

Oracle Insurance

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

Release 5.6

November 2018

Copyright © 2005, 2018, Oracle and/or its affiliates. All rights reserved.

Oracle Insurance Insbridge Enterprise Rating ESI for Java User Guide

Release 05.06.xx

Part # E96973-01

Library # E96993-01

November 2018

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	VIII
	Audience	viii
	Related Documents	viii
	Conventions	viii
	System Requirements.....	ix
	Manual History	ix
CHAPTER 1		
	INTRODUCTION TO INSBRIDGE EXTENDED SERVICES INTERFACE	10
CHAPTER 2		
	USING ESI: IBSS - EXAMPLE	13
	Creating a Rate Object	13
	IBRequestResponse Interface	13
	Creating a Rate Object	13
	Input Batch – Creating, Submitting, Executing and Monitoring	15
	Result Batch.....	16
CHAPTER 3		
	BASE DOMAIN TYPES	17
	Use Case Summary.....	18
	Quick Guide Table	23
CHAPTER 4		
	RATEMANAGER INTERFACE.....	25
	EsiRateManager	25
	EsiRateManager	25
	Environment Items – getAvailableEnvironments, All Non-Secured.....	25
	Environment Items – getAvailableEnvironments, Secured.....	26
	General – createLauchDefinition (Selected Version, Network Login)	26
	General – createLauchDefinition (First Version, Network Login)	26
	General – createLauchDefinition (Selected Version, RateManager Login).....	26
	General – createLauchDefinition (First Version, RateManager Login).....	27
	Global Items – CreateCategory	27
	Global Items – CreateCategory with Fields	27
	Global Items – CreateCategoriesFromXML.....	27
	Global Items – CreateField	28
	Global Items – CreateProduct.....	29
	Global Items – DeleteCategory.....	29
	Global Items – DeleteField.....	29
	Global Items – DeleteProduct.....	29
	Global Items – GetAllProductsList	29
	Global Items – GetCategoryXML	30
	Global Items – GetDomainList.....	30
	Global Items – GetFieldsXML.....	30

Global Items – GetProgramList.....	30
Global Items – GetProjectList	30
Global Items – GetReleases	31
Global Items – UpdateCategory.....	31
Global Items – UpdateField	31
Global Items – UpdateProduct.....	31
Program Items – AddProgramVersionNote	31
Program Items – AddTableVariableRevisionNote	32
Program Items – CopyProgramVersion	32
Program Items – CreateFolderGroup	32
Program Items – CreateProgram.....	32
Program Items – CreateProgramVersion	33
Program Items – CreateTableVariableRevision	33
Program Items – DeleteFolderGroup.....	33
Program Items – DeleteProgram	33
Program Items – DeleteProgramVersionNote	34
Program Items – DeleteTableVariableRevisionNotes	34
Program Items – DeleteUnlockedProgramVersions	34
Program Items – ExportTableVariableDatatoText	34
Program Items – ExportMultipleTableVariableDataToExcel	34
Program Items – ImportExcelDatatoTableVariable	35
Program Items – GetAvailableFolderGroups.....	35
Program Items – GetAvailableFolderGroupsXML	36
Program Items – GetProgramVersionNotes	36
Program Items – GetProgramVersionReportXML	36
Program Items – GetProgramVersions.....	36
Program Items – GetSubFolderList	36
Program Items – GetTableVariableRevisionNotes	37
Program Items – SetTableVariableActiveRevision	37
Program Items – UpdateFolderGroup	37
Program Items – UpdateProgram	37
Program Items – UpdateProgramVersionLock	38
Program Items – UpdateProgramVersionName	38
Program Items – UpdateProgramVersionNote	38
Program Items – UpdateTableVariableRevisionNote	38
Releases Items – AddReleaseProgram.....	38
Releases Items – CreateRelease	39
Releases Items – DeleteAllReleases.....	39
Releases Items – DeleteMultipleReleases	39
Releases Items – DeleteRelease.....	39
Releases Items – DeployRelease.....	40
Releases Items – GetReleasePrograms	40
Releases Items – GetReleaseProgramsPackageStatus	40
Releases Items – PackageReleasePrograms	40

Releases Items – RemoveReleaseProgram.....	41
Security Items – CreateUser.....	41
Security Items – GetAvailableUsers	41
Security Items – GetAvailableUserGroups	41
Security Items – TerminateSession	41
Security Items – UpdateUser	42
SRP Items – CreateSRP.....	42
Program Versioning Data – InsertProgramVersionDataRow	42
Program Versioning Data – InsertProgramVersionDataRows.....	42
Program Versioning Data – DeleteProgramVersionDataRow	43
Program Versioning Data – UpdateProgramVersionDataCell.....	43

CHAPTER 5

FRAMEWORK ADMINISTRATOR INTERFACE.....	44
EsiFrameworkAdministrator.....	44
EsiFrameworkAdministrator.....	44
SRP Items – CopySRP	44
SRP Items – DeleteSRP	45
SRP Items – DeleteOrUnload.....	45
SRP Items – ExportSRP	45
SRP Items – GetSRPList, ESI Created Only.....	46
SRP Items – GetSRPList, Selected SRP Type	46
SRP Items – ImportSRP	46
SRP Items – LoadSRP	46
SRP Items – MoveSRP.....	47
SRP Items – SaveFile.....	47
SRP Items – UnloadSRP	47

CHAPTER 6

INPUTBATCH	48
Create New Rate.....	48
Add Batch to Runtime DB	48
Execute Batch	48
Retrieve Input Rates	48
Submit Insbridge XML from Memory	49
Submit Insbridge XML Files from Disk.....	49
Submit Rate Object to Runtime DB	49

CHAPTER 7

RESULT BATCH.....	50
Create New Result.....	50
Retrieve Results.....	50

CHAPTER 8

JOB	51
Retrieve Job Status.....	51
Stop the Job	51

CHAPTER 9

RATE OR RESULT 52
 Add Rating Details 52
 Add Program Details 52
 Add Parent Category 52
 Add Child Category 52
 Add Category Inputs 52

CHAPTER 10

ESI USER SETUP AND AUTHENTICATION 53
 Adding a User 53

CHAPTER 11

ESI SYSTEM CONFIGURATION 55
 ESI JAVA TESTER CONFIGURATION CHANGES 56

CHAPTER 12

ESI JAVA TESTER 57
 Installation 57
 Prerequisites 57
 Test IBSS Features/Methods 62
 Database Functions 63
 ESI IBSS Testing 63
 Submit Insbridge XML as XML 64
 Submit Insbridge XML as Objects 64
 Submit Insbridge XML Files to Queue 64
 Submit Insbridge XML Files as Objects To Queue 65
 DATABASE FUNCTIONS 66
 Get Input XML by Policy Number 66
 Get Input XML by policy Number as Object 67
 Get Result XML by Policy Number 67
 Get Result XML by Policy Number as Object 67
 Copy Batch 68
 BATCH – STARTING AND MONITORING 69
 Start Batch 69
 Stop Batch 69
 Get Batch Status 70
 SOFTRATER 70
 RM Features/Methods 71
 Login 71
 Test Security 72
 Test Globals/Universals 73
 Test Program Items 74
 Test IBFA 76
 Test Releases 77
 Test List Items 78

	Test Versioning Data	84
SUPPORT		
	CONTACTING SUPPORT	85
INDEX		
	INDEX	86

LIST OF FIGURES

FIGURE 1 ADDING A NEW USER.....	54
FIGURE 2 ALLOWING EXTERNAL API ACCESS.....	55
FIGURE 3 SELECTING INSTANCE TO TEST.....	61
FIGURE 4 CONNECTION STATUS.....	61
FIGURE 5 TESTING IBSS.....	62
FIGURE 6 IBSS ESI TESTER LANDING PAGE.....	63
FIGURE 7 IBSS ESI TESTER INSBRIDGE XML.....	66
FIGURE 8 IBSS TESTER BATCH.....	69
FIGURE 9 IBSS ESI TESTER FOR SOFTRATER.....	70
FIGURE 10 LOGGING INTO ESI.....	71
FIGURE 11 TEST SECURITY.....	72
FIGURE 12 TEST GLOBALS/UNIVERSALS.....	73
FIGURE 13 TEST PROGRAM ITEMS.....	74
FIGURE 14 TEST IBFA.....	76
FIGURE 15 TEST RELEASES.....	77
FIGURE 16 JAVA ESI TESTER FOR RM.....	78
FIGURE 17 TESTER OUTPUT.....	83
FIGURE 18 TEST VERSIONING DATA.....	84

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
bold	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.
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.

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

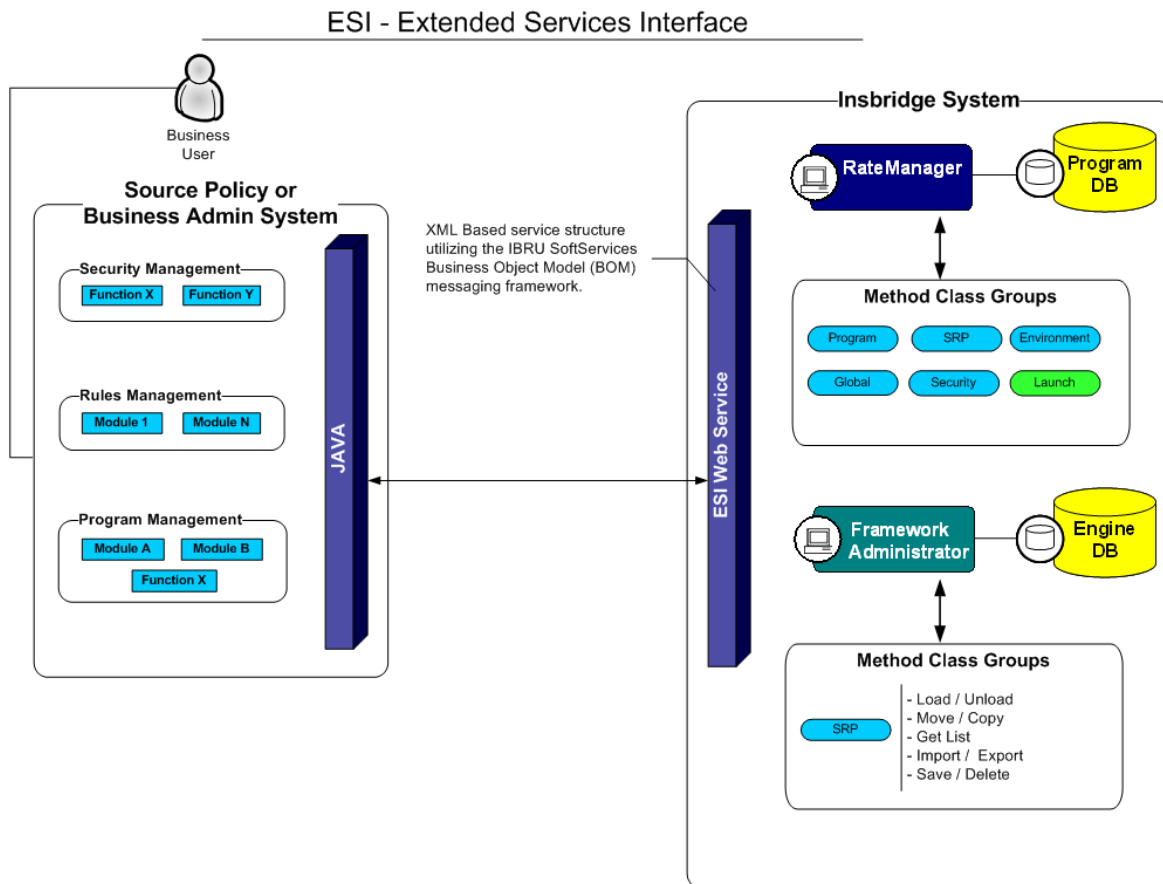
New editions incorporate any updates issued since the previous edition.

