ORACLE
INSURANCE

Oracle Insurance

Insbridge Rating and Underwriting

SoftRater Server User Guide

Release 4.7.1

November 2013

Copyright © 2005, 2013, Oracle and/or its affiliates. All rights reserved.

Oracle Insurance Insbridge Rating and Underwriting SoftRater Server User Guide

Release 4.7.1

Part # E50925-01

Library # E50929-01

November 2013

Primary Author: Mary Elizabeth Wiger

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this software or related documentation is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS

Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are “commercial computer software” or “commercial technical data” pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

This software is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications which may create a risk of personal injury. If you use this software in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure the safe use of this software. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software in dangerous applications.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

This software and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Where an Oracle offering includes third party content or software, we may be required to include related notices. For information on third party notices and the software and related documentation in connection with which they need to be included, please contact the attorney from the Development and Strategic Initiatives Legal Group that supports the development team for the Oracle offering. Contact information can be found on the Attorney Contact Chart.

The information contained in this document is for informational sharing purposes only and should be considered in your capacity as a customer advisory board member or pursuant to your beta trial agreement only. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described in this document remains at the sole discretion of Oracle.

This document in any form, software or printed matter, contains proprietary information that is the exclusive property of Oracle. Your access to and use of this confidential material is subject to the terms and conditions of your Oracle Software License and Service Agreement, which has been executed and with which you agree to comply. This document and information contained herein may not be disclosed, copied, reproduced, or distributed to anyone outside Oracle without prior written consent of Oracle. This document is not part of your license agreement nor can it be incorporated into any contractual agreement with Oracle or its subsidiaries or affiliates.

CONTENTS

LIST OF FIGURES

PREFACE

Welcome to the *Oracle Insurance Insbridge Rating and Underwriting SoftRater Server User Guide.* This manual describes the settings and options of Oracle Insurance Insbridge Rating and Underwriting SoftRater Server (IBSS). IBSS is the administrative tool for the SoftRater engine. The SoftRater engine is a multi-platform component within the Oracle Insurance Insbridge Rating and Underwriting (IBRU) system that executes the rules, rating, and underwriting instructions as defined by the user in Oracle Insurance Insbridge Rating and Underwriting RateManager.

The SoftRater engine is an EJB component hosted in the Application Server and accessible through HTTP SOAP Proxy, HTTP POST, and EJB – Direct JNDI interfacing. A database management system is used for content storage with support for all major DBMS vendors.

IBSS runs on:

- Oracle WebLogic (WebLogic)
- JBoss, a division of Red Hat (JBoss)
- IBM WebSphere (WebSphere)

AUDIENCE

This manual is intended for system administrators who are tasked with setting up and managing Insbridge rating environments. Users should be familiar with their company's environments, databases and the requirements of the end users.

RELATED DOCUMENTS

For more information, refer to the following Oracle resources:

- The Oracle Insurance Insbridge Rating and Underwriting Framework Administrator User Guide.
- The Oracle Insurance Insbridge Rating and Underwriting SoftRater User Guide.
- You can view these guides in-line at this address:

  http://www.oracle.com/technetwork/documentation/insurance-097481.html

CONVENTIONS

The following text conventions are used in this document:

Convention	Description
bold	Boldface type indicates graphical user interface elements associated with an action.
Italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`Monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

SYSTEM REQUIREMENTS

For minimum operating system and hardware requirements, please see the Hardware Software requirements guide.

Manual History

New editions incorporate any updates issued since the previous edition.

Edition	Publication Number	Product Version	Publication Date	Comment
1st Edition	P01-729-21	V 4.7	November 2013	UI Update

Chapter 1

SOFTRATER SERVER

Oracle Insurance Insbridge Rating and Underwriting SoftRater Server (IBSS) is an administrative tool used in conjunction with SoftRater for WebLogic, JBoss, or WebSphere. It allows you to:

- Configure the SoftRater engine
- Rate test files
- Execute a SoftData request
- Enter a SoftLibrary
- Manage an IBSS Clustered environment

CLUSTERED ENVIRONMENTS

IBSS allows for clustered environments. A clustered environment is where multiple instances of IBSS are created on one or more servers of the same type, sharing the same configuration. The benefits to a cluster are that all instances in the cluster work together to provide high availability, reliability, scalability, and increased performance.

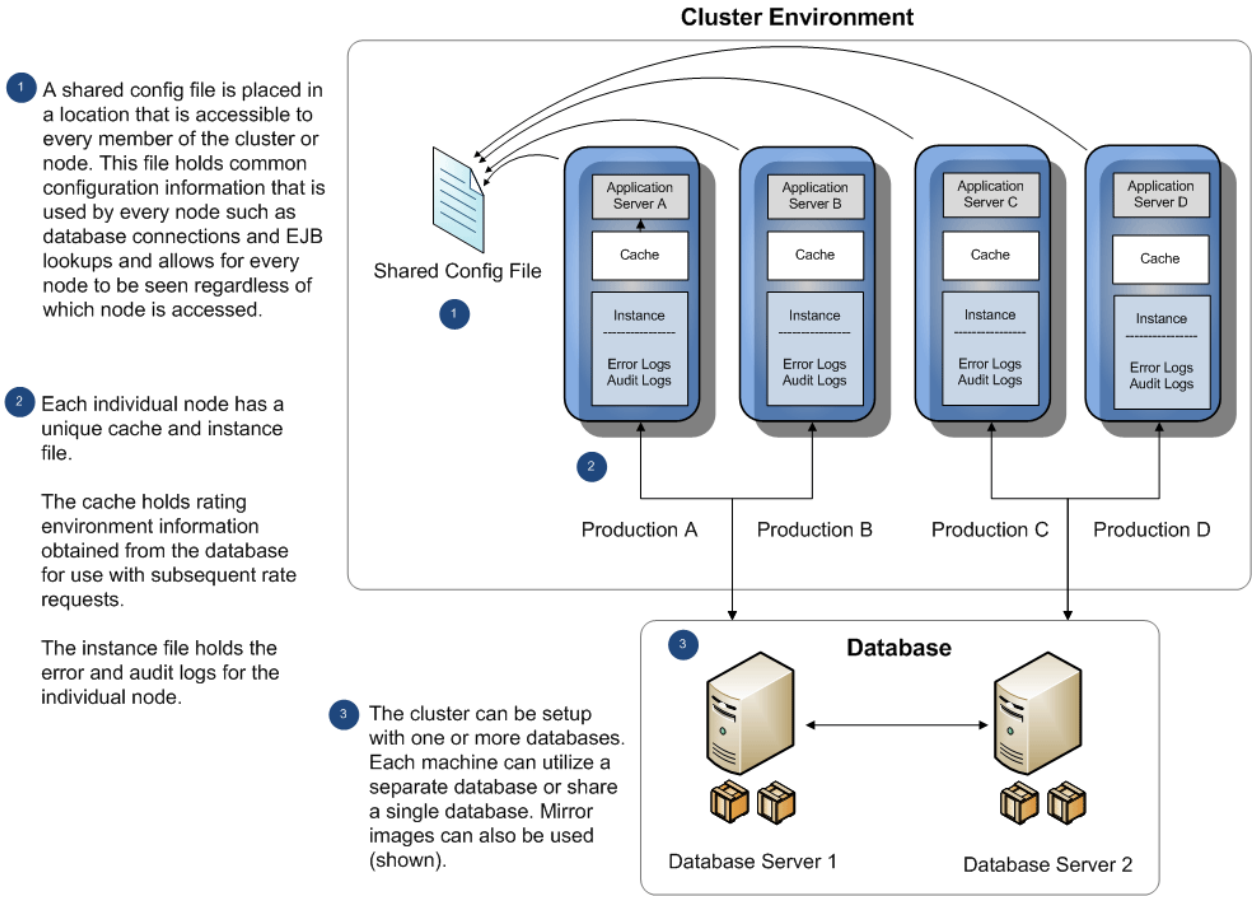


Figure 1 IBSS Cluster

Shared Information in an IBSS Cluster

Clusters in IBSS share common information and common features but each individual instance holds unique cache and log files. The common information includes the config.xml file, database environments, and SoftLibraries. This allows for information that is shared to be consistent throughout the IBSS cluster. For example, if you make a change to a database environment, the config.xml file gets updated. Since every IBSS instance in the cluster has access to the updated config.xml file, every instance is updated. Likewise if you make an update to a SoftLibrary, you do not need to update every IBSS instance. You can do one update. All IBSS instances in the cluster will have access to the updated SoftLibrary.

When you set up a cluster, each individual IBSS deployment is entered as a node. The node is the endpoint of the IBSS instance. Each node has an IBSS home page and can be presented on the home page of each IBSS instance. It does not matter from where you enter the environment or subscriber information. Each node shares the same configuration file so each node sees the same information.

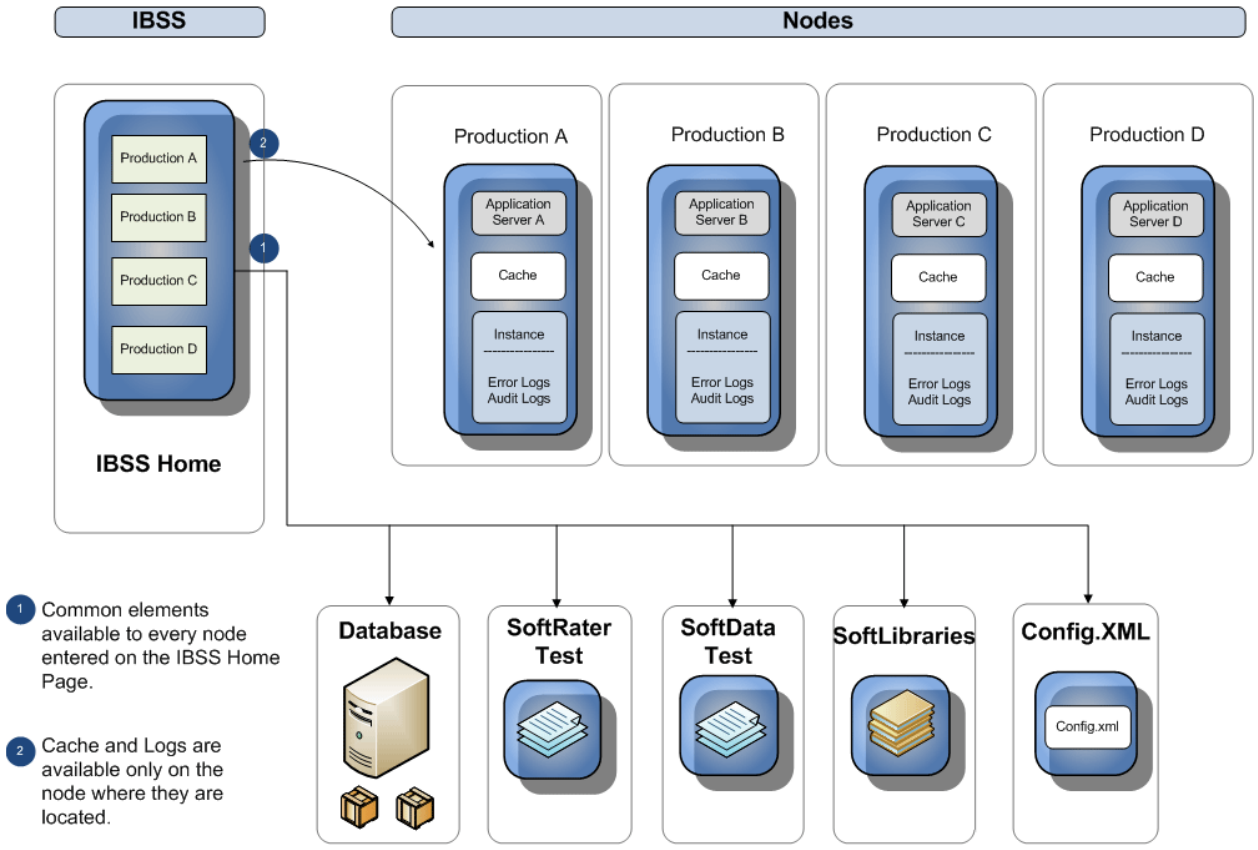


Figure 2 Shared Information in an IBSS Cluster

Once nodes are created in the IBSS cluster, SoftRater and SoftData can utilize any node for testing. For example, if you wanted to test a SoftData call on node A, you would select node A from the node option on the SoftRater Test Interface page. You would not have to enter the node A IBSS instance and perform the test from there. You can test from any node in the cluster. If you then wanted to test node B, you would select node B from the option. You would not have to enter the node B IBSS instance.

The areas that are unique to each node in the cluster are cache and Instance files. Cache holds the rating environment information obtained from the database. This information is used in subsequent rate requests. Cache can be purged across all members of the cluster or only for the selected node.

Another area unique to each node is the Instance file. The Instance file holds error and audit logs for each member of the cluster. These logs are unique to the instance and no information is shared. Individual error and audit logs can assist when debugging issues. You can see the actions performed by each IBSS instance without having to sort through logs that do not refer to the node in question.

IBSS Clusters

Any SoftRater for Java engine (WebLogic, WebSphere, or JBoss) can utilize clustering. The actual setup of a cluster on the application server(s) is up to the discretion of the installer, but there are a few requirements for a cluster such as each instance points to the same configuration file and that all instances must be of the same application server type. Once the cluster has been created, the installer will need to provide the server host(s) and the port(s) used along with any database information.

Creating a cluster in IBSS:

- Enter the subscriber information
- Enter the environments
- Enter the node information
- Enter the cluster in IBFA

An IBSS cluster can consist of one or more instances. Cluster information can be entered on any member of the cluster. The cluster setup must also be entered in the corresponding IBFA.

Standalone Instances

It is possible to have multiple IBSS instances that are not part of a cluster. These standalone instances will each have their own config file and cannot be managed from a common instance because there is no shared config file. This means that every standalone IBSS instance would require subscriber, environment, and node information to be entered. This is not the recommended setup.

Ports Used by WebLogic, JBoss, and WebSphere

WebLogic, JBoss, and WebSphere utilize ports to access IBSS:

- WebLogic uses port 7001
- JBoss uses port 8080
- WebSphere uses port 9080

NOTE: *These are the most commonly used ports. The port number may be changed when domains, cells, nodes, servers or instances are created.*

If the designated port is unavailable or there is a conflict, your system administrator can change the port. For example, you can change the WebLogic default port (7001) during the Server Domain creation.

If you cannot find the IBSS instance you need at the port location specified, please check the port being used with your system administrator.

NOTE: *The IBSS parameter in the application URL must be capitalized.*

WebSphere

For a WebSphere environment, IBSS can be accessed by going to http://<yourserver>:9080/IBSS, where <yourserver> is the server the IBSS is installed on.

If you do not know this information, contact your system administrator.

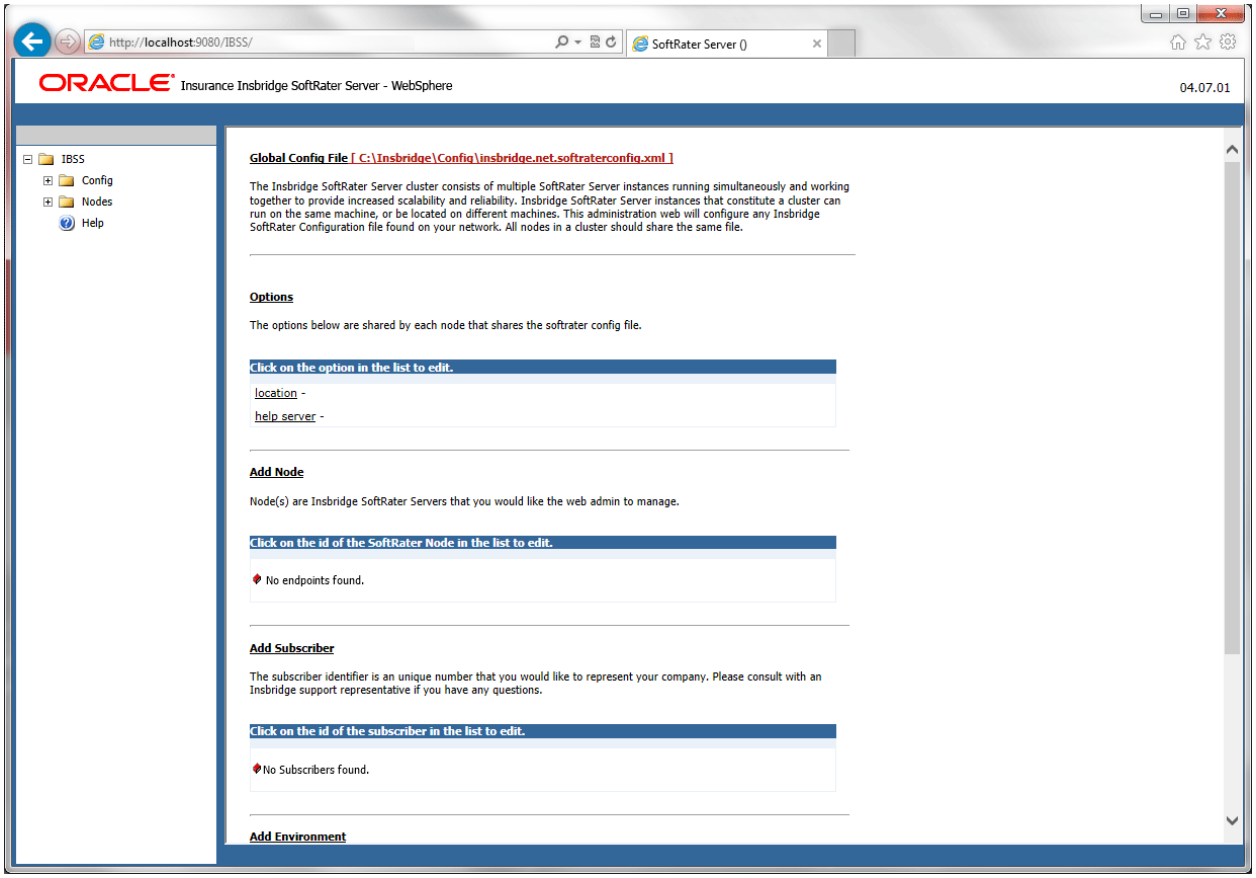


Figure 3 SoftRater Server for WebSphere

NOTE: *The top of the screen lists the platform you are on and the version of IBSS.*

JBoss

For a JBoss environment, IBSS can be accessed by going to http://<yourserver>:8080/IBSS, where <yourserver> is the server the IBSS is installed on.

If you do not know this information, contact your system administrator.

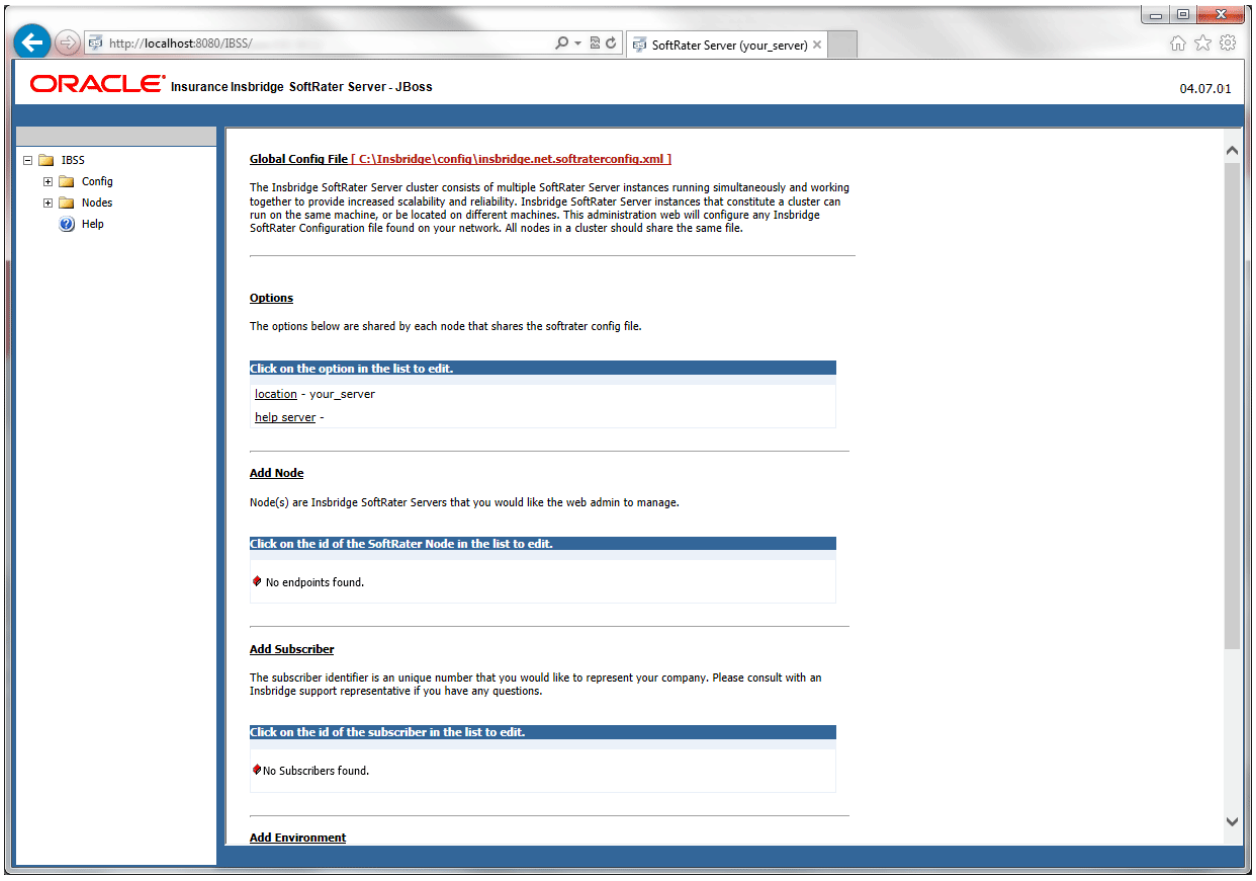


Figure 4 SoftRater Server for JBoss

NOTE: *The top of the screen lists the platform you are on and the version of IBSS.*

WebLogic

For a WebLogic environment, IBSS can be accessed by going to http://<yourserver>:7001/IBSS, where <yourserver> is the server the IBSS is installed on.

If you do not know this information, contact your system administrator.

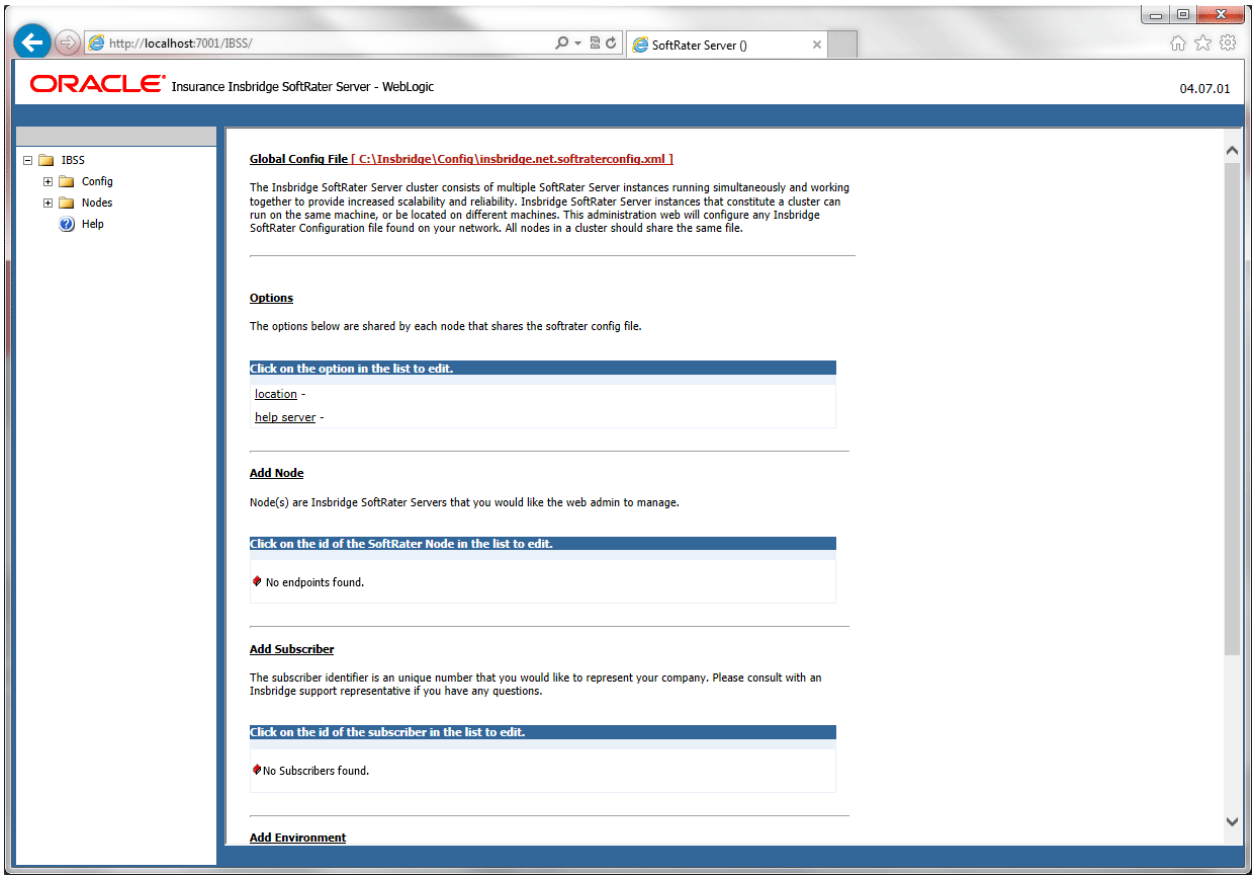


Figure 5 SoftRater Server for WebLogic

NOTE: *The top of the screen lists the platform you are on and the version of IBSS.*

Chapter 2

IBSS HOME PAGE

The IBSS Home page is the landing page whenever you open IBSS. Before you can use the SoftRater engine, subscriber, environments, and node information must be entered. Subscriber and environment information is generally done at installation time, but may be entered or updated at a later date if necessary. Node information can be done later as well.

The IBSS Home page allows you to set up and manage:

- Subscribers
- Environments
- Nodes
- Options
- View the configuration file

This page operates the same way for WebLogic, JBoss, or WebSphere.

