BigFix
BigFix 10 Insights Operations Guide

Special notice

Before using this information and the product it supports, read the information in Notices.

Edition notice

This edition applies to BigFix version 10 and to all subsequent releases and modifications until otherwise indicated in new editions.

Contents

Chapter 1. Version History

Date	Version	Note
30/11/2021	1.0	Initial release

Chapter 2. Introduction

BigFix Insights is an application-driven extract, transform, and load (ETL) that helps you consolidate all the BigFix data into a single data warehouse.

The Bigfix Insights Operations Guide is a reference to assist in the day-to-day operations of the Bigfix Insights environment. In order to get the most out of this guide, it is assumed that you have performed the required activities in the Bigfix Insights Getting Started Guide and have implemented a functioning Bigfix Insights environment.

Many of the items described here would be performed by a Bigfix Master Operator in practice, or by the Bigfix Insights Adminitrator if there is a separation of duties in your organization.

Chapter 3. Working with datasources and sites

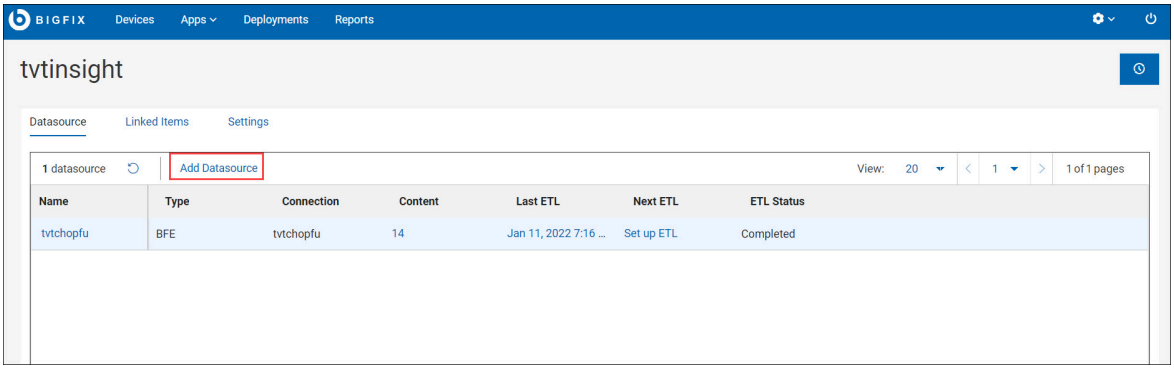
This section describes in detail about adding, deleting, and editing a datasource in BigFix 10 Insights. Datasource represents an existing BigFix Enterprise (BFEnterprise) database containing Fixlets, analyses, computers, sites, and results that Insights uses to visualize important data across your organization. Multiple datasources can be added to Insights to consolidate and correlate data for endpoints spread across multiple BigFix Enterprise Servers. In each datasource, individual sites can be added or excluded for use in BigFix Insights. Matching sites across multiple datasources are represented as a Linked Item in Insights

Adding a datasource

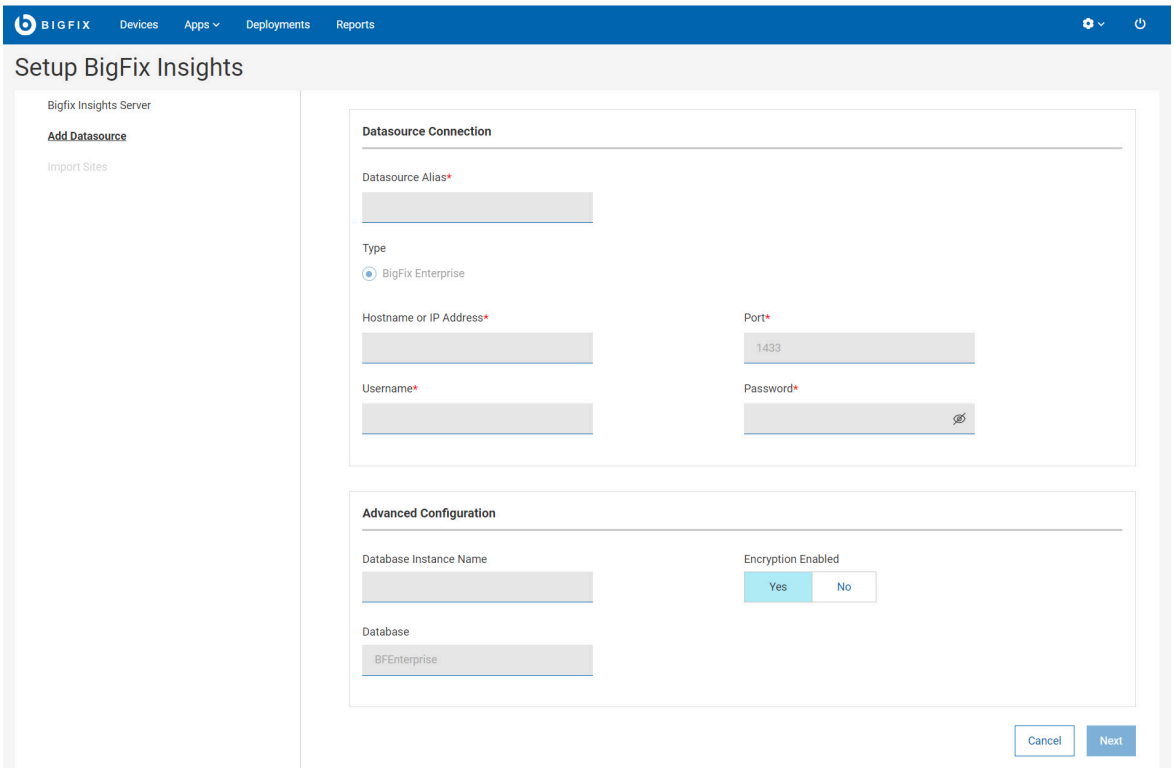
The Datasource section is used for BigFix datasource management. In this view, you can add, delete, and edit datasource connections and settings. After setting up BigFix Insights, the **Insights server home page** page is displayed.

Do the following steps to add a Datasource:

1. Within the Insights application, navigate to the **Datasources** tab.



2. Click **Add Datasource**. The **Setup BigFix Insights** page with **Datasource Connection** and **Advanced Configuration** details is displayed as shown below.



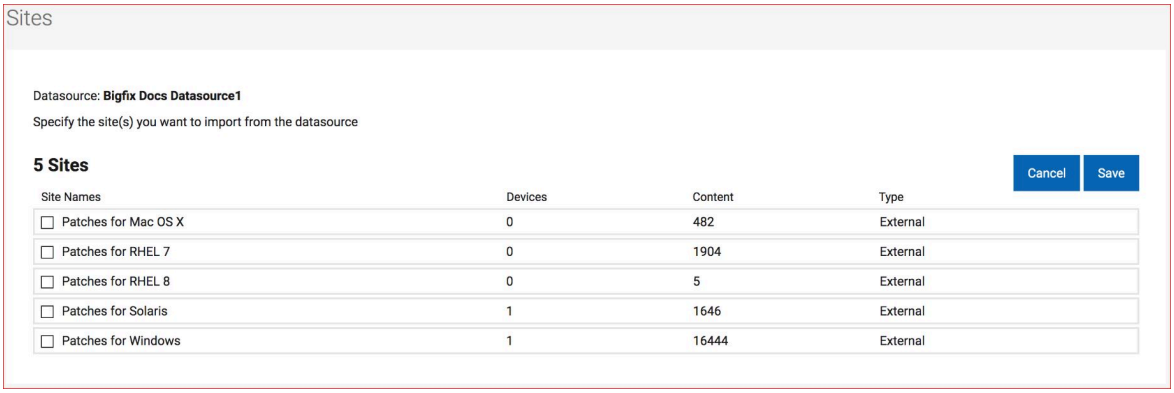
3. Enter the Datasource Connection details and Database Instance Name. This defines the datasource location and how to authenticate with the datasource. You **must** use SQL Authentication to connect to a datasource.

   Note: Fields marked with asterisk (*) are mandatory.

4. Click **Next**. The **Sites** page provides a list of external sites that are available for importing from this data source. This workflow allows you to include or exclude a site from the data source import.

   Note: The **Sites** page provides a dialog for users to include or exclude sites. You should only add sites that are interesting to be reported upon. For example, Checklist sites and Patch content sites may be of interest to report upon, however the remote control content site do not provide content to support interesting reports. This is up to your discretion. But in general, if more sites are added, the more data is retrieved and ETL's process needs a bit more time.

Once the sites are added to be imported, you cannot remove the sites(this is to assure the historical consistency of the Insights application). In general, consider having the sites imported to a minimum to start, and then add in additional sites if time and resourcing permits.



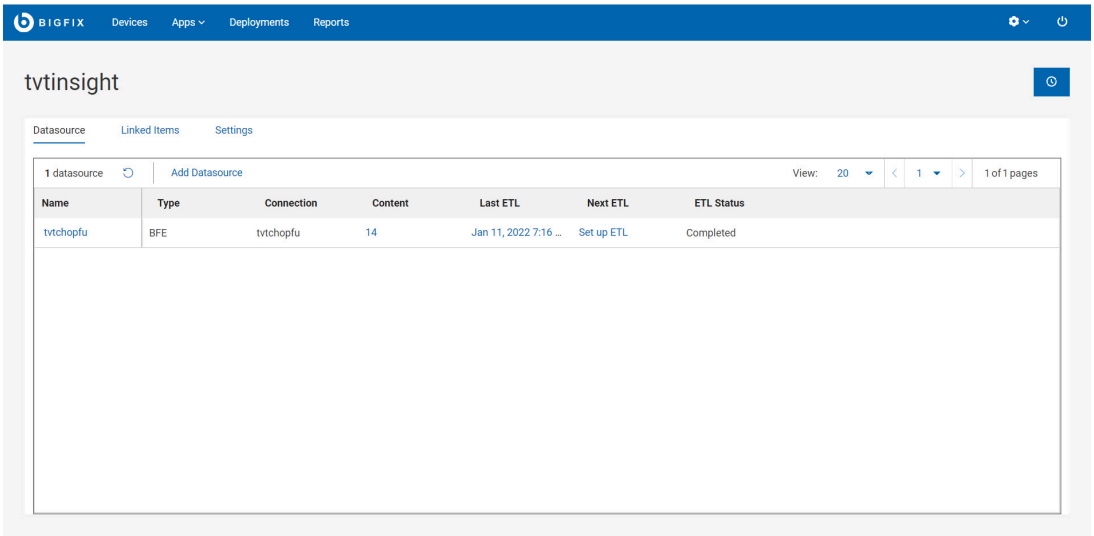
5. Include or exclude the site(s) that you want to import from the datasource by selecting or deselecting the check box.
    - If you include a site that is never imported in to any other datasource previously, it is set as a *primary* site because it is the only current representation of the content.
    - A primary datasource is the datasource whose represented content (such as Fixlet, Site, Task, Analysis and so on) is used as a single record to correlate to. A site that is primary cannot be excluded. To learn how to change the primary datasource for a site, see Working with Linked Items *(on page 10)*.
6. Click **Save**.

You have successfully added a datasource. You can view the newly added datasource in the **Datasources** tab of the **Details** page. The number of included sites is listed under **Sites**.

Viewing existing datasources

Within the Insights app, navigate to the Datasources tab. The current datasource definitions are displayed.

Figure 1. View existing datasources

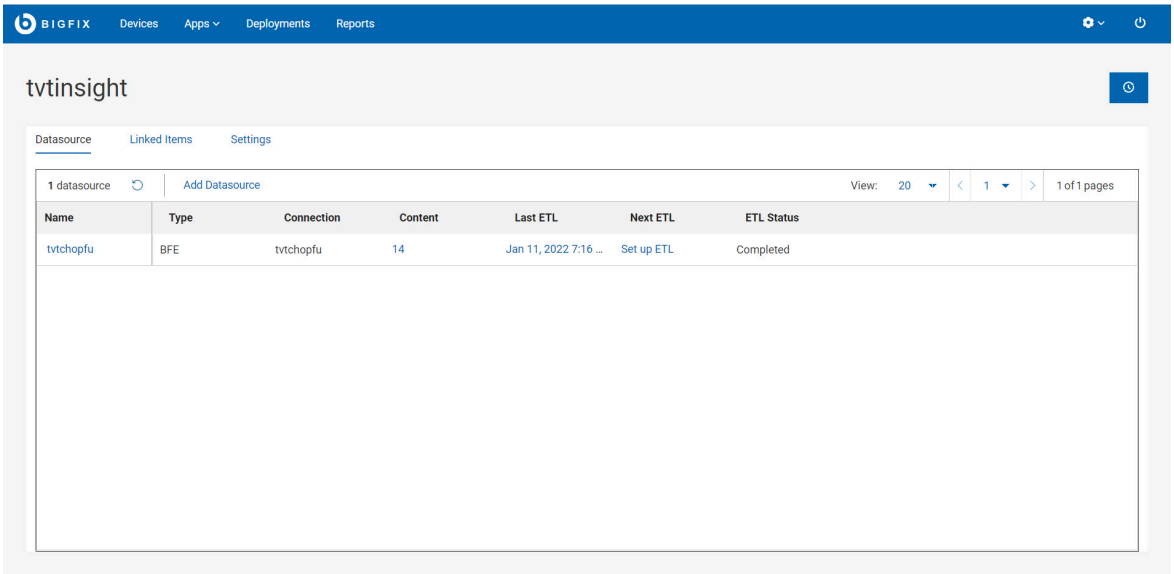


Editing the sites in a datasource

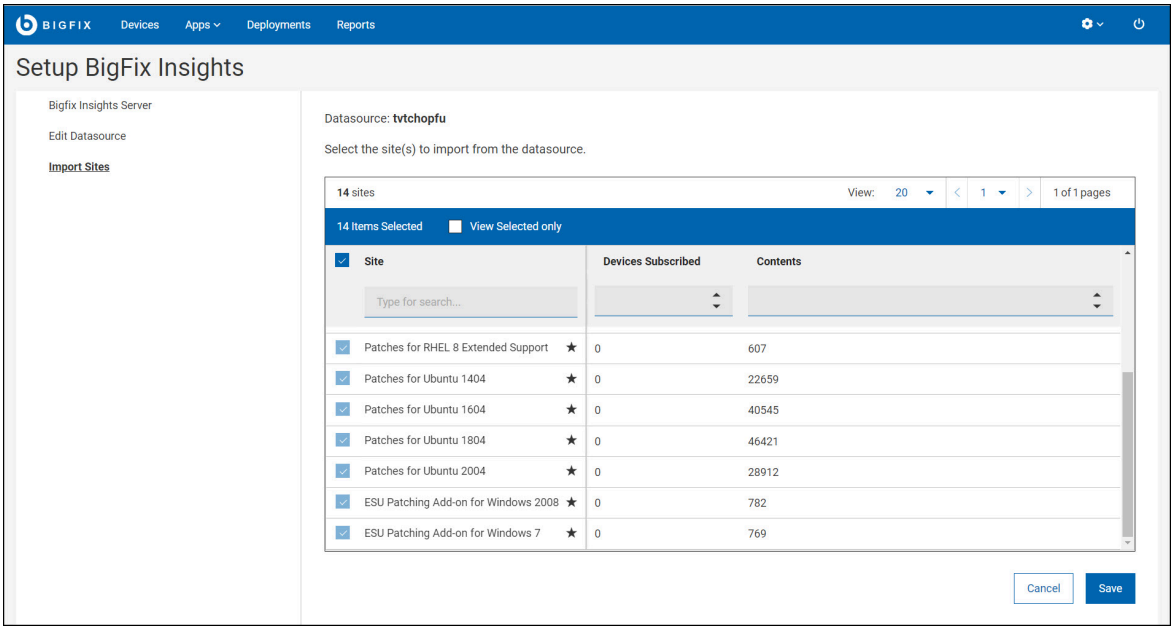
1. Click the **Datasources** tab in the BigFix Insights Details page.

   Note: You cannot edit the sites in a datasource when the **ETL Status** is **Running**.

2. To change the imported sites, click the **Content** link of a desired datasource. The sites link is represented as a number hyperlink within the sites column. This value represents the number of sites that is imported from the given datasource.



3. The sites within the datasource are listed. This represents that all the sites are available for import from the given datasource.



Note: A star next to a site name indicates that it is a *primary* site. You cannot remove a site that is defined as a primary site. A primary site represents

the site that other databases are correlated to. The primary site must be reassigned to another database with the same site prior to removal.

4. Include or exclude the site(s) from the import by checking or clearing the check box.
5. Click **Save**.

You have successfully updated the imported sites of a datasource.

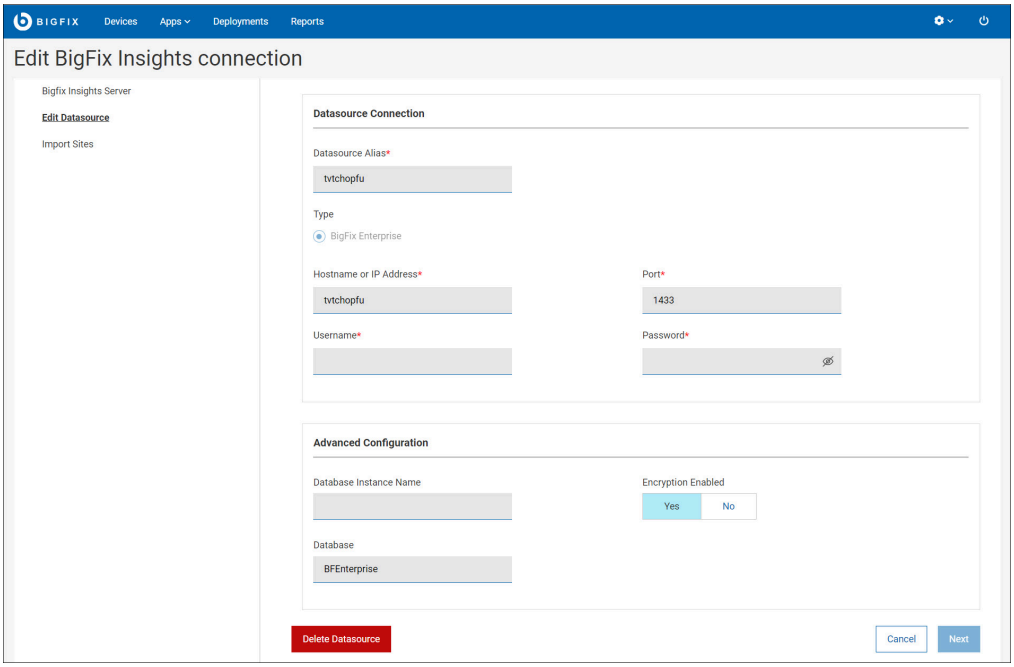
Editing a datasource

Do the following steps to edit an existing datasource.

1. Within the Insights app, navigate to the **Datasources** tab. All the datasources that you have created are listed.

Note: You cannot edit the sites in a datasource when the **ETL Status** is **Running**.

2. Click the **Alias** of the datasource you want to edit.



3. Edit the fields as required.
4. Click **Next**.
   The datasource is updated.

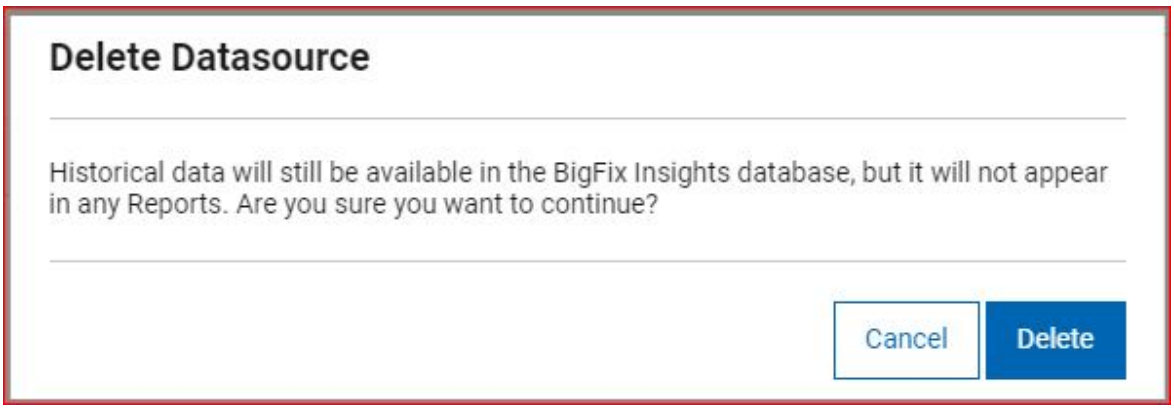
Deleting a Datasource

This task describes how to delete a datasource.

Note:

- A datasource cannot be deleted if it has one or more primary sites. To change the primary datasource for a site, see Working with Linked Items *(on page 10)*.
- You cannot edit the sites in a datasource when the **ETL Status** is **Running**.

1. Within the Insights application, navigate to the **Datasources** tab.
2. Click the **Datasource Alias** you want to delete.
   The **Edit Datasource** page is displayed
3. Click **Delete Datasource**.
   The **Delete Datasource** message box is displayed.



