** icon.
3. Follow the steps in the procedure above for installing EBM, selecting the edocs' Tools feature to install.

Payment

The EBM installation procedure automatically installs and configures Payment with Platform Services.

By default, Payment is installed in: *C:\edocs\TBM\Payment*.

(The default installation path for Payment is also referred to as PAYMENT_HOME, or *EBM_HOME/Payment*)

The EBM Directory Structure

The EBM home directory contains all the files you need to create and configure the EBM production database. When you install EBM components, InstallAnywhere prompts you to specify a destination directory. You can use the default or specify another directory.

The default EBM installation directories are:

- EBM : *\edocs\TBM\*
- Platform Services: *\edocs\TBM\estatement*
- Edocs Tools: *\edocs\TBM\estatement\bin*

Tip edocs recommends that you install EBM in the same top-level directory on both the database server and the application servers.

Where to Find Database Components

edocs\TBM\estatement\db contains platform-specific subdirectories for database creation and configuration.

Where to Find Application Server Components

edocs\TBM\estatement\J2EEApps contains platform-specific subdirectories for edocs J2EE and web applications to be deployed to your application server. Be sure to deploy the correct version for your platform.

Where to Find Input and Output Data

edocs\TBM\estatement\AppProfiles stores information on each new EBM application created in the Command Center. *edocs\TBM\estatement\Input* is the default input directory used by each Command Center job. *edocs\TBM\estatement\Data* stores data processed by the EBM Command Center. *edocs\TBM\estatement\Output* stores the output of jobs.

An additional directory, *edocs/TBM/estatement/Store*, appears when the first Command Center job runs. The *Store* directory holds temporary files created during job run time. When the job completes, EBM automatically cleans up these temporary files.

3

Configuring Your Database Server

Overview

This chapter assumes in-depth understanding of and practical experience with database administration. Consult your database documentation as necessary. For distributed environments, make sure you have any required database client software installed on your application server and any other client machines of your database server.

edocs recommends that you install and configure EBM in the same top-level directory structure, first on the database server, then the application server.

This chapter provides instructions for configuring your database server to support a **new** EBM database. It includes:

- Windows environment variables for your database server
- Using database partitioning with EBM

Caution: The installation and configuration examples shown in this guide use default EBM pathnames, privileges, and permissions. If you choose not to accept the default values, make sure your values are consistent on all servers across your installation of EBM.

SQL Server Database Server Environment Variables

VARIABLE	DESCRIPTION	WINDOWS	CUSTOM
ADMIN_NAME	Admin name	**sa**	
ADMIN_PASSWD	Admin password	Leave blank	
DB_USERNAME	Database user name	**edx_dba**	
DB_PASSWD	Database password	**edx**	
DB_NAME	Database name	**edx0**	
SERVER_NAME	Server name	**localhost**	
EDX_HOME	Platform Services home	**c:\edocs\TBM\estatement**	

VARIABLE	DESCRIPTION	WINDOWS	CUSTOM
EBM_HOME	EBM home path	**`c:\edocs\TBM`**	
DB_DATAPATH	Database Data File Path	**`c:\sqldir\data`**	
LOG	Database Log File Path	**`c:\sqldir\log`**	

Using Database Partitioning with EBM

Database partitioning (partition splitting) reduces the number of tables the system must scan when indexing your data. You specify the number of partitions when you create a DDN in the Command Center. At the first run of the Indexer job, EBM creates and populates a set of partitioned index tables to maintain your dynamic data.

SQL Server supports partitioned views. Index tables are created with a check-constraint, and the view is created on the underlying index tables. edocs recommends using 4 or 12 partitions for quarterly or monthly index tables.12 partitions are recommended for maximum performance.

For more information on using partitions with your DDNs, see the *EBM Administration Guide*.

Configuring a New SQL Server Database for Windows

Windows administrators need to complete two steps (and a third for WebSphere) to create and configure your Platform Services database:

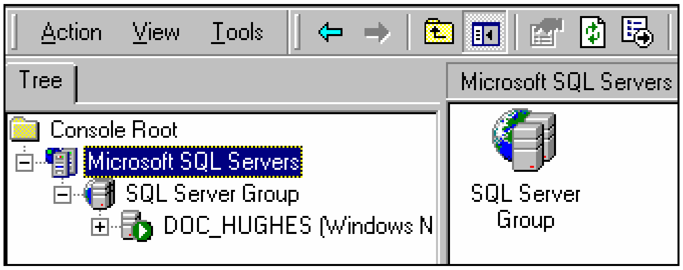
- Increasing the Default Width for SQL Server Numeric Columns
- Using the DBConfigTool

Increasing the Default Width for SQL Server Numeric Columns

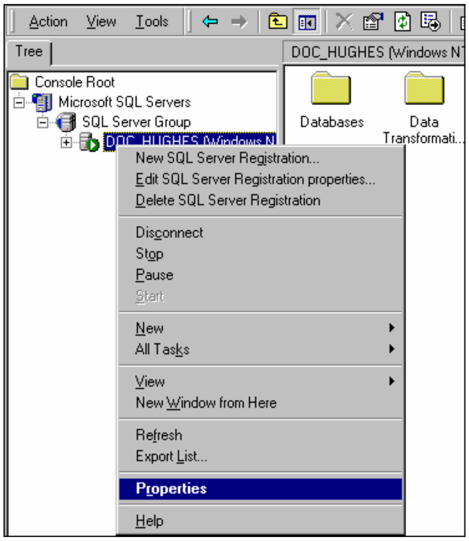
The SQL Server default width for numeric columns is 28 characters. However, EBM requires a minimum width of **38 characters** for numeric columns. You must increase these defaults **before** creating the database.

To increase the default width of numeric columns:

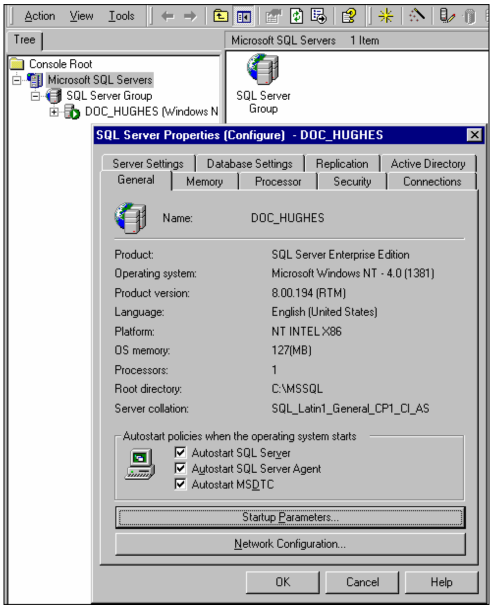
1. From the Start menu, select **`Programs`** and **`Microsoft SQL Server,`** and click **`Enterprise Manager`**. The SQL Enterprise Manager window appears.
2. In the left pane, expand **`Microsoft SQL Servers`** and **`SQL Server Group`** to show your server. The sample screen shows a SQL server named DOC_HUGHES.



3. In the left pane, highlight your SQL server by right clicking on its name, and then click **`Properties`** from the menu.



The SQL Server Properties dialog appears.



SQL Server Properties (Configure) - DOC_HUGHES

Server Settings | Database Settings | Replication | Active Directory
General | Memory | Processor | Security | Connections

Name: DOC_HUGHES

Product: SQL Server Enterprise Edition
Operating system: Microsoft Windows NT - 4.0 (1381)
Product version: 8.00.194 (RTM)
Language: English (United States)
Platform: NT INTEL X86
OS memory: 127(MB)
Processors: 1
Root directory: C:\MSSQL
Server collation: SQL_Latin1_General_CP1_CI_AS

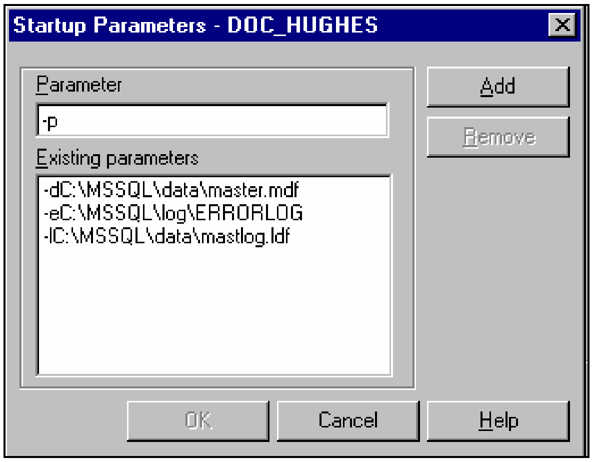
Autostart policies when the operating system starts
☑ Autostart SQL Server
☑ Autostart SQL Server Agent
☑ Autostart MSDTC

Startup Parameters...
Network Configuration...

OK Cancel Help

4. Click **Startup Parameters** at the bottom of the dialog. The Startup Parameters dialog appears showing the name of the server.

