TIBCO® RTView® for TIBCO BusinessWorks™ User's Guide

Version 7.1

TIBCO Software Inc.
3303 Hillview Avenue
Palo Alto, CA 94304-1213

CUSTOMER SUPPORT
For an overview of TIBCO Support Services, and information about getting started with TIBCO Product Support, visit this site:

http://www.tibco.com/services/support/default.jsp

If you already have a valid maintenance or support contract, visit this site:

http://support.tibco.com

Entry to this site requires a username and password. If you do not have a username, you can request one.

Preface

Welcome to the TIBCO® RTView® for TIBCO BusinessWorks™ User's Guide

Document Conventions

This guide uses the following standard set of typographical conventions.

Convention	Meaning
italics	Within text, new terms and emphasized words appear in italic typeface.
boldface	Within text, directory paths, file names, commands and GUI controls appear in bold typeface.

Courier	Code examples appear in Courier font:
	`amnesiac > enable`
	`amnesiac # configure terminal`

< >	Values that you specify appear in angle brackets: **interface <ipaddress>**

Third Party Notices

Please refer to the **LICENSES_thirdparty.txt** file from your product installation.

CHAPTER 1 Quick Start

This chapter is designed for those customers evaluating TIBCO® RTView® for TIBCO BusinessWorks™ for purchase and describes the basic steps required to install, configure, and start the Monitor using default settings while using Eclipse Jetty (which is delivered with the Monitor) as the application server. The steps listed in this chapter represent only the basic flow needed to get the Monitor up and running. See Introduction to the Monitor, Configuration, and Deployment for additional set up, configuration, and deployment options/details. Most of the configurations in this chapter are defined using the RTView Configuration Application. See RTView Configuration Application for more information.

By default, "Login" is disabled for the Monitor, which means that logging in will not be required when initially accessing the Monitor. See Enabling Login in the Monitor for more information on enabling Login. **Note**: This is only relevant if you are using the classic (non-HTML) displays.

This chapter contains:

Prerequisites for Windows and UNIX/Linux Installations

- TIBCO HAWK 5.1.x
- TIBCO RV 8.4.2+
- If you are using TIBCO Businessworks Version 6, you must enable TIBCO Hawk or install the OSGI Plugin. See Enable Monitoring via TIBCO Hawk for Versions 5 and 6 and Enable Monitoring via OSGi Plugin for Version 6 for more information.
- If you are upgrading to a newer version and want to copy your existing configurations/properties into the new version, see Upgrade Notes for information.
- Java JDK 8, 9, 11, or 17
- Application Server (for example, Eclipse Jetty (delivered with the Monitor), or Tomcat 8.5+)

For a more complete list of system requirements, refer to the **README_sysreq.txt** file from your product installation.

UNIX/Linux Quick Start Steps

1. Download **TIB_rtview-bw_<version>.zip** to your local UNIX/Linux server.

Note: If using UNIX, do not include spaces in your installation directory path. The **start_server.sh** and **stop_server.sh** scripts will not function properly if spaces are included in the installation directory path.

2. Extract the files:

   unzip -a TIB_rtview-bw_<version>.zip

   If correctly installed, you should see an **TIB_rtview-bw** directory with an **rtvapm** sub-directory, which should include **bwmon** and **bw6mon** as subdirectories.

3. Set the **JAVA_HOME** environment variable to point to your Java installation. For example:

   export JAVA_HOME=/opt/Java/jdk1.9.0

4. Navigate to the **TIB_rtview-bw** directory and type:

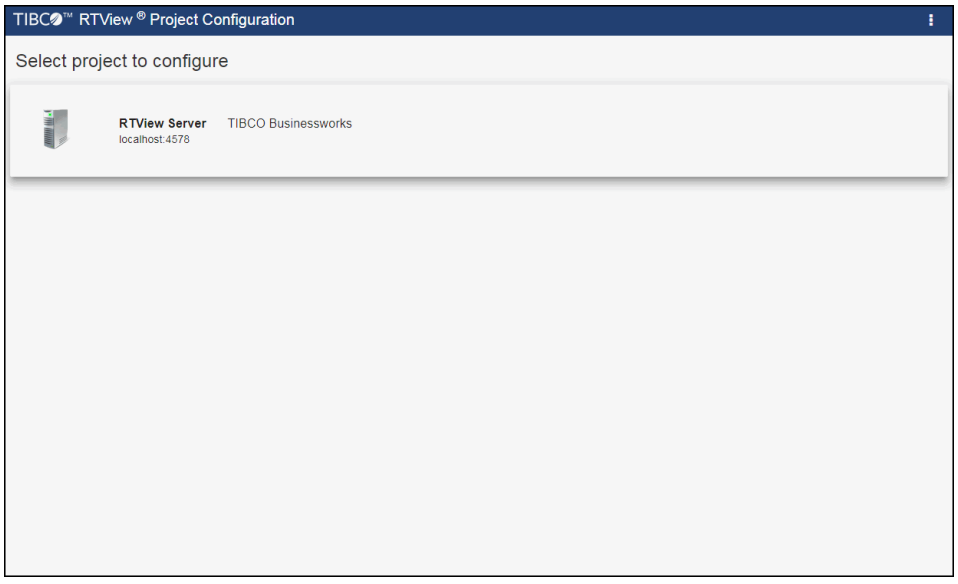
   start_server.sh

5. Open a browser and type the following URL to open the RTView Configuration Application:

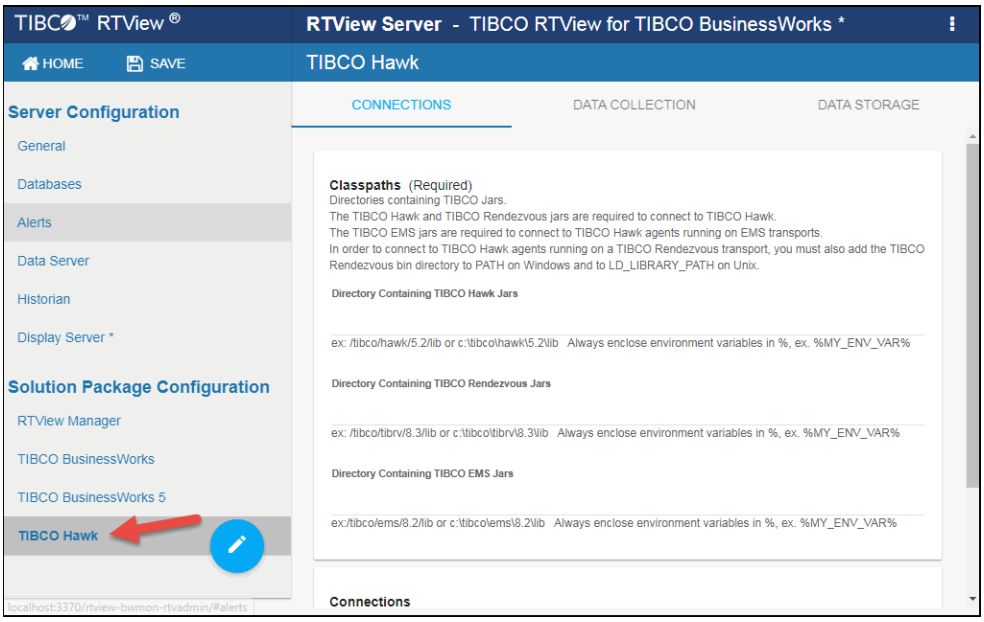
   http:/localhost:3370/rtview-bwmon-rtvadmin

   Use rtvadmin/rtvadmin as the username/password.

   The RTView Configuration Application displays.

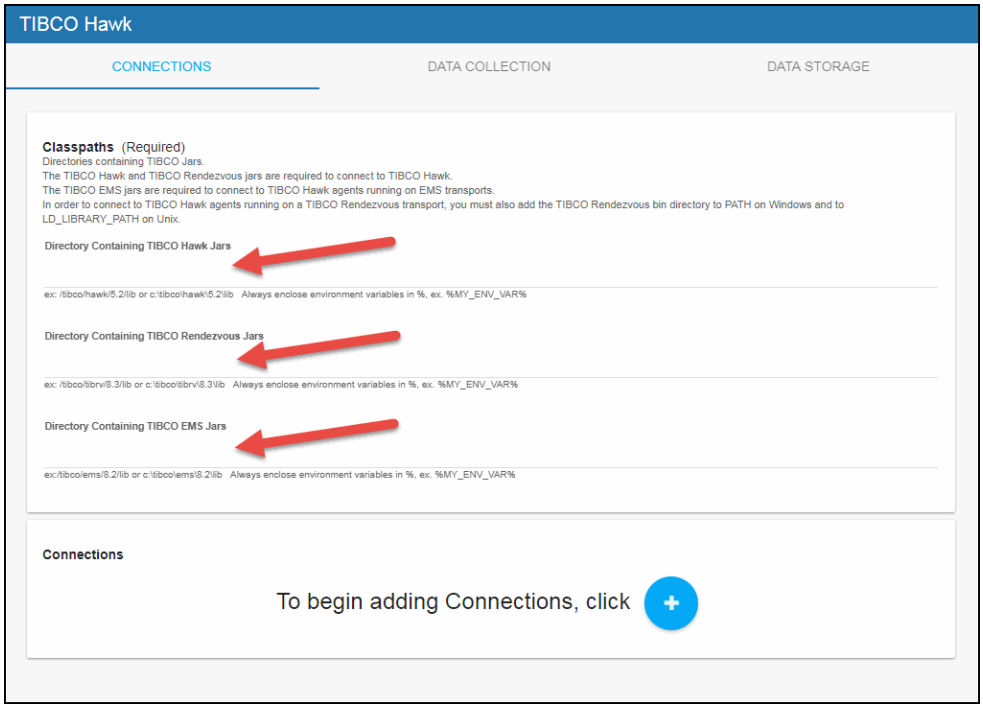


6. Select the **RTView Server** project, and then select **TIBCO Hawk** under **Solution Package Configuration**.



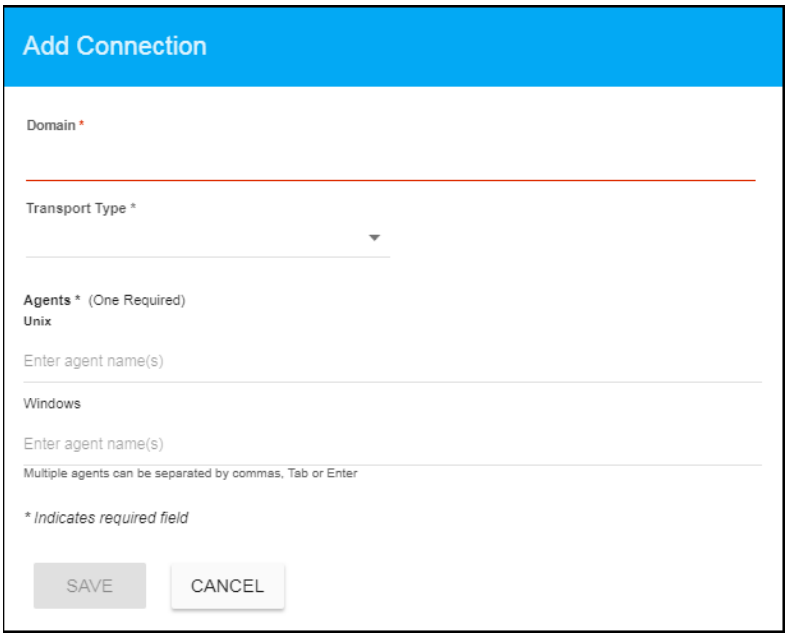
7. In the **Connections** tab, specify the classpaths for the TIBCO Hawk jar files, the TIBCO Rendezvous jar files, and the TIBCO EMS jar files.

Note: If you are using Version 6 and have installed the OSGI plugin, then you do not need to add TIBCO Hawk connections. Skip to Step 14.

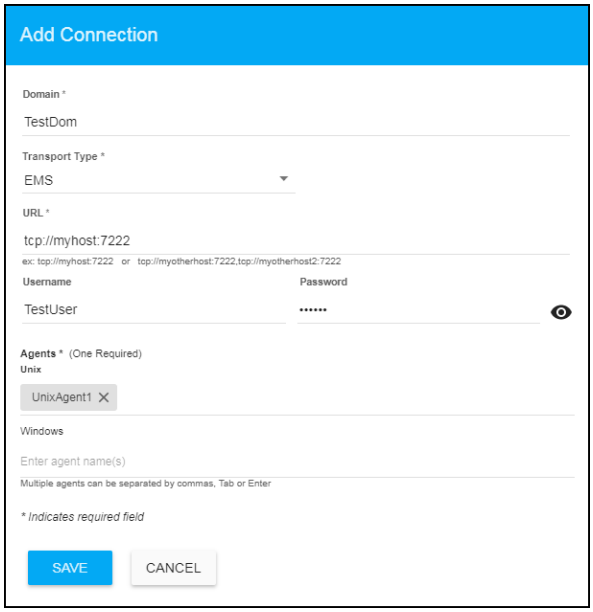


8. Click the + button in the **Connections** region.

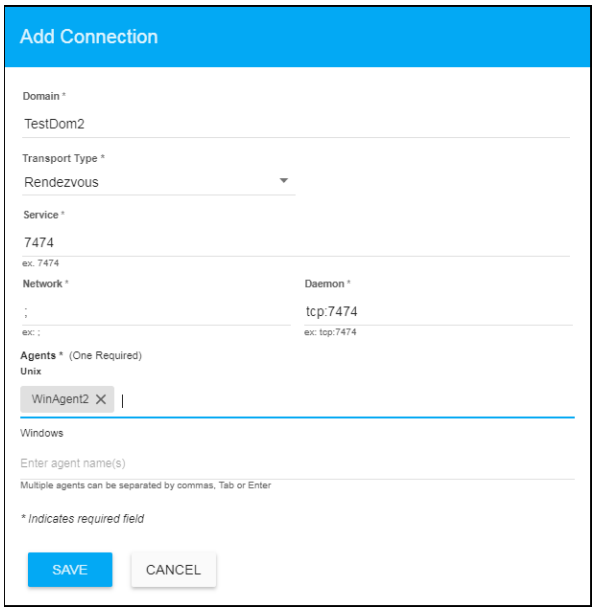
The **Add Connection** dialog displays.



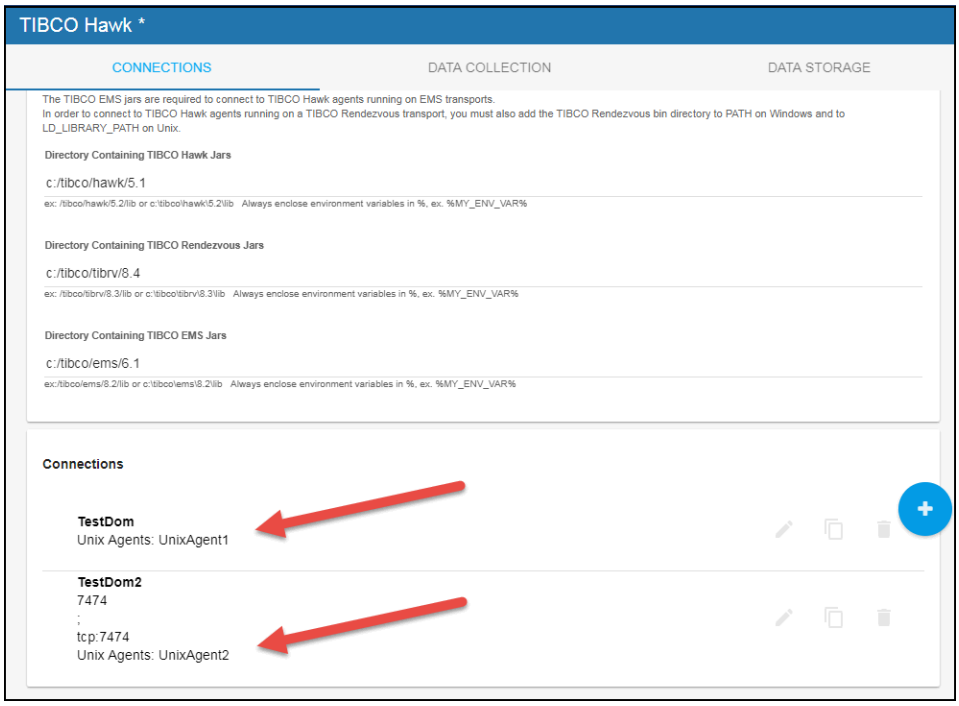
9. Enter a connection for each TIBCO Hawk domain used to monitor BusinessWorks. For a TIBCO Hawk domain using an EMS transport, specify the **Domain**, the **Transport Type** = **EMS**, the **URL**, the associated **Username** and **Password**, the associated **Unix** and **Windows** agent(s), and click **Save**. Repeat for all TIBCO Hawk domains running on EMS transports.



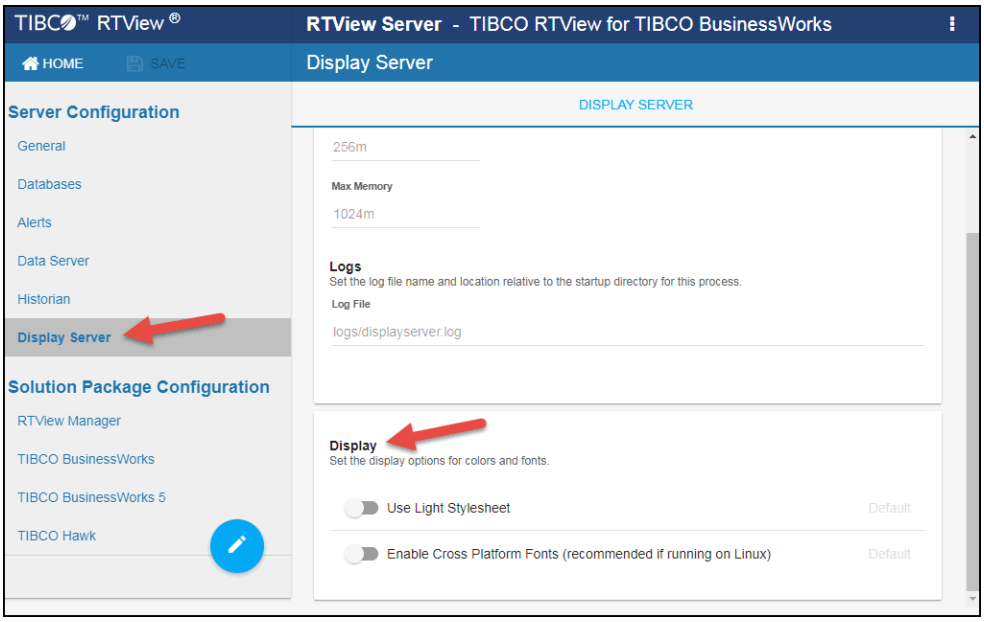
For TIBCO Hawk domains that use Rendezvous transports, specify the **Domain**, the **Transport Type** = **Rendezvous**, the **Service**, the **Network,** the **Daemon**, the associated **Unix** and **Windows** agent(s), and click **Save**. Repeat for all TIBCO Hawk domains running on Rendezvous transports.



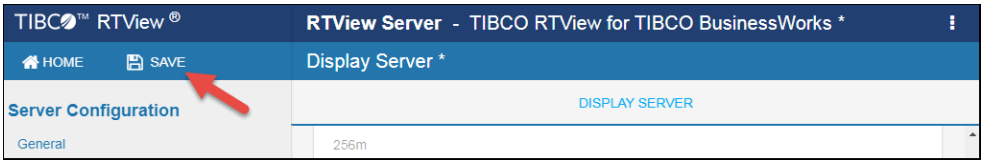
The connections you created display at the bottom of the **Connections** tab.



10. Optionally, if you are using the Classic (non-HTML) displays and want to use the white style sheet instead of the dark style sheet, and also want to enable Cross Platform Fonts, navigate to the **Display Server** option under **Server Configuration** and enable the **Use Light Stylesheet** and **Enable Cross Platform Fonts** toggles in the **Display** region.



11. Click the **SAVE** button in the RTView Configuration Application.



12. In order to apply the changes made in the RTView Configuration Application, you need to stop and restart all services by clicking the button in the upper right hand corner.

You could also restart the servers by navigating to the **TIB_rtview-bw** directory in a command window and typing:

stop_servers.sh

Once all services have stopped, type:

start_servers.sh

13. Check the log files in the **projects/rtview-server/logs** directory for errors.

14. You can verify that your caches are collecting data by accessing the following URL:

http(s)://localhost:3370/common

The **RTView Cache Viewer** application displays, which allows you to view the details for the caches that are collecting data.

15. To view the monitor with HTML displays, open a browser and navigate to **http://localhost:3370/rtview-bwmon** while using **rtvadmin**/**rtvadmin** as the login/password.

For the classic displays, open a browser and view the monitor using **http://localhost:3370/rtview-bwmon-classic** while using **rtvadmin**/**rtvadmin** as the login/password (if Login is enabled).

Windows Quick Start Steps

1. Download **TIB_rtview-bw_<version>.zip** to your local Windows server.
2. Extract the files in **TIB_rtview-bw_<version>.zip**using right mouse-click >"**Extract All...**"

   If correctly installed, you should see an **TIB_rtview-bw** directory with an **rtvapm** sub-directory, which should include **bwmon** and **bw6mon** as subdirectories.
3. Set the **JAVA_HOME** environment variable to point to your Java installation. For example:

   set JAVA_HOME=C:\Java\jdk1.9.0
4. Navigate to the **TIB_rtview-bw** directory and type:

   start_servers.bat (in a command window)

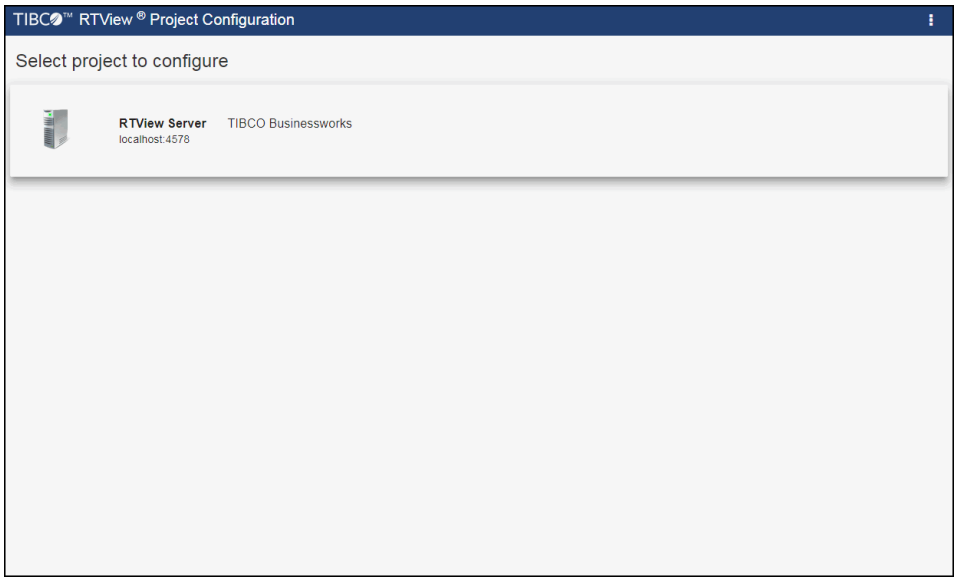
   or

   Double-click **start_servers.bat** in Windows
5. Open a browser and enter the following to open the RTView Configuration Application:

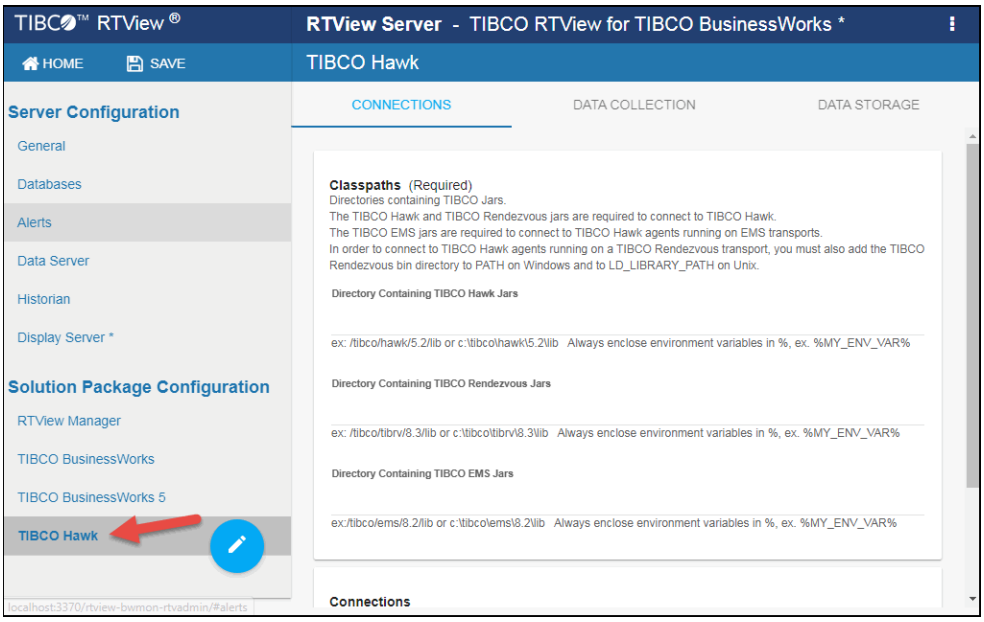
   http:/localhost:3370/rtview-bwmon-rtvadmin

   Use rtvadmin/rtvadmin as the username/password.

   The RTView Configuration Application displays.

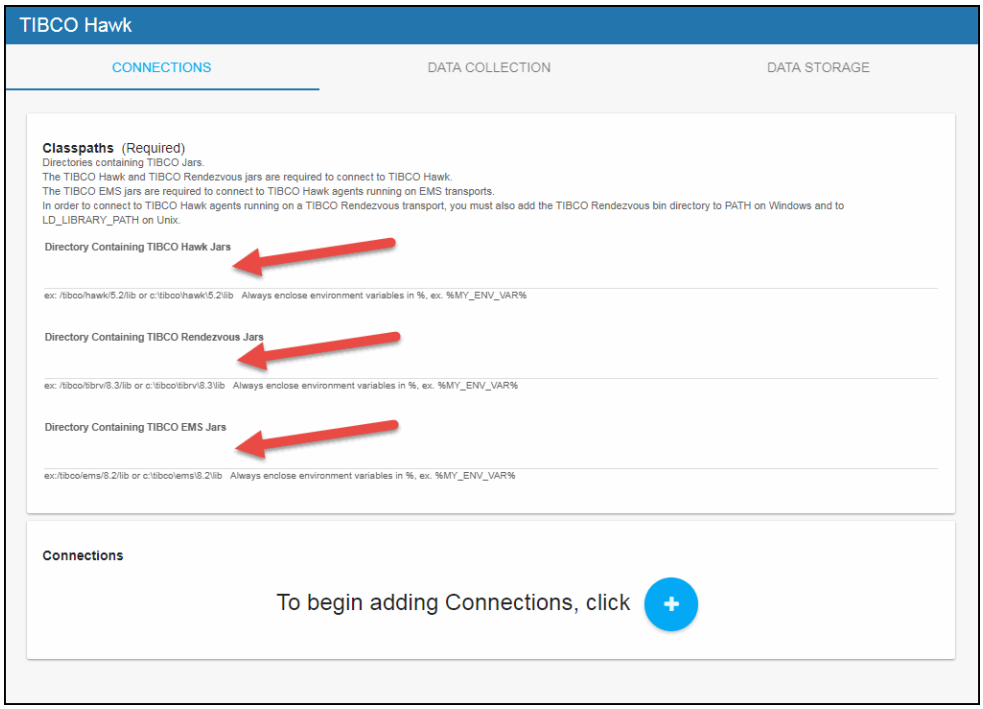


6. Select the **RTView Server** project, and then select **TIBCO Hawk** under **Solution Package Configuration**.



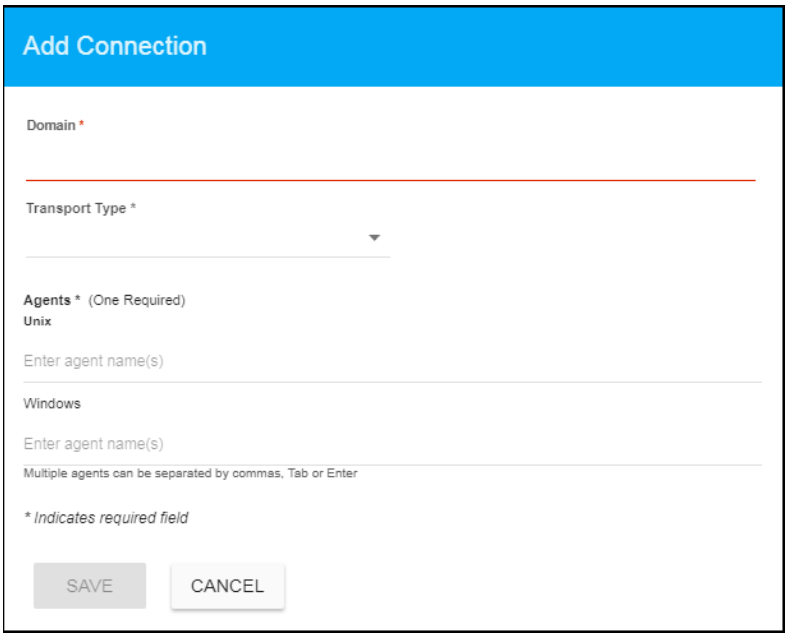
7. In the **Connections** tab, specify the classpaths for the TIBCO Hawk jar files, the TIBCO Rendezvous jar files, and the TIBCO EMS jar files.

Note: If you are using Version 6 and have installed the OSGI plugin, then you do not need to add TIBCO Hawk connections. Skip to Step 14.

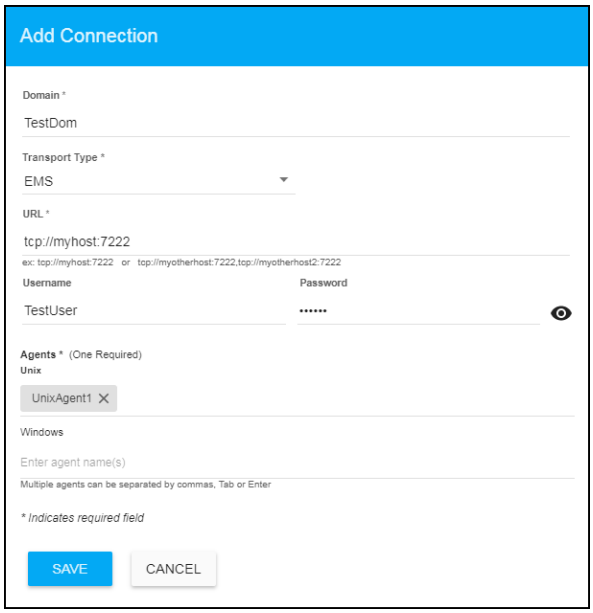


8. Click the button in the Connections region.

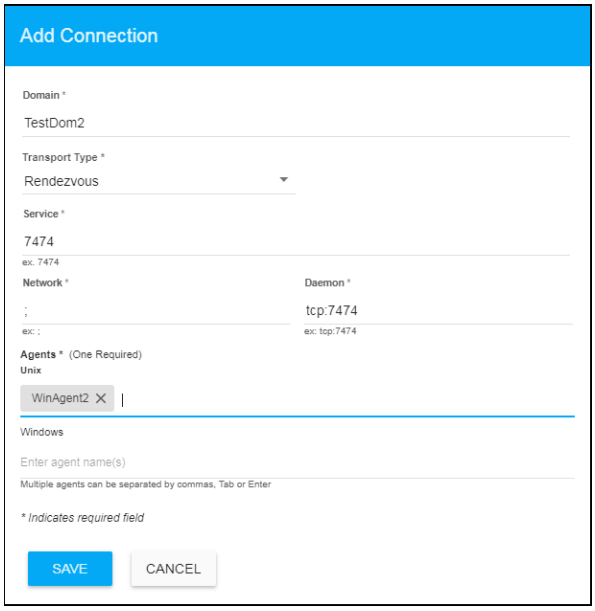
The **Add Connection** dialog displays.



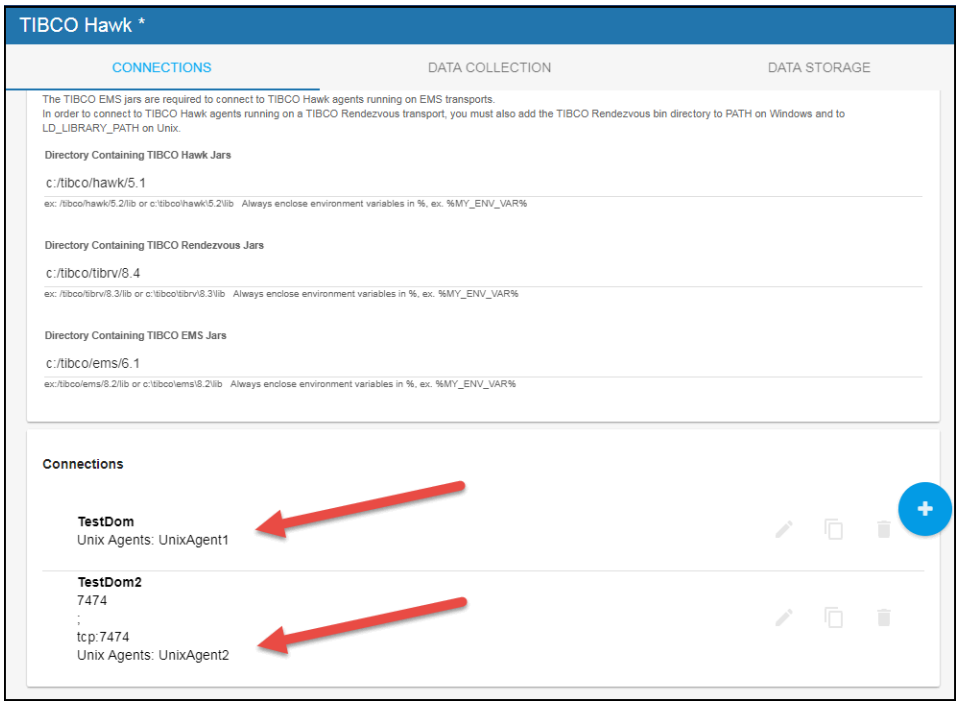
9. Enter a connection for each TIBCO Hawk domain used to monitor BusinessWorks. For a TIBCO Hawk domain using an EMS transport, specify the **Domain**, the **Transport Type** = **EMS**, the **URL**, the associated **Username** and **Password**, the associated **Unix** and **Windows** agent(s), and click **Save**. Repeat for all TIBCO Hawk domains running on EMS transports.



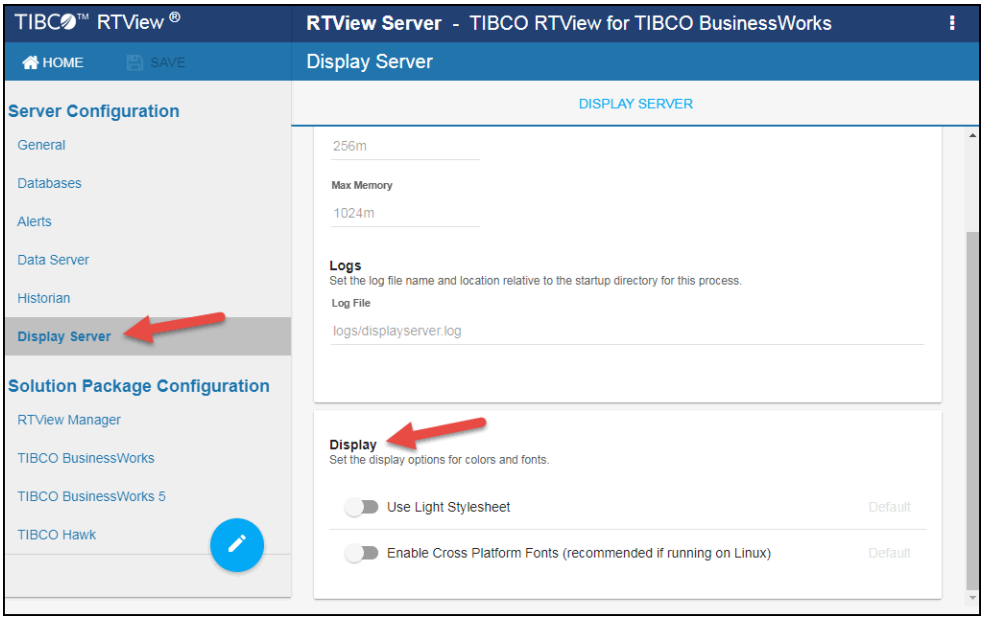
For TIBCO Hawk domains that use Rendezvous transports, specify the **Domain**, the **Transport Type** = **Rendezvous**, the **Service**, the **Network,** the **Daemon**, the associated **Unix** and **Windows** agent(s), and click **Save**. Repeat for all TIBCO Hawk domains running on Rendezvous transports.



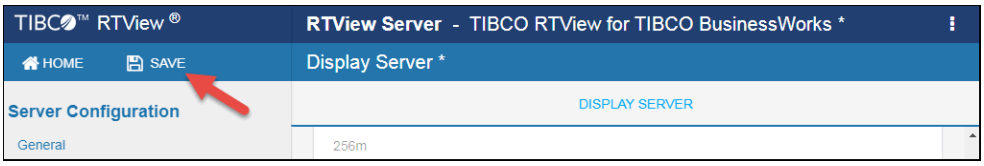
The connections you created display at the bottom of the **Connections** tab.



10. Optionally, if you are using the Classic (non-HTML) displays and want to use the white style sheet instead of the dark style sheet, and also want to enable Cross Platform Fonts, navigate to the **Display Server** option under **Server Configuration** and enable the **Use Light Stylesheet** and **Enable Cross Platform Fonts** toggles in the **Display** region.



11. Click the **SAVE** button in the RTView Configuration Application.



12. In order to apply the changes made in the RTView Configuration Application, you need to stop and restart all services by clicking the button in the upper right hand corner.

You could also restart the servers by navigating to the **TIB_rtview-bw** directory and typing:

stop_servers.bat (in a command window)

or by

Double-clicking **stop_servers.bat** in Windows

Once all services have stopped, type:

start_servers.bat (in a command window)

or

Double-click **start_servers.bat** in Windows

13. Check the log files in the **projects\rtview-server\logs** directory for errors.

14. You can verify that your caches are collecting data by accessing the following URL:

http(s)://localhost:3370/common

The **RTView Cache Viewer** application displays, which allows you to view the details for the caches that are collecting data.

15. To view the monitor with HTML displays, open a browser and navigate to **http://localhost:3370/rtview-bwmon** while using **rtvadmin**/**rtvadmin** as the login/password.

For the classic displays, open a browser and view the monitor using **http://localhost:3370/rtview-bwmon-classic** while using **rtvadmin**/**rtvadmin** as the login/password (if Login is enabled).

See Configuration, Deployment, and Using the Monitor for additional information about the configuration options, deployment options, and the various displays available in the Monitor.

CHAPTER 2 Introduction to the Monitor

This section contains the following:

Overview

The Monitor takes the time and guesswork out of monitoring and troubleshooting TIBCO® BusinessWorks System™ deployments, providing a centralized view of both real-time and historical performance metrics across numerous BusinessWorks™ Servers.

The Monitor enables TIBCO users to continually assess and analyze the health and performance of their TIBCO BusinessWorks™ infrastructure, gain early warning of issues with historical context, and effectively plan for capacity of their BusinessWorks™ Servers. It does so by aggregating and analyzing key performance metrics across all servers, engines, processes and activities, and presents the results, in real time, through meaningful dashboards as data is collected.