Edition	Publication Number	Product Version	Publication Date	Comment
1 st Edition	P01-783-01	R 4.1	December 2010	Initial release
2 nd Edition	P01-783-02	R 4.5	May 2011	Update
3 rd Edition	P01-783-03	R 4.5.1	September 2011	Update
4 th Edition	P01-783-04	R 4.6	May 2012	Update
5 th Edition	P01-783-05	R 4.6.1	November 2012	Update Release
6 th Edition	P01-783-06	R 4.7	September 2013	Update Release
7 th Edition	P01-783-07	R 4.7.1	November 2013	Update
8 th Edition	P01-783-08	R 4.8	August 2014	Update
9 th Edition	P01-783-09	R 4.9	December 2014	Update
10 th Edition	P01-783-10	R 5.1	December 2015	Update
11 th Edition	P01-783-11	R 5.2	July 2016	Update
12 th Edition	P01-783-12	R 5.3	October 2016	Update
13 th Edition	P01-783-13	R 5.4	January 2017	Update
14 th Edition	P01-783-14	R 5.5	September 2017	Update
15 th Edition	P01-783-15	R 5.6	August 2018	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 Java 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 Java is designed to operate on Insbridge version 04.01.00 or higher and connects with:

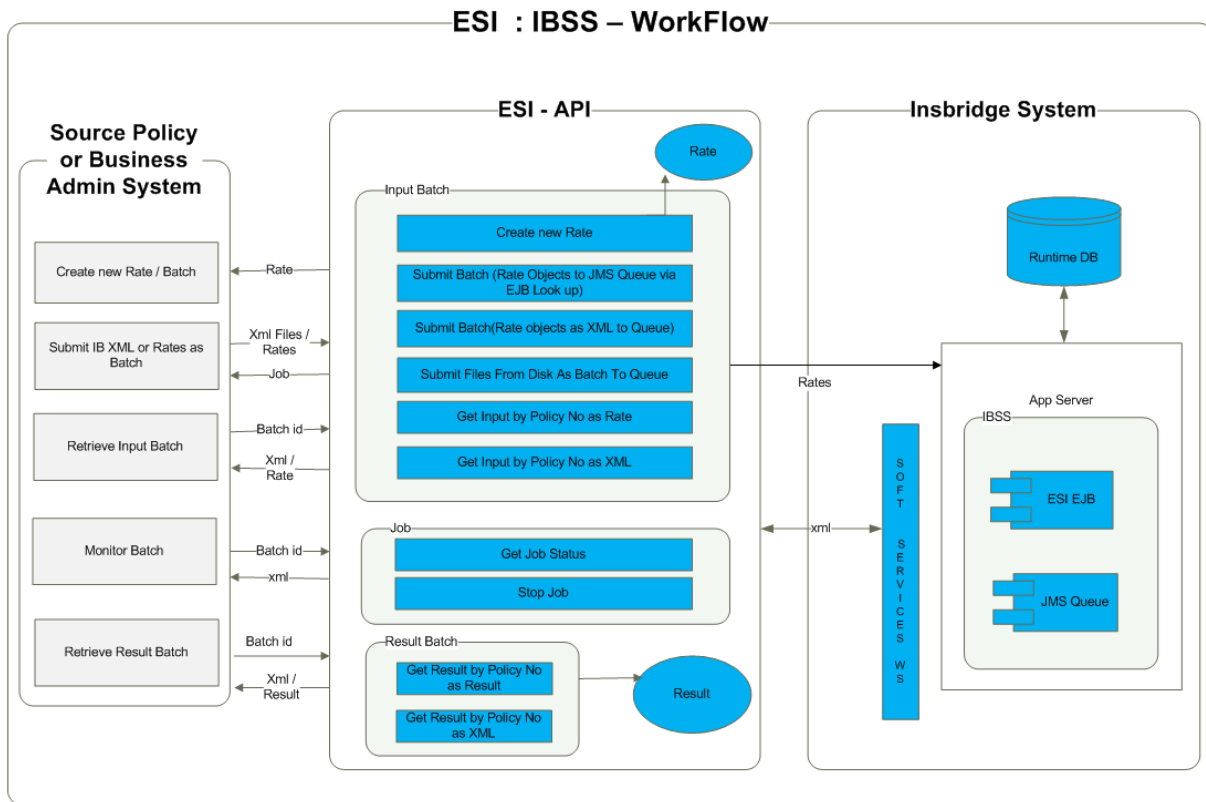
- Oracle Insurance RateManager Interface
- Oracle Insurance Framework Administrator Interface
- Oracle Insurance Insbridge SoftRater Server – Release 5.5 or later

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

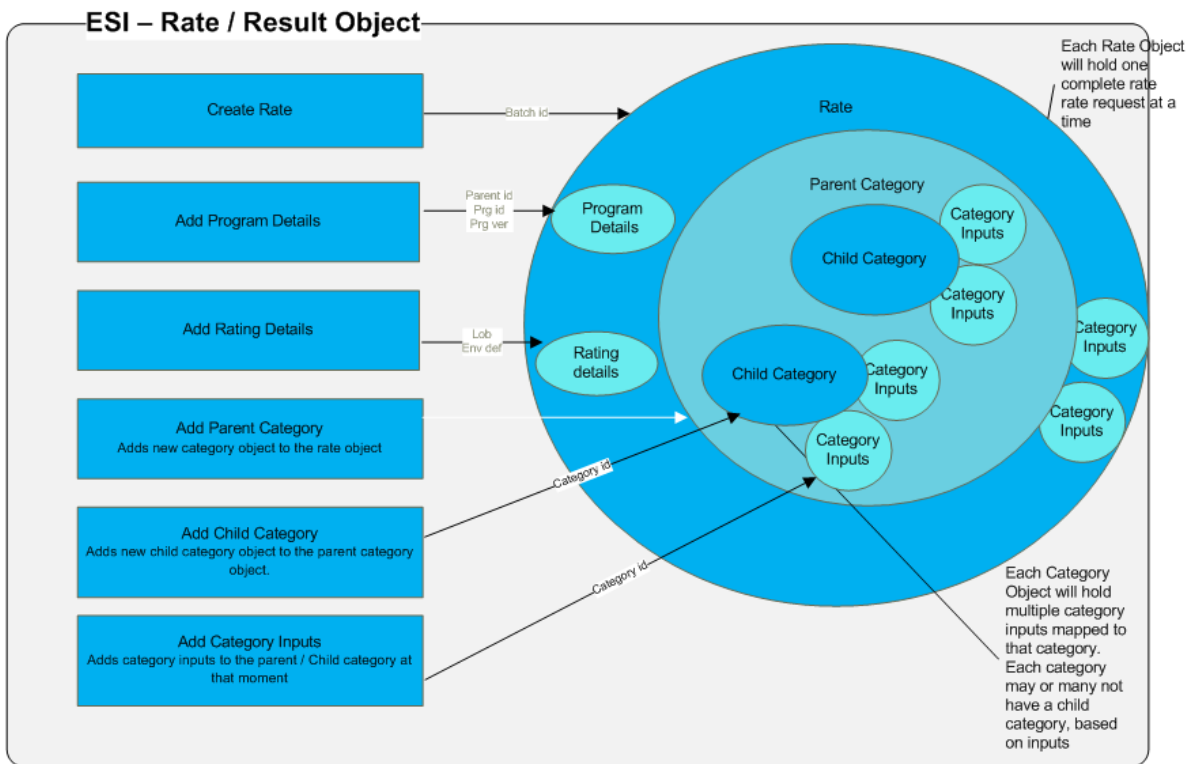
- ESI for Java is available on Insbridge Enterprise Rating version 04.01.00 or higher.
- ESI is shipped with the IBFA installation. To locate ESI, look under the Insbridge directory for the SDK/ESI/Java file; [INSTALL_DIR]\Insbridge\SDK\ESI\Java\lib.
- ESI entries are immediate. There is no lag time 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.
- 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 following table is a quick guide to the functions available for ESI.

Business Use Case	Methods to Use	Result
General – Open URL		
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.
Security Items		
Add	<u>Security Items – CreateUser</u>	Creates a new user.
List	<u>Security Items – GetAvailableUsers</u>	Returns a list of all available users for the subscriber.
List	<u>Security Items – GetAvailableUserGroups</u>	Returns a list of all available user groups.
Terminate Session	<u>Security Items – TerminateSession</u>	Logs out the user that is currently logged in.
Update	<u>Security Items – UpdateUser</u>	Updates a user's information.
Global/Universal Items		
Add	<u>Global Items – CreateField</u>	Creates a new field.
Add	<u>Global Items – CreateCategory</u>	Creates a new category.

Add	<u>Global Items – CreateCategory with</u>	Creates a new category that includes fields.
Add	<u>Global Items – CreateCategoriesFromXML</u>	Creates a new category using an XML string.
Add	<u>Global Items – CreateProduct</u>	Creates a new product.
Delete	<u>Global Items – DeleteField</u>	Removes a field.
Delete	<u>Global Items – DeleteCategory</u>	Removes a category.
Delete	<u>Global Items – DeleteProduct</u>	Removes a product.
List	<u>Global Items – GetAllProductsList</u>	Gets an XML string of all products, including all the active and inactive product details.
List	<u>Global Items – GetCategoryXML</u>	Returns an XML string of all categories in a project.
List	<u>Global Items – GetDomainList</u>	Gets an XML string of all RateManager domains.
List	<u>Global Items – GetFieldsXML</u>	Returns an XML string of all fields in a project.
List	<u>Global Items – GetProgramList</u>	Gets an XML string of available programs in a selected folder.
List	<u>Global Items – GetProjectList</u>	Gets an XML string of all the projects in a domain.
List	<u>Global Items – GetReleases</u>	Returns a list of releases.
Updates	<u>Global Items – UpdateCategory</u>	Updates a category.
Updates	<u>Global Items – UpdateField</u>	Updates a field.
Update	<u>Global Items – UpdateProduct</u>	Updates a product.
Folder Groups		
Add	<u>Program Items – CreateFolderGroup</u>	Creates a new folder.
Delete	<u>Program Items – DeleteFolderGroup</u>	Removes a folder.
List Folders	<u>Program Items – GetAvailableFolderGroups</u>	Returns a list of folders in a product.
List Folders	<u>Program Items – GetAvailableFolderGroupsXML</u>	Returns an XML string of all available folders in a product.
List	<u>Program Items – GetSubFolderList</u>	Returns an XML string of available sub folders for a particular folder in a product of a project.
Update	<u>Program Items – UpdateFolderGroup</u>	Updates the name or location of a folder.
Programs		
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 Version	<u>Program Items – CopyProgramVersion</u>	Copies an existing Insbridge program version to a new specified version.
Delete	<u>Program Items – DeleteProgram</u>	Removes a program.
Delete	<u>Program Items – DeleteUnlockedProgramVersions</u>	Deletes multiple unlocked program versions of a program.
Update	<u>Program Items – UpdateProgram</u>	Updates a program.
Update	<u>Program Items – UpdateProgramVersionLock</u>	Updates a program version lock.
Update	<u>Program Items – UpdateProgramVersionName</u>	Add or updates a program version name.
Program Version Notes		
Add	<u>Program Items – AddProgramVersionNote</u>	Creates a new program version note.
Delete	<u>Program Items – DeleteProgramVersionNote</u>	Removes a program version note.
List	<u>Program Items – GetProgramVersionNotes</u>	Returns list of program version notes.
List	<u>Program Items – GetProgramVersionReportXML</u>	Returns an XML string of a program version report for a particular program version.
List	<u>Program Items – GetProgramVersions</u>	Returns an XML string of the available versions of a particular program.
Update	<u>Program Items – UpdateProgramVersionNote</u>	Updates a program version note.
Table Data		
Add	<u>Program Items – CreateTableVariableRevision</u>	Creates a new table revision.
Add	<u>Program Items – AddTableVariableRevisionNote</u>	Creates a table revision note.
Delete	<u>Program Items – DeleteTableVariableRevisionNotes</u>	Deletes a note from a table revision.
Export	<u>Program Items – ExportTableVariableData</u>	Export table variable data.
List	<u>Program Items – GetTableVariableRevisionNotes</u>	Returns a list of notes on a table revision.
Set	<u>Program Items – SetTableVariableActiveRevision</u>	Sets the table variable active revision.
Update	<u>Program Items – UpdateTableVariableRevisionNote</u>	Updates a note on a table revision.

SRP – RateManager		
Add	<u>SRP Items – CreateSRP</u>	Creates an SRP and returns the SRP object.
Environment Items		
List	<u>Environment Items – GetAvailableEnvironments Non-Secured</u>	Returns a list of all available non-secured environments.
List	<u>Environment Items – GetAvailableEnvironments, Selected</u>	Returns a list of all available environments within a selected environment type.
SRP– IBFA		
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.
Delete	<u>SRP Items – DeleteOrUnload</u>	Unload/delete multiple SRPs from a particular Environment.
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, ESI Created Only</u>	Returns a list of all ESI created SRPs in a select environment.
List	<u>SRP Items – GetSRPList, Selected SRP Type</u>	Returns a list of certain types of SRPs in a select environment.
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.
Releases		
Add	<u>Releases Items – CreateRelease</u>	Creates a new release.
Delete	<u>Releases Items – DeleteRelease</u>	Removes a release.
Delete	<u>Releases Items – DeleteAllReleases</u>	Removes all releases.
Delete	<u>Releases Items – DeleteMultipleReleases</u>	Removes multiple existing releases.
Deploy	<u>Releases Items – DeployRelease</u>	Deploys and loads a release in a selected environment.

List	<u>Releases Items – GetReleasePrograms</u>	Returns a list of all programs in release.
List	<u>Releases Items – GetReleaseProgramsPackageStatus</u>	Returns an XML string of all the programs of a release with package status.
Update	<u>Releases Items – AddReleaseProgram</u>	Add a program version to a release.
Package	<u>Releases Items – PackageReleasePrograms</u>	Creates a SoftRater Package (SRP) for a release.
Delete	<u>Releases Items – RemoveReleaseProgram</u>	Removes a program from the release.
Program Versioning Data		
Add	<u>Program Versioning Data – InsertProgramVersionDataRow</u>	Adds a row to program versioning data.
Add	<u>Program Versioning Data – InsertProgramVersionDataRows</u>	Adds multiple rows to program versioning data
Delete	<u>Program Versioning Data – DeleteProgramVersionDataRow</u>	Removes a row from program versioning data.
Update	<u>Program Versioning Data – UpdateProgramVersionDataCell</u>	Updates cell data in program versioning data.

QUICK GUIDE TABLE

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

Business Use Case	Methods to Use	Result
Input Batch		
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.
Result Batch		
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.
Job		
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 – createLaunchDefinition (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 – createLaunchDefinition (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 – createLaunchDefinition (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 – CreateCategory

- **Business Functionality**
 - Creates a category in a particular product.
- **Inputs**
 - ESI_Category object
- **Outputs**
 - Returns the ID of the newly created Category if succeeded or -1 if failed.

Global Items – CreateCategory with Fields

- **Business Functionality**
 - Creates a category with fields in a particular project.
- **Inputs**
 - ESI_Category object
 - List of ESI_Field objects
- **Outputs**
 - No outputs.

Global Items – CreateCategoriesFromXML

- **Business Functionality**
 - Creates one or more categories using an XML string.
- **Inputs**
 - None
- **Outputs**
 - Returns a List of ESI_Category.

Global Items – CreateField

- **Business Functionality**
 - Creates a new field in a particular project.
- **Inputs**
 - ESI_Field object
- **Outputs**
 - Returns the ID of the newly created field if succeeded or -1 if failed.

Global Items – CreateProduct

- **Business Functionality**
 - Creates a new product. The product is activated and available in RateManager. In order for any users to use the new product, groups must have access privileges granted.
- **Inputs**
 - ESI_PRODUCT object
- **Outputs**
 - Returns the ID of the newly created product if succeeded or -1 if failed.