5. Enter the startup parameter **-p** in the Parameter text field.



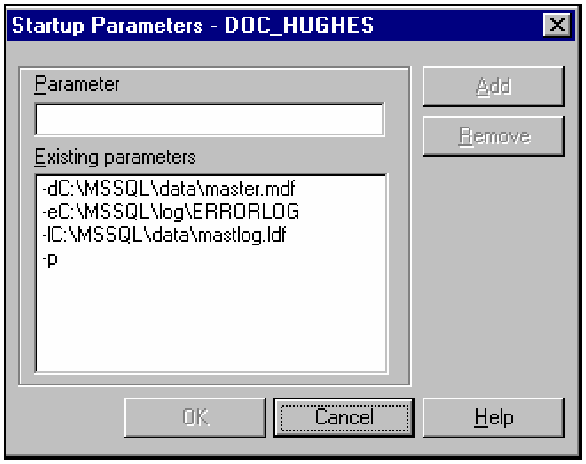
Startup Parameters - DOC_HUGHES

Parameter: -p

Existing parameters:

```
-dC:\MSSQL\data\master.mdf
-eC:\MSSQL\log\ERRORLOG
-lC:\MSSQL\data\mastlog.ldf
```

6. Click **Add.** The new startup parameter is added to the list of existing parameters.



Startup Parameters - DOC_HUGHES

Parameter:

Existing parameters:

```
-dC:\MSSQL\data\master.mdf
-eC:\MSSQL\log\ERRORLOG
-lC:\MSSQL\data\mastlog.ldf
-p
```

7. Click **OK** to close the dialog.

8. Stop and start your SQL server for the new startup parameter to take effect.

Using the DBConfigTool

Windows administrators use the Java database configuration tool, **DBConfigTool.jar**, to create and configure the production database. This wizard guides you to specify information about the database including the username and password needed to access it, the name of the database, the server on which it is installed, and the pathnames for the SQL Data and Log files

The DBConfigTool requires a Java SDK installed on the machine that runs it.

Caution When installing Microsoft SQL Server 2000, select **Mixed Mode (Windows Authentication and SQL Server Authentication)** as your default authentication mode. Selecting only Windows Authentication Mode might cause database creation and configuration to fail.

To configure a new EBM database for SQL Server on Windows:

1. Open a Command Prompt window and change directory to your EBM database home directory. For example:

   ```
   C:\> cd edocs\TBM\estatement\db\mssql
   ```

2. Run the Java database creation and configuration tool **`DBConfigTool.jar`**. For example:

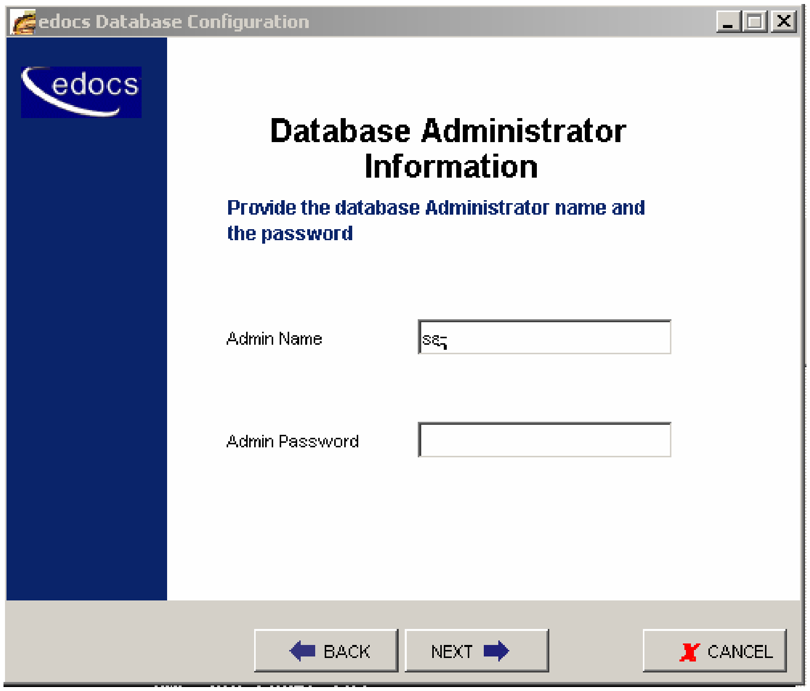
   ```
   \bea\jdk141_05\bin\java –jar DBConfigTool.jar
   ```

   Note that the above java command example uses the JDK provided by WebLogic. Your JDK location may differ.

   The Database Configuration screen appears.

3. Select **`Create and configure new database`** (default)**,** and then click **`Next`**. The Database Configuration screen appears.

4. Click **`Next`**. The Database Administrator Information screen appears.



Database Administrator Information

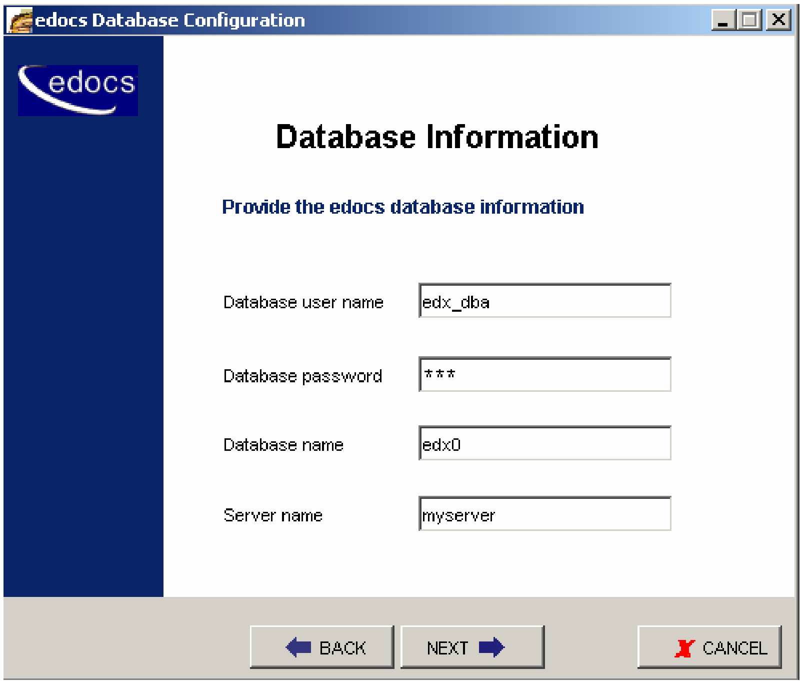
Provide the database Administrator name and the password

Admin Name: sa

Admin Password:

5. Enter the Admin Name and Admin Password (if your database has one) for the SQL database. If you are using the default values provided by SQL Server, the Admin Name is **sa** and the default password is left blank as shown in the example. Please consult for SQL Sever DBA for the correct values for your database server.

Click **Next**. The Database Information screen appears.



edocs Database Configuration
edocs
Database Information
Provide the edocs database information

Database user name: edx_dba

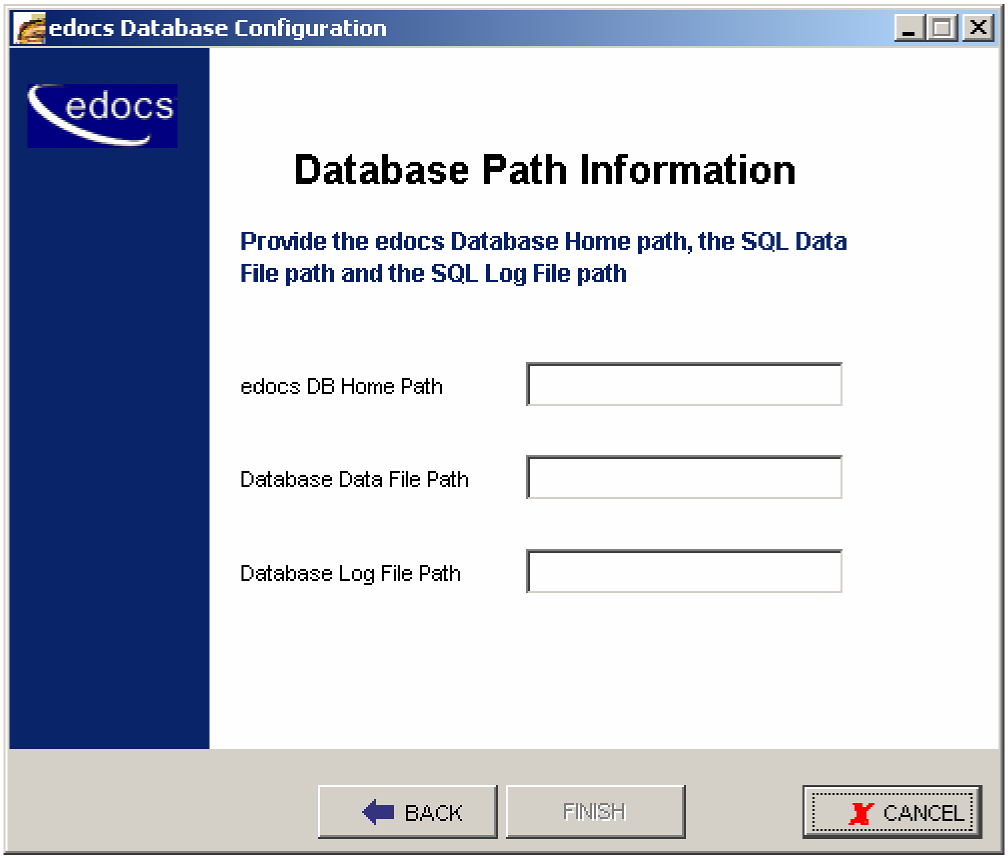
Database password: ***

Database name: edx0

Server name: myserver

6. Enter your database user name, password, database name, and the name of your database server, as shown in the example above (your server name will differ).