Users also benefit from predefined rules and alerts that pin-point critical areas to monitor in most BusinessWorks™ environments and allow for customization of thresholds to let users fine-tune when alert events should be activated.

The Monitor also contains alert management features so that the life cycle of an alert event can be managed to proper resolution. All of these features allow you to know exactly what is going on at any given point, analyze the historical trends of the key metrics, and respond to issues before they can degrade service levels in high-volume, high-transaction environments.

The Monitor can be deployed as a Web application run in a browser.

System Requirements

Please refer to the **README_sysreq.txt** from your product installation. A copy of this file is also available on the product download page.

Upgrade Notes

This section describes the steps necessary to upgrade existing Monitor applications.

Follow the steps for each version between the version you are upgrading from and the version to which you are upgrading:

- 7.1 -- See these steps to upgrade to version 7.1.
- 7.0 -- See these steps to upgrade to version 7.0.

- 6.7 -- See these steps to upgrade to version 6.7.
- 6.6 -- No upgrade steps required.
- 6.5 -- See these steps to upgrade to version 6.5.
- 6.4 -- No upgrade steps required.
- 6.3 -- See these steps to upgrade to version 6.3.

7.1

Log4j2

The syntax used in a Log4j properties file was changed completely by Apache in version 2. The **sl.log4j.properties** file distributed with RTView has been changed to use the version 2 syntax. If, in previous RTView versions, you customized that file or specified your own custom log4j properties file (e.g. using the "log4jprops" option), you'll need to remake those customization using the version 2 syntax.

Customization changes should be made to the copy of **sl.log4j.properties** in the **projects** directory, instead of the copy under **rtvapm\common\conf** in order to make it easier toupgrade to future releases.

Note that the default logging behavior has been changed: In this release by default messages are appended to the existing **logs/X.log** file (where X is "dataserver", or "historian", etc depending on the name of the server) until it reaches a size of 50MB. Then it is renamed to X.log.N (where N = 1 - 9) and a new empty X.log file is created. So at any time the logs directory may contain X.log (newest, up to 50MB in size), and X.log.N where N = 1 - 9, each approx 50 MB, where 1 is the oldest and 9 is the newest. Once N = 9 is reached, on the next rollover X.log.1 is deleted and each remaining X.log.N is renamed to X.log.N-1.

SNMP Notifications

If you are upgrading from a previous release that sent SNMP notifications, you need to update the MIB in your SNMP receiver. The MIB definition in **rtvapm\common\lib\SL-RTVIEW-EM-MIB.txt** has changed to include a new field for this.

Alert Notification

In previous releases, any notification other than executing the **my_alert_actions** scripts had to be configured on the **CUSTOM PROPERTIES** tab of the RTView Configuration Application. If you are running a project configured with a previous release that has alert notifications defined in the **CUSTOM PROPERTIES** tab, they will continue to work as they did before with no changes. However, if you want to use the new **Alert Notification** configuration page, you will need to first delete the **CUSTOM PROPERTIES** for alert notifications, then recreate them in the **Alert Notification** page. Otherwise, the **CUSTOM PROPERTIES** will override the settings in the **Alert Notification** page.

Admin Displays (HTML UI)

For improved security the following displays have been moved from **assets/packages/common** to **assets/packages/admin**:

- Alert Administration (rtv_alerts_admin_table.html)
- Alert Overrides Admin (rtv_alerts_admin_overrides.html)

- Component Alert Configuration (rtv_alerts_admin_detail.html)

Any existing browser bookmarks to those displays should be updated or recreated.

Project Directory Structure

1. Copy ***.properties** and ***.properties.json** from your old project directory to the **projects\rtview-server** directory.

2. If you modified the **rtvservers.dat** in your old project, make the same changes to **projects\rtview-server\rtvservers.dat**.

3. If you are not using jetty, deploy the **.war** files from **projects\rtview-server** to your application server.

4. If you are using Jetty, open **project.properties** and **project.properties.json** in a text editor and replace **../../webapps/** with **./** in all war file paths in both files. If upgrading a project that used the bw6mon war files, also change **bw6mon** to **bwmon** in the war file names.

5. Use the top level **start**/**stop**/**status** scripts to start, stop and check the status of your monitor processes.

6. There were previously separate monitors for BusinessWorks 5 and 6. These have been combined into a single monitor running on port prefix **33**. You will now access the monitor for both at **http://localhost:3370/rtview-bwmon** if you are using jetty.

7. Optionally follow the directions to remove the version of BusinessWorks that you are not using from the monitor.

Removing BusinessWorks 5 or 6 from the Monitor

1. To remove displays from the classic (thin client) user interface, open **projects\rtview-server\bwmon_navtree.xml** in a text editor. Delete the nodes for the version of BusinessWorks that you do not want to see. The nodes that start with BW are for BusinessWorks 6 and the nodes that start with BW5 are for BusinessWorks 5. Save the file.

2. To remove displays from the html user interface, **cd** to the **projects\rtview-server** directory in a terminal or command prompt and type the following:

   jar -xvf rtview-bwmon.war setup.js

3. Open **setup.js** in a text editor and edit the **window.EM_PACKAGE_LIST** array to remove either **bwmon** (for BusinessWorks 5) or **bw6mon** (for BusinessWorks 6). Save the file.

4. Type **jar -uf rtview-bwmon.war setup.js**

5. Run the Monitor.

6. Open the Configuration Application at: **http://localhost:3370/rtview-bwmon-rtvadmin**

7. Click on the **TIBCO RTView for TIBCO BusinessWorks** project.

8. In the left Navigation Panel, click on the pencil button to bring up the **Solution Packages** dialog. In the **Solution Packages** dialog, deselect the version of BusinessWorks that you are not using and click **SAVE** to close the dialog. Click the **SAVE** button in the top left corner and then click the **RESTART SERVERS** button in the upper right-hand corner.

7.0

Property File Handling and the RTView Configuration Application

Property file handling has been modified in order to support the "RTView Configuration Application". Existing applications will continue to work as before with no changes. However, customers should be aware of the following if they want to merge their old properties into the new version.

In previous releases, each sample project defined its own ports, sender target, and server identification properties. These properties have been removed from the sample properties and should be defined in the project properties instead. Sample projects have been updated to include these properties. Upward compatibility support is included for projects created prior to 6.8. In 6.8, the **rtview.properties** files in all sample projects were replaced with **project.properties** files. Any project with an **rtview.properties** file is recognized as a project created with a previous release. In that case, RTView will automatically read in the old ports, sender target, and server identification properties for all versions in the **rtview.properties** file. Therefore, projects created with previous versions will continue to run with no modifications. However, projects containing an **rtview.properties** file cannot be configured using the new RTView Configuration Application.

There are three options for merging properties from a previous version into the new version:

- The first option is to use the RTView Configuration Application to reapply the settings. If you have a lot of connections, this is not very practical but, if you only have a few, it could be worthwhile since you will be able to use the RTView Configuration Application for everything moving forward. To do this, run the new version, open the RTView Configuration Application, and apply all configurations that were part of your previous project. See Configuration for more information on how to configure your project using the RTView Configuration Application.
- The second option is to use your old properties files instead of the RTView Configuration Application. This has the downside that you cannot use the RTView Configuration Application moving forward. To do this, do the following in the sample project directory

1. Copy the properties files from the old project directory into the new project directory, including the old **rtview.properties**.

2. Remove the **project.properties** and **project.properties.json** from the new project directory.

3. Add your properties files to the appropriate lines in **servers\rtvservers.dat**.

- The third option for applying solution package server settings from a previous version is a combination of the above, which has the benefit of allowing you to use the RTView Configuration Application without having to re-enter all of your connections.

1. Copy the properties files from the old project directory into the new project directory. Do NOT copy the old **rtview.properties** into the new project directory.

2. Edit the properties file you just copied over to comment out or remove all non-connection properties.

3. Run the new project and use the RTView Configuration Application to apply all settings from your previous project except connections. See Configuration for more information on how to configure your solution package servers using the RTView Configuration Application.

4. Add your properties files from step 2 to the appropriate lines in **servers\rtvservers.dat**.

5. Moving forward, new connections can be added via the RTView Configuration Application or by hand editing the properties file from step 2, whichever is more convenient. However, only connections added via the RTView Configuration Application will be editable in the RTView Configuration Application.

Note: The following files are read and written by the Configuration Application and should never be manually edited: **project.properties** and **project.properties.json**.

6.7

OSGI Plugin

For existing users of TIBCO BusinessWorks Version 6 who are currently using the OSGI plugin, you will need to re-install the OSGI plugin after installing this version of TIBCO BusinessWorks. See Enable Monitoring via OSGi Plugin for Version 6 for more information on installing the plugin.

Version 6 Sample Project and Ports

Also for existing users of TIBCO BusinessWorks Version 6, the bw6mon sample project no longer contains the displays for BusinessWorks version 5. The ports used by the bw6mon sample project have also changed. Projects created in previous releases will continue to use the old ports, while users should be aware the ports have been changed in the new sample projects as follows:

For TIBCO BusinessWorks Monitor Version 6, the new port prefix is 45. This results in the following default port assignments:

dataserver data port 4578

dataserver JMX port 4568

datserver SC port 4570

dataserver rtvhttp port 4575

dataserver rtvagent port 4572

dataserver sender data port 4576

dataserver sender JMX port 4566

displayserver data port 4579

displayserver JMX port 4569

historian JMX port 4567

database (hsqldb) JMX port 4561

Property File Handling and the RTView Configuration Application

Property file handling has been modified in order to support the RTView Configuration Application. Existing applications will continue to work as before with no changes. However, customers should be aware of the following if they want to merge their old properties into the new version.

In previous releases, each sample project defined its own ports, sender target, and server identification properties. These properties have been removed from the sample properties and should be defined in the project properties instead. Version 5 and Version 6 sample projects have been updated to include these properties. Upward compatibility support is included for projects created prior to 6.7. In 6.7, the **rtview.properties** files in all sample projects were replaced with **project.properties** files. Any project with an **rtview.properties** file is recognized as a project created with a previous release. In that case, RTView will automatically read in the old ports, sender target, and server identification properties for all versions in the **rtview.properties** file. Therefore, projects created with previous versions will continue to run with no modifications. However, projects containing an **rtview.properties** file cannot be configured using the new RTView Configuration Application.

There are three options for merging properties from a previous version into the new version:

- The first option is to use the RTView Configuration Application to reapply the settings. If you have a lot of connections, this is not very practical but, if you only have a few, it could be worthwhile since you will be able to use the RTView Configuration Application for everything moving forward. To do this, run the new version (Version 5 or Version 6), open the RTView Configuration Application, and apply all configurations that were part of your previous project. See Configuration for more information on how to configure your project using the RTView Configuration Application.
- The second option is to use your old properties files instead of the RTView Configuration Application. This has the downside that you cannot use the RTView Configuration Application moving forward. To do this, do the following in the bwmon or bw6mon sample project directory

1. Copy the properties files from the old project directory into the new project directory, including the old **rtview.properties**.

2. Remove the **project.properties** and **project.properties.json** from the new project directory.

3. Add your properties files to the appropriate lines in **servers\rtvservers.dat**.
   - The third option for applying solution package server settings from a previous version is a combination of the above, which has the benefit of allowing you to use the RTView Configuration Application without having to re-enter all of your connections.

1. Copy the properties files from the old project directory into the new project directory. Do NOT copy the old **rtview.properties** into the new project directory.

2. Edit the properties file you just copied over to comment out or remove all non-connection properties.

3. Run the new project and use the RTView Configuration Application to apply all settings from your previous project except connections. See Configuration for more information on how to configure your solution package servers using the RTView Configuration Application.

4. Add your properties files from step 2 to the appropriate lines in **servers\rtvservers.dat**.

5. Moving forward, new connections can be added via the RTView Configuration Application or by hand editing the properties file from step 2, whichever is more convenient. However, only connections added via the RTView Configuration Application will be editable in the RTView Configuration Application.

Note: The following files are read and written by the Configuration Application and should never be manually edited: **project.properties** and **project.properties.json**.

6.6

No upgrade steps required.

6.5

Sender/receiver deployments

If you are using the sender/receiver deployment and upgrading projects from versions previous to 3.6, you need to modify properties files after upgrading in the following cases:

1. If the project properties files overwrite the **sender.sl.rtview.sub=$rtvAgentTarget** property, change it to use the new **sender.sl.rtvapm.dataxfr.target** property using the URL you specified for the **$rtvAgentTarget**. For example:

sender.sl.rtview.sub=$rtvAgentTarget:'localhost:3172'

would be changed to

sender.sl.rtvapm.dataxfr.target=id=default url=localhost:3172 packages=all

2. If the project properties file adds additional targets using the **sender-.sl.rtview.cache.config** property, change it to use the new **sender-.sl.rtvapm.dataxfr.target** property using the URL you specified for the **$rtvAgentTarget** and a new unique ID. For example:

sender.sl.rtview.cache.config=pck_rtvagent_sender.rtv $rtvAgentTarget:'otherhost:3172'

would be changed to

sender.sl.rtvapm.dataxfr.target=id=target2 url=otherhost:3172 packages=all

If your project properties file did not overwrite either of the above, the default sender/receiver properties values were used and therefore no changes are needed.

6.4

No upgrade steps required.

6.3

The **Active** and **DeltaCreated** data type columns in the BW_PROCESSES table were converted to real numbers to account for the loss of resolution when compaction is taking place by averaging the metrics. To synchronize cache data with data in the database, use ALTER TABLE SQL as show below for your supported DB platform (Oracle is not required):

- DB2
- SQL Server

- MySQL

DB2

ALTER TABLE "BW_PROCESSES"

ALTER COLUMN "Active" SET DATA TYPE DOUBLE;

ALTER TABLE "BW_PROCESSES"

ALTER COLUMN "DeltaCreated" SET DATA TYPE DOUBLE;

ALTER TABLE "BW_PROCESS_TOTALS"

ALTER COLUMN "Active" SET DATA TYPE DOUBLE;

SQL Server

ALTER TABLE [BW_PROCESSES]

ALTER COLUMN [Active] FLOAT

ALTER TABLE [BW_PROCESSES]

ALTER COLUMN [DeltaCreated] FLOAT

ALTER TABLE [BW_PROCESS_TOTALS]

ALTER COLUMN [Active] FLOAT

MySQL

ALTER TABLE "BW_PROCESSES"

MODIFY "Active" DOUBLE ,

MODIFY "DeltaCreated" DOUBLE ;

ALTER TABLE "BW_PROCESS_TOTALS"

MODIFY "Active" DOUBLE ;

Installation

The Monitor can be installed as a stand-alone monitoring system for technical support teams. The Monitor can also be installed as a Solution Package within the RTView© Enterprise product.

Download the **TIB_rtview-bw_<version>.zip** file and unzip the **TIB_rtview-bw_<version>.zip** file into a directory of your choosing.

For more information about RTView Enterprise see the RTView© Enterprise User's Guide, available on the SL Product Documentation website.

See Quick Start for more information on how to install the Monitor.

File Extraction Considerations

On Windows systems, using the extraction wizard of some compression utilities might result in an extra top-level directory level based on the name of the .zip file. The additional directory is not needed because the .zip files already contain the top-level directory. This extra directory must be removed before clicking the Next button that performs the final decompression.

On UNIX/Linux systems, use the -a option to properly extract text files.

Setup

This section describes how to setup your system for the Monitor. This section includes:

- Enabling Login in the Monitor: This is optional.
- Enabling Monitoring in TIBCO BusinessWorks

Enabling Login in the Monitor

Note: The following only enables login in the Display Server (Classic) version of the User Interface. The following steps do not enable login for the HTML User Interface, which has login enabled by default.

By default, "Login" is disabled for BW Monitor, which means that logging in will not be required when initially accessing BW Monitor. To enable RTView Role Based Security, follow the instructions below. This will enable the following users and their associated passwords:

rtvadmin (password: rtvadmin)

admin (password: admin)

rtvuser (password: rtvuser)

user (password: user)

To define your own users and roles or to integrate with LDAP or other user and security management systems, see Role-based Security in the RTView Core User's Guide at http://www.sl.com/support/documentation.

To enable Login, perform the following steps:

1. Navigate to **TIB_rtview-bw/projects/rtview-server**.

2. Extract the **rtvdisplay.properties** file from the **rtview-bwmon-classic.war** file by typing:

   jar -xf rtview-bwmon-classic.war WEB-INF/classes/gmsjsp/rtvdisplay.properties

   The **WEB-INF/classes/gmsjsp** directory path is created, which contains the **rtvdisplay.properties** file.

3. Edit the **WEB-INF/classes/gmsjsp/rtvdisplay.properties** file, set **LoginEnabled=true**, and save your changes.

4. Update the **rtview-bwmon-classic.war** file with your changes by typing:

   jar -uf rtview-bwmon-classic.war WEB-INF/classes/gmsjsp/rtvdisplay.properties

5. If you are using Tomcat as your application server, copy the **TIB_rtview-bw/projects/rtview-server/rtview-bwmon-classic.war** file to the Tomcat **webapps** directory. If you are using Eclipse Jetty as your application server (which is delivered with BW Monitor), there are no further steps.

Enabling Monitoring in TIBCO BusinessWorks

Some setup is required in TIBCO BusinessWorks Versions 5 and 6 to enable monitoring. For Version 6, there are two options: You can either install the OSGI plugin in every AppSpace you want to monitor, or you can enable the TIBCO Hawk MicroAgent in each AppSpace you want to monitor. Monitoring via the OSGI plugin is recommended for better performance and reliability.

For Version 5, monitoring must be done via TIBCO Hawk, which is enabled by default, but you also need to install the RTViewBWAgent plug-in microagent. If you enable JMX monitoring, you can access additional data such as CPU and Memory data. For BWSE engines, additional setup is needed to allow access to AMX Node data.

This section contains:

- Enable Monitoring via OSGi Plugin for Version 6
- Enable Monitoring via TIBCO Hawk for Versions 5 and 6
- Enable Monitoring via JMX for Version 5
- Enable Monitoring of BWSE Engines for Version 5

Enable Monitoring via OSGi Plugin for Version 6

This section describes the Enabling Monitoring in TIBCO BusinessWorks and steps required to install the RTView OSGI (Open Service Gateway Initiative) BusinessWorks plugin into TIBCO BusinessWorks, BWCE: Docker, and BWCE: Cloud Foundry.

Configuration

The RTView OSGi plugin is configured with the following properties:

Property	Definition
sl.rtview.rtvagent.name	This is the name by which the plugin will be identified in the RTView displays. It should be unique and should identify the sender. For example: **example: sl.rtview.rtvagent.name=hostname** **default: updater**
sl.rtview.rtvagent.target	This is the host and port of the receiving RTView dataserver. For example: **sl.rtview.rtvagent.target=receiver:3272** **default: localhost:3272** You can send to multiple receivers using the plugin property **sl.rtview.rtvagent.target**. Specify a comma-separated list of receivers, in which case the plugin will send to multiple targets. For example: **sl.rtview.rtvagent.target=receiver1:3272,receiver2:3272,receiver3:4272**
sl.rtview.rtvagent.update	This is how often the plugin sends data, in seconds. For example:

Property	Definition
	sl.rtview.rtvagent.update=30 **default: 10**
sl.rtview.bw.activities.enabled	This enables BW6 activity metrics to be collected. For example: **sl.rtview.bw.activities.enabled=true** **default: false**

Note that the receiving RTView data server must be run using the **-propfilter:receiver** flag. This is the default for all RTView distributions except RTView data collector. If you are using that distribution, you should add the flag to your **start_servers** command. For example:

start_servers.sh -propfilter:receiver

Containers

With the container deployments you may also use the following properties to provide specific names for the domain, appspace, and appnode.

sl.rtview.bw.domain

sl.rtview.bw.appspace

sl.rtview.bw.appnode

For example:

sl.rtview.bw.domain=Containers

sl.rtview.bw.appspace=Docker

sl.rtview.bw.appnode=docker-1

If you supply specific names, then your containers will appear in the displays as if they were appnodes in an appspace. Furthermore, if multiple containers are instances of the same application, the metrics of the container appnodes will be summed for the application, as if it were deployed to the appspace. If you don't supply specific names, the following names will be used:

domain: "standalone"

appspace: same as appnode

appnode: container ID (Docker) or container GUID (PCF)

If you do supply specific names you can include variables from the container environment in the names, by bracketing the environment variable name with '**%**'.

For example, PCF containers have environment variables such as **CF_INSTANCE_GUID** and **CF_INSTANCE_INDEX** (the former is used for the appnode name if one is not supplied). If you wanted to name your appnodes "instance-1", "instance-2", and so on, then you could supply an appnode name such as:

sl.rtview.bw.appnode=instance-%CF_INSTANCE_INDEX%

TIBCO BusinessWorks

Prerequisites

BW 6.3 or later; Hotfix 2 or later

RTVAPM X.X

To enable the plugin in RTView® TIBCO® BusinessWorks™, perform the following:

1. Stop all AppSpaces that are to be monitored and stop the bwagent, if running.
2. Copy the plugin to **$TIBCO_HOME/bw/6.x/system/shared**.
3. Start the bwagent. Do not start the AppNode(s) or AppSpace(s) yet.

   Note: Continue to step 4 if you want to modify the default property values. You may skip steps 4-7 entirely if you want to use the default property values.

4. Navigate to the system **config** folder: **$TIBCO_HOME/bw/6.x./config/**.
   $TIBCO_HOME/bw/6.x/domains/<domain>/appnodes/<appspace>/<appnode>/config
5. Copy the **appspace_config.ini_template** file to a temporary file.
6. Edit the temporary file to include the **bw.appnode.clean.config.folder.on.startup=true** property and set the properties provided above under Configuration as needed.

   Note: You can send to multiple receivers using the plugin property **sl.rtview.rtvagent.target**. Specify a comma-separated list of receivers, in which case the plugin will send to multiple targets. For example: **sl.rtview.rtvagent.target=receiver1:3272,receiver2:3272,receiver3:4272**.

7. Use the **bwadmin config** command to push the configuration to the AppSpace:
   bwadmin config -d <domain> -cf <temporary-file> appspace <appspace>
8. (Optional) For any AppNode, to see plugin output (INFO or TRACE) in the **bwappnode.log** file, add the following to the "user properties" section of the **logback.xml** file (**$TIBCO_HOME/bw/<version>/domains/<domain>/appnodes/<appspace>/<appnode>/logback.xml**):

   ```
   <logger name="com.sl.rtvapm.osgi">
   <level value="INFO"/>
   </logger>
   ```

   Note: When debugging the behavior of the plugin, it is recommended to set level value to "TRACE."

9. Restart the AppSpace(s).
10. Repeat steps 6-9 for every AppSpace to be monitored.

BWCE: Docker

Prerequisites

BWCE 2.4.2 or later

RTVAPM X.X

You will build your Docker image using your application with the TIBCO base image and including the plugin jar. In the example below, we use **tibco.bwce.sample.BookStore** as the application:

1. Download **bwce-runtime-<version>.zip** from TIBCO and copy it into your BWCE installation at the following location:

   $TIBCO_HOME/bwce/<version>/docker/resources/bwce-runtime

2. Build the image:

   cd $TIBCO_HOME/bwce/<version>/docker

   docker build -t tibco/bwce:latest .

3. Add the RTView OSGi plugin to your application by copying the application archive and the plugin jar (**com.sl.rtvapm.osgi_<version>.jar**) to your project directory.

4. Create a Docker file containing:

   FROM tibco/bwce:latest

   MAINTAINER Tibco

   ADD tibco.bwce.sample.BookStore.ear /

   RUN mkdir -p /resources/addons/jars

   ADD com.sl.rtvapm.osgi_<version>.jar /resources/addons/jars

   EXPOSE 8080

5. Build the application:

   docker build -t tibco/bookstore:latest .

6. Set the plugin properties described above using Java system properties, **BW_JAVA_OPTS**, and the Docker run command. For example:

   **OPTS="-Dsl.rtview.rtvagent.target=localhost:3272 **

   **-Dsl.rtview.rtvagent.name=<hostname> **

   **-Dsl.rtview.bw.domain=Containers **

   -Dsl.rtview.bw.appspace=Docker"

   **docker run -i **

   **-e ... **

   **-e BW_JAVA_OPTS="$OPTS" **

   **-e ... **

   tibco/bookstore:latest

7. The plugin logs to the console at levels INFO and TRACE. You can set the log level for all packages with BW_LOGLEVEL. For example:

   -e BW_LOGLEVEL=INFO

BWCE: Cloud Foundry

Prerequisites

BWCE 2.4.2 or later

RTVAPM X.X

To enable the plugin in Cloud Foundry, perform the following:

1. Download the **bwce-buildpack_cf-<version>.zip** file from TIBCO.

2. Insert the plugin jar into the zip file at **/resources/addons/jars**. To insert plugin jar in the buildpack zip, proceed as follows on a unix system:

   - make a copy of the buildpack zip. For example: **bwce-buildpack_cf_rtv.zip**

   - make the directories **./bw-buildpack/resources/addons/jars** and put the plugin jar inthe **jars** directory.

   - update the buildpack zip: **zip -u -r bwce-buildpack_cf_rtv.zip bw-buildpack**

   - remove **./bw_buildpack...**

3. Upload the buildpack to your cloud. For example:

   cf create-buildpack bw-buildpack bwce-buildpack_cf-rtv.zip 1

4. Set the plugin properties described above using Java system properties and **BW_JAVA_OPTS** in the **manifest.yml** file. For example:

   . . .

   env:

   . . .

   BW_JAVA_OPTS: '-Dsl.rtview.rtvagent.target=<receiver>:3272 -Dsl.rtview.rtvagent.name=<hostname> -Dsl.rtview.bw.domain=Containers -Dsl.rtview.bw.appspace=PCF'

   . . .

5. Push the application. For example:

   cf push

Enable Monitoring via TIBCO Hawk for Versions 5 and 6

See the appropriate instructions:

- TIBCO BusinessWorks Version 6
- TIBCO BusinessWorks Version 5

TIBCO BusinessWorks Version 6

Perform these instructions if you are monitoring BusinessWorks version 6:

1. Enable your applications for statistics collection. You can do this using the TIBCO BusinessWorks administrator CLI with commands such as:

   bwadmin enablestats -d MyDomain -a MyAppSpace process MyAppName MyAppVersion

   Repeat for each application you wish to monitor.

2. Enable the Hawk MicroAgent in your AppNodes for each AppSpace you wish to monitor. Refer to the **Enabling TIBCO Hawk MicroAgent** section of the *TIBCO BusinessWorks6 Administration Guide*.

TIBCO BusinessWorks Version 5

This section is for BusinessWorks Version 5 users.

> **Note:** This section does not apply if all your engines are deployed as BusinessWorks Service Engines (BWSE).

Install the RTViewBWAgent plug-in microagent in the Hawk Agent for each domain you have configured to communicate with the Monitor.

RTViewBWAgent detects deployed engines and gets their maximum heap size metrics when the Hawk agent is started. If RTViewBWAgent is not installed, deployed engines are not detected until they have been started and report data to the Monitor. When live data is received the engine is added and its **Status** is set to **LIMITED**. The **Status** remains **LIMITED** because, although live data metrics are available, the deployment and maximum heap size metrics are still unavailable.

Note: After installation, you can use the Hawk Display to view the RTViewBWAgent microagent and invoke its methods: GetBWDeploymentNames and GetBWDeploymentMaxHeapSizes.

You can also configure the agent to detect deployed engines and make data updates at more frequent, specified intervals. To specify the update interval you uncomment the **-update** argument in the **BWAgentPlugin.hma** file and specify a non-zero value. When the **-update** argument is not used (is commented out), the Monitor does not report that an engine has been deployed or undeployed until the Hawk agent is restarted.

1. Navigate to the **agents/BWAgentPlugin** directory of your Monitor installation and locate the following two files:
   - **BWAgentPlugin.jar**
   - **BWAgentPlugin.hma**

2. For a given domain, find the plug-in directory via this path:
   <TIBCO-home>/tra/domain/<domain-name>

3. Repeat Step 2 for each Hawk domain you have configured to communicate with the Monitor.

4. To (optionally) set RTViewBWAgent to make data updates at more frequent, specified intervals, open the **BWAgentPlugin.hma** file, uncomment the **-update** argument and specify a non-zero value. The value, which defaults to 300, represents the update interval in seconds. For example, a value of **3600** updates every hour:

```
<arguments>
      <arg>-update:3600</arg>
      . .
</arguments>
```

5. Copy the **BWAgentPlugin.jar** file and **BWAgentPlugin.hma** file into the plug-in directory and restart the Hawk Agent.

Enable Monitoring via JMX for Version 5

TIBCO BusinessWorks version 5 engines can also be enabled for JMX monitoring as documented in *TIBCO BusinessWorks™ Administration, Monitoring the BusinessWorks Engine Using JMX*:

To enable monitoring of BW5 engines via JMX:

1. To enable local JMX monitoring, add the following properties to **bwengine.tra**:

Jmx.Enabled=true
java.property.com.sun.management.jmxremote=true

2. To enable remote JMX monitoring, add the following properties to **bwengine.tra**: (Note <**port_number**> can be any available port)

java.property.com.sun.management.jmxremote.port=<port_number>
java.property.com.sun.management.jmxremote.authenticate=false
java.property.com.sun.management.jmxremote.ssl=false

For example, the BW Engine **MyDomain.MyApp.Procs** can be enabled for remote JMX monitoring by adding the following lines to the file

C:\Tibco\tra\domain\MyDomain\application\MyApp\MyApp-Procs.tra:

```
#
# Enable JMX on port 9000
#
Jmx.Enabled=true
java.property.com.sun.management.jmxremote=true
java.property.com.sun.management.jmxremote.port=9000
java.property.com.sun.management.jmxremote.authenticate=false
java.property.com.sun.management.jmxremote.ssl=false
```

3. After the BW Engine is enabled for JMX monitoring and restarted, it can be monitored by adding a JMX Connection property in the RTView Configuration Application where the **Connection** name is the Engine name. See To configure RTView Manager: for more information.

Enable Monitoring of BWSE Engines for Version 5

This section is for TIBCO ActiveMatrix (AMX) users, and describes how to configure BW Monitor to monitor BWSE engines. BW Monitor needs access to AMX Node data stored in EMS message queues on the AMX Host system. To make this data available to BW Monitor you will create EMS topics with bridges from the queues.

The TIBCO BusinessWorks Service Engine (BWSE) is an ActiveMatrix (AMX) component that enables BW engines to participate in the implementation of AMX services. In this case, the BWSE engines run within an AMX Node and are not visible to BW Monitor. However, you can configure BW Monitor to display these engines, as well as to gather JVM memory metrics for the AMX Nodes in which they are running.

To Configure for BWSE engines:

1. To configure the AMX Host, execute the following commands in the EMS administration tool (tibemsadmin):

   create topic rtv.amx.governance.stats

create bridge source=queue:amx.governance.stats target=topic:rtv.amx.governance.stats

2. In AMX Administrator, in the properties for each BWSE engine, set HawkEnabled to **true**. See BWSE Components for limitations for BWSE components.

CHAPTER 3 Configuration

This chapter describes how to configure the Monitor components. Most sections in this chapter use the RTView Configuration Application for setting up their various configurations. See RTView Configuration Application for more information. This section includes:

Overview

This section describes how to configure the Monitor.

Basic Steps

Some of the configuration steps described here are required (where noted) and others are optional.

- Configure Data Collection: (**Required**) Define the TIBCO Servers and destinations to be monitored. This step must be performed before running any deployment of the Monitor. This section also describes configuring historical data collection, which is not required.
- Configure Ports: (**Required**) Configure the RTView Servers.
- Configure the Database: (Optional) Configure a production database.
- Configure Alert Notifications: (Optional) Configure alerts to execute an automated action (for example, to send an email alert).
- Configure High Availability: (Optional) Configure high availability components.

Assumptions

This document assumes that you:

- installed the Monitor per instructions in Installation.
- completed instructions in Setup.

Initializing a Command Prompt or Terminal Window

To start any RTView process (Data Server, Historian, and so forth), you must first initialize a command line window on the host. Unix scripts are Bourne shell compatible.

To initialize a command line window, execute the **rtvapm_init** script. For example:

Windows

Go to your Monitor installation directory and type:

rtvapm_init

UNIX

The script used to initialize a terminal window depends on whether you are in csh or rsh (e.g. Linux, Mac OS X). With a Bourne shell, open a terminal window, go to your Monitor installation directory and type:

. ./rtvapm_init.sh

Configure Data Collection

This section describes how to configure the data source connections for each TIBCO BusinessWorks component that you want to monitor.

For most installations, the default Monitor property settings are sufficient. Consult Technical Support before modifying other configurations to avoid upgrade issues.

At this point you have:

- Verified System Requirements.
- Performed Monitor Quick Start
- Performed Setup instructions

This section includes:

- Configure for Hawk (for BW5 and BW): Define the classpaths for TIBCO jar files and various connections for TIBCO BusinessWorks.
- Configure for RTView Manager: Configure data collection and historical data collection for RTView Manager.
- Configure for TIBCO BusinessWorks: Configure data collection and historical data collection for TIBCO BusinessWorks.
- Configure for TIBCO BusinessWorks 5: Configure data collection and historical data collection for TIBCO BusinessWorks Version 5.

Configure for Hawk (for BW5 and BW)

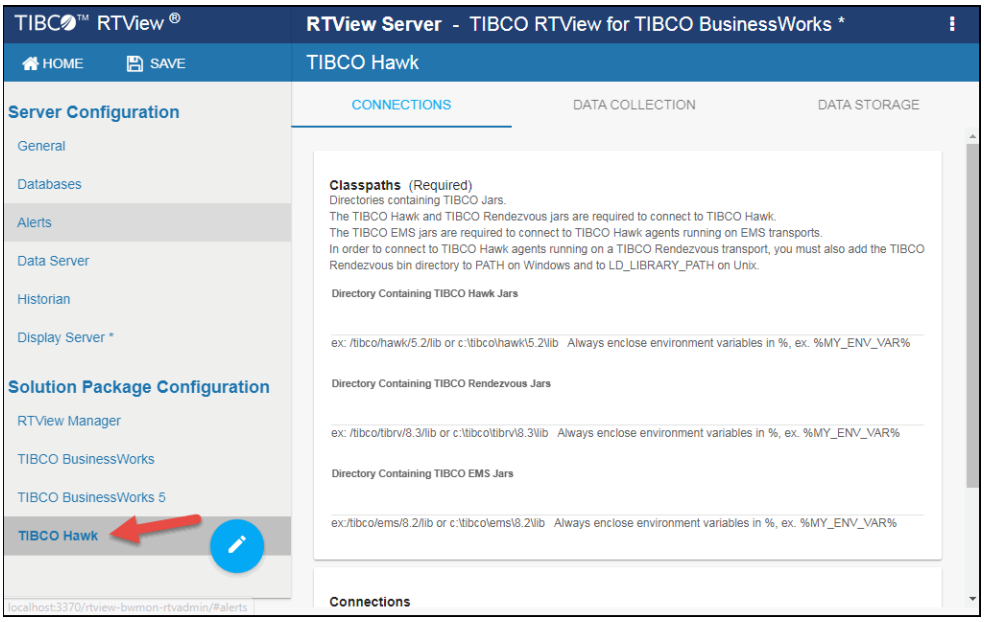
This section contains:

- Configuring Data Collection for Hawk: Configure data collection in Hawk that is required for TIBCO BusinessWorks and TIBCO BusinessWorks 5.

Configuring Data Collection for Hawk

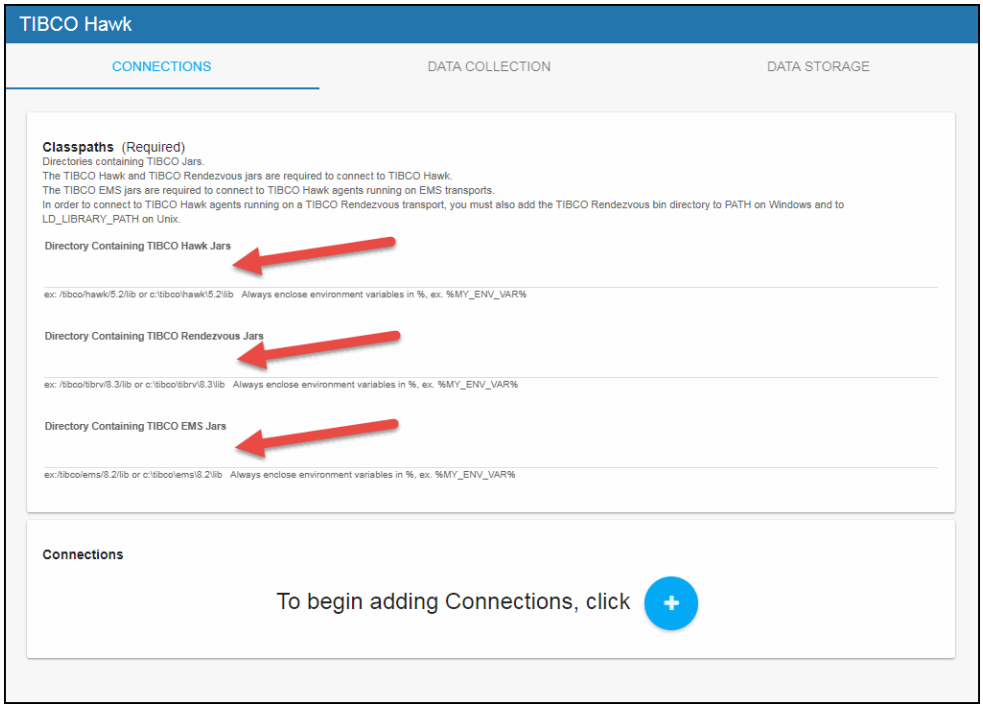
Note: Only the **Classpath** and **Connections** regions on the **CONNECTIONS** tab need to be set up for TIBCO BusinessWorks (Version 5 or 6). The **DATA COLLECTION** and **DATA STORAGE** tabs do not need to be configured.

1. Navigate to the RTView Configuration Application > **(Project Name)** > **Solution Package Configuration** > **TIBCO Hawk** > **CONNECTIONS** tab.



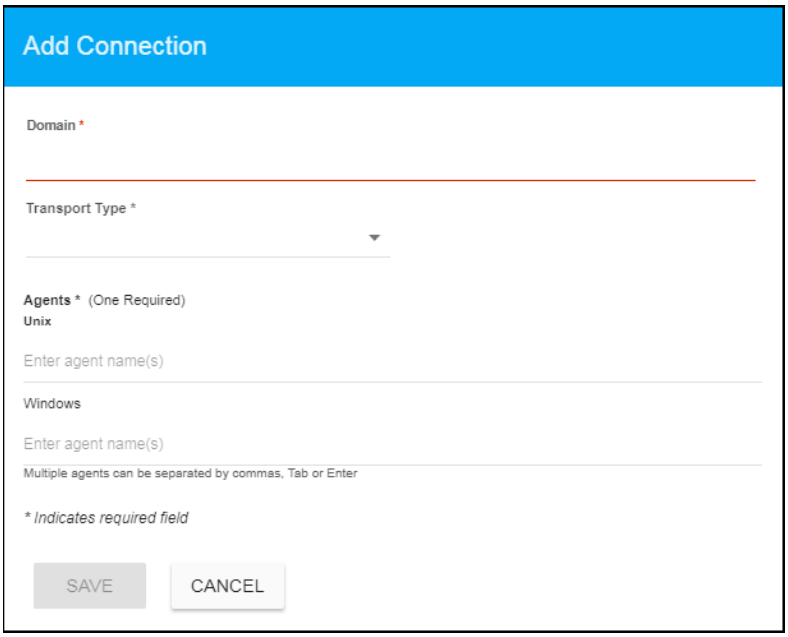
2. In the **CONNECTIONS** tab, specify the classpaths for the TIBCO Hawk jar files, the TIBCO Rendezvous jar files, and the TIBCO EMS jar files.

