

## Table of Contents

1	Overview .....	2
2	Functionality & Scope .....	2
3	Installing the Ticket Synchronization Framework.....	3
4	Administrator Workspaces .....	6
4.1	Constants .....	7
4.2	Ticket Sharing & Synchronization .....	9
4.2.1	View.....	9
4.2.2	Edit .....	10
4.2.3	Delete .....	<b>Error! Bookmark not defined.</b>
5	Connector Service Transaction .....	16
6	Logging .....	31
7	IMPORTANT .....	34
8	Appendix I – Other Details .....	35
8.1	Scheduled Job .....	35

## 1 Overview

The Ticket Synchronization Framework enables the Ivanti Administrator to integrate the various defect trackers viz. VSTFS, JIRA, Mantis, etc. into the Ivanti SM.

Through the integration, incidents created in Ivanti SM are synchronized as issues in the target defect trackers. The attachments and comments of the incident can also be synchronized.

This user manual illustrates the functionality and usage of the Ticket Synchronization Framework and how the administrator can configure the defect trackers.

## 2 Functionality & Scope

The Ticket Synchronization Framework comprises the following functionality and scope.

Defect Trackers supported currently.

1. JIRA
2. GitHub
3. Bugzilla
4. Redmine
5. Mantis
6. VSTS
7. Salesforce
8. HP Quality Center

Functionality:

1. Creating an Incident in Ivanti SM creates an issue in defect tracker
2. Creating Comments in Incident creates comments in defect tracker
3. Creating Attachments in Incident creates attachments in defect tracker
4. Changing the status of the bug in defect tracker to closed updates the status of the incident in Ivanti SM

Connector	Authentication	Defect Sync Support	Comment Support	Incoming Attachment	Outgoing Attachment
JIRA	Basic	Yes	Yes	Yes	Yes
GitHub	Basic	Yes	Yes	No	No
Bugzilla 5.0.3+	Query Params	Yes	Yes	No	No
Redmine 3.3+	Basic	Yes	Yes	Yes	Yes
Mantis 2.1.0+	SOAP	Yes	Yes	Yes	Yes
VSTS / TFS 2017+	Basic	Yes	Yes	Yes	Yes
SalesForce	OAuth	Yes	Yes	No	No
HP Quality Center	Cookies	No	No	Yes	Yes

Table 1

**Important:** The Synchronization between Ivanti SM and Defect Trackers is ONLY Uni-directional and NOT Bi-Directional always.

### 3 Installing the Ticket Synchronization Framework

The Ticket Synchronization Framework comprises two packages to be imported sequentially.

Package 1 – TicketSharingandSynchronization.MetadataPackage

Package 2 - TicketSharingandSynchronizationSampleData.MetadataPackage

1. Login into the Ivanti Service Manager management system Configuration Settings as Administrator
2. From the left side Settings pane, Navigate to Build -> Ivanti Development Package -> Package
3. Click on the Import Package link on the top right
4. A pop up would be displayed as in Fig.1

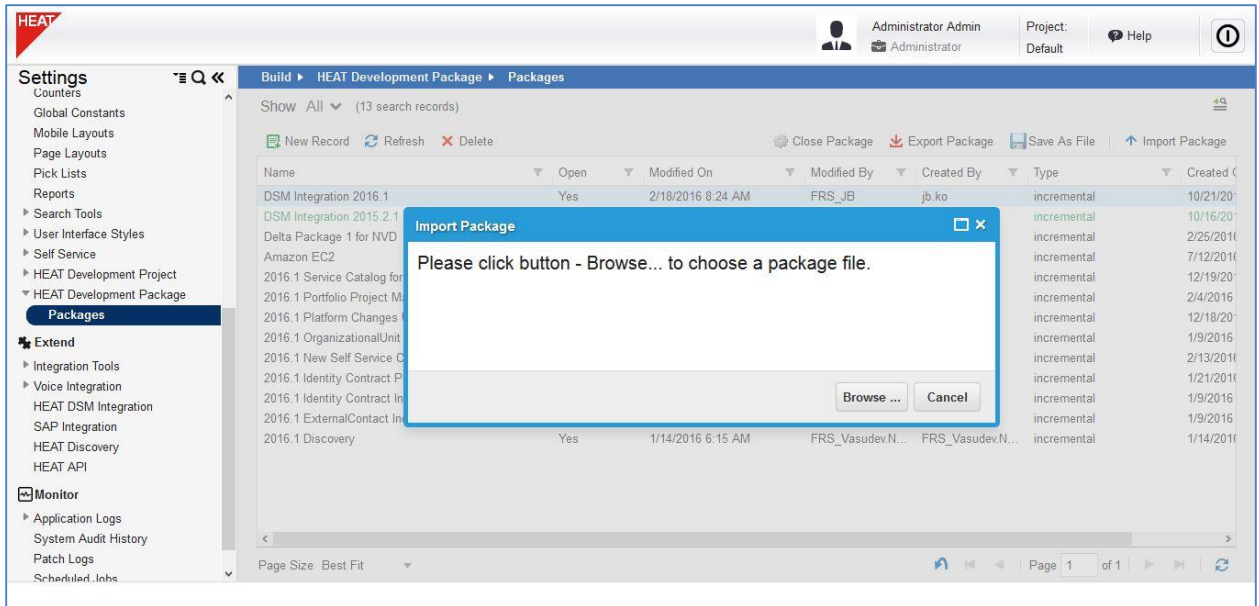


Fig.1

5. Click Browse
6. Navigate to the location where the package 1 file is available as shown in Fig.2

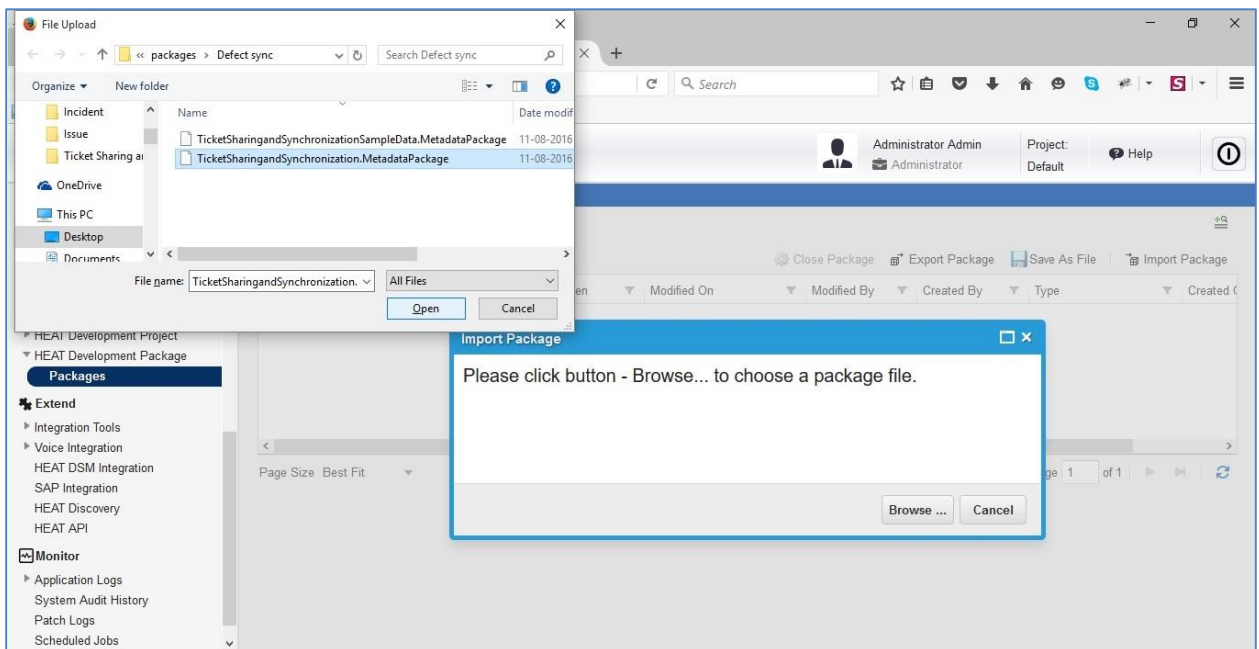


Fig.2

7. Select the Package 1 ***TicketSharingandSynchronization.MetadataPackage***
8. Click Open
9. A Pop up is displayed as shown in Fig.3

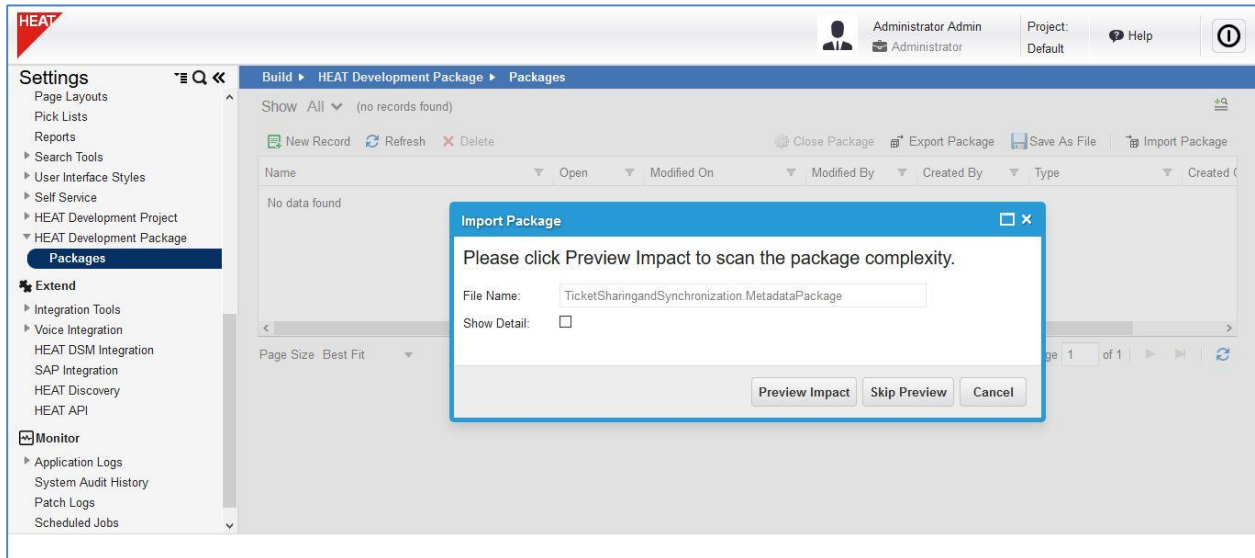


Fig.3

10. Click Skip Preview
11. The pop up with the three Operation Types are displayed as shown in Fig.4

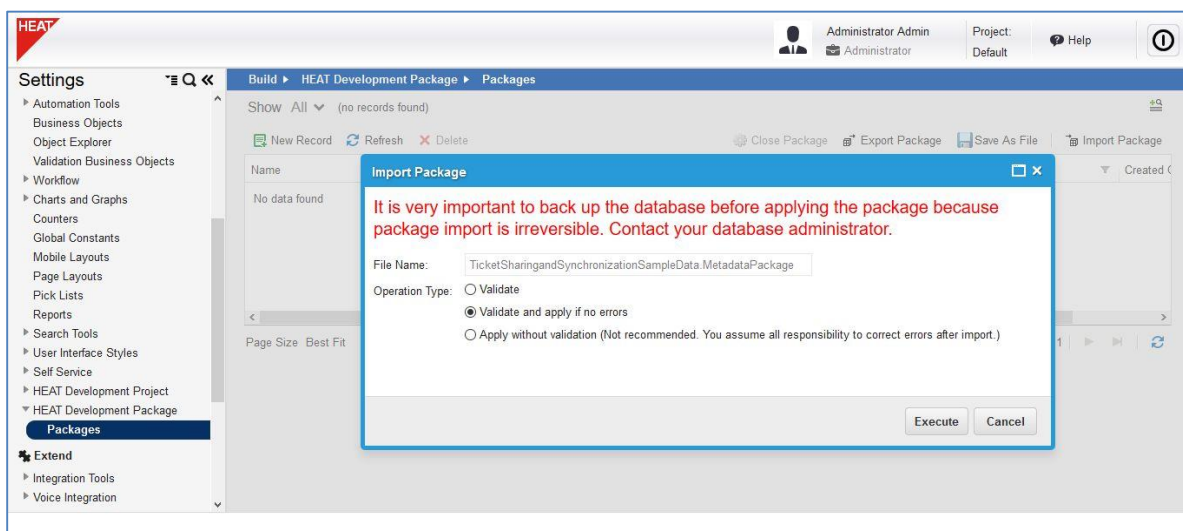


Fig. 4

12. Select the second Operation Type “Validate and Apply if no errors”
13. Click Execute
14. The package would be imported and applied successfully and a message displayed
15. Repeat the steps from 7 to 13 for Package 2

Note:

1. The package 1 - ***TicketSharingandSynchronization.MetadataPackage*** after import comprises the web scripts / quick actions for all connectors and sample data for JIRA and VSTFS connector services ONLY
2. If the customer wants the sample data for other supporting connector services, then import the package 2 – ***TicketSharingandSynchronizationSampleData.MetadataPackage***

## 4 Administrator Workspaces

1. Login as Administrator
2. Click More
3. Check for the Defect Sync specific workspaces as shown in Fig.7
  - a. Constants
  - b. Connector Service
  - c. Connector Service Transaction
  - d. Ticket Sharing and Synchronization

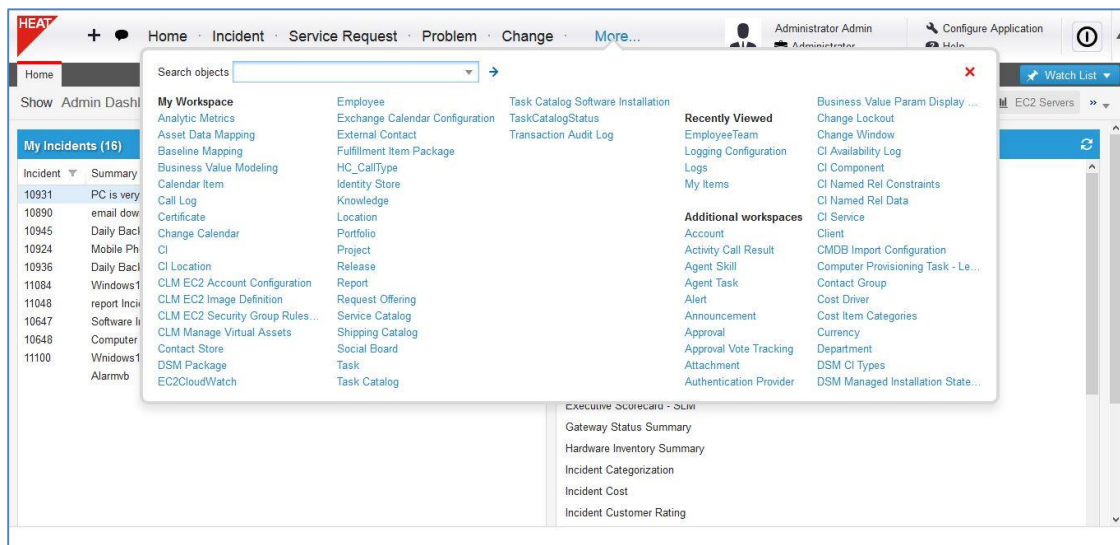


Fig. 7

## 4.1 Constants

The connector web service scripts will interact directly with the Ivanti Service Manager data using the web service API. To do so, the scripts need access to the API, so login credentials and location are required, and can be supplied through this dialog.