   Click **Next**. The Database Path Information screen appears.



Database Path Information

Provide the edocs Database Home path, the SQL Data File path and the SQL Log File path

edocs DB Home Path

Database Data File Path

Database Log File Path

7. Enter the pathname to the database installation directory, location of the SQL data file, and the location of the SQL log file. By default those values are as follows:
   - edocs DB Home Path: **`c:\edocs\TBM\estatement`**
   - Database Data File Path: **`c:\sqldir\data`**
   - Database Log File Path: **`c:\sqldir\log`**
8. Click **`Finish`**. The DBConfigTool will create your database.
9. A Success screen appears if the database creation process completed without errors.
10. If a Failed screen appears, see the next section.

Run Setup Scripts

On the database server, in EBM_HOME\db\mssql, run the following scripts (in sequence):

1. Run Add_pwc_tasks.sql
2. Run Add_pwc_job_types.sql
3. Run create_pmtdue_notification_Tables.sql

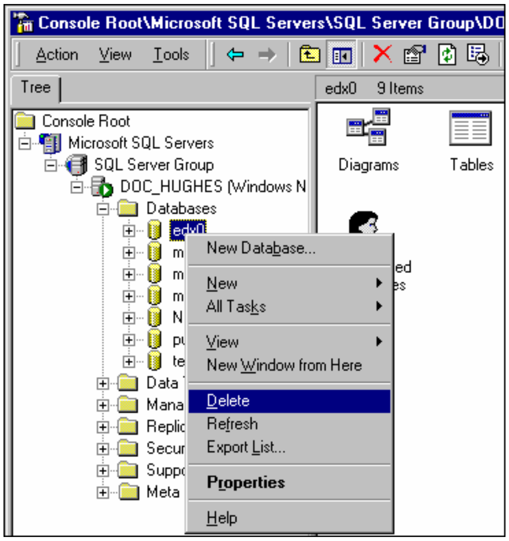
4. Run AddressBook.sql
5. Run createMessengerQueueTables.sql
6. Run create_unbilled.sql (Sample unbilled data set)

What to Do if Database Configuration Fails

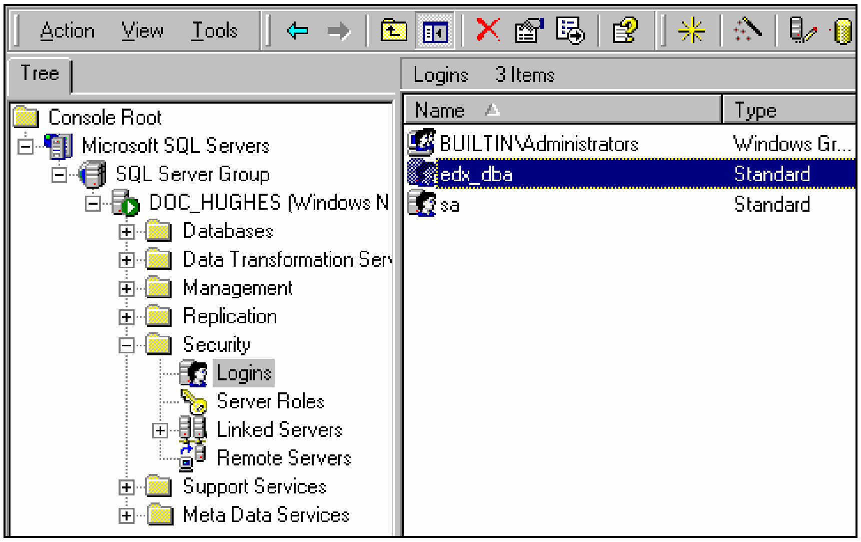
If you encounter errors during database creation and configuration, you must first remove the partially configured database before configuring the database again.

To recover from a failed database configuration for SQL Server:

1. Make sure your database server is running.
2. From the Start menu, select **Programs** and **Microsoft SQL Server**, and click **Enterprise Manager**. The Enterprise Manager screen opens.
3. Expand the SQL Server Group and click the **Database** folder.
4. Right-click the name of the newly created database (for example, **edx0**) and delete it from the list of installed databases.



5. Scroll down to the **Security** folder and expand it to show **Logins**. Delete the database user (for example, **edx_dba**) for the database that you just deleted.



Logins 3 Items

Name	Type
BUILTIN\Administrators	Windows Gr...
edx_dba	Standard
sa	Standard

6. Open a command line window and run the database configuration tool again, making sure that the values on each screen are correct before proceeding to the next screen.

Creating the Payment Database

Configuring the Payment database consists of running a script that creates tables and indexes in the EBM database, which Payment uses.

Running the Database Creation Script for MSSQL:

1. In the directory *%PAYMENT_HOME%\db\mssql*, edit *set_isql_options.bat* and enter the correct information for the database, user name and password (Payment uses the same database as EBM) on the line that starts with:

   ```
   set ISQL_OPTIONS......
   ```

2. Run *%PAYMENT_HOME%\db\mssql\create_payment_db.bat* to create payment databases.

Tip If you run the database creation script from a command prompt, you will see the database creation process and messages.

3. Restart the database.

Preparing for a Database Failure

The database for Payment should be backed-up regularly, for example nightly, to maximize the recovery effort in the event of a system failure. For example, if the Payment Database Server failed for some reason, the database administrator could restore from the latest backed-up version. A failure to the Payment Database Server also requires that the database administrator restart the Payment services in order to reestablish the database connection pool.

Payment will automatically handle single transaction failures by rolling back to the transaction's original state. This happens because each Payment operation is handled within its own transaction context.

Start and Test Your Database Server

Start and test your database server using the server documentation for your platform. If you encounter any errors, double-check the steps in these chapters before proceeding.

Once your database server starts successfully with the EBM database installed, you can proceed to configure your application server.

4 Configuring the WebLogic Server for Windows

Overview

This chapter assumes in-depth understanding of and practical experience with application server administration. Consult WebLogic Server documentation at http://bea.com as necessary.

You must start your WebLogic Server instance and bring up the Administrative Console before you begin this chapter.

Caution If you cannot bring up the WebLogic Console, you will be unable to proceed with configuring your application server for EBM.

edocs recommends that you install and configure EBM in the same top-level directory structure, first on the database server, then the application server.

If you have not already installed database server components and configured the database server for EBM, do so now.

For distributed environments, ensure that you have any required **database client software** installed on WebLogic Server and any other client machines of your database server.

This chapter provides instructions for configuring WebLogic Server to support EBM. It includes:

- Starting and Stopping WebLogic Server
- Capturing your Windows environment for EBM
- Passing Windows Environment Data to WebLogic
- Windows Services for EBM

Caution The installation and configuration examples shown in this guide use default EBM pathnames, privileges, and permissions. If you choose not to accept the default values, make sure your values are consistent on all servers across your installation of EBM.

About the Sample Windows Domain Used in this Guide

This guide uses the default WebLogic domain **`%WL_HOME%\edocs\domains\mydomain`** (**`%WL_HOME%`** is the directory path where you installed WebLogic). WebLogic users may use the Domain Configuration Wizard to create this domain or replace these pathnames with a custom domain created by your system administrator.

Caution If you use a custom domain, the examples in this guide must be changed accordingly or they may not work. edocs does not recommend that you accept the default path of **`\user_projects.`**

Starting and Stopping WebLogic Server

Developers and system administrators need to be familiar with how to stop and start WebLogic Server and any active web applications for your platform. Consult your BEA WebLogic documentation for instructions on how to do this.

Sourcing Your Configuration

About Sourcing Your Configuration

Before you start your WebLogic server instance, you must edit its startup script to **source** your customized version of the configuration file **`edx.config`**, which passes your EBM environment to WebLogic Server at startup.