Note: If using Version 6 and you have installed the OSGI plugin, TIBCO Hawk connections are not needed. Skip to Step 5.

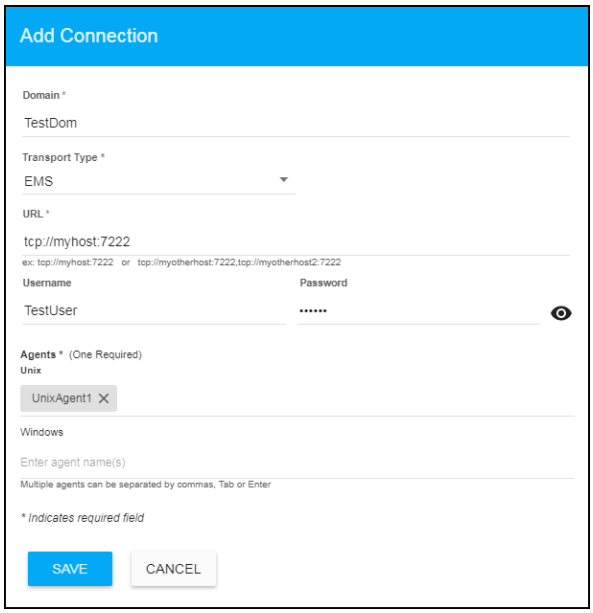


3. Click the + icon.

The **Add Connection** dialog displays.



4. For TIBCO Hawk domains running on **EMS** transports, specify the connection information and click **Save** where:



Domain: Enter the name of the domain.

Transport Type: Select **EMS** from this drop down list.

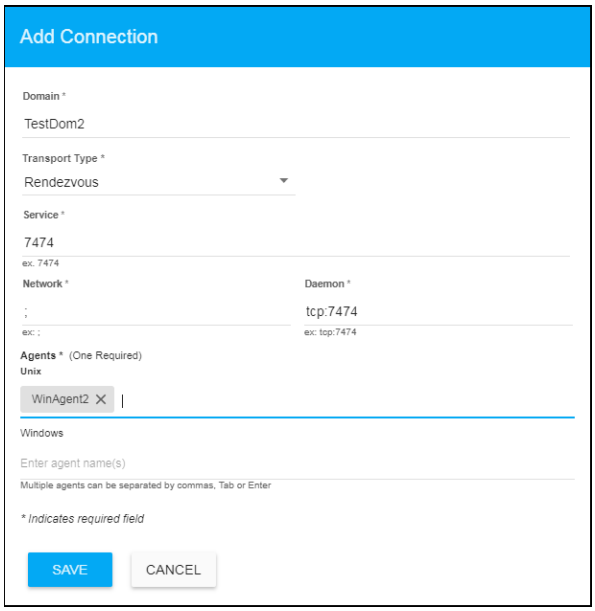
URL: Enter the complete URL for the EMS connection.

Username: The username is used when creating the EMS connection. This field is optional.

Password: This password is used when creating the EMS connection. This field is optional. By default, the password entered is hidden. Click the icon to view the password text.

Agents: Enter the associated Unix/Windows agents. The agent name displays in the field after entering the name and typing a comma or by clicking the Tab or Enter key. You can enter more than one agent in the fields. Once the agent is specified, you can delete the agent by clicking the **X** next to their name.

For TIBCO Hawk domains running on **Rendezvous** transports, specify the connection information and click **Save** where:



Domain: Enter the name of the domain.

Transport Type: Select **Rendezvous** from this drop down list.

Service: Enter the Service for the Rendezvous connection.

Network: Enter the Network for the Rendezvous connection.

Daemon: Enter the Daemon for the Rendezvous connection.

Agents: Enter the associated Unix/Windows agents. The agent name displays in the field after entering the name and typing a comma or by clicking the Tab or Enter key. You can enter more than one agent in the fields. Once the agent is specified, you can delete the agent by clicking the **X** next to their name.

Note: After you complete these configuration steps and start the RTView Data Server, you can verify your Hawk configuration by viewing the **dataserver.log** file, located in the **projects/rtview-server/logs** directory. For example:

```
2013-05-08 13:39:48,009 INFO rtv_stdout - [rtview] ... AppMgr.initApp

2013-05-08 13:39:48,009 INFO rtv_stdout - [rtview] ... BWMON Manager AppMgr.initApp

2013-05-08 13:39:48,025 INFO rtv_stdout - [rtview] ... using filters file <bwmon_
filters.xml>

2013-05-08 13:39:49,056 INFO rtv_stdout - [rtview] ... startApplication()

2013-05-08 13:39:49,056 INFO rtv_stdout - [rtview] ... startApplication()

2013-05-08 13:39:49,056 INFO rtv_stdout - [rtview] ----------------------------
```

```
2013-05-08 13:39:49,056 INFO rtv_stdout - [rtview] Group: WIN_AGENTS
2013-05-08 13:39:49,056 INFO rtv_stdout - [rtview] Agent: demo1(domain1)
2013-05-08 13:39:49,056 INFO rtv_stdout - [rtview] Agent: demo2(domain1)
2013-05-08 13:39:49,056 INFO rtv_stdout - [rtview] Agent: demo3(domain1)
2013-05-08 13:39:49,056 INFO rtv_stdout - [rtview] ----------------------------
2013-05-08 1339:49,056 INFO rtv_stdout - [rtview] ----------------------------
2013-05-08 13:39:49,056 INFO rtv_stdout - [rtview] Group: UNIX_AGENTS
2013-05-08 13:39:49,072 INFO rtv_stdout - [rtview] Agent: demo4(domain2)
2013-05-08 13:39:49,072 INFO rtv_stdout - [rtview] Agent: demo5(domain2)
2013-05-08 13:39:49,072 INFO rtv_stdout - [rtview] Agent: demo6(domain2)
2013-05-08 13:39:49,072 INFO rtv_stdout - [rtview] ----------------------------
```

5. **SAVE** your changes in the RTView Configuration Application (upper left-hand corner), and then stop and restart your project using the **RESTART SERVERS** button (in the upper right-hand corner of the RTView Configuration Application).

Configure for RTView Manager

The configuration defined in this section is only relevant for Version 5 users that have enabled JMX. Version 6 users do not need to complete this section.

Note: This is also only relevant if you are using the Classic (non-HTML) displays.

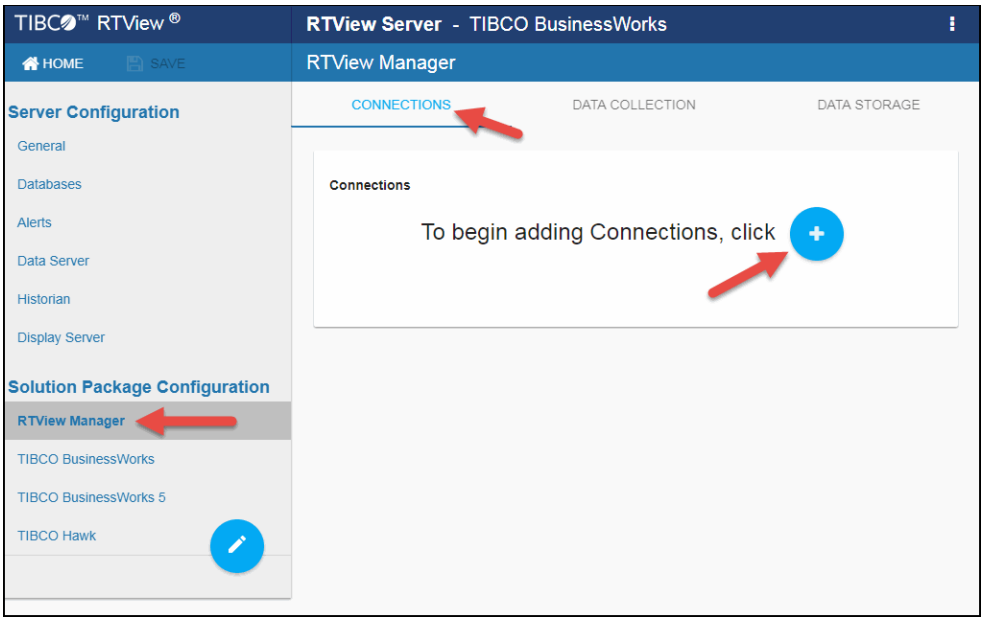
This section contains:

- Configuring Data Collection for RTView Manager: Defines the steps required to configure data collection in the RTView Configuration Application for RTView Manager. (Required)
- Configuring Historical Data for RTView Manager: Describes the steps required to configure historical data collection for RTView Manager. (Optional)

Configuring Data Collection for RTView Manager

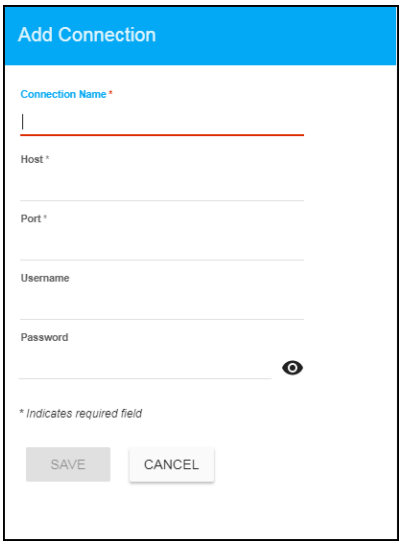
To configure RTView Manager:

1. Navigate to the RTView Configuration Application > **(Project Name)** > **Solution Package Configuration** > **RTView Manager** > **CONNECTIONS** tab.



2. Click the + icon.

The **Add Connection** dialog displays.



3. Add a connection for each BusinessWorks engine for which you enabled monitoring via JMX. See Enable Monitoring via JMX for Version 5 for more information. JMX data is available in the Monitor in the RTView Servers view.

Connection Name: Use the Engine name for the **Connection Name** field.

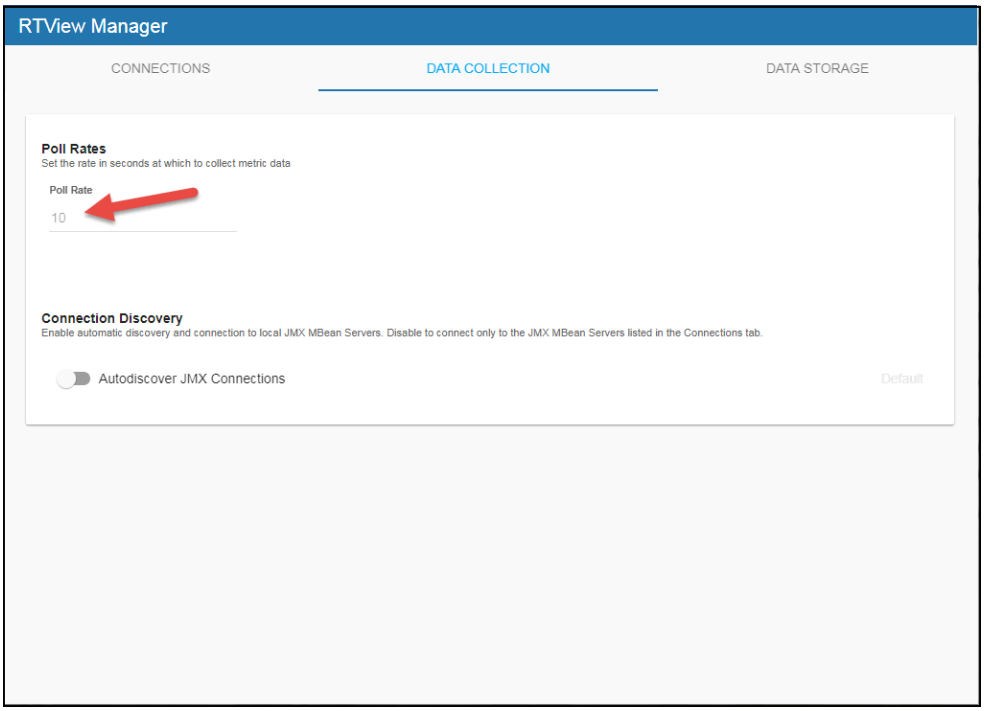
Host: Enter the host for the engine.

Port: Enter the Port specified in your **.tra** file. JMX data is available in the Monitor in the RTView Servers view.

Username: The username is used when creating the connection. This field is optional.

Password: This password is used when creating the connection. This field is optional. By default, the password entered is hidden. Click the 👁 icon to view the password text.

4. Navigate to the RTView Configuration Application > **(Project Name)** > **Solution Package Configuration** > **RTView Manager** > **DATA COLLECTION** > **Poll Rates** to update the default polling rates for all RTView Manager caches.



5. **SAVE** your changes in the RTView Configuration Application (upper left-hand corner), and then stop and restart your project by clicking the **RESTART SERVERS** button (which appears after you save your changes) in the upper right-hand corner.

Configuring Historical Data for RTView Manager

You can specify the number of history rows to store in memory, the compaction rules, the duration before metrics are expired and deleted, and the different types of metrics that you want the Historian to store in the **Data Storage** tab in the RTView Configuration Application. This section contains the following:

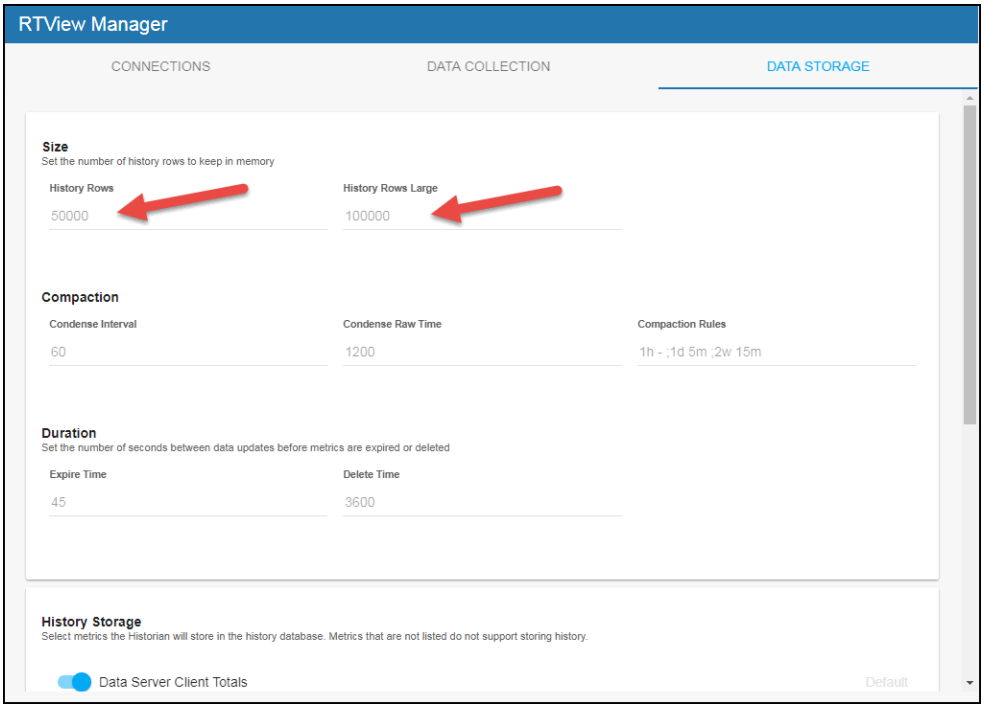
- Defining the Storage of In Memory RTView Manager History
- Defining Compaction Rules for RTView Manager
- Defining Expiration and Deletion Duration for RTView Manager Metrics
- Enabling/Disabling Storage of Historical Data for RTView Manager
- Defining a Prefix for All History Table Names for RTView Manager Metrics

Defining the Storage of In Memory RTView Manager History

You can modify the maximum number of history rows to store in memory in the **Data Storage** tab. The **History Rows** property defines the maximum number of rows to store for the JvmGcInfo, JvmMemoryPool, RtvDataServerManager, RtvDisplayServerManager, RtvDataServerClientTotals, TomcatGlobalRequestStats, and TomcatWebModuleTotals caches. The **History Rows Large** property defines the maximum number of rows to store for the JvmOperatingSystem, JvmThreading, JvmMemory, RtvDataServerClientStats, and TomcatWebModuleStats caches. The default setting for **History Rows** is 50,000 and the default setting for **History Rows Large** is 100,000. To update the default settings:

1. Navigate to the RTView Configuration Application > (**Project Name**) > **Solution Package Configuration** > **RTView Manager** > **DATA STORAGE** tab.

2. In the **Size** region, click the **History Rows** and **History Rows Large** fields and specify the desired number of rows.



Defining Compaction Rules for RTView Manager

Data compaction, essentially, is taking large quantities of data and condensing it using a defined rule so that you store a reasonably sized sample of data instead of all of your data, thus preventing you from potentially overloading your database. The available fields are:

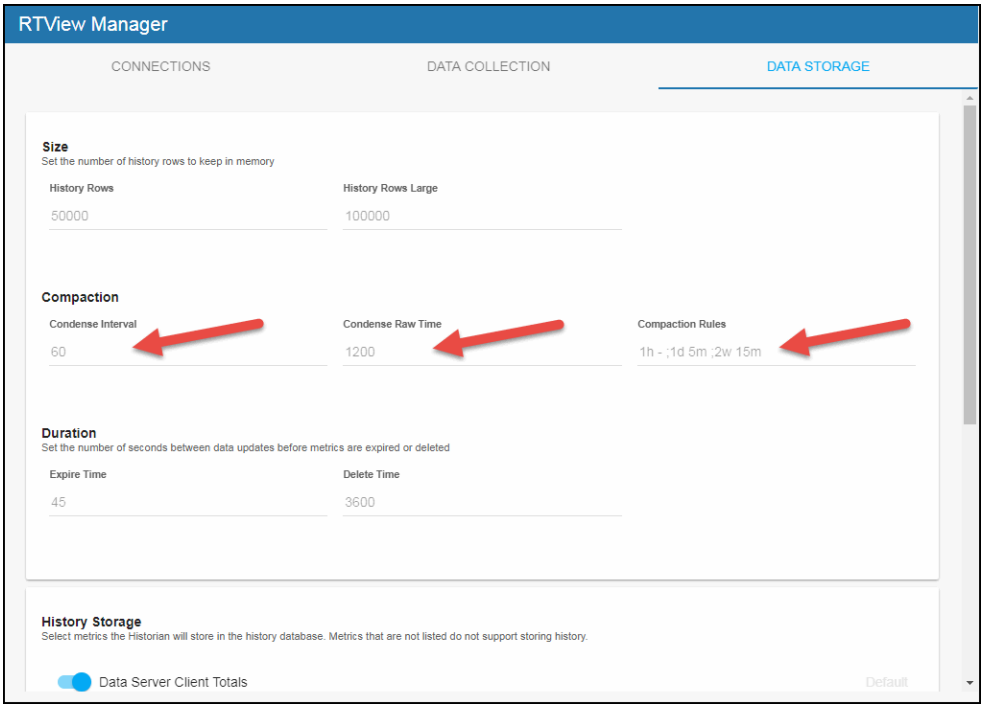
- **Condense Interval** -- The time interval at which the cache history is condensed. The default is 60 seconds. The following caches are impacted by this setting: JvmGcInfo, JvmMemoryPool, JvmOperatingSystem, JvmThreading, JvmMemory, RtvDataServerManager, and RtvDataServerClientTotals.
- **Condense Raw Time** -- The time span of raw data kept in the cache history table. The default is 1200 seconds. The following caches are impacted by this setting: JvmGcInfo, JvmMemoryPool, JvmOperatingSystem, JvmThreading, JvmMemory,

RtvDataServerManager, RtvDataServerClientTotals, TomcatWebModuleStats, TomcatGlobalRequestStats, and TomcatWebModuleTotals.

- **Compaction Rules** -- This field defines the rules used to condense your historical data in the database. By default, the columns kept in history will be aggregated by averaging rows with the following rule 1h -;1d 5m;2w 15m, which means the data from 1 hour will not be aggregated (1h - rule), the data over a period of 1 day will be aggregated every 5 minutes (1d 5m rule), and the data over a period of 2 weeks old will be aggregated every 15 minutes (2w 15m rule). The following caches are impacted by this setting: JvmOperatingSystem, JvmThreading, JvmMemory, RtvDataServerManager, RtvDataServerClientTotals, TomcatWebModuleStats, TomcatGlobalRequestStats, and TomcatWebModuleTotals.

1. Navigate to the RTView Configuration Application > (**Project Name**) > **Solution Package Configuration** > **RTView Manager** > **DATA STORAGE** tab.

2. In the **Compaction** region, click the **Condense Interval**, **Condense Raw Time**, and **Compaction Rules** fields and specify the desired settings.



Defining Expiration and Deletion Duration for RTView Manager Metrics

The data for each metric is stored in a specific cache and, when the data is not updated in a certain period of time, that data will either be marked as expired or, if it has been an extended period of time, it will be deleted from the cache altogether.

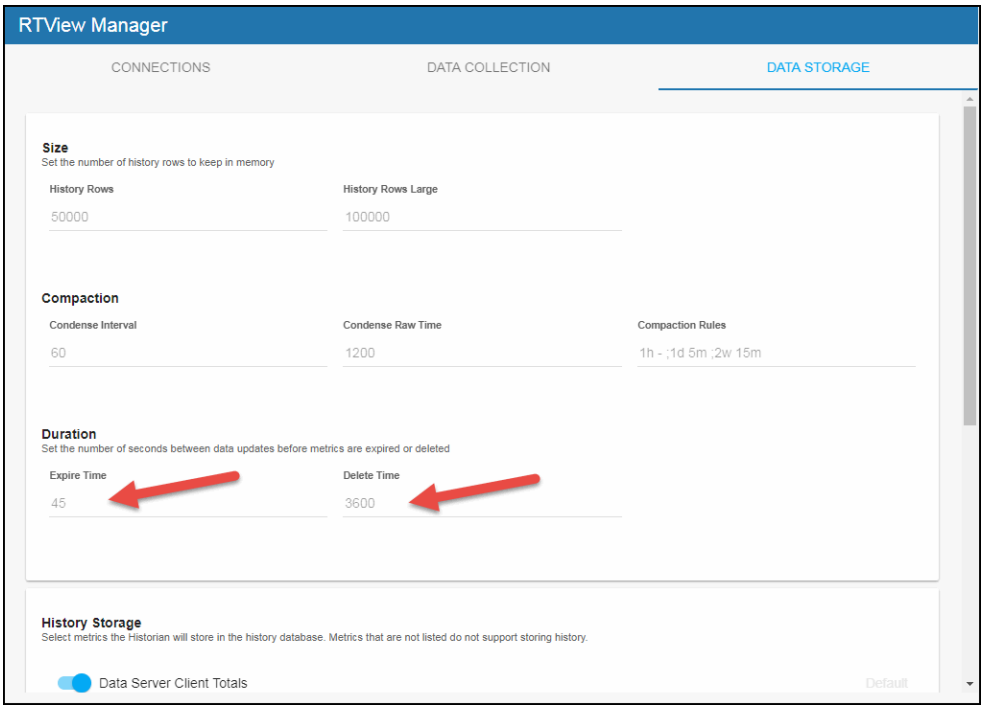
The **Expire Time** field, which sets the expire time for the JvmConnections, JvmGcInfo, JvmMemoryPool, JvmClassLoading, JvmCompilation, JvmOperatingSystem, JvmThreading, JvmMemory, JvmMemoryManager, JvmSystemProperties, RtvDataServerManager, RtvDisplayServerManager, RtvHistorianManager, RtvDataServerClientStats, RtvDataServerClientTotals, RtvServerVersion, TomcatWebModuleStats,

TomcatConnectorInfo, TomcatGlobalRequestStats, TomcatHostInfo, and TomcatWebModuleTotals caches, defaults to 45 seconds.

The **Delete Time**, which sets the delete time for the JvmConnections, JvmGcInfo, JvmMemoryPool, JvmClassLoading, JvmCompilation, JvmOperatingSystem, JvmRuntime, JvmThreading, JvmMemory, JvmMemoryManager, JvmSystemProperties, RtvDataServerManager, RtvDisplayServerManager, TomcatWebModuleStats, TomcatGlobalRequestStats, TomcatWebModuleTotals, RtvHistorianManager, RtvDataServerClientStats, RtvDataServerClientTotals, RtvServerVersion, TomcatWebModuleStats, TomcatConnectorInfo, TomcatGlobalRequestStats, TomcatHostInfo, and TomcatWebModuleTotals caches, defaults to 3600 seconds. To modify these defaults:

1. Navigate to the RTView Configuration Application > (**Project Name**) > **Solution Package Configuration** > **RTView Manager** > **DATA STORAGE** tab.

2. In the **Duration** region, click the **Expire Time** and **Delete Time** fields and specify the desired settings.

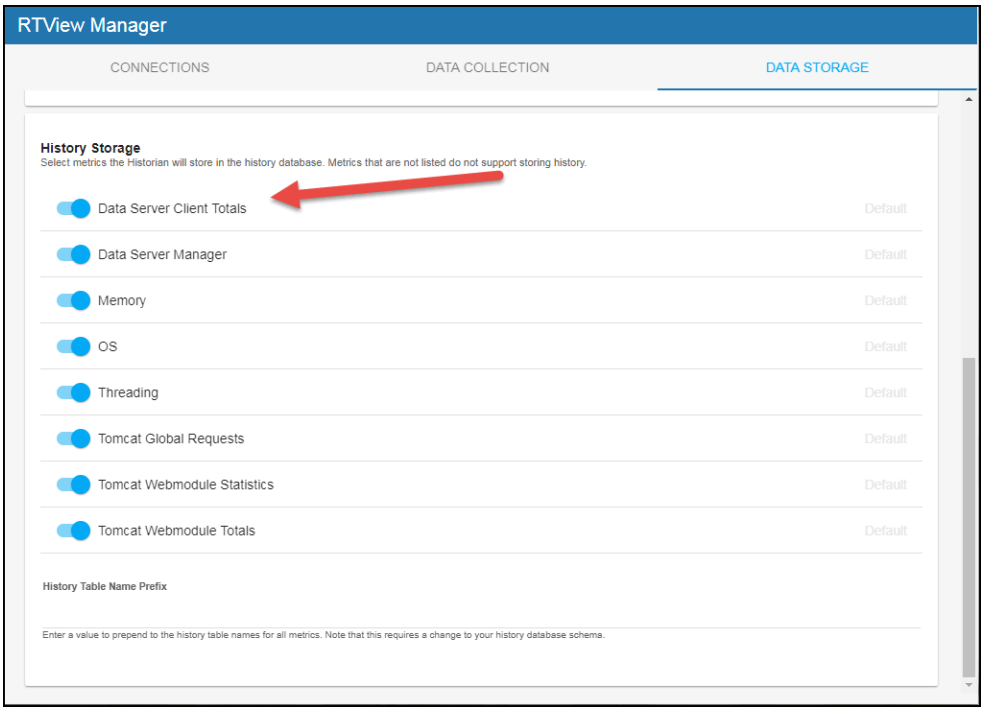


Enabling/Disabling Storage of Historical Data for RTView Manager

The **History Storage** region allows you to select which metrics you want the Historian to store in the history database. By default, all RTView Manager historical data is saved to the database. To enable/disable the collection of historical data, perform the following steps:

1. Navigate to the RTView Configuration Application > (**Project Name**) > **Solution Package Configuration** > **RTView Manager** > **DATA STORAGE** tab.

2. In the **History Storage** region, select the toggles for the various metrics that you want to collect/deselect for the metrics that you do not want to collect. Blue is enabled, gray is disabled.



Defining a Prefix for All History Table Names for RTView Manager Metrics

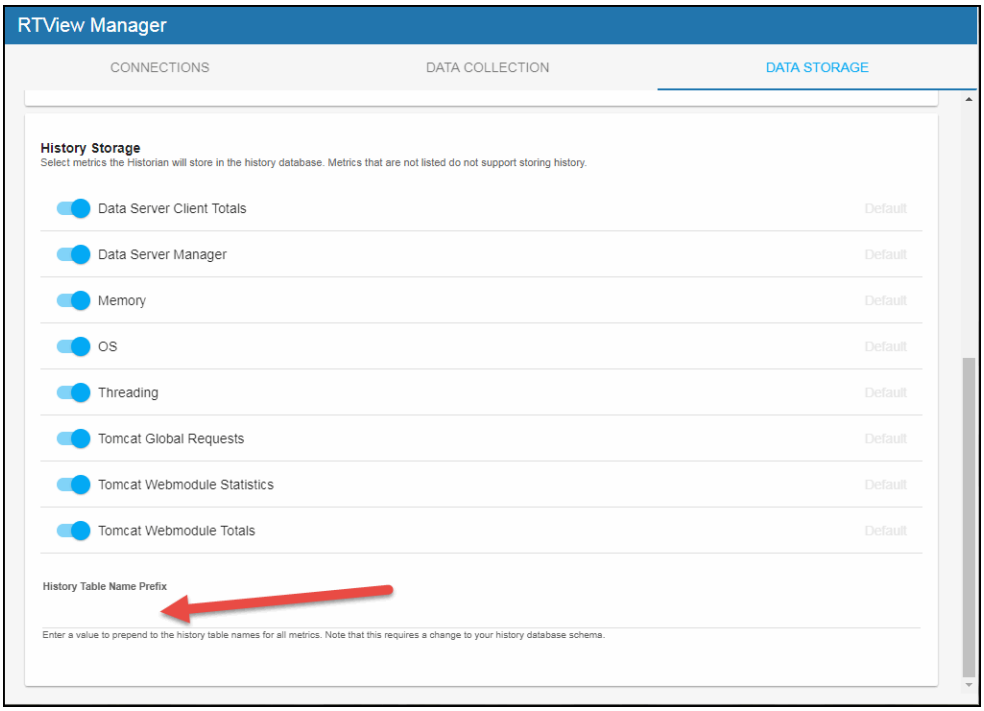
The **History Table Name Prefix** field allows you to define a prefix that will be added to the database table names so that the Monitor can differentiate history data between data servers when you have multiple data servers with corresponding Historians using the same solution package(s) and database. In this case, each Historian needs to save to a different table, otherwise the corresponding data server will load metrics from both Historians on startup. Once you have defined the **History Table Name Prefix**, you will need to create the corresponding tables in your database as follows:

- Locate the .sql template for your database under **RTVAPM_HOME**/**rtvmgr**/**dbconfig** and make a copy of it
- Add the value you entered for the **History Table Name Prefix** to the beginning of all table names in the copied .sql template
- Use the copied .sql template to create the tables in your database

To add the prefix:

1. Navigate to the RTView Configuration Application > **(Project Name)** > **Solution Package Configuration** > **RTView Manager** > **DATA STORAGE** tab.

2. Click on the **History Table Name Prefix** field and enter the desired prefix name.



Configure for TIBCO BusinessWorks

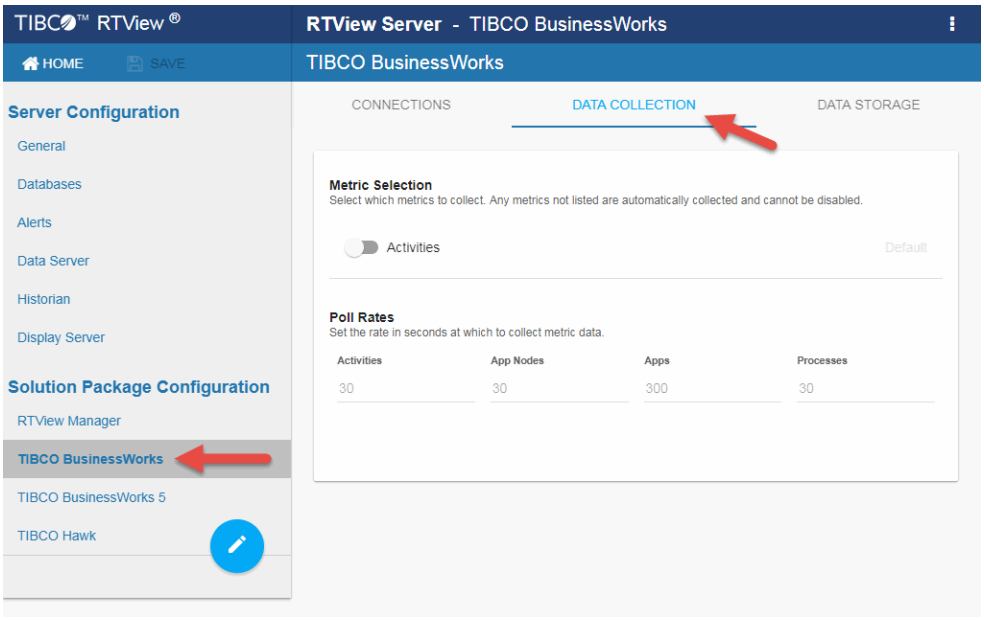
This section contains:

- Configuring Data Collection in the RTView Configuration Application for Version 6: Defines the steps required to configure data collection in the RTView Configuration Application for TIBCO BusinessWorks. (Required)
- Configuring Historical Data for Version 6: Describes the steps required to configure historical data collection. (Optional)

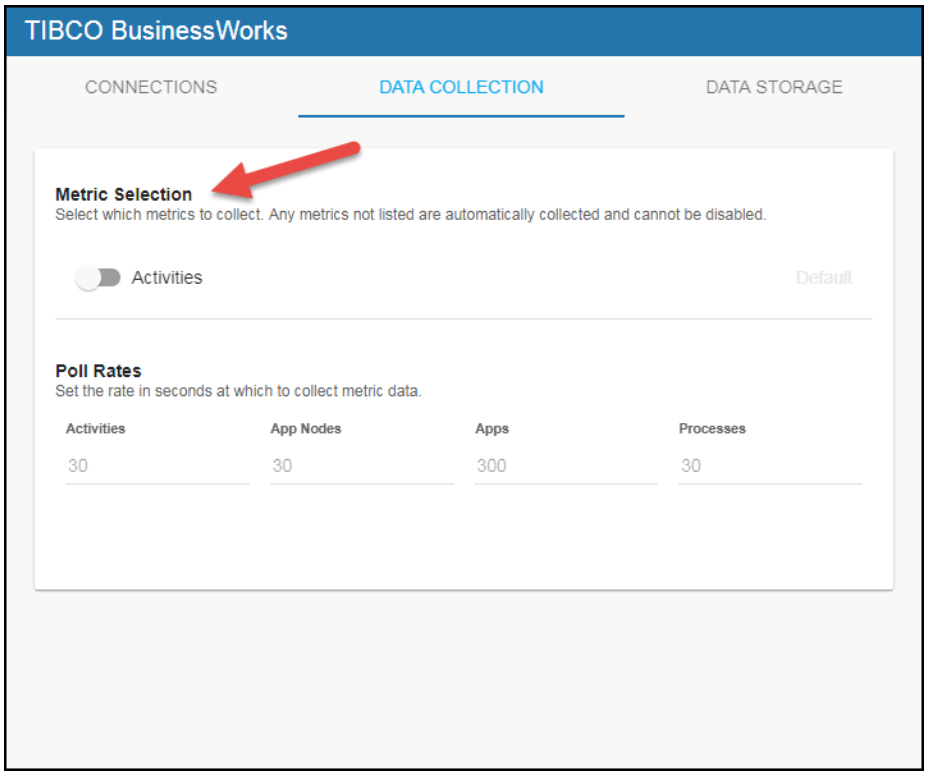
Configuring Data Collection in the RTView Configuration Application for Version 6

To configure data collection for Version 6:

1. Navigate to the RTView Configuration Application > **(Project Name)** > **Solution Package Configuration** > **TIBCO BusinessWorks** > **DATA COLLECTION** tab.



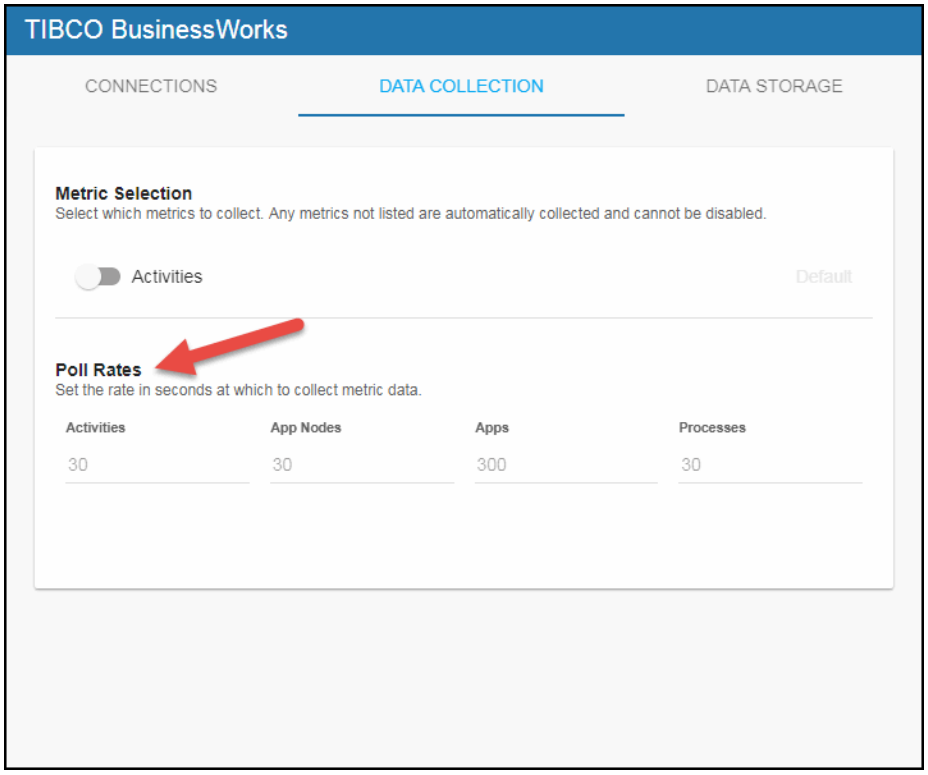
2. By default, collecting activities data for TIBCO ActiveMatris Businessworks is disabled. To enable collecting activities (Bw6Activities cache) data, navigate to the RTView Configuration Application > **TIBCO BusinessWorks** > **DATA COLLECTION** tab > **Metric Selection** section and enable the **Activities** toggle. Grey toggle is disabled, blue toggle is enabled.



3. If you want to modify the default values for the update rates for the TIBCO BusinessWorks caches, you can update the default polling rates in RTView Configuration

Application > **(Project Name)** > **Solution Package Configuration** > **TIBCO BusinessWorks** > **DATA COLLECTION** > **Poll Rates**.

Modify the value for the **Activities** field to modify the default polling rate for the _Bw6HawkActivityStatistics, _Bw6Activities, Bw6Activities, and Bw6ActivityTotalsByProcess caches. Modify the value for the **App Nodes** field to modify the polling rate for the _Bw6HawkAppNodeProcessInfo, and _Bw6AppNodes caches. Modify the value for the **Apps** field to modify the polling rate for the _Bw6HawkApps, Bw6AppSlices, __Bw6Apps, _Bw6Apps, _Bw6AppsDelta, and Bw6Apps caches. Modify the value for the **Processes** field to modify the polling rate for the _Bw6HawkProcessStatistics, _Bw6Processes, Bw6Processes, Bw6ProcessTotalsByAppNodeAndApp, Bw6ProcessTotalsByAppNode, Bw6ProcessTotalsByApp caches.