4. Click **Delete** to delete the datasource.

Chapter 4. Working with Linked Items

Learn more about Linked Items in BigFix 10 Insights.

A Linked Item indicates a mapping of the same content across multiple datasources to prevent its duplication in BigFix Insights. In the case of sites, it indicates that certain sites between different datasources are the same, and so there might be content between the sites that are the same. To know more about datasources, see Working with Datasources and Sites *(on page 3)*.

In the mappings within a Linked Item, there is always one datasource that is set as *Primary*. The primary linked item is a datasource whose represented objects (for example, an external site) are used as a single record to correlate to. The primary linked item determines when there is a difference between the content of mapped represented objects and what record is going to be chosen as the representation of the content. By default, the first datasource to import the represented object is set as its primary. You can change the primary datasource for a represented object under the **Linked Items** tab.

Example:

The following example describes how a primary datasource affects BigFix Insights.

Assume there are two datasources, Datasource West and Datasource East with the same external site, BES Support. On Datasource East, BES Support has a slightly older version of the site and has 20 fixlets. On Datasource West, BES Support has the newest version of the site and has 30 fixlets. Between the both of them, they have 20 fixlets that are the same.

If Datasource West's BES Support is set as the primary site and the ETL runs, BigFix Insights shows a total of 30 fixlets represented instead of 50 because BigFix Insights maps the 20 shared fixlets between the linked sites and adds all 10 of Datasource West's BES Support's unique fixlets. Inversely, if Datasource East's BES Support is initially set as the primary, BigFix Insights shows only 20 shared fixlets represented between the datasources.

If you switch the primary datasource from Datasource East to Datasource West after an ETL has run, all the unique BES Support fixlets from Datasource West are added upon the next ETL, therefore there are 30 fixlets represented. If you switch the primary datasource from Datasource West to Datasource East after an ETL has run, BigFix Insights still shows a

total of 30 fixlets; however, only the 20 shared fixlets are updated and Datasource West's 10 unique BES Support fixlets are not updated.

To navigate to **Linked Items Overview**, click the **Linked Items** tab in the BigFix Insights Details page. BigFix Insights automatically creates linked items for the sites that are found in the datasources.

The **Linked Items Overview** contains the following columns:

Table 1. Linked Items Overview

Column Name	Description
Link Alias	The alias of the linked item. Defaults to the site name.
Type	The type of the site included in the data-source, which could be Internal or Exter-nal
Datasources	The number of datasources for this linked item.
Primary	The primary datasource of the linked item.
Last Updated	The timeline at which the property of the linked item was last updated.

Editing Link item

Do the following steps to edit the link alias and update the primary datasource for a linked item:

1. Within the Insights app, navigate to the **Linked Items** tab.
2. Click the item in the Link Alias column.
3. Edit the **Link Alias** name.
4. Select the **Primary Linked Datasource** from the list.

The Primary Linked Datasource is used to map content for all the non-primary datasources; hence choose the datasource that is a preferred source of data.

5. Click **Save**.

Chapter 5. Scheduling an ETL

The ETL (Extract, Transform, Load) process pulls data from a datasource and stores it in the Insights' database. An ETL process may consume significant time and resources and hence it is run on a customizable schedule to minimize disruption. In BigFix Insights, you can schedule an ETL daily, weekly, or monthly and also determine the history of an ETL for a datasource and a set of ETLs for all datasources.

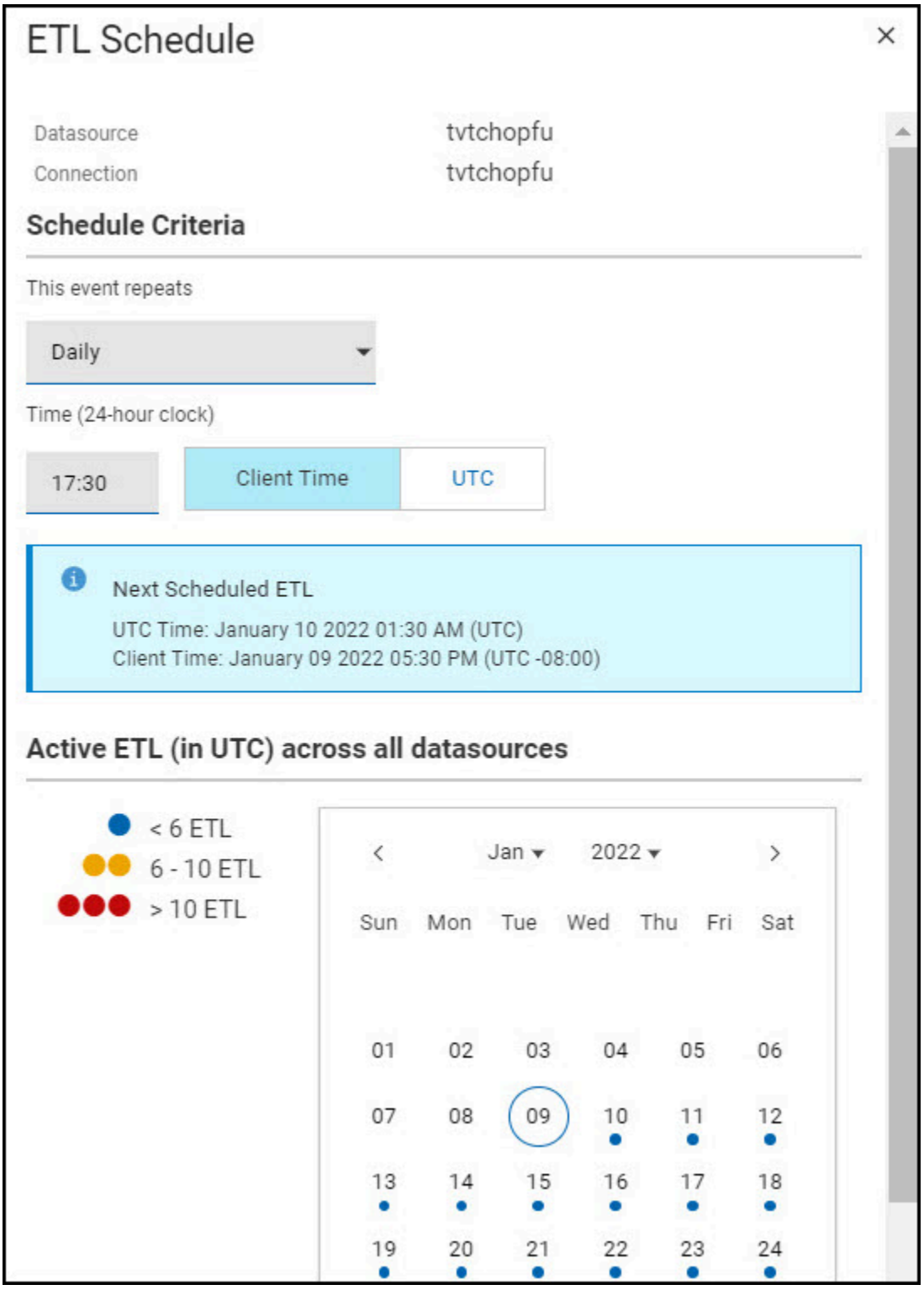
Setting up ETL

Note: To minimize disruption in your deployments, follow the recommendations provided in the capacity guidelines.

Perform the following steps to schedule an ETL in Insights:

1. Navigate to the **Datasources** tab in Insights.

2. Click **Set up ETL** for the datasource for which you want to schedule an ETL.
   The ETL scheduler is displayed:


ETL Schedule
Datasource
tvtchopfu
Connection
tvtchopfu
Schedule Criteria
This event repeats
Daily
Time (24-hour clock)
17:30
Client Time
UTC
Next Scheduled ETL
UTC Time: January 10 2022 01:30 AM (UTC)
Client Time: January 09 2022 05:30 PM (UTC -08:00)
Active ETL (in UTC) across all datasources
< 6 ETL
6 - 10 ETL
> 10 ETL
Jan
2022
Sun Mon Tue Wed Thu Fri Sat
01 02 03 04 05 06
07 08 09 10 11 12
13 14 15 16 17 18
19 20 21 22 23 24

Note: Client Time refers to the time on the datasource. The circled date in the calendar indicates the current day.

The ETL calendar has dates with dots of colors Green, Blue, or Red.

- **Blue**: Indicates that less than 6 ETLs are scheduled for that current day.
- **Green**: Indicates that 6-10 ETLs are scheduled for that current day.
- **Red**: Indicates that more than 10 ETLs are scheduled for that current day.

3. Select the frequency at which you want to run the ETL and enter values for the other fields.

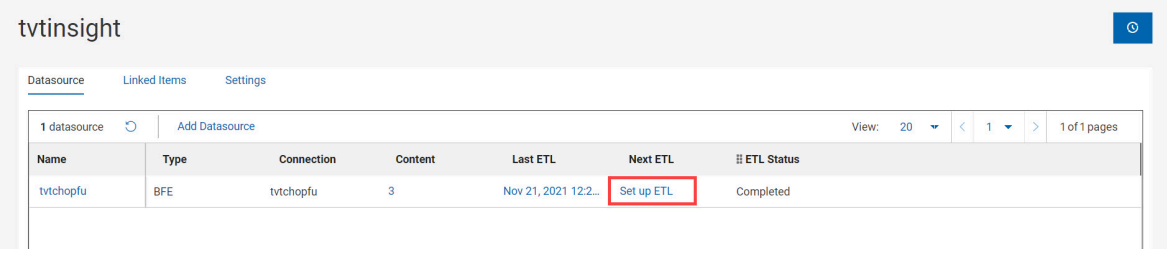
Note:

- Click **Queue ETL Now** to run an ETL now. If another ETL is in progress, **Queue ETL Now** adds the ETL to the queue.
- For a given datasource, if an ETL initiated by **Queue ETL Now** is running when a scheduled ETL is expected to run, the scheduled ETL is skipped and will run at its next scheduled ETL time.

4. Select the time when you want to run the ETL. You can change the meridiem by clicking the meridiem.
5. Click **Save**.

   You have successfully scheduled an ETL in Insights.

   You can view the next ETL event from the **Next ETL** column in the Insights Details page:



Rescheduling an ETL

Perform the following steps to reschedule an ETL:

1. Navigate to the **Datasource** tab.
2. Click the date in the **Next ETL** column for the datasource for which you want to edit the ETL Schedule.
3. Edit the fields as required.
4. Click **Save**.
   The **Next ETL** column for the datasource is updated with the updated schedule.

Note: You cannot edit an ETL schedule when the **ETL Status** is **Running**.

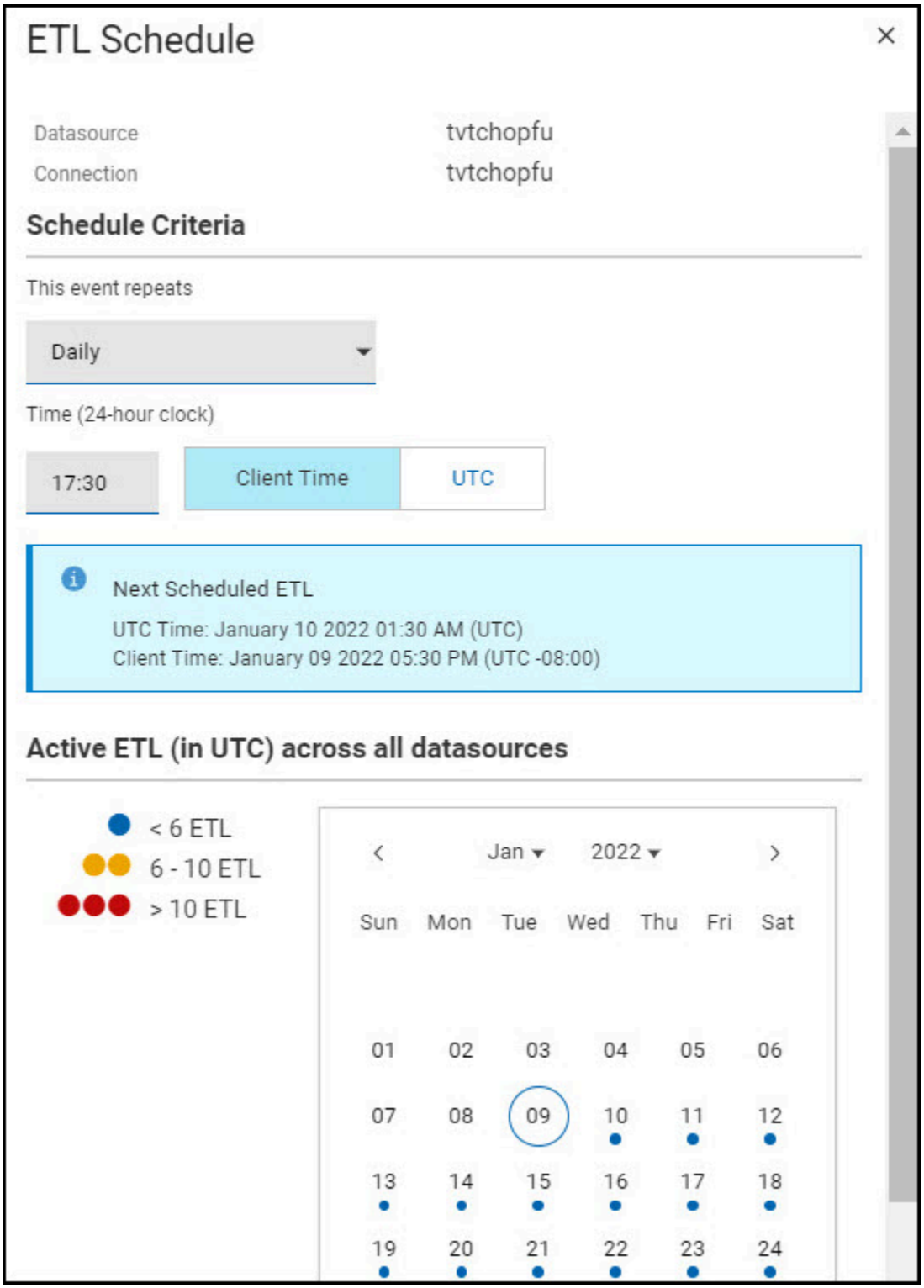
ETL History

You can view the ETL history by clicking the **Last ETL** column on the Insights Details page (in the **Datasource** tab). The history displays all the completed runs for the selected ETL. The circled date in the calendar indicates the present day and all the dates that are highlighted are days that the ETL has ran.

Note: You can view the ETL history only after a successful ETL.

To display the duration, status, start time and end time for the ETL that ran on that date, click on a highlighted date in the ETL History calendar.

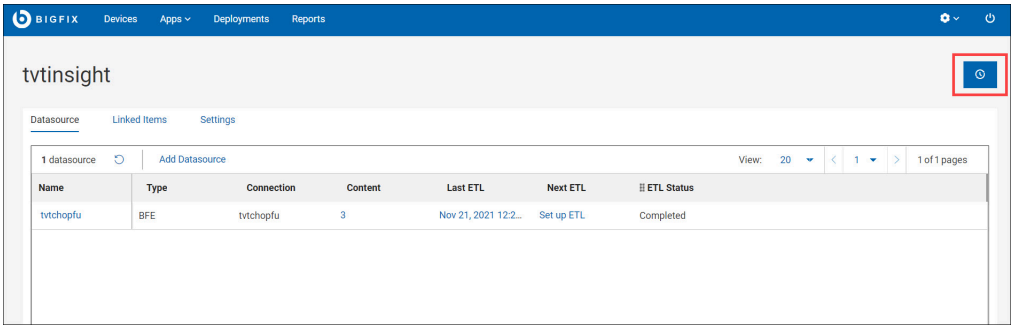
To refresh the Last Successful ETL and Next ETL fields with the latest updates, click the Refresh icon. Clicking the Next ETL link brings you to the ETL Scheduler for that datasource.


ETL Schedule
Datasource
tvtchopfu
Connection
tvtchopfu
Schedule Criteria
This event repeats
Daily
Time (24-hour clock)
17:30
Client Time
UTC
Next Scheduled ETL
UTC Time: January 10 2022 01:30 AM (UTC)
Client Time: January 09 2022 05:30 PM (UTC -08:00)
Active ETL (in UTC) across all datasources
< 6 ETL
6 - 10 ETL
> 10 ETL
Jan
2022
Sun Mon Tue Wed Thu Fri Sat
01 02 03 04 05 06
07 08 09 10 11 12
13 14 15 16 17 18
19 20 21 22 23 24

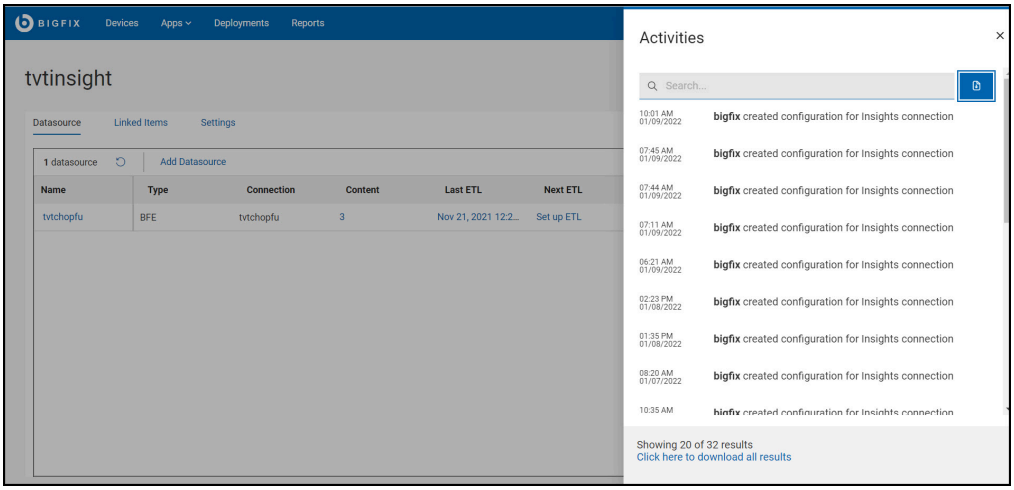
Chapter 6. Working with activities history

This topic describes how to view the activities history and export the logs.

1. Within the Insights app, navigate to the **Datasource** tab.
2. Click the **Activities** icon placed below the gear icon.



The **Activities** window displays a log of all the activities tracked by Insights along with their timestamp in UTC format.



Note: When you delete a datasource, all the activities related to the datasource are in read-only mode. They can no longer be linked to the datasource.

3. To perform more actions or to know more about a particular item, click any hyperlink in the log list.

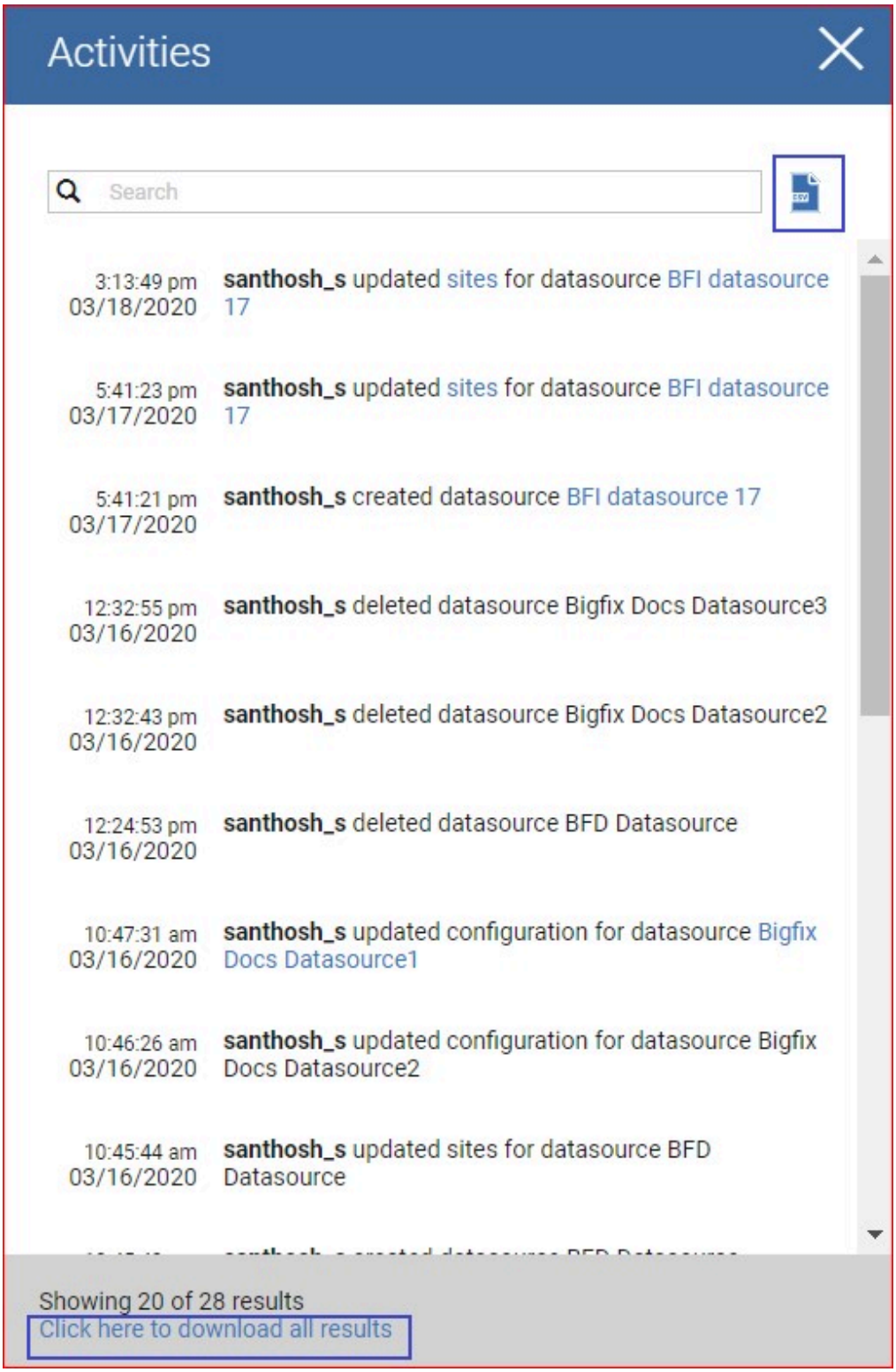
The following actions are possible:

- **ETL schedule** - Opens the datasource's ETL Schedule
- **Sites** - Opens the datasource's sites page
- **Datasource's alias** - Opens the datasource's edit page
- **Linked item's alias** - Opens the linked item's edit page
- **BigFix Insights' alias** - Opens the BigFix Insights connection details page

To filter entries in the **Activities** window, use the **Search** option.

