Ikasan Dashboard User Manual

Version 1.1.0

EMAIL: info@ikasan.org

WEBSITE: www.ikasan.org

***Table of Contents***

1 Introduction 4

1.1 About 4

1.2 Audience 4

1.3 How to Use This Guide 4

1.4 Overview 4

2 Accessing the Ikasan Dashboard 5

3 Platform Setup 8

4 Menu Navigation 10

5 General 11

5.1 User Profile 11

5.2 Dashboard 12

6 Services 13

6.1 Topology View 13

6.1.1 Topology Tree 15

6.1.1.1 Populating the Topology Tree 17

6.1.1.2 Categorised Errors 18

6.1.1.3 Start up Control 20

6.1.1.4 Component Configuration 21

6.1.1.5 Wiretap Configuration 24

6.1.2 Business Stream Tab 26

6.1.2.1 Creating a New Business Stream 26

6.1.3 Wiretap Tab 28

6.1.3.1 Wiretap Details Window 29

6.1.4 Errors Tab 30

6.1.4.1 Error Occurrence Details Window 31

6.1.5 Exclusions Tab 32

6.1.5.1 Exclusion Event Details Window 33

6.1.6 Actioned Exclusions Tab 35

6.1.6.1 Actioned Exclusion Event Details Window 36

6.1.7 Systems Events Tab 37

6.1.8 Categorised Errors Tab 38

6.1.8.1 Categorised Error Occurrence Details Window 39

6.1.9 Filters Tab 41

6.1.10 Mapping Actions 43

6.1.10.1 Creating a New Mapping Client 43

6.1.10.2 Creating a New Mapping Configuration Type 43

6.1.10.3 Creating a New Mapping Configuraiton Context 44

6.1.10.4 Creating a New Mapping 44

6.1.10.5 Importing a Mapping Configuration 45

6.1.11 Mapping Configuration Search 46

6.1.12 Managing a Mapping Configuration 47

6.2 Monitoring View 49

7 Administration 51

7.1 User Administration 51

7.2 Group Administration 53

7.3 Role Administration 54

7.4 Policy Administration 56

7.5 Managing User Directories 58

7.6 Platform Configuration 60

7.7 Notifications 61

7.7.1 Configuring a Notification 62

# Introduction

## About

This document provides a user guide for the Ikasan Dashboard. This is part of the documentation suite for the Ikasan Enterprise Integration Platform.

This console is independently deployed as a self-contained WAR to the runtime container. This guide assumes packaging and deployment to runtime has been completed and focuses purely on the operations of the console itself from initial first time usage through to administration and management.

## Audience

This guide is targeted at users wishing to track and view business events flowing over one or more Ikasan integration modules as well as Ikasan associated services.

## How to Use This Guide

This guide provides a reference that can be read front to back or dipped into as required for the operation of the Ikasan Dashboard.

## Overview

The Ikasan User Console provides a web front end for searching and tracking events passing through Ikasan Integration Modules.

This guide covers the user operations within this console including:

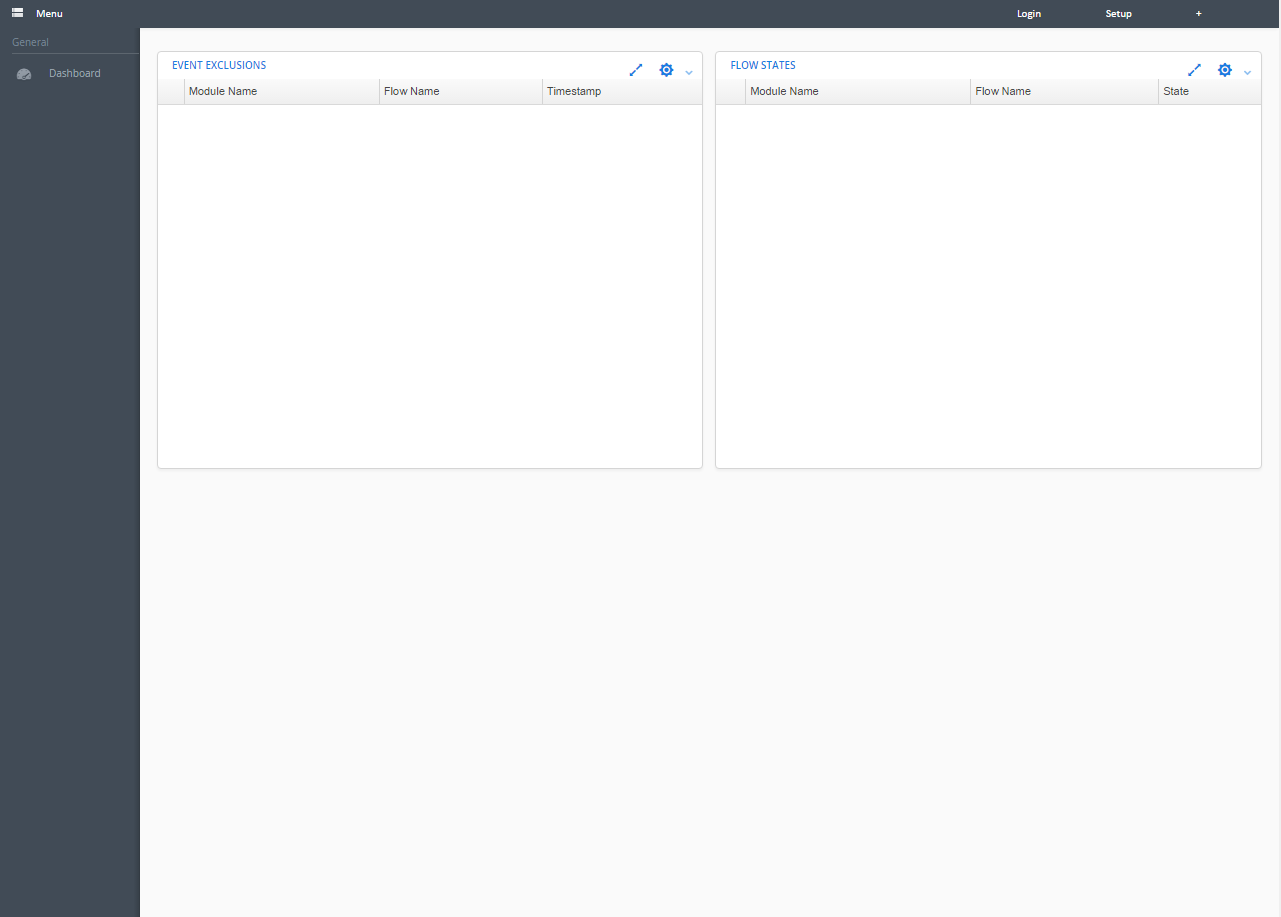
* Event Searching and Tracking
* User Administration and Management
* Ikasan Support and Resources
* Security and access management
* Data mapping
* Error and event exclusion lifecycle
* Business stream control
* Monitoring

# Accessing the Ikasan Dashboard

The Ikasan Dashboard is bound to the root context of “ikasan-dashboard” at runtime. In order to access the Ikasan User Console deployed to server “*foo*” (assuming default port 8080) you would open a web browser specifying the URL:

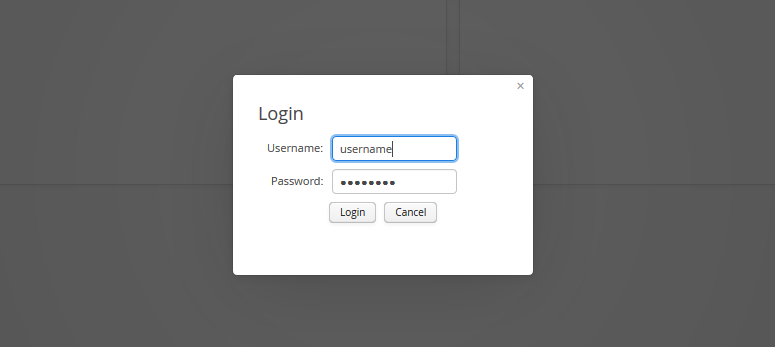
**http://foo:8080/ikasan-dashboard**

This would return the landing page below.

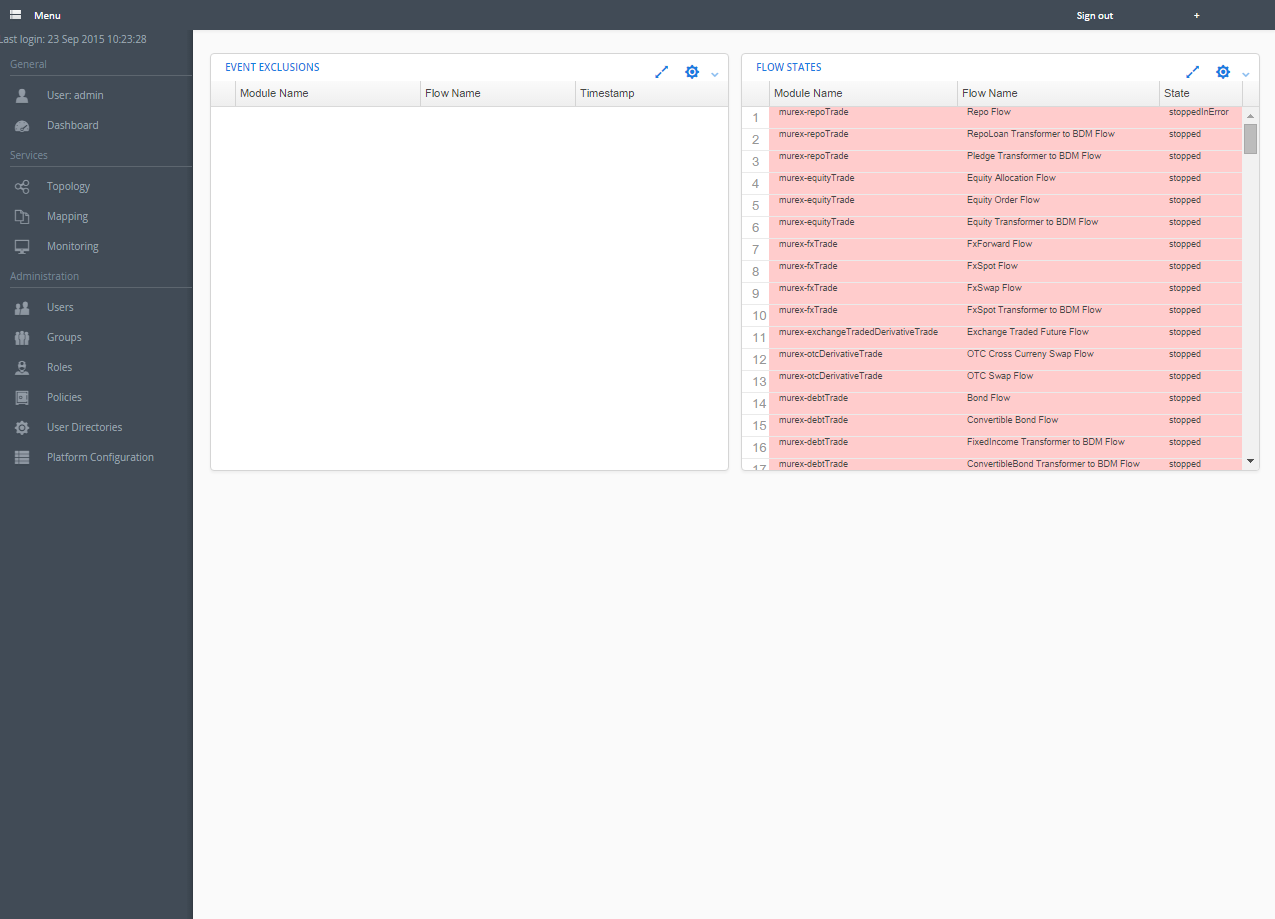


Once you have arrived at the dashboard landing screen you can log in by clicking on the ‘login’ link in the navigation bar at the top of the screen.

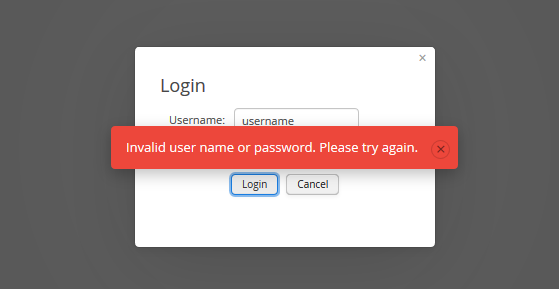
You will then be presented with the login dialog in which the Ikasan dashboard credentials are entered.



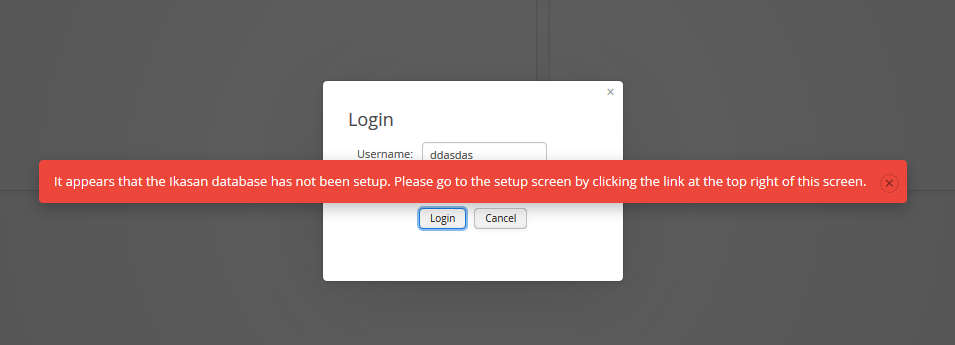
If your credentials are valid you will be presented with a screen similar to below with available functionality appearing in the menu bar on the left of the screen. If you have successfully logged in skip section 3 of this document.



If your credentials are invalid you will be presented with a screen similar to below and you will be required to enter your credentials again.



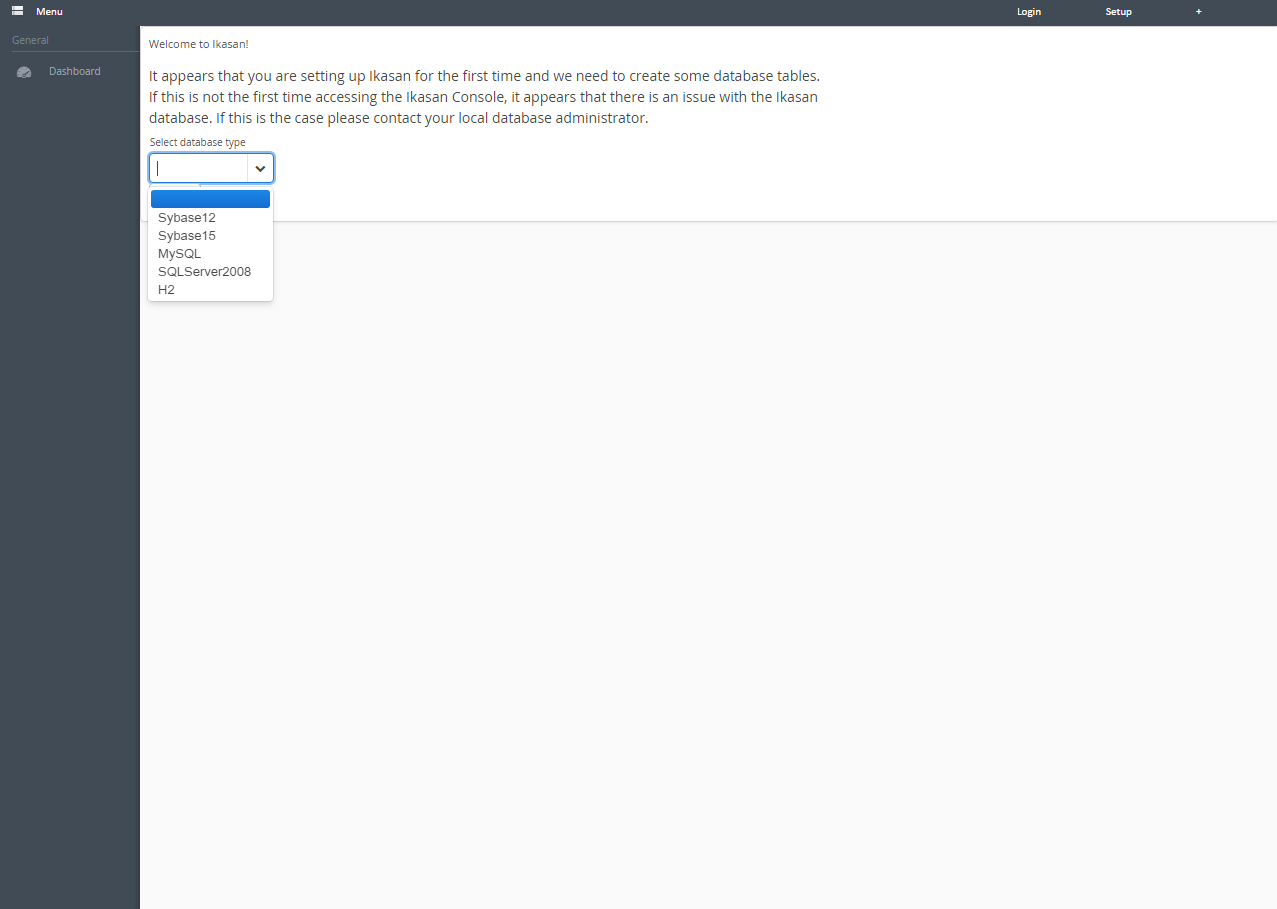
However it is also possible that this is the first time that the dashboard is being accessed in which case you will be required to follow the setup steps outlined in section 3 of this document. If you receive the message below then proceed with the Ikasan setup outlined in section 3.



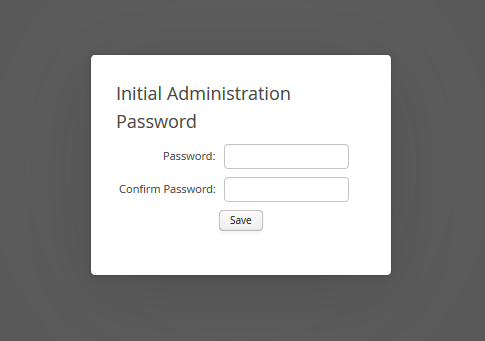
# Platform Setup

If you are setting up Ikasan for the first time, you will be required to go through the platform setup step prior to accessing the dashboard. This screen is accessed by clicking on the setup link on the right hand side of the ‘Navigation Bar’ at the top of the screen.

Upon arriving on the ‘Platform Setup’ screen select the appropriate database that is being used as the Ikasan database and press the create button.



You will then be presented with a dialog in which you must enter the ‘admin’ user password that will be used during the initial setup.



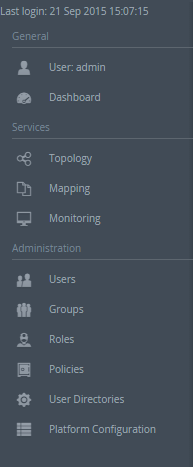
Once you have entered the user password press the save button. This action will then perform the task of creating the required database tables and populating the database with the data required for the dashboard to become operational. Once this is complete you will successfully be able to log into the application as described above. The credentials for the initial login are username ‘admin’ along with the password entered for the administration user in the setup step.

The next logical step in setting up the application in an enterprise context is to setup a user directory (LDAP) and synchronise this application with this directory. The [Managing User Directories](#_Managing_User_Directories) section of this document outlines the steps required to do this in more detail. However the dashboard is now fully operational for the admin user that has been created.

# Menu Navigation

Once logged in a menu will appear on the left hand side of the screen. The menu options that are available to a user are dependent upon the role of the user within the application. The image below shows all menu options that are available within the application. The following sections of this document will describe each functional area in more detail. Each can be accessed by clicking on the appropriate link in the menu below.

The last login time for the currently logged in user is displayed at the top of the menu.

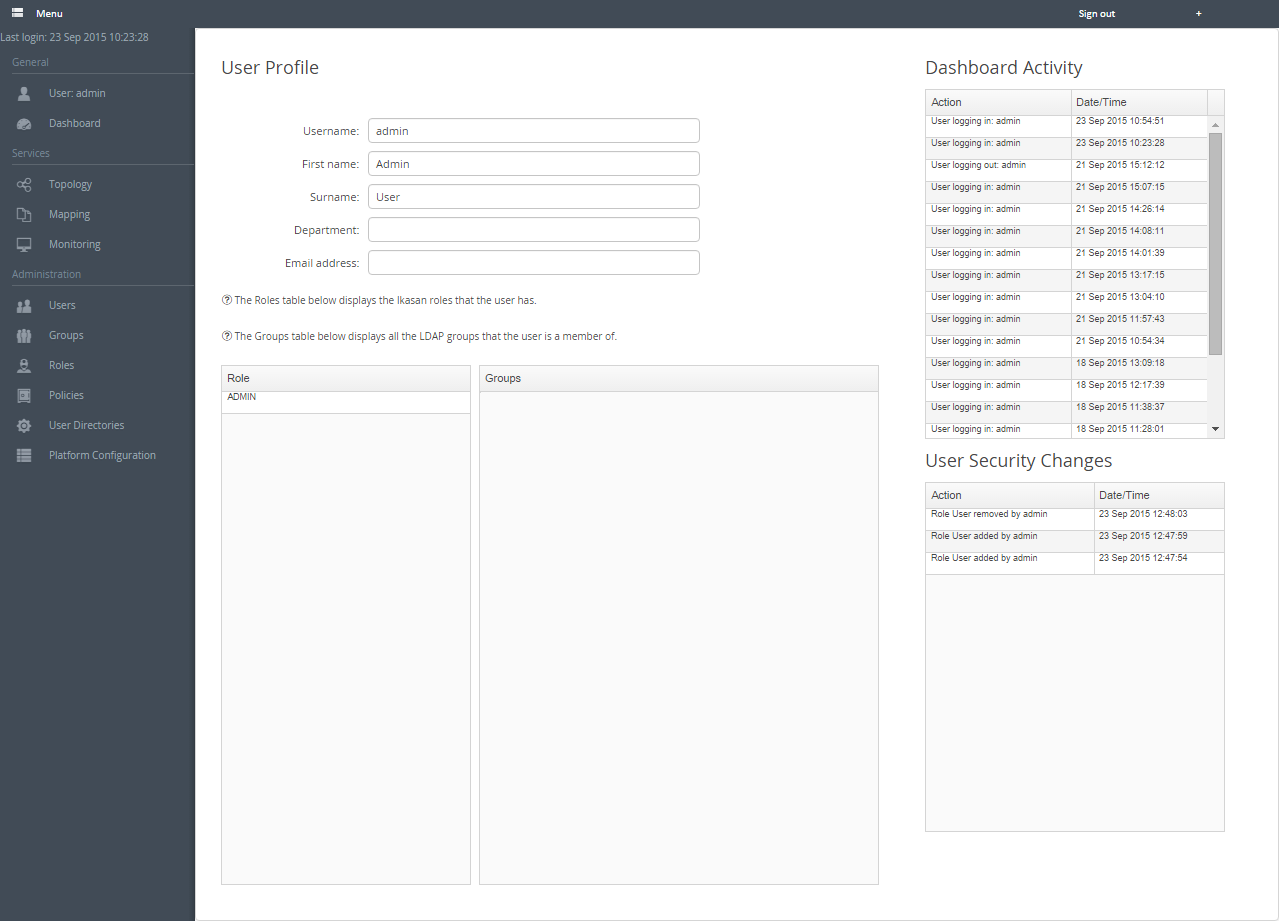


# General

## User Profile

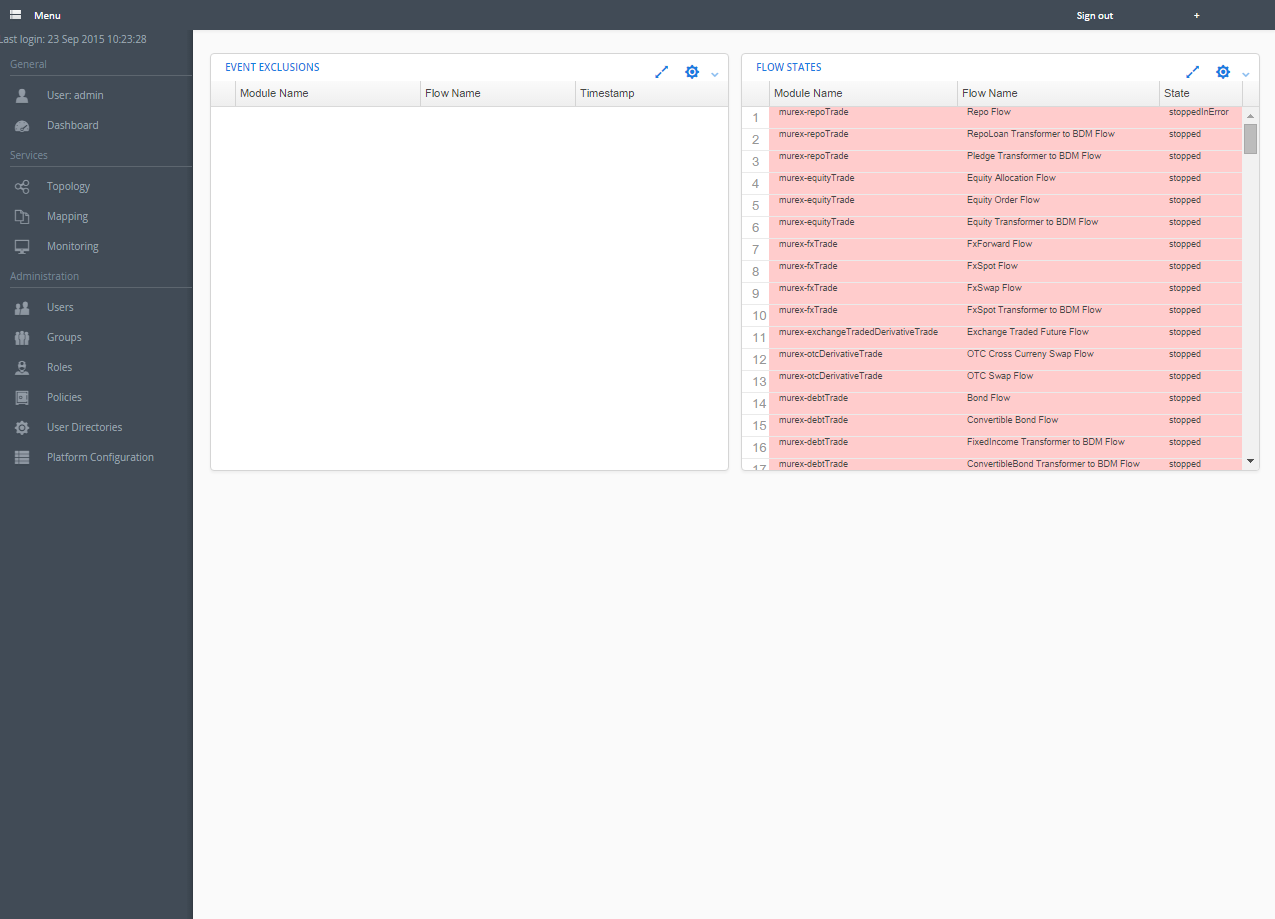
The User Profile screen is purely an informational screen providing details of the currently logged in user. It displays the following information:

* General user information such as name department and email address are shown in the form at the top left of the screen.
* The ‘Role’ table displays all roles that the current user has. The roles are managed within the context iof the Ikasan dashboard.
* The ‘Groups’ table displays all the groups that this user is a member. These groups are sourced from the LDAP repository and are not managed from within the Ikasan dashboard.
* The ‘Dashboard Activity’ table displays information about the current user’s previous access times for the Ikasan dashboard.
* The ‘User Security Changes’ table provides an audit trail of all changes to this current users security permissions.



## Dashboard

The ‘Dashboard’ screen is a work in progress. The intention is that this screen will provide a relevant view based on the users profile.

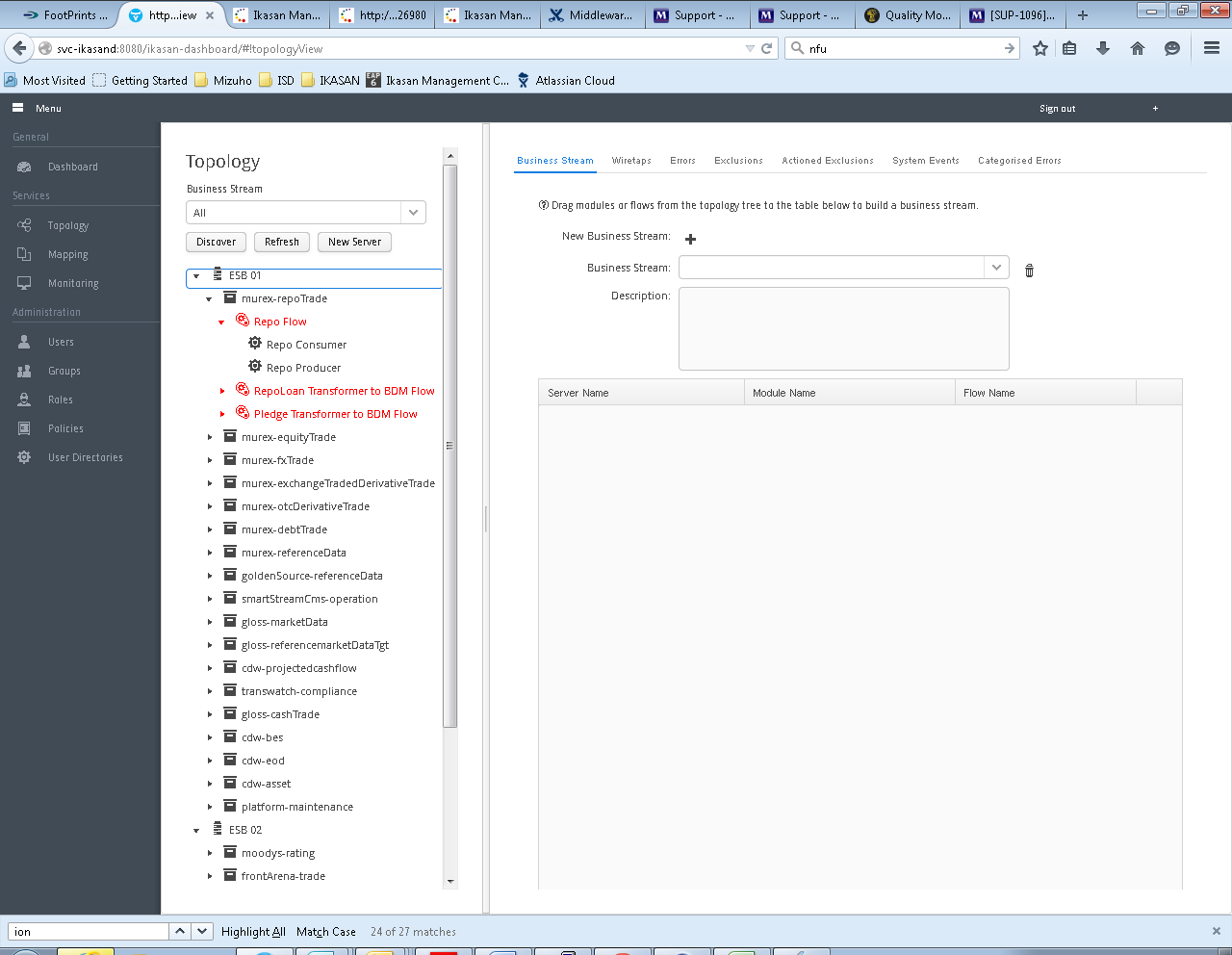


# Services

## Topology View

The ‘Topology View’ exposes a view onto the underlying integration elements that form the Enterprise Service Bus. The users ability to access individual tabs is driven by the roles that the user has. It provides the following functional areas:

* [**Topology Tree**.](#_Topology_Tree) The topology tree provides a hierarchical view on all the servers, integration modules, flows, and component elements and provides functionality to allow for the control and administration of these elements.
* [**Business Stream Tab.**](#_Business_Stream_Tab)The business stream tab allows users to create new business streams and to association relevant flows with a business stream.
* [**Wiretaps Tab.**](#_Wiretap_Tab)The wiretaps tab allows a user to search for data that has flowed through the ESB, which has been wiretapped at relevant points within a flow. The tab provides the ability to filter the results based on criteria defined by the user. Users can drill down on details of wiretapped events.
* [**Errors Tab.**](#_Errors_Tab)The errors tab exposes errors that have occurred within the ESB. These errors can be filtered based on criteria defined by the user. Users can drill down on details of the error.
* [**Exclusions Tab.**](#_Exclusions_Tab)The exclusions tab provides the user with access to events that have been excluded by the ESB. Exclusions typically occur when a business exception has occurred, for example due to missing reference data. Business exceptions are distinctly separate to technical exceptions which are normally transient. For technical exceptions, the ESB will attempt to self-recover and as such these events do not need to be excluded because once the underlying issue is resolved, the data will flow as normal. However, business exceptions are excluded which allows for the root cause to be remediated, for example reference data fixed, and the data subsequently resubmitted into the ESB to flow to its intended destination. This tab allows the user to drill down into details of the excluded data, the error that caused it and provides functionality to allow the user to action the exclusion.
* [**Actioned Exclusions Tab.**](#_Actioned_Exclusions_Tab) The actioned exclusion tab provides an audit view of exclusions that have been actioned.
* [**System Event Tab.**](#_Systems_Events_Tab) The system event tab provides a view on system level events that have occurred within the ESB. These events include flow startup control events, mapping changes, dashboard and module access, and user security permission changes to name a few.
* [**Categorised Errors Tab.**](#_Categorised_Errors_Tab)The categorised errors tab provides a view on all errors that have been deemed categorised errors. The user can drill down on details and action if applicable.



### Topology Tree

The topology tree provides a hierarchical view on all the servers, integration modules, flows, and component elements and provides functionality to allow for the control and administration of these elements. The following functionality is available within the tree (dependant on the permissions you have within the application):

ESB wide functionality:

* [Add a new server.](#_Adding_a_New)
* Discover new modules.

At the server level:

* The ability to view details of the server. (Not currently implemented)
* [The ability to categorise an error.](#_Categorised_Errors)
* Edit server details.

At the module level:

* The ability to view details of the module. (Not currently implemented)
* The ability to view the module diagram. (Not currently implemented)
* [The ability to categorise an error.](#_Categorised_Errors)

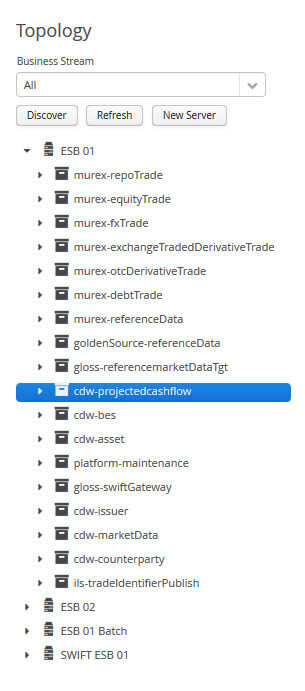
At the flow level:

* The ability to control the flow (Start, Stop, Pause, Resume).
* [The ability to configure the start-up type.](#_Start_up_Control)
* [The ability to categorise an error.](#_Categorised_Errors)

At the component level:

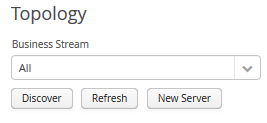
* [The ability to configure a component.](#_Component_Configuration)
* [The ability to add a wiretap to a component](#_Wiretap_Configuration).
* [The ability to categorise an error.](#_Categorised_Errors)

There are a number of icons in the topology tree which represent the following:

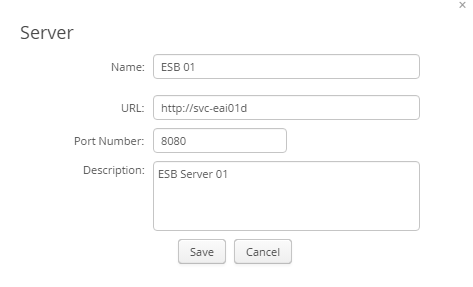
* Represents a server to which Ikasan modules are deployed.
*  Represents an Ikasan module.
*  Represents a flow within an Ikasan module.
*  Represents a component within an Ikasan flow which can be configured.
*  Represents a component within an Ikasan flow which cannot be configured.

#### Populating the Topology Tree

When an Ikasan module is deployed to a server, the module updates the underlying data store with details about itself. The topology tree is then populated when the ‘Discover’ button is pressed, by inspecting the servers that have been configured within the Dashboard, and attempting to discover modules that have registered themselves with the underlying data store. Once a module is discovered, it describes itself, and all of its details are inserted into the underlying data store. The ‘Topology Tree’ is then refreshed by pressing the ‘Refresh’ button in order to display the newly discovered modules. The image below shows the buttons that perform these functions which can be found at the top of the topology tree.



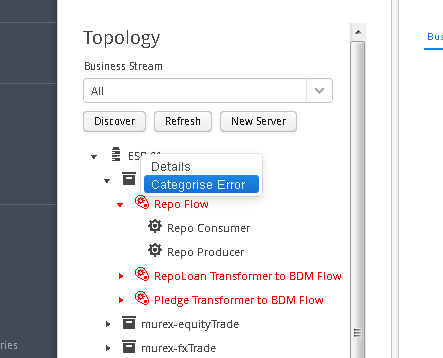
Prior to initiating the ‘Discovery’ process a server or servers must be added to the Ikasan Dashboard. This is done by pressing the ‘New Server’ button. The ‘Server’ window will appear and the user is required to fill all fields for the server prior to pressing the ‘Save’ button.



#### Categorised Errors

A categorised error is a decoration of an underlying ESB error occurrence. It allows the user to provide a criticality as well as a user friendly error message for errors of a certain type, which caused a certain action at a certain location within the ESB. These errors will then be displayed on the ‘[Categorised Errors Tab’](#_Categorised_Errors_Tab) where they can be filtered based on relevant criteria.

In order to create a categorised error, select the location within the ESB where you want to categorise an error by expanding the dashboard topology tree. Right click on the node in the tree and select ‘Categorised Error’ from the menu that appears as seen in the image below.



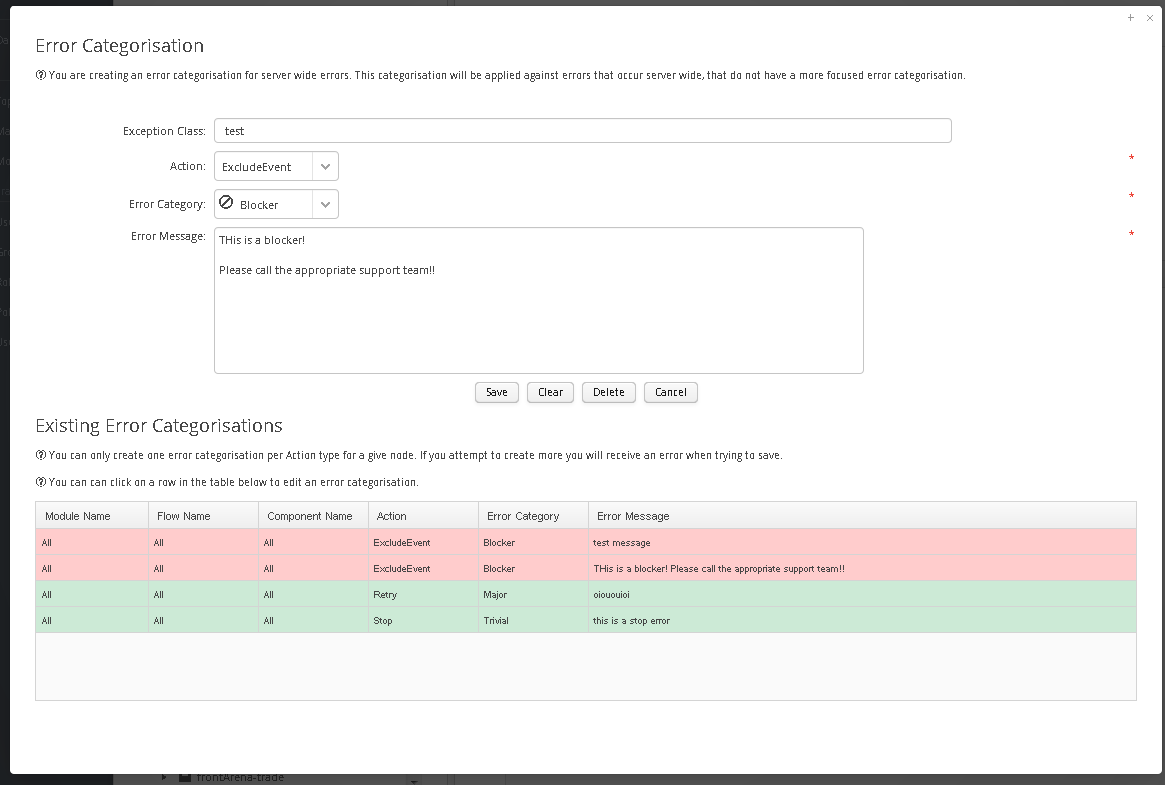
Once you have selected the ‘Categorise Error’ menu item, a window will appear. This window allows you to create, modify or delete categorised errors associated with the node. There are some basic rules for the creation of a ‘Categorised Error’.

1. For a given node (ESB Element) within the topology tree, the ‘Action’ / ‘Exception Class’ combination must be unique.
2. ‘Categorised Errors’ are applied to the underlying error occurrence in a cascading manner from the most relevant to the least relevant configured ‘Categorised Error’. For example, if you have a ‘Categorised Error’ configured at the component level that is a direct match for the ‘Exception Class / Action’ combination, then this is the ‘Categorised Error’ that will be displayed in the ‘Categorised Error’ tab when searched. However, if there is no direct match at the component level, but there are ‘Categorised Errors’ configured at the flow, module or server level that match on the ‘Exception Class / Action’ combination, then this will cause this error to be included in the results on the Categorised Error tab.

The use of the ‘Error Categorisation’ window is straight forward. The form at the top allows the following to be added:

* **Exception Class** – free form text containing the fully qualified java exception class to match on.
* **Action** – the action taken by Ikasan when the exception occurs.
* **Error Category** – how this particular error should be categorised.
* **Error message** - the user friendly error message that will be displayed with the categorised error.

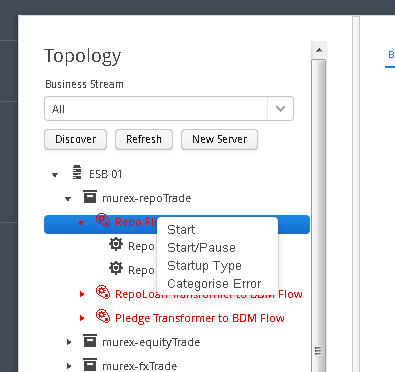
Once the ‘Categorised Error’ is saved it will appear in the table of ‘Categorised Errors’ at the bottom of the window. Any of these can be selected and modified or deleted from this table.



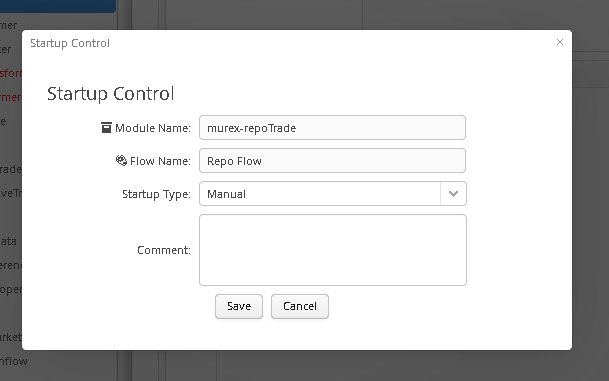
#### Start up Control

The start-up control functionality is available at the flow level within the ‘Topology Tree’. It can be accessed by right clicking on the relevant flow node in the topology tree that you want to set the start-up control for. There are 3 settings for start-up control:

1. Manual – when a server restarts, it relies on the user to manually restart the flow.
2. Automatic – when a server restarts, the flow will be automatically restarted.
3. Disabled – the flow is disabled and cannot be restarted.



Once selected the ‘Startup Control’ window will be displayed. The user can then select the appropriate startup type. A comment is mandatory when setting the startup type to ‘Disabled’.

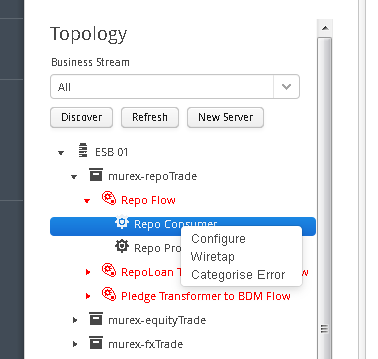


#### Component Configuration

The component configuration functionality is available at the component level within the ‘Topology Tree’. It can be accessed by right clicking on the relevant component node in the topology tree that you want to create or modify the configuration for.

Components in the topology tree are denoted by 2 different cog icons

*  Denotes components that are configurable, hence will have the ‘Configure’ menu option available when right mouse clicked on.
*  Denotes components that are not configurable and do not have the ‘Configure’ menu item available.



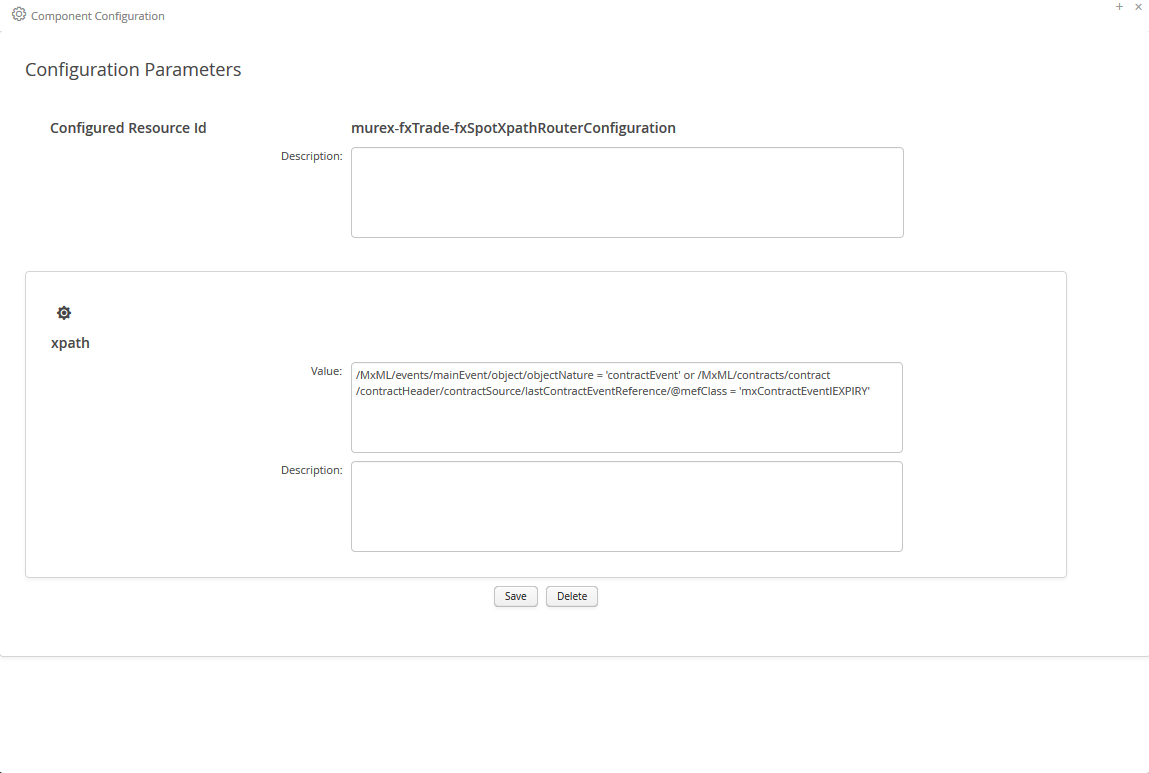
Once the ‘Configure’ menu item in the topology tree has been selected, the ’Component Configuration’ window will appear. The action of opening the window is twofold.

1. If the ‘Configuration’ does not yet exist, a new one will be created. Any default field vales that have been defined in the containing module will be set on the appropriate configuration parameter when the ‘Component Configuration’ window opens.
2. If the ‘Configuration’ already exists then all values currently set on the configuration in the database will be used to populate the ‘Component Configuration’ window.

Ikasan provides support for a number of different configuration parameter data types:

* Integer
* Boolean
* List of Strings
* Map of String key and value
* String
* Long
* Masked String

These datatypes are defined on the underlying configuration object that has been created within the associated Ikasan module. All form validation that occurs when a ‘Configuration’ is saved will validate against the relevant datatypes and the user will be presented with an appropriate error message if the configuration parameter cannot be converted to the expected datatype.

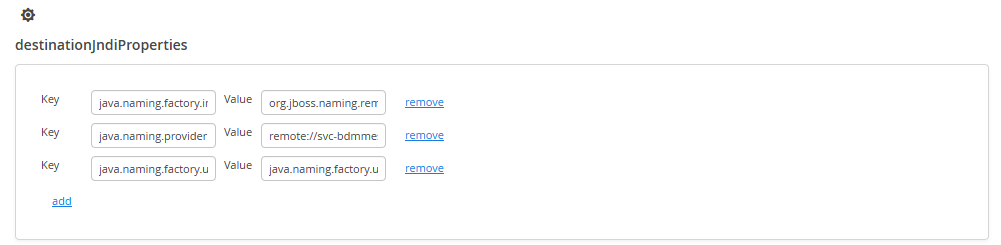
****

There are 2 actions that can be performed on the ‘Component Configuration’ window:

1. **Save**. Will validate all datatypes that have been entered into the form with the expected underlying datatypes. Validation errors will be raised if any validation rules are violated, otherwise the configuration will be saved. **NB It is important to note, newly configured configuration values will not take effect until the associated flow is restarted.**
2. **Delete.** This will delete the configuration from the underlying data store.

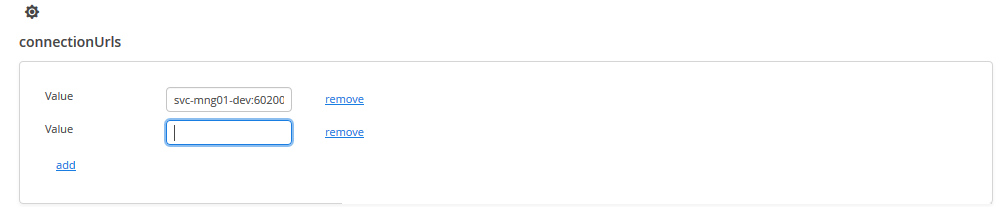
##### Managing Map Configuration Parameters

Map datatypes can be used as configuration parameters. These are stored as a Map with keys which are String datatypes and values which are also String datatypes. The image below shows a Map being managed from the ‘Component Configuration’ screen. To add a new key-value pair, click the ‘add’ link at the bottom of the existing key-value pairs. To remove an existing key-value pair, click on the ‘remove’ link to the right of the key-value pair you wish to remove. The changes to the Map will not be applied to the underlying data store until the ‘save’ button of the ‘Component Configuration’ window is pressed.



##### Managing List Configuration Parameters

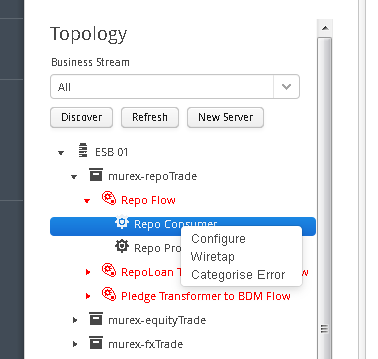
List datatypes can be used as configuration parameters. These are stored as a List of String elements. The image below shows a List being managed from the ‘Component Configuration’ screen. To add an element to the list, click the ‘add’ link at the bottom of the existing elements. To remove an existing element click on the ‘remove’ link to the right of the element you wish to remove. The changes to the List will not be applied to the underlying data store until the ‘save’ button of the ‘Component Configuration’ window is pressed.



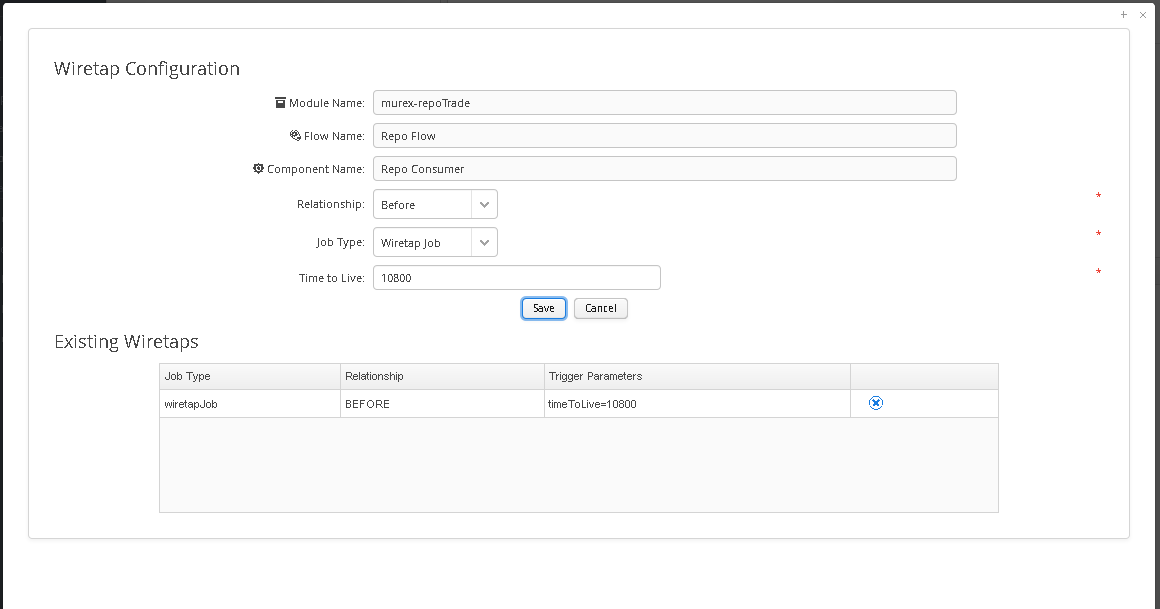
#### Wiretap Configuration

The Ikasan framework supports the common EAI concept of wiretaps. A wiretap is a trigger that allows for data events that occur with the ESB to be captured in order to be search and viewed. They create a temporal trail of events flowing through the ESB and are an invaluable tool when supporting the ESB. This section explains the creation of wiretaps, while the [Wiretap Tab](#_Wiretap_Tab) section has more details about searching for wiretapped events.

Wiretaps are configured at the component level. This is done by right clicking on the component you wish to wiretap and selecting the ‘Wiretap’ menu item.



Once the ‘Wiretap’ menu item has been selected, the ‘Wiretap Configuration’ will appear as seen below.



The ‘Configuration Window’ will have the Module Name, Flow Name, and Component Name fields pre-populated based on the node in the topology tree from which with ‘Wiretap’ menu item was selected. In order to create a ‘Wiretap’ the user will be required to provide values for the following fields:

* **Relationship**. The relationship determines when the trigger is fired. There are 2 choices. ‘Before’ which causes the wiretap trigger to be fired prior to the event being processed by the component being wiretapped. And ‘After’ which causes the wiretap trigger to be fired after the component has processed the event.
* **Job Type**. There are 2 job types. ‘Logging Job’ which when fired, writes details of the event to the log file that is associated with the Ikasan module that the component belongs to. ‘Wiretap Job’ which when fired, writes details of the event to the underlying data store which has been associated with Ikasan. Wiretaps can then be searched and viewed via the Ikasan dashboard as defined in section [Wiretap Tab](#_Wiretap_Tab).
* **Time to Live**. The ‘Time to Live’ field is only relevant for ‘Wiretap Jobs’. It is configured with the ‘Wiretap Job’ to inform scheduled housekeeping processes to remove the wiretap after the ‘Time to Live’ has expired. This field is expecting a numerical value and is expressed as a time to live in seconds.

**Once the wiretap form has been completed, the save button will store the wiretap which will take immediate effect for all subsequent events received by the associated component**.

Once a ‘Wiretap’ has been saved it will appear in the ‘Existing Wiretaps’ table at the bottom of the wiretaps window. This table provides a view on all the wiretaps that have been created for a component. Wiretaps can be deleted by clicking on the delete icon in the right hand column of this table.

### Business Stream Tab

A ‘Business Stream’ is a logical grouping of Ikasan flows that together form a view on the ESB that constitutes a view that is of interest to a set of users, whether business users or technical users, who are required to have a more narrow or focused view of events, errors or exclusions that occur across the flows that the Business Stream encapsulates.

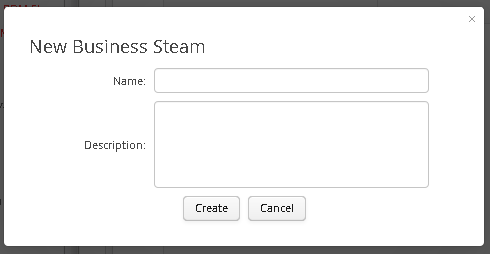
From the ‘Business Stream’ tab a user can create a new ‘Business Stream’, modify an existing ‘Business Stream’ or delete a ‘Business Stream’

#### Creating a New Business Stream

In order to create a new ‘Business Stream’ click on the  icon denoted by the New Business Stream label on the ‘Business Stream Tab’.



This will cause the ‘New Business Stream’ window to appear.

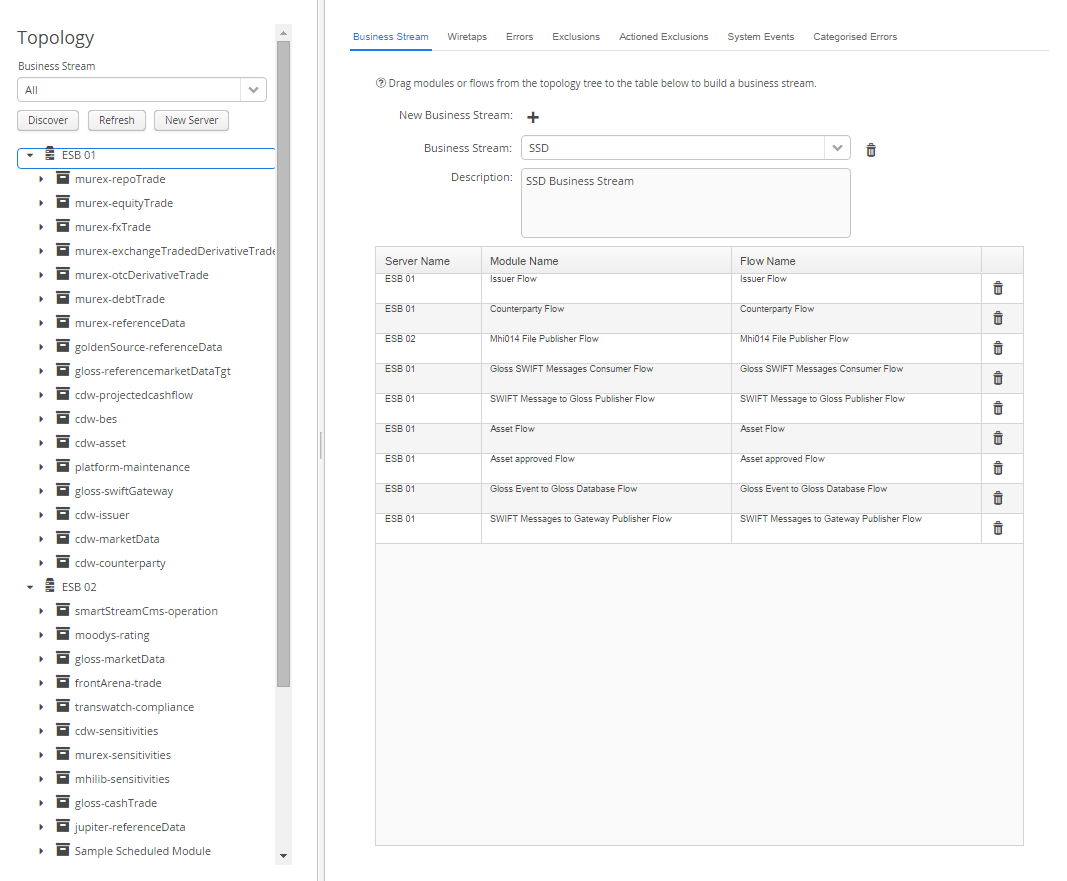


In order to create a new ‘Business Stream’ a user must supply a Name and Description and press the ‘Save’ button.

The new Business Stream will then appear in the Business Stream Combo as seen below:



Once a ‘Business Stream’ has been created, users are able to start adding flows to it by dragging with Modules  or Flows  out of the ‘Topology Tree’ and dropping them into the table on the ‘Business Stream Tab’. This will automatically persist the addition of these flows to the ‘Business Stream’ into the underlying data store.



Individual flows can be removed form a ‘Business Stream’ by clicking on the trash can in the table row for the flow you wish to delete. Once again this action will automatically cause the flow to be removed from the ‘Business Stream’ in the underlying data store.



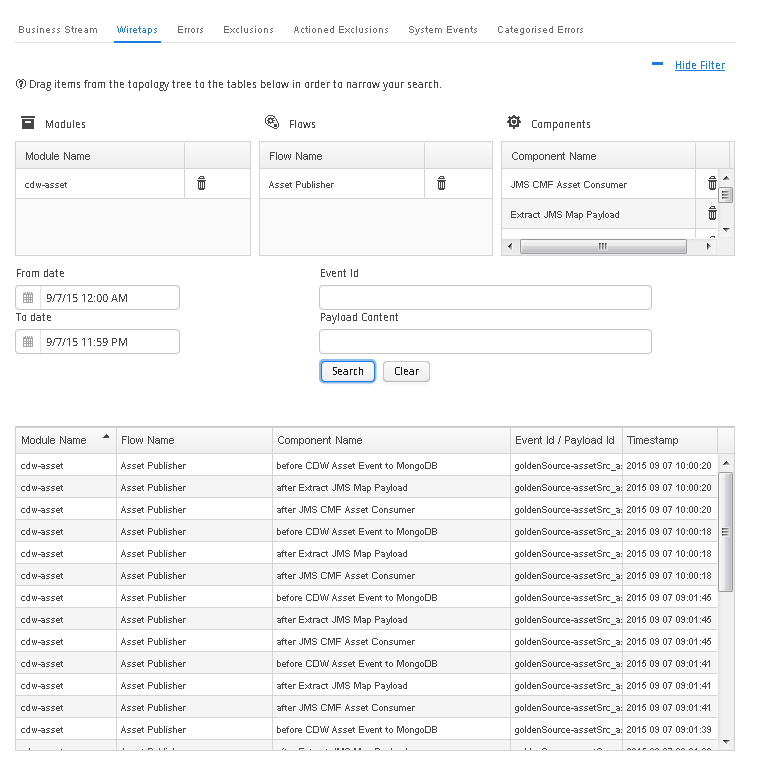
Lastly a ‘Business Stream’ can be deleted completely by selecting the ‘Business Stream’ in the combo box and clicking the trash can to its right.



### Wiretap Tab

The ‘Wiretaps Tab’ allows the user to search for and view all wiretaps that have been [configured](#_Wiretap_Configuration_1) within the topology tree.

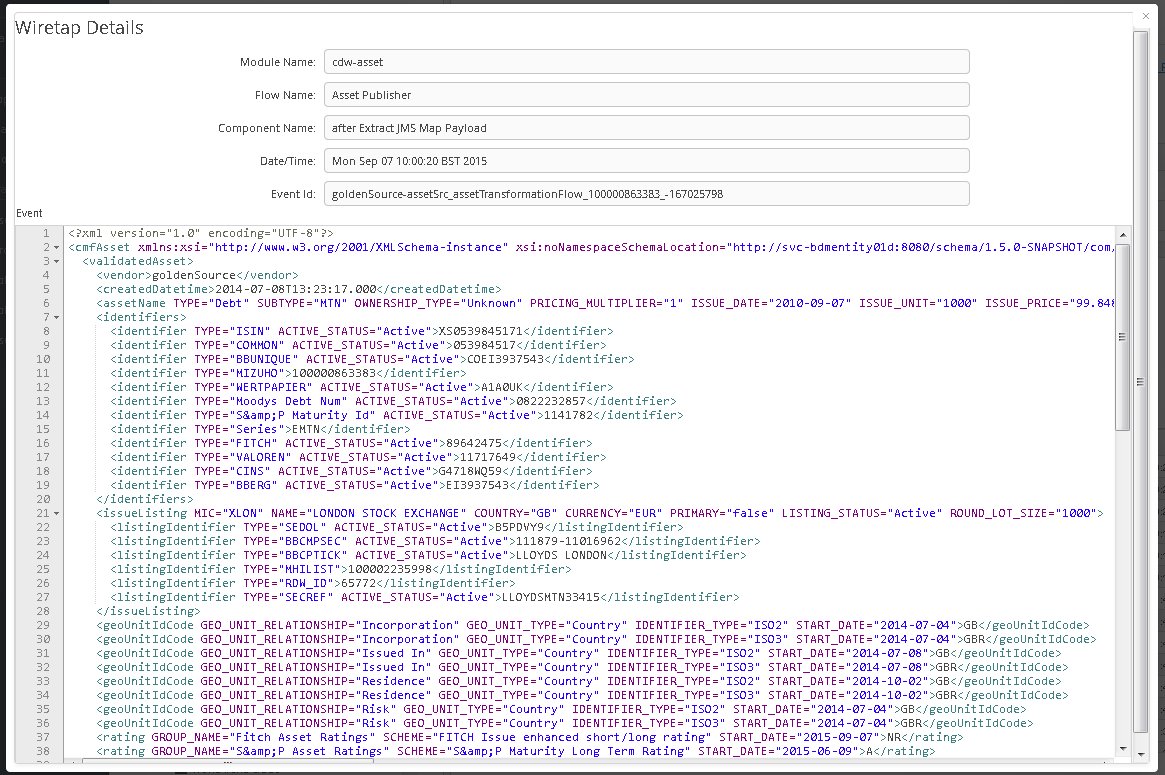
The ‘Wiretap Tab’ consists of 2 main sections. The top section allows the user to provide filtering on the ‘Wiretap’ search. In order to filter the search ‘Modules’, ‘Flows’ and ‘Components’ can be dragged from the topology tree and dropped into their respective tables. It is also possible to narrow by the date and time that the ‘Wiretap Event’ occurred and also search using the ‘Event id’ or ‘Payload Content’. Once the search filter has been defined the user can click the ‘Search’ button to perform the search and results will appear in the table below. The search filter can be cleared using the ‘Clear’ button.



The filter can be expanded and collapsed using the link at the top right of the screen.

#### Wiretap Details Window

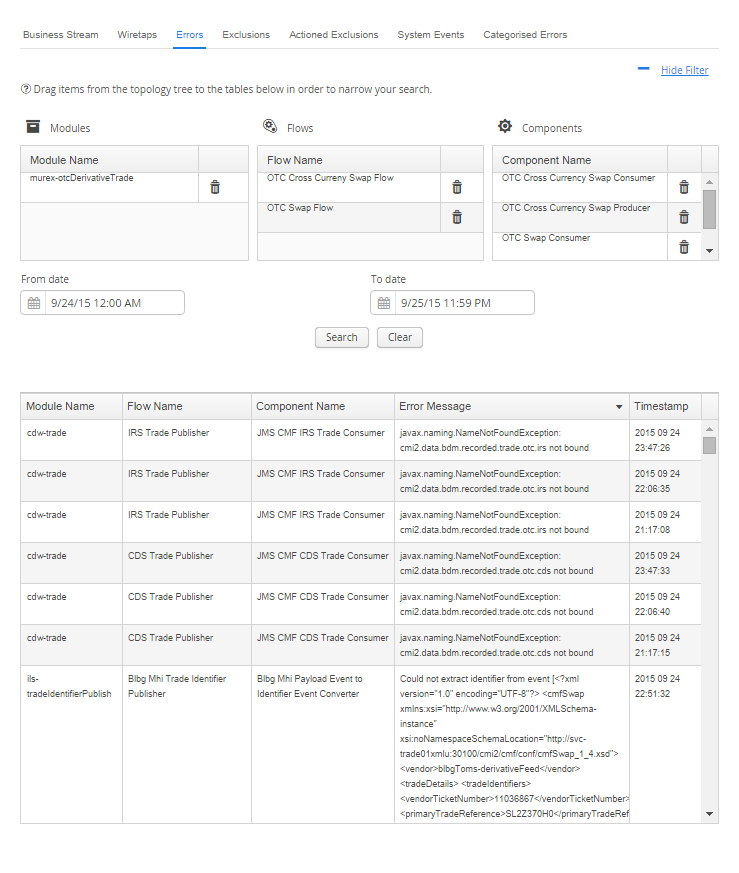
Once a ‘Wiretap’ search has been performed, it is possible to click on a row within the search results table. This will cause the ‘Wiretap Details’ window to open. The ‘Wiretap Details’ window provides further information regarding the ‘Wiretapped Event’ including the payload of the event.



### Errors Tab

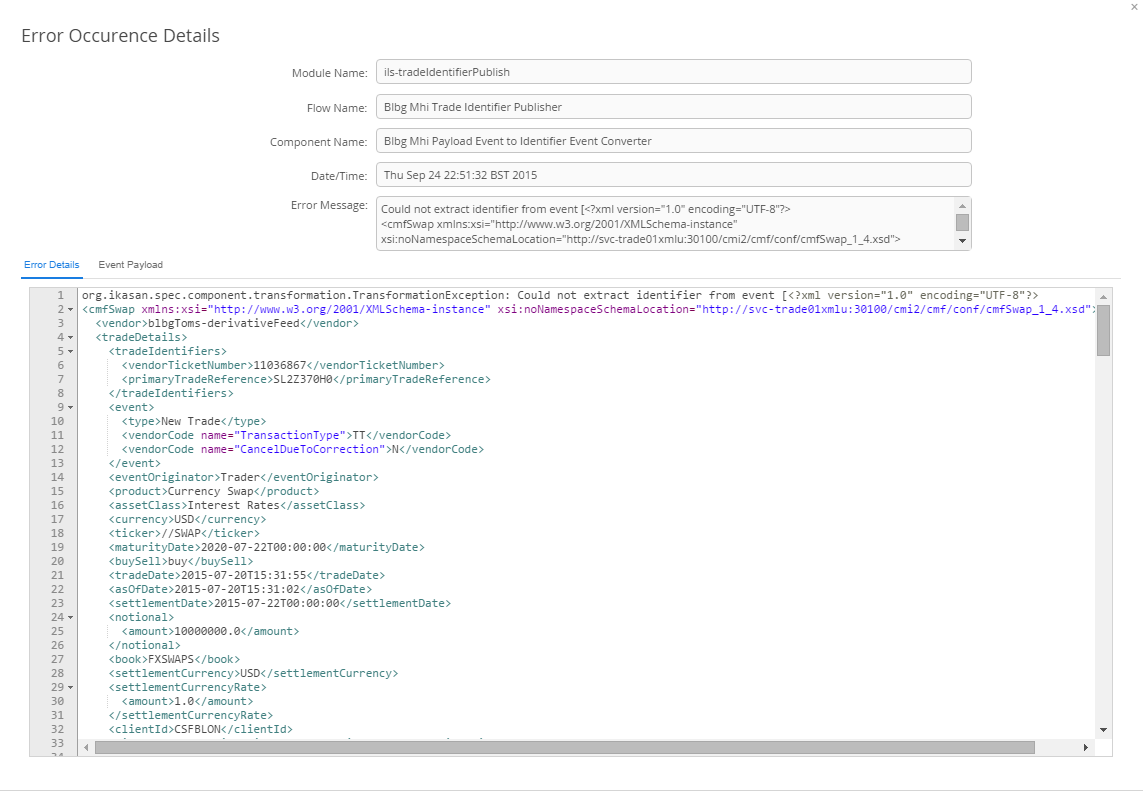
The ‘Errors Tab’ allows the user to search for and view all errors that have occurred within the ESB.

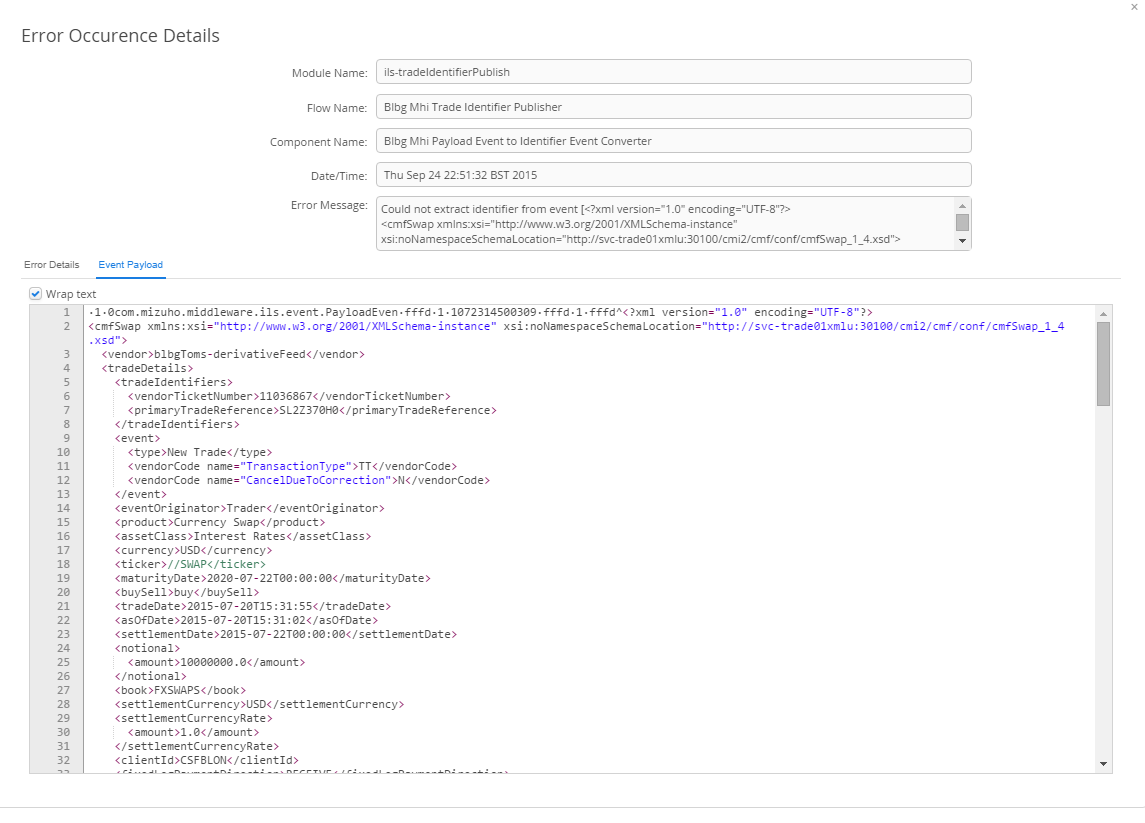
The ‘Errors Tab’ consists of 2 main sections. The top section allows the user to provide filtering on the ‘Errors’ search. In order to filter the search ‘Modules’, ‘Flows’ and ‘Components’ can be dragged from the topology tree and dropped into their respective tables. It is also possible to narrow by the date and time that the error occurred. Once the search filter has been defined the user can click the ‘Search’ button to perform the search and results will appear in the table below. The search filter can be cleared using the ‘Clear’ button.



#### Error Occurrence Details Window

Once an ‘Error’ search has been performed, it is possible to click on a row within the search results table. This will cause the ‘Error Occurrence Details’ window to open. The ‘Error Occurrence Details’ window provides further information regarding the ‘Error’ including the payload of the error event along with the error message and a full java stack trace.

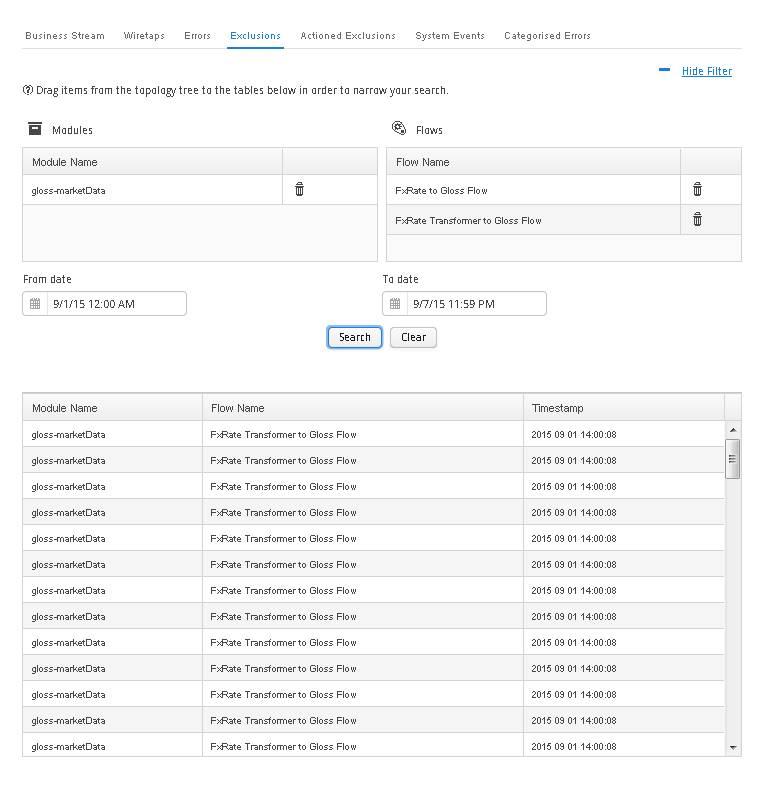




### Exclusions Tab

The ‘Exclusions Tab’ allows the user to search for and view all exclusions that have occurred within the ESB.

The ‘Exclusions Tab’ consists of 2 main sections. The top section allows the user to provide filtering on the ‘Exclusions’ search. In order to filter the search ‘Modules’ and ‘Flows’ can be dragged from the topology tree and dropped into their respective tables. It is also possible to narrow by the date and time that the exclusion occurred. Once the search filter has been defined the user can click the ‘Search’ button to perform the search and results will appear in the table below. The search filter can be cleared using the ‘Clear’ button.

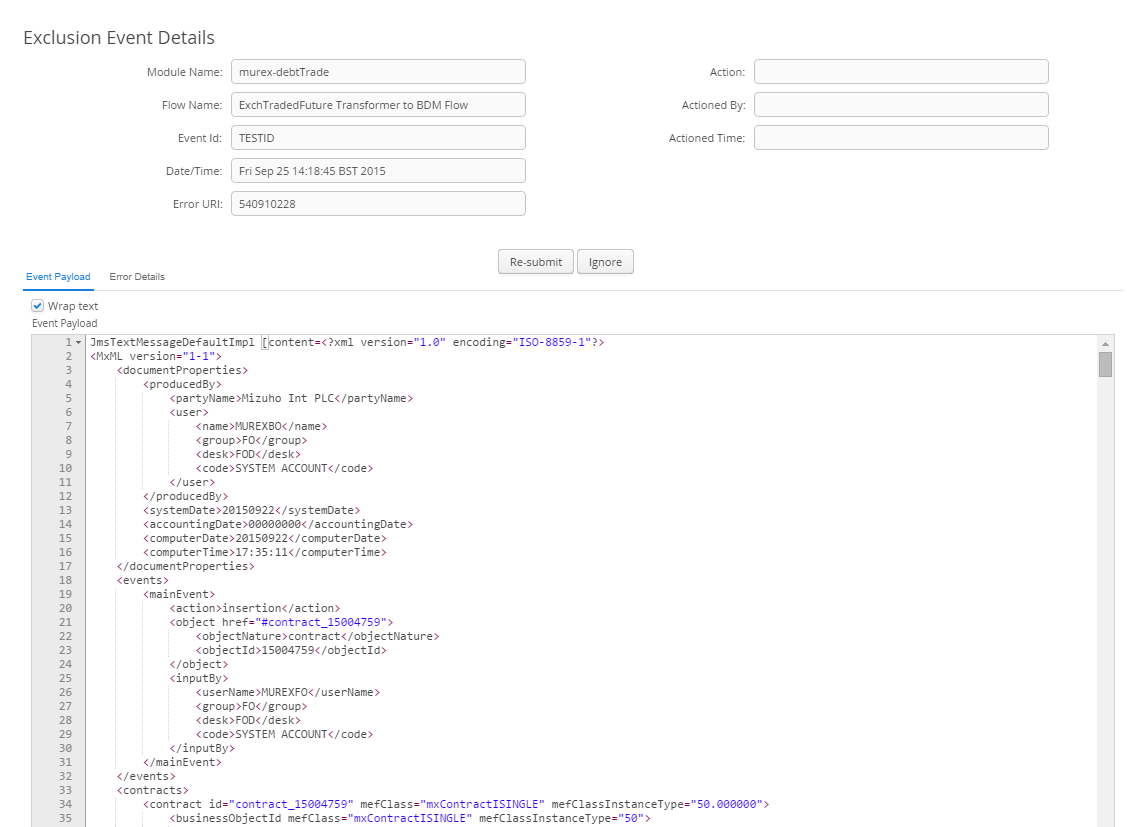


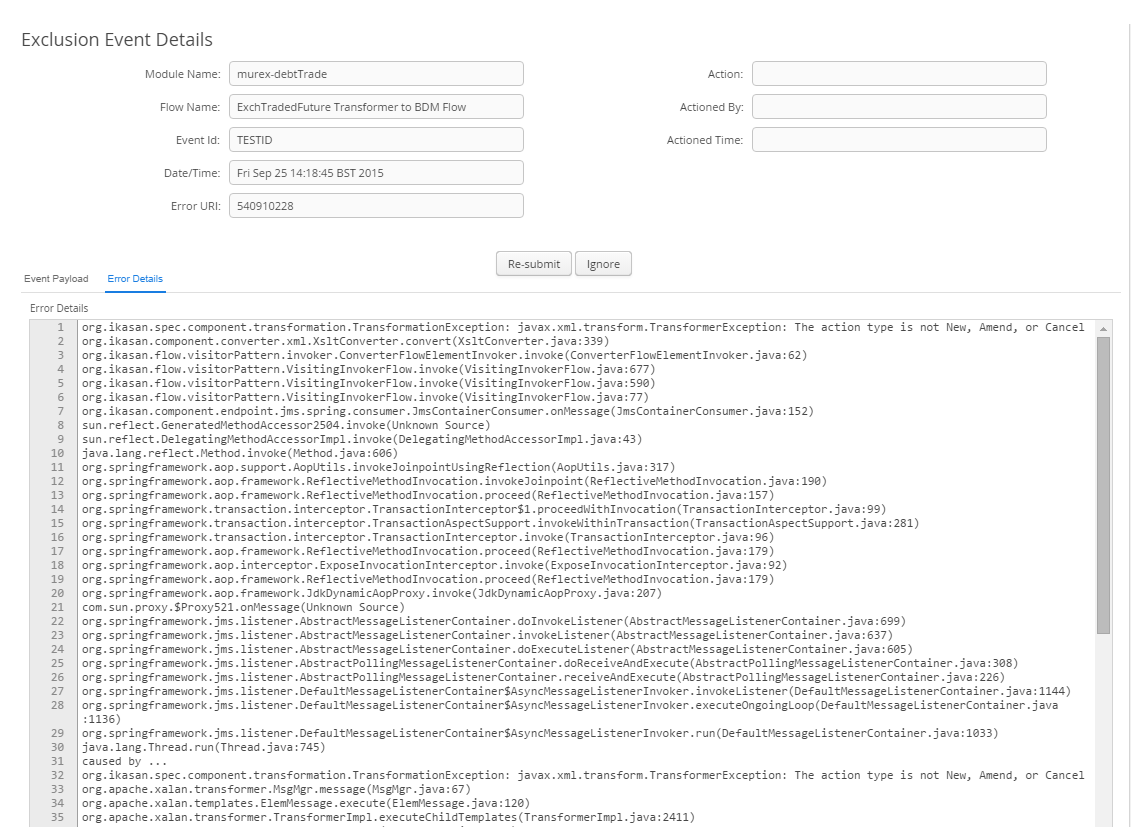
#### Exclusion Event Details Window

Once an ‘Exclusions’ search has been performed, it is possible to click on a row within the search results table. This will cause the ‘Exclusion Event Details’ window to open. The ‘Exclusion Event Details’ window provides further information regarding the ‘Exclusion’ including the payload of the event along with the error details.

There are 2 actions that can be performed on an ‘Excluded Event’:

1. The user can choose to resubmit the ‘Excluded Event’. This will cause the event, in its original state, to be submitted back into the flow from which it was excluded. The action taken, who it was taken by, and at what time will be recorded against the ‘Excluded Event’. The ‘Excluded Event’ will no longer appear in future ‘Excluded Event’ searches and will become an ‘Actioned Excluded Event’.
2. The user can choose to ignore the ‘Excluded Event’. The action taken, who it was taken by, and at what time will be recorded against the ‘Excluded Event’. The ‘Excluded Event’ will no longer appear in future ‘Excluded Event’ searches and will become an ‘Actioned Excluded Event’ who has been ignored.

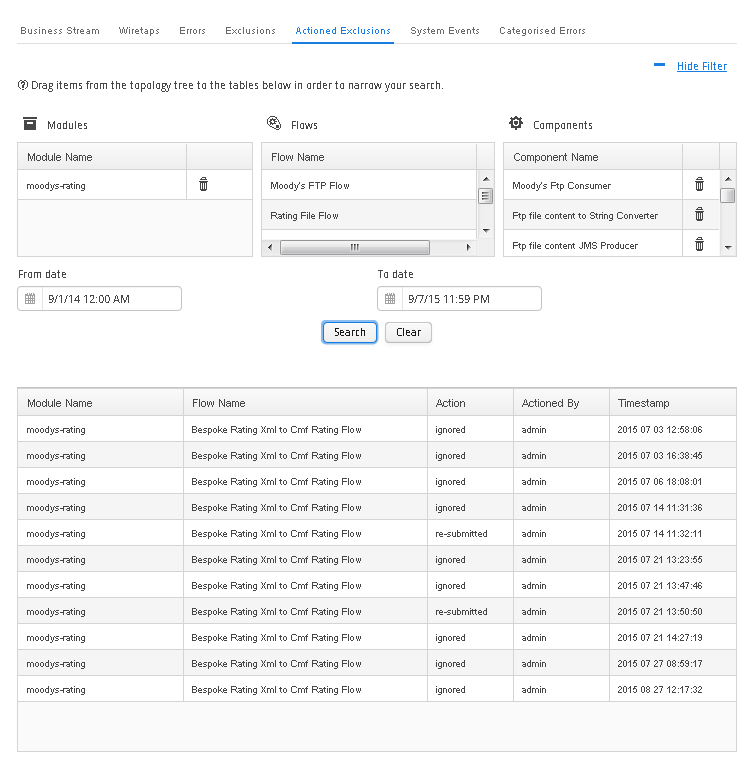




### Actioned Exclusions Tab

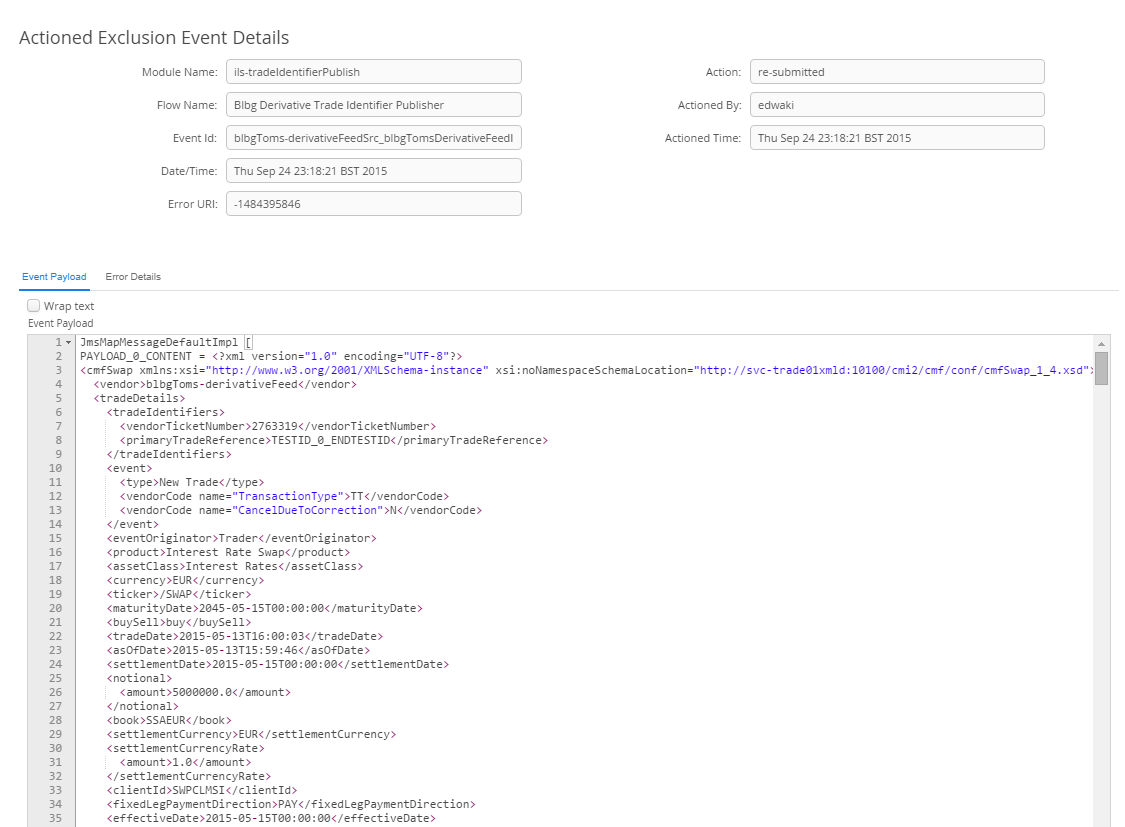
The ‘Actioned Exclusions Tab’ allows the user to search for and view all actioned exclusions that have occurred within the ESB. An ‘Actioned Exclusion’ is an ‘Excluded Event’ that has either been resubmitted or ignored by a user.

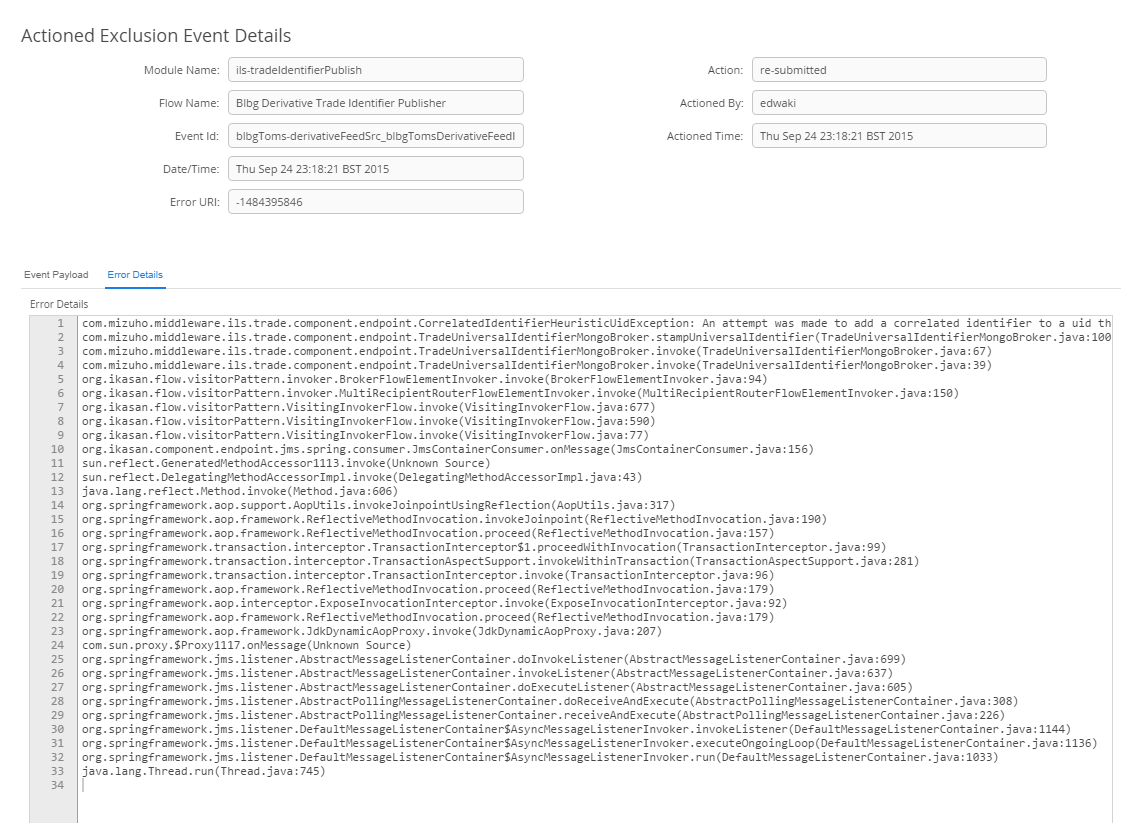
The ‘Actioned Exclusions Tab’ consists of 2 main sections. The top section allows the user to provide filtering on the ‘Exclusions’ search. In order to filter the search ‘Modules’ and ‘Flows’ can be dragged from the topology tree and dropped into their respective tables. It is also possible to narrow by the date and time that the exclusion occurred. Once the search filter has been defined the user can click the ‘Search’ button to perform the search and results will appear in the table below. The search filter can be cleared using the ‘Clear’ button.



#### Actioned Exclusion Event Details Window

Once an ‘Actioned Exclusions’ search has been performed, it is possible to click on a row within the search results table. This will cause the ‘Actioned Exclusion Event Details’ window to open. The ‘Actioned Exclusion Event Details’ window provides further information regarding the ‘Actioned Exclusion’ including the payload of the event along with the error details.

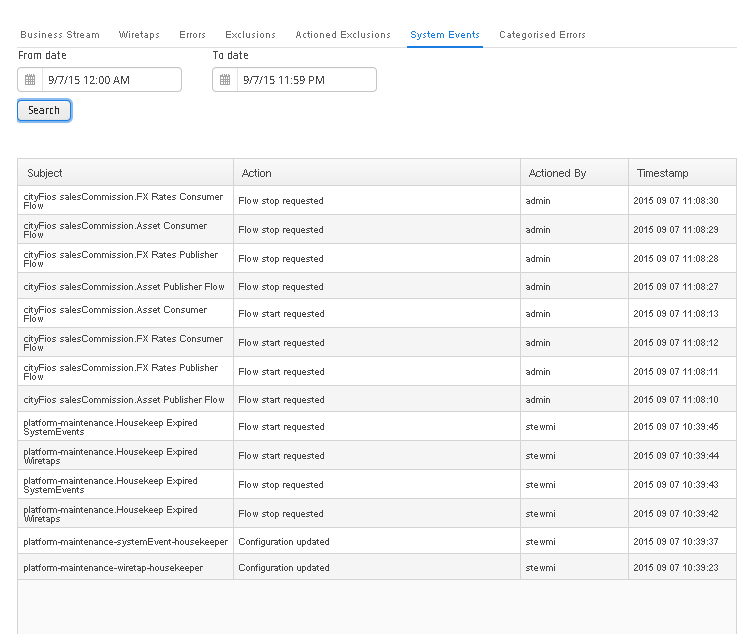




### Systems Events Tab

The ‘System Events Tab’ allows the user to search for and view all system events that have occurred within the ESB. The following system events can occur:

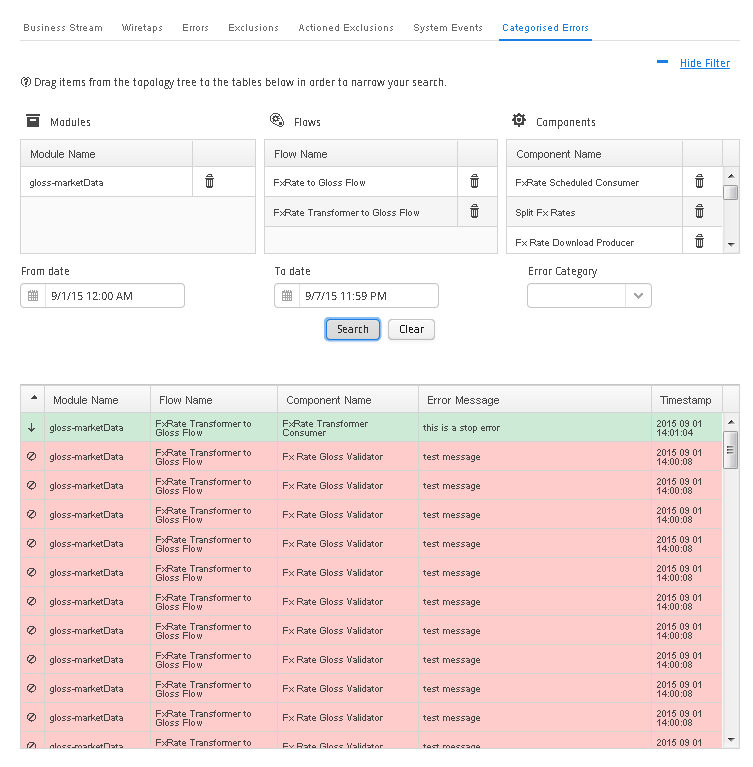
* A user starting or stopping a module.
* A user creating, updating or deleing a configuration.
* When a user’s security role changes.
* When a user logs in or out of the dashboard,
* When a user creates, modifies or deletes a ‘[Mapping Configuration’](#_Managing_a_Mapping).



### Categorised Errors Tab

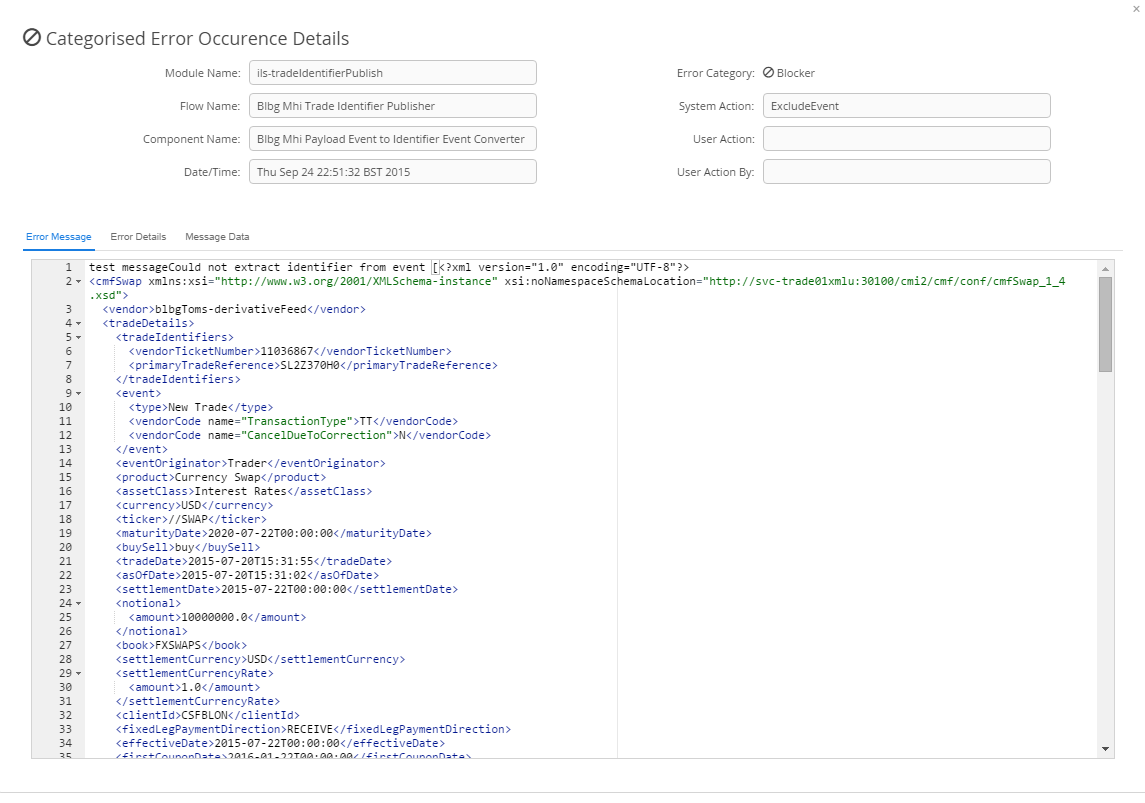
The ‘Categorised Errors Tab’ allows the user to search for and view all errors that have occurred within the ESB and that have a ‘[Categorised Error’](#_Business_Stream_Tab) associated with them.

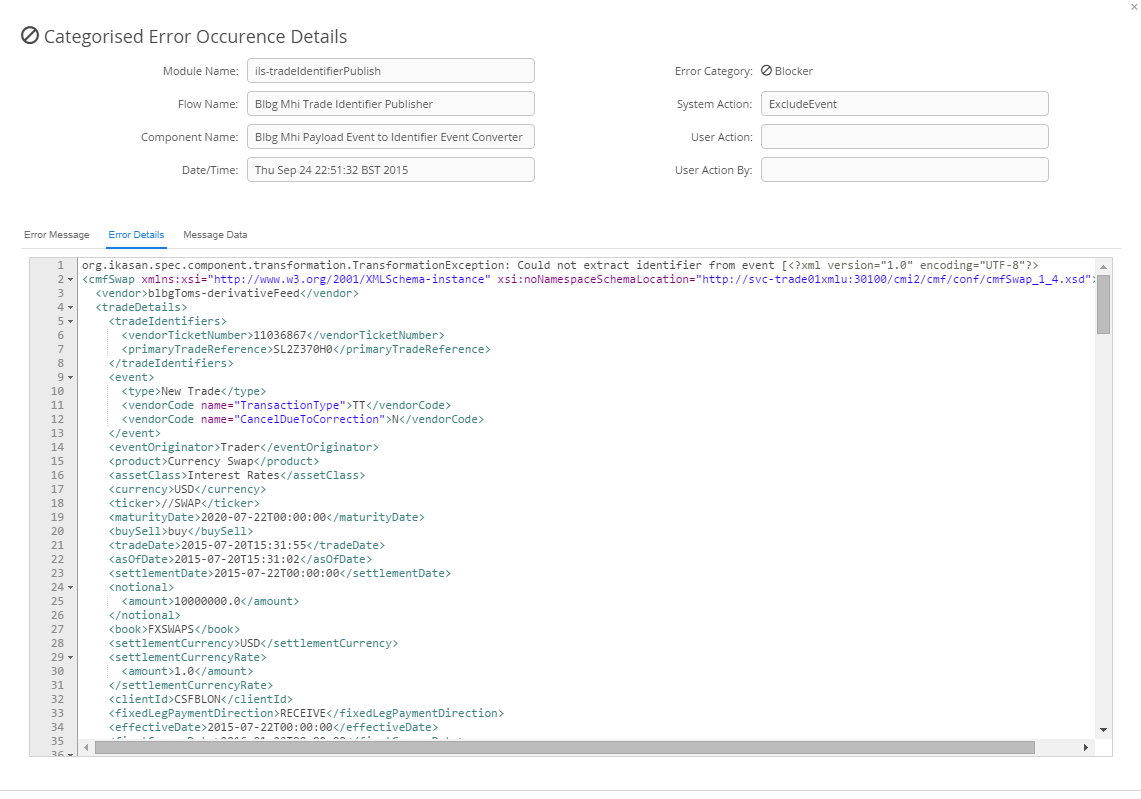
The ‘Categorised Errors Tab’ consists of 2 main sections. The top section allows the user to provide filtering on the ‘Categorised Errors’ search. In order to filter the search ‘Modules’, ‘Flows’ and ‘Components’ can be dragged from the topology tree and dropped into their respective tables. It is also possible to narrow by the date and time that the error occurred. Once the search filter has been defined the user can click the ‘Search’ button to perform the search and results will appear in the table below. The results are colour coded based on the error category associated with them. The search filter can be cleared using the ‘Clear’ button.

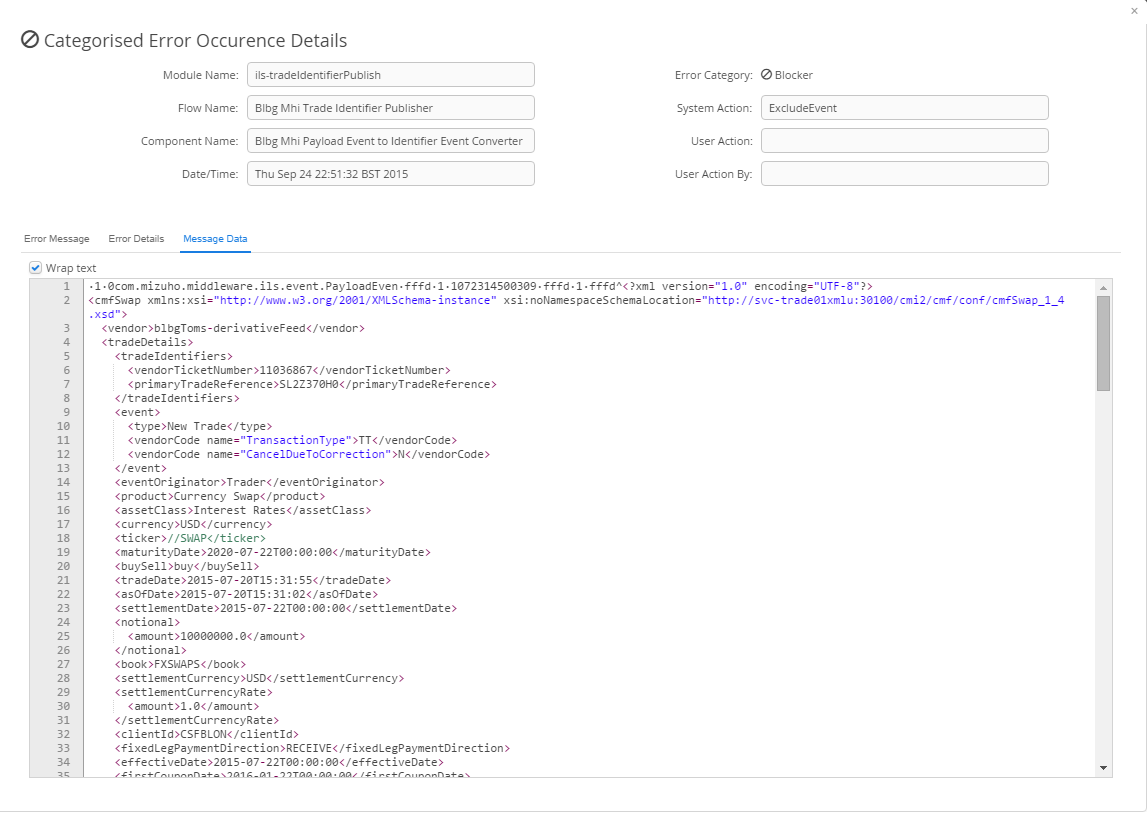


#### Categorised Error Occurrence Details Window

Once a ‘Categorised Error’ search has been performed, it is possible to click on a row within the search results table. This will cause the ‘Categorised Error Occurrence Details’ window to open. The ‘Categorised Error Occurrence Details’ window provides further information regarding the ‘Error’ including the payload of the error event along with the error message and a full java stack trace and user friendly error message.

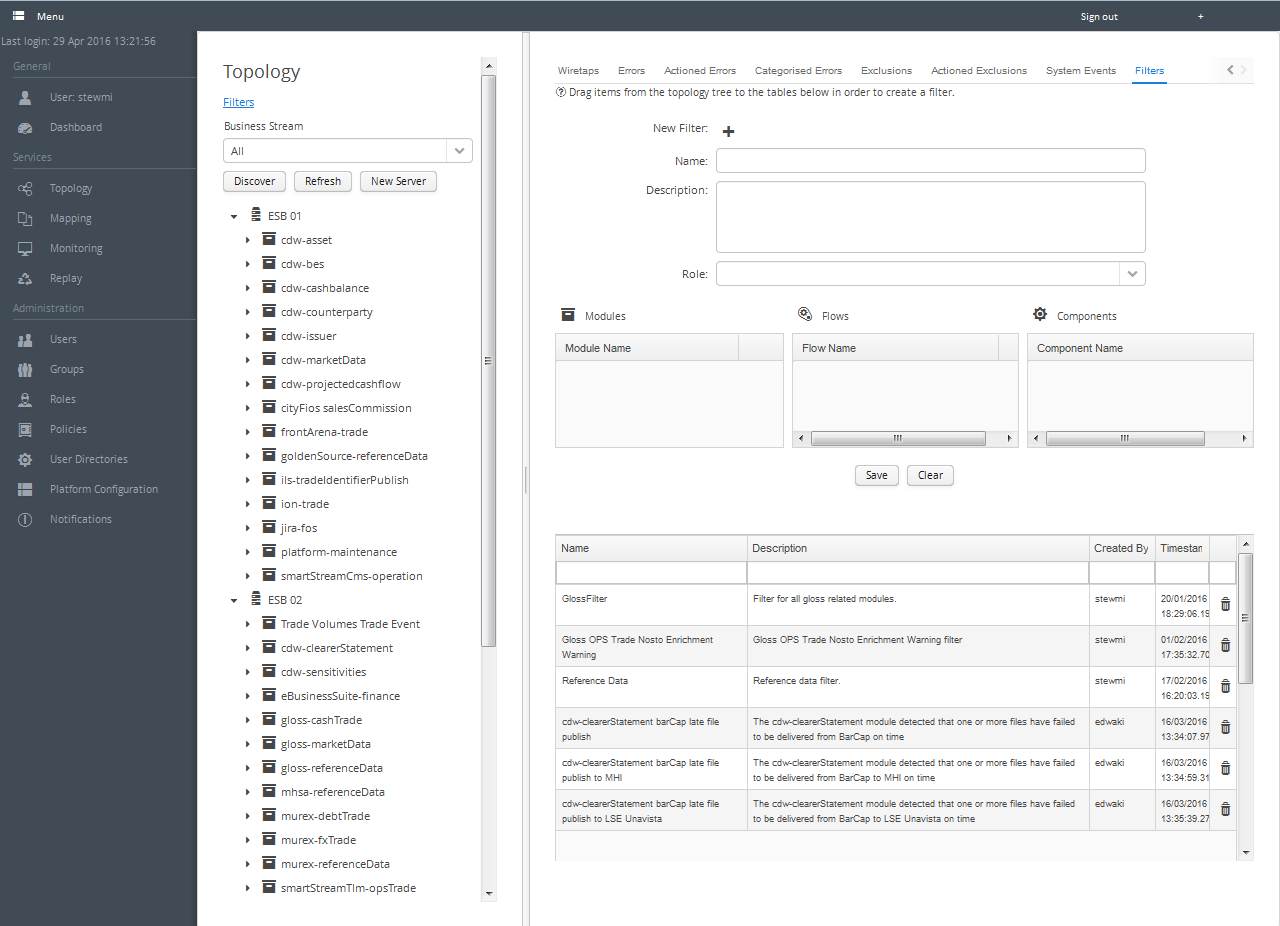




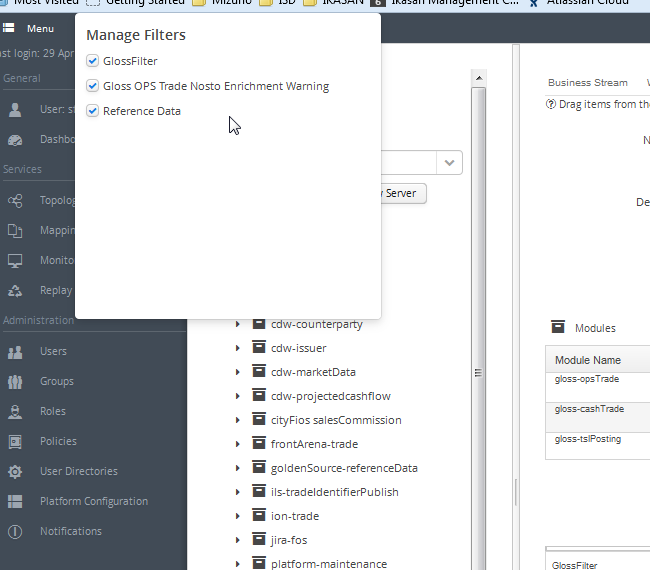


### Filters Tab

Filters are created and managed on the filter tab. A filter is a collection of Ikasan modules, flows and components that are logically grouped within the filter. A filter requires a name and description. If a filter is assigned to a role, then the users who are assigned this role will have this filter automatically applied to any searches that they perform within the dashboard. Modules, components or flows can be added to a filter by being dragged and dropped into the appropriate tables.



If a user has a filter assigned to a role that they have been assigned, they can toggle the use of the filter by clicking the filter link at the top of the topology tree.

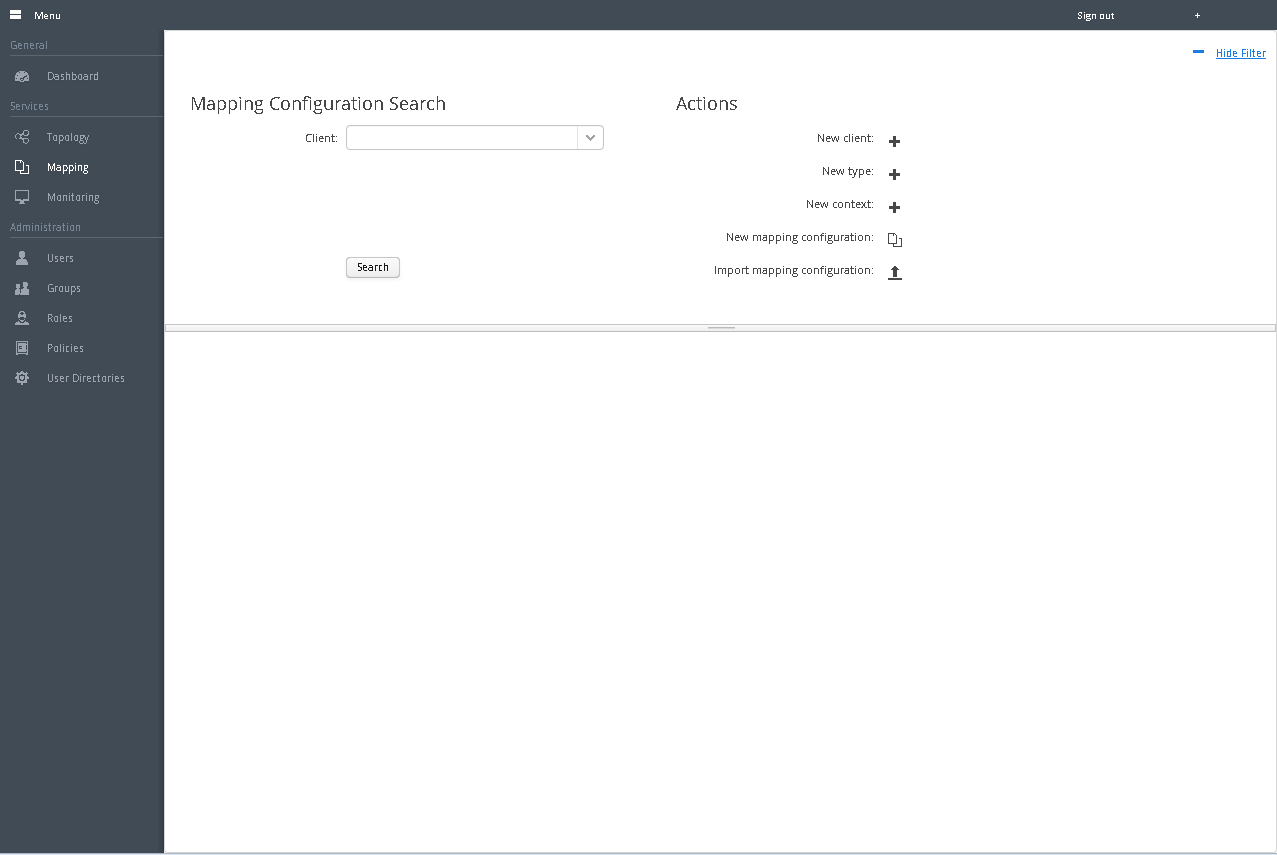
 Mapping View

As part of the Ikasan offering, a Mapping Service is available. The Mapping Service is designed to allow data to be mapped from one context to another. For example, ‘System A’ might represent a product type or a user id in one format whereas ‘System B’ requires a different format. The Mapping Service can have a mapping set up in order to provide the translation form one domain to another.

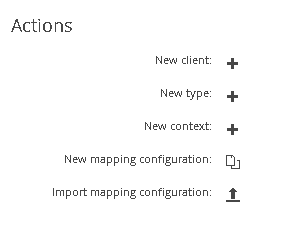
The Mapping Service allows for a mapping whose cardinality is many source values to one target value. Each individually configured mapping is uniquely identified by the following 4 fields:

* **Client.** The client of the mapping service. Is entered as free form text.
* **Type.** The type of the mapping, for example ‘Product Type’. Is entered as free form text.
* **Source context.** The context of the source of the data, for example ‘System A’. Is entered as free form text.
* **Target context.** The context of the target of the data, for example ‘System B’. Is entered as free form text.

The Ikasan Dashboard provides functionality that allows for mappings to be administered.

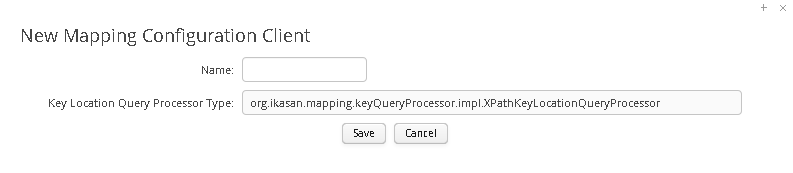


### Mapping Actions

The ‘Mapping’ landing screen has a number of actions available. In order to perform any of these actions, click on the icon associated with the action.

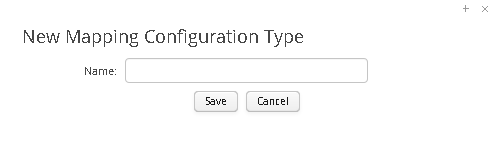
#### Creating a New Mapping Client

When creating a ‘New Mapping Configuration Client’, the user will be presented with a simple window. The user is expected to provide a ‘Name’ which must be unique. The ‘Key Location Query Processor’ field is pre-populated and this should not be changed. Press ‘Save’ and the new client will be persisted into the database and will become available to be used when creating a new mapping.



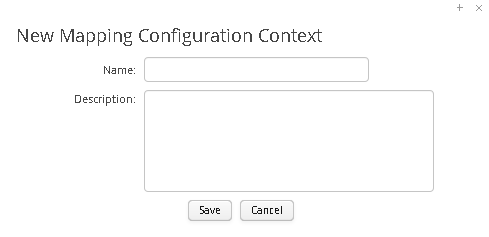
#### Creating a New Mapping Configuration Type

When creating a ‘New Mapping Configuration Type, the user will be presented with a simple window. The user is expected to provide a ‘Name’ which must be unique. Press ‘Save’ and the new type will be persisted into the database and will become available to be used when creating a new mapping.



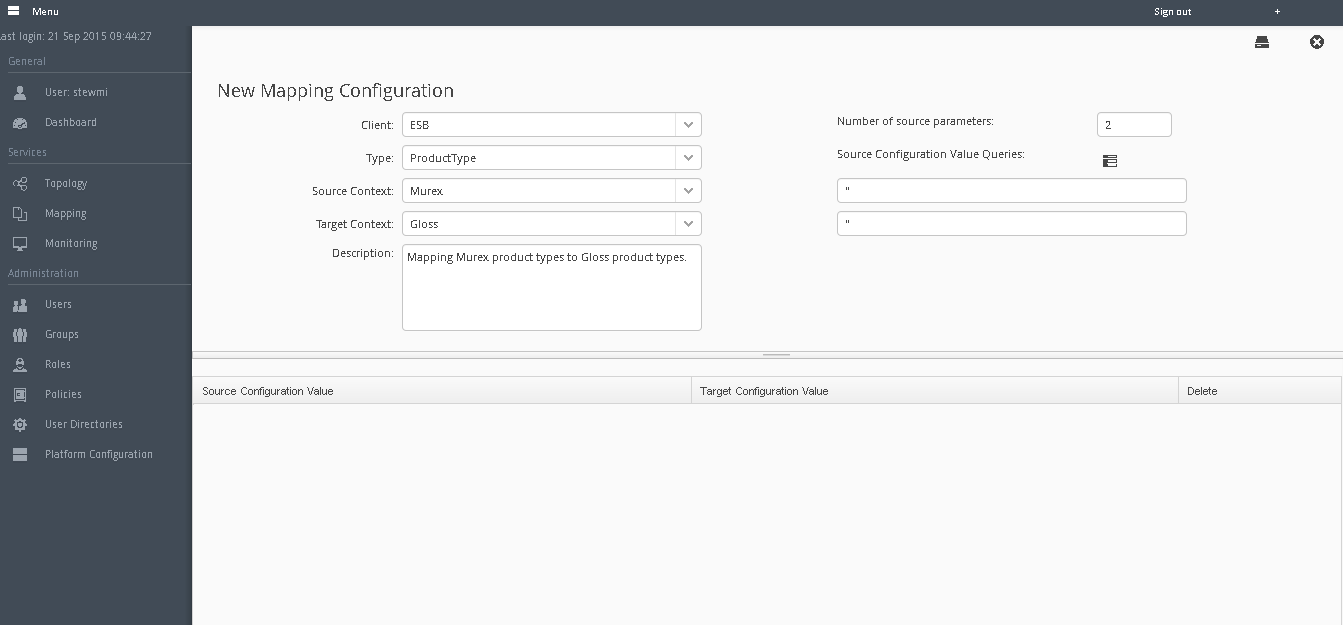
#### Creating a New Mapping Configuraiton Context

When creating a ‘New Mapping Configuration Context’, the user will be presented with a simple window. The user is expected to provide a ‘Name’ which must be unique as well as a description. Press ‘Save’ and the new context will be persisted into the database and will become available to be used when creating a new mapping. A newly created context can be used as both a source or target context when associated with a mapping.



#### Creating a New Mapping

All fields on the ‘New Mapping Configuration’ screen are mandatory. Select the appropriate ‘Client’, ‘Type’, ‘Source Context’ and ‘Target Context’ from the relevant combo boxes. A freeform ‘Description’ is entered in the text area. In order to define the number of source parameters enter the ‘Number of Source Parameters’ into the test area provided. Once done select the  icon to the right and a number of fields will appear. If you are using XPath to narrow the source values for the mapping, then enter the XPath queries into these fields. Otherwise leave as empty strings surrounded by single quotes. Once all of the form fields have been filled, press the  icon to save the new mapping. At any point in time prior to saving, the creation of the new mapping can be cancelled by pressing the  icon.

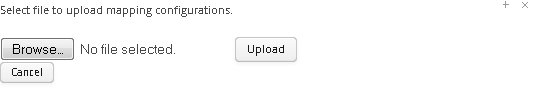


#### Importing a Mapping Configuration

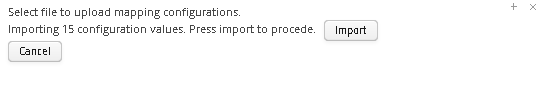
In order to import a mapping it is important to confirm that the ‘Client’, ‘Type’, ‘Source Context’ and ‘Target Context’ for the mapping you intend to import already exist in the mapping service. If not, create these as per the previous steps.

Once all prerequisite steps are complete, press the  icon on the mapping view landing screen. This will cause an import window as seen below.

Press the browse button to select the file containing the mapping xml you wish to upload. The schema that this mapping file must conform to is bundled with the dashboard and can be accessed with the following URL: http:/<dashboard-url>:<dashboard-port>/ikasan-dashboard/static/org/ikasan/dashboard/mappingConfigurationImportExport.xsd.

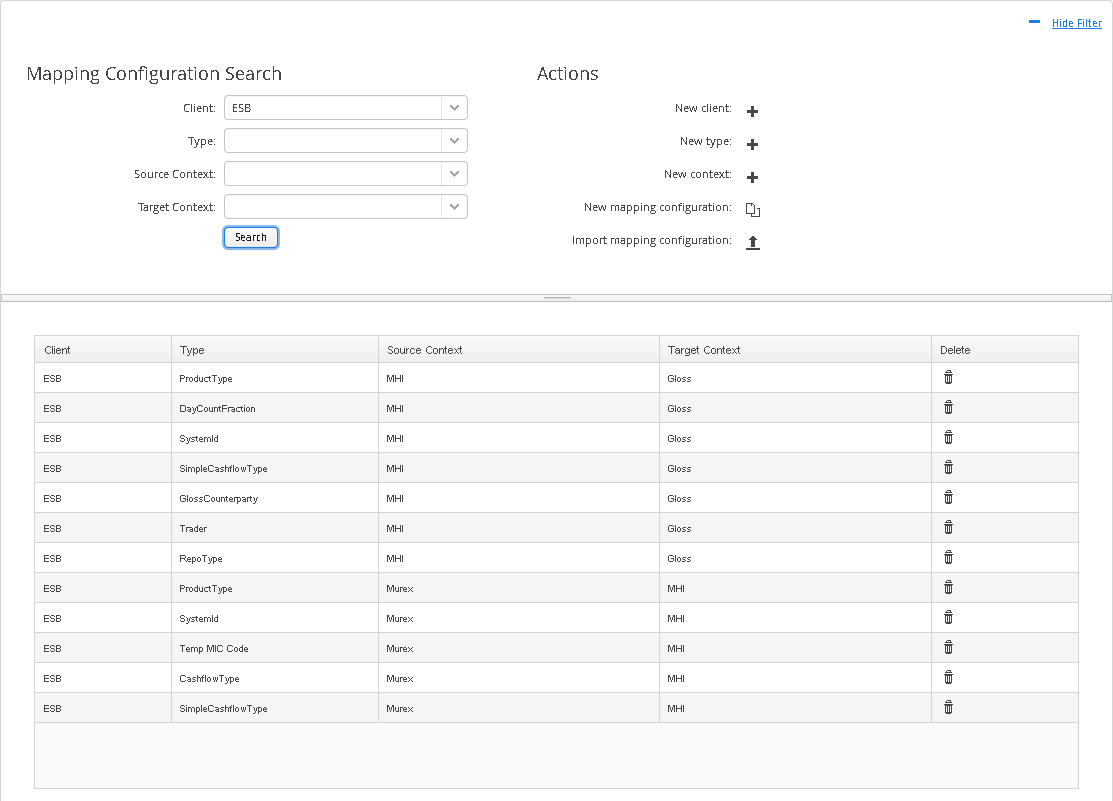


Once the appropriate file has been selected, press the ‘Upload’ button. This will cause the file to be uploaded and processed. The window will then display details about the number of mapping values that will be uploaded. Once happy, press the ‘Import’ button and the imported mapping will be added to the database and can then be [managed](#_Managing_a_Mapping) from the dashboard.



### Mapping Configuration Search

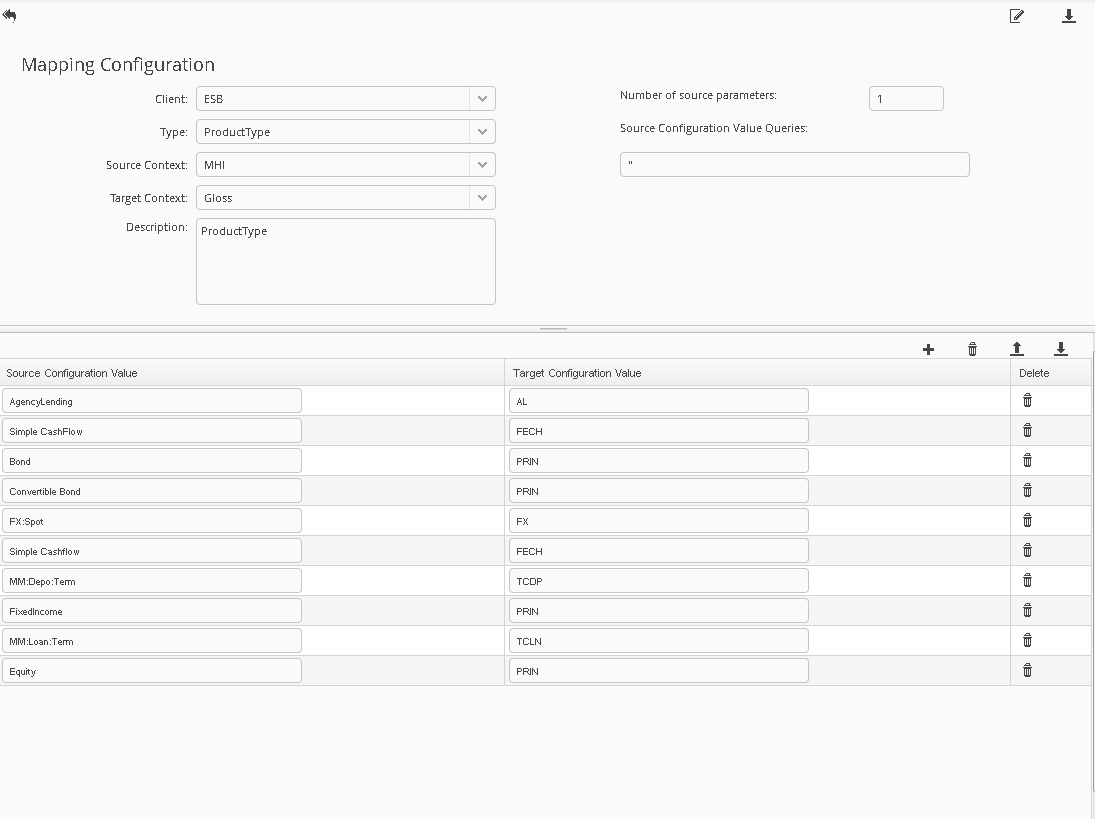
Mappings can be searched for on the ‘Mapping View’ screen. Result sets can be narrowed by selecting from the ‘Client’, ‘Type’, ‘Source Context’ and ‘Target Context’ combo boxes. Once the search criteria have been defined, press the ‘Search’ button. Results will be returned in the lower pane of the screen. Any row in the ‘Search Results’ table can clicked on which will open the mapping so that is can be [managed](#_Managing_a_Mapping).



### Managing a Mapping Configuration

There are a number of controls available which allow a ‘Mapping Configuration’ to be managed. A ‘Mapping Configuration’ is first opened in a read only mode. The following controls allow the mapping to be managed:

*  **Edit**. Clicking on this icon causes the mapping to become editable. All values in the top form can be edited except for the number of source parameters. Also the table of source and target values in the bottom table can be edited.
*  **Download Mapping**. The download icon in the top right hand corner of the screen allows for the user to download the mapping.
*  **Save**. When the mapping is in the editable mode, the save icon becomes available. When clicked this save any changes made to the mapping to the database.
*  **Cancel**. When the mapping is in the editable mode, this button will cancel the edit action and will revert the mapping to the state it was in prior to editing.

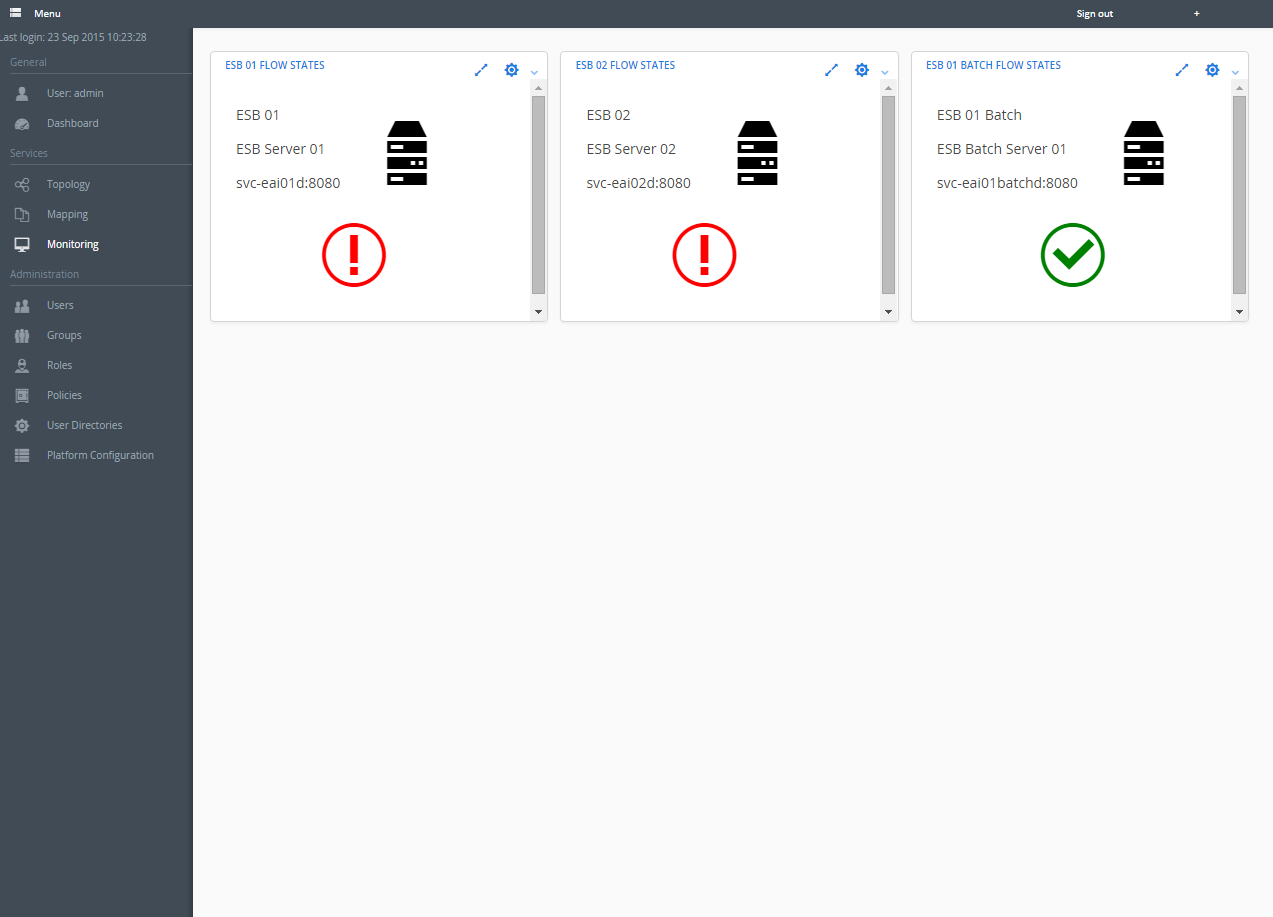


The following controls above the source and target values on the lower part of the screen are focused on items in this table.

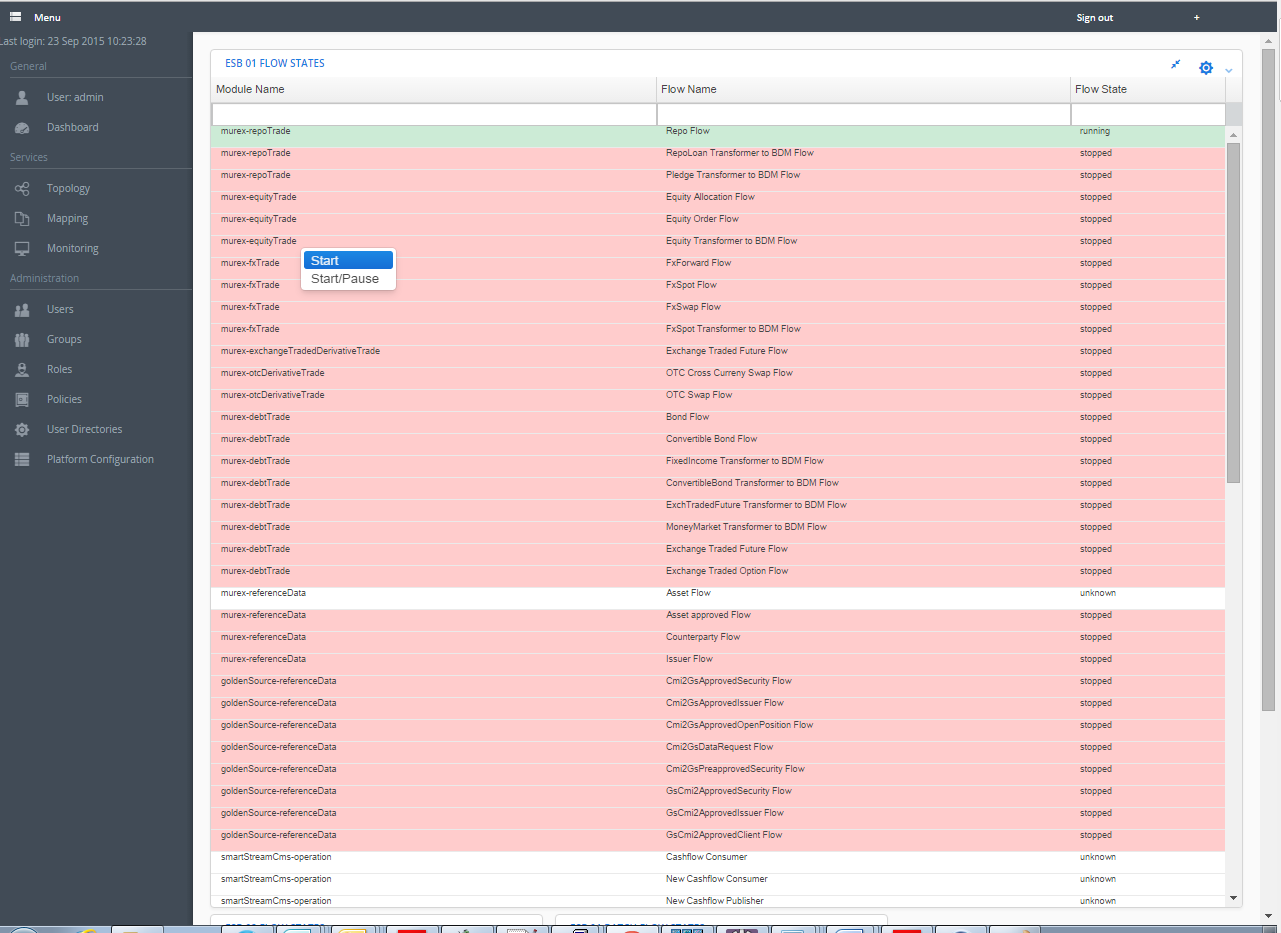
*  **Add**. When clicking on this item, a new row will appear in the mapping values result table below, allowing the user to add new mapping values. Once new values have been added, the user MUST save the mapping for these to be persisted to the database.
*  **Delete**. The delete icon at the top of the mapping values table will delete all values associated with the mapping. The delete icon located in each row of the mapping values table will only delete the values for which the icon resides.
*  **Download**. The download button allows for the values component of the mapping to be download
*  **Upload**. The upload button allows for mapping values to be uploaded. The document being uploaded must conform to http:/<dashboard-url>:<dashboard-port>/ikasan-dashboard/static/org/ikasan/dashboard/mappingConfigurationValuesImportExport.xsd

## Monitoring View

The monitoring view provides a heads up view of the state of the servers that have been configured in the ‘Topology View’. The status that is reported is quite basic in its current form. If all Ikasan modules that are deployed to that server are running, then a green tick will indicate that that particular server is healthy. If however a server has a module which is stopped, stopped in error, paused or recovering, then red explanation mark indicates that that server is not healthy.



It is then possible to drill down into the server itself, by selecting the expand icon associated with the server box that you wish to see further details within. When drilling down, the server box will expand and show all flows which are represented as a row in the table that is displayed. The row will be colour coded to so that it becomes quickly apparent as to which flows are not in a green running state. The table can be sorted and filtered so that it is easy to isolate individual flows that are of interest.



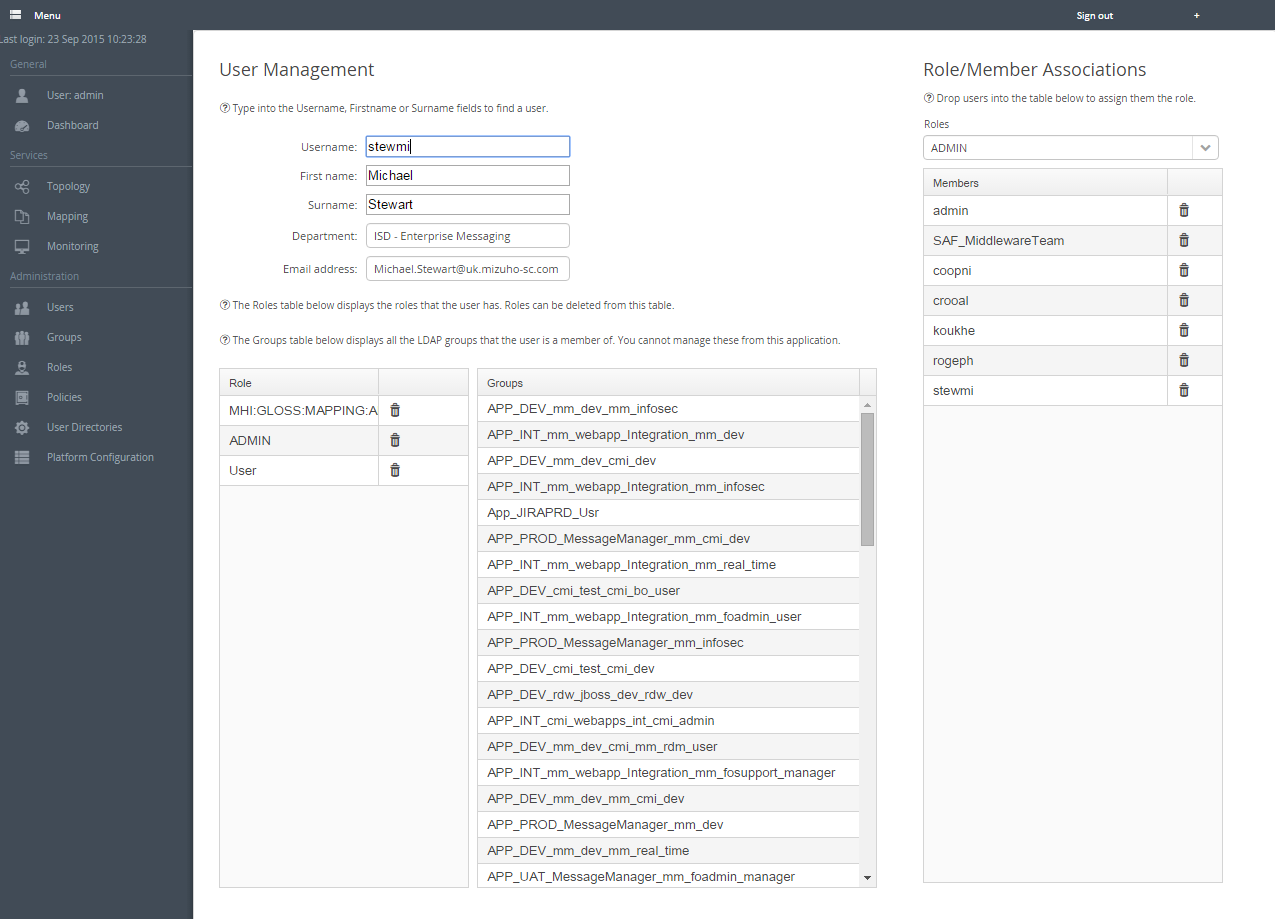
# Administration

## User Administration

In order to find a user it is possible to begin to type the user’s username, first name or surname into the appropriate field on the ‘User Management’ form. There is pop up a suggestion window for the list of users who match the data that has been entered. From this suggestion list, select the user whose details you wish to manage. Once selected all relevant details of the user will be populated into the screen.

From this screen it is possible to assign or remove roles for the user and also to provide visibility on the LDAP groups the user is a member of if the application has been synchronised with an LDAP server.

In order to assign a role to a user, select the role you wish to assign from the ‘Roles’ combo box in the right hand pane on the screen. All users who have this role will then appear in the roles table below. Drag the username text box and drop it onto the roles table. The user will now have that role. This role will appear in the ‘Role’ table below the ‘User Management’ form which displays all roles that the current user has.

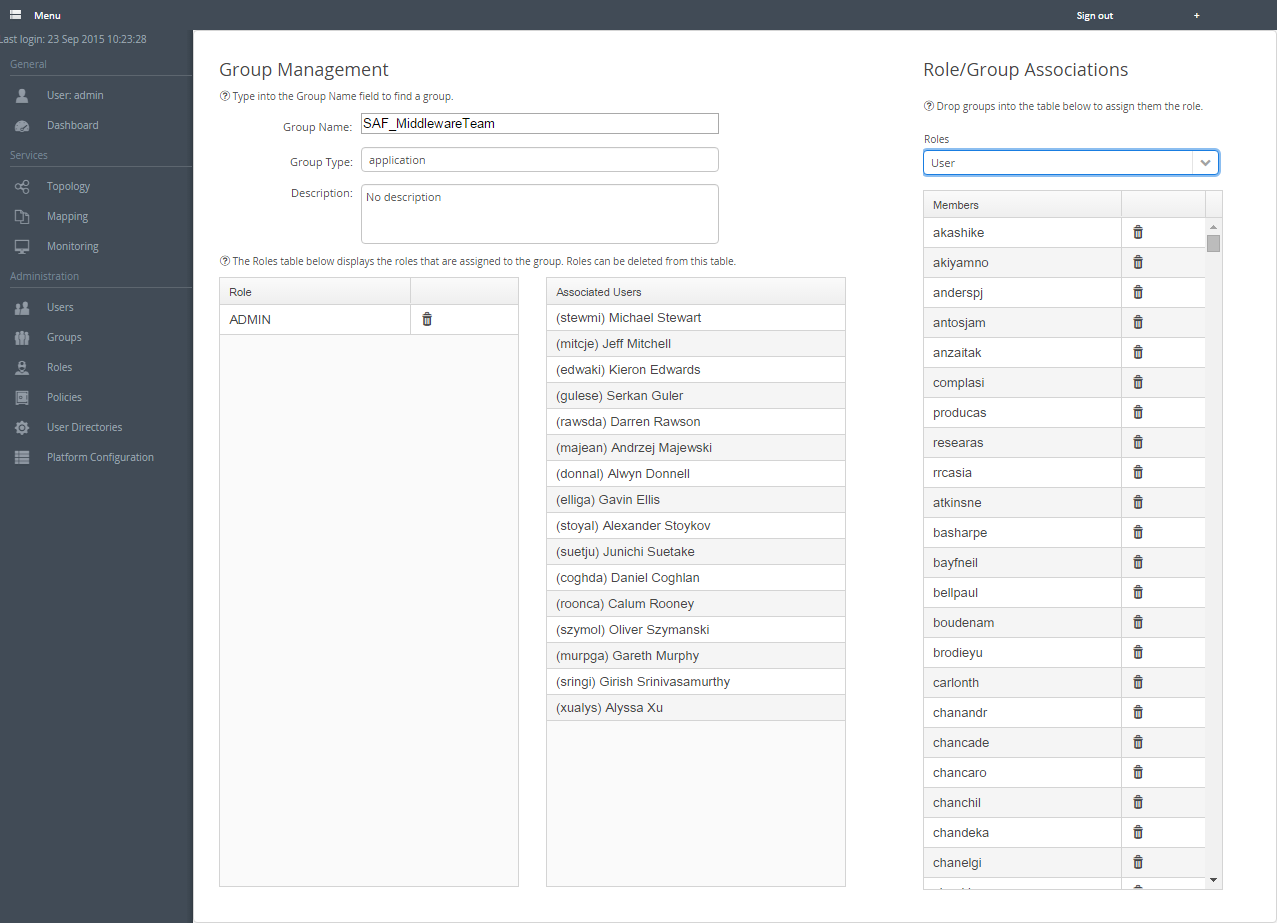


In order to remove a role that a user currently has, the trash can in the ‘Role’ table below the ‘User Management’ form which displays all roles that the current user has can be clicked and the User will no longer have the role. Alternatively the trash can in the Members table on the right hand side of the screen, can be clicked to remove the desired user.

## Group Administration

The ‘Group Administration’ screen works in a very similar way to the ‘User Administration’ screen. Groups can be searched for by beginning to type the name of the group into the ‘Group Name’ text box. This will present a number of suggestions from which the group you wish to manage can be selected. Once selected, the screen will be populated with relevant information for the group.

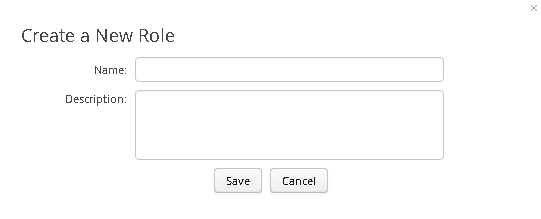
Like users, a group can be given a role. This is done by selecting the role you wish to assign the group from the ‘Roles’ combo box and dragging and dropping the ‘Group Name’ text box into the ‘Members’ table.



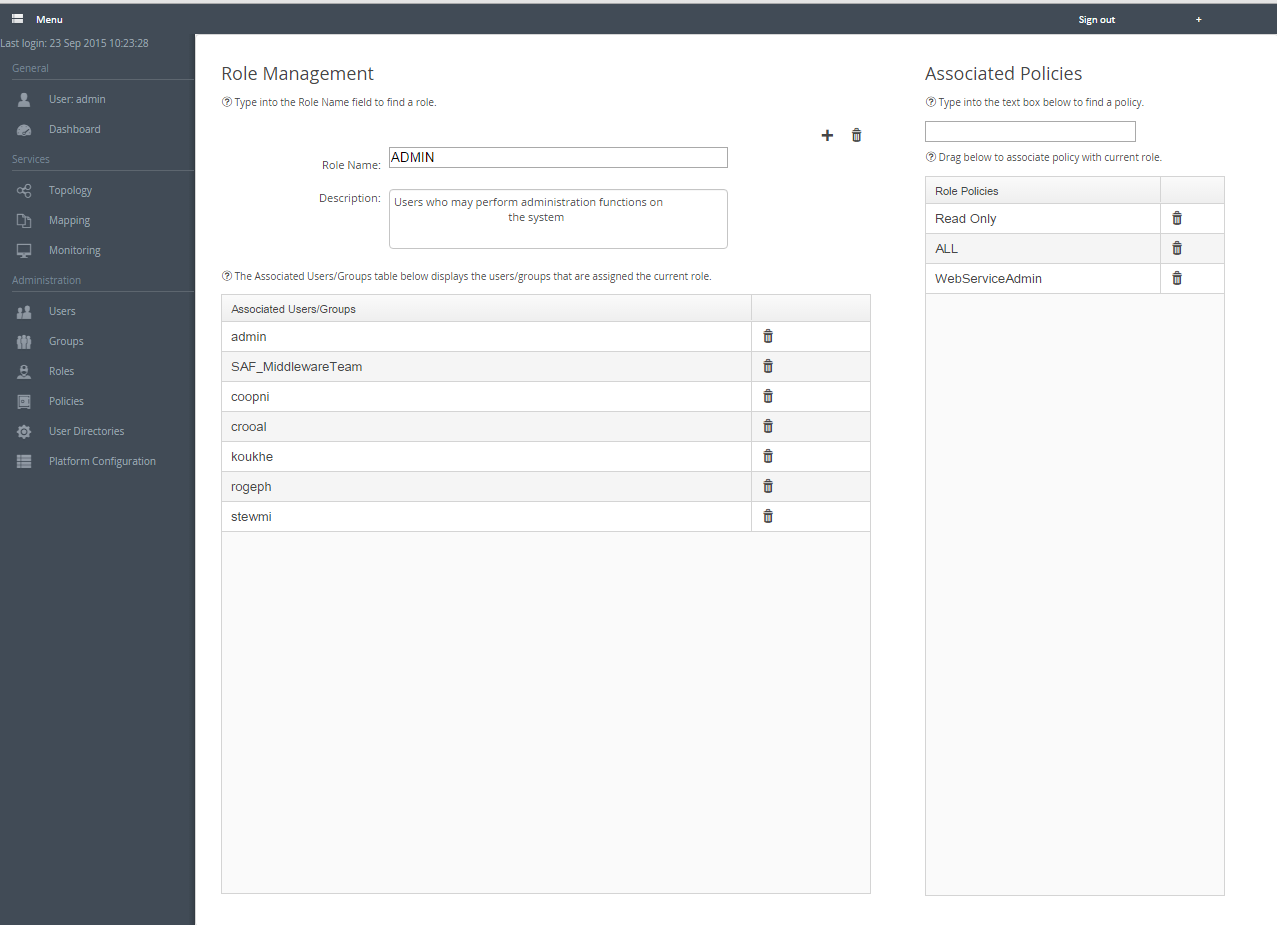
In order to remove a role that a group currently has, the trash can in the ‘Role’ table below the ‘Group Management’ form which displays all roles that the current group has, can be clicked and the User will no longer have the role. Alternatively the trash can in the Members table on the right hand side of the screen, can be clicked to remove the desired ‘Group’.

## Role Administration

A Role is a grouped set of policies that can be assigned to a User or a Group. The ‘Role Administration’ provides the ability create and delete a role as well as associate policies with a role. In order to add a role, press the  icon. The user will then be presented with a window in which a new role can be created. Enter the name and description of the role and press save.



In order to delete a role select the icon to the right of the ‘Role Management’ form.



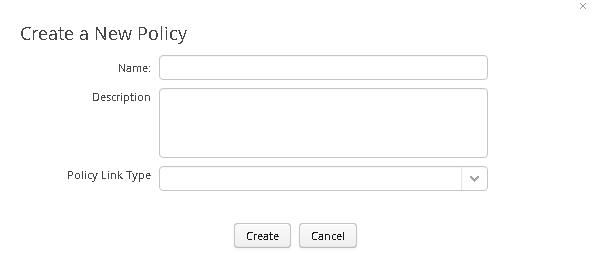
It is also possible to search for existing roles by beginning to type into the ‘Role Name’ text box. Suggestions will be presented and it is possible to select the role you searching for. The screen will be populated with details relating to the relevant role.

The ‘Associated Users/Groups’ table provides a view on all user and groups who have the current role. It is possible to select the  in any row in this table and disassociate the user or group from this role.

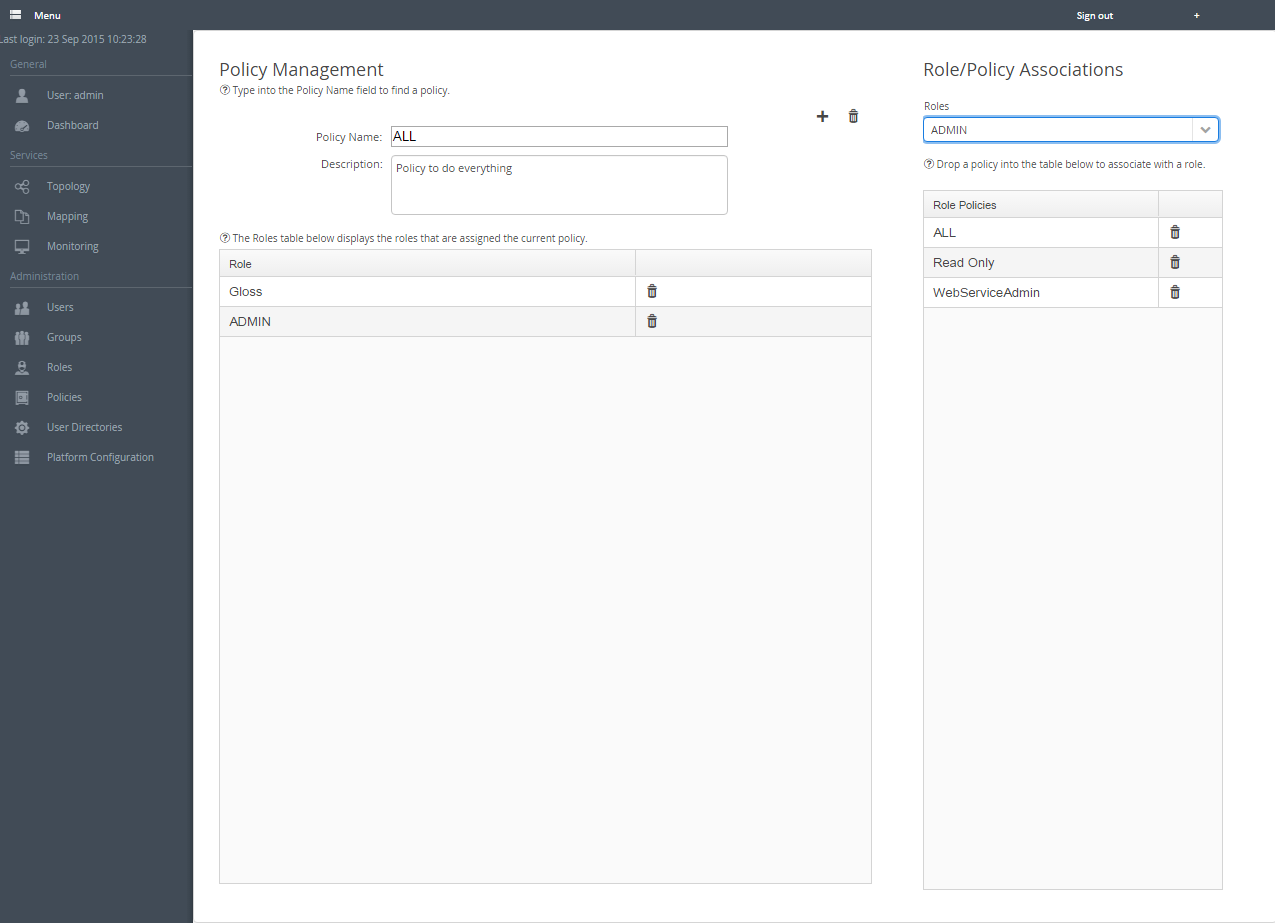
The ‘Associated Policies’ pane allows for policies to be associated with the current role. This is done by typing the policy name into the policy text box and selecting the appropriate suggestion. Once selected drag the policy text box into the ‘Role Policies’ table below and the current role has now been assigned that policy.

## Policy Administration

A Policy is the most granular security element and typically manages a single piece of functionality or related functionality. The ‘Policy Administration’ provides the ability create and delete a policies as well as associate policies with a role. In order to add a policy, press the  icon. The user will then be presented with window in which a new policy can be created. Enter the name and description of the role and press save.



In order to delete a policy select the icon to the right of the ‘Policy Management’ form.



It is also possible to search for existing policies by beginning to type into the ‘Role Name’ text box. Suggestions will be presented and it is possible to select the policy you searching for. The screen will be populated with details relating to the relevant policy.

The Role table below provides a view on all roles that have the current policy. It is possible to select the  in any row in this table and disassociate the role from this policy.

The ‘Role/Policies Associations’ pane allows for the current policy to be associated with the selected role. This is done by selecting the role you wish to add the policy to from the ‘Roles’ combo box. Once selected drag the policy text box into the ‘Role Policies’ table and the current policy has now been assigned to the selected role.

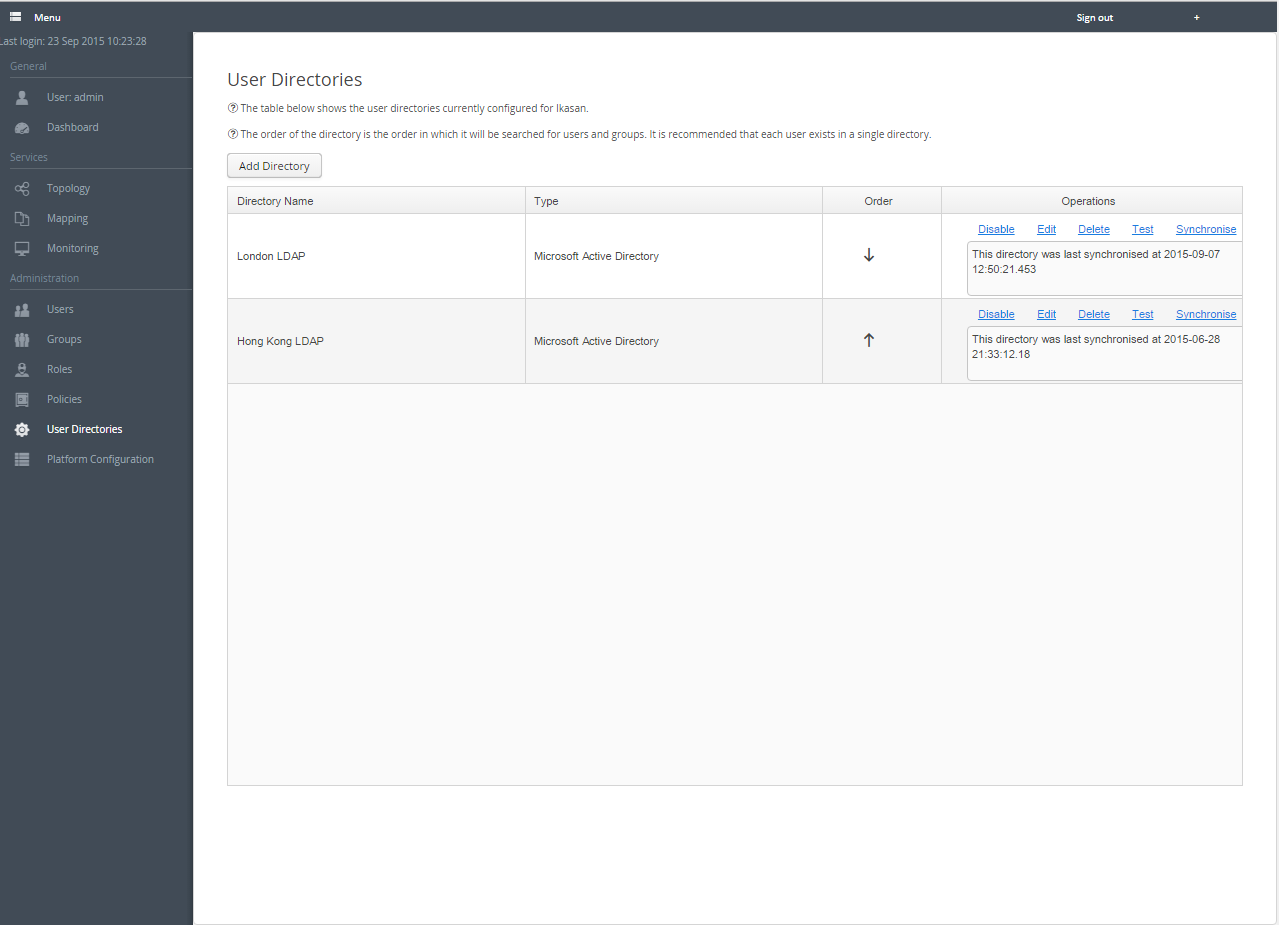
## Managing User Directories

The Dashboard provides functionality that allows it to connect and synchronise with one or more LDAP repositories to allow to Active Directory authentication as well as the ability to pull LDAP groups into the Dashboard domain to allow for different levels of security control within the application.

There are a number of functions available to a configured ‘User Directory’.

* Enable/Disable. This enables or disables authentication against this directory. The directory must be enabled to allow users to authenticate against the repository.
* Edit the ‘User Directory’ configuration.
* Delete the ‘User Directory’ configuration.
* Test the connectivity the ‘User Directory’
* Synchronise the dashboard with the ‘User Directory’. This should be done once the directory has been set up. Without this step, users will not be able to authenticate against the repository.

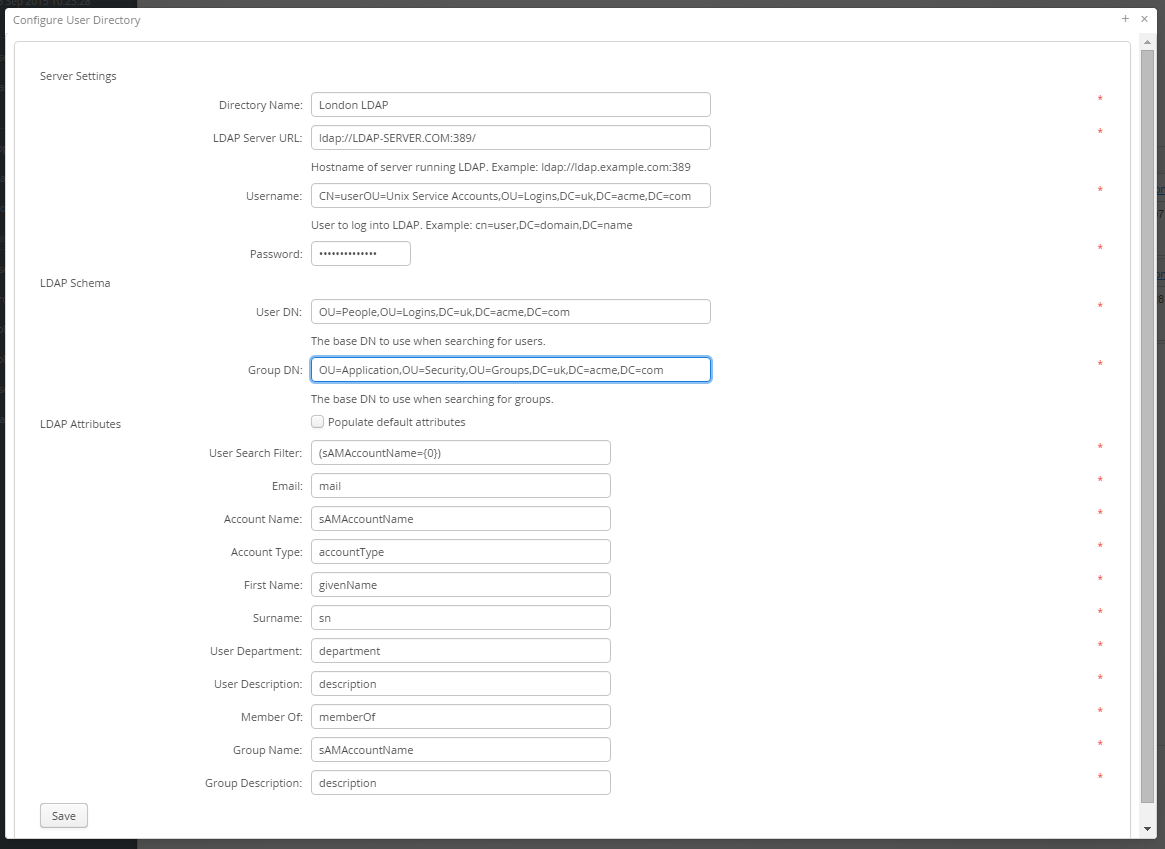
The user can also choose to add a directory.



The ‘User Directory’ is presented whenever a user either creates a new or edits an existing ‘User Directory’.

The form is broken into distinct areas:

* ‘Server Settings’ contains the details of how to connect to the directory.
* ‘LDAP Schema’ contains details of where to get user and group information from within the directory.
* ‘LDAP Attributes’ contain the names of attributes from which data elements are retrieved from during the synchronisation process. You can have this filled with default values which is sufficient in most cases, by selecting the Populate default attributes check box.

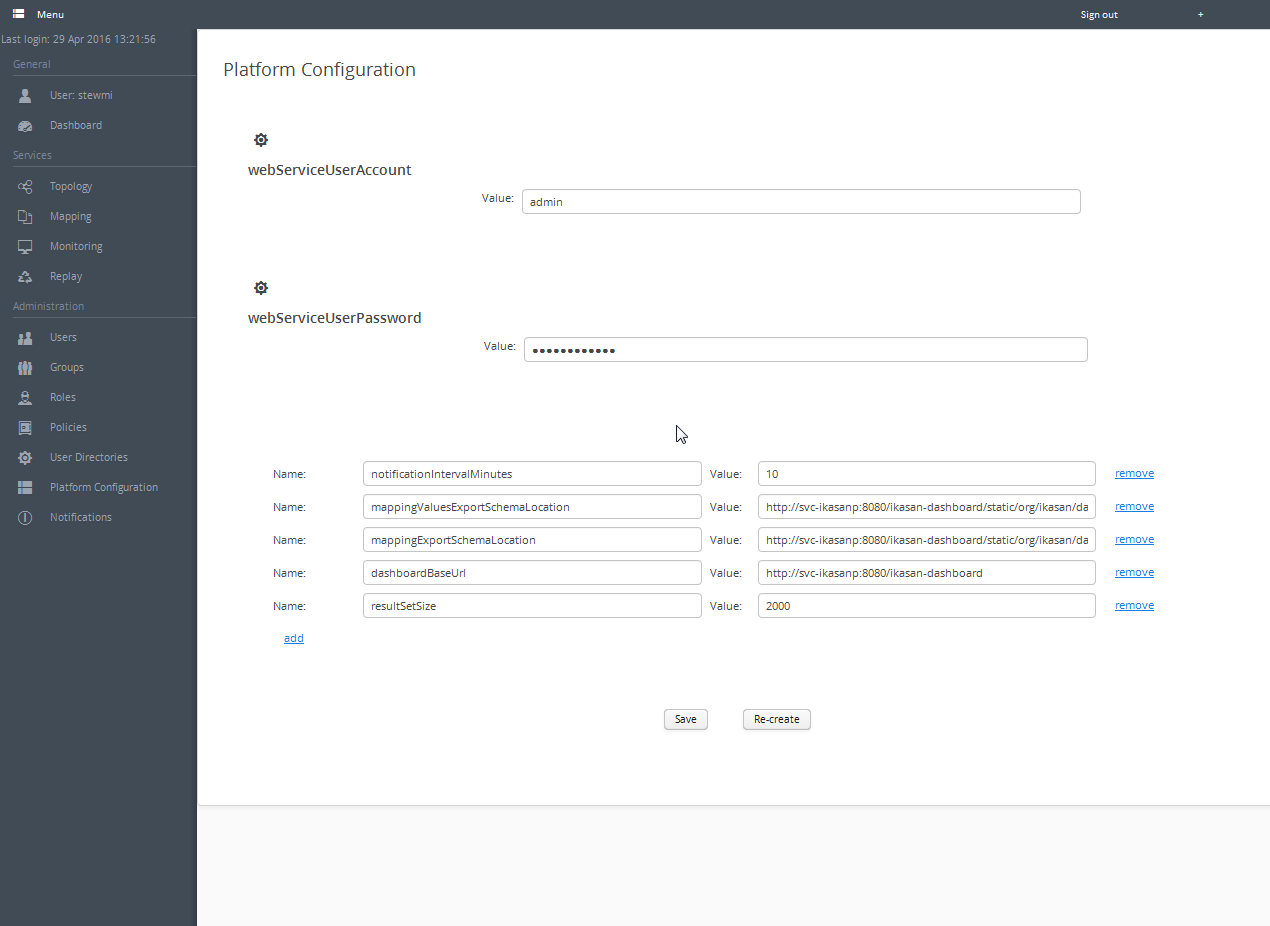


## Platform Configuration

The Platform Configuration screen provides a place to externalise global configurations for the Ikasan platform including the Dashboard.

These configurations are stored in the Ikasan configuration service with a configured resource id of ‘platform-configuration’.

There are a number of ‘Platform Configuration’ fields required to be entered:

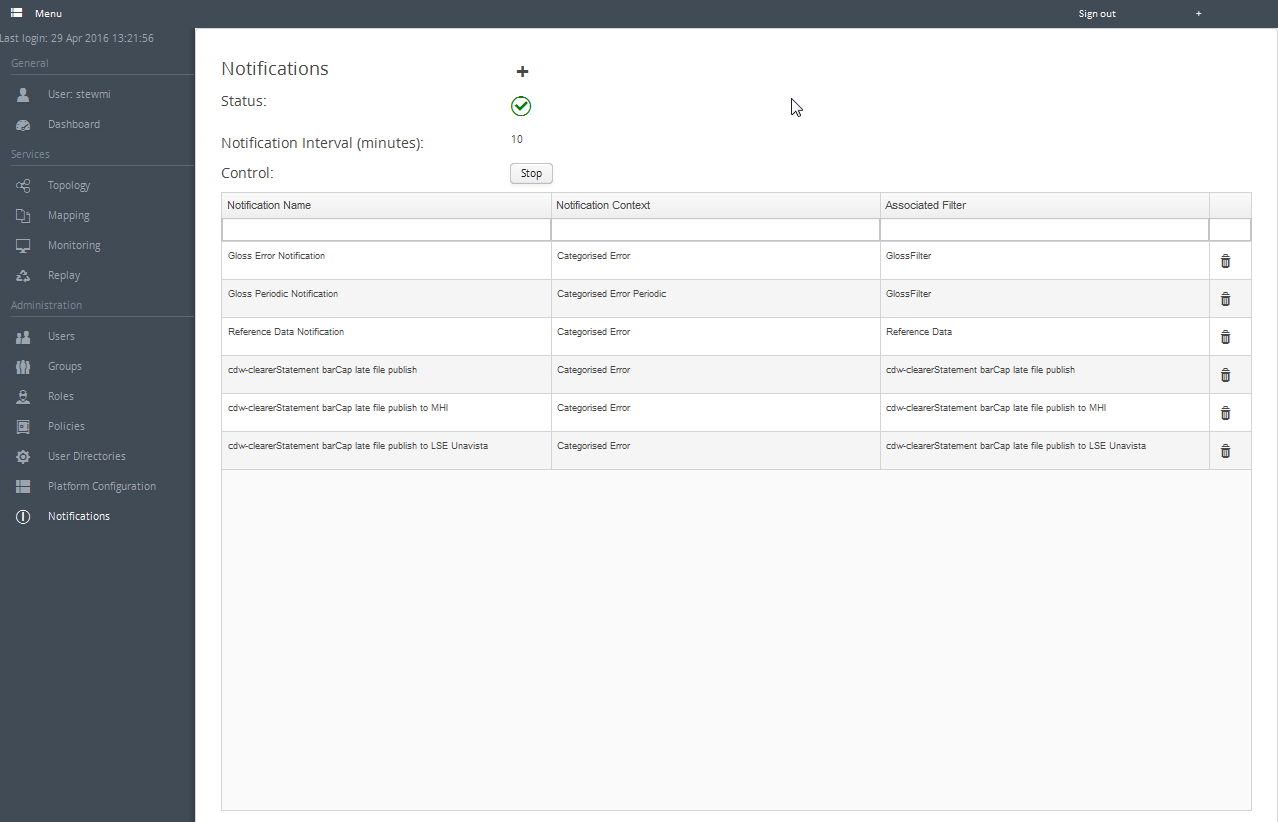
1. **webServiceUserAccount**: The username of any account who has a role containing the ‘ALL’ or ‘WebServiceAdmin’ policy.
2. **webServiceUserPassword**: The password of the above account.
3. **mappingExportSchemaLocation**: The path to the export schema in the form http:/<dashboard-url>:<dashboard-port>/ikasan-dashboard/static/org/ikasan/dashboard/mappingConfigurationImportExport.xsd#
4. **mappingValuesExportSchemaLocation**: The path to the export schema in the form http:/<dashboard-url>:<dashboard-port>/ikasan-dashboard/static/org/ikasan/dashboard/mappingValuesConfigurationImportExport.xsd
5. **dashboardBaseUrl**: http:/<dashboard-url>:<dashboard-port>/ikasan-dashboard
6. **resultSetSize:** the number of results to be returned to the dashboard in a search.
7. **notificationIntervalMinutes**. The interval in minutes between dashboard notifications being sent.

## Notifications

The notification service allows for notifications to be sent from the dashboard. The concept of notifications is a follows.

*A notification is associated with a filter. If a categorised error is raised against a module, flow or component that is contained within the filter that is associated with a notification, then that notification will be sent. Currently only email notifications are supported, however there are plans to extend this into other notification types such as SMS or even the automatic creation of JIRA tickets.*

Notifications run as a background thread in the dashboard and the thread runs on a scheduler. They are scheduled bashed on the value of notificationIntervalMinutes that is set on the platform configuration screen. The control button allows you to stop and start the notification thread. PLEASE NOTE. If you change the notificationIntervalMinutes configuration value, you will need to stop and start the notification thread in order for the new interval to be applied to the scheduler.



### Configuring a Notification

A notification requires a name, context, filter, comma separated list of recipients, subject and a body.

There are 2 types of contexts.

1. Categorised Error: If this context is selected, then a notification will be sent if a categorised error is raised in the period since the notification thread last fired.
2. Categorised Error Periodic: If this context is selected, then a notification will be sent if a categorised error is raised in the period since the notification thread last fired. It will then continue to send a notification every hour until the categorised error is cleared.