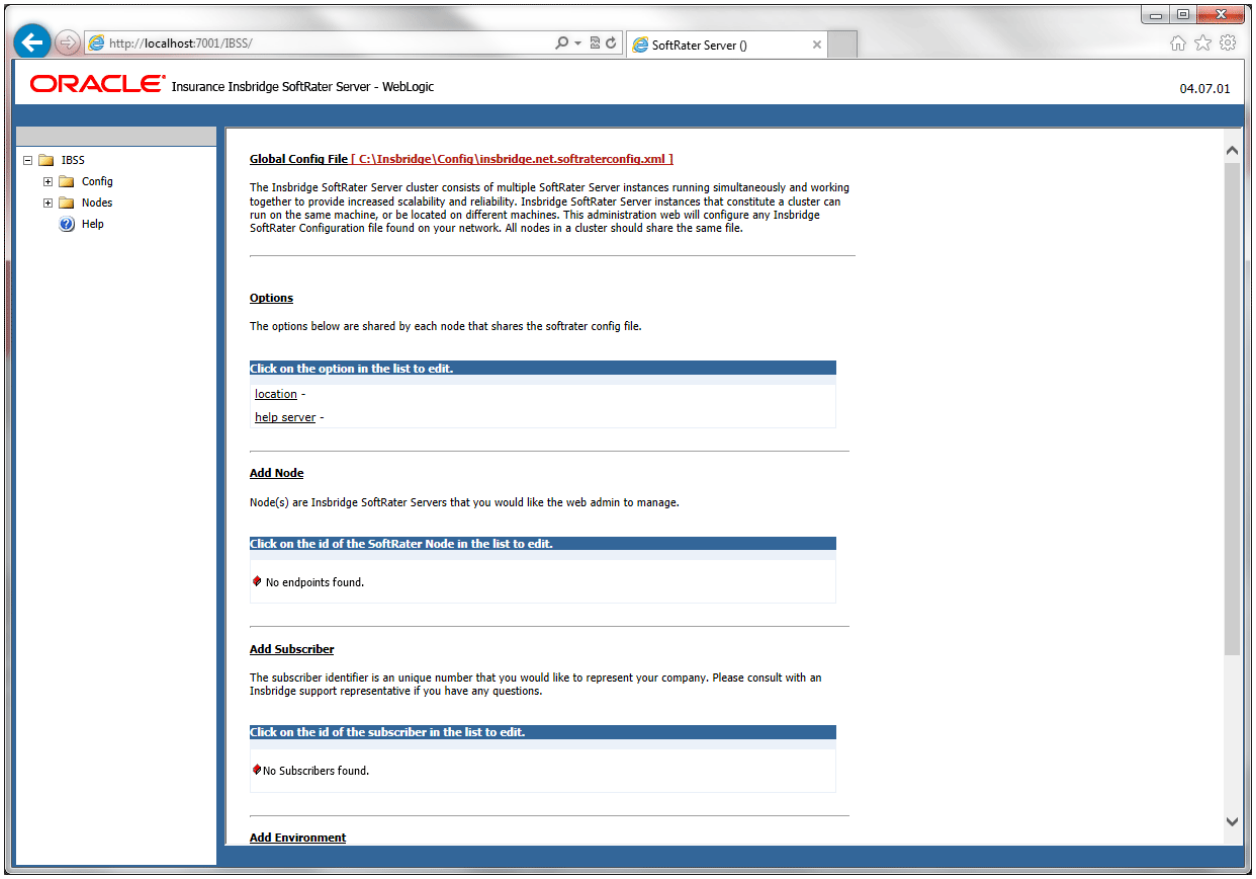


Figure 6 SoftRater Page

CONFIGURATION FILE

IBSS utilizes a configuration file (Insbridge.net.softraterconfig.xml) that is parked on the server where IBSS is deployed. When you make a change to subscribers, environments, etc. using IBSS, the configuration file gets updated. If the configuration file is shared across multiple IBSS instances, for example in a clustered environment, then every IBSS instance in the cluster has access to the update. The configuration file contains settings for:

- Configuration information
- Environments
- Database Connections
- SoftLibrary information

NOTE: *Clustered environments must use the same config file.*

Viewing the Configuration File

Complete file information is not available until you have configured all environments.

1. On the IBSS Home page, click the underlined configuration file path at the top; i.e. **C:\INSBRIDGE\Config\insbridge.net.softraterconfig.xml.**

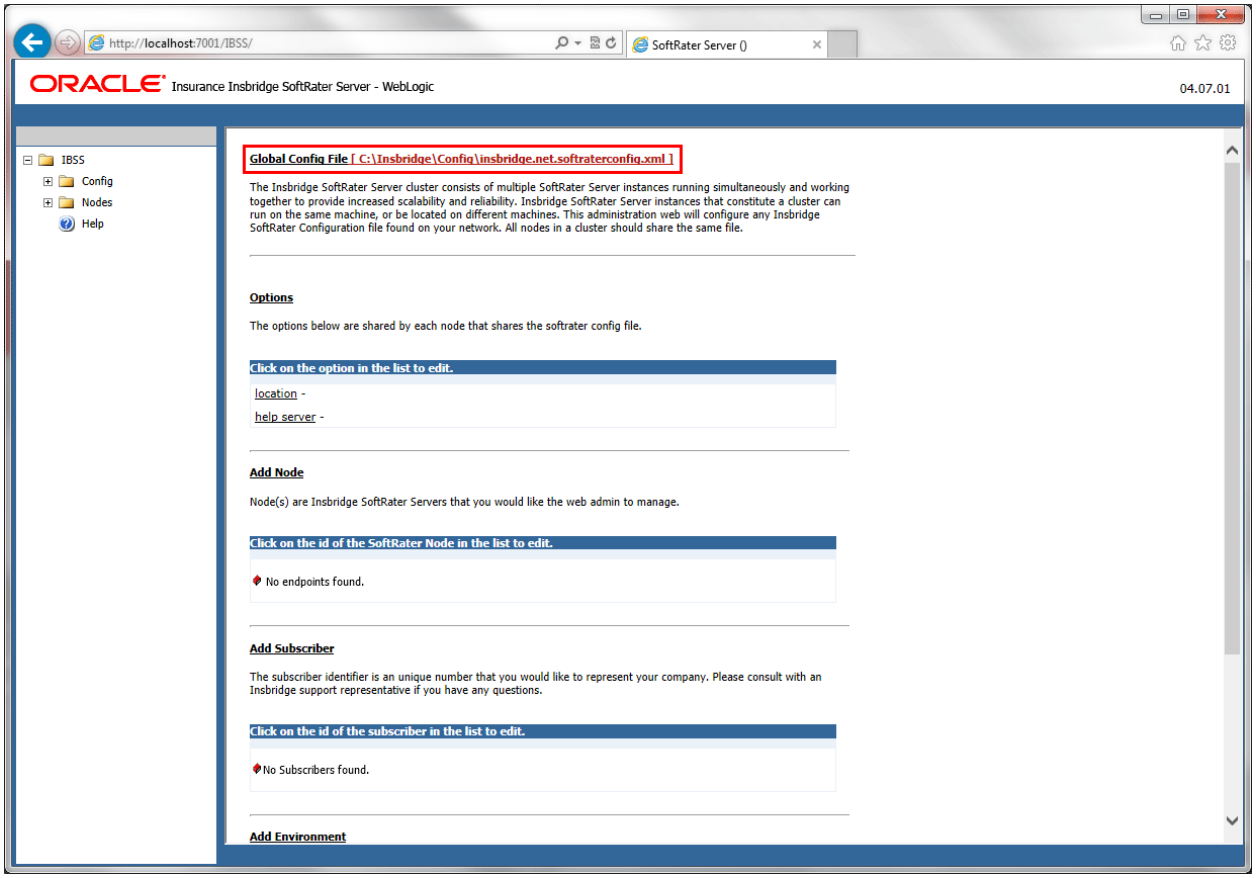


Figure 7 Insbridge Page Configuration File Path

2. The configuration file opens in a new window.

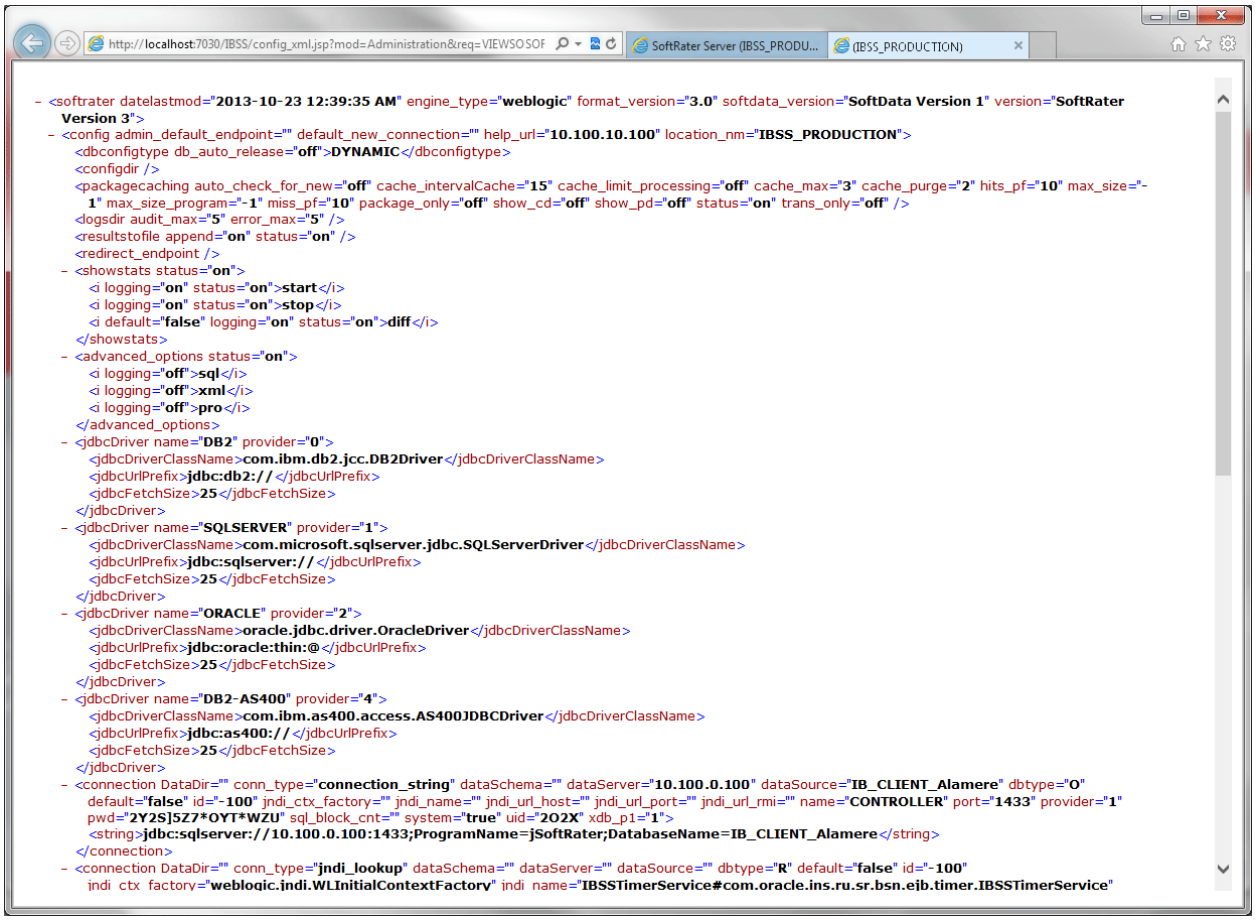


Figure 8 Configuration File New Window

NOTE: *The configuration path is set at installation. You may have a different configuration path than the one in the example.*

In a clustered environment, the same configuration path is displayed for every member of the cluster.

If you have multiple IBSS instances that do not share the same config file, the configuration path is the path for the current instance only.

OPTIONS

Options include the friendly name used to identify the server and the location of the help server. Options are shared by each node.

The location and help server fields are validated for entries not for content.

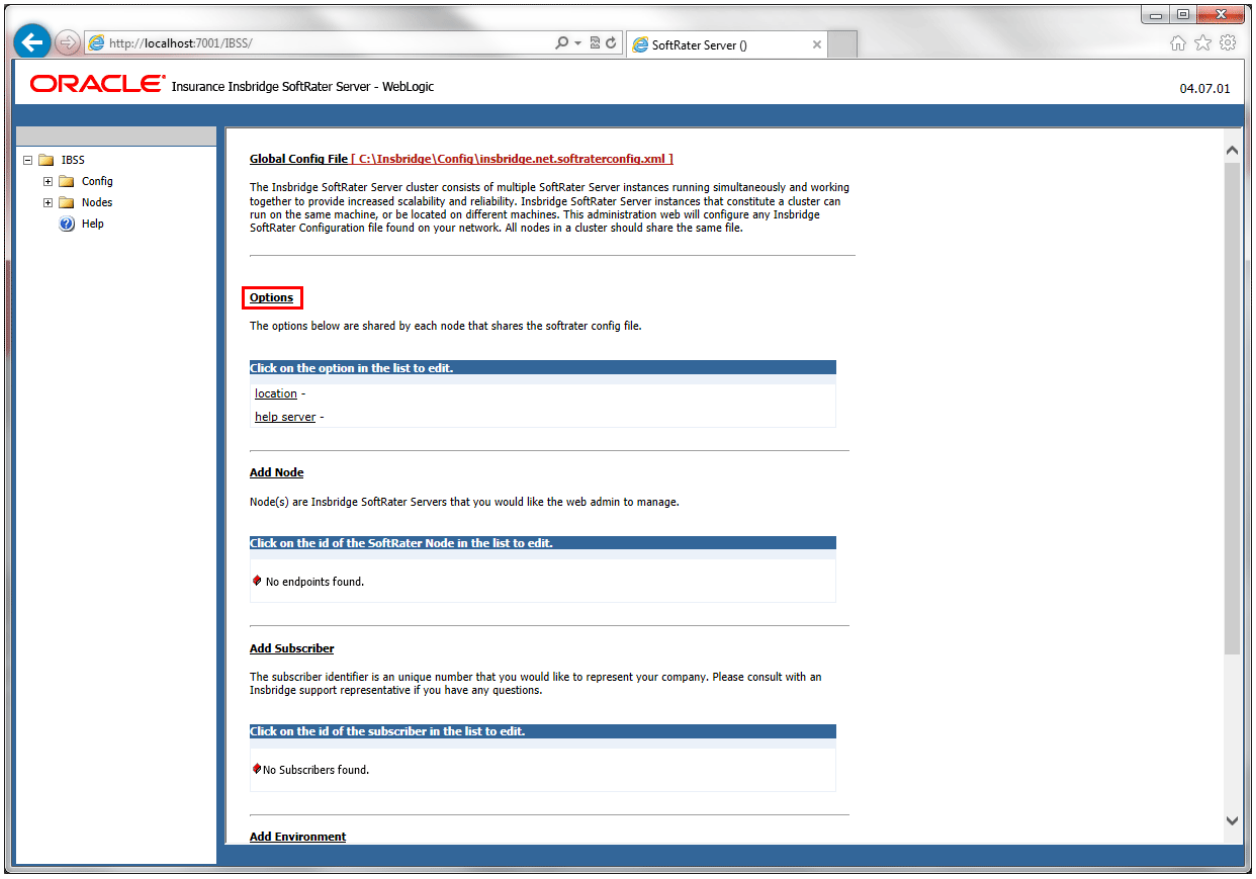


Figure 9 Option on the IBSS Home Page

Location Name

The location name is the name of the server where IBSS is located. The Location Name is the friendly name that is returned as part of the XML response content for each transaction to show the originating site for the result. The location name also is displayed on the top SoftRater Server tab on the browser.

Help Server

Help files are installed with IBFA. Enter the name or IP address of the server where IBFA was installed. This allows users to access the help files when they click on the help link. Leaving this field blank results in an error when users click the Help link on the IBSS home page.

You may need to contact your system administrator for the location of IBFA.

Setting Options

The options window can be opened by clicking the Options link on the IBSS home page or clicking the location or help server links.

1. Click the Options link on the IBSS home page. A separate window is displayed.
2. Enter in the Location Name. This can be a server name or IP address.
3. Enter in the Help Server. This can be a server name or IP address.
4. Click Save. The entries are displayed on the IBSS home page immediately.

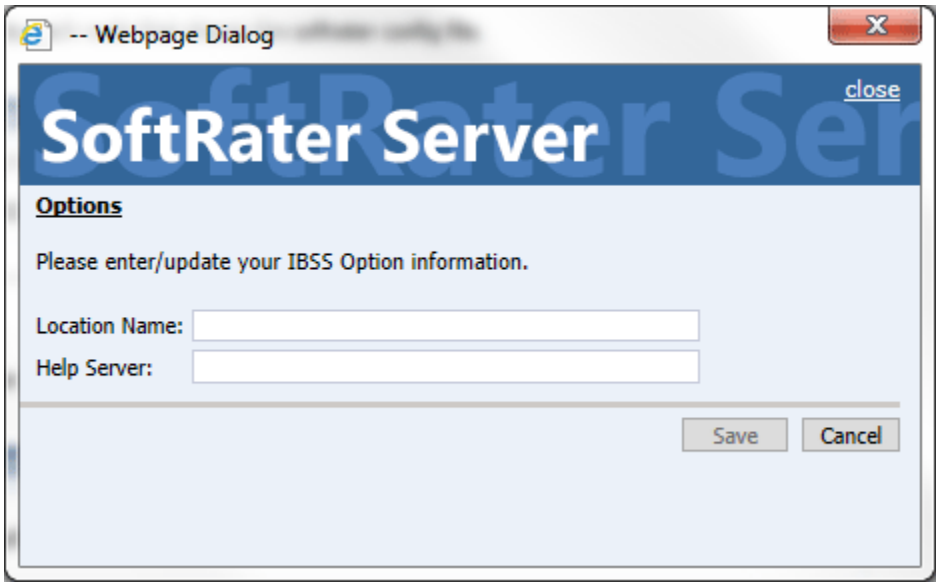


Figure 10 Entering Options

Editing Options

Options can be edited at any time. Once an entry has been made in the field, the field must have a value. You cannot remove the entry and leave the field blank.

Any updates or changes must be saved prior to closing the window. Cancel disregards any changes and closes the window.

Deleting Options

Options cannot be deleted.

SUBSCRIBERS

A *subscriber* is a group or collective of users and environments under a single identity. This identity is used to manage and configure the group's activities and databases. A subscriber may be a single company, a group within a company, or a single person.

Any subscriber that is in IBFA must be created here also.

Adding a Subscriber

You can add a subscriber when IBSS is installed or later. The same subscriber information entered in IBFA should be entered in IBSS. The difference between an IBFA subscriber and an IBSS subscriber is the description. A description can be entered in IBFA but is not needed in IBSS.

The Subscriber IDs must match between IBFA and IBSS. The subscriber ID can be a 1-5 digit number and must unique. The ID is fixed after you save and cannot be edited. In IBSS, only the name can be edited. If you have entered the subscriber ID incorrectly, you must re-enter the subscriber.

If you are using nodes, you only have to enter the subscriber information once. Each node is populated with the same subscriber information.

1. From the IBSS home page, click the **Add Subscriber** link.

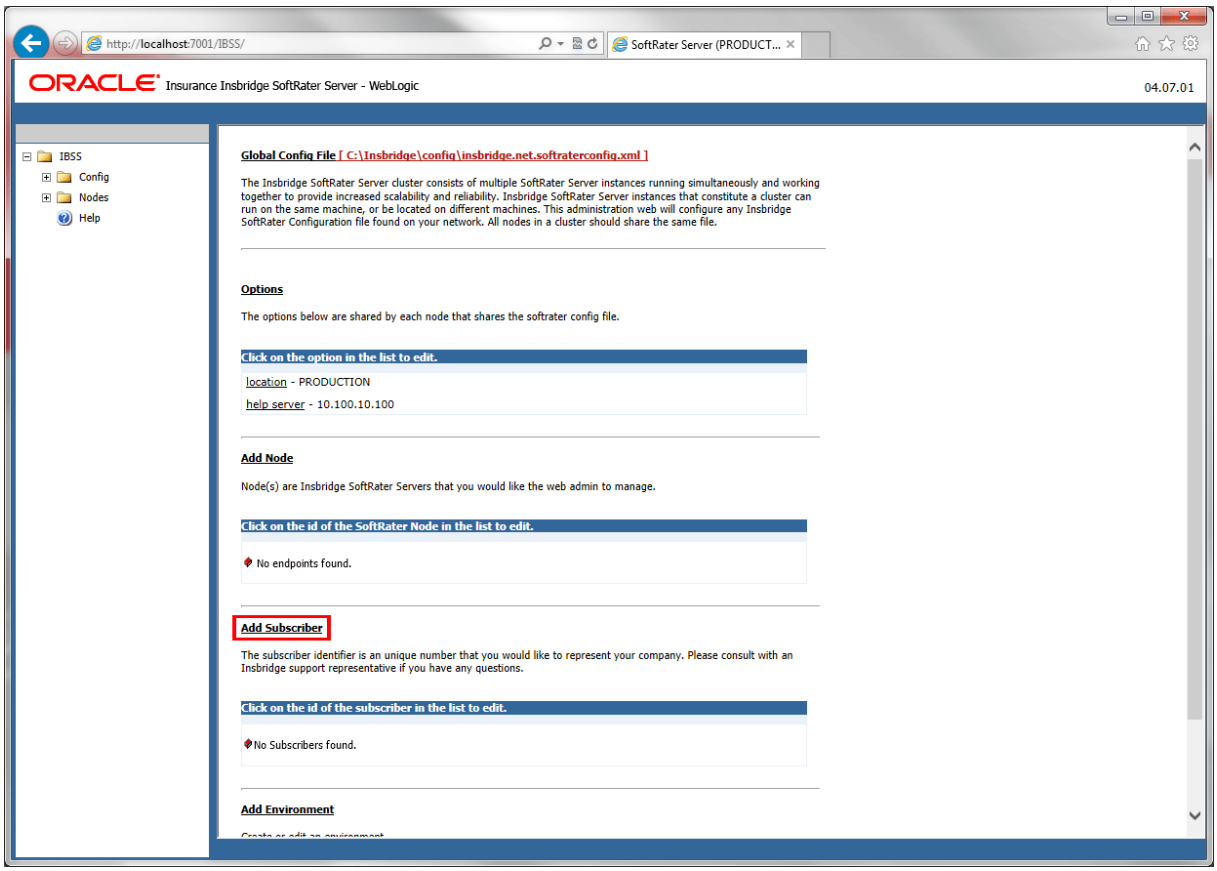


Figure 11 Adding Subscriber Links

2. This opens the **Add Subscriber** window.

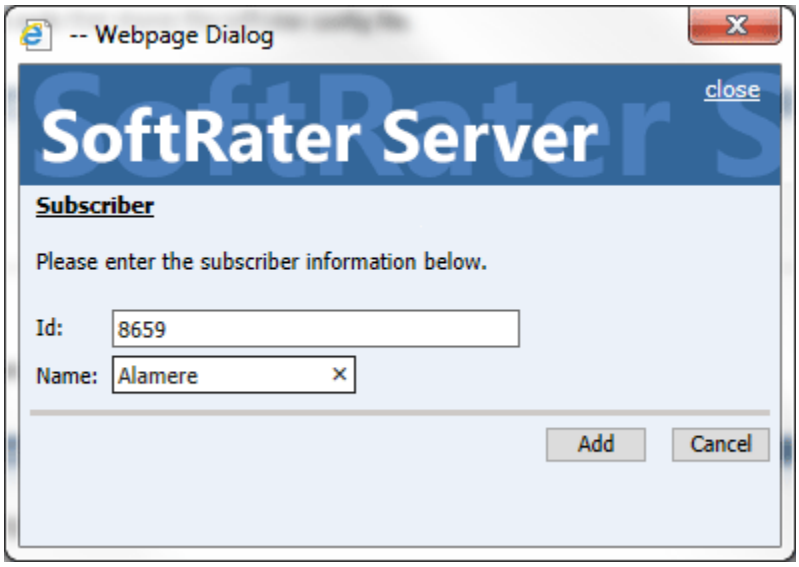


Figure 12 Adding a New Subscriber

3. Enter the **ID**, and **Name** for the new subscriber. When you finish entering the information, click **Add**.

4. The new subscriber is now the list of subscribers.

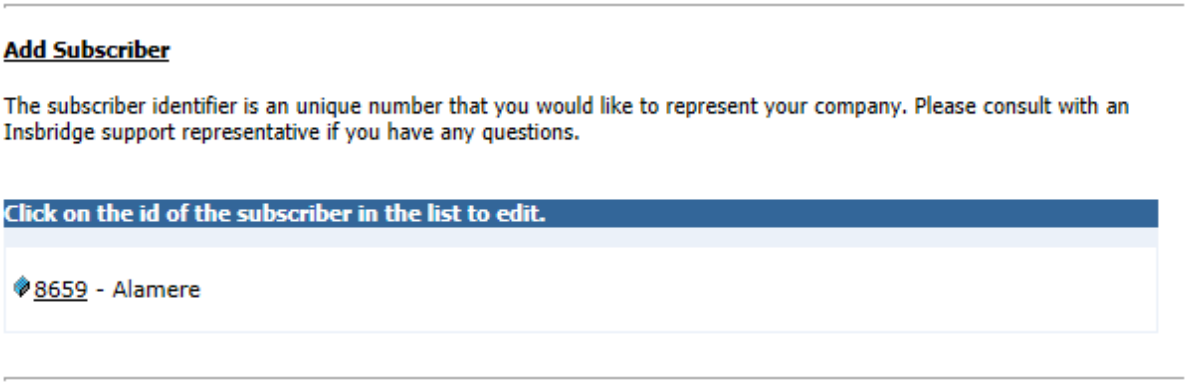


Figure 13 Subscriber List

Editing a Subscriber

The only edit allowed to a subscriber is the Name.

To edit a Subscriber, click on the hyperlinked Subscriber ID number. This brings up the Edit Subscriber window. Make any changes and click Save to save your changes and return to the previous window.

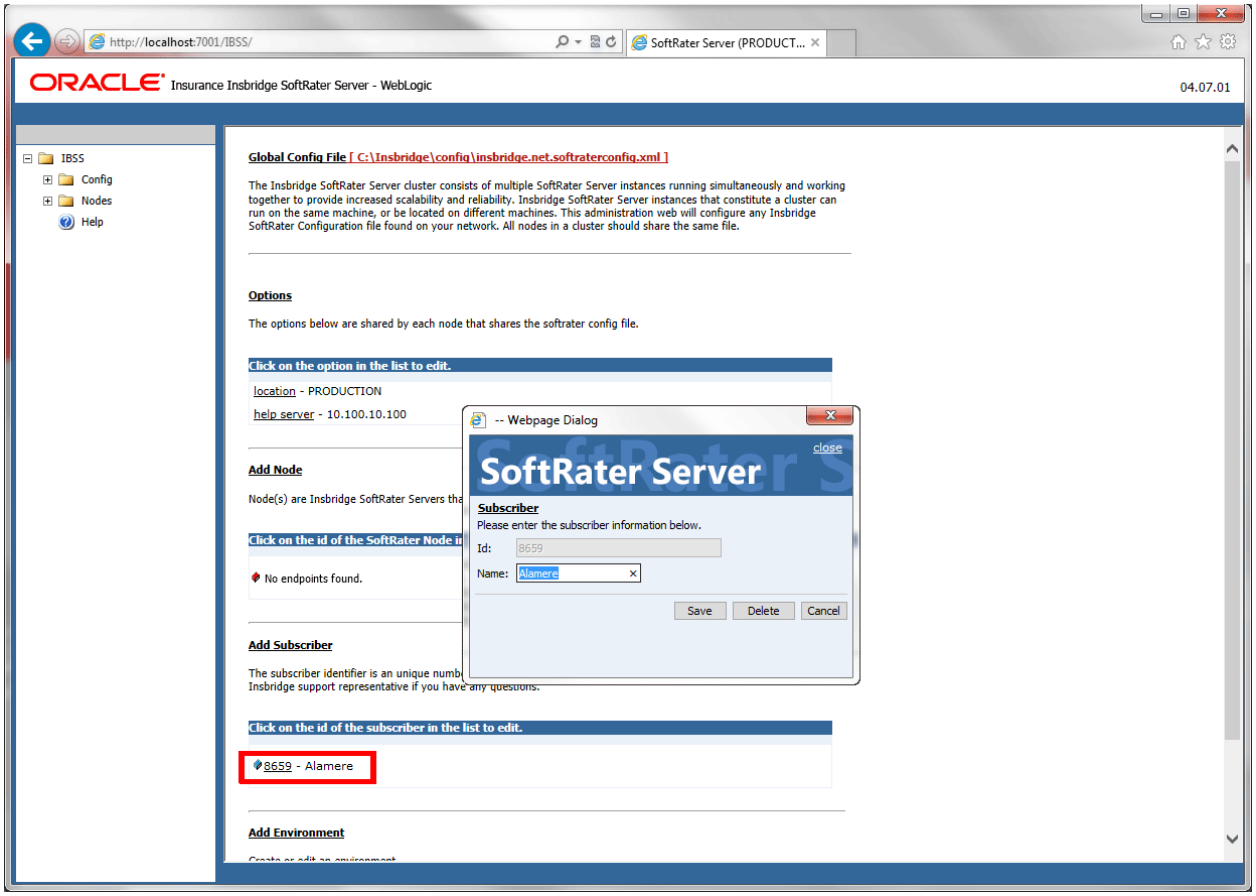


Figure 14 Editing Subscriber

Deleting a Subscriber

If a subscriber is no longer needed, they can be deleted. Make certain a subscriber is not needed before deleting them, as this action cannot be undone. When a subscriber is deleted, all associated database connections and environment settings are also deleted.

1. From the IBSS home page, click the hyperlinked Subscriber ID number.
2. This opens the **Edit Subscriber** window.

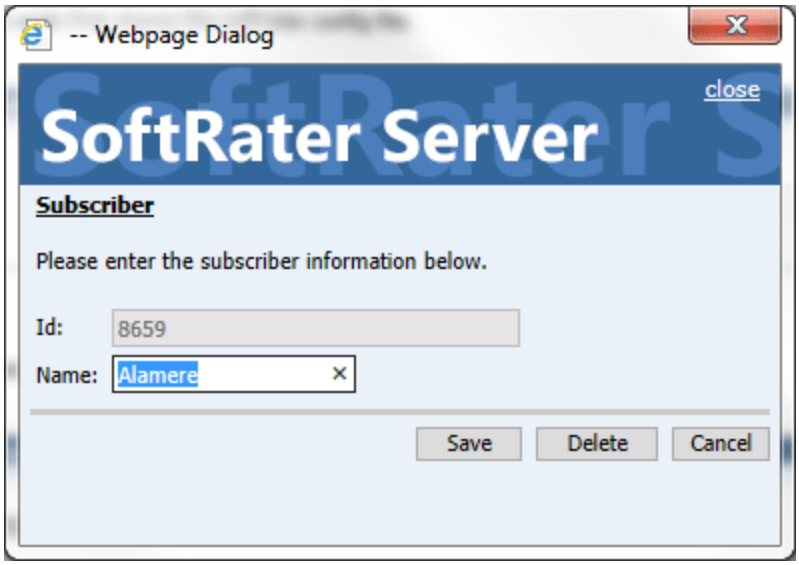


Figure 15 Deleting a Subscriber

3. Click the **Delete** button. A warning message is displayed.

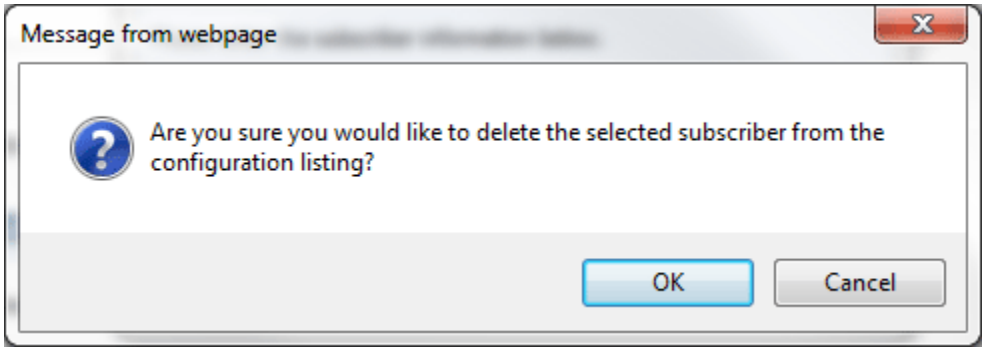


Figure 16 Confirm Deleted Subscriber

4. Click **OK** to remove the Subscriber or **Cancel** to return to the previous window.

WARNING: ***Deleting a subscriber also deletes all database connections and environments setup for that subscriber.***

ENVIRONMENTS

The **Environment** section allows you to configure rating environments. An environment is a pointer to a location where SoftRater Packages (SRP's) are stored and loaded. There is no limit to the number of environments you may have. The Add Environment link is located on the IBSS home page.

Enter any environment that you want to share across the cluster. Environments entered here can be discovered by the IBFA instance when creating the IBSS cluster environment. IBSS cluster environments must be entered in IBFA. You only have one space to enter an environment, but as long as the cluster has all the databases entered, IBFA will be able to locate every environment entered.

Adding an Environment

Prior to creating up an environment in IBSS you will need the database server, port, database name, user ID, and password information. You also may need Query block or schema name if you are entering a DB2 or Oracle database.

1. To create a new environment, click the Add Environment link on the IBSS home page.

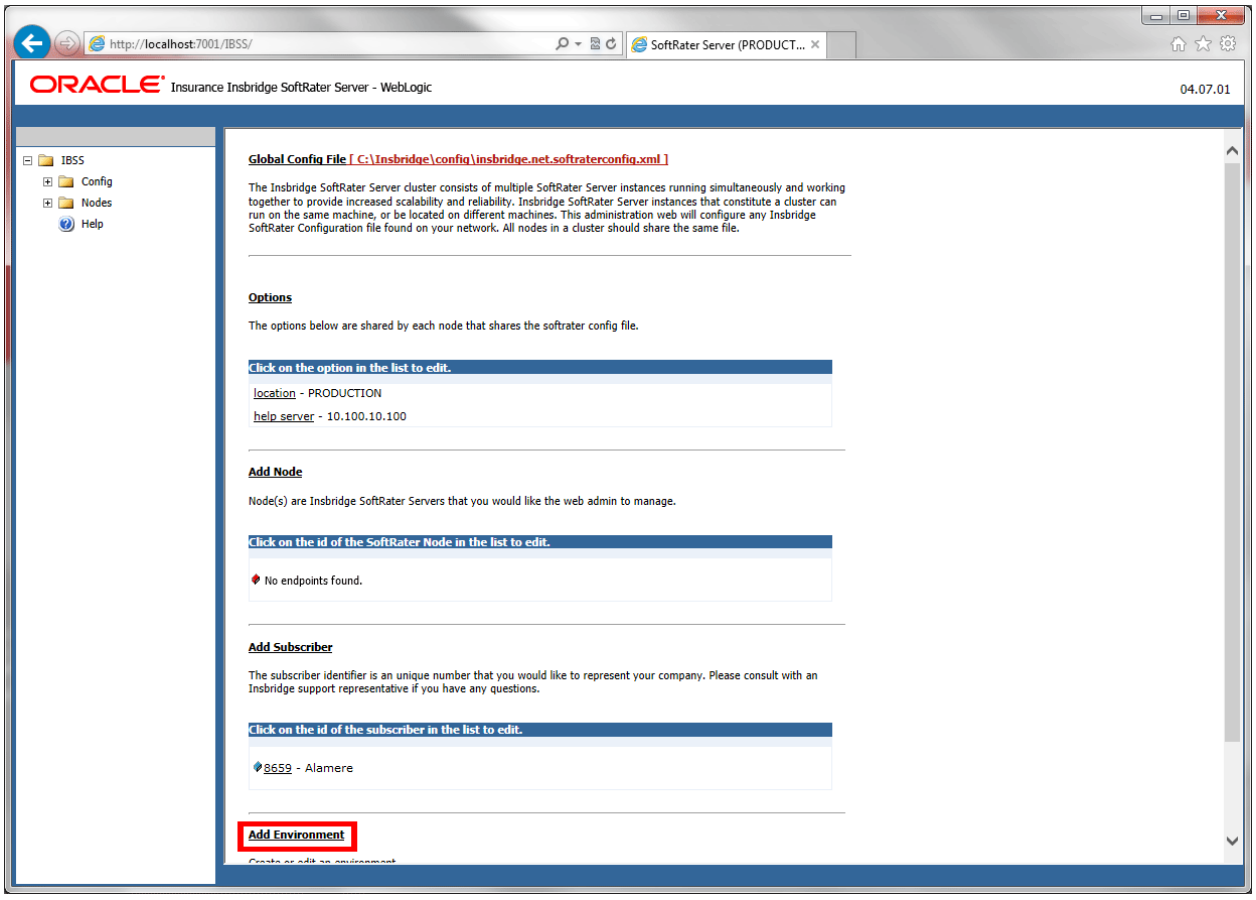


Figure 17 Add Environment

2. A separate screen is displayed. Enter the following information:

For DSN Entries

Subscriber: Select the name of the subscriber from the menu.

Environment Name: A name for the environment. This is the friendly name.

Environment Type: Select the type of connection:

- **DSN** – The connection information is held by the IBRU system.
- **JNDI** – The connection is an EJB Lookup and information is not held by the IBRU system.

Database Server: The name or IP address of the server where packages are loaded.

Database Port: Listening port for the target Database Management System.

Default port numbers:

- SQL Server – 1433
- Oracle – 1521
- DB2 – 50000

Query Block: For certain Database Management Systems (DBMS), system batch query optimization is required for performance and/or because of limits on/in the system. This value will control the maximum number of batch query operations that can be submitted to the target DBMS in any one transaction.

For DB2, enter a query size that is relative to the SQL statement heap size. For example, if your SQL statement heap size is 2048, you may want to enter a query size of 5. The larger the heap size, the larger the query size can be.

You may also need to increase the application heap size.

Query block is not needed for SQL Server or Oracle databases.

Database Name: The name of the database where packages are loaded.

Schema Name: The schema name for the database. Schema Name is not needed for SQL connections.

Database User Id: The username to access the database.

Database Password: The password for the username entered.

Provider Type: The type of the database

- **ORACLE**
- **DB2**
- **DB2(AS400)**
- **MS SQL SERVER**

Database Type: The type of database being accessed:

- **SoftRater – The default. This option should be used for most installations.**
- **RateManager** – Supported for selected installations. Please contact Support.
- **OTHER** – Not used at this time.

Data Dir: The location of the SoftRater Packages (SRP's) on the server.

Default Environment: Select if this is to be used as the default environment. The default environment is used when no environment is specified. There can be only one default environment. The default environment is indicated with a red icon.

3. When you are finished entering information, click **Save**.

Figure 18 Environment Setup

4. Before you can use the new or edited environment for rating, you need to reload the configuration XML file. This is done by resetting the environments. On the left hand side of the IBSS home page, expand Nodes and select the Reset Environments option. You will be placed on the Reset Environment Configuration page. Click **Execute**. The screen refreshes and each node will be reset. Click IBSS to return to the home page.

For JNDI Entries

Subscriber: Select the name of the subscriber from the menu.

Environment Name: A name for the environment. This is the friendly name.

Environment Type: Select the type of connection:

- **DSN** – The connection information is held by the IBRU system.
- **JNDI** – The connection is an EJB Lookup and information is not held by the IBRU system.

Name: The name of the data source.

Context Factory: A URL context factory is a special object factory that creates contexts for resolving URL strings.

RMI: The Java Remote Method Invocation used.

URL Host: The name or IP of the host server.

URL Port: The port number used by the host server.

Provider Type: The type of the database

- **ORACLE**
- **DB2**
- **DB2(AS400)**
- **MS SQL SERVER**

Database Type: The type of database being accessed:

- **SoftRater – The default. This option should be used for most installations.**
- **RateManager** – Supported for selected installations. Please contact Support.
- **OTHER** – Not used at this time.

Data Dir: The location of the SoftRater Packages (SRP's) on the server.

Default Environment: Select if this is to be used as the default environment. The default environment is used when no environment is specified. There can be only one default environment. The default environment is indicated with a red icon. 

5. When you are finished entering information, click **Save**.

Figure 19 Creating a JNDI Environment

6. Before you can use the new or edited environment for rating, you need to reload the configuration XML file. This is done by resetting the environments. On the left hand side of the IBSS home page, expand Nodes and select the Reset Environments option. You will be placed on the Reset Environment Configuration page. Click **Execute**. The screen refreshes and each node will be reset. Click IBSS to return to the home page.

Editing an Environment

Environments can be edited at any time. The subscriber cannot be edited. If you have selected the wrong subscriber you must delete the incorrect entry and re-enter with the correct subscriber.

1. To edit an environment, click the Environment link on the IBSS home page that you want to edit.

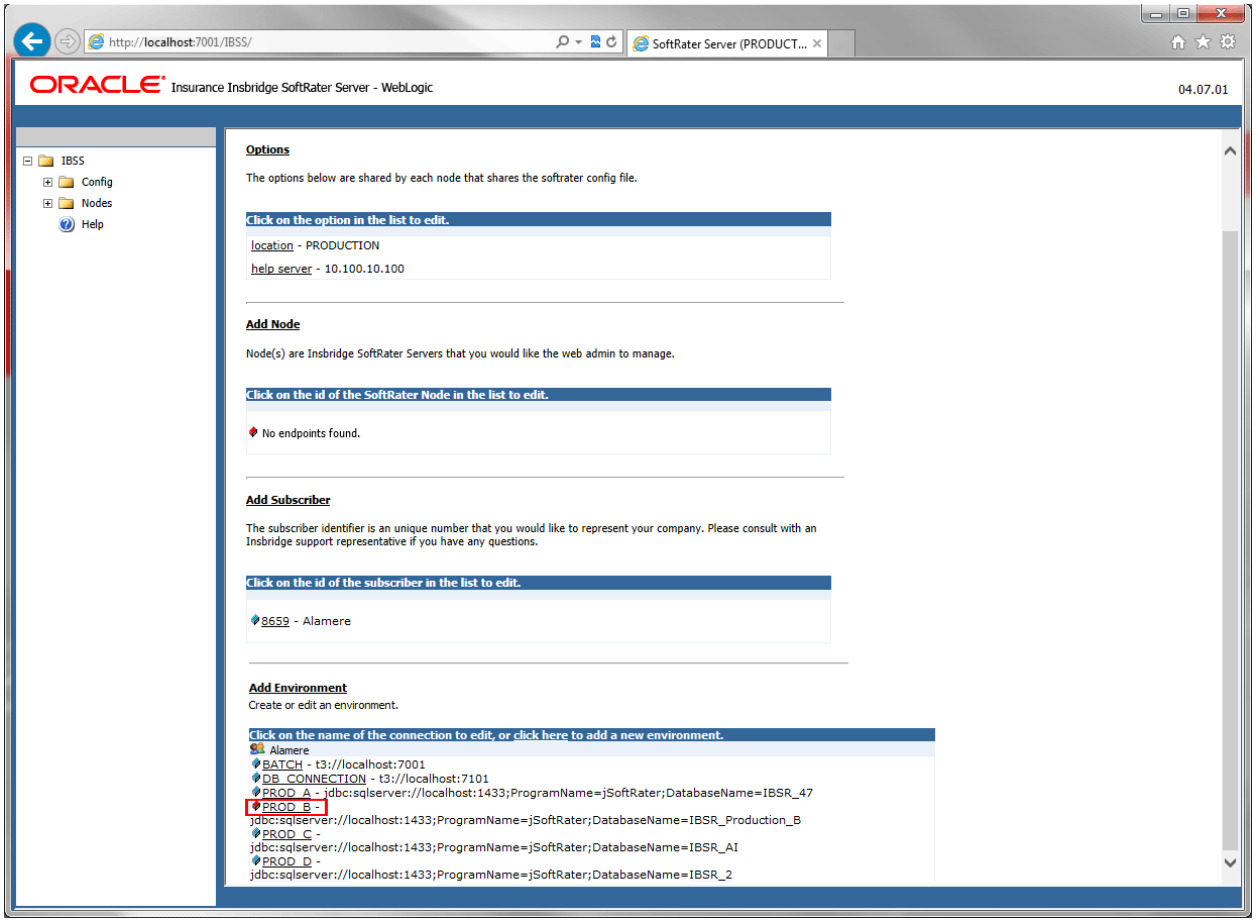


Figure 20 Add Environment

2. The environment information window is presented. The subscriber information cannot be edited. All other fields are open for editing. The field highlighted in blue is the field you are currently in.

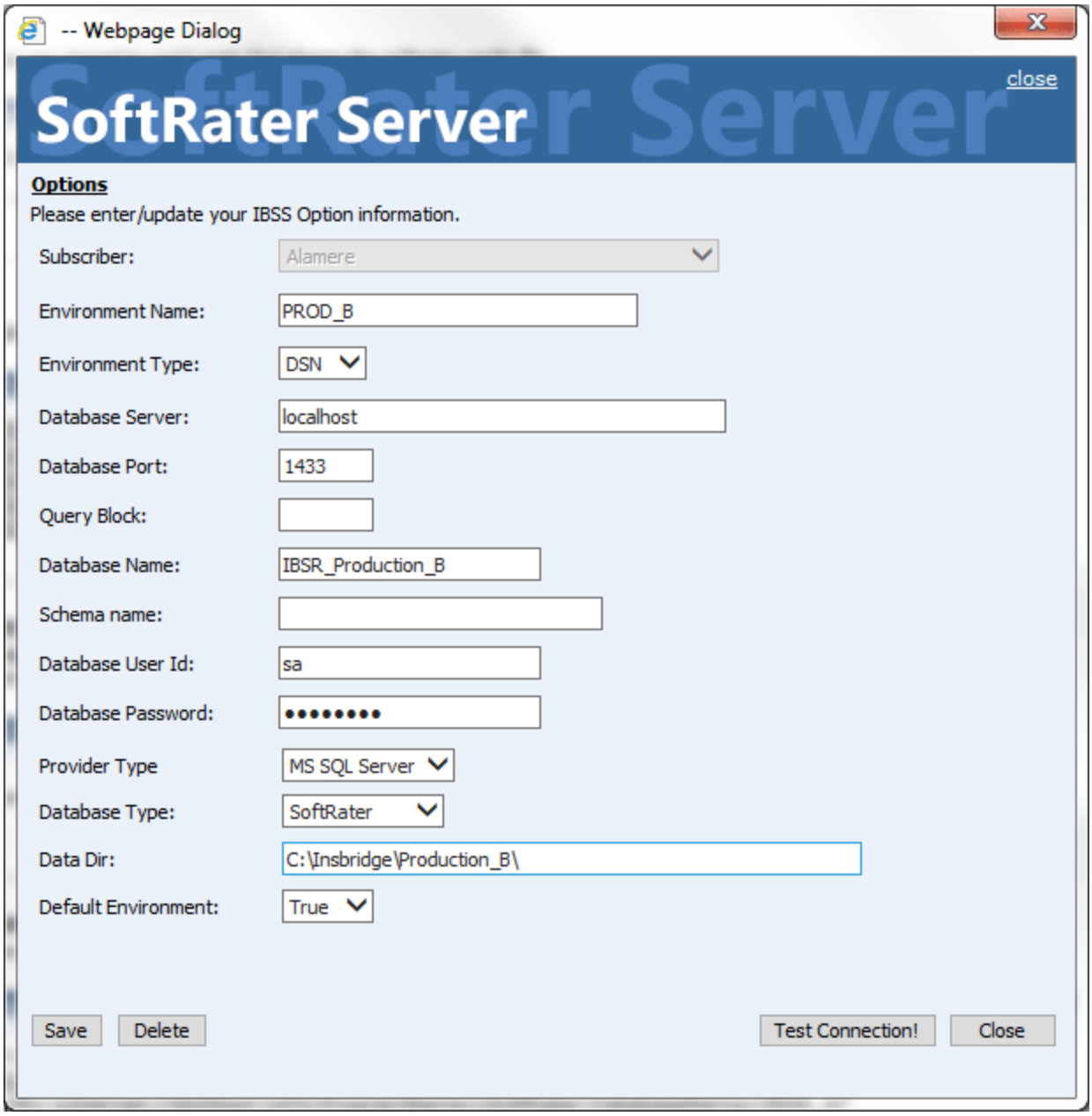


Figure 21 Editing an Environment

3. When you are finished editing information, click **Save**.

4. Before you can use the new or edited environment for rating, you need to reload the configuration XML file. This is done by resetting the environments. On the left hand side of the IBSS home page, expand Nodes and select the Reset Environments option. You will be placed on the Reset Environment Configuration page. Click **Execute**. The screen refreshes and each node will be reset. Click IBSS to return to the home page.

Testing an Environment

You should test your environment after saving to ensure the connection properties are valid. After saving the environment entry, click **Test Connection**. The results will be displayed in a message window.

Success Message

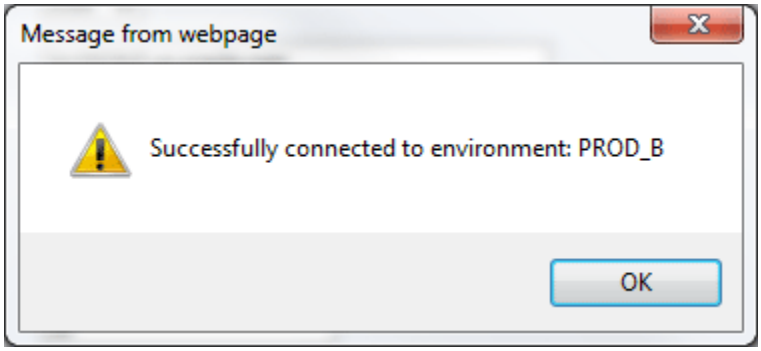


Figure 22 Successful Test Connection

Failed Messages

Failed messages from the test connection can show some common errors, such as typos, bad logins and failing to save.

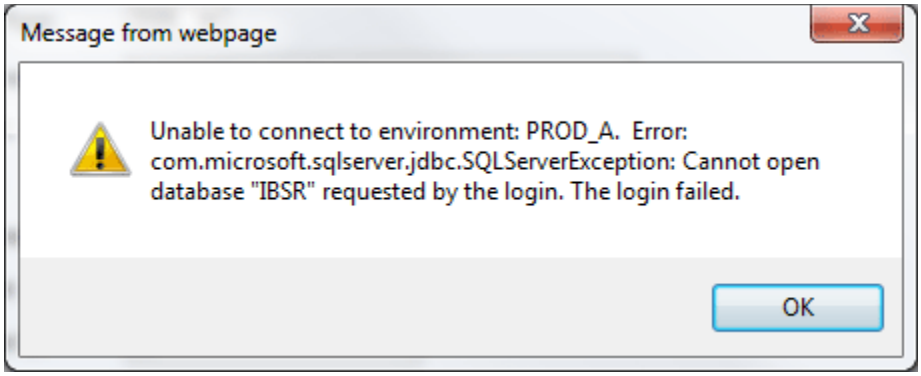


Figure 23 Failed Test Connection Bad Login Information

The cannot open database…the login failed error can sometimes results from bad user ID or password being entered.

If a database location or port number is incorrect, the information returned in the message may direct you to verify the connection properties, check the instance is running, check that the instance ports are open or that no fire wall is prohibiting the connection.

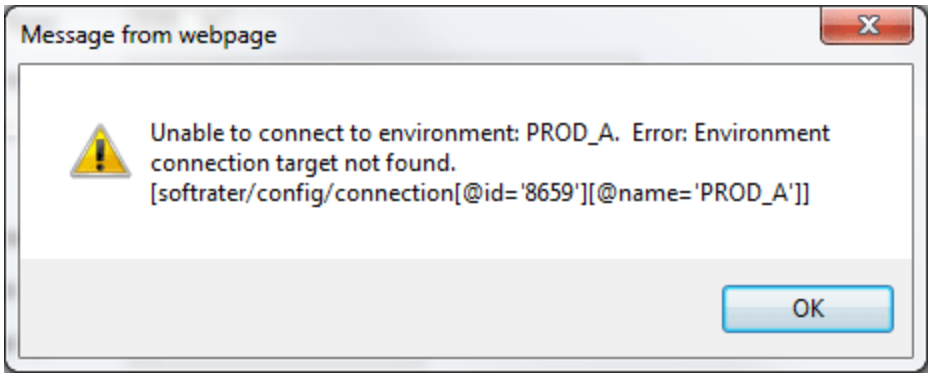


Figure 24 Failed Test Connection Not Saved

An environment connection target not found error sometimes results from the environment not being saved after updates. Save the environment and try again.

Deleting an Environment

An environment can be deleted at any time.

1. To delete an environment, click the Environment link on the IBSS home page that you want to permanently remove.
2. The environment information window is presented, click **Delete**.
3. You must confirm deleting the environment.

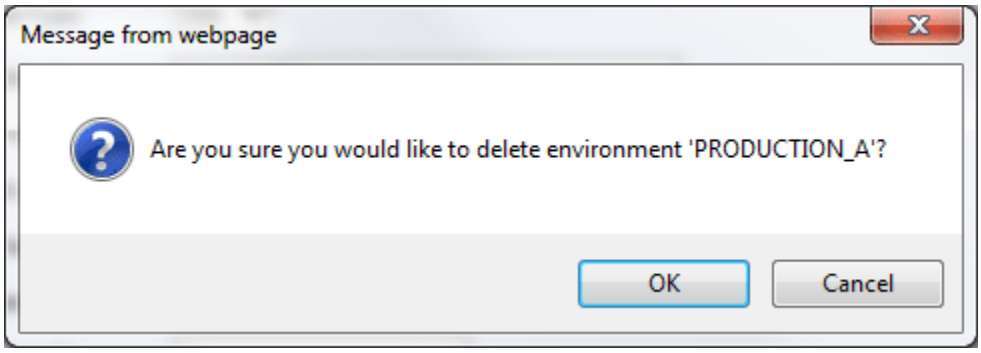


Figure 25 Deleting an Environment

4. Click **OK** to delete the environment or **Cancel** if you do not wish to delete the environment.
5. The environment will be deleted.
6. The deleted environment will still be available for rating against until the configuration XML file is reloaded. This is done by resetting the environments.

WARNING: *Make sure this is the action you want to take.* ***This action cannot be undone.***

NODES

A node is the endpoint of an IBSS instance. Every instance of IBSS must have a node created. Nodes allow you to manage the cache and log files and to reset the environment of the node. Nodes have the additional advantage of allowing for clustered environments to be managed from one location.

Nodes are created and managed on the IBSS home page.

NODE MANAGEMENT

In Figure 27 Cluster/Node Structure, the IBSS instance on the left is the instance that has been accessed using the IBSS URL (http://<*YOUR SERVER*>:<*port*>/IBSS. This could be any one of the member nodes. All nodes have the same view. This is because of the shared config.xml file.

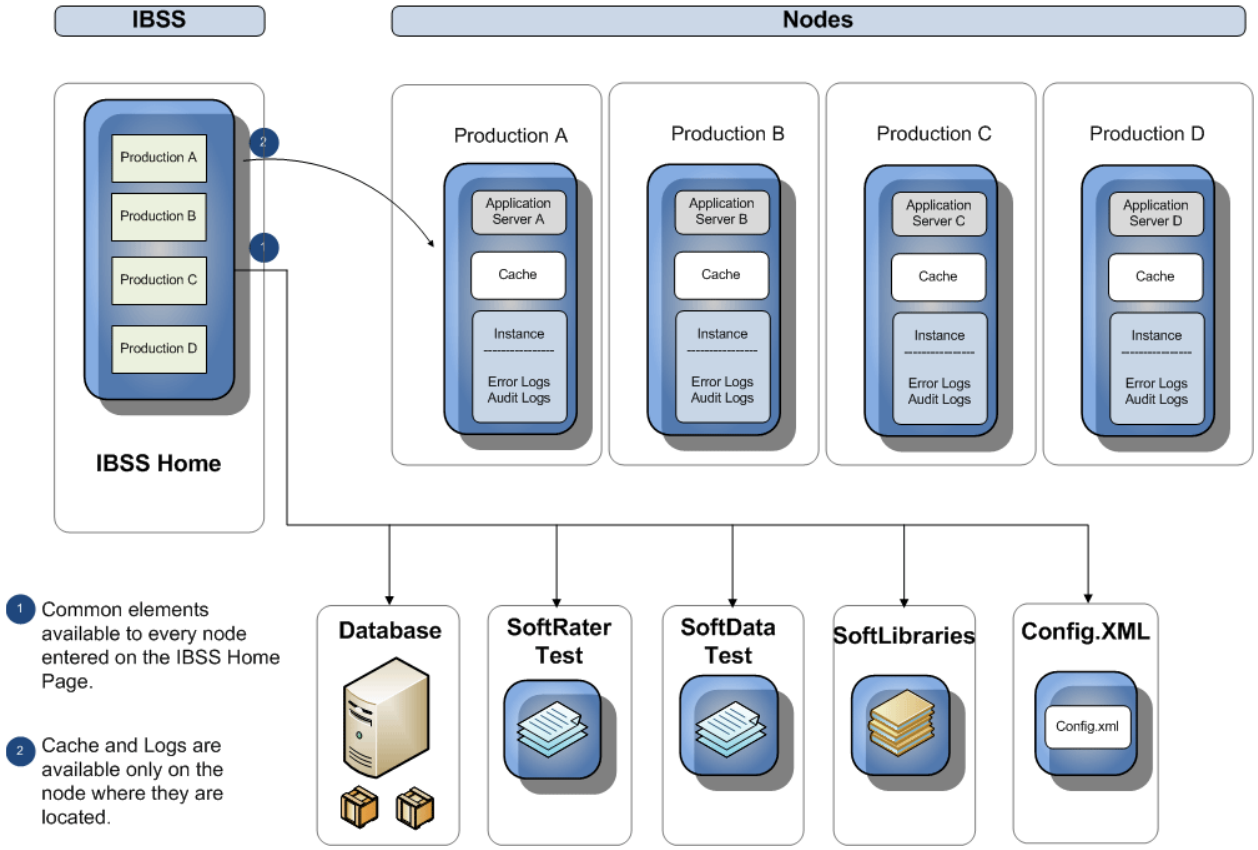


Figure 26 Cluster/Node Structure

The IBSS instance that has been accessed is also presented as a node. There is no distinction between the node you have accessed and other member nodes. You will see the same information presented on the IBSS home page and have access to the same databases, SoftRater Test and SoftData Test options, and SoftLibraries.

There are features that can be accessed on a selected node only:

- Reset a Single Node Environment
- Cache
- Logs

These features can only be accessed under the node where you want to view or perform an action.

ADDING A NODE

Every IBSS instance requires a node, which should be entered at installation. Additional nodes can be entered then or at a later time. Nodes are listed under the Add Node section on the IBSS home page. Nodes are also listed on the left side of the window in alphabetical order. To view nodes, expand the Nodes folder.

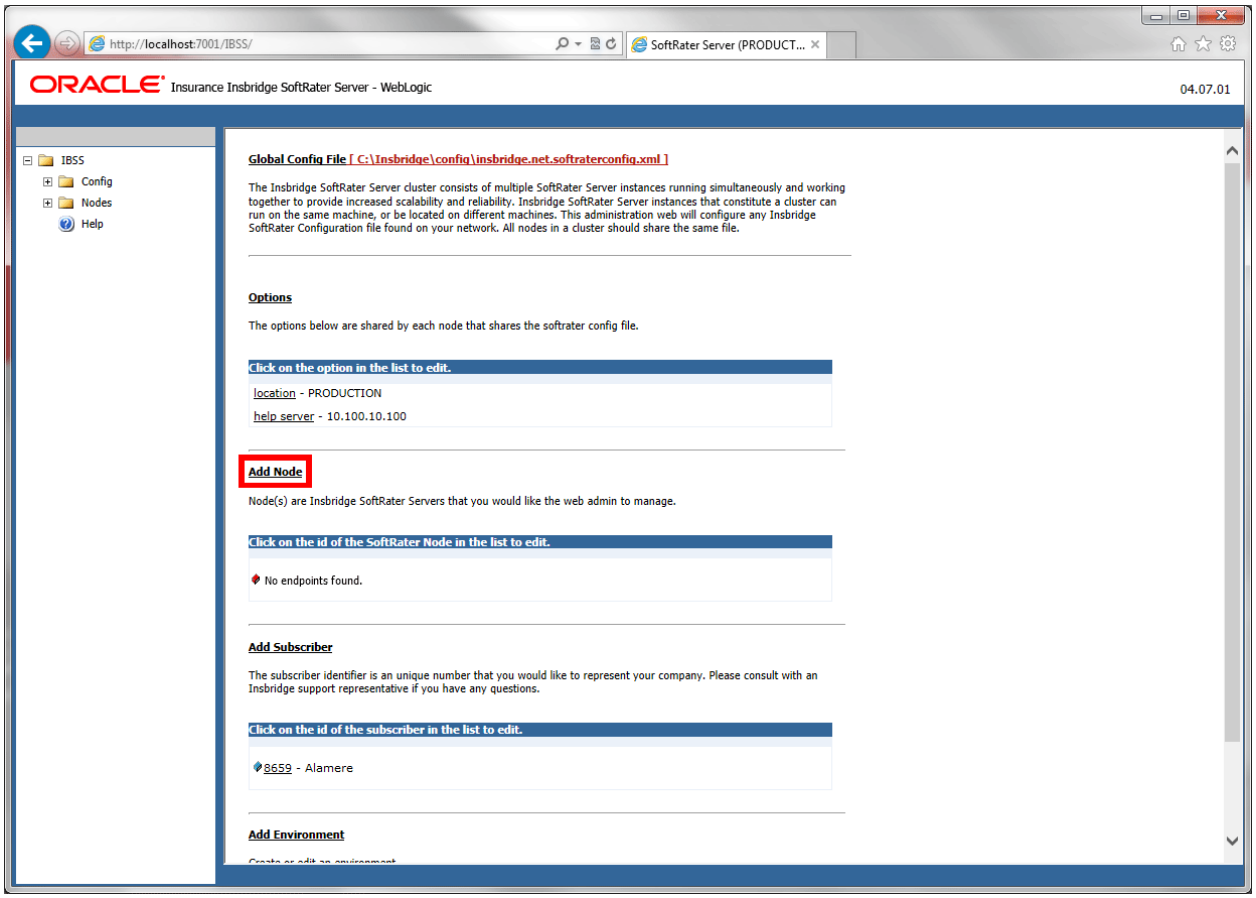


Figure 27 Adding a Node

To create a node, server information is needed:

- **Server OS**
- **Server Name or IP Address**
- **Port Used**

1. From the IBSS home page, click the **Add Node** link.

2. This opens the **Add Subscriber** window.

Figure 28 Adding a Node

3. Select the server operating system from the drop down. There are three choices, Linux, Mainframe and Windows.

4. Enter the server host name or IP address and port. The default ports:
   - **WebLogic:** Port 7001
   - **WebSphere:** Port 9080
   - **JBoss:** Port 8080

The format for this is: [SERVER NAME]:[PORT NUMBER]

5. Enter a friendly name. This is the name displayed for the node.

6. Check if this is a secure environment. A secure environment connects using HTTPS.

7. It is recommended that the Ping Server option be used to test the connection. This is an optional step but can be helpful. If a connection cannot be made, the node cannot be saved.

8. Click **Add** to add the node. If a node cannot be added, an error message is displayed. Please correct the error and try again.

NOTE: *It is recommended that server names or IP addresses be used over localhost.*

EDITING A NODE

Nodes can be edited at any time. The server name or IP address cannot be edited. If you have entered an incorrect server name or IP address, you must delete the incorrect entry and re-enter with the correct information.

1. To edit a node, click the Node link on the IBSS home page that you want to edit.

2. The node information window is presented. The server name or IP address cannot be edited. All other fields are open for editing. The field highlighted in blue is the field you are currently in.

Figure 29 Editing a Node

3. When you are finished editing information, click **Save**.

4. Before you can use the new or edited environment for rating, you need to reload the configuration XML file. This is done by resetting the environments. On the left hand side of the IBSS home page, expand Nodes and select the Reset Environments option. You will be placed on the Reset Environment Configuration page. Click **Execute**. The screen refreshes and each node will be reset. Click IBSS to return to the home page.

Testing a Node

You can test for connectivity using the Ping Server option of the Node instance window. The Ping Server will let you know immediately if a connection can be made to the node.

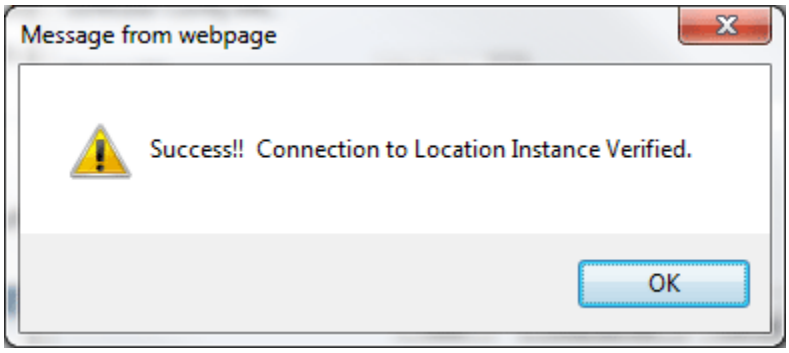


Figure 30 Successful Node Connection

If connectivity cannot be established, a fail message is returned.

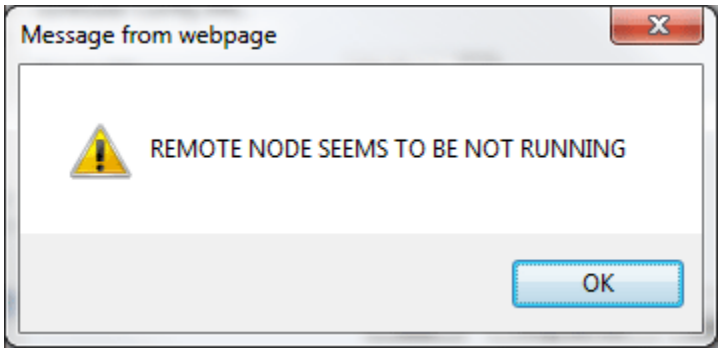


Figure 31 Down Node Message

DELETING A NODE

A node can be deleted at any time.

1. To delete a node, click the node link on the IBSS home page that you want to delete.
2. The environment information window is presented, click **Delete**.
3. You will be asked to confirm deleting the node.

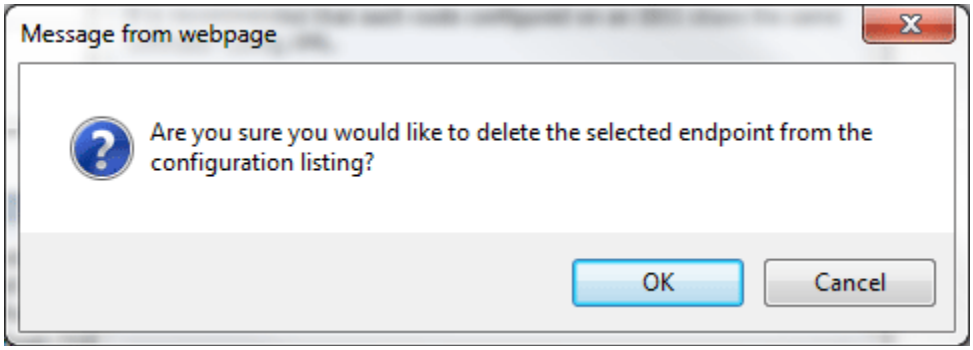


Figure 32 Deleting a Node

4. Click **OK** to delete the node or **Cancel** if you do not wish to delete the node.
5. The node will be deleted.
6. The deleted node may still be listed until the configuration XML file is reloaded. This is done by resetting the environments.

NODE INFORMATION PAGE

Nodes are listed in the Nodes section on the IBSS home page and on the left side of the screen under the nodes folder.

The nodes folder holds the shared functionality availed to all nodes and the details and functionality available to each individual node. To view the details of an individual node, expand the Nodes folder and select the node where you want to view or edit details.

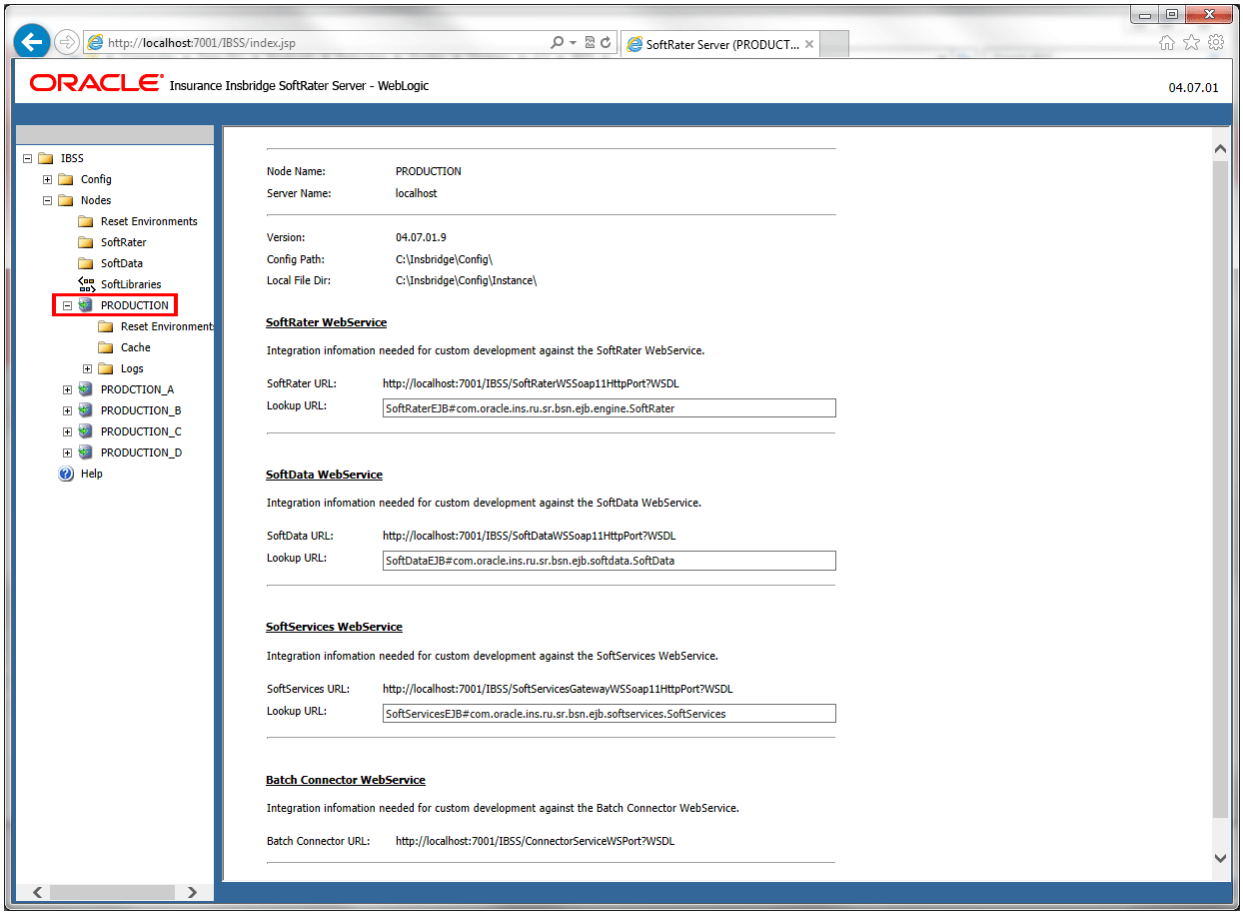


Figure 33 Node Management Window

To view the properties of the node, click the node name. The Node Information page is displayed. A few fields can be edited on this page. Most fields must be edited in other areas or cannot be edited.

- **Node Name** – The friendly name of the node. This is the name entered when the node was created. This can be edited on the IBSS home page by editing the node.
- **Server Name** – The server name or IP address where the node is located. This is a non-editable field.
- **Version** – The current IBSS version. Support may need this information if you have to contact them. This is a non-editable field.
- **Config Path** – The path to the shared config.xml file. This is a non-editable field.

- **Local File Dir** – The location of the Instance folder. The instance folder holds the error and audit logs for the node. This directory should be unique per node. This is a non-editable field.
- **SoftRater WebService** – The integration information needed for custom development against the SoftRater WebService. The Lookup URL can be modified.
- **SoftData WebService** – The integration information needed for custom development against the SoftData WebService. The Lookup URL can be modified.
- **SoftServices WebService** – The integration information needed for custom development against the SoftServices WebService. The Lookup URL can be modified.
- **Batch Connector WebService** – The integration information needed for custom development against the Batch Connector WebService. This is a non-editable field.

Editing Lookup URLs

To update Lookup URLS, enter in the new URL and click Save at the bottom of the window. Your entry is saved immediately. You must save your entry. If you leave the page without saving, the URL will not be updated. No warning message is displayed if you make changes and leave the page.

Node Status

Every node is represented by name on the left menu. The icon next to the node indicates whether the node is up or down. Node status is based on availability.

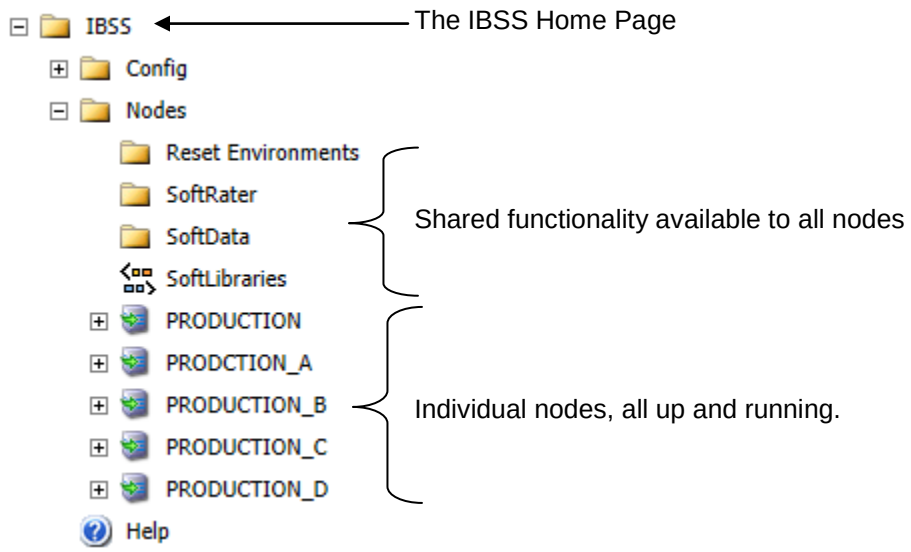


Figure 34 IBSS Instance will All Nodes Accessible

Nodes have two statuses:

	Indicates the node is down and not accessible by IBSS.
	Indicates the node is up and accessible.

If the node is down, it means that IBSS cannot access it. There could be a few reasons IBSS cannot access the node. The first place to check is the application server to make sure that the server is up and running properly and the port is accessible.

When a node is down, you will not be able to access the individual node features or reset the environment.

If a node is accessible, then it presents as up.

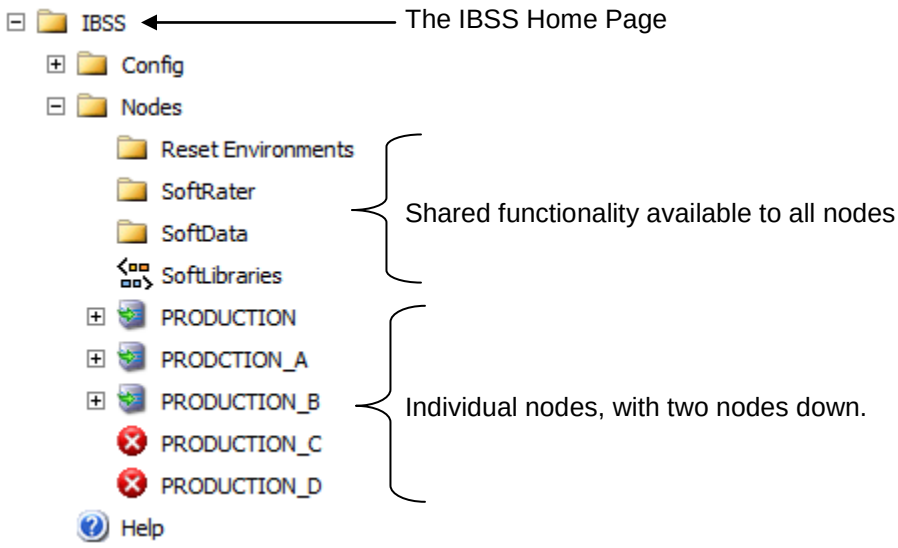


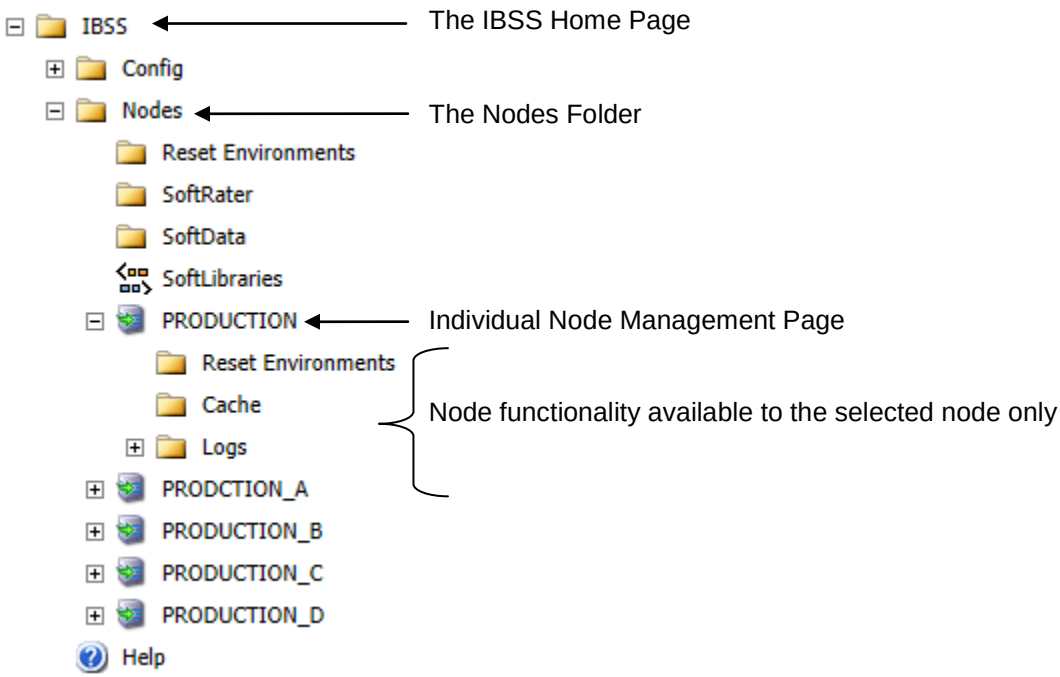
Figure 35 IBSS Instance with Two Nodes Down

To see if a node has come back on line, refresh the window. The icon should change from down to up if the node is accessible.

NOTE: *An inaccessible or missing config.xml file does not present as a down node. An error message will be displayed on the IBSS home page.*

NODE FUNCTIONALITY

Under the Nodes folder, there is shared functionality that is available to all nodes. Under each individual node is functionality that is available only to the selected node.



The functionality available to individual nodes only is:

- **Reset Environment** – Resets the environment for the selected node only. Other nodes will continue using the config.xml that is in cache.
- **Cache** – Displays the current program listing in cache. On this page you can refresh cache to get the latest program listing or clear cache for the selected node.
- **Logs** – Displays the error and audit logs for the selected node. On this page you can view all general, batch, or impact analysis error logs or XML, SQL, or SoftRater Runtime Export audit logs.

These actions can only be performed under the node where you want to action to take place.

Reset Environments

To reset an environment for a specific node, select the **Reset Environments** link under the node that you want to reset. The Reset Environment Configuration:<*NODE*> window is displayed. The node that is going to be reset is displayed.

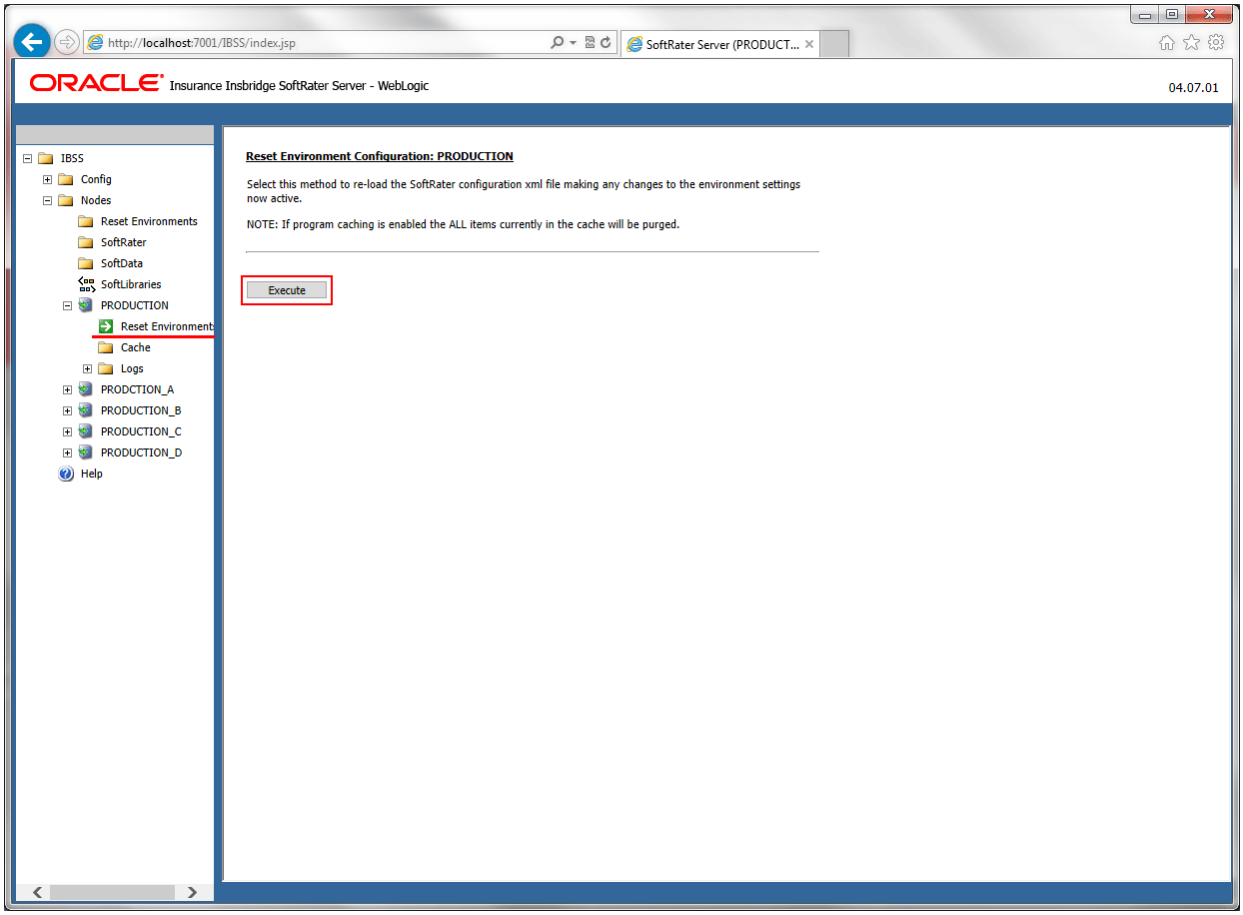


Figure 36 Resetting a Node

1. Click the Execute button. A confirmation window is displayed.

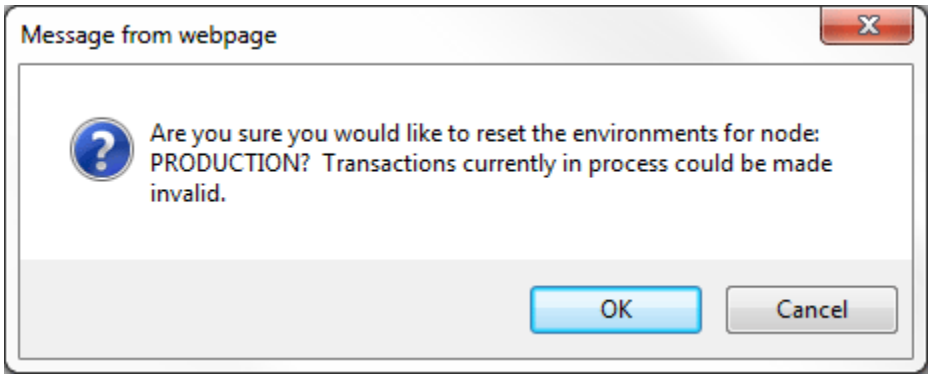


Figure 37 Confirming a Node Reset

2. Click OK to reset the environment. Click Cancel to return to the previous screen. No action will be taken.

3. The screen refreshes and the environment has been reset.

4. A confirmation window is displayed with a success or fail message.

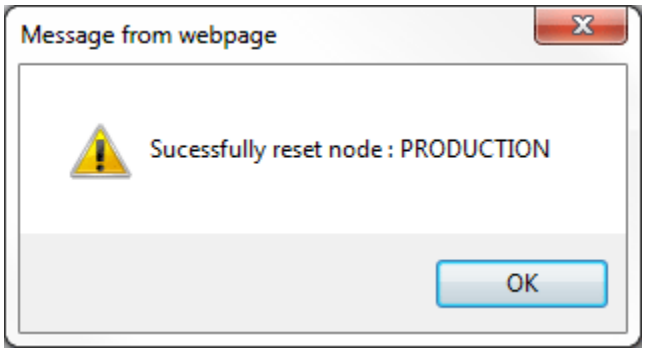


Figure 38 Successful Rest

An individual environment may need to be reset to activate changes to an environment. Changes do not take effect until the SoftRater configuration XML file is reloaded. This is the purpose of an environment reset.

NOTE: *Resetting environments causes all information in the cache to be purged and any transactions currently processing could be made invalid.*

CACHE

Cache is available to the individual node. Select the Cache link under the node where you want to view or clear cache. The Program Cache Options window is displayed. This page allows you to view a listing of all the programs currently in the cache and remove a specific program from the cache or clear the entire cache. If there is no cache, check to see if cache has been enabled on the Config->Engine page.

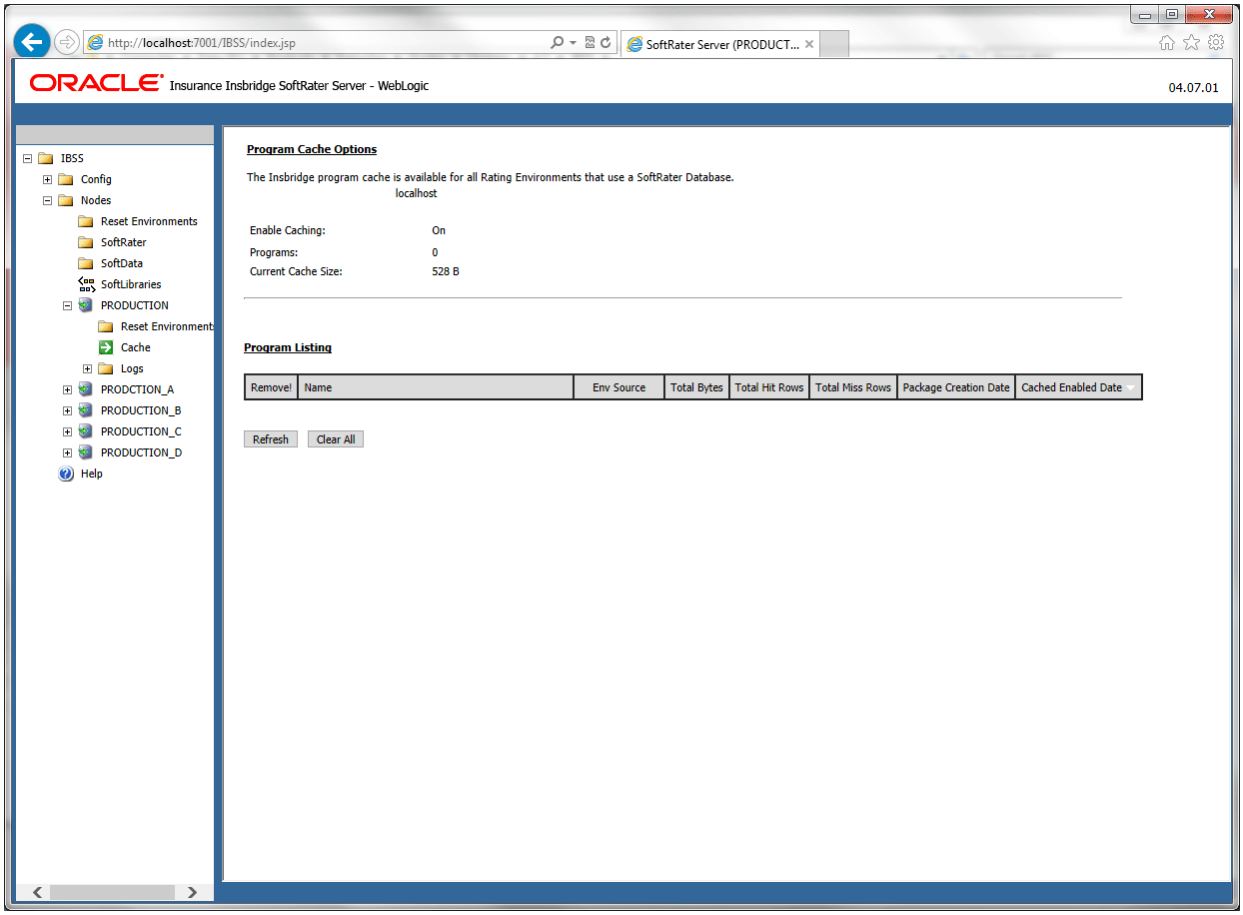


Figure 39 Program Cache

Removing a Program from the Cache

To remove a specific program from the cache, use the **Remove** hyperlink next to the desired program. All information for that program (in the specific environment) will be removed from the cache. If information exists in the cache for the same program in different environments, removing the information from one environment does not remove it from the other environments.

If no programs are available for removal, the Remove hyperlink will not be displayed.

Clearing the Cache

To remove all information from the cache, click **Clear All**. All information will be removed from the cache.

LOGS

There are two Logs pages. One is for errors and one is for audits. Both pages show a listing of the logs available. By default, information is only logged when an error occurs. Audit logs must be enabled. This can be done on the Config->Engine page.

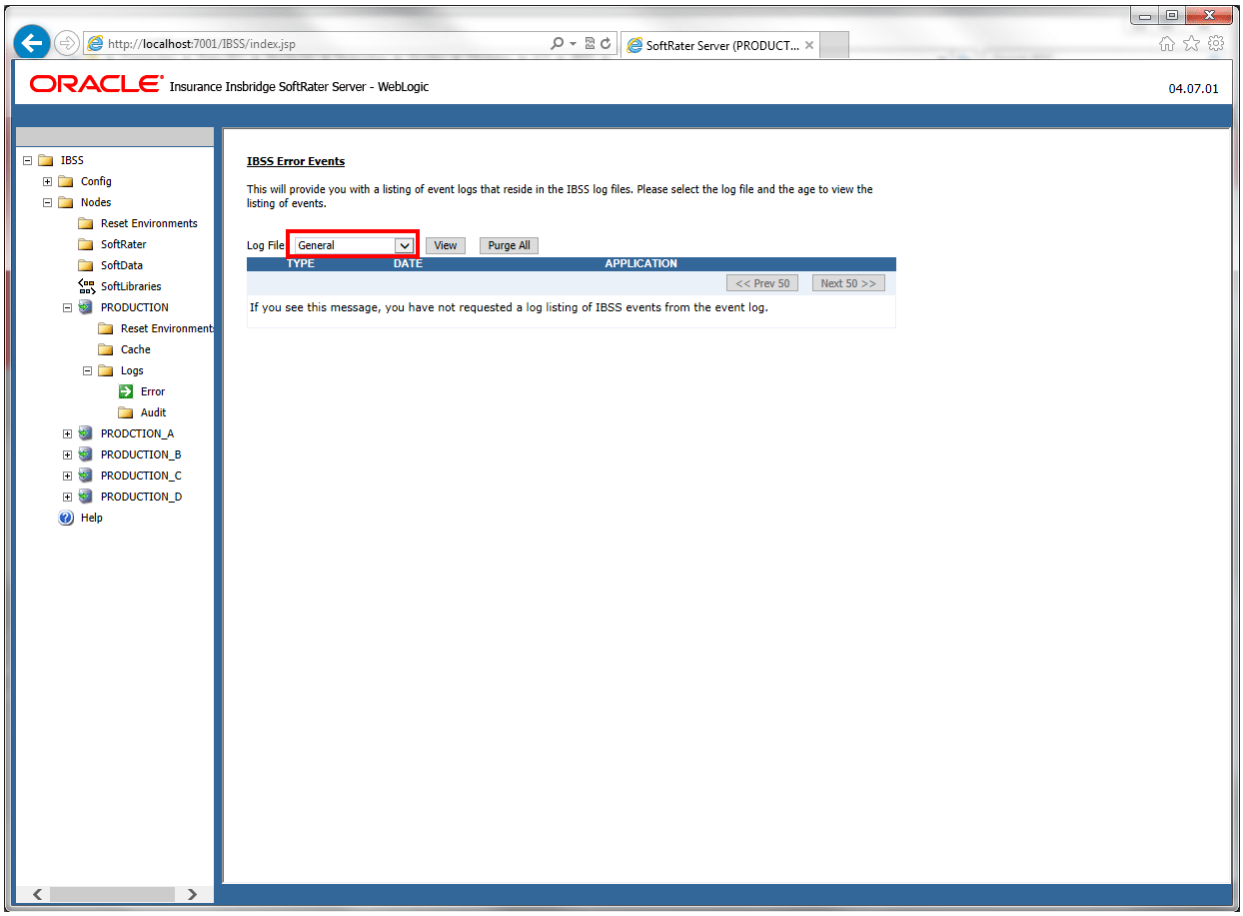


Figure 40 Logs Page

Error Logs

General: General error message logs.

Batch: Error logs for batch files.

Impact Analysis: Error logs for Impact Analysis.

Audit Logs

XML: Operational XML that triggered some exception processing.

SQL: SQL text that has generated some exception processing.

SoftRater Runtime Export: Provides an XML view of the entire program that was executed.

To view an error log, click the hyperlink. The log will be returned in the text area at the bottom of the screen. If auditing has not been turned on, there will not be any audit logs displayed. To turn audit logging on, please go to the Config->Engine page.

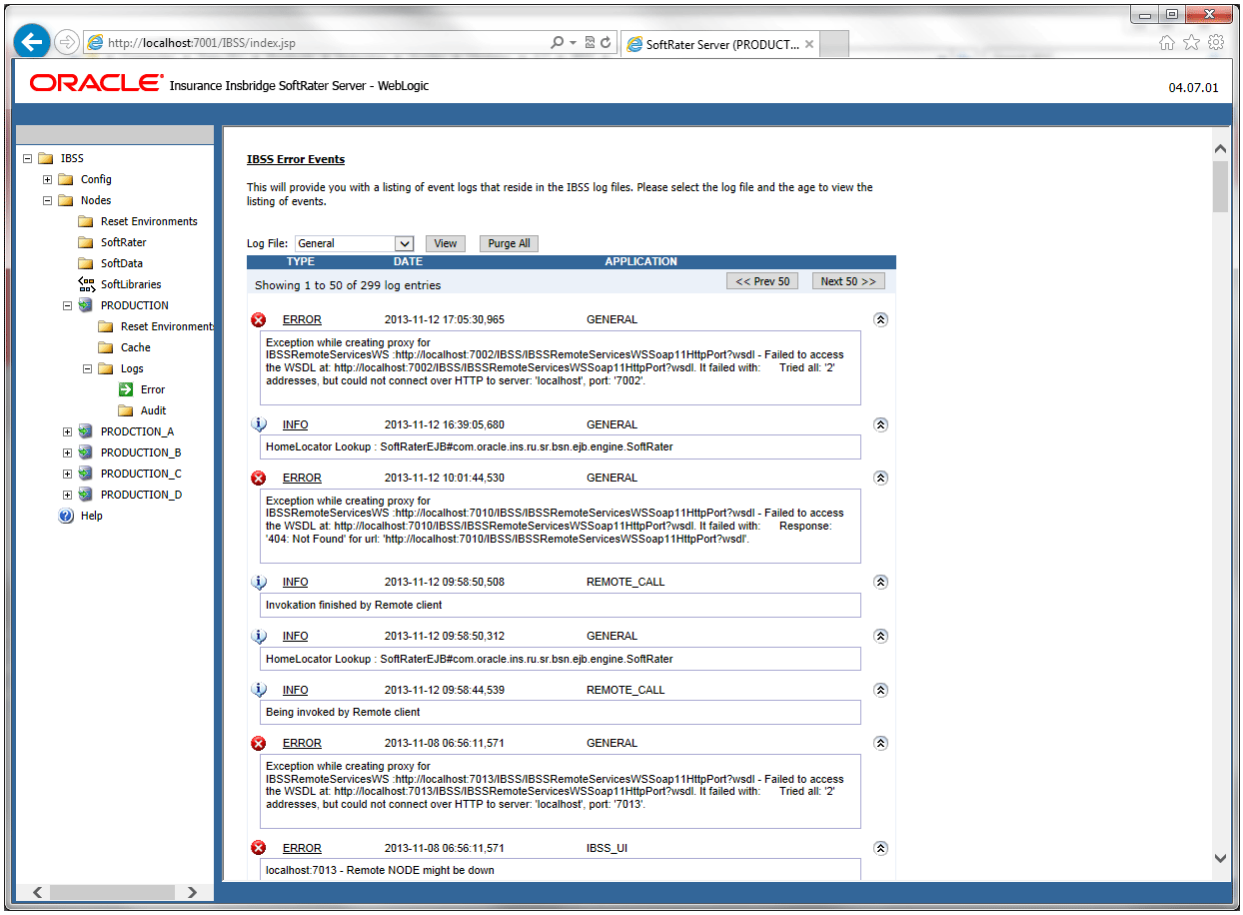


Figure 41 Error Logs

Viewing Logs on the Server

If you have access to the server where IBSS was deployed, you can view log files in the Instance folder of the node. Each node has an instance folder where logs are stored. The location of the Instance folder was determined at installation time.

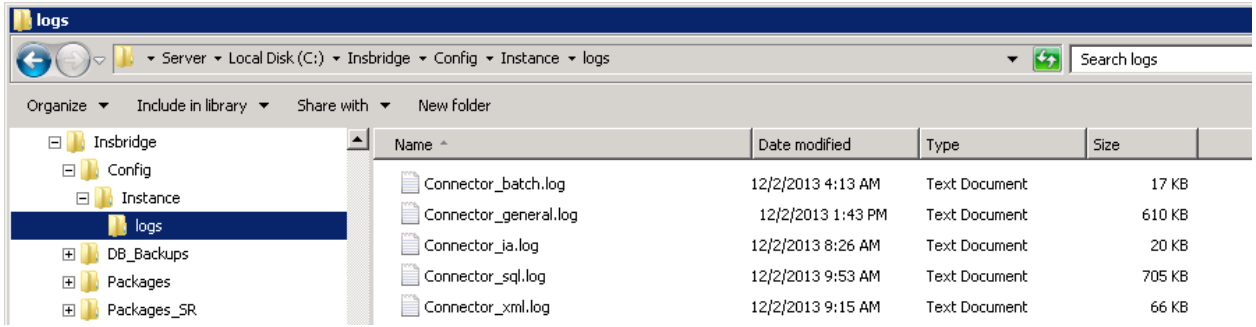


Figure 42 Logs stored on the Server

These are the logging options available and a description of what is being logged in the Instance file.

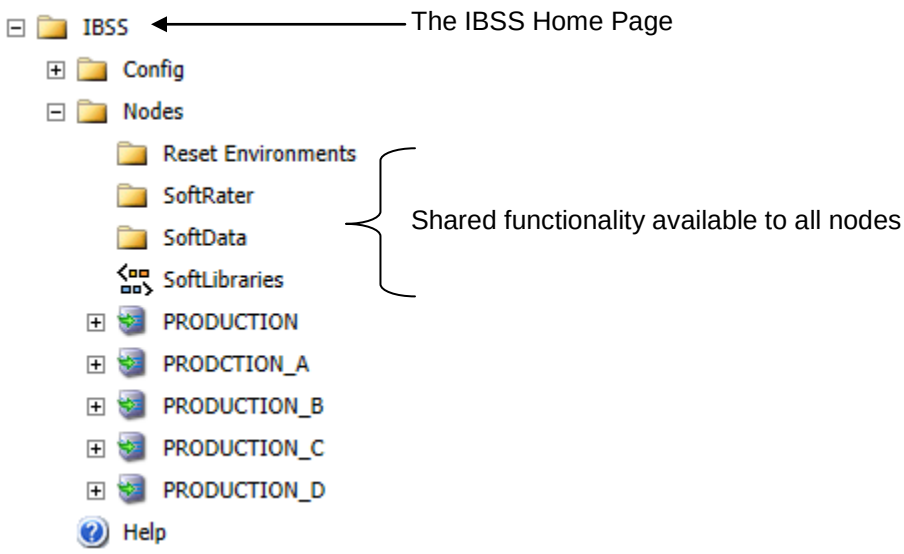
- **Connector_batch.log:** This is the error plus information log for batches.
- **Connector_general.log:** This is the error plus information log. All errors, warnings, requests, results, and such are logged here. Any SoftRater exceptions are logged here also.
- **Connector_ai.log:** This is the error plus information log for Impact Analysis.
- **Connector_sql.log:** This log records all database queries sent by SoftRater, including table variable lookup, and global versioning selection.
- **Connector_xml.log:** This log records all the rating request and result XMLs. The SoftRater.xml log only logs the rating request XML submitted to a SoftRater instance for both synchronous and batch rating requests. At the point a rating request makes it to SoftRater, the request does not distinguish between the two, since the IBFA Spindle web service is what handles the threading of the requests and the aggregation of the results

If XML Audit Logging option is enabled, it contains both <rate> and <result> nodes, so it logs both the request XML and the result XML.

NOTE: *Library logging is done on the SoftLibraries page on a per program basis.*

CONFIG

The Config section of IBSS has two pages:



The Config section of IBSS has two pages:

- **Engine**
- **Logs**

ENGINE

The Engine page is where you can select the information to be automatically logged and the statistical information to be returned in result XML files. The setting here apply to all nodes.

To navigate to the Engine page, select IBSS->Config->Engine.

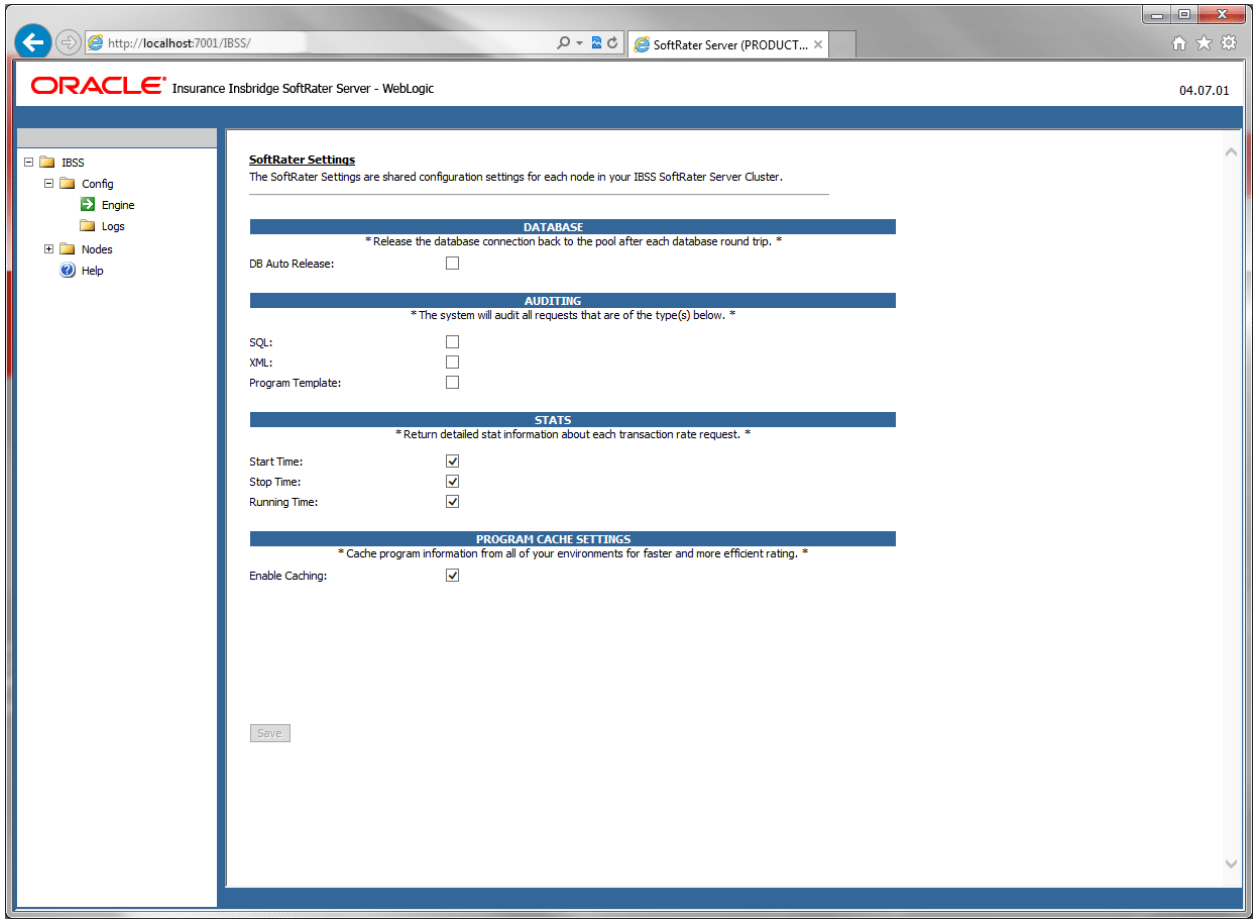


Figure 43 Engine Page

DB Auto Release

A check in this box will enable this feature. If this feature is enabled, the database connections are released as soon as the transaction or request is complete. In the case of a Program to Program (P2P) call, the connection is released as soon as the calling program is finished. If enabled, every time you call the database, a connection will need to be re-opened. This may slow down performance because connections will be opened and closed after each call. You may require this option if you have a limited number of database connections, licensing restrictions, or you are unable to keep connections open.

No check in this box disables this feature. If this feature is disabled, the database connection remains open. You may want keep this feature disabled if you have heavy usage, a large pool of connections, or you are able to keep connections open. Disabling the feature may result in faster performance. However, connections will scale up and down as demand is placed on the DB. If

you have a limited number of connections and heavy demand, disabling DB auto release may not give you the performance increase you need. Pool settings can be set by the DB administrator to allow for a larger connection pool. If disabled at the SoftRater level, on most DB servers, there is a setting that can be used.

DB Auto Release is disabled by default.

Auditing

Allows you to select what information should be logged for each transaction. These options should typically be left unchecked, unless directed to check them by a member of the Oracle Insurance Support team. Leaving auditing on may result in exceptionally large log files.

SQL: The SQL script that was executed. If this option is not selected, this information will still be logged if an error occurs.

XML: The XML that was used. If this option is not selected, this information will still be logged if an error occurs.

Program Template: Provides an XML view of the entire program that was executed. This information is used to help Oracle Insurance determine why a program is not rating correctly.

NOTE: *SoftLibrary logging is done on the SoftLibraries page.*

Stats

Allows you to select if and what statistical information will be included in the return XML. Any combination of options may be selected. The statistical information is returned in the <stats> node.

Start: Time stamp of when rating started. Returned in the <start_time> node.

Stop: Time stamp of when rating finished. Returned in the <stop_time> node.

Running: The **Start Time** minus the **Stop Time**. The result is the length of time rating took, in milliseconds (1000 milliseconds = 1 second). Returned in the <running_time> node.

NOTE: ***Start Time*** *and* ***Stop Time*** *are shown in microseconds (10^{-3} seconds), while the* ***Running Time*** *is shown in milliseconds (10^{-2} seconds).*

Cache Setting

The first time a rate request is sent down, for example, from a policy admin system; the application server that holds the selected rating environment retrieves the requested rate package from the database and stores it in cache. When another rate request is sent down that

utilizes the same package, if it is already stored in cache, the cached package is used. This increases efficiency and allows for rate requests to be returned faster. Packages are cleared out of cache when an updated package is loaded from IBFA or when environments are reset.

The Cache Settings allows you to enable caching. A check indicates the cache is enabled. If cache is not enabled, then the rate package is retrieved from the database every time. This may increase rating times.

LOGS

The Logs page is where you can set the log size for error and audit logs. Every node has individual error and audit logs. The size set on the Logs page applies to each node, not to the collective amount between all nodes. For example, if you set a size of 4, then every node will have a log size of 4. The sizes are in MB. If a log file reaches the max size, then log files roll off with the oldest being purged first.

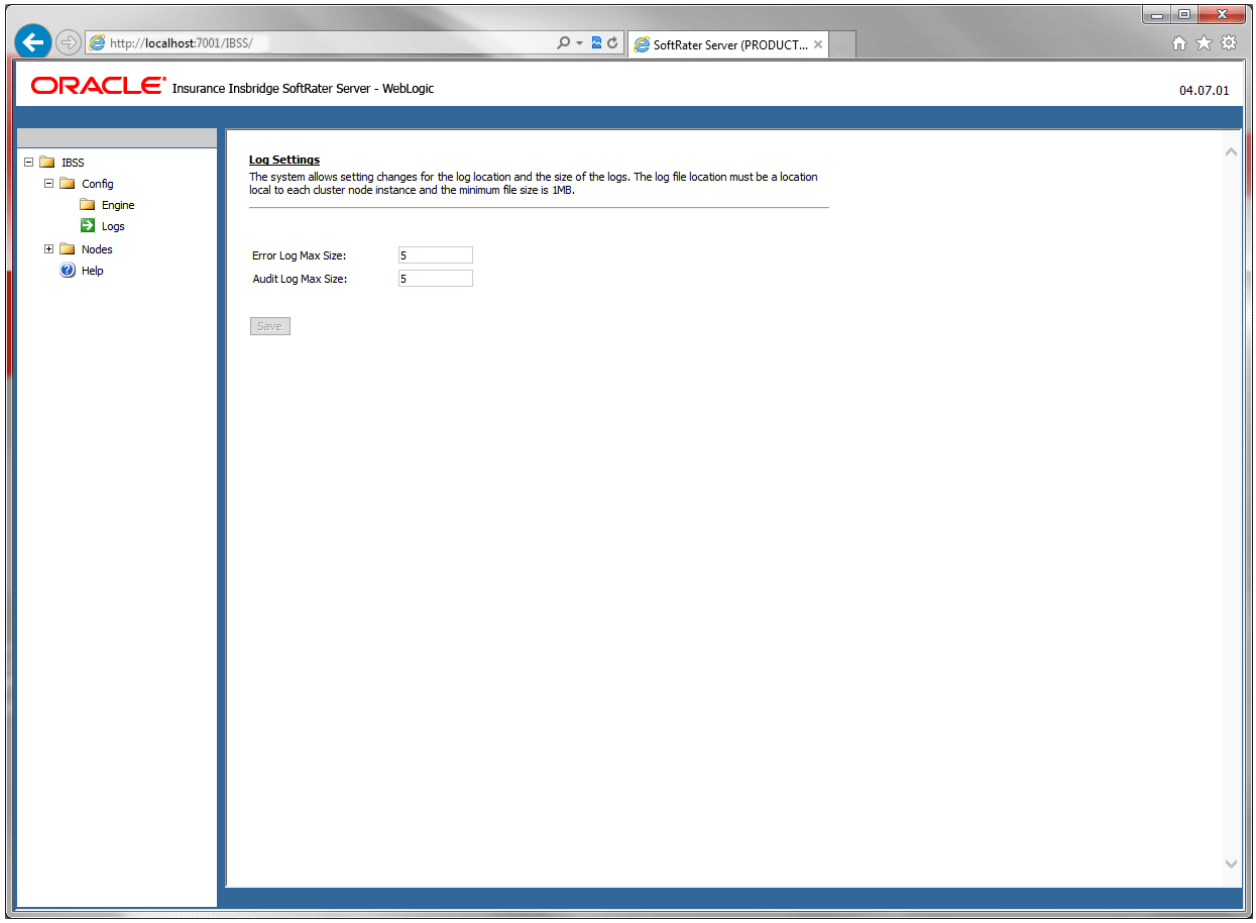


Figure 44 Logs Page

To set the size, enter in a number and click Save.

SOFTRATER TEST INTERFACE

Use the SoftRater Test Interface when you want to:

- Compare rates from different servers
- Verify a package is loaded
- Verify SoftRater is using the correct version of a program
- See how long SoftRater is taking to rate a program

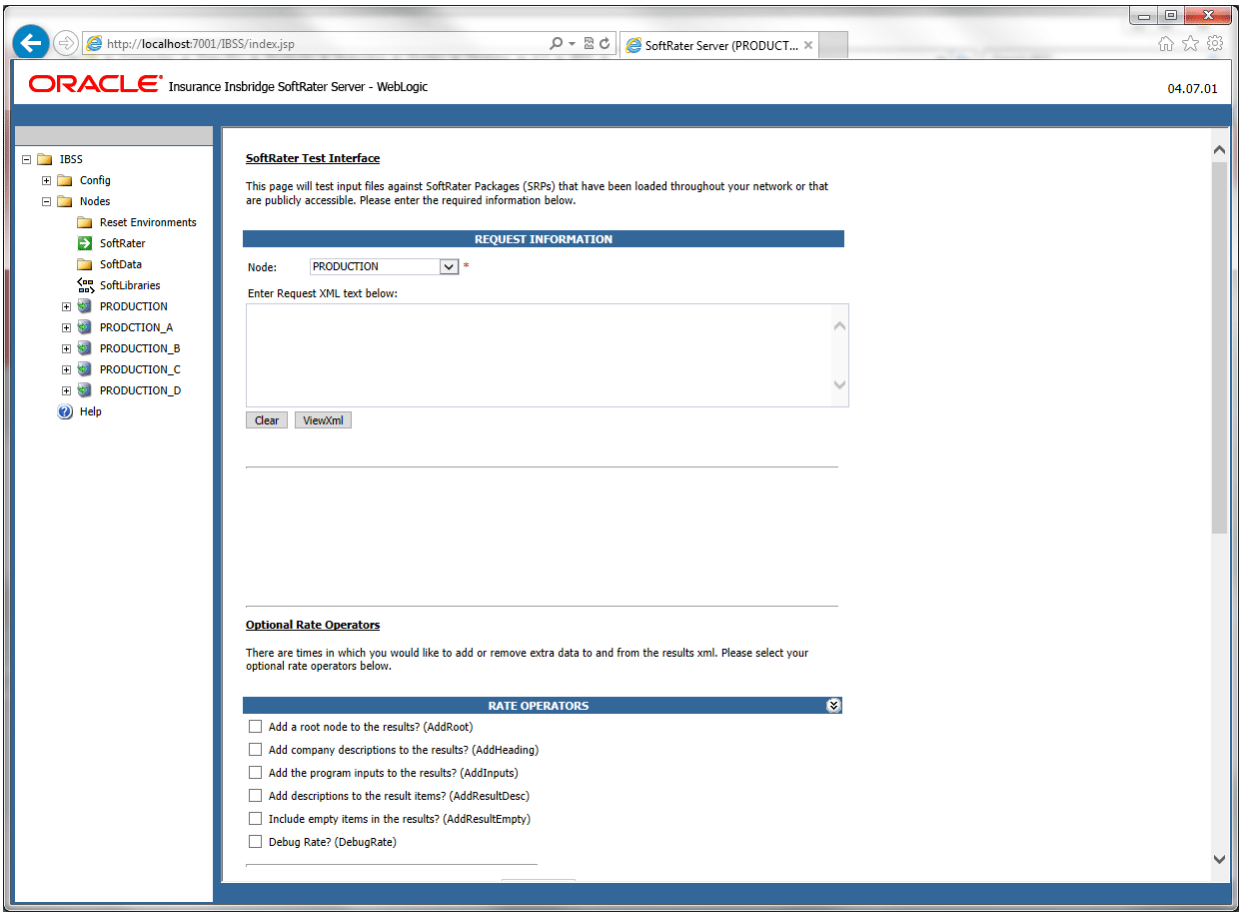
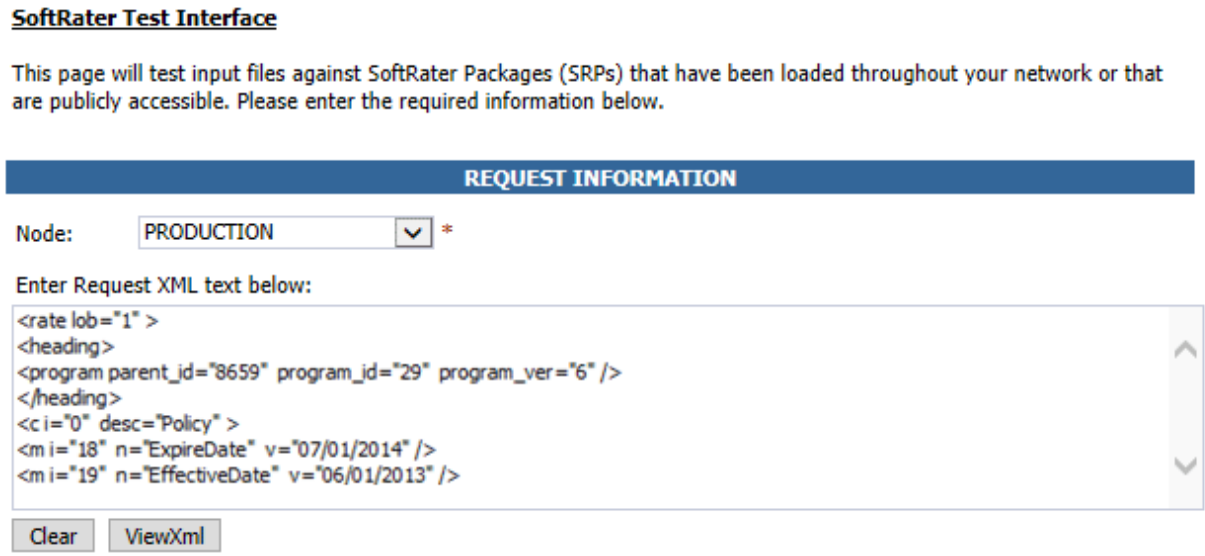


Figure 45 SoftRater Test Interface

The SoftRater test interface can be set to run synchronously.

1. On the SoftRater Test Interface page, select the node that you want to rate against from the menu.

2. Paste the XML file into the large text box.



3. Scroll down the page and choose any optional operators SoftRater should use when rating the file.

4. When you are ready to rate the file, scroll to the bottom of the page and click **Execute**.

5. The file is rated and the results are shown in the results box.

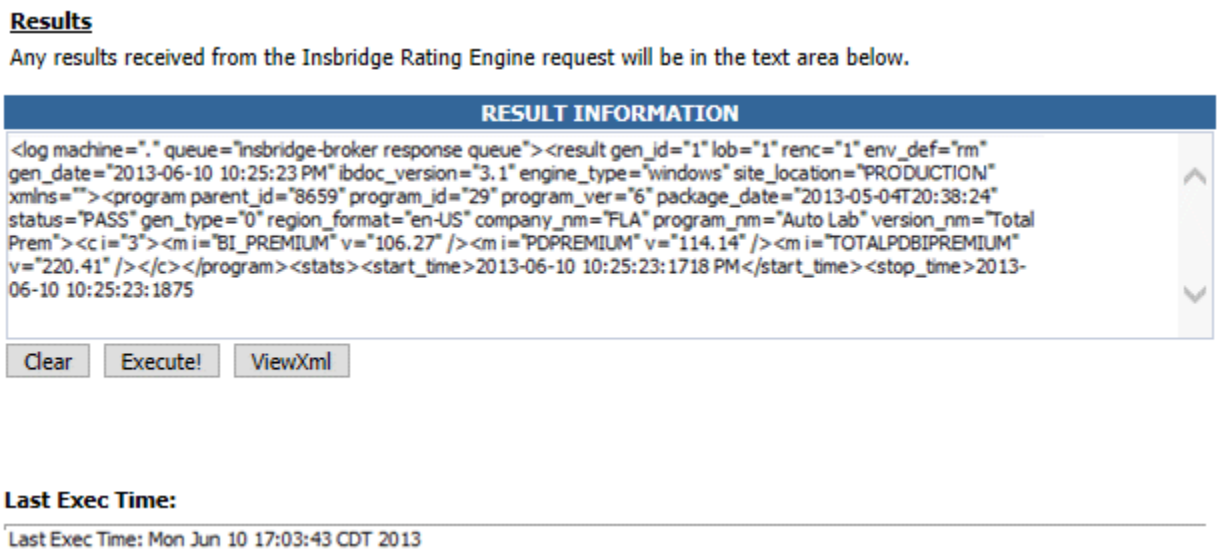


Figure 46 Test Results

6. To view the file in XML, click **ViewXml**.

Rating a File Using the SoftRater Test Interface

When rating a file through the SoftRater Test Interface, you have the ability to instruct SoftRater on how to rate the file and how to return the results. These options are divided into two categories:

- Rate Operators
- Map Request Operators

Rate Operators

Rate operators are optional and function to add or remove information from the result XML. They also enable you to override the default result and environment definitions. To view or edit optional rate operators, click the expand icon on the far right hand side of the RATE OPERATORS header.

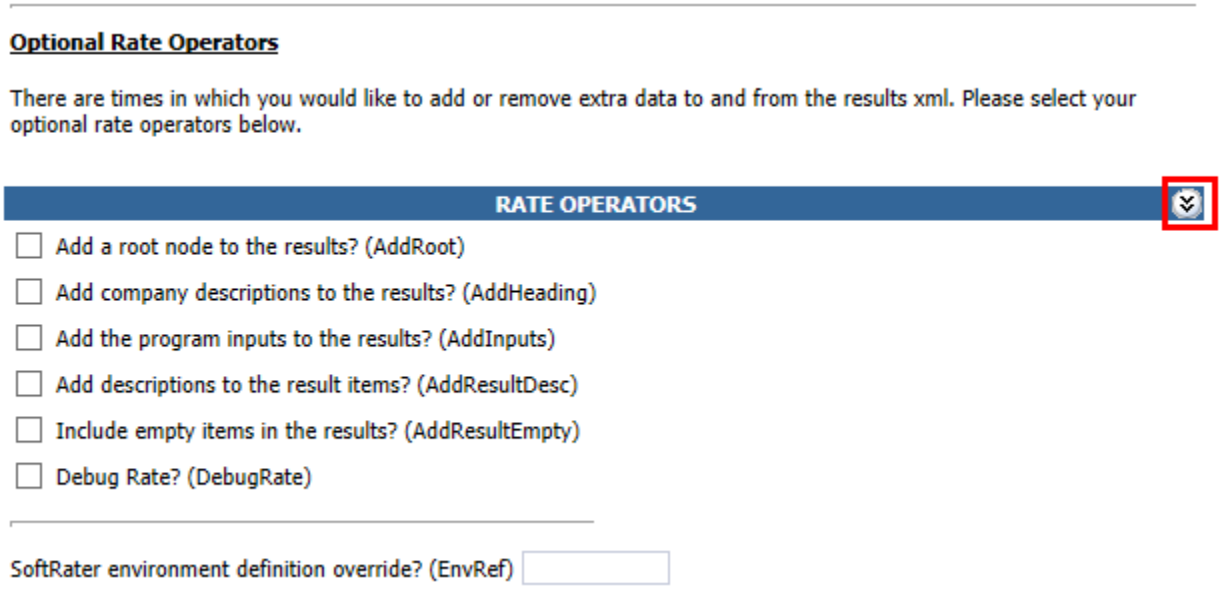


Figure 47 Optional Rate Operators

NOTE: *You can use any combination of rate operators.*

The following options require you to check a box to enable the item:

Add a root node to the results? (AddRoot): Add the root node when you want to determine how long SoftRater is taking to rate a policy. The root node contains the time rating started and the time rating stopped. It also shows you the number of input files and the number of output files. The root node is always the first line in the results and begins with `<ibdoc>.`

Add company descriptions to the results? (AddHeading): Add the folder pathway when you want to see the name of the path of the folder, program and program version.

Add the program inputs to the results? (AddInputs): Add the program inputs when you want to verify that SoftRater is using the correct input values.

Add descriptions to the result items? (AddResultDesc): Add descriptions to the results when you want to see the name of the variable or input along with the result ID. The variable or input name will be shown after the `d`, while the result ID is shown after the `i.` In the following example output, `VehPrincipallyDriven` is the input name, and `Primary Vehicle` is the result ID:

```
<m i="Primary Vehicle" d="VehPrincipallyDriven" v="1" />
```

Include empty items in the results? (AddResultEmpty): By default, SoftRater does not output results that do not have a value. If you would like to see all results, check this box.

Debug Rate? (DebugRate): Check this box if you would like to see the debugged XML along with the results.

The following option requires you to enter specific information into the text box:

SoftRater environment definition override? (EnvRef): If you do not want to rate against the default SoftRater environment, enter the environment that should be rated against in this box.

Map Request Operators

Map request operators are optional and enable you to submit custom XML to the SoftRater engine and receive custom XML back from it.

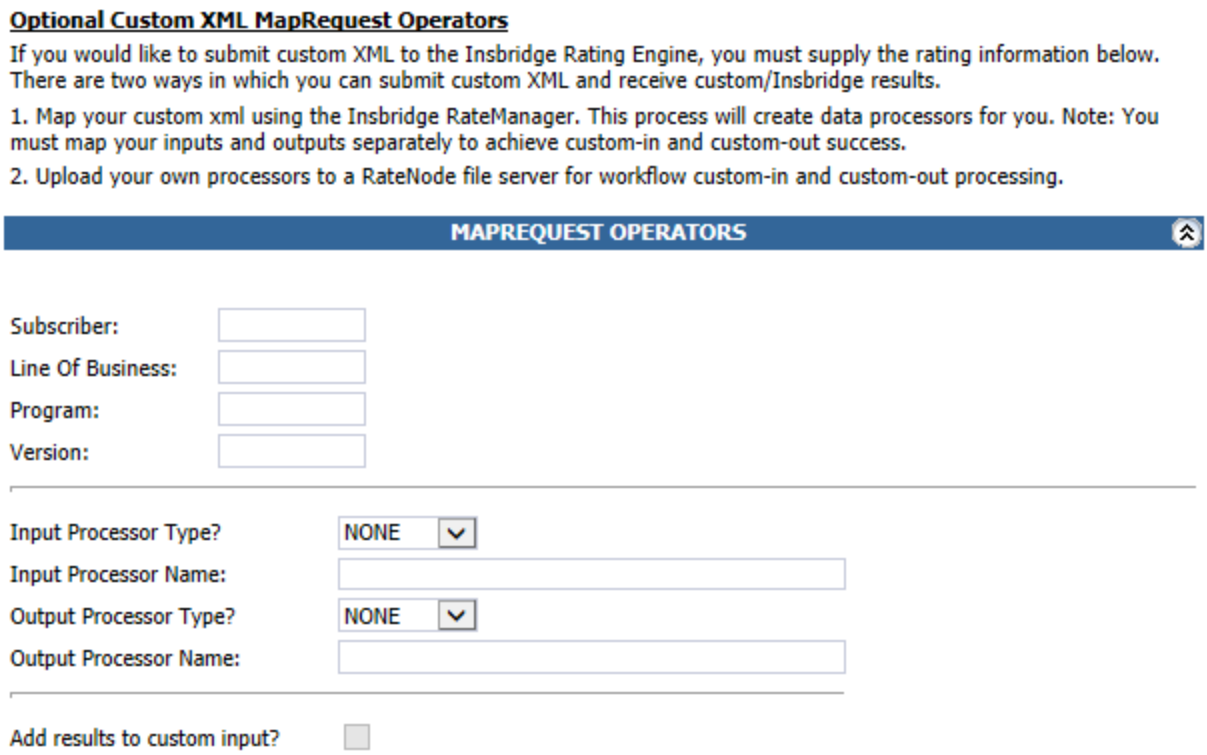


Figure 48 Custom XML Map Request Operators

The following information is required if you are using a mapping:

Subscriber: Your subscriber ID.

Line of Business: The number for the line of business you are rating. For example, if you are rating a file for the fire line of business, then enter 5.

Program: The program ID you are rating.

Version: The program version you are rating.

The following information is required only if using custom input XML:

Input Processor Type: The type of input mapping to use.

GLOBAL: A mapping that can be used by all programs under a specific line of business. Global mappings are created in RateManager from the **Global Inputs** window.

LOCAL: A mapping that is specific to a particular program version. Local mappings are created in RateManager from the **Program Listing** window.

CUSTOM: A mapping that was created outside of RateManager and then imported.

Input Processor Name: The unique identifier that was entered for the mapping.

The following information is required only if you want to receive custom output XML:

Output Processor Type: The type of output mapping to use.

LOCAL: A mapping that is specific to a particular program version. Local mappings are created in RateManager from the **Program Listing** window.

CUSTOM: A mapping that was created outside of RateManager and then imported.

Output Processor Name: The unique identifier that was entered for the mapping.

You can optionally check the following:

Add results to custom input: If you want the original input file to be appended at the beginning of the result file, select this box.

SOFTDATA TEST INTERFACE

The SoftData Test Interface allows you to test a SoftData request file that is in InsbridgeDataRequest.XML format. You can select the node you want to rate against. The file can be copied from a source and pasted in the text box.

SoftData is a common feature available to every node. SoftData is located under the Nodes folder.

1. Select **SoftData** from the menu on the left. This opens the **SoftData Test Interface** page.

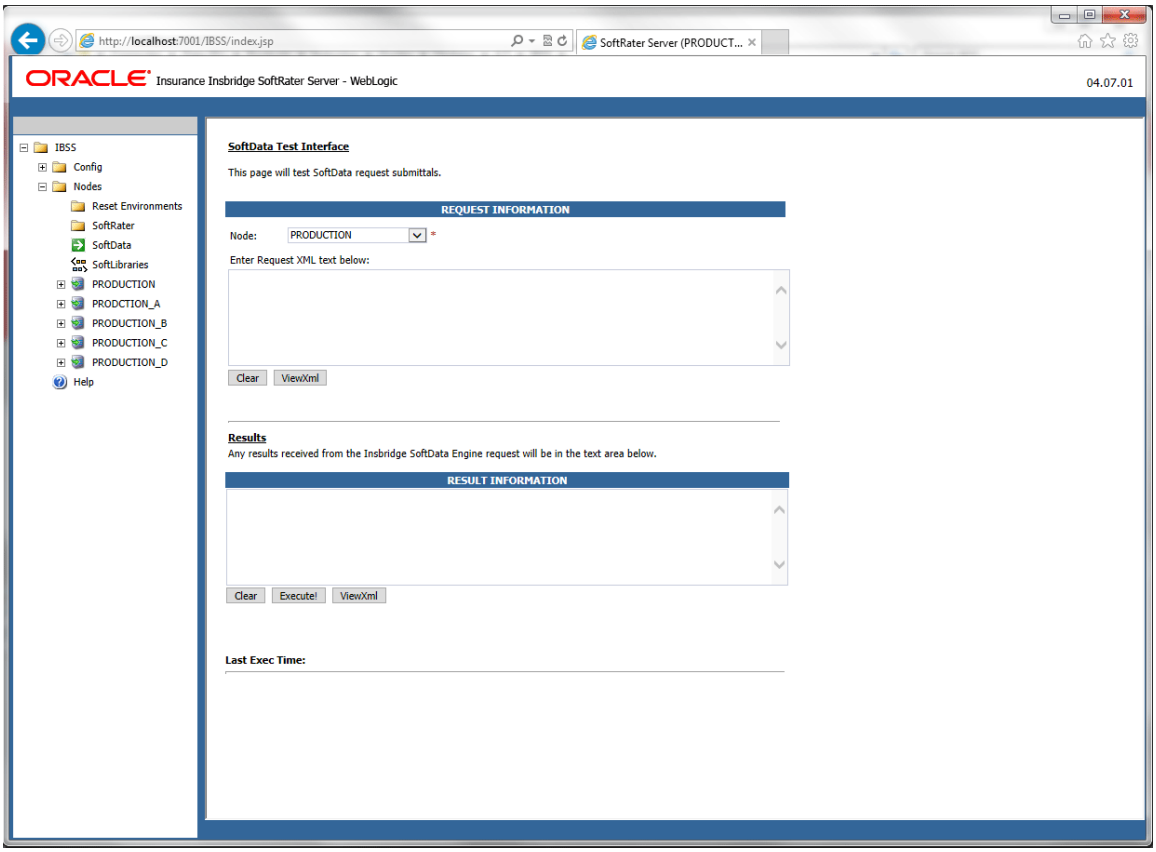


Figure 49 SoftData Test Interface

2. Select the node that you want to test against.
3. Click **Execute**. The file will be submitted and the results are shown in the results box.
4. To view the XML file in Internet Explorer, click **ViewXML**.

NOTE: *For SoftData format, please see the SoftData User Guide.*

Chapter 7

SOFTLIBRARIES

The SoftLibraries section of IBSS allows an administrator to add, edit or delete external libraries. SoftLibraries can be created by developers and can be setup as an application or a lookup and are available to all nodes. After being added to IBSS, SoftLibraries must be added and activated in IBFA. Only active SoftLibraries can be used by callout programs in RateManager. In SoftLibraries you can:

- **Add Libraries**
- **Edit Libraries**
- **Delete Libraries**

SoftLibraries are an advanced topic. This chapter defines the screen features and the functionality. Developing the code that is called will not be discussed here.

SoftLibraries can be found under Nodes -> SoftLibraries.

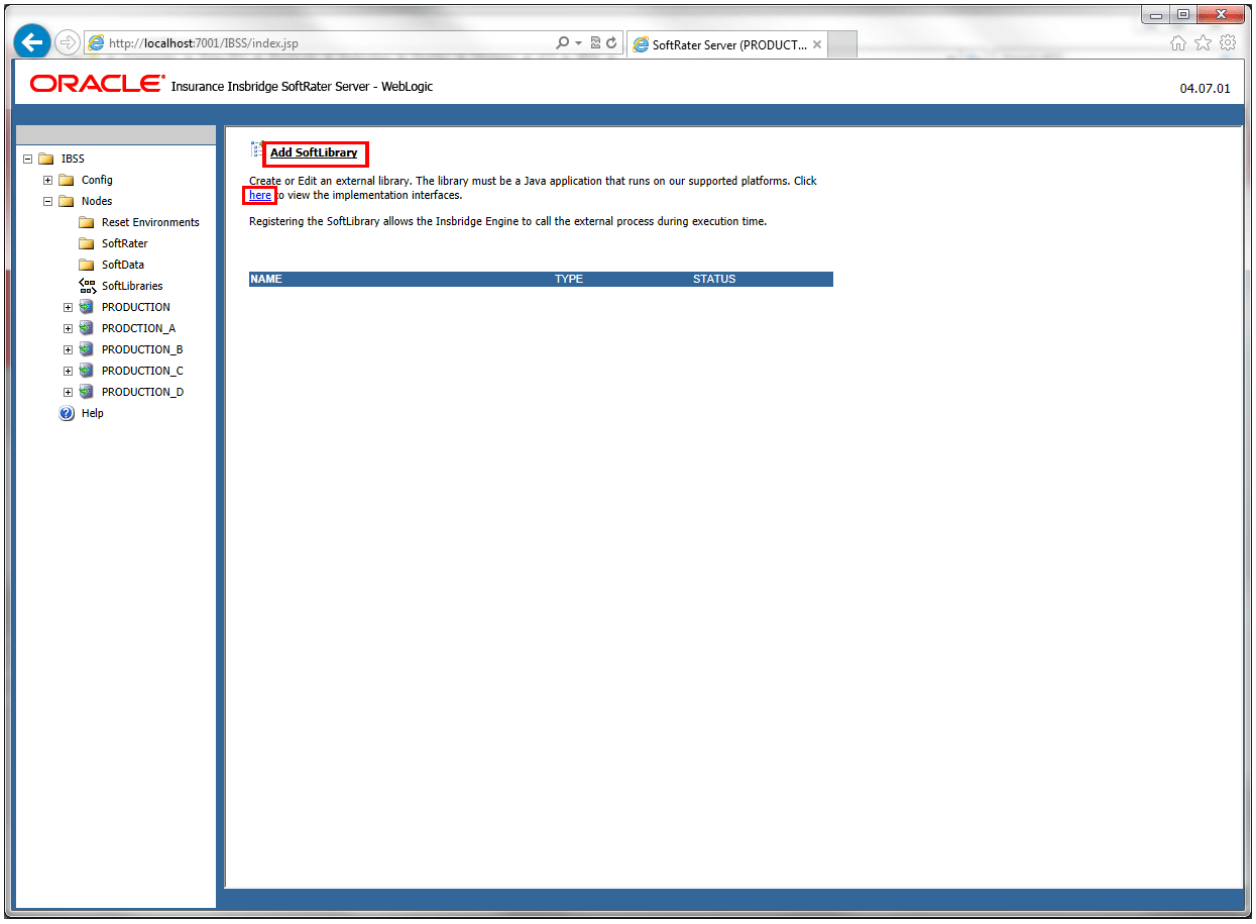


Figure 50 SoftLibraries Main Menu

The main SoftLibraries page lists any entered SoftLibraries. Other information includes:

Name: The name of the SoftLibrary.

Type: The type of SoftLibrary, either .NET Web Service, WebSphere or WebLogic.

Status: The status, either active or disabled.

NOTE: *The status determines whether or not a SoftLibrary is available to IBFA. If a SoftLibrary is disabled, it will not be displayed in IBFA. If a SoftLibrary is active, it will be displayed.*

The following information is found at the top of the window:

Add SoftLibrary: This link places you on the new SoftLibrary window.

Click here link: Found on "Click here to view the implementation interfaces." This pulls up a separate window that contains the implementation interfaces required by SoftLibraries. This window can be left open while you work.

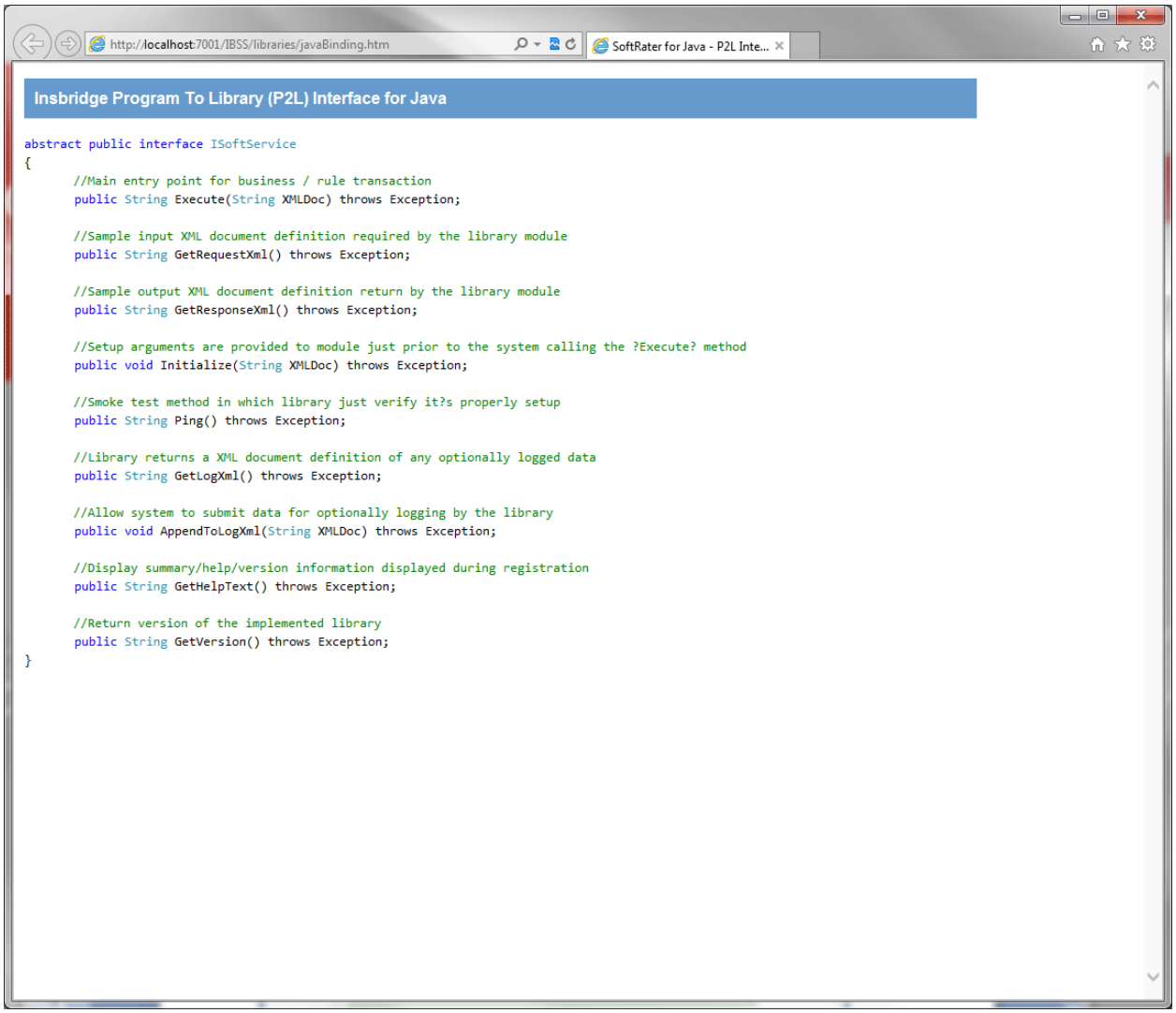


Figure 51 SoftLibrary Implementation Information

ADDING SOFTLIBRARIES

Adding a SoftLibrary is a multi-step process.

If you want to create a Java class library, the .JAR file must be placed in a specific location and then entered into IBSS. After entering in IBSS, the SoftLibrary can be added in IBFA. Please have the Class Name and Endpoint information ready.

Step 1: Placement of Library on App Server

NOTE: *For the first part of adding a SoftLibrary, you must have access to the application server where IBSS is located.*

1. Have your Java class library ready as a .JAR file.
2. On the application server where IBSS is located, stop the app server.
3. Place the .JAR file in the lib location designated by application:
    a. **WebLogic:** Place the .JAR file in the /$WL_Home/user_projects/domains/YOUR_DOMAIN/lib.
    b. **WebSphere:** Place the .JAR file in the WebSphere install directory /$WS_Home/AppServer/lib.
    c. **JBoss:** Place the .JAR file in the JBoss install directory /$JB_Home/server/SR_Server/lib.

    Where$_Home is the relative path to the app server installation location.
4. Restart the app server.

NOTE: *To enter a library in IBSS, you must have the Class Name.*

Step 2: Entering Library in IBSS

1. On the SoftLibrary page, click the **Add SoftLibrary link** at the top of the window. A separate window is displayed.

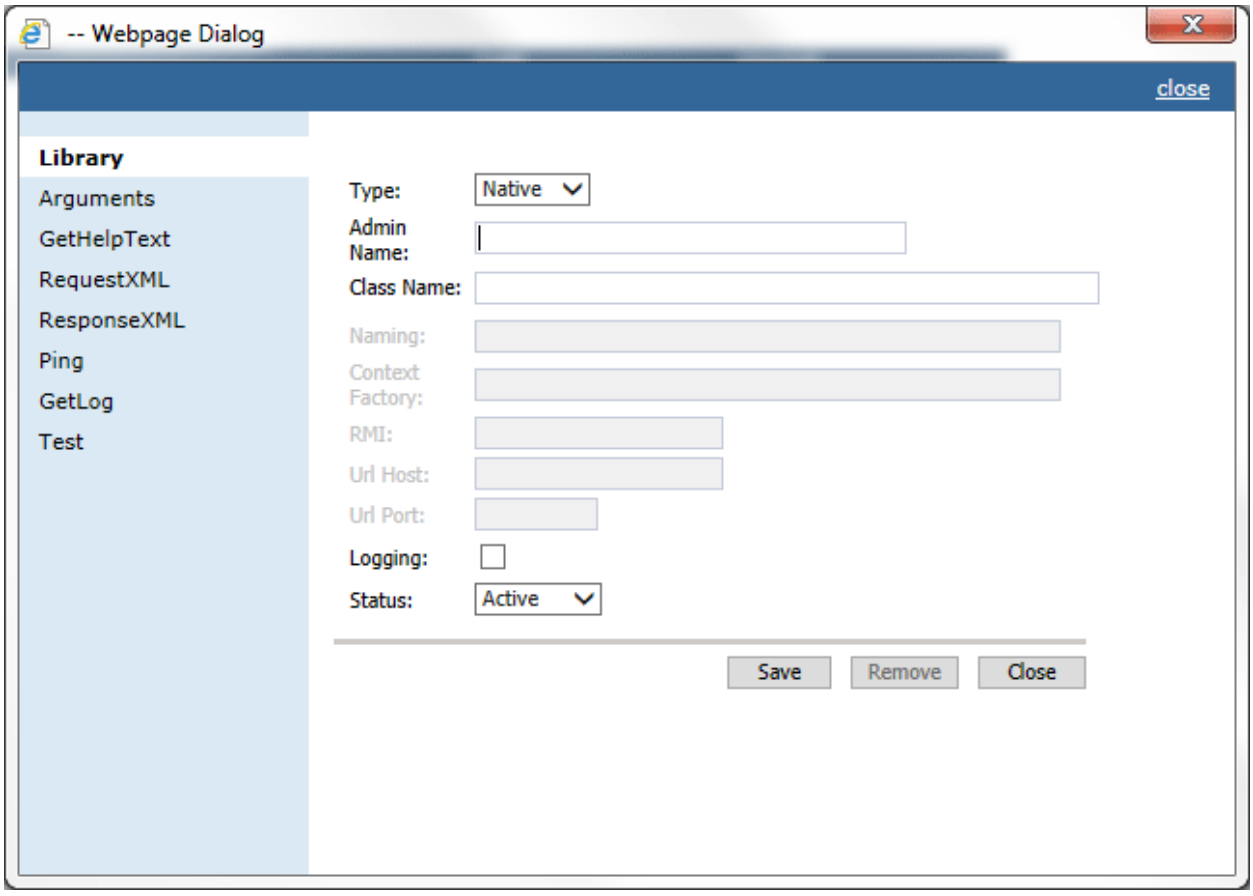


Figure 52 SoftLibraries Main Screen

The SoftLibrary window allows you to enter in basic information and to view critical information about the SoftLibrary. There are eight options on the side menu:

- **Library:** The basic SoftLibrary information is entered here.
- **Arguments:** Enter or edit the arguments for the SoftLibrary here.
- **GetHelpText:** This is an information window. If you want a quick view of the SoftLibrary, this window contains the basic information.
- **Request XML:** The request XML will be displayed here. This will tell you what requests you are sending out. This is an information only window.
- **Response XML:** The response XML will be displayed here. This will tell you exactly what you are getting back. This is an information only window.
- **Ping:** Will ping the server to verify server response.
- **GetLog:** If logging is turned on, this window will display logs.

- **Test:** Allows you to test the SoftLibrary.

There are three buttons located in the Library Add/Edit section.

- **Save** – Saves your work.
- **Remove** – Removes the entire library.
- **Close** – Closes the window without saving.

2. The first window is the Library information window. Information must be entered here before you can continue.
   - Select the Type. There are two types of SoftLibraries in IBSS:
     - **Native** – The library is installed as an application on the app server.
     - **Lookup** – The library uses an EJB Lookup.

For Native Entries

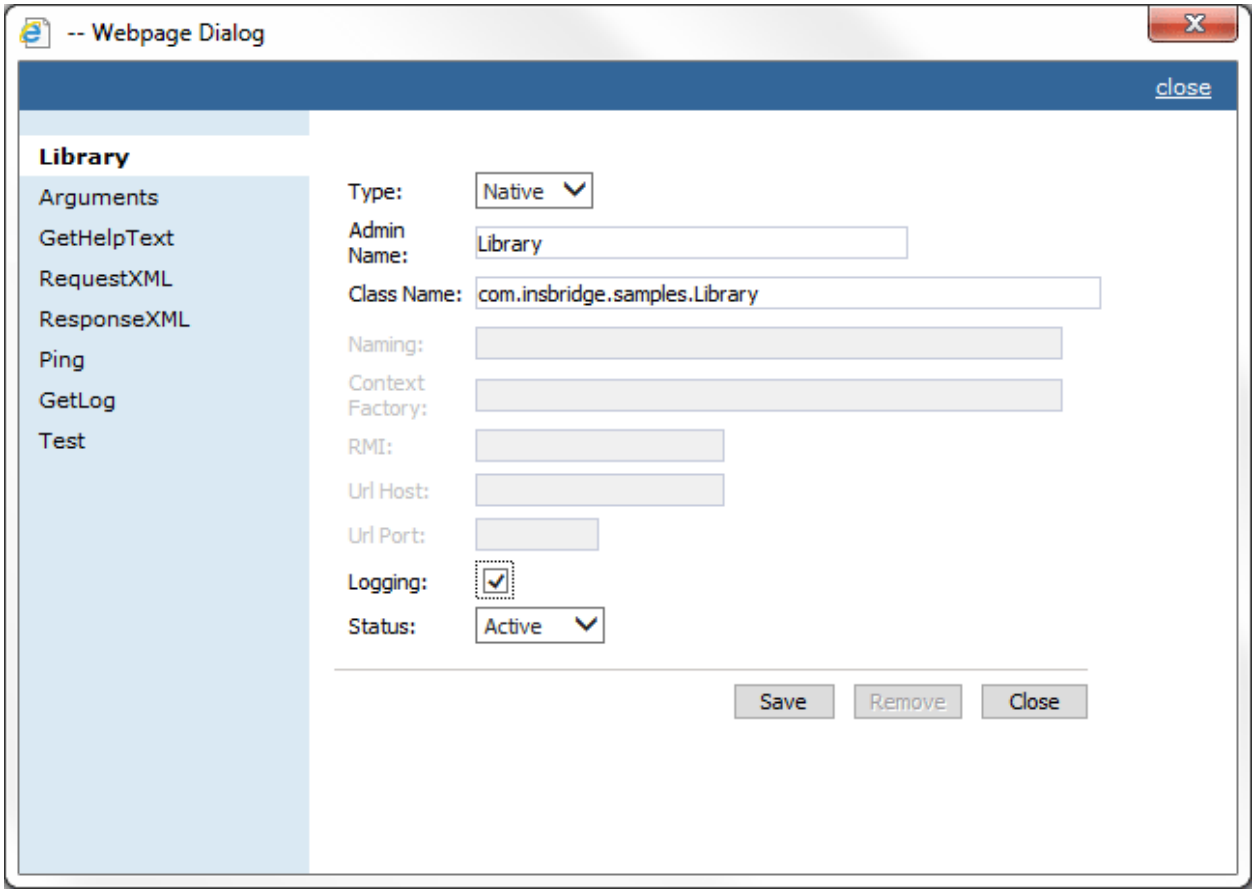


Figure 53 SoftLibrary Native Type

- **Admin Name:** This is the name of the SoftLibrary as it is displayed in RateManager. It is also the name that will be displayed on the lower section of the SoftLibraries page. Admin names must be unique.
- **Class Name:** This is the name of the actual SoftLibrary created by the developer from the program where it was created. This information can be obtained from the developer. Class names must be unique.
- **Logging:** Check this box if you want logging for this program. Leave it blank for no logging. The default is for no logging take place.
- **Status:** The status of the SoftLibrary. If active is selected, the SoftLibrary will be displayed to RateManager users. If disable is selected, the SoftLibrary will not will be displayed to users. The default status is disabled.

For Lookup Entries

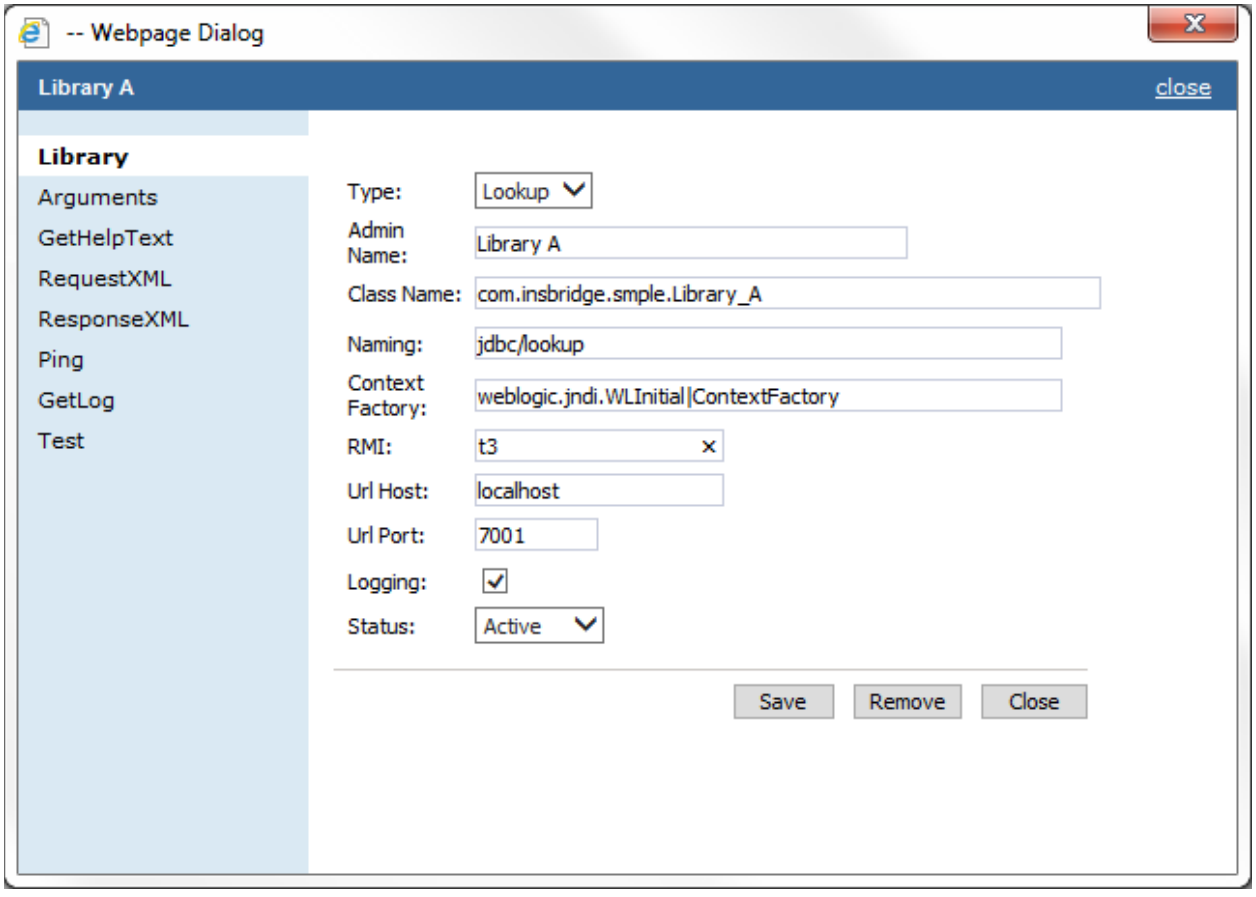


Figure 54 SoftLibrary Lookup Type

- **Admin Name:** This is the name of the SoftLibrary as it is displayed in RateManager. It is also the name that will be displayed on the lower section of the SoftLibraries page. Admin names must be unique.

- **Class Name:** This is the name of the actual SoftLibrary created by the developer from the program where it was created. This information can be obtained from the developer. Class names must be unique.
- **Naming:** The name of the data source.
- **Context Factory:** A URL context factory is a special object factory that creates contexts for resolving URL strings.
- **RMI:** The Java Remote Method Invocation used.
- **URL Host:** The name or IP of the host server.
- **URL Port:** The port number used by the host server.
- **Logging:** Check this box if you want logging for this program. Leave it blank for no logging. The default is for no logging take place.
- **Status:** The status of the SoftLibrary. If active is selected, the SoftLibrary will be displayed to RateManager users. If disable is selected, the SoftLibrary will not will be displayed to users. The default status is disabled.

3. Click **Save** to add your SoftLibrary. It will be displayed on the lower portion of the screen.

If there are any errors with the new SoftLibrary, a message will be displayed at the top of the screen explaining the conflict.

Step 3: Entering in IBFA

SoftLibraries entered in IBSS will not be displayed in RateManager until added and activated in IBFA. You will need to have the Admin Name, Class Name, and Endpoint in order to add the SoftLibrary in IBFA. Please see the IBFA User Guide for more information.

Arguments

After a library is created, arguments can be entered. Arguments are settings that the library should use.

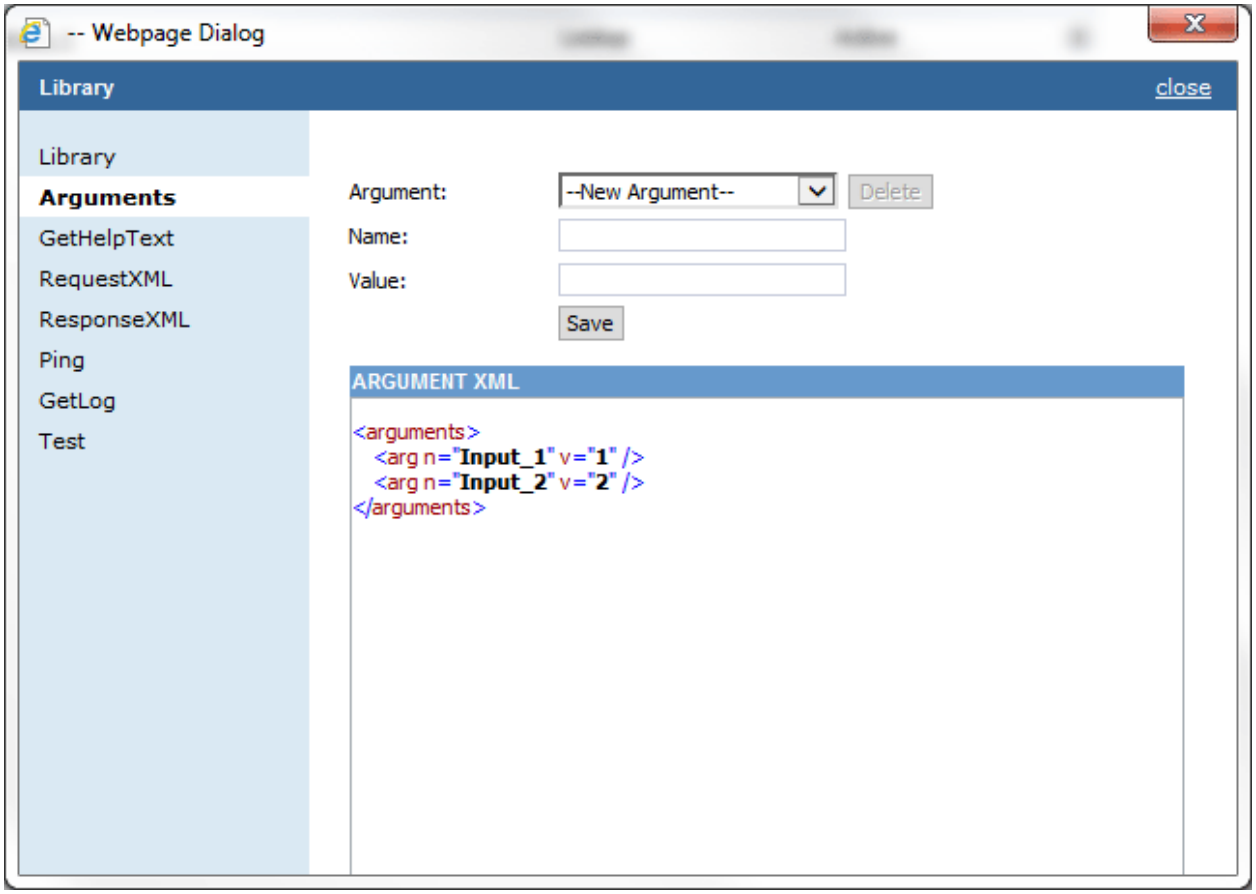


Figure 55 Editing Arguments

Adding an Argument

To create an argument, select **--New Argument--** in the Argument drop down list. Enter in a **Name** and a **Value**. Click **Save**. Your argument will be listed below.

To add an argument, select the **Argument** you want from the drop down menu. The argument name will populate the **Name** field. The current value will populate the **Value** field. If these arguments are correct, click **Save**. If you need to, you can edit both the name field and the value field by typing in what you need.

Editing an Argument

To edit an argument, select the argument from the Argument drop down list. Edit the **Name** and a **Value**. Click **Save**. Your argument will be listed below with the new values.

Deleting an Argument

If there is an argument that you do not need, you can delete it. Select the argument from the drop down menu and click the **Delete** button. The argument will be removed.

GetHelpText

If you want to check your SoftLibrary, continue on the **GetHelpText** option on the side menu. The basic information will be listed.

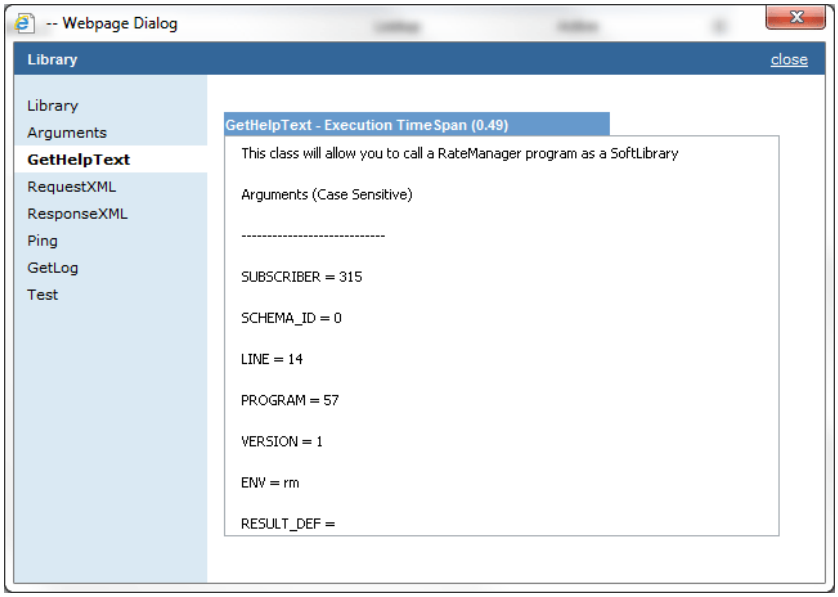


Figure 56 GetHelpText Window

RequestXML

The next information window is the **RequestXML** window. This window will display the request XML that is going out.

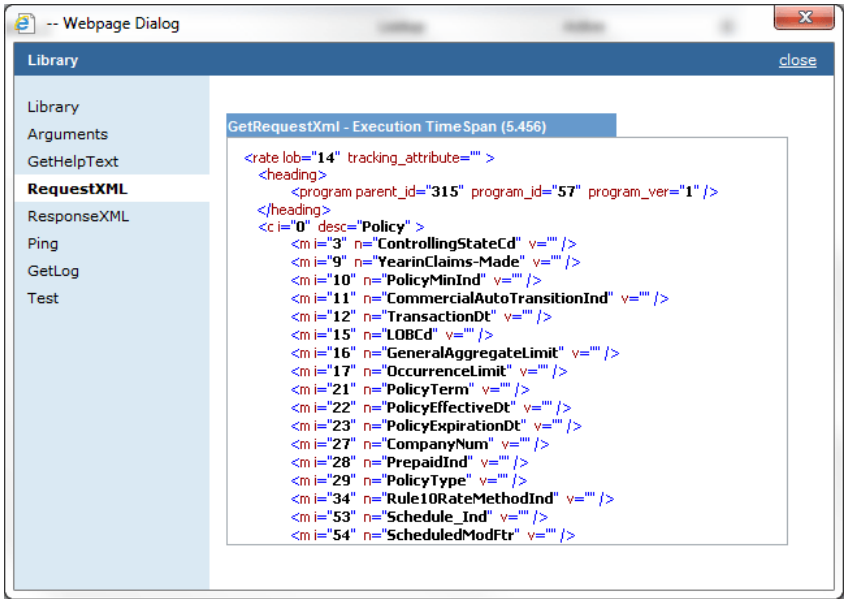


Figure 57 RequestXML Window

ResponseXML

The **ResponseXML** window will tell you exactly what you are getting back.

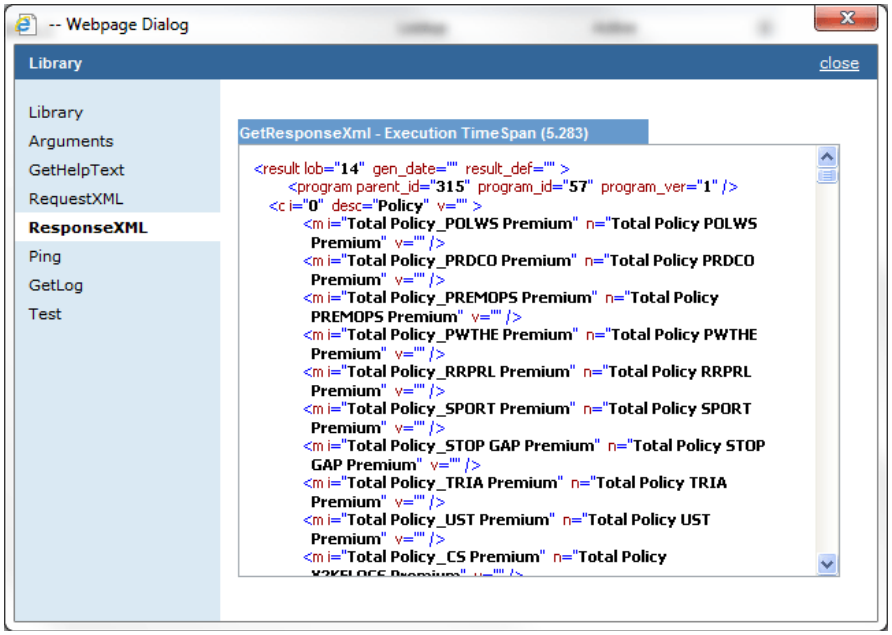


Figure 58 ResponseXML Window

Ping

If you would like to ping the server, click the **Ping** menu option.

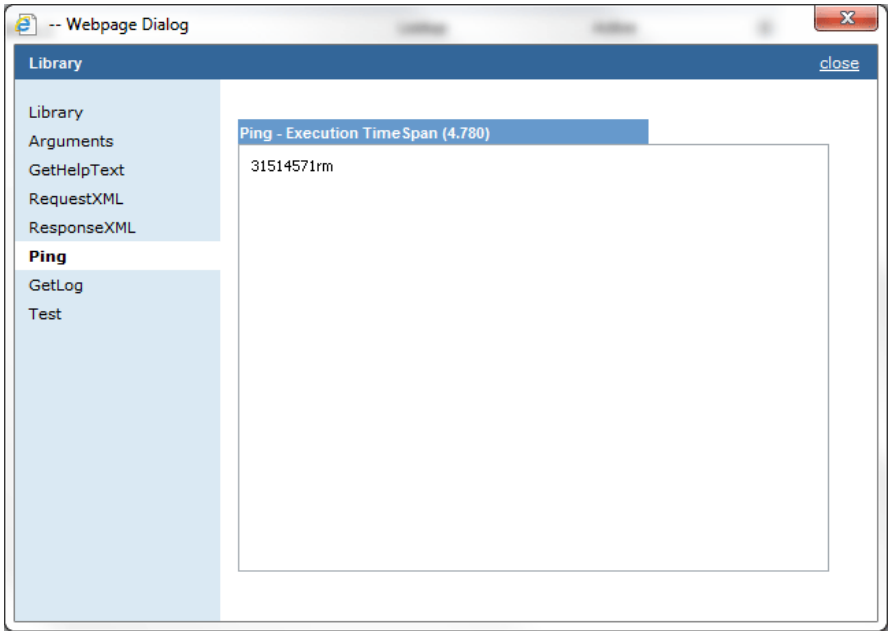


Figure 59 Ping Window

GetLog

To view the logs, click the **GetLog** menu option.

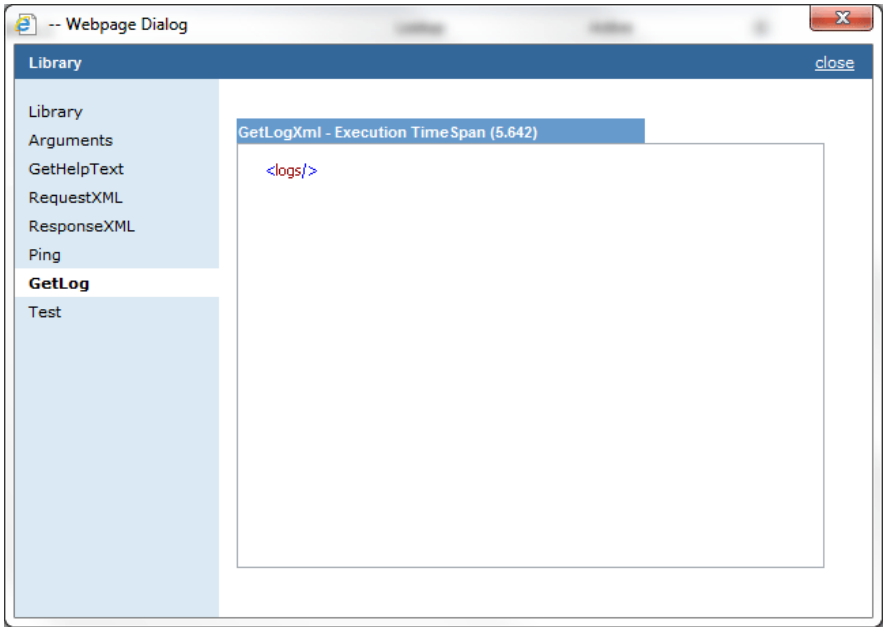


Figure 60 GetLog Window

Test

The last menu option is Test. Test has three parts

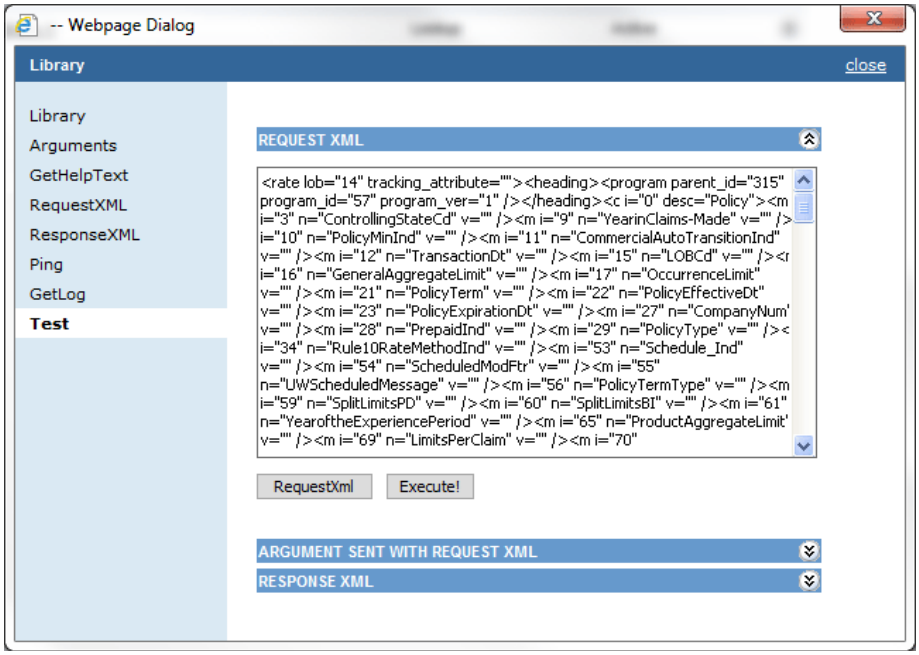


Figure 61 Test Window – Request

- First you must enter in the RequestXML. You can enter in your own XML or you can click the RequestXml button and have the XML auto populate for you.
- You can view the arguments sent with the request XML.
- Click the Execute! button to execute the XML. The responseXml will be displayed. To view the Arguments sent or to view the RequestXML, click the expand button .

After a SoftLibrary has been entered, it is listed on the main SoftLibrary page. You can view quick details by expanding the arrow on at the end of the field.

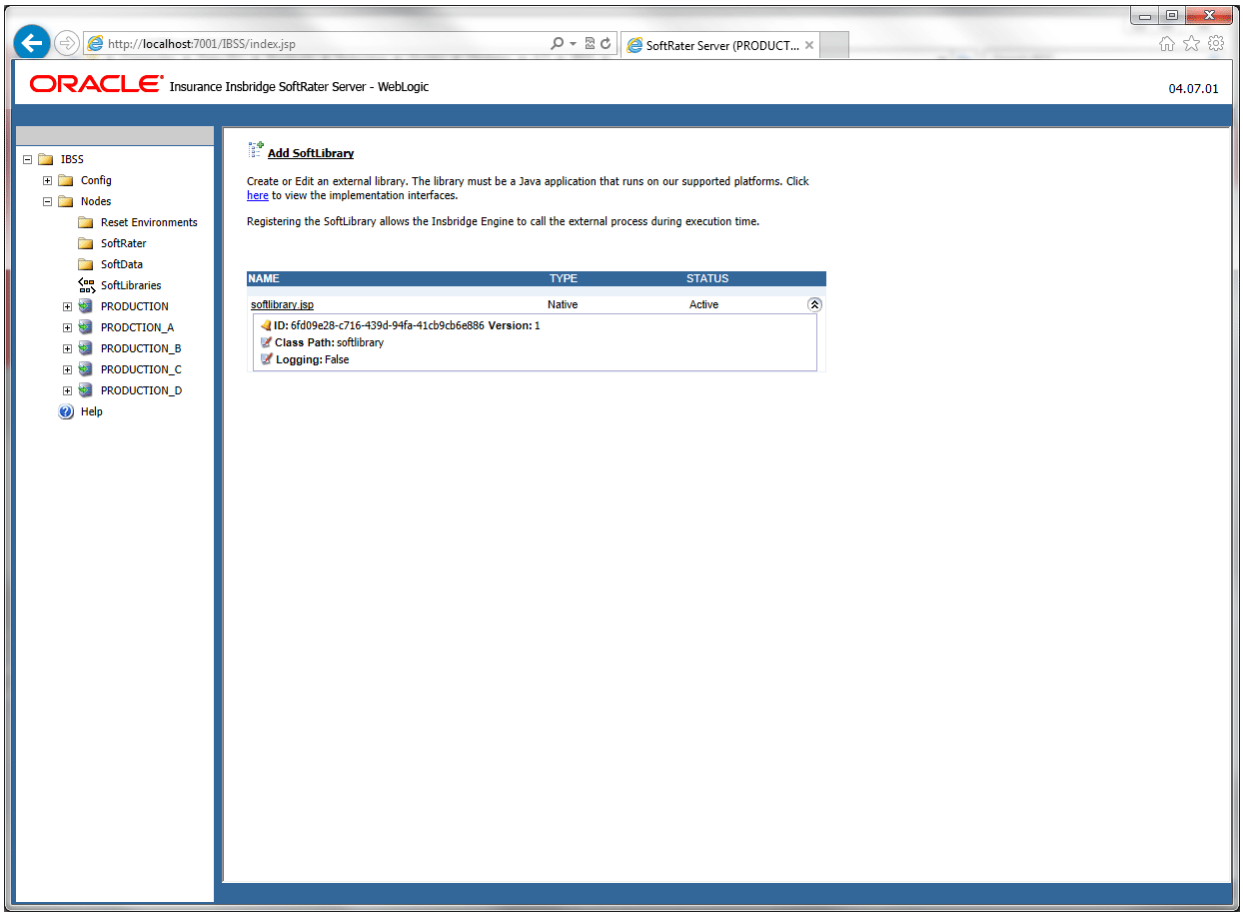


Figure 62 Completed SoftLibrary

EDITING SOFTLIBRARIES

Any SoftLibrary can be edited at any time. Be aware that changes to a SoftLibrary in IBSS may not be automatically carried over to IBFA. If you edit the Admin Name or Class Name, you will have to edit the same information in IBFA. You also could delete the SoftLibrary in IBFA and then re-add it. Be aware that changes to a SoftLibrary here will affect any programs in RateManager that currently call this SoftLibrary without sending notice to users.

To edit a SoftLibrary, click on the name. The SoftLibrary window will be displayed. Make your edits and click **Save**.

Deleting SoftLibraries

Any SoftLibrary can be deleted at any time. Be aware that changes to a SoftLibrary here will affect any programs in RateManager that currently call this SoftLibrary without sending notice to users. SoftLibraries will also have to be deleted out of IBFA.

To delete a SoftLibrary, click the name. The SoftLibrary window will be displayed. Click the **Remove** button. A warning message will be displayed, click OK to remove the SoftLibrary. Click Cancel to return to the SoftLibrary window.

Sample for Sending a Message to Call a SoftLibrary

This sample can be used when you have a third party application that needs to communicate with an IBSS SoftLibrary. This is the message that gets sent to the SoftLibrary services.

Sample Message:

```
<o_skywire>
<softservices type='lib'>
        <msg command='Execute' name='MyServiceName'>
                <my_xml_root>
                        <My_Message> Abc-123</My_Message>
                </my_xml_root>
        </msg>
</softservices>
</o_skywire>
```

Where:

Command	Definition
type='lib'	Standard argument for requesting (SS) libraries. This is a required entry.
command='*xxx*'	Where *xxx* equals the option to be performed. Options include: 'Execute' GetRequestXml' 'GetResponseXml' 'Ping' 'GetLogXML' 'AppendToLogXml' 'GetHelpText' GetArgs' **NOTE:** Values are **NOT** case sensitive. This is a required entry.
name='*xxx*'	Where *xxx* equals the admin name of the (SS) from the IBSS (SS) admin console. **NOTE:** Values **ARE** case sensitive. This is a required entry.
Abc-123	Where *Abc-123* equals your XML.

CONTACTING SUPPORT

If you need assistance with an Oracle Insurance Insbridge Rating and Underwriting System product, please log a Service Request using My Oracle Support at https://support.oracle.com/.

Address any additional inquiries to:

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200
oracle.com

TTY Access to Oracle Support Services

Oracle provides dedicated Text Telephone (TTY) access to Oracle Support Services within the United States of America 24 hours a day, seven days a week. For TTY support, call 800.446.2398.

Deaf/Hard of Hearing Access to Oracle Support Services

To reach Oracle Support Services, use a telecommunications relay service (TRS) to call Oracle Support at 1.800.223.1711. An Oracle Support Services engineer will handle technical issues and provide customer support according to the Oracle service request process. Information about TRS is available at http://www.fcc.gov/cgb/consumerfacts/trs.html, and a list of phone numbers is available at http://www.fcc.gov/cgb/dro/trsphonebk.html.

GLOSSARY TERMS

A	
Administrator:	The person designated by your company who has the authority to create and change groups, usernames, passwords and restrictions.
Algorithm:	A sequence of steps used to perform a calculation.
Assigned Driver:	A driver who has been assigned to a vehicle.
Assigned Vehicle:	A vehicle that has been assigned a driver.
Authoring Environment:	The physical machine where RateManager is installed.
C	
Calculated Variable:	Calculated Variables are used when a result cannot best be derived from simple data mapping in a table. For example, if age is not passed as an input, but is a criteria needed in determining other factors, you would use a Calculated Variable to calculate driver age from the inputs of effective date and driver date of birth. Calculated Variables look at every node (driver, vehicle, location, etc.) independently and create a result for each. Once a calculated variable is created, the result can be used in any other variable.
Callouts:	A set of a single or multiple programs and/or SoftLibraries that allows users to call needed operations at a specific time from either inside the system or outside the system.
Category:	A user defined group of information that defines inputs, variables, algorithms and the overall structure of the program.
Criteria:	An input or variable used in a mapped variable to determine which value to return. Any input or variable can be used as a criteria.
D	
Data Type:	The type of data associated with a parameter. RateManager supports three (decimal, integer and string) data types for variables and inputs and a fourth (date) for inputs and result variables only.
Date:	A data type supported for inputs and result variables only. For more information, see Dates in the Contents section.
Decimal:	A data type supported for all types of variables and inputs. Examples of decimal values are 3.1415, 18 and 0.995.
Default Value:	Used by a mapped variable if no match is found based on the criteria.
Dependency:	When copying a variable, algorithm or driver assignment, any element that requires another element to be present or defined will be listed. All dependencies must be resolved.
Driver Assignment:	Driver assignment is an auto insurance specific method of assigning a particular driver to a particular vehicle on a policy, based on certain criteria. Criteria often differ on a carrier basis, and sometimes even on a program level.
Driver Assignment Scenario:	A driver assignment scenario is a list of instructions that define the main sequence of operations to properly define the driver assignment logic for a carrier. A scenario is based upon the same algorithm principle used

	throughout the RateManager software.
E	
Export:	Exports allow users to export all elements of a program from one database to another database or within the same database. Or to export data in tab-delimited form, from a RateManager table to an outside location.
F	
Flag:	A variable that holds a 1 for true and a 0 for false.
Flag Driver Algorithm:	An algorithm used by a Flag Driver Function to flag drivers based on certain criteria.
Flag Driver Function:	A built-in function used in the main driver assignment that defines the scope of the flagging operation. Most require an association with a predefined Flag Driver Algorithm.
Flag Vehicle Algorithm:	An algorithm used by a Flag Vehicle Function to flag vehicles based on certain criteria.
Flag Vehicle Function:	A built-in function used in the main driver assignment that defines the scope of the flagging operation. Most require an association with a predefined Flag Vehicle Algorithm.
G	
Global:	An input or variable that is available to all programs under a specific subline.
Global Input:	A value that is passed into the rating system.
Global Result:	A value that is passed out of the system after rating.
Group:	A set of users that have the same access rights.
I	
IBFA:	Insbridge Framework Administrator. IBFA is an administrative tool used to configure Insbridge applications and setup RateManager database connections. IBFA will be located on a Windows Server machine. IBFA/SR-WIN is an Insbridge Framework Administrator/SoftRater for Windows.
IBSS:	Insbridge SoftRater Server. IBSS is the administrative tool for the SoftRater engine. The SoftRater engine is a multi-platform component within IBRU that executes the rules, rating and underwriting instructions as defined by the user in RateManager. IBSS is usually located on a Java machine. IBSS/SR-JAVA is an Insbridge SoftRater Server/SoftRater for Java.
IBRU:	Insbridge Rating and Underwriting System. This is the entire system.
If:	A step type available for use in calculated variables and algorithms. For more information, see If in the contents section.
Import:	Import allows users to bring in programs from an outside location into RateManager. Or to import data in tab-delimited form, into a RateManager table.
Input:	A value that is passed into the rating system.
Integer:	A data type supported for all types of variables and inputs. Examples of integer values are 3, 1859865 and -47.
Interpolation:	An estimated value derived from two known values.
L	
Library:	The Library is where templates are stored and managed.
Linked Variables:	Two or more mapped variables that have been associated with one another

	because they use the same criteria.
Lock:	A lock will close all associated Variables, Algorithms, Driver Assignments, Sequencing and Result Mappings in a program version from deletions and edits.
Logical Environment:	An environment created for a subscriber in IBFA. It defines package location, engine location and database location in addition to several other supporting data items. This environment is used for rating and/or SRP management. Each database connection will have a logical environment.
M	
Mapped Variable:	A variable that uses other variables and inputs as criteria in determining the appropriate value. See Mapped Variables in the Contents section for more information.
Mask:	A feature that allows the customer to determine how data should be interpreted. See Masking in the Contents section for more information.
N	
Normal Rating Algorithm:	The most common type of algorithm. Examples of what it can be used for are: • Determine premiums • Calculate differences in limits being passed into the system vs. limits being rated by the system • Assign tiers
O	
Operator:	A built-in mathematical function used in calculations and comparisons.
P	
Package:	A small file that holds all the RateManager logic for a specific program and version.
Package Location:	A pointer to a location where SoftRater Packages (SRP's) are stored.
Physical Environment	A physical environment is generally referred to as a physical machine.
Program:	A planned group of procedures executed in a specific order to return a rating. Programs in RateManager typically correspond to rate manuals. Programs can be either created by the users or imported.
Program Date Mask:	Specifies how SoftRater interprets dates being passed into an input file.
Program Folders:	A RateManager file management system that functions in much the same way as Microsoft Windows Explorer. This multi level setup allows for an unlimited number of program folders and subfolders to be placed underneath a subline.
R	
Rank Driver Algorithm:	An algorithm used by a Rank Driver Function to rate drivers based on certain criteria.
Rank Driver Function:	A built-in function used in the main driver assignment that defines the scope and sorting order of the ranking operation. Most require an association with a predefined Rank Driver Algorithm.
Rank Vehicle Algorithm:	An algorithm used by a Rank Vehicle Function to rate vehicles based on certain criteria.
Rank Vehicle Function:	A built-in function used in the main driver assignment that defines the scope and sorting order of the ranking operation. Most require an association with a predefined Rate Vehicle Algorithm.

Rating Environment:	The physical machine(s) where SoftRater is installed. This is typically the same as a SoftRater node.
Reconcile:	A comparison feature that compares one program version against another version in the same program and generate a report of the differences.
Restrictions:	Limitations on viewing and editing pages and fields in the system. Restrictions are assigned and changed by the Administrator.
Result Mapping:	A defined set of results, inputs and variables displayed in the output file.
Revision:	A variable specific type of versioning. See Versioning in the Contents section for more information.
RM:	RateManager. RateManager is a component within IBRU that enables users to manage the product definition and modification process, including rating and underwriting logic.
S	
Sequence:	The order in which algorithms run. See Sequencing in the Contents section for more information.
SoftLibrary:	A SoftLibrary is a specially developed program that performs a specific task. SoftLibraries may run their own code or call upon other systems to obtain information outside of RateManager, for example, obtaining a credit score
SoftRater Node	A SoftRater node is either an IBFA (without RateManager) or IBSS instance on a physical environment.
Source:	The source is the creator of a template and will also be the name of the new subline.
SR:	SoftRater. The engine that executes the rating, rules and underwriting instructions defined within RateManager. The rating environment for runtime execution and processing of business content. SoftRater can be further defined by the operating system where it has been loaded.
SRP:	SoftRater Packages. A package that holds all the RateManager logic for a specific program and version.
SR-JAVA:	SoftRater for Java. This is also another name for IBSS.
SR-WIN:	– SoftRater for Windows. This is also another name for IBFA.
String:	A data type supported for all types of variables and inputs. Examples of string values are "2.718", "The quick brown fox jumps over the lazy dog." and "001".
Subline:	Sublines are classifications that fall in between lines of business and program folders. Sublines allow for the separation of programs by source.
T	
Tab-delimited:	A type of text file in which columns are separated by tabs. This is the required format for importing tables into RateManager.
Template:	Templates are exact copies of existing programs within a line of business that can be from within your own user group, any other user group within the company or even from an outside company.
U	
Unassigned Driver:	A driver who has not been assigned to a vehicle.
Unassigned Vehicle:	A vehicle that has not been assigned a driver.
Underwriting Algorithm:	A type of algorithm used to determine if a policy meets the requirements of the company.

V	
Variable:	A name used to represent a value that can change. See Variables in the Contents section for more information.
Version:	One of a sequence of copies of a program, each incorporating new modifications. See Versioning in the Contents section for more information.
VFS:	Virtual File Servers. Virtual file server management allows you to set up servers that are in different locations where packages can be downloaded.
W	
Wildcard:	An option available for mapped variables that tells RateManager that one or more rows ignore the value passed in for the criteria. See Variables in the Contents section for more information.
Workflow:	A collection of programs from all lines of business combined with result group mappings and assigned to execute in sequence that returns a single or multiple results.
Working Category:	A classification used to define how elements should run. See Categories in the Contents section for more information.
X	
XML ID:	A number automatically assigned by RateManager to identify inputs and categories.

INDEX

E

F

G

H

I

J

L

M

S

T

V

W

X