4. To export the results, click **Export to CSV** that is next to the search bar.
   You can also export the results by clicking **Click here to download all results** located at the bottom of the **Activities** window.

Note: This option is displayed only when there are more than 20 search results.


Activities
Search
3:13:49 pm 03/18/2020 santhosh_s updated sites for datasource BFI datasource 17
5:41:23 pm 03/17/2020 santhosh_s updated sites for datasource BFI datasource 17
5:41:21 pm 03/17/2020 santhosh_s created datasource BFI datasource 17
12:32:55 pm 03/16/2020 santhosh_s deleted datasource Bigfix Docs Datasource3
12:32:43 pm 03/16/2020 santhosh_s deleted datasource Bigfix Docs Datasource2
12:24:53 pm 03/16/2020 santhosh_s deleted datasource BFD Datasource
10:47:31 am 03/16/2020 santhosh_s updated configuration for datasource Bigfix Docs Datasource1
10:46:26 am 03/16/2020 santhosh_s updated configuration for datasource Bigfix Docs Datasource2
10:45:44 am 03/16/2020 santhosh_s updated sites for datasource BFD Datasource
Showing 20 of 28 results
Click here to download all results

Chapter 7. Updating WebUI

Learn how to update the BigFix Insights WebUI application.

The update procedure for BigFix Insights is dependent on the WebUI application updates.

Complete the following steps to update the BigFix Insights WebUI application:

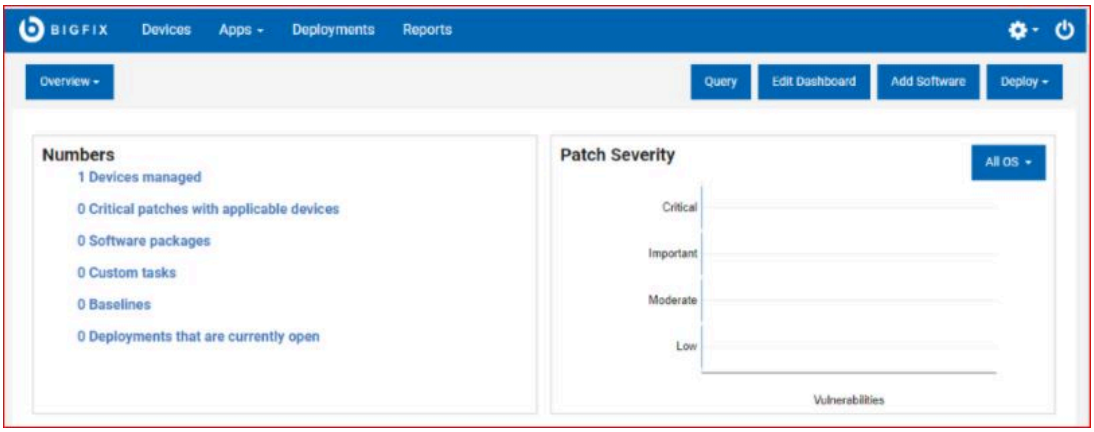
1. Navigate to the **Application Updates** page in WebUI.

   Note: You need to have Master Operator credentials.

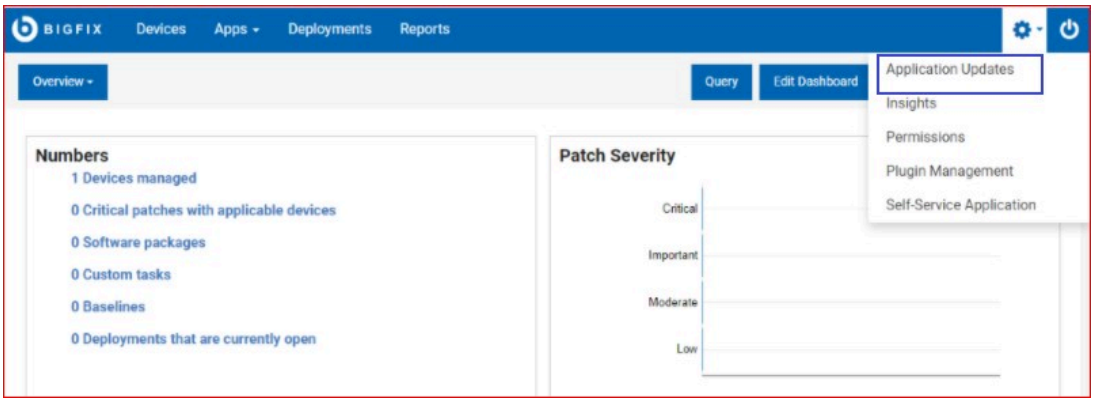
   Do the following to navigate to Application Updates page in WebUI:

   a. Log in into WebUI by using your Master Operator credentials.

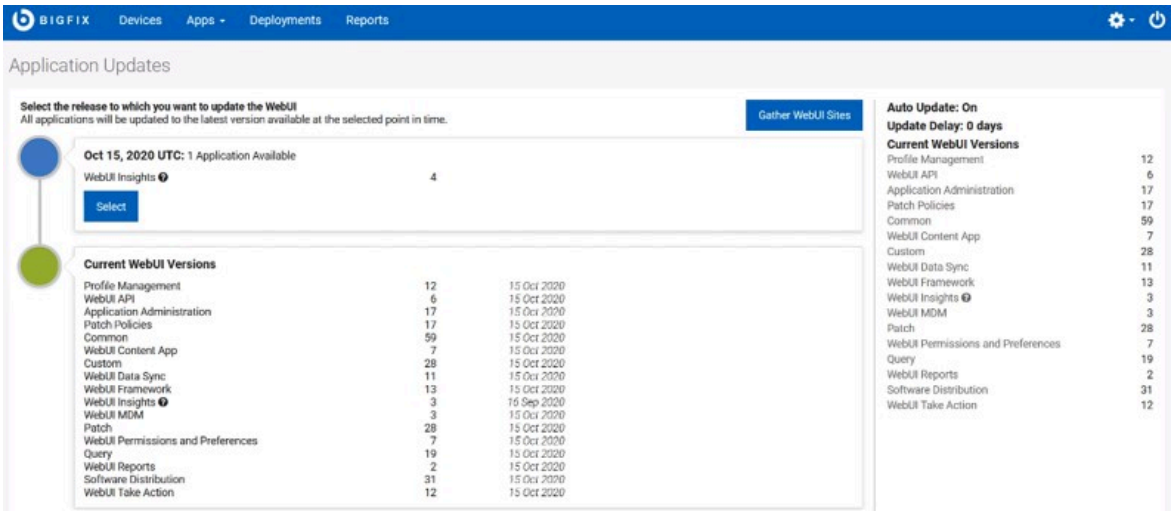
      The WebUI home page is displayed.



   b. Click the gear icon in the navigation bar and select **Application Updates**.

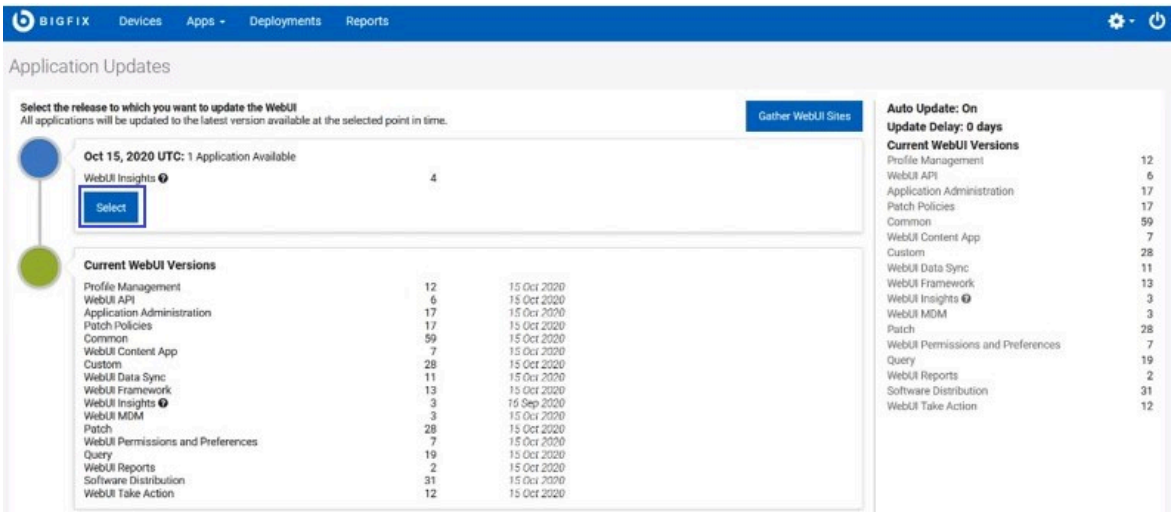


The **Application Updates** page is displayed.



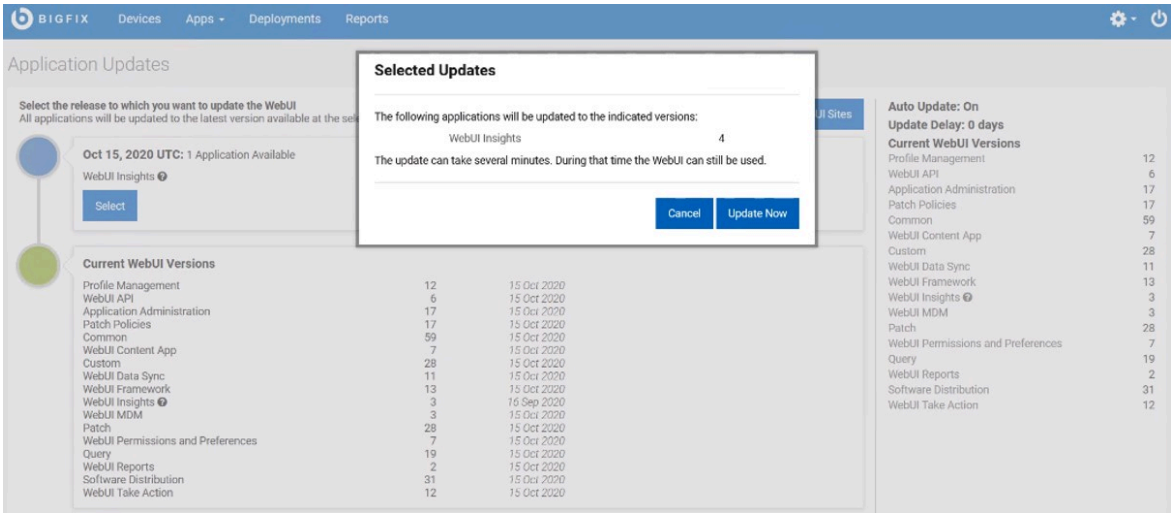
The **Application Updates** page notifies you about the available application updates. The version number of the available update in the update notification is different (and higher) than the current WebUI versions that are displayed under the **Current WebUI Versions** pane.

2. Click **Select**.



The **Selected Updates** dialog opens.

3. Click **Update Now** to confirm the update or click **Cancel** to exit the update.



The update begins and is completed in the background. You can use the current WebUI interface during the update. After the update is completed, you will be using the latest version when you log in to the WebUI interface the next time.

Chapter 8. Power BI Reports

Microsoft Power BI Desktop is a tool that helps you present the data that BigFix generates in a meaningful and comprehensible way. You can add an existing Power BI Report to your BigFix Insights implementation.

The following tasks are the prerequisites for generating data for Power BI Reports:

- Download the no-cost Power BI Desktop package from the Microsoft website and install Power BI Desktop with access to the BigFix Insights database.

  The instructions below are based on the July 2021 Power BI desktop version.
- Set up BigFix Insights.
- Add a datasource to BigFix Insights.
- Run an successful extract, transform, and load (ETL) process.
- Download and install the following fixlets from the Console:
  - Bigfix Insights – Power BI – Patch Remediation
  - Bigfix Insights – Power BI – Device Inventory

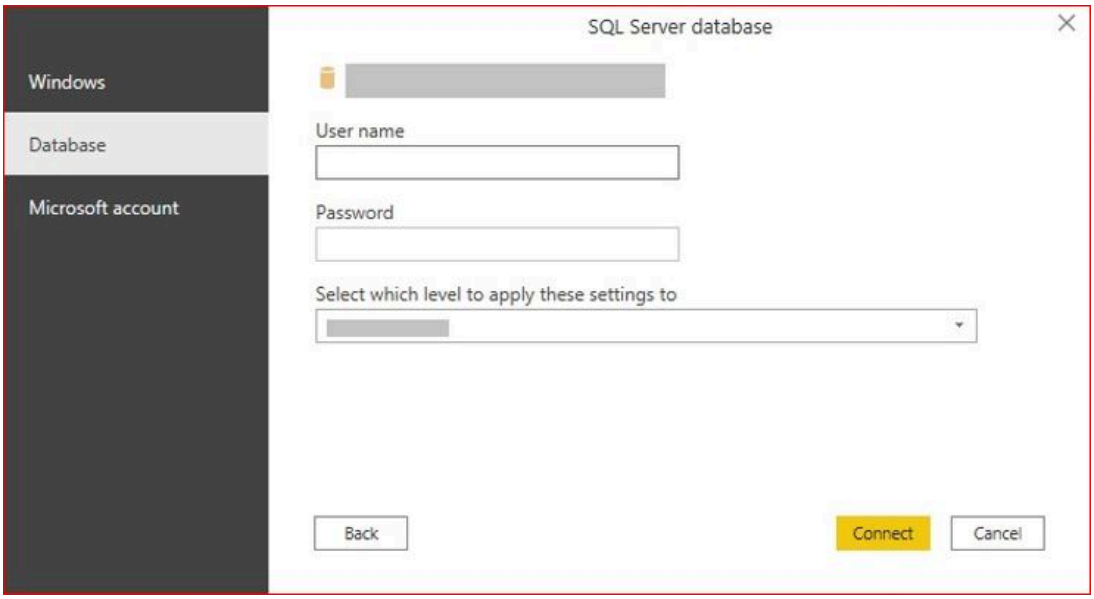
Repeat the following steps for all the Power BI Reports that HCL provides. The reports currently include Patch Remediation and Device Inventory.

Initial Configuration

1. Open one of the `*.pbix` files in Power BI Desktop.

   a. The authentication dialog box for the SQL server is displayed.

   b. Click **Database** in the left pane.
   The following dialog box is displayed.



c. Edit the following items that match your environment:

Server	Enter the IP address or the DNS name for your Microsoft SQL Server. HCL has provided all the Power BI Reports with the common DNS name of *bigfix.insights.database* for your convenience. You can either add this DNS record to your DNS servers or even add it to the local HOSTS file on the computer on which you are running the Power BI Desktop.
Database	Enter the name of the BigFix Insights database. This is the default database name used by WebUI when setting up BigFix Insights for the first time.

Username	The provided HCL Power BI Reports uses SQL authentication. You must enter the username when using SQL authentication.
Password	Enter the password when using SQL authentication.

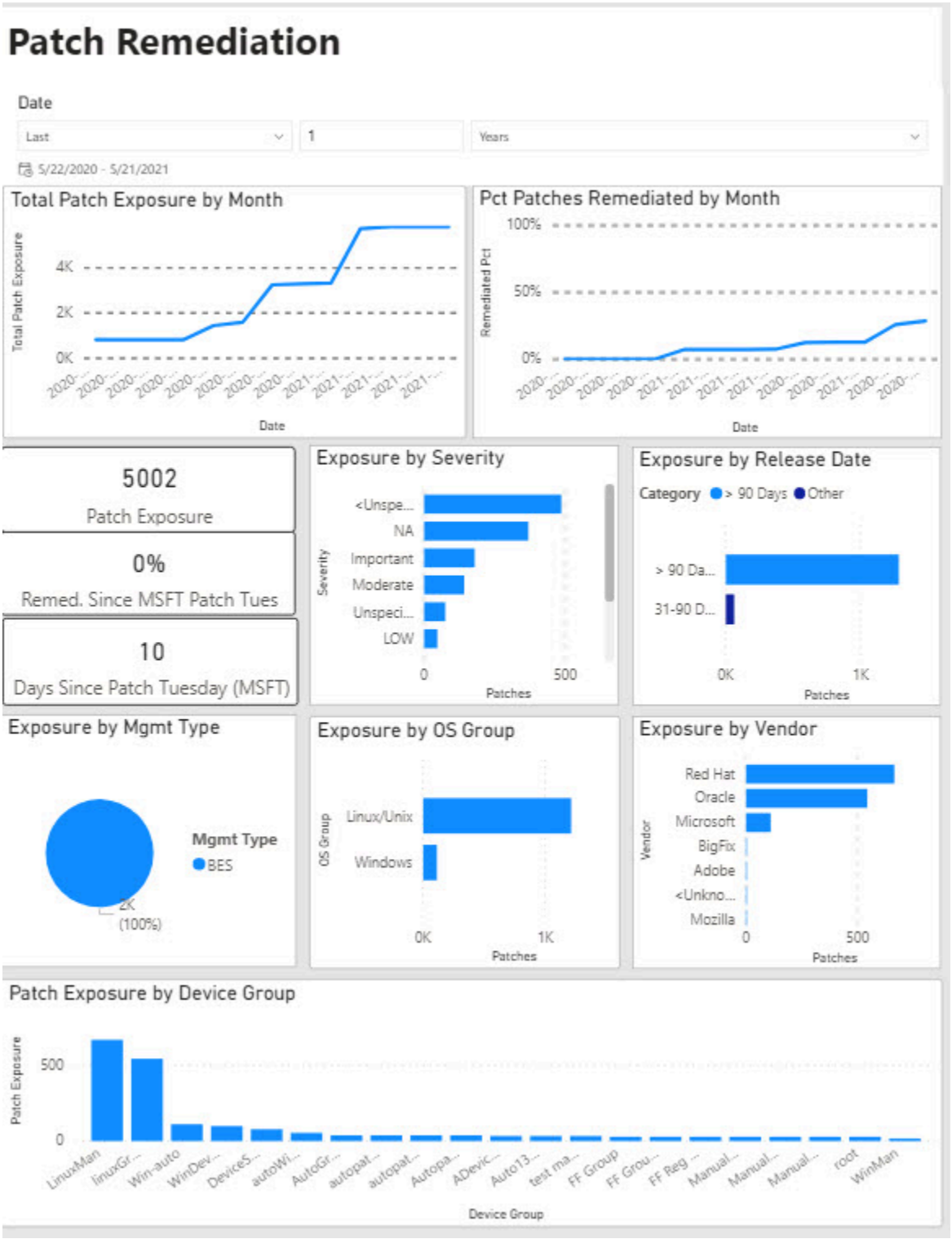
Note: If you have a Windows authentication enabled on SQL Server and the user that runs Power BI Desktop has access to the BigFix Insights database, then you can change the authentication method to **Use Windows Authentication**. When publishing your workbooks to the Tableau Server, the Tableau Server uses this account as a default to access your SQL Server. After you enter the correct data entries, the report continues to load. You are ready to publish your workbook.

