

## HCL Detect v12.1.9 Admin Guide



# Contents

- Chapter 1. Introduction..... 3
- Chapter 2. User & Role Management..... 4
- Chapter 3. System Health..... 17
- Chapter 4. Applications, Aggregates & Enrichments..... 19
- Chapter 5. Subscriber Segments.....26

# Chapter 1. Introduction

This guide helps administrators to configure user roles and permissions, monitor feed applications and system health, and manage subscriber segments. The administrator can access the Administration menu in the Header to navigate to User roles and management, System health and Settings option to manage the application.

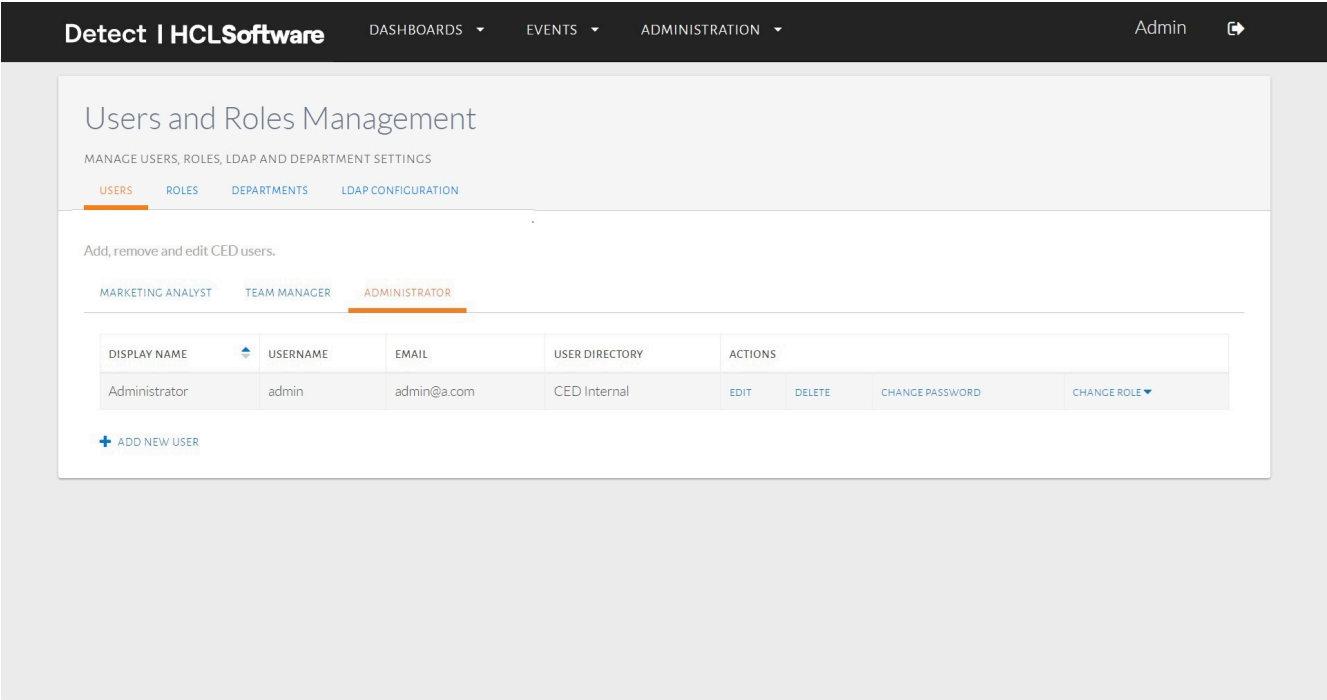
# Chapter 2. User & Role Management

## USERS

The only user that comes pre-configured with HCL Detect is the `admin` user.

The `admin` user can add additional users, including other administrators, as well as perform other user management tasks by accessing the `User And Role Management` link located under the `Administration` category in the top navigation bar.

The user management screen is shown below:



The `UserAndRoleManagement` page.

In the `Users` tab, the various roles that are configured in the system and the users that exist for each such role are depicted.

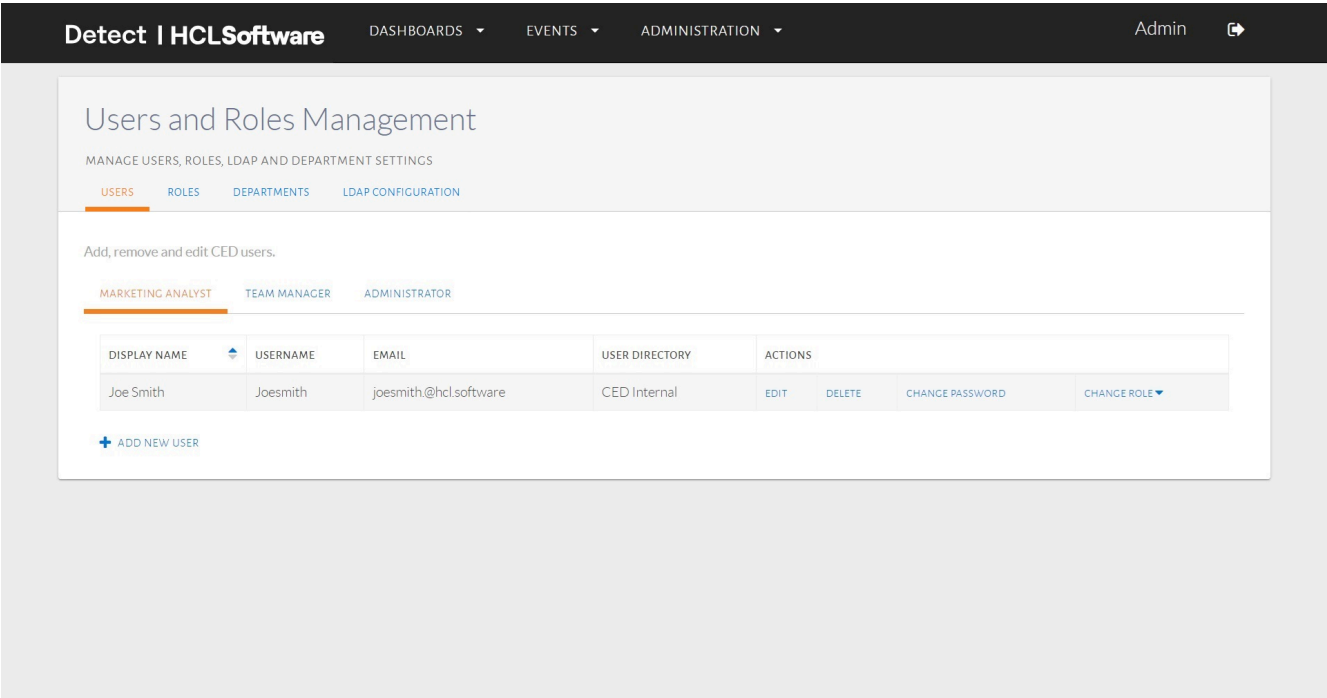
To add a new user, the administrator must first decide the role for such a user, for example, the `MarketingAnalyst` role can be selected by picking the corresponding tab in the `User Management` page and by clicking on the `Add New User` link:

The screenshot shows the 'Detect | HCLSoftware' user management interface. The top navigation bar includes 'DASHBOARDS', 'EVENTS', and 'ADMINISTRATION'. The 'ADMINISTRATION' dropdown is open, showing 'MANAGE USERS, ROLES, LDAP AND DEPARTMENT SETTINGS'. Below this, there are tabs for 'USERS', 'ROLES', 'DEPARTMENTS', and 'LDAP CONFIGURATION'. The 'USERS' tab is selected, and the sub-tab 'MARKETING ANALYST' is active. The main content area is titled 'Add, remove and edit CED users.' and contains a form for adding a new user. The form fields are: USERNAME (with a red border and a message 'A valid username is required (alphanumeric, no spaces).'), EMAIL, FIRST NAME, LAST NAME, DISPLAY NAME (with a message 'This is the name displayed to other users system wide.'), and PASSWORD (with a 'SHOW' button and a message 'The user password must be at least 6 characters in length, must contain at least one number, must contain at least one upper case letter.'). At the bottom of the form are 'SAVE' and 'CANCEL' buttons.

Adding a new user.

Subsequently, the Username, Email, First Name, Last name, Display name and Password attributes should be filled out. Finally, the Save button can be pressed to store this data.

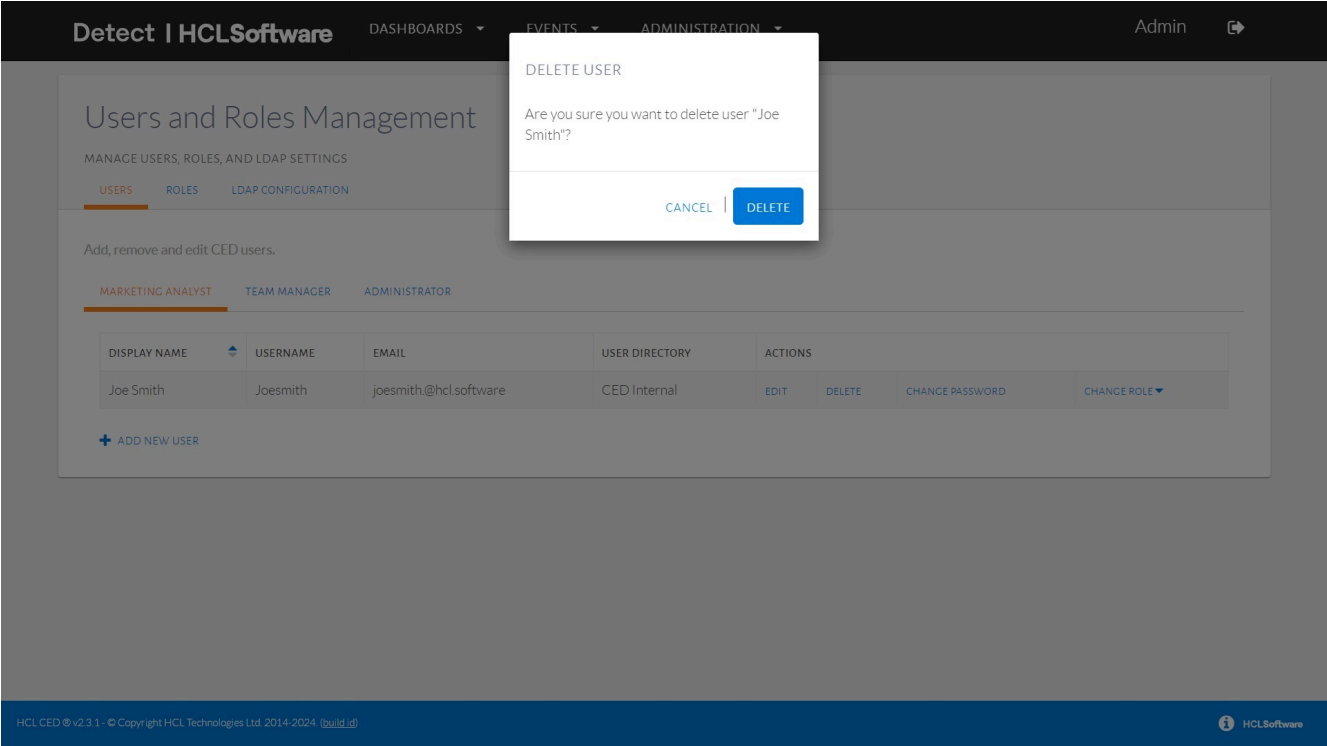
Once completed, the newly added user should be visible in the user interface:



A newly added user.

The data for an existing user can be modified and the user itself can be removed by another user with user management entitlement.