Configuring Historical Data for Version 6

You can specify the number of history rows to store in memory, the compaction rules, the duration before metrics are expired and deleted, and the different types of metrics that you want the Historian to store in the **DATA STORAGE** tab in the RTView Configuration Application. This section contains the following:

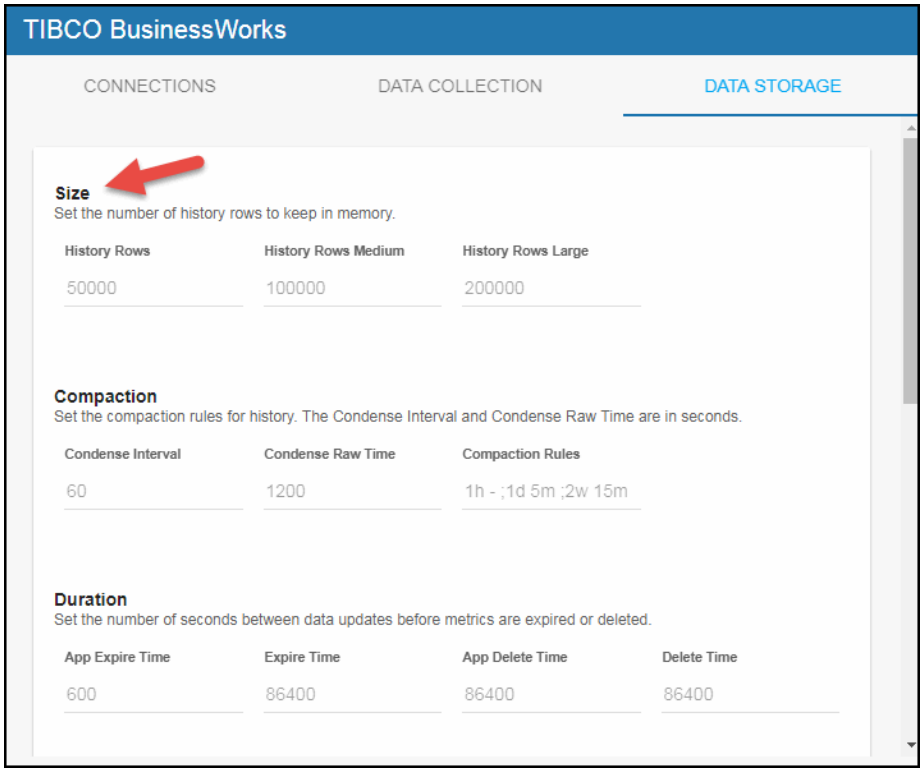
- Defining the Storage of In Memory BWMON History
- Defining Compaction Rules for BWMON
- Defining Expiration and Deletion Duration for BWMON Metrics
- Enabling/Disabling Storage of BWMON Historical Data

- Defining a Prefix for All History Table Names for Metrics

Defining the Storage of In Memory BWMON History

You can modify the maximum number of history rows to store in memory in the Data Storage tab. The **History Rows** property defines the maximum number of rows to store for the Bw6AppNodes, Bw6ProcessTotalsByAppNode, Bw6ProcessTotalsByApp, and Bw6ProcessTotalsByAppNodeAndApp caches. The **History Rows Medium** property defines the maximum number of rows to store for the Bw6ActivityTotalsByProcess and Bw6Processes caches. The **History Rows Large** property defines the maximum number of rows to store for the Bw6Activities cache. The default setting for **History Rows** is 50,000, the default setting for **History Rows Medium** is 100,000, and the default setting for **History Rows Large** is 200,000. To update the default settings:

1. Navigate to the RTView Configuration Application > **(Project Name)** > **Solution Package Configuration** > **TIBCO BusinessWorks** > **DATA STORAGE** tab.
2. In the **Size** region, click the **History Rows**, **History Rows Medium**, and **History Rows Large** fields and specify the desired number of rows.



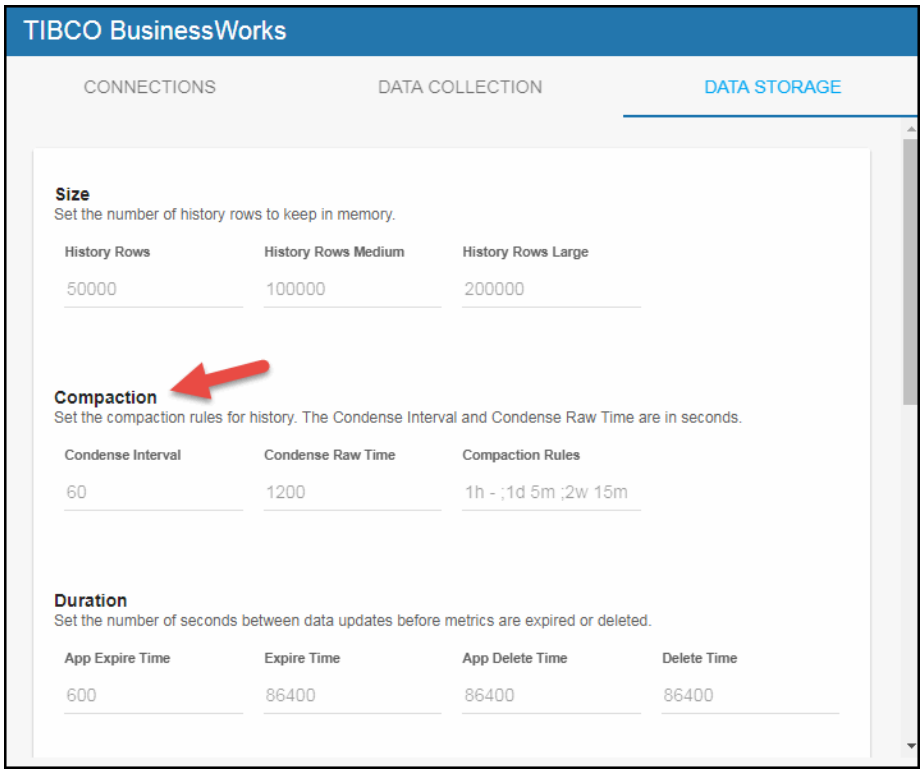
Defining Compaction Rules for BWMON

Data compaction, essentially, is taking large quantities of data and condensing it using a defined rule so that you store a reasonably sized sample of data instead of all of your data, thus preventing you from potentially overloading your database. The available fields are:

- **Condense Interval** -- The time interval at which the cache history is condensed. The default is 60 seconds. The following caches are impacted by this setting: Bw6Activities, Bw6ActivityTotalsByProcess, Bw6AppNodes, Bw6Processes, Bw6ProcessTotalsByAppNode, Bw6ProcessTotalsByApp, and Bw6ProcessTotalsByAppNodeAndApp.
- **Condense Raw Time** -- The time span of raw data kept in the cache history table. The default is 1200 seconds. The following caches are impacted by this setting: Bw6Activities, Bw6ActivityTotalsByProcess, Bw6AppNodes, Bw6Processes, Bw6ProcessTotalsByAppNode, Bw6ProcessTotalsByApp, and Bw6ProcessTotalsByAppNodeAndApp.
- **Compaction Rules** -- This field defines the rules used to condense your historical data in the database. By default, the columns kept in history will be aggregated by averaging rows with the following rule 1h -;1d 5m;2w 15m, which means the data from 1 hour will not be aggregated (1h - rule), the data over a period of 1 day will be aggregated every 5 minutes (1d 5m rule), and the data over a period of 2 weeks old will be aggregated every 15 minutes (2w 15m rule). The following caches are impacted by this setting: Bw6Activities, Bw6ActivityTotalsByProcess, Bw6AppNodes, Bw6Processes, Bw6ProcessTotalsByAppNode, Bw6ProcessTotalsByApp, and Bw6ProcessTotalsByAppNodeAndApp.

1. Navigate to the RTView Configuration Application > **(Project Name)** > **Solution Package Configuration** > **TIBCO BusinessWorks** > **DATA STORAGE** tab.

2. In the **Compaction** region, click the **Condense Interval**, **Condense Raw Time**, and **Compaction Rules** fields and specify the desired settings.

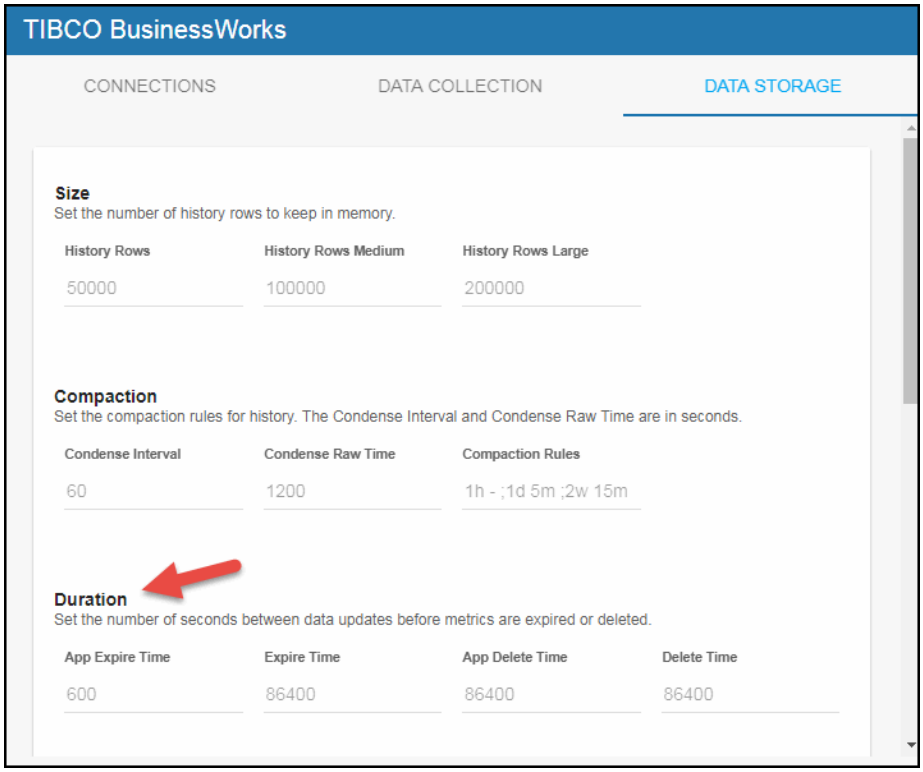


Defining Expiration and Deletion Duration for BWMON Metrics

The data for each metric is stored in a specific cache and, when the data is not updated in a certain period of time, that data will either be marked as expired or, if it has been an extended period of time, it will be deleted from the cache altogether. The **App Expire Time** field, which sets the expire time for the Bw6Apps cache, defaults to 600 seconds. The **Expire Time** field, which sets the expire time for the Bw6Activities, Bw6AppSlices, Bw6OsgiAgents, and Bw6Processes caches, defaults to 75 seconds. The **App Delete Time**, which sets the delete time for the Bw6Apps cache, defaults to 86,400 seconds. The **Delete Time**, which sets the delete time for the Bw6Activities, Bw6AppSlices, Bw6AppNodes, Bw6AppNodes, Bw6OsgiAgents, and Bw6Processes caches, defaults to 3600 seconds. To modify these defaults:

1. Navigate to the RTView Configuration Application > **(Project Name)** > **Solution Package Configuration** > **TIBCO BusinessWorks** > **DATA STORAGE** tab.

2. In the **Duration** region, click the **App Expire Time**, **Expire Time**, **App Delete Time**, and **Delete Time** fields and specify the desired settings.

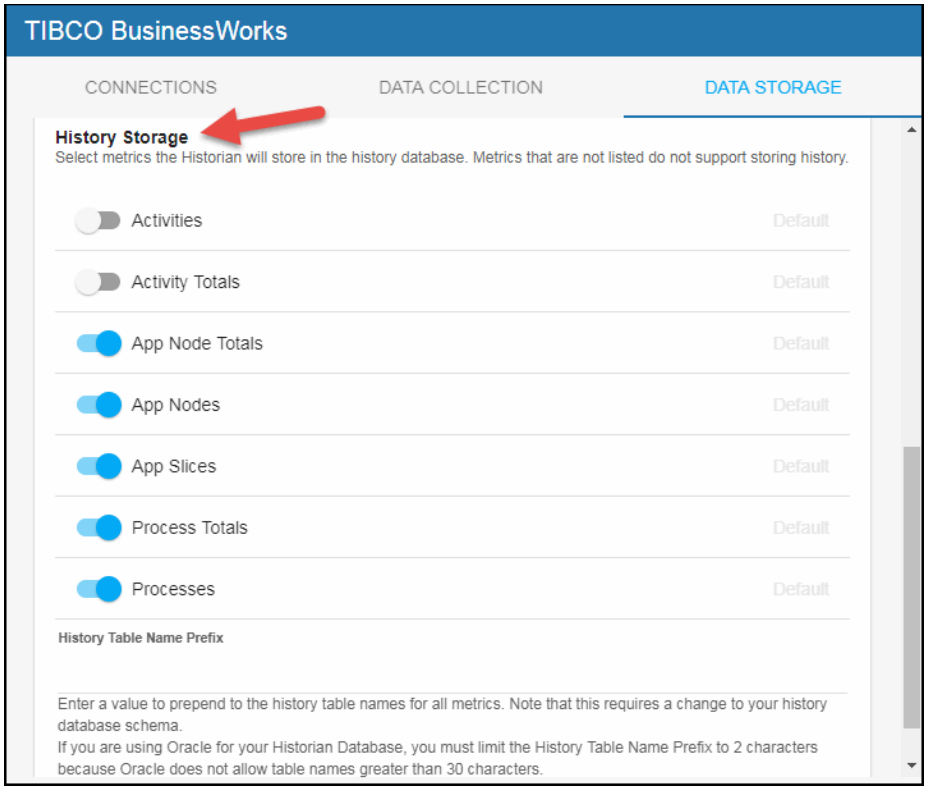


Enabling/Disabling Storage of BWMON Historical Data

The **History Storage** region allows you to select which metrics you want the Historian to store in the history database. By default, historical Activities and Activity Totals data is not saved to the database. All other metrics are saved by default. To enable/disable the collection of historical data, perform the following steps:

1. Navigate to the RTView Configuration Application > **(Project Name)** > **Solution Package Configuration** > **TIBCO BusinessWorks** > **DATA STORAGE** tab.

2. In the **History Storage** region, select the toggles for the various metrics that you want to collect/deselect for the metrics that you do not want to collect. Blue is enabled, gray is disabled.



Defining a Prefix for All History Table Names for Metrics

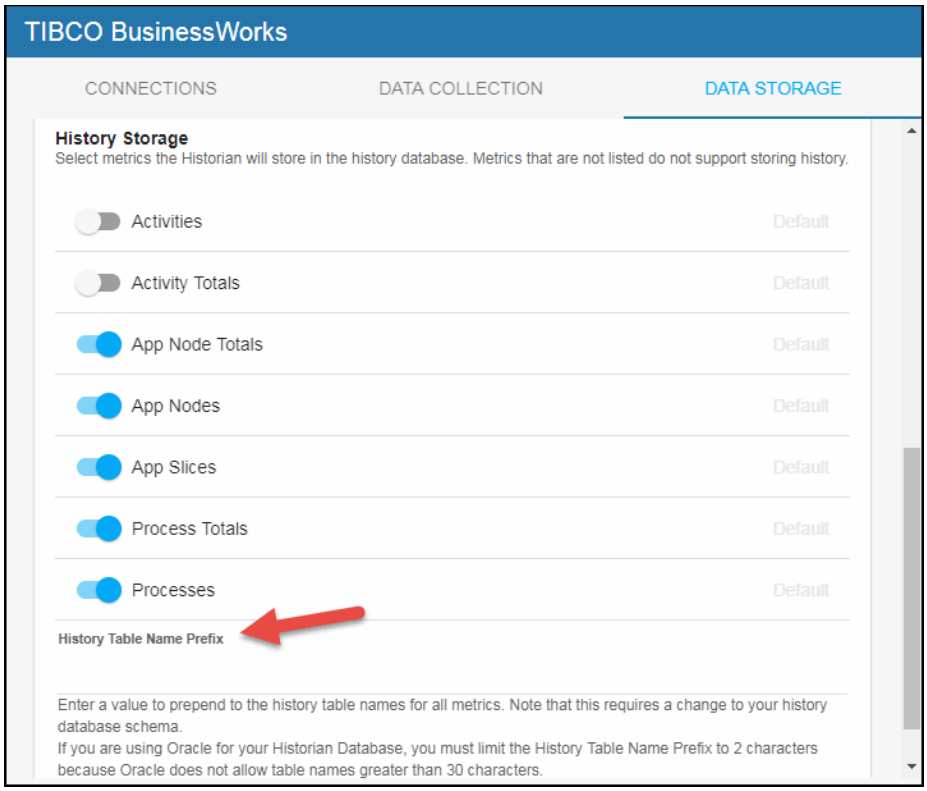
The **History Table Name Prefix** field allows you to define a prefix that will be added to the database table names so that the Monitor can differentiate history data between data servers when you have multiple data servers with corresponding Historians using the same solution package(s) and database. In this case, each Historian needs to save to a different table, otherwise the corresponding data server will load metrics from both Historians on startup. Once you have defined the **History Table Name Prefix**, you will need to create the corresponding tables in your database as follows:

- Locate the .sql template for your database under **RTVAPM_HOME**/**bw6mon**/**dbconfig** and make a copy of it
- Add the value you entered for the **History Table Name Prefix** to the beginning of all table names in the copied .sql template
- Use the copied .sql template to create the tables in your database

To add the prefix:

1. Navigate to the RTView Configuration Application > **(Project Name)** > **Solution Package Configuration** > **TIBCO BusinessWorks** > **DATA STORAGE** tab.

2. Click on the **History Table Name Prefix** field and enter the desired prefix name.



Configure for TIBCO BusinessWorks 5

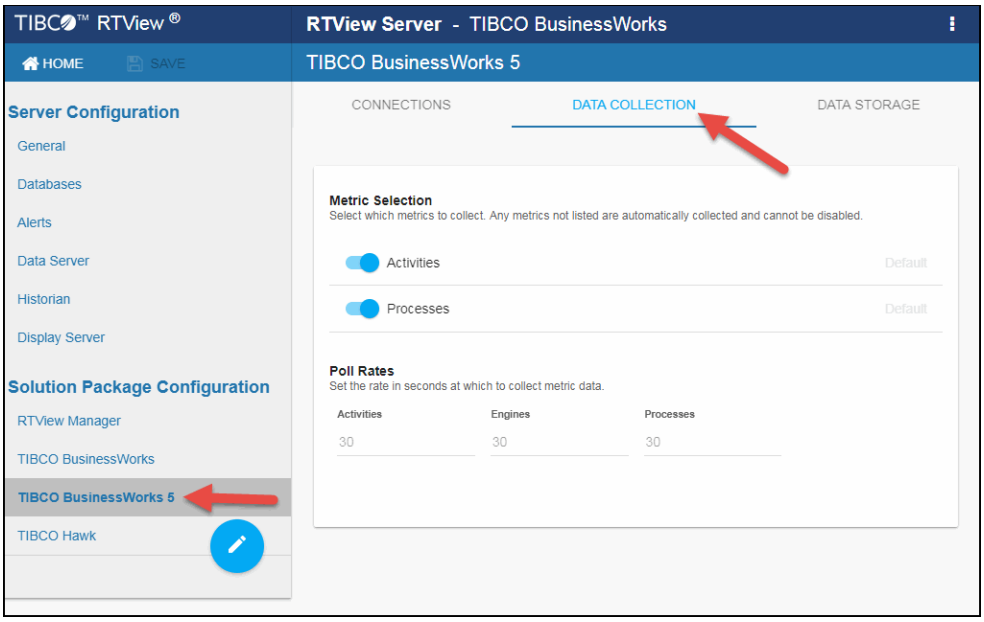
This section contains:

- Configuring Data Collection in the RTView Configuration Application for Version 5: Defines the steps required to configure data collection in the RTView Configuration Application for TIBCO BusinessWorks. (Required)
- Configuring Historical Data for Version 5: Describes the steps required to configure historical data collection. (Optional)
- Configure for BWSE Engines for Version 5: Describes how to configure BW Monitor to monitor BWSE engines for TIBCO ActiveMatrix (AMX) users.
- Create Customized Filters for Version 5: Describes how to create customized filters for BusinessWorks version 5.
- Enable BW Servers Displays for Version 5: Describes how to make the **BW Servers - Server Processes** and Single Server Process - Summary displays visible in the Monitor for BusinessWorks version 5. By default, these displays are not enabled.
- Reduce Collection of Process Data for Version 5: Describes how to modify data collection for BusinessWorks version 5.

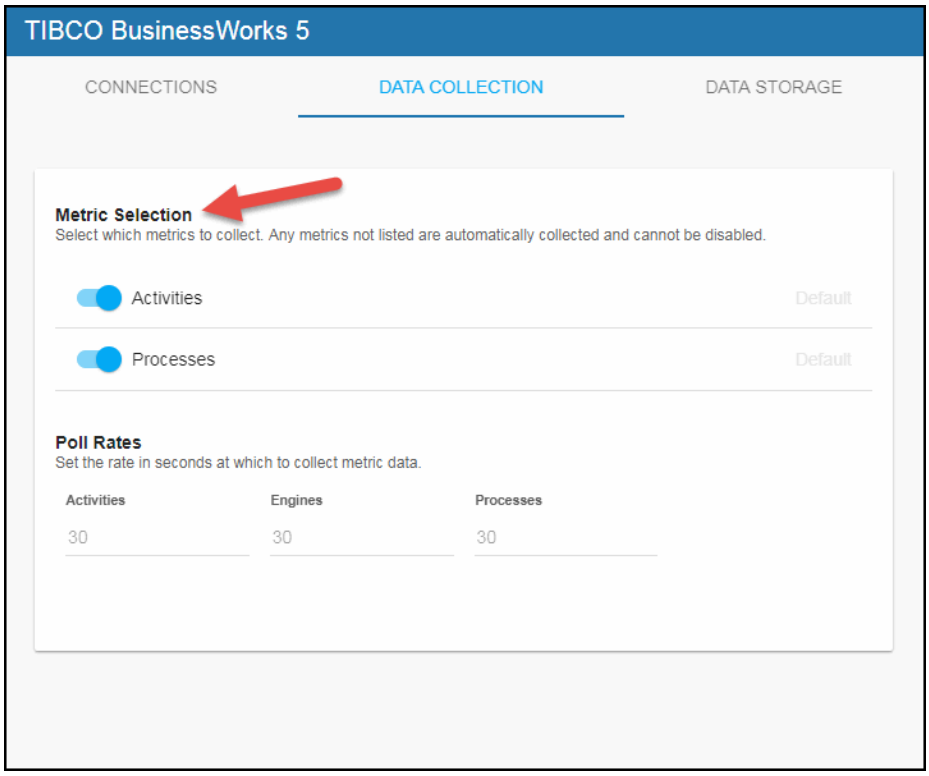
Configuring Data Collection in the RTView Configuration Application for Version 5

To configure data collection for Version 5:

1. Navigate to the RTView Configuration Application > **(Project Name)** > **Solution Package Configuration** > **TIBCO BusinessWorks 5** > **DATA COLLECTION** tab.

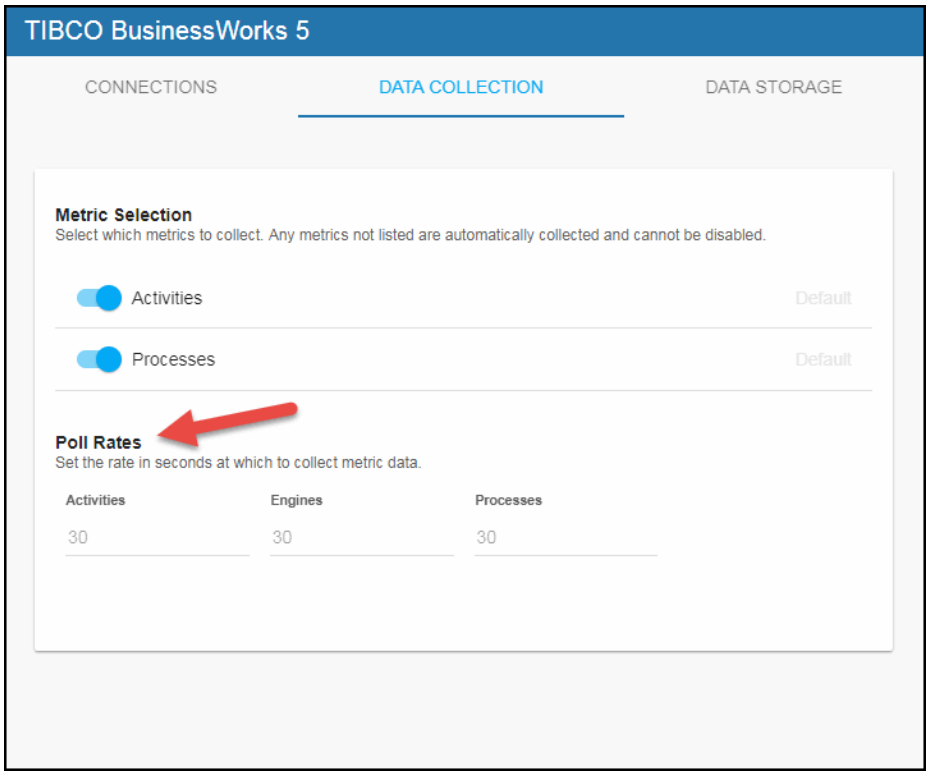


2. By default, collecting activities and processes data for TIBCO BusinessWorks 5 is enabled. To disable collecting activities (BwActivities cache) and processes (BwProcesses cache) data, navigate to the RTView Configuration Application > **TIBCO BusinessWorks 5** > **DATA COLLECTION** tab > **Metric Selection** section and disable the **Activities** and **Processes** toggles. Grey toggle is disabled, blue toggle is enabled.



3. If you want to modify the default values for the update rates for the TIBCO BusinessWorks 5 caches, you can update the default polling rates in RTView Configuration Application >**(Project Name)** > **Solution Package Configuration** > **TIBCO BusinessWorks 5** > **DATA COLLECTION** > **Poll Rates**.

Modify the value for the **Activities** field to modify the default polling rate for the BwActivities and BwActivityTotalsByProcess caches, which will update at approximately this rate, but will get occasional extra updates. Modify the value for the **Engines** field to modify the polling rate for the BwUndeployedEngines, BwEngines, BwEnginesDeployment, BwEngineState caches, which will update at approximately this rate, but will get occasional extra updates. Modify the value for the **Processes** field to modify the polling rate for the BwProcess and BwProcessTotalsByEngine caches, which will update at approximately this rate, but will get occasional extra updates.



Configuring Historical Data for Version 5

You can specify the number of history rows to store in memory, the compaction rules, the duration before metrics are expired and deleted, and the different types of metrics that you want the Historian to store in the **DATA STORAGE** tab in the RTView Configuration Application. This section contains the following:

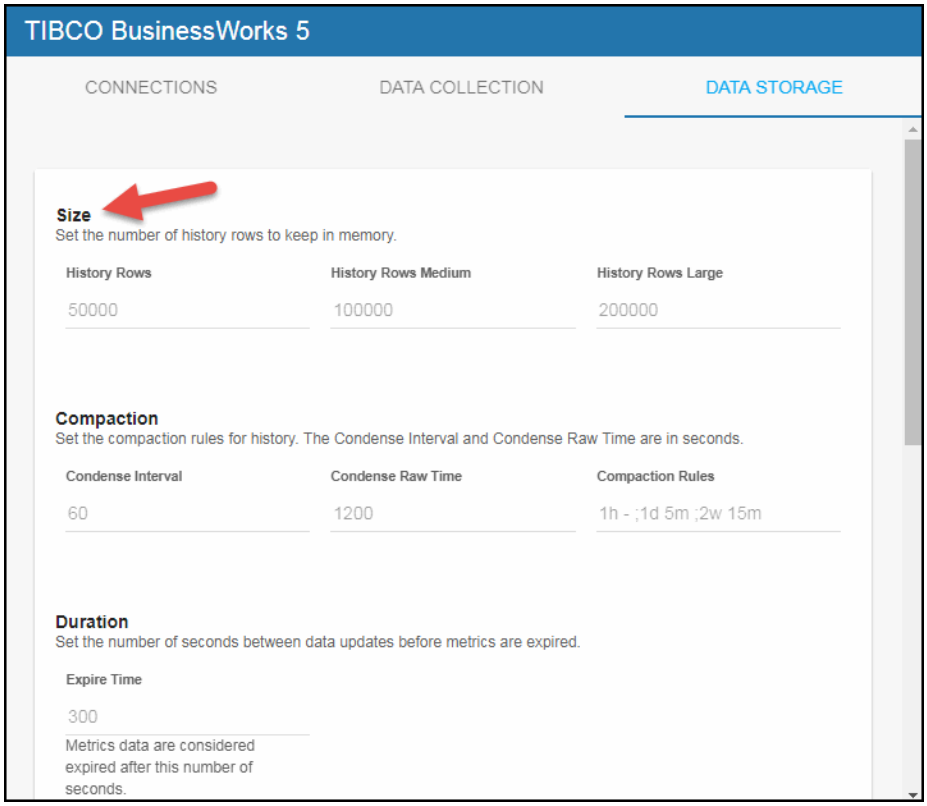
- Defining the Storage of In Memory BWMON5 History
- Defining Compaction Rules for BWMON5
- Defining Expiration and Deletion Duration for BWMON5 Metrics
- Enabling/Disabling Storage of Historical Data for BWMON5
- Defining a Prefix for All History Table Names for BWMON5 Metrics

Defining the Storage of In Memory BWMON5 History

You can modify the maximum number of history rows to store in memory in the **Data Storage** tab. The **History Rows** property defines the maximum number of rows to store for the BwEngines, BwProcessTotalsByEngine, and BwServers caches. The **History Rows Medium** property defines the maximum number of rows to store for the BwActivityTotalsByProcess and BwProcesses caches. The **History Rows Large** property defines the maximum number of rows to store for the BwActivities cache. The default setting for **History Rows** is 50,000, the default setting for **History Rows Medium** is 100,000, and the default setting for **History Rows Large** is 200,000. To update the default settings:

1. Navigate to the RTView Configuration Application > (**Project Name**) > **Solution Package Configuration** > **TIBCO BusinessWorks 5** > **DATA STORAGE** tab.

2. In the **Size** region, click the **History Rows**, **History Rows Medium**, and **History Rows Large** fields and specify the desired number of rows.



Defining Compaction Rules for BWMON5

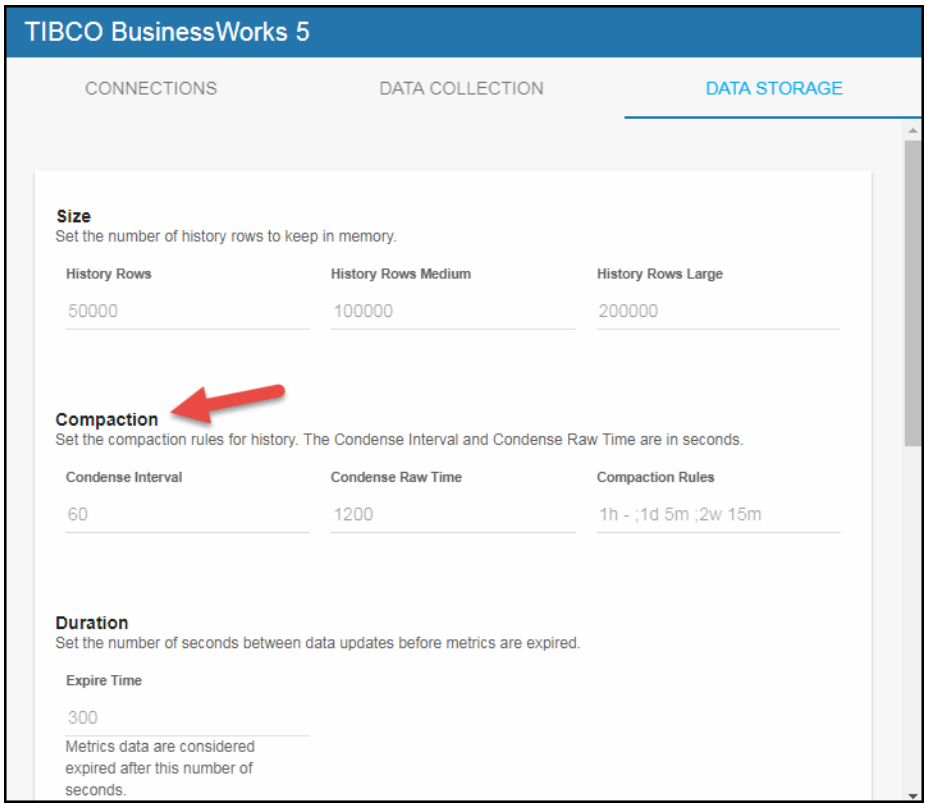
Data compaction, essentially, is taking large quantities of data and condensing it using a defined rule so that you store a reasonably sized sample of data instead of all of your data, thus preventing you from potentially overloading your database. The available fields are:

- **Condense Interval** -- The time interval at which the cache history is condensed. The default is 60 seconds. The following caches are impacted by this setting: BwActivities, BwActivityTotalsByProcess, BwEngines, BwProcesses, BwProcessTotalsByEngine, and BwServers.
- **Condense Raw Time** -- The time span of raw data kept in the cache history table. The default is 1200 seconds. The following caches are impacted by this setting: BwActivities, BwActivityTotalsByProcess, BwEngines, BwProcesses, and BwProcessTotalsByEngine.
- **Compaction Rules** -- This field defines the rules used to condense your historical data in the database. By default, the columns kept in history will be aggregated by averaging rows with the following rule 1h -;1d 5m;2w 15m, which means the data from 1 hour will not be aggregated (1h - rule), the data over a period of 1 day will be aggregated every 5 minutes (1d 5m rule), and the data over a period of 2 weeks old will be aggregated every 15 minutes (2w 15m rule). The following caches are impacted

by this setting: BwActivities, BwActivityTotalsByProcess, BwEngines, BwProcesses, BwProcessTotalsByEngine, and BwServers.

1. Navigate to the RTView Configuration Application > **(Project Name)** > **Solution Package Configuration** > **TIBCO BusinessWorks 5** > **DATA STORAGE** tab.

2. In the **Compaction** region, click the **Condense Interval**, **Condense Raw Time**, and **Compaction Rules** fields and specify the desired settings.

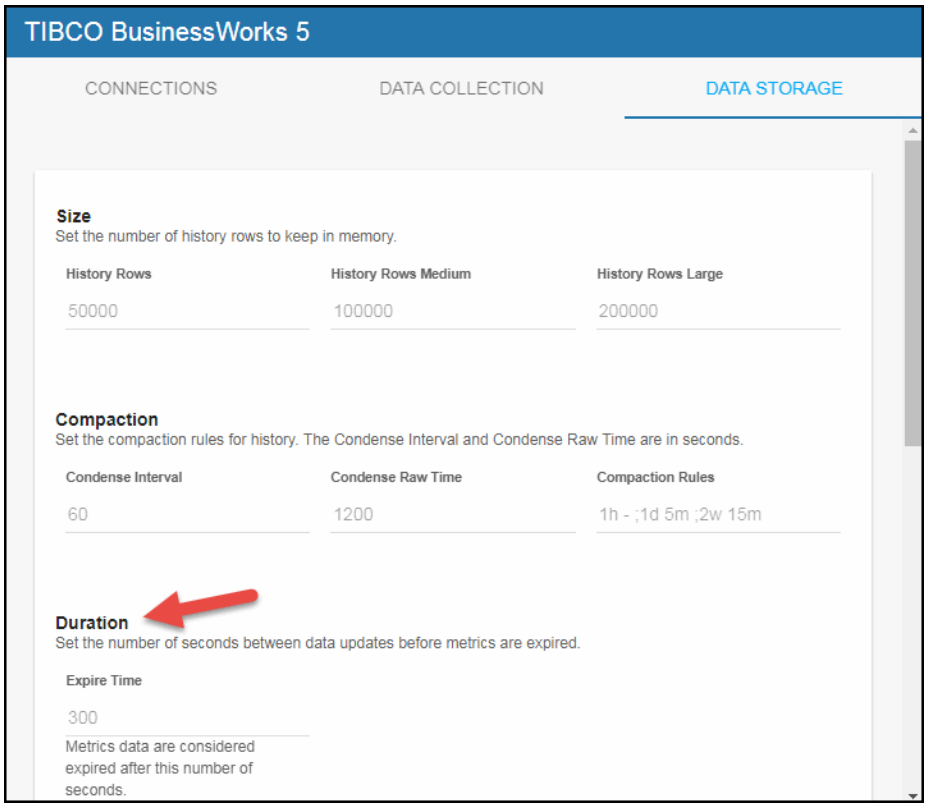


Defining Expiration and Deletion Duration for BWMON5 Metrics

The data for each metric is stored in a specific cache and, when the data is not updated in a certain period of time, that data will be marked as expired. The **Expire Time** field, which sets the expire time for the BwServers cache, defaults to 75 seconds. To modify this default:

1. Navigate to the RTView Configuration Application > **(Project Name)** > **Solution Package Configuration** > **TIBCO BusinessWorks 5** > **DATA STORAGE** tab.

2. In the **Duration** region, click the **Expire Time** field and specify the desired settings.

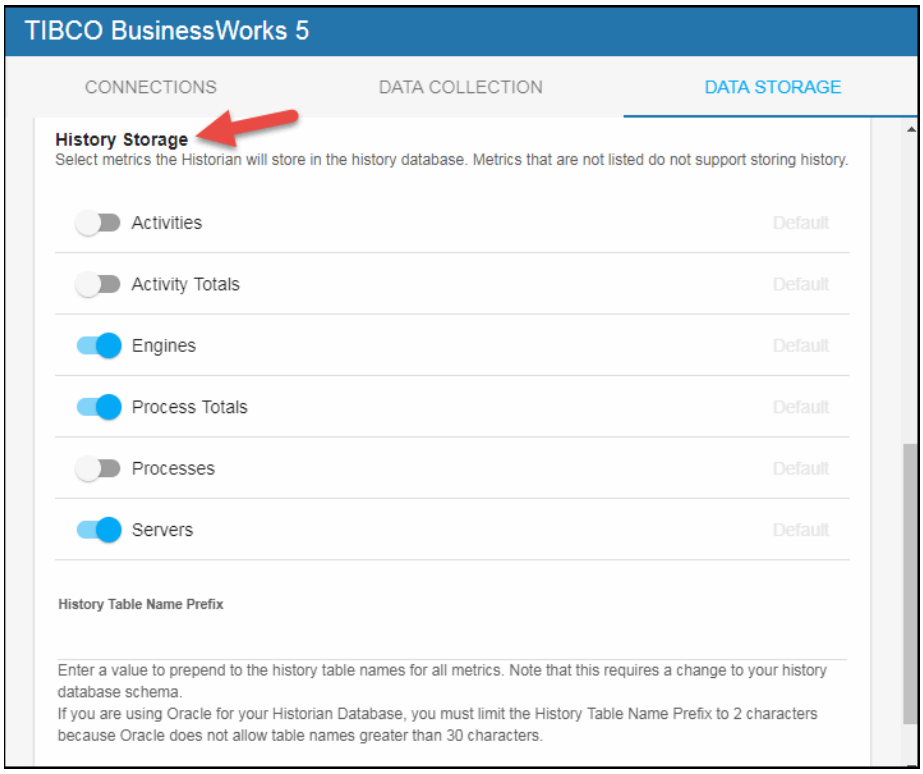


Enabling/Disabling Storage of Historical Data for BWMON5

The **History Storage** region allows you to select which metrics you want the Historian to store in the history database. By default, historical Activities, Activity Total, and Processes data is not saved to the database. All other metrics are saved by default. To enable/disable the collection of historical data, perform the following steps:

1. Navigate to the RTView Configuration Application > **(Project Name)** > **Solution Package Configuration** > **TIBCO BusinessWorks 5** > **DATA STORAGE** tab.