Starting and Stopping an Active Application Server

Improperly starting or stopping an application server in an active EBM production environment can produce unexpected and unintended results. You can create custom startup and shutdown scripts that include all your command parameters, as well as the command used to start or stop the Scheduler, to schedule and run jobs in the EBM Command Center.

The default command-line startup shell scripts are fine for an inactive production environment where there are no running jobs. However, the startup process will stop immediately if you enter a **`Ctrl+C`** (often used to force a hard shutdown of the server) in the startup directory, or if you close the terminal session.

By default, if you use the Windows Control Panel to stop a server instance, the Windows Service Control Manager (SCM) kills the server's Java Virtual Machine (JVM). If you kill the JVM, the server immediately stops all processing. Any session data is lost. If you kill the JVM for an Administration Server while the server is writing to the config.xml file, you can corrupt the config.xml file. See BEA Documentation for Enabling Graceful Shutdowns from the Control Panel .

Capturing Your Windows Environment for EBM

EBM installs several configuration files that you use to define your EBM environment. These configuration scripts are required **only on the application server**:

`%EDX_HOME%\config\edx_load.config.bat`	Editable configuration file stores database Java options required by your application server
`%EDX_HOME%\config\edx.config.bat`	shell script passes the environment data in **`edx_env.bat`** and **`edx_load.config.bat`** to your application server when called in your startup script

This section describes how to run **`edx.config`** to capture your environment variables.

Using *edx.config.bat* and *edx_load.config.bat* to Store Environment Data

You must edit the configuration file **`edx_load.config.bat`** to set values for your database user, password, and server name. Then you must run **`edx.config.bat`**.

Use the Appendix Quick Reference to enter application server environment variables for your platform for each of the specified parameters. You may want to print the *Environment Variables* sections for easy reference. You can accept the default values, if appropriate, or enter your own.

You may want to make backup copies of the following files:

- edx.config.bat
- edx_env.bat
- edx_load.config.bat

To edit Windows environment data with *edx.config.bat*:

1. Navigate to **`%EDX_HOME%\config`** and edit **`edx.config.bat`**.

2. Add the following lines to the **`:wlMain`** section:

```
@set CLASSPATH=%CLASSPATH%;C:\edocs\TBM\lib\commons-
logging.jar
@set CLASSPATH=%CLASSPATH%;C:\edocs\TBM\lib\Configuration.jar
@set CLASSPATH=%CLASSPATH%;C:\edocs\TBM\lib\javachart.jar
@set CLASSPATH=%CLASSPATH%;C:\edocs\TBM\lib\ldeprotocol.jar
@set CLASSPATH=%CLASSPATH%;C:\edocs\TBM\lib\log4j.jar
@set CLASSPATH=%CLASSPATH%;C:\edocs\TBM\config
```

3. Still in *edx.config.bat*, add **`|com.edocs.domain.telco.lde.protocol`** to the following line:

```
@set JAVA_OPTIONS=%JAVA_OPTIONS% -Dedx.home="%EDX_HOME%" -
Djava.protocol.handler.pkgs="com.edocs.protocol"
```

to make:

```
@set JAVA_OPTIONS=%JAVA_OPTIONS% -Dedx.home="%EDX_HOME%" -Djava.protocol.handler.pkgs="com.edocs.protocol|com.edocs.domain.telco.lde.protocol"
```

4. Save and close all files.

To edit Windows environment data with edx.config.bat

1. Navigate to **`%EDX_HOME%\config`** and open **`edx_env.bat`**.
2. Modify the default settings to reflect your environment. For example:

```
@rem define APP_SERVER
@set APP_SERVER=wl
@rem define APP_SERVER

@rem define EDX_HOME
@set EDX_HOME=C:\edocs\TBM\estatement
@rem define EDX_HOME

@rem define JAVA_HOME
@set JAVA_HOME=C:\bea\jdk141_05
@rem define JAVA_HOME

@rem define WL_HOME
@set WL_HOME=C:\bea\weblogic81
@rem define WL_HOME
```

Caution Make sure you set all paths to the appropriate point releases/patches for WebLogic Server and JDK, if necessary. Check the Release Notes and your system documentation for updated requirements.

To edit Windows Java Options in *edx_load.config.bat:*

1. Navigate to **`%EDX_HOME%\config`** and open **`edx_load.config.bat.`**
2. Modify the default settings for **`com.edocs.tasks.loader`** to reflect your database user, password, and server name. Use the settings for your database as described in the previous chapter. For example:

```
@set JAVA_OPTIONS=%JAVA_OPTIONS% -Dcom.edocs.tasks.loader.user=edx_dba
@set JAVA_OPTIONS=%JAVA_OPTIONS% -Dcom.edocs.tasks.loader.password=edx
@set JAVA_OPTIONS=%JAVA_OPTIONS% -Dcom.edocs.tasks.loader.alias=localhost
```

3. Save and close the file.

Caution Make sure you set all paths to the appropriate point releases/patches for WebLogic Server and JDK, if necessary. Check the Release Notes and your system documentation for updated requirements.

Passing Windows Environment Data to WebLogic

edx.config.bat is a shell script that you call and process in your application server startup script to pass your EBM environment (stored in **edx_env.bat** and **edx_load.config.bat**) to WebLogic.

Caution: Do not confuse **edx.config.bat** with **edx_env.bat**, in which you enter the environment data to pass to the server.

This section describes how to use **edx.config.bat** to pass your environment data to WebLogic at server startup.

To pass your EBM environment to WebLogic (overview):

1. Determine whether you wish to start WebLogic as a Windows Service or directly from the startup script. Use the appropriate procedure for your service or startup scripts.
2. In your **domain** service or startup script, set your EBM home directory, **%EBM_HOME%.**
3. In your **domain** service or startup script, call and process the configuration script **edx.config.bat.** This procedure is called **sourcing** your configuration.
4. In the **master** service or startup script, set your CLASSPATH, to use the classpath defined in **edx.config.bat.**

Passing Your Configuration to WebLogic Running as a Windows Service

edocs recommends installing WebLogic Server as a Windows Service, and modifying the script that calls that service. For WebLogic, this file is **%WL_HOME%\edocs\domains\mydomain\InstallService.cmd.**This script calls the master startup script **InstallSvc.cmd**.

Before editing either of these files, be sure to save a backup copy **in a different directory**.

Example InstallService.cmd for WebLogic:

Bold indicates text that you should add or change from the default.

```
.
.
.
@rem Set WLS_USER equal to your system username and WLS_PW equal
@rem to your system password for no username and password prompt
@rem during server startup.  Both are required to bypass the startup
@rem prompt.
```

```
set WLS_USER=
set WLS_PW=

.
.
.

set EDX_HOME=C:\edocs\TBM\estatement
call %EDX_HOME%\config\edx.config.bat

@rem Set JAVA_OPTIONS to the java flags you want to pass to the vm. i.e.:
@rem set JAVA_OPTIONS=-Dweblogic.attribute=value -Djava.attribute=value

set JAVA_OPTIONS=%JAVA_OPTIONS%

@rem Set JAVA_VM to the java virtual machine you want to run. For instanc
@rem set JAVA_VM=-server

set JAVA_VM=

@rem Set MEM_ARGS to the memory args you want to pass to java. For instan
@rem set MEM_ARGS=-Xms32m -Xmx200m

set MEM_ARGS="-ms128m -mx128m -Xss1m -noclassgc"

.
.
.

:installSvc

rem *** Set up extrapath for win32 and win64 platform separately

if not "%WL_USE_64BITDLL%" == "true" set
EXTRAPATH=%WL_HOME%\server\bin;%JAVA_HOME%\jre\bin;%JAVA_HOME%\bin;%WL_HO
ME%\server\bin\oci920_8;%EDX_HOME%\lib

if "%WL_USE_64BITDLL%" == "true" set
EXTRAPATH=%WL_HOME%\server\bin\win64;%WL_HOME%\server\bin;%JAVA_HOME%\jre
\bin;%JAVA_HOME%\bin;%WL_HOME%\server\bin\win64\oci920_8;%EDX_HOME%\lib

rem *** Install the service

"%WL_HOME%\server\bin\beasvc" -install -svcname:"beasvc
%DOMAIN_NAME%_%SERVER_NAME%" -javahome:"%JAVA_HOME%" -
execdir:"%USERDOMAIN_HOME%" -extrapath:"%EXTRAPATH%" -password:"%WLS_PW%"
-cmdline:%CMDLINE%

:finish

ENDLOCAL
```

To edit InstallService.cmd for WebLogic:

See the example above for default settings. Make sure to change these as needed for your environment.

Caution Make sure you set all paths to the appropriate point releases/patches for WebLogic Server and JDK, if necessary. Check the Release Notes and your system documentation for updated requirements.

1. Stop WebLogic Server and all application server instances.
2. Navigate to the **`%WL_HOME%\user_project\domains\mydomain`** subdirectory of your application server home directory.
3. Open **`InstallService.cmd`** by right clicking on its name, and selecting **`Edit`**.