As the integration was created under the HEAT Software corporation, the values use HEAT instead of Ivanti.

1. Login as Administrator
  - e. Click More and search for the Constants workspace
  - f. The Constants workspace would be displayed with pre-configured values (Refer Fig.8)

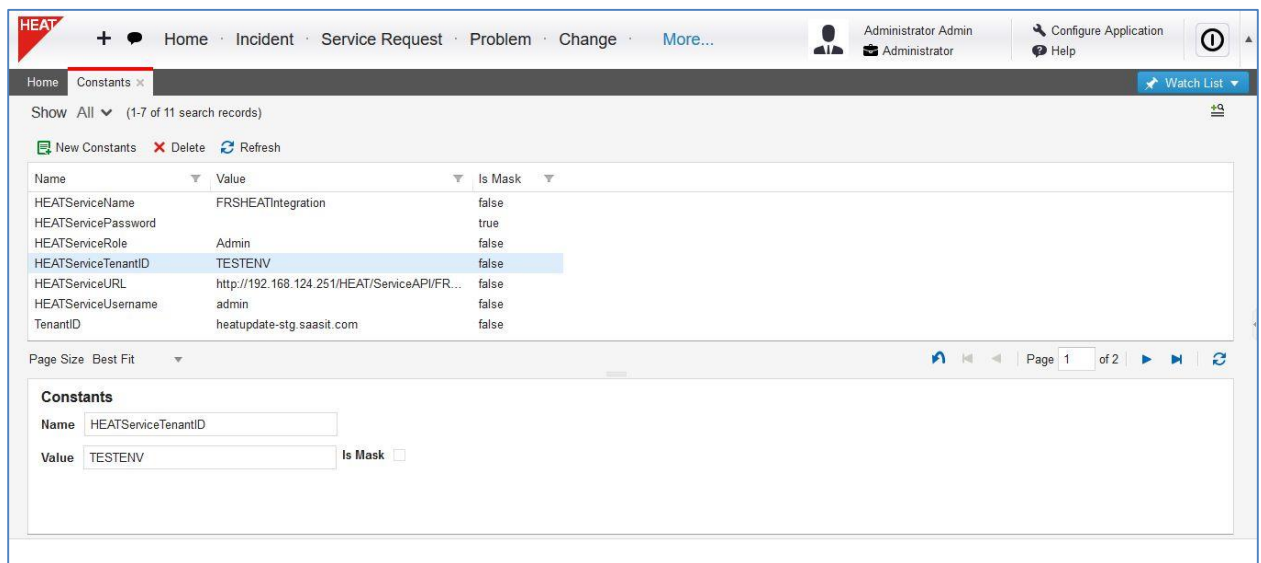


Fig. 8

To edit a particular constant value, double click the constant record.

The constant edit view would be displayed as shown in Fig.9

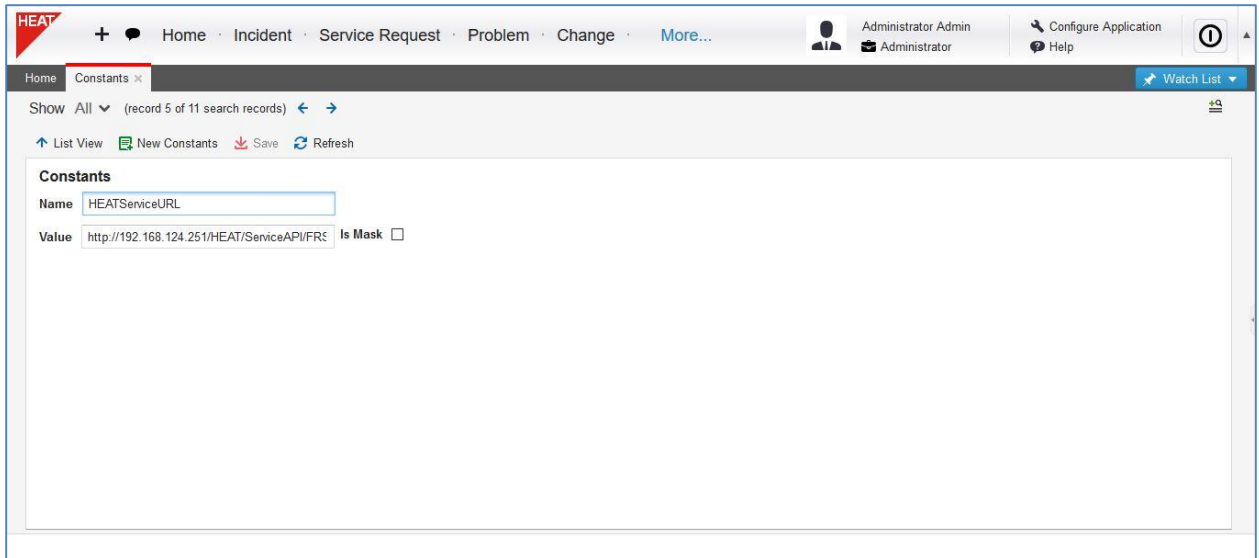


Fig.9

Edit the values in the Constant record to match your tenant.

Check the Is Mask option if you want the Value field data to be masked.

Click Save

The new values would be updated.

Each of the Constants fields is explained below.

**1. HEATServiceName:**

- Name of the Service. This is fixed and should normally not need to be changed.

**2. HEATServiceUsername:**

- Username for accessing the HEAT Web Service API.

**3. HEATServicePassword:**

- Password for accessing the HEAT Web Service API

**4. HEATServiceRole:**

- Role for the accessing the HEAT Web Service API



**5. HEATServiceTenantID:**

- Tenant ID that the API will use for the Web Service API. This should be your current tenant name.

**6. HEATServiceURL:**

- URL of the HEAT Web Service API.
  - If you are a cloud tenant, this should be in the format  
`https://yourtenant.saasit.com/ServiceAPI/FRSHEATIntegration.asmx/`
  - If you are a premise tenant, this should be in the format  
`https://yourtenant.saasit.com/HEAT/ServiceAPI/FRSHEATIntegration.asmx/`

**7. TenantID:**

- Tenant URL, same as HEATServiceURL

**8. TenantPassword:**

- Password of the Tenant, same as HEATServicePassword

**9. TenantRole:**

- Role of the Tenant, same as HEATServiceRole

**Note:**

1. Constants 1 to 6 are used for Attachment Sync
2. Constants 7 to 9 are used for Issue / Comment Sync

## 4.2 Ticket Sharing & Synchronization

This workspace must be configured by the administrator to define the credentials to access the Defect Management system, and to define the field and data mappings.

### 4.2.1 View

The list of all the integrations is displayed as shown in Fig.10

The list displays the following details

1. Name (Integration Name)

2. Business Object Name
3. Service Name
4. Status

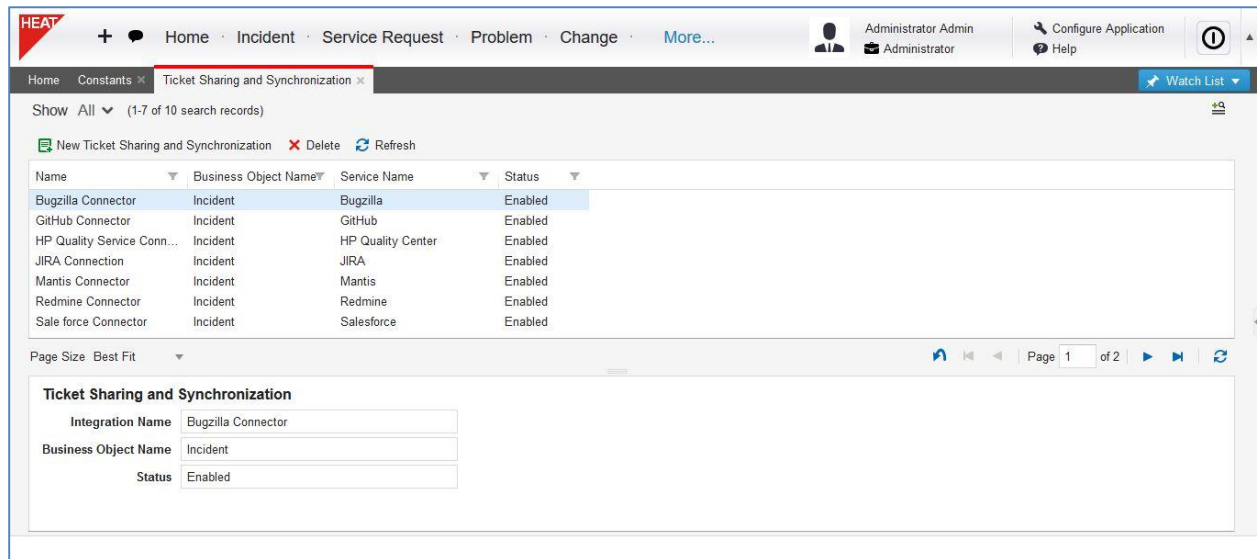


Fig.10

#### 4.2.2 Edit

Following is the sequence of steps to edit an existing ticket sharing and synchronization.

1. Double click a Connector Service Connection from the list
2. The edit view of the selected Connector Service Connection is displayed with Prepopulated data as shown in Fig.11

The screenshot displays the 'Ticket Sharing and Synchronization' configuration page in the Ivanti Ticket Synchronization Framework. The interface includes a top navigation bar with links for Home, Incident, Service Request, Problem, Change, and More... The user is logged in as 'Administrator Admin'. The page shows a list of search records (10 of 11) and a 'Watch List' button. The main configuration area is titled 'Ticket Sharing and Synchronization' and contains the following fields:

- Integration Name: VSTFS Connection
- Business Object Name: Incident
- Status: Enabled

Below these fields are tabs for 'Credentials', 'Filters (1)', 'Field Mapping (2)', 'Data Mapping (2)', 'Data Rules (1)', 'Additional Operations', and 'Comment Field Mapping (2)'. The 'Credentials' tab is active, showing the following fields:

- Service Name: VSTFS
- Service URL: https://heatconnector.visualstudio.com/
- Username: HEAT Admin
- Password: [Redacted]

Fig.11

3. Set the Status depending on whether you want the integration to run or not
4. In the credentials tab, update the fields as required
  - a. The Service URL is the url which the Defect Management application host is accessed.
    - i. E.g. salesforce.com might be <https://na3.salesforce.com> depending on your location.
    - ii. If the url to the Web Service API of the integration in question is different than your tenant, use the host name where the integration is accessed.
5. Edit Filter
  - a. Click the Filters Tab
  - b. Double Click the filter from the list
  - c. A pop up is displayed with the filter values prepopulated as shown in Fig.12

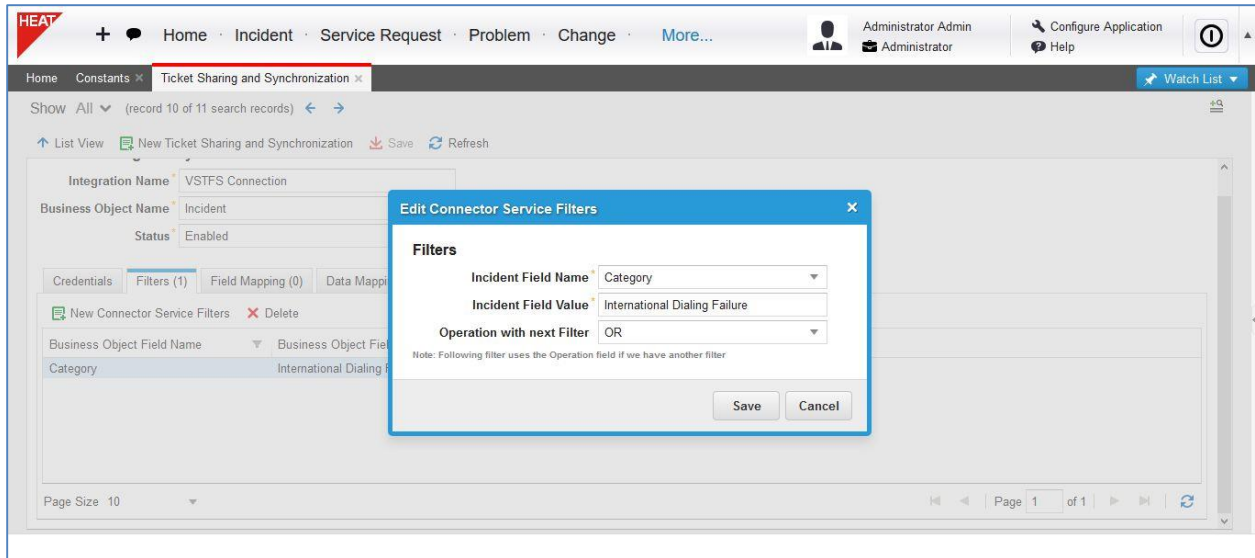


Fig.12

- d. Edit one or more values
  - e. Click Save
  - f. The modified values are updated
  - g. Click Cancel to retain the existing values
6. Edit Field Mapping
- a. Click the Field Mapping Tab
  - b. Double Click the mapping from the list
  - c. A pop up is displayed with the values prepopulated as shown in Fig.13

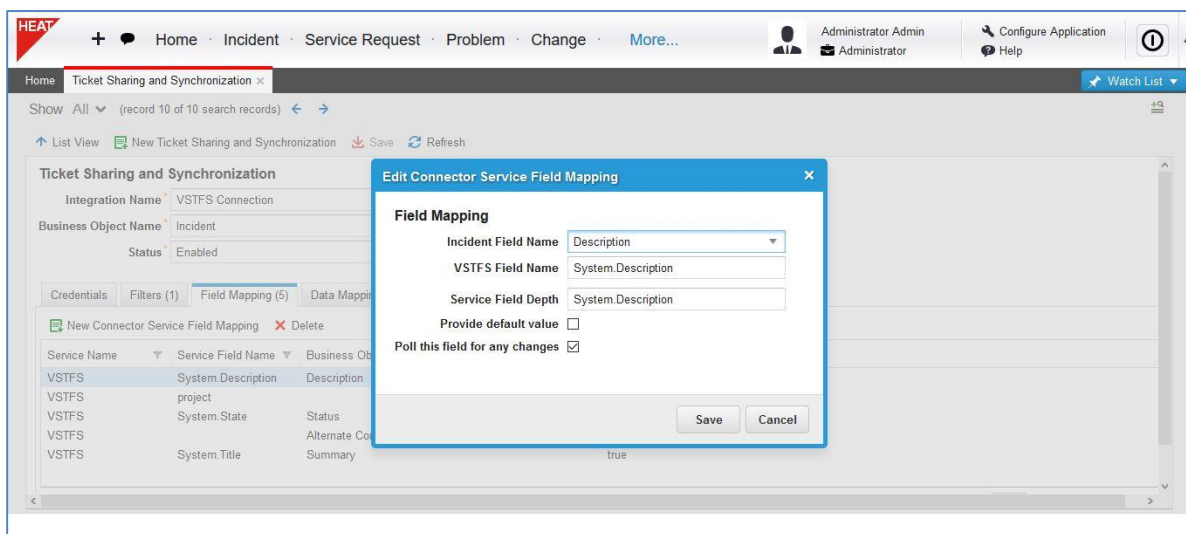


Fig.13

- d. Edit one or more values
  - e. Click Save
  - f. The modified values are updated
  - g. Click Cancel to retain the existing values
7. Edit Data Mapping
- a. Click the Data Mapping Tab
  - b. Double Click the data mapping from the list
  - c. A pop up is displayed with the values prepopulated as shown in Fig. 14

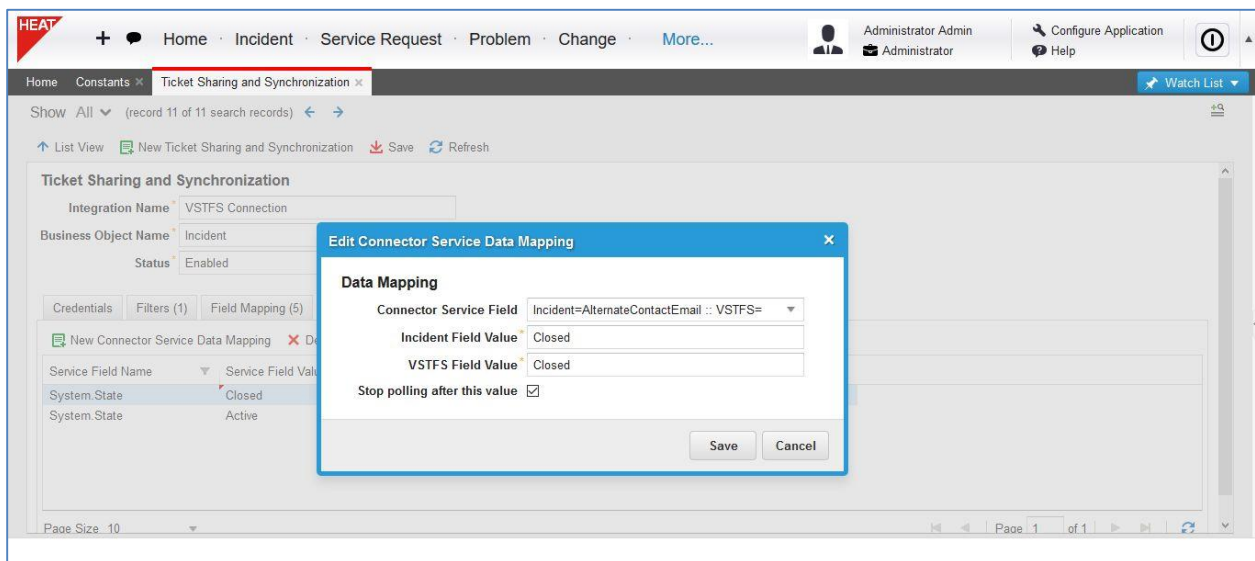


Fig.14

- d. Edit one or more values
  - e. Click Save
  - f. The modified values are updated
  - g. Click Cancel to retain the existing values
8. Edit Data Rules
- a. Click the Data Rules Tab
  - b. Double Click the Data Rule from the list
  - c. A pop up is displayed with the values prepopulated as shown in Fig.15

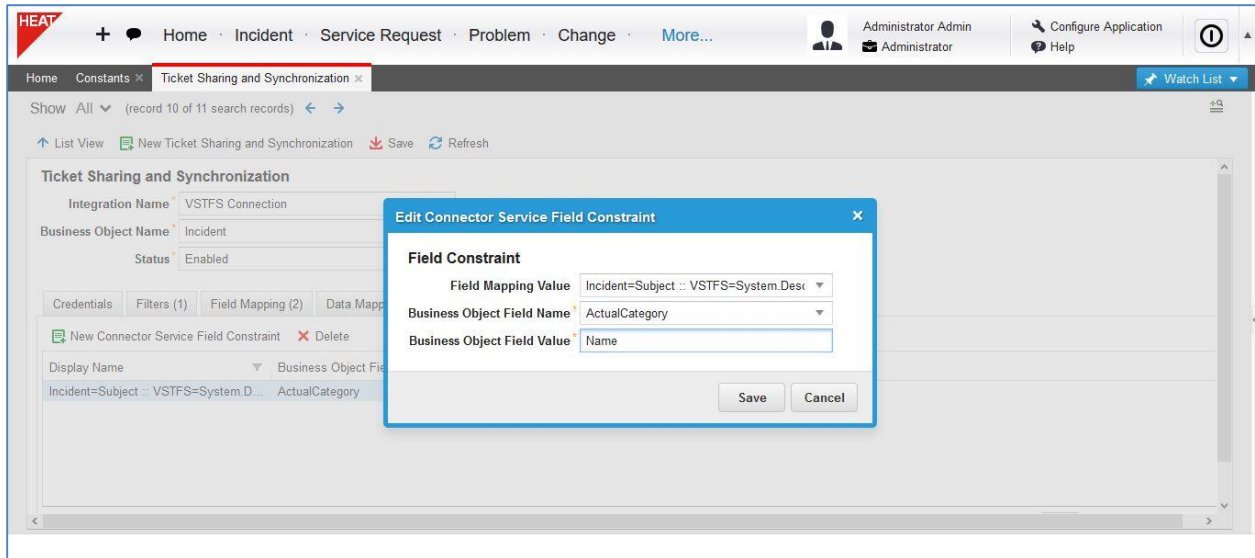


Fig.15

- d. Edit one or more values
- e. Click Save
- f. The modified values are updated
- g. Click Cancel to retain the existing values
9. Edit Additional Operations
  - a. Click the Additional Operations tab
  - b. The additional options are displayed as shown in Fig.16

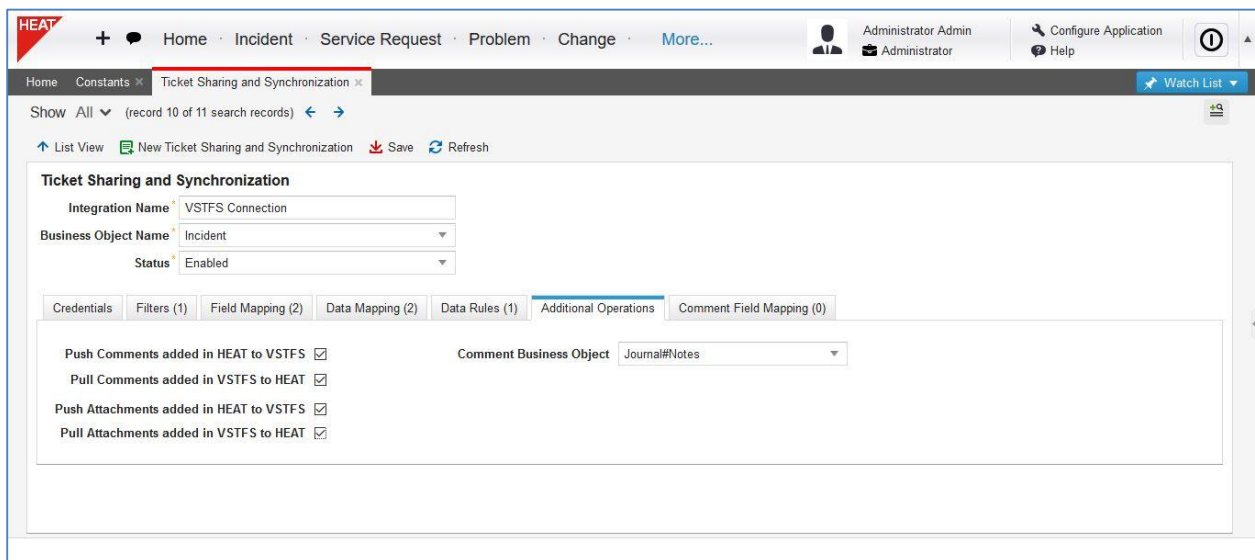


Fig.16

- c. Check / Uncheck one or more options
  - d. Click Save
  - e. The modified values are updated
10. Edit Comment Field Mapping
- a. Click the Comment Field Mapping Tab
  - b. Double Click the mapping from the list
  - c. A pop up is displayed with the values prepopulated as shown in Fig.17

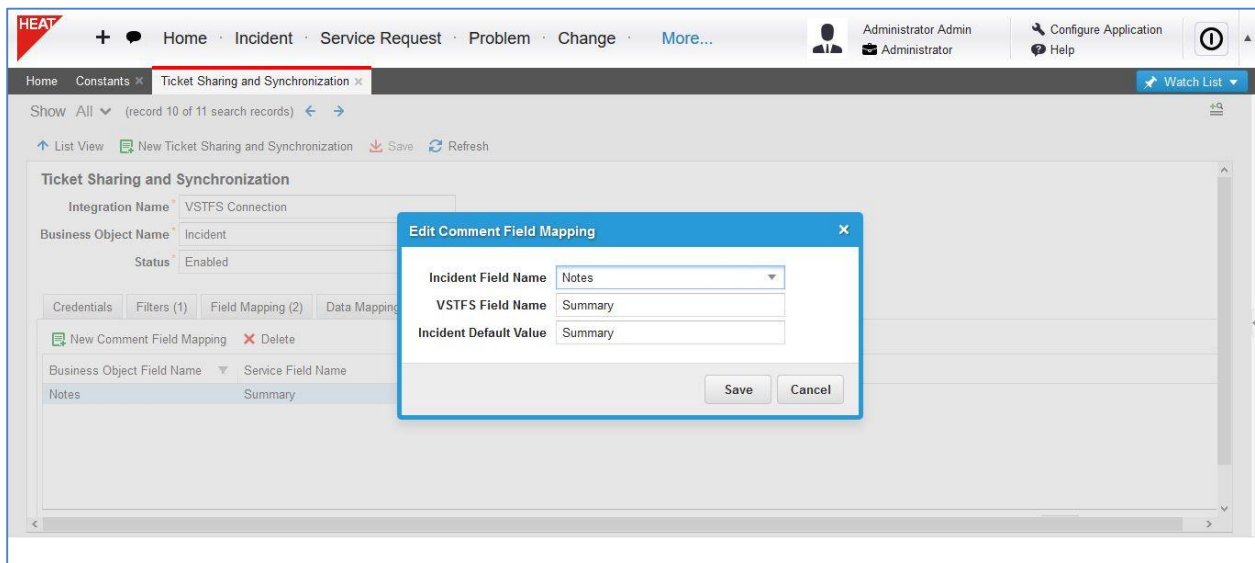


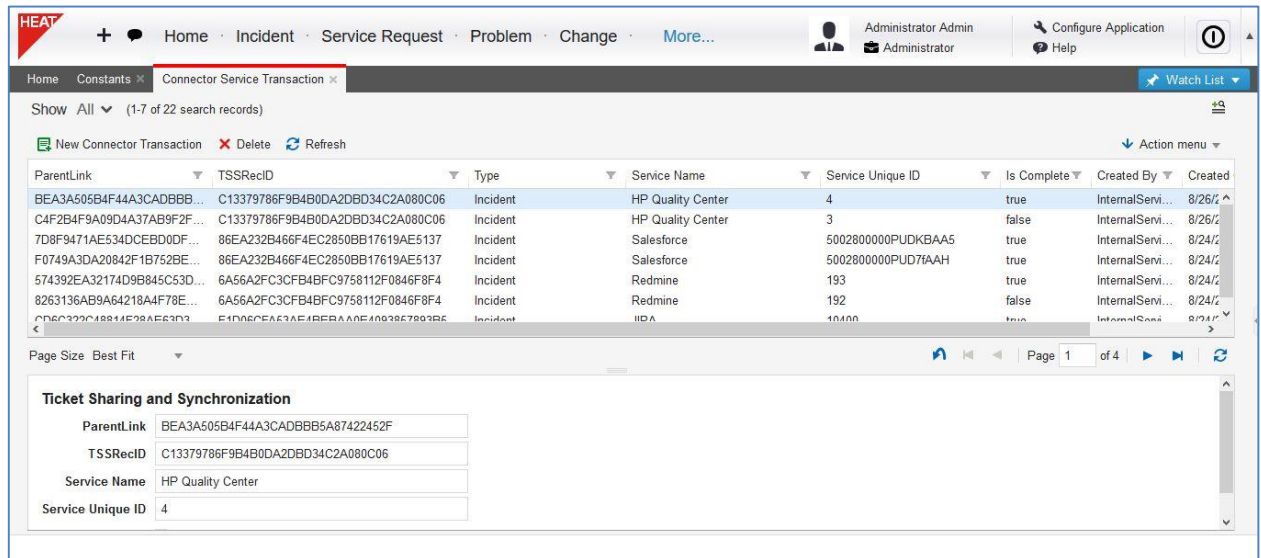
Fig.17

- d. Edit one or more values
  - e. Click Save
  - f. The modified values are updated
  - g. Click Cancel to retain the existing values
11. Click Save
12. The modified data is updated

## 5 Connector Service Transaction

The connector service transaction workspace lists all the transactions for the synchronizations between Ivanti and Ticket Synchronization Frameworks.

The list of transaction records is displayed as shown in Fig. 18



The screenshot shows the Ivanti Connector Service Transaction workspace. At the top, there is a navigation bar with links: Home, Incident, Service Request, Problem, Change, and More... The user is logged in as Administrator Admin. Below the navigation bar, there is a tab for 'Connector Service Transaction'. The main area displays a table of transaction records with columns: ParentLink, TSSRecID, Type, Service Name, Service Unique ID, Is Complete, Created By, and Created. Below the table, there is a section titled 'Ticket Sharing and Synchronization' with input fields for ParentLink, TSSRecID, Service Name, and Service Unique ID.

ParentLink	TSSRecID	Type	Service Name	Service Unique ID	Is Complete	Created By	Created
BEA3A505B4F44A3CAD8BB...	C13379786F9B4B0DA2DBD34C2A080C06	Incident	HP Quality Center	4	true	InternalSeri...	8/26/2...
C4F2B4F9A09D4A37AB9F2F...	C13379786F9B4B0DA2DBD34C2A080C06	Incident	HP Quality Center	3	false	InternalSeri...	8/26/2...
7D8F9471AE534DCEBD0DF...	86EA232B466F4EC2850BB17619AE5137	Incident	Salesforce	5002800000PUDKBAA5	true	InternalSeri...	8/24/2...
F0749A3DA20842F1B752BE...	86EA232B466F4EC2850BB17619AE5137	Incident	Salesforce	5002800000PUD7FAAH	true	InternalSeri...	8/24/2...
574392EA32174D9B845C53D...	6A56A2FC3CFB4BFC9758112F0846F8F4	Incident	Redmine	193	true	InternalSeri...	8/24/2...
8263136AB9A64218A4F78E...	6A56A2FC3CFB4BFC9758112F0846F8F4	Incident	Redmine	192	false	InternalSeri...	8/24/2...
C76C322C18814E28AE63D3...	E1D06CEA51AE4BEBA0E4093867893B6	Incident	HP Quality Center	10000	true	InternalSeri...	8/24/2...

**Ticket Sharing and Synchronization**

ParentLink: BEA3A505B4F44A3CAD8BB5A87422452F

TSSRecID: C13379786F9B4B0DA2DBD34C2A080C06

Service Name: HP Quality Center

Service Unique ID: 4

Fig. 18

The list displays,

1. ParentLink
2. TSSRecID
3. Type
4. Service Name
5. Service Unique ID
6. Is Complete
7. Created By
8. Created On

The transaction records are created and displayed for each of the following synchronization action.

1. Create Issue
2. Comments Outgoing
3. Comments Incoming
4. Attachments Outgoing
5. Attachments Incoming



## 6. Sync Issue

The list of Synchronization Actions for all the Ticket Synchronization Framework Services is displayed as shown in Fig.19

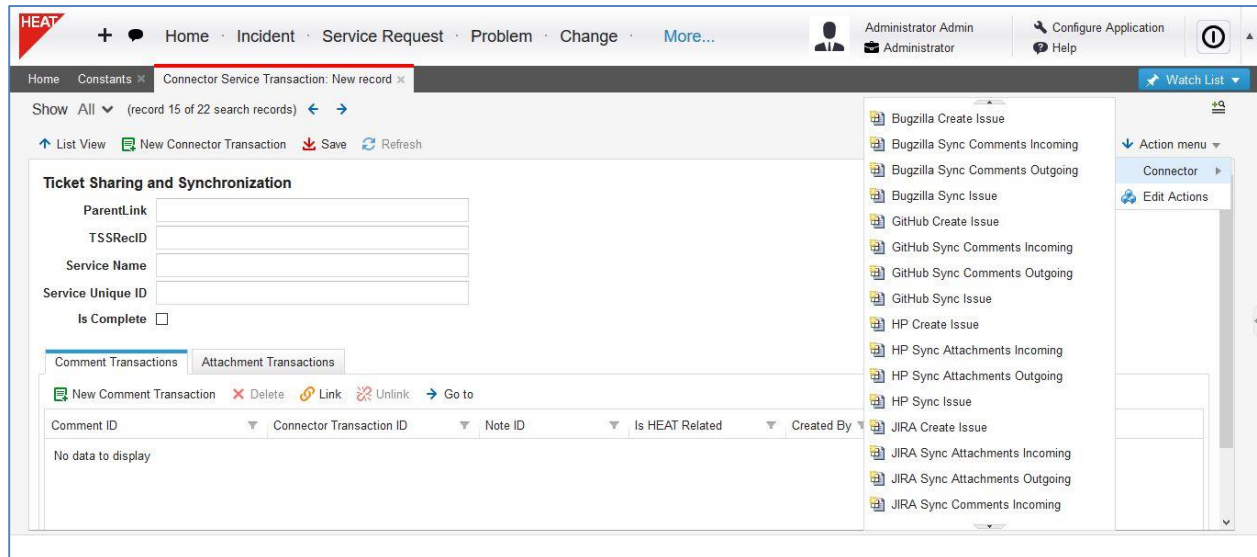


Fig.19

## Executing Action Scripts

This section describes the usage of each of the Action Scripts.

The user can execute each of the Action scripts for the various configured Ticket Synchronization Frameworks as follows.

1. In the Connector Service Transaction list, double click a transaction record for a specific connector
2. The details of the transaction are displayed as shown in Fig.20
  - a. Parent Link
  - b. TSSRecID
  - c. Service Name
  - d. Service Unique ID (Unique ID generated in the connector service)
  - e. Is Complete (Flag whether the corresponding incident is Closed or Not)
  - f. Two tabs
    - i. Comment Transactions
    - ii. Attachment Transactions

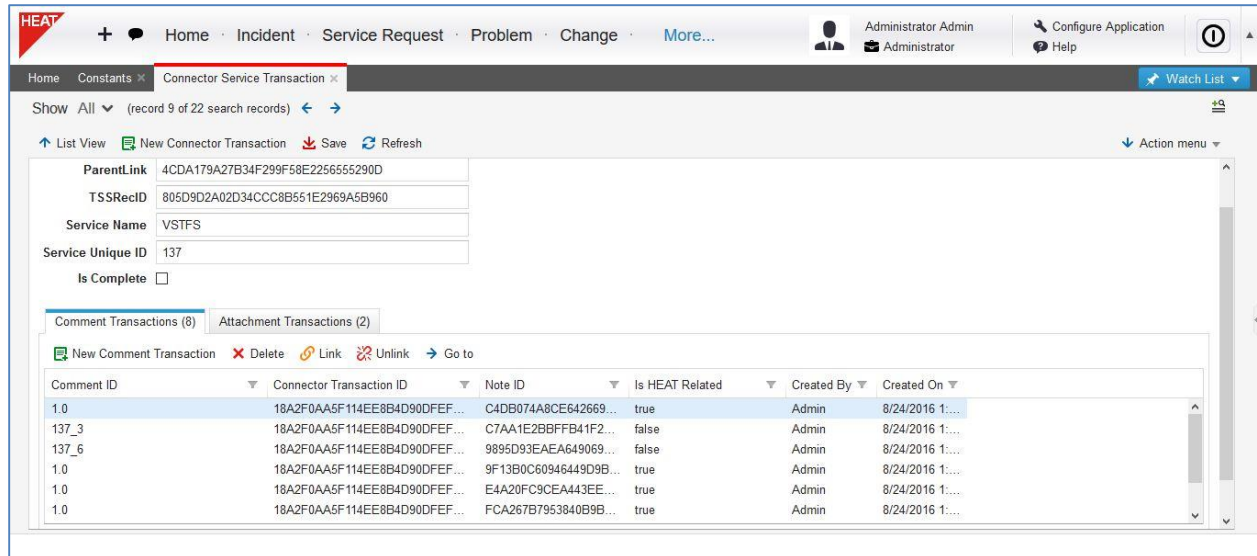


Fig.20

## Create Issue

- Create an Incident in Ivanti SM; creates an issue in the target Defect Connector and then the corresponding transaction record gets created
- User can validate whether the Issue got created in the target defect connector by checking the Service Unique ID value
  - If the value is not displayed immediately, click the Refresh link at the top of the grid
  - If the value is displayed, that means the Issue got created
- For some reason, if the issue does not get created in the defect connector service, the service unique Id will be blank as shown in Fig.21

The screenshot shows the Ivanti HEAT interface with the 'Connector Service Transaction' tab selected. A table lists various transactions with columns: ParentLink, TSSRecID, Type, Service Name, Service Unique ID, Is Complete, Created By, and Created. The first record is highlighted, showing a Service Unique ID of 4.

ParentLink	TSSRecID	Type	Service Name	Service Unique ID	Is Complete	Created By	Created
E9A7B6C7F4354EE1B32311...	805D9D2A02D34CCC8B551E2969A5B960	Incident	VSTFS	4	false	InternalSeri...	8/29/2...
BEA3A505B4F44A3CADBBB...	C13379786F9B4B0DA2DBD34C2A080C06	Incident	HP Quality Center	3	true	InternalSeri...	8/26/2...
C4F2B4F9A09D4A37AB9F2F...	C13379786F9B4B0DA2DBD34C2A080C06	Incident	HP Quality Center	3	false	InternalSeri...	8/26/2...
7D8F9471AE534DCEBD0DF...	86EA232B466F4EC2850BB17619AE5137	Incident	Salesforce	5002800000PUDKBAA5	true	InternalSeri...	8/24/2...
F0749A3DA20842F1B752BE...	86EA232B466F4EC2850BB17619AE5137	Incident	Salesforce	5002800000PUD7FAAH	true	InternalSeri...	8/24/2...
574392EA3217AD9B845C53D...	6A56A2FC3CFB4BFC9758112F0846F8F4	Incident	Redmine	193	true	InternalSeri...	8/24/2...
8263136AB0A64218A4E78E...	6A56A2FC3CFB4BFC9758112F0846F8F4	Incident	Redmine	193	false	InternalSeri...	8/24/2...

Below the table, the 'Ticket Sharing and Synchronization' section shows the following details:

- ParentLink: E9A7B6C7F4354EE1B32311765F0FD6F3
- TSSRecID: 805D9D2A02D34CCC8B551E2969A5B960
- Service Name: VSTFS
- Service Unique ID: (empty)

Fig.21

- In this case, the user can select the transaction record related to the connector service, click the Action Menu -> Create Issue (For the selected Connector Service) to explicitly create the Issue
- The Issue gets created and the Service Unique ID is displayed as shown in Fig.22

The screenshot shows the Ivanti HEAT interface with the 'Connector Service Transaction' tab selected. The table is identical to Fig.21, but the 'Service Unique ID' column is highlighted for the first record, showing a value of 138.

ParentLink	TSSRecID	Type	Service Name	Service Unique ID	Is Complete	Created By	Created
E9A7B6C7F4354EE1B32311...	805D9D2A02D34CCC8B551E2969A5B960	Incident	VSTFS	138	false	InternalSeri...	8/29/2...
BEA3A505B4F44A3CADBBB...	C13379786F9B4B0DA2DBD34C2A080C06	Incident	HP Quality Center	4	true	InternalSeri...	8/26/2...
C4F2B4F9A09D4A37AB9F2F...	C13379786F9B4B0DA2DBD34C2A080C06	Incident	HP Quality Center	3	false	InternalSeri...	8/26/2...
7D8F9471AE534DCEBD0DF...	86EA232B466F4EC2850BB17619AE5137	Incident	Salesforce	5002800000PUDKBAA5	true	InternalSeri...	8/24/2...
F0749A3DA20842F1B752BE...	86EA232B466F4EC2850BB17619AE5137	Incident	Salesforce	5002800000PUD7FAAH	true	InternalSeri...	8/24/2...
574392EA3217AD9B845C53D...	6A56A2FC3CFB4BFC9758112F0846F8F4	Incident	Redmine	193	true	InternalSeri...	8/24/2...
8263136AB0A64218A4E78E...	6A56A2FC3CFB4BFC9758112F0846F8F4	Incident	Redmine	193	false	InternalSeri...	8/24/2...

Below the table, the 'Ticket Sharing and Synchronization' section shows the following details:

- ParentLink: E9A7B6C7F4354EE1B32311765F0FD6F3
- TSSRecID: 805D9D2A02D34CCC8B551E2969A5B960
- Service Name: VSTFS
- Service Unique ID: 138

Fig.22

## 1. Comments Outgoing

- Navigate to the created incident record and add a comment (Under Journal – Create Notes) as shown in Fig.23

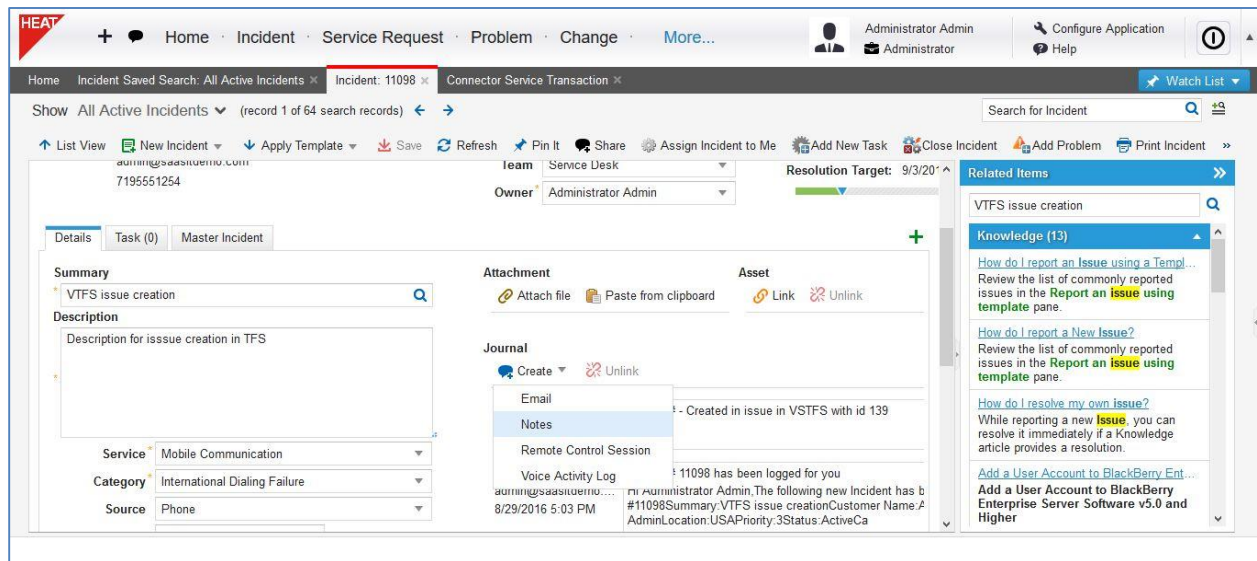


Fig.23

- A comment would be added for the incident as shown in Fig.24

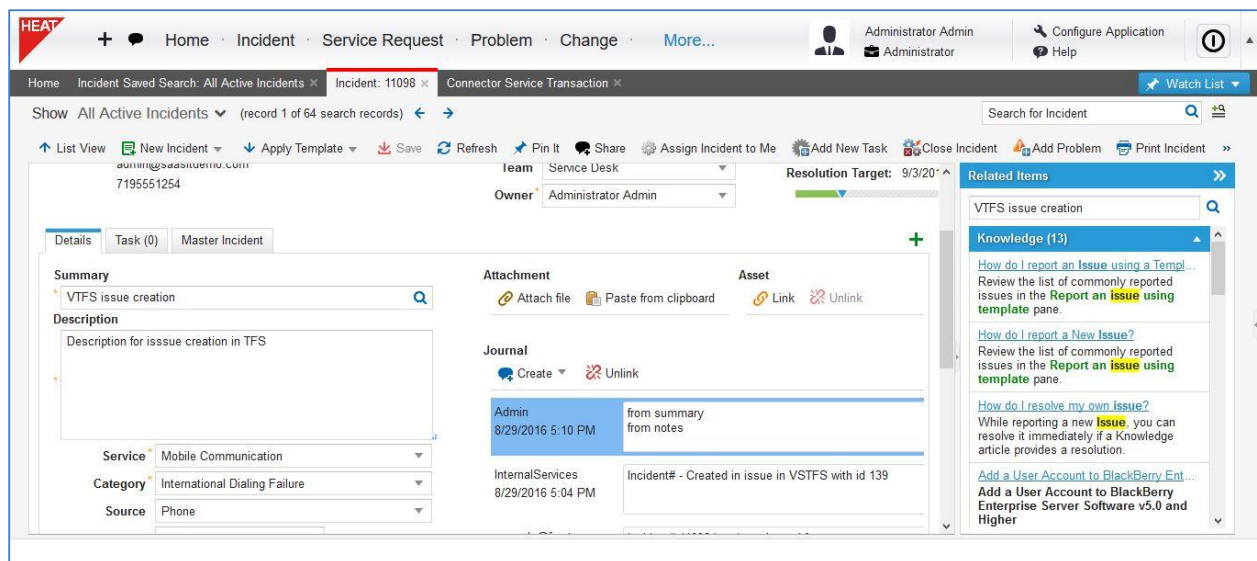


Fig.24

- Now navigate back to the transaction record view
- Click the Action Menu at the top right
- Click Connector
- List of all action scripts for the various integrated connector services are displayed
- Scroll to the required connector service item
- For example, click TFS Sync Comments Outgoing as shown in Fig.25

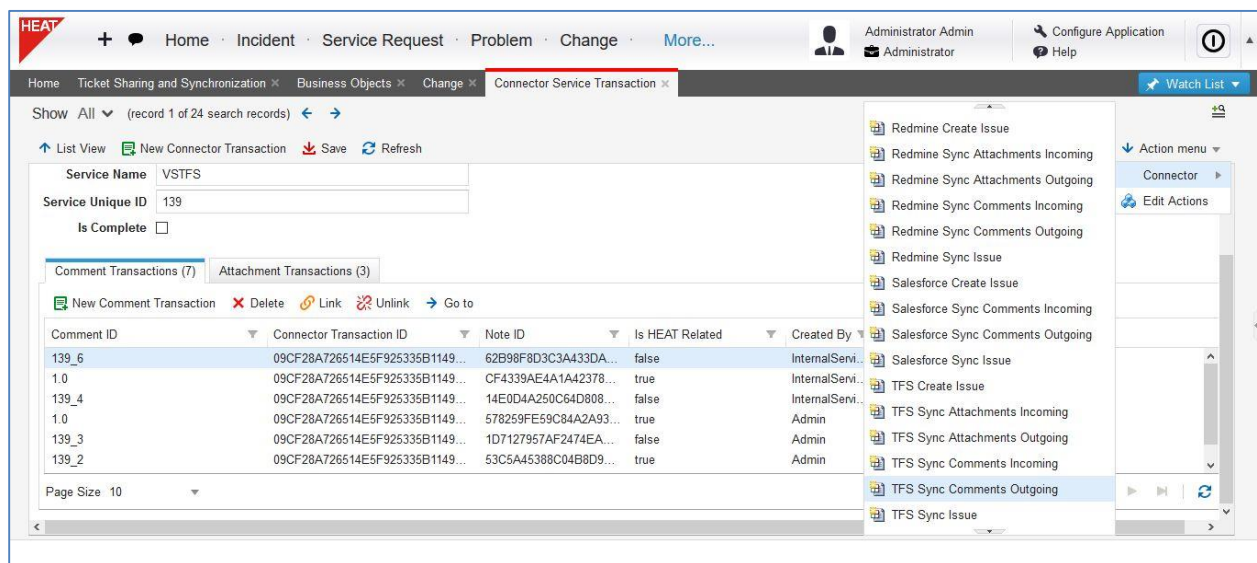


Fig.25

- A new Comment transaction record would be created and displayed in the Comment Transactions tab
- Login and Check in the TFS Connector Service
- The comment added in Ivanti Incident would be synced successfully and displayed in the TFS Issue as shown in the Fig.26

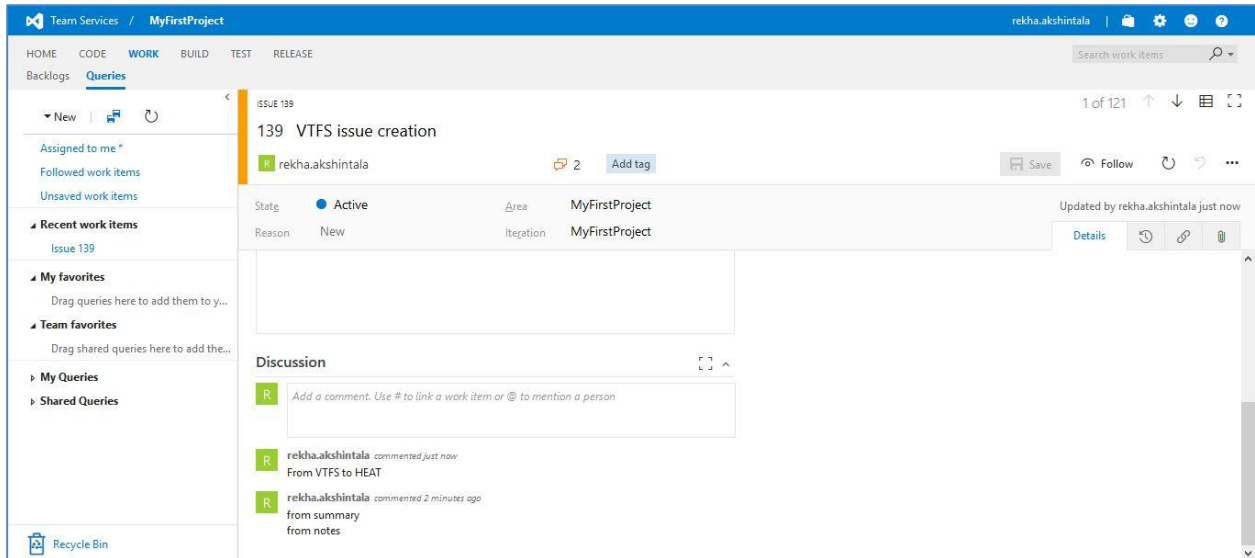


Fig.26

## 2. Comments Incoming

- Navigate to the TFS Issue
- Add a comment to that issue as displayed in Fig.27

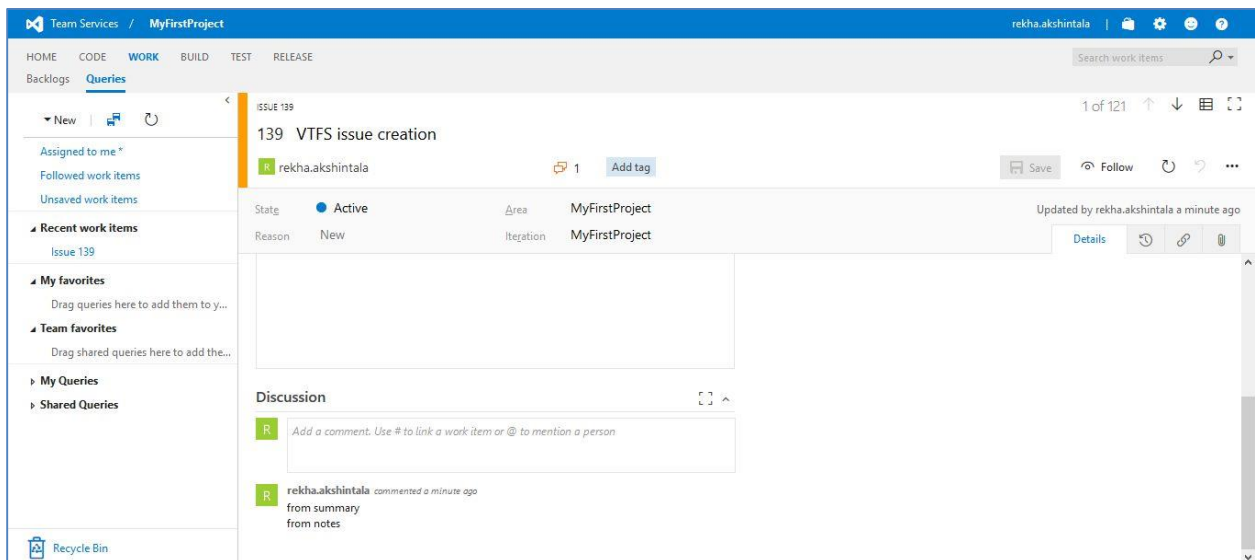


Fig.27

- Now navigate back to the transaction record detail view
- Click the Action Menu at the top right
- Click Connector
- Scroll to the required connector service item



- Click TFS Sync Comments Incoming as shown in Fig.28

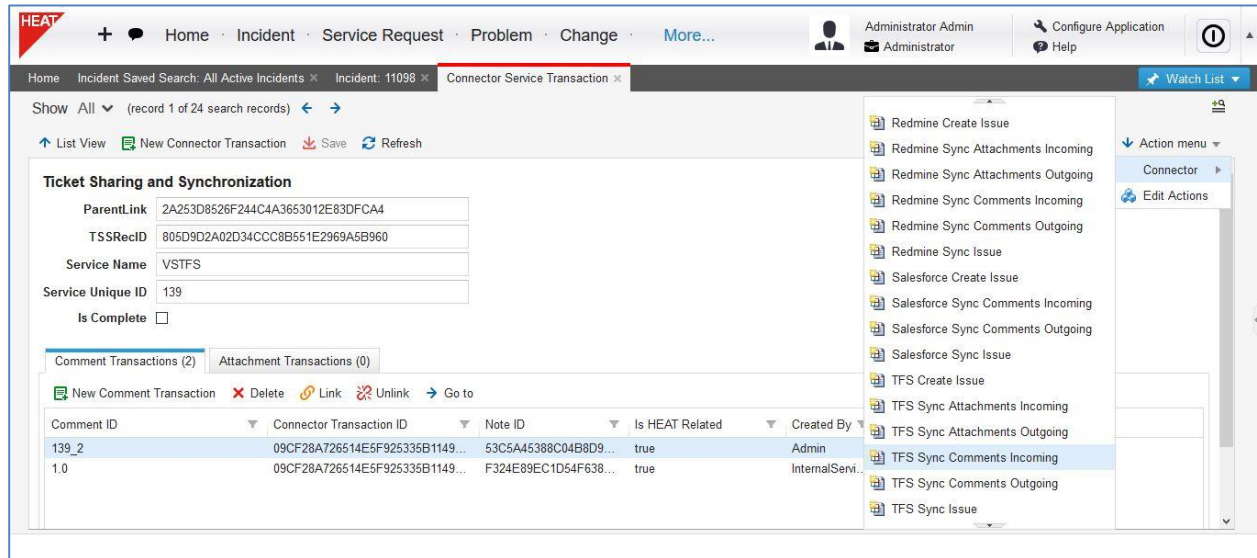


Fig.28

- A new Comment transaction record would be created and displayed in the Comment Transactions tab as shown in Fig.29

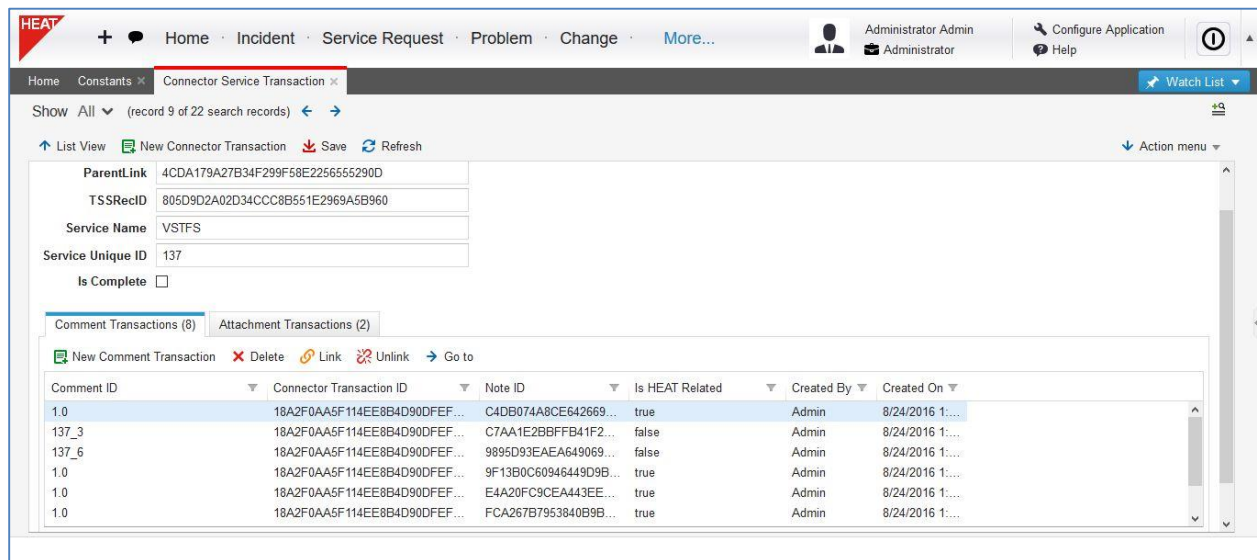


Fig.29

- Check the incident record in Ivanti, a new comment would be synced from TFS and displayed successfully as shown in Fig.30



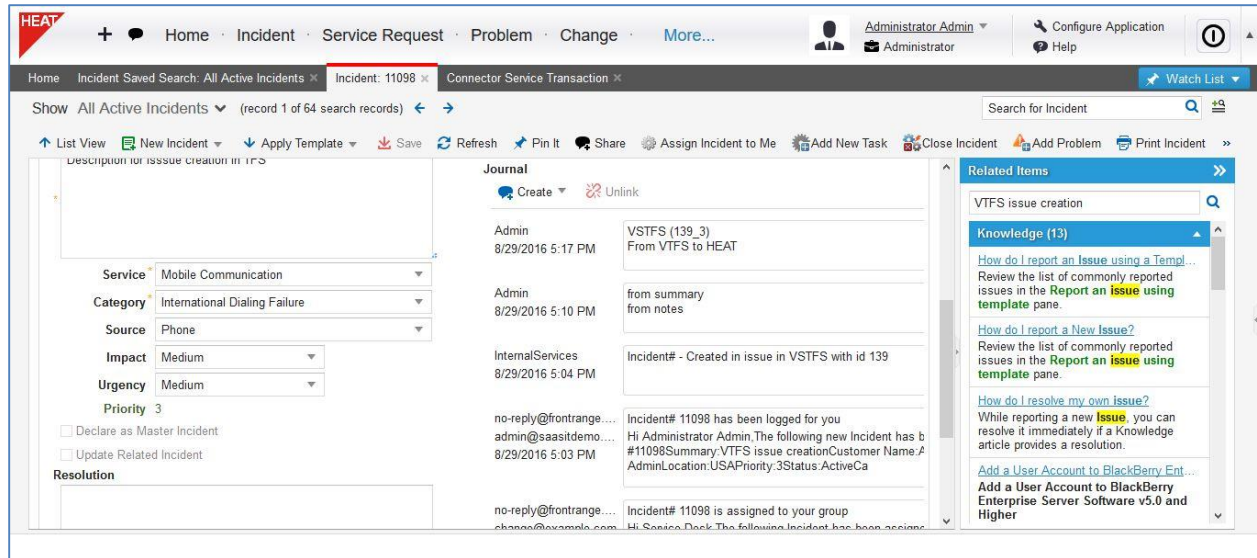


Fig.30

### 3. Attachments Outgoing

- Navigate to the created incident record and add an attachment as shown in Fig.31

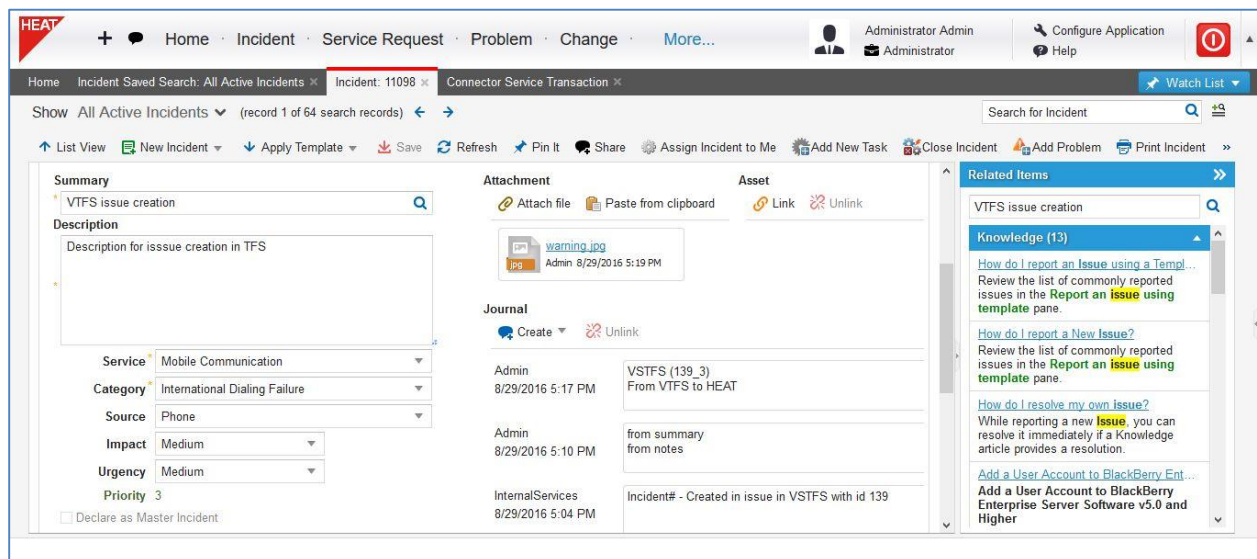


Fig.31

- Now navigate back to the transaction record view
- Click the Action Menu at the top right
- Click Connector
- Scroll to the required connector service item
- For example, click TFS Sync Attachments Outgoing as shown in Fig.32

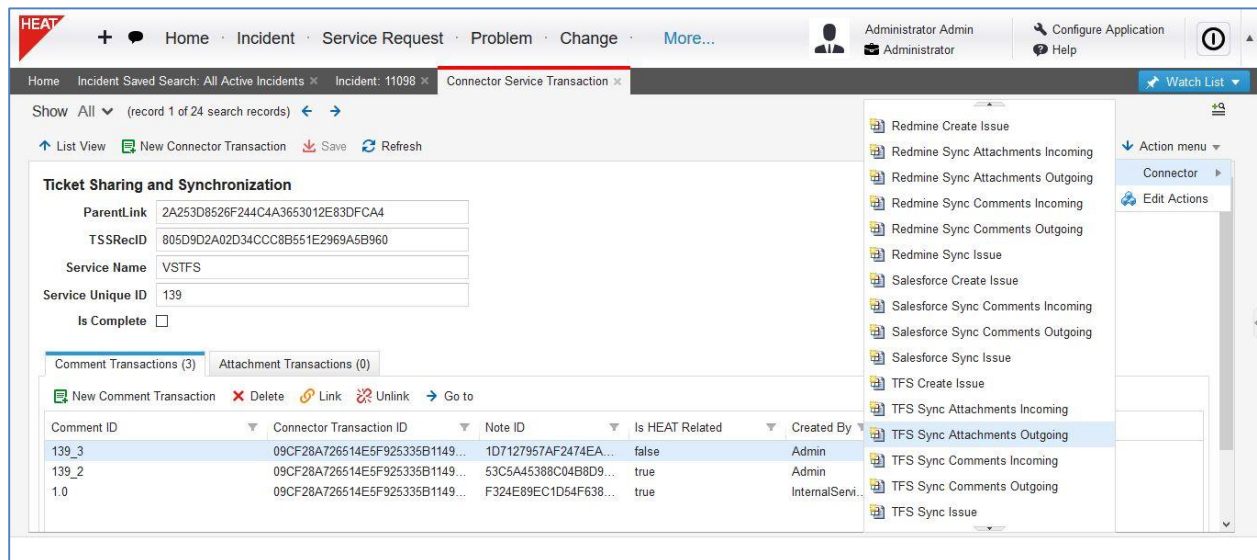


Fig.32

- A new attachment transaction record would be created and displayed in the Attachment Transactions tab as shown in Fig.33

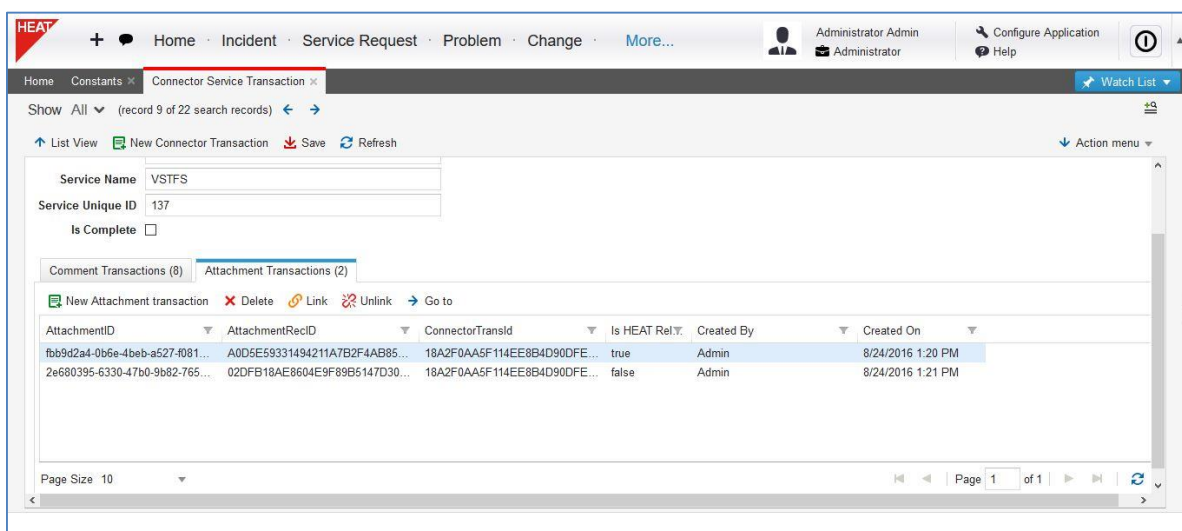


Fig.33

- Login and Check in the TFS Connector Service
- The Attachment added in Ivanti Incident would be synced successfully and displayed in the TFS Issue as shown in Fig.34

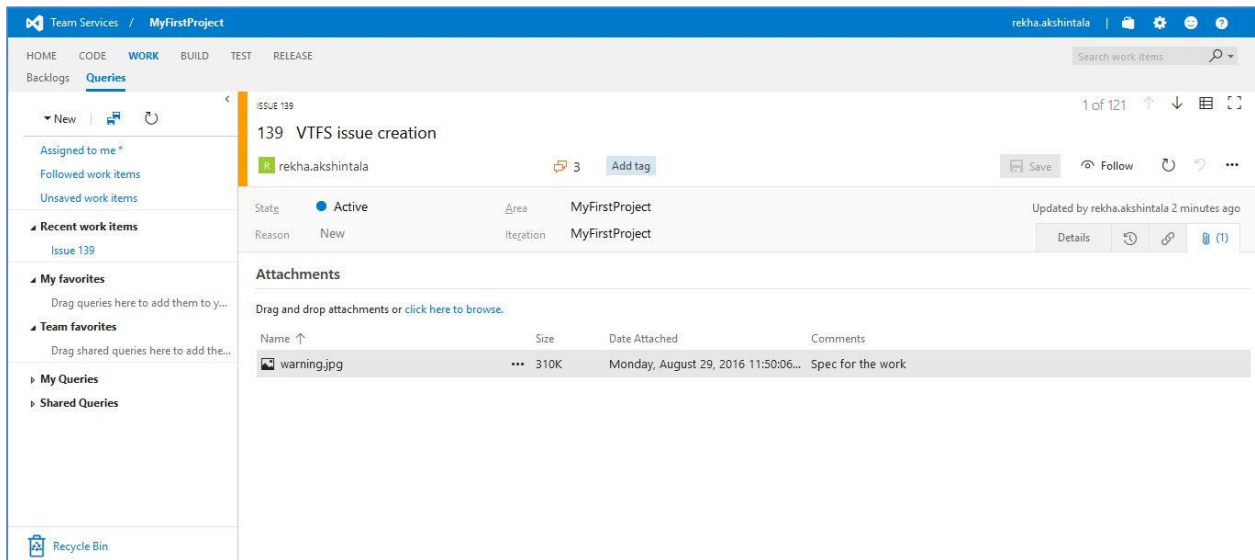


Fig.34

#### 4. Attachments Incoming

- Navigate to the TFS Issue
- Add an Attachment to that issue
- Now navigate back to the transaction record detail view
- Click the Action Menu at the top right
- Click Connector
- Scroll to the required connector service item
- Click TFS Sync Attachments Incoming as shown in Fig.35

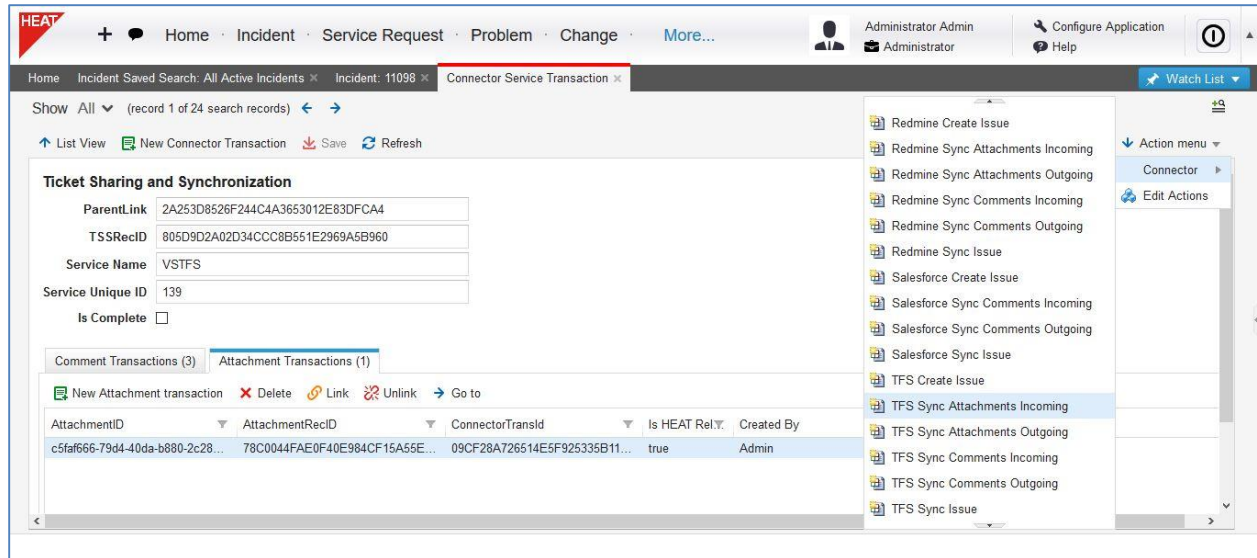


Fig.35

- A new Attachment transaction record would be created and displayed in the Attachment Transactions tab
- Check the incident record in Ivanti, a new attachment would be synced from TFS and displayed successfully as shown in Fig.36

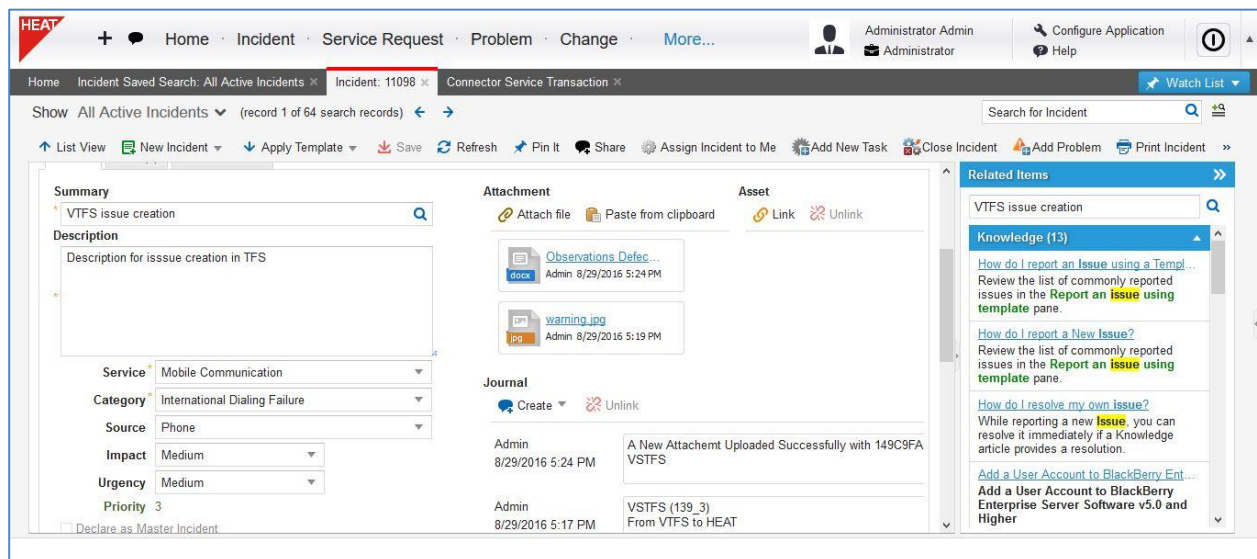


Fig.36

**Note:** The maximum size of the attachment is as allowed by the Ivanti SM. Currently it is 5 MB.

## 5. Sync Issue

- Navigate to the TFS Issue
- Update the status of the issue to Closed as shown in Fig.37

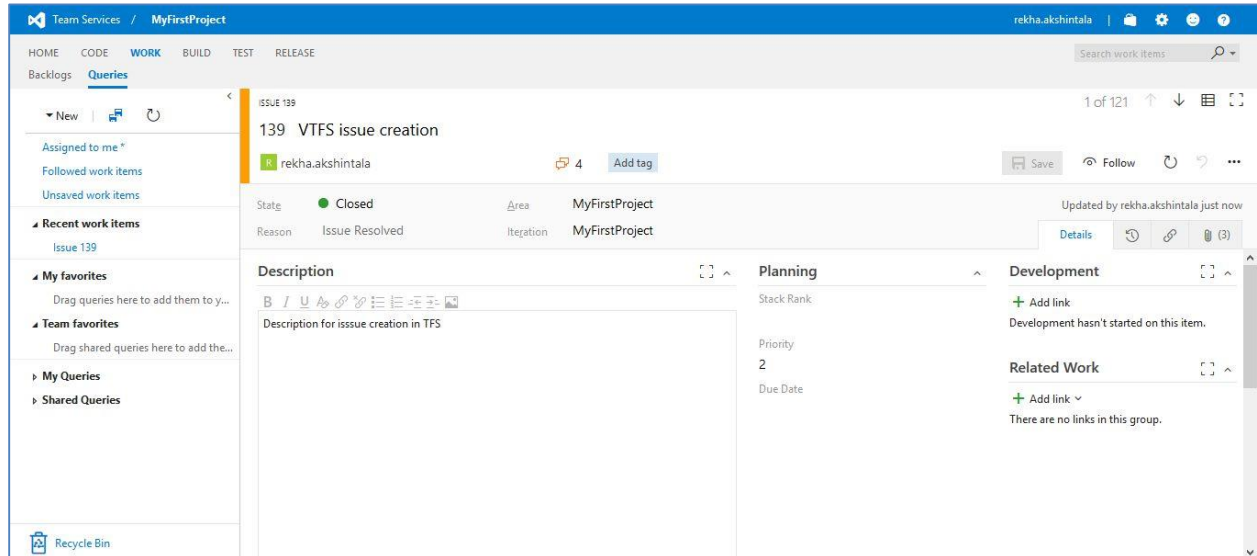


Fig.37

- Now navigate back to the transaction record view in Ivanti SM
- Click the Action Menu at the top right
- Click Connector
- Scroll to the required connector service item
- Click TFS Sync Issue as shown in Fig.38



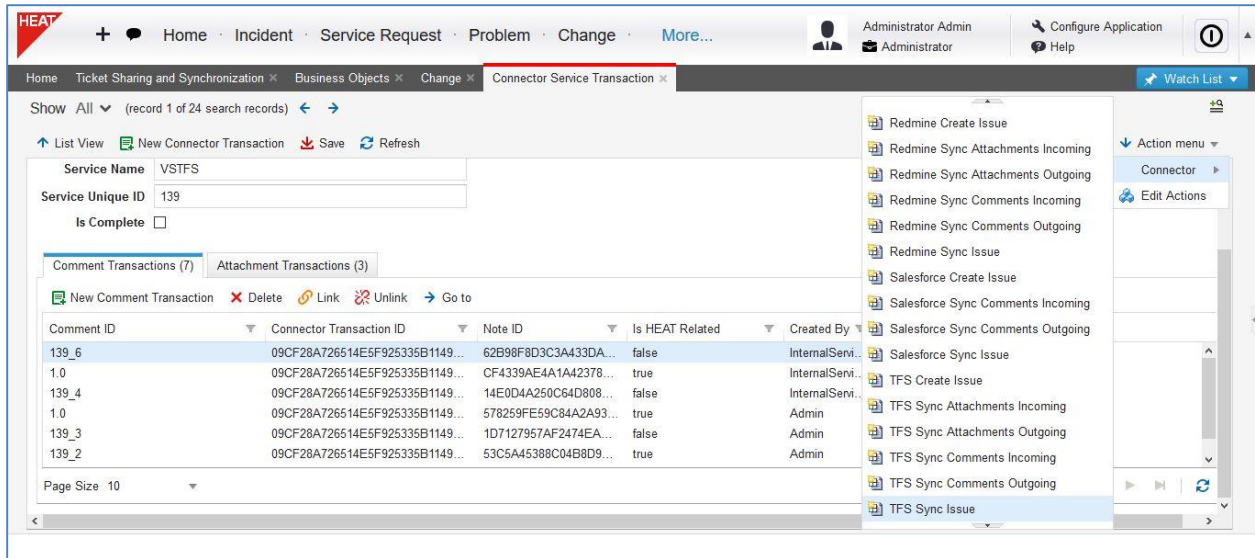


Fig.38

- Navigate to the Incident record
- Check the status
- It would be Closed as shown in Fig.39

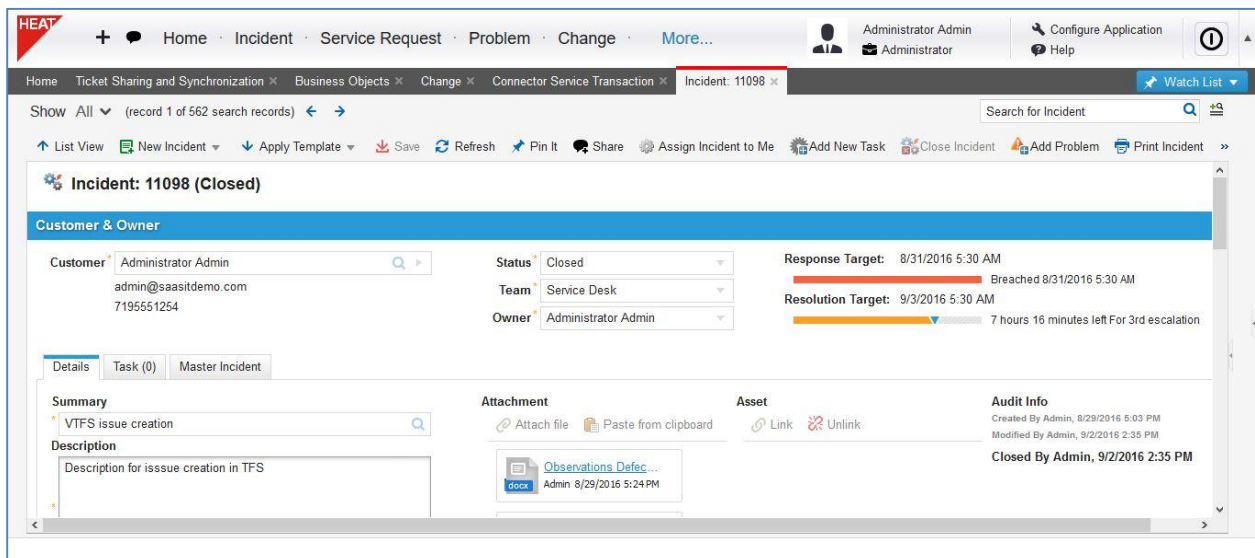


Fig.39

- In the transaction record, check the Is Complete option
- It would be checked indicating that the incident has been closed as show in Fig.40

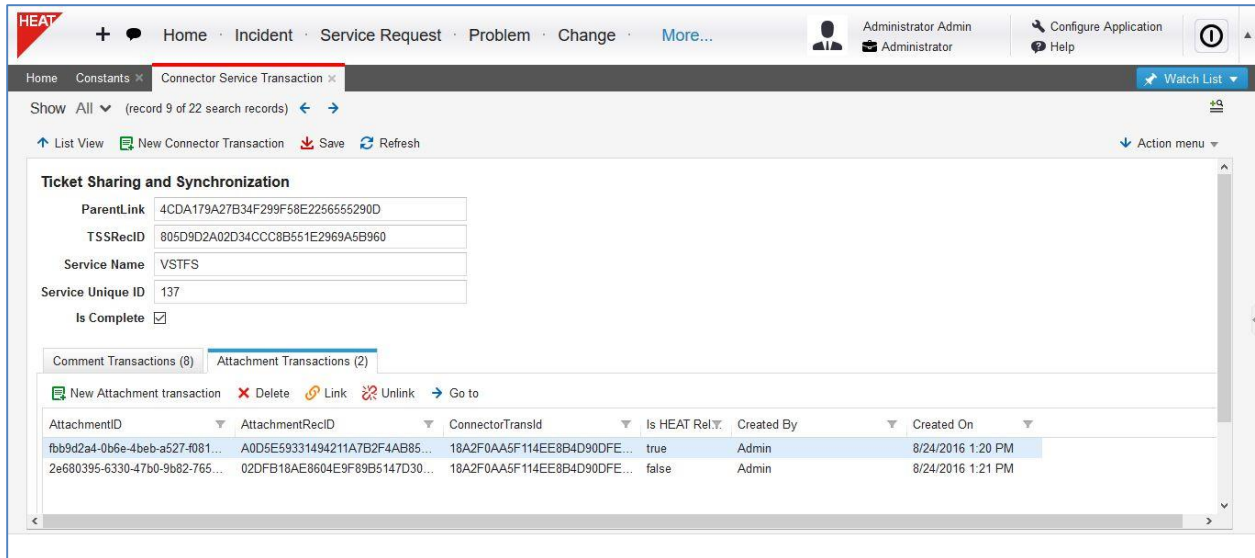


Fig.40

**Note:**

1. Not all the Connector Services have all the Action items. Please refer the table in Section 2.
2. Repeat the above steps 1 to 6 as applicable for each of the connector service

## 6 Logging

The Administrator can refer the log file or Logs workspace for any troubleshooting or Analysis.

The error / activity information is logged into a log file AppServer which is located in the path C:\Logs\ on the server the application is hosted. This is then automatically uploaded into the Logs workspace.

The Administrator should ensure that the configuration of the log is properly updated.

1. Navigate to the workspace Logging Configuration
2. Double click the SCRIPT Service Name
3. Set the Log Level to ALL as shown in Fig.41
4. Click Save

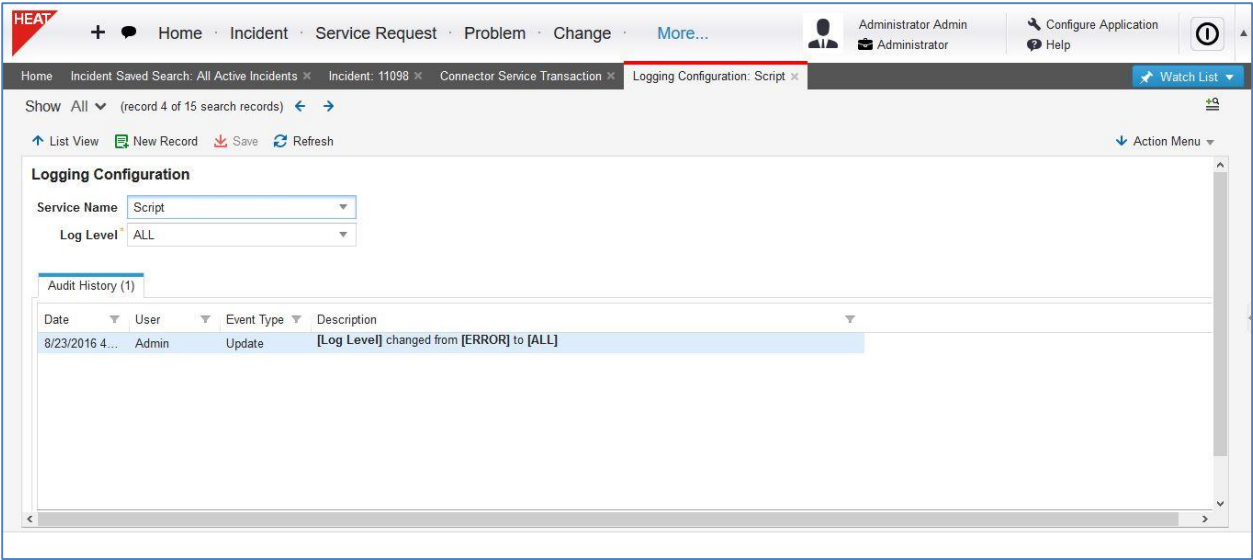
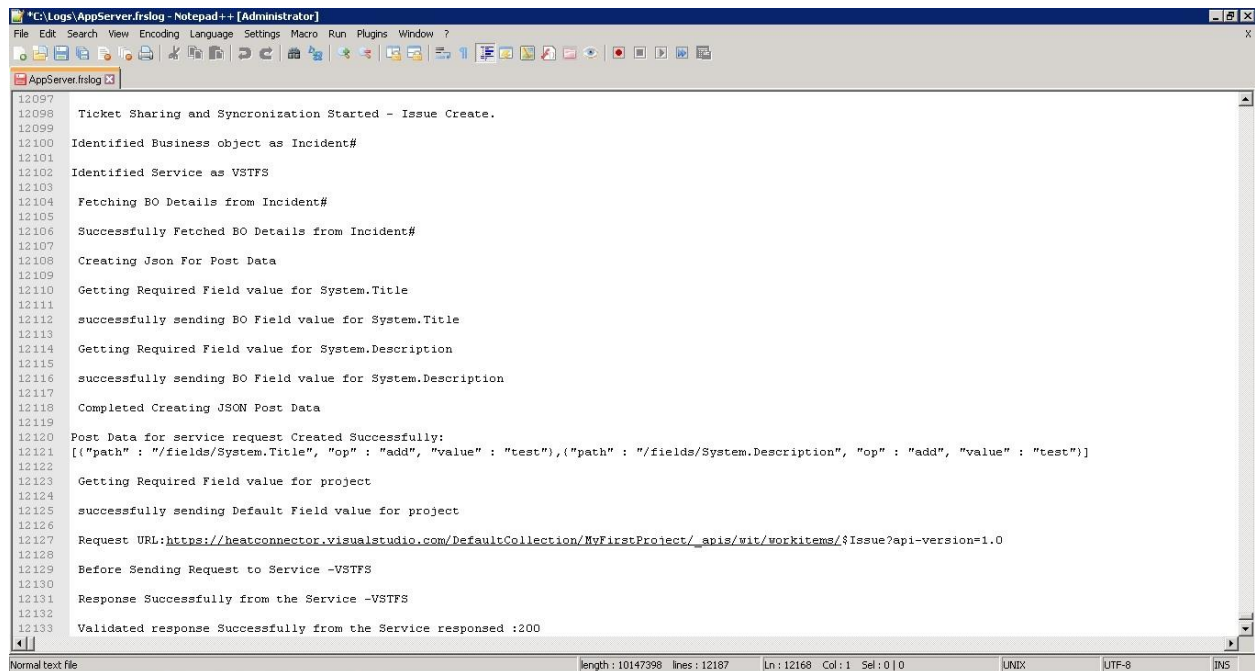


Fig.41



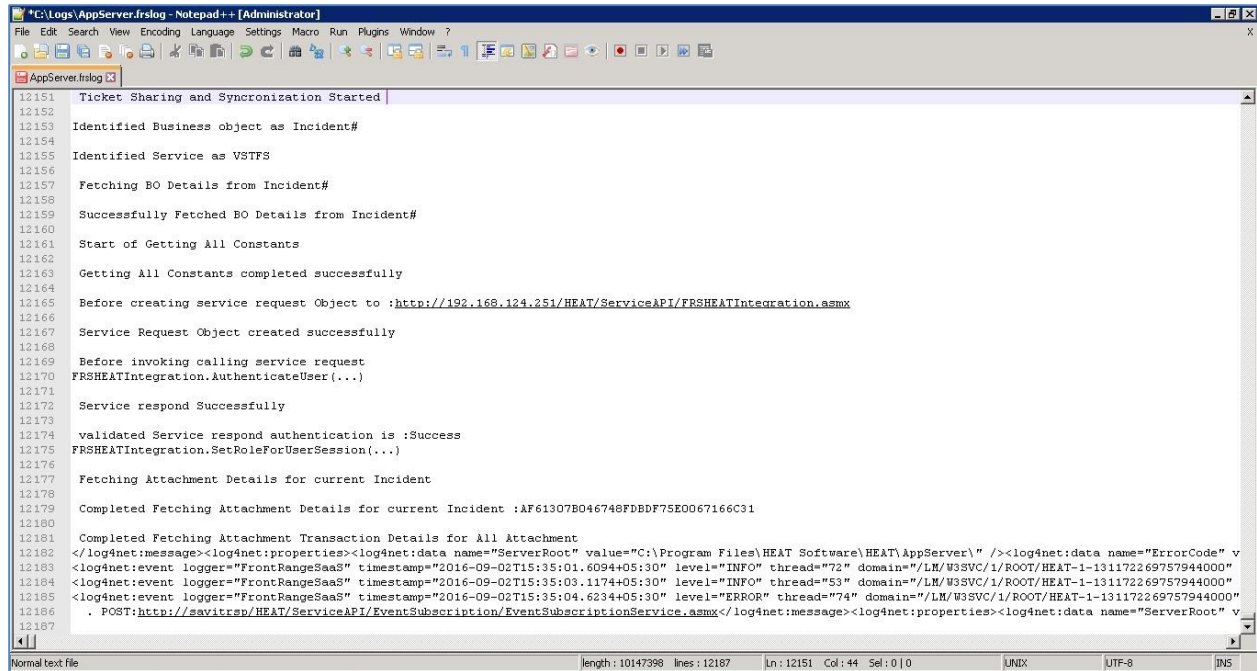
Following is the sequence of steps to check the logs.

1. If using the Log files:
  - a. In the Application Server, navigate to the folder C:\Logs or the Logs workspace
  - b. Check for the file AppServer of type FRSLOG
  - c. Open the file
  - d. The file displays any errors or activity logged as shown in Fig.42, Fig 43
2. If using the Logs workspace.
  - a. Open the workspace
  - b. Filter for appropriate logs
  - c. The records display any errors or activity logged. This will be similar to what is shown in Fig.42, Fig 43 but will be in log workspace format



```
12097 Ticket Sharing and Synchronization Started - Issue Create.
12098
12099 Identified Business object as Incident#
12100
12101 Identified Service as VSTFS
12102
12103 Fetching BO Details from Incident#
12104
12105 Successfully Fetched BO Details from Incident#
12106
12107 Creating Json For Post Data
12108
12109 Getting Required Field value for System.Title
12110
12111 successfully sending BO Field value for System.Title
12112
12113 Getting Required Field value for System.Description
12114
12115 successfully sending BO Field value for System.Description
12116
12117 Completed Creating JSON Post Data
12118
12119 Post Data for service request Created Successfully:
12120 [{"path" : "/fields/System.Title", "op" : "add", "value" : "test"}, {"path" : "/fields/System.Description", "op" : "add", "value" : "test"}]
12121
12122 Getting Required Field value for project
12123
12124 successfully sending Default Field value for project
12125
12126 Request URL:https://heatconnector.visualstudio.com/DefaultCollection/MyFirstProject/_apis/wit/workitems/$Issue?api-version=1.0
12127
12128 Before Sending Request to Service -VSTFS
12129
12130 Response Successfully from the Service -VSTFS
12131
12132 Validated response Successfully from the Service responded :200
12133
```

Fig.42



```
*C:\Logs\AppServer.frslog - Notepad++ [Administrator]
File Edit Search View Encoding Language Settings Macro Run Plugins Window ?
AppServer.frslog
12151 Ticket Sharing and Synchronization Started
12152
12153 Identified Business object as Incident#
12154
12155 Identified Service as VSTFS
12156
12157 Fetching BO Details from Incident#
12158
12159 Successfully Fetched BO Details from Incident#
12160
12161 Start of Getting All Constants
12162
12163 Getting All Constants completed successfully
12164
12165 Before creating service request Object to :http://192.168.124.251/HEAT/ServiceAPI/FRSHEATIntegration.asmx
12166
12167 Service Request Object created successfully
12168
12169 Before invoking calling service request
12170 FRSHEATIntegration.AuthenticateUser(...)
12171
12172 Service respond Successfully
12173
12174 validated Service respond authentication is :Success
12175 FRSHEATIntegration.SetRoleForUserSession(...)
12176
12177 Fetching Attachment Details for current Incident
12178
12179 Completed Fetching Attachment Details for current Incident :AF61307B046748FDBDF75E0067166C31
12180
12181 Completed Fetching Attachment Transaction Details for All Attachment
12182 </log4net:message><log4net:properties><log4net:data name="ServerRoot" value="C:\Program Files\HEAT Software\HEAT\AppServer\" /><log4net:data name="ErrorCode" v
12183 <log4net:event logger="FrontRangeSaaS" timestamp="2016-09-02T15:35:01.6094+05:30" level="INFO" thread="72" domain="/LM/W3SVC/1/ROOT/HEAT-1-131172269757944000"
12184 <log4net:event logger="FrontRangeSaaS" timestamp="2016-09-02T15:35:03.1174+05:30" level="INFO" thread="53" domain="/LM/W3SVC/1/ROOT/HEAT-1-131172269757944000"
12185 <log4net:event logger="FrontRangeSaaS" timestamp="2016-09-02T15:35:04.6234+05:30" level="ERROR" thread="74" domain="/LM/W3SVC/1/ROOT/HEAT-1-131172269757944000"
12186 . POST:http://savitrsp/HEAT/ServiceAPI/EventSubscription/EventSubscriptionService.asmx</log4net:message><log4net:properties><log4net:data name="ServerRoot" v
12187
Normal text file length: 10147398 lines: 12187 Ln: 12151 Col: 44 Sel: 0 | 0 UNIX UTF-8 JNS
```

Fig.43

## 7 IMPORTANT

Please note the following important aspects in the Ticket Synchronization Framework.

1. Once the package is imported and Ticket Synchronization Framework is installed, the configuration values for the connectors display some pre-configured sample data. **The User MUST update / change the sample data and add / remove the fields as need for his current Ticket Synchronization Framework instance.**
2. For the Ticket Synchronization Framework Bugzilla, during configuration ensure that the Description field the option “**Poll this field for any changes**” should be unchecked always
3. The comments in Salesforce should be added in Case Comment section
4. This User Manual only illustrates the configuration for the Incident Business Object; if the Ivanti SM instance has a different business object to be mapped, then the Administrator has to create the new business object and configure. Please contact your System Administrator for advance configuration.

## 8 Appendix I – Other Details

These details cover settings that are normally not required to be adjusted, but are included here in case new integrations are being designed.

### 8.1 Scheduled Job

After the package(s) have been imported successfully, a pre-configured scheduled job would be created. This scheduled job executes every 1 hour and runs the different active quick actions.

1. Navigate to Settings -> Monitor -> Scheduled Jobs
2. Search for the job “Hourly Schedule” of Type “Hourly Schedule”
3. The Schedule Job is displayed in the list as shown in Fig.44

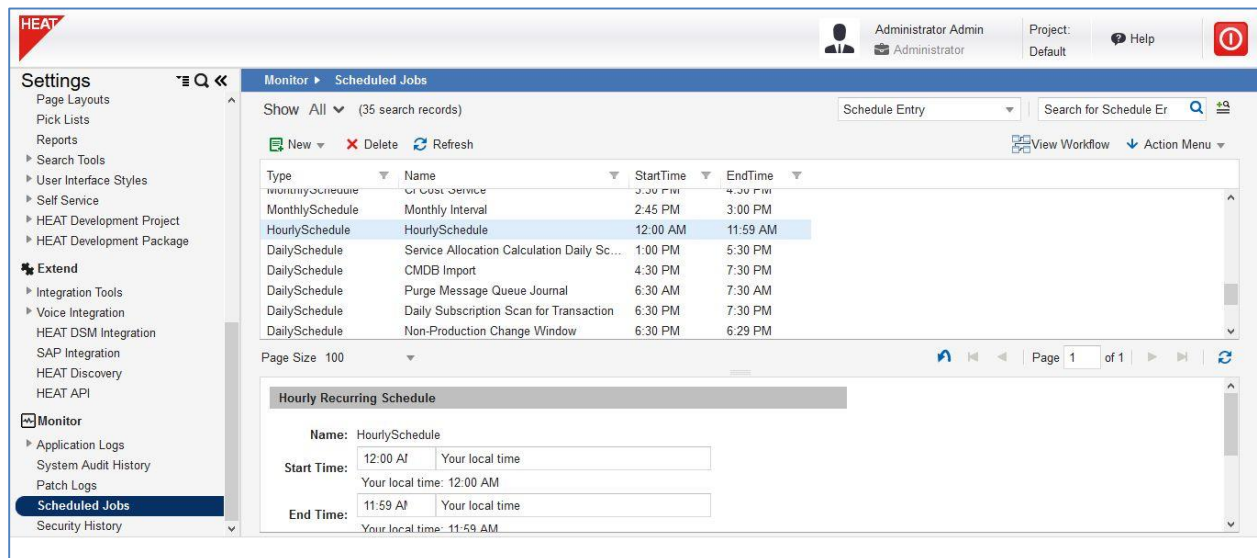


Fig.44

4. Click the record to view the details
5. The details are displayed as shown in Fig.45
6. The administrator can update the settings as required

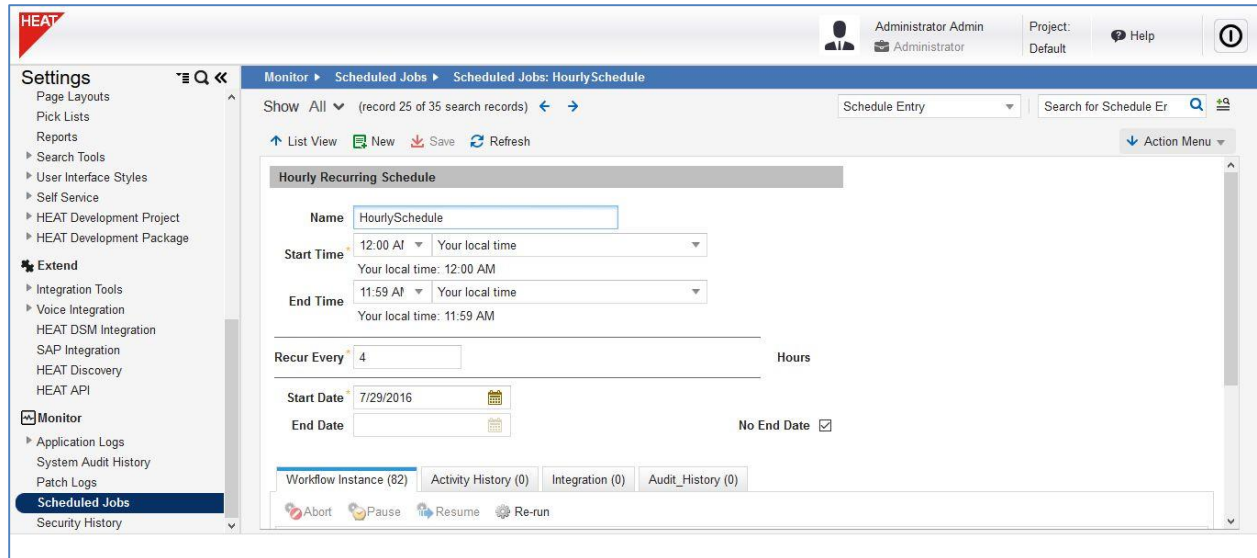


Fig.45