Global Items – DeleteCategory

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

Global Items – DeleteField

- **Business Functionality**
 - Deletes a field in a particular project.
- **Inputs**
 - ESI_Field
- **Outputs**
 - True if successful or false if failed.

Global Items – DeleteProduct

- **Business Functionality**
 - Deletes an existing custom product.
- **Inputs**
 - ProductID
- **Outputs**
 - True if successful or exception if failed.

Global Items – GetAllProductsList

- **Business Functionality**
 - Gets an XML string of all products, including all the active and inactive product details.
- **Inputs**
 - ProjectID
- **Outputs**
 - Returns an XML string that includes all the products for that project. 1 for active and -1 for disabled and deactivated.

Global Items – GetCategoryXML

- **Business Functionality**
 - Gets an XML string of all the categories in a particular project.
- **Inputs**
 - ESI_Category object
- **Outputs**
 - Return an XML string.

Global Items – GetDomainList

- **Business Functionality**
 - Gets an XML string of all the RateManager domains.
- **Inputs**
 - ESI_Domain object
- **Outputs**
 - Returns an XML string that includes all domains.

Global Items – GetFieldsXML

- **Business Functionality**
 - Gets an XML string of all the fields of a particular project.
- **Inputs**
 - ProjectID
- **Outputs**
 - Returns an XML string that includes all the fields of that project.

Global Items – GetProgramList

- **Business Functionality**
 - Gets an XML string of all the programs in a selected folder.
- **Inputs**
 - ProjectID
- **Outputs**
 - Returns an XML string that includes all the programs of a particular folder.

Global Items – GetProjectList

- **Business Functionality**
 - Gets an XML string of all the projects in a domain.
- **Inputs**
 - ESI_Domain object
- **Outputs**

- Returns an XML string that includes all projects in a domain.

Global Items – GetReleases

- **Business Functionality**
 - Gets an XML string of all RateManager Releases.
- **Inputs**
 - No inputs are required.
- **Outputs**
 - Returns an XML string of all RateManager Releases.

Global Items – UpdateCategory

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

Global Items – UpdateField

- **Business Functionality**
 - Updates a field in a particular project.
- **Inputs**
 - ESI_Field
- **Outputs**
 - True if successful or false if failed.

Global Items – UpdateProduct

- **Business Functionality**
 - Updates an existing product: name, date mask or activation status.
- **Inputs**
 - ESI_PRODUCT object
- **Outputs**
 - Returns "True" if succeeded or "False" if failed.

Program Items – AddProgramVersionNote

- **Business Functionality**
 - Creates a new note on the selected program version.
- **Inputs**
 - ESI_Program object

- **Outputs**
 - ESI_Program object.

Program Items – AddTableVariableRevisionNote

- **Business Functionality**
 - Adds a new note on the selected table revision.
- **Inputs**
 - ESI_Program object
 - ESI_Table object
- **Outputs**
 - ESI_Table object.

Program Items – CopyProgramVersion

- **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 ESI_ProgramVersion object.

Program Items – CreateFolderGroup

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

Program Items – CreateProgram

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

Program Items – CreateProgramVersion

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

Program Items – CreateTableVariableRevision

- **Business Functionality**
 - Creates a new revision of an existing table variable.
- **Inputs**
 - EsiTableVariableRevision object - The Table Variable Revision where the new revision is being created. For global and universal table variables, set both programId and programVersion to 0.
 - includeData- True if table data needs to be copied from the selected revision to the new revision, or false if no data is being copied.
 - setActiveRev- True if Table Variable Revision needs to be set as active revision, or false otherwise.
- **Outputs**
 - Returns True if succeeded or exception if failed

Program Items – DeleteFolderGroup

- **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 are now unlinked.
- **Inputs**
 - ESI_FolderGroup object
- **Outputs**
 - Returns True if succeeded or False 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**
 - ESI_Program object
- **Outputs**
 - Returns a pass/fail value. Pass = true and Fail = false.

Program Items – DeleteProgramVersionNote

- **Business Functionality**
 - Deletes a note on the selected program version.
- **Inputs**
 - ESI_Program object
- **Outputs**
 - Returns a pass/fail value. Pass = true and Fail = false.

Program Items – DeleteTableVariableRevisionNotes

- **Business Functionality**
 - Deletes a note from a particular table variable revision.
- **Inputs**
 - ESI_Program object
- **Outputs**
 - Deletes a note from a particular table variable revision.

Program Items – DeleteUnlockedProgramVersions

- **Business Functionality**
 - Deletes multiple unlocked program versions of a program.
- **Inputs**
 - EsiProgram - the program with the program versions to be deleted.
 - progVers - the program versions array which needs to be deleted.
- **Outputs**
 - Returns a pass/fail value. Pass = true and Fail = false.

Program Items – ExportTableVariableDatatoText

- **Business Functionality**
 - Exports the table variable revision data for a selected table variable. For global and universal table variables, set TableVarRevision.ProgramId to 0. If the SaveLocation is not provided, the default location is the Insbridge Framework temp folder.
- **Inputs**
 - EsiTableVariableRevision - the Table Variable Revision where data is being exported.
 - SaveLocation - the absolute path and filename where the data is being exported to.
- **Outputs**
 - Returns a tab delimited file.

Program Items – ExportMultipleTableVariableDataToExcel

- **Business Functionality**
 - Exports one or more table variables data to an Excel spreadsheet.
- **Inputs**
 - **FileToSave** - file path and name where the exported file is being saved.
 - **NewSheetName** - desired sheet name in the Excel file to export one TV and it is null to export multiple table variables, the sheet names will be save with table variable name.
 - **isOverwriteMode** - set to true to delete existing file and create a new one, or set to false to create new sheet in existing file.
 - **isWildcardSymbolForNull** - set to true to export null/blank cells for wildcard qualifiers as "*"
 - **ProjectID** - the project id of the table variable.
 - **ProgramID** - the program id of the table variable .
 - **tableVarIDs** - the string array of table variable id's.
 - **tableVarRevisions** - the string array of table variables revision number.
 - **storedProcFetchSize** - the database page size.
 - **isLoggingOn** - set to true to see detailed data export logging. The default is false.
 - **LogFileDir** - a directory where log files will be saved if isLoggingOn is set to true.
 - **SortString** - the table data sort column SQL string.
 - **FilterString** - the table data filter SQL string.
 - **Port** – the database server port number.
- **Outputs**
 - Returns true if succeeded or exception if failed.

Program Items – ImportExcelDatatoTableVariable

- **Business Functionality**
 - This method imports excel data to single or multiple table variables active revisions or selected revisions.
- **Inputs**
 - **FilePath** - file path and name where the imported file is being saved.
 - **ProjectID** - the project id of the table variable.
 - **ProductID** - the product id of the table variable.
 - **ProgramID** - the program id of the table variable.
 - **ProgVer** - the program version of the table variable id.
 - **DeleteAllRows** - True to delete data before inserting, else False.
 - **tablevarIDs** - the string array of table variable ids.
 - **tableVarRevs** – the table variable revisions array, null to import data to the active revisions of table variables else the string array of table variable revisions to import data to that selected revision.
 - **SheetType** - the Excel file type: Insbridge or Custom.
 - **Port** – the database server port number
- **Outputs**
 - Returns Success message or exception if failed.

Program Items – GetAvailableFolderGroups

- **Business Functionality**
 - Gets a list of Folders of a particular product.
- **Inputs**
 - ProductID
- **Outputs**

- Returns a list of ESI_FolderGroup objects.

Program Items – GetAvailableFolderGroupsXML

- **Business Functionality**
 - Gets an XML string of available Folders of a particular product.
- **Inputs**
 - ProductID
- **Outputs**
 - Returns a table with available Folders of a specific product.

Program Items – GetProgramVersionNotes

- **Business Functionality**
 - Gets a list of notes on a selected program version.
- **Inputs**
 - ESI_Program object
- **Outputs**
 - Returns a list of notes for a specific program version.

Program Items – GetProgramVersionReportXML

- **Business Functionality**
 - Gets an XML string of a Program Version Report for a particular Program Version.
- **Inputs**
 - ESI_Program object
- **Outputs**
 - Returns an XML string of a Program Version Report of a particular Program Version.

Program Items – GetProgramVersions

- **Business Functionality**
 - Gets an XML string of the available versions of a particular program.
- **Inputs**
 - ESI_Program object
- **Outputs**
 - Returns an XML string of the available versions of a particular program.

Program Items – GetSubFolderList

- **Business Functionality**
 - Gets an XML string of available sub folders for a particular folder in a product of a project.
- **Inputs**

- ProductID
- **Outputs**
 - Returns an XML string of available sub folders for a particular folder in a product of a project.

Program Items – GetTableVariableRevisionNotes

- **Business Functionality**
 - Gets a list of notes on a selected table revision. For global and universal table variables, set TableVarRevision.ProgramId to 0.
- **Inputs**
 - ESI_Program object
 - ESI_Table object
- **Outputs**
 - Returns a list of notes for a specific table revision.

Program Items – SetTableVariableActiveRevision

- **Business Functionality**
 - Set the active revision of an existing table variable.
- **Inputs**
 - EsiTableVariableRevision object - The Table Variable Revision where the active revision is set
 - parentProgID - For local table variable the parentProgID = program ID of local TV and for global and universal TV the parentProgID = 0
- **Outputs**
 - Returns True if succeeded or exception if failed.

Program Items – UpdateFolderGroup

- **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**
 - ESI_FolderGroup object
- **Outputs**
 - Returns the updated Folder ID if succeeded or -1 if failed.

Program Items – UpdateProgram

- **Business Functionality**
 - Updates a Program in the folder of your choosing. If culture, decimal symbol and group symbol are not provided, they will default to "en-US", "." and ",".
- **Inputs**
 - ESI_Program
- **Outputs**

- True if successful or exception if failed.

Program Items – UpdateProgramVersionLock

- **Business Functionality**
 - Locks or unlocks a program version.
- **Inputs**
 - ESI_ProgramVersion
 - LockOption
- **Outputs**
 - True if successful or exception if failed.

Program Items – UpdateProgramVersionName

- **Business Functionality**
 - Adds or updates the program version name.
- **Inputs**
 - ESI_ProgramVersion
- **Outputs**
 - True if successful or exception if failed.

Program Items – UpdateProgramVersionNote

- **Business Functionality**
 - Updates a note on the selected program version.
- **Inputs**
 - ESI_Program object
- **Outputs**
 - Returns a pass/fail value. Pass = true and Fail = false.

Program Items – UpdateTableVariableRevisionNote

- **Business Functionality**
 - Updates a note on the selected table revision.
- **Inputs**
 - ESI_Program object
 - ESI_Table object
- **Outputs**
 - Returns a pass/fail value. Pass = true and Fail = false.

Releases Items – AddReleaseProgram

- **Business Functionality**

- Add a program version or global versioning to an existing Release.
- **Inputs**
 - EsiRelease - the release the program version or global versioning is added to.
 - EsiProgramVersion - the program version or global versioning that is being added. To add global versioning set parentProgramId= 0 and versionNumber = 0.
- **Outputs**
 - True if successful or exception if failed.

Releases Items – CreateRelease

- **Business Functionality**
 - Creates a new Release.
- **Inputs**
 - ReleaseName – the release name
 - ReleaseDescription – the release description
 - ReleaseNotes – the release notes
 - AdminEmail – the release admin email
 - TargetEnvironment – the environment
- **Outputs**
 - Esi_Release object.

Releases Items – DeleteAllReleases

- **Business Functionality**
 - Deletes all existing unlocked releases.
- **Inputs**
 - No inputs are required to delete all unlocked releases.
- **Outputs**
 - True if successful or exception if failed.

Releases Items – DeleteMultipleReleases

- **Business Functionality**
 - Deletes multiple existing unlocked releases.
- **Inputs**
 - List<EsiRelease> - the list of releases that are being deleted
- **Outputs**
 - True if successful or exception if failed.

Releases Items – DeleteRelease

- **Business Functionality**
 - Deletes an existing release.
- **Inputs**
 - ReleaseID – the release being deleted
 - ReleaseName – the release name
 - ReleaseDescription – the release description

- ReleaseNotes – the release notes
- AdminEmail – the release admin email
- **Outputs**
 - True if successful or exception if failed.

Releases Items – DeployRelease

- **Business Functionality**
 - Deploys and loads a release in a selected environment.
- **Inputs**
 - EsiRelease - The Release to be Deployed.
 - environment - The environment the release is to be deployed to.
 - envType -The environment type the release is to be deployed to.
 - loadRelease -True if release needs to be loaded to the selected environment, else false.
- **Outputs**
 - Returns a success message or exception if failed.

Releases Items – GetReleasePrograms

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

Releases Items – GetReleaseProgramsPackageStatus

- **Business Functionality**
 - Gets an XML string of all the programs in a release. If the program is successfully packaged then it returns the SRP ID value i.e., "srp_package_id". If the programs is not packaged then the SRP ID attribute value is empty and there will be log info.
- **Inputs**
 - ESI_Release – the object being retrieved.
- **Outputs**
 - Returns the xml string of release programs.

Releases Items – PackageReleasePrograms

- **Business Functionality**
 - Creates a SoftRater Package (SRP) for the Release programs.
- **Inputs**
 - EsiRelease - The release, with a list of release programs where SRPs need to be created, that needs to have multiple programs packaged. To package all programs in the release, only the release name and release ID are required.
 - packageAllProgs - True if all programs in the release need to be package else false.

- **Outputs**
 - Success message or exception if failed.

Releases Items – RemoveReleaseProgram

- **Business Functionality**
 - Removes a program version or global versioning from an existing Release.
- **Inputs**
 - EsiRelease - the release the program version or global versioning is removed from.
 - EsiProgramVersion - the program version or global versioning that is being removed. To remove global versioning set parentProgramId = 0 and versionNumber = 0.
- **Outputs**
 - True if successful or exception if failed.