2. Repeat steps 1 and 2 for each additional `*.pbix` file to view or edit.

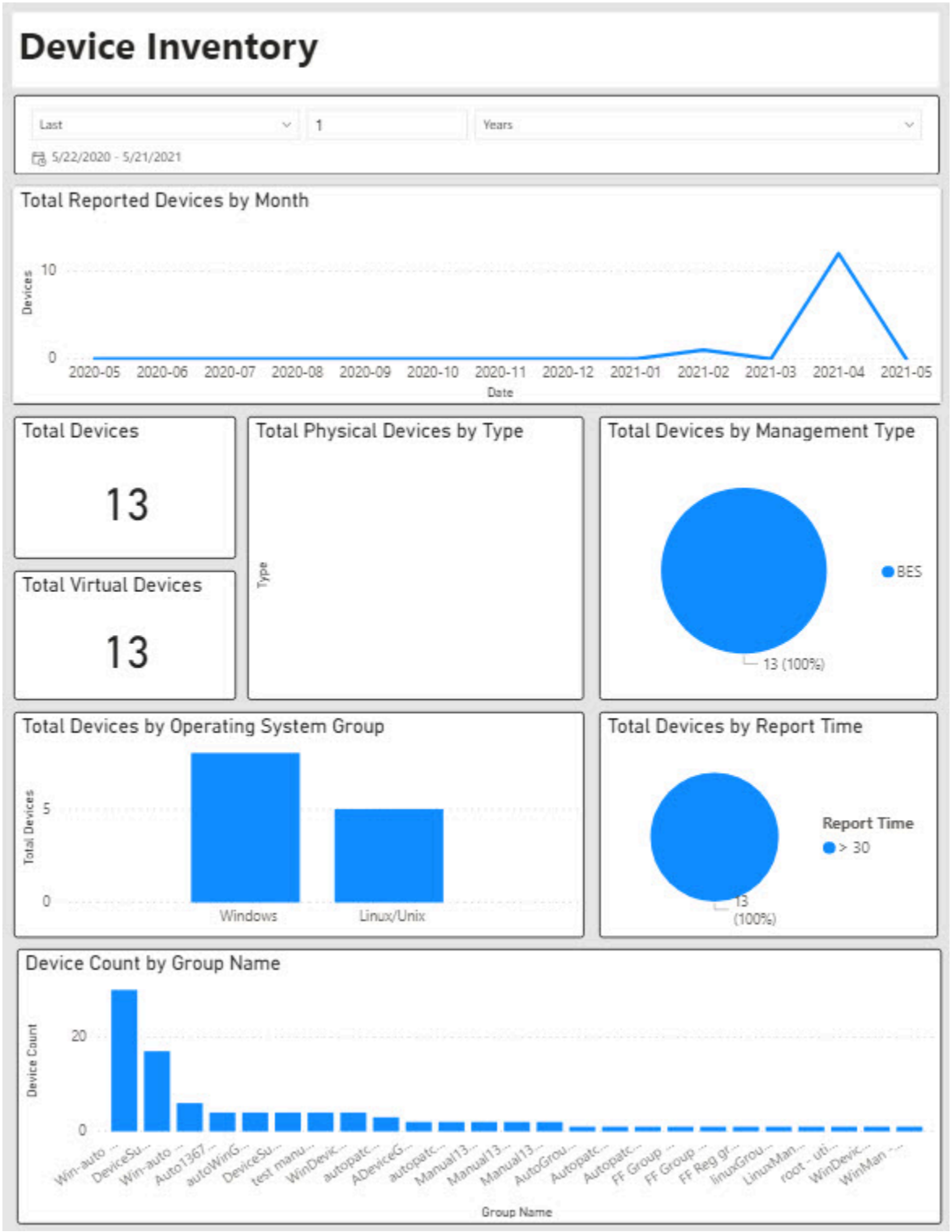
Sample Reports

Sample reports screenshots are shown below:

Patch Remdiation sample report



Device Inventory sample report


Device Inventory
Last
1
Years
5/22/2020 - 5/21/2021
Total Reported Devices by Month
Devices
10
0
2020-05
2020-06
2020-07
2020-08
2020-09
2020-10
2020-11
2020-12
2021-01
2021-02
2021-03
2021-04
2021-05
Date
Total Devices
13
Total Virtual Devices
13
Total Physical Devices by Type
Type
Total Devices by Management Type
BES
13 (100%)
Total Devices by Operating System Group
Total Devices
5
0
Windows
Linux/Unix
Total Devices by Report Time
Report Time
> 30
13
(100%)
Device Count by Group Name
Device Count
20
0
Win-auto ...
DeviceSu...
Win-auto ...
Auto1367...
autoWinG...
DeviceSu...
test manu...
WinDevic...
autopatc...
ADeviceG...
autopatc...
Manual13...
Manual13...
Manual13...
AutoGrou...
Autopatc...
Autopatc...
FF Group ...
FF Group ...
FF Reg gr...
linuxGrou...
LinuxMan...
root - uti...
WinDevic...
WinMan ...
Group Name

Chapter 9. Tableau Workbooks

Tableau Desktop and Tableau Server are powerful tools that help you present the valuable data that BigFix generates in a meaningful and ingestible way. This chapter describes how to add an existing Tableau Workbook to your BigFix Insights implementation.

The following are the prerequisites to generate Tableau Workbooks:

- Purchase appropriate licenses for Tableau. You need a Creator license to run the tasks listed in this topic.
- Tableau Desktop and Tableau Server must be at version 2019.2 or greater to support the featured utilized within the provided Workbooks
- Install the Tableau Server with access to the BigFix Insights database.
- Install the Tableau Desktop with access to the BigFix Insights database.
- Setup BigFix Insights.
- Add a datasource to BigFix Insights.
- Run an ETL and ensure that it is successful.

BigFix Insights include 4 Tableau Workbook samples:

- Patch Rhythm
- Device Inventory
- Operating System Migrations
- Deployment Progress

1. Within BigFix, take action on the appropriate Fixlet(s) located in the Insights' external site. These Fixlets download the necessary `.sql` and `.twb` files to your desired location.
    a. BigFix Insights – Tableau - PatchRhythm
    b. BigFix Insights – Tableau - DeviceInventory
    c. BigFix Insights – Tableau - Operating SystemMigrations
    d. BigFix Insights – Tableau - DeploymentProgress
2. Open Microsoft SQL Server Management Studio (SSMS) and connect to the BigFix Insights database.

3. Build the required Stored Procedures. You can build the required Stored Procedures as described below:

    a. Each sample report comes with 1 or more `.sql` files.
        - Patch Rhythm contains PatchRhythm.sql and PatchRhythm_1.sql
        - Device Inventory contains DeviceInventory.sql.
        - Operating System Migrations contains OperatingSystemMigrations.sql, OperatingSystemMigrations_1.sql, OperatingSystemMigrations_2.sql, OperatingSystemMigrations_3.sql, OperatingSystemMigrations_4.sql, and OperatingSystemMigrations_5.sql.
        - Deployment Progress contains DeploymentProgress.sql, DeploymentProgress_1.sql, and DeploymentProgress_2.sql.

    b. Open the `.sql` files in SSMS.

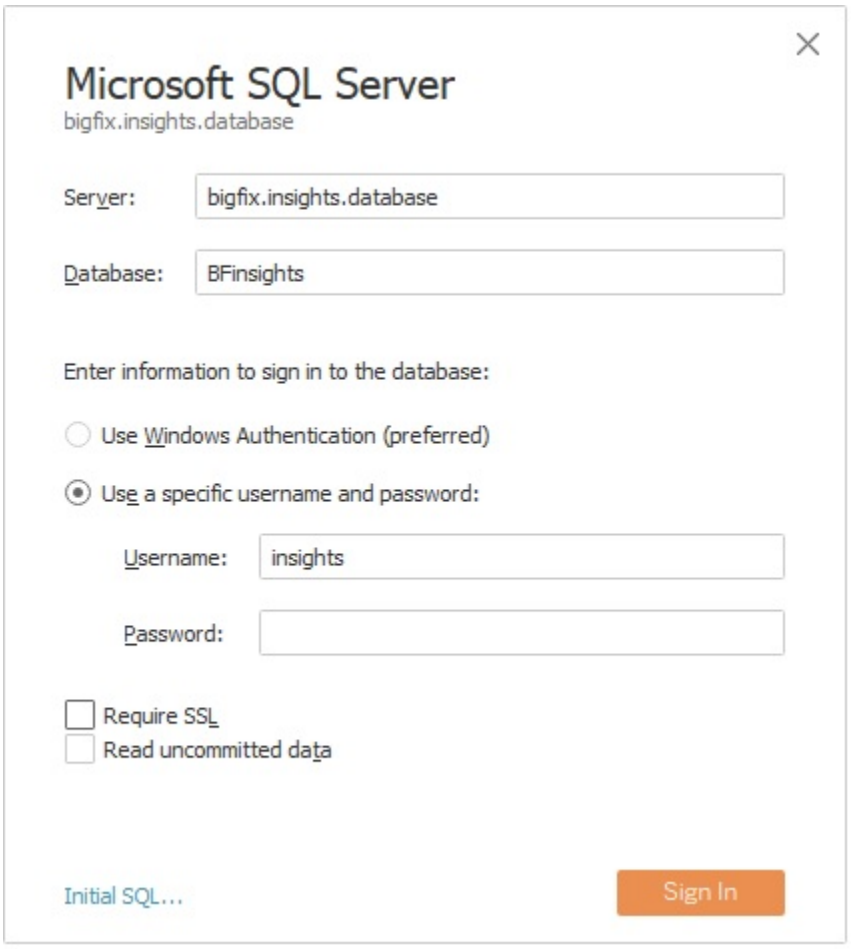
    c. Choose one of the open tabs in SSMS, select your BigFix Insights database in the **Available Databases** drop-down and click **Execute**.

    d. Repeat step 3 for each open tab.
    The Microsoft SQL Server setup is completed.
4. Open the Tableau Workbooks. Open one of the `.twb` files in **Tableau Desktop**. The authentication dialog for the SQL server is displayed:

If another dialog box is displayed, there is also an **Edit connection** link on that dialog box as well.

Now you need to edit the connection settings for the Workbook to function correctly.

a. Click **Edit connection** on the top right corner.
   The following dialog is displayed.



b. Change the following four items that match your environment:

Server

Enter the IP address or the DNS name for your Microsoft SQL Server. HCL has provided all the Tableau Workbooks with the common DNS name of *bigfix.insights.database* for your convenience. You can either add this DNS record to your DNS servers or add it to the local HOSTS file on the computer on which you are running Tableau Desktop.

Database

Enter the name of the BigFix Insights database. This is the default database name used by WebUI when setting up BigFix Insights.

Username

The Tableau Workbooks provided by HCL utilizes SQL authentication. If you are using SQL authentication, enter the username.

Password

If you are using SQL authentication, enter the password.

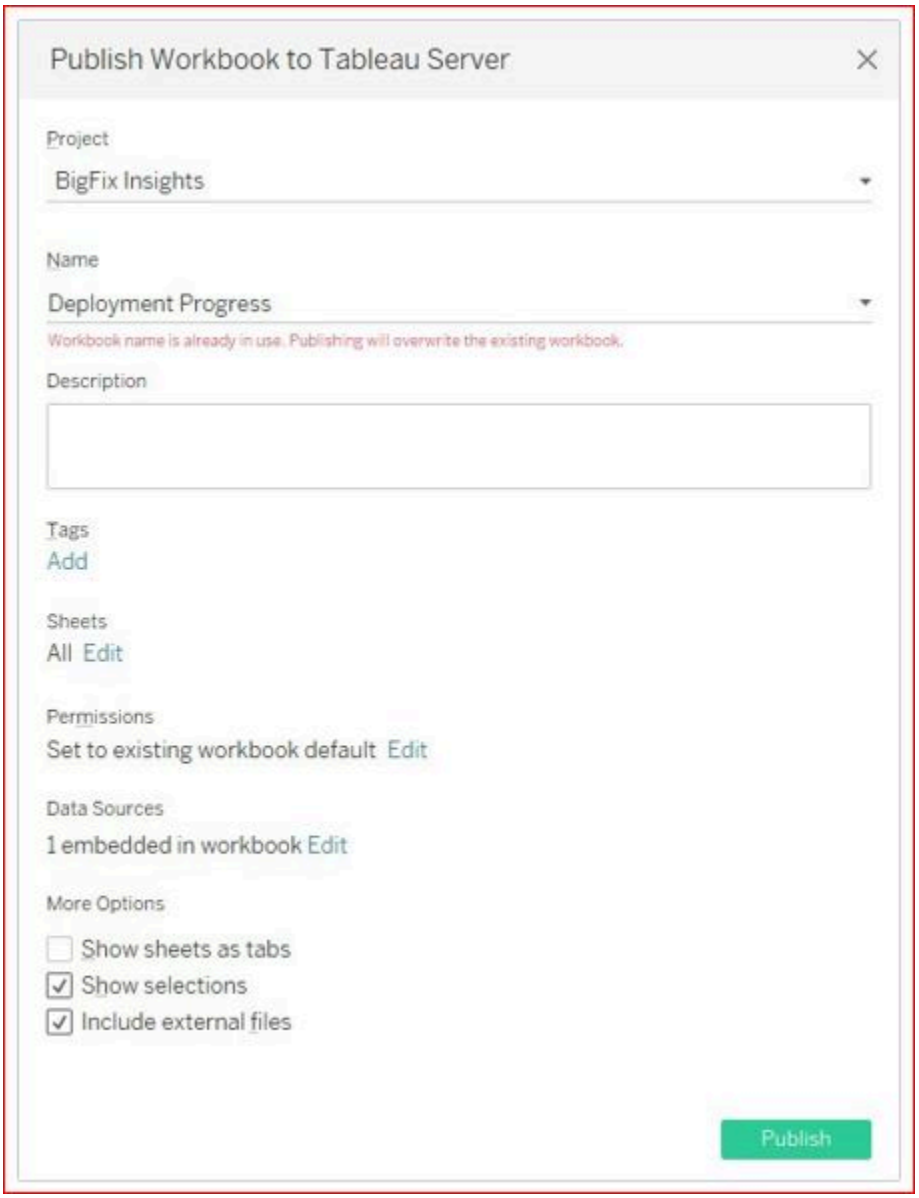
Note: If Windows Authentication is enabled on the SQL Server and the user running Tableau Desktop has access to the BigFix Insights database, then change the authentication to **Use Windows Authentication (preferred)**. When publishing your workbooks to the Tableau Server, Tableau Server uses the Windows account by default to access your SQL Server. While using the Windows Authentication and if the Tableau Desktop user account is different than the user account you are accessing the SQL database with on the Tableau Server, please change the username and password manually from within the Tableau Server after publishing the workbook.

After successfully completing the above data entry items, click **Sign In**. The report will continue to load. You are now ready to publish your workbook.

5. Execute the following steps to publish your workbook:

   a. Open your **Tableau Desktop** and click **Server > Publish Workbook**

   b. The Publish Workbook to Tableau Server dialog is displayed:



c. Select a project into which you want to publish the workbook.

d. Select a name for the workbook.

Note: If a name already exists, enter a new name. Publishing a workbook with a name that already exists prompts you to decide whether to overwrite the existing workbook.

e. Click **Publish**.

Once the workbook is published successfully, a web browser opens (you will be logged into Tableau Server automatically) to the newly published workbook for viewing.

Using the Stored Procedure samples

The Stored Procedures provided utilize a number of custom attributes that you define to suite your needs. For details, see Adding and editing custom attributes *(on page 39)*.

1. All of the Stored Procedures require a *time zone* attribute. This allows your reports to be displayed in the time zone of your choice.
   Attribute Name = time zone and Attribute Value = Pacific Standard Time. Choose your desired time zone from the list of available time zones supported by your version of Microsoft SQL Server. If a time zone is not provided, the reports defaults to UTC time zone.
2. The default severity order for the Patch Rhythm report is the number of relevant patches per severity in descending order, then by the name of the severity in alphabetical order. You could customize the sort order by creating one or more custom attributes.
   Attribute Name = Critical and Attribute Value = 1 and Category = severity_order_by. The Category is required by all applicable entries in the custom attributes table to let the Stored Procedure determine which entries to read. Critical identifies the severity provided by BigFix to assign this sort order value. The **1** represents the value the Stored Procedure will use to perform the sort. The lower the value, the higher the severity that shows up in your list of severities. You can assign the same value to multiple severities. The Stored Procedure then drops down to sorting alphabetically for those severities.
3. The Patch Rhythm report also supports renaming severities from the default name. A vendor may label their severities one way, where another vendor labels their severities in another. This custom attribute allows you to fix the issue.

Attribute Name = n/a, Attribute Value = N/A and Catetory = severity_rename. The category is required by all applicable entries in the custom attributes table to let the Stored Procedure determine which entries to read. In this example, change the label n/a to N/A for readability purposes. If this entry is not added to the custom attributes table, then you would see two rows representing n/a and N/A as separate severities. Adding this custom attribute combines these two values into a single row.

4. The Device Inventory report supports shortening long operating system names.
   Attribute Name = Win10 10.0.18362.592 (1903), Attribute Value = Win10 (1903) and Category = os_shortener. The Category is required by all applicable entries in the custom attributes table to let the Stored Procedure determine which entries to read.
5. The Device Inventory report also support grouping operating systems into appropriate Families.
   Attribute Name = Win10 (1903), Attribute Value = Windows, and Category = os_families. The Category is required by all applicable entries in the custom attributes table to let the Stored Procedure determine which entries to read. Based on this entry, the Stored Procedure assigns the shortened operating system name to a Windows grouping. You would repeat this entry for Linux, Mac OS X, and/or any other operating systems your BigFix environment supports.
6. The Operating System Migrations report also utilizes the os_shortener and os_families custom attributes.
7. The Operating System Migrations report includes the support for defining supported and non-supported operating systems. By default, this Stored Procedure looks at the Fixlets for unsupported operating systems and check applicability to determine if an operating system is supported by the vendor or not. Currently, these Fixlets are only associated with Microsoft operating system. You can define your own vendor support status utilizing the following custom attribute.
   Attribute Name = Win7, Attribute Value = 0, and and Category = os_vendor_support. The Category is required by all applicable entries in the custom attributes table to let the Stored Procedure determine which entries to read. If your operating system matches Win7, then this report displays Win7 as not supported by the vendor. This example assumes you have [os_families]Win7 defined. The day Windows 7 was listed as end-of-life, BigFix did not have a matching unsupported Fixlet so that the Stored

Procedure could mark all Windows 7 devices as not supported. Adding this custom attribute allowed the report to display all Windows 7 devices as not supported. Currently, BigFix does include the unsupported Fixlet for Windows 7, so this custom attribute for that purpose is no longer necessary. If you set the Attribute value to 1, then the operating system is marked as supported, even if there is an existing unsupported Fixlet that is applicable. This would be used if you have purchased extended support from Microsoft for Windows 7.

8. The Deployment Progress report also utilizes the os_shortener and os_families Category entries.
9. The Deployment Progress report includes the support for identifying which one of the 30+ Action Result Statuses are deemed Success, Pending, or Failure. If you do not predefine these statuses, then the Action Result is labeled as [unknown].
   To view the complete list of all possible Action Result statuses and their descriptions, view the `action_state_strings` table in the BigFix Insights database. All IDs less than 0 are automatically defined as **Failure**.
   - Attribute Name = Success, Attribute Value = 0|3|5|17, and Category = action_result_status. These Action Result Status Id's are defined by HCL as Success in this Stored Procedure. You should see four different IDs delimited by a pipe character, "|". You can modify these IDs as you see fit. Each Id represents the status a device has reported back to the BigFix server during an Action Deployment.
   - Attribute Name = Pending, Attribute Value = 1|2|6|7|8|9|10|11|12|13|14|15|16|19|20|21|22|24|25|27|28|29|30|33|34|35|36|37, and Category = action_result_status. These Action Result Status IDs are defined by HCL as Pending in this Stored Procedure.
   - Attribute Name = Failure, Attribute Value = 4|18|23|26|31|32, and Category = action_result_status. These Action Result Status IDs are defined by HCL as Failure in this Stored Procedure.

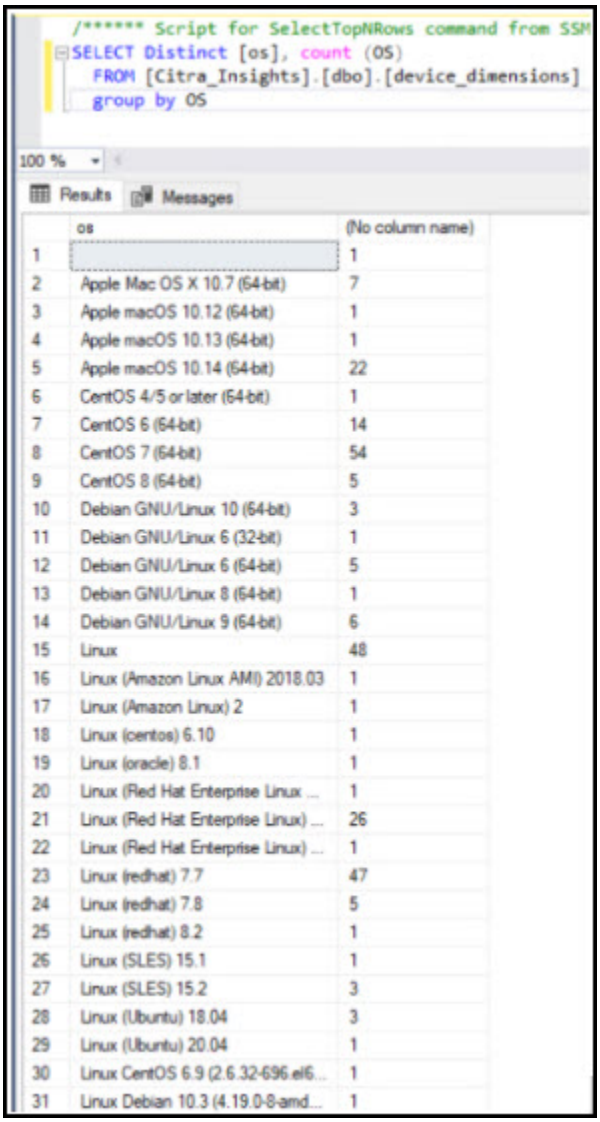
Chapter 10. Adding and editing custom attributes

This topic describes how to create, edit and delete custom attributes.

Custom attributes are used to add external information within the Insights database. The information can then be used to assist with reporting use cases. When a custom attribute is created, it can be leveraged to alias values within an Insights report. A practical use case of this functionality is provided below as an example.

Use Case: BigFix provides the literal interpretation of the operating system as installed on systems. This is reported through the operating system API and depending on the OS the resulting string may differ slightly. See the following screenshot.

Figure 2. Example



```
/****** Script for SelectTopNRows command from SSM
SELECT Distinct [os], count (OS)
  FROM [Citra_Insights].[dbo].[device_dimensions]
  group by OS
```

100 %

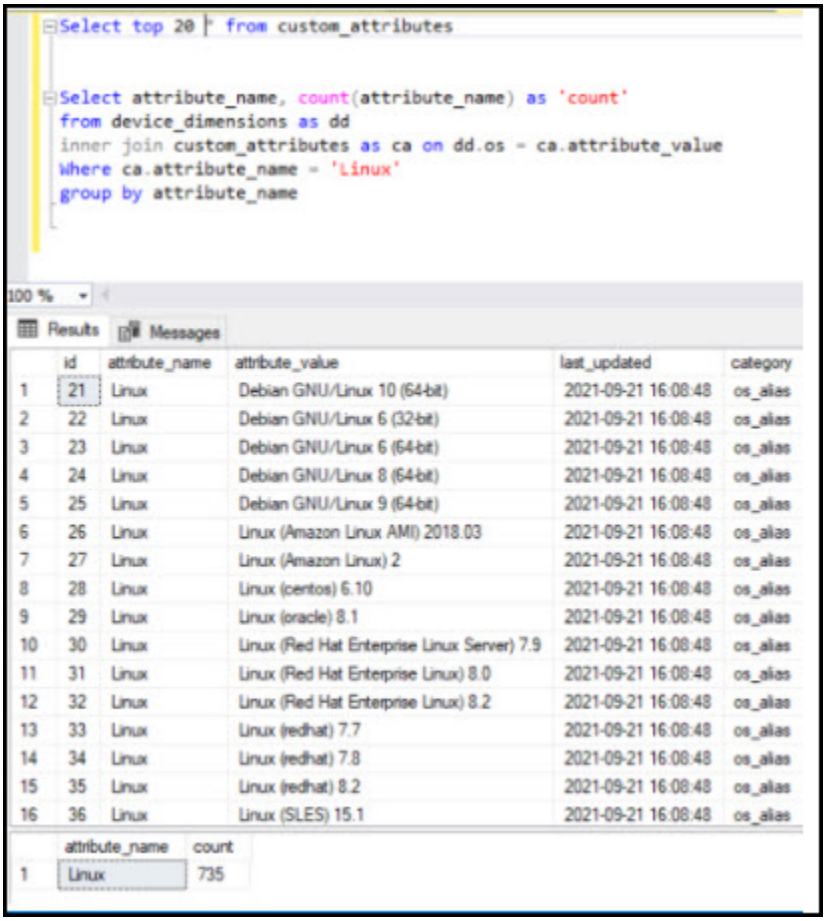
Results | Messages

	os	(No column name)
1		1
2	Apple Mac OS X 10.7 (64-bit)	7
3	Apple macOS 10.12 (64-bit)	1
4	Apple macOS 10.13 (64-bit)	1
5	Apple macOS 10.14 (64-bit)	22
6	CentOS 4/5 or later (64-bit)	1
7	CentOS 6 (64-bit)	14
8	CentOS 7 (64-bit)	54
9	CentOS 8 (64-bit)	5
10	Debian GNU/Linux 10 (64-bit)	3
11	Debian GNU/Linux 6 (32-bit)	1
12	Debian GNU/Linux 6 (64-bit)	5
13	Debian GNU/Linux 8 (64-bit)	1
14	Debian GNU/Linux 9 (64-bit)	6
15	Linux	48
16	Linux (Amazon Linux AMI) 2018.03	1
17	Linux (Amazon Linux) 2	1
18	Linux (centos) 6.10	1
19	Linux (oracle) 8.1	1
20	Linux (Red Hat Enterprise Linux ...	1
21	Linux (Red Hat Enterprise Linux) ...	26
22	Linux (Red Hat Enterprise Linux) ...	1
23	Linux (redhat) 7.7	47
24	Linux (redhat) 7.8	5
25	Linux (redhat) 8.2	1
26	Linux (SLES) 15.1	1
27	Linux (SLES) 15.2	3
28	Linux (Ubuntu) 18.04	3
29	Linux (Ubuntu) 20.04	1
30	Linux CentOS 6.9 (2.6.32-696.el6...	1
31	Linux Debian 10.3 (4.19.0-8-amd...	1

Consider the desire was to provide a rollup count for all given Linux distributions vs the count by individual Linux distributions. This is one of the many use cases that leveraging Custom Attributes can assist with. Observe the data represented below (notice there are 2 queries). The first query shows the contents of the custom_attributes table. Notice that the table has been populated with various distributions. The second query demonstrates leveraging the table to normalize and group the data accordingly. Using the two pieces

together now enables the primary use case of rolling up operating system for all Linux Distributions.

Figure 3. Use case - Example



```
Select top 20 * from custom_attributes


Select attribute_name, count(attribute_name) as 'count'
from device_dimensions as dd
inner join custom_attributes as ca on dd.os = ca.attribute_value
Where ca.attribute_name = 'Linux'
group by attribute_name
```

100 %

Results | Messages

	id	attribute_name	attribute_value	last_updated	category
1	21	Linux	Debian GNU/Linux 10 (64-bit)	2021-09-21 16:08:48	os_alias
2	22	Linux	Debian GNU/Linux 6 (32-bit)	2021-09-21 16:08:48	os_alias
3	23	Linux	Debian GNU/Linux 6 (64-bit)	2021-09-21 16:08:48	os_alias
4	24	Linux	Debian GNU/Linux 8 (64-bit)	2021-09-21 16:08:48	os_alias
5	25	Linux	Debian GNU/Linux 9 (64-bit)	2021-09-21 16:08:48	os_alias
6	26	Linux	Linux (Amazon Linux AMI) 2018.03	2021-09-21 16:08:48	os_alias
7	27	Linux	Linux (Amazon Linux) 2	2021-09-21 16:08:48	os_alias
8	28	Linux	Linux (centos) 6.10	2021-09-21 16:08:48	os_alias
9	29	Linux	Linux (oracle) 8.1	2021-09-21 16:08:48	os_alias
10	30	Linux	Linux (Red Hat Enterprise Linux Server) 7.9	2021-09-21 16:08:48	os_alias
11	31	Linux	Linux (Red Hat Enterprise Linux) 8.0	2021-09-21 16:08:48	os_alias
12	32	Linux	Linux (Red Hat Enterprise Linux) 8.2	2021-09-21 16:08:48	os_alias
13	33	Linux	Linux (redhat) 7.7	2021-09-21 16:08:48	os_alias
14	34	Linux	Linux (redhat) 7.8	2021-09-21 16:08:48	os_alias
15	35	Linux	Linux (redhat) 8.2	2021-09-21 16:08:48	os_alias
16	36	Linux	Linux (SLES) 15.1	2021-09-21 16:08:48	os_alias

	attribute_name	count
1	Linux	735

In short, the custom attributes table in combination with the specified query can be used together to normalize data for grouping in this use case. Custom_Attributes are leveraged within several of the BigFix published sample reports.

Assigning permissions to custom attributes

Perform the following steps to enable WebUI operators to add, edit, or delete custom attributes.

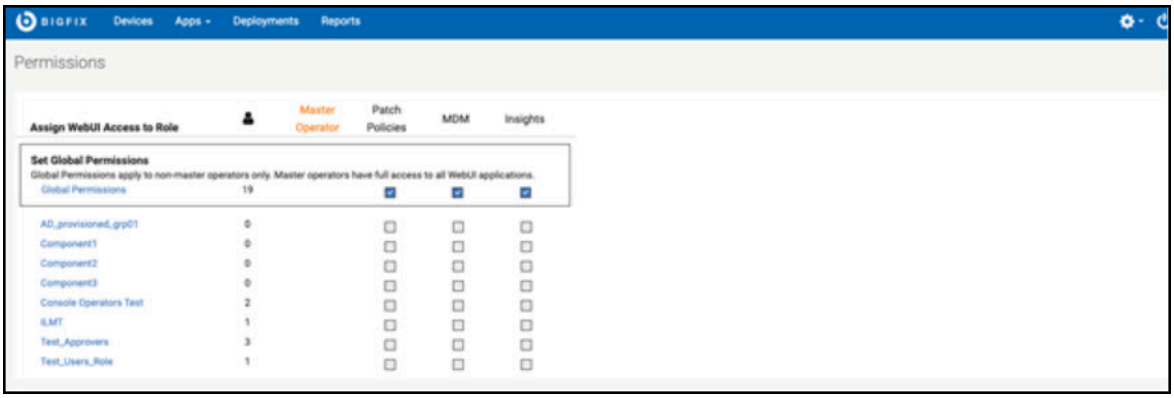
Note: The **Set Global Permissions** checkbox of the **Report Attribute: Create/Edit/Delete** should be selected to add, edit, or delete a custom attribute.

1. Log in to WebUI as a Master Operator.

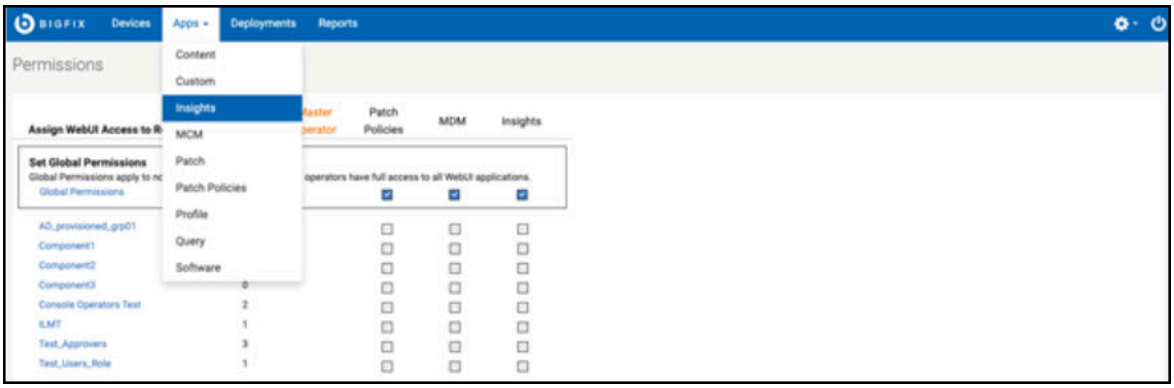
   Note: Master Operator permissions are required to assign permissions.

2. Navigate to **Settings** gear and click **Permissions**.

3. Assign the desired roles access to the Insights app.



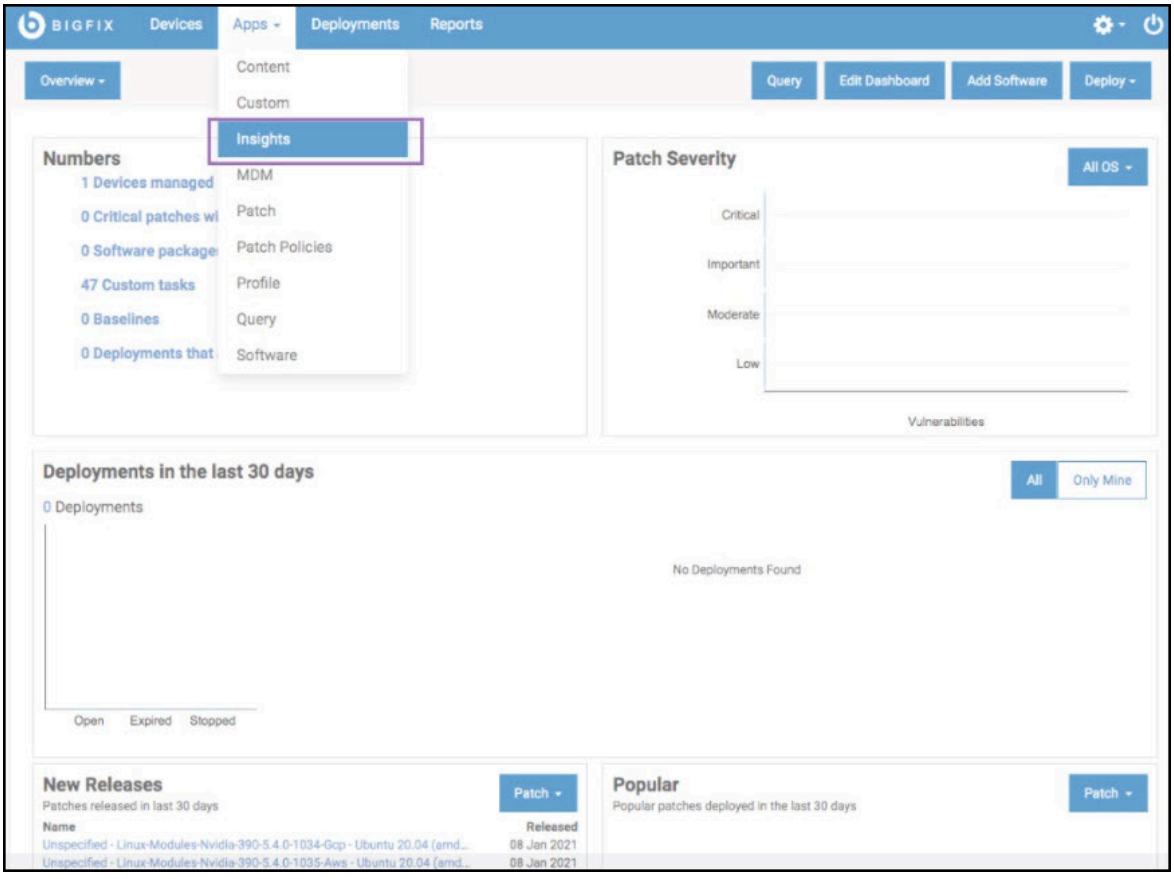
4. Once a user has been granted access to the Insights app, the user will see the Insights app in the WebUI/Apps dropdown.



Adding custom attributes

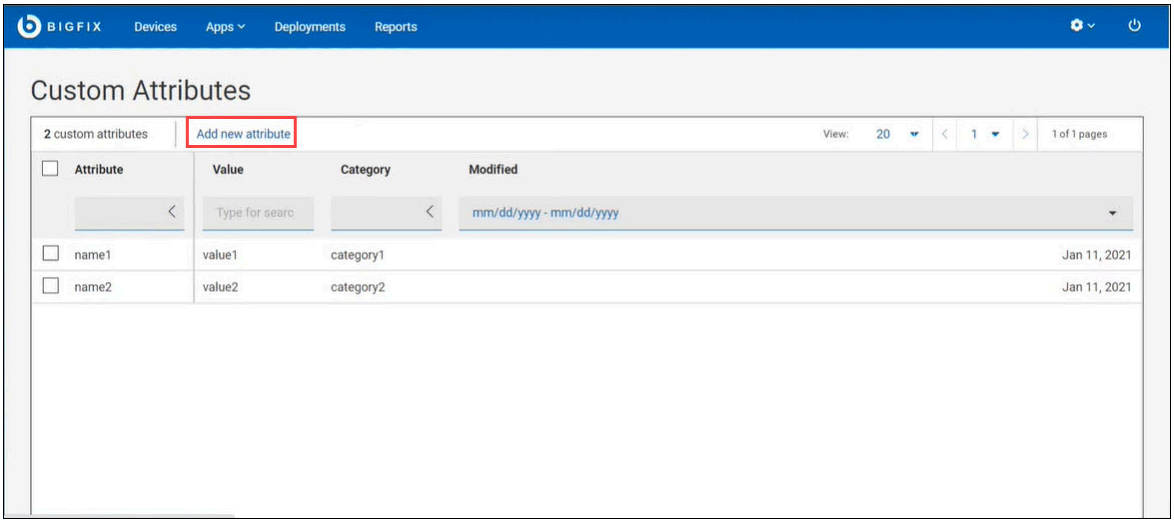
Perform the following steps to add a custom attribute:

1. Log in to WebUI.
2. Click**Apps** tab and click**Insights**.

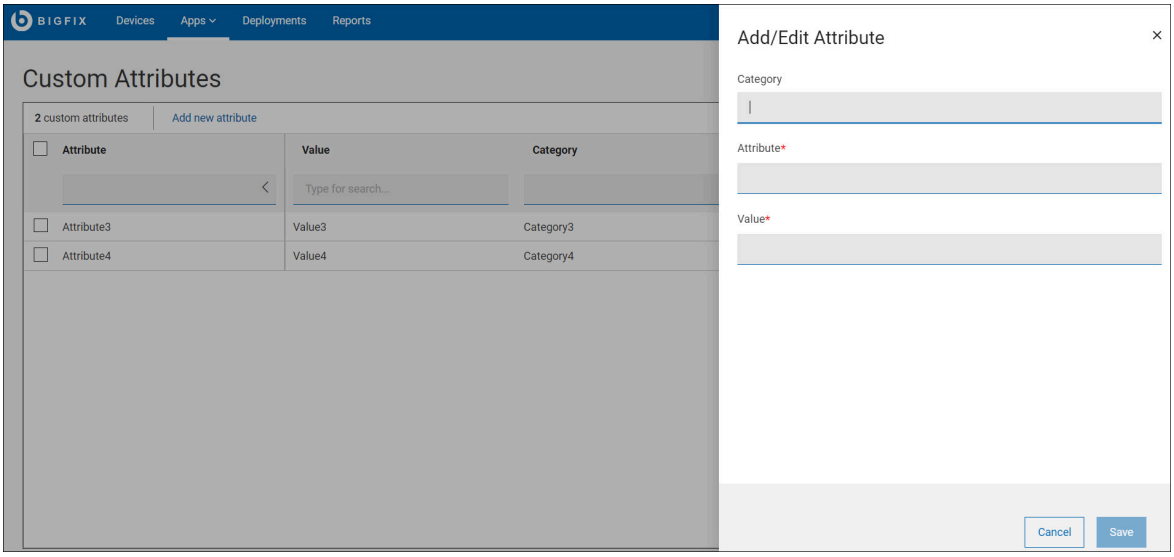


Data grid view enables you to quickly view the list of custom attributes, value, and category. Every attribute gives an option to search or filter. You can add, remove, and resize columns.

3. Click **Add new attribute** in the **Custom Attributes** page.



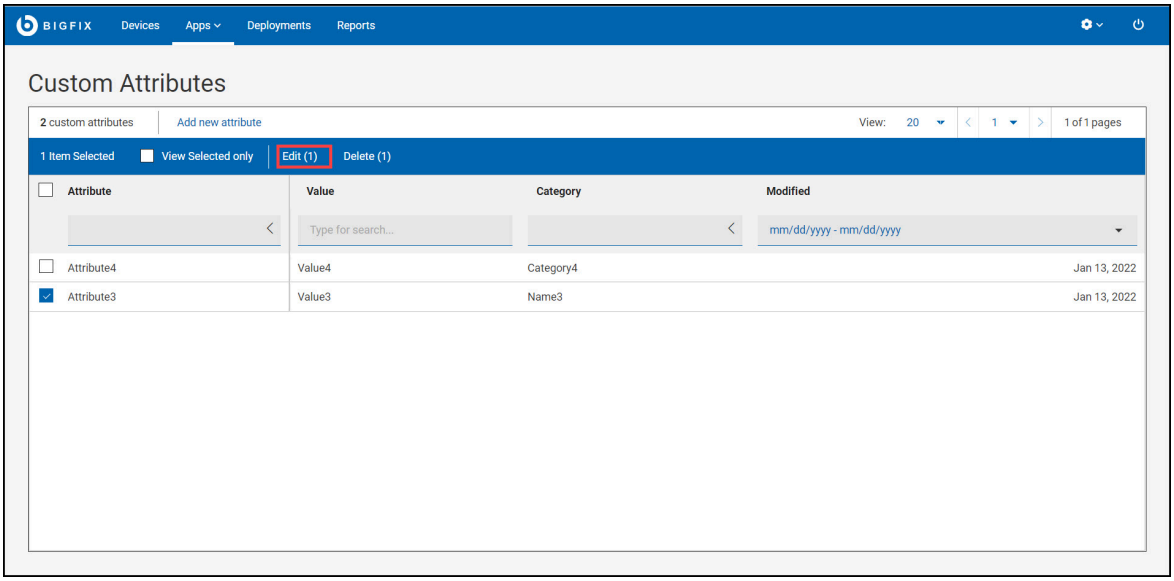
4. Enter the **Category Name**, **Attribute Name**, and **Value** in the Add/edit Attribute slide out window and click **Save**.



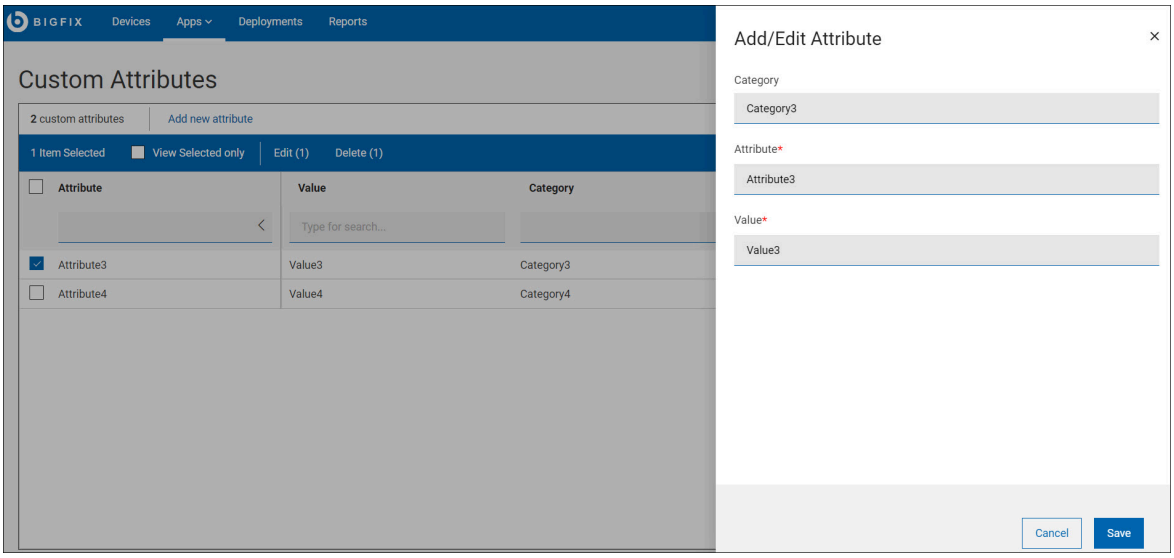
Editing custom attributes

Perform the following steps to edit a custom attribute:

1. Navigate to the Custom Attributes page in WebUI.
2. Search or select the attribute from the list that you want to edit and click the edit icon.



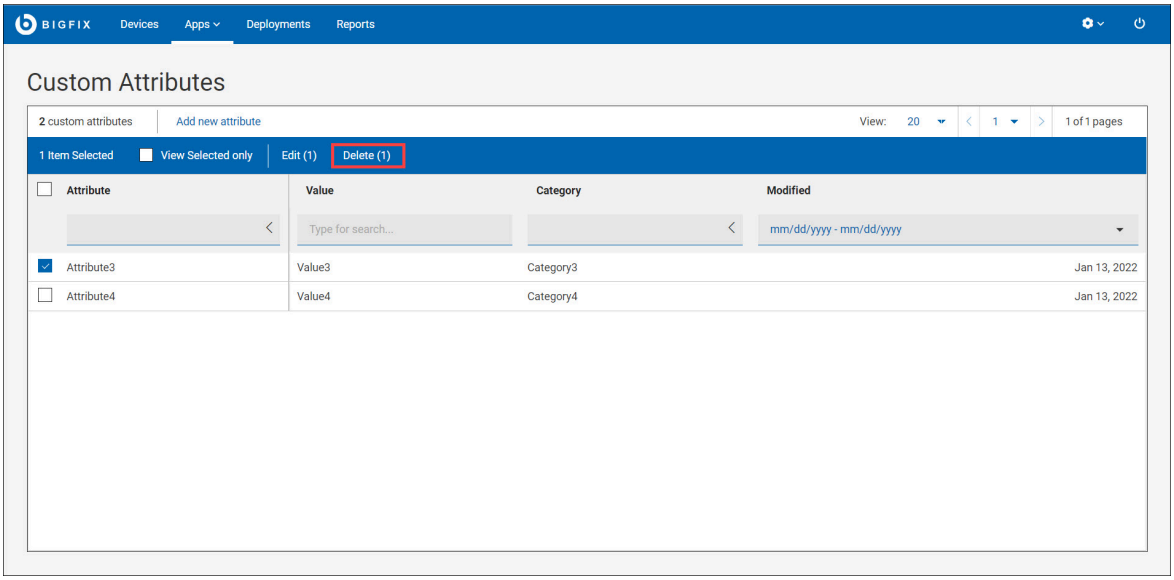
3. Edit the **Category**, **Attribute**, and **Value** and click **Save**.



Deleting custom attributes

Perform the following to delete an existing custom attribute.

1. Navigate to the **Custom Attributes** page in the WebUI.
2. Select/search the attribute(s) from the list that you want to delete and click **Delete**.



3. Once clicked, you will be prompted to confirm your deletion.

Chapter 11. Insights schema

The tables in this section provide detailed information about the BigFix 10 Insights schema.

Table 2. action_state_strings

The action_state_strings table stores the string representation of various possible action states. datasource_action_results contains a tinyint state column. To get a human-readable form of the action result state, users can refer to dbo.action_state_strings.

Name	Type	Description
state_number	int NOT NULL	The state number that is stored in `datasource_action_results`
datasource_id	int NOT NULL	The id of the datasource from which the `state_number` - `state_string` pair came
state_string	nvarchar(max) NOT NULL	The human-readable version of the state

Table 3. content_results

The content_results table stores results for all datasources associated back to datasource_sites, datasource_devices, datasource_analyses and datasource_fixlets. Fixlet and Analysis relevance results are in this table.

Name	Type	Description
datasource_site_id	bigint NOT NULL	The id of `dbo.datasource_sites`

Table 3. content_results

The content_results table stores results for all datasources associated back to datasource_sites, datasource_devices, datasource_analyses and datasource_fixlets. Fixlet and Analysis relevance results are in this table.

(continued)

Name	Type	Description
datasource_device_id	bigint NOT NULL	The id of `dbo.data-source_devices`
datasource_content_id	bigint NOT NULL	The id of `dbo.data-source_fixlet` OR `dbo.datasource_analysis` To check type column to determine which object result row refers to
type	tinyint	Type of fixlet. Valid values: 0 - Fixlet 1 - Task 2 - Baseline 3 - Analysis 4 - Computer Group
relevant	bit	Whether this content is relevant (0/1)
applicable	bit	Whether the content relevance is applicable (0/1)
resolved	bit	Whether the action is resolved (0/1)

Table 3. content_results

The content_results table stores results for all datasources associated back to datasource_sites, datasource_devices, datasource_analyses and datasource_fixlets. Fixlet and Analysis relevance results are in this table.

(continued)

Name	Type	Description
reverted	bit	Whether the action is reverted (0,1)
last_non_relevant	datetime	The last time the action was non-relevant
first_relevant	datetime	The first time the action was relevant
last_relevant	datetime	The last time the action was relevant
valid_from	datetime	The timestamp of the last ETL
valid_to	datetime	The timestamp indicating until when this row is valid

Table 4. custom_attributes

The custom_attributes table stores a set of key or value pairs that store additional data a user might need. These values are used in Tableau and PowerBI reports.

Name	Type	Description
id	int NOT NULL	The id of the name - value attribute pair
attribute_name	nvarchar(128) NOT NULL	The name of the attribute
category	nvarchar(128) NOT NULL	The value of the category

Table 4. custom_attributes

The custom_attributes table stores a set of key or value pairs that store additional data a user might need. These values are used in Tableau and PowerBI reports.

(continued)

Name	Type	Description
attribute_value	nvarchar(128) NOT NULL	The value of the attribute
last_updated	datetime2 NOT NULL	The timeline at which the attribute was last updated

Table 5. datasource_action_results

The datasource_action_results table stores the action results from all ETL'd datasources. Information is pulled from BFE actionresults table.

Name	Type	Description
datasource_device_id	bigint NOT NULL	The id of the device to which the result belongs and which references *dbo.datasource_devices.id*
datasource_action_id	bigint NOT NULL	The id of the action to the result belongs and which references *dbo.datasource_actions.id*
start_time	datetime NULL	Pulled from StartTime column in `BFE ACTIONRESULTS`
end_time	datetime NULL	Pulled from EndTime column in `BFE ACTIONRESULTS`
try_count	smallint NOT NULL	Pulled from TryCount column in `BFE ACTIONRESULTS`

Table 5. datasource_action_results

The datasource_action_results table stores the action results from all ETL'd datasources. Information is pulled from BFE actionresults table.

(continued)

Name	Type	Description
retry_count	smallint NOT NULL	pulled from RetryCount column in `BFE ACTIONRESULTS`
line_number	smallint NOT NULL	Pulled from LineNumber column in `BFE ACTION-RESULTS`
state	smallint NOT NULL	Pulled from State column in `BFE ACTIONRESULTS` and contains a numerical representation. The human-readable format is stored in *dbo.action_state_strings*
report_number	bigint NOT NULL	Pulled from ReportNumner column in `BFE ACTION-RESULTS`
exit_code	int NULL	pulled from ExitCode column in `BFE ACTIONRESULTS`
valid_from	datetime2(3) GENERATED ALWAYS AS ROW START NOT NULL	Timestamp of the last ETL
valid_to	datetime2(3) GENERATED ALWAYS AS ROW END NOT NULL	Timestamp indicating until when the row is valid

Table 6. datasource_actions

The datasource_actions table stores the actions for all ETL'd datasources,. The actions are pulled from BFE ACTIONS table.

Name	Type	Description
id	bigint IDENTITY(1,1) NOT NULL	The BigFix Insights db id of the action
remote_id	bigint NOT NULL	The id of the action object from `BFE ACTIONS` table
datasource_id	int NOT NULL	The id of the datasource the action was ETL'ed from, references dbo.data-source.id
datasource_site_id	bigint NOT NULL	The id of the site the action was ETL'ed from, references *dbo.datasource_site-.id*
type	int NOT NULL	Type of the action
deleted	tinyint NOT NULL	Whether the action has been deleted or not
parent_id	int NULL	Pulled from ParentID column in `BFE ACTIONS`
is_subscription	int NOT NULL	Pulled from IsSubscription column in `BFE ACTIONS`
is_setting	int NOT NULL	Pulled from IsSetting column in `BFE ACTIONS`
inception_time	datetime NULL	Pulled from InceptionTime column in `BFE ACTIONS`

Table 6. datasource_actions

The datasource_actions table stores the actions for all ETL'd datasources,. The actions are pulled from BFE ACTIONS table.

(continued)

Name	Type	Description
expiration_time	datetime NULL	Pulled from ExpirationTime column in `BFE ACTIONS`
stopped_at	datetime NULL	Pulled from StoppedAt column in `BFE ACTIONS`
stopped_by	int NULL	Pulled from StoppedBy column in `BFE ACTIONS`
name	nvarchar(1024) NOT NULL	Name of the action
state	tinyint NOT NULL	The state of the action, calculated by looking at the IsStopped and ExpirationTime columns from `BFE ACTIONS`: • 0 - Expired • 1 - Stopped • 2 - Open
creator_id	int NOT NULL	Pulled from CreatorID column in `BFE ACTIONS`
creation_time	datetime NOT NULL	Pulled from CreationTime column in `BFE ACTIONS`
targeting_method	tinyint NOT NULL	Pulled from TargetingMethod column in `BFE ACTIONS`

Table 6. datasource_actions

The datasource_actions table stores the actions for all ETL'd datasources,. The actions are pulled from BFE ACTIONS table.

(continued)

Name	Type	Description
targeting_relevance	nvarchar(max) NULL	Pulled from TargetingRelevance column in `BFE ACTIONS`
targeting_properties	nvarchar(max) NULL	Pulled from TargetingProperties column in `BFE ACTIONS`
source_site_url	nvarchar(1024) NULL	Pulled from SourceSiteURL column in `BFE ACTIONS`
source_site_id	bigint NULL	Pulled from SourceSiteID column in `BFE ACTIONS`
source_content_id	int NULL	Pulled from SourceContentID column in `BFE ACTIONS`
source_site_name	nvarchar(1024) NULL	Pulled from SourceSiteName column in `BFE ACTIONS`

Table 7. datasource_actionsite_properties

The datasource_properties table contains all properties that are pulled from the BES.properties table that exist in the actionsite. This is a temporary table. The properties are used within the ETL to populate the datasource_property_map table which can be used to generate dimensions tables.

Name	Type	Description
id	bigint NOT NULL IDENTITY(1, 1)	Datasource property id
datasource_id	int NOT NULL	dbo.datasources id
remote_content_id	bigint NOT NULL	remote content id
remote_site_id	bigint NOT NULL	remote site id
remote_property_id	tinyint NOT NULL	remote property id
name	nvarchar(512) NOT NULL	property name

Table 8. datasource_analyses

The datasource_analyses table contains external and custom analyses from ETL'd datasources.

Name	Type	Description
id	bigint NOT NULL IDENTITY(1, 1)	The datasource analysis id
remote_id	bigint NOT NULL	The remote analysis id
datasource_id	int NOT NULL	The dbo.datasources id
datasource_site_id	bigint NOT NULL	The dbo.datasource_sites id
name	nvarchar(255) NOT NULL	The name of the analysis

Table 8. datasource_analyses

The datasource_analyses table contains external and custom analyses from ETL'd datasources.

(continued)

Name	Type	Description
deleted	bit NOT NULL	Whether the content is deleted
valid_from	datetime2 (2) GENERATED ALWAYS AS ROW START NOT NULL	The timestamp from the last ETL
valid_to	datetime2 (2) GENERATED ALWAYS AS ROW END NOT NULL	The timestamp indicating until when the row is valid

Table 9. datasource_analysis_properties

The datasource_analysis_properties table stores property data associated with dbo.datasource_analyses.

Name	Type	Description
id	bigint NOT NULL IDENTITY(1, 1),	The datasource analysis property ID
remote_id	bigint NOT NULL	The remote content id
datasource_analysis_id	bigint NOT NULL	The dbo.datasource_analyses id
name	nvarchar(512) NOT NULL	The name of the property
deleted	bit NOT NULL	Whether the content is deleted

Table 9. datasource_analysis_properties

The datasource_analysis_properties table stores property data associated with dbo.datasource_analyses.

(continued)

Name	Type	Description
valid_from	datetime2 (2) GENERATED ALWAYS AS ROW START NOT NULL	The timestamp from the last ETL
valid_to	datetime2 (2) GENERATED ALWAYS AS ROW END NOT NULL	The timestamp indicating until when the row is valid

Table 10. datasource_computer_groups

The datasource_computer_groups table provides relationship of computers to groups. It displays all computer groups memberships across all datasource devices. This is more like a flattened computer group membership results that is not linked and there is no global representation within this table.

Name	Type	Description
id	bigint NOT NULL IDENTITY(1, 1)	The datasource computer group ID
datasource_id	int NOT NULL	The datasources id
datasource_site_id	bigint NOT NULL	The dbo.datasource_sites id
datasource_group_id	bigint NOT NULL	The dbo.datasource_groups id
datasource_device_id	bigint NOT NULL	The dbo.datasource_devices id

Table 10. datasource_computer_groups

The datasource_computer_groups table provides relationship of computers to groups. It displays all computer groups memberships across all datasource devices. This is more like a flattened computer group membership results that is not linked and there is no global representation within this table.

(continued)

Name	Type	Description
version	varbinary	The group version
is_member	tinyint	Whether the computer is a member of the group
valid_from	AS ROW START NOT NULL	The timestamp from the last ETL
valid_to	AS ROW END NOT NULL	The timestamp indicating until when the row is valid

Table 11. datasource_databases

The datasource_databases table contains datasource definitions and parameters of how to target and authenticate to the datasource.

Name	Type	Description
id	int NOT NULL IDENTITY(1, 1)	The datasource database id
host	nvarchar(512) NOT NULL	The hostname or IP address
database	nvarchar(128) NOT NULL	The database name
type	nvarchar(8) NOT NULL CONSTRAINT "DF_ca681866723fd-	The type of datasource

Table 11. datasource_databases

The datasource_databases table contains datasource definitions and parameters of how to target and authenticate to the datasource.

(continued)

Name	Type	Description
	b25046ba6b637c" DEFAULT 0	• BFE - BigFix Enterprise • BFI - BigFix Inventory • SCA - BigFix Software and Computer Analytics
username	nvarchar(128) NOT NULL	The database username (encrypted)
password	nvarchar(1024)	The database password (encrypted)
port	int	The database port
domain	nvarchar(512)	The database Windows domain
instance_name	nvarchar(512)	The database SQL server instance name

Table 12. datasource_devices

The datasource_devices table stores all devices from all ETLed BES deployments.

Name	Type	Description
id	bigint NOT NULL IDENTITY(1, 1)	The datasource device id
datasource_id	int NOT NULL	The dbo.datasources id

Table 12. datasource_devices

The datasource_devices table stores all devices from all ETLed BES deployments.

(continued)

Name	Type	Description
remote_id	bigint NOT NULL	The remote device id
name	nvarchar(512)	The device name
report_number	bigint NOT NULL	The number of the last report received
last_report_time	datetime	The last time a report was received. Format: unix epoch timestamp in UTC
last_report_sent	datetime	The last time a report was sent. Format: unix epoch timestamp in UTC
locked	bit NOT NULL CONSTRAINT "DF_-f7c2c0228e83297504c49ac4074" DEFAULT 0	Whether the device is locked
deleted	bit NOT NULL CONSTRAINT "DF_-4079840396b41f9b717aed30e-f9" DEFAULT 0	Whether the device is deleted
valid_from	datetime2 (2) GENERATED ALWAYS AS ROW START NOT NULL	The timestamp from last ETL

Table 12. datasource_devices

The datasource_devices table stores all devices from all ETLed BES deployments.

(continued)

Name	Type	Description
valid_to	datetime2 (2) GENERATED ALWAYS AS ROW END NOT NULL	The timestamp indicating until when the row is valid

Table 13. datasource_fixlets

The datasource_fixlets table displays all content across all datasources. These are flattened content objects that are not linked and there is no global representation within this table.

Name	Type	Description
id	bigint NOT NULL IDENTITY(1, 1)	The datasource content id
datasource_id	int NOT NULL	The dbo.datasources id
datasource_site_id	bigint NOT NULL	The dbo.datasource_sites id
remote_id	bigint NOT NULL,	The remote content id
name	nvarchar(1024) NOT NULL	The fixlet name
class	nvarchar(32)	The fixlet class name
category	nvarchar(255)	The fixlet category name
is_task	tinyint NOT NULL	Whether the fixlet is a task
severity	nvarchar(64)	The fixlet severity
source	nvarchar(255)	The fixlet source

Table 13. datasource_fixlets

The datasource_fixlets table displays all content across all datasources. These are flattened content objects that are not linked and there is no global representation within this table.

(continued)

Name	Type	Description
source_release_date	nvarchar(32)	The fixlet source release date
source_id	nvarchar(1024)	The source id for this fixlet
download_size	bigint	The download size for this fixlet
cve	nvarchar(max)	The CVE id list
cvss	cvss nvarchar(127)	The CVSS id list
sans	nvarchar(255)	The sans id list
hidden	bit NOT NULL CONSTRAINT "DF_-9da82af0b7111ef-be9652888c6b" DEFAULT 0	Whether this fixlet is hidden
deleted	bit NOT NULL CONSTRAINT "DF_-fc594b434464ca05741816b75fe" DEFAULT 0	Whether this fixlet is deleted
valid_from	datetime2 (2) GENERATED ALWAYS AS ROW START NOT NULL	The timestamp from last ETL

Table 13. datasource_fixlets

The datasource_fixlets table displays all content across all datasources. These are flattened content objects that are not linked and there is no global representation within this table.

(continued)

Name	Type	Description
valid_to	datetime2 (2) GENERATED ALWAYS AS ROW END NOT NULL	The timestamp indicating until when the row is valid

Table 14. datasource_groups

The datasource_groups table has a listing of all datasources computer groups. These are flattened content results that is not linked. There is no global representation within this table.

Name	Type	Description
id	bigint NOT NULL IDENTITY(1, 1)	The datasource group id
datasource_id	int NOT NULL	The dbo.datasources id
datasource_site_id	bigint NOT NULL	The dbo.datasource_sites id
remote_id	bigint NOT NULL	The remote group id
name	nvarchar(512) NOT NULL	The group name
type	tinyint NOT NULL	The group type. Values: 0 - manual 1 - automatic
is_client_evaluated	bit NOT NULL CONSTRAINT "DF_-	Whether the client evaluates that it is a member of

Table 14. datasource_groups

The datasource_groups table has a listing of all datasources computer groups. These are flattened content results that is not linked. There is no global representation within this table.