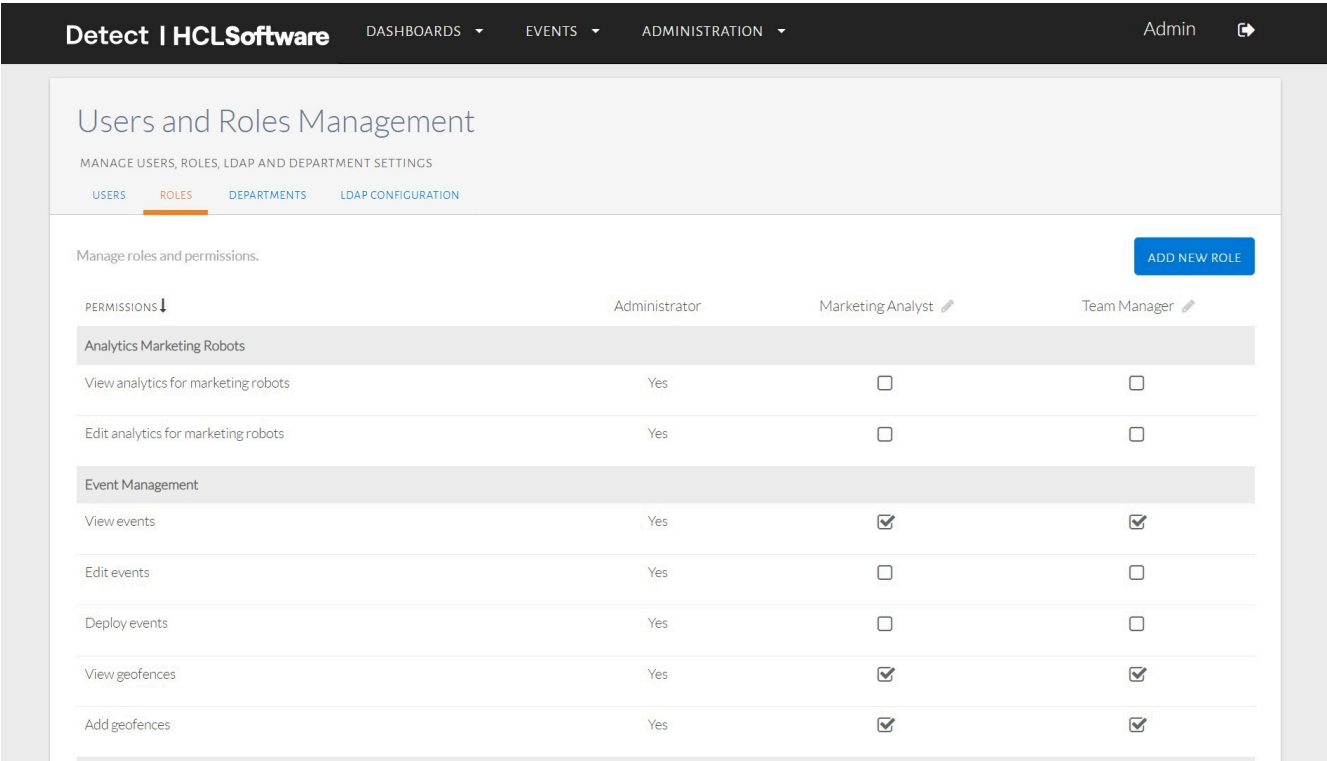
The deletion and editing are performed using `Delete` and `Edit` button under `Actions` head of the table:



Removing or updating a user's data.

**ROLES**

The `Roles` page can be accessed by accessing the `Roles` tab in the navigation bar:

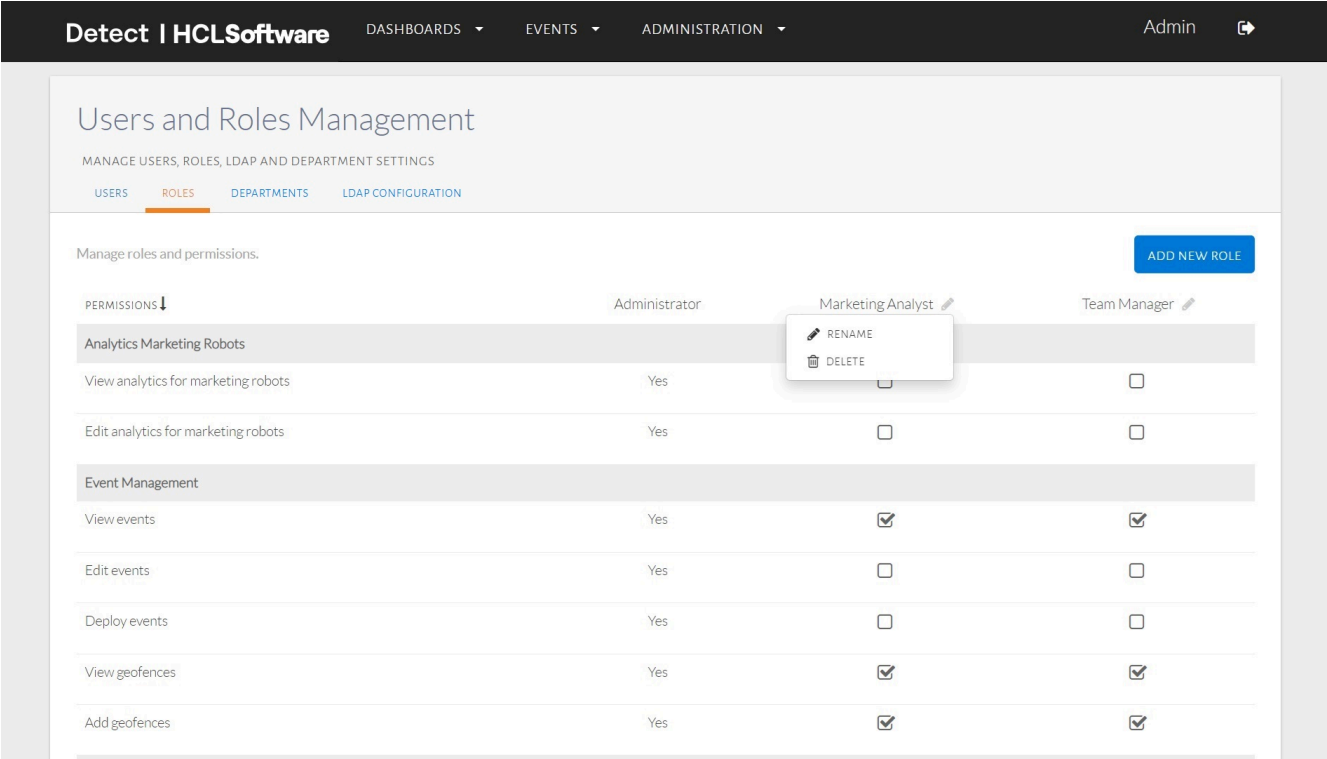


The HCL Detect, by default, comes with three default roles: the Administrator, the Team Manager and the Marketing Analyst:

- The Administrator has unrestricted permissions in HCL Detect.
- The TeamManager does not have the permissions associated with user and role management, but has all the other permissions.
- The MarketingAnalyst has permissions that are a notch below a Team Manager, excluding, for instance, the ability to edit the feeds.

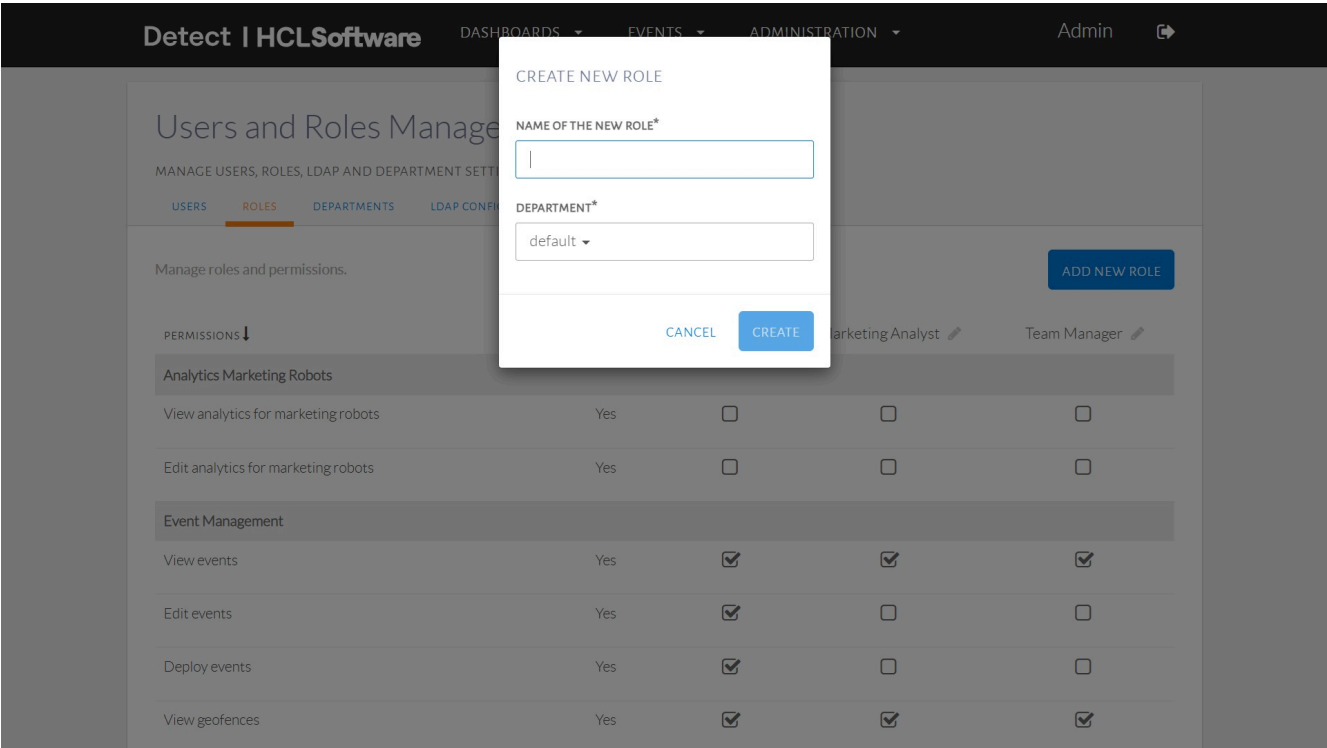
The removal of a role can only be accomplished if no user(s) with that role exists. The user interface shows an appropriate error when an attempt is made to delete a role that has user(s) assigned to it:



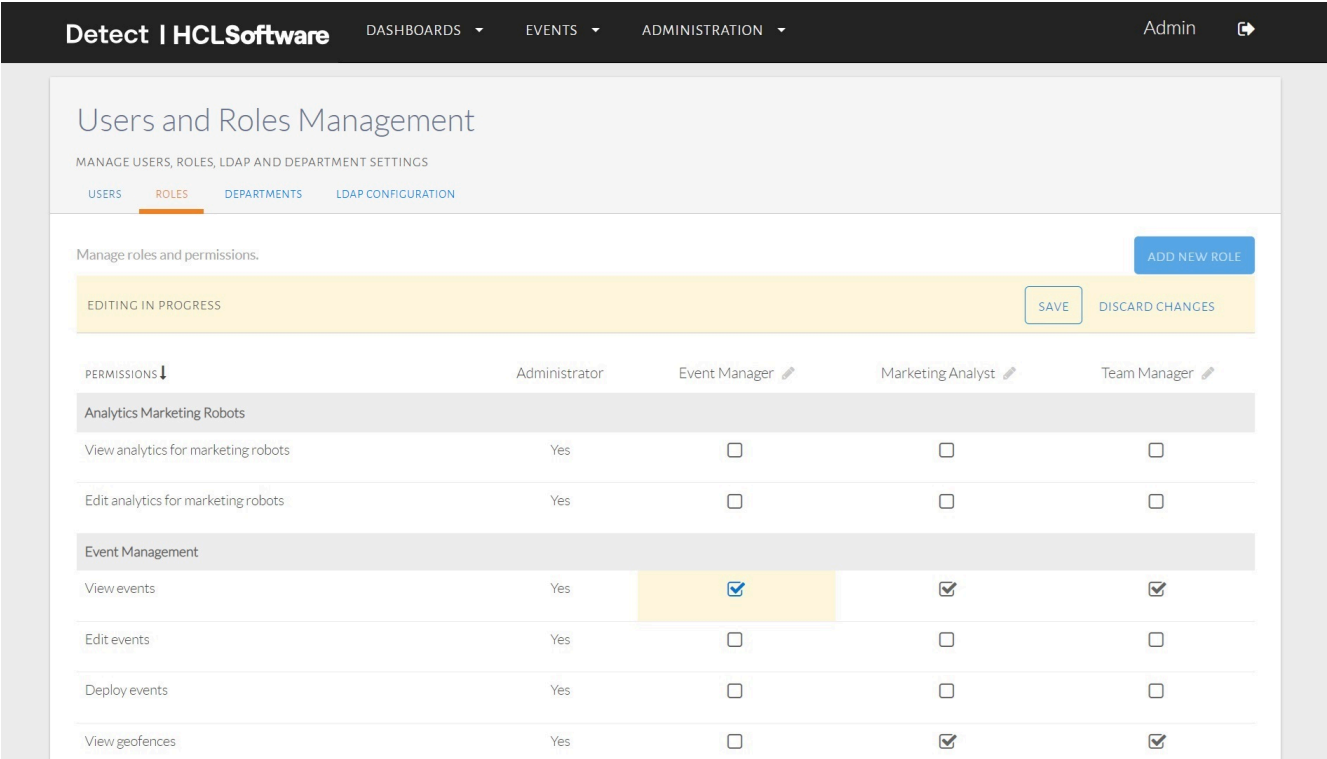


A new role can be created by clicking on the `AddNewRole` button located at the top-right corner.

The resulting pop-up asks the user to enter the name of the new role and, upon clicking on the `Create` button, the corresponding new role is created.



A new role is created without any permissions and assign the roles to department. Click the corresponding check-boxes to add permissions to the newly created role.



A newly created role (Event Manager) with View events permissions only.

## DEPARTMENTS

Departments enables users to organize and group projects under specific departments, streamlining project management and enhancing collaboration within teams.

The screenshot shows the 'Users and Roles Management' interface with the 'DEPARTMENTS' tab selected. The interface includes a header with navigation links (DASHBOARDS, EVENTS, ADMINISTRATION) and a user profile (Admin). The main content area displays a table of departments and an 'ADD NEW DEPARTMENT' button.

**Users and Roles Management**  
MANAGE USERS, ROLES, LDAP AND DEPARTMENT SETTINGS

USERS ROLES **DEPARTMENTS** LDAP CONFIGURATION

Manage your departments effectively from this section.

NAME	DESCRIPTION	ACTIONS
CreditCard	Credit Card offers	<a href="#">EDIT</a> <a href="#">DELETE</a>

[+ ADD NEW DEPARTMENT](#)

HCL Detect © v12.1.9 • © Copyright HCL Technologies Ltd. 2014-2024. (build.id) HCLSoftware

To add a department, click **ADD NEW DEPARTMENT** at the bottom of the page, and enter the department name and description. The newly created department will be listed in the departments page.



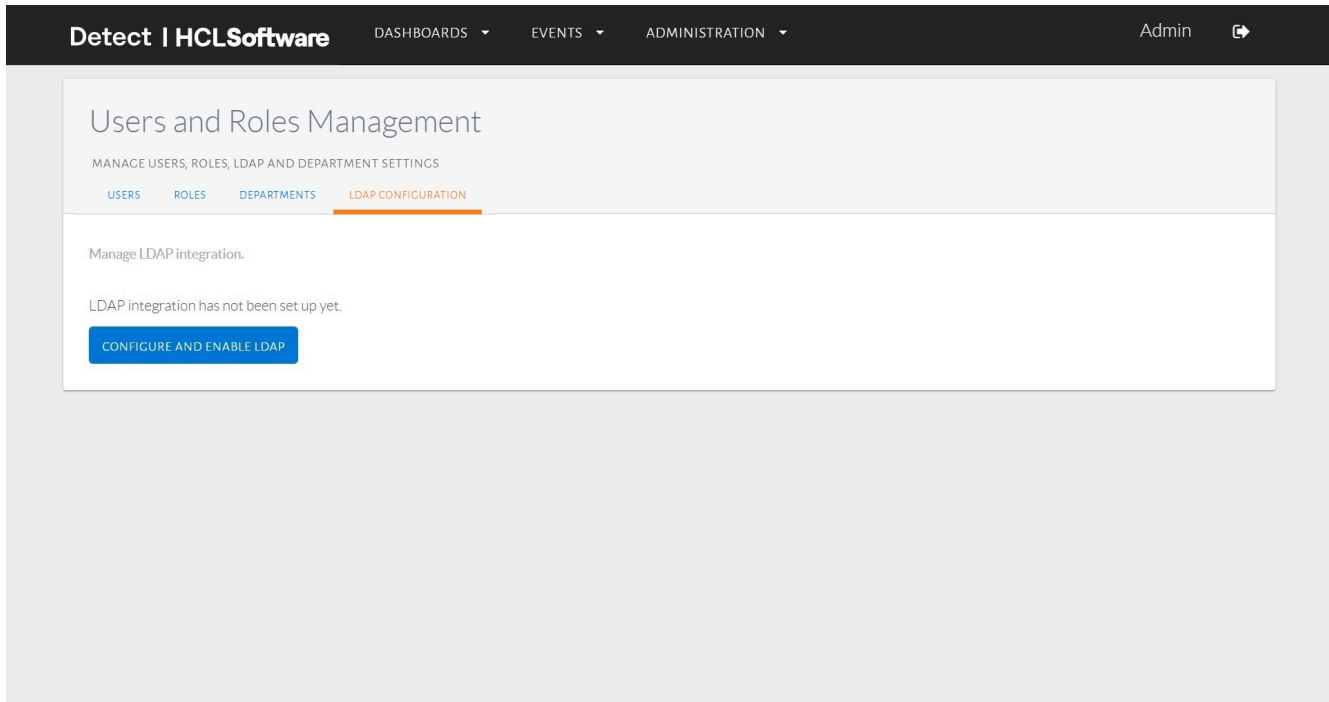
**Note:** **DEPARTMENT NAME** and **DESCRIPTION** fields are mandatory fields.

The screenshot displays the HCL Detect Admin interface. The top navigation bar includes the logo 'Detect | HCLSoftware' and menu items: DASHBOARDS, EVENTS, ADMINISTRATION, and Admin. The main content area is titled 'Users and Roles Management' with the subtitle 'MANAGE USERS, ROLES, LDAP AND DEPARTMENT SETTINGS'. Below this, there are four tabs: USERS, ROLES, DEPARTMENTS (which is highlighted with an orange underline), and LDAP CONFIGURATION. The 'DEPARTMENTS' tab contains the text 'Manage your departments effectively from this section.' followed by two input fields: 'DEPARTMENT NAME\*' and 'DESCRIPTION\*'. Below these fields are two buttons: 'SAVE' and 'CANCEL'. The footer of the interface shows the version 'HCL Detect © v12.1.9 - © Copyright HCL Technologies Ltd. 2014-2024. (build id)' and the HCLSoftware logo.

To update the existing departments, click **Edit** and update the details. The removal of the department is done by clicking **Delete**.

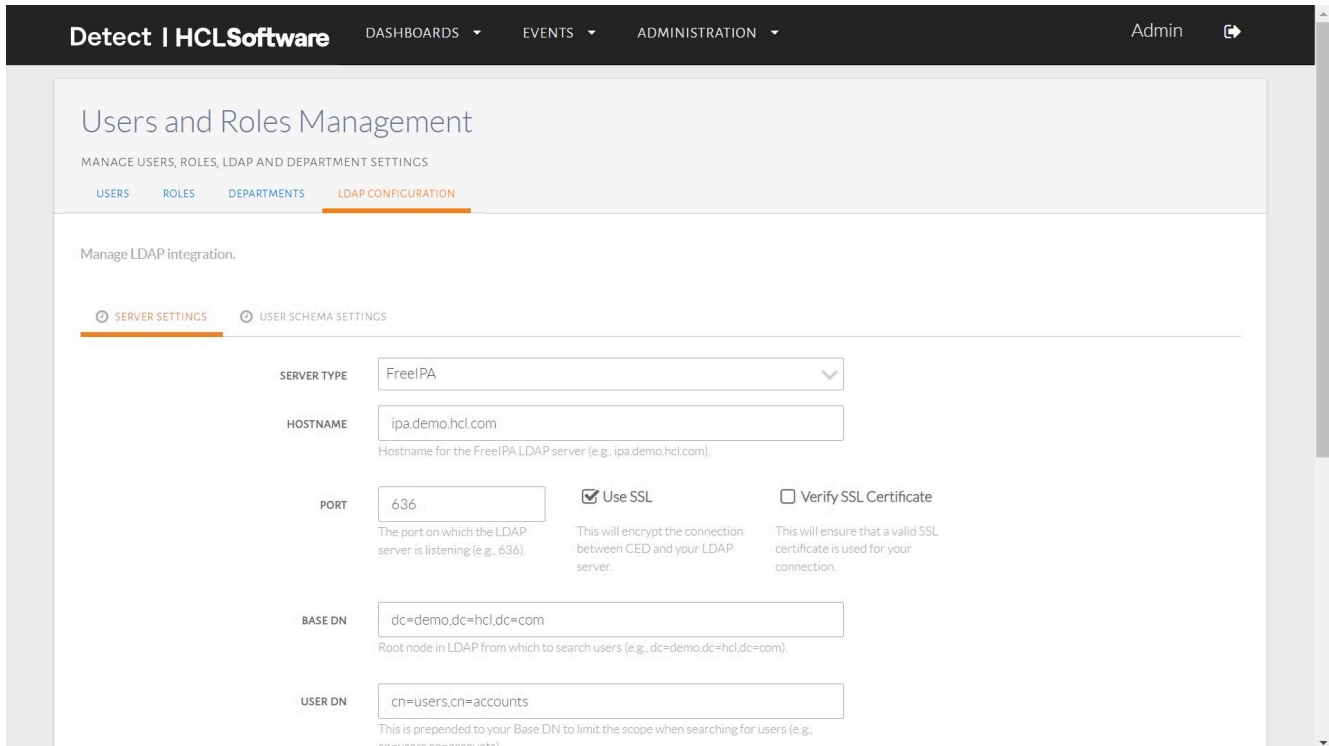
## LDAP

HCL Detect can be optionally configured to do LDAP based authentication. The `LDAPConfiguration` page can be used to configure the `LDAP Configuration` tab in the navigation bar:



The `LDAPConfiguration` tab.

To enable LDAP based authentication, we need to click on `ConfigureAndEnableLDAP` button:



Configuring the `LDAPConfiguration` tab.

Subsequently, the `ServerType`, `Hostname`, `Port`, `Base DN`, `User DN`, `Group DN`, `Search User Name` and `Search User Password` attributes should be filled out. Current two type are servers are supported, i.e., `Free IPA` and `Active Directory`. Finally, the `Test And Save` button can be pressed to test the server setting:

The screenshot displays the 'Detect | HCLSoftware' admin interface. The top navigation bar includes 'DASHBOARDS', 'EVENTS', and 'ADMINISTRATION'. The 'ADMINISTRATION' section is active, showing 'SERVER SETTINGS' and 'USER SCHEMA SETTINGS' tabs. The 'SERVER SETTINGS' tab is selected, showing configuration for an 'ActiveDirectory' server. The form includes fields for 'HOSTNAME' (ipa.demo.hcl.com), 'PORT' (389), 'BASE DN' (dc=unicindia,dc=com), 'USER DN' (cn=Users), 'GROUP DN' (empty), 'SEARCH USER NAME' (unicindia\administrator), and 'SEARCH USER PASSWORD' (masked). There are checkboxes for 'Use SSL' and 'Verify SSL Certificate'. At the bottom, there are buttons for 'DISCARD CHANGES', 'TEST AND SAVE', 'TEST CURRENT SETTINGS', and 'NEXT'. A message at the bottom left states 'No changes to save'.

Testing the `ServerSettings`.

Press `Next` Button to start configuring the `User Schema Settings` tab. Now `user Object Class`, `User Object Filter`, `Username Attribute`, `User Name RDN Attribute`, `User First Name Attribute`, `User Lastname Attribute`, `User Display Name Attribute`, `User Email Attribute` and `User Distinguished Name Attribute` attribute needs to be filled based on your organization's LDAP configurations:

**Detect | HCLSoftware** DASHBOARDS ▾ EVENTS ▾ ADMINISTRATION ▾ Admin ↗

✓ SERVER SETTINGS ✓ **USER SCHEMA SETTINGS**

**USER OBJECT CLASS**   
The LDAP user object class type to use when loading users (e.g., inetorgperson).

**USER OBJECT FILTER**   
The filter to use when searching for user objects (e.g., (objectClass=inetorgperson)).

**USERNAME ATTRIBUTE**   
The object attribute to use when loading the user's sign in username (e.g., uid).

**USER NAME RDN ATTRIBUTE**   
The RDN to use when loading the username (e.g., uid).

**USER FIRST NAME ATTRIBUTE**   
The object attribute to use when loading the user's first name (e.g., givenName).

**USER LAST NAME ATTRIBUTE**   
The object attribute to use when loading the user's last name (e.g., sn).

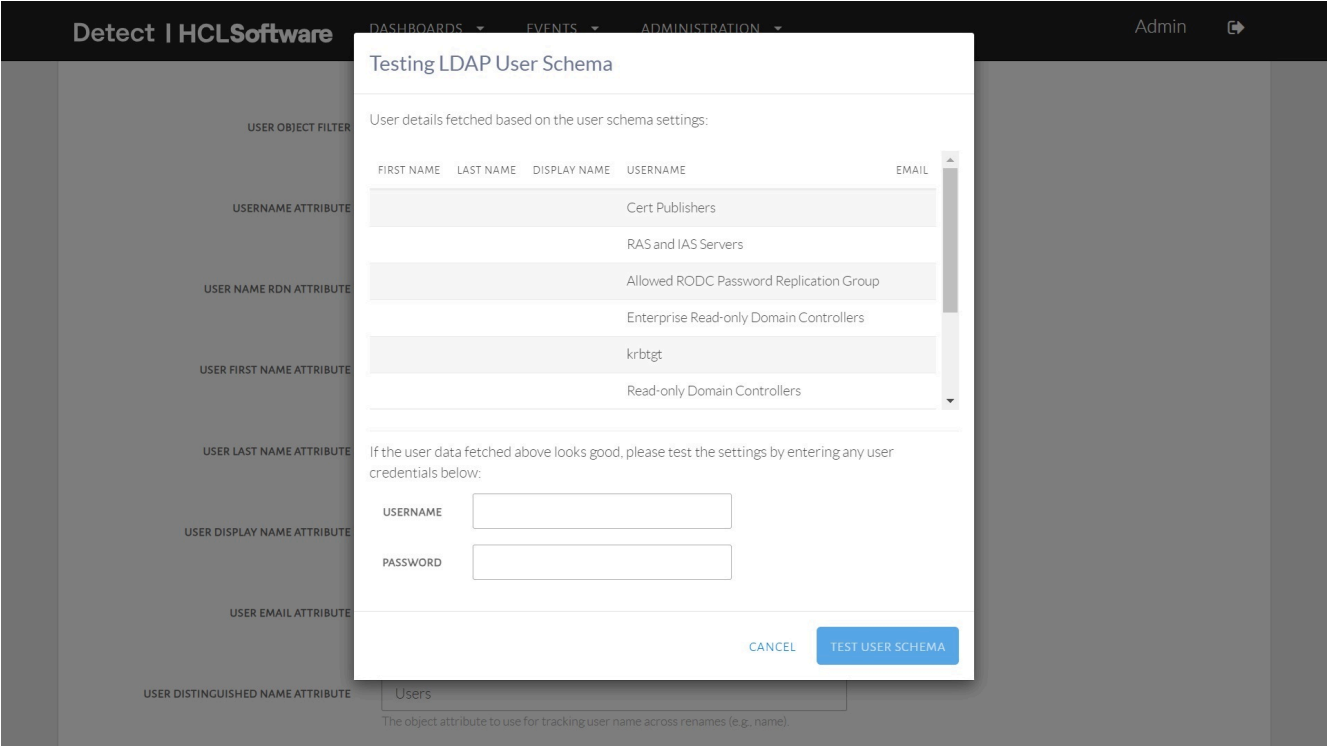
**USER DISPLAY NAME ATTRIBUTE**   
The object attribute to use when loading the user's display name (e.g., displayName).

**USER EMAIL ATTRIBUTE**   
The object attribute to use when loading the user's email (e.g., mail).

**USER DISTINGUISHED NAME ATTRIBUTE**   
The object attribute to use for tracking user name across renames (e.g., name).

Configuring the `UserSchemaSettings`.

Now Click on `TestAndSave` button to test the configuration. Test it by filling `Username` and `Password` for any existing LDAP user and clicking `Test And Save` on the pop-up display.



Testing the `UserSchemaSettings`.

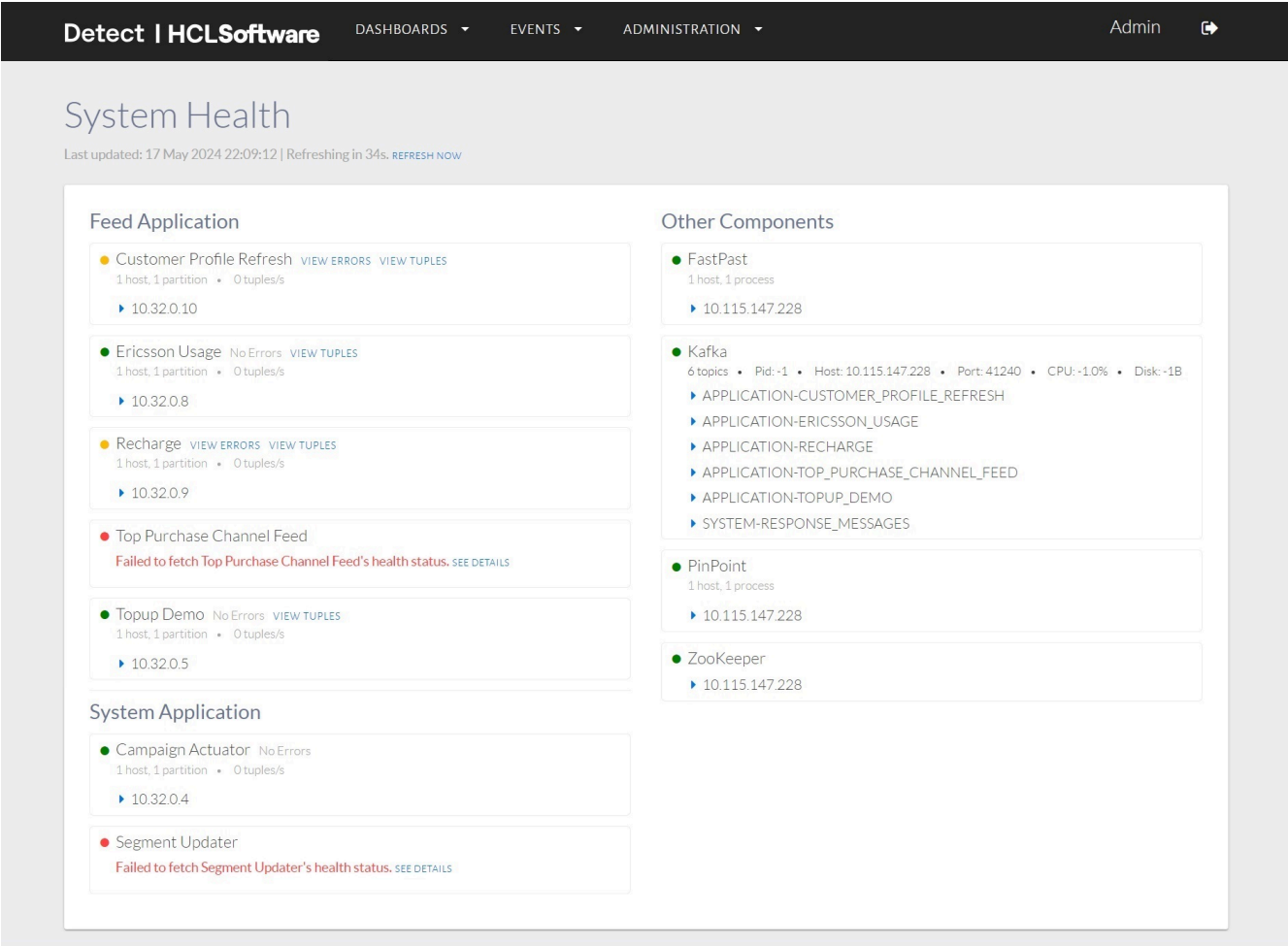


# Chapter 3. System Health

The `SystemHealth` dashboard can be accessed by clicking on the `System Health` link in the top navigation bar.

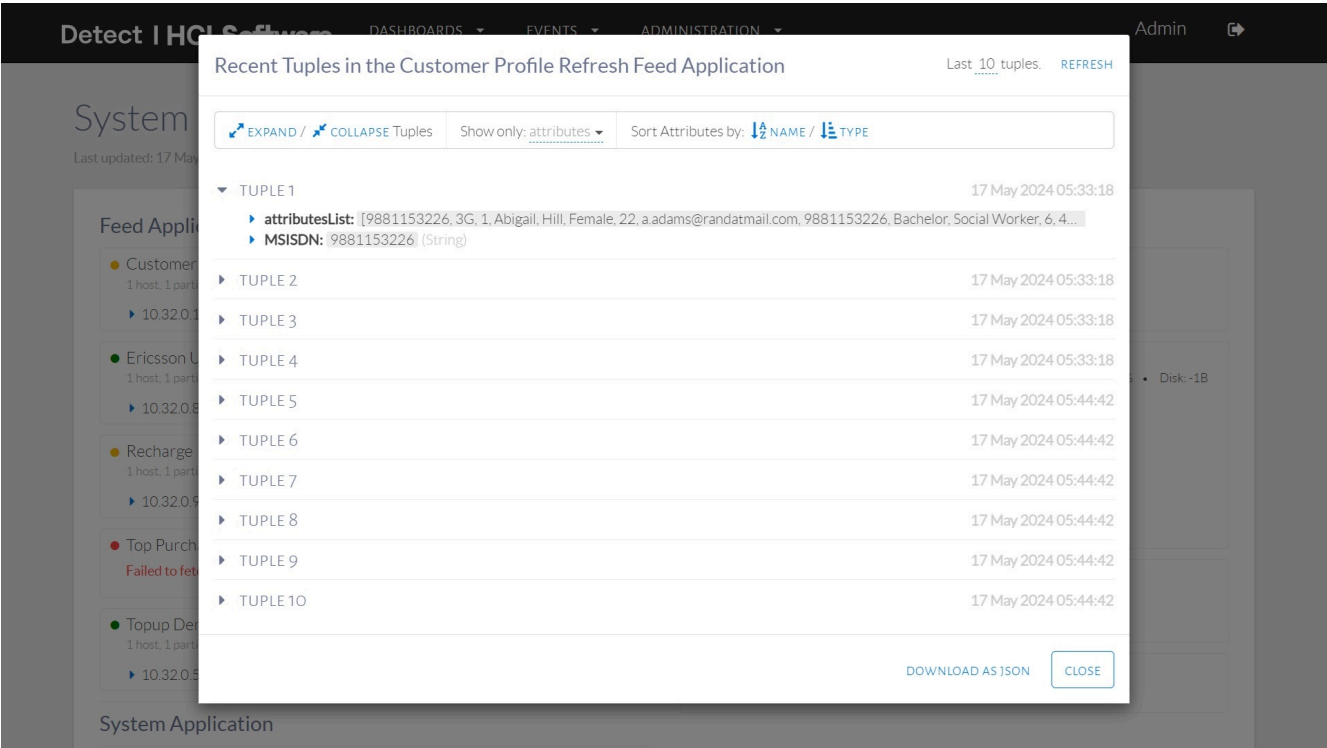
The dashboard shows the status of the `AnalyticsApplications`, `Feed Applications`, `System Applications` and `Other Components`.

`FeedApplications` are used for ingesting the incoming streaming data, `Analytics Applications` are used for stream or Batch based analytics jobs, `System Applications` are built-in applications used for different important functions of application like detecting events or managing batch segment updates. `Other Components` are the HCL Detect supporting components, including the `FastPast`, the `PinPoint`, the `Kafka` and the `Zookeeper`:



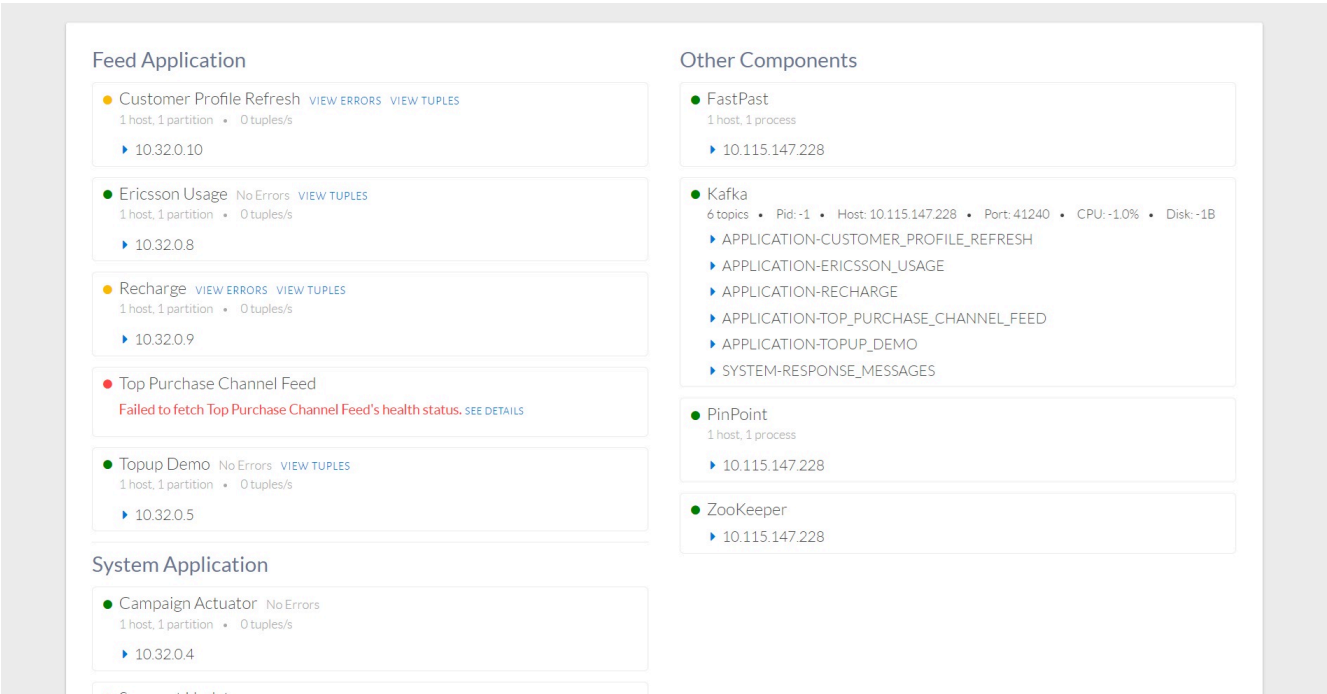
The `SystemHealth` dashboard.

`FeedApplications` can also be used to see the last few data rows/tuples begin processed by clicking `View Tuples` button:



`ViewTuples` from feed applications.

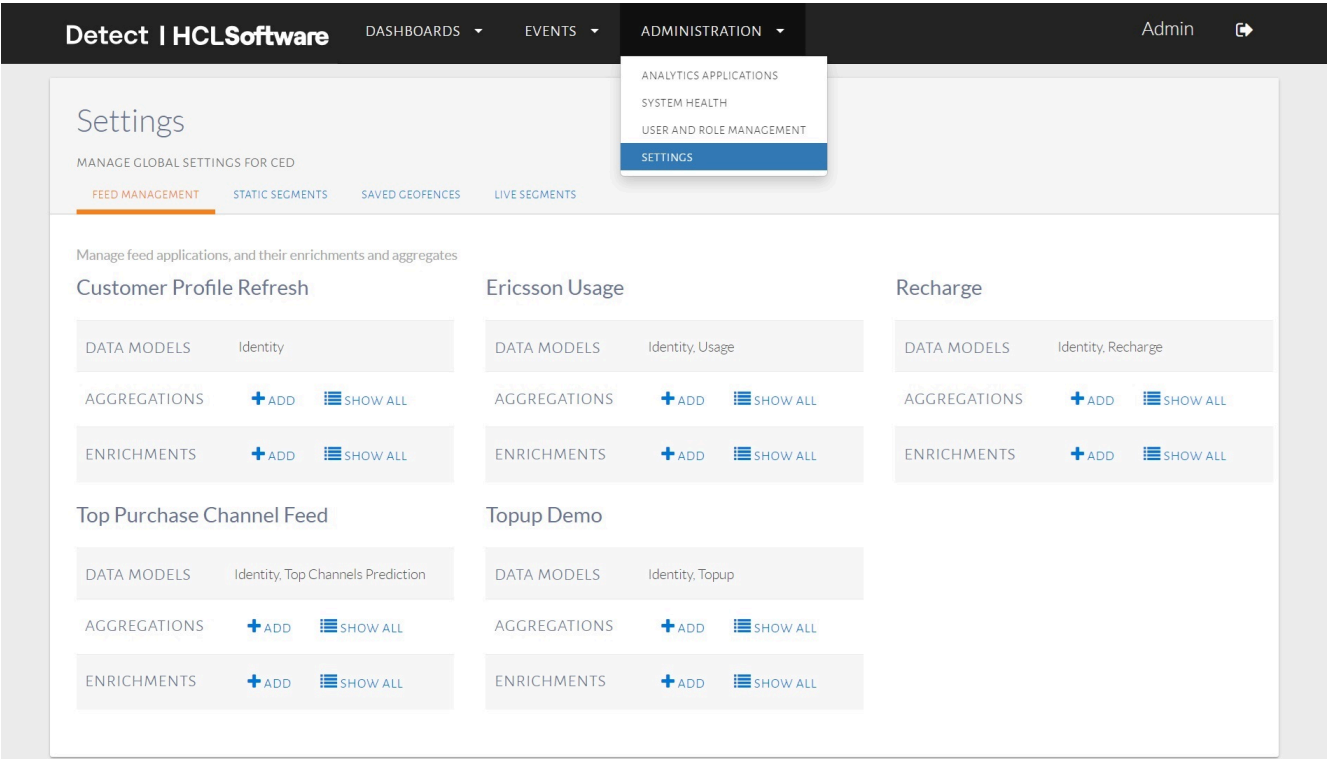
`FeedApplications` also shows the resource utilization of a feeds and parallel processing flow of each feeds.



partitions and statistics of a feed application.

# Chapter 4. Applications, Aggregates & Enrichments

All of the feed applications configured in HCL Detect can be seen by clicking on the `Administration` link in the top navigation bar and then by clicking the `Settings` menu item. We have to use `Feed Management` tab to see feed applications.



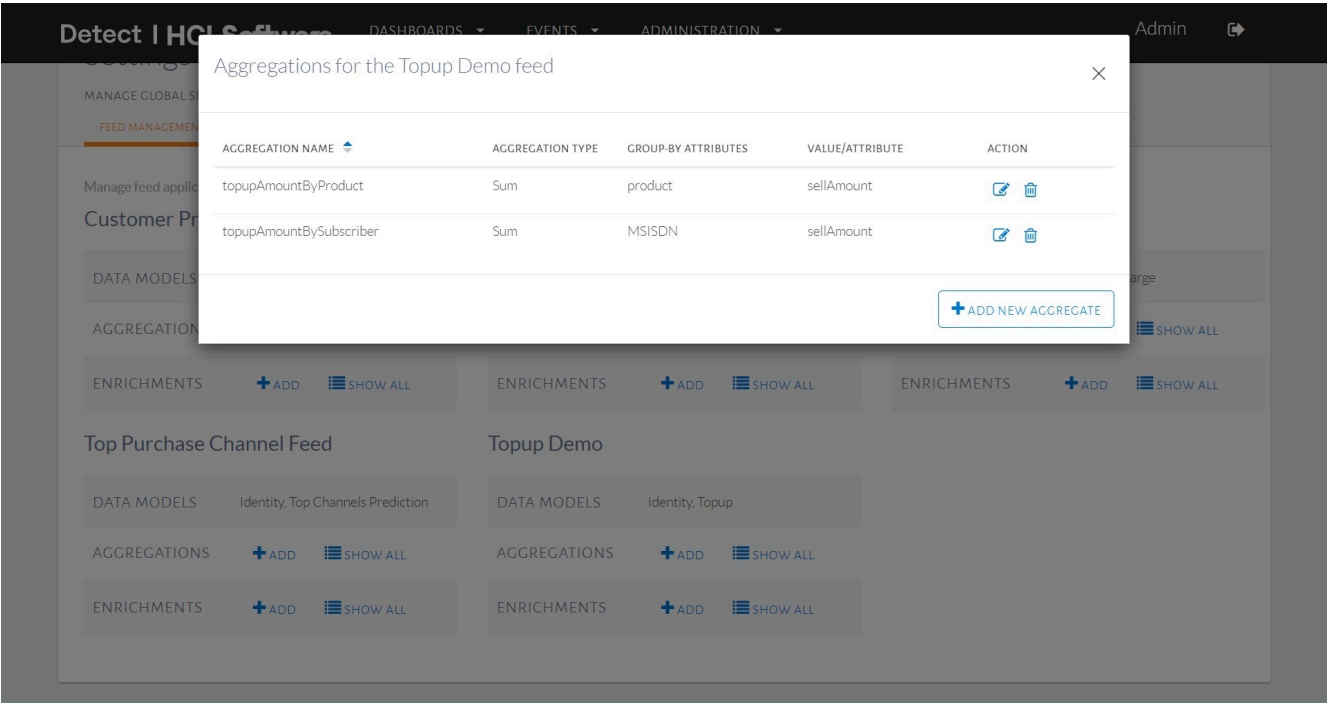
The applications configured in HCL Detect.

This dashboard shows the applications and their associated data model, e.g., 'Topup', 'Usage', etc.

Users with the appropriate permissions can use this page to view, edit, add and/or delete the aggregates and enrichments associated with a given feed application.

The existing aggregates configured for a feed application can be seen by clicking on the `ShowAll` button in the `Aggregations` row for that application.

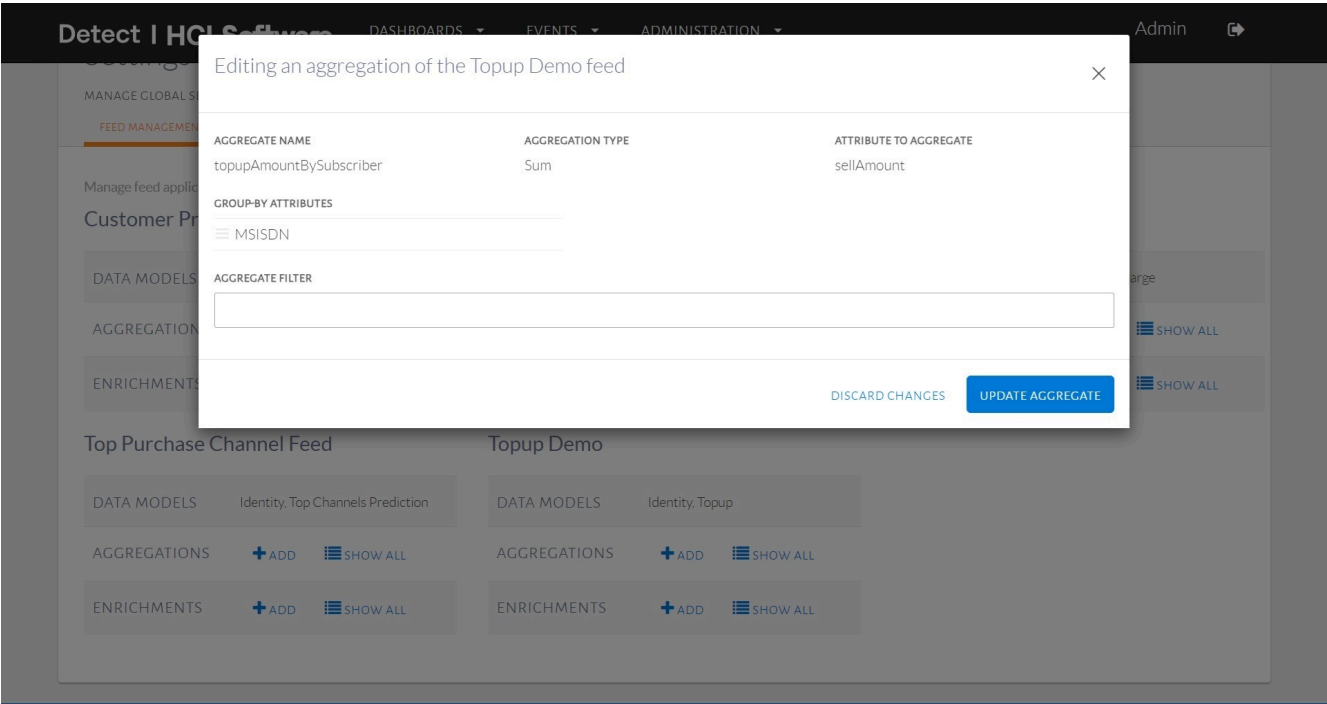
The dialog showing the aggregates associated with the `TopupDemo` application is below:



The aggregates associated with the `TopupDemo` feed.

The aggregates associated with a feed can be edited and removed.

To explain the aggregates a bit further, let us consider the aggregate called `topupAmountBySubscriber`. The dialog that opens up upon clicking on the edit button is shown below:



Editing an aggregate.

This aggregate is designed to count the number of calls by subscriber over various time windows (e.g., `currentDay`, `lastDay`, etc.).

`topupAmountBySubscriber`, i.e. the aggregate `Name`, is the name that is used to refer to the aggregate when using it in a trigger.

The `AggregationType`, in this case `Sum`, is used to accumulate the topup amounts, and the group-by attribute, in this case, `MSISDN`, is used to produce per-user tallies, since it identifies individual subscribers.

Additional group-by attributes can be added in a manner similar to the *group-by* clause used in traditional relational database management system (RDBMS) query languages.

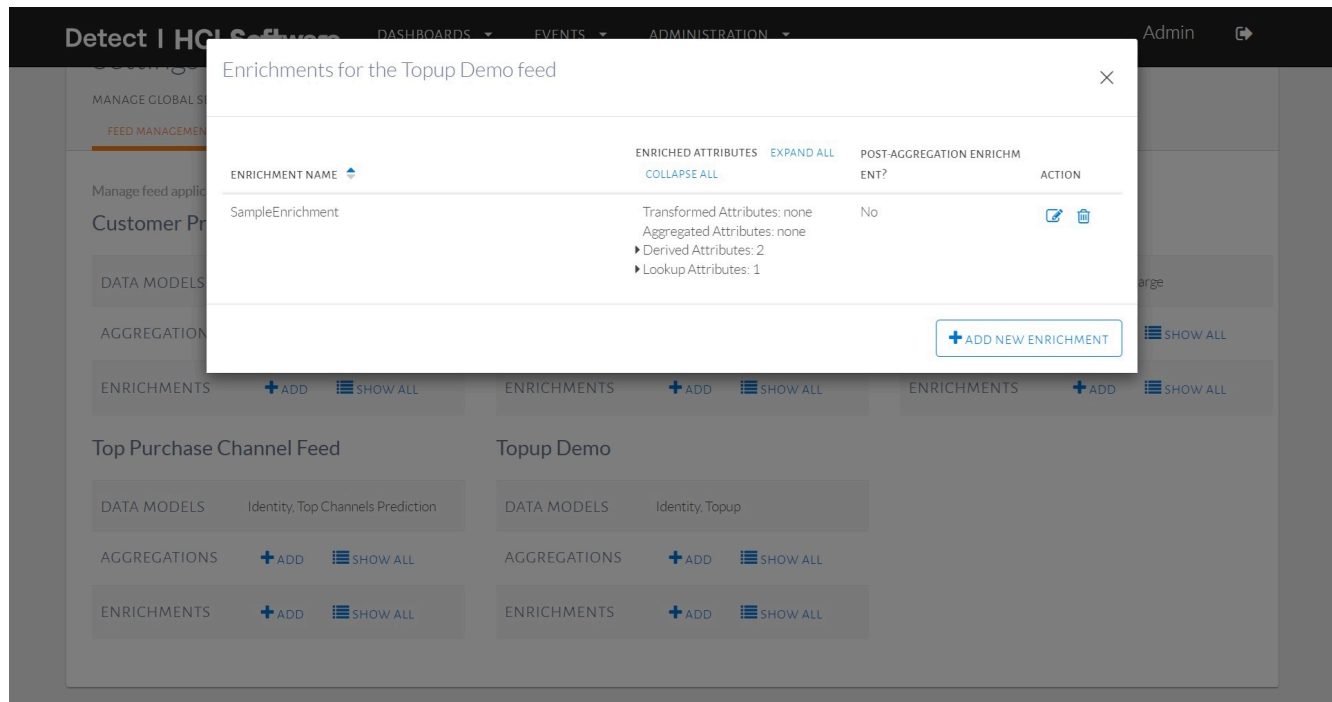
A filter condition can also be used when only a subset of tuples is to be considered for an aggregation.

For instance, if the incoming tuples in the feed contain a Boolean `isDroppedCall`, indicating a dropped call, then one can calculate `droppedCallsBySubscriber` by using `isDroppedCall` equals to `True` as a filter condition with the `MSISDN` as the group-by attribute and `Count` as the aggregation type.

To add a new aggregate, we need to click on `Add` button for that feed application. A dialog with form needs to be filled by filling `Aggregate Name`, `Aggregation Type`, `Attribute to Aggregate`, `Group by Attribute` and `Aggregation Filter`:

Adding new aggregate to a feed.

The list of enrichments associated with a feed can be seen by clicking on the `Show` button in the `Enrichments` row. The enrichments associated with the `Topup Demo` feed can be seen below:



The enrichments associated with the `TopupDemo` feed.

HCL Detect supports 4 kinds of enrichments:

- The *transformed attributes* enrichment enables one to add attributes to the tuple that have either a constant value or a value that can easily be derived from an UEL expression. Such transformed attribute may be needed by other enrichments or retained as an attribute in the enriched tuple. Note that retaining and forwarding the constant attributes as part of the output tuple from the enrichment step is optional.
- The *lookup-based* enrichment, which retrieves attributes from the profile store and possibly add them to the outgoing tuple. Note that retaining and forwarding the looked up attributes as part of the output tuple from the enrichment step is optional. We have to first select a key attribute present in the tuple, the associate `Pinpoint` table will be listed in `Lookup Table` drop-down, finally we have to select the attribute from the lookup table to be looked up.
- The *aggregate-based* enrichment, which fetches data from `FastPast` and possibly adds such attributes to the outgoing tuple. As part of configuring an aggregated attribute, it is necessary to (1) select an aggregate from the list of available aggregates for the feed, (2) select the window length unit, which can be `Minute`, `Hour`, `Day`, `Month` or `Year`, (3) select the period which can be one of `Current` or `Last` and (4) optionally select `Window length`, which can be used to indicate that only a subset of an aggregate is to be used. For instance, when the `Day` window length unit and the `Last` period with `Window Length` set to 7 is chosen, the aggregated value for the last 7 days can be computed. Note that when using the `Minute` window length unit, `Window Length` needs to be multiple of 10-minute i.e., backend configured sub-hour buckets length. Similarly to lookup-based enrichment, aggregate-based enrichment attributes may or may not be retained and forwarded as part of the output tuple.
- The *derived attributes* enrichment, which enables the execution of an external Python function. One such a function takes in a tuple (as well as any other required additional parameters, if any), performs a user-defined computation, and produces a result. This function invocation's return value can then be retained and forwarded as part of an

outgoing tuple. This capability can be used, for instance, to invoke a scoring function on an incoming tuple, which computes a score (e.g., based on previously mined data) that can then be added as an attribute to the corresponding outgoing tuple. The *derived attributes* enrichment are basically of three types, i.e., *enrichment function based*, *expression based* and *scorer function based*. *expression based* takes an UEL expression explained in Miscellaneous » UEL section.

The interface for editing enrichments is shown below:

Editing an enrichment of the Topup Demo feed

ENRICHMENT NAME: SampleEnrichment POST-AGGREGATION ENRICHMENT: No

TRANSFORMED ATTRIBUTES (0) LOOKUP ATTRIBUTES (1) AGGREGATED ATTRIBUTES (0) DERIVED ATTRIBUTES (2)

ATTRIBUTE NAME	LOOKUP TABLE	KEY ATTRIBUTE	LOOKED UP ATTRIBUTE	TYPE	RETAINED?	
lastMSISDN	CUSTOMER_PROFILE	MSISDN	lastMSISDN	String	<input type="checkbox"/>	

ADD LOOKUP ATTRIBUTE

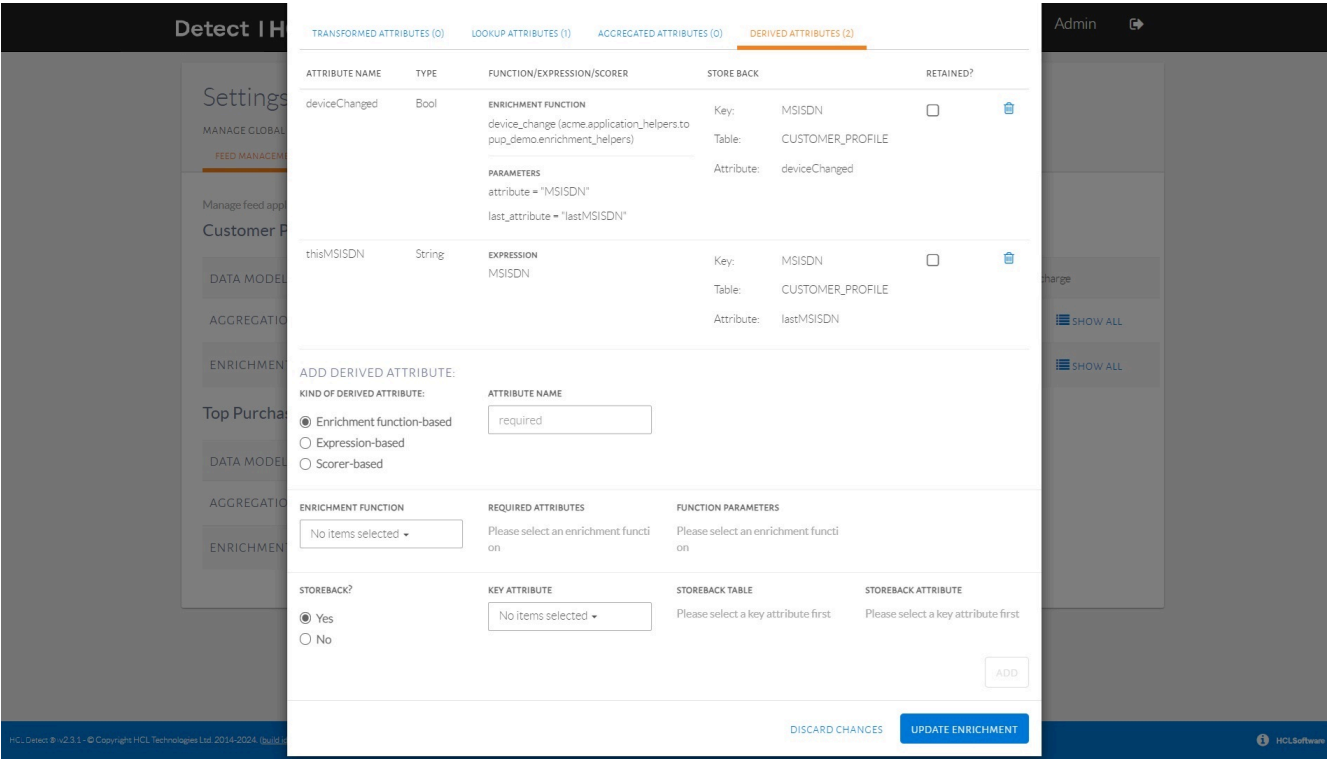
KEY ATTRIBUTE:

LOOKUP TABLE:

SELECT LOOKUP ATTRIBUTES:

Interface for editing enrichments.

The interface for editing derived attribute type enrichments is shown below:



Interface for editing derived attribute type enrichments.

Enrichment operations can be configured to before aggregations are performed or after aggregations are performed. The value of a enriched *aggregate-based* attribute may change based on whether it configured to run before or after the aggregation operation. We can choose this while creating an enrichment by checking the `Post-Aggregation Enrichment` check-bit as shown in the below image:



The screenshot displays the 'Detect | HCL Software' interface with a modal window titled 'Editing an enrichment of the Topup Demo feed'. The modal contains the following elements:

- ENRICHMENT NAME:** SampleEnrichment
- POST-AGGREGATION ENRICHMENT:** No
- Attribute Tabs:** TRANSFORMED ATTRIBUTES (0) (selected), LOOKUP ATTRIBUTES (1), AGGREGATED ATTRIBUTES (0), DERIVED ATTRIBUTES (2)
- Content:** No transformed attributes
- ADD TRANSFORMED ATTRIBUTE Section:**
  - ATTRIBUTE NAME:** required
  - ATTRIBUTE DATA TYPE:** No items selected (dropdown)
  - EXPRESSION:** (empty text field)
  - ADD:** (button)
- Buttons:** DISCARD CHANGES, UPDATE ENRICHMENT

The background interface shows a sidebar with 'Settings' and 'FEED MANAGEMENT' sections, and a main area with 'Top Purchase Channel Feed' and 'Topup Demo' sections, each containing 'DATA MODELS', 'AGGREGATIONS', and 'ENRICHMENTS' sub-sections.

Interface for creating a new enrichment.

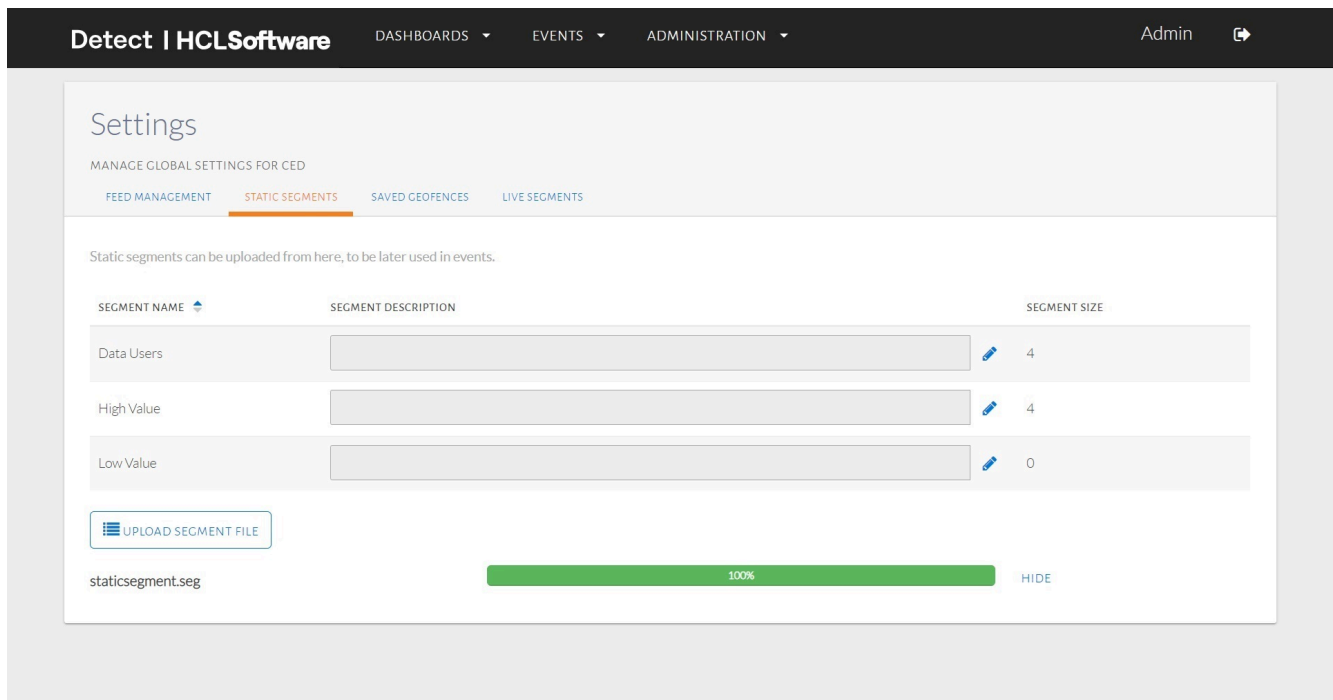
# Chapter 5. Subscriber Segments

Segments are groups of customers with some similarities, setup segments based on preset conditions (Live), or through user selection list (Static). Grouping customers into Segments helps target them more appropriately or continuously across events with Detect. Detect supports 2 types of segments, Live & Static HCL Detect can make use of *subscriber segments*. Such segments can be used to include or exclude a group of subscribers from being considered as subjects for a trigger.

## Static Segments

The set of `StaticSegments` can be inspected by clicking on `Settings` from the `Administration` menu and then selecting the tab titled `Static Segments`.

The example below shows a view of the page with 2 segments: the `DataUsers` and the `High Value` users.



Uploading a segment file.

To add, edit or delete a segment, a text file with its contents specified in a particular format can be uploaded to HCL Detect using the `UploadSegmentFile` button.

Two formats are supported. The first is a plain-text file with a simple linear organization as shown below:

```
[Data Users]: add
911323232323
918787879988
918787989021
919898990906
[High Value]: add
912323989899
912367367676
```

```

912398982337
919828738787
[High Value]: remove
912388728787
[Low Value]: remove_all

```

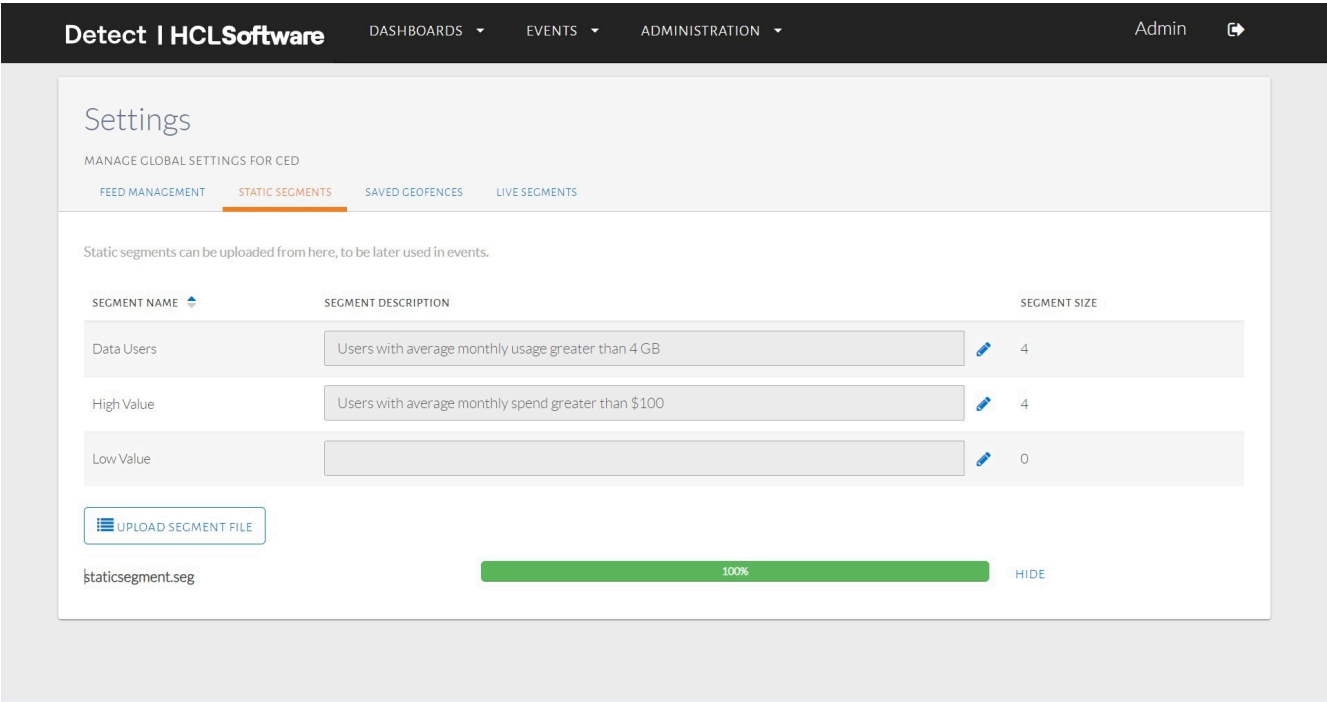
Alternatively, the file can also be specified in JSON format, in which case it must have a `.json` extension. An example is given below (note that the order of the fields is important):

```

{
  "segmentUploads": [
    {
      "operation": "Add",
      "segment": "Data Users",
      "users": [
        "911323232323",
        "918787879988",
        "918787989021",
        "919898990906"
      ]
    },
    {
      "operation": "Add",
      "segment": "High Value",
      "users": [
        "912323989899",
        "912367367676",
        "912398982337",
        "919828738787"
      ]
    },
    {
      "operation": "Remove",
      "segment": "High Value",
      "users": [
        "912388728787"
      ]
    },
    {
      "operation": "RemoveAll",
      "segment": "Low Value"
    }
  ]
}

```

Once uploaded, one can also add a description for the segments:



Segments and their descriptions.

**Live Segments**

Detect allows the user to enter a set of audience conditions and saves customers who meet those criteria in a Live Segment. Detect checks all customers against the defined conditions and maintains the segment over time. This means that Detect adds or removes customers from the segment as they fulfill or stop fulfilling the defined criteria. For example: if you'd like to frequently target customers who have the main balance atleast \$100 & who have installed a Mobile App i.e., Facebook, this can be saved as a Live Segment.

Detect will check all customers against the defined Main balance check & app installation status and sort the appropriate customers to the segment. Conditions can be setup as shown below:

**Detect | HCLSoftware** DASHBOARDS ▾ EVENTS ▾ ADMINISTRATION ▾ Admin ↗

**Demo Event** [edit](#) PROJECT: Demo STATUS: **DRAFT**

Demo Event

CREATED BY: admin

Audience ✓ Event Triggers ✓ Events ✓ Details ✓ [REVIEW AND SUBMIT FOR APPROVAL](#)

### Audience Criteria

All of the following: ▾

- Main balance amount ▾  
Subscribers who have the main balance at least ▾ \$ 100.00..
- Installed mobile applications ▾  
Subscribers who have installed MobileApp1 ▾ mobile application(s)

ADD ANOTHER CONDITION SAVE AS A LIVE SEGMENT

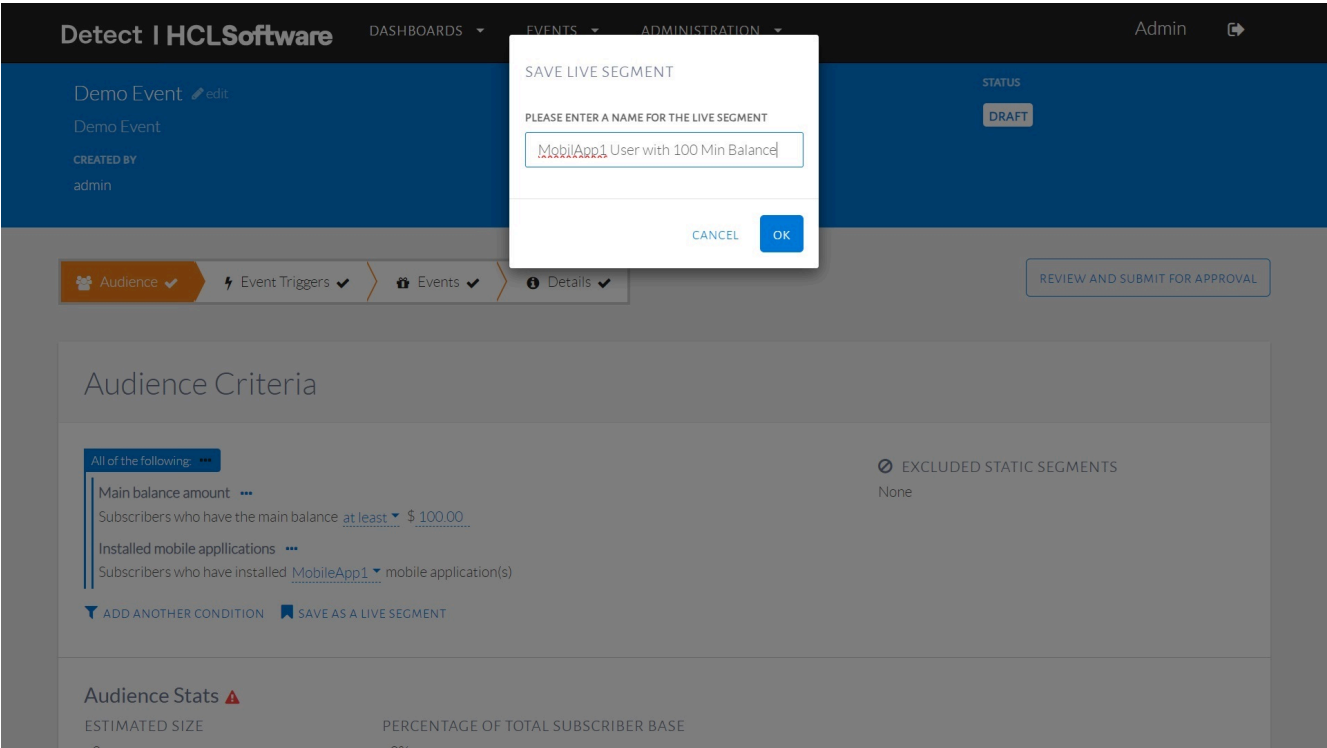
EXCLUDED STATIC SEGMENTS  
None

**Audience Stats** ⚠

ESTIMATED SIZE PERCENTAGE OF TOTAL SUBSCRIBER BASE

Creating an audience condition.

To save above audience condition as reusable live segment, we need to click on `SaveasLiveSegment` button. A dialog will open to get the name of the live segment:



Saving a live segment.

Once the live segment is saved, the condition will be wrapped as a live segments. This live segment can be replaced and edited by clicking three dots on the live segment shown:

**Detect | HCLSoftware** DASHBOARDS ▾ EVENTS ▾ ADMINISTRATION ▾ Admin

Demo Event PROJECT Demo STATUS **DRAFT**

CREATED BY admin

**Audience** ✓ ⚡ Event Triggers ✓ 📅 Events ✓ ⓘ Details ✓ [REVIEW AND SUBMIT FOR APPROVAL](#)

### Audience Criteria

All of the following: \*\*\*

- Live segment \*\*\*
- MobilApp1 User with 100 Min Balance

ADD ANOTHER CONDITION SAVE AS

SHOW CONDITIONS  
REPLACE WITH CONDITIONS  
REMOVE CONDITION

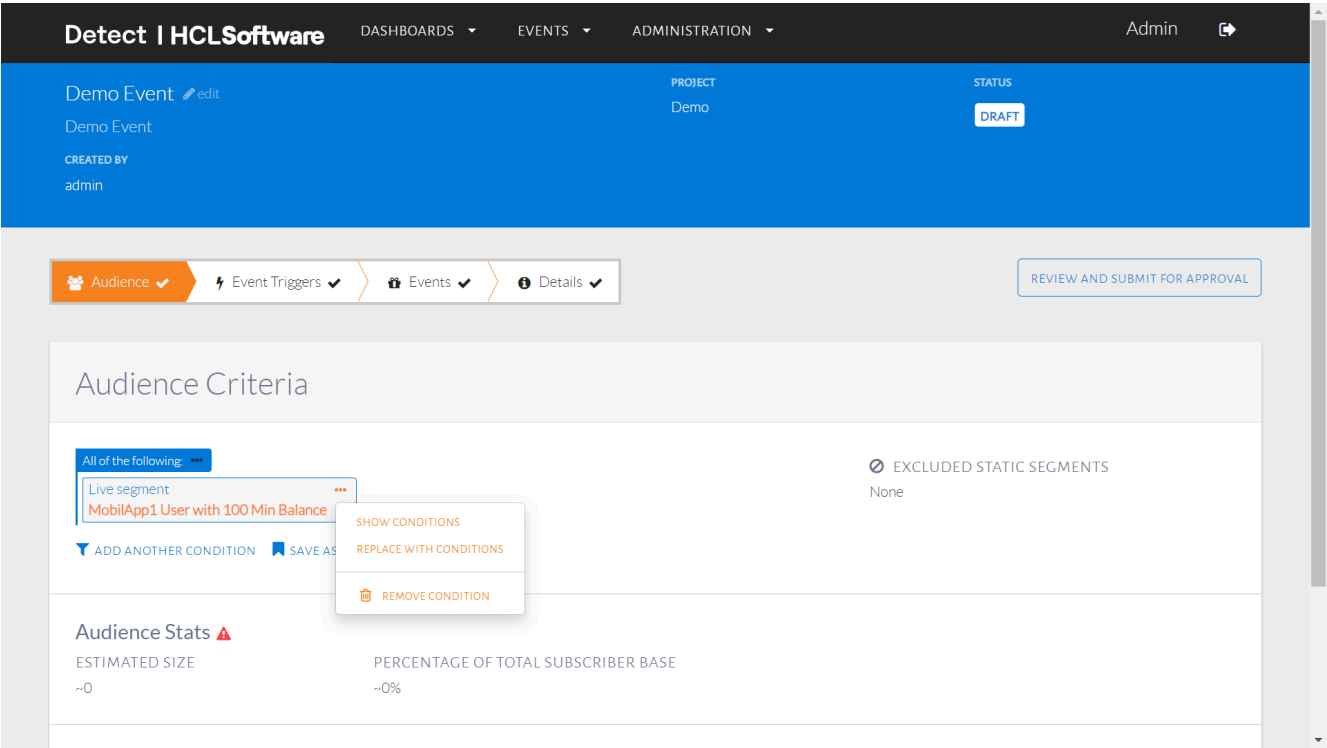
**Audience Stats** ⚠

ESTIMATED SIZE	PERCENTAGE OF TOTAL SUBSCRIBER BASE
~0	~0%

EXCLUDED STATIC SEGMENTS: None

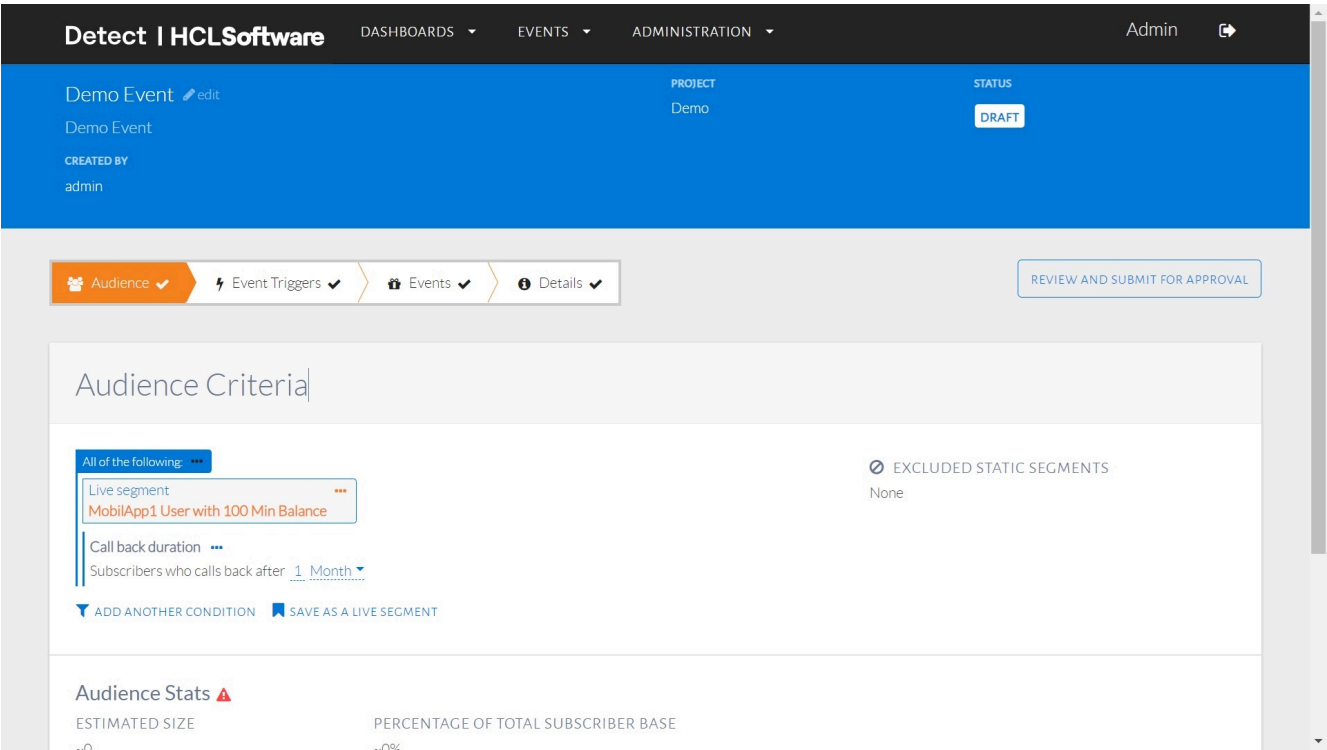
Using a live segment and editing it.

The set of `LiveSegments` can be inspected by clicking on `Settings` from the `Administration` menu and then selecting the tab titled `Live Segments`.



List of live segment.

Customers belonging to Live or Static Segments can be used as audience criteria by any user, in any event. Segments can be used along with other audience conditions as well, and can be added as shown below:





Audience condition using live segment and other audience conditions.