4. Before the **JAVA_OPTIONS** definition, set **EDX_HOME** and call **edx.config.bat**. For example:

   ```
   set EDX_HOME=C:\edocs\TBM\estatement
   call %EDX_HOME%\config\edx.config.bat
   ```

5. Set **JAVA_OPTIONS** to **%JAVA_OPTIONS%**.
6. Set **JAVA_VM** to null.
7. Optimize JVM Memory by increasing the memory arguments allocated to the Java Virtual Machine (JVM) on the application server. For example (quotes are optional):

   ```
   set MEM_ARGS=“-ms128m –mx128m –Xss1m –noclassgc”
   ```

8. Add your EBM **\lib** directory to the **EXTRAPATH** setting. See the examples above.
9. (Optional) You can set your application server user and password in the script (to bypass entering it in a console window) by specifying them for **WLS_USER** and **WLS_PW**.
10. Save and close **InstallService.cmd**.

To edit InstallSvc.cmd for WebLogic:

Caution

This procedure is required by a defect in WebLogic that does not correctly pass classpath settings from the domain to the master when the master script is called. Consult your WebLogic administrator when editing scripts that control multiple domains.

1. Stop WebLogic Server and all application server instances.
2. Navigate to the **%WL_HOME%\weblogic81\server\bin** subdirectory of your application server home directory.
3. Open **InstallSvc.cmd** by right clicking on its name, and selecting **Edit**.
4. Set your EBM home directory and call your EBM environment script right after the CLASSPATH setting. For example:

   ```
   set EDX_HOME=C:\edocs\TBM\estatement
   call %EDX_HOME%\config\edx.config.bat
   ```

5. Save and close **InstallSvc.cmd**.

Passing Your Configuration in a Startup Script for WebLogic

You can also choose to start WebLogic Server directly by modifying the server startup script to source your configuration. WebLogic recommends that you start up the server from your domain, using **%WL_HOME%\edocs\domains\mydomain\startWebLogic.cmd**.

Before editing either of these files, be sure to save a backup copy **in a different directory**.

Example startWebLogic.cmd for WebLogic:

```
.
.
.
@REM Initialize the common environment.

set WL_HOME=C:\bea\weblogic81
for %%i in ("%WL_HOME%") do set WL_HOME=%%~fsi

set PRODUCTION_MODE=true

set JAVA_VENDOR=Sun

set JAVA_HOME=C:\bea\jdk141_05

for %%i in ("%JAVA_HOME%") do set JAVA_HOME=%%~fsi

set MEM_ARGS=-ms128m -mx128m -Xss1m -noclassgc

@REM Call commEnv here AFTER setting the java_vendor to get common
environmental settings.

call "%WL_HOME%\common\bin\commEnv.cmd"

@REM Set SERVER_NAME to the name of the server you wish to start up.

set SERVER_NAME=myserver

set CLASSPATH=%WEBLOGIC_CLASSPATH%;%POINTBASE_CLASSPATH%;%JAVA_HOME%\jre\
lib\rt.jar;%WL_HOME%\server\lib\webservices.jar;%CLASSPATH%

set EDX_HOME=C:\edocs\TBM\estatement
call %EDX_HOME%\config\edx.config.bat

.
.
.
```

To edit startWebLogic.cmd for WebLogic:

See the example above for default settings. Make sure to change these as needed for your environment.

1. Stop WebLogic Server and all application server instances.
2. Edit **`%WL_HOME%\edocs\domains\mydomain\startWebLogic.cmd.`**
3. Optimize JVM Memory by increasing the memory arguments allocated to the Java Virtual Machine (JVM) on the application server. For example (quotes are optional):

   ```
   set MEM_ARGS=-ms128m –mx128m –Xss1m –noclassgc
   ```

4. Set your edocs home directory **`%EDX_HOME%`** and call **`edx.config.bat`** just after the set CLASSPATH statement. For example:

   ```
   set EDX_HOME=C:\edocs\TBM\estatement
   call %EDX_HOME%\config\edx.config.bat
   ```

5. Add the **`classpath`** parameter to the **`java`** command at the end so it uses the paths to the EBM classes. For example:

```
%JAVA_HOME%\bin\java %JAVA_VM% %MEM_ARGS% %JAVA_OPTIONS% -classpath
"%CLASSPATH%" -Dweblogic.Name=%SERVER_NAME% -
Dweblogic.ProductionModeEnabled=%PRODUCTION_MODE% -
Djava.security.policy="%WL_HOME%\server\lib\weblogic.policy"
weblogic.Server
```

6. Save and close **startWebLogic.cmd**.

Windows Services for EBM

Setting Up a WebLogic Server Instance as a Windows Service

If you want a WebLogic Server instance to start automatically when you boot a Windows host computer, you can set up the server as a Windows service.

For detailed instructions on setting up WebLogic Server as a Windows Service, see the BEA documentation.

Setting Up the Platform Services Scheduler as a Windows Service

After all EBM EAR files have been deployed to the application server **and** WebLogic is running, you must start the EBM Scheduler in order to schedule and run jobs in the EBM Command Center. If you attempt to run a new job with the Scheduler not running, the job will not run and you will see ‘Not yet started’ as its status.

To install the Scheduler as a Windows Service, you must modify the Scheduler template file **SCH.txt**, installed to the **bin** directory for EBM.

To install the Scheduler as a Windows Service:

1. Navigate to the **bin** directory for Platform Services, or **%EDX_HOME%\bin**.

2. Open the Scheduler template file **SCH.txt** and modify the Java classpath to reflect your active Java environment. For example:

```
classpath=C:\bea\jdk141_05\lib\tools.jar;c:\bea\wlserver\lib\w
eblogic.jar;c:\edocs\TBM\estatement\lib\edx_client.jar;c:\edoc
s\TBM\estatement\lib\edx_common.jar
```

Caution Make sure you set all paths to the appropriate point releases/patches for WebLogic Server and JDK, if necessary. Check the Release Notes and your system documentation for updated requirements.

3. Confirm that the following line of code is present in the file for your host and port:

```
-Djava.naming.provider.url=t3://localhost:7001
```

4. If you want the Scheduler to log information to a file rather than to the console, add the following value in **SCH.txt**:

```
-Dcom.edocs.pwc.debug=true scheduler_logfile_name
```

5. Confirm that all the directory references in **`SCH.txt`** are correct.

6. Save and close **`SCH.txt`**.

7. Open a command prompt window, and then change directory to **`%EDX_HOME%\bin`**. Use the **`schedulersvc`** command to install the Scheduler as a Windows Service, for example:

   ```
   C:\> schedulersvc -install C:\edocs\TBM\estatement\bin\SCH.txt
   ```

8. If the Scheduler service is installed successfully, a confirmation message appears.

Configuring Payment

Updating the Payment and EBM configuration files:

1. Log on as the WebLogic server owner, and change your working directory to *%PAYMENT_HOME%\config,* for example:

   ```
   cd \edocs\TBM\Payment\config
   ```

2. If you are not using the default EBM and Payment directories, then edit the *edx_payment.config* file, and correct the entry that defines PAYMENT_HOME.

3. Copy the updated *edx_payment.config* file to *%EDX_HOME%\config*.

JTA timeout configuration

If pmtCheckSubmit will process a large number of checks, the JTA timeout value in WebLogic must be increased to keep the connection open long enough to process all the checks. For every 1500 checks to be processed, the timeout value should be increased by 30 seconds.

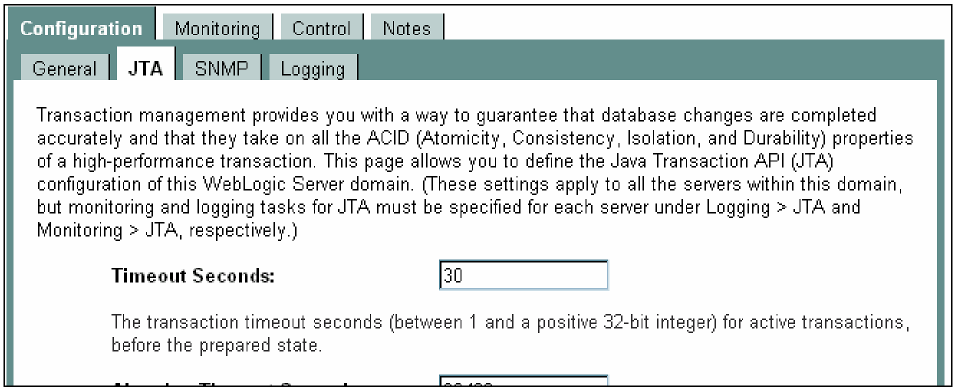
The JTA timeout value allows the system to recover a "lost" transaction. Your application code should handle commit and rollback of database transactions. But, if there is a programming error such that transactions are not committed or rolled back, a timeout provides a way for the application server to release those transactions.

The default timeout (30 seconds) is fine for web-based transactions, but it is too small for batch based transactions.

To configure the JTA Timeout setting:

1. Enter the URL to log on to the WebLogic console

2. Click on the **`JTA`** in the left window, and a screen similar to the following appears:



Transaction management provides you with a way to guarantee that database changes are completed accurately and that they take on all the ACID (Atomicity, Consistency, Isolation, and Durability) properties of a high-performance transaction. This page allows you to define the Java Transaction API (JTA) configuration of this WebLogic Server domain. (These settings apply to all the servers within this domain, but monitoring and logging tasks for JTA must be specified for each server under Logging > JTA and Monitoring > JTA, respectively.)

Timeout Seconds: 30

The transaction timeout seconds (between 1 and a positive 32-bit integer) for active transactions, before the prepared state.

3. Change the Timeout Seconds field to the multiple of 30 seconds that you require, and click the **`Apply`** button.

Configuring Support for VeriSign Processing

If you are going to use a VeriSign credit card gateway, then you must edit the classpath in your application server startup script, and configure your java security file.

To edit the classpath in the application server startup script

Edit the startup script for your application server to add *jsse.jar*, *jnet.jar* and *jcert.jar*. These files are located in the lib subdirectory of *%EBM_HOME%\payment*.

These JSEE JAR files must be added to the classpath **before** the WebLogic JAR files. For example,

```
set
CLASSPATH=<JSSE_PATH>\jsse.jar;<JSSE_PATH>\jcert.jar;<JSSE_PAT
H>\jnet.jar;%CLASSPATH%
```

Where <JSSE_PATH> is the path to the JSEE JAR files.

Configuring java security

Modify the *java.security* file to configure Java to use the JSSE, which is located in *%EBM_HOME%\payment\lib* to add the following entries:

```
security.provider.1=sun.security.provider.Sun
security.provider.2=com.sun.net.ssl.internal.ssl.Provider
```

If the file already uses the numbers shown above, then use the next available numbers that maintain a contiguous sequence.

Caution Be sure that you edit the correct java security file. Many installations have more than one JDK installed. Edit the one that the application server uses.

Payment Configuration

After configuring the database and the application server for Payment, the following steps are required to implement online bill payment:

1. Configure a payment gateway for online check and/or credit card processing.
2. Enroll customers for online bill viewing and payment.
3. Set up Payment jobs to process payments and optionally send reminders.

See the *EBM Administration Guide* for information about configuring a payment gateway, enrolling customers and other operational issues.

Troubleshooting Tips for WebLogic on Windows

If the WebLogic service is unable to find *edx_load.config.bat* at startup:

Add the path **`-extrapath:C:\bea\weblogicXX\server\bin;%PATH%`** to **`installNtService.cmd`** (substituting your drive where necessary). For example

```
"C:\bea\weblogic81\server\bin\beasvc" –install –svcname:myserver
–javahome:"%JAVA_HOME%" –execdir:"C:\bea\weblogic81"
–extrapath:"C:\bea\weblogic81\server\bin;%PATH%" –cmdline:%CMDLINE%
-password:lovelyday
```

If you encounter problems when running customized Web pages:

1. From your domain in the WebLogic Server Console, select **`Servers`** and **`myserver`** A tabbed dialog with the name of your server appears in the right pane.
2. On the Configuration/Compilers tab, change the Java Compiler value from the default **`javac`** to the location of **`javac`** in the JDK installed **with WebLogic Server**.
3. Restart WebLogic Server.

Caution Make sure you set all paths to the appropriate point releases/patches for WebLogic Server and JDK, if necessary. Check the Release Notes and your system documentation for updated requirements.

Editing XMA Properties

On the application server, you must edit both hibernate.properties and *hierarchy.hibernate.properties* in *%EBM_HOME\config* for the new database:

1. Edit the following statement to replace *localhost* with the IP address of your database host server, and *1433* with the database port number (if you do not use the default port number):

```
hibernate.connection.url=jdbc:inetpool:inetdae7://localhost:14
33
```

Edit this statement to replace *edx_dba* with the name of your database user:

```
hibernate.connection.username=edx_dba
```

Edit this statement to replace *edx* with the name of your database user’s password:

```
hibernate.connection.password=edx
```

2. Edit *log4j.xml* in EBM_HOME*/config*:

   Edit the following line and change the installation path as necessary:

   ```
   <param name="File"
   value="C:\edocs\TBM\estatement\logs\log4j.log">
   ```

5

Configuring Java Resources for WebLogic

Overview

This chapter assumes in-depth understanding of and practical experience with application server administration. It is designed for experienced WebLogic administrators and primarily presents only the steps and settings specific to EBM.

See WebLogic Server documentation at http://bea.com/ for detailed step-by-step instructions on Java resource configuration, performance, and tuning. You must also consult your application server administrator for settings that may be specific to your configuration.

You must start your WebLogic Server instance and bring up the Administrative Console before you begin this chapter.

Caution If you cannot bring up the WebLogic Console, you will be unable to proceed with configuring your application server for EBM.

Configuring Java Database Connectivity (JDBC) for EBM

After you have successfully configured the EBM database, you must configure Java Database Connectivity (JDBC) resources on the EBM application server. JDBC Connections on the application server support data retrieval from relational databases and other data sources.

WebLogic Environment Variables

Caution Make sure you set all paths to the appropriate point releases/patches for WebLogic Server and JDK, if necessary. Check the Release Notes and your system documentation for updated requirements to these environment variables.

VARIABLE	DESCRIPTION	WINDOWS
APP_PORT	app server port	`7001`
ADMIN_PORT	app server admin port	`7002`
JAVA_HOME	Java home directory	`%WLHOME%\jdk141_05`

About JDBC Connections for EBM

JDBC connection pools contain named groups of JDBC Connections that are created when the connection pool is registered, usually when starting up WebLogic Server. WebLogic Server opens JDBC Connections to the database during startup and adds these connections to the pool. A J2EE web application borrows a connection from the pool, uses it, and then returns it to the pool by closing it.

JDBC data sources enable JDBC clients to obtain a connection to a Database Management System (DBMS). Each data source points to the value specified for the Name attribute when a JDBC connection pool was configured.

EBM requires three sets of **JDBC Connection Pools** and related **JDBC Data Sources**:

- **edxAdmin** supports the Command Center through the EBM web application
- **edxLogger** supports EBM logging through the EBM web application
- **edxUser** supports user data retrieval through custom web applications

For more details on configuring JDBC Connections, please see the JDBC documentation for your application and database servers.

Tip **`edxAdmin`** connection pools support concurrency for scheduling multiple jobs. Tuning **`edxAdmin`** connection capacity and threads can improve EBM email performance.

Configuring JDBC Connections for WebLogic

You must create three sets of JDBC connection pools and three sets of transaction datasources. Their names are specific to edocs across all platforms, but JDBC properties vary by both application server and database server.

See Appendix A for appropriate WebLogic JDBC configuration settings for SQL Server.

You must enter the same information six times: one connection pool and one data source each for **`Admin`**, **`User`**, and **`Logger`**. Make sure you have chosen the correct properties for your application server and database server, and that each datasource and its properties maps to the connection pool of the same name.

Caution Make sure you are using the correct properties for your combination of application server, database, and JDBC resource.

For more details on how to configure JDBC connections, see WebLogic Server documentation at http://bea.com.

To configure JDBC Connections for WebLogic:

1. Create a JDBC Connection Pool each for **`edxAdmin`**, **`edxLogger`**, and **`edxUser`**. Use the appropriate JDBC values for your database server.
2. Create a JDBC Data Source each for **`edxAdmin`**, **`edxLogger`**, and **`edxUser`**. Use the appropriate JDBC values for your database server.
3. Review your connections. Each data source should target the connection pool of the same name (**`Admin`**, **`User`**, or **`Logger`**).
4. When you are finished, proceed to the next section to configure Java Messaging Services (JMS) for EBM.

JDBC – WebLogic for Windows

Tip You may save time by **cloning** additional Java resources. Right-click a resource and select **`Clone <name>`**, then change the resource name and properties as required.

JDBC Connection Pools for WebLogic

Create three JDBC Connection Pools, using WebLogic Server documentation at http://bea.com. Use the same **Properties** for all three connection pools. Make sure to deploy them to the server you are configuring for EBM (in the examples of this guide, the default **`myserver`**).

WebLogic creates a new JDBC Connection Pool using a wizard. Follow the prompts, and enter:

- **Database type** = MS SQL Server
- **Database Drive**r = Other

For each of three connection pools, using the following names and properties:

Pool 1: Admin	**Pool 2: User**	**Pool 3: Logger**
`edxAdminConnectionPool`	**`edxUserConnectionPool`**	**`edxLoggerConnectionPool`**

General Tab	
URL	**`jdbc:inetpool:inetdae7://localhost:1433`**
Driver Classname	**`com.inet.pool.PoolDriver`**
Database User	Enter the database user name. This document uses **`edx_dba`**.
Password	Enter the password for the database user. This document uses **`edx`**.

After the wizard completes, go to the Configuration page to make adjustments using the values shown in the following table (on the Connections tab, click **`Show`** for Advanced Options):

Connections Tab	
Initial Capacity	**`1`**
Maximum Capacity	**`20`**
Capacity Increment	**`5`**
Login Delay	**`1`**
Statement Cache Size	**`300`**
Test Frequency	**`60`**
Allow Shrinking	**`True`** (box checked)
Shrink Frequency	**`15`**
Test Reserved Connections	**`TRUE (checked)`**
Test Released Connections	**`FALSE (unchecked)`**
Test Table Name	**`job`**

Click **`Apply`** to save these values for each connection pool.

JDBC Data Sources for WebLogic

Create three transaction data sources, using WebLogic Server documentation at http://bea.com.

	Datasource 1: Admin	Datasource 2: User	Datasource 3: Logger
Name	`edxAdminDataSource`	`edxUserDataSource`	`edxLoggerDataSource`
JNDI Name	`edx.databasePool`	`edx.user.databasePool`	`edx.logger.databasePool`
Pool Name	`edxAdminConnectionPool`	`edxUserConnectionPool`	`edxLoggerConnectionPool`

Configuration Tab - Advanced Options (use defaults)	
Emulate Two-Phase Commit for non-XA Driver	`FALSE (unchecked)`
Row Prefetch Enabled	`FALSE (unchecked)`
Stream Chunk Size: bytes	`256`

On the **`Targets`** tab, select the server that will use this Data Source.

Configuring Java Messaging Services (JMS) for EBM

After you have successfully configured JDBC Connections, you must now configure Java Messaging Services (JMS) on the application server for EBM. EBM requires three sets of JMS resources:

- **edxAnnotation** supports Line Item Dispute and Annotation features
- **edxDispute** supports Line Item Dispute and Annotation features
- **edxLogger** supports EBM logging through the EBM web application. It requires **five** JMS consumers and session pools.

Tip If your web application does not implement Line Item Dispute and Annotation, you need only configure JMS resources for **`edxLogger`**.

To configure each set of JMS resources, see Appendix A for JMS Resources settings for WebLogic Server.

About JMS Resources for EBM

JMS enable web application components to asynchronously send and receive messages.

- **JMS Connection Factories** are data objects that enable Java Messaging Service (JMS) clients to create JMS connections. You define and configure one or more connection factories to create connections with predefined attributes. WebLogic Server adds the connection factories to the JNDI space during startup, and each J2EE web application retrieves a connection factory using the JNDI on the application server.
- **JMS Stores** store persistent messages in a database accessed through a designated JDBC connection pool. The JMS database can be any database that is accessible through a WebLogic-supported JDBC driver. When creating a JMS Store, you must define the name of the **backing store**, and the **JDBC connection pool** and **database table name prefix** for use with multiple instances
- **JMS Servers** manage connections and message requests on behalf of clients.
- **JMS Topics** can be one of two destinations that you can configure for a JMS server. The other destination is a JMS queue. WebLogic Server allows you to configure one or more destinations for the JMS server. You can configure destinations explicitly or with a **destination template** (useful for multiple destinations with similar attribute values).
- **JMS Session Pools** allow a JMS listener (called a **Consumer** in WebLogic) to have multiple threads that improve performance under heavy load. Each JMS consumer requires its own session pool.

Configuring JMS Resources for WebLogic

You must enter very similar information many times: one set of JMS resources each for annotation and dispute, and FIVE sets for logging. Make sure you have chosen the correct properties for the resource you are creating, and that each resource maps to others of the **same name**.

Use the JMS settings in Appendix A to configure JMS settings for WebLogic Server. For general information about configuring Java resources for WebLogic, see WebLogic Server documentation at http://bea.com.

Tip If your web application does not use Line Item Dispute and Annotation, you need only configure JMS for edxLogger.

To configure JMS for WebLogic:

1. Create one JMS Connection Factory each for **`edxAnnotation`**, **`edxDispute`**, and **`edxLogger`**
2. Create one JMS Store each for each for **`edxAnnotation`**, **`edxDispute`**, and **`edxLogger`**.
3. Create one JMS Server each for each for **`edxAnnotation`**, **`edxDispute`**, and **`edxLogger`**.

4. Create one JMS Topic each for the JMS servers of each for **`edxAnnotation`**, **`edxDispute`**, and **`edxLogger`**.

5. Create one JMS Session Pool each for **`edxAnnotation`** and **`edxDispute`**.

6. Create one JMS Consumer each for **`edxAnnotation`** and **`edxDispute`**.

7. Create FIVE session pools and consumers for **`edxLogger`** as listed.

Tip

You may **clone** additional Java resources. Right-click a resource and select **`Clone <name>`**, then change the resource name and properties as required.

JMS Connection Factories

Create three JMS connection factories, using WebLogic Server documentation at http://bea.com. You may accept the default **Properties** for all three connection factories, or consult your application server administrator to tune these values.

	1: Annotation	**2: Dispute**	**3: Logger**
Name	**`edxAnnotationTCF`**	**`edxDisputeTCF`**	**`edxLoggerTCF`**
JNDI Name	**`edx/tcf/annotate`**	**`edx/tcf/dispute`**	**`edx/tcf/log`**

On the Targets tab, select the Servers that will use each JMS Connection Factory.

JMS (JDBC) Stores

Create three JMS JDBC Stores, using WebLogic Server documentation at http://bea.com. You may accept the default **`Prefix Name=<NULL>`** for all three stores, or consult your application server administrator to tune these values.

Name (of JMS Store)	**Connection Pool**
`edxAnnotationStore`	**`edxUserConnectionPool`**
`edxDisputeStore`	**`edxUserConnectionPool`**
`edxLoggerStore`	**`edxLoggerConnectionPool`**

JMS Servers

Create three JMS Servers, using WebLogic Server documentation at http://bea.com. You may accept the default **Properties** for all three servers, or consult your application server administrator to tune these values.

	1: Annotation	**2: Dispute**	**3: Logger**
Name	**`edxAnnotationServer`**	**`edxDisputeServer`**	**`edxLoggerServer`**
(Persistent) Store	**`edxAnnotationStore`**	**`edxDisputeStore`**	**`edxLoggerStore`**

Targets Tab	
Targets-Server	[select **`myserver`** from drop-down menu]

JMS Topics

Create three JMS Topics, using WebLogic Server documentation at http://bea.com. For WebLogic 7 select **`Destinations`** under each defined Server, then click on **`Configure a new JMSTopic`**. Make sure to create the matching topic for each server.

	1: Annotation	**2: Dispute**	**C3: Logger**
Name	**`edxAnnotationTopic`**	**`edxDisputeTopic`**	**`edxLoggerTopic`**
JNDI Name	**`edx/jms/annotate`**	**`edx/jms/dispute`**	**`edx/jms/log`**
Enable Store	**`True`**		

JMS Session Pools and Consumers for Annotation and Dispute

Tip

If your deployment does not use annotation and dispute, you can skip to configuring session pools and consumers for Logger.

Create one pair of JMS Session Pools and Consumers each for Annotation and Dispute, using WebLogic Server documentation at http://bea.com. Set **`Acknowledge Mode`** to **`auto`** and **`Sessions Maximum`** to **`-1`** for all three Session Pools.

Session Pool	**1: Annotation**	**2: Dispute**
Name	**`edxAnnotationPool`**	**`edxDisputePool`**
Connection Factory	**`edx/tcf/annotate`**	**`edx/tcf/dispute`**
Listener Class	**`com.edocs.services.annotation.Listener`**	**`com.edocs.services.dispute.Listener`**

Tip

For each session pool, -1 specifies no session maximum. Tune each Session Maximum to the maximum number of threads for each pool.

Consumer	1: Annotation	2: Dispute
Name	`edxAnnotationConsumer`	`edxDisputeConsumer`
Messages Maximum	`10`	`10`
Selector	`JMSType='USER'`	`JMSType='USER'`
Destination	`edx/jms/annotate`	`edx/jms/dispute`

JMS Session Pools and Consumers for Logging

Create FIVE pairs of **JMS Session Pools and Consumers** for **Logger,** using WebLogic Server documentation at http://bea.com. **`Set Acknowledge Mode`** to **`auto`** and **`Sessions Maximum`** to **`-1`** for all five Session Pools.

Tip

For each session pool, -1 specifies no session maximum. Tune each Session Maximum to the maximum number of threads for each pool.

1) Admin Activity

JMS session pool - Configuration Tab

Property	Value
Name	`edxLoggerAdminActivityPool`
Connection Factory	`edx/tcf/log`
Listener Class	`com.edocs.fs.logging.sub.AdminActivityListener`
Acknowledge Mode	`auto`
Sessions Maximum	`-1`

JMS Consumer- Configuration Tab

Property	Value
Name	`edxLoggerAdminActivityConsumer`
Messages Maximum	`10`
Selector	`JMSType='ADM'`
Destination	`edx/jms/log`

2) CSR Activity

JMS session pool- Configuration Tab

Property	Value
Name	`edxLoggerCSRActivityPool`
Connection Factory	`edx/tcf/log`
Listener Class	`com.edocs.fs.logging.sub.CSRActivityListener`
Acknowledge Mode	`auto`
Sessions Maximum	`-1`

JMS consumer- Configuration Tab

Property	Value
Name	`edxLoggerCSRActivityConsumer`
Messages Maximum	`10`
Selector	`JMSType='CSR'`
Destination	`edx/jms/log`

3) Message Log

JMS session pool- Configuration Tab

Property	Value
Name	`edxLoggerMessageLogPool`
Connection Factory	`edx/tcf/log`
Listener Class	`com.edocs.fs.logging.sub.MessageLogListener`
Acknowledge Mode	`auto`
Sessions Maximum	`-1`

JMS consumer- Configuration Tab

Property	Value
Name	`edxLoggerMessageLogConsumer`
Messages Maximum	`10`
Selector	`JMSType='MSG'`
Destination	`edx/jms/log`

4) System Activity

JMS session pool- Configuration Tab

Property	**Value**
Name	`edxLoggerSystemActivityPool`
Connection Factory	`edx/tcf/log`
Listener Class	`com.edocs.fs.logging.sub.SystemActivityListener`
Acknowledge Mode	`auto`
Sessions Maximum	`-1`

JMS consumer

Property	**Value**
Name	`edxLoggerSystemActivityConsumer`
Messages Maximum	`10`
Selector	`JMSType='SYS'`
Destination	`edx/jms/log`

5) UserActivity

JMS session pool

Property	**Value**
Name	`edxLoggerUserActivityPool`
Connection Factory	`edx/tcf/log`
Listener Class	`com.edocs.fs.logging.sub.UserActivityListener`
Acknowledge Mode	`auto`
Sessions Maximum	`-1`

JMS consumer

Property	**Value**
Name	`edxLoggerUserActivityConsumer`
Messages Maximum	`10`
Selector	`JMSType='USER'`
Destination	`edx/jms/log`

Deploying EBM

After configuring your WebLogic domain server, you can deploy the EAR files to the appropriate servers:

- **Application servers**: Deploy the edocs Platform Services (ear-eStatement.ear) and the EBM ears.
- **Database servers**: Deploy edocs Platform Services (ear-eStatement.ear).

The ear files are located at:

Feature	**Location**	**File Name**
Edocs Platform Services	%EBM_HOME%\J2EEApps\weblogic	ear-eStatement.ear
EBM	%EBM_HOME%\J2EEApps\weblogic	EBM Consumer: ear-tbm.ear EBM Business: ear-tbm-b2b.ear
EBM CSR	%EBM_HOME%\J2EEApps\weblogic	ear-tbm-csr.ear

Consult your BEA WebLogic documentation on how to deploy applications.

After successfully deploying the application, you can log into the EBM Command Center:

1. In your browser, point to http://localhost:7001/edocs (where `localhost` is your server name if you are on a different machine). The initial default Administrator ID is **`admin`** and the Password is **`edocs`**.

2. Once in the Command Center, change your password. Click the Help button in the Command Center for information changing passwords.

Starting the Scheduler

You can start the Scheduler from a command line or as a Windows service (as described in the previous chapter). Before starting the Scheduler, check the file to make sure it has the correct JAVA_HOME value.

Edit wl_scheduler.bat

In **`%EDX_HOME%\bin,`** edit *wl_scheduler.bat* to set correct value for **`JAVA_HOME`**.

To start the Scheduler from a Windows command prompt:

1. Open a command prompt window and change directory to the **`bin`** directory of your EBM installation, **`%EDX_HOME%\bin`**.

2. Run the Scheduler command for WebLogic Server (**`wl_scheduler`**), host, and port. This example shows the Scheduler command for WebLogic:

   ```
   C:\> wl_scheduler –start –url t3://localhost:7001
   ```

3. If the Scheduler starts successfully, a start-up message with the name of the log file appears in the command prompt window. **Do not close this window while EBM is running,** as closing it will stop the Scheduler. A log file is created in **`%EDX_HOME%\Logs`**.

4. You can stop the Scheduler by replacing the **`-start`** parameter with the **`-stop`** parameter, or simply by closing the command prompt window.

To start the EBM Scheduler as a Windows Service:

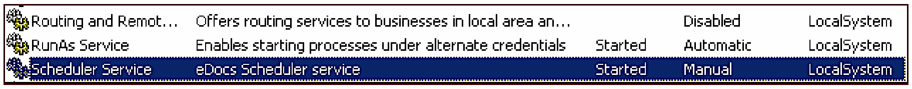
1. Install the Scheduler as a Windows Service if you have not already done so.

2. Start WebLogic Server if it is not already running.

3. From the Start menu, select **Settings**>**Control Panel**.

4. Double-click **Administrative Tools**, then double-click **Services**.

5. Right-click **Scheduler Service** and select **Start**. You may also click the Start icon.



Routing and Remot...	Offers routing services to businesses in local area an...		Disabled	LocalSystem
RunAs Service	Enables starting processes under alternate credentials	Started	Automatic	LocalSystem
Scheduler Service	eDocs Scheduler service	Started	Manual	LocalSystem

6. To start the Scheduler automatically at startup, right-click the service and select **Properties**. From the Startup Type menu, select **Automatic**.

Where to Go From Here

Once you have successfully configured the WebLogic Server and deployed the EBM application, you can proceed to deploying any custom J2EE applications. This requires customizing each web application’s deployment code for your environment and platform. For details, see WebLogic Server documentation at http://bea.com and *Deploying and Customizing J2EE Applications.*

If you want to deploy and configure edocs Payment, proceed to the next chapter.

Appendix A: Uninstalling EBM

Uninstalling EBM

You can uninstall and remove EBM components, deployed J2EE applications, and Windows services using the EBM Uninstaller.

Uninstall EBM from the **database server** first, then the **application server**.

The uninstaller does **not** delete any directories that contain files modified since installation. Instead, it lists these items, which you must then remove manually.

Before uninstalling EBM components, you must:

- Stop your application server.
- Stop your database instance.
- Stop your database server.

To uninstall EBM:

1. Navigate to the **`Uninstall`** folder of your EBM home directory, **`$EBM_HOME.`**
2. Windows users may run the command-line script **`Uninstall_TBM.exe`**, or select Start Menu>Programs>edocs>Uninstall EBM.
3. Click **`Uninstall`**. A second uninstall screen appears showing EBM components being removed from your machine.

   When the uninstaller is finished, a screen appears listing any items that could not be removed.
4. Change the directory to your EBM home directory and manually remove any remaining files and directories as necessary.
5. Click **`Done`** to close the uninstaller.
6. Repeat this procedure on your application server and any other installations.

Uninstalling Windows Services

Tip This section applies to all Windows platforms.

Uninstalling WebLogic Server as a Windows Service

Uninstalling this Windows Service will require that you start and stop WebLogic Server from the command line or the administrative console.

To uninstall WebLogic Server as a Windows Service:

1. Open a Command Prompt window, and change directory to the **`domain`** directory of your application server home directory.

   ```
   C:\> cd %WL_HOME%\config\mydomain
   ```

2. Uninstall WebLogic Server as a Windows Service with the **`uninstallNtService`** command:

   ```
   C:\> uninstallNtService.cmd
   ```

   You do not have to specify the WebLogic service name on the command line, because it is named in **`uninstallNtService.com`**.

Uninstalling the EBM Scheduler as a Windows Service

Uninstalling this Windows Service will require that you start and stop the Scheduler from the command line.

To uninstall the Scheduler as a Windows Service:

1. Open a command prompt window, and change directory to the **`\bin`** directory of your EBM home directory.

   ```
   C:\> cd %EDX_HOME%\bin
   ```

2. Uninstall the Scheduler as a Windows Service with the **`–remove`** command:

   ```
   C:\> Schedulersvc –remove
   ```

Uninstalling Payment

Removing the Payment Database

Follow the steps below to remove the Payment database tables and indexes. The process involves running an SQL script as the as the owner of the database. You should be aware that this procedure **completely** removes the payment database elements and should be used with care.

Caution

If you wish to remove the Payment database tables and indexes, you must do that before removing the Payment database package.

To remove the Payment database for MSSQL:

1. Change your working directory to the *%PAYMENT_HOME\db\mssql* directory.
2. Run the file *drop_payment_db.bat*.

Index