(continued)

Name	Type	Description
	9878d124eced9143031872264f8d" DEFAULT 0	group. This is always 1 for automatic groups. For manual groups, this is 1 when the group is "Determined by client settings"
creator_id	int NOT NULL	The user that created this group, if any.
creation_time	datetime2	The creation time. Format: Unix epoch timestamp in UTC
deleted	bit NOT NULL CONSTRAINT "DF_c3d-b464e009de0643d2455c0f99" DEFAULT 0	Whether the content is deleted
valid_from	datetime2 (2) GENERATED ALWAYS AS ROW START NOT NULL	The timestamp from the last ETL
valid_to	datetime2 (2) GENERATED ALWAYS AS ROW END NOT NULL	The timestamp indicating until when the row is valid

Table 15. datasource_property_map

The datasource_property_map table includes all properties across all deployments. Data is pulled from [BFInsights].datasource_actionsite_properties, [BFInsights].datasource_analysis_properties, [BFInsights].datasource_analyses. This is a temporary table and is used to generate the dimension table [BFInsights].[device_dimensions].

Name	Type	Description
id	bigint NOT NULL IDENTITY(1, 1)	The datasource property map id
datasource_id	int NOT NULL	The dbo.datasources id
datasource_site_id	bigint NOT NULL	The dbo.datasource_sites id
remote_content_id	bigint NOT NULL	The remote content id
remote_property_id	bigint NOT NULL	The remote property id
name	nvarchar(512) NOT NULL	The name of the property
type	tinyint	The property type. Following are the values: 0 - Reserved. This is a read-only predefined property. 1 - Default. This is a predefined property that can be edited. 2 - Custom. This is a user created property. Relevance: custom flag, default flag, reserved flag

Table 15. datasource_property_map

The datasource_property_map table includes all properties across all deployments. Data is pulled from [BFInsights].datasource_actionsite_properties, [BFInsights].datasource_analysis_properties, [BFInsights].datasource_analyses. This is a temporary table and is used to generate the dimension table [BFInsights].[device_dimensions].

(continued)

Name	Type	Description
deleted	bit NOT NULL	Whether the property is deleted
valid_from	datetime2 (2) GENERATED ALWAYS AS ROW START NOT NULL	The timestamp from the last ETL
valid_to	datetime2 (2) GENERATED ALWAYS AS ROW END NOT NULL	The timestamp indicating until when the row is valid

Table 16. datasource_property_results

The datasource_property_results table contains full list of property values across all deployments. This is a temporary table and is used in the webui to generate the [BFInsights].device_dimensions table.

Note: The values are pulled from QUESTIONRESULTS and LONGQUESTIONRESULTS, but only where:

- ***A valid WebUISiteID exists (NOT NULL)***
- ***The property type is either Default (0) Reserved (1)***

Name	Type	Description
id	bigint NOT NULL IDENTITY(1, 1)	The datasource property id

Table 16. datasource_property_results

The datasource_property_results table contains full list of property values across all deployments. This is a temporary table and is used in the webui to generate the [BFInsights].device_dimensions table.

Note: The values are pulled from QUESTIONRESULTS and LONGQUESTIONRESULTS, but only where:

- ***A valid WebUISiteID exists (NOT NULL)***
- ***The property type is either Default (0) Reserved (1)***

(continued)

Name	Type	Description
datasource_id	int NOT NULL	The dbo.datasources id
remote_device_id	bigint NOT NULL	The remote device id
remote_site_id	bigint NOT NULL	The remote site id
remote_content_id	bigint NOT NULL	The remote content id
remote_property_id	bigint NOT NULL	The remote property id
value	nvarchar(4000)	The property value. **Note:** This value is nullable.

Table 17. datasource_sequences

The datasource_sequences table tracks sequence ranges for every ETL'd table. Once the ETL is started, you record the current latest sequence from BigFix Enterprise DB and record it here and next time the ETL is ran, it would begin sequence in ETL.

Name	Type	Description
datasource_id	int NOT NULL	The dbo.datasources id

Table 17. datasource_sequences

The datasource_sequences table tracks sequence ranges for every ETL'd table. Once the ETL is started, you record the current latest sequence from BigFix Enterprise DB and record it here and next time the ETL is ran, it would begin sequence in ETL.

(continued)

Name	Type	Description
entity_name	nvarchar(128) NOT NULL	The name of the entity being tracked
last_sequence	binary(8) NOT NULL CONSTRAINT "DF_1dbb74c62a71e48bed0b4d1eb9e" DEFAULT 0	The last sequence pulled for entity

Table 18. datasource_sites

The datasource_sites table holds the list of sites ETL'd from each datasource. The data comes from dbo.SITES table in BigFix Enterprise.

Name	Type	Description
id	bigint NOT NULL IDENTITY(1, 1)	The datasource site id, primary key, used to identify site within the Insights DB.
datasource_id	int NOT NULL	The dbo.datasources id, datasource id from which the site comes from.
remote_id	bigint NOT NULL	The remote site id, references SiteID within the BigFix Enterprise DB.

Table 18. datasource_sites

The datasource_sites table holds the list of sites ETL'd from each datasource. The data comes from dbo.SITES table in BigFix Enterprise.

(continued)

Name	Type	Description
name	nvarchar(512) NOT NULL	The name of the site, references Name column within the BigFix Enterprise DB.
display_name	nvarchar(512)	The display name of the site, option, display name of the site
url	nvarchar(1024) NOT NULL	The gather URL of the site.
version	int NOT NULL	The version of the site.
type	tinyint NOT NULL	The site type. Following are the values: 0 - External 1 - ActionSite 2 - Custom 3 - Operator Relevance: type
is_excluded	bit NOT NULL CONSTRAINT "DF_10df8b1811fbcc64562a31ae28d" DEFAULT 0	Whether the site is excluded from ETL, set from etl_sites is_excluded column
deleted	bit NOT NULL CONSTRAINT "DF_042faf76cefda84e084362635e2" DEFAULT 0	Whether the site is deleted

Table 18. datasource_sites

The datasource_sites table holds the list of sites ETL'd from each datasource. The data comes from dbo.SITES table in BigFix Enterprise.

(continued)

Name	Type	Description
valid_from	datetime2 (2) GENERATED ALWAYS AS ROW START NOT NULL	The timestamp from the last ETL
valid_to	datetime2 (2) GENERATED ALWAYS AS ROW END NOT NULL	The timestamp indicating until when the row is valid

Table 19. datasources

The datasources table represents all datasources added by the user when configuring ETLs.

Name	Type	Description
id	int NOT NULL IDENTITY(1, 1)	The datasource id.
name	uniqueidentifier NOT NULL DEFAULT NEWSEQUENTIALID()	The uuid for the datasource
excluded	bit NOT NULL CONSTRAINT "DF_8baa2c7f1a5bacdff54a25e37fb" DEFAULT 0	Whether the datasource is excluded from ETLs. Format: Boolean.
version	nvarchar(32)	Version of the datasource.
user_alias	nvarchar(128)	The datasource alias, assigned by the user.

Table 19. datasources

The datasources table represents all datasources added by the user when configuring ETLs.

(continued)

Name	Type	Description
last_modified	datetime2 NOT NULL CONSTRAINT "DF_-749648a827211eafd-f548bfcc6a" DEFAULT GETUTCDATE()	The last time the data-source was modified.
bes_database_id	int NOT NULL	The bes database_data-source id.

Table 20. etl_metrics

The etl_metrics table gives you the basic metrics information about ETLs that transpire. The purpose: Front-end WebUI app exposes data to user.

Note: Timestamps are generated from Insights SQL server via dbo.get_time().

Name	Type	Description
id	bigint NOT NULL IDENTI-TY(1,1)	The unique id
datasource_id	int NOT NULL	the id of ETL'ed datasource
start_time	datetime	the start time of ETL **Note:** This is ap-proximate and dif-fers slightly from

Table 20. etl_metrics

The etl_metrics table gives you the basic metrics information about ETLs that transpire.

The purpose: Front-end WebUI app exposes data to user.

Note: Timestamps are generated from Insights SQL server via dbo.get_time().

(continued)

Name	Type	Description
		what you find in ETL_TIMES.
end_time	datetime	The end time of ETL
duration_ms	bigint DEFAULT 0	The duration in milliseconds of ETL
status	tinint	The status of ETL. Following are the values: 0 - RUNNING 1 - SUCCESS 2 - FAILED
preflights	varchar(max)	the preflight data collected prior to runs of the etl

Table 21. etl_sites

The etl_sites table contains all sites from all datasources and acts as a state manager for the webui front-end to inform on whether a given site is "excluded" and/or "primary" based on a user selection.

Definitions:

**** "excluded": A BigFix site is considered excluded if it has made the appropriate selection witihin the WebUI frontend. Content from a site that is excluded will no longer be drawn and therefore any existing content within the database will remain stale. That is if you decide to exclude a site after having run an ETL, future updates to content within that site will not be pulled and you will end up seeing stale content.***

**** "primary": A BigFix site is considered primary if a user selects a site to be primary, or if it is the first time this site has been imported via the WebUI datasource import process. The primary flag is set at datasource import time. If there exists more than one datasource within BFInsights, then a primary site is used to inform [BFInsights].Global entities (global_fixlets, global_analyses, global_site) when these entities are updated after an ETL.***

Name	Type	Description
id	bigint not null identity(1, 1)	The ETL site id.
datasource_id	int not null	The dbo.datasources id.
remote_site_id	bigint not null	The remote site id.
name	nvarchar(512) not null	The masthead name of the site
display_name	nvarchar(512)	The propagated display name of the site.
type	tinyint not null	The site type. Following are the values: 0 - External

Table 21. etl_sites

The etl_sites table contains all sites from all datasources and acts as a state manager for the webui front-end to inform on whether a given site is "excluded" and/or "primary" based on a user selection.

Definitions:

**** "excluded": A BigFix site is considered excluded if it has made the appropriate selection witihin the WebUI frontend. Content from a site that is excluded will no longer be drawn and therefore any existing content within the database will remain stale. That is if you decide to exclude a site after having run an ETL, future updates to content within that site will not be pulled and you will end up seeing stale content.***

**** "primary": A BigFix site is considered primary if a user selects a site to be primary, or if it is the first time this site has been imported via the WebUI datasource import process. The primary flag is set at datasource import time. If there exists more than one datasource within BFInsights, then a primary site is used to inform [BFInsights].Global entities (global_fixlets, global_analyses, global_site) when these entities are updated after an ETL.***

(continued)

Name	Type	Description
		1 - ActionSite 2 - Custom 3 - Operator Relevance: type
url	navchar(512)	The gather URL for the site.
device_count	int not null	The number of devices subscribed to the site.
content_count	int not null	The number of content associated with the site.

Table 21. etl_sites

The etl_sites table contains all sites from all datasources and acts as a state manager for the webui front-end to inform on whether a given site is "excluded" and/or "primary" based on a user selection.

Definitions:

**** "excluded": A BigFix site is considered excluded if it has made the appropriate selection witihin the WebUI frontend. Content from a site that is excluded will no longer be drawn and therefore any existing content within the database will remain stale. That is if you decide to exclude a site after having run an ETL, future updates to content within that site will not be pulled and you will end up seeing stale content.***

**** "primary": A BigFix site is considered primary if a user selects a site to be primary, or if it is the first time this site has been imported via the WebUI datasource import process. The primary flag is set at datasource import time. If there exists more than one datasource within BFInsights, then a primary site is used to inform [BFInsights].Global entities (global_fixlets, global_analyses, global_site) when these entities are updated after an ETL.***

(continued)

Name	Type	Description
is_excluded	bit not null constraint "DF_cbe84844d-c63d312aac5d151072" default 0	Whether the site is excluded from ETL.
is_primary	bit not null	Whether the site is the primary site for all similar sites from all datasources.
deleted	bit not null constraint "DF_-2a71f0da409e899e54c3c7d-d3cf" default 0	Whether the site is deleted.

Table 21. etl_sites

The etl_sites table contains all sites from all datasources and acts as a state manager for the webui front-end to inform on whether a given site is "excluded" and/or "primary" based on a user selection.

Definitions:

**** "excluded": A BigFix site is considered excluded if it has made the appropriate selection witihin the WebUI frontend. Content from a site that is excluded will no longer be drawn and therefore any existing content within the database will remain stale. That is if you decide to exclude a site after having run an ETL, future updates to content within that site will not be pulled and you will end up seeing stale content.***

**** "primary": A BigFix site is considered primary if a user selects a site to be primary, or if it is the first time this site has been imported via the WebUI datasource import process. The primary flag is set at datasource import time. If there exists more than one datasource within BFInsights, then a primary site is used to inform [BFInsights].Global entities (global_fixlets, global_analyses, global_site) when these entities are updated after an ETL.***

(continued)

Name	Type	Description
link_user_alias	nvarchar(128)	When the site is the primary site, alias for the group of all linked items.
link_updated	datetime	When the site is the primary site, last time the metadata for the linked item group has changed.
version	int not null	Site version from datasource

Table 22. etl_times

The etl_sites table tracks the ETL Start time and validity for SCD. Internal tables should not be used for reporting; pleae use time_dimensions table.

Name	Type	Description
start_time	datetime2 NOT NULL	The start time of the ETL
datasource_id	int NOT NULL	The datasource id.
valid_to	datetime2 NOT NULL	The name of the site.

Table 23. global_analyses

The global_analyses table lists the global representation of an analysis object.

Name	Type	Description
id	bigint IDENTITY(1,1) NOT NULL	The global analysis id
datasource_analyses_remote_id	bigint NOT NULL	The remote_id from dbo.datasource_analyses
global_site_id	bigint NOT NULL	The dbo.global_sites id
datasource_analyses_id	bigint NOT NULL	The dbo.datasource_analyses id
name	nvarchar(512) NOT NULL	The name of the global analysis
user modified	bit NOT NULL CONSTRAINT "DF_-d0c00b9a184f80a8ae481c357cd" DEFAULT 0	Whether the analysis has been modified by the user
deleted	bit NOT NULL CONSTRAINT "DF_-	Whether the analysis has been deleted

Table 23. global_analyses

The global_analyses table lists the global representation of an analysis object.

(continued)

Name	Type	Description
	9c98cb597a5aade71b3f2ee5983" DEFAULT 0	

Table 24. global_analyses_associations

The global_analyses_associations maps dbo.global_analyses and dbo.datasource_analyses objects together. One global analysis could have multiple datasource analyses associated with it.

Name	Type	Description
global_analyses_id	bigint NOT NULL	References dbo.global_-analyses.id
datasource_analyses_id	bigint NOT NULL	References dbo.data-source_analyses.id
datasource_id	int NOT NULL	The id of the datasource to which the datasource_-analysis belongs, for ease of joining with specific datasources
deleted	bit NOT NULL	Whether the datasource_-analysis or global_analysis object has been deleted

Table 25. global_fixlet_associations

The global_fixlet_associations table tracks mapping between datasource fixlets and global fixltes table. One global fixlet can have mulitple datasource fixlets associated with it based on the datasource fixlet `id` which reference PK in dbo.datasource_fixlets table. Datasource fixlet `remote_id` columns reference ContentID columns in BigFix Enterprise database in external_fixlets and custom_fixlets tables

Name	Type	Description
global_fixlet_id	bigint not null	The id of a global fixlet; references dbo.global_fixlets ID column
datasource_fixlet_id	bigint not null	The id of datasource fixlet; references dbo.datasource_fixlets ID **column**
datasource_id	int not null	The id of the datasource to which the global content belongs, for ease of joining **with specific datasource**
deleted	bit not null	Whether the datasource_fixlet or global_fixlet object has **been deleted**

Table 26. global_site_associations

The global site association table tracks mapping between datasource sites and global sites. One global site can have mulitple datasource sites associated with it based on datasource site `id` which reference PK in Datasource site table. Datasource site remote_id columns reference SiteID columns in BigFix Enterprise database.

Name	Type	Description
global_site_id	bigint not null	The id of the global site; references dbo.global_-sites table ID column
datasource_site_id	bigint not null	The datasource Site ID; references datasource_sites **table**
datasource_id	int not null	**The id of the datasource**
deleted	bit not null	Whether the given global site is **deleted**

Table 27. global_site_metadata

The global_site_metadata table allows user-configured fields to be associated with the global sites. global_site_metadata references dbo.global_sites through global_site_id. All fields are user modifiable.

Name	Type	Description
global_site_id	bigint not null	The global site id; has one to one relationship with the global site
configuration	nvarchar(128) null	**Custom field; user-defined**
release_version	nvarchar(128) null	**Custom field; user-defined**

Table 27. global_site_metadata

The global_site_metadata table allows user-configured fields to be associated with the global sites. global_site_metadata references dbo.global_sites through global_site_id. All fields are user modifiable.

(continued)

Name	Type	Description
vendor	nvarchar(128) null	**Custom field; user-defined**
description	nvarchar(512) null	**Custom field; user-defined**
updated_at	datetime2(0) not null	Has trigger on update , set **sbo.get_date() function**

Table 28. global_sites

The Global Representation of Sites, Global site represent external site object which can combine external sites from multiple datasources for reporting purposes. The linkage works through dbo.global_site_associations table. The key is datasource_sites 'remote_id' columns which link datasource sites to BigFix Enterprise site object, every external site has the same `remote_id`.

Name	Type	Description
id	bigint NOT NULL IDENTITY(1, 1)	The global site id, primary keym which identifies global site with in the deployment.
datasource_site_remote_id	bigint NOT NULL	The dbo.datasource_sites remote site id, references dbo.datasource_sites `remote_id` column.
name	nvarchar(512) NOT NULL	The name of the site, populated from datasource_-

Table 28. global_sites

The Global Representation of Sites, Global site represent external site object which can combine external sites from multiple datasources for reporting purposes. The linkage works through dbo.global_site_associations table. The key is datasource_sites 'remote_id' columns which link datasource sites to BigFix Enterprise site object, every external site has the same `remote_id`.

(continued)

Name	Type	Description
		sites name colums which is marked as master.
display_name	nvarchar(512)	The display name of the site, populated from data-source_sites name colums which is marked as master.
url	nvarchar(1024) NOT NULL	The gather url of the site, populated from data-source_sites name colums which is marked as master.
version	int NOT NULL	The version of the site.
type	tinyint NOT NULL	The site type. Values: 0 - External 1 - ActionSite 2 - Custom 3 - Operator Relevance: type
user_modified	bit NOT NULL CONSTRAINT "DF_-	Whether the site has been modified by the user, if site is marked as user modified

Table 28. global_sites

The Global Representation of Sites, Global site represent external site object which can combine external sites from multiple datasources for reporting purposes. The linkage works through dbo.global_site_associations table. The key is datasource_sites 'remote_id' columns which link datasource sites to BigFix Enterprise site object, every external site has the same `remote_id`.

(continued)

Name	Type	Description
	e9fc5a1422e7419b6127e01553541" DEFAULT 0	is not going to particiapte in ETL and should be updated manually.
deleted	bit NOT NULL CONSTRAINT "DF_-1777d4164442db2017e-f27482e9" DEFAULT 0	Whether the site is deleted, can be deleted only by a user.
updated_at	datetime2 (0)	The timestamp from last updated date.

Table 29. staging_fixlet_fields

The staging_fixlet_fields table dumps the raw data from BFE external_fixlet_fields and custom_fixlet_fields. As a staging table, the data in dbo.staging_fixlet_fields is used by Insights later on and not intended for user interaction.

Name	Type	Description
datasource_fixlet_id	bigint NOT NULL	Fixlet of the field object, references dbo.data-source_fixlet.id
datasource_id	int NOT NULL	Datasource of the fixlet, references dbo.data-source.id

Table 29. staging_fixlet_fields

The staging_fixlet_fields table dumps the raw data from BFE external_fixlet_fields and custom_fixlet_fields. As a staging table, the data in dbo.staging_fixlet_fields is used by Insights later on and not intended for user interaction.

(continued)

Name	Type	Description
name	nvarchar(1024) NOT NULL	Name of the field.
value	nvarchar(max) NOT NULL	Value of the field.
deleted	big NOT NULL	Whether the field has been deleted or not.

Table 30. staging_fixlet_results

The staging_fixlet_results table is a list of all fixlet results for all ETLed BES databases pulled from the BFE FIXLETRESULTS table. As a staging table, the data is meant to be consumed by other Insights tables and not intended for users.

Name	Type	Description
datasource_id	int NOT NULL	The dbo.datasources id.
datasource_site_remote_id	bigint NOT NULL	The remote site id pulled from BFE FIXLETRESULTS WebuiSiteID column , can be used to reference dbo.datasource_site.remote_id.
datasource_device_remote_id	bigint NOT NULL	The remote device id pulled from BFE FIXLETRESULTS ComputerID column, can be used to refer-

Table 30. staging_fixlet_results

The staging_fixlet_results table is a list of all fixlet results for all ETLed BES databases pulled from the BFE FIXLETRESULTS table. As a staging table, the data is meant to be consumed by other Insights tables and not intended for users.