2. In the **History Storage** region, select the toggles for the various metrics that you want to collect/deselect for the metrics that you do not want to collect. Blue is enabled, gray is disabled.



Defining a Prefix for All History Table Names for BWMON5 Metrics

The **History Table Name Prefix** field allows you to define a prefix that will be added to the database table names so that the Monitor can differentiate history data between data servers when you have multiple data servers with corresponding Historians using the same solution package(s) and database. In this case, each Historian needs to save to a different table, otherwise the corresponding data server will load metrics from both Historians on startup. Once you have defined the **History Table Name Prefix**, you will need to create the corresponding tables in your database as follows:

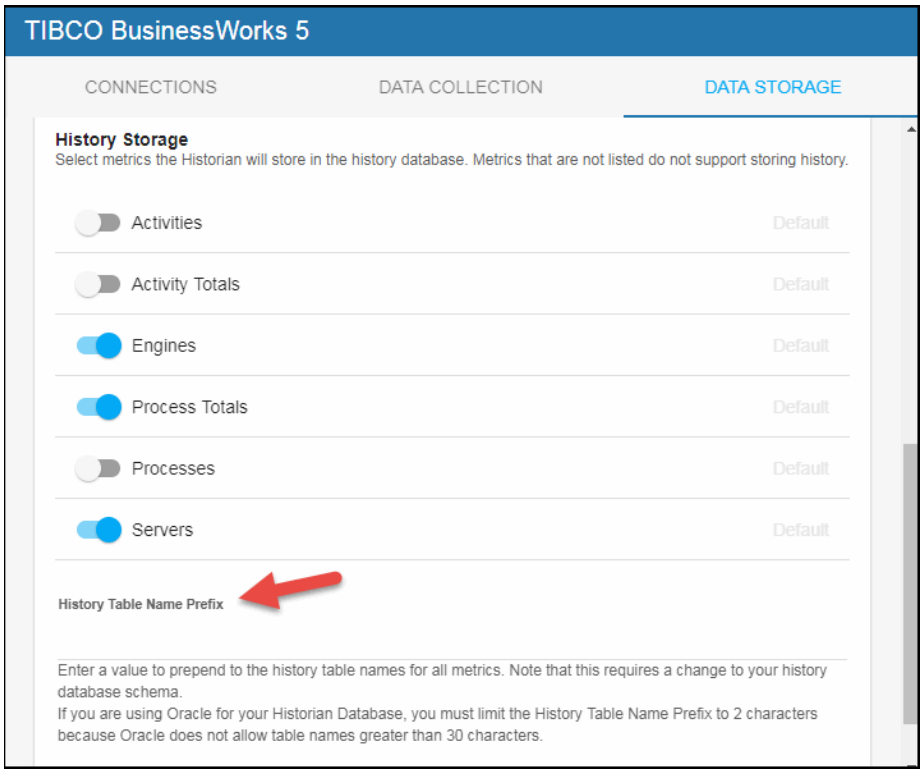
- Locate the .sql template for your database under **RTVAPM_HOME**/**bwmon**/**dbconfig** and make a copy of it
- Add the value you entered for the **History Table Name Prefix** to the beginning of all table names in the copied .sql template

Use the copied .sql template to create the tables in your database

To add the prefix:

1. Navigate to the RTView Configuration Application > **(Project Name)** > **Solution Package Configuration** > **TIBCO BusinessWorks 5** > **DATA STORAGE** tab.

2. Click on the **History Table Name Prefix** field and enter the desired prefix name.



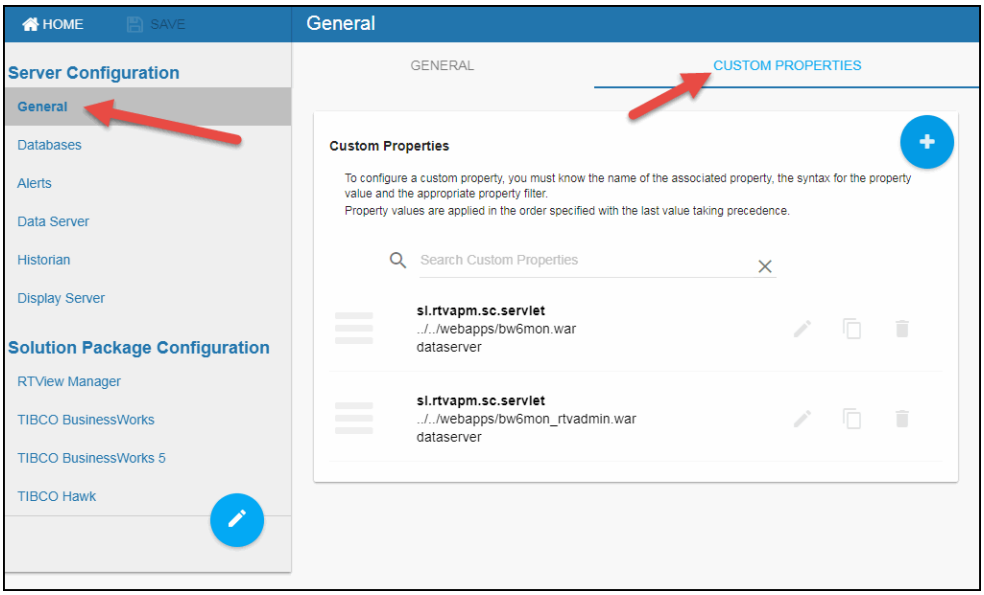
Configure for BWSE Engines for Version 5

This section is for TIBCO ActiveMatrix (AMX) users, and describes how to configure BW Monitor to monitor BWSE engines. BW Monitor needs access to AMX Node data stored in EMS message queues on the AMX Host system. To make this data available to BW Monitor you will create EMS topics with bridges from the queues.

The TIBCO BusinessWorks Service Engine (BWSE) is an ActiveMatrix (AMX) component that enables BW engines to participate in the implementation of AMX services. In this case, the BWSE engines run within an AMX Node and are not visible to BW Monitor. However, you can configure BW Monitor to display these engines, as well as to gather JVM memory metrics for the AMX Nodes in which they are running.

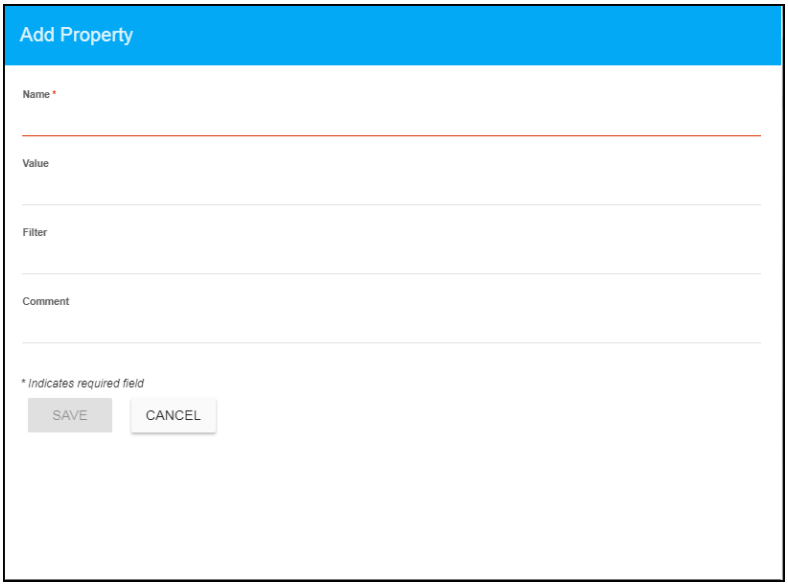
To Configure for BWSE engines:

1. Navigate to the RTView Configuration Application > **(Project Name)** > **Server Configuration** > **General** > **CUSTOM PROPERTIES** tab.



2. Click the icon.

The **Add Property** dialog displays.



3. Create the following custom properties, one at a time, and click **Save** after creating each:

Name: sl.rtview.cache.config
Value: bw_engine_microagents.rtv
Filter: collector

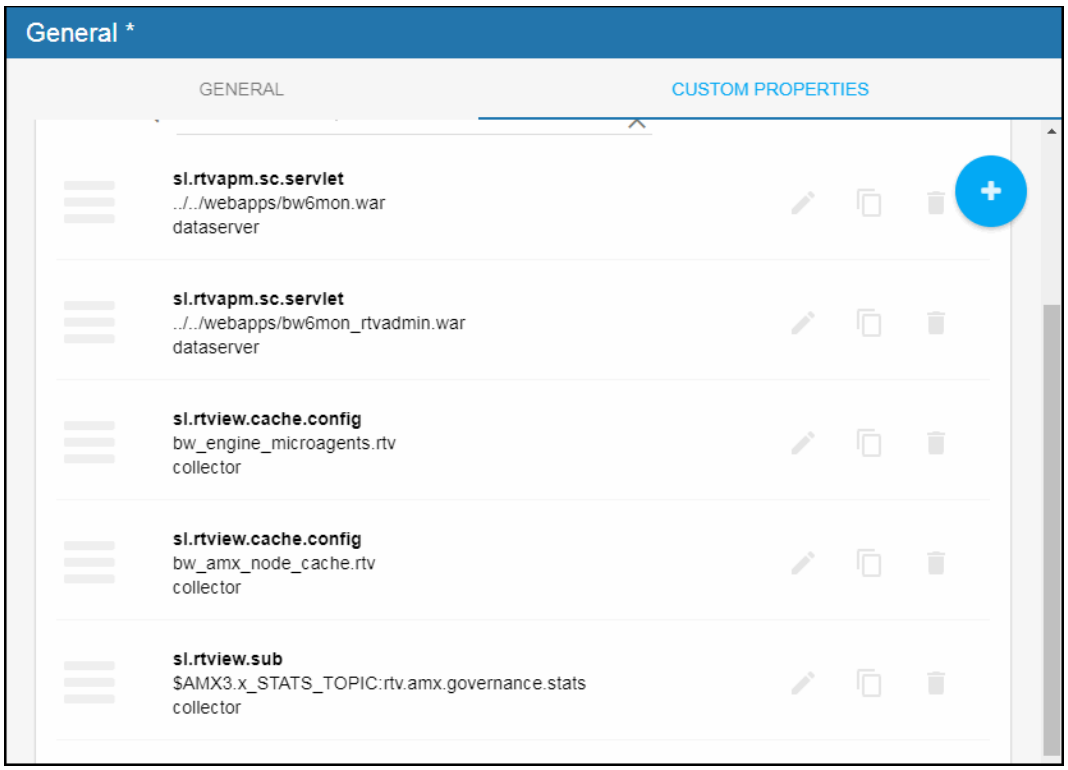
Name: sl.rtview.cache.config
Value: bw_amx_node_cache.rtv
Filter: collector

Name: sl.rtview.sub

Value: $AMX3.x_STATS_TOPIC:rtv.amx.governance.stats

Filter: collector

Once all three are created and saved, the newly created properties display in the **Custom Properties** tab.



4. For each AMX host, click the + icon, add the following custom properties, and click Save after entering each:

Name: sl.rtview.jms.jmsconn

Value: **local** com.tibco.tibjms.TibjmsTopicConnectionFactory **tcp://localhost:7222** admin **- - - - -**

Filter: collector

(where **local** is the connection name and **tcp://localhost:7222** is the URL for your ems server)

Name: sl.rtview.cache.config

Value: bw_amx_node_cache_source.rtv $jms_conn:**local**

Filter: collector

(where **local** is the connection name)

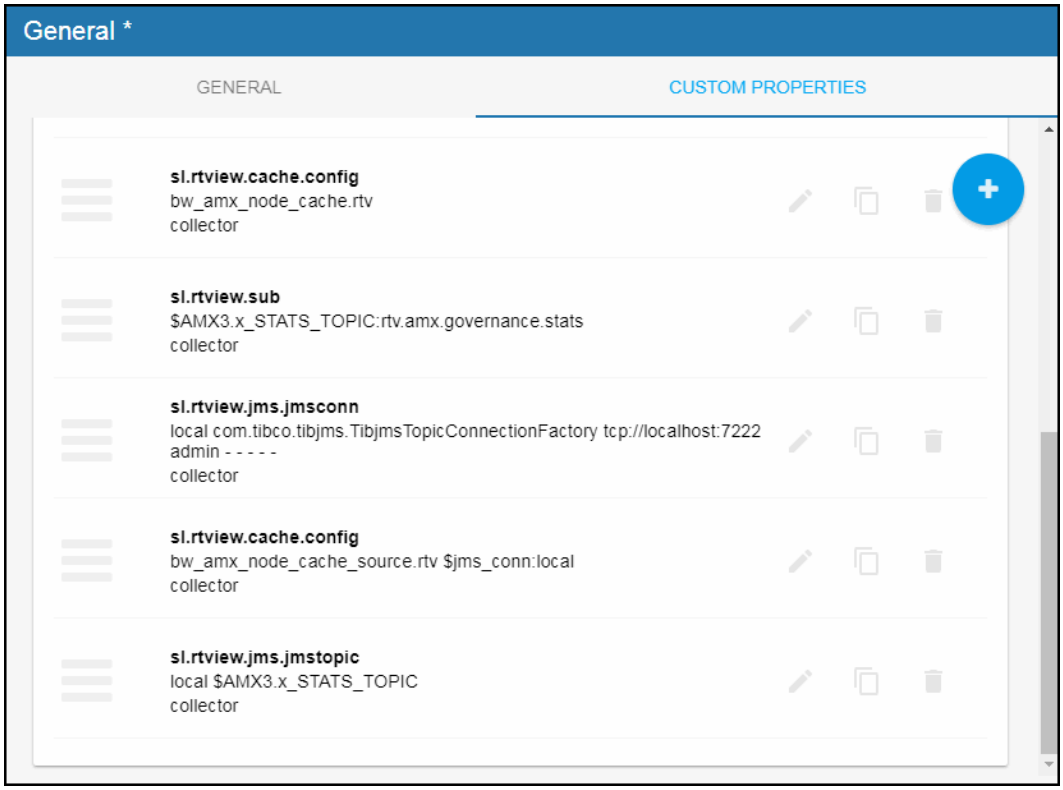
Name: sl.rtview.jms.jmstopic

Value: local $AMX3.x_STATS_TOPIC

Filter: collector

(where **local** is the connection name)

Once all three are created and saved, the newly created properties display in the **Custom Properties** tab.



5. Click **Save** and restart the data server to apply your changes.

Create Customized Filters for Version 5

Note: You can only create customized filters in the Classic (non-HTML) displays.

This section applies to BusinessWorks version 5, and describes how to create filtering options for the **Filter:** drop-down menu. By default, the **Filter:** drop-down menu only contains the **No Filter** option.

You can create filtering options that limit display data based on a combination of domain, engine, process, and activity names. You configure the filtering options prior to running the Monitor.

To create your filtering options edit the **bwmon_filters.xml** file, located in your project directory. Edit by inserting regular expressions for each type of name you want filter by, as well as a name for the filter. The filter name becomes the option in the **Filters:** drop-down menu. Instructions and examples are provided in the **bwmon_filters.xml** file.

Enable BW Servers Displays for Version 5

Note: This only applies to the Classic (non-HTML) displays.

This section applies to BusinessWorks **version 5**, and describes how to make the **BW Servers -** Server Processes and Single Server Process - Summary displays visible in the Monitor. By default, these displays are not enabled.

The **Server Processes** and **Single Server Summary** displays show information about BW Server operating system processes. Due to limitations in TIBCO Hawk, the data they display is not available from IBM AIX or HP-UX servers.

To enable the displays:

1. Open the **bwmon_navtree.xml** file, located in your project directory. For details about the project directory, see Enabling Login in the Monitor.

2. Uncomment the following two lines,

<!-- <node label="Server Processes" display="bw_server_processes"/> -->

<!-- <node label="Server Process Summary" display="bw_server_process_summary"/> -->

3. Save the file.

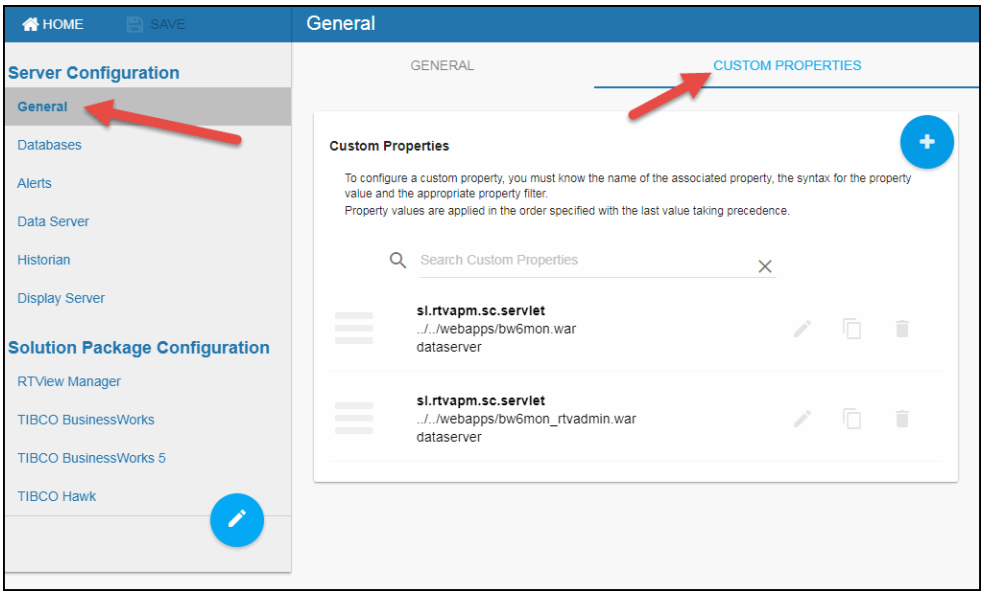
4. Restart the Monitor.

5. Verify the displays appear under **BW Servers** in the navigation tree.

Reduce Collection of Process Data for Version 5

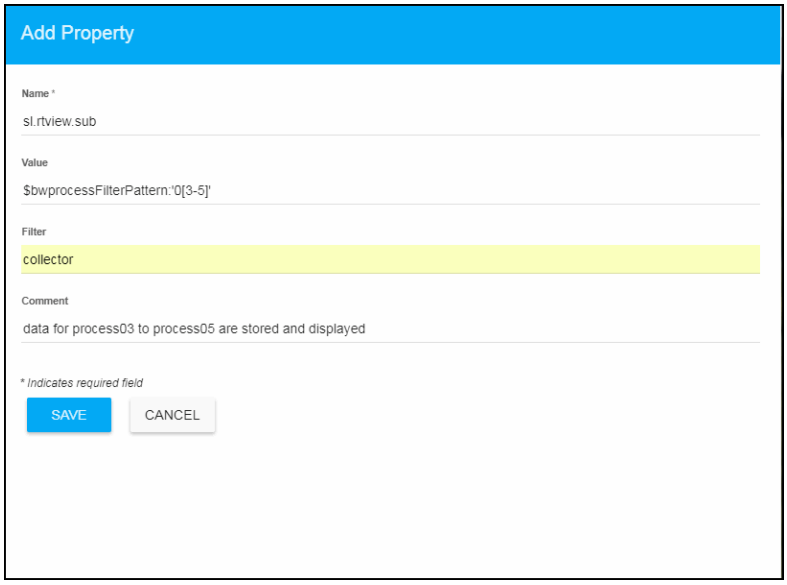
This section describes how to exclude BW5 process data that is collected by the Monitor but not of interest to you. By default, all process data is included. Excluding data stops it from being stored in the cache and removes it from displays. To exclude (or include) data, create a custom property in the RTView Configuration Application as such:

1. Navigate to the RTView Configuration Application > **(Project Name)** > **Server Configuration** > **General** > **CUSTOM PROPERTIES** tab.



2. Click the icon.

The **Add Property** dialog displays.



3. Define the values for the desired property. Each property specifies a regular expression that is applied to a process name. If the name matches the pattern, then the process is included. To exclude processes, start the filter pattern with **^** (negation).

For example, if you have the following processes:

process01.process

process02.process

process03.process

process04.process

process05.process

process06.process

process07.process

and you set the first property as follows:

Name: sl.rtview.sub

Value: $bwprocessFilterPattern:'0[3-5]'

Filter: collector

Comment: (description of the filter)

then the data for process03 to process05 is stored and displayed:

process03.process

process04.process

process05.process

If you set the second property as follows:

Name: sl.rtview.sub

Value: $bwprocessFilterPattern:'0[^4]'

Filter: collector

Comment: (description of the filter)

Then data from process04 is excluded and you continue getting data from:

process03.process

process05.process

4. Once all your properties have been added, click **Save**.

5. Restart the data server so that your changes take effect.

Design Notes

This section contains:

- Supported API Actions
- Filenames
- Sample json
- Adding, Editing, Deleting JsonPrimitive Properties
- Adding and Editing JsonObject Properties
- Deleting JsonObject Properties
- Updating vs. Restarting Data Servers
- High Availability

Supported API Actions

The REST API supports several actions. To get the list of actions, go to the sample application as described above and execute the following on the command line:

node main.js -action=getActions

To get the description of a single action:

node main.js -action=getActions -name=actionName

You can also execute any action that start with get in a browser as follows (where **host**, **port** and **rtvadmin** are the values you specified in the sample application):

http://host:port/rtvadmin/api?action=getActions&name=actionName

Filenames

When using the REST API to import initial properties into the Configuration Application, the filename must be **project**. This is because the Configuration Application reads and writes the project properties files and all RTView projects automatically read them. When using the REST API to automatically update properties that are not included in the Configuration application, the filename must match the **-properties** argument in the **rtvservers.dat** file and must NOT be **project**.

Sample json

You can optionally use the Configuration Application to generate sample json to get you started. Properties saved from the Configuration Application are in **projects\rtview-server\project.properties.json**.

Adding, Editing, Deleting JsonPrimitive Properties

All primitive json values must be enclosed in quotes, even boolean and number values. The top level solution package element must be included.

The following example uses **solmon** properties to illustrate. See the generating sample json properties for details about generating properties for your solution package.

Example:

```
{
	"solmon": {
		"expiretime":"10000"
	}
}
```

Adding and Editing JsonObject Properties

Solution package connections are arrays of JsonObjects. The property descriptions indicate which fields in the json object are required and which are indexes. When adding a new connection (or other JsonObject), you must include all of the required and index fields or the property will not be saved. The top level solution package element must be included.

The following example uses **solmon** properties to illustrate. See the generating sample json properties for details about generating properties for your solution package.

Example:

```
{
    "solmon": {
        "conn": [{
                "iscloudvmr": "true",
                "__name": "conn2",
                "url": "http://host2:8080/SEMP",
                "version": "7.4VMR",
                "vpnnamelist": "vpn1;vpn2"
            },
            {
                "iscloudvmr": "true",
                "__name": "conn3",
                "url": "http://host3:8080/SEMP"
            }
        ]
    }
}
```

When adding connections to an existing file, you can either merge the new connections into the existing connection list or you can replace the whole list with the connections. This is controlled by the merge parameter. When merge is true, the indexes are used to control whether a new connection is added or an existing connection is modified.

Deleting JsonObject Properties

Solution package connections are arrays of JsonObjects. The property descriptions indicate which fields in the json object are indexes. When deleting a connection (or other JsonObject), only the index fields are required. The top level solution package element must be included.

The following example uses **solmon** properties to illustrate. See the generating sample json properties for details about generating properties for your solution package.

Example:

```
{
    "solmon": {
        "conn": [{
                "__name": "conn2"
            },
            {
                "__name": "conn3",
            }
        ]
    }
}
```

Updating vs. Restarting Data Servers

All connection properties support updates. Once you have added, edited or deleted connections using the REST API, you can apply those changes with the updatePropertiesOnServer action. Restart is not required. Note that when connections are

removed from your configuration, they are not immediately removed from the monitor. They stay in the caches (and display) but do not receive further updates. They will expire and be removed based on the settings in the DATA STORAGE tab of the Configuration Application. All non-connections properties are applied on restart, so they must be applied with the restartServers action. Restarting your servers will also cause any deleted connections to be immediately removed from the caches and displays.

High Availability

To edit properties for HA-configured servers, first follow the instructions in the **High Availability** section of this document to configure the rtvadmin servlet for High Availability.

Configure Ports

This section describes how to configure the ports for the RTView Servers. This step is required if you need to modify port settings or deploy Java processes on different hosts. Otherwise, this step is optional.

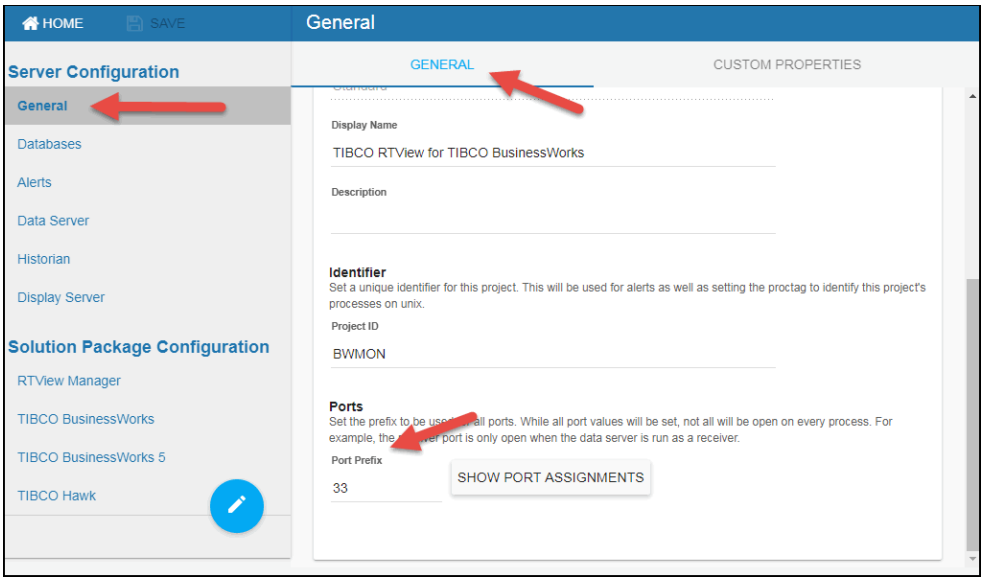
Java Processes

There are several Java processes included with the Monitor that are used during browser deployment. By default, it is assumed that these Java processes run on one host and that no configuration changes are needed. However, if these processes are distributed across several hosts, or if the default port definitions for these processes need to be modified, then configuration file settings must also be modified to allow all Monitor components to communicate with each other.

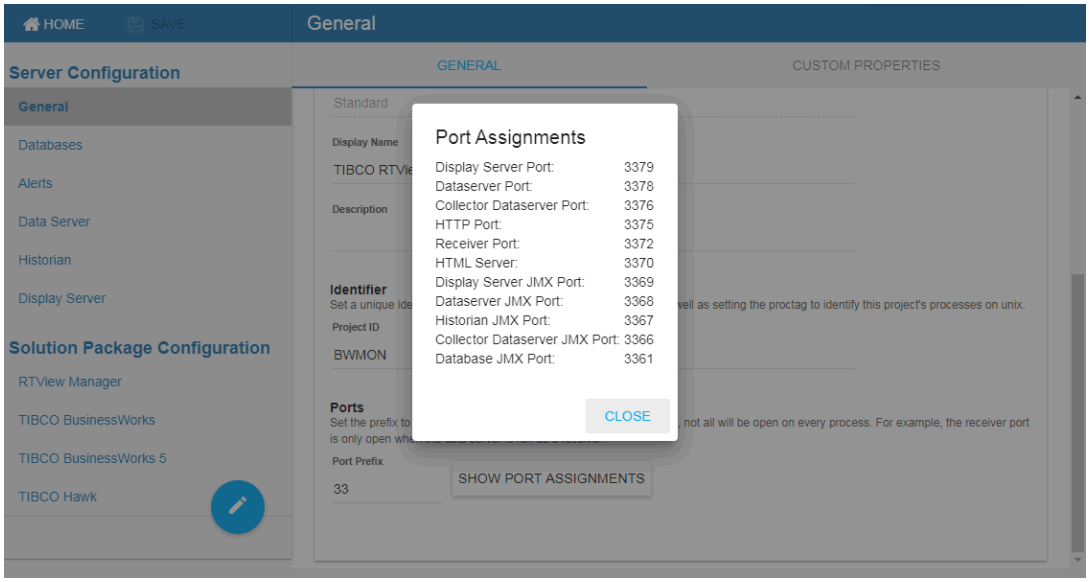
Java Process	Description	Default Port(s)
RTView Data Server	Gathers performance metrics.	Default Port= **3378** Default JMX Port = **3368**
RTView Historian	Retrieves data from the RTView Data Server and archives metric history to a database.	Default JMX Port= **3367**
RTView Display Server	Collects the data and generates the displays that the Application Server uses to produce the web pages.	Default Port= **3379** Default JMX Port = **3369**
Eclipse Jetty	Hosts the RTView Servlets.	Default Port= **3370**

To modify port settings or deploy Java processes on different hosts (rather than on a single host):

1. Navigate to the RTView Configuration Application > **(Project Name)** > **Server Configuration** > **General** > **GENERAL** tab.



2. In the **Ports** region, click the **Port Prefix** field and specify the port prefix that you want to use. You can click the **Show Port Assignments** button to view the port numbers that are created using the **Port Prefix** you specified.



3. Click **Save** in the RTView Configuration Application and restart the dataserver.

4. Edit the **update_wars** (**.bat** or **.sh**) file (in the **TIB_rtview-bw/projects/rtview-server** directory) and change the port prefix for all ports to the prefix specified in Step 2.

5. Rebuild the war files for your application server by executing the following scripts (in the **TIB_rtview-bw/projects/rtview-server** directory):

   Windows:

   update_wars.bat

   UNIX:

   ./update_wars.sh

6. Deploy all of your war files to your application server. For example:

   If monitoring using Tomcat as your application server, copy all the **.war** files located in the **TIB_rtview-bw\projects\rtview-server** directory to the Tomcat **webapps** directory.

Configure the Database

The Monitor is delivered with a default memory resident HSQLDB database, which is suitable for evaluation purposes. However, in production deployments, we recommend that you deploy one of our supported databases. For details, see the TIBCO© RTView© Standard Monitor User's Guide.

This section describes how to setup an alternate (and supported) database.

Database Requirements

The Monitor requires two database connections that provide access to the following information:

- **Alert Settings**

  The ALERTDEFS database contains alert administration and alert auditing information. The values in the database are used by the alert engine at runtime. If this database is not available, the Self-Service Alerts Framework under which alerts are executed will not work correctly.

- **Historical Data**

  The RTVHISTORY database contains the historical monitoring data to track system behavior for future analysis, and to show historical data in displays.

To Configure the Monitor Database:

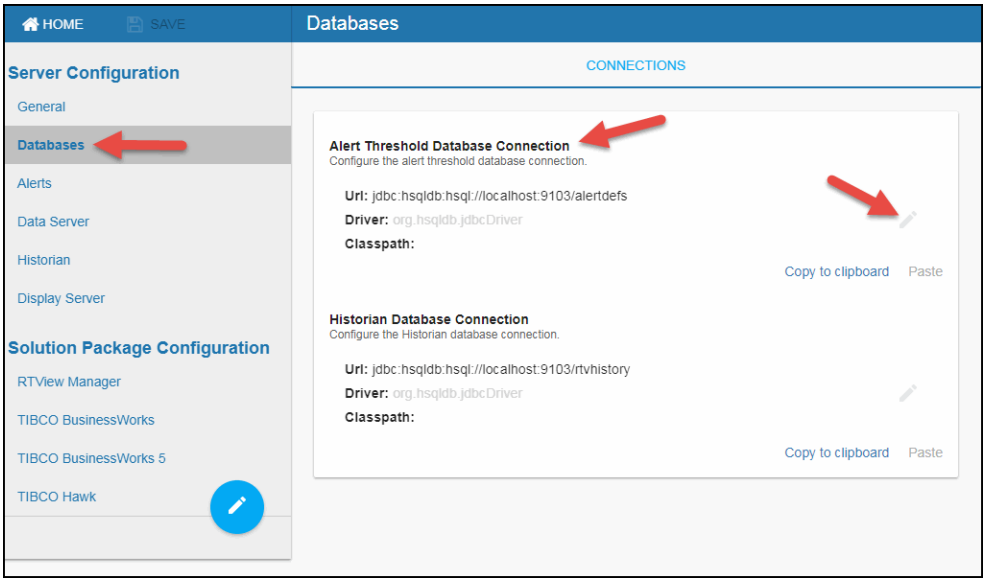
You configure the database by defining database configurations in the RTView Configuration Application. You will also copy portions of the **database.properties** template file (located in the **rtvapm\common\dbconfig** directory) into the RTView Configuration Application.

1. Install a database engine of your choice. Supported database engines are Oracle, Microsoft SQL Server, MySQL, and DB2.

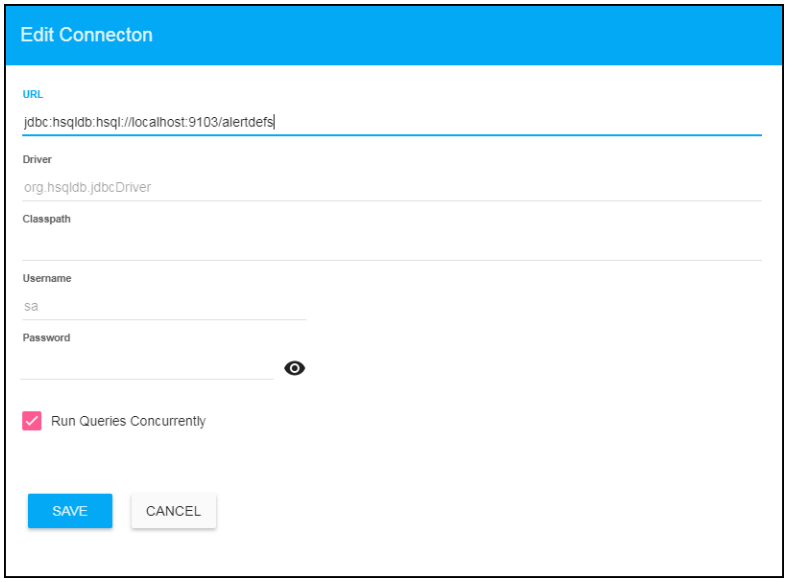
   NOTE: The default page size of DB2 is 4k. It is required that you create a DB2 database with a page size of 8k. Otherwise, table indexes will not work.

2. Open the **database.properties** template file, which is located in the **rtvapm\common\dbconfig** directory, and find the line that corresponds to your supported database from the "Define the ALERTDEFS DB" section.

3. Navigate to the RTView Configuration Application > **(Project Name)** > **Server Configuration** > **Databases** and click the Edit icon in the **Alert Threshold Database Connection** region.



The **Edit Connection** dialog displays.



4. Enter the information from Step 2 into the **Edit Connection** dialog and click **Save**.

URL - Enter the full database URL to use when connecting to this database using the specified JDBC driver.

Driver - Enter the fully qualified name of the JDBC driver class to use when connecting to this database.

Classpath - Enter the location of the jar where the JDBC driver resides in your environment.

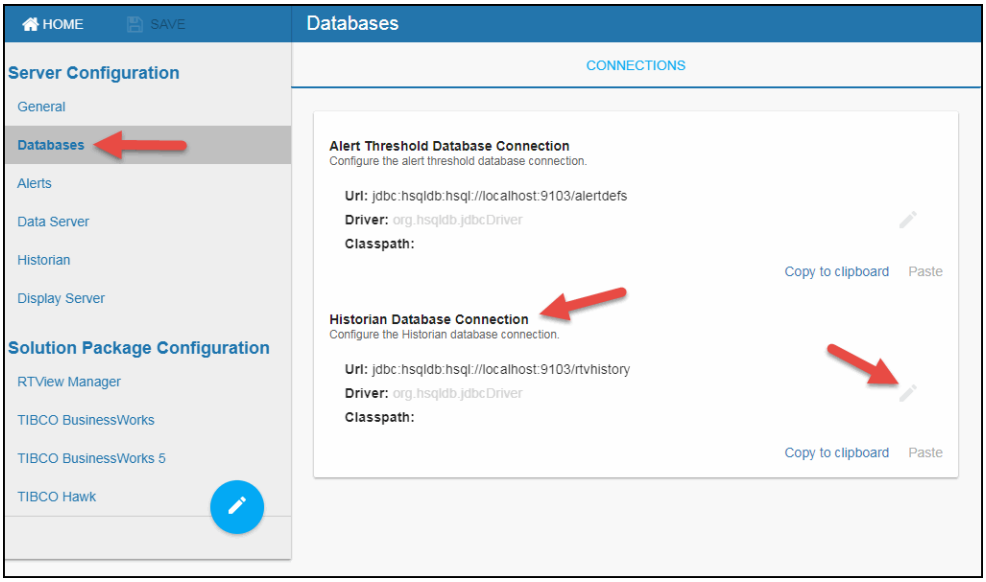
Username - Enter the username to enter into this database when making a connection.

Password - Enter the password to enter into this database when making a connection. If there is no password, use "**-**".

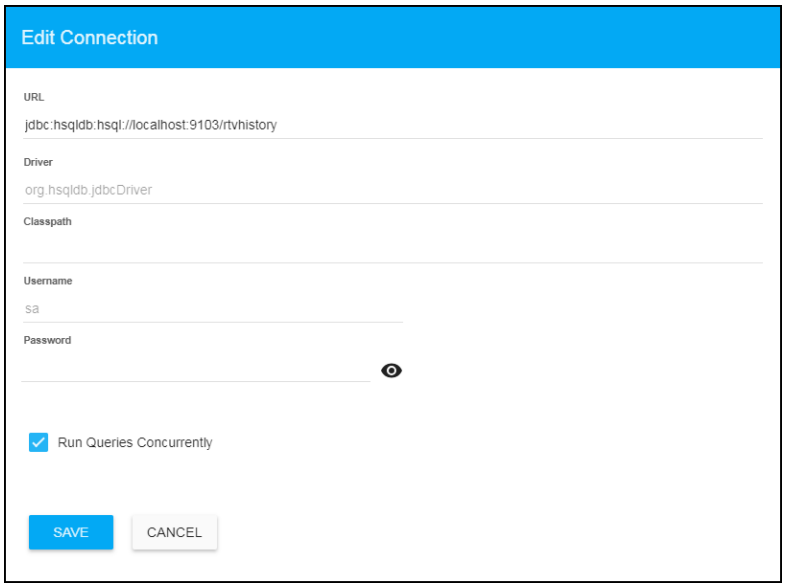
Run Queries Concurrently - Select this check box to run database queries concurrently.

5. Go back to the **database.properties** template file, which is located in the **rtvapm\common\dbconfig** directory, and find the line that corresponds to your supported database from the "Define the RTVHISTORY DB" section.

6. Navigate to the RTView Configuration Application > **(Project Name)** > **Server Configuration** > **Databases** and then click the Edit icon in the **Historian Database Connection** region.



The **Edit Connection** dialog displays.



7. Enter the information from Step 5 into the **Edit Connection** dialog and click **Save**.

URL - Enter the full database URL to use when connecting to this database using the specified JDBC driver.

Driver - Enter the fully qualified name of the JDBC driver class to use when connecting to this database.

Classpath - Enter the location of the jar where the JDBC driver resides in your environment.

Username - Enter the username to enter into this database when making a connection.

Password - Enter the password to enter into this database when making a connection. If there is no password, use "**-**".

Run Queries Concurrently - Select this check box to run database queries concurrently.

8. Click **Save** in the RTView Configuration Application and restart your data server.

9. Manually create database tables. If your configured database user has table creation permissions, then you only need to create the Alerts tables. If your configured database user does not have table creation permission, then you must create both the Alert tables and the History tables.

To create tables for your database, use the **.sql** template files provided for each supported database platform, which is located in the **dbconfig** directory of the **common**, **bwmon**, **bw6mon**, and **rtvmgr** directories:

- **Alerts**

rtvapm/common/dbconfig/create_common_alertdefs_tables_<db>.sql

- **History**

rtvapm/bwmon/dbconfig/create_bwmon_history_tables_<db>.sql

rtvapm/bw6mon/dbconfig/create_bwmon_history_tables_<db>.sql

rtvapm/rtvmgr/dbconfig/create_rtvmgr_history_tables_<db>.sql

where **<db>** ={**db2**, **mysql**, **oracle**, **sqlserver**}

NOTE: The standard SQL syntax is provided for each database, but requirements can vary depending on database configuration. If you require assistance, consult with your database administrator.

The most effective method to load the **.sql** files to create the database tables depends on your database and how the database is configured. Some possible mechanisms are:

- **Interactive SQL Tool**
  Some database applications provide an interface where you can directly type SQL commands. Copy/paste the contents of the appropriate **.sql** file into this tool.
- **Import Interface**
  Some database applications allow you to specify a **.sql** file containing SQL commands. You can use the **.sql** file for this purpose.

Before loading the **.sql** file, you should create the database and declare the database name in the command line of your SQL client. For example, on MySQL 5.5 Command Line Client, to create the tables for the Alert Settings you should first create the database:

create database myDBName;

before loading the **.sql** file:

mysql -u myusername -mypassword myDBName < create_common_alertdefs_tables_mysql.sql;

If you need to manually create the Historical Data tables, repeat the same process. In some cases it might also be necessary to split each of the table creation statements in the **.sql** file into individual files.

Third Party Application

If your database does not have either of the two above capabilities, a third party tool can be used to enter SQL commands or import **.sql** files. Third party tools are available for connecting to a variety of databases (RazorSQL, SQLMaestro, Toad, for example).

You have finished configuring the databases. Proceed to Configure Alert Notification.

Configure Alert Notification

This section describes how to configure alerts to execute an automated action (such as sending an email alert). To setup alert notification you select the event you want to notify on and then select the action to execute.

You set alerts to execute notifications based on the following events:

- when a new alert is created
- the first time the **Severity** level on an alert changes
- when an alert is cleared
- periodically renotify unacknowledged alerts

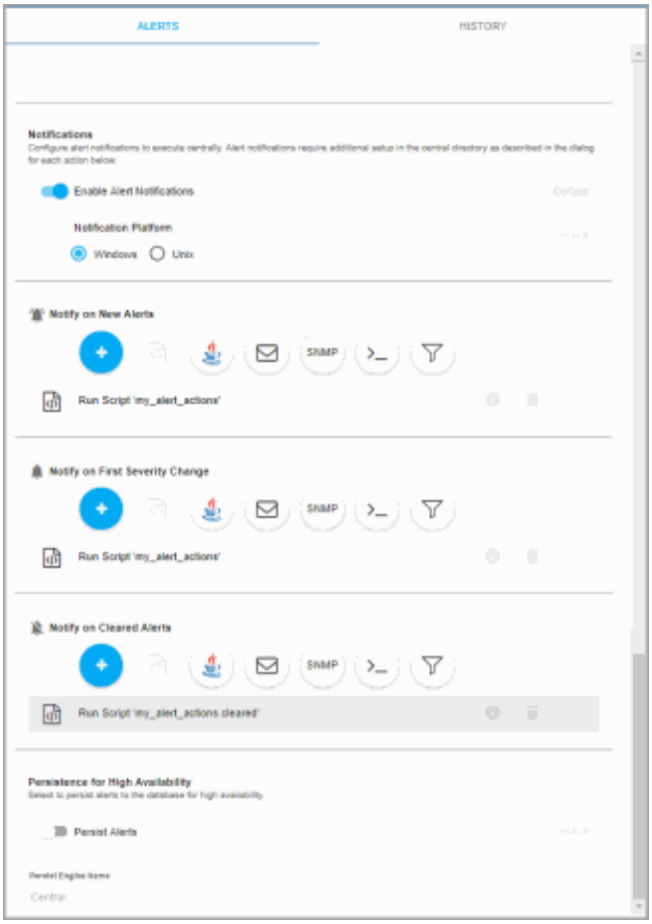
By default, a **.bat** script is executed for new alerts and on the first severity change for an alert. The script, by default, is not configured to execute an automated action. However, you can uncomment a

line in the script that prints alert data to standard output. Or, you can modify the script to execute an automated action (such as sending an email alert). The following is a sample output from the alert command script:

----- Alert command script executed: DOMAINNAME=MYMON-1, ALERTNAME=someAlert, ALERTINDEX-X=alertIndex1~alertIndex2, ALERTID=1075, ALERTSEVERITY=2, ALERTTEXT=High Alert Limit exceeded current value: 100.0 limit: 80.0 #####

To configure Alert Notification:

1. Open the RTView Configuration Application, select **Alerts** (in the navigation tree) and then the **Alerts** tab.



2. Toggle on **Enable Alert Notifications** and select the **Notification Platform** type (**Windows or Unix**).

3. Select an alert event that you want to notify on by clicking  next to the option.

Alert Event Options

- **Notify on New Alerts**:A notification is executed every time a new alert is created.
- **Notify on First Severity Change**: A notification is executed the first time the **Severity** changes for each alert.
- **Notify on Cleared Alerts**: A notification is executed every time an alert is cleared.

- **Periodically Renotify on Unacknowledged Alerts**: Enter the **Renotification Interval** (number of seconds). A notification is executed for each unacknowledged alert per the interval you specify here. If the Renotification Interval is greater than **0** and no actions are defined, the **New Alerts** action will be used for renotifications.

4. Select the alert action(s) you want to execute.

Alert Action Options


Run a Script


Execute Java Code

Add Email Notification


Send SNMP Trap


Run Command String


Conditional Filter

You can choose multiple actions.

5. Click  to close the dialog and  (in title bar) to save your changes.

6. Some alert notification actions require additional setup as described in the dialog for each action. See the descriptions of each action below for details on the dialogs and additional setup for each action.

7. Click  to apply changes.

Run a Script

This alert notification action executes the following script in the **projects/rtview-server** directory for RTViewCentral and in the **projects/rtview-manager** directory for RTView Manager:

- **my_alert_actions.bat/sh** – New and First Severity Change
- **my_alert_actions.cleared.bat/sh** – Cleared
- **my_alert_actions.renotify.bat/sh** – Periodically Renotify

This action can only be added once per notification type. In addition to selecting this action in the Configuration Application, you must also modify the appropriate script to execute the actions for your notification. This script has access to the following fields from the alert: **Alert Name**, **Alert Index**, **ID**, **Alert Text** and **Severity**.

Return to Alert Event Options.

Execute Java Code

This alert notification action allows you to implement your alert notification actions using Java code. It executes the **my_alert_notification.$domainName.$alertNotifyType.$alertNotifyCol** command in your Custom Command Handler and passes the row from the alert table that corresponds to the alert.

This action can only be added once per notification type. In addition to selecting this action the Configuration Application you must also modify the custom command handler to execute the actions for

your notification. A sample custom command handler is included under **projects/custom**. It prints the alert notification to the console. You will modify this command handler to implement your own notification actions.

Make the following entries:

- **Custom Command Handler Class Name**: Enter the fully qualified name of the Custom Command Handler class. This defaults to the sample Custom Command Handler in the **projects/custom** directory.
- **Custom Command Handler Jar**: Enter the path and name of the jar containing the Custom Command Handler class. The path may be absolute or relative to the location of data server. This defaults to the sample Custom Command Handler in the **projects/custom** directory.

Note that if you can only have one custom command handler per Data Server, so changing these settings for one notification event will change them for the rest of the notification events.

Customizing the Custom Command Handler

The source for the Custom Command handler is provided in the **RtvApmCommandHandler.java** file, located in the **RTViewEnterpriseMonitor\projects\custom\src\com\sl\rtvapm\custom** directory. By default, the handler prints the alert data to standard output. To change this behavior perform the following steps:

1. Open the **RtvApmCommandHandler.java** file.
2. Modify the **OutputAlertString** method as needed. You can replace this method with your own if you modify the **invokeCommand** method to call it, and your method accepts the same arguments as **OutputAlertString**.
3. Save the **RtvApmCommandHandler.java** file.
4. Compile **RtvApmCommandHandler.java** and rebuild **rtvapm_custom.jar** using the supplied script (**make_all.bat** or **make_all.sh**) in **projects\custom\src** directory.

Return to Alert Event Options.

Add Email Notification

This alert notification action sends an email. This action can be added multiple times per notification type. No additional setup is required beyond filling in the **Add Email Notification** dialog in the Configuration Application.

Make the following entries:

- **SMTP Host**: The SMTP host address. This is required. Consult your administrator.
- **SMTP Port**: The SMTP port number. This is required. Consult your administrator.
- **From**: The email address from which to send the email. This is required.
- **To**: The email address to which to send the email. This is required and may contain multiple entries.
- **Subject**: The subject for the email. This is required. You can include the value from any column in the alert table in your subject. Click **Insert $alert<Value>** and select one or more applicable alert value(s).
- **Body**: The body of the email. This is optional. Click **Insert $alert<Value>** and select one or more applicable alert value(s).

- **User**: The user name for the account from which you are sending the email. This is optional.
- **Password**: The password for the account from which you are sending the email. This is optional.

Return to Alert Event Options.

Send SNMP Trap

This alert notification action sends an SNMP Trap as described in **rtvapm/common/lib/SL-RTVIEW-EM-MIB.txt**. This action can be added multiple times per notification type. No additional setup is required beyond filling in the **Add SNMP Trap Notification** dialog in the Configuration Application.

Make the following entries:

- **Trap Type**: Select the SNMP version of the trap. This is required.
- **Destination Address**: The system name or IP address of the receiving system. This is required.
- **Destination Port**: The UDP port on the receiving system. This is required.
- **Community Name**: (This field is visible when **Trap Type v2/v3** is selected.) The SNMP v2 Community Name string. This is required.

Return to Alert Event Options.

Run Command String

This alert notification action executes a specified command. This action can be added multiple times per notification type. Make the following entry:

Command String: Enter the command string for any command supported by RTView. To enter a command string, you must know the correct syntax for the command. Contact Technical Support for assistance on syntax. You can include the value from any column in the alert table using the syntax in the Show More link at the bottom of the dialog.

Return to Alert Event Options.

Conditional Filter

This alert notification action alert allows you to execute different actions for different alerts based on information in the alert. For example, you can configure EMS alerts to send emails to your EMS team and Solace alerts to send emails to your Solace team. This action can be added multiple times per notification type.

To create a condition, make the following entries:

- **Alert Field**: Select an alert field: **Alert Name**, **Alert Index**, **Category, CI Name, Owner, Package, Primary Service** or **Severity**. This is required.
- **Operator**: Select one - **EQUALS**, **DOES NOT EQUAL**, **STARTS WITH**, **ENDS WITH** or **CONTAINS**. This is required.
- **Value**: Enter the value to which to compare the Alert Field. Cannot contain wildcard characters. This is required.
- **Action(s)**: Select one or more actions to execute when this condition is met - Run a Script , Execute Java Code , Send SNMP Trap , Add Email Notification or Run Command String.

Return to Alert Event Options.

Configure High Availability

High Availability (HA) mitigates single point of failure within BW Monitor by providing a means of defining redundant system components, together with failover capability, for users of those components.

When using HA, components are designated **PRIMARY** and **BACKUP**. If the **PRIMARY** component fails, failover occurs to the **BACKUP** component. If the **PRIMARY** component is subsequently restarted, the **BACKUP** component allows the newly restarted component to take the primary role and return to its backup role.

This section contains the following:

Overview of High Availability Architecture

Data Server High Availability

The primary and backup data servers connect to each other via socket. If the primary data server stops, then the backup server takes over. If the primary then comes back online, then the primary takes over again and the backup returns to standby mode. The data client connections will move between the two servers accordingly.

NOTE: Be aware that data clients can connect to the standby server using a non-fault tolerant URL and still get data because of a proxy feature where the standby server forwards data requests to the primary server. This can be confusing when you use the HTML Cache Viewer (**http://localhost:3370/common**) on the standby server to view cache contents because it looks like the standby server caches are updating, but you are really viewing the data in the primary server and not in the standby server.

Display Server High Availability (Classic User Interface)

In display server deployments, the primary display server and backup display server do not connect to each other. The rtvdisplay servlet is configured to connect first to the primary and, if that fails, it tries to connect to the backup. At any point, if the one it is connected to becomes unavailable, then it will try to connect to the other. You can configure whether to have the rtvdisplay server connect back to the primary server when it comes back online or stay connected to the backup server until it goes offline.

HTML User Interface High Availability

The HMTL UI client connects to the data server via an HA configured rtvquery servlet.

Historian High Availability

The primary and backup historian connect to each other via socket. If the primary historian stops, then the backup takes over. If the primary historian comes back online, then the primary takes over again and the backup returns to standby mode. Only the active historian writes to the database.

The historian is a data client of the data server and connects to it via a fault tolerant URL (socket only), which means that the data servers and historians can fail over separately or together.

Requirements for Configuring High Availability

The following are minimum requirements for High Availability:

- Two host machines, one for the primary host and one for the backup host.
- Both hosts must be configured such that the RTView processes on each host can connect to each other via socket.
- Both hosts must be able to access:
  - the same data connections
  - the same historian database
  - the alert threshold database
- The RTView processes on both hosts must be able to run against identical properties files. In the case where drivers or other third party jars are located in different directories on the two hosts, create a directory in the same location in each host, copy the jar files into and reference that directory in your properties.
- Tomcat or other Application Server
  - The HTML UI and rtv servlets must be deployed on an application server other than the internal Jetty server. Note that this requires extra configuration of the servlet **.war** files in the application server.

Steps for Configuring High Availability

To Configure High Availability:

1. On both the primary and backup hosts, define the following environment variables:
   - **PRIMARYHOST** - the IP Address or hostname of the host running the primary servers (for example, **set PRIMARYHOST=MyHost**).
   - **BACKUPHOST** - the IP Address or hostname of the host running the backup servers (for example, **set BACKUPHOST=OtherHost**).
2. Install the Monitor on both the primary host and the backup host.
3. Configure your servlets to be HA and deploy them to your application server:
   - **cd projects\rtview-server**

- In a text editor, open **update_wars**(**.bat** or **.sh**) and fill in the values for **HOST**, **HA_HOST**, **HA_DISPLAYHOST**, and **HA_FAILBACK** as described in the script.
- Run the **update_wars**(**.sh** or **.bat**) script.
- Copy the generated war files to the **webapps** directory of your application server.

4. To run High Availability, you must run the following from the command line:

Windows

- From the command line on the primary host, **cd** to <top level directory> and type **start_server -haprimary**.
- From the command line on the backup host, **cd** to <top level directory> and type **start_server -habackup**.

Unix

- From the command line on the primary host, **cd** to <top level directory>and type **start_server.sh -haprimary**.
- From the command line on the backup host, **cd** to <top level directory> and type **start_server.sh -habackup**.

5. Configure the Monitor on the primary host using the RTView Configuration Application (see Quick Start and RTView Configuration Application for more information on configuring the Monitor). Make sure to configure data collection, configure server options and databases, and enable alert persistence.

Note that the RTView Configuration Application must be able to connect both the primary and backup servers in order to enable editing. The same properties are saved to both servers. The **RESTART SERVERS** button (in the RTView Configuration Application) restarts both the primary and backup servers at the same time. If you want to stagger the restarts, use the scripts under **TIB_rtview-bw** to stop and then start your servers after making changes in the RTView Configuration Application.

Note: Jetty does not have to be disabled, but data clients will not be able to make high availability connections to the data server using the Jetty URL. However, the Jetty URL can still be used to configure the application.

Verifying the High Availability Configuration

Verify failover and failback configurations by looking for the following in the log files.

Note: If the PRIMARYHOST and/or BACKUPHOST environment variable(s) is/are not set, you will get the following error in the log files and HA will be disabled:

```
ERROR: Disabling HA because the PRIMARYHOST and/or BACKUPHOST environment
variable is not set.
```

Primary Data Server Log File

```
startup

[rtview] Starting as primary HA data server accessible via
//primaryhostname:3378,//backuphostname:3378

[rtview] DataServerHA: connected to backuphostname:3378
```

```
[rtview] DataServerHA: run as primary server, backuphostname:3378 has lower
priority than this server

[rtview] leaving standby mode
```

Backup Data Server Log File

```
startup

[rtview] Starting as backup HA data server accessible via
//primaryhostname:3378,//backuphostname:3378

rtview] entering standby mode

after failover (primary data server exits)

[rtview] DataServerHA: error receiving message: java.net.SocketException:
Connection reset (primaryhostname:3378)

[rtview] DataServerHA: becoming primary server, lost connection to primary server
primaryhostname:3378

[rtview] leaving standby mode

after failback (primary data server comes back up)

[rtview] DataServerHA: resigning as primary server, got standby directive from
other server primaryhostname:3378

[rtview] connected to primaryhostname:3378

[rtview] entering standby mode
```

Primary Historian Log File

```
[rtview] Starting as primary HA historian paired with backup historian at
<backuphostname>:3322

[rtview] ServerGroup: status of member <backuphostname>:3322: primary, priority= 1,
started=Wed Nov 14 12:56:01 PST 2018

[rtview] ServerGroup: primary server = local

[rtview] ServerGroup: becoming primary server
```

Backup Historian Log File

```
[rtview] Starting as backaup HA histoiran paired with primary historian at
<primaryhostname>:3322

[rtview] ServerGroup: status of member <primaryhostname>:3322: primary, priority= ,
started=Wed Nov 14 12:56:01 PST 2018

[rtview] ServerGroup: primary server = <primaryhostname>:3322

after failover (primary historian exits):

[rtview] error receiving message: java.io.EOFException (primaryhostname:3322)

[rtview] ServerGroup: disconnected from primaryhostname:3322
```

```
[rtview] ServerGroup: primary server = local

after failback (primary historian starts back up):

[rtview] ServerGroup: status of member primaryhostname:3322: primary, priority= 2,
started= Tue Nov 20 09:12:43 PST 2018

[rtview] ServerGroup: connected to primaryhostname:3322

[rtview] ServerGroup: primary server = primaryhostname:3322
```

Primary Display Server Log File

```
2018-11-19 14:08:09,366 INFO main - [rtview] Starting as primary HA display server
paired with backup display server on <backuphostname>
```

Backup Display Server Log File

```
2018-11-19 14:08:09,366 INFO main - [rtview] Starting as backup HA display server
paired with primary display server on <primaryhostname>
```

CHAPTER 4 Deployment

This section describes how to deploy the Monitor components. This section includes:

Overview

The Monitor is deployed as a web application that runs in a browser. By default, the web application is run in Eclipse Jetty hosted in a data server, though you can optionally use your own application server instead. Evaluation environments can use the provided HSQLDB database. Production environments require a supported JDBC- or ODBC-enabled relational database to store historical information. Supported databases are MySql, Oracle, SqlServer and DB2.

The RTView Historian and RTView Data Server are typically deployed on the same host. However, these processes can optionally be configured on separate hosts. Doing so can increase performance in deployments that need to support many end users or systems with large TIBCO servers.

To deploy the Monitor as a web application:

- Web Application Deployment: Clients need only a browser and Adobe Flash installed to run as a web application. The RTView Display Server, RTView Data Server, RTView Historian and Application Server are typically installed on the same host.

To configure an RTView process to run as a Windows Service

- RTView Server Components as Windows Services: The RTView Data Server, Historian, and Display Server can be run as Windows Services.

Web Application Deployment

This section describes how to deploy the Monitor as a web application. You start the Monitor using the **start_rtv** script (and stop the Monitor using the **stop_rtv** script). For web application deployments the following processes are started: the RTView Data Server, Historian, and Display Server, as well as the database and an application server.

This section contains:

Windows

Note: You can skip Step 1 and Step 2 if you are using Eclipse Jetty, which is delivered with the Monitor, as your application server.

1. Copy the **.war** files, located in the **rtvapm\<*mon>\webapps** directory, where **<*mon>** is the Monitor you are deploying (for example, **bw6mon, bwmon**, **tbemon** and **emsmon**), and deploy them to your Application Server.

2. Start your Application Server if using Tomcat or an application server other than Eclipse Jetty.

3. Initialize a command window. Go to your Monitor installation directory and type:

   rtvapm_init

4. You can skip this step if you are using Eclipse Jetty. The RTView Configuration Application uses digest authentication for security, and only allows access to users with the "rtvadmin" role. In order to allow access to the RTView Configuration Application in your application server, you need to add a user with the "rtvadmin" role. For example, if using Tomcat, follow the instructions below. For other application servers, refer to their documentation for adding users.
   - Edit <installation directory>\conf\tomcat-user.xml
   - Add the following lines inside the tomcat-users tag:

```
<role rolename="rtvquery"/>
<user username="rtvquery" password="rtvadmin" roles="rtvquery"/>
```

5. Change directory (**cd**) to the **<*mon>\projects\mysample** directory.

6. In the **projects\mysample** directory, start the Monitor applications by typing:

   start_rtv all

NOTE: The **start_rtv all** command starts all the Monitor applications at once. You can start each Monitor application individually: **start_rtv default database** starts the default HSQLDB database (suitable for testing purposes), **start_rtv default dataserver** starts the Data Server, **start_rtv default historian** starts the Historian and **start_rtv default displayserver** starts the Display Server. Use the **stop_rtv** script to stop Monitor applications.

7. Open a Web browser and browse to the following URL to open the Monitor:

   If using Eclipse Jetty as your application server:

   http://localhost:3370/bwmon (for Version 5)

   or

   http://localhost:4570/bw6mon (for Version 6)

   If using your own application server:

   http://host:port/<*mon>

   Where **host** is the IP or host name where your Application Server is running, **port** is the port used by your Application Server and **<*mon>** is the Monitor you are deploying. The login display opens in the Web browser.

   Login. The default user name and password are:

User Name: **demo**

Password: **demo**

The main Monitor display opens.

UNIX/Linux

Note: You can skip Step 1 and Step 2 if you are using Eclipse Jetty, which is delivered with the Monitor, as your application server.

1. Copy the **.war** files, located in the **rtvapm/<*mon>/webapps** directory, where **<*mon>** is the Monitor you are deploying (for example, **bw6mon**, **bwmon**, **tbemon** and **emsmon**), and deploy them to your Application Server.

   Note: You can skip this step if you are using Eclipse Jetty, which is delivered with the Monitor, as your application server.

2. Start your Application Server if using Tomcat or an application server other than Eclipse Jetty, which is delivered with the Monitor.

3. Initialize a terminal window. The script used to initialize a terminal window depends on whether you are in csh or rsh (e.g. Linux, Mac OS X). With a Bourne shell, open a terminal window, go to your Monitor installation directory and type:

   ./rtvapm_init.sh

4. You can skip this step if you are using Eclipse Jetty. The RTView Configuration Application uses digest authentication for security, and only allows access to users with the "rtvadmin" role. In order to allow access to the RTView Configuration Application in your application server, you need to add a user with the "rtvadmin" role. For example, if using Tomcat, follow the instructions below. For other application servers, refer to their documentation for adding users.
   - Edit <installation directory>\conf\tomcat-user.xml
   - Add the following lines inside the tomcat-users tag:

```
<role rolename="rtvquery"/>
<user username="rtvquery" password="rtvadmin" roles="rtvquery"/>
```

5. Change directory (**cd**) to the **<*mon>/projects/mysample** directory.

6. In the **projects/mysample** directory, start the Monitor applications by typing:

   start_rtv.sh all

NOTE: The **start_rtv.sh all** command starts all the Monitor applications at once. You can start each Monitor application individually: **start_rtv.sh default database** starts the default HSQLDB database (suitable for testing purposes), **start_rtv.sh default dataserver** starts the Data Server, **start_rtv.sh default historian** starts the Historian and **start_rtv.sh default displayserver** starts the Display Server. Use the **stop_rtv.sh** script to stop Monitor applications.

7. Open a Web browser and browse to the following URL to open the Monitor:

   If using Eclipse Jetty as your application server:

   http://localhost:3370/bwmon (for Version 5)

   or

   http://localhost:4570/bw6mon (for Version 6)

If using your own application server:

http://host:port/<*mon>

Where **host** is the IP or host name where your Application Server is running, **port** is the port used by your Application Server and **<*mon>** is the Monitor you are deploying. The login display opens in the Web browser.

Login. The default user name and password are:

User Name: **demo**

Password: **demo**

The main Monitor display opens.

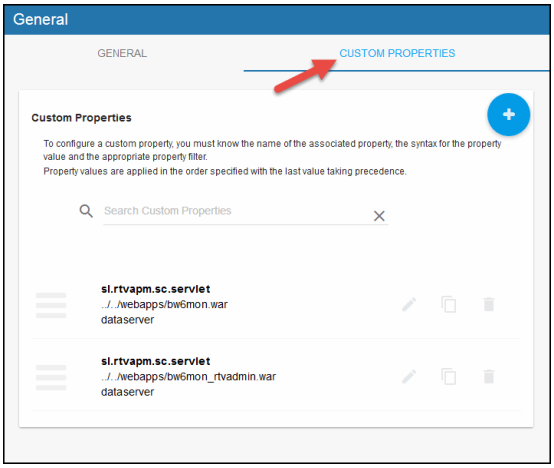
See Quick Start for a more detailed example.

RTView Server Components as Windows Services

This section describes how to configure an RTView process (Data Server, Historian, Display Server) to run as a Windows service. To illustrate, we use the EMS Monitor in our examples.

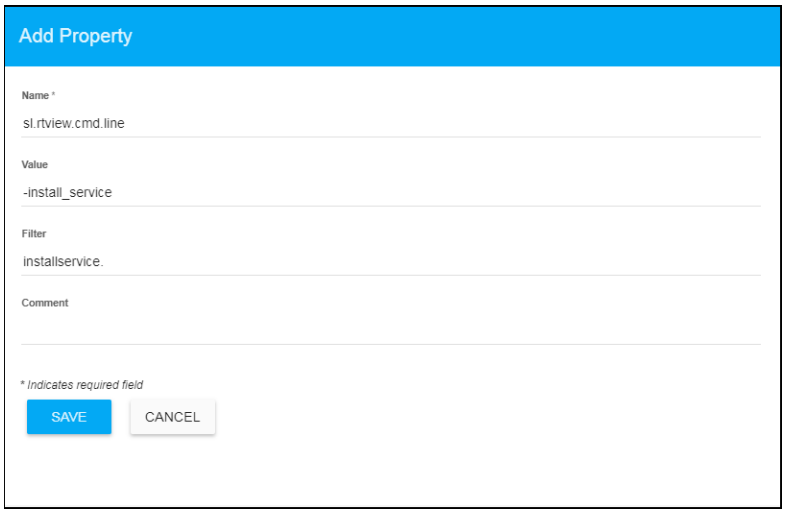
To Configure the Data Server, Historian or Display Server to run as a Windows Service

1. Navigate to the RTView Configuration Application > **(Project Name)** > **Server Configuration** > **General** > **Custom Properties** tab.



2. Click the icon.

The **Add Property** dialog displays.



3. Define the values for each of the following properties and click **Save**:

Name: sl.rtview.cmd_line
Value: -install_service
Filter: installservice
Comment: (description of the filter)

Name: sl.rtview.cmd_line
Value: -dir:%RTVAPM_STARTUP%
Filter: installservice
Comment: (description of the filter)

Name: sl.rtview.cmd_line
Value: -uninstall_service
Filter: uninstallservice
Comment: (description of the filter)

Note: The environment variable %RTVAPM_STARTUP% is set by run script to the directory where the script was started.

4. For each Windows service you want to create, add the following property and replace ServiceName in the value and filter fields with a name you choose for the service:

Name: sl.rtview.cmd_line
Value: -service:ServiceName
Filter: ServiceName

For example, (using BW Monitor to illustrate), choose BWMonData as the name for starting a Data Server as a Windows service and BWMonDisp to indicate a name for starting a Display Server as a Windows service.

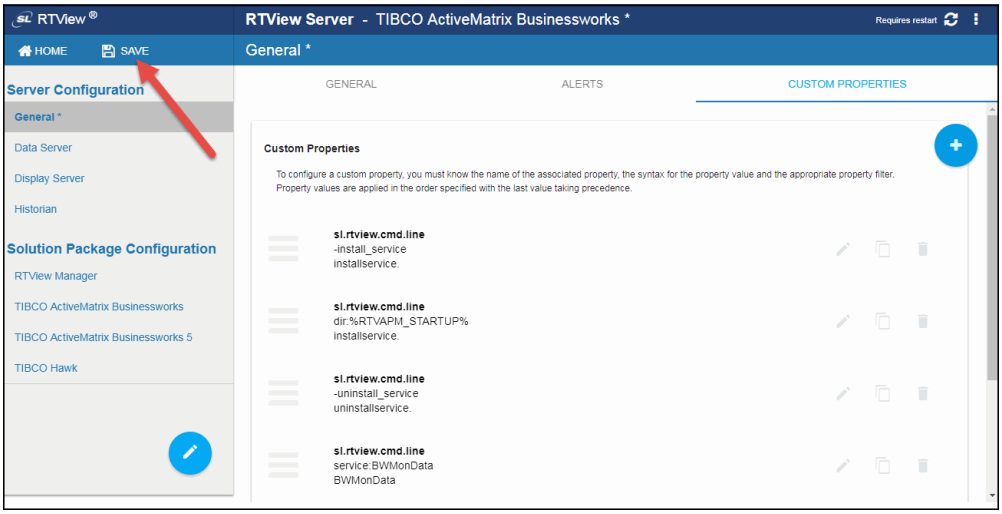
Name: sl.rtview.cmd_line

Value: -service:BWMonData
Filter: BWMonData

Name: sl.rtview.cmd_line
Value: -service:BWMonDisp
Filter: BWMonDisp

Note: Each service must have a unique name and the beginning of the property entered must match the name of the service.

5. Once all your properties have been added, click **Save**.



6. Restart the data server so that your changes take effect.

7. **To install and run**

Execute the following scripts to start the service:

NOTE: These scripts must be run in an initialized command window.

rundata -propfilter:installservice -propfilter:EMSMonData
rundisp -propfilter:installservice -propfilter:EMSMonDisp

8. **To uninstall**

Execute the following scripts to uninstall the services:

NOTE: These scripts must be run in an initialized command window.

rundisp -propfilter:uninstallservice -propfilter:EMSMonDisp
rundata -propfilter:uninstallservice -propfilter:EMSMonData

Troubleshooting

Log Files

When the Monitor encounters an error, Monitor components output an error message to the console and/or to the corresponding log files. Log files are located in the project directory**\logs** directory. Look at the following log files on the machine where the components are running:

- **dataserver.log**
- **displayserver.log**
- **historian.log**

To enable logging, make sure the directory logs exists in your project directory.

Verifying Data Received from Data Server

Open the **Cache Viewer Display** to verify data is arriving correctly from the Data Server. To access the **Cache Viewer Display**, choose **Administration** in the navigation tree, then choose **RTView Cache Tables** display or the **RTView Cache Overview** display. You should see all caches being populated with monitoring data (number of rows > 0). Otherwise, there are problems with the connection to the Data Server.

Restarting the Data Server

If the Display Server or the Historian fails to connect to the Data Server or receives no data, verify the ports are assigned correctly in your properties files and then restart the Data Server.

Multiple Distributed Data Servers

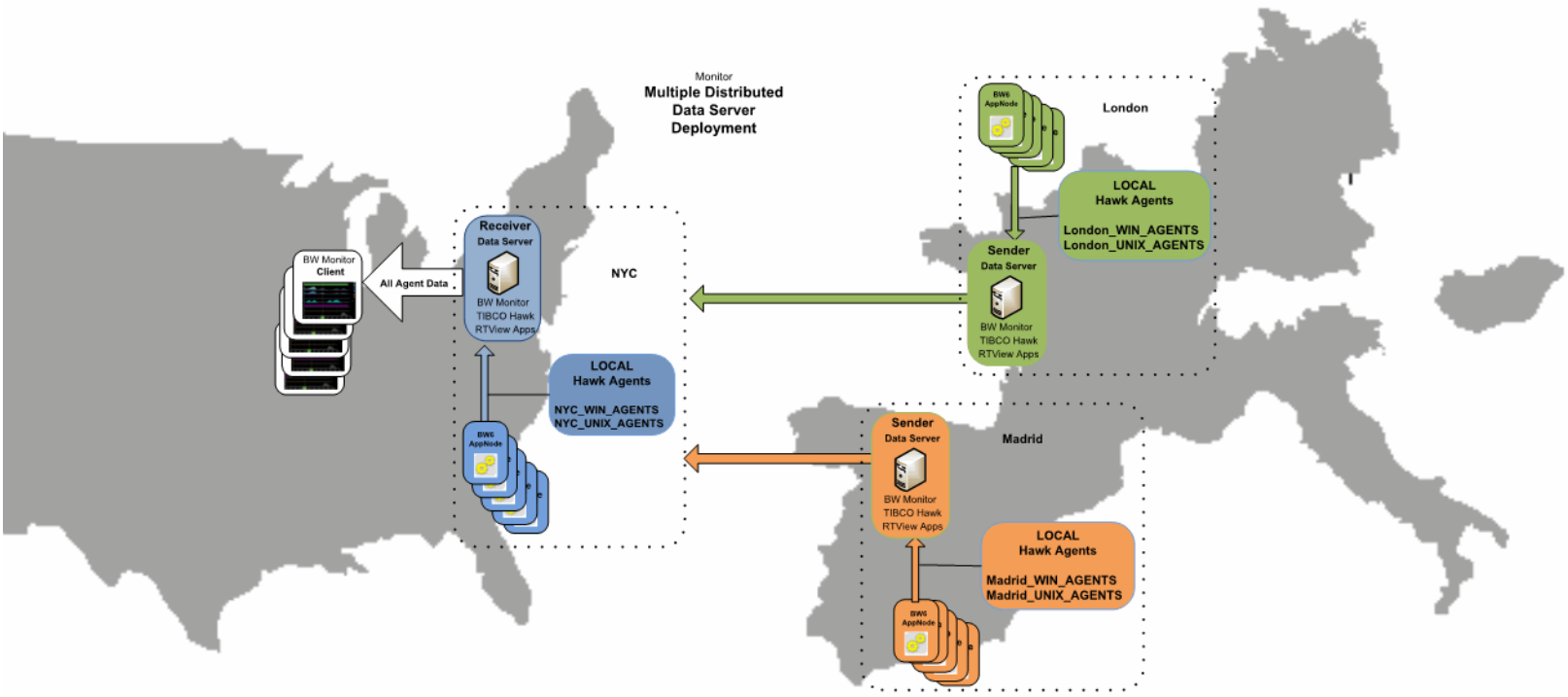
If you have systems running outside your LAN subnet, this deployment option enables you to centralize the Monitor deployment while locating each Data Server near the servers they monitor, thereby optimizing Monitor performance.

> **Note:** If the Monitor runs on a single server and monitors servers within a single, local subnet, you do not need to consider this type of deployment.

To set up this type of deployment, determine which Data Server is to be the central Data Server and configure all the other Data Servers to be the senders. You also need to configure each Data Server (receiver and senders) to automatically detect and gather data strictly from their local Hawk Agents. The sender Data Servers then send their local Hawk Agent data to the receiver Data Server. The receiver Data Server aggregates all data received and makes it available to Monitor displays.

For example, the following illustrates a Monitor deployment for a company headquartered in New York City. The company has systems running locally, as well as outside the LAN in London and Madrid. The NYC Data Server is designated as the receiver, and the London and Madrid Data Servers are designated as senders.

Note: Each Data Server gathers data only for its LAN.



NYC Data Server (Receiver)	London Data Server (Sender)	Madrid Data Server (Sender)
• Automatically detects and gathers data for its local Agents (**NYC_WIN_ AGENTS** and **NYC_UNIX_ AGENTS**). • Receives data from London and Madrid senders. • Aggregates data. • Provides data to the Monitor displays. Can see all Agents local to London and Madrid Data Servers, but is configured only to obtain data from its local systems.	• Automatically detects and gathers data for its local Agents (**London_WIN_ AGENTS** and **London_ UNIX_AGENTS**). • Sends data from its local Agents to NYC Data Server.	• Automatically detects and gathers data for its local Agents (**Madrid_WIN_ AGENTS** and **Madrid_UNIX_ AGENTS**). • Sends data from its local Agents to NYC Data Server.

Receiver Configuration

This section assumes you have already installed TIBCO BusinessWorks (Version 5 or 6) on the system where you will be running the receiver, and also that you have created a project directory. See Setup for information on installing TIBCO BusinessWorks and Quick Start for an example of how to configure TIBCO BusinessWorks.

Set up your receiver data server

1. Start the project using **start_server**. See Quick Start for more information.

2. By default, the receiver is setup to receive data on port 3372 (for Version 5) or 3272 (for Version 6). If your senders cannot access the system on which the receiver is running, they can send data to the rtvagent servlet instead, which will forward the data to the receiver. To deploy the rtvagent servlet:

   If you are using Eclipse Jetty (the default application server):

   There are no required steps.

   If you are using Tomcat/a different application server:

   Copy all of the **.war** files located in the **TIB_rtview-bw\projects\rtview-server** directory to the Tomcat **webapps** directory.

3. Remove any TIBCO Hawk connections that will be serviced by a sender in the RTView Configuration Application > (**PROJECT NAME) > Solution Package Configuration** > **TIBCO Hawk** > **CONNECTIONS**. If all connections will be serviced by senders, any connections created in the **CONNECTIONS** tab need to be removed. See Configure for RTView Manager for more information.

4. Restart the project using **stop_server** and **start_server**. See Quick Start for more information.

Collector Configuration

This section assumes you have already installed TIBCO BusinessWorks (Version 5 or 6) on the system where you will be running the sender, and also that you have created a project directory. See Setup for information on installing TIBCO BusinessWorks and Quick Start for an example of how to configure TIBCO BusinessWorks. You can run as many senders on as many systems as needed.

1. In the **rtvservers.dat** file located in your project directory, add **-propfilter:sender** to the end of the dataserver line and comment out the display server, historian and database processes as follows (since they are not used by sender data servers):

```
default . dataserver rundata -propfilter:sender
#default . historian runhist -ds
#default . displayserver rundisp -ds
#default . database rundb
```

2. Start the sender project using **start_server**. See Quick Start for more information.

   Note: If you are running multiple senders on the same system or running the sender on the same system as the receiver, you need to change the port prefix for the sender so that you do not get a port conflict. To do so, use the following on the command line as follows: **-portprefix:XX** where XX is the port prefix. To save this to your properties file so you do not need to specify it on the command line, add the **-saveportprefix** command line option. For example: **-portprefix:55 -saveportprefix**

3. Open the RTView Configuration Application > (**Project Name**) > **Server Configuration** > **Data Servers** > **COLLECTOR** tab.

4. In the **Targets** region, click the  icon to add a target as follows:

   ID: A unique name for the target.

URL: Specify the URL for the receiver. The url can be **host:port** (for example, somehost:3372 or somehost:3272) or an **http url** for the rtvagent servlet on the receiver (for example, **http://somehost:8068/rtview-bwmon-rtvagent**.

Targets: Select the **All solution packages** option.

Enabled: Select this check box to enable the target.

5. Click **Save** to exit the **Add Target** dialog.

6. Fill in a unique value for this sender in the **Identifier** field on the **COLLECTOR** tab. This should be unique across all senders.

7. Click on the RTView Configuration Application > (**Project Name**) > **Solution Package Configuration** > **TIBCO Hawk** > **CONNECTIONS** tab and verify that this sender is configured to collect only from its local Hawk connections (see Configure for RTView Manager for more information).

   Note: Perform this step only if you are using Version 5 or if you are using TIBCO Hawk with Version 6 instead of the OSGI plugin.

8. If you changed the port prefix in step 2, click on the RTView Configuration Application > (**Project Name**) > **Server Configuration** > **General** > **GENERAL** tab and confirm the port prefix is set to the correct value. If not, modify it accordingly.

9. Click **Save** in the RTView Configuration Application and restart your project using **stop_server** and **start_server**. See Quick Start for more information.

CHAPTER 5 Using the Monitor -- HTML Displays

This section describes Monitor features, graphs and functionality as well as Monitor displays. This section includes:

- Overview: Describes the Monitor features and functionality.
- TIBCO BusinessWorks Monitor HTML Views/Displays: Describes the Views and Displays available in TIBCO BusinessWorks.
- TIBCO BusinessWorks 5 Monitor HTML Views/Displays: Describes the Views and Displays available in TIBCO BusinessWorks 5.
- Drilldowns: Displays contained under this View are only available by clicking on buttons/links in other displays. You cannot access these displays directly via the left menu.
- Alerts: Displays the **Alert Table**, which presents detailed information about all alerts that have occurred in your system
- Admin: Provides additional alert and cache displays that enable you to set alert thresholds, observe how alerts are managed, and view internal data gathered and stored by RTView.

Overview

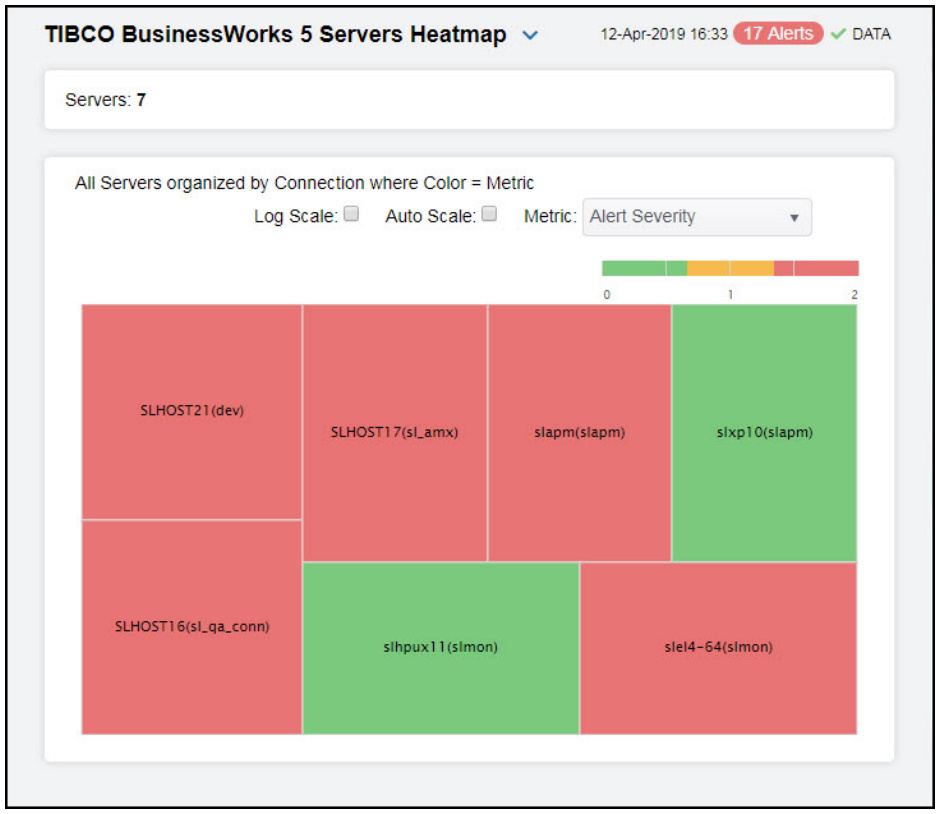
This section describes the main Monitor features, how to read Monitor objects, GUI functionality and navigation. This section includes:

- Heatmaps: Describes how to read heatmaps and heatmap functionality.
- Tables: Describes how to read tables and table functionality.
- Trend Graphs: Describes how to read trend graphs and trend graph functionality.
- Using the Monitor -- HTML Displays: Describes the top layer of the title bar shared by Monitor displays.
- Export Report: Describes how to export reports from the Monitor.

Heatmaps

Heatmaps organize your BW resources (servers, processes, and so forth) into rectangles and use color to highlight the most critical values in each. Heatmaps enable you to view various metrics in the same heatmap using drop-down menus. Each metric has a color gradient bar that maps relative values to colors. In most heatmaps, the rectangle size represents the number of resources in the rectangle; a larger size is a larger value. Heatmaps include drop-down menus to filter data by. The filtering options vary among heatmaps.

For example, each rectangle in the **TIBCO BusinessWorks 5 Heatmap** represents an application, where color is representative of the selected **Metric**.



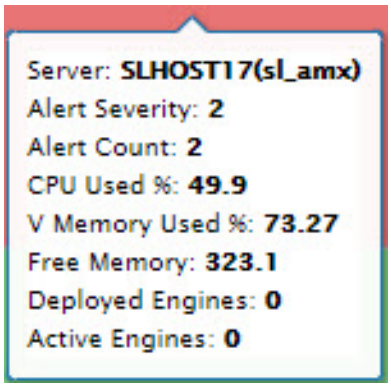
The **Metric** drop-down menu in this heatmap contains options to show **Alert Severity**, **Alert Count**, as well as other metrics. Menu options vary according to the data populating the heatmap. **Alert Severity** is selected and its corresponding color gradient  bar is shown. **Alert Severity** is the maximum level of alerts in the heatmap rectangle. Values range from **0** - **2**, as indicated in the color gradient  bar, where **2** is the highest **Alert Severity**:

- Red indicates that one or more resources associated with that application currently has an alert in an alarm state.
- Yellow indicates that one or more resources associated with that application currently have an alert in a warning state.
- Green indicates that no resources associated with that application have alerts in a warning or alarm state.

In most heatmaps, you can also drill-down to a *Summary* display containing detailed data for the resource (in this case, you drill-down to detailed data for the selected application in the **TIBCO BusinessWorks 5 Server Summary** display) by clicking on a rectangle. The drill-down opens a display that contains relevant and more detailed data.

Mouse-over

The mouse-over functionality provides additional detailed data in an over imposed pop-up window when you mouse-over a heatmap. The following figure illustrates mouse-over functionality in a heatmap object.



Log Scale

Typically, heat maps provide the Log Scale option, which enables visualization on a logarithmic scale. This option should be used when the range in your data is very broad. For example, if you have data that ranges from the tens to the thousands, then data in the range of tens will be neglected visually if you do not check this option. This option makes data on both extreme ranges visible by using the logarithmic of the values rather than the actual values.

Auto Scale

When checked, the values of the selected metric are auto-scaled to its highest defined value. When unchecked, the values of the selected metric display based on the threshold defined for the alert associated with the selected metric. Selecting **Auto Scale** helps to visualize the range of the values currently present for the selected metric instead of the threshold of the alert that has been associated with the metric. All metrics that have not been associated in the heatmap defaults with alerts use a monochromatic color gradient bar (whites and greens). All metrics that have been associated in the heatmap defaults with alerts use a multi-chromatic color gradient bar (reds, yellows, white, and greens).

Tables

Monitor tables contain the same data that is shown in the heatmap in the same View. Tables provide you a text and numeric view of the data shown in that heatmap, and additional data not included the heatmap. For example, the **TIBCO BusinessWorks 5 Servers Table** display (shown below) shows the same data as the **TIBCO BusinessWorks 5 Servers Heatmap** display (shown previously).

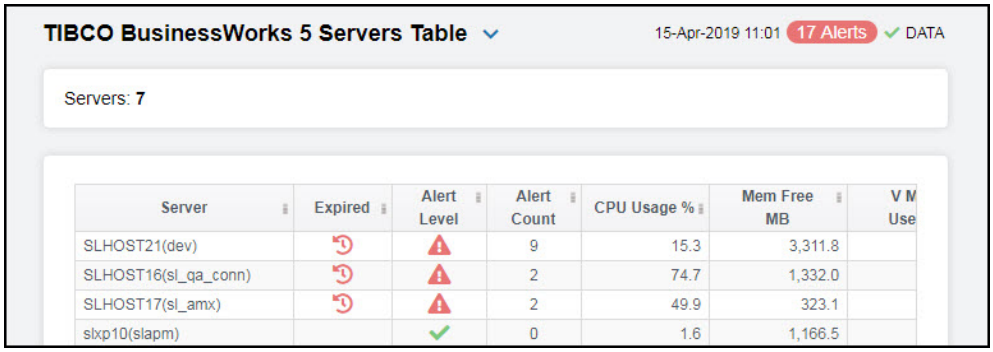


Table rows also sometimes use color to indicate the current most critical alert state for all resources associated with a given row. For example, the color coding is typically as follows:

Red indicates that one or more resources associated with that application currently has an alert in an alarm state.

Yellow indicates that one or more resources associated with that application currently have an alert in a warning state.

Green indicates that no resources associated with that application currently have an alert in a warning or alarm state.

Tables support several interactive features: filtering on multiple columns, sorting, column reordering, and hiding columns. Many of these features are accessed from the column menu, shown in the screen shot above, which you open by clicking on the menu icon in a column's header.

Additional features are:

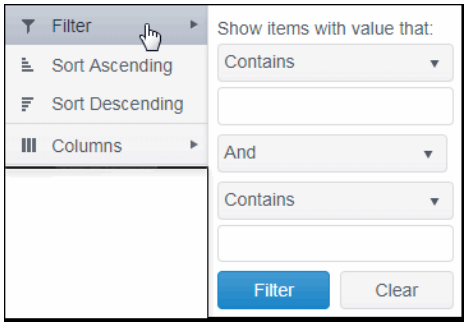
- Column Filtering
- Multiple Column Sorting
- Column Visibility
- Column Reordering
- Row Paging

Column Filtering

You can create a filter on any column. If filters are created on multiple columns, then only the rows that pass all of the filters are displayed. That is, if there are multiple filters they are logically "ANDed" together to produce the final result.

The background of a column's menu icon changes to white to indicate that a filter is defined on that column. This is intended to remind you which columns are filtered.

You can configure a filter on any column by clicking on the column's menu icon and choosing **Filter** from the menu. This opens the **Column Filter** dialog:



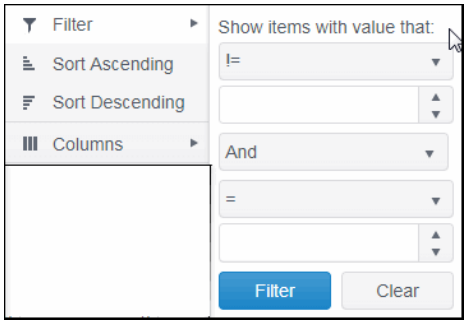
Options in the **Column Filter** dialog vary according to the data type of the selected column:

- **String columns**: You can enter a filter string such as "abc" and, from the dropdown list, select the operator (equal to, not equal to, starts with, contains, etc) to be used when comparing the filter string to each string in the column. All of the filter comparisons on strings are case-insensitive. You can optionally enter a second filter string (e.g. "xyz") and specify if an AND or OR combination should be used to combine the first and second filter results on the column.
- **Numeric columns**: You can enter numeric filter values and select arithmetic comparison operators, (=, !=, >, >=, <, <=). You can optionally enter a second filter

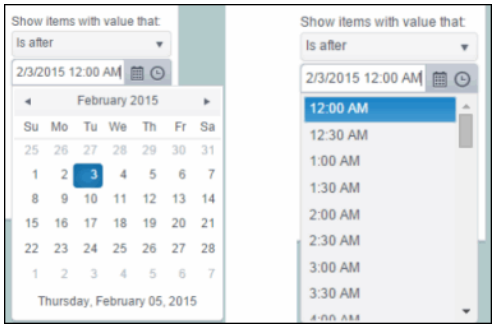
value and comparison operator, and specify if an AND or OR combination should be used to combine the first and second filter results.

- **Boolean columns**: You simply select whether matching items should be true or false.

The numeric and boolean filter dialogs are shown below.



- **Date columns**: You can select a date and time and choose whether matching items should have a timestamp that is the same as, before, or after the filter time. The date is selected by clicking on the calendar icon and picking a date from a calendar dialog. The time is selected by clicking on the time icon and picking a time from a dropdown list:



Alternatively, a date and time can be typed into the edit box. The strings shown in a date column are formatted by the Display Server using its time zone. But if a filter is specified on a date column, the date and time for the filter are computed using the client system's time zone. This can be confusing if the Display Server and client are in different time zones.

Data updates to the grid are suspended while the filter menu is opened. The updates are applied when the menu is closed.

Column filtering is reflected in an export to HTML and Excel.

Multiple Column Sorting

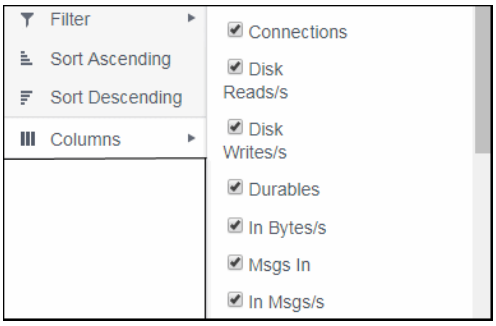
Click on a column header and select **Sort Ascending** or **Sort Descending** to sort the table by that column. To sort multiple columns, click on the column header for each column you want to sort. The sorting is performed in the order that the column headers were clicked. Multiple column sorting is a very useful feature, but can also cause confusion if you intend to sort on a single column, but forget to "unsort" any previously selected sort columns first. You should check for the up/down sort icon in other column headers if a sort gives unexpected results.

The grid's row selection is cleared if the sort is changed or if columns are resized or reordered.

Column sorting is reflected in an export to HTML and Excel.

Column Visibility

You can hide or show columns in the table by clicking on any column's menu icon, and choosing **Columns** from the menu. This opens a submenu with a check box for each column that toggles the visibility of the column. All columns in the data table appear in the Columns menu, even those that are initially hidden.



The leftmost column (the row header column) cannot be hidden.

Column visibility changes are NOT reflected in an export to HTML and Excel.

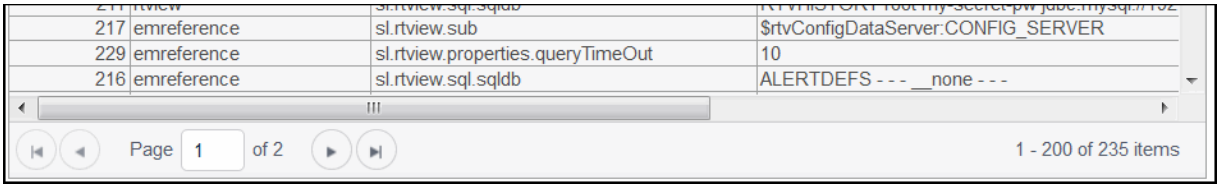
Column Reordering

You can reorder the grid columns by dragging and dropping a column's header into another position.

Column reordering is NOT reflected in an export to HTML and Excel.

Row Paging

If the data table contains more than one 200 rows, page controls appear at the bottom of the grid.



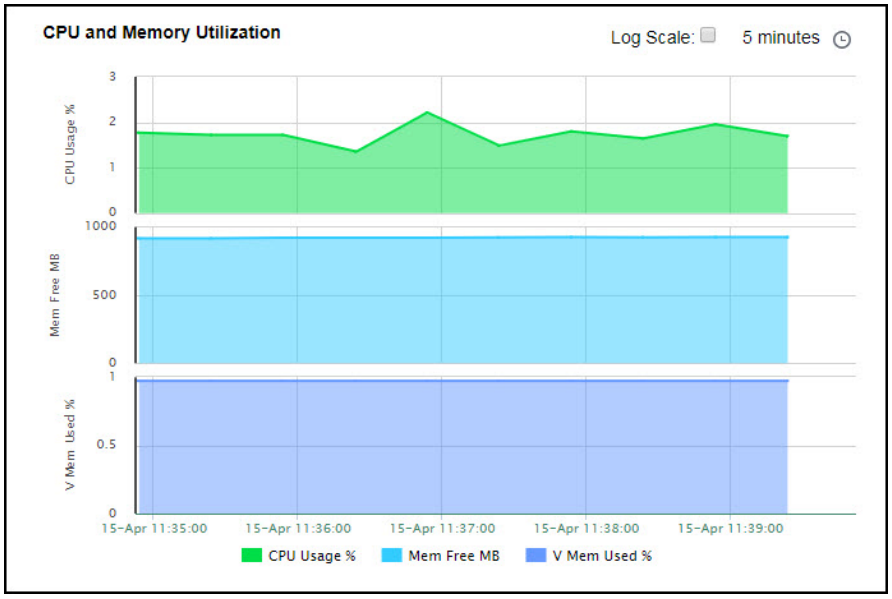
Expired Rows

When a data row has a checked symbol in the **Expired** column, performance data has not been received within the time specified (in seconds) in the **Expire Time** field in the **Duration** region in the RTView Configuration Application > (Project Name) > **Solution Package Configuration** > **TIBCO BusinessWorks** > **DATA STORAGE** tab. The **Delete Time** field (also in the **Duration** region) allows you to define the amount of time (in seconds) in which the row will be removed from the table if there is no response.

Trend Graphs

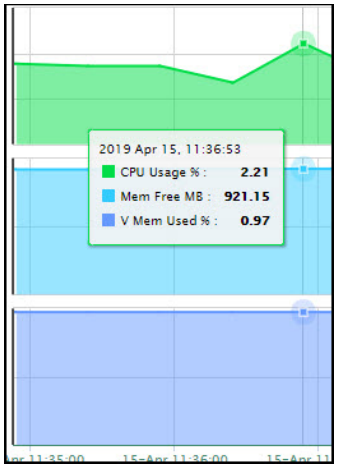
Monitor trend graphs enable you to view and compare performance metrics over time. You can use trend graphs to assess utilization and performance trends.

For example, the following figure illustrates a typical Monitor trend graph.



Mouse-over

The mouse-over functionality provides additional detailed data in an over imposed pop-up window when you mouse-over trend graphs. The following figure illustrates mouse-over functionality. In this example, when you mouse-over a single dot, or data point, a pop-up window shows data for that data point.



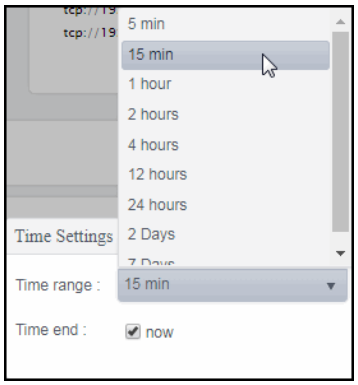
Log Scale

Typically, trend graphs provide the Log Scale option. Log Scale enables you to see usage correlations for data with a wide range of values. For example, if a minority of your data is on

a scale of tens, and a majority of your data is on a scale of thousands, the minority of your data is typically not visible in non-log scale graphs. Log Scale makes data on both scales visible by applying logarithmic values rather than actual values to the data.

Time Settings

Select a time range from the drop down menu varying from **5 Minutes** to **Last 7 Days**. By default, the time range end point is the current time.

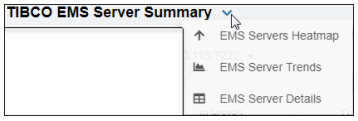
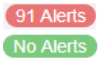


To change the time range, deselect the **now** toggle, which displays some additional date fields. You can click the left and right arrow buttons to decrease the end time by one time period (the time selected in the **Time range** drop down) per click, or you can choose the date and time from the associated calendar and clock icons. You can also enter the date and time in the text field using the following format: **MMM dd, YYYY HH:MM:ss**. For example, Aug 21, 2018 12:24 PM. Click the **now** toggle to reset the time range end point to the current time.

Title Bar Functionality

Displays share the same top layer in the title bar, as shown and described below.

The following table describes the functionality in the display title bar.

	Displays the name of the data server for which you are viewing data.
	Navigates to a display that is most commonly accessed from the current display. The target display differs among displays.
	Displays the current number of alerts. Clicking on this "button" opens the Alerts Table display in a new window.

The current date and time. If the time is incorrect, this might indicate that RTView stopped running. When the date and time is correct and the **Data OK** indicator is green, this is a strong indication that the platform is receiving current and valid data.

The data connection state. Red indicates the data source is disconnected (for example, if the Data Server is not receiving data, or if the Display Server does not receive data from the Data Server, this will be red). Green indicates the data source is connected. When the date and time is correct and the **Data OK** indicator is green, this is a strong indication that the platform is receiving current and valid data.

Export Report

You can quickly export reports for tables in a display by right-clicking on the display and selecting **Export to Excel > Current Page or All Pages**.

TIBCO BusinessWorks Monitor HTML Views/Displays

The following views are available:

- TIBCO BusinessWorks Overview Display: Describes the TIBCO BusinessWorks Overview display.
- BW Applications - HTML: The displays in this View present BusinessWorks application performance metrics.
- "BW Containers - HTML": The displays in this View present BusinessWorks container performance metrics.
- "BW Application Nodes - HTML": The displays in this View present BusinessWorks AppNode performance metrics.
- "BW Application Slices - HTML": The displays in this View present BusinessWorks AppSlice performance metrics.
- "BW Processes - HTML": The displays in this View present BusinessWorks process performance metrics.
- "BW Activities - HTML": The displays in this View present BusinessWorks activity performance metrics.

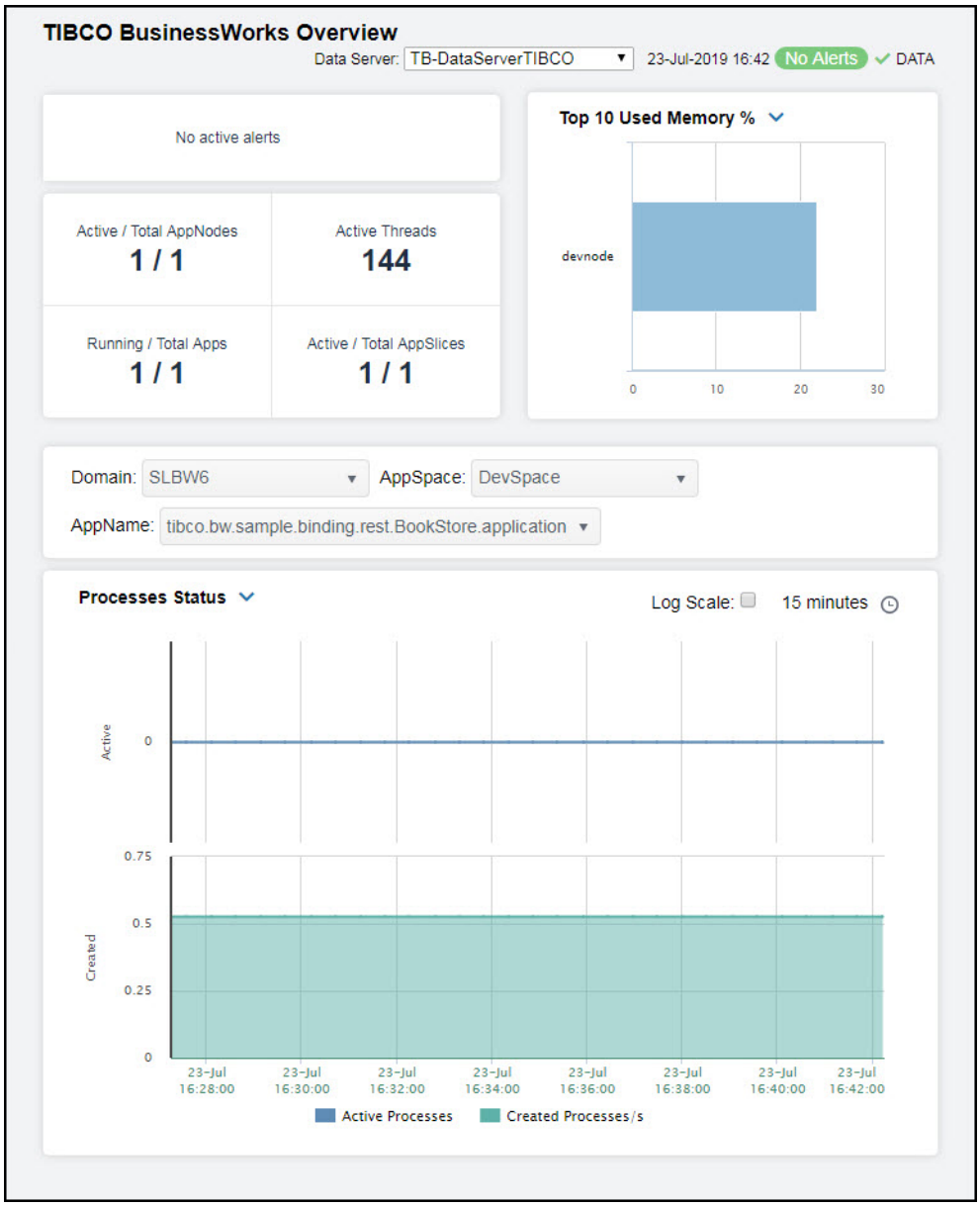
TIBCO BusinessWorks Overview Display

The **TIBCO BusinessWorks Overview** is the top-level display for the TIBCO Enterprise BusinessWorks Monitor, which provides a good starting point for immediately getting the status of all your AppNodes, AppSlices, threads, and processes on your Data Server. You can select the RTView DataServer for which you want to see data and easily view the current data for that DataServer including:

- The total number of active alerts for the selected DataServer, including the total number of critical and warning alerts.
- The number of active AppNodes and the total number of AppNodes.
- The number of active threads on your connected DataServer.
- The number of running and total applications on your connected DataServer.
- The number of active and total AppSlices on your connected DataServer.
- A visual list of the top 10 servers containing the highest used CPUpercentage/used memory percentage/on your connected DataServer.

You can hover over each region in the upper half of the Overview to see more detail. You can also drill down to see even more detail by clicking on each respective region in the Overview. For example, clicking on the alerts in the **CRITICAL** and **WARNING** alerts region opens the **Alerts Table by Components** display.

The bottom half of the display provides a processes status and processes performance trend graph for a selected server. You can hover over the trend graph to see the values at a particular time. You can specify the time range for the trend graph and view data based on a log scale, which enables visualization on a logarithmic scale and should be used when the range in your data is very broad.



BW Applications - HTML

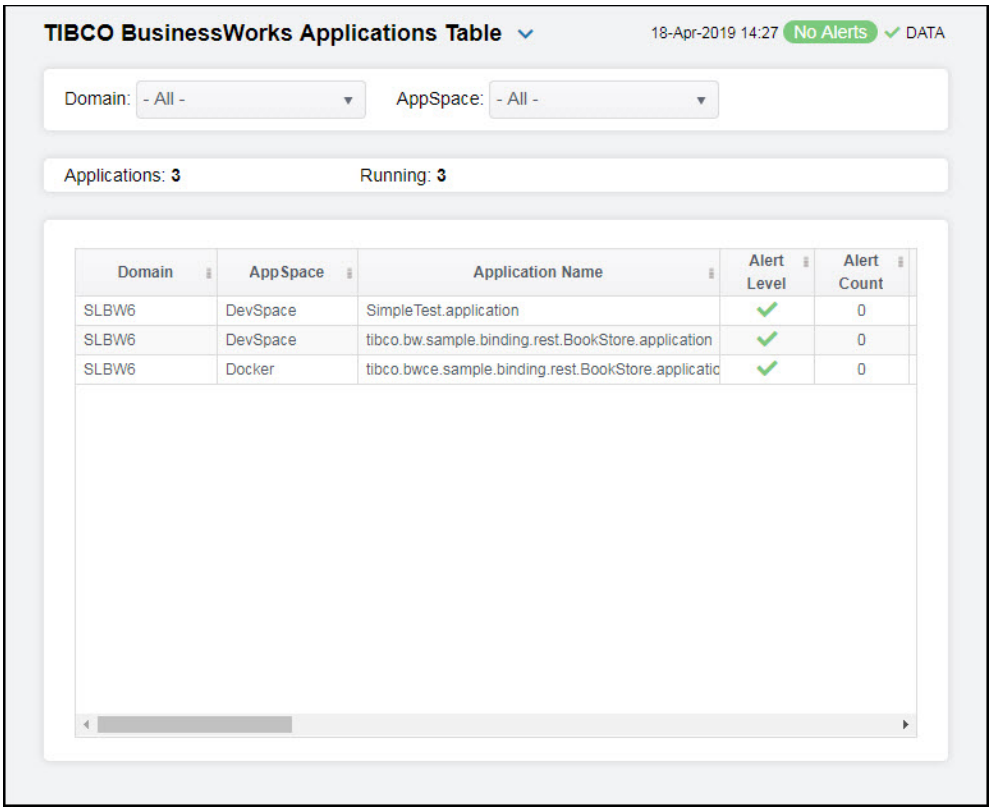
These displays present process performance data for your BusinessWorks applications and AppSpaces across BusinessWorks Domains. Process metrics are totaled by application. Use these displays to monitor critical alerts for all your BusinessWorks applications, and investigate those alerts in lower-level displays. Clicking **BW Applications** from the left/navigation menu opens the TIBCO BusinessWorks Applications Table - HTML display, which shows all available utilization metrics for all BW applications. The options available under **BW Applications** are:

- **BW Applications Heatmap**: Opens the TIBCO BusinessWorks Applications Heatmap - HTML, which shows server and alert status for all BW5 applications.

- **BW Application**: Opens the TIBCO BusinessWorks Application Summary - HTML display, which shows information for a single application.

TIBCO BusinessWorks Applications Table - HTML

Investigate detailed utilization metrics for all BW applications. The **TIBCO BusinessWorks Applications Table** contains all metrics available for applications, including the number of active, failed, suspended, and created applications. Each row in the table contains data for a particular application. Choose a **Domain** and **AppSpace** from the drop-down menus to display activities for the selected Domain/AppSpace combination, or choose **All** from the drop downs to view all applications. Click a column header to sort column data in ascending or descending order. Double-click on a table row to drill-down to the TIBCO BusinessWorks Application Summary - HTML display and view metrics for that particular application. Toggle between the commonly accessed **Table** and **Heatmap** displays by clicking the drop down list on the display title.



Filter By:

The display might include these filtering options:

Domain: Select the domain for which you want to view data in the display.

AppSpace Select the AppSpace for which you want to view data in the display.

Fields and Data:

Applications:	The total number of applications in the AppSpace.
Running	The total number of applications currently running in the AppSpace.

Table:
Each row in the table is a different application.

Domain	The domain in which the application resides.
AppSpace	The AppSpace in which the application resides.
Name	The name of the application.
Alert Level	The most critical alert state for alerts in the row: Red indicates that one or more metrics exceeded their ALARM LEVEL threshold. Yellow indicates that one or more metrics exceeded their WARNING LEVEL threshold. Green indicates that no metrics have exceeded their alert thresholds.
Alert Count	The total number of active alerts for the application.
State	The current status of the application. Valid values are **Running** and **Stopped**.
Deployment	The type of deployment.
AppNodes	The total number of AppNodes associated with the application.
Active Processes	The number of currently active application processes.
Active/s	The rate of processes becoming active per second.
Created Processes	The number of application processes that have been created.
Created/s	The number of application processes created per second.
Completed Processes	The number of completed application processes.
Completed/s	The rate of processes being completed.
Most Recent Exec Time ms	The number of milliseconds for the most recently executed process.
Rate Exec Time ms/s	The number of processes executed, in milliseconds per second.
Suspended Processes	The number of suspended application processes.
Failed Processes	The number of failed application processes.
Version	The application version.
Module	The application module.
Shared Module	The shared module, if any.
Expired	When checked, performance data has not been received within the time specified (in seconds) in the **Expire Time** field in the **Duration** region in the RTView Configuration Application > (**Project Name**) > **Solution Package Configuration** > **TIBCO BusinessWorks** > **DATA STORAGE** tab. The **Delete Time** field (also in the **Duration** region) allows you to define the amount of time (in seconds) in which the row will be removed from the table if there is no response.
Time Stamp	The date and time the row data was last updated.

TIBCO BusinessWorks Applications Heatmap - HTML

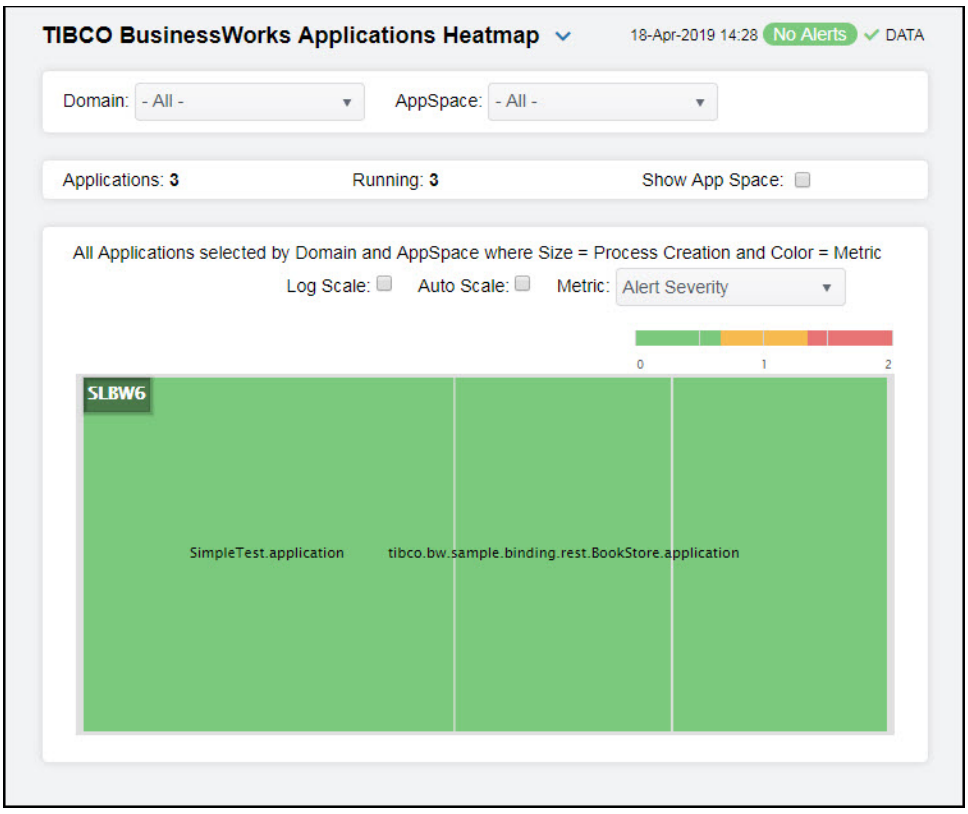
Clicking **BW Applications Heatmap** in the left/navigation menu opens the **TIBCO BusinessWorks Applications Heatmap**, which allows you to view the most critical BusinessWorks application alert states pertaining to process creation and execution for all

nodes on which the applications are deployed. Use this display to quickly identify applications with critical alerts.

Each rectangle in the heatmap represents an application. The rectangle color indicates the most critical alert state associated with the application. The rectangle size represents process creation across applications; a larger size is a larger value.

Choose a domain and AppSpace from the drop-down menus. Choose a different metric to display from the **Metric** drop-down menu. Use the **Show AppSpace** check-box ☑ to include or exclude labels in the heatmap. Mouse over a rectangle to see additional metrics. By default, this display shows **Alert Severity**.

Drill-down and investigate an application by clicking a rectangle in the heatmap to view details in the TIBCO BusinessWorks Application Summary - HTML display.



Filter By:

The display might include these filtering options:

Domain: Select the domain for which you want to view data in the display.

AppSpace Select the AppSpace for which you want to view data in the display.

Fields and Data:

Applications: The total number of Applications.

Running The total number of Applications currently running.

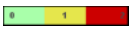
Show AppSpace Displays the name of the associated AppSpace in the heatmap when selected.

Log Scale Select to enable a logarithmic scale. Use **Log Scale** to see usage correlations for data with a wide range of values. For example, if a minority of your data is on a

scale of tens, and a majority of your data is on a scale of thousands, the minority of your data is typically not visible in non-log scale graphs. **Log Scale** makes data on both scales visible by applying logarithmic values rather than actual values to the data.

Auto Scale Select to enable auto-scaling. When auto-scaling is activated, the color gradient bar's maximum range displays the highest value. NOTE: Some metrics auto-scale automatically, even when **Auto** is not selected.

Metric Select the metric driving the heatmap display. The default is **Alert Severity**. Each **Metric** has a color gradient bar that maps values to colors. The heatmap organizes the servers by domain, where each rectangle represents an application. Mouse-over any rectangle to display the current values of the metrics for the application. Click on a rectangle to drill-down to the associated TIBCO BusinessWorks Application Summary - HTML display for a detailed view of metrics for that particular application.

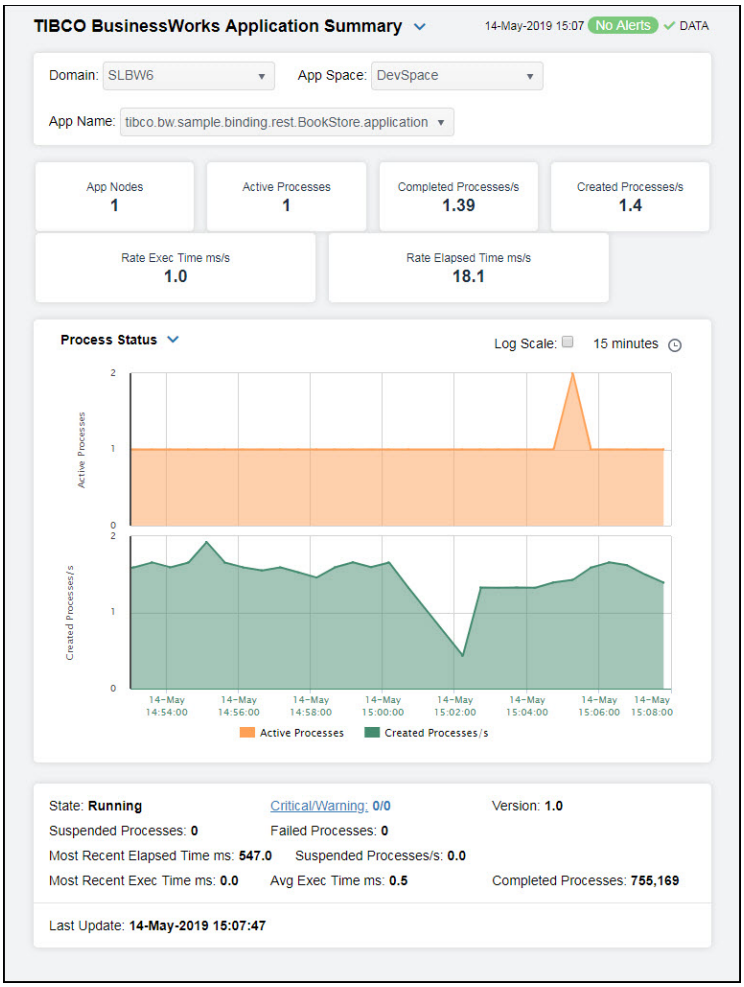
- **Alert Severity** The maximum level of alerts in the heatmap rectangle. Values range from **0** - **2**, as indicated in the color gradient  bar, where **2** is the highest Alert Severity:
  - Red indicates that one or more metrics exceeded their ALARM LEVEL threshold.
  - Yellow indicates that one or more metrics exceeded their WARNING LEVEL threshold.
  - Green indicates that no metrics have exceeded their alert thresholds.
- **Alert Count** The total number of critical and warning alerts in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average alert count.
- **Active Count** The total number of active processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
- **Completed Count** The total number of completed processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
- **Suspended Count** The total number of suspended processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
- **Failed Count** The total number of failed processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
- **Created / sec** The number of processes created per second in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.

Suspended / sec	The number of suspended processes per second in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Failed / sec	The number of failed processes per second in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Exec Time / sec	The process execution time per second in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Most Recent Exec Time	The execution time for the most recently executed process in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Average Elapsed Time	The average elapsed time for all processes in the heatmap rectangle, calculated by dividing the delta elapsed time for the interval by the delta completed, or the number of process instances that completed in the interval. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.

TIBCO BusinessWorks Application Summary - HTML

Clicking **BW Application** in the left/navigation menu opens the **TIBCO BusinessWorks Application Summary** display, which allows you to view current and historical metrics for a single BusinessWorks application across multiple nodes. Use this display to investigate performance issues of application AppNodes within an AppSpace. Use this display to view all available data for each AppNode by Domain and AppSpace.

Clicking on the information boxes at the top of the display takes you to the TIBCO BusinessWorks Application Nodes Table - HTML display or the TIBCO BusinessWorks Processes Table - HTML display, where you can view additional AppNode and Processes data. You can select from two different trend graphs: **Process Status** and **Process Performance**. In the **Process Status** trend graph region, you can view the created processes rate and number of active processes over a selected time range. In the Process Performance trend graph region, you can view the elapsed time rate and execution time rate over a selected time range. Clicking the **Critical/Warning** link at the bottom of the display opens the Alerts Table by Component display.



Filter By:

The display might include these filtering options:

Domain:	Select the domain for which you want to view data in the display.
AppSpace	Choose the AppSpace for which you want to view data in the display.
AppName:	Choose the AppName for which you want to view data in the display.

Fields and Data:

AppNodes	The number of AppNodes running on this application.
Active Processes	The number of active processes for this application.
Completed Processes/s	The rate of completed processes, per second, for this application.
Created Processes/s	The rate of processes being created, per second, on this application.
Rate Exec Time ms/s	The rate at which the application is accumulating process execution time, in milliseconds per second.
Rate Elapsed Time ms/s	The rate at which the application accumulates process elapsed time, in milliseconds per second.
	Process Status
Trend Graphs	Traces the sum of process metrics across all processes in all slices of the selected application. **Active Processes** -- Traces the number of currently active application processes.

Created Processes/s -- Traces the rate of created application processes.

Process Performance

Traces the sum of process metrics across all processes in all slices of the selected application.

Rate Exec Time ms/s -- Traces the rate at which the application is accumulating process execution time, in milliseconds per second.

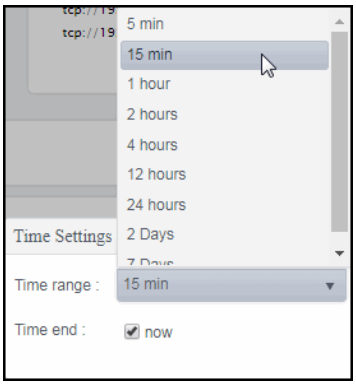
Rate Elapsed Time ms/s -- Traces the rate at which the application accumulates process elapsed time, in milliseconds per second.

Log Scale

Select to enable a logarithmic scale. Use **Log Scale** to see usage correlations for data with a wide range of values. For example, if a minority of your data is on a scale of tens, and a majority of your data is on a scale of thousands, the minority of your data is typically not visible in non-log scale graphs. **Log Scale** makes data on both scales visible by applying logarithmic values rather than actual values to the data.

Time Settings

Select a time range from the drop down menu varying from **5 Minutes** to **Last 7 Days**. By default, the time range end point is the current time.



To change the time range, deselect the **now** toggle, which displays some additional date fields. You can click the left and right arrow buttons to decrease the end time by one time period (the time selected in the **Time range** drop down) per click, or you can choose the date and time from the associated calendar and clock icons. You can also enter the date and time in the text field using the following format: **MMM dd, YYYY HH:MM:ss**. For example, Aug 21, 2018 12:24 PM. Click the **now** toggle to reset the time range end point to the current time.

State	The current status of the application. Valid values are **Running** and **Stopped.**
Suspended Processes	The number of suspended application processes.
Suspended Processes/s	The rate of processes being suspended.

Completed Processes	The number of completed processes.
Critical/ Warning	The number of critical and warning alerts.
Failed Processes	The number of failed processes.
Most Recent Exec Time ms	The number of milliseconds for the most recently executed process.
Version	The application version.
Most Recent Elapsed Time ms	The most recent elapsed time for a process, in milliseconds.
Avg Exec Time ms	The average number of milliseconds for processes to execute for the selected application.
Last Update	The date and time of the last data update.

BW Containers - HTML

These displays present process performance data for your BusinessWorks containers across BusinessWorks Domains. Use these displays to monitor critical alerts for all your BusinessWorks containers, and investigate those alerts in lower-level displays. Clicking **BW Containers** from the left/navigation menu opens the TIBCO BusinessWorks Containers Table - HTML display, which shows a tabular view of all available container performance data. The options available under **BW Containers** are:

- **BW Containers Heatmap**: Opens the TIBCO BusinessWorks Containers Heatmap - HTML, which is a color-coded heatmap view of selected container performance metrics.
- **BW Container Summary**: Opens the TIBCO BusinessWorks Container Summary - HTML display, which shows current and historical metrics for a single container.

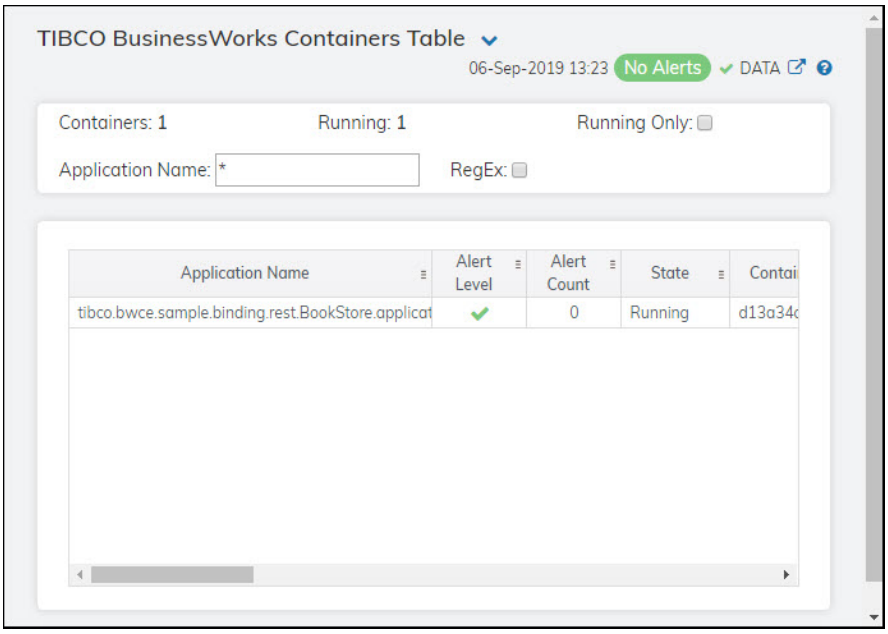
TIBCO BusinessWorks Containers Table - HTML

This display provides a view of the most critical BusinessWorks container alert states pertaining to process creation and execution for all nodes on which the containers are deployed in a tabular format. Use this display to quickly identify containers with critical alerts.

Each row in the table is a container in the selected domain. Check performance and utilization metrics for processes (for example, **Active Processes**, **Active/s**, **Created Processes**, **Completed Processes**, **Suspended Processes**), execution rates (for example, **Rate Exec Time ms/s**), memory and CPU utilization (for example, **Used Memory** and **Used CPU%**) as well as many others.

By default, all containers are listed in the table, but you can enter a string in the **Container Name** filter field to limit the list of containers shown in the display. Click a column header to sort column data in numerical or alphabetical order. Click the **Running Only** check box to only view containers that are up and running in the table.

To view additional details for a specific container, drill-down and investigate by clicking the row in the table for the desired container, which opens the TIBCO BusinessWorks Container Summary - HTML display.



Filter By:

The display might include these filtering options:

Containers:	The total number of containers listed in the table.
Running	The total number of containers that are currently running.
Running Only	Select to show only running containers in the table.
Domain:	Choose a domain to show data for in the display.
Application Name	Enter a string (all or part of a container name) to filter the data shown in the display. If you enter part of an container name, you must enter "*****" before and/or after the string. For example, if you have a container named ContNameOne, you could filter using *Name*, *NameOne, or ContName*.
RegEx	Toggles the **Application Name** filter to accept Regular Expressions for filtering. For example, if your application name is ContNameOne and this option was toggled on, you could enter "Name" (without using "*****"to display the container in the table).

Table:

Each row in the table is a different application.

Application Name	The name of the application.
Alert Level	The most critical alert state for alerts in the row: Red indicates that one or more metrics exceeded their ALARM LEVEL threshold. Yellow indicates that one or more metrics exceeded their WARNING LEVEL threshold. Green indicates that no metrics have exceeded their alert thresholds.
Alert Count	The total number of active alerts in the container.
State	The current status of the application. Valid values are **Running** and **Stopped.**
Container ID	The ID of the container.
Active Processes	The number of currently active processes in the container.
Active/s	The rate of processes becoming active.

Created Processes	The number of processes that have been created since the last data update.
Created/s	The number of processes created per second in the container.
Completed Processes	The number of completed processes in the container.
Completed/s	The rate of processes being completed.
Most Recent Exec Time ms	The number of seconds for the most recently executed process.
Rate Exec Time ms/s	The rate of processes executed in the container, in milliseconds per second.
Suspended Processes	The number of suspended application processes in the container.
Failed Processes	The number of failed processes in the container.
Version	The version of the container.
Module	The name of the container module.
Shared Module	The name of the shared module, if any.
Time Stamp	The date and time the row data was last updated.

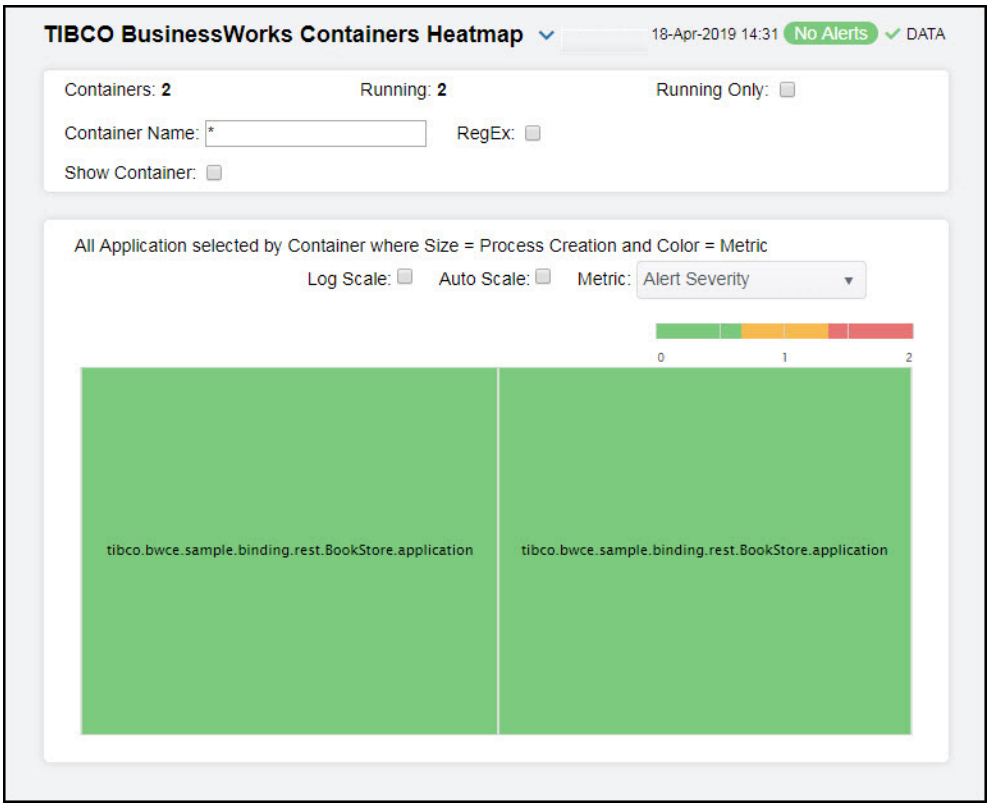
TIBCO BusinessWorks Containers Heatmap - HTML

Clicking **BW Containers Heatmap** in the left/navigation menu opens the **TIBCO BusinessWorks Container Heatmap**, which allows you to view the most critical BusinessWorks container alert states pertaining to process creation and execution for all nodes on which the containers are deployed. Use this display to quickly identify containers with critical alerts.

Each rectangle in the heatmap represents a container. The rectangle color indicates the most critical alert state associated with the container. The rectangle size represents process creation across containers; a larger size is a larger value.

By default, all containers are listed in the heatmap, and is based on the **Alert Severity** metric. You can select a different metric from the **Metric** drop-down menu to view the heatmap based on a different metric. To view data shown for a specific container(s) in the display, enter a string in the **Container Name** filter field. Use the **Container Names** check-box ☑ to include or exclude labels in the heatmap. You can mouse over a rectangle to see additional metrics. Toggle between the commonly accessed **Table** and **Heatmap** displays by clicking the drop down list on the display title.

Drill-down and investigate details for a specific container by clicking a rectangle in the heatmap, which opens the details for the selected container in the TIBCO BusinessWorks Container Summary - HTML display.



Filter By:

The display might include these filtering options:

Containers:	The total number of containers currently shown in the heatmap.
Running	The total number of containers currently running.
Running Only	Select to show only running containers in the heatmap.
Container Name	Enter a string (all or part of a container name) to filter the data shown in the display. If you enter part of a container name, you must enter "*****" before and/or after the string. For example, if you have an container named ContNameOne, you could filter using *Name*, *NameOne, or ContName*. You can also enable the **RegEx** toggle to just enter a portion of the container name.
RegEx	Toggles the **Container Name** filter to accept Regular Expressions for filtering. For example, if your container name is ContNameOne and this option was toggled on, you could enter "Name" (without using "*****"to display the container in the heatmap).
Show Container	Check to include container name labels in the heatmap.

Fields and Data:

Log Scale	Select to enable a logarithmic scale. Use **Log Scale** to see usage correlations for data with a wide range of values. For example, if a minority of your data is on a scale of tens, and a majority of your data is on a scale of thousands, the minority of your data is typically not visible in non-log scale graphs. **Log Scale** makes data on both scales visible by applying logarithmic values rather than actual values to the data.
Auto Scale	Select to enable auto-scaling. When auto-scaling is activated, the color gradient bar's maximum range displays the highest value. NOTE: Some metrics auto-scale automatically, even when **Auto** is not selected.
Metric	Select the metric driving the heatmap display. The default is **Alert Severity**.

Each **Metric** has a color gradient bar that maps values to colors. The heatmap consists of multiple rectangles, where each rectangle represents a container. Mouse-over any rectangle to display the current values of the metrics for the container. Click on a rectangle to drill-down to the associated TIBCO BusinessWorks Container Summary - HTML display for a detailed view of metrics for that particular container.

Alert Severity	The maximum level of alerts in the heatmap rectangle. Values range from **0** - **2**, as indicated in the color gradient  bar, where **2** is the highest Alert Severity: Red indicates that one or more metrics exceeded their ALARM LEVEL threshold. Yellow indicates that one or more metrics exceeded their WARNING LEVEL threshold. Green indicates that no metrics have exceeded their alert thresholds.
Alert Count	The total number of critical and warning alerts in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average alert count.
Active	The total number of active processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Completed	The total number of completed processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Suspended	The total number of suspended processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Failed	The total number of failed processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Created/s	The number of processes created per second in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Suspended/s	The number of suspended processes per second in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Failed/s	The number of failed processes per second in the heatmap rectangle. The color gradient  bar, populated by the

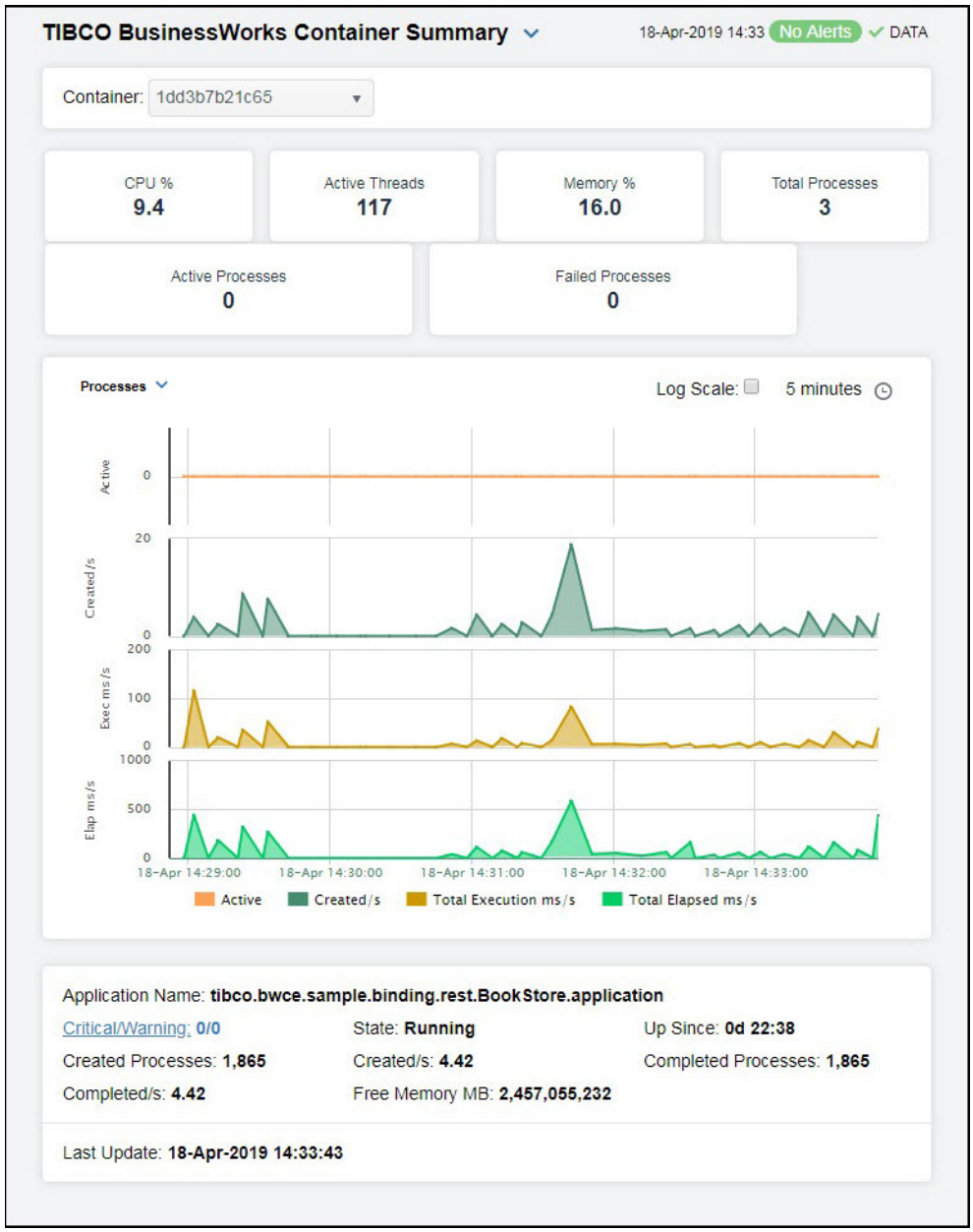
	current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Exec Time/s	The process execution time per second in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Most Recent Exec Time	The execution time for the most recently executed process in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Avg Exec Time	The average execution time for all processes in the heatmap rectangle, calculated by dividing the delta execution time for the interval by the delta completed, or the number of process instances that completed in the interval. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Avg Elapsed Time	The average elapsed time for all processes in the heatmap rectangle, calculated by dividing the delta elapsed time for the interval by the delta completed, or the number of process instances that completed in the interval. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.

TIBCO BusinessWorks Container Summary - HTML

Clicking **BW Container Summary** in the left/navigation menu opens the **TIBCO BusinessWorks Container Summary** display, which allows you to track utilization and performance metrics for specific BW containers. Clicking on the information boxes at the top of the display takes you to the TIBCO BusinessWorks Containers Table - HTML display, where you can view additional container data.

You can toggle between a **Processes** trend graph and a **Resources** trend graph. The **Processes** trend graph shows the number of active containers, the container created rate, the total execution rate, and the total elapsed rate over a selected time range. The **Resources** trend graph shows **CPU%** utilization, current number of **Threads**, **Memory%** utilization and **Bytes** used over a selected time range.

Clicking the **Critical/Warning** link at the bottom of the display opens the **Alerts Table by Component** display.



Filter By:

The display might include these filtering options:

Container Choose the container for which you want to view data in the display.

Fields and Data:

CPU % The percentage of CPU used by the AppNode.

Active Threads The total number of active threads on the AppNode.

Memory % The percentage of memory used by the AppNode.

Total Processes The total number of processes on the container.

Active Processes The total number of active processes on the AppNode.

Failed Processes The total number of failed processes on the AppNode.

Processes Trend Graph Traces the sum of process metrics across all processes in all slices of the selected container.

Active -- Traces the number of currently active application

processes on the container.

Created/s -- Traces the number of created application processes on the container.

Total Execution ms/s -- Traces the rate at which the application is accumulating process execution time, in milliseconds per second, on the container.

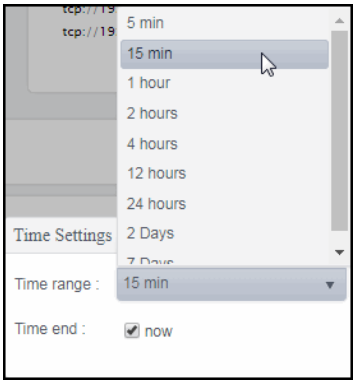
Total Elapsed ms/s -- Traces the rate at which the application accumulates process elapsed time, in milliseconds per second, on the container.

Log Scale

Select to enable a logarithmic scale. Use **Log Scale** to see usage correlations for data with a wide range of values. For example, if a minority of your data is on a scale of tens, and a majority of your data is on a scale of thousands, the minority of your data is typically not visible in non-log scale graphs. **Log Scale** makes data on both scales visible by applying logarithmic values rather than actual values to the data.

Time Settings

Select a time range from the drop down menu varying from **5 Minutes** to **Last 7 Days**. By default, the time range end point is the current time.



To change the time range, deselect the **now** toggle, which displays some additional date fields. You can click the left and right arrow buttons to decrease the end time by one time period (the time selected in the **Time range** drop down) per click, or you can choose the date and time from the associated calendar and clock icons. You can also enter the date and time in the text field using the following format: **MMM dd, YYYY HH:MM:ss**. For example, Aug 21, 2018 12:24 PM. Click the **now** toggle to reset the time range end point to the current time.

Name	The name of the application.
Up Since	The length of time that the application has been up and running.
Completed Processes	The number of completed processes.

Critical/Warning	The number of critical and warning alerts.
Created Processes	The number of created processes.
Completed/s	The rate of completed processes.
State	The current status of the application. Valid values are **Running** and **Stopped.**
Created/s	The rate at which processes are being created.
Free Memory MB	The amount of free memory, in megabytes.
Last Update	The date and time of the last data update.

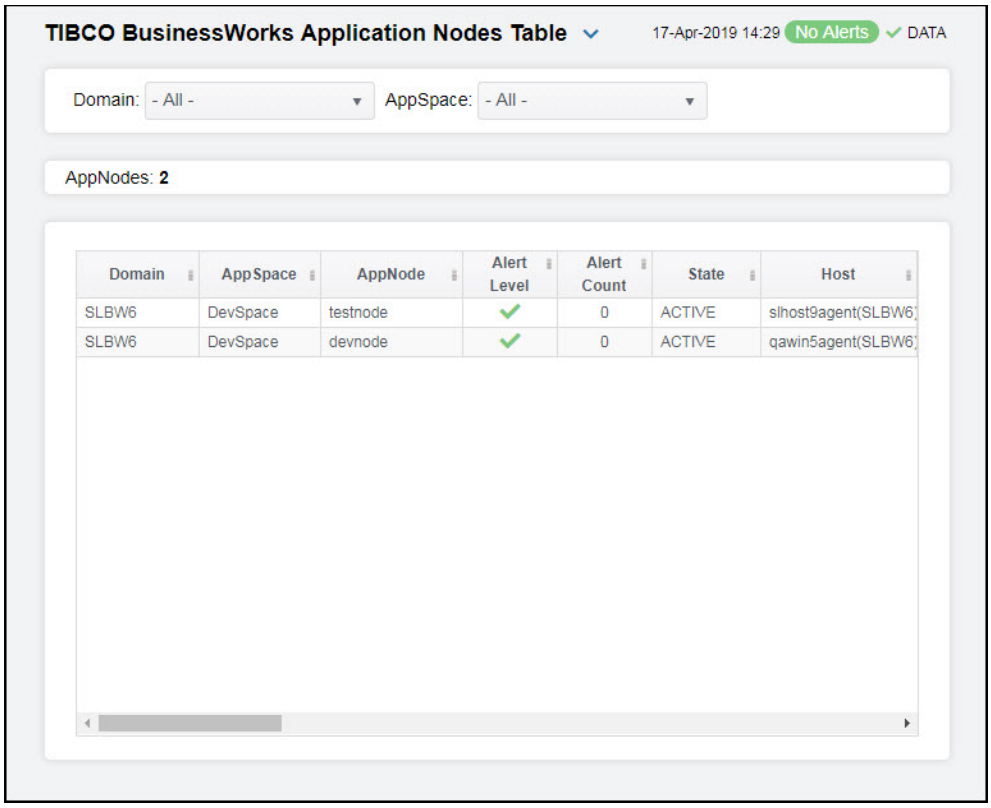
BW Application Nodes - HTML

These displays present internal JVM memory and host CPU utilization for BusinessWorks AppNodes and their resources, which can be useful because the AppNode performance is dependent on both internal and external factors and they sometimes interact. Clicking **BW Application Nodes** from the left/navigation menu opens the TIBCO BusinessWorks Application Nodes Table - HTML display, which shows a tabular view of all available utilization data. The options available under **BW Application Nodes** are:

- **BW Application Nodes Heatmap**: Opens the TIBCO BusinessWorks Application Nodes Heatmap - HTML, which shows a color-coded heatmap view of utilization metrics.
- **BW Application Node**: Opens the TIBCO BusinessWorks Application Node Summary - HTML display, which shows Current and historical metrics for a single AppNode.

TIBCO BusinessWorks Application Nodes Table - HTML

View BusinessWorks data shown in the TIBCO BusinessWorks Application Nodes Heatmap - HTML display, and additional details, in a tabular format. Use this display to view all available data for each AppNode by Domain and AppSpace. Each row in the table is an AppNode. Choose a domain and AppSpace from the drop-down menus. Click a column header to sort column data in numerical or alphabetical order. Drill-down and investigate by clicking a row to view details for the selected AppNode in the TIBCO BusinessWorks Application Node Summary - HTML display.



Filter By:

The display might include these filtering options:

Domain: Choose a domain to show data for in the display.

AppSpace Choose an AppSpace to show data for in the display.

AppNodes: The total number of rows in the table.

Table:

Column values describe the AppNode.

Domain The domain in which the AppNode resides.

AppSpace The AppSpace in which the AppNode resides.

AppNode The name of the AppNode.

Alert Level The most critical alert state for alerts in the row:

- Red indicates that one or more metrics exceeded their ALARM LEVEL threshold.
- Yellow indicates that one or more metrics exceeded their WARNING LEVEL threshold.
- Green indicates that no metrics have exceeded their alert thresholds.

Alert Count The total number of active alerts for the AppNode.

State The current status of the application. Valid values are **Running** and **Stopped.**

Host The host on which the AppNode resides.

Deployment The type of deployment.

Processes The number of processes running.

Active Threads The number of currently active threads.

Total Memory The total amount of used and free memory, in megabytes.

Used Memory The amount of used memory, in megabytes.

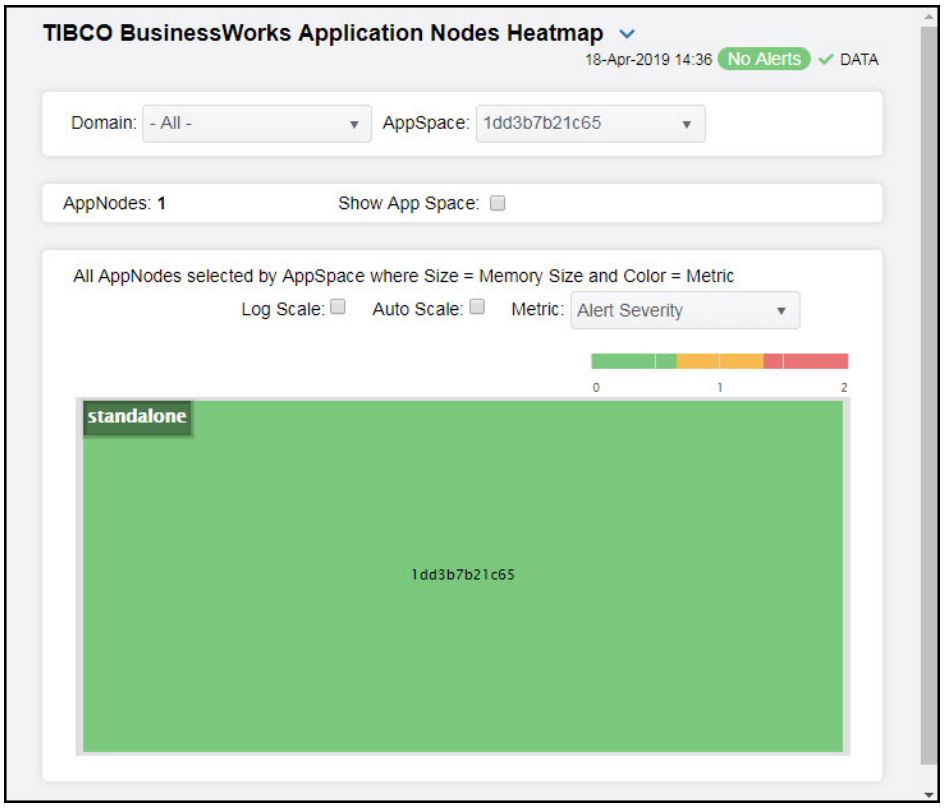
Free Memory	The amount of free memory, in megabytes.
Used Memory %	The percent (%) used memory.
Used CPU %	The percent (%) used CPU.
Active Processes	The number of currently active application processes.
Active/s	The rate of application processes becoming active.
Created Processes	The number of application processes that have been created.
Created /s	The number of application processes created per second.
Completed Processes	The number of completed application processes.
Completed/s	The rate of application processes being completed.
Most Recent Exec Time ms	The number of seconds for the most recently executed process, in milliseconds.
Rate Exec Time ms/s	The number of application processes executed per second.
Suspended Processes	The number of suspended application processes.
Failed Processes	The number of failed application processes.
System Process ID	A unique string identifier for the process.
Up Since	The date and time the AppNode was last started.
Time Stamp	The date and time the row data was last updated.
Source	Name of RTView Data Server sending this data (or localhost).
Expired	When checked (true), performance data has not been received within the time specified (in seconds) in the **Expire Time** field in the **Duration** region in the RTView Configuration Application > (**Project Name**) > **Solution Package Configuration** > **TIBCO BusinessWorks** > **DATA STORAGE** tab. The **Delete Time** field (also in the **Duration** region) allows you to define the amount of time (in seconds) in which the row will be removed from the table if there is no response.

TIBCO BusinessWorks Application Nodes Heatmap - HTML

Clicking **BW Application Nodes Heatmap** in the left/navigation menu opens the **TIBCO BusinessWorks Application Nodes Heatmap**, which allows you to view the most critical JVM memory and host resource utilization for BusinessWorks AppNodes. Use this display to quickly identify AppNodes with critical alerts.

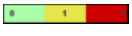
Each rectangle in the heatmap represents an AppNode. The rectangle color indicates the most critical alert state associated with the AppNode. The rectangle size represents the maximum memory used in the rectangle; a larger size is a larger value. Choose a domain and AppSpace from the drop-down menus. Choose a different metric to display from the **Metric** drop-down menu.

Use the **Show AppSpace** check-box ☑ to include or exclude labels in the heatmap. Mouse over a rectangle to see additional metrics. By default, this display shows **Alert Severity**. Drill-down and investigate by clicking a rectangle in the heatmap to view details for the selected application in the TIBCO BusinessWorks Application Node Summary - HTML display. Toggle between the commonly accessed **Table** and **Heatmap** displays by clicking the drop down list on the display title.



Filter By:

The display might include these filtering options:

Domain:	Choose a domain to show data for in the display.
AppSpace	Choose an AppSpace to show data for in the display.
AppNodes:	The total number of AppNodes in the AppSpace.
Show AppSpace	When selected, the names of the AppSpaces associated with the AppNodes display in the heatmap.
Log Scale	Select to enable a logarithmic scale. Use **Log Scale** to see usage correlations for data with a wide range of values. For example, if a minority of your data is on a scale of tens, and a majority of your data is on a scale of thousands, the minority of your data is typically not visible in non-log scale graphs. **Log Scale** makes data on both scales visible by applying logarithmic values rather than actual values to the data.
Auto Scale	Select to enable auto-scaling. When auto-scaling is activated, the color gradient bar's maximum range displays the highest value. NOTE: Some metrics auto-scale automatically, even when **Auto** is not selected.
Metric	Select the metric driving the heatmap display. The default is **Alert Severity**. Each **Metric** has a color gradient bar that maps values to colors. The heatmap organizes the servers by host, where each rectangle represents a server. Mouse-over any rectangle to display the current values of the metrics for the Server. Click on a rectangle to drill-down to the associated TIBCO BusinessWorks Application Node Summary - HTML display for a detailed view of metrics for that particular AppNode.
Alert Severity	The maximum level of alerts in the heatmap rectangle. Values range from **0** - **2**, as indicated in the color gradient  bar, where **2** is the highest Alert Severity: ● Red indicates that one or more metrics exceeded their ALARM LEVEL threshold. ● Yellow indicates that one or more metrics exceeded their WARNING LEVEL threshold.

	Green indicates that no metrics have exceeded their alert thresholds.
Alert Count	The total number of critical and warning alerts in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average alert count.
CPU Used%	The percent (%) CPU used in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum CPU used percentage in the heatmap. The middle value in the gradient bar indicates the average amount.
Memory Used%	The percent (%) memory used in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum memory used percentage in the heatmap. The middle value in the gradient bar indicates the average amount.
Active Processes	The number of currently active processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum number of active processes in the heatmap. The middle value in the gradient bar indicates the average count.
Created Processes	The number of processes created in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum number of created processes in the heatmap. The middle value in the gradient bar indicates the average count.
Created/sec	The number of processes created per second in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum creation rate in the heatmap. The middle value in the gradient bar indicates the average count.
Failed Processes	The number of failed processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum number of failed processes in the heatmap. The middle value in the gradient bar indicates the average number.

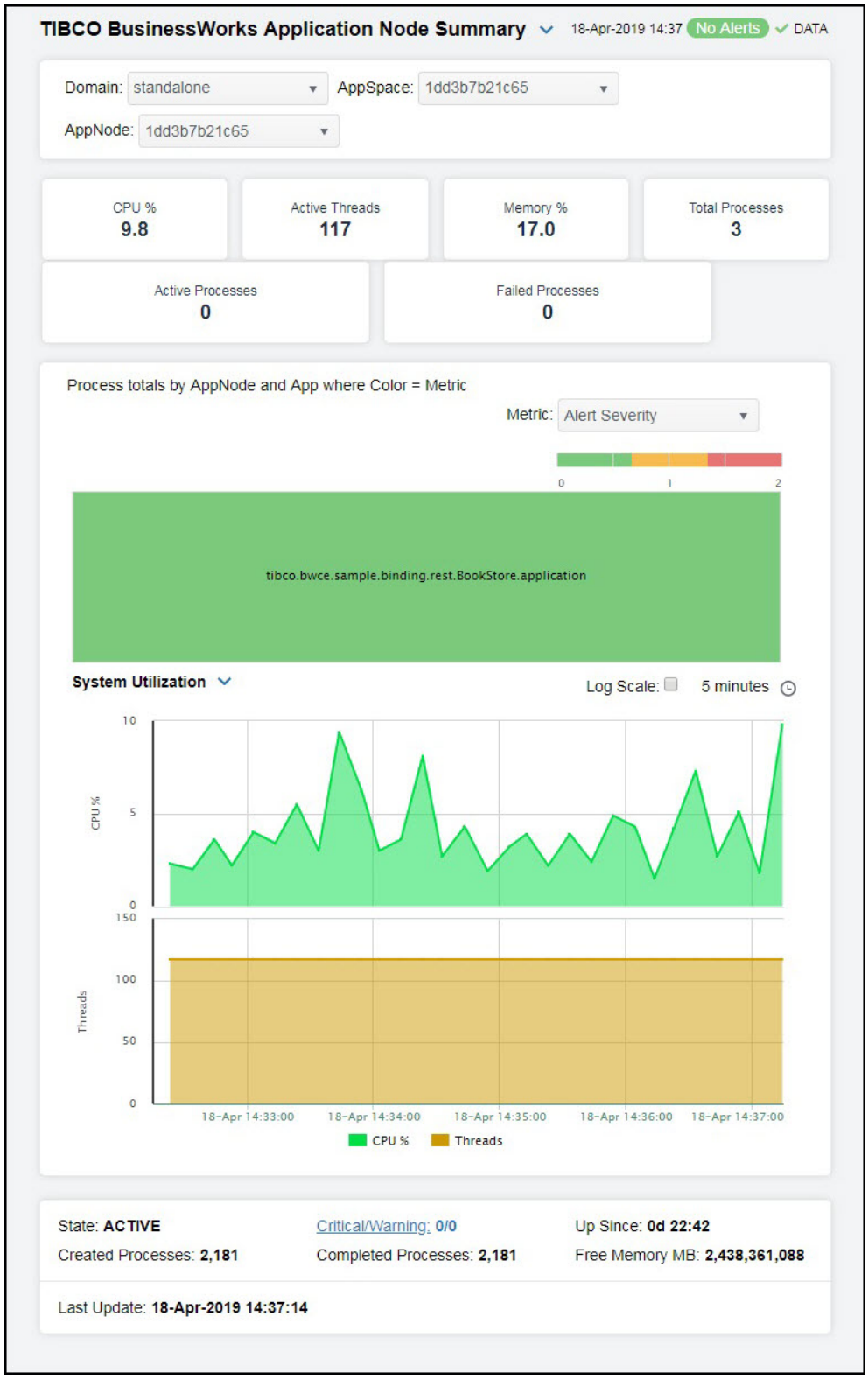
TIBCO BusinessWorks Application Node Summary - HTML

Clicking **BW Application Node** in the left/navigation menu opens the **TIBCO BusinessWorks Application Node Summary** display, which allows you to view current and historical utilization and performance metrics for a single BusinessWorks AppNode. Use this display to investigate performance issues on an AppNode. Clicking on the information boxes at the top of the display takes you to the TIBCO BusinessