



Enterprise Exception Handling

Getting Started Guide

Arunava Majumdar
Sr. IT Specialist
Focused Technology Practice
(Connectivity/BPM)
IBM Corporation.

Vineet Gupta
IBM Certified IT Specialist
Application Integration and Middleware
IBM Software Services for Websphere
IBM India Pvt. Ltd.

WebSphere. software

Contents

<i>Modification History</i>	5
<i>Legal Disclaimer</i>	6
<i>Acknowledgement</i>	8
<i>Scope of the Document</i>	9
1. Core EEH Setup	10
1.1. Setting up EEH_PATH.....	10
1.2. Setting up the Database.....	11
1.2.1. Setting up DB2.....	11
1.2.2. Setting up Derby.....	12
1.2.3. Setting up Solid DB.....	13
1.3. Setting up WebSphere MQ.....	15
1.3.1. Starting EEH Daemon for DB2.....	15
1.3.2. Starting EEH Daemon for Derby.....	15
1.3.3. Starting EEH Daemon for Derby.....	16
1.3.4. Shutting down EEH Daemon.....	17
2. EEH Eclipse Editor Installation	18
3. Running EEH Sample	26
3.1. Importing Exception Database files.....	26
3.2. Connecting to the Exception Database.....	29
3.2.1. Connecting to DB2.....	30
3.2.2. Connecting to Derby.....	32
3.2.3. Connecting to Solid DB.....	33
3.3. Importing Sample Exception Catalog.....	34
3.4. Adding the Sample Application.....	36
3.5. Running the Sample Application.....	42
3.6. Viewing Exceptions.....	43
3.7. Custom Renderers (Preview).....	48
4. Running EEH Web Analyzer	50
4.1. Installing on Tomcat 7.0.....	50
4.2. Installing on WebSphere® Application Server 8.0.....	55
5. Troubleshooting Tips	79
6. Service Offering from IBM	80

Table of Figures

Figure 1 - Solid DB Shortcut	13
Figure 2 - Solid DB startup splash.....	14
Figure 3 – Application running on the task bar	14
Figure 4 – EEH Daemon Started	16
Figure 5 – EEH Daemon Shutdown	17
Figure 6 – Eclipse 3.2 Splash.....	18
Figure 7 – Installation Menu.....	18
Figure 8 – Installation Wizard: Step 1	19
Figure 9 – Installation Wizard: Step 2	19
Figure 10 – Installation Wizard: Step 2.1: Selecting EEH Site	20
Figure 11 – Installation Wizard: Step 2.2: Name the EEH Site	20
Figure 12 – Installation Wizard: Step 2.3: Local Site added	21
Figure 13 – Installation Wizard: Step 3	22
Figure 14 – Installation Wizard: Step 4	23
Figure 15 – Installation Wizard: Step 5	24
Figure 16 – Installation Wizard: Step 6	25
Figure 17 – Installation Wizard: Step 7	25
Figure 18 – Import supplied sample eclipse project	26
Figure 19 – Select Exception Database project	27
Figure 20 – Select the ExcDB project	27
Figure 21 - Imported sample ExcDB project in the workspace	28
Figure 22 – Exception Database Editor.....	29
Figure 23 – Add DB2 JDBC jars and Category – Step I	30
Figure 24 – Add DB2 JDBC jars and Category – Step II.....	30
Figure 25 – Add DB2 JDBC jars and Category – Step III.....	30
Figure 26 – Add DB2 JDBC jars and Category – Step IV.....	31
Figure 27 – Add DB2 JDBC jars and Category – Step V	31
Figure 28 – Connected to DB2 database	31
Figure 29 – Add Derby JDBC jars and Category – Step I	32
Figure 30 – Add Derby JDBC jars and Category – Step II	32
Figure 31 – Connected to Derby database.....	32
Figure 32 – Add SolidDB JDBC jars and Category – Step I.....	33
Figure 33 – Add SolidDB JDBC jars and Category – Step II.....	33
Figure 34 – Connected to SolidDB database	33
Figure 35 – Import the Exception Catalog	34
Figure 36 – Select the sample Exception Catalog.....	34
Figure 37 – Imported Exception Catalog	35
Figure 38 – Add Application	36
Figure 39 – Add Application Wizard	36
Figure 40 – Added Application.....	37
Figure 41 – Add Application Header	37
Figure 42 – Add Application Header Wizard	38
Figure 43 – Added Application Header	38
Figure 44 – Add Application Header Wizard for adding a second application header.....	39
Figure 45 - Second application header added	39
Figure 46 – Associating Application Header with Application.....	40
Figure 47 – Application Headers associated with the Application.....	40
Figure 48 – Associating Exception Catalog with the Application	41
Figure 49 – Exception Catalog mapped to the Application	41
Figure 50 - Sample Application Running.....	42
Figure 51 – Exception Database Editor: Analyzer page.....	43
Figure 52 – Exception Page: Exception Details	44
Figure 53 – Exception Page: Exception Dump.....	44
Figure 54 – Exception Page: Exception Dump: Hex Dump Renderer	45
Figure 55 – Exception Page: Exception Data.....	45
Figure 56 – Exception Page: Exception Data: Hex Data Renderer	46
Figure 57 – Exception Page: Stack Trace	46
Figure 58 – Exception Page: Closing	47
Figure 59 – Add Renderer Library	48
Figure 60 - Select Renderer jar.....	48
Figure 61 – Added Renderer Library	49
Figure 62 – EEH Web Analyzer	51
Figure 63 – EEH Web Analyzer: Exception Details.....	52

Figure 64 – EEH Web Analyzer: Exception Dump.....	52
Figure 65 – EEH Web Analyzer: Exception Dump (Hex).....	53
Figure 66 – EEH Web Analyzer: Exception Data.....	53
Figure 67 – EEH Web Analyzer: Call Stack and Exception Stack Trace.....	54
Figure 68 – EEH Web Analyzer: Data Editing and Re-queuing History	54
Figure 69 – Start Application Server.....	55
Figure 70 - Administrative Console.....	55
Figure 71 - Application Server Log in	55
Figure 72 – WebSphere® Application Server Welcome screen.....	56
Figure 73 - Application Server menu option.....	56
Figure 74 - Define Application Server	56
Figure 75 - JDBC Resource Definition menu option	57
Figure 76 - Create New JDBC Provider	57
Figure 77 - Create New JDBC Provider: Step 1	58
Figure 78 – Create New JDBC Provider: Step 2.....	59
Figure 79 – Create New JDBC Provider: Step 3.....	60
Figure 80 – JDBC Provider created.....	60
Figure 81 - Save JDBC Provider configuration.....	60
Figure 82 – Configure Data Sources.....	61
Figure 83 – Create New Data Source definition.....	61
Figure 84 – Create New Data Source: Step 1.....	62
Figure 85 – Create New Data Source: Step 2.....	62
Figure 86 – Create New Data Source: Step 3.....	63
Figure 87 88 – Create New Data Source: Step 4.....	63
Figure 89 – Save Data Source definition.....	64
Figure 90 – Add Security information to Data Source	65
Figure 91 - New JAAS definition.....	66
Figure 92 – Database connection User Id and Password	66
Figure 93 – Save JAAS definition	67
Figure 94 - Navigate back to Data Sources.....	67
Figure 95 - Assign the defined Security definition	68
Figure 96 - Apply and Save the Data Source definition.....	69
Figure 97 - Test the Data Source connection.....	69
Figure 98 - Install the EEH Web Analyzer	70
Figure 99 – Install Enterprise Application	70
Figure 100 – EEH Web Analyzer install: War location	71
Figure 101 – EEH Web Analyzer install: Fast Path.....	71
Figure 102 – EEH Web Analyzer install: Step 1	72
Figure 103 – EEH Web Analyzer install: Step 2	73
Figure 104 – EEH Web Analyzer install: Step 3: Browse	73
Figure 105 – EEH Web Analyzer install: Step 3: Select the Target JNDI.....	74
Figure 106 – EEH Web Analyzer install: Step 3: Database JNDI selected.....	74
Figure 107 – EEH Web Analyzer install: Step 4	75
Figure 108 – EEH Web Analyzer install: Step 5	75
Figure 109 – EEH Web Analyzer install: Step 6	76
Figure 110 – Save the configuration	77
Figure 111 – Review that the Application is started.....	78
Figure 112 – EEH Web Analyzer	78

Modification History

Date	Version	Author	Description
05/01/2012	1.0.0	Arunava Majumdar, Vineet Gupta	Final release for Support Pac id08

Legal Disclaimer:

Information provided has been developed as a collection of the experiences of technical services professionals over a wide variety of customer and internal IBM environments, and may be limited in application to those specific hardware and software products and levels

The information contained in this document has not been submitted to any formal IBM test. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk, and in some environments may not achieve all the benefits described.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of this publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

IBM may not offer the products, services, or feature discussed in this document in all countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service. IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785 U.S.A.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials of this IBM product and use of those Web sites is at your own risk.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products. All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice and represent goals and objectives only.

All prices shown are IBM's suggested list prices and are subject to change without notice. Dealer prices may vary.

Any performance data contained in this document was determined in a controlled environment. Therefore the results obtained in other operating environments may vary significantly. Some measurements quoted in this document may have been made on development-level systems. There is no guarantee that these measurements will be the same on generally available systems. Some measurements quoted in the document may have been estimated through extrapolation. Actual results may vary. Users of this presentation should verify the applicable for their specific environment.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrates programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purpose of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platforms for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

In memory of my sister Anusree Majumdar

Acknowledgement:

I would like to thank my co-author Vineet Gupta for joining the EEH Asset team and add the Web Analyzer to the package. This provides a useful browser-based interface to search for exceptions in the database.

I would also like to thank Suman Kalia, Anthony O'Dowd and Eugene Kharmalov from the development team for their Support and technical help.

I would like to extend my special thanks to Sandra Raleigh and Guy Hochstetler for their continued support on the Exception Handler publication and setting up of the EEH Community and Services Offering.

And last but not the least, I would like to acknowledge my wife Sonia and my son Abhi's support during the very difficult times as this version of the Exception Handler was being published.

I also acknowledge the Open Source contributions on the following packages used in the delivery of the product.

<p>This product includes software developed by the DOM4J Project (http://www.dom4j.org/). This product includes software developed by the SAXPath Project (http://www.saxpath.org/). This product includes software developed by the JAXEN Project (http://jaxen.codehaus.org/). This product includes software developed by the JAXEN Project (http://logging.apache.org/log4j/1.2/).</p>
--

Scope of the Document:

The scope of the document is limited to providing a brief guideline to setup Enterprise Exception Handling in the least amount of time for the organization. The authors have attempted to limit longer paragraphs and provided spaces for user notes to make this a working cookbook for users to refer to in the future. There is not attempt to duplicate product documentation in this guide. Feedback will be appreciated.

1. Core EEH Setup:

To get started with Enterprise Exception Handling, there are only three basic requirements:

- Set the environment variable EEH_PATH to the directory where the package is unzipped
- Setup a database and create the EEH schema
- Setup MQ Middleware to communicate between the applications and the Exception Handling Daemon

1.1. *Setting up EEH_PATH*

Use forward slash “/” for path separator and no slashes at the end of the path to point to the installation directory of EEH, e.g. `EEH_PATH=C:/IBM/Asset/EEH`.

1.2. Setting up the Database

EEH package provides DDL scripts to setup the database for the EEH Schema based on the specifications. The scripts are located for several databases in the [EEH_PATH/ddl](#) directory.

1.2.1. Setting up DB2

Open DB2 prompt
Change directory to [EEH_PATH/ddl/DB2](#)

```
db2 create db EEH01
```

Run the DDL script [DB2.exccat.cmd](#)

1.2.2. Setting up Derby

Start Interactive SQL for Derby

```
set DERBY_HOME=Z:/_Backup/Assets/Assets.TEST/install/derby-10.5.3.0
java -jar %DERBY_HOME%/lib/derbyrun.jar ij
```

```
connect 'jdbc:derby:eeh01;create=true';

run 'Derby.exccat.sql';
disconnect;
exit;
```

1.2.3. Setting up Solid DB

From the Start IBM solidDB shortcut open the solidDB database in Z:_Backup\Assets\Assets.TEST\database\solidDB\eeh01

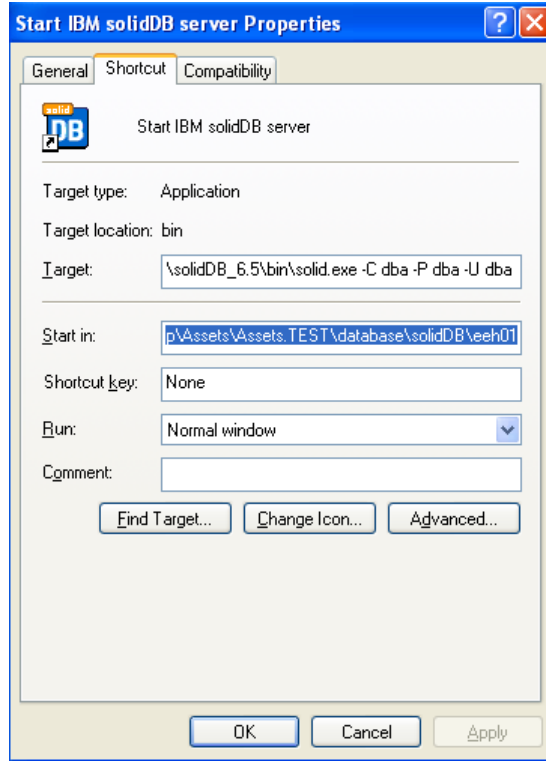


Figure 1 - Solid DB Shortcut

If the database does not exist this creates the database in the specified path.



Figure 2 - Solid DB startup splash

The startup splash is shown while starting the solid DB server and then the application is minimized to the taskbar as shown in the following figure.



Figure 3 – Application running on the task bar

1.3. Setting up WebSphere MQ

The MQSC scripts for setting up the MQ queues is provided in the samples directory. Change directory to [EEH_PATH/samp/mqsc](#)

`runmqsc QM.EEH.01 < queues.mqsc`

The names of the queues may be changed based on the implementation. Below are the definitions of the least number of queues for a basic EEH implementation.

```
def ql('Q.EEH.FAIL') bothresh(3) boqname('Q.EEH.FAIL.BAK') replace
def ql('Q.EEH.FAIL.BAK') replace
def ql('Q.EEH.CMD') replace
def ql('Q.EEH.EXPIRED') replace
```

The exception handler daemon can now be started. Batch files are provided to facilitate the user to start the EEH daemon. Copy and customize the sample batch file for the respective databases from the to [EEH_PATH/bin/win](#) directory.

1.3.1. Starting EEH Daemon for DB2

Edit [daemon.bat](#)

JDBC Jar: [db2jcc.jar](#); [db2jcc_license_cisuz.jar](#)

```
call bin/win/exchd.bat -drv "com.ibm.db2.jcc.DB2Driver"
-url "jdbc:db2://localhost:50001/EEH01"
-usr pi-eadmin -pwd passw0rd
-q Q.EEH.FAIL -m QM.EEH.01 -cmdQ Q.EEH.CMD
```

1.3.2. Starting EEH Daemon for Derby

Edit [daemonDerby.bat](#)

JDBC Jar: [derbyclient.jar](#)

```
call bin/win/exchd.bat -drv "org.apache.derby.jdbc.ClientDriver"
-url "jdbc:derby://localhost:1527/EEH01"
-usr assetadm -pwd passw0rd -dbprefix app
-q Q.EEH.FAIL -m QM.EEH.01 -cmdQ Q.EEH.CMD
```

1.3.3. Starting EEH Daemon for Derby

Edit [daemonSolidDB.bat](#)

JDBC Jar: [SolidDriver2.0.jar](#)

```
call bin/win/exchd.bat -drv "solid.jdbc.SolidDriver"  
-url "jdbc:solid://localhost:2315"  
-usr dba -pwd dba -dbprefix dba  
-q Q.EEH.FAIL -m QM.EEH.01 -cmdQ Q.EEH.CMD
```

The following figure shows that the EEH daemon was started normally.

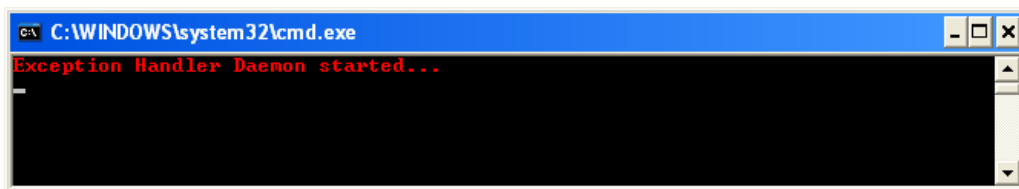


Figure 4 – EEH Daemon Started

1.3.4. Shutting down EEH Daemon

Edit [shutdown.bat](#)

```
call bin/win/shutdown.bat -m QM.EEH.01 -cmdQ Q.EEH.CMD -shutdown
```

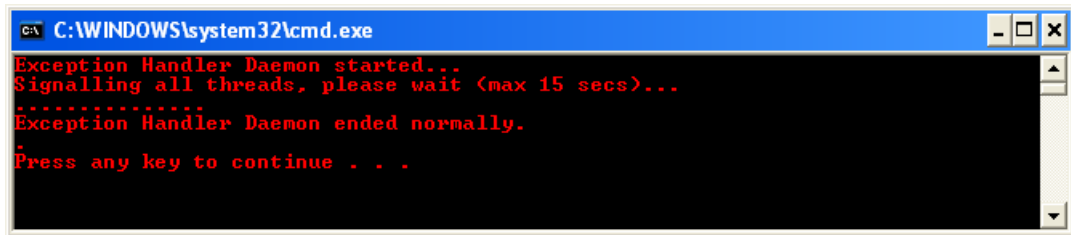


Figure 5 – EEH Daemon Shutdown

2. EEH Eclipse Editor Installation:

The eclipse editor is packaged as a eclipse site that provides 1 feature and 2 plug-ins:

Feature: [com.ibm.mq.exc.feature_1.0.0.jar](#)
Plug-in: [com.ibm.mq.exc.plugin_1.0.0.jar](#)
[com.ibm.mq.exc.plugin.service_1.0.0.jar](#)

The eclipse site is packaged under [EEH_PATH/eclipse/site](#)

Minimum requirement for the plug-in is version 3.2.2 of eclipse and Java 1.5.

The installation steps are illustrated in the following figures. It demonstrates installation on base eclipse 3.2. Other eclipse platforms would have a similar installation.

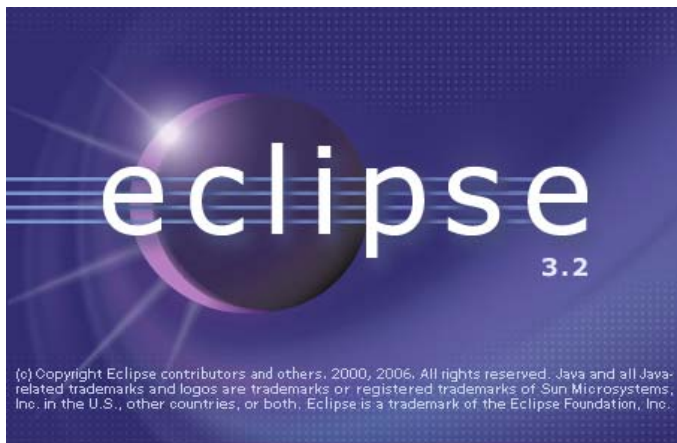


Figure 6 – Eclipse 3.2 Splash

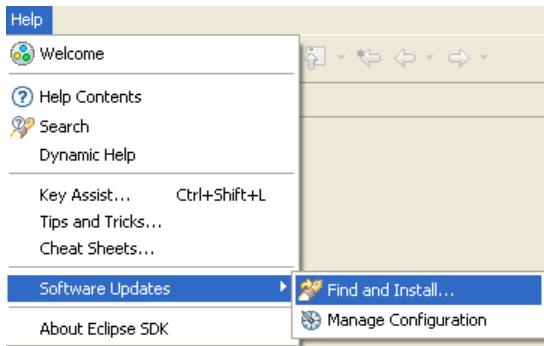


Figure 7 – Installation Menu

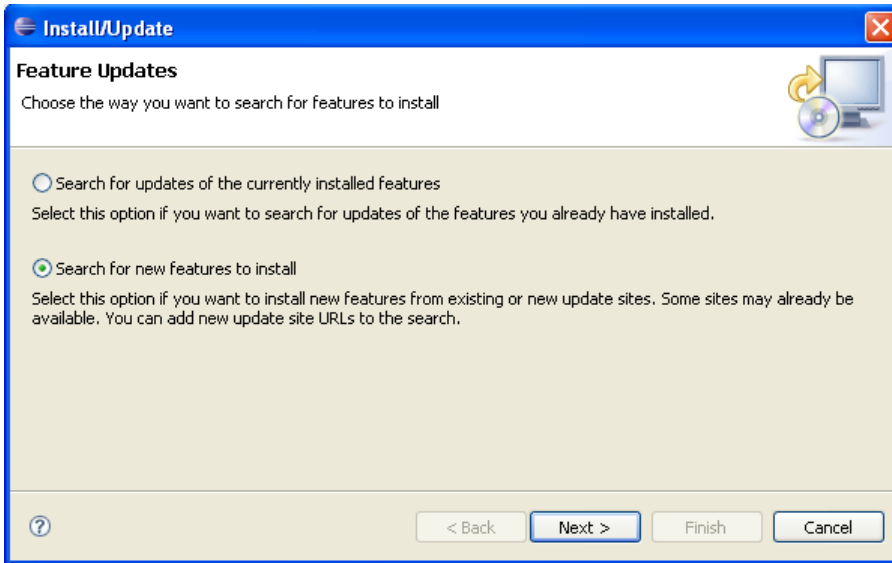


Figure 8 – Installation Wizard: Step 1

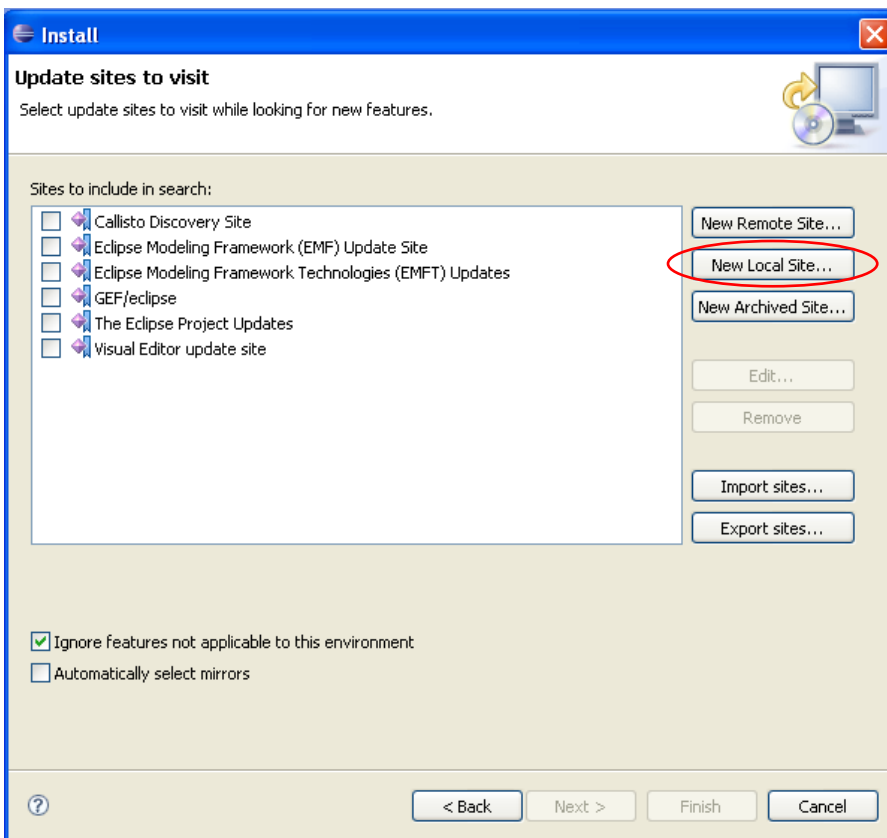


Figure 9 – Installation Wizard: Step 2

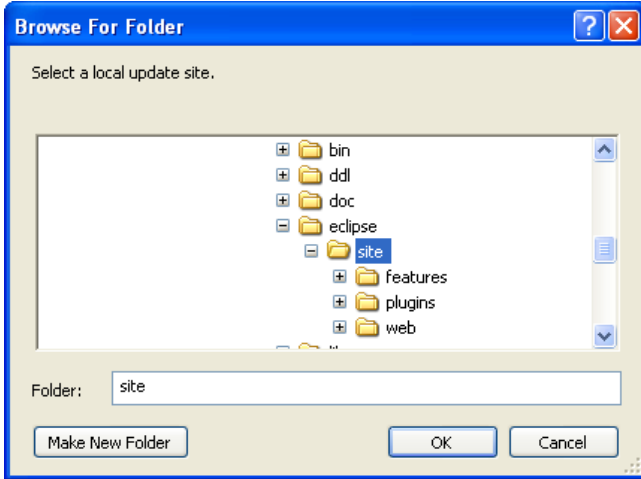


Figure 10 – Installation Wizard: Step 2.1: Selecting EEH Site

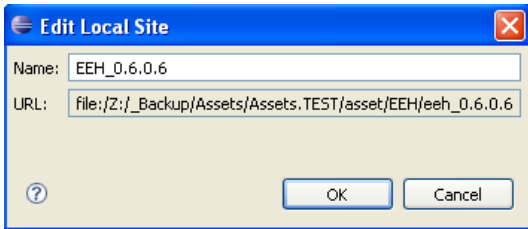


Figure 11 – Installation Wizard: Step 2.2: Name the EEH Site

[N.B. Recommendation: do not use the version appended to EEH unless you have multiple installations of EEH unless working with simultaneously. This helps in updating the package in a common EEH_PATH.]

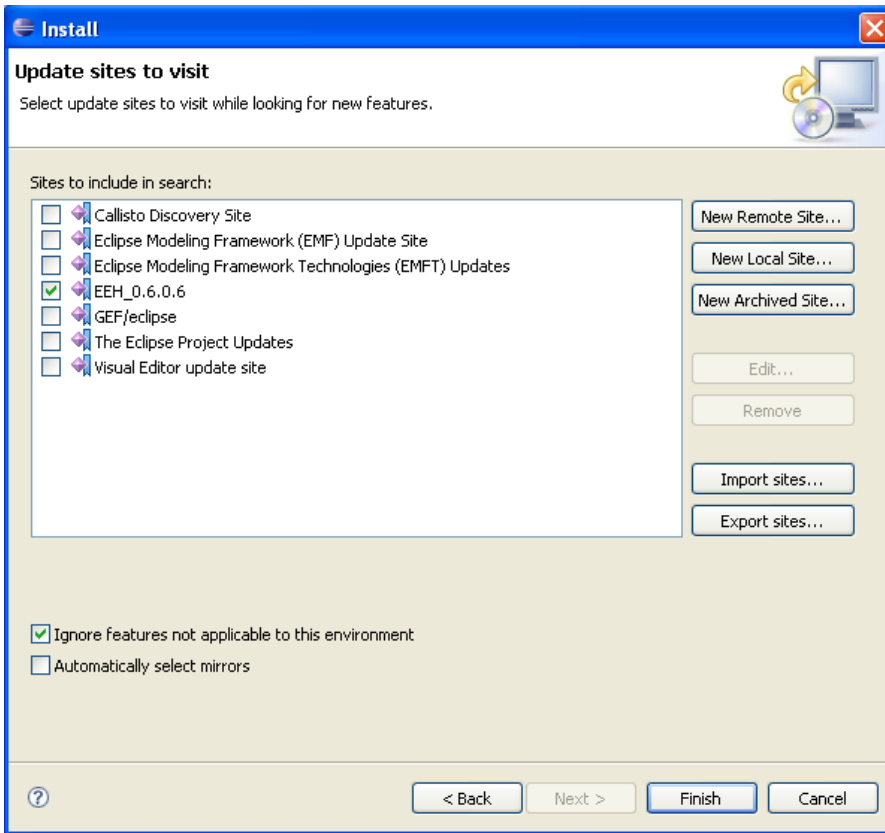


Figure 12 – Installation Wizard: Step 2.3: Local Site added

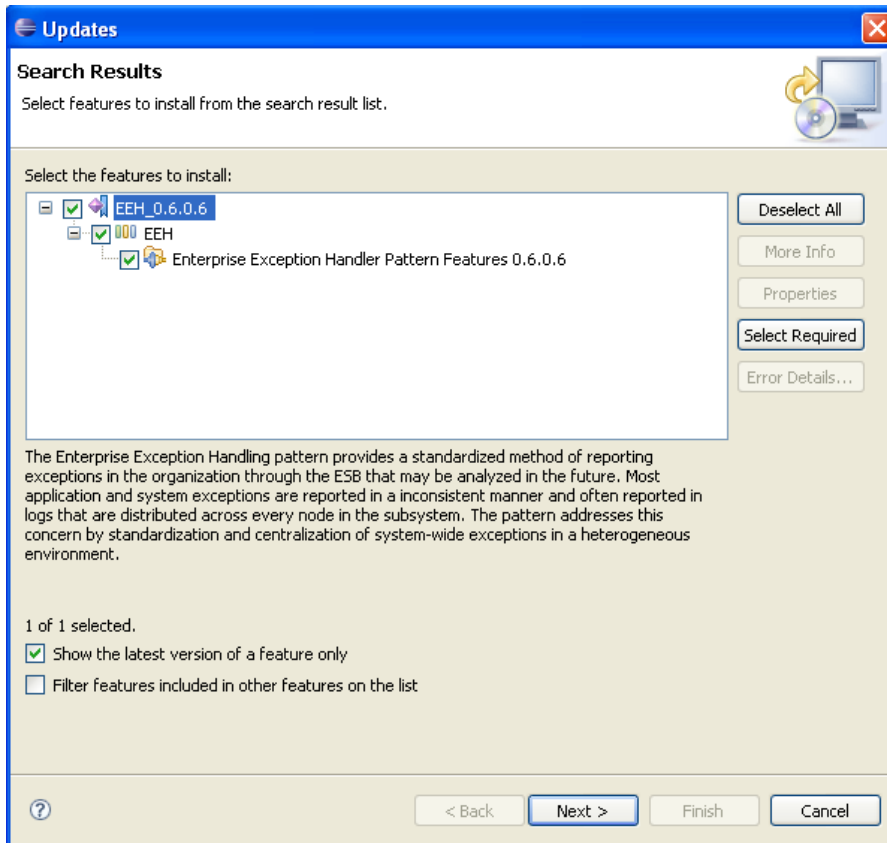


Figure 13 – Installation Wizard: Step 3

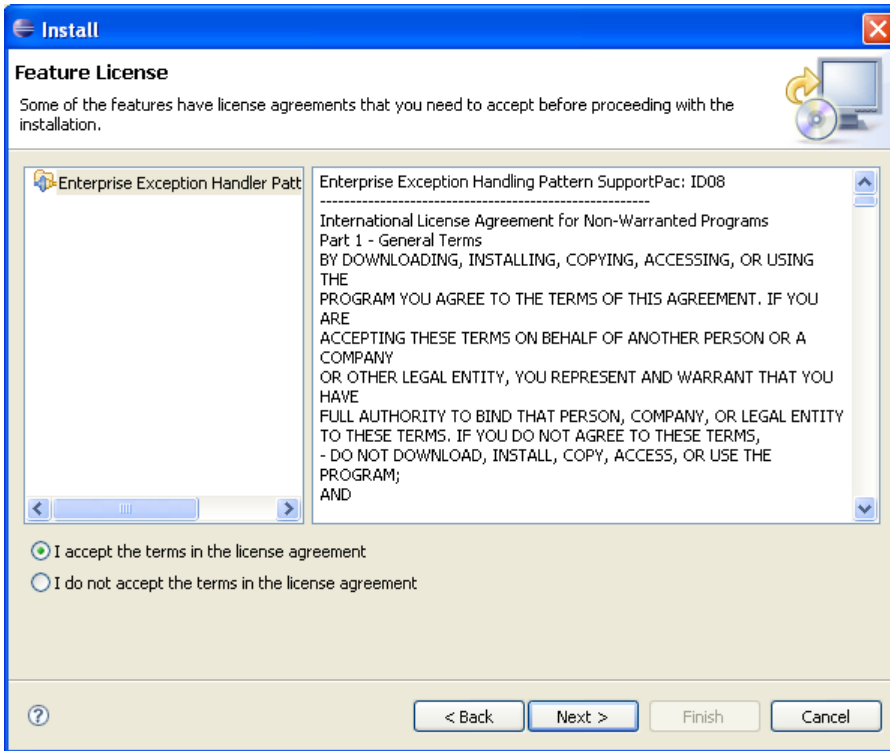


Figure 14 – Installation Wizard: Step 4

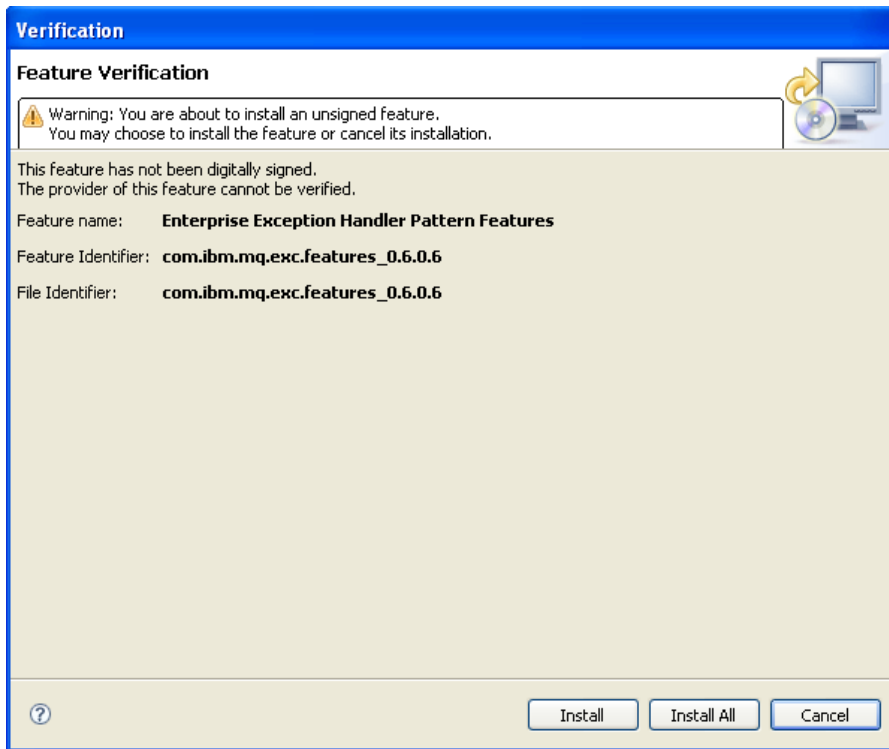


Figure 16 – Installation Wizard: Step 6

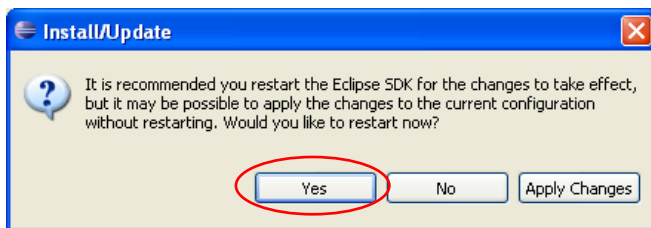


Figure 17 – Installation Wizard: Step 7

[N.B. Restart of the eclipse session is necessary for the plug-ins to take effect.]

3. Running EEH Sample:

To run the sample application and view the exceptions in the database follow the following steps:

- Import the Exception Database files
- Connect to the database
- Import the Sample Exception Catalog
- Add the application and headers and associations
- Run the Sample Application
- View the exceptions in the database

3.1. Importing Exception Database files

Sample exception database editor files are provided as part of a sample eclipse project. The files may be copied to any project. The location is [EEH_PATH/samp/excdb](#).

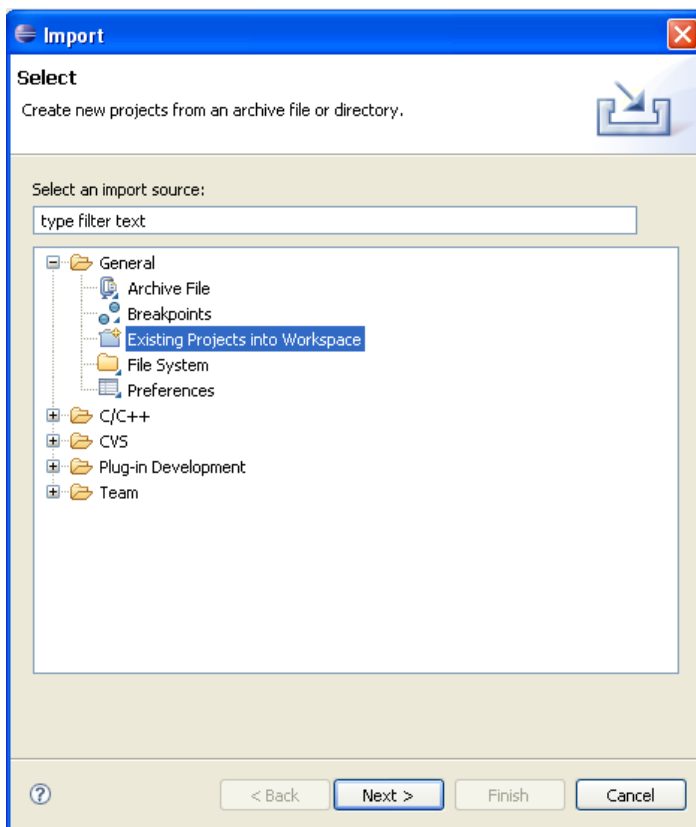


Figure 18 – Import supplied sample eclipse project

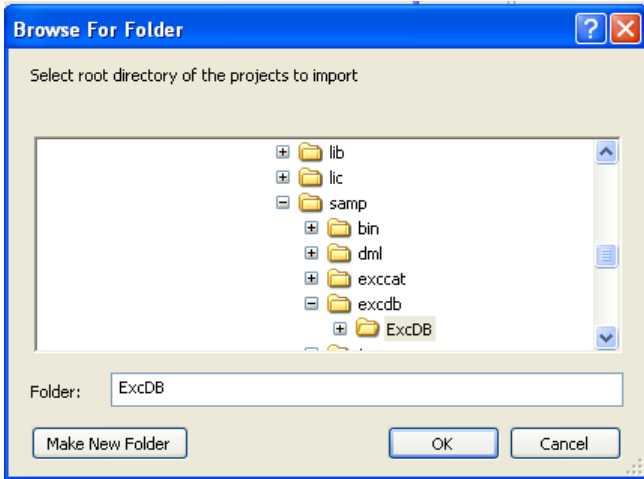


Figure 19 – Select Exception Database project

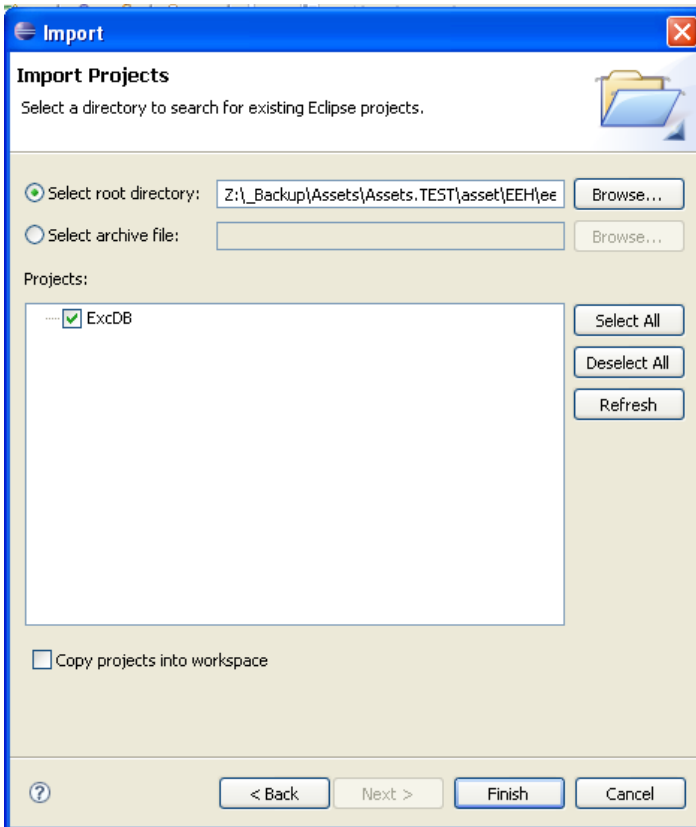


Figure 20 – Select the ExcDB project

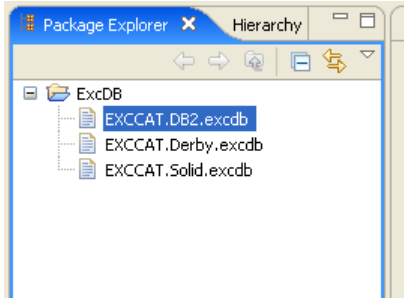


Figure 21 - Imported sample ExcDB project in the workspace

After importing the project all the **excdb** files should be visible under the project. They provide example to getting started with the respective databases. The files may be copied into any location within the workspace and has no requirements to be added to a specific type of project.

3.2. Connecting to the Exception Database

The Exception Database Editor is opened by double clicking on the **excdb** file.

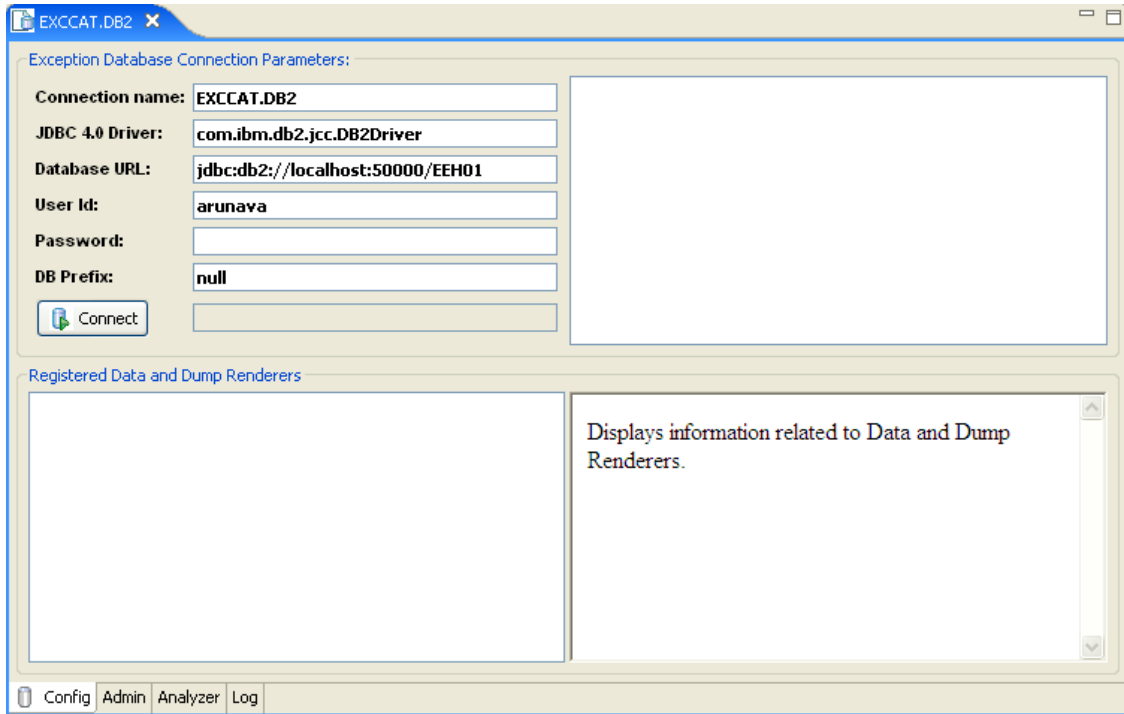


Figure 22 – Exception Database Editor

3.2.1. Connecting to DB2

For connecting to DB2, add the DB2 JDBC jars from the editor. This updates the EEH service plug-in [com.ibm.mq.exc.plugin.service_1.0.0.jar](#) from inside the editor and adds the database jars inside it. This needs to be done once per installation and provides portability for the eclipse installation to be moved or cloned to another location.

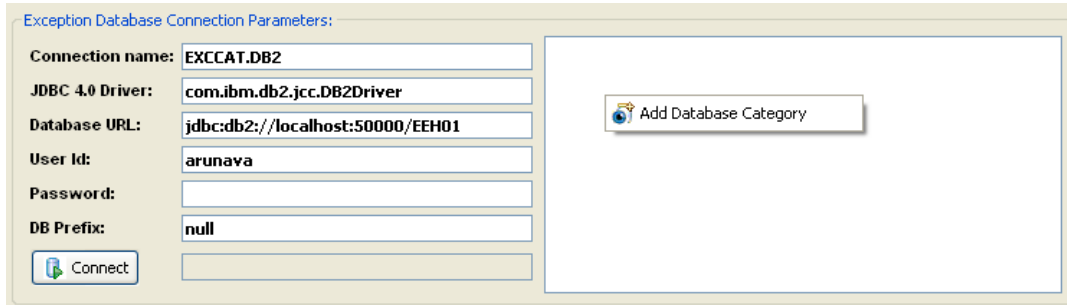


Figure 23 – Add DB2 JDBC jars and Category – Step I

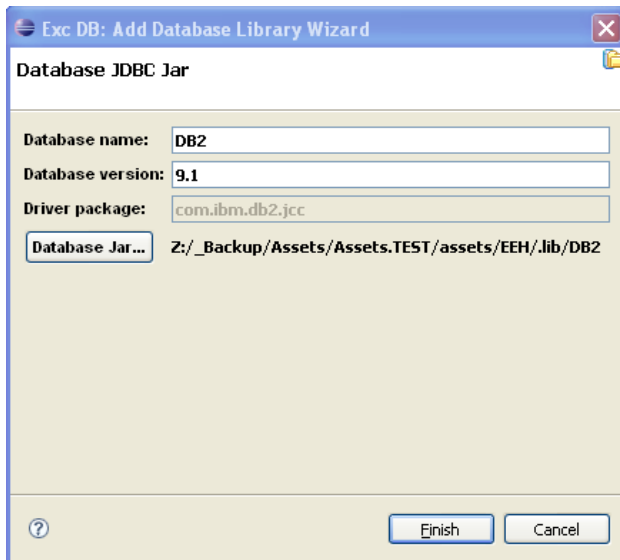


Figure 24 – Add DB2 JDBC jars and Category – Step II

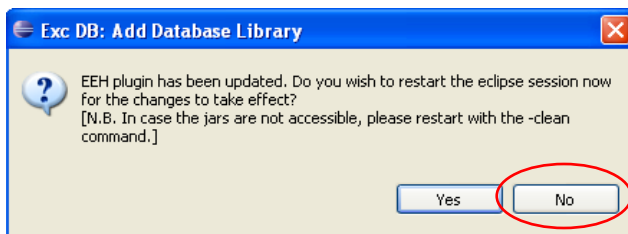


Figure 25 – Add DB2 JDBC jars and Category – Step III

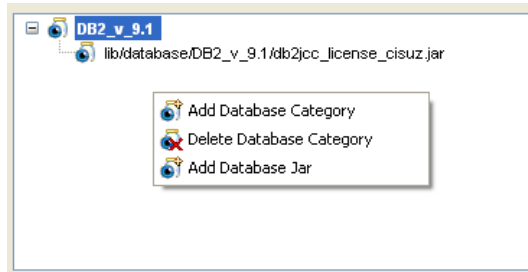


Figure 26 – Add DB2 JDBC jars and Category – Step IV

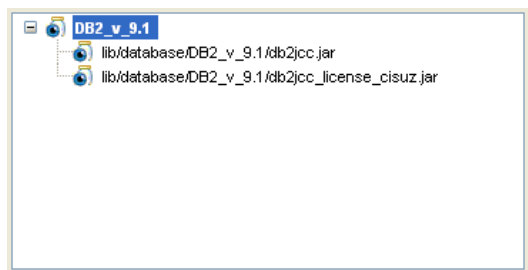


Figure 27 – Add DB2 JDBC jars and Category – Step V

After adding the DB2 jars please **restart** the eclipse session and connect to the EEH database.

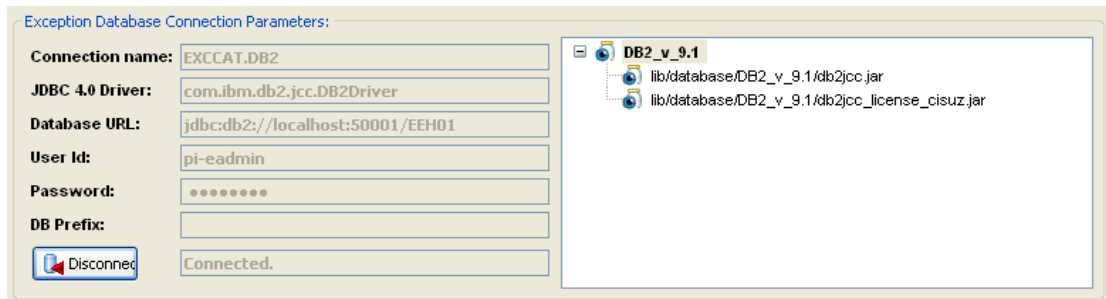


Figure 28 – Connected to DB2 database

3.2.2. Connecting to Derby

Add the Derby database JDBC jars similar to the DB2 explained in the previous section.

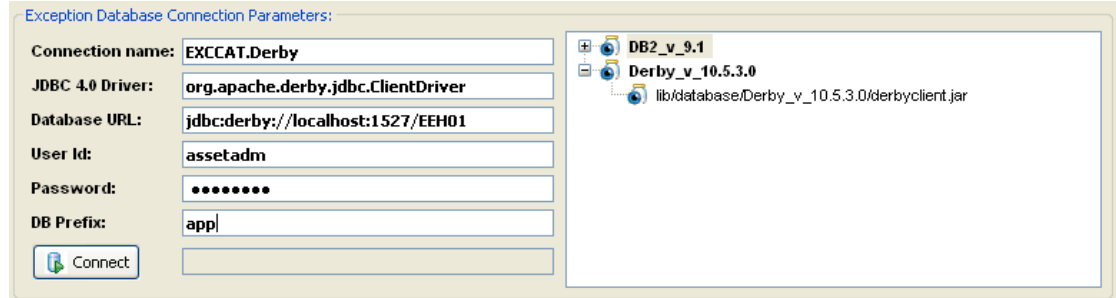


Figure 29 – Add Derby JDBC jars and Category – Step I

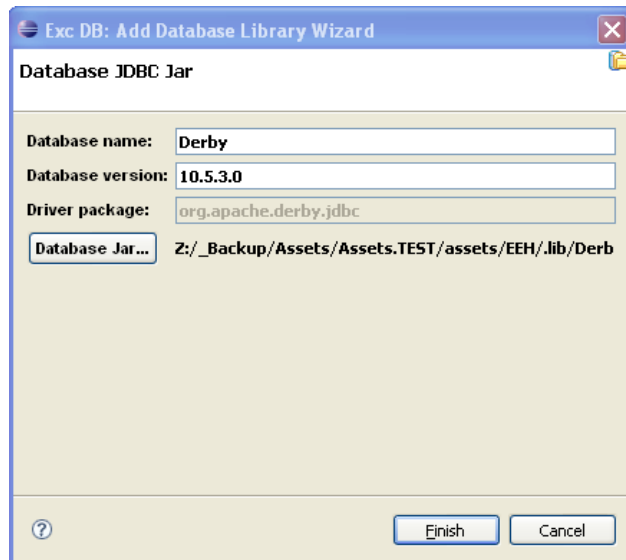


Figure 30 – Add Derby JDBC jars and Category – Step II

After adding the Derby jars please **restart** the eclipse session and connect to the EEH database.

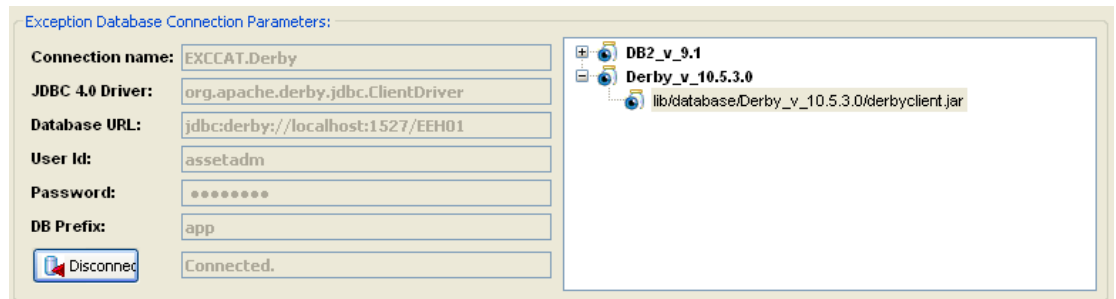


Figure 31 – Connected to Derby database

3.2.3. Connecting to Solid DB

Add the Solid DB database JDBC jars similar to the DB2 explained in the previous section.

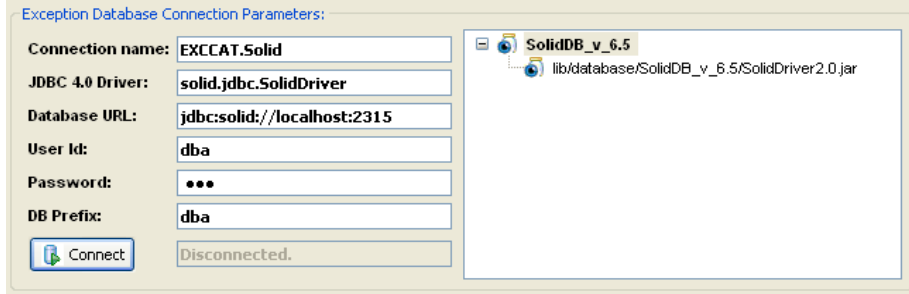


Figure 32 – Add SolidDB JDBC jars and Category – Step I

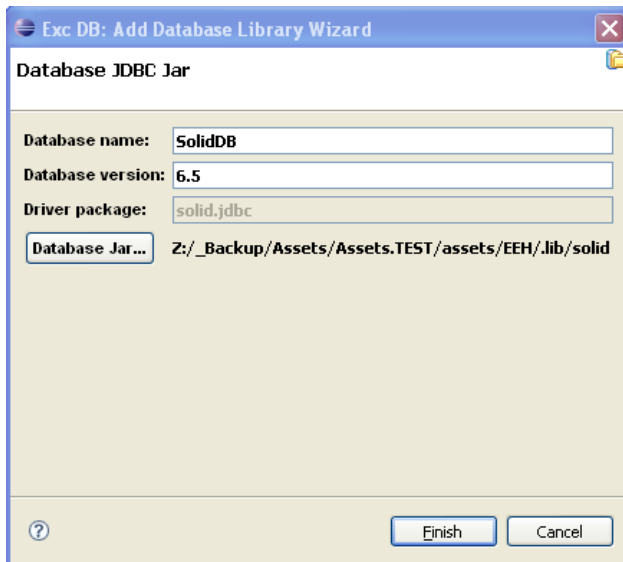


Figure 33 – Add SolidDB JDBC jars and Category – Step II

After adding the Solid DB jars please **restart** the eclipse session and connect to the EEH database.

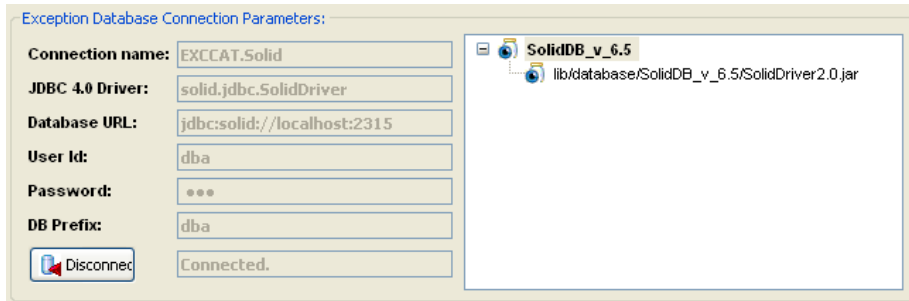


Figure 34 – Connected to SolidDB database

3.3. Importing Sample Exception Catalog

The sample Exception Catalog may be imported into the database only when actively connected. The option is present on the *Admin* page.

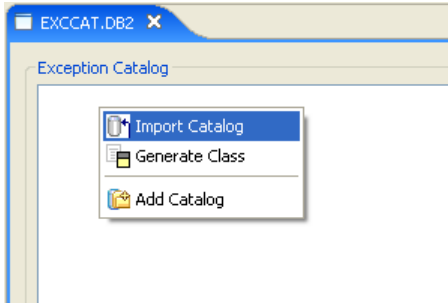


Figure 35 – Import the Exception Catalog

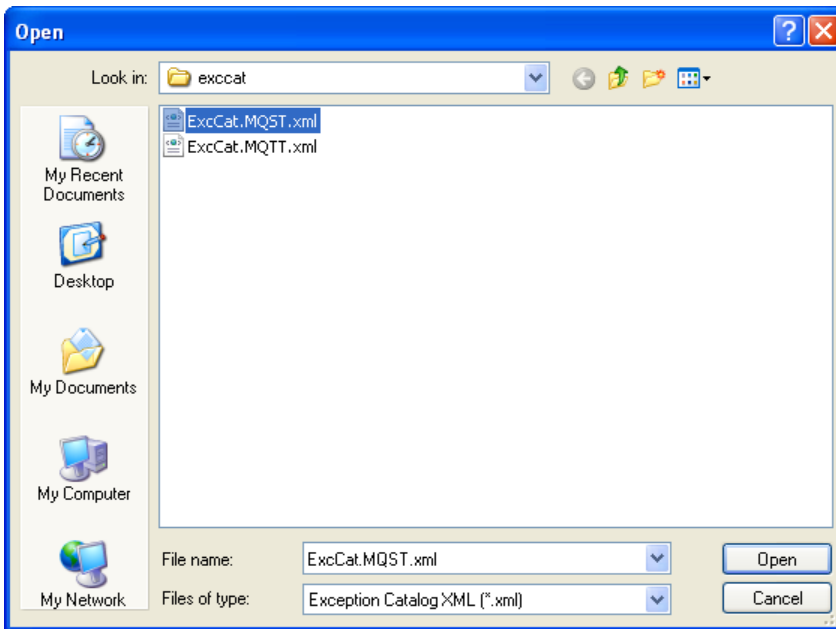


Figure 36 – Select the sample Exception Catalog

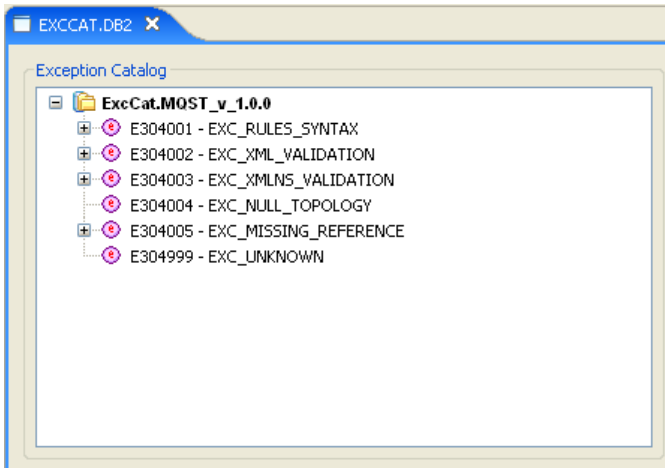


Figure 37 – Imported Exception Catalog

3.4. Adding the Sample Application

Add the sample application to the database. Registering application in the database is important to trace exceptions and to catch unwarranted use of catalog without registration leading to erroneous exception information during analysis.

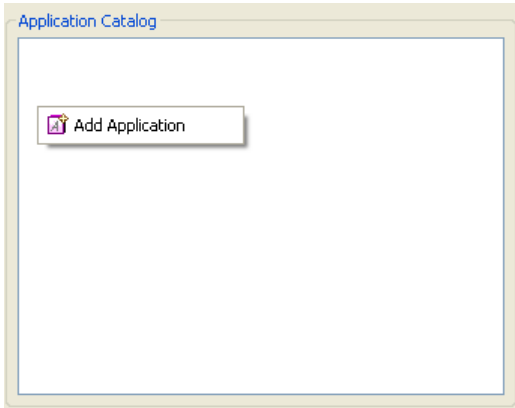


Figure 38 – Add Application

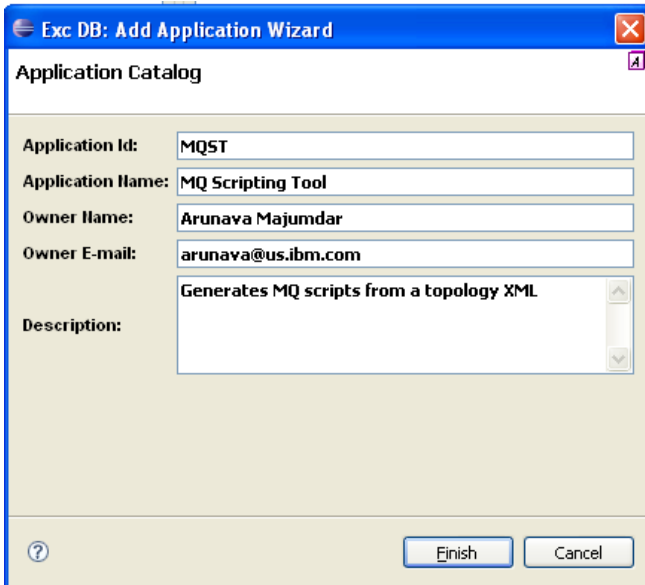


Figure 39 – Add Application Wizard

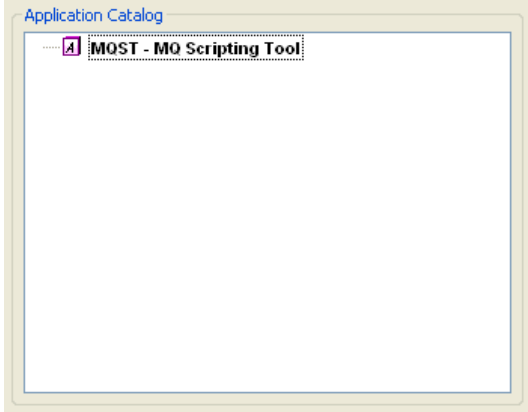


Figure 40 – Added Application

Application Headers are loosely couple with applications and have a many-to-many relationship for carrying important application information, state, etc. at the moment of the failure. The sample application uses two such headers for demonstration.

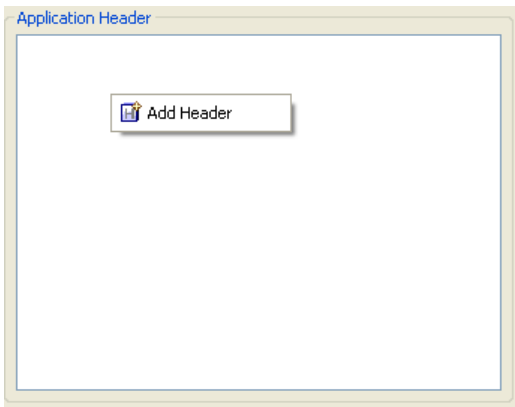


Figure 41 – Add Application Header

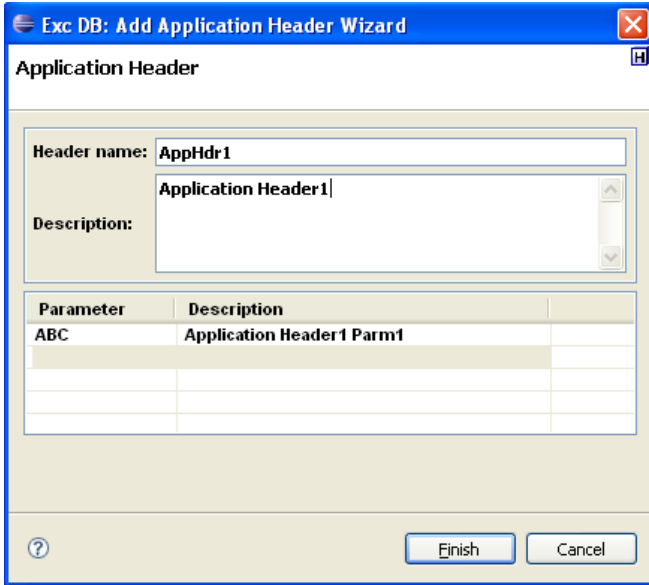


Figure 42 – Add Application Header Wizard

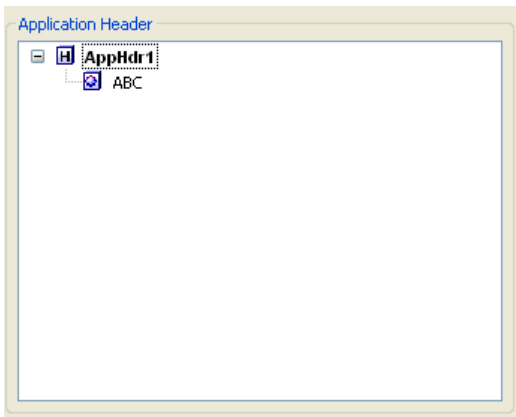


Figure 43 – Added Application Header

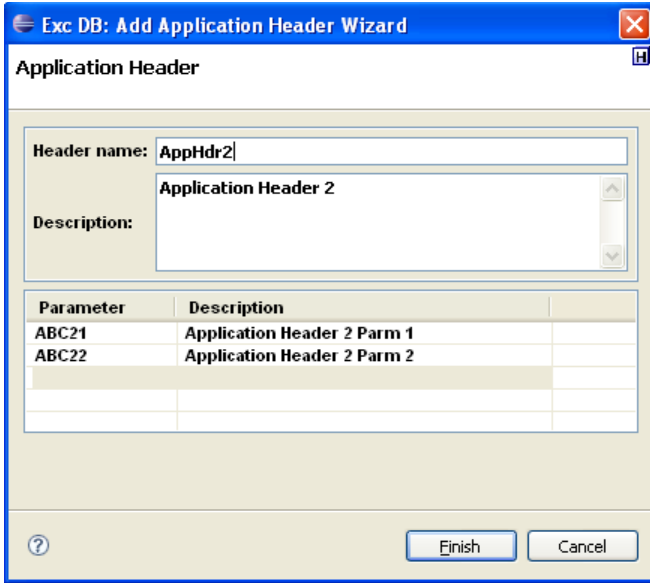


Figure 44 – Add Application Header Wizard for adding a second application header

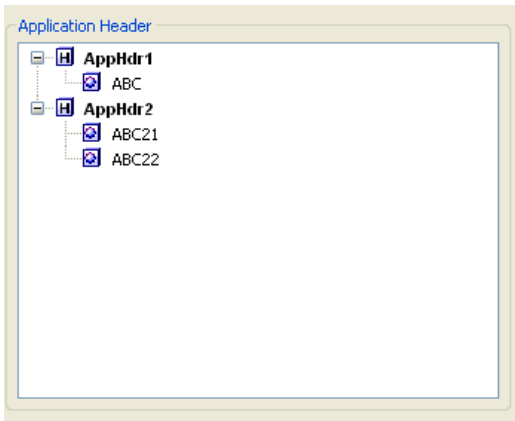


Figure 45 - Second application header added

The applications header must be associated to the application. Drag the header to the application to add the association. The association of the application headers is illustrated in the tree at the bottom.

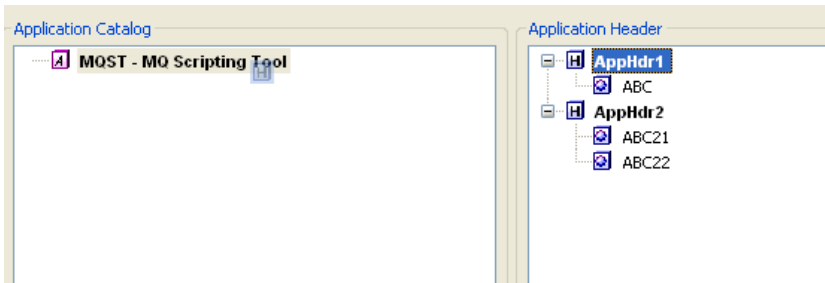


Figure 46 – Associating Application Header with Application

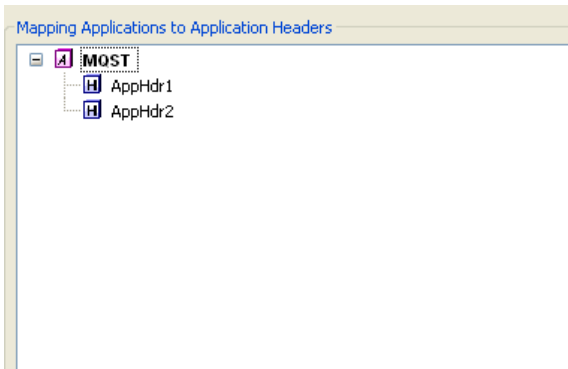


Figure 47 – Application Headers associated with the Application

The exception catalog, now, must be associated with the application. Drag the exception catalog to the application to create the association. The associations between exceptions and applications are illustrated in the tree below. Complete catalogs as well as individual exceptions can be associated to applications based on the exception handling design for the organization.

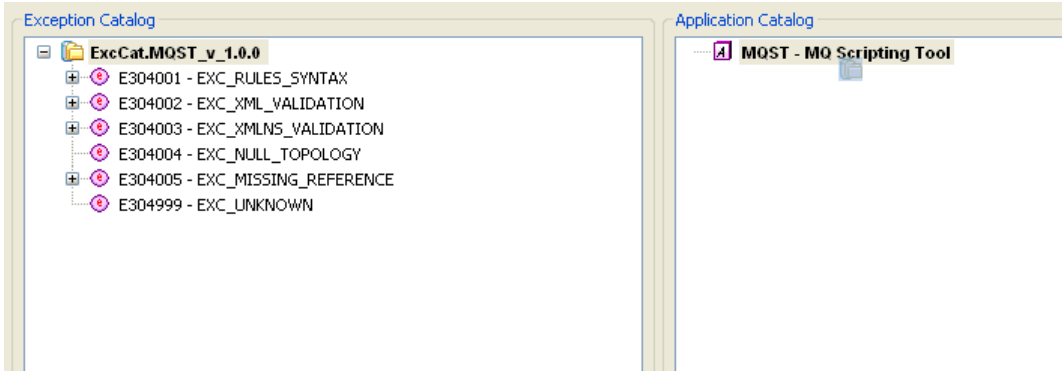


Figure 48 – Associating Exception Catalog with the Application



Figure 49 – Exception Catalog mapped to the Application

3.5. Running the Sample Application

Customize the sample [samp/bin/Exch_Java_01_OK.bat](#)

```
java com.ibm.mq.exc.samp.Exch_Java_01_OK QM.EEH.01
```

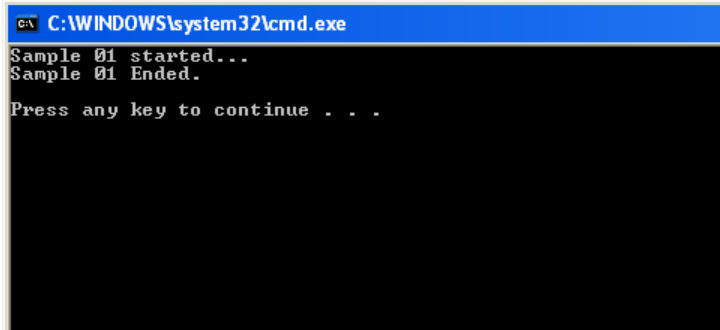


Figure 50 - Sample Application Running

The application puts an exception message on the exception queue [Q.EEH.FAIL](#). The EEH daemon processes the message and inserts it into the database.

3.6. Viewing Exceptions

For viewing exception from the database, the analyzer page provides selection criteria to narrow down the search. The retrieve data button runs the query and shows the exceptions in the table below.

The screenshot shows the 'EXCCAT.DB2' application window. The 'Selection Criteria' section includes a table with the following data:

AppId	Application
MQST	MQ Scripting Tool

Below this table are dropdown menus for 'Catalog' and 'Version'. To the right are input fields for 'Start Date', 'Start Time' (with a checkbox and spinners), 'End Date', and 'End Time' (with a checkbox and spinners). A 'Retrieve Data' button is positioned at the bottom right of the selection criteria area.

The 'Details' section displays a table with the following data:

Catalog	Version	Code	App Id	Timestamp	Exception UUID	Data UUID
ExcCat.MQST	1.0.0	F404003	MQST	2011-09-28 09:43:09.301	3cb11651-e0e1-4fc7-abd...	

At the bottom of the window, there is a navigation bar with buttons for 'Config', 'Admin', 'Analyzer', and 'Log'.

Figure 51 – Exception Database Editor: Analyzer page

The details of the exceptions are shown on a separate page added dynamically to the editor and generated on double clicking on the exception on the table on the *Analyzer* page.

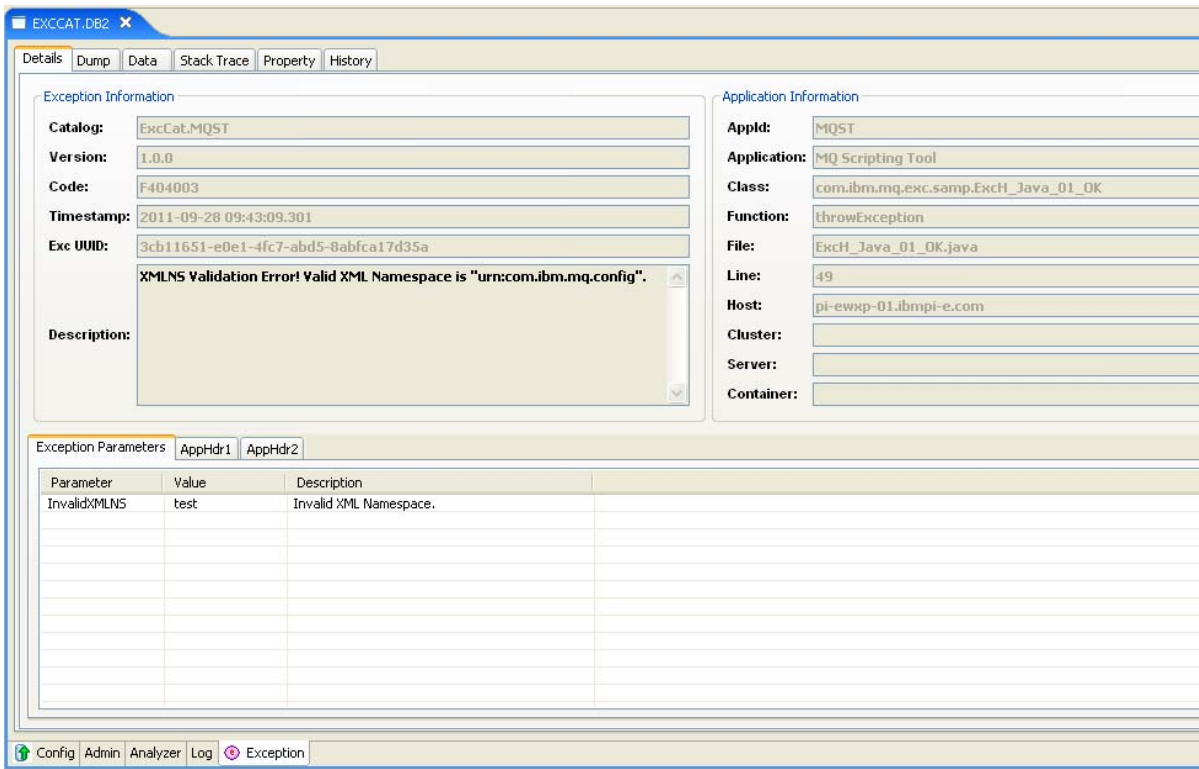


Figure 52 – Exception Page: Exception Details

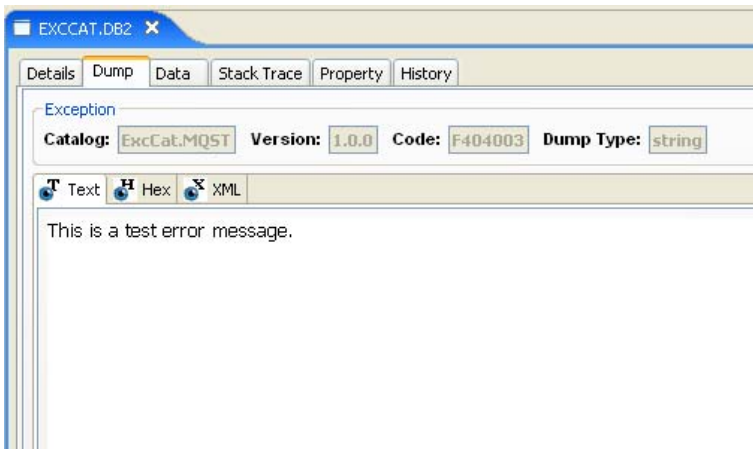


Figure 53 – Exception Page: Exception Dump

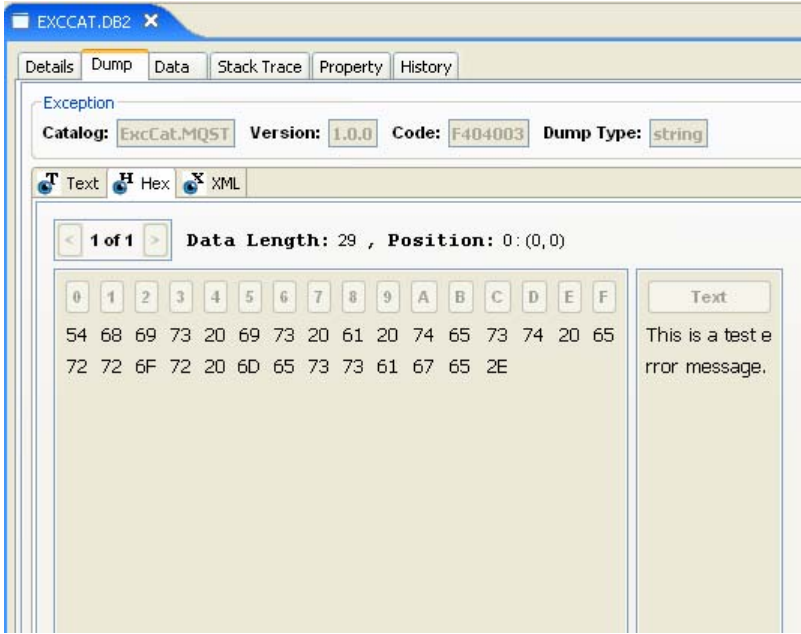


Figure 54 – Exception Page: Exception Dump: Hex Dump Renderer

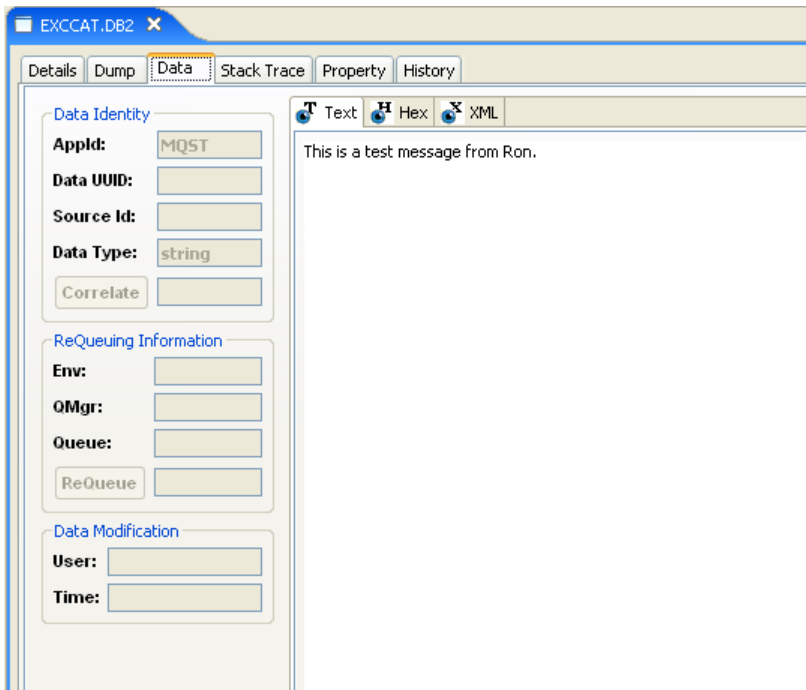


Figure 55 – Exception Page: Exception Data

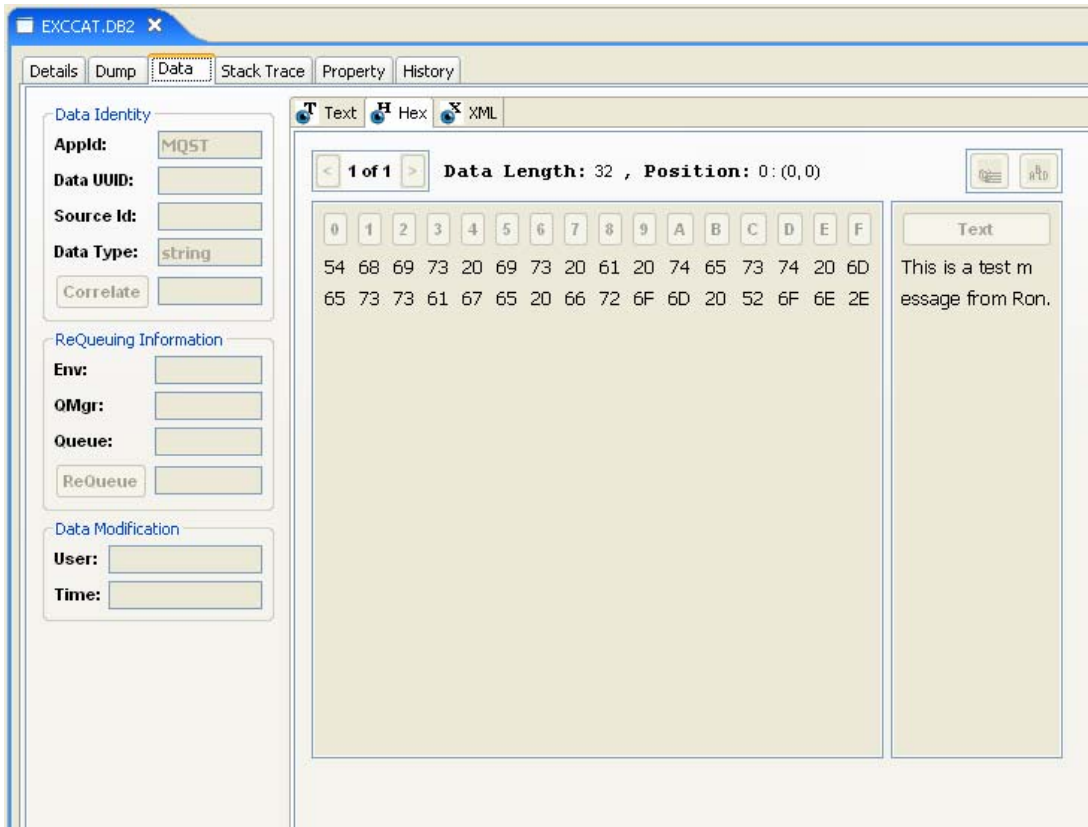


Figure 56 – Exception Page: Exception Data: Hex Data Renderer

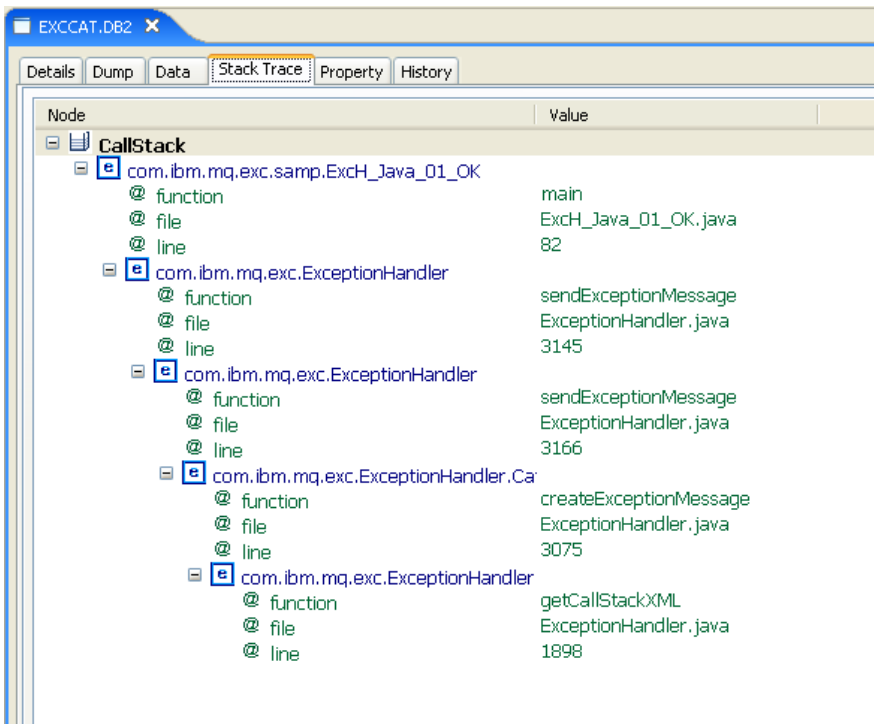


Figure 57 – Exception Page: Stack Trace

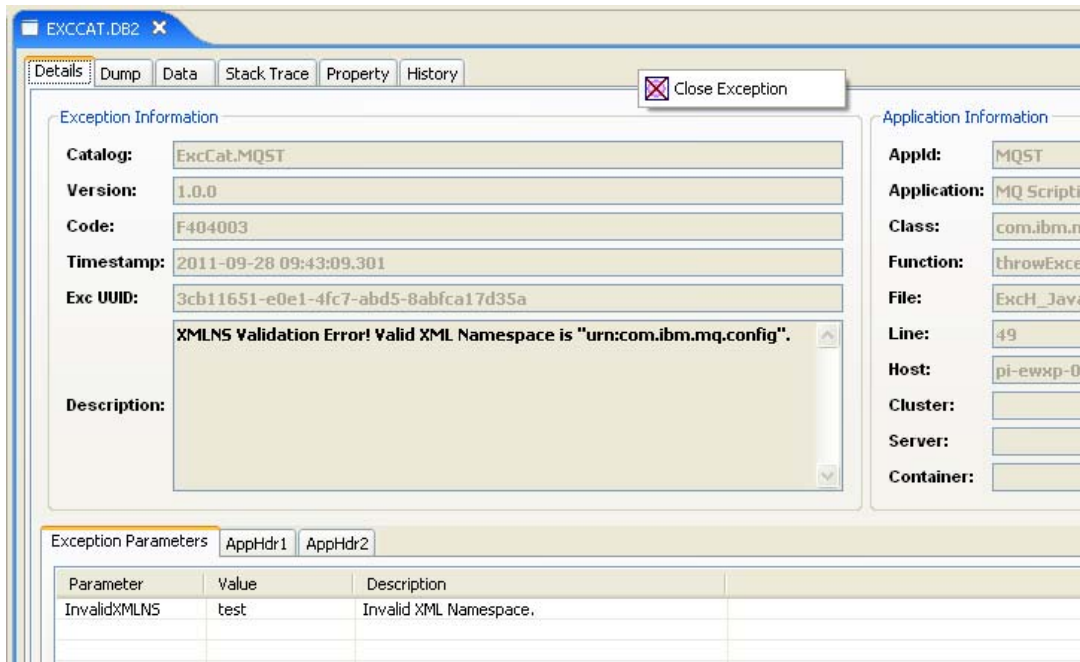


Figure 58 – Exception Page: Closing

To close the exception page right on the exception page after the last tab (*History*) and the menu for close will be enabled.

3.7. Custom Renderers (Preview)

The plug-in provides the interface to add custom *Renderers* but does not provide the corresponding interface to render either data or the dump.

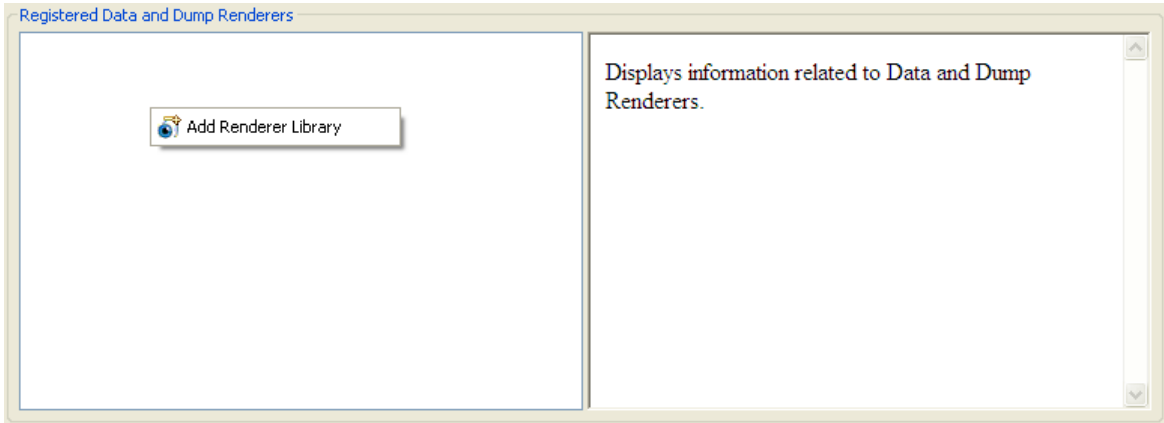


Figure 59 – Add Renderer Library

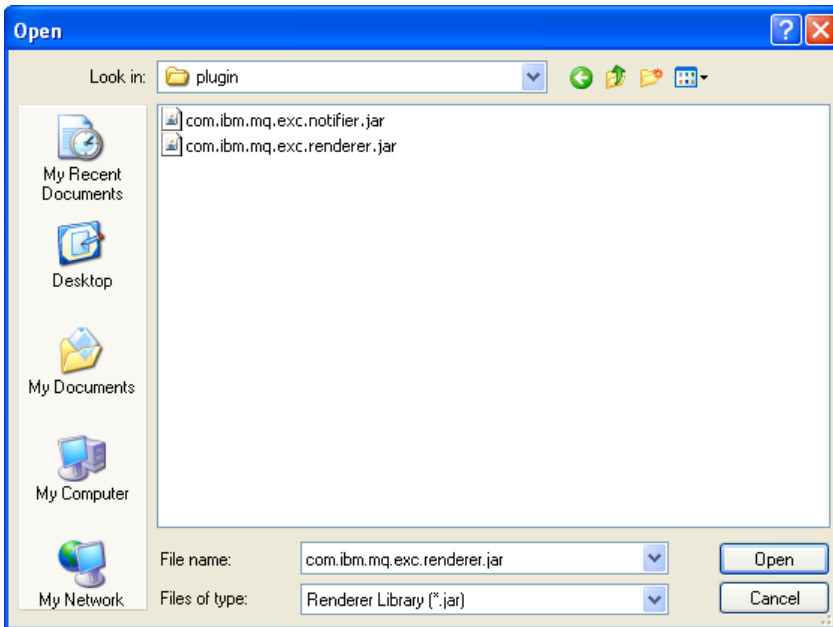


Figure 60 - Select Renderer jar



Figure 61 – Added Renderer Library

4. Running EEH Web Analyzer:

To run the EEH Web Analyzer application to view exceptions in the database follow the following steps:

- Set the JNDI Context for JDBC
- Load the JDBC Driver jars and MQ jars in the Application Server
- Deploy the Web Archive to the Application Server

4.1. Installing on Tomcat 7.0

Create JDBC reference to the database by adding the resource entry for the database in the following file [CATALINA_BASE/conf/context.xml](#)

```
<Resource name="jdbc/EXPDB"
  auth="Container"
  type="javax.sql.DataSource"
    username=<username of the database>
    password=<password of the database>
    driverClassName=<driver class name>
    url=<Url of the Database>
    maxActive="8"
    maxIdle="4"/>
```

DB2 JDBC entry in [CATALINA_BASE/conf/context.xml](#)

```
<Resource name="jdbc/EXPDB"
  auth="Container"
  type="javax.sql.DataSource"
    username="pi-eadmin"
    password="****"
    driverClassName="com.ibm.db2.jcc.DB2Driver"
    url="jdbc:db2://localhost:50001/EEH01"
    maxActive="8"
    maxIdle="4"/>
```

Add the DB2 JDBC jar files to the [CATALINA_BASE/lib](#)

```
db2jcc.jar
db2jcc_license_cisuz.jar
```

For re-queuing of exception data, the following jars must be added to the **CATALINA_BASE/lib** directory.

```
com.ibm.mq.jar
com.ibm.mq.commonservices.jar
com.ibm.mq.headers.jar
com.ibm.mq.jmqi.jar
com.ibm.mq.jms.Nojndi.jar
com.ibm.mq.pcf.jar
com.ibm.mqjms.jar
connector.jar
dhbcore.jar
fscontext.jar
jms.jar
jndi.jar
jta.jar
ldap.jar
```

Deploy the Web Analyzer by copying the WAR file from **EEH_PATH/web/EEH.war** to **CATALINA_BASE/webapps**. Catalina will extract it at startup.

Customize the following script to start the Tomcat server

```
@echo off
set JRE_HOME=D:\_IBM\WMB_7.0\jre16
set CATALINA_BASE=Z:\_Backup\Assets\Assets.TEST\webserver\Tomcat
set CATALINA_HOME=Z:\_Backup\Assets\Assets.TEST\.install\apache-tomcat-7.0.22

%CATALINA_HOME%\bin\startup
```

URL to the Web Analyzer:

<http://<tomcat server>:<tomcat port>/EEH>

Catalog	Version	Code	App Id	TimeStamp	Exception UUID	Data UUID	Requeue QMgr	Requeue Queue	Description
ExcCat.MQST	1.0.0	04003	MQST	2012-03-12 09:03:26.733	ce8e2c7e-673d-4f1b-a840-f28ac3c38a3a				XMLNS Validation Error! Valid XML Namespace is "urn:com.ibm.mq.config".
ExcCat.MQST	1.0.0	04003	MQST	2012-03-12 09:03:47.466	326b7b0d-f371-41f8-847a-0c192c789f0c				XMLNS Validation Error! Valid XML Namespace is "urn:com.ibm.mq.config".
ExcCat.MQST	1.0.0	04003	MQST	2012-03-13 01:58:00.025	21fc560e-3ebc-41b8-9a6d-0cab34d85167				XMLNS Validation Error! Valid XML Namespace is "urn:com.ibm.mq.config".
ExcCat.MQST	1.0.0	04003	MQST	2012-03-13 01:58:02.034	7ebad3a5-15a5-4b31-b856-8f4959f7d3cd				XMLNS Validation Error! Valid XML Namespace is "urn:com.ibm.mq.config".
ExcCat.MQST	1.0.0	04003	MQST	2012-04-03 19:55:57.159	fe18b8bf-c682-490b-aab1-f5ab2161ff7a				XMLNS Validation Error! Valid XML Namespace is "urn:com.ibm.mq.config".
ExcCat.MQST	1.0.0	04003	MQST	2012-04-04 19:33:54.319	e0c2c157-b09d-4bd2-819e-81e0ad6d1dec				XMLNS Validation Error! Valid XML Namespace is "urn:com.ibm.mq.config".

Figure 62 – EEH Web Analyzer

This page is automatically refreshed with the selected search criteria. The exception details may be viewed by double-clicking on the individual exception.

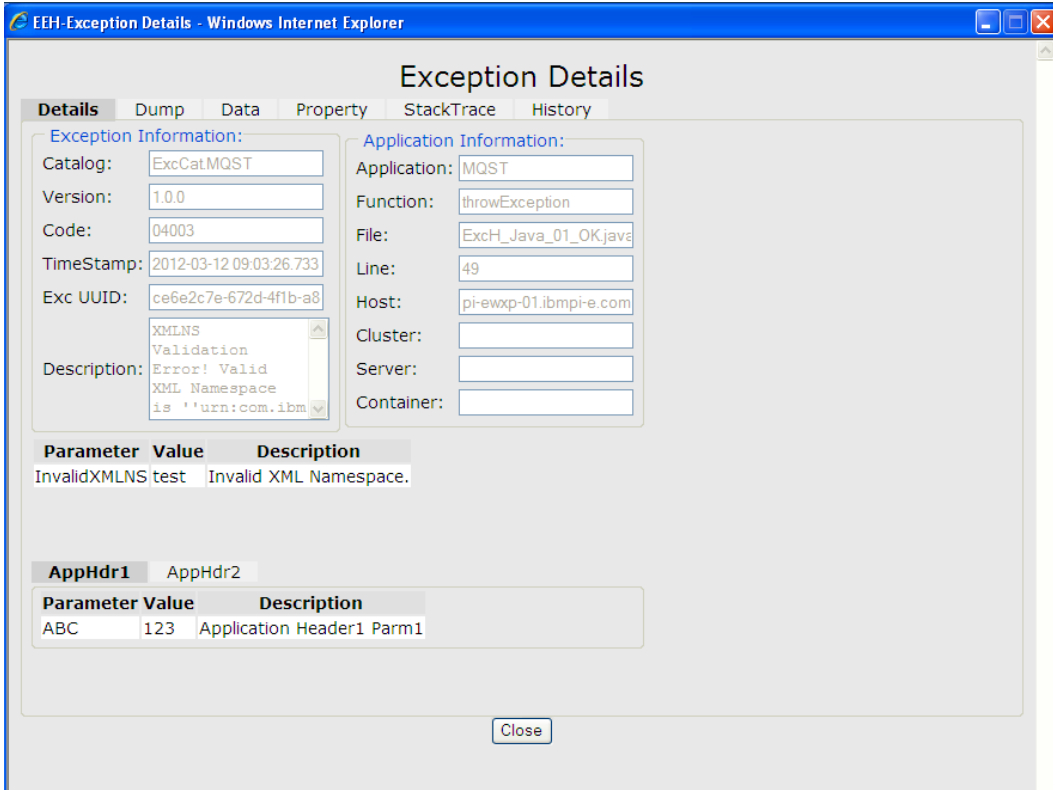


Figure 63 – EEH Web Analyzer: Exception Details

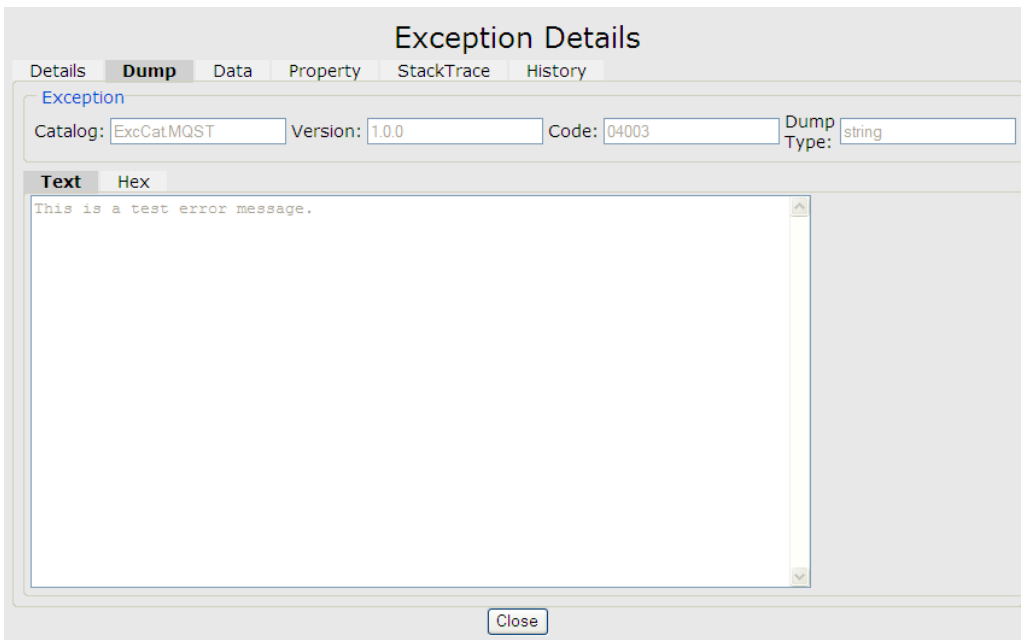


Figure 64 – EEH Web Analyzer: Exception Dump

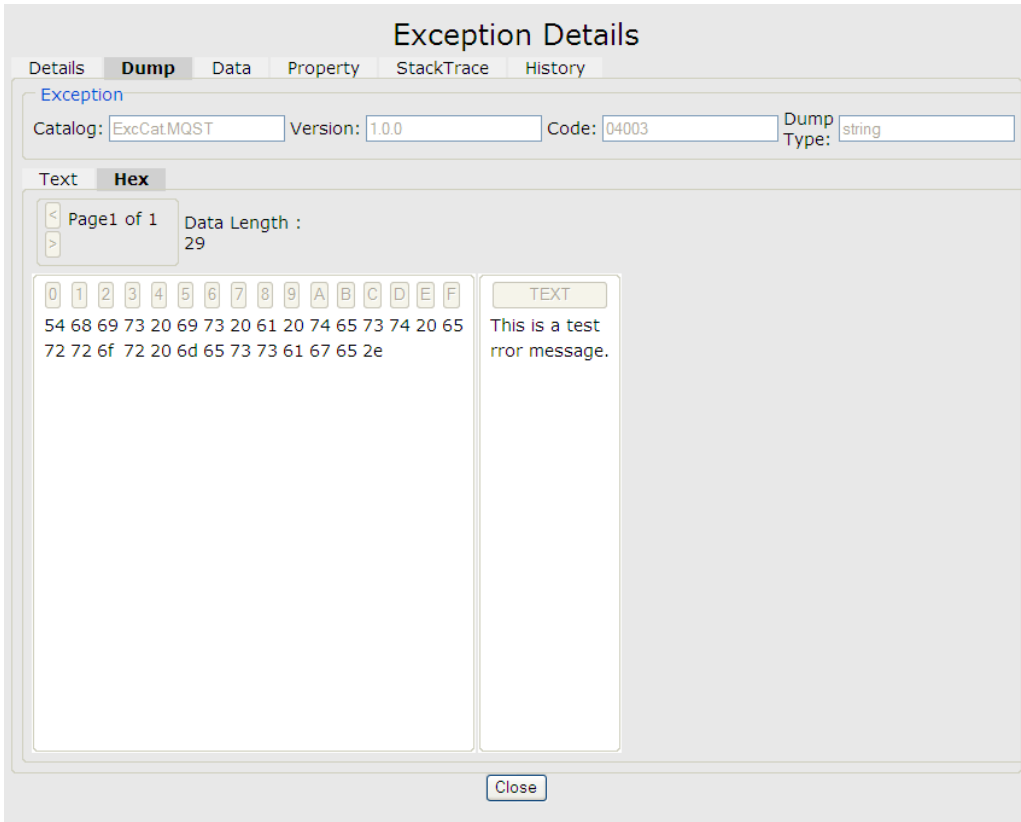


Figure 65 – EEH Web Analyzer: Exception Dump (Hex)

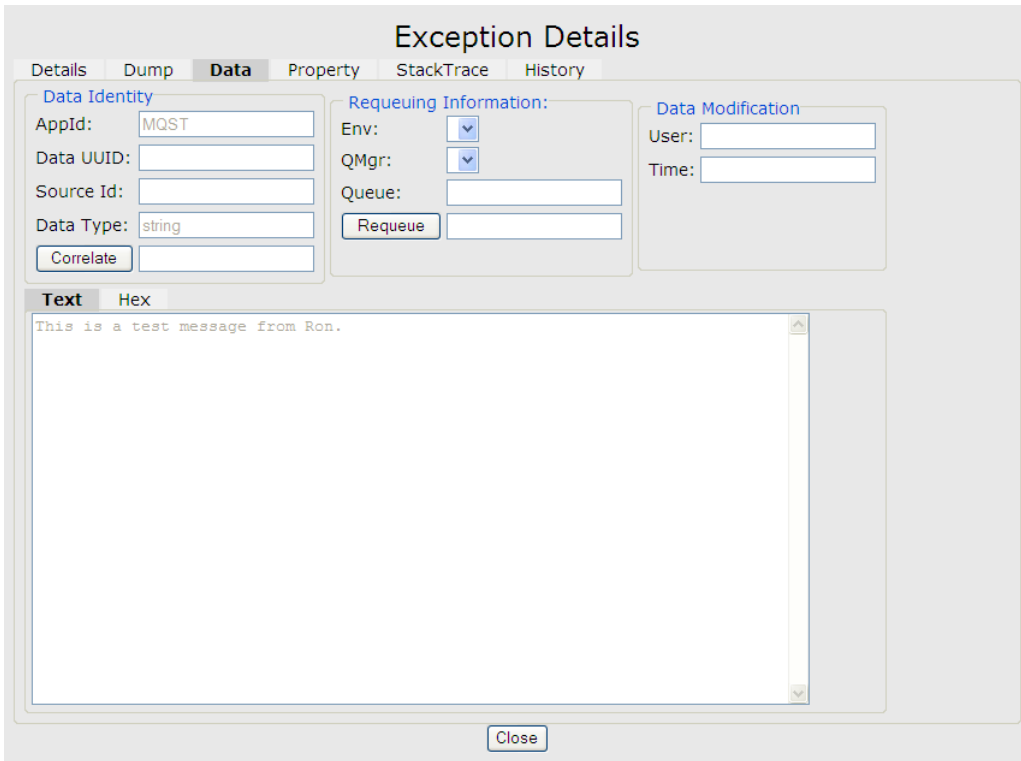


Figure 66 – EEH Web Analyzer: Exception Data

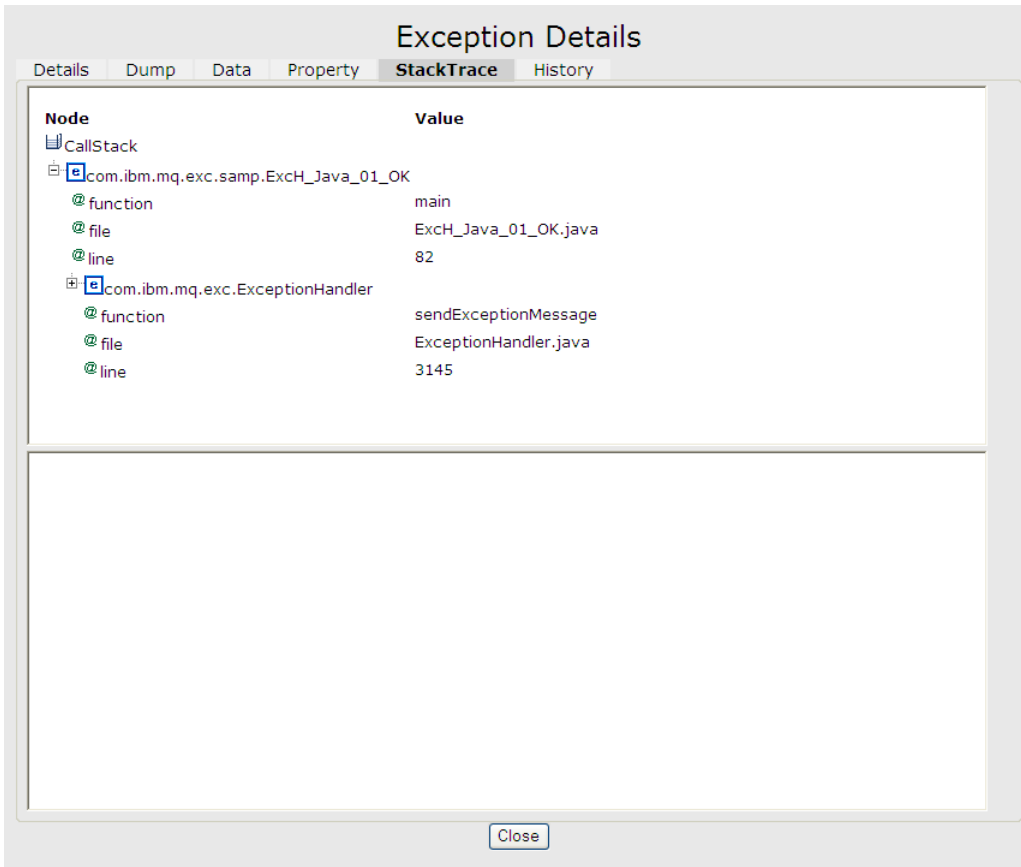


Figure 67 – EEH Web Analyzer: Call Stack and Exception Stack Trace

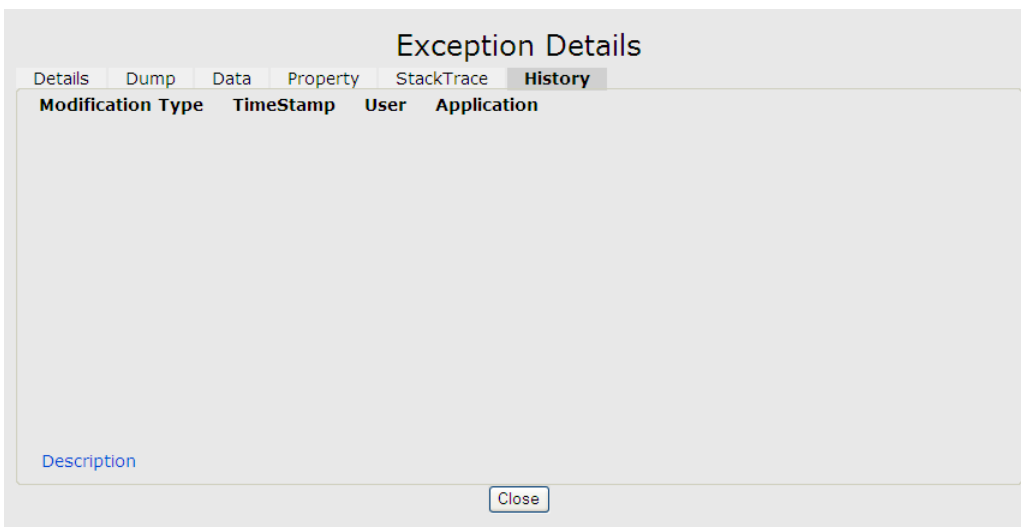


Figure 68 – EEH Web Analyzer: Data Editing and Re-queuing History

4.2. Installing on WebSphere® Application Server 8.0

Start the application server for the profile.



Figure 69 – Start Application Server

Start the administrative console.

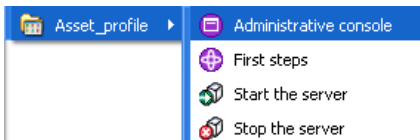


Figure 70 - Administrative Console

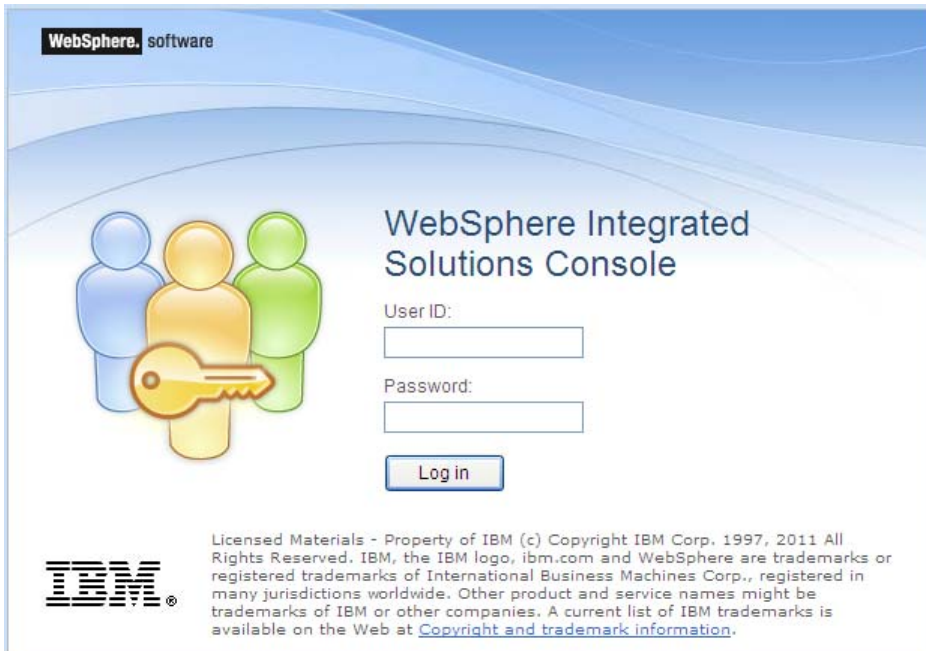


Figure 71 - Application Server Log in

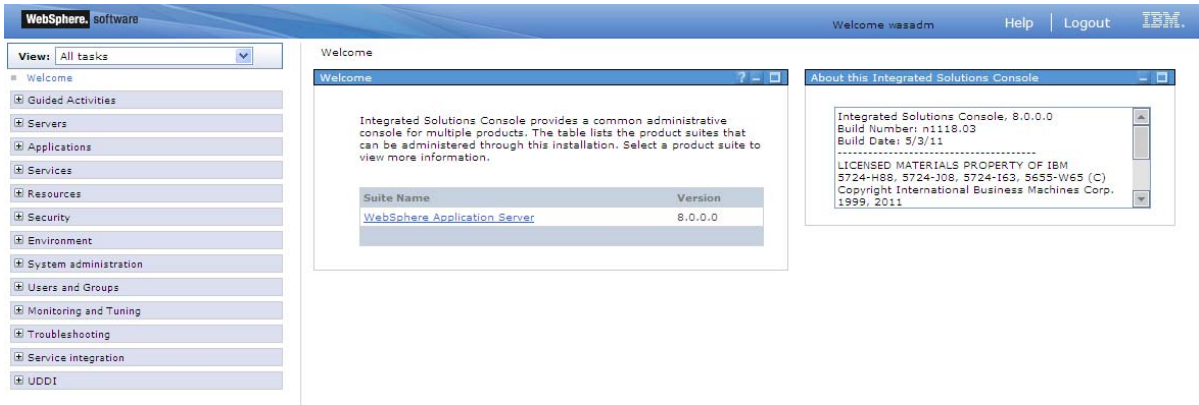


Figure 72 – WebSphere® Application Server Welcome screen

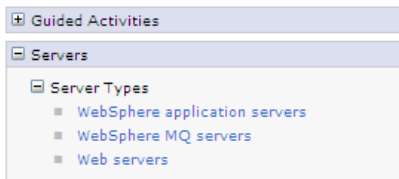


Figure 73 - Application Server menu option

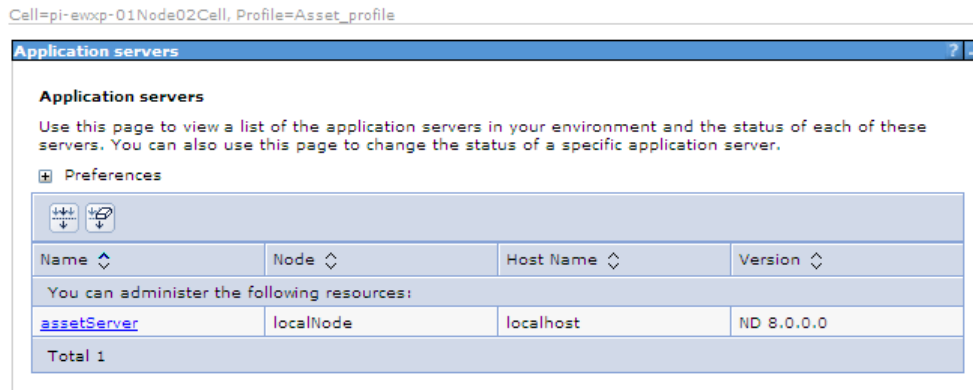


Figure 74 - Define Application Server

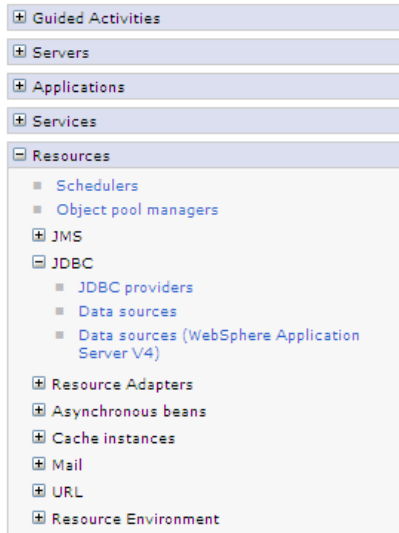


Figure 75 - JDBC Resource Definition menu option

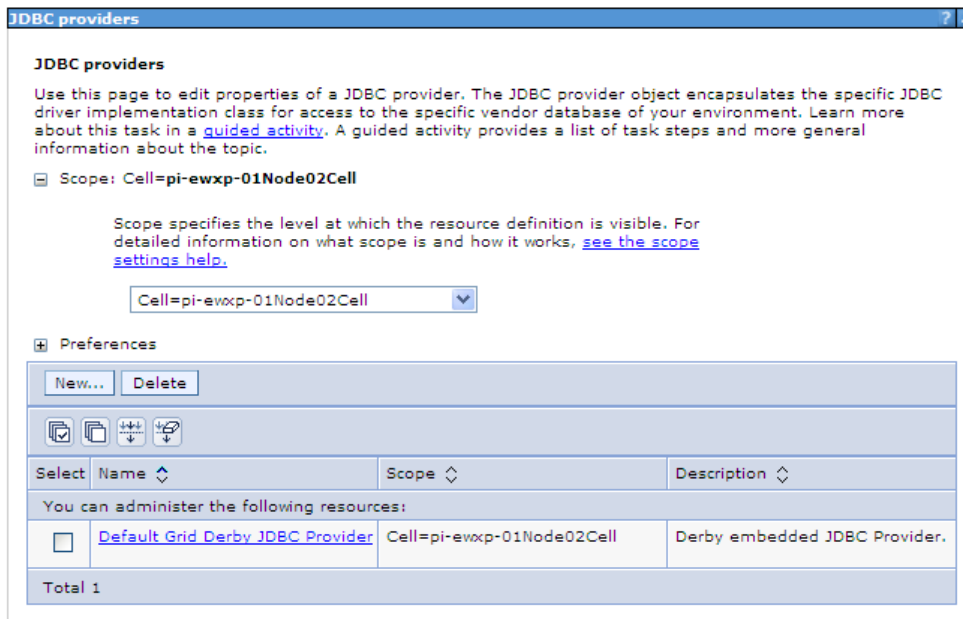


Figure 76 - Create New JDBC Provider

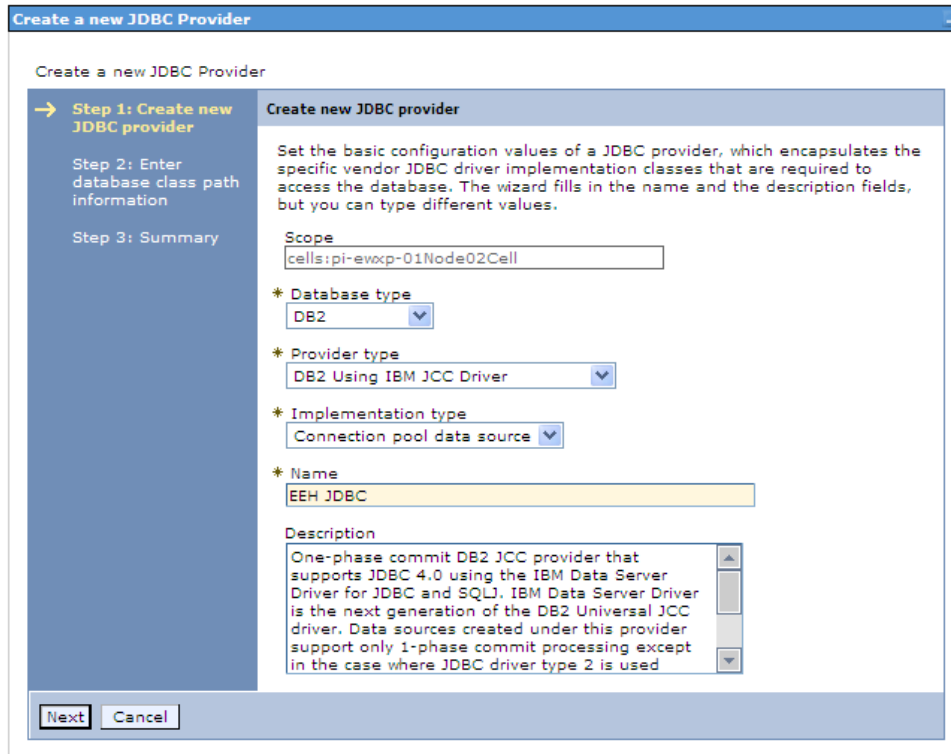


Figure 77 - Create New JDBC Provider: Step 1

Create a new JDBC Provider

Create a new JDBC Provider

Step 1: Create new JDBC provider

→ **Step 2: Enter database class path information**

Step 3: Summary

Enter database class path information

Set the class path for the JDBC driver class files, which WebSphere(R) Application Server uses to define your JDBC provider. This wizard page displays a default list of jars and allows you to set the environment variables that define the directory locations of the files. Use complete directory paths when you type the JDBC driver file locations. For example: C:\SQLLIB\java on Windows(R) or /home/db2inst1/sqllib/java on Linux(TM).

Entries are separated by using the ENTER key and must not contain path separator characters (such as ';' or '|'). If a value is specified for you, you may click Next to accept the value.

Class path:

Directory location for "db2jcc4.jar, db2jcc_license_cisuz.jar" which is saved as WebSphere variable \${DB2_JCC_DRIVER_PATH}

Directory location for "pdq.jar, pdqgmt.jar" which is saved as WebSphere variable \${PUREQUERY_PATH}

Native library path

Directory location which is saved as WebSphere variable \${DB2_JCC_DRIVER_NATIVEPATH}

Figure 78 – Create New JDBC Provider: Step 2

[N.B. Please specify the location where the database jar files are present.]

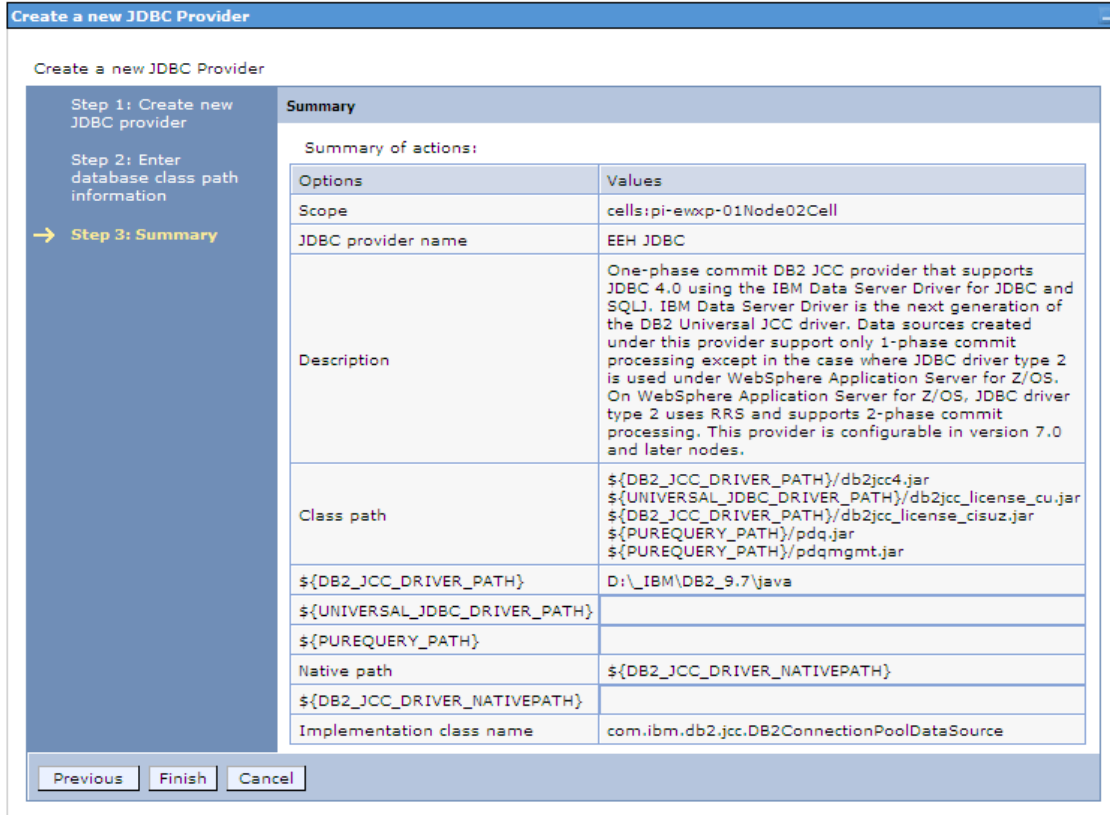


Figure 79 – Create New JDBC Provider: Step 3

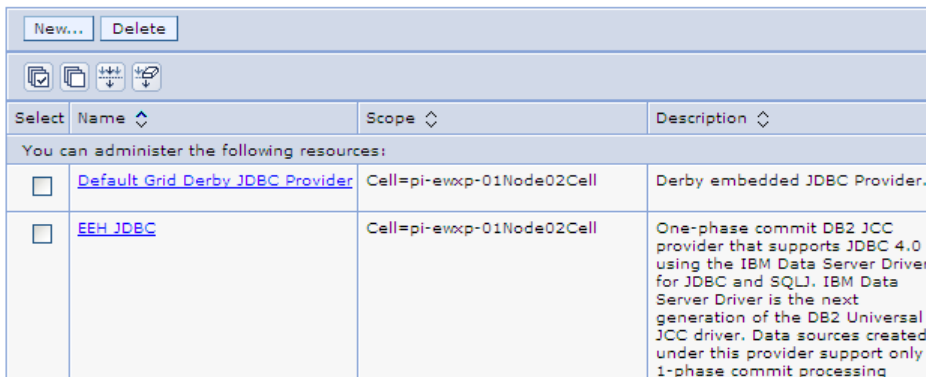


Figure 80 – JDBC Provider created

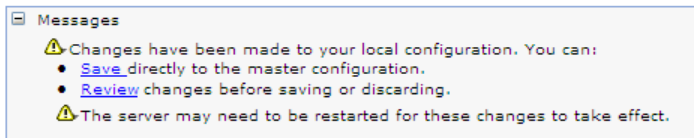


Figure 81 - Save JDBC Provider configuration

[JDBC providers](#) > EEH JDBC

Use this page to edit properties of a Java Database Connectivity (JDBC) provider. The JDBC provider object encapsulates the specific JDBC driver implementation class for access to the specific vendor database of your environment.

Configuration

<p>General Properties</p> <p>* Scope cells:pi-ewcxp-01Node02Cell</p> <p>† Name EEH JDBC</p> <p>Description One-phase commit DB2 JCC provider that supports JDBC 4.0 using the IBM Data Server Driver for JDBC and SQLJ. IBM Data Server Driver is the next generation of the DB2 Universal JCC driver. Data sources created under this provider support only 1-</p> <p>Class path \${UNIVERSAL_JDBC_DRIVER_PATH}/db2jcc_license_cu.jar \${DB2_JCC_DRIVER_PATH}/db2jcc_license_cisuz.jar \${PUREQUERY_PATH}/pdq.jar \${PUREQUERY_PATH}/pdqmgmt.jar</p> <p>Native library path \${DB2_JCC_DRIVER_NATIVEPATH}</p> <p><input type="checkbox"/> Isolate this resource provider</p> <p>* Implementation class name com.ibm.db2.jcc.DB2ConnectionPoolDataSource</p> <p>Apply OK Reset Cancel</p>	<p>Additional Properties</p> <ul style="list-style-type: none"> ▪ Data sources ▪ Data sources (WebSphere Application Server V4)
--	--

Figure 82 – Configure Data Sources

JDBC providers

[JDBC providers](#) > [EEH JDBC](#) > [Data sources](#)

Use this page to edit the settings of a datasource that is associated with your selected JDBC provider. The datasource object supplies your application with connections for accessing the database. Learn more about this task in a [guided activity](#). A guided activity provides a list of task steps and more general information about the topic.

⊕ Preferences

New... Delete Test connection Manage state...

⊞ ⊞ ⊞ ⊞

Select	Name	JNDI name	Scope	Provider	Description	Category
None						
Total 0						

Figure 83 – Create New Data Source definition

Create a data source

Create a data source

- Step 1: Enter basic data source information
- Step 2: Enter database specific properties for the data source
- Step 3: Setup security aliases
- Step 4: Summary

Enter basic data source information

Set the basic configuration values of a datasource for association with your JDBC provider. A datasource supplies the physical connections between the application server and the database.

Requirement: Use the Datasources (WebSphere(R) Application Server V4) console pages if your applications are based on the Enterprise JavaBeans (TM) (EJB) 1.0 specification or the Java(TM) Servlet 2.2 specification.

Scope
cells:pi-ewxp-01Node02Cell

JDBC_provider_name
EEH JDBC

* Data source name
EEH

* JNDI name
jdbc/EXPDB

Next Cancel

Figure 84 – Create New Data Source: Step 1

Create a data source

Create a data source

- Step 1: Enter basic data source information
- Step 2: Enter database specific properties for the data source
- Step 3: Setup security aliases
- Step 4: Summary

Enter database specific properties for the data source

Set these database-specific properties, which are required by the database vendor JDBC driver to support the connections that are managed through the datasource.

Name	Value
* Driver type	4
* Database name	EEH01
* Server name	localhost
* Port number	50001

Use this data source in container managed persistence (CMP)

Previous Next Cancel

Figure 85 – Create New Data Source: Step 2

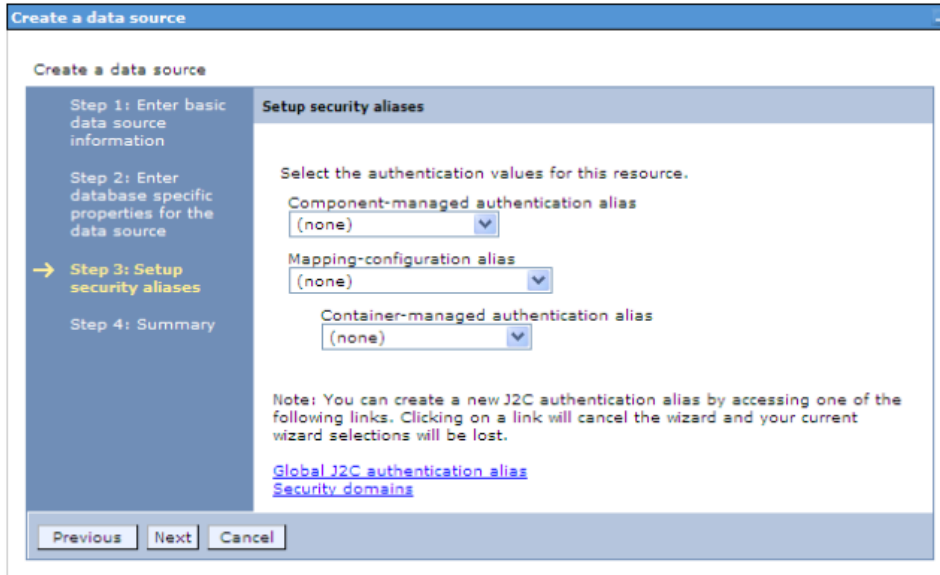


Figure 86 – Create New Data Source: Step 3

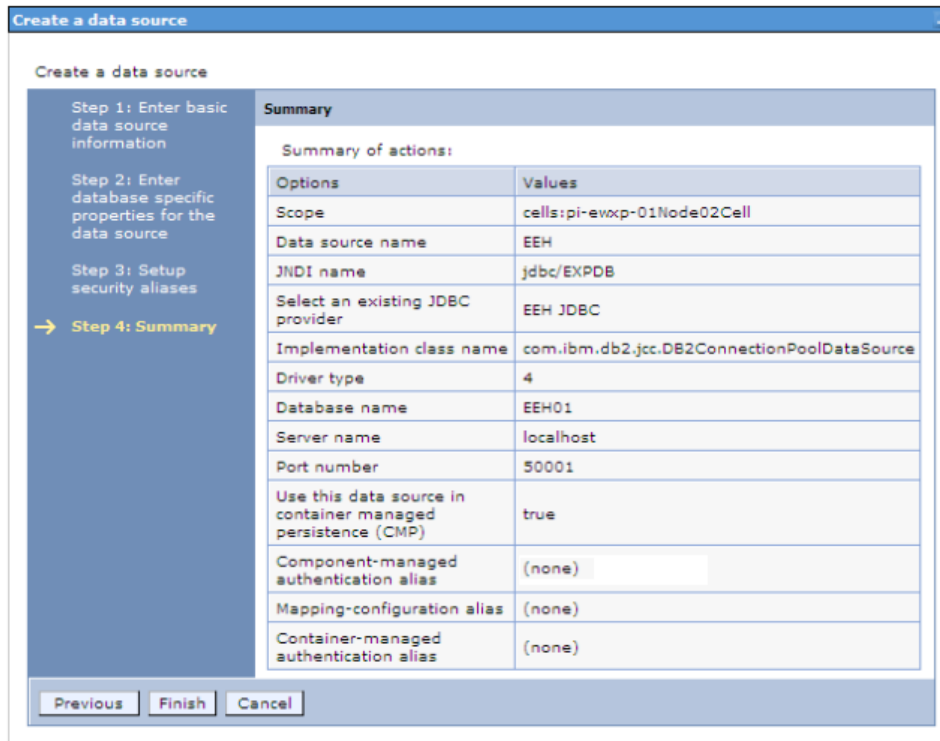


Figure 87 88 – Create New Data Source: Step 4

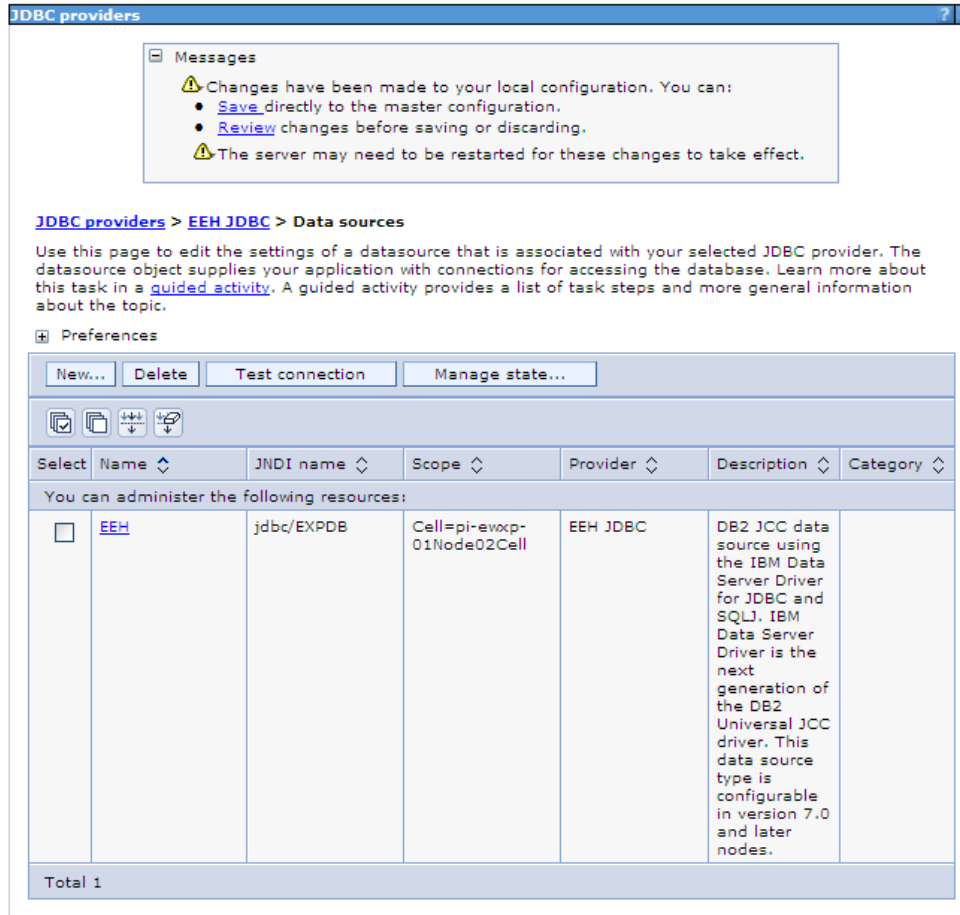


Figure 89 – Save Data Source definition

Data sources

Data sources > EEH

Use this page to edit the settings of a datasource that is associated with your selected JDBC provider. The datasource object supplies your application with connections for accessing the database.

Configuration

Test connection

General Properties

* Scope
cells:pi-ewxp-01Node02Cell

* Provider
EEH JDBC

* Name
EEH

JNDI name
jdbc/EXPDB

Use this data source in container managed persistence (CMP)

Description
DB2 JCC data source using the IBM Data Server Driver for JDBC and SQLJ. IBM Data Server Driver is the next generation of the DB2 Universal JCC driver. This data source type is configurable in version 7.0 and later nodes.

Category

Data store helper class name

Select a data store helper class

Data store helper classes provided by WebSphere Application Server

DB2 Universal data store helper
(com.ibm.websphere.rsadapter.DB2UniversalDataStoreHelper)

DB2 for iSeries data store helper
(com.ibm.websphere.rsadapter.DB2AS400DataStoreHelper)

Specify a user-defined data store helper

Enter a package-qualified data store helper class name

Security settings

Select the authentication values for this resource.

Component-managed authentication alias
(none)

Mapping-configuration alias
(none)

Container-managed authentication alias
(none)

Common and required data source properties

Name	Value
* Driver type	4
* Database name	EEH01
* Server name	localhost
* Port number	50001

Apply OK Reset Cancel

Additional Properties

- Connection pool properties
- WebSphere Application Server data source properties
- Custom properties

Related Items

- JAAS - J2C authentication data

Figure 90 – Add Security information to Data Source



Figure 91 - New JAAS definition



Figure 92 – Database connection User Id and Password

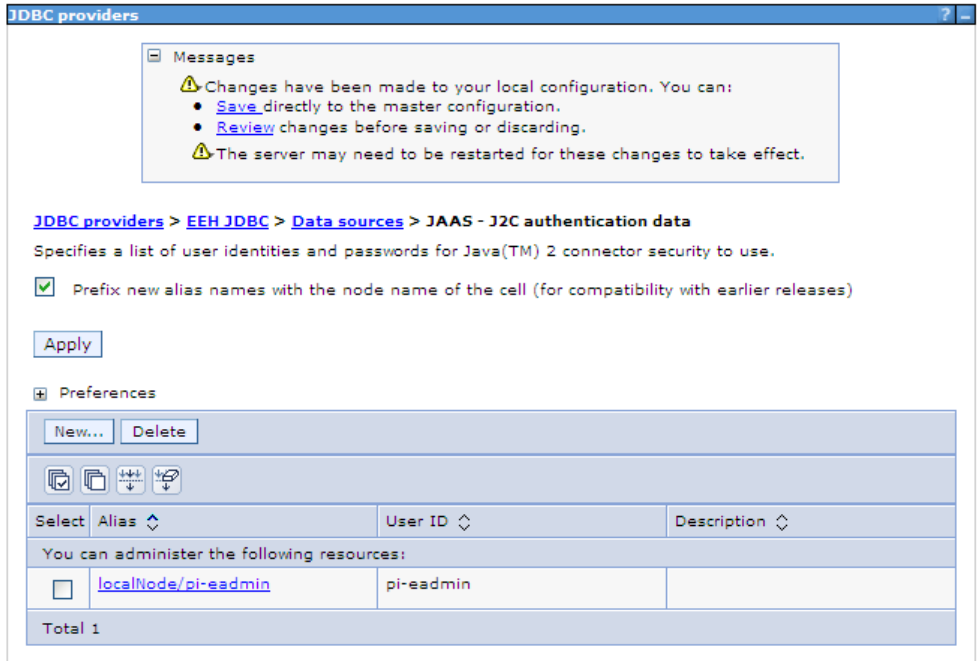


Figure 93 – Save JAAS definition

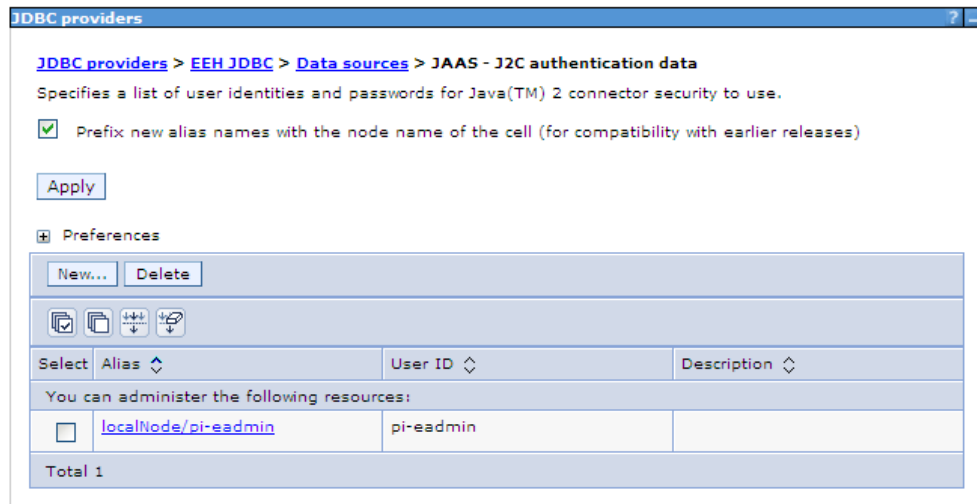


Figure 94 - Navigate back to Data Sources

Data sources

[Data sources](#) > EEH

Use this page to edit the settings of a datasource that is associated with your selected JDBC provider. The datasource object supplies your application with connections for accessing the database.

Configuration

Test connection

General Properties

* Scope

* Provider

* Name

JNDI name

Use this data source in container managed persistence (CMP)

Description

Category

Data store helper class name

Select a data store helper class

Data store helper classes provided by WebSphere Application Server

DB2 Universal data store helper
(com.ibm.websphere.rsadapter.DB2UniversalDataStoreHelper)

DB2 for iSeries data store helper
(com.ibm.websphere.rsadapter.DB2AS400DataStoreHelper)

Specify a user-defined data store helper

Enter a package-qualified data store helper class name

Additional Properties

- [Connection pool properties](#)
- [WebSphere Application Server data source properties](#)
- [Custom properties](#)

Related Items

- [JAAS - J2C authentication data](#)

Security settings

Select the authentication values for this resource.

Component-managed authentication alias

Mapping-configuration alias

Container-managed authentication alias

Common and required data source properties

Name	Value
* Driver type	<input type="text" value="4"/>
* Database name	<input type="text" value="EEH01"/>
* Server name	<input type="text" value="localhost"/>
* Port number	<input type="text" value="50001"/>

Figure 95 - Assign the defined Security definition

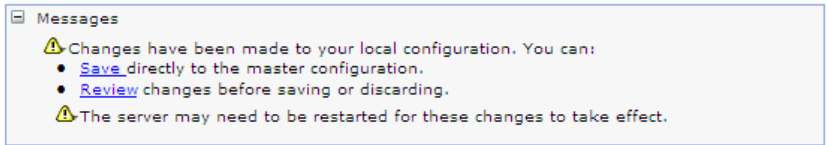


Figure 96 - Apply and Save the Data Source definition

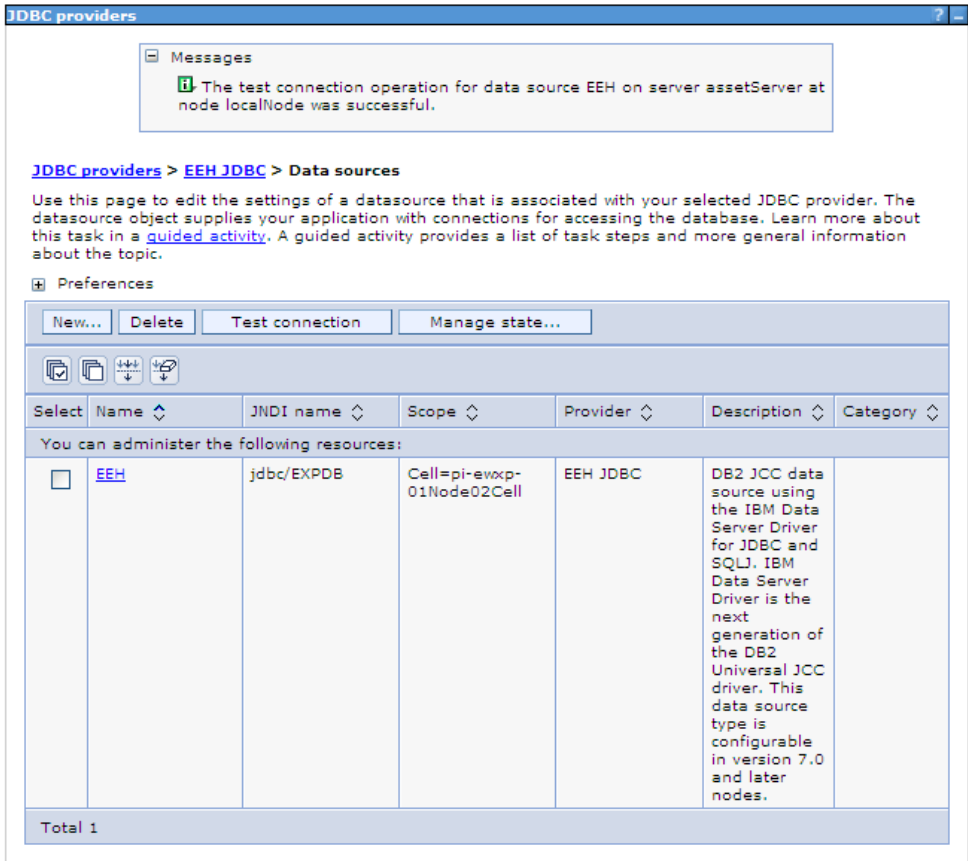


Figure 97 - Test the Data Source connection



Figure 98 - Install the EEH Web Analyzer

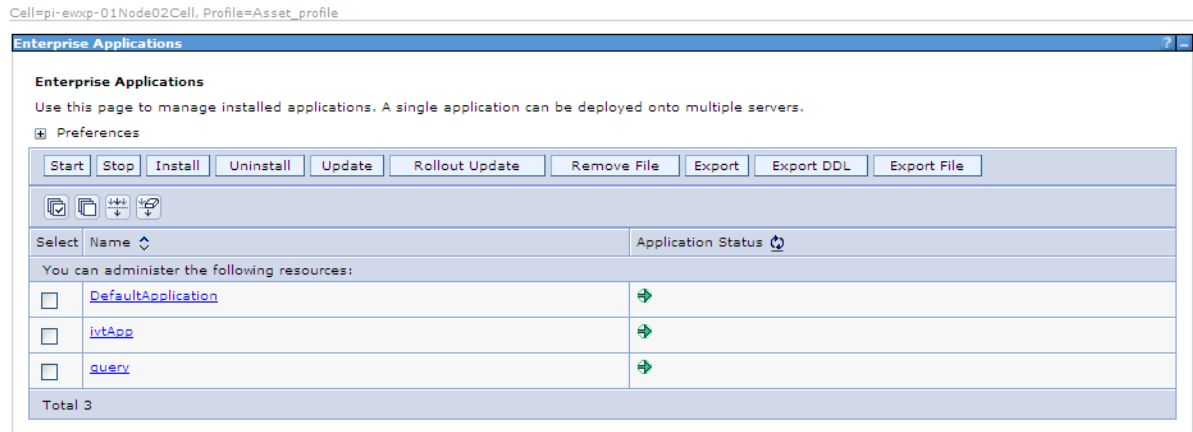


Figure 99 – Install Enterprise Application

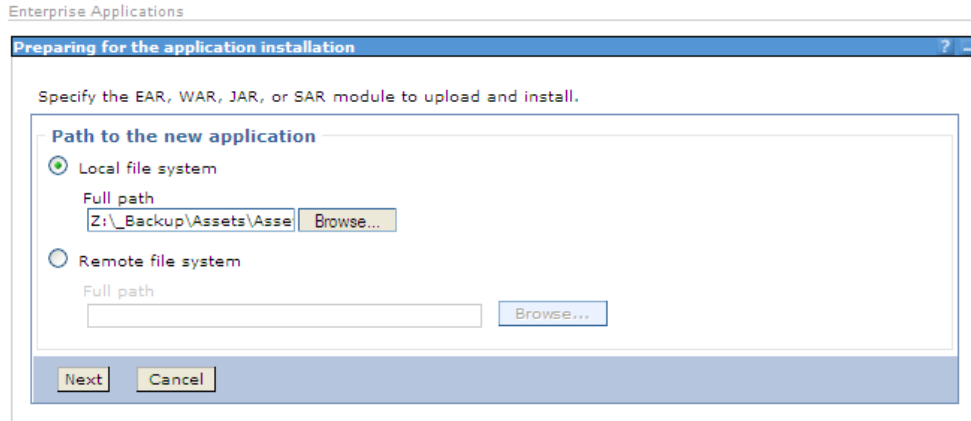


Figure 100 – EEH Web Analyzer install: War location

Specify the location of EEH.war.

The EEH.war is located at [%EEH_PATH%/web](#)

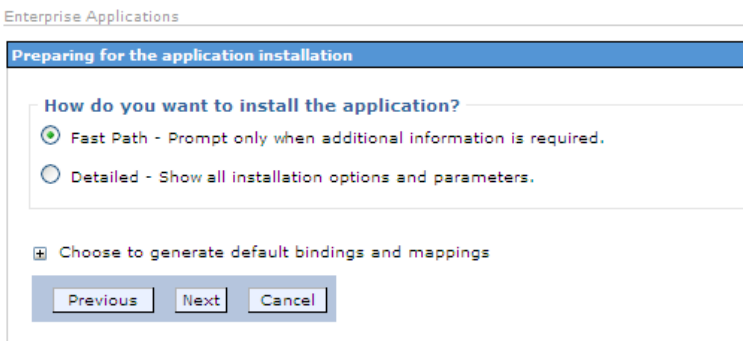


Figure 101 – EEH Web Analyzer install: Fast Path

Install New Application

Specify options for installing enterprise applications and modules.

→ **Step 1: Select installation options**

[Step 2: Map modules to servers](#)

✳ [Step 3: Map resource references to resources](#)

✳ [Step 4: Map virtual hosts for Web modules](#)

✳ [Step 5: Map context roots for Web modules](#)

[Step 6: Summary](#)

Select installation options

Specify the various options that are available for your application.

Precompile JavaServer Pages files

Directory to install application

Distribute application

Use Binary Configuration

Deploy enterprise beans

Application name

Create MBeans for resources

Override class reloading settings for Web and EJB modules

Reload interval in seconds

Deploy Web services

Validate Input off/warn/fail

Process embedded configuration

File Permission

Allow all files to be read but not written to
Allow executables to execute
Allow HTML and image files to be read by everyone

Application Build ID

Allow dispatching includes to remote resources

Allow servicing includes from remote resources

Business level application name

Asynchronous Request Dispatch Type

Allow EJB reference targets to resolve automatically

Deploy client modules
Client deployment mode

Validate schema

Figure 102 – EEH Web Analyzer install: Step 1

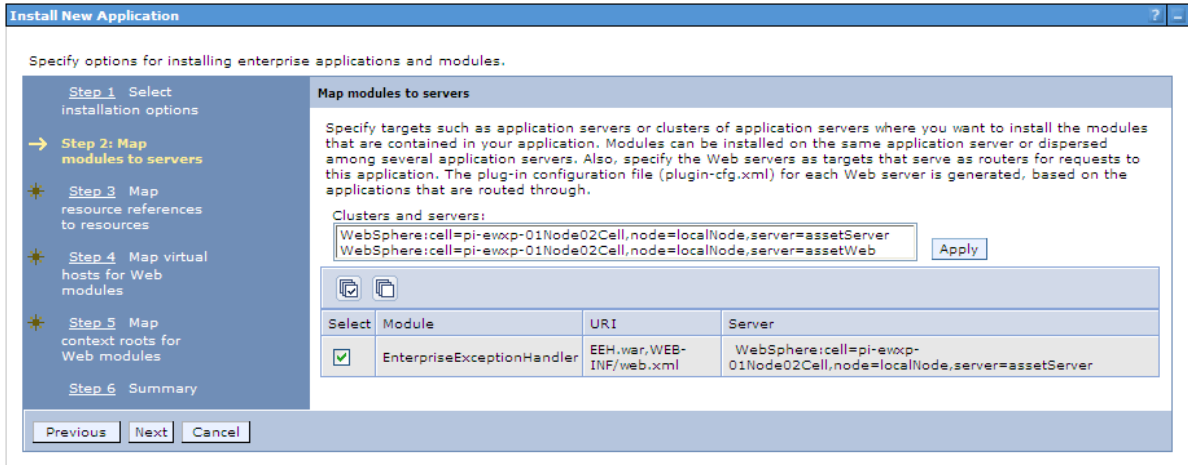


Figure 103 – EEH Web Analyzer install: Step 2

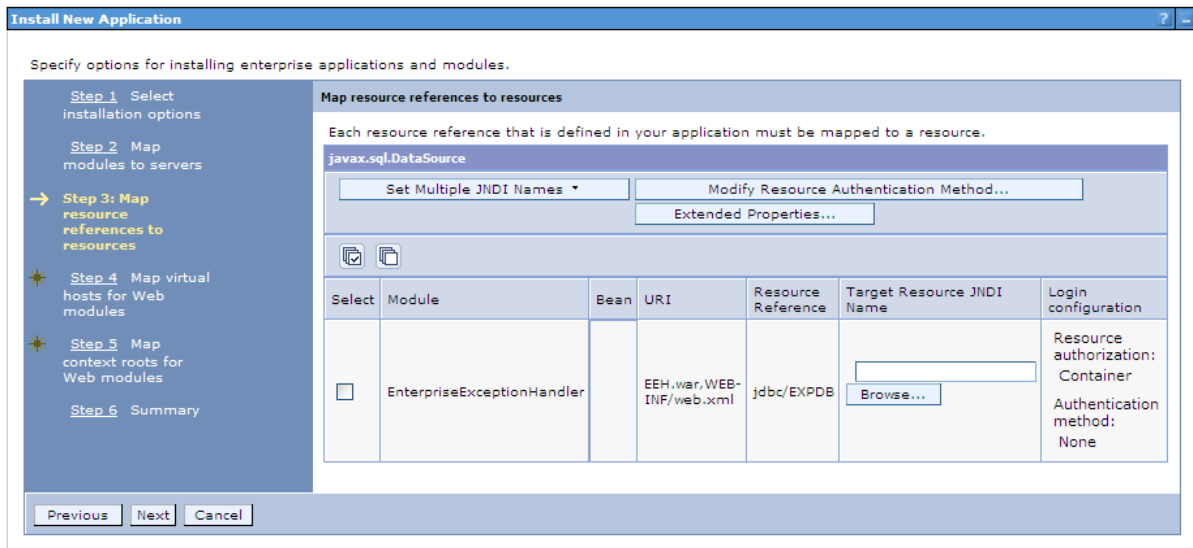


Figure 104 – EEH Web Analyzer install: Step 3: Browse

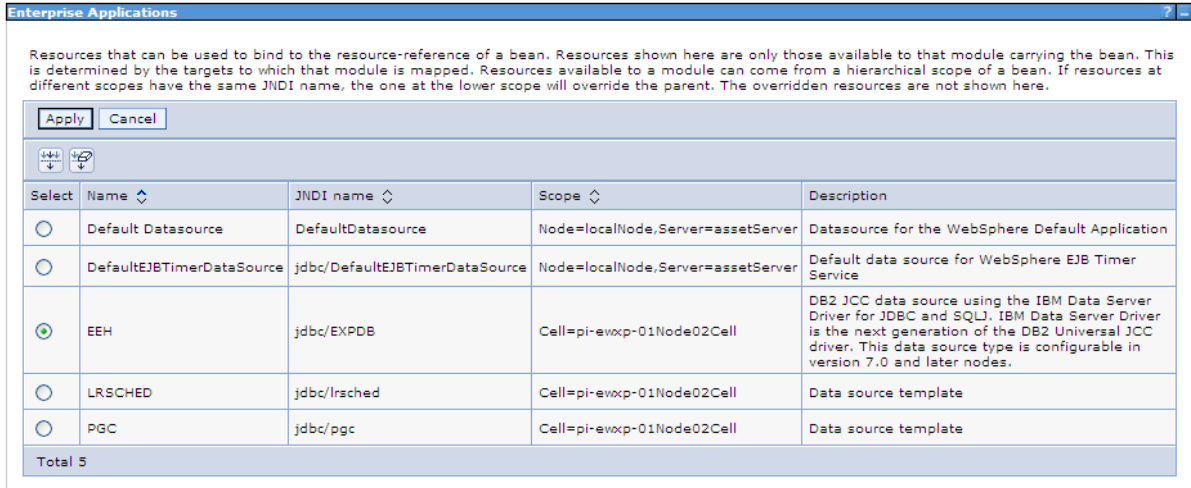


Figure 105 – EEH Web Analyzer install: Step 3: Select the Target JNDI

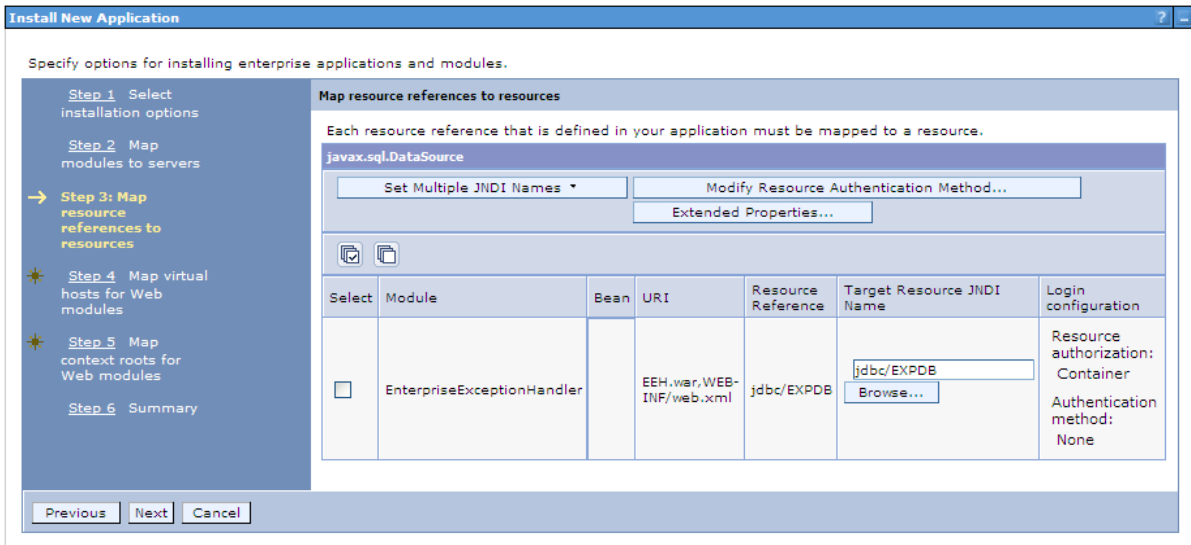


Figure 106 – EEH Web Analyzer install: Step 3: Database JNDI selected

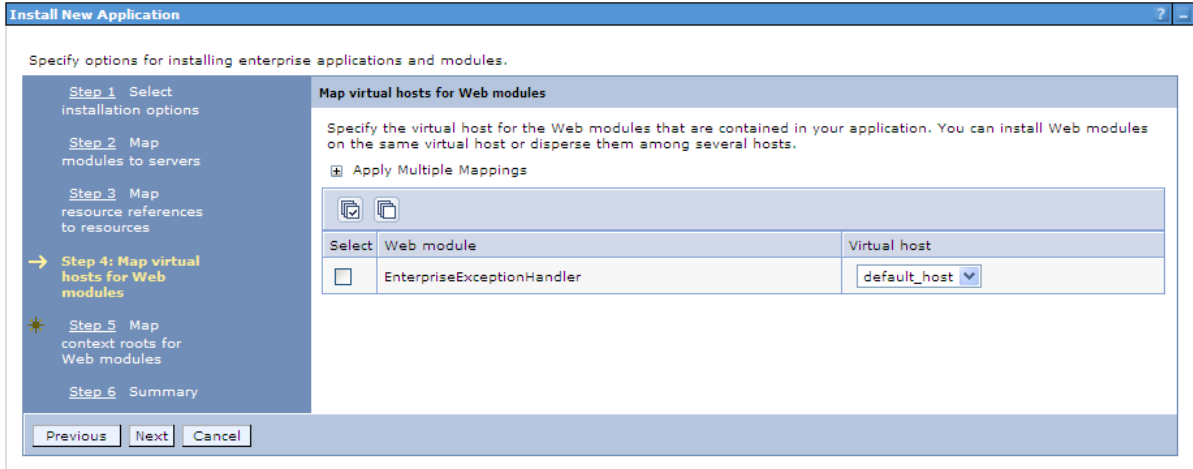


Figure 107 – EEH Web Analyzer install: Step 4

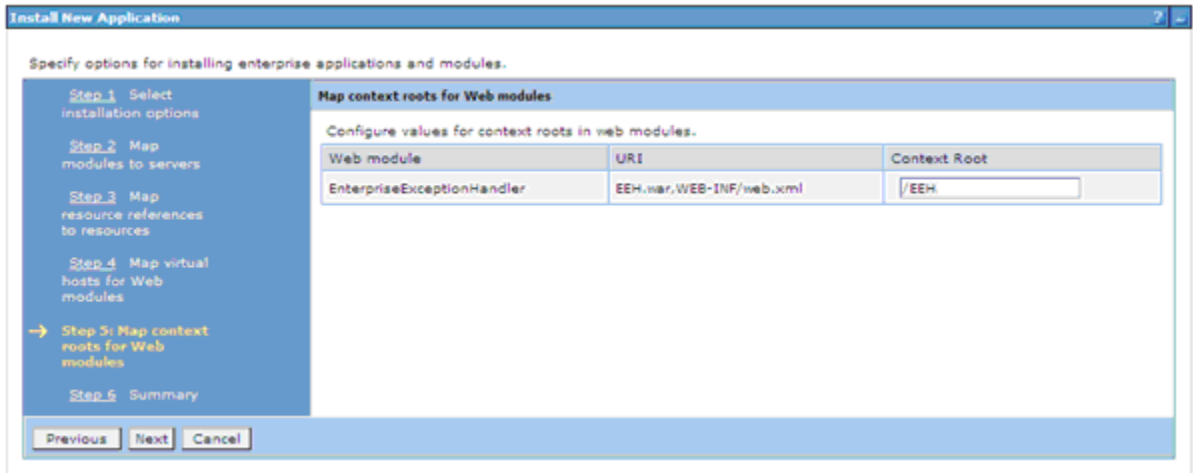


Figure 108 – EEH Web Analyzer install: Step 5

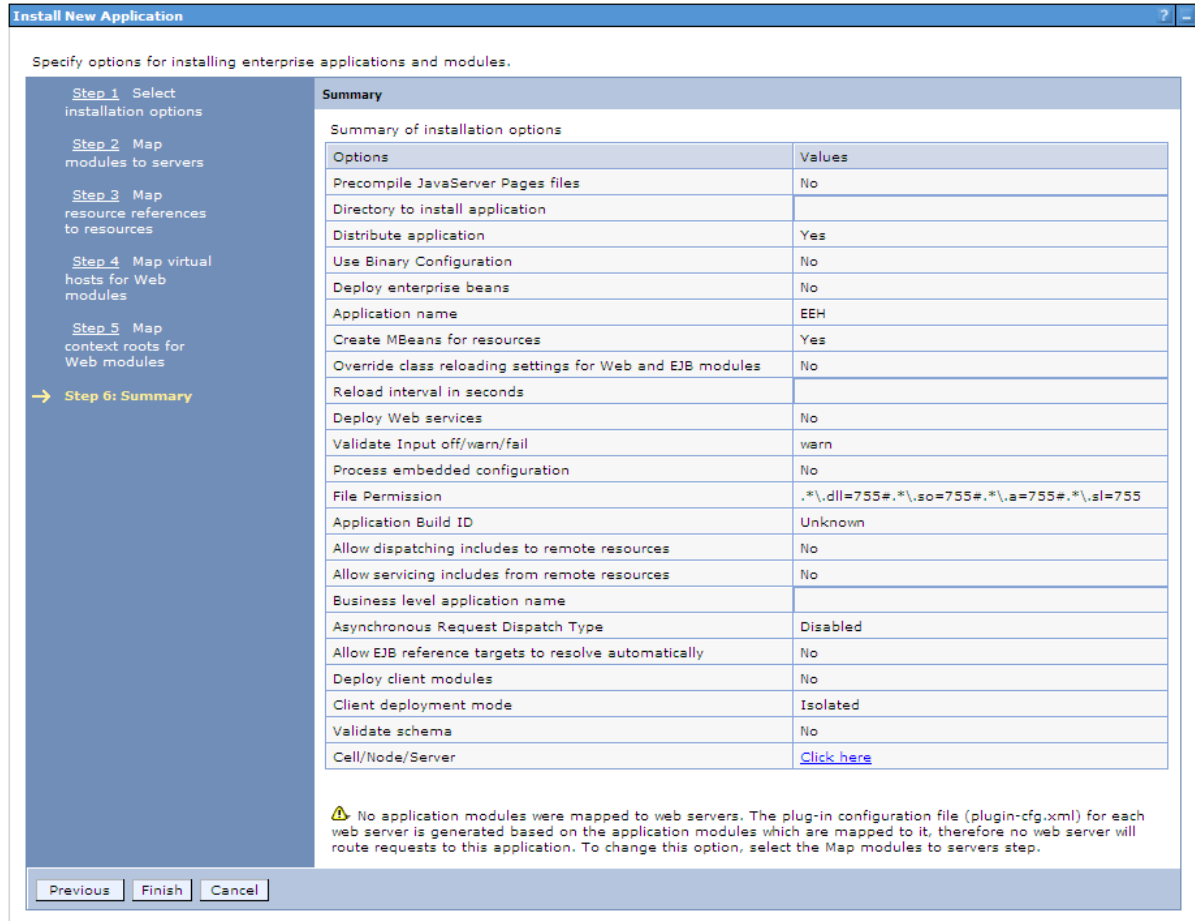


Figure 109 – EEH Web Analyzer install: Step 6

Clicking Finish started installing the EEH Web Analyzer and the log is displayed.

Installing...

If there are enterprise beans in the application, the EJB deployment process can take several minutes. Do not save the configuration until the process completes.

Check the SystemOut.log on the deployment manager or server where the application is deployed for specific information about the EJB deployment process as it occurs.

ADMA5016I: Installation of EEH started.

ADMA0159W: Duplicate root context(/) was found on the same node localNode and same host default_host

ADMA5068I: The resource validation for application EEH completed successfully, but warnings occurred during validation.

ADMA5058I: Application and module versions are validated with versions of deployment targets.

ADMA5005I: The application EEH is configured in the WebSphere Application Server repository.

ADMA5005I: The application EEH is configured in the WebSphere Application Server repository.

ADMA5081I: The bootstrap address for client module is configured in the WebSphere Application Server repository.

ADMA5053I: The library references for the installed optional package are created.

ADMA5005I: The application EEH is configured in the WebSphere Application Server repository.

ADMA5001I: The application binaries are saved in D:_IBM\WAS_8.0\was_nd\profiles\Asset_profile\wstempl-794994399\workspace\cells\pi-ewxp-01\Node02Cell\applications\EEH.ear\EEH.ear

ADMA5005I: The application EEH is configured in the WebSphere Application Server repository.

SECJ0400I: Successfully updated the application EEH with the appContextIDforSecurity information.

ADMA5005I: The application EEH is configured in the WebSphere Application Server repository.

ADMA5005I: The application EEH is configured in the WebSphere Application Server repository.

ADMA5113I: Activation plan created successfully.

ADMA5011I: The cleanup of the temp directory for application EEH is complete.

ADMA5013I: Application EEH installed successfully.

Application EEH installed successfully.

To start the application, first save changes to the master configuration.

Changes have been made to your local configuration. You can:

- [Save](#) directly to the master configuration.
- [Review](#) changes before saving or discarding.

To work with installed applications, click the "Manage Applications" link.

[Manage Applications](#)

Figure 110 – Save the configuration

RESTART the Application Server for the application to function.

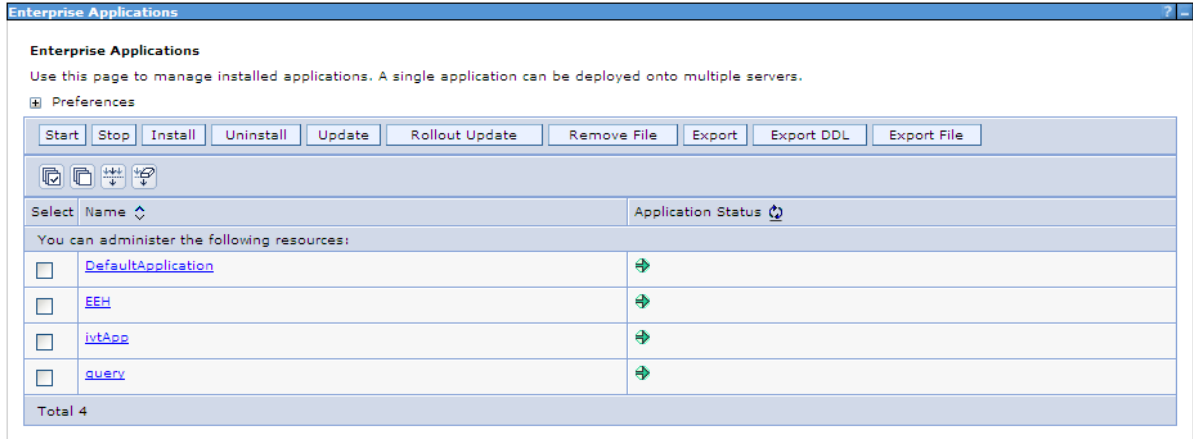


Figure 111 – Review that the Application is started

URL to the Web Analyzer:
<http://<WAS server>:<WAS port>/EEH>

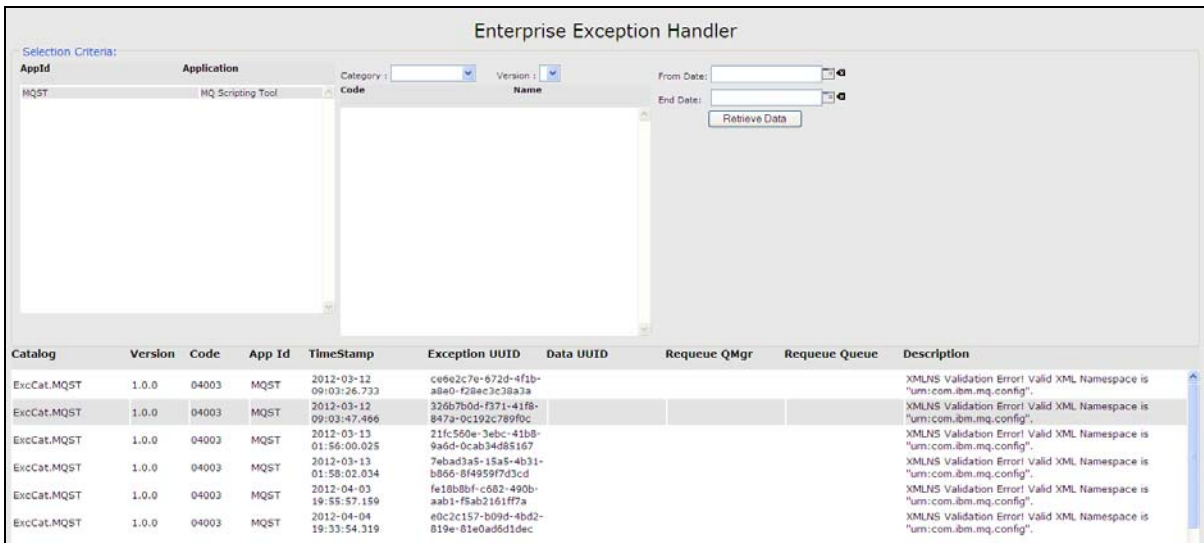


Figure 112 – EEH Web Analyzer

5. Troubleshooting Tips:

This chapter is created as an evolutionary document of troubleshooting problems and their solution. The best location to post your questions is in the Enterprise Exception Handling forums. Two public forums have been created to capture new feature ideas and for resolving technical issues with regards to this asset.

IBM Services Asset: Enterprise Exception Handling: Ideas and Features

This forum discusses any new Ideas and Features that can be implemented for future releases of Enterprise Exception Handler (EEH) Pattern. This is an opportunity for not only propose a Feature Request but to involve in discussions on the usefulness of the feature. The moderators of the forum will forward any new ideas to the EEH development team and feedback from the team will be posted here. EEH development may also participate in discussions.

<http://www.ibm.com/developerworks/forums/forum.jspa?forumID=2760>

IBM Services Asset: Enterprise Exception Handling: Technical Discussions

All technical discussions on the Enterprise Exception Handling (EEH) Pattern are hosted at this forum. The forum brings experts and architects and developers new to the EEH technology together to help implement EEH in the organization.

<http://www.ibm.com/developerworks/forums/forum.jspa?forumID=2761>

Valuable troubleshooting tips will be added to this chapter after moderating problems and solutions from the technical forum.

Adding and deleting database jars in the eclipse Exception Database Editor is now failing to load new ones. How do I fix the plug-in?

The Exception Database Editor has a dependent plug-in that is updated when database jars are added or deleted. This OSGi bundle for loading database jars, com.ibm.mq.exc.plugin.service_1.0.0, is independent of the editor plug-in com.ibm.mq.exc.plugin_1.0.0.jar. To fix any issues with the service plug-in, please replace it from the EEH install location:

[%EEH_PATH%/eclipse/site/plugins/com.ibm.mq.exc.plugin.service_1.0.0](#)
to the eclipse instance plugins directory. Please restart eclipse with the `-clean` option.

6. Service Offering from IBM:

This Category 01 SupportPac is an IBM Services Asset for WebSphere Software and may be freely downloaded under the license agreement. Please visit the [IBM Software Services Zone for WebSphere](#) website for detailed information on services offerings and to contact a services representative.

Engage the Experts: IBM Software Services for WebSphere (ISSW)

Engage IBM to provide knowledge and expertise on the Enterprise Exception Handling (EEH) framework for your organization:

- **Enterprise Exception Handling Workshop:** A discussion forum for gathering knowledge on the scale and the complexity of the implementation of EEH. Over the 2-day workshop IBM consultant will demonstrate the capabilities and the advantages for using the framework and extensibility. High-level planning guidelines for implementation of the standard.
- **Enterprise Exception Handling Strategy and Planning:** Introducing a standard at an organization level requires planning and strategy. This is true for engineering existing applications with EEH. IBM will assist in looking at development life cycle of different application teams to coordinate the effort to provide governance around exceptions in the system and the rollout of the exception handling framework. Typically an engagement would require 4-12 weeks.
- **Enterprise Exception Handling Implementation:** IBM will provide assistance in setting up the EEH environment with the database schema and implement exception handling for a number of Java applications or message flows to provide mentoring for implementation of the pattern. Typical engagement of 3-5 weeks is recommended.
- **Enterprise Exception Handling Extensibility:** EEH provides capabilities for the implementation of plugins – Renderers, Exception Notifiers, etc. IBM can provide value added services to create plugins for specific requirements. IBM can also provide services to customize the asset for client requirements or additional integration points that are not implemented. This activity depends on the requirements and scope of the implementation. It is recommended to conduct a workshop to determine the scope of the engagement.