(continued)

Name	Type	Description
		ence dbo.datasource_device.remote_id.
datasource_content_remote_id	bigint NOT NULL	The remote content id pulled from BFE FIXLETRESULTS ID column. Can be used to reference dbo.datasource_fixlets.remote_id OR datasource_analyses.remote_id. Need to check type column to determine the object result.
relevant	bit NOT NULL	To check whether fixlet is relevant.
type	tinyint NOT NULL	This indicates which type of fixlet: 0 - Fixlet 1 - Task 2 - Baseline 3 - Analysis 4 - ComputerGroup
first_relevant	datetime	The first time the action was non-relevant.

Table 30. staging_fixlet_results

The staging_fixlet_results table is a list of all fixlet results for all ETLed BES databases pulled from the BFE FIXLETRESULTS table. As a staging table, the data is meant to be consumed by other Insights tables and not intended for users.

(continued)

Name	Type	Description
last_relevant	datetime	The last time the action was relevant.
last_non_relevant	datetime	The last time the action was non-relevant.
updated_at	datetime2(2) NOT NULL CONSTRAINT "DF_62411c85e6f-f6c0c6ed7a96e3b4" DE-FAULT GETUTCDATE()	The timestamp when ETL last updated this row.

Table 31. time_dimensions

The time_dimensions table tracks all imports that have been ran on BigFix Enterprise database. This table helps to build a temporal view of the data over time. In order to see you data at specific time in the past, time_dimensions id column needs to be used with `as of <data>` query.

Name	Type	Description
	datetime2(2) NOT NULL	ID as timestamp, a time of ETL(Import) was ran, every successful import should have a record in this table.

Table 31. time_dimensions

The time_dimensions table tracks all imports that have been ran on BigFix Enterprise database. This table helps to build a temporal view of the data over time. In order to see you data at specific time in the past, time_dimensions id column needs to be used with `as of <data>` query.

(continued)

Name	Type	Description
datasource_id	int NOT NULL	dbo.datasources id, ID of the datasource for which import was ran.
day_of_week	tinyint NOT NULL	Indicates day of the week of timeslice. This column allow an easy filtering for specific time frame.
day_of_month	tinyint NOT NULL	Indicates day of the month of timeslice.
week_of_year	tinyint NOT NULL	Indicates week of year of timeslice.
month_of_year	tinyint NOT NULL	Indicates month of year of timeslice.
quarter	tinyint NOT NULL	Indicates quarter of timeslice.
year	smallint NOT NULL	Indicates year of timeslice.

Table 32. activity_history

The activity_history table is used to track ongoing status, events, and information from the activities panel.

Name	Type	Description
id	bigint NOT NULL IDENTITY(1, 1)	The activity ID.
user_ID	int NOT NULL	The user ID.
user_name	nvarchar(128) NOT NULL	The user name.
method	nvarchar(128) NOT NULL	Action taken. Values: create update delete
parent_id	nvarchar(128)	The parent ID.
entity_alias	nvarchar(128)	The datasource/server alias.
entity_type	nvarchar(36) NOT NULL	The entity type. Values • server • datasource • schedule • site
entity_id	int NOT NULL	The entity ID.
description	nvarchar(128)	The description of the activity.

Table 33. webui.etl_schedules

The webui.etl_schedules table is used to track ongoing ETLs schedules from datasources.

Name	Type	Description
id	bigint NOT NULL IDENTITY(1, 1)	The activity ID.
datasource_database_id	int NOT NULL	The dbo.datasource_database ID.
time_type	nvarchar(128) NOT NULL	The schedule period. Values: • Daily • Weekly • Monthly
day_of_week	nvarchar(128)	The day of the week. Values: MONDAY - FRIDAY
week_of_month	nvarchar(128)	The week of the month. Values: 1st-31st
day_after	int	The number of days after day_of_week.
hour	int	The hour.
minute	int	The minute.
next_etl	nvarchar(36)	The next etl time.
etl_status	nvarchar(36)	The etl status. Values:

Table 33. webui.etl_schedules

The webui.etl_schedules table is used to track ongoing ETLs schedules from datasources.

(continued)

Name	Type	Description
		• succeeded • failed • running

Table 34. device_dimensions

The device_dimensions table lists the detailed attributes for scanned devices in the BigFix domain.

Name	Type	Description
datasource_device_id	bigint not null	the device id – references datasource_devices
computer_name	nvarchar(512)	the name of the computer
locked	bit not null	the locked status 1 - locked 0 - not locked
os	nvarchar(512)	the operating system
cpu	nvarchar(512)	the cpu
relay	nvarchar(512)	the relay
dns_name	nvarchar(4000)	the DNS name
active_directory_path	nvarchar(4000)	the AD path
ip_address	nvarchar(4000)	the IP address(es)
ipv6_address	nvarchar(4000)	the IPV6 address(es)

Table 34. device_dimensions

The device_dimensions table lists the detailed attributes for scanned devices in the BigFix domain.

(continued)

Name	Type	Description
agent_version	nvarchar(128)	the agent version
device_type	nvarchar(128)	the device type
computer_type	nvarchar(128)	the computer type
user_name	nvarchar(512)	the username
ram	nvarchar(512)	the RAM
subnet_address	nvarchar(4000)	the subnet address
valid_from	datetime2	the valid_from date
valid_to	datetime2	the valid to date

Table 35. datasource_table_metrics

The datasource_table_metrics table lists the detailed attributes of ETL load procedures into the Insights database.

Name	Type	Description
etl_metric_id	bigint not null	the unique id of associated metric. References etl_metrics
datasource_id	int not null	the datasource id
table_name	nvarchar(128)	the name of the table
schema_name	nvarchar(128)	the name of the schema
rows	int not null	the number of rows

Table 35. datasource_table_metrics

The datasource_table_metrics table lists the detailed attributes of ETL load procedures into the Insights database.

(continued)

Name	Type	Description
total_space_mb	decimal(36,2) not null	the total space in megabytes
used_space_mb	decimal(36,2) not null	the used space in megabytes
unused_space_mb	decimal(36,2) no null	the unused space in megabytes

Chapter 12. Best practices

Learn more about the best practices in BigFix 10 Insights.

To minimize the impact of the data added and to optimize the ETL process duration, the following general best practices are recommended:

- Reduce the amount of sites imported (that is, dialog provided per data set). If you do not plan to historically report on the data, do not import the data into insights. Treat the data load like a debt and be aware of the accumulated workload.
- You must maintain the data of primary sites historically, starting from when you added them. You can never delete the data. Hence, be prescriptive of which sites to import at the time of setting up Insights itself.
- Plan to run the ETL at a frequency that is useful to your business operations (less is more, daily is reasonable). Balance the performance and managing expectations as appropriate.
- Reduce the site subscriptions of checklist to machines that are intended to evaluate.

Insights SQL Configuration

Configuration	Description \ reasoning
Verify SQL Server Collation matches all BigFix Enterprise datasources.	Collation setting for all databases in the system must match. The default collation for SQL Server databases is SQL_Latin1_General_CP1_CI_AS. Please refer to https://docs.microsoft.com/en-us/sql/relational-databases/collations/set-or-change-the-server-collation?view=sql-server-ver15 for more information.
TempDB IO channel must be a dedicated channel and not shared.	BigFix Insights leverages TempDB heavily. Try to isolate TempDB from other workloads.

Configuration	Description \ reasoning
Configure memory limitations within SQL rightly.	Make sure that the SQL memory is capped to allow at least 8 GB for the OS. This is configured within the SQL Server properties.
Configure virus scanners to exclude the SQL file storage location including all data file sets	When you configure your antivirus software settings, make sure that you exclude the following files or directories(as applicable) from virus scanning. This improves the performance of the files and makes sure that the files are not locked when the SQL Server service must use them. Refer to instructions from your virus scanner on how to set this exclusion rule. Reference: https://support.microsoft.com/en-us/help/309422/choosing-antivirus-software-for-computers-that-run-sql-server
Do not use file indexing or file compression on supporting SQL data files.	Similar to the reasoning behind A/V exclusions apply the same to HIPS based applications or File indexing operations that potentially could lock the data files in use.
Initially establish the sizes of Tlog and MDF to 80% of size projection and set auto growth appropriately.	Leveraging the sizing information provided assures that the MDF and LDF data files are initially established prior to setup of the initial database. This will allow the system to minimize auto growth during the initial ETL.
Configure auto growth of SQL Database files to be substantial vs minimal.	The Supporting DB File can become quite sizeable within the SQL in support of the Insights DB. To minimize the time and re-

Configuration	Description \ reasoning
	sources the system is taking growing the supporting data files, assure the growth characteristics are changed from default value to be more than 2 GB autogrow, or by 10% every time.
Assure soft NUMA is in place	Per Microsoft SQL 2017 and 2019, soft NUMA configurations should be managed by default. Make sure that this setting has not been altered from the default. For more information, see https://docs.microsoft.com/en-us/sql/database-engine/configure-windows/soft-numa-sql-server?view=sql-server-ver15

Chapter 13. BigFix Insights Troubleshooting

This topic helps you in troubleshooting various issues encountered in BigFix 10 Insights.

App Log Location

BigFix Insights application is an app of the WebUI Server that is configured as the ETL server. As such, logs are located within the native WebUI Location. This should generally be located within the following directory:

```
<Installation Drive>:\Program Files (x86)\BigFix Enterprise\BES WebUI\WebUI\logs
```

Note: The WebUI application logs are configured to be stored in an alternative directory. The BigFix Insights application logs are present within the alternate directory if the WebUI application logs are configured to do so.

BigFix Insights maintains a log within the log directory. The application name is Insights, thus the log is called `insights.log` within this directory. The log is configured to roll as the logged events grow. Depending on the total size of the log set, you may notice log files that are suffixed within the integer representations (that is `insights.log.1`, `insights.log.2`, and so on). Take note of the file modification time when evaluating the log file as the log rolls forward.

Advanced Logging

To enable verbose logging, you should modify the WebUI settings *_WebUI_Logging_Filter* to include bf* (including asterisk). The application or the WebUI service should be restarted for the changes to take effect. This log output is very expansive, so it is advisable that the setting should be returned to its previous value once your debugging session is ended.

The insights application log output will be made to be verbose.

Also, additional logging will now exist at the following file path: `<Installation Drive>:\Program Files (x86)\BigFix Enterprise\BES WebUI\WebUI\sites\WebUI Insights_ <hash><site_version><timestamp>\insights-app\logs\imports`

SQL Tables of interest

ETL_Metrics

BigFix Insights maintains a table log of all running and completed ETLs. This data is captured within the *ETL_Metrics* table. This table should never be modified or altered manually. This table can be located on your Insights SQL Server Database, within the *ETL_Metrics* table. The following SQL query can be used to select information from this table:

```
SELECT TOP 1000 [id]
      ,[datasource_id]
      ,[start_time]
      ,[end_time]
      ,[duration_ms]
      ,[status]
      ,[detail_log]
,[preflights]
  FROM dbo].[etl_metrics]
```

The *ETL_Metrics* table documents each ETL from each Datasource and maintains a `detail_log` column that represents the metrics for the given ETL. The column value is structured as a JSON value. The following is an example of the JSON value that is written per ETL.

```
{"step1":{"entity":"DatasourcePropertyResult","startSequence":"0x000000001f
c08879",
"endSequence":"0x000000001fcd3510","startTime":"2020-03-28T18:26:19.463Z",
"endTime":"2020-03-28T18:26:20.363Z","durationMs":900},
"step2":{"entity":"ActionStateString","startSequence":"0x000000001fc08879",
"endSequence":"0x000000001fcd3510","startTime":"2020-03-28T18:26:20.376Z",
"endTime":"2020-03-28T18:26:20.503Z","durationMs":127},
"step3":{"entity":"DatasourceSite","startSequence":"0x000000001fc08879",
"endSequence":"0x000000001fcd3510","startTime":"2020-03-28T18:26:20.516Z",
"endTime":"2020-03-28T18:26:20.743Z","durationMs":227},
```

```
"step4":{"entity":"DatasourceActionsiteProperty","startSequence":"0x0000000
01fc08879",
"endSequence":"0x000000001fcd3510","startTime":"2020-03-28T18:26:20.743Z",
"endTime":"2020-03-28T18:26:20.870Z","durationMs":127},
"step5":{"entity":"DatasourceDevice","startSequence":"0x000000001fc08879",
"endSequence":"0x000000001fcd3510","startTime":"2020-03-28T18:26:20.980Z",
"endTime":"2020-03-28T18:26:21.103Z","durationMs":123},
"step6":{"entity":"DatasourceFixlet","startSequence":"0x000000001fc08879",
"endSequence":"0x000000001fcd3510","startTime":"2020-03-28T18:26:21.103Z",
"endTime":"2020-03-28T18:26:24.896Z","durationMs":3793},
"step7":{"entity":"DatasourceAnalysis","startSequence":"0x000000001fc08879"
,
"endSequence":"0x000000001fcd3510","startTime":"2020-03-28T18:26:24.910Z",
"endTime":"2020-03-28T18:26:25.053Z","durationMs":143},
"step8":{"entity":"DatasourceAnalysisProperty","startSequence":"0x000000001
fc08879",
"endSequence":"0x000000001fcd3510","startTime":"2020-03-28T18:26:25.053Z",
"endTime":"2020-03-28T18:26:25.210Z","durationMs":157},
"step9":{"entity":"DatasourceAction","startSequence":"0x000000001fc08879",
"endSequence":"0x000000001fcd3510","startTime":"2020-03-28T18:26:25.223Z",
"endTime":"2020-03-28T18:26:25.633Z","durationMs":410},
"step10":{"entity":"DatasourceGroup","startSequence":"0x000000001fc08879",
"endSequence":"0x000000001fcd3510","startTime":"2020-03-28T18:26:25.633Z",
"endTime":"2020-03-28T18:26:25.693Z","durationMs":60},
"step11":{"entity":"DatasourceComputerGroup","startSequence":"0x000000001fc
08879",
"endSequence":"0x000000001fcd3510",
"startTime":"2020-03-28T18:26:25.693Z","endTime":"2020-03-28T18:26:25.756Z"
,"durationMs":63},
"step12":{"entity":"DatasourcePropertyMap","startSequence":"0x000000001fc08
879",
"endSequence":"0x000000001fcd3510","startTime":"2020-03-28T18:26:25.773Z",
```

```
"endTime":"2020-03-28T18:26:25.900Z","durationMs":127},
"step13":{"entity":"StagingFixletField","startSequence":"0x000000001fc08879
",
"endSequence":"0x000000001fcd3510","startTime":"2020-03-28T18:26:25.913Z",
"endTime":"2020-03-28T18:26:27.510Z","durationMs":1597},
"step14":{"entity":"StagingFixletResult","startSequence":"0x000000001fc0887
9",
"endSequence":"0x000000001fcd3510","startTime":"2020-03-28T18:26:27.526Z",
"endTime":"2020-03-28T18:26:27.590Z","durationMs":64},
"step15":{"entity":"DatasourceActionResult","startSequence":"0x000000001fc0
8879",
"endSequence":"0x000000001fcd3510","startTime":"2020-03-28T18:26:27.590Z",
"endTime":"2020-03-28T18:26:27.763Z","durationMs":173},
"step16":{"entity":"ContentResult","startSequence":"0x000000001fc08879",
"endSequence":"0x000000001fcd3510","startTime":"2020-03-28T18:26:27.780Z",
"endTime":"2020-03-28T18:26:28.030Z","durationMs":250}}
```


The log is viewed easily in a JSON viewer (you get several applications online). The JSON is structured as an array of variables, where one step is a variable within the array. An example of a given step variable within the array is:


```
{"step1":{"entity":"DatasourcePropertyResult","startSequence":"0x000000001f
c08879",
"endSequence":"0x000000001fcd3510","startTime":"2020-03-28T18:26:19.463Z",
"endTime":"2020-03-28T18:26:20.363Z","durationMs":900},"
```


The above example indicates that step 1 is the *DatasourcePropertyResult* step. The ETL was configured to ingest from sequence `0x000000001fc08879` and ended at sequence `0x000000001fcd3510`. The step started at 2020-03-28T18:26:19.463Z and ended at 2020-03-28T18:26:20.363Z. This comprised a total duration of 900ms (or .9 seconds). The same principle is used to analyze the entirety of the resulting log value. Understanding the results can provide some context for ongoing ETL operations. The Sequence start and end range indicate the data that was retrieved within the step. Sequences are logged within the ingesting BFE server and represent a means to identify if something has changed, and

therefore must be imported. If the Start and End range sequence are the same value, this indicates there is nothing that has changed within the resulting BFE database to ingest.

Problem sets and potential remediations

Table 36. Problem sets and potential remediations

Problem or Issue	Course of Action
Issue in deploying the Insights DBMS	Verify the user credentials that are provided in the Insights setup has the appropriate permissions to create the target DB on the Insights SQL Server. Verify the proper target Insights SQL Server is accessible over the port specified (if none is specified, port 1433 to be used). A simple verification exercise is to review the SQL logs on the Insights DB server to verify the connection is being made and authentication is taking place (SQL server can be configured to audit successful logins and unsuccessful logins).
Data not updating for a given datasource	Insights ingests from a database target that is a hosted replica of the live BigFix Enterprise Server. Verify the replica has been updated with a recent dataset (by the backup or replication etc). A simple examination of the *ETL_Metrics* table can confirm if data has changed on the ingestion target. The *ETL_Metrics* table logs every ETL to every datasource. The section above itemizes how to interpret the data within that table. Comparing the previous successful run step with the most

Table 36. Problem sets and potential remediations (continued)

Problem or Issue	Course of Action
	recent step sequence id can confirm if data has indeed evolved on the underlying database. If the data is changing, Insights should be ingesting from the datasource, review the ETL logs and confirm that the ETL is successfully completed. If the ETL is not completing successfully on a repeating basis contact support.
Unable to delete a datasource	Within BigFix Insights there is a concept of a primary linked item. The primary linked item is assigned when an external site is selected to be ingested. BigFix Insights does not allow a user to remove a datasource that is associated as the *primary* for a given site. The primary for the given site must be first reassigned to another datasource before the datasource can be deleted.
Unable to connect to a datasource	Insights retrieves information from a corresponding datasource. Verify that the user credentials provided are correct and has the appropriate permissions to read from the target datasource DB. Verify that the proper target datasource SQL Server is accessible over the port specified (if none is specified, port 1433 should be used). A simple verification exercise is to review the SQL logs on the datasource DB server to verify that the connection is

Table 36. Problem sets and potential remediations (continued)

Problem or Issue	Course of Action
	being made and authentication is taking place (SQL server can be configured to audit successful logins and unsuccessful logins).
“Start Fresh” with Insights – Revert to a “New Install” because there are data-sources that you want to remove	To completely reset, the customer should run the following query against their BFEnterprise database to which the Web-UI reports: `delete from [BFEnterprise].[dbo].[webui_data] where App = 'insights'` The Insights database can be deleted or retained. When you re-visit the Insights WebUI Application, you should either connect to an existing Insights database or create a new database. If you want to retain your previous Insights database, provide a new name to the newly created Insights database.

Chapter 14. BigFix Insights Change Log

This topic lists out the change log in BigFix Insights.

January 2022 Release – Insights 10.0.8

- Fixes various bugs and defects
  - Fixes issues with the insights_table_metrics table population where duplicate key was possible
  - Fixes issues with tracking sequences for new entities
  - Fixes issue with Custom Attributes that would overwrite existing entries with the same name
  - Fixes issue with etl_metrics - adding support for binary type columns
  - Fixes issue where datasource_fixlets was not accurately setting hidden for custom content
- Added features
  - Site inclusion changes from excluded to included now cause a last_sequence reset for all entities associated with this datasource
  - Custom Attributes view adds a selected-only filter
  - Custom Attributes view adds validation for user input
  - Removes possible sensitive data from all future error output.
  - datasource_sites now includes operator sites
  - [schema change] device_dimensions added colums:
    - agent_type
  - [schmea change] datasource_sites added columns:
    - opsite_datasource_user_id
  - [schmea change] datasource_fixlets adds columns:
    - user_remote_id
    - creation_time
    - last_modification_time
  - [schmea change] added new entities (tables):
    - datasource_baselines
    - datasource_baseline_action_settings

- datasource_baseline_action_settings_user_groups
- datasource_baseline_component_actions
- datasource_baseline_component_action_successes
- datasource_baseline_component_groups
- datasource_baseline_components
- datasource_baseline_fields
- datasource_baseline_relevances
- datasource_custom_site_readers
- datasource_custom_site_writers
- datasource_device_baselines
- datasource_external_site_visibilities
- datasource_fixlet_actions
- datasource_content_comments
- datasource_fixlet_relevances
- datasource_role_group_assignments
- datasource_role_site_assignments
- datasource_role_user_assignments
- datasource_roles
- datasource_site_users
- datasource_users
- etl_entity_inclusions