

**Oracle Insurance**

**Insbridge Enterprise  
Rating  
ESI for Java User Guide**

Release 5.2

July 2016

Copyright © 2005, 2016, Oracle and/or its affiliates. All rights reserved.

Oracle Insurance Insbridge Enterprise Rating ESI for Java User Guide

Release 05.02.xx

Part # E76660-01

Library # E76672-01

July 2016

Primary Authors: Aidi Nui, 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 is software or related documentation that 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 END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware 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 that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware 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.

# CONTENTS

	PREFACE .....	VII
	Audience .....	vii
	Related Documents .....	vii
	Conventions .....	vii
	System Requirements.....	viii
	Manual History .....	viii
<b>CHAPTER 1</b>		
	INTRODUCTION TO INSBRIDGE EXTENDED SERVICES INTERFACE .....	9
<b>CHAPTER 2</b>		
	USING ESI: IBSS - EXAMPLE .....	12
	Creating a Rate Object .....	12
	IBRequestResponse Interface .....	12
	Creating a Rate Object .....	12
	Input Batch – Creating, Submitting, Executing and Monitoring .....	14
	Result Batch.....	15
<b>CHAPTER 3</b>		
	BASE DOMAIN TYPES .....	16
	Use Case Summary.....	17
	Quick Guide Table .....	21
<b>CHAPTER 4</b>		
	RATEMANAGER INTERFACE.....	23
	EsiRateManager .....	23
	EsiRateManager .....	23
	Environment Items – getAvailableEnvironments, All Non-Secured.....	23
	Environment Items – getAvailableEnvironments, Secured.....	24
	General – createLauchDefinition (Selected Version, Network Login) .....	24
	General – createLauchDefinition (First Version, Network Login) .....	24
	General – createLauchDefinition (Selected Version, RateManager Login).....	24
	General – createLauchDefinition (First Version, RateManager Login).....	25
	Global Items – createInput.....	25
	Global Items – updateInput.....	25
	Global Items – deleteInput .....	25
	Global Items – getInputsArray .....	26
	Global Items – createCategory .....	26
	Global Items – createCategory with Inputs.....	26
	Global Items – getCategoryXML.....	26
	Global Items – updateCategory .....	26
	Global Items – deleteCategory .....	27

Program Items – createLine.....	27
Program Items – updateLine.....	27
Program Items – deleteLine.....	27
Program Items – createFolder .....	27
Program Items – updateFolder .....	28
Program Items – deleteFolder .....	28
Program Items – getFoldersXML.....	28
Program Items – createProgram.....	28
Program Items – updateProgram.....	28
Program Items – deleteProgram.....	29
Program Items – createProgramVersion .....	29
Program Items – copyProgram .....	29
Program Items – updateProgramVersionLock.....	29
Releases Items – createRelease .....	30
Releases Items – deleteRelease .....	30
Releases Items – addReleaseProgram .....	30
Releases Items – removeReleaseProgram .....	30
Releases Items – getReleasePrograms .....	31
Security Items – login.....	31
Security Items – networkLogin.....	31
Security Items – logout .....	31
Security Items – createUser.....	31
Security Items – updateUser.....	32
Security Items – getUsers .....	32
Security Items – getUserGroups.....	32
SRP Items – createSRP .....	32

**CHAPTER 5**

FRAMEWORK ADMINISTRATOR INTERFACE.....	33
EsiFrameworkAdministrator.....	33
EsiFrameworkAdministrator.....	33
SRP Items – copySRP.....	33
SRP Items – deleteSRP.....	34
SRP Items – exportSRP .....	34
SRP Items – getSRPList.....	34
SRP Items – getBytesFromFile.....	34
SRP Items – importSRP .....	35
SRP Items – loadSRP .....	35
SRP Items – moveSRP .....	35
SRP Items – saveFile .....	35
SRP Items – unloadSRP.....	36

**CHAPTER 6**

INPUTBATCH .....	37
Create New Rate.....	37
Add Batch to Runtime DB .....	37

	Execute Batch .....	37
	Retrieve Input Rates .....	37
	Submit Insbridge XML from Memory .....	38
	Submit Insbridge XML Files from Disk.....	38
	Submit Rate Object to Runtime DB .....	38
<b>CHAPTER 7</b>		
	RESULT BATCH.....	39
	Create New Result.....	39
	Retrieve Results.....	39
<b>CHAPTER 8</b>		
	JOB .....	40
	Retrieve Job Status.....	40
	Stop the Job .....	40
<b>CHAPTER 9</b>		
	RATE OR RESULT .....	41
	Add Rating Details .....	41
	Add Program Details .....	41
	Add Parent Category .....	41
	Add Child Category.....	41
	Add Category Inputs .....	41
<b>CHAPTER 10</b>		
	ESI USER SETUP AND AUTHENTICATION.....	42
	Adding a User .....	42
<b>CHAPTER 11</b>		
	ESI SYSTEM CONFIGURATION .....	44
	ESI – IBSS SYSTEM CONFIGURATION.....	45
	LIST OF FILES.....	45
	CONFIGURATION CHANGES .....	45
	ESI IBSS Testing .....	47
	Submit Insbridge XML as XML .....	47
	Submit Insbridge XML as Objects.....	47
	Submit Insbridge XML Files to Queue .....	47
	Submit Insbridge XML Files as Objects To Queue.....	48
	DATABASE FUNCTIONS .....	48
	Get Input XML by Policy Number.....	48
	Get Input XML by policy Number as Object.....	49
	Get Result XML by Policy Number .....	49
	Get Result XML by Policy Number as Object.....	49
	Copy Batch.....	50
	BATCH – STARTING AND MONITORING .....	50
	Start Batch .....	50
	Stop Batch.....	50
	Get Batch Status .....	50

**CHAPTER 12**

ESI TESTER ..... 52  
 IBSS Features/Methods ..... 52  
 Database Functions ..... 53  
 RM Features/Methods ..... 54  
 Test Security ..... 54  
 Test Globals ..... 55  
 Test Program Items ..... 56  
 Test IBFA ..... 57  
 Test Releases ..... 58  
 Test List Items ..... 59

**SUPPORT**

CONTACTING SUPPORT ..... 63

**INDEX**

INDEX ..... 64

# LIST OF FIGURES

- FIGURE 1 ADDING A NEW USER..... 43
- FIGURE 2 ALLOWING EXTERNAL API ACCESS..... 44
- FIGURE 3 SELECTING INSTANCE TO TEST..... 52
- FIGURE 4 CONNECTION STATUS..... 52
- FIGURE 5 TESTING IBSS..... 53
- FIGURE 6 TEST SECURITY ..... 54
- FIGURE 7 TEST GLOBALS..... 55
- FIGURE 8 TEST PROGRAM ITEMS..... 56
- FIGURE 9 TEST IBFA ..... 57
- FIGURE 10 TEST RELEASES ..... 58
- FIGURE 11 JAVA ESI TESTER FOR RM ..... 59
- FIGURE 12 TESTER OUTPUT ..... 62
- FIGURE 13 PROGRAM RESULT XML FILE..... 62

## PREFACE

Welcome to the *Oracle Insurance Insbridge Enterprise Rating ESI for Java Guide*. This guide describes how you can configure and use the Insbridge Extended Services Interface (ESI).

## AUDIENCE

This guide is intended for application developers, programmers and others tasked with configuring the ESI interface. Users should be familiar with Java IDE.

## RELATED DOCUMENTS

For more information, refer to the following Oracle resources:

- The Oracle Insurance Insbridge Enterprise Rating RateManager User Guide.
- The Oracle Insurance Insbridge Enterprise Rating Javadoc File.
- The Oracle Insurance Insbridge Enterprise Rating Framework Administrator User Guide.
- You can view these guides in-line at this address:

<http://www.oracle.com/technology/documentation/insurance.html>

## CONVENTIONS

The following text conventions are used in this document:

Convention	Description
<b>bold</b>	Boldface type indicates graphical user interface elements associated with an action.
<i>italic</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
<code>monospace</code>	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.

Oracle Insurance recommends the following system requirements for client workstations for optimal performance. Currently, Insbridge ESI only supports US English.

### *Insbridge ESI Clients:*

- Operating System – Windows 7 or later
- JDK 7
- Unzip utility for using ESI system configuration

## Manual History

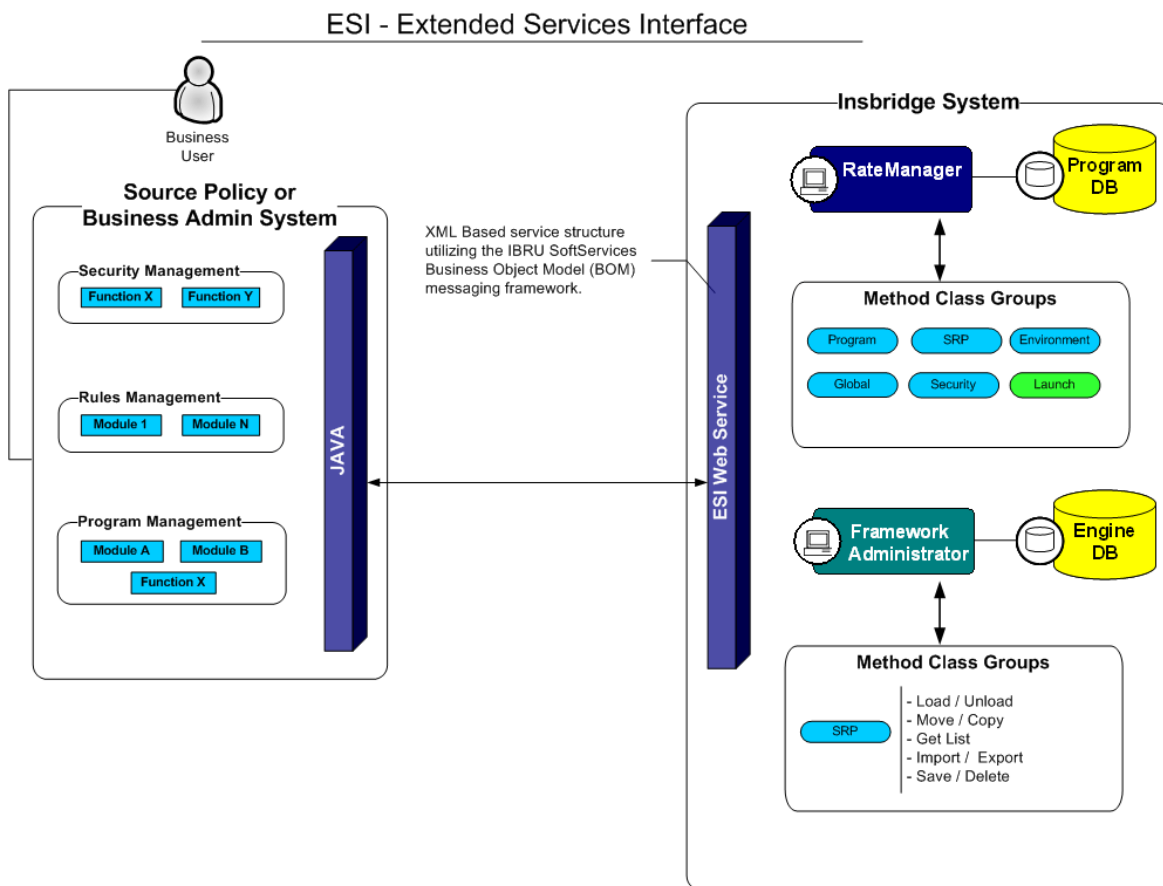
New editions incorporate any updates issued since the previous edition.

<b>Edition</b>	<b>Publication Number</b>	<b>Product Version</b>	<b>Publication Date</b>	<b>Comment</b>
1 <sup>st</sup> Edition	P01-783-01	R 4.1	December 2010	Initial release
2 <sup>nd</sup> Edition	P01-783-02	R 4.5	May 2011	Update
3 <sup>rd</sup> Edition	P01-783-03	R 4.5.1	September 2011	Update
4 <sup>th</sup> Edition	P01-783-04	R 4.6	May 2012	Update
5 <sup>th</sup> Edition	P01-783-05	R 4.6.1	November 2012	Update Release
6 <sup>th</sup> Edition	P01-783-06	R 4.7	September 2013	Update Release
7 <sup>th</sup> Edition	P01-783-07	R 4.7.1	November 2013	Update
8 <sup>th</sup> Edition	P01-783-08	R 4.8	August 2014	Update
9 <sup>th</sup> Edition	P01-783-09	R 4.9	December 2014	Update
10 <sup>th</sup> Edition	P01-783-10	R 5.1	December 2015	Update
11 <sup>th</sup> Edition	P01-783-11	R 5.2	July 2016	Update

## INTRODUCTION TO INSBRIDGE EXTENDED SERVICES INTERFACE

Oracle Insurance Insbridge Enterprise Rating Extended Services Interface (ESI) is a library module designed to provide remote services to the Insbridge Enterprise Rating (Insbridge) business services and Insbridge Soft Service (IBSS) without directly utilizing the system User Interfaces (UI).

### ESI for Insbridge



ESI allows users to pass information between a user's source policy or business admin system and the RateManager system or the Insbridge Framework Administrator (IBFA) without using either the RateManager or IBFA UI. ESI itself is not a web service. You reference it in your applications and **consume it natively**. ESI then calls the web services in RateManager or IBFA. Both RateManager and IBFA have a web service interface that ESI can call.

ESI is designed to operate on Insbridge version 04.01.00 or higher and connects with:

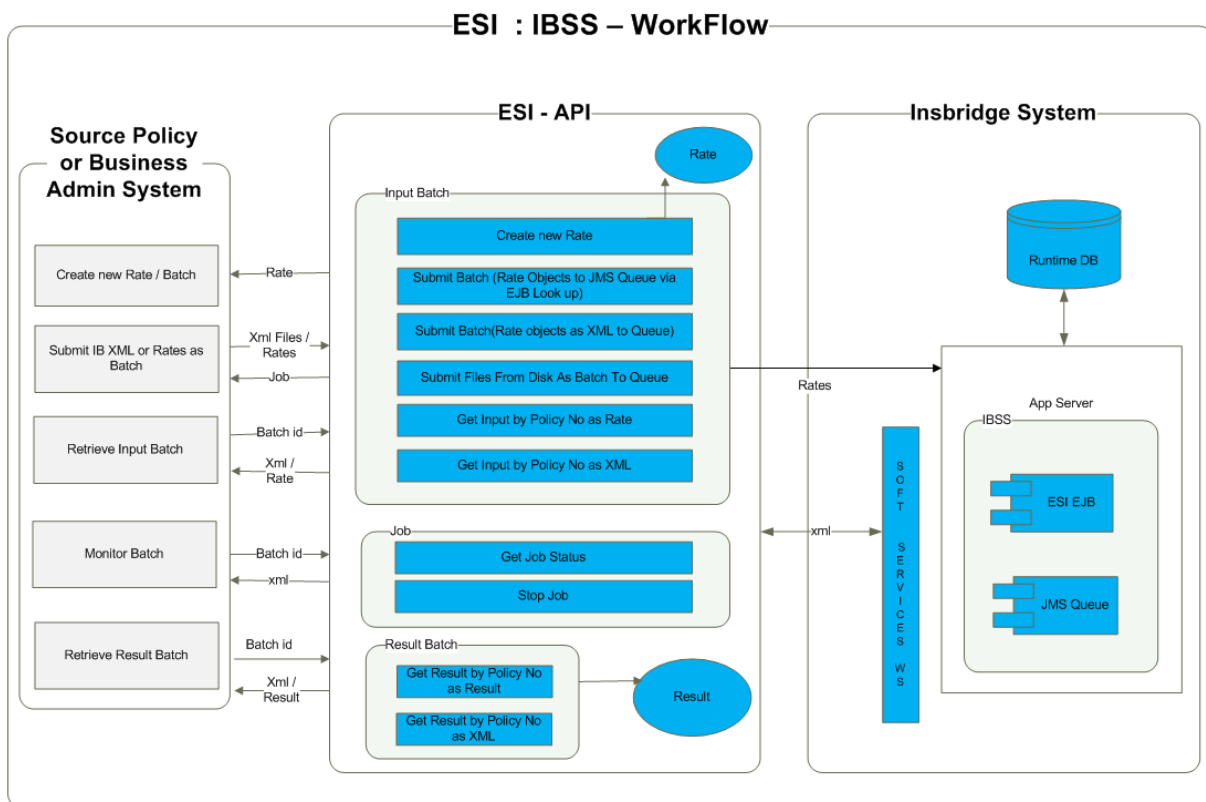
- Oracle Insurance RateManager Interface
- Oracle Insurance Framework Administrator Interface

For example, if you had a vendor Policy Administration system and wanted to provide integrated product management through your custom solution; ESI would allow you to model and use the workflow on the Insbridge business services and functions but control all the user presentation aspects through your custom product interface.

### Rule and Functionality Highlights

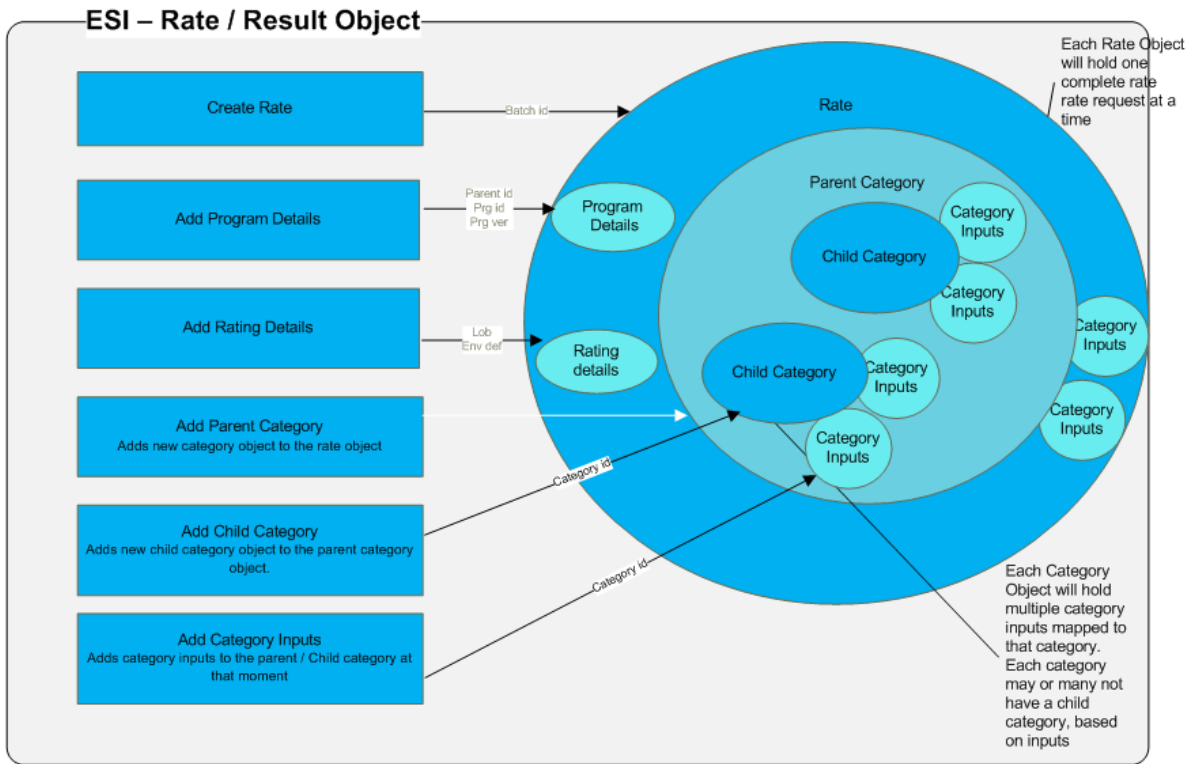
- ESI is available on Insbridge Enterprise Rating version 04.01.00 or higher.
- ESI is shipped with the IBFA installation. To locate ESI, look under the Insbridge directory for the SDK/ESI/Java file; [INSTALL\_DIR]\Insbridge\SDK\ESI\Java. This file contains five items:
  - InsbridgeEsiServices.jar
  - EsiSwingTester.cmd
  - EsiSwingTester.jar
  - EsiSwingTester Source Code.zip
  - Javadoc.zip
- ESI entries are immediate. There is no lag time or system sync required.
- An ESI cannot be recalled or undone. If an ESI needs to be edited, for example a folder has an incorrect name; you must enter RateManager or IBFA and make the changes there.
- ESI works with the default subline in the folder of your choice for any line of business.
- Unless specifically stated as Optional, all Inputs are **Required**.

### ESI Work Flow for IBSS



**NOTE:** For further details, please see the Javadoc file.

**Rate Model ESI: IBSS**



**Functionality Highlights**

- A rate request now can be an Object (Rate) with rate request details or an Insbridge XML.
- With the ESI API, you can submit an InputBatch as a collection of rate objects or Insbridge XML, to the runtime database of IBSS.
- Using the InputBatch, input rates can be retrieved either as an XML or as a rate object.
- Using the Result Batch, the result of a rate can be retrieved either as an XML or as a result object.
- When an InputBatch is submitted for execution, a Job is returned that can be used to get the status of the Job, or stop the Job even if it is being executed.
- The status can be retrieved either as an XML or as a status object.



# USING ESI: IBSS - EXAMPLE

## Creating a Rate Object

ESI has a Rate Object that can be used to create a rate request. A Rate Object is the representation of a rate request. A collection of rate requests is an InputBatch that can be submitted to the runtime database for execution as a batch.

## IBRequestResponse Interface

Rate implements the IBRequestResponse interface. You can use the interface to create rate objects to add to the runtime database as a batch for execution.

```
public interface IBRequestResponse {
    public void addRatingDetails(String lob, String policyNumber, String env_def);
    public void addProgramDetails(String parent_id, String tiering_id, String program_id, String
    program_ver);
    public int addParentCategory(String catID, String desc);
    public int addChildCategory(String catID, String desc);
    public void addCategoryInputs(String id, String name, String value);
    public boolean clear() ;
    public int getParentCategorySize();
    public HashMap getParentCategoryMap();
    public void setCurrentCategory(CategoryVO currentParent);
    public CategoryVO getCurrentCategory();
    public void setRateDetails(RateVO rateDetails);
    public RateVO getRateDetails();
    public void setProgramDetails(ProgramVO programDetails);
    public ProgramVO getProgramDetails();
}
```

## Creating a Rate Object

**Step 1:** Create an InputBatch.

**Step 2:** Use the createNewRate() method to create a new rate request object.

**Step 3:** Use the addRatingDetails() method to add rating details to the rate.

**Step 4:** Use the `addProgramDetails()` method to add program details to the rate.

**Step 5:** Use the `addParentCategory` method to add a parent category to the rate.

A parent category is always at the top level. Child categories can be added underneath the parent category.

**Step 6:** Use the `addCategoryInputs()` method to add category inputs to the category created in Step 5.

The `addCategoryInputs()` method adds category inputs to the current category defined in Step 5.

**Step 7:** Use the `addChildCategory()` method to add a child category to the parentCategory or current category.

**Step 8:** Use the `addCategoryInputs()` method to add category inputs to the category.

**Step 9:** Use the `setCurrentCategory()` method to set the current category where you want to work adding a child category or category inputs.

### Example

```
rateRequest.setCurrentCategory((rateRequest.getCurrentCategory().getParentCategory()).getParentCategory());
```

### Sample Code for Creating a Rate Object

```
public class SampleRateObject {
    public static void main(String[] args) {

        InputBatch inputBatch = new InputBatch();
        Rate rateRequest = inputBatch.createNewRate();

        //create rate and program objects for the batch

        rateRequest.addRatingDetails("111", "", "rm");
        rateRequest.addProgramDetails("775", "", "2", "1");

        //Start creating new category objects by adding new Parent category

        rateRequest.addParentCategory("12", "vehicle");

        //add category inputs for the Parent category

        rateRequest.addCategoryInputs("20", "CompanyCd", "Alamere");
        rateRequest.addCategoryInputs("21", "UseLowerFalseClaimFactorInd", "");
        rateRequest.addCategoryInputs("24", "GroupCd", "");
        rateRequest.addCategoryInputs("46", "EffectiveDt", "03/17/2015");

        // adding child category

        rateRequest.addChildCategory("12_1", "vehicle_1_CAR");
        rateRequest.addCategoryInputs("12_1", "CompanyCd", "gershome");
        rateRequest.addCategoryInputs("12_1", "GroupCd", "gershome");
        rateRequest.addCategoryInputs("12_1", "EffectiveDt", "03/17/2015");
        rateRequest.addCategoryInputs("12_1", "UseLowerFalseClaimFactorInd", "");
    }
}
```

```

rateRequest.addChildCategory("12_1_1", "vehicle_1_CAR_1");
rateRequest.addCategoryInputs("12_1_1", "CompanyCd", "gershome");
rateRequest.addCategoryInputs("12_1_1", "GroupCd", "gershome");
rateRequest.addCategoryInputs("12_1_1", "EffectiveDt", "03/17/2015");
rateRequest.addCategoryInputs("12_1_1", "UseLowerFalseClaimFactorInd", "");

// How to change from one Parent Category to another

rateRequest.setCurrentCategory(rateRequest.getCurrentCategory().getParentCategory());

rateRequest.addChildCategory("12_1_2", "vehicle_1_1CAR_2");
rateRequest.addCategoryInputs("12_1_2", "CompanyCd", "gershome");
rateRequest.addCategoryInputs("12_1_2", "GroupCd", "gershome");
rateRequest.addCategoryInputs("12_1_2", "EffectiveDt", "03/17/2015");
rateRequest.addCategoryInputs("12_1_2", "UseLowerFalseClaimFactorInd", "");

// Sample setting up parent category for new Child categories at different levels

rateRequest.setCurrentCategory((rateRequest.getCurrentCategory().getParentCategory()).getParentCategory());
rateRequest.addChildCategory("12_2", "vehicle_1_bus");
rateRequest.addCategoryInputs("12_2", "CompanyCd", "gershome1");
rateRequest.addCategoryInputs("12_2", "CompanyCd", "gershome2");

// adds a parent category at top level

rateRequest.addParentCategory("13", "Driver");
rateRequest.addCategoryInputs("20", "CompanyCd", "Alamere");
rateRequest.addCategoryInputs("21", "UseLowerFalseClaimFactorInd", "");
rateRequest.addCategoryInputs("24", "GroupCd", "");
rateRequest.addCategoryInputs("46", "EffectiveDt", "03/17/2015");
}}

```

---

**NOTE:** Once moved up from a child category to its parent category, system cannot go down to the child category.

---

## Input Batch – Creating, Submitting, Executing and Monitoring

### Creating an InputBatch

- An input batch is created using InputBatch.
- Create rates for the InputBatch using the createNewRate() method.
- Rates created with the createNewRate method are added to the InputBatch using the addRateToBatch() method.



## Submitting an InputBatch

- Once all the rates have been added to the InputBatch, the InputBatch is added to the runtime database using the addInputBatchToRuntimeDB() method.

## Executing an InputBatch

- An InputBatch in the runtime database can be executed to rate against the engine from ESI using the executeBatch() method.
- When an InputBatch is executed, a Job is returned to the user.

## Monitoring a Batch

When the Job is returned from the executing of the InputBatch, you can monitor the batch.

Monitoring includes:

- Get the status of the InputBatch
  - getJobStatusAsXML to get the status as XML
  - getJobStatus() to get the status as status object (StatusVO)
- Stop the InputBatch (Job) using stopJob() method, even if the Job is currently being executed.

## Retrieving Inputs

Input rates can be retrieved from the runtime database in two ways.

- Retrieve input rate by policy number as a rate object using the getInputRateByPolicyNumber() method.
- Retrieve input rate by policy number as XML using the getInputXMLByPolicyNumber() method.

## Result Batch

### Retrieving Results

Result can be retrieved from the runtime database in two ways.

- Retrieve result by policy number as a result object using the getResultByPolicyNumber () method.
- Retrieve result by policy number as XML using the getResultXMLByPolicyNumber () method.
- Once results have been retrieved, they can be added to the ResultBatch using the addResultToResultBatch() method.

## **BASE DOMAIN TYPES**

Base ESI domain types represent the business nouns within the Insbridge system. These data structures represent the majority of the business payload.

Please see the Javadoc file for details.

## USE CASE SUMMARY

The Use Case Summary table is a quick guide to the function available for ESI for Insbridge.

Business Use Case	Methods to Use	Result
<b>General – Open URL</b>		
Create Launch Definition – using a Network login	<u>General – createLauchDefinition (Selected Version, Network Login)</u>	Method returns a URL for a selected program version using a network login.
Create Launch Definition – using a Network login	<u>General – createLauchDefinition (First Version, Network Login)</u>	Method returns a URL for the first program version using a network login.
Create Launch Definition – using a RateManager login	<u>General – createLauchDefinition (Selected Version, RateManager Login)</u>	Method returns a URL for a selected program version using a RateManager login.
Create Launch Definition – using a RateManager login	<u>General – createLauchDefinition (First Version, RateManager Login)</u>	Method returns a URL for the first program version using a RateManager login.
<b>Programs</b>		
Add	<u>Program Items – createProgram</u>	Create a new program.
Add New Version	<u>Program Items – createProgramVersion</u>	Creates a new version of an existing program.
Copy a Program	<u>Program Items – copyProgram</u>	Copies an existing Insbridge program.
Delete	<u>Program Items – deleteProgram</u>	Removes a program.
Update	<u>Program Items – updateProgram</u>	Updates a program.
Update	<u>Program Items – updateProgramVersionLock</u>	Updates a program version lock.

<b>Folder Groups</b>		
Add	<u>Program Items – createFolder</u>	Creates a new folder.
Delete	<u>Program Items – deleteFolder</u>	Removes a folder.
List Folders	<u>Program Items – getFoldersXML</u>	Returns an XML string of all available folders in an LOB.
Update	<u>Program Items – updateFolder</u>	Updates the name or location of a folder.
<b>Line of Business</b>		
Add	<u>Program Items – createLine</u>	Creates a new Line.
Delete	<u>Program Items – deleteLine</u>	Removes an Line.
Update	<u>Program Items – updateLine</u>	Updates an Line.
<b>SRP – RateManager</b>		
Add	<u>SRP Items – createSRP</u>	Creates an SRP and returns the SRP object.
<b>Global Items</b>		
Add	<u>Global Items – createInput</u>	Creates a new input.
Add	<u>Global Items – createCategory</u>	Creates a new category.
Add	<u>Global Items – createCategory with Inputs</u>	Creates a new category with inputs.
Delete	<u>Global Items – deleteInput</u>	Removes an input.
Delete	<u>Global Items – deleteCategory</u>	Removes a category.
Updates	<u>Global Items – updateInput</u>	Updates a category.
Updates	<u>Global Items – updateCategory</u>	Updates an input.
List Inputs	<u>Global Items – getInputsArray</u>	Returns an array of all inputs in a Line.
List Inputs	<u>Global Items – getCategoryXML</u>	Returns an XML string of all categories in a Line.
<b>Security Items</b>		
Add	<u>Security Items – createUser</u>	Creates a new user.
List	<u>Security Items – getUsers</u>	Returns a list of all available users for the subscriber.
List	<u>Security Items – getUserGroups</u>	Returns a list of all available user groups.
Login	<u>Security Items – login</u>	Logs in the user.

Login	<u>Security Items – networkLogin</u>	Logs in the user with network login.
Logout	<u>Security Items – logout</u>	Logs out the user.
Update	<u>Security Items – updateUser</u>	Updates a user's information.
<b>Environment Items</b>		
List	<u>Environment Items – getAvailableEnvironments, All Non-Secured</u>	Returns a list of all available non-secured environments.
List	<u>Environment Items – getAvailableEnvironments, Secured</u>	Returns a list of all available environments within a selected environment type.
<b>Releases</b>		
Add	<u>Releases Items – createRelease</u>	Creates a new release.
Delete	<u>Releases Items – deleteRelease</u>	Removes a release.
List	<u>Releases Items – getReleasePrograms</u>	Returns a list of all programs in release.
Update	<u>Releases Items – addReleaseProgram</u>	Add a program version to a release.
Delete	<u>Releases Items – removeReleaseProgram</u>	Removes a program from the release.
<b>SRP– IBFA</b>		
Copy	<u>SRP Items – copySRP</u>	Copies an SRP from the current environment into a selected environment.
Delete	<u>SRP Items – deleteSRP</u>	Deletes an SRP from the disk.
Export	<u>SRP Items – exportSRP</u>	Exports an SRP from IBFA and converts it to bytes.
Import	<u>SRP Items – importSRP</u>	Imports SRP bytes into an IBFA environment.
List	<u>SRP Items – getSRPList</u>	Returns a list of all SRPs in a select environment.
List	<u>SRP Items – getBytesFromFile</u>	Returns the bytes from a file.
Load	<u>SRP Items – loadSRP</u>	Loads an SRP into an IBFA environment.
Move	<u>SRP Items – moveSRP</u>	Moves an SRP from the current environment to the passed environment.
Save	<u>SRP Items – saveFile</u>	Saves the bytes you pass into a physical file.
Unload	<u>SRP Items – unloadSRP</u>	Unloads an SRP from a selected environment.

**The new ESI features added in 5.1.0 are:**

Add/update program version name

Add/update program versioning data (Delete, Update, Insert with multiple rows)

Add/update notes to a program version (Retrieve all, Add, Update, Delete)

Add/update notes to a table revision (Retrieve all, Add, Update, Delete)

Define a new table revision

Export tables in the tab-delimited tab format

## QUICK GUIDE TABLE

The quick guide table shows the functions available for ESI for IBSS.

Business Use Case	Methods to Use	Result
<b>Input Batch</b>		
Create new rate objects	createNewRate	Creates a new rate object that can be added to an InputBatch.
Add batch to DB	addInputBatchToRuntimeDB	Once all the rates have been added to the InputBatch, the batch is added to the runtime DB.
Execute the batch submitted to Runtime DB	ExecuteBatch	After an InputBatch is added to the runtime DB, it is executed (rated) against the engine.
Retrieve input rate	getInputRateByPolicyNumber	The input rate request is retrieved from the runtime DB either as an XML or as a rate object.
Submit Insbridge XML files from disk as batch	submitIBFilesFromDiskAsBatchToQueue	A collection of XML rate files in the disk is submitted as an InputBatch.
Submit Insbridge XML from memory	submitIBXMLFromMemoryToDB	A single XML rate from memory is submitted to the runtime DB as a batch or added to a batch.
Submit single rate object to database	submitRateObjectToDB	A single rate object is submitted to the runtime DB as a batch.
<b>Result Batch</b>		
Create new result object	createNewResult	Creates a new result object.
Get result by policy number	getResultByPolicyNumber	Returns a result either as an XML or a result object.
<b>Job</b>		
Get the status of the job	getJobStatus	Returns the status of the job submitted for execution.
Stop the job	stopJob	Stops the submitted job.

Rate / Result		
Add rating details to rate	addRatingDetails	Adds rating details to the rate.
Add program details to rate	addProgramDetails	Adds program details to the rate.
Add parent category to rate	addParentCategory	Adds parent category details to the rate.
Add child category to rate	addChildCategory	Adds child category details to the rate.
Add category inputs to category	addCategoryInputs	Adds category inputs to the category.



## RATEMANAGER INTERFACE

### Constructor Detail

#### EsiRateManager

```
public EsiRateManager(int aSubscriber)
    throws java.lang.Exception
```

Constructor for EsiRateManager using LOCALHOST as the target server

**Parameters:**

aSubscriber - the licensed subscriber ID

**Throws:**

java.lang.Exception

---

#### EsiRateManager

```
public EsiRateManager(int aSubscriber,
    java.lang.String aTargetServer)
    throws java.lang.Exception
```

Constructor for EsiRateManager

**Parameters:**

aSubscriber - the licensed subscriber ID

aTargetServer - target server name or IP address

**Throws:**

java.lang.Exception

---

## Environment Items – getAvailableEnvironments, All Non-Secured

- **Business Functionality**
  - Gets all the available non-secured environments.
  - This ESI interfaces provides the ability for an external system to programmatically retrieve meta-data about all the available non-secured Environments in order for the system to programmatically invoke the other ESI deployment related API's.
- **Inputs**
  - No inputs
- **Outputs**
  - List of all non-secured EsiEnvironment objects.

## Environment Items – getAvailableEnvironments, Secured

- **Business Functionality**
  - Gets all the available environments for a particular environment type.
  - This ESI interfaces provides the ability for an external system to programmatically retrieve meta-data about the available Environments in order for the system to programmatically invoke the other ESI deployment related API's and pass in the correct environment data object.
- **Inputs**
  - aType – the environment type
  - isIncludeSecure – Include Secured or Not
- **Outputs**
  - List of ESIEEnvironment objects.

## General – createLauchDefinition (Selected Version, Network Login)

- **Business Functionality**
  - Method creates a URL of an existing program version. This method uses the Network login for authentication. Requires custom authentication setup in RateManager Security. See Authentication.
- **Inputs**
  - aLine – Line ID
  - aProgram – Program ID
  - aProgramVersion – Program Version
- **Outputs**
  - Returns a URL string to launch a Program.

## General – createLauchDefinition (First Version, Network Login)

- **Business Functionality**
  - Method creates a URL of the first program version. This method uses the Network login for authentication. Requires custom authentication setup in RateManager Security. See Authentication.
- **Inputs**
  - aLine – Line ID
  - aProgram – Program ID
- **Outputs**
  - Returns a URL string to launch a Program.

## General – createLauchDefinition (Selected Version, RateManager Login)

- **Business Functionality**
  - Method creates a URL of an existing program version. This method uses the standard RateManager login for authentication.
- **Inputs**
  - aUser – non-network user ID
  - aLine – Line ID

- aProgram – Program ID
- aProgramVersion – Program Version
- **Outputs**
  - Returns a URL string to launch a Program.

### General – createLaunchDefinition (First Version, RateManager Login)

- **Business Functionality**
  - Method creates a URL of the first program version. This method uses the standard RateManager login for authentication.
- **Inputs**
  - aUser – non-network user ID
  - aLine – Line ID
  - aProgram – Program ID
- **Outputs**
  - Returns a URL string to launch a Program.

### Global Items – createInput

- **Business Functionality**
  - Creates a new global level input in a particular Line.
- **Inputs**
  - EsilInput
- **Outputs**
  - Returns the ID of the newly created Input if succeeded or exception if failed.

### Global Items – updateInput

- **Business Functionality**
  - Updates a global level input in a particular Line.
- **Inputs**
  - EsilInput
- **Outputs**
  - True if successful or false if failed.

### Global Items – deleteInput

- **Business Functionality**
  - Deletes a global level input in a particular Line.
- **Inputs**
  - EsilInput
- **Outputs**
  - True if successful or false if failed.

## Global Items – getInputsArray

- **Business Functionality**
  - Lists all the inputs of a particular Line.
- **Inputs**
  - EsiInput
- **Outputs**
  - Returns an array of EsiInputs that includes all the inputs of that Line.

## Global Items – createCategory

- **Business Functionality**
  - Creates a category in a particular Line.
- **Inputs**
  - EsiCreateCategory
- **Outputs**
  - Returns the ID of the newly created Category if succeeded or exception if failed.

## Global Items – createCategory with Inputs

- **Business Functionality**
  - Creates a category with inputs in a particular Line.
- **Inputs**
  - EsiCategory
  - List of EsiInput objects
- **Outputs**
  - No outputs.

## Global Items – getCategoryXML

- **Business Functionality**
  - Gets an XML string of all the categories in a particular Line.
- **Inputs**
  - getCategoryXML
- **Outputs**
  - Return an XML string.

## Global Items – updateCategory

- **Business Functionality**
  - Updates a category in a particular Line.
- **Inputs**
  - EsiCategory
- **Outputs**
  - True if successful exception if failed.

## Global Items – deleteCategory

- **Business Functionality**
  - Deletes a category in a particular Line.
- **Inputs**
  - EsiCategory
- **Outputs**
  - True if successful exception if failed.

## Program Items – createLine

- **Business Functionality**
  - Creates a new line of business. The line of business, if activated, is displayed on the RateManager navigation bar immediately. In order for any users to use the new line, groups must have access privileges granted.
- **Inputs**
  - EsiLine
- **Outputs**
  - Returns the ID of the newly created Line if succeeded or -1 if failed.

## Program Items – updateLine

- **Business Functionality**
  - Updates a line of business.
- **Inputs**
  - EsiLine
- **Outputs**
  - True if successful or exception if failed.

## Program Items – deleteLine

- **Business Functionality**
  - Deletes a line of business.
- **Inputs**
  - EsiLine
- **Outputs**
  - True if successful or exception if failed.

## Program Items – createFolder

- **Business Functionality**
  - Creates a new folder in a particular Line.
- **Inputs**
  - EsiFolder
- **Outputs**
  - Returns an EsiFolder object.

## Program Items – updateFolder

- **Business Functionality**
  - Updates the name or parent of an existing program folder. When you change the parent of any folder, the folders underneath are changed as well.
- **Inputs**
  - EsiFolder
- **Outputs**
  - Returns the updated Folder ID if succeeded or exception if failed.

## Program Items – deleteFolder

- **Business Functionality**
  - Deletes an existing program folder. Make absolutely sure that the programs for the folder are no longer needed. Deleting a folder deletes all programs under that folder only. Subfolders and any programs in the subfolders are not deleted but will now be unlinked.
- **Inputs**
  - EsiFolder
- **Outputs**
  - Returns True if succeeded or exception if failed.

## Program Items – getFoldersXML

- **Business Functionality**
  - Gets an XML string of available Folders of a particular Line.
- **Inputs**
  - getFoldersXML
- **Outputs**
  - Returns an XML string of available Folders of a specific Line.

## Program Items – createProgram

- **Business Functionality**
  - Creates a new Insbridge Program in the folder of your choosing.
- **Inputs**
  - EsiProgram
- **Outputs**
  - EsiProgram object.

## Program Items – updateProgram

- **Business Functionality**
  - Updates a Program in the folder of your choosing.
- **Inputs**
  - EsiProgram

- **Outputs**
  - True if successful or exception if failed.

## Program Items – deleteProgram

- **Business Functionality**
  - Deletes an existing Insbridge Program. When a program is deleted, all data (variables, algorithms, sequencing, result groups and, if an auto program, driver assignment scenarios) and versions are deleted. Make sure this is the action you want to take. Locked programs cannot be deleted and returns a fail.
- **Inputs**
  - EsiProgram
- **Outputs**
  - True if successful or exception if failed.

## Program Items – createProgramVersion

- **Business Functionality**
  - Creates a new version of an existing program.
- **Inputs**
  - EsiProgramVersion
- **Outputs**
  - Returns an EsiProgramVersion object.

## Program Items – copyProgram

- **Business Functionality**
  - Copies an existing Insbridge program version to a new specified program version with the option to copy the definition only or copy the definition and all data.
    - **Copy Definitions Only:** This option copies all elements of the program with the exception of the data contained in mapped variables.
    - **Copy Definitions & All Data:** This option copies all elements of the program, including the data contained in mapped variables.
- **Inputs**
  - New Program Version
  - Source Program Version
  - Copy Option
- **Outputs**
  - Returns an EsiProgram object.

## Program Items – updateProgramVersionLock

- **Business Functionality**
  - Locks or unlocks a program version.
- **Inputs**
  - EsiProgramVersion

- aLockOption
- **Outputs**
  - True if successful or exception if failed.

## Releases Items – createRelease

- **Business Functionality**
  - Creates a new Release.
- **Inputs**
  - aReleaseName – the release name
  - aReleaseDescription – the release description
  - aReleaseNotes – The release notes
  - aAdminEmail – the release admin email
  - aTargetEnvironment – the environment
- **Outputs**
  - EsiRelease object.

## Releases Items – deleteRelease

- **Business Functionality**
  - Deletes an existing release.
- **Inputs**
  - aTargetRelease – the release being deleted
- **Outputs**
  - True if successful or exception if failed.

## Releases Items – addReleaseProgram

- **Business Functionality**
  - Add a program version to a Release.
- **Inputs**
  - aTargetRelease – the release the program version is being added to
  - aTargetProgramVersion – the program version that is being added
- **Outputs**
  - True if successful or exception if failed.

## Releases Items – removeReleaseProgram

- **Business Functionality**
  - Removes a program version from a Release.
- **Inputs**
  - aTargetRelease – the release the program version is being removed from
  - aTargetProgramVersion – the program version that is being removed
- **Outputs**
  - True if successful or exception if failed.



## Releases Items – getReleasePrograms

- **Business Functionality**
  - Lists the programs in a Release.
- **Inputs**
  - aTargetRelease – the EsiRelease object being retrieved
- **Outputs**
  - The EsiRelease object including the programs contained.

## Security Items – login

- **Business Functionality**
  - Logs a User in to the RateManager System.
- **Inputs**
  - none
- **Outputs**
  - none

## Security Items – networkLogin

- **Business Functionality**
  - Logs a User in to the RateManager System using a Windows login.
- **Inputs**
  - none
- **Outputs**
  - none

## Security Items – logout

- **Business Functionality**
  - Logs a User out of the RateManager System.
- **Inputs**
  - none
- **Outputs**
  - none

## Security Items – createUser

- **Business Functionality**
  - Creates a new User.
- **Inputs**
  - EsiUser
- **Outputs**
  - Returns new user ID if successful or exception if failed.

## Security Items – updateUser

- **Business Functionality**
  - Updates an existing User. Password Reset must be done inside RM.
- **Inputs**
  - EsiUse
- **Outputs**
  - Returns True if succeeded or False if failed.

## Security Items – getUsers

- **Business Functionality**
  - Gets a list of all the available users for the subscriber.
- **Inputs**
  - No inputs
- **Outputs**
  - Returns a list of EsiUsers.

## Security Items – getUserGroups

- **Business Functionality**
  - Gets all the available user groups.
- **Inputs**
  - No inputs
- **Outputs**
  - Returns a list of EsiUserGroup.

## SRP Items – createSRP

- **Business Functionality**
  - Creates a SoftRater Package (SRP) in a particular environment.
  - ESI created SRP has the identifier “API” in the filename and shows an icon on IBFA SRP list.
- **Inputs**
  - aTargetProgram – the program the SRP to be created on
  - aProgram Version – the program version the SRP to be created on
  - aTargetEnvironment – the environment the SRP to be created on
- **Outputs**
  - Returns the newly created EsiSRP object.

## FRAMEWORK ADMINISTRATOR INTERFACE

### Constructor Detail

#### EsiFrameworkAdministrator

```
public EsiFrameworkAdministrator(java.lang.String aApplicationTokenKey,  
                                int aSubscriber)  
    throws java.lang.Exception
```

Constructor for FrameworkAdministrator using LOCALHOST as the target server

**Parameters:**

aApplicationTokenKey - the Secure token to validate ESI session  
aSubscriber - the Licenced Subscriber ID

**Throws:**

java.lang.Exception

---

#### EsiFrameworkAdministrator

```
public EsiFrameworkAdministrator(java.lang.String aApplicationTokenKey,  
                                int aSubscriber,  
                                java.lang.String aTargetServer)  
    throws java.lang.Exception
```

FrameworkAdministrator Constructor

**Parameters:**

aApplicationTokenKey - the Secure token to validate ESI session  
aSubscriber - the Licenced Subscriber ID  
aTargetServer - the Target server name or IP address

**Throws:**

java.lang.Exception

### SRP Items – copySRP

- **Business Functionality**
  - Copies an SRP from the current environment to the selected environment. To determine the current environment, run the GetSRPList. This returns a list of Esi\_SRP's. Every Esi\_SRP has the environment and other information. CopySRP gets the current environment from the Esi\_SRP object that you pass in.
- **Inputs**

- aTargetEnvironment – The environment the SRP to be copied to.
- aTargetSRP – The SRP to be copied.
- **Outputs**
  - True if succeeded or exception if failed.

## SRP Items – deleteSRP

- **Business Functionality**
  - Deletes an SRP from disk. To determine (or control) the environment that the SRP is deleted from, run the GetSRPList to get the Esi\_SRP's. Every Esi\_SRP has the environment and other information. This tells you which environment the SRP is currently in.
  - Once deleted, this package cannot be reloaded.
- **Inputs**
  - aTargetSRP – The SRP to be deleted
- **Outputs**
  - True if succeeded or exception if failed

## SRP Items – exportSRP

- **Business Functionality**
  - Exports an SRP from an IBFA environment and converts it to bytes.
  - You can use the bytes directly or save them to a physical file by using SaveFile() method.
  - To use the SRP bytes, use a straight binary read into a byte array.

```
FileStream fs = File.OpenRead(@"C:\TEMP\abc.srp");
byte[] srpBytes = new byte[fs.Length];
fs.Read(srpBytes, 0, srpBytes.Length);
```
- **Inputs**
  - aTargetSRP – The SRP to be exported
- **Outputs**
  - The bytes of the SRP

## SRP Items – getSRPList

- **Business Functionality**
  - Gets a list of all SRPs in a particular environment.
- **Inputs**
  - aTargetEnvironment – The environment from where the SRPs are to be retrieved.
  - aSRPType – theSRP type, 0=all SRPs; 1=ESI created SRPs
- **Outputs**
  - A list of EsiSRP objects.

## SRP Items – getBytesFromFile

- **Business Functionality**
  - Gets the bytes from the file you pass in.

- **Inputs**
  - aFile – the file to get bytes from.
- **Outputs**
  - The bytes of the file.

### SRP Items – importSRP

- **Business Functionality**
  - Imports SRP bytes into an IBFA environment.
- **Inputs**
  - aTargetEnvironment – The environment the SRP is to be imported into
  - aSRPBytes – The SRP bytes to be imported
  - aSRPFileName – The SRP file name to be saved as
- **Outputs**
  - True if succeeded or exception if failed

### SRP Items – loadSRP

- **Business Functionality**
  - Loads an SRP into an IBFA environment so that the rating algorithms can be executed against it.
  - Loading a package takes the logic stored in the package and places it in the database so it can be executed.
  - Packages should be able to be loaded within an ESI Environment.
- **Inputs**
  - aTargetEnvironment – The environment the SRP is to be loaded in.
  - aTargetSRP – The SRP to be loaded.
- **Outputs**
  - True if succeeded or exception if failed.

### SRP Items – moveSRP

- **Business Functionality**
  - Given an Esi\_SRP object as well as an ESI Environment, the API moves the SRP from the current environment to the passed-in environment.
- **Inputs**
  - aTargetEnvironment – The environment the SRP is to be moved to.
  - aTargetSRP – The SRP to be moved.
- **Outputs**
  - True if succeeded or exception if failed.

### SRP Items – saveFile

- **Business Functionality**
  - Saves the bytes you pass in to a physical file.
- **Inputs**

- aData – The bytes to be saved
- aPath – The location and name the bytes are to be saved
- **Outputs**
  - True if succeeded or exception if failed

## SRP Items – unloadSRP

- **Business Functionality**
  - Unloads an SRP from a particular Environment. To determine (or control) the environment that the SRP is unloaded from, run the GetSRPList to get the Esi\_SRP's. Every Esi\_SRP has the environment and other information. This tells you which environment the SRP is currently in.
  - The package itself remains on disk until deleted. It is not necessary to unload a package before loading another package for the same program and version.
  - If a package is loaded for a program and version that already has a package loaded, the old package is unloaded before the new package is loaded.
- **Inputs**
  - aTargetSRP – The SRP to be unloaded.
- **Outputs**
  - True if succeed or exception if failed.

# INPUTBATCH

## Create New Rate

- **Business Functionality**
  - Creates new rate object to be added to an input batch. A rate object is the object representation of a rate request.
- **Inputs**
  - Rating details
  - Program details
  - Parent category
  - Child category
  - Category inputs
- **Outputs**
  - Rate Object

## Add Batch to Runtime DB

- **Business Functionality**
  - Once all the rates have been added to the InputBatch, it can be added to the runtime database where the input batch will be picked up and rated against the SoftRater engine and results will be stored back in the database.
- **Inputs**
  - Rates
- **Outputs**
  - Batch ID

## Execute Batch

- **Business Functionality**
  - When an InputBatch is added to the runtime database, it can be rated against the SoftRater engine.
- **Inputs**
  - Batch Id
- **Outputs**
  - Job

## Retrieve Input Rates

- **Business Functionality**
  - A rate request after being added to the runtime database can be retrieved by the policy number.
  - Rate request can be retrieved either as an XML or as a rate object.
- **Inputs**
  - Batch Id
  - Policy Number

- **Outputs**
  - XML or Rate Object

### Submit Insbridge XML from Memory

- **Business Functionality**
  - Insbridge XML from memory can be submitted as a batch to the runtime database.
- **Inputs**
  - Insbridge XML
- **Outputs**
  - Batch ID

### Submit Insbridge XML Files from Disk

- **Business Functionality**
  - Insbridge XML files from disk can be submitted as a batch to the runtime database.
- **Inputs**
  - Insbridge XML files disk path
- **Outputs**
  - Batch ID

### Submit Rate Object to Runtime DB

- **Business Functionality**
  - A rate object can be submitted as a batch to the runtime database.
- **Inputs**
  - Rate Object
- **Outputs**
  - Batch ID



## RESULT BATCH

### Create New Result

- **Business Functionality**
  - Creates new result object to be added to an InputBatch. A rate object is the object representation of a rate request.
- **Inputs**
  - Rating details
  - Program details
  - Parent category
  - Child category
  - Category inputs
- **Outputs**
  - Result Object

### Retrieve Results

- **Business Functionality**
  - A result can be retrieved by policy number.
  - A result can be retrieved either as an XML or as a result object.
- **Inputs**
  - Batch Id
  - Policy Number
- **Outputs**
  - XML or Result Object

## JOB

### Retrieve Job Status

- **Business Functionality**
  - Once a Job is submitted for execution, the status of the Job can be retrieved here.
- **Inputs**
  - Batch Id
- **Outputs**
  - Status XML or Status Object

### Stop the Job

- **Business Functionality**
  - Once a Job is submitted for execution, it can be stopped even if it is being executed.
- **Inputs**
  - Batch Id
- **Outputs**
  - status

## RATE OR RESULT

### Add Rating Details

- **Business Functionality**
  - Adds the rating details to the rate object.
- **Inputs**
  - LOB
  - Policy No
  - Environment definition

### Add Program Details

- **Business Functionality**
  - Adds the program details to the rate object
- **Inputs**
  - Parent ID
  - Program Id
  - Program Version

### Add Parent Category

- **Business Functionality**
  - Adds the parent category in the rate object.
- **Inputs**
  - Category Id
  - Category Description

### Add Child Category

- **Business Functionality**
  - Adds child category to the current parent category.
- **Inputs**
  - Category ID
  - Category Description

### Add Category Inputs

- **Business Functionality**
  - Adds the category inputs to the current category.
- **Inputs**
  - ID
  - Name
  - Value

# ESI USER SETUP AND AUTHENTICATION

There are two types of authentication in ESI RateManager Interface:

- **Regular login:** users send down a valid RateManager username and ESI logs the user in with that.
- **Network login:** ESI gets your windows credentials and logs you in with the same username and password as the user you login to windows

You must set up a network user account if you want to use windows integrated login in RateManager.

## Adding a User

New users can be added from the User Management screen. All new users are assigned the default password, password. New users should log in as soon as possible and change their password.

---

**NOTE:** *Prior to creating a user, make sure you have your user groups created.*

---

## To Add a New User

1. In RateManager, Tools→Security→**User Management**.
2. Click **NEW** to open the Add User tab.
3. Select the **Group(s)** the new user belongs to by selecting a group or groups from the Select Groups list and then clicking Select Arrow – Move Left. Upon saving, the user belongs to that group.
4. Select the **Company** the user belongs to from the drop down listing.

Subscriber information may be needed by users when logging into the system. If more than one subscriber was created, users must be directed to the subscriber where their logins were created. Subscriber information is necessary for custom XML creation and testing and is also needed for Libraries.

If you have not created subscribers in IBFA, you will not be able to assign users to a company.

5. Enter a **First** and **Last** name for the user in the appropriate text boxes.
6. Enter a **Username** for the new user. The username must be at least six, but fewer than twelve, alphanumeric characters and can include an underscore (–) or period (.).
7. Enter a **Department** for the user.
8. Select whether the new user is a **Network User** by selecting **True** or **False** from the drop down. If

you select **True**, the user's credentials (username and password) are validated using Windows Authentication Tokens. The username must match the network user ID. If you select **False**, the user must enter a valid username and password via the RateManager login screen.

9. Optionally, enter a **Phone Number** and **Email Address** for the new user.
10. If you want the user's password to expire, click the box next to **Password Expires**. The Preferences screen contains the number of days required for a password change.
11. When you are finished, click **SAVE**. The user is added to the system and the user list is updated. The new user can now log in using the username you assigned them and the default password, **password**.
12. The user should change their password the first time they log in. See Changing Your Password for more information.

The screenshot shows the 'Add User' form in the Oracle Insurance Insnbridge RateManager application. The form is titled 'Add User' and includes a warning message: 'You have selected to add/edit a user that will belong to a 'Group' that will control his/her rights for all Insnbridge applications and a 'Company' that offers high level grouping for security. All new users will have a default password that should be changed on a successful login for security reasons. The required information is noted with an asterisk.' The form fields are as follows:

- Groups:** Member Of: ADMIN; Select Groups: RateLoaders, QA, RateAnalysis, BA.
- Company:** alamere (dropdown menu).
- Firstname:** (text input).
- Lastname:** (text input).
- Username:** (text input).
- Department:** (text input).
- Network User:** False (dropdown menu).
- Phone:** (text input).
- Email:** (text input).
- Password Expires:** (checkbox, currently unchecked).

The 'Save' button is highlighted with a red box, and the 'Cancel' button is also visible.

Figure 1 Adding a New User

## ESI SYSTEM CONFIGURATION

To use ESI's IBFA Framework functionality, you need to send down the API key when you do the ESI API call.

When IBFA is installed, a default API key is generated and saved in IBFA. You can find it in *IBFA* → *Insbridge* → *Security* → *External API Access*. You can disable ESI's IBFA functionality by setting it to "Disabled". For security purpose, you may want to change the key once in a while. To change the key, click on "Generate Key" and then click on "Update". Every key generated is unique.

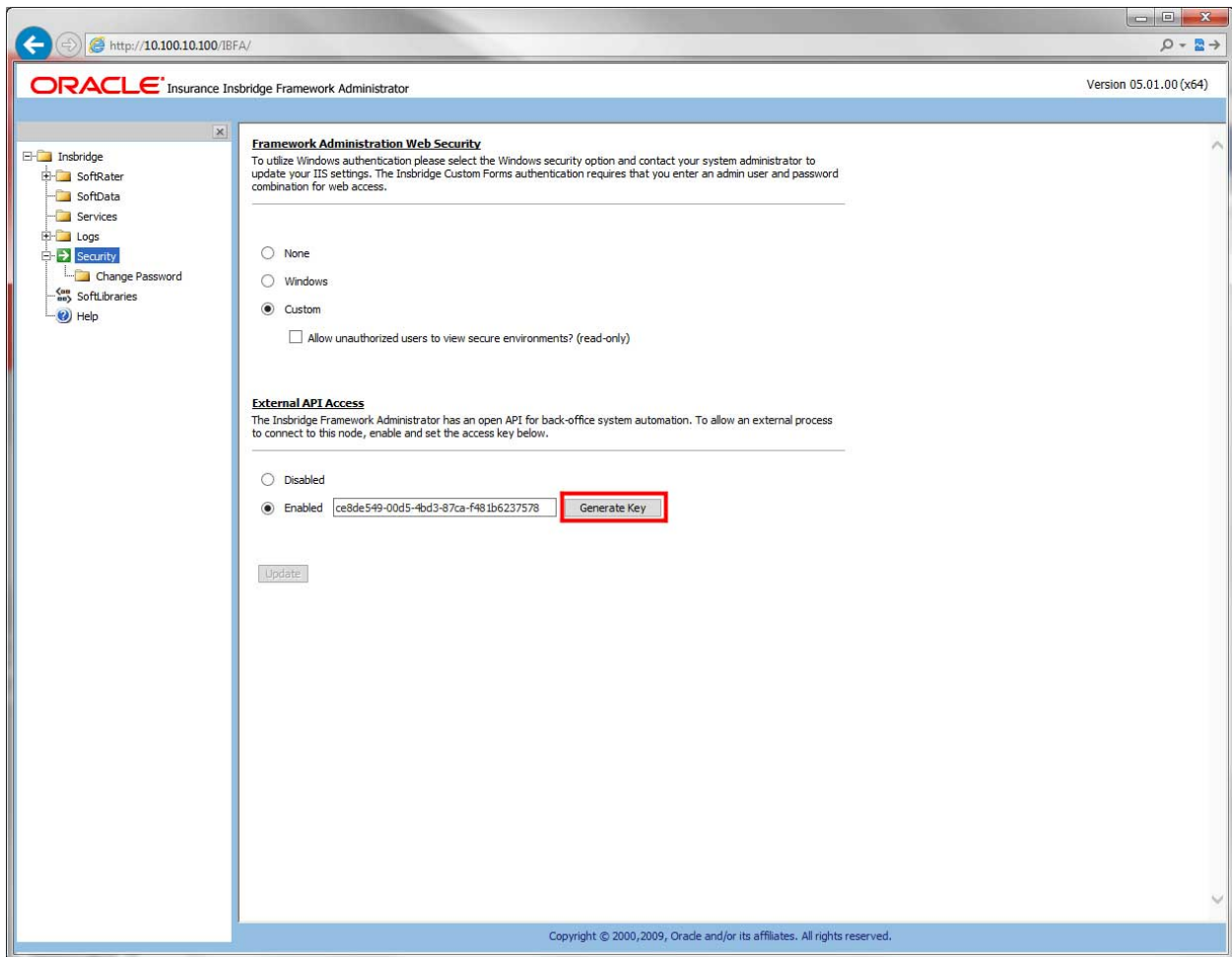


Figure 2 Allowing External API Access

## ESI – IBSS SYSTEM CONFIGURATION

### LIST OF FILES

The following are the list of files needed to Test ESI - IBSS

File	Usage
EsiSwingTester.cmd	Command file that will start the swings tester Application.
EsiSwingTester.jar	EsiSwingTester– Swings App
InsbridgeESIServices.jar	Esi Services are packaged to this jar

#### Prerequisite:

- Set up JAVA\_HOME correctly to the location where java is installed in the system.
- InsbridgeESIServices.jar must be in the same folder as the esiTester.cmd. If not, then it must be configured in the classpath variable of cmd file.

#### EsiSwingTester.cmd

This cmd file starts the esi-tester swings app.

#### EsiSwingTester.jar

- The tester swing app can help in testing the ESI.
- Both IBSS and IBFA testing are integrated. This document covers only the IBSS Testing. Please see ESI for Windows for more information.

#### InsbridgeESIServices.jar

- The ESI Logic and the services part are packaged in this .jar file.
- There is a config.properites file in this .jar that used for setting up the configuration for ESI.

## CONFIGURATION CHANGES

The config.properties requires editing prior to usage.

#### Config.properties

1. Open the InsbridgeESIServices.Jar file using an unzip utility and open the config.properties file for editing.

2. Change the following to match your server and EJB details where IBSS is deployed.
  - ESI\_IN\_SERVER=false
  - hostName=localhost:7101
  - contextRoot=ins.ru.sr.ui-context-root

If ESI is deployed to an Application Server, then the following details need to be configured.

#### **Weblogic Configuration**

- ESI\_EJB\_JNDI=ESI#com.oracle.ins.ru.sr.bsn.ejb.esi.ESI
- INITIAL\_CONTEXT\_FACTORY=weblogic.jndi.WLInitialContextFactory
- PROVIDER\_URL=t3://127.0.0.1:7103

#### **WebSphere Configuration**

- ESI\_EJB\_JNDI=com.oracle.ins.ru.sr.bsn.ejb.esi.ESI
- INITIAL\_CONTEXT\_FACTORY= com.ibm.websphere.naming.WsnInitialContextFactory
- PROVIDER\_URL=iiop://127.0.0.1:7103

#### **JBOSS Configuration**

- ESI\_EJB\_JNDI= IBSS/ins.ru.sr.bsn-1.0.0/SoftRaterEJB!  
com.oracle.ins.ru.sr.bsn.ejb.esi.ESI
- INITIAL\_CONTEXT\_FACTORY= org.jnp.interfaces.NamingContextFactory
- PROVIDER\_URL=jnp://127.0.0.1:7103



## **ESI IBSS TESTING**

### **Submit Insbridge XML as XML**

1. Open the Insbridge XML tab
2. Select Insbridge XML as XML Type
3. Select the test type as XML
4. Copy and paste the Insbridge XML to the Insbridge XML Text area
5. Enter the subscriber ID
6. Enter the Environment ID
7. Click on submit to Runtime DB
8. Result will be your batch id or Error message in case of errors

### **Submit Insbridge XML as Objects**

1. Open the Insbridge XML tab
2. Select insbridge XML as XML Type
3. Select the test type as Object
4. Copy and paste the Insbridge XML to the Insbridge XML Text area
5. Enter the subscriber ID
6. Enter the Environment ID
7. Click on submit to Runtime DB
8. Result will be your batch id or Error message in case of errors

### **Submit Insbridge XML Files to Queue**

1. Open the Insbridge XML tab
2. Select File as XML Type
3. Select the test type as XML
4. Enter the path to the location of the xml files
5. Enter a name for the batch
6. Enter the subscriber ID
7. Enter the Environment ID

8. Click on submit to Runtime DB
9. Check Broker messages with status as pending to see if the files are added to the queue.
10. Result will be your batch id or Error message in case of errors

### **Submit Insbridge XML Files as Objects To Queue**

1. Open the Insbridge XML tab
2. Select XML Type as File
3. Select the test type as Object
4. Enter the path to the location of the xml files
5. Enter a name for the batch
6. Enter the subscriber ID
7. Enter the Environment ID
8. Click on submit to Runtime DB
9. Check Broker messages with status as pending to see if the files are added to the queue.
10. Result will be your batch id or Error message in case of errors

## **DATABASE FUNCTIONS**

ESI currently supports 3 database related functions as listed below:

### **GET INPUT XML BY POLICY NUMBER**

1. Open the Insbridge XML tab
2. Select XML Type as DB
3. Select Get Input XML as DB Method
4. Select the test type as XML
5. Enter the batch id and Policy Number
6. Enter the subscriber ID
7. Enter the Environment ID
8. Click on submit to Runtime DB
9. Result will be your input xml or Error message in case of errors

### **Get Input XML by policy Number as Object**

1. Open the Insbridge XML tab
2. Select XML Type as DB
3. Select Get Input XML as DB Method
4. Select the test type as Object
5. Enter the batch id and file Policy Number
6. Enter the subscriber ID
7. Enter the Environment ID
8. Click on submit to Runtime DB
9. Result will be your input xml or Error message in case of errors

### **Get Result XML by Policy Number**

1. Open the Insbridge XML tab
2. Select DB as XML Type
3. Select Get Result XML as DB Method
4. Select the test type as XML
5. Enter the batch id and Policy Number
6. Enter the subscriber ID
7. Enter the Environment ID
8. Click on submit to Runtime DB
9. Result will be your Result xml or Error message in case of errors

### **Get Result XML by Policy Number as Object**

1. Open the Insbridge XML tab
2. Select DB as XML Type
3. Select Get Result XML as DB Method
4. Select the test type as Object
5. Enter the batch id and Policy Number
6. Enter the subscriber ID
7. Enter the Environment ID

8. Click on submit to Runtime DB
9. Result will be your Result xml or Error message in case of errors

### **Copy Batch**

1. Open the Insbridge XML tab
2. Select DB as XML Type
3. Select Copy Batch as DB Method
4. Select the test type as XML
5. Enter the batch id
6. Enter the subscriber ID
7. Enter the Environment ID
8. Click on submit to Runtime DB
9. Result will be your New Batch ID or Error message in case of errors

## **BATCH – STARTING AND MONITORING**

### **Start Batch**

1. Enter Batch ID, Subscriber ID, Environment ID, File Ref id, Notification details and read write option.
2. Click on start Batch
3. Result will be your request id or error in case of errors

### **Stop Batch**

1. Enter request Id
2. Click on Stop Batch
3. Result will be the status of the stop batch request

### **Get Batch Status**

1. Enter request Id

2. Click on Get Batch Status
3. Result will be the status of the batch request

## ESI TESTER

To use the ESI Tester, launch the .cmd file. The ESI tester runs on a Windows machine. The Java version of ESI allows for IBSS testing as well as RateManager. Select the option you want to test.

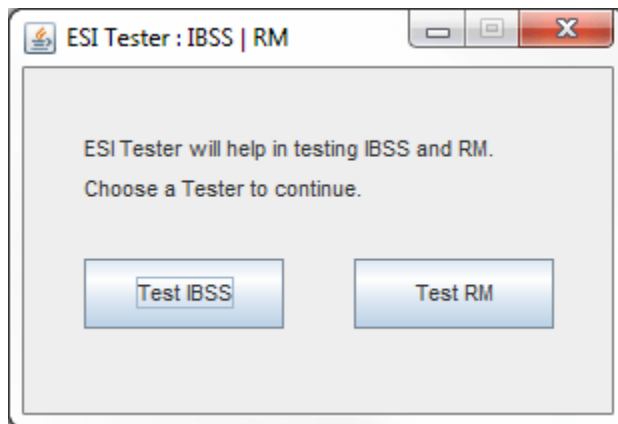


Figure 3 Selecting Instance to Test

## IBSS FEATURES/METHODS

The IBSS features.

- Insbridge XML
- XCategory
- Batch

Status is displayed in the upper corner. Offline indicates that webservice is not available. Check the configuration settings to make sure that correct configuration is being used. Refresh to update the status.

If the incorrect configuration for the application server is used, that status displays as offline.

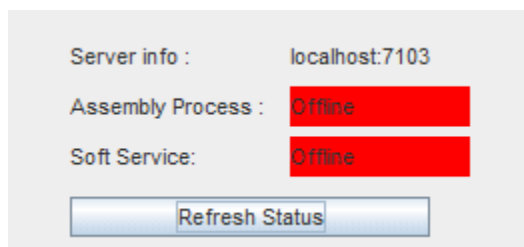


Figure 4 Connection Status

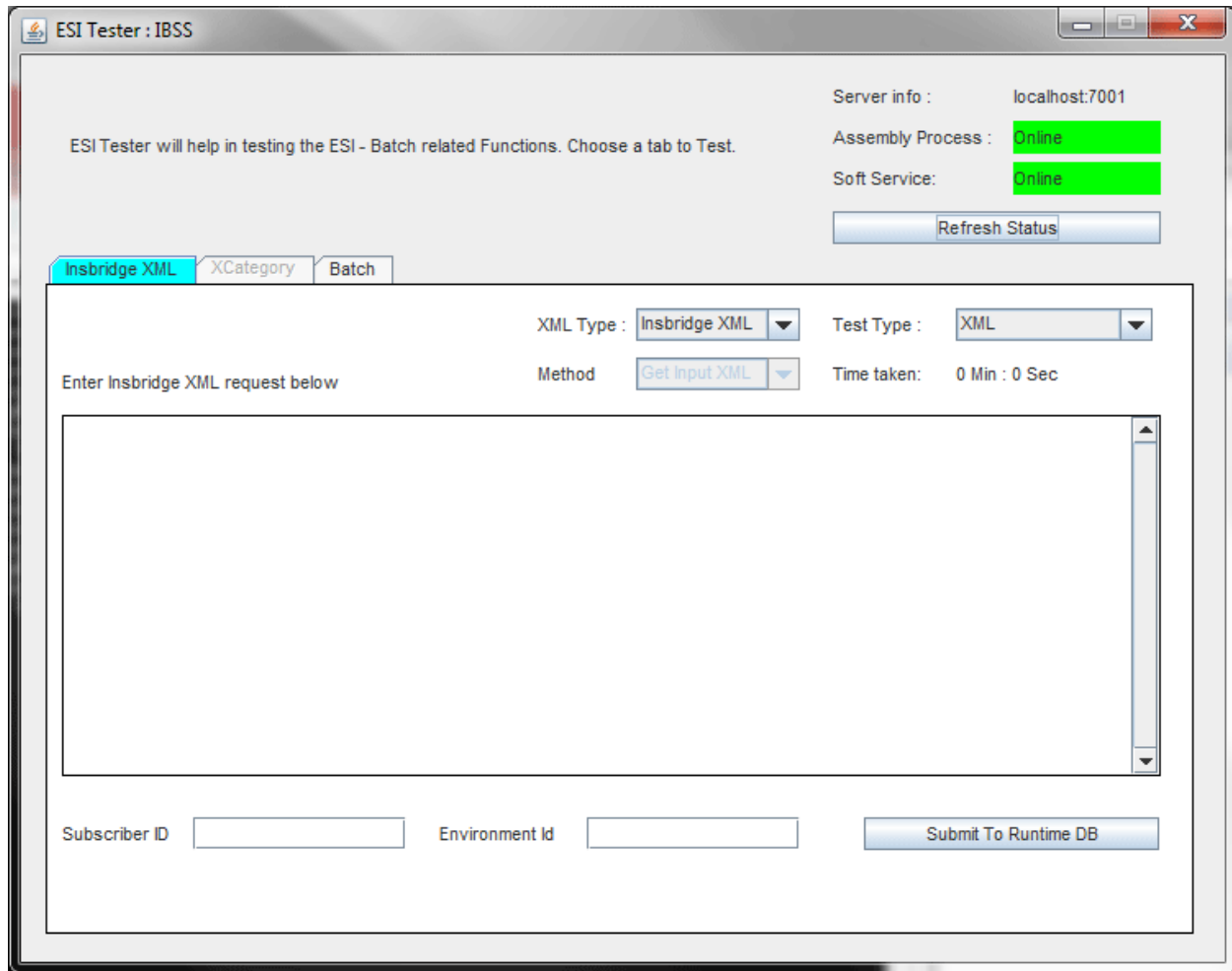


Figure 5 Testing IBSS

Testing IBSS allows for:

- Submit Insbridge XML as XML
- Submit Insbridge XML as Objects
- Submit Insbridge XML Files to Queue
- Submit Insbridge XML Files as Objects To Queue

## Database Functions

ESI currently supports database related functions:

- Get Input XML by Policy Number
- Get Input XML by policy Number as Object
- Get Result XML by Policy Number
- Get Result XML by Policy Number as Object
- Copy Batch

Select the types and paste in the XML request. Enter in Subscriber and environment Id. Click Submit to Runtime DB.

## RM FEATURES/METHODS

- Test Security
- Test Globals
- Test Program Items
- Test IBFA
- Test Releases
- Test List Items

### Test Security

Allows for the adding and editing of User Groups and Users.

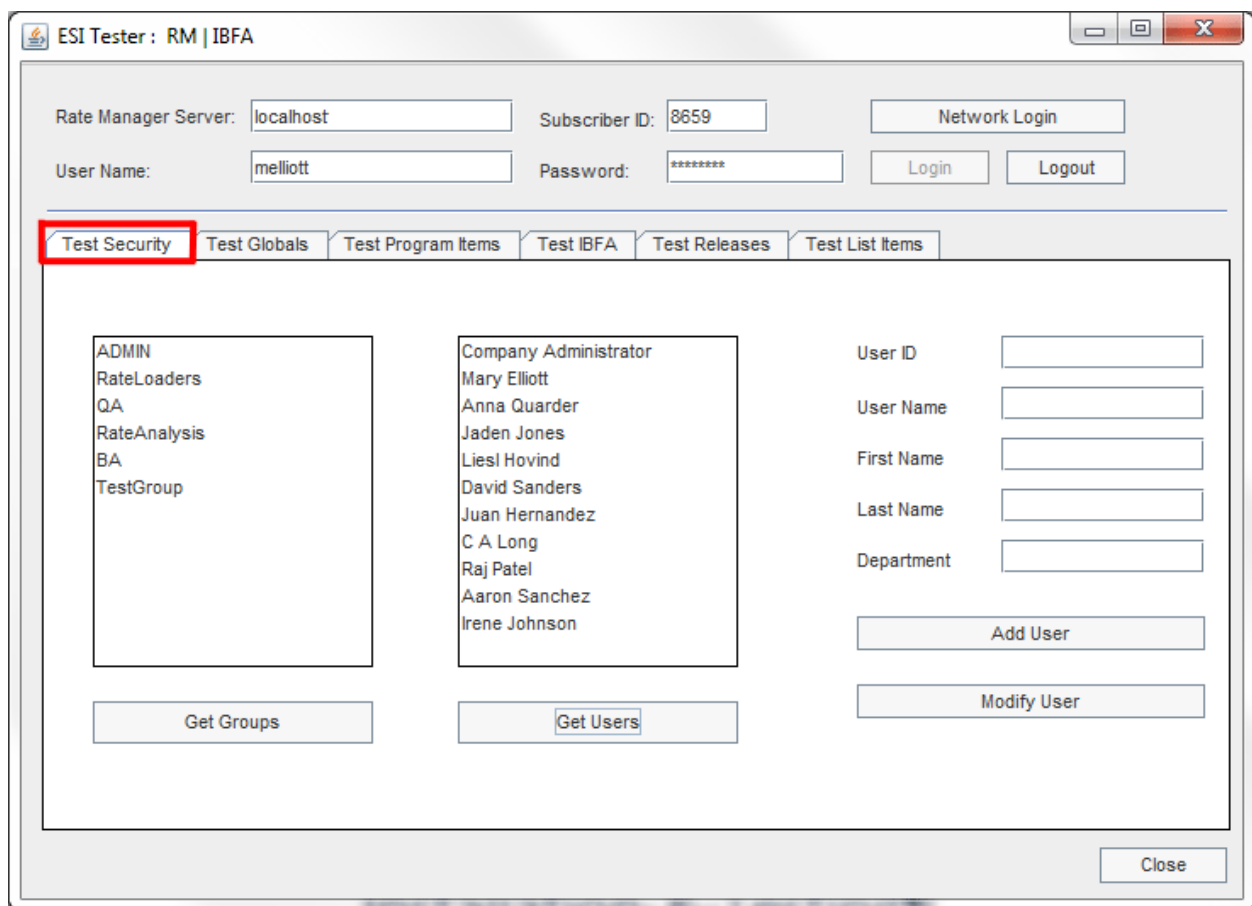


Figure 6 Test Security

- Get Groups returns the current groups in the system
- Get Users returns a list of the current users in the system.
- Add User and Modify User requires:
  - User Name



- First Name
- Last Name

Other fields are optional

## Test Globals

Allows for the adding, editing and deleting of LOBs, Categories, and Inputs.

The screenshot shows the ESI Tester application window titled "ESI Tester : RM | IBFA". The interface includes a login section with fields for "Rate Manager Server" (localhost), "Subscriber ID" (8659), "User Name" (melliott), and "Password" (\*\*\*\*\*). There are buttons for "Network Login", "Login", and "Logout". Below the login section is a tabbed interface with tabs for "Test Security", "Test Globals" (highlighted with a red box), "Test Program Items", "Test IBFA", "Test Releases", and "Test List Items". The "Test Globals" tab contains a form with three input fields: "Line ID" (1), "Cat ID", and "Input ID". Below these fields are several buttons: "Create LOB", "Create Category", "Create Input", "Create Category w/ inputs", "Get Category XML", "Update LOB", "Update Category", "Update Input", "Get Inputs XML", "Delete LOB", "Delete Category", and "Delete Input". A "Close" button is located at the bottom right of the window.

Figure 7 Test Globals

LOB requests require a Line ID.

- Get Category XML returns the details of the categories in the selected line.
- Get Inputs XML returns the details of the inputs in the selected line.

## Test Program Items

Allows for viewing, adding, editing, copying and deleting of program items.

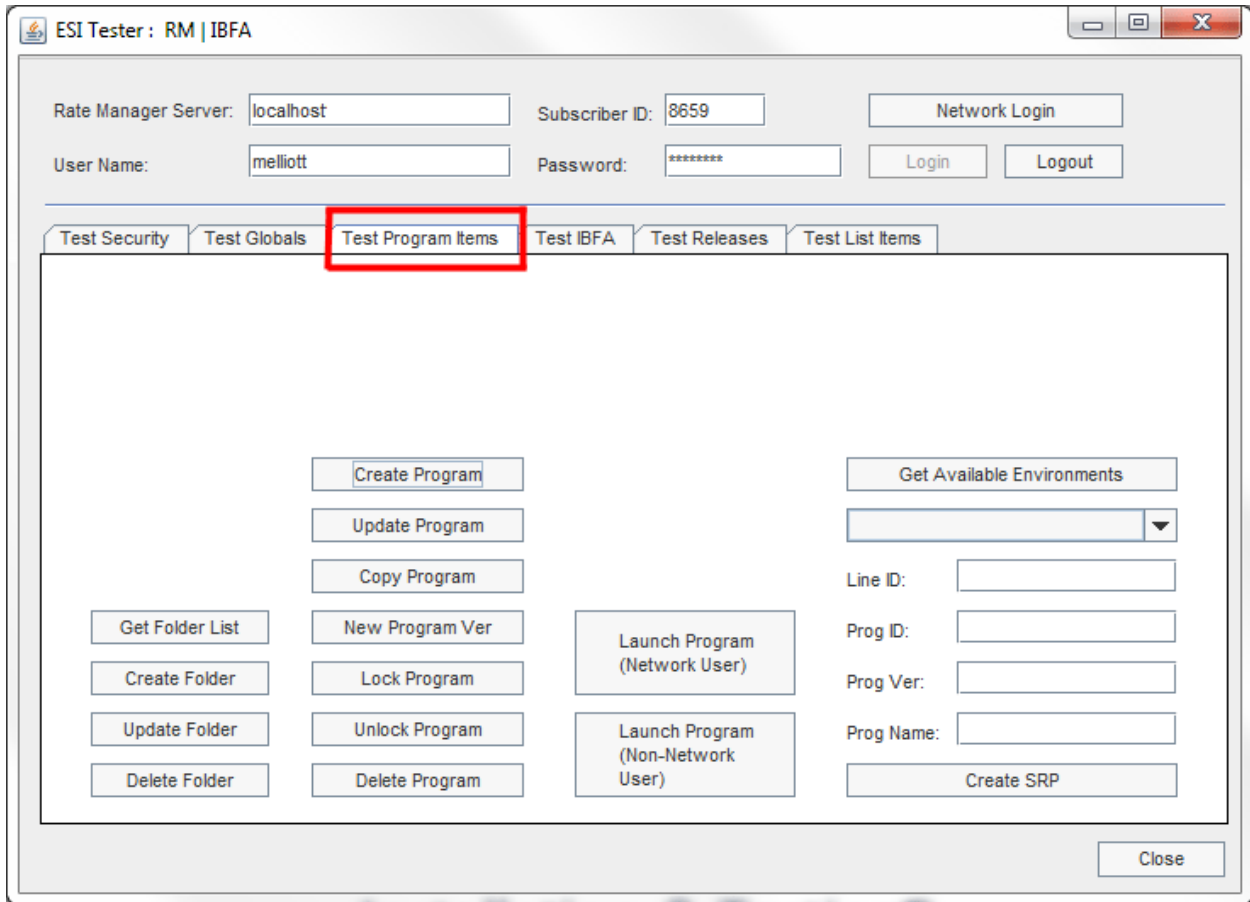


Figure 8 Test Program Items

Get Available Environments returns the environments available. The list is populated drop down.

Get Folder List requires a Line ID and returns a list of folders in the line.

## Test IBFA

Allows for SRPs to be managed.

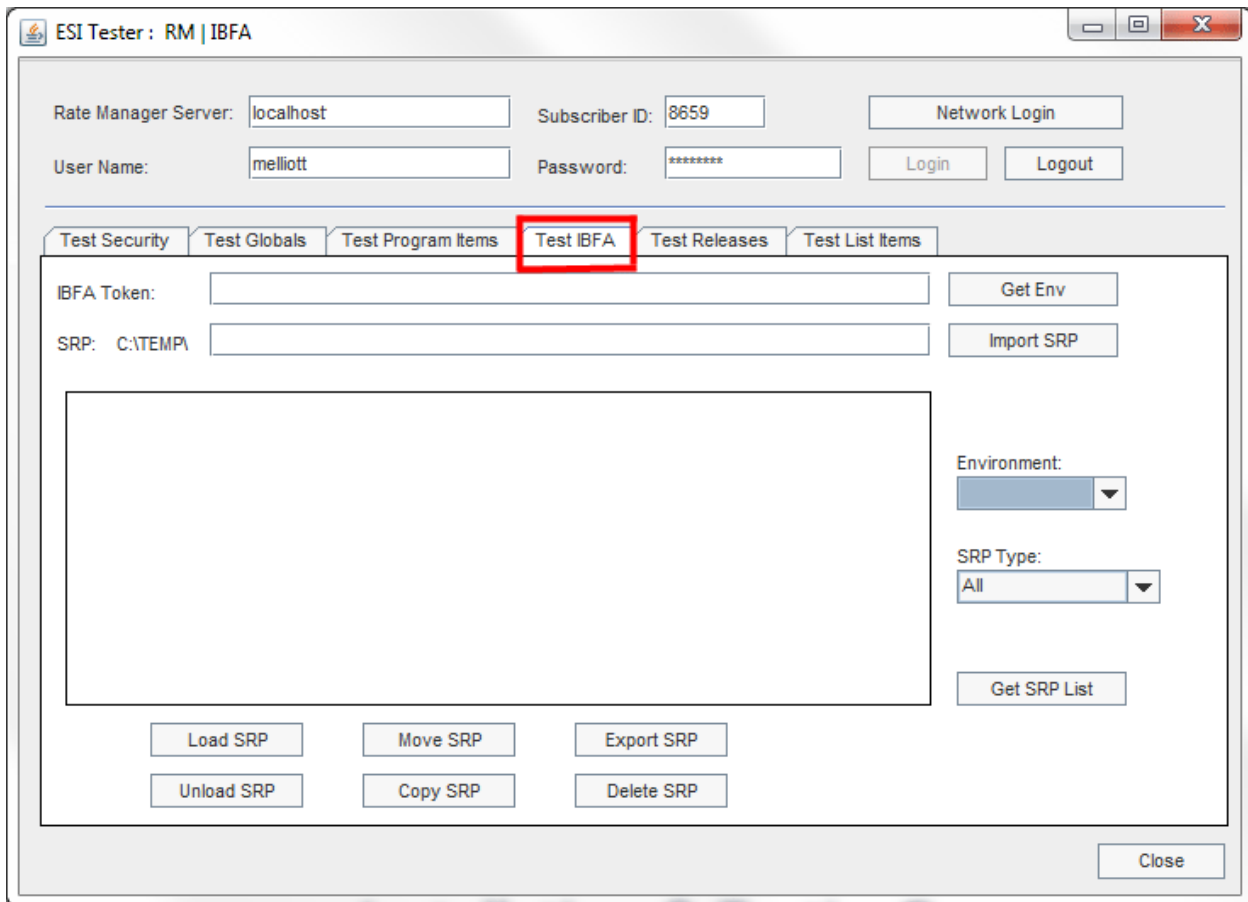


Figure 9 Test IBFA

Get Env returns the environments available. The list is populated Environment drop down.

## Test Releases

Allows for the management of releases.

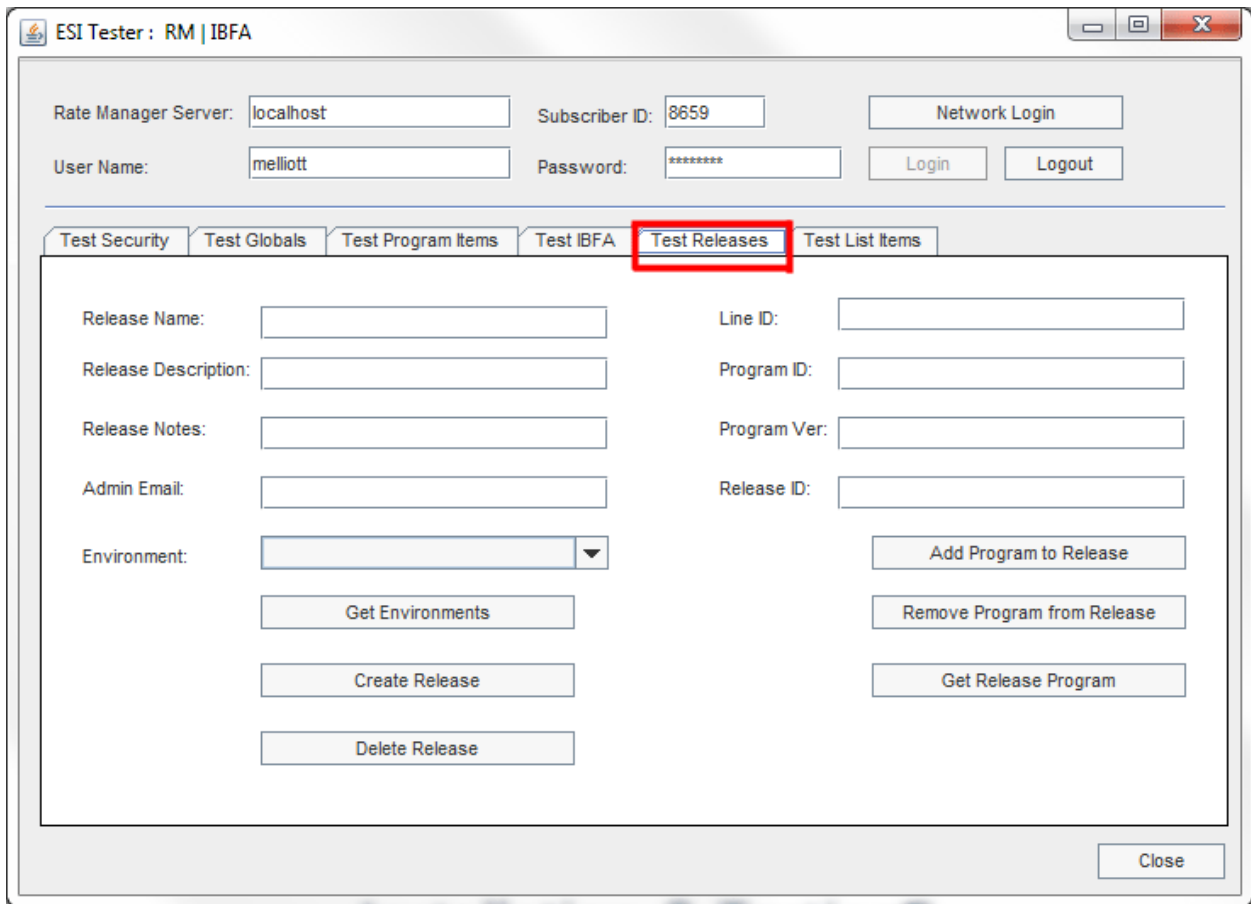


Figure 10 Test Releases

Get Environments returns the environments available. The list is populated Environment drop down.

## Test List Items

- Return a list of lines of business
- Return a list of schemas for a particular line of business
- Return a list of subline (schema) folders for a particular schema
- Return a list of folder for a particular subline folder
- Return a list of programs for a subline folder
- Return a list of program versions for a particular program
- Return the XML for the program version report for a particular program version

**NOTE:** For the above all features it will return the list in the XML format.

Test List Items allows you to build a treeview that follows the program explorer in RateManager and for the XML used for the program version report to be returned.

The columns on the right follow the treeview found on the Program Explorer in RateManager.

Figure 11 Java ESI Tester for RM

## To Use

Options are available after you enter in the Subscriber ID, username and password.

Enter in any required input and click the button of the items where you want returned. The output is returned in XML format.

### GET LOB LIST

- **Input:** No input is required.
- **GET LOB LIST:** Gets an XML string of all the line of business including active and inactive lobs.
- **Output:** Returns an XML string of available LOB's. XML string has:
  - LOB\_ID
  - LOB\_NAME
  - ACTIVE attributes

### GET SCHEMA LIST

- **Inputs:** Line ID.
- **GET SCHEMA LIST:** Gets an XML string of available schemas of a particular LOB.
- **Outputs:** Returns an XML string that includes all the schemas of that LOB. XML string has:
  - LOB\_ID
  - SCHEMA\_ID
  - SCHEMA\_NAME

### GET FOLDER LIST

- **Inputs:** Line ID and Schema ID.
- **GET FOLDER LIST:** Gets an XML string of available folders of a particular Schema.
- **Outputs:** Returns the XML string that includes all the folders of that Schema. XML String has:
  - LOB\_ID
  - SCHEMA\_ID
  - FOLDER\_ID
  - FOLDER\_NAME

### GET SUB FOLDER LIST

- **Inputs:** Line ID, Schema ID and Folder ID.
- **GET SUB FOLDER LIST:** Gets an XML string of available sub folders of a particular subline folder/ folder.
- **Outputs:** Returns an xml string that includes all the sub folders of that Subline Folder/Folder. XML string has

- LOB\_ID
- SCHEMA\_ID
- FOLDER\_ID
- FOLDER\_NAME
- FOLDER\_PARENT\_ID

### GET PROGRAM LIST

- **Inputs:** Line ID, Schema ID and Folder ID.
- **GET PROGRAM LIST:** Gets an XML string of available Programs of a particular folder.
- **Outputs:** Returns an xml string that includes all programs of that Folder.  
XML string has:
  - LOB\_ID
  - SCHEMA\_ID
  - FOLDER\_ID
  - PROGRAM\_ID
  - PROGRAM\_NAME

### GET PROGRAM VERSIONS LIST

- **Inputs:** Line ID, Schema ID, Folder ID and Program ID.
- **GET PROGRAM VERSIONS:** Gets an XML string of available Program versions of a particular program.
- **Outputs:** Returns an xml string that includes all program versions of that Program.  
XML string has:
  - LOB\_ID
  - SCHEMA\_ID
  - FOLDER\_ID
  - PROGRAM\_ID
  - VERSION

### GET PROGRAM VERSION REPORT XML

- **Inputs:** Line ID, Schema ID, Folder ID, Program ID, Program Ver and Program Name.
- **GET PROGRAM VER REPORT XML:** Gets an XML string of program version report of a particular program version.
- **Outputs:** Returns an XML string of program version report.

### GET PROGRAM VERSION REPORT LOCATION

This method returns the saved location of the program version report.

- **Inputs:** Line ID, Schema ID, Folder ID, Program ID, Program Ver , Program Name and Folder Path(optional).
- **Outputs:** Returns the saved location of the program version report.

**NOTE:** The input Folder Path is optional. Folder Path is the Folder/Directory location of the report where it has to save. If we give the folder path it will generate the report and saves in the given location. If we don't give the Folder Path then by default system will create one location and it will save that location.

## Output

Output is in XML format.

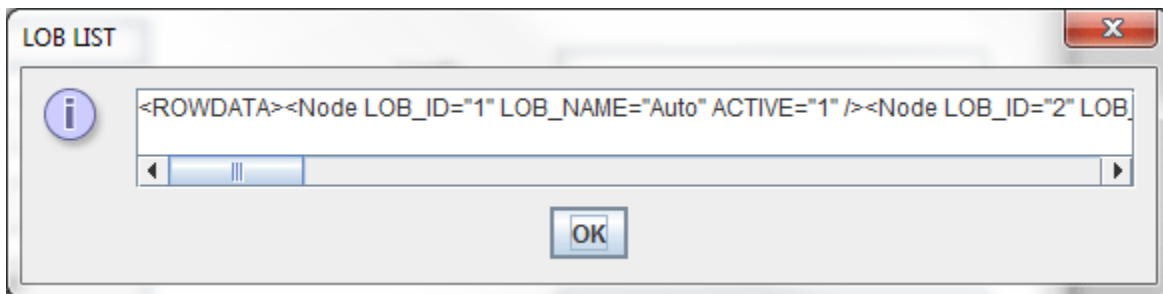


Figure 12 Tester Output

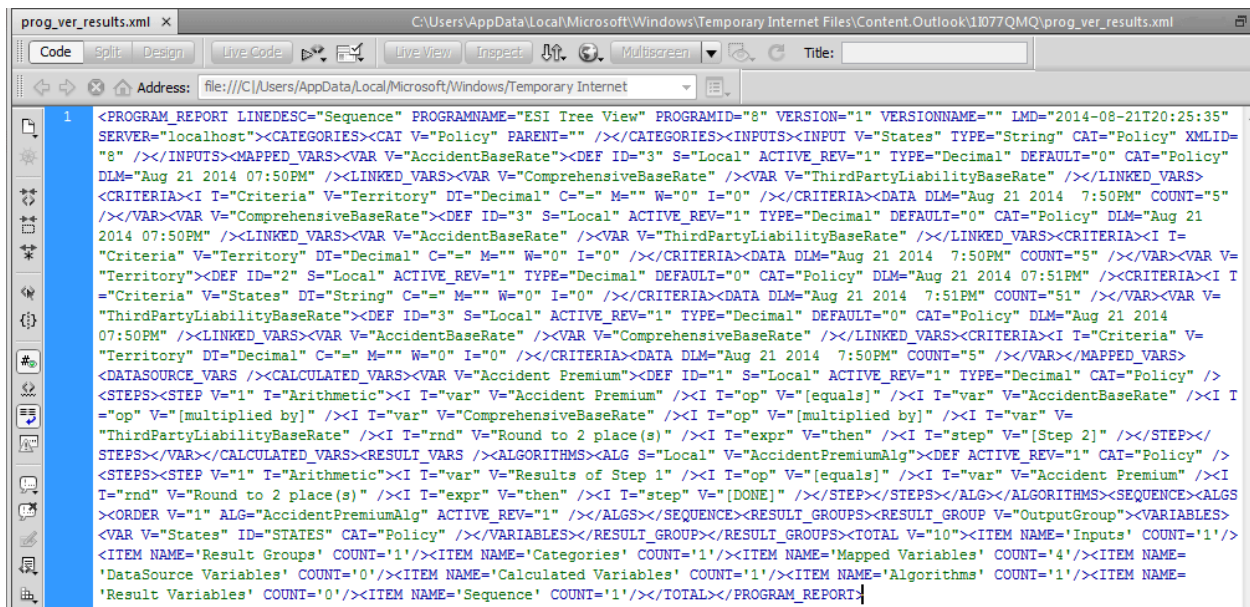


Figure 13 Program Result XML File



## CONTACTING SUPPORT

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

Oracle customers have access to electronic support through My Oracle Support. For information, visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> or visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> if you are hearing impaired.

**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](http://oracle.com)

## INDEX

### A

- Add Batch to Runtime DB
  - InputBatch, 36
- Add Category Inputs
  - Rate or Result, 40
- Add Child Category
  - Rate or Result, 40
- Add Parent Category
  - Rate or Result, 40
- Add Program Details
  - Rate or Result, 40
- Add Rating Details
  - Rate or Result, 40
- Adding
  - New User in User Management, 41
- API Key, 43
- Authentication
  - ESI User, 41

### B

- Base ESI Domain Types, 16
- Business Use Case, 17

### C

- Category
  - Create, 25
  - Create with Inputs, 25
  - Delete, 26
  - Get, XML, 25
  - Update, 25
- Class Definition
  - IBFA, 32
  - RateManager Interface, 22
- Company
  - Selecting for New User, 41
- Constructor Detail
  - RateManager Interface, 22
- Copy Definitions & All Data
  - Copying Programs, 28
- Copy Definitions Only
  - Copying Programs, 28
- Copy Program, 28
- Copy SRP, 32
- Create, 38
  - Network Login, 41
- Create Category, 25
  - With Inputs, 25
- Create Folders, 26
- Create Input, 24
- Create Line, 26
- Create New User, 30

- Create Program, 27
- Create Program Version, 28
- Create Release, 29
- Create SRP, 31
- CreateNew Rate
  - InputBatch, 36
- Creating
  - InputBatch, 14
  - Rate Object, 12

### D

- Default
  - API Key, 43
- Delete Category, 26
- Delete Folder, 27
- Delete Line, 26
- Delete Program, 28
- Delete Release, 29
- Delete SRP, 33
- Deletes Input, 24

### E

- Edition Notice, 2
- Environments
  - Get Available, Non-Secured, 22
  - Get Available, Secured, 23
- ESI
  - Overview, 9
  - Rules, 10
- ESI Domain Types, 16
- Example
  - Business Use Case, 17
- ExecuteBatch
  - InputBatch, 36
- Executing
  - InputBatch, 15
- Export SRP, 33

### F

- False
  - New Users, 42
- Folder
  - Create, 26
  - Delete, 27
  - Get XML, 27
  - Update, 27
- Functionality Highlights
  - ESI, 11

### G

- Get

- User Groups, 31
- Get Available
  - Environments, Non-Secured, 22
  - Environments, Secured, 23
  - User Groups, 31
- Get Bytes From File, 33
- Get Category XML, 25
- Get Folders XML, 27
- Get Inputs Array, 25
- Get SRP List, 33
- Group
  - Selecting for New User, 41

**I**

- IBFA
  - Class Definition, 32
  - System Configuration, 43
- IBRequestResponse Interface, 12
- Import SRP, 34
- InputBatch
  - Add Batch to Runtime DB, 36
  - Create New Rate, 36
  - Creating, 14
  - Execute Batch, 36
  - Executing, 15
  - Monitoring, 15
  - Retrieve Input Rate, 36
  - Retrieving Inputs, 15
  - Submit Insbridge XML Files from Disk, 37
  - Submit Insbridge XML from Memory, 37
  - Submit Rate Object to Runtime DB, 37
  - Submitting, 15
- InputObject, 12
- Inputs
  - Create New, 24
  - Delete, 24
  - Get, Array, 25
  - Update, 24

**J**

- Job
  - Retrieve Job Status, 39
  - Stop the Job, 39

**L**

- Launch Definition
  - First Version, 23, 24
  - Selected Version, 23
- Line
  - Create, 26
  - Delete, 26
  - Program, 28
  - Update, 26
- List Release, 30
- Load SRP, 34
- login, 30
- logout, 30

**M**

- Monitoring
  - InputBatch, 15
- Move SRP, 34

**N**

- Network Login
  - ESI Authentication, 41
- Network User
  - New User, 41
- networkLogin, 30
- New User in User Management
  - Adding, 41

**O**

- Objects
  - Rate, 12
- Overview
  - ESI, 9

**P**

- Password
  - Default, 41
- Password Expires
  - New User, 42
- Program
  - Copy, 28
  - Copy Definitions & All Data, 28
  - Copy Definitions Only, 28
  - Create, 27
  - Create New Version, 28
  - Update, 27
  - Update Version Lock1, 28

**Q**

- Quick Guide for ESI, 20

**R**

- Rate Model for IBSS, 11
- Rate Object
  - Creating, 12
- Rate Objects, 12
- Rate or Result
  - Add Category Inputs, 40
  - Add Child Category, 40
  - Add Parent Category, 40
  - Add Program Details, 40
  - Add Rating Details, 40
- RateManager Login
  - Create Launch First Version, 23, 24
  - Create Launch Selected Version, 23
  - ESI Authentication, 41
- Release
  - Create, 29
  - Delete, 29

- Delete Program, 29
- List Program, 30
- Update, 29
- Result Batch
  - Create New Result, 38
  - Retrieve Results, 38
  - Retrieving Results, 15
- Retrieve, 38, 39
- Retrieve Input Rates
  - InputBatch, 36
- Retrieving Inputs
  - InputBatch, 15
- Retrieving Results, 15
  - Result Batch, 15
- Rules
  - ESI, 10
  - Username, 41

**S**

- Save File SRP, 34
- SRP
  - Copy, 32
  - Create, 31
  - Delete, 33
  - Export, 33
  - Get Bytes from File, 33
  - Get List, 33
  - Import, 34
  - Load, 34
  - Move, 34
  - Save File, 34
  - Unload, 35
- Stop, 39
- Submit Insbridge XML Files from Disk
  - InputBatch, 37
- Submit Insbridge XML from Memory
  - InputBatch, 37
- Submit Rate Object to Runtime DB
  - InputBatch, 37
- Submitting
  - InputBatch, 15

- System Configuration
  - IBFA, 43
- System Requirements, viii

**T**

- Tester, 46
- True
  - New Users, 42

**U**

- Unload SRP, 35
- Update Category, 25
- Update Folder, 27
- Update Input, 24
- Update Line, 26
- Update Program, 27
- Update Program Version Lock, 28
- Update Release, 29
- Update User, 31
- User
  - Create New, 30
  - Login, 30
  - Logout, 30
  - Network Login, 30
  - Network User for New, 41
  - Selecting Company for New, 41
  - Selecting Group for New, 41
  - Update, 31
  - Username for New, 41
- User Authentication, 42
- User Groups
  - Get, 31
  - Get Available, 31
- Username
  - New User, 41
  - Rules, 41
- Users
  - Password Expires, 42