Security Items – CreateUser

- **Business Functionality**
 - Creates a new User.
- **Inputs**
 - ESI_User object
- **Outputs**
 - Returns True if succeeded or False if failed.

Security Items – GetAvailableUsers

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

Security Items – GetAvailableUserGroups

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

Security Items – TerminateSession

- **Business Functionality**
 - Logs out the current user.
- **Inputs**
 - ESI_User object
- **Outputs**
 - Returns True if succeeded or False if failed.

Security Items – UpdateUser

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

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**
 - ESI_Program object
 - Program Version
 - ESI_Environment object
- **Outputs**
 - Returns the newly created ESI_SRP object.

Program Versioning Data – InsertProgramVersionDataRow

- **Business Functionality**
 - Inserts a new row in program versioning data.
- **Inputs**
 - ProjectID
 - ProductID
 - ProgID
 - ProgVer
 - RowID
- **Outputs**
 - True if successful or exception if failed.

Program Versioning Data – InsertProgramVersionDataRows

- **Business Functionality**
 - Inserts multiple rows in program versioning data.

- **Inputs**
 - ProjectID
 - ProductID
 - ProgID
 - ProgVer
 - RowID
- **Outputs**
 - True if successful or exception if failed.

Program Versioning Data – DeleteProgramVersionDataRow

- **Business Functionality**
 - Deletes a row in program versioning data.
- **Inputs**
 - ProjectID
 - ProductID
 - ProgID
 - ProgVer
 - RowID
- **Outputs**
 - True if successful or exception if failed.

Program Versioning Data – UpdateProgramVersionDataCell

- **Business Functionality**
 - Updates a cell value in program versioning data.
- **Inputs**
 - ProjectID
 - ProductID
 - ProgID
 - ProgVer
 - RowID
 - ColumnID
- **Outputs**
 - True if successful or exception if failed.

NOTE: *Versioning Criteria must be in place to view changes in cell values.*

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**
 - The environment the SRP to be copied to.

- The SRP to be copied.
- **Outputs**
 - True if succeeded or False 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**
 - The SRP to be deleted
- **Outputs**
 - True if succeeded or False if failed

SRP Items – DeleteOrUnload

- **Business Functionality**

Unload or delete multiple SRPs from a particular Environment.
To determine that the SRPs are unloaded/deleted from, run the GetSRPList to get the ESI_SRP's
- **Inputs**
 - List<EsiSRP> - The list of SRP's to be unloaded/deleted.
 - type - the type of action, whether it's unload or delete.
- **Outputs**
 - True if succeed or False 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**
 - The environment the SRP is to be exported
 - The SRP to be exported
- **Outputs**
 - The bytes of the SRP

SRP Items – GetSRPList, ESI Created Only

- **Business Functionality**
 - Gets a list of all ESI created SRPs in a particular environment.
- **Inputs**
 - The environment from where the SRPs are to be retrieved.
- **Outputs**
 - A list of ESI_SRP objects.

SRP Items – GetSRPList, Selected SRP Type

- **Business Functionality**
 - Gets a list of SRPs including Releases SRPs of certain type in a particular environment.
- **Inputs**
 - The environment from where the SRPs are to be retrieved.
 - SRP type: 0 = all SRPs; 1 = ESI created SRPs.
- **Outputs**
 - A list of ESI_SRP objects.

SRP Items – ImportSRP

- **Business Functionality**
 - Imports SRP bytes into an IBFA environment.
- **Inputs**
 - The environment the SRP is to be imported into
 - SRP bytes to be imported
 - SRP file name to be saved as
- **Outputs**
 - True if succeeded or False 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**
 - The environment the SRP is to be loaded in.
 - The SRP to be loaded.
- **Outputs**
 - True if succeeded or False 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**
 - The environment the SRP is to be moved to.
 - The SRP to be moved.
- **Outputs**
 - True if succeeded or False if failed.

SRP Items – SaveFile

- **Business Functionality**
 - Saves the bytes you pass in to a physical file.
 - It gives ASP.NET Machine Account ("ASPNET") full control of that file.
- **Inputs**
 - The bytes to be saved
 - The location and name the bytes are to be saved
- **Outputs**
 - True if succeeded or False 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 will be unloaded before the new package is loaded.
- **Inputs**
 - The SRP to be unloaded.
- **Outputs**
 - True if succeed or False 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.

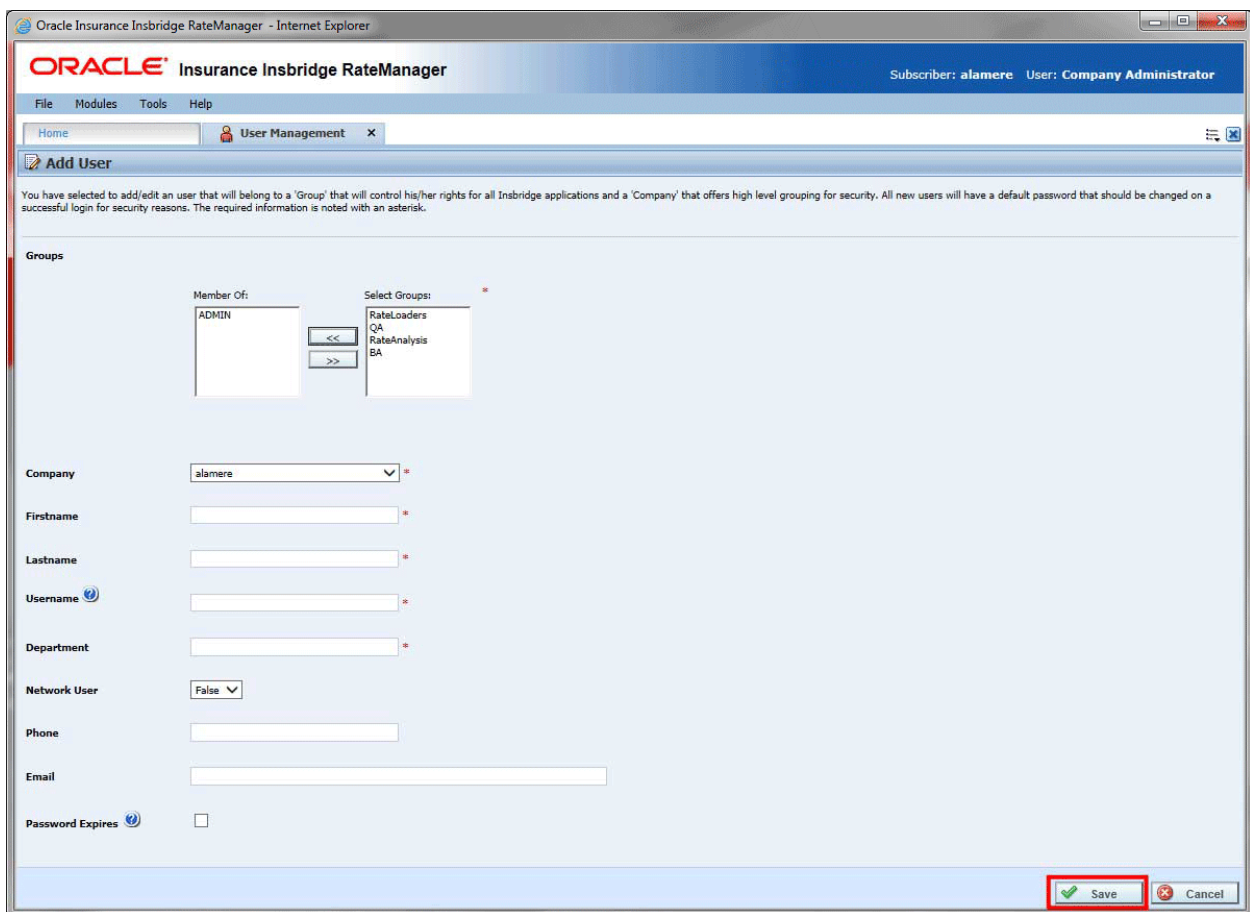


Figure 1 Adding a New User

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.

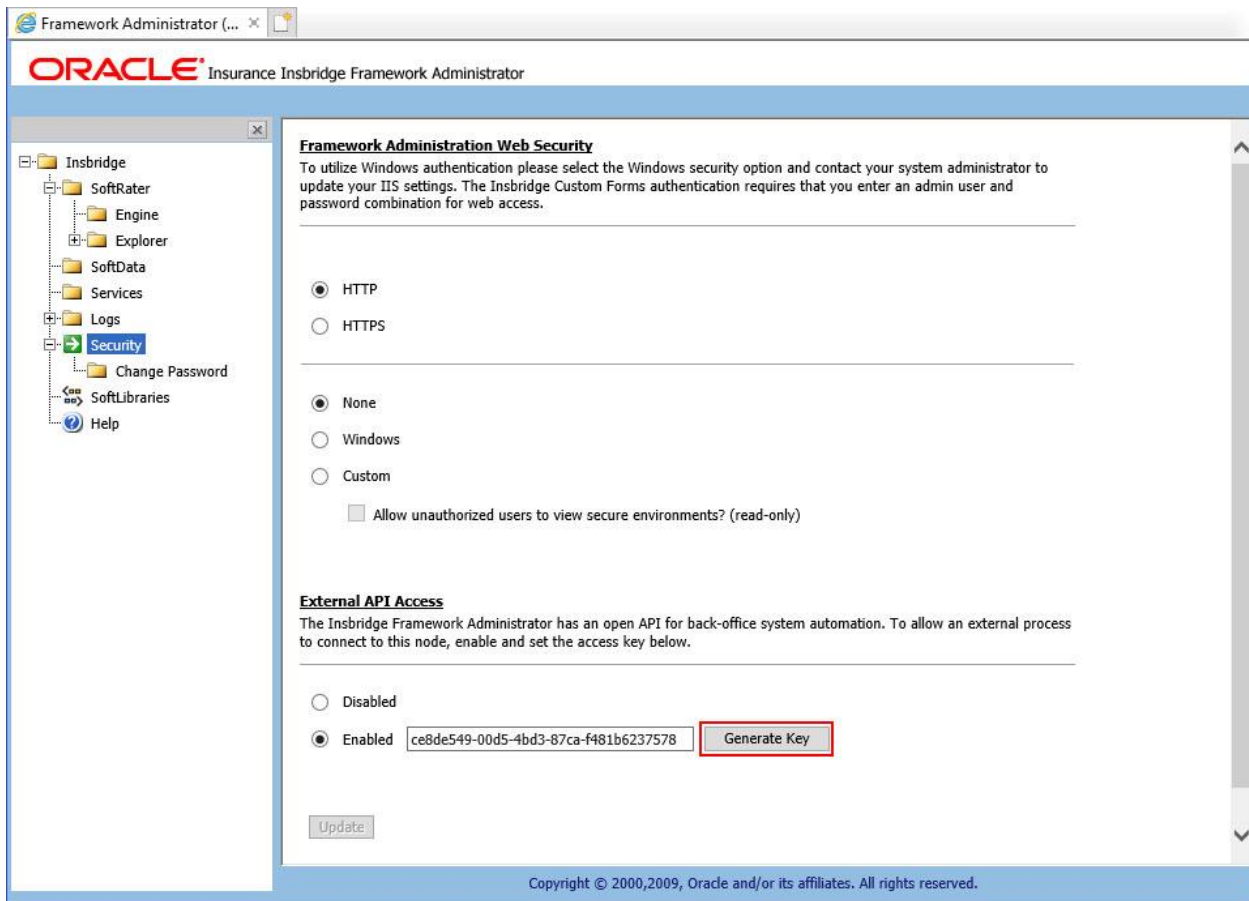


Figure 2 Allowing External API Access

ESI JAVA TESTER CONFIGURATION CHANGES

Configuration changes must be implemented prior to running the ESI Tester. The ESI Java Tester runs on Windows.

Prerequisites

- JAVA_HOME Location on the machine where the ESI Java Tester will be used. The JAVA_HOME location cannot have spaces in the path.

– For example: C:\Java\jre1.8.0_171

If the JAVA_HOME path has a space, (For example, C:\Program Files\Java\jdk1.8.0_171) an error will be displayed when starting.

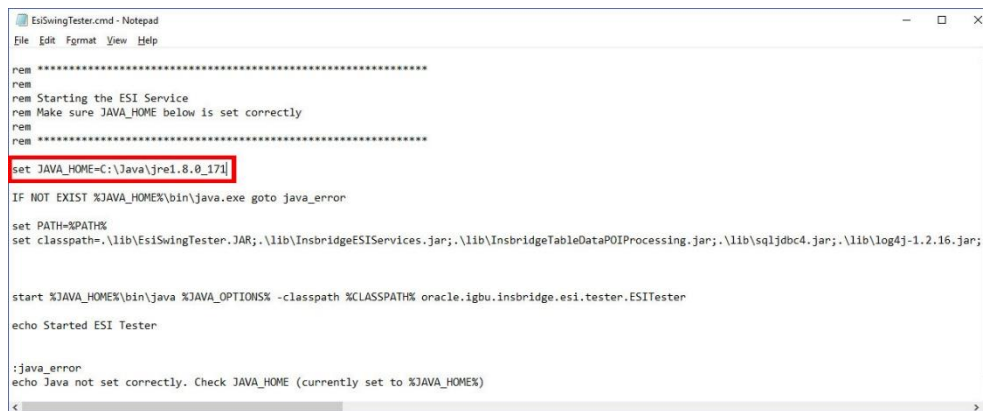
- URL or name and port number of the IBSS server that will be used.
- Zip utility

1. Copy the Esi Java Tester files from the server where Insbridge was installed to your local machine.

[INSTALL_DIR]\Insbridge\SDK\ESI\Java\lib.

Where [INSTALL_DIR] is the location of the server where Insbridge was installed.

2. In the ESI Java Tester file, select the EsiSwingTester.cmd.
3. Right click and select Edit.
4. Update the JAVA_HOME location to the location where the tester will be used.



```
EsiSwingTester.cmd - Notepad
File Edit Format View Help

rem *****
rem
rem Starting the ESI Service
rem Make sure JAVA_HOME below is set correctly
rem *****
set JAVA_HOME=C:\Java\jre1.8.0_171
IF NOT EXIST %JAVA_HOME%\bin\java.exe goto java_error
set PATH=%PATH%
set classpath=.\lib\EsiSwingTester.JAR;.\lib\InsbridgeESIServices.jar;.\lib\InsbridgeTableDataPOIProcessing.jar;.\lib\sqljdbc4.jar;.\lib\log4j-1.2.16.jar;

start %JAVA_HOME%\bin\java %JAVA_OPTIONS% -classpath %CLASSPATH% oracle.igbu.insbridge.esi.testner.ESITester
echo Started ESI Tester

:java_error
echo Java not set correctly. Check JAVA_HOME (currently set to %JAVA_HOME%)
```

5. Save your entry.

ESI JAVA TESTER

The ESI Tester is user interface for working with ESI. The ESI tester runs on a local Windows machine. The Java version of ESI allows for IBSS testing as well as RateManager.

INSTALLATION

List of Files

Following are the list of files needed to test ESI

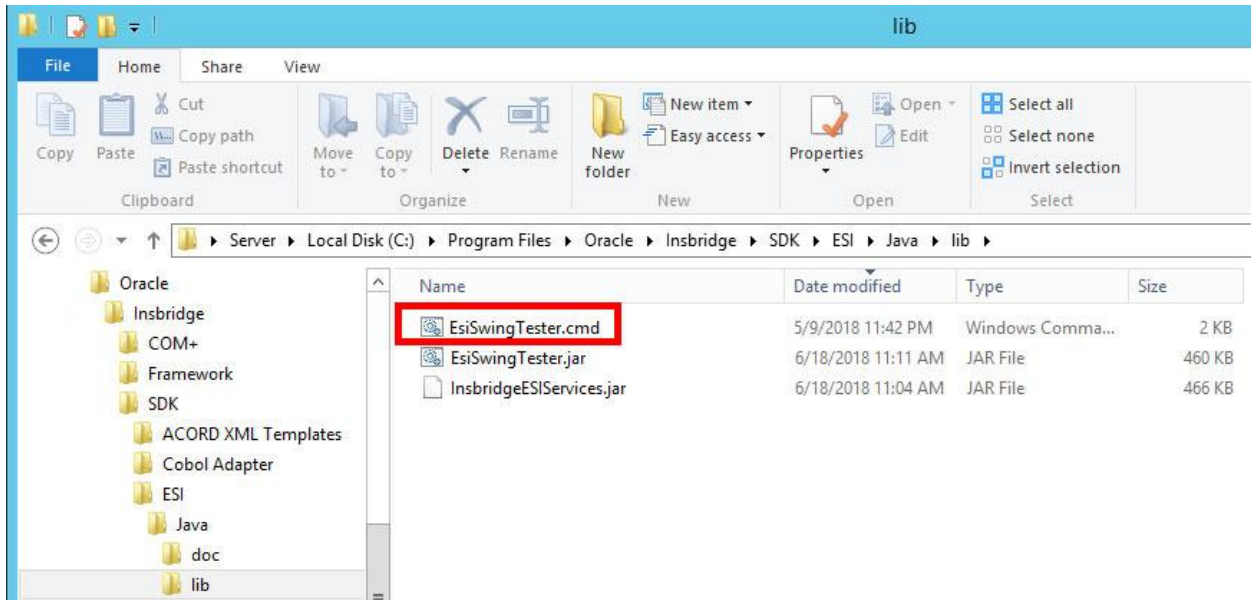
File	Usage
ESISwingTester.cmd	A Windows command file to launch the tester.
ESISwingTester.jar	ESI Tester executable.
InsbridgeESIServices.jar	ESI Tester executable.

Prerequisites

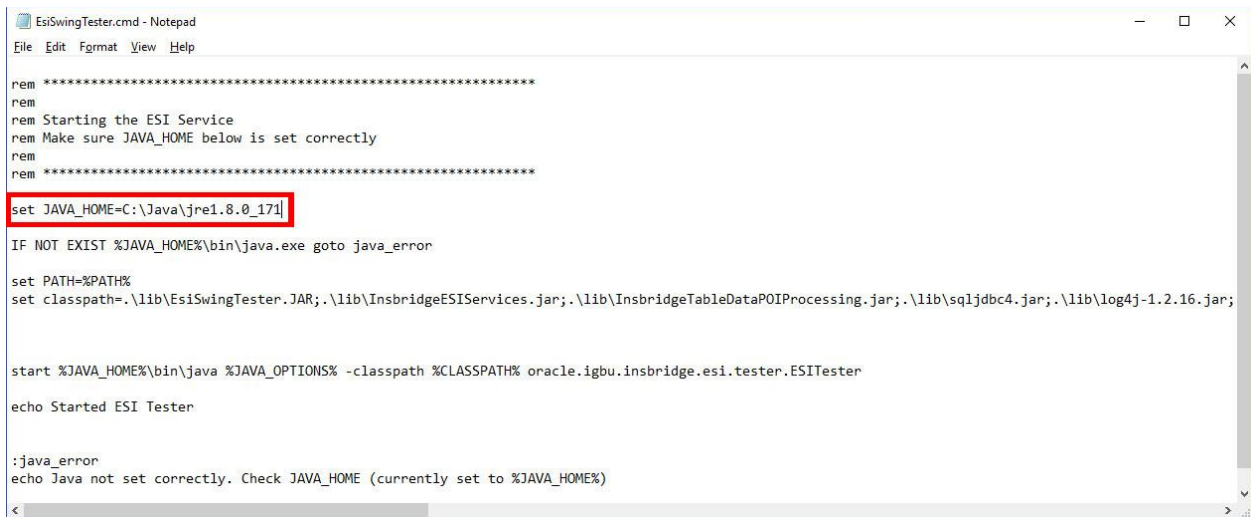
- ESI for Java requires Java 1.8. Java must be parked in a location where the path does not contain spaces. For example, C:\Java\jre1.8.0_171
- The URL and port number for the IBSS instance where you want to connect.
- Unzip utility for editing the Tester.jar file.
- Access to the server where RateManager was installed.
- 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

Entering Java Location

1. Locate the ESI files on the server where Insbridge was installed. For Example:
C:\Program Files\Oracle\INSBRIDGE\SDK\ESI\Java\lib
2. Copy files to your local Windows machine.
3. The Java location needs to be entered in the ESISwingTester.cmd file.



4. Right click to edit.

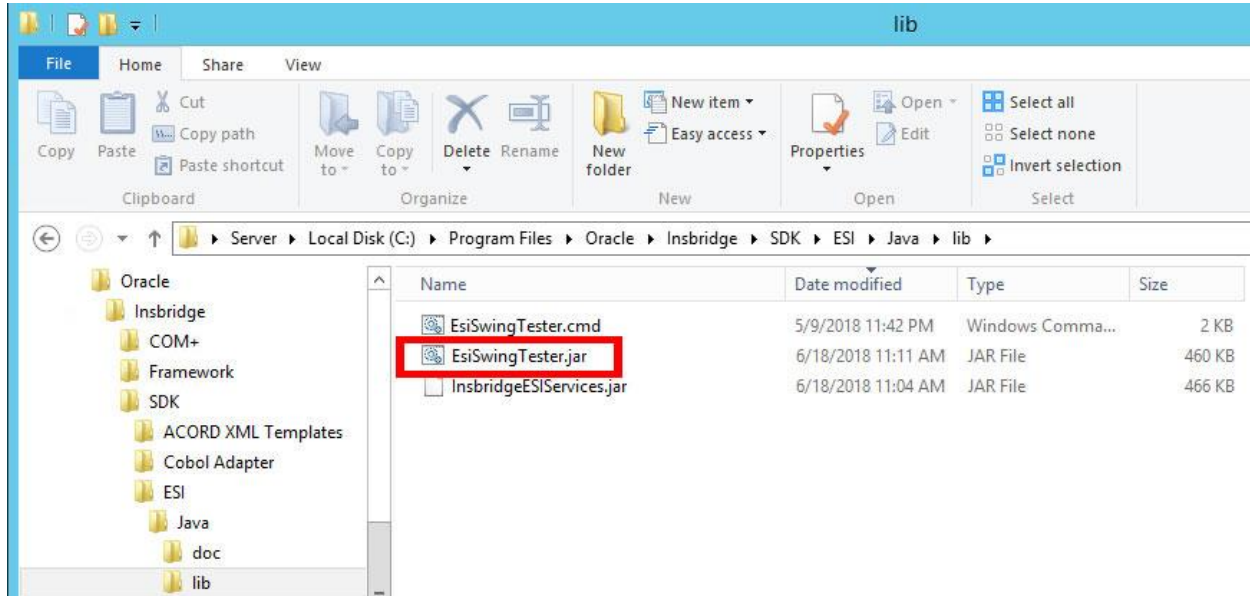


5. Enter the location of Java. The location cannot contain any spaces.

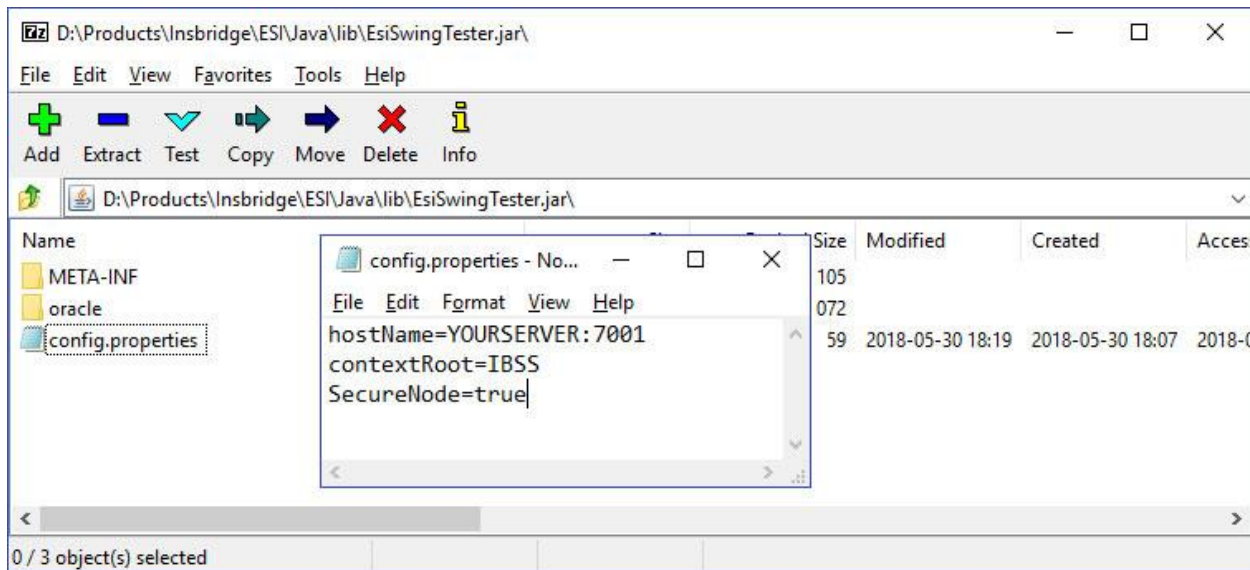
6. Save and close.

Entering the IBSS Instance

If you want to utilize the features for IBSS, you must enter the instance in the EsiSwingTester.jar file. You will need an unzip program to open the jar file.



1. Right click the EsiSwingTester.jar file and open the archive.



2. Locate the config file. Right click to edit.
3. Enter in the location:port for the IBSS instance you want to use.
4. Enter in a SecureNode value. True will connect using HTTPS. False connects using HTTP.
5. Save and close.

Using ESI on Application Server

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

Launching ESI Tester

To launch the ESI Java Tester, launch the .cmd file. Select the option you want to test.

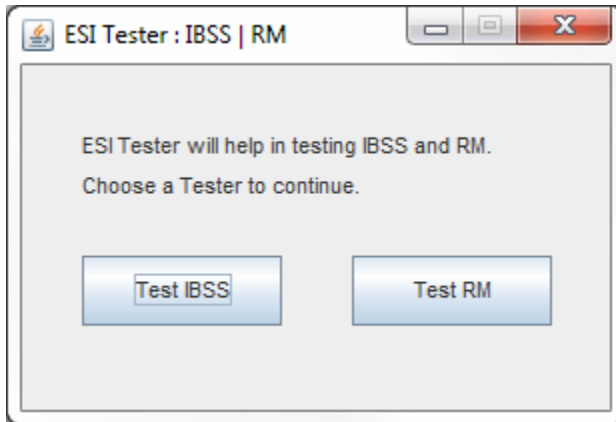


Figure 3 Selecting Instance to Test

Status is displayed in the upper corner. Offline indicates that webservices 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.

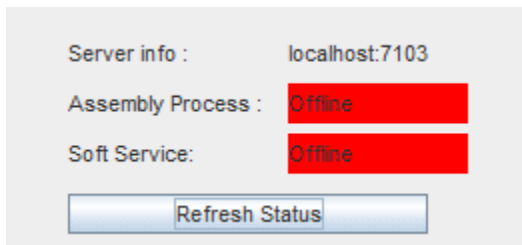


Figure 4 Connection Status

TEST IBSS FEATURES/METHODS

The IBSS features.

- Insbridge XML
- Batch
- SoftRater

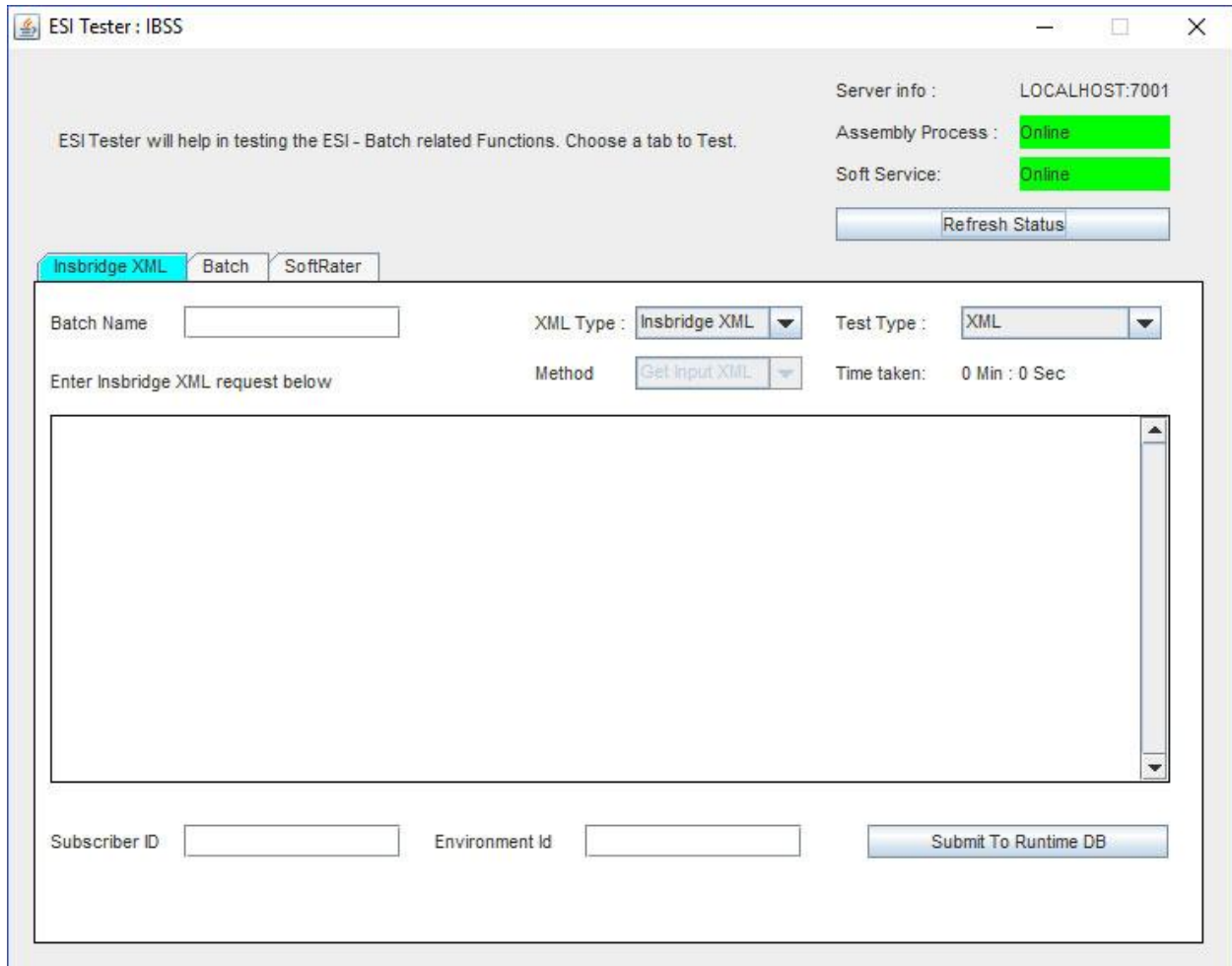


Figure 5 Testing IBSS

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 ID and environment Id. Click Submit to Runtime DB.

ESI IBSS TESTING

ESI Tester : IBSS

Server info : LOCALHOST:7001

Assembly Process : Online

Soft Service: Online

Refresh Status

ESI Tester will help in testing the ESI - Batch related Functions. Choose a tab to Test.

Insbridge XML | Batch | SoftRater

Batch Name:

XML Type : Insbridge XML

Test Type : XML

Enter Insbridge XML request below

Method : Get Input XML

Time taken: 0 Min : 0 Sec

Subscriber ID: Environment Id:

Submit To Runtime DB

Figure 6 IBSS ESI Tester Landing Page

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

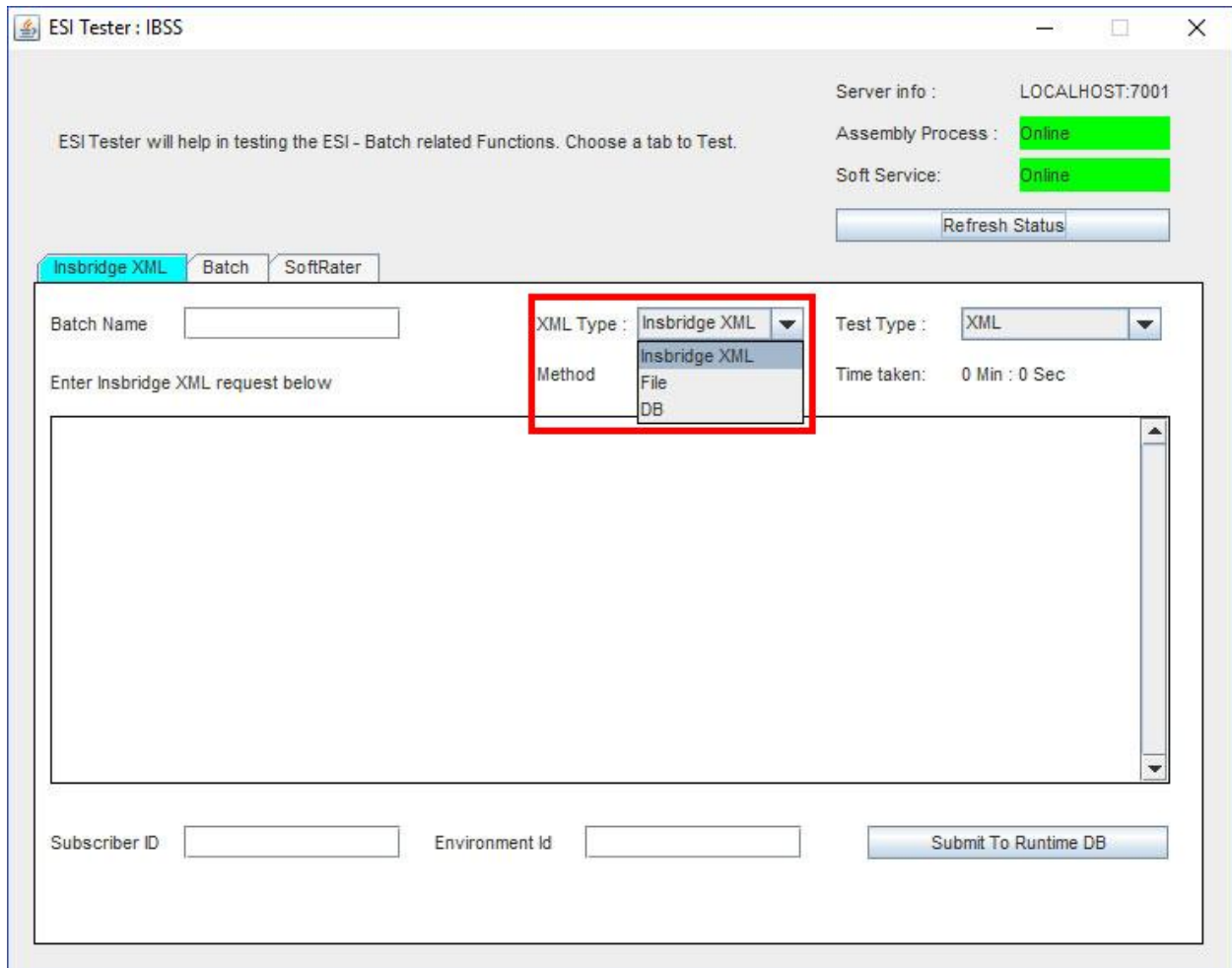


Figure 7 IBSS ESI Tester Insbridge XML

ESI currently supports the 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

ESI Tester will help in testing the ESI - Batch related Functions. Choose a tab to Test.

Server info : LOCALHOST:7001
 Assembly Process : Online
 Soft Service : Online

Refresh Status

Insbridge XML **Batch** SoftRater

Submit a batch Request :

Batch Id
 Subscriber Id
 Environment Id
 File Ref Id
 Read Write Option
 Result Batch Id

Choose the Type of Notifications you like to Receive

Email Notification
 File Notification
 Queue Notification

Seperate Emails with ','
 Location of Notifying File :
 JMS Queue to Look up :
 Process Type Batch

Start Batch Show Request XML

Monitor Batch here.

Enter Request ID Stop Batch Get Batch Status

Figure 8 IBSS Tester Batch

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

SOFTRATER

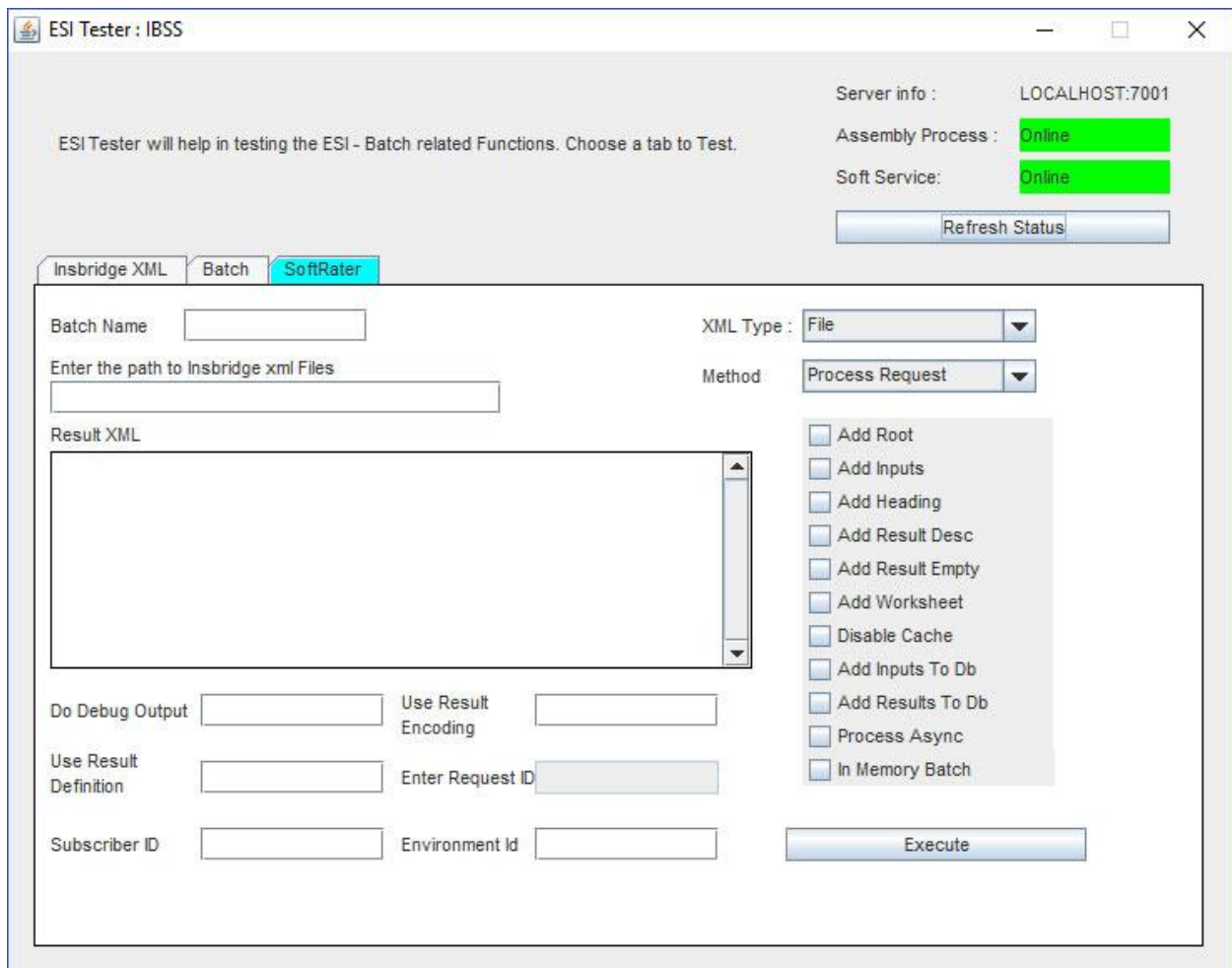


Figure 9 IBSS ESI Tester for SoftRater

Functions in the same way as SoftRater on IBSS. Please see the IBSS Guide for more information.

RM FEATURES/METHODS

- Test Security
- Test Globals/Universals
- Test Program Items
- Test IBFA
- Test Releases
- Test List Items
- Test Versioning Data

Login

A RateManager user ID and password can be used or a network login can be used. The RateManager service defaults to localhost and the first subscriber found in the Insbridge config file. This information can be updated. Enter the RateManager server you need and the subscriber you need.

The screenshot shows the 'ESI Tester: RM | IBFA' application window. At the top, there are input fields for 'Rate Manager Server' (localhost) and 'Subscriber ID' (8659), with a 'Network Login' button. Below these, the 'User Name' (melliott) and 'Password' (masked with asterisks) fields are highlighted with a red rectangle, with a 'Login' button next to them. The main area of the window is divided into tabs: 'Test Security', 'Test Glb/Uni', 'Test Program Items', 'Test IBFA' (selected), 'Test Releases', 'Test List Items', and 'Test Versioning Data'. Under the 'Test IBFA' tab, there are fields for 'IBFA Token' and 'SRP: C:\TEMP\'. To the right of these fields are buttons for 'Get Env' and 'Import SRP'. Below the SRP field is a large empty rectangular box. To the right of this box are dropdown menus for 'Environment' and 'SRP Type' (set to 'All'), and a 'Get SRP List' button. At the bottom of the main area, there are several buttons: 'Load SRP', 'Move SRP', 'Export SRP', 'Unload SRP', 'Copy SRP', 'Delete SRP', 'Add SRP(s) to Unload / Delete', 'Unload SRP's', and 'Delete SRP's'. A 'Close' button is located in the bottom right corner of the window.

Figure 10 Logging Into ESI

Test Security

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

Figure 11 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

NOTE: The Close option at the bottom of the ESI window closes the ESI tester. To navigate, click the tab where you want to test.

Test Globals/Universals

Allows for the adding, editing and deleting of Products, Categories, and Fields at the global or universal level.

The screenshot shows the 'ESI Tester: RM | IBFA' application window. At the top, there are login fields: 'Rate Manager Server' (localhost), 'Subscriber ID' (8659), 'User Name' (melliott), and 'Password' (*****). Buttons for 'Network Login', 'Login', and 'Logout' are present. Below the login section is a tabbed menu with 'Test Glib/Uni' selected. The main content area contains four input fields: 'Product ID', 'Project ID', 'Cat ID', and 'Field ID'. Below these are several buttons: 'Create Product', 'Create Category', 'Create Field', 'Create Category w/ Fields', 'Get Category XML', 'Update Product', 'Update Category', 'Update Field', 'Get Fields XML', 'Delete Product', 'Delete Category', and 'Delete Field'. A 'Close' button is located at the bottom right of the window.

Figure 12 Test Globals/Universals

Get requests require a Project ID.

- Get Category XML returns the details of the categories in the selected project.
- Get Fields XML returns the details of the fields in the selected project.

NOTE: The ESI tester does create users and other items, some of the values used are place holders that can be edited in RateManager.

Test Program Items

Allows for viewing, adding, editing, copying and deleting of program items. Also allows for the exporting table variable data and setting of active table revisions.

The screenshot shows the 'ESI Tester: RM | IBFA' application window. At the top, there are fields for 'Rate Manager Server' (localhost), 'Subscriber ID' (8659), 'User Name' (melliott), and 'Password' (*****). Below these are 'Network Login', 'Login', and 'Logout' buttons. The main interface has a tabbed menu with 'Test Program Items' selected and highlighted in red. The main area contains several sections:

- Database and File Settings:** Fields for DB Server, RM DB Catalog, DB Port (1433), DB User ID, DB Password, Log Dir, Save to file, Sheet name/Type, Overwrite Existing File (No), Table Var ID, Prog ID, Product ID, ProgVer, Wildcard Symbol, Data Fetch Size, Table Var Rev, and Delete Data checkbox.
- Export/Import:** Buttons for 'Export Table Data to Excel' and 'Import Excel Data to TV'. A blue button labeled '*import d...' is also present.
- Program Management:** A grid of buttons including 'Get Folder List', 'Create Folder', 'Update Folder', 'Delete Folder', 'Update Prog Version Name', 'Create Program', 'Update Program', 'Copy Program', 'New Program Ver', 'Lock Program', 'Unlock Program', 'Delete Program', 'Delete Unlocked Program Versions', 'Get Available Environments', 'Create SRP', 'New Table Var Revision', 'Set as Active Revision', 'Launch Program (Network User)', 'Launch Program (Non-Network User)', 'Get Program Version Notes', and 'Get Table Revision Notes'.
- Program Details:** Input fields for Project ID, Product ID, Folder ID, Prog ID, Prog Ver, Prog Name, Ver Name, Note ID, Note Text (200 Max), Export TV Data to, Export Data to Text, Parent Prog ID, TV Name, TV ID, and TV Rev.
- Notes:** Buttons for 'Add Program Ver Note', 'Update Program Ver Note', 'Delete Program Ver Note', 'Add Table Rev Note', 'Update Table Rev Note', and 'Delete Table Rev Note'.

 A 'Close' button is located at the bottom right of the window.

Figure 13 Test Program Items

- Get Available Environments returns the environments available. The list is populated drop down.
- Get Folder List requires a Product ID and returns a list of folder IDs in the Project.
- Get Program Version Notes returns notes for a specific program.
- Get Table Revision Notes returns the notes for a specific Table in a specific Project.
- Export Table Var Data exports data from the specified table revision in tab delimited format to a default or named location. The location can be specified in the Export TV Data to: field.
- Launch Program (Network User) launches RateManager for users with network login credentials.

- Launch Program (Non-network User) launches RateManager for users not using network login credentials.

Test IBFA

Allows for SRPs to be managed.

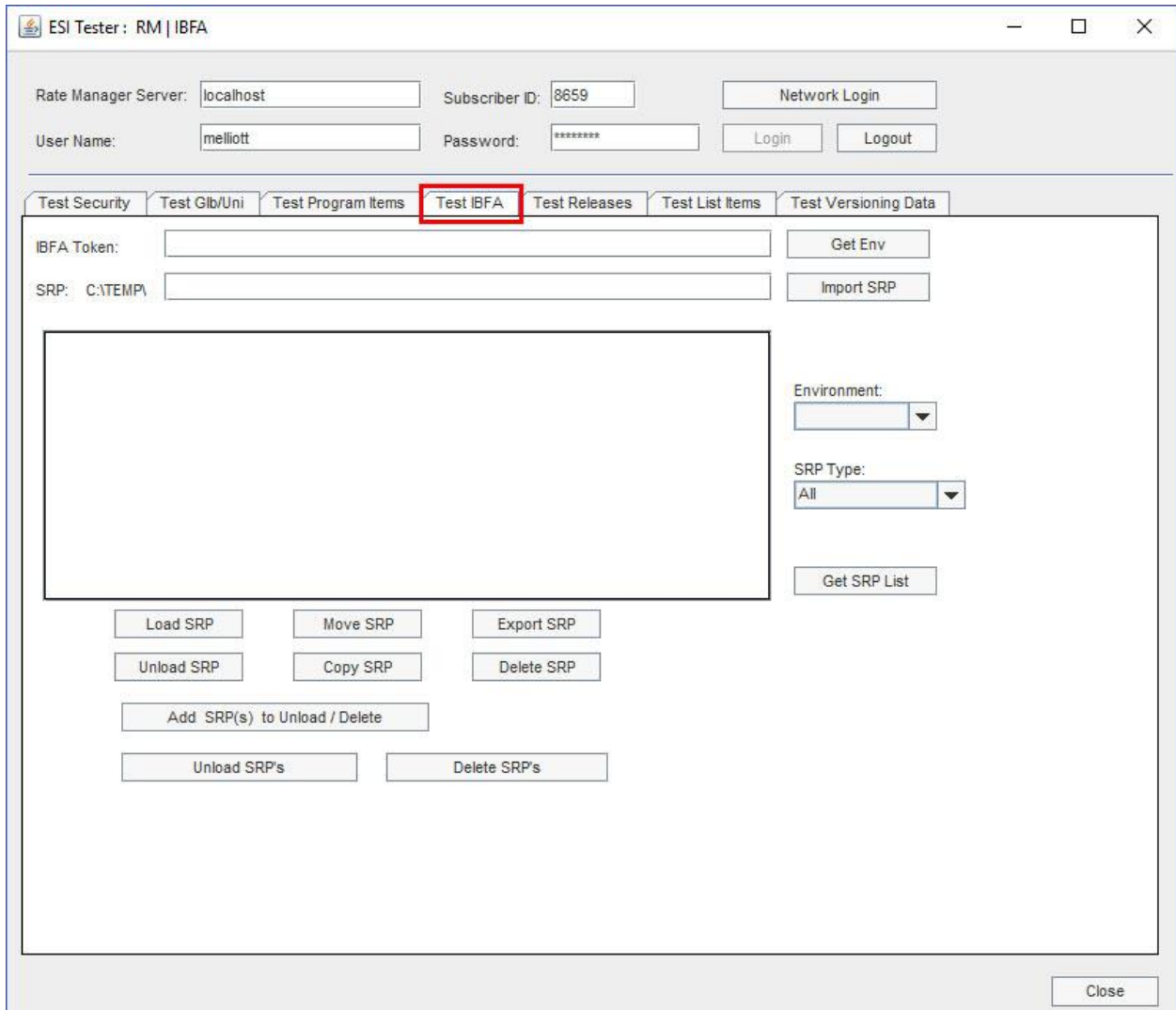


Figure 14 Test IBFA

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

Test Releases

Allows for the management of releases.

ESI Tester: RM | IBFA

Rate Manager Server: localhost Subscriber ID: 8659 Network Login

User Name: mellott Password: ***** Login Logout

Test Security Test Glb/Uni Test Program Items Test IBFA **Test Releases** Test List Items Test Versioning Data

Release Name: Project ID:

Release Description: Product ID:

Release Notes: Program ID:

Admin Email: Program Ver:

Environment: Release ID:

Get Environments

Create Release Add Program to Release Add Global Versioning to Release

Delete Release Remove Program from Release Remove Global Versioning from Release

Add Release(s) to Delete Get Release Program

Delete All Releases Add Program(s) to Package Get Release Programs Package Status

Delete Multiple Releases Package Select All Programs to Package

Environment Type: **Development** Deploy Load Release to Selected Environ...

Close

Figure 15 Test Releases

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

Test List Items

Figure 16 Java ESI Tester for RM

- Return a list all active products
- Return a list of domains
- Return a list of projects for a domain
- Return a list of products for a particular project
- Return a list of folders for a particular product
- Return a list of sub folders for a particular folder
- Return a list of programs for a folder
- Return a list of program versions for a particular program
- Return the XML for the program version report for a particular program version
- Return the location of a program version report
- Return a list of releases

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.

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 ALL PRODUCTS

- **Input:** No input is required.
- **GET ALL PRODUCTS:** Returns an XML string of all products. Active products have an active status of 1. Inactive or deactivated products have an active status of -1.
- **Output:** Returns an XML string of available Products. XML string has:
 - PRODUCT_NAME
 - PRODUCT_ID
 - ACTIVE attributes
 - DATE MASK

GET DOMAIN LIST

- **Input:** No input is required.
- **GET DOMAIN LIST:** Returns an XML string of all domains.
- **Output:** Returns an XML string of available domains. XML string has:
 - DOMAIN_NAME
 - DOMAIN_ID

GET PROJECT LIST

- **Inputs:** Domain ID.
- **GET PROJECT LIST:** Gets an XML string of available projects within the domain ID.
- **Outputs:** Returns an XML string that includes all the projects of that domain. XML string has:
 - PROJECT_NAME
 - PROJECT_ID
 - DATE_LAST_MODIFIED

GET PRODUCT LIST

- **Inputs:** Project ID.
- **GET PROJECT LIST:** Gets an XML string of available products within the project ID.
- **Outputs:** Returns an XML string that includes all the products of that project. XML string has:
 - PRODUCT_NAME
 - PRODUCT_ID
 - DATE_MASK
 - CULTURE
 - DECIMAL_SYMBOL
 - GROUP_SYMBOL
 - ABBR (Abbreviation used)

GET FOLDER LIST

- **Inputs:** Project ID and Product ID.
- **GET FOLDER LIST:** Gets an XML string of available folders of a particular product.
- **Outputs:** Returns the XML string that includes all the folders of that product. XML String has:
 - PROJECT_ID
 - PRODUCT_ID
 - FOLDER_ID
 - FOLDER_NAME

GET SUB FOLDER LIST

- **Inputs:** Project ID, Product ID and Folder ID.
- **GET SUB FOLDER LIST:** Gets an XML string of available sub folders of a particular product folder/ folder.
- **Outputs:** Returns an xml string that includes all the sub folders of that product Folder/Folder. XML string has:
 - PROJECT_ID
 - PRODUCT_ID
 - FOLDER_ID
 - FOLDER_NAME
 - FOLDER_PARENT_ID

GET PROGRAM LIST

- **Inputs:** Project ID, Product 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:
 - PROJECT_ID
 - PRODUCT_ID
 - FOLDER_ID
 - PROGRAM_ID
 - PROGRAM_NAME

GET PROGRAM VERSIONS LIST

- **Inputs:** Project ID, Product 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:

- PROJECT_ID
- PRODUCT_ID
- FOLDER_ID
- PROGRAM_ID
- VERSION

GET PROGRAM VERSION REPORT

- **Inputs:** Project ID, Product ID, Folder ID, Program ID, Program Ver and Program Name.
- **GET PROGRAM VER REPORT:** 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:** Project ID, Product 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.*

GET RELEASES

- **Input:** No input is required.
- **GET ALL PRODUCTS:** Returns an XML string of all releases.
- **Output:** Returns an XML string of all releases. XML string has:
 - RELEASES_ID
 - RELEASES_NAME
 - LOCK_STATUS

Output

Output is in XML format. In this example, All Products are listed. Active products have an active status of 1. Inactive or deactivated products have an active status of -1.

```

<ROWDATA><Node PRODUCT_NAME="Artisan" PRODUCT_ID="21" ACTIVE="1"
DATE_MASK="YYYY-MM-DD" /><Node PRODUCT_NAME="Auto"
PRODUCT_ID="1" ACTIVE="1" DATE_MASK="YYYY-MM-DD" /><Node
PRODUCT_NAME="Automated Underwriting" PRODUCT_ID="12" ACTIVE="1"
DATE_MASK="YYYY-MM-DD" /><Node PRODUCT_NAME="Boiler And
Machinery" PRODUCT_ID="17" ACTIVE="1" DATE_MASK="YYYY-MM-DD"
/><Node PRODUCT_NAME="BOP" PRODUCT_ID="4" ACTIVE="1"
DATE_MASK="YYYY-MM-DD" /><Node PRODUCT_NAME="Commercial Auto"
PRODUCT_ID="11" ACTIVE="1" DATE_MASK="YYYY-MM-DD" /><Node
PRODUCT_NAME="Commercial Inland Marine" PRODUCT_ID="29" ACTIVE="-1"
DATE_MASK="YYYY-MM-DD" /><Node PRODUCT_NAME="Commercial
Package" PRODUCT_ID="30" ACTIVE="1" DATE_MASK="YYYY-MM-DD"
/><Node PRODUCT_NAME="Commercial Property" PRODUCT_ID="15"
ACTIVE="1" DATE_MASK="YYYY-MM-DD" /><Node PRODUCT_NAME="Crime
And Fidelity" PRODUCT_ID="31" ACTIVE="1" DATE_MASK="YYYY-MM-DD"
/><Node PRODUCT_NAME="Cyber" PRODUCT_ID="102" ACTIVE="1"
DATE_MASK="MM/DD/YYYY" /><Node PRODUCT_NAME="Directors And
Officers" PRODUCT_ID="13" ACTIVE="1" DATE_MASK="YYYY-MM-DD" /><Node
PRODUCT_NAME="Disability" PRODUCT_ID="10" ACTIVE="-1"
DATE_MASK="YYYY-MM-DD" /><Node PRODUCT_NAME="Errors and
Omissions" PRODUCT_ID="101" ACTIVE="1" DATE_MASK="MM/DD/YY"
/><Node PRODUCT_NAME="General Liability" PRODUCT_ID="14" ACTIVE="1"
DATE_MASK="YYYY-MM-DD" /><Node PRODUCT_NAME="Home"
PRODUCT_ID="2" ACTIVE="1" DATE_MASK="YYYY-MM-DD" /><Node
PRODUCT_NAME="Medical" PRODUCT_ID="19" ACTIVE="1"
DATE_MASK="YYYY-MM-DD" /><Node PRODUCT_NAME="Package"
PRODUCT_ID="18" ACTIVE="1" DATE_MASK="YYYY-MM-DD" /><Node
PRODUCT_NAME="Pet" PRODUCT_ID="103" ACTIVE="1"
DATE_MASK="MM/DD/YY" /><Node PRODUCT_NAME="PowerSport"
PRODUCT_ID="104" ACTIVE="0" DATE_MASK="MM/DD/YYYY" /><Node
PRODUCT_NAME="Public" PRODUCT_ID="100" ACTIVE="0"
DATE_MASK="MM/DD/YY" /><Node PRODUCT_NAME="Rules"
PRODUCT_ID="99" ACTIVE="1" DATE_MASK="YYYY-MM-DD" /><Node
PRODUCT_NAME="Umbrella" PRODUCT_ID="6" ACTIVE="1"
DATE_MASK="YYYY-MM-DD" /><Node PRODUCT_NAME="Valuables"
PRODUCT_ID="7" ACTIVE="1" DATE_MASK="YYYY-MM-DD" /><Node
PRODUCT_NAME="Watercraft" PRODUCT_ID="8" ACTIVE="-1"
DATE_MASK="YYYY-MM-DD" /><Node PRODUCT_NAME="Workers Comp"
PRODUCT_ID="20" ACTIVE="1" DATE_MASK="YYYY-MM-DD" /></ROWDATA>

```

Figure 17 Tester Output

Test Versioning Data

Allows for the testing of program versioning data.

The screenshot shows the 'ESI Tester: RM | IBFA' application window. At the top, there are fields for 'Rate Manager Server' (localhost), 'Subscriber ID' (8659), 'User Name' (melliott), and 'Password' (*****). There are 'Network Login', 'Login', and 'Logout' buttons. Below this is a tabbed interface with tabs for 'Test Security', 'Test Glb/Uni', 'Test Program Items', 'Test IBFA', 'Test Releases', 'Test List Items', and 'Test Versioning Data'. The 'Test Versioning Data' tab is selected and highlighted with a red box. The main area contains a grid of 10 input fields for Project ID, Product ID, Prog ID, Row ID, and 10 qualifiers (Q1-Q10). Below the grid are 'Insert' and 'Delete Row' buttons. At the bottom, there is an 'Update Data Cell' section with a dropdown for 'Column ID / Qualifier Number', a 'Data' input field, and an 'Update Cell' button. A 'Close' button is located in the bottom right corner.

Figure 18 Test Versioning Data

Up to 10 qualifiers can be entered.

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

INDEX

A

- Add Batch to Runtime DB
 - InputBatch, 48
- Add Category Inputs
 - Rate or Result, 52
- Add Child Category
 - Rate or Result, 52
- Add Parent Category
 - Rate or Result, 52
- Add Program Details
 - Rate or Result, 52
- Add Rating Details
 - Rate or Result, 52
- Add Release Program, 38
- Adding
 - New User in User Management, 53
- API Key, 55
- Authentication
 - ESI User, 53

B

- Base ESI Domain Types, 17
- Business Use Case, 18, 22

C

- Category
 - Create, 27
 - Create from XML, 27
 - Create with Fields, 27
 - Delete, 29
 - Get, XML, 30
 - Update, 31
- Class Definition
 - IBFA, 44
 - RateManager Interface, 25
- Company
 - Selecting for New User, 53
- Constructor Detail
 - RateManager Interface, 25
- Copy Definitions & All Data
 - Copying Programs, 32
- Copy Definitions Only
 - Copying Programs, 32
- Copy Program Version, 32
- Copy SRP, 44
- Create, 50
 - Network Login, 53
- Create Category, 27
 - From XML, 27
 - With Fields, 27
- Create Field, 28

- Create Folder Groups, 32
- Create Product, 29
- Create Program, 32
- Create Program Version, 33
- Create Program Version Note, 31
- Create Release, 39
- Create SRP, 42
- Create Table Revision, 33
- Create Table Revision Note, 32
- Create User, 41
- CreateNew Rate
 - InputBatch, 48
- Creating
 - InputBatch, 15
 - Rate Object, 13

D

- Database Type
 - IBSS Tester, 66
- Default
 - API Key, 55
- Delete All Releases, 39
- Delete Category, 29
- Delete Folder Groups, 33
- Delete Multiple Releases, 39
- Delete Product, 29
- Delete Program, 33
- Delete Program Version Note, 34
- Delete Program Versioning Data, 43
- Delete Release, 39
- Delete SRP, 45
- Delete Table Variable Revision Notes, 34
- Delete Unlocked Program Versions, 34
- Deletes Field, 29
- Deploy Release, 40
- Domain
 - Get, List XML, 30

E

- Edition Notice, 2
- Environments
 - Get Available, Non-Secured, 25
 - Get Available, Secured, 26
- ESI
 - Overview, 10
 - Rules, 11
- ESI Domain Types, 17
- Example
 - Business Use Case, 18, 22
- ExecuteBatch
 - InputBatch, 48
- Executing
 - InputBatch, 16

Export Multiple Table Data Excel, 34
 Export SRP, 45
 Export Table Revision Data, 34

F

Fail to Delete, 33
 False
 New Users, 54
 Fields
 Create New, 28
 Delete, 29
 Get, XML, 30
 Update, 31
 Folder
 Create, 32
 Delete, 33
 Get Available, 35
 Get Available, XML, 36
 Get Sub List, 36
 Update, 37
 Functionality Highlights
 ESI, 12

G

Get All Products
 Test List Items, 79
 Get Available
 Environments, Non-Secured, 25
 Environments, Secured, 26
 User Groups, 41
 Users, 41
 Get Available Folder Groups, 35
 Get Available Folder Groups XML, 36
 Get Category XML, 30
 Get Domain List
 Test List Items, 79
 Get Domain List XML, 30
 Get Fields XML, 30
 Get Folder List
 Test List Items, 80
 Get Product List
 Test List Items, 80
 Get Program List
 Test List Items, 80
 Get Program List XML, 30
 Get Program Version List
 Test List Items, 81
 Get Program Version Notes, 36
 Get Program Version Report
 Test List Items, 81
 Get Program Version Report Location
 Test List Items, 81
 Get Program Version Report XML, 36
 Get Program Versions, 36
 Get Project List
 Test List Items, 79
 Get Project List XML, 30
 Get Release Program Package Status, 40
 Get Release Programs, 40
 Get Releases, 31

 Test List Items, 82
 Get SRP List
 All, 46
 Selected, 46
 Get Sub Folder List
 Test List Items, 80
 Get SubFolder List, 36
 Get Table Revision Notes, 37
 GetAllProductsList XML, 29
 Group
 Selecting for New User, 53

I

IBFA
 Class Definition, 44
 System Configuration, 55
 IBRequestResponse Interface, 13
 IBSS Tester, 62
 Database Type, 66
 Test Batch, 69
 Test Insbridge XML, 63
 Test SoftRater, 70
 Import Excel Table Data, 35
 Import SRP, 46
 InputBatch
 Add Batch to Runtime DB, 48
 Create New Rate, 48
 Creating, 15
 Execute Batch, 48
 Executing, 16
 Monitoring, 16
 Retrieve Input Rate, 48
 Retrieving Inputs, 16
 Submit Insbridge XML Files from Disk, 49
 Submit Insbridge XML from Memory, 49
 Submit Rate Object to Runtime DB, 49
 Submitting, 16
 InputObject, 13
 Insert Multiple Rows Program Versioning Data, 42
 Insert Rows Program Versioning Data, 42

J

Job
 Retrieve Job Status, 51
 Stop the Job, 51

L

Launch Definition
 First Version, 26, 27
 Selected Version, 26
 Load SRP, 46
 Locked Program
 Fail to Delete, 33
 Login
 RateManager Tester, 71

M

Monitoring
 InputBatch, 16
Move SRP, 47

N

Network Login
 ESI Authentication, 53
Network User
 New User, 53
New User in User Management
 Adding, 53
Note
 Create Program Version, 31
 Create Table Revision, 32
 Delete Program Version, 34
 Update Program Version, 38
 Update Table Revision, 38
Notes
 Get Program Version, 36
 Get Table Revision, 37

O

Objects
 Rate, 13
Output
 Test List Items, 83
Overview
 ESI, 10

P

Package Release Programs, 40
Password
 Default, 53
Password Expires
 New User, 54
Product
 Create, 29
 Delete, 29
 Update, 31
ProductsList
 Get All, XML, 29
Program
 Copy Definitions & All Data, 32
 Copy Definitions Only, 32
 Copy Version, 32
 Create, 32
 Create New Version, 33
 Delete, 33
 Delete Unlocked Versions, 34
 Get Version Report, 36
 Get Versions, 36
 Get, List XML, 30
 Update, 37
 Update Version Lock, 38
 Update Version Name, 38
Program Versioning Data
 Delete, 43

 Insert Multiple, 42
 Insert Row, 42
 Update, 43
Project
 Get, List XML, 30

Q

Quick Guide for ESI, 23

R

Rate Model for IBSS, 12
Rate Object
 Creating, 13
Rate Objects, 13
Rate or Result
 Add Category Inputs, 52
 Add Child Category, 52
 Add Parent Category, 52
 Add Program Details, 52
 Add Rating Details, 52
RateManager Login
 Create Launch First Version, 26, 27
 Create Launch Selected Version, 26
 ESI Authentication, 53
RateManager Tester, 71
 Login, 71
 Test Globals Universals, 73
 Test IBFA, 76
 Test List Items, 78
 Test Program Items, 74
 Test Releases, 77
 Test Security, 72
 Test Versioning Data, 84
Release
 Add Program, 38
 Create, 39
 Delete, 39
 Delete All, 39
 Delete Multiple, 39
 Deploy, 40
 Get, 31
 Get Program Package Status, 40
 Get Programs, 40
 Package Programs, 40
 Remove Program, 41
Remove Release Program, 41
Result Batch
 Create New Result, 50
 Retrieve Results, 50
 Retrieving Results, 16
Retrieve, 50, 51
Retrieve Input Rates
 InputBatch, 48
Retrieving Inputs
 InputBatch, 16
Retrieving Results, 16
 Result Batch, 16
Rules
 ESI, 11
 Username, 53

S

Save File SRP, 47
Sessions
 Terminate, 41
Set Table Variable Active Revision, 37
SRP
 Copy, 44
 Create, 42
 Delete, 45
 Export, 45
 Get List, All, 46
 Get List, Selected, 46
 Import, 46
 Load, 46
 Move, 47
 Save File, 47
 Unload, 47
 Unload or Delete, 45
Stop, 51
Submit Insbridge XML Files from Disk
 InputBatch, 49
Submit Insbridge XML from Memory
 InputBatch, 49
Submit Rate Object to Runtime DB
 InputBatch, 49
Submitting
 InputBatch, 16
System Configuration
 IBFA, 55
System Requirements, ix

T

Table Variable
 Delete Revision Notes, 34
Tables
 Create New Revision, 33
 Export Multiple Excel Table Data, 34
 Export Table Revision Data, 34
 Import Excel Table Data, 35
 Set Active Revision, 37
Terminate Session, 41
Test Batch
 IBSS Tester, 69
Test Globals Universals
 RateManager Tester, 73
Test IBFA
 RateManager Tester, 76
Test Insbridge XML
 IBSS Tester, 63
Test List Items
 Get All Products, 79
 Get Domain List, 79
 Get Folder List, 80

 Get Product List, 80
 Get Program List, 80
 Get Program Version List, 81
 Get Program Version Report, 81
 Get Program Version Report Location, 81
 Get Project List, 79
 Get Releases, 82
 Get Sub Folder List, 80
 Output, 83
 RateManager Tester, 78
Test Program Items
 RateManager Tester, 74
Test Releases
 RateManager Tester, 77
Test Security
 RateManager Tester, 72
Test SoftRater
 IBSS Tester, 70
Test Versioning Data
 RateManager Tester, 84
Tester, 61
True
 New Users, 54

U

Unload SRP, 47
UnloadOrDelete SRP, 45
Update Category, 31
Update Field, 31
Update Folder Groups, 37
Update Product, 31
Update Program, 37
Update Program Version Lock, 38
Update Program Version Name, 38
Update Program Version Note, 38
Update Program Versioning Data, 43
Update Table Revision Note, 38
Update User, 42
User
 Create New, 41
 Network User for New, 53
 Selecting Company for New, 53
 Selecting Group for New, 53
 Update, 42
 Username for New, 53
User Authentication, 54
User Groups
 Get Available, 41
Username
 New User, 53
 Rules, 53
Users
 Get Available, 41
 Password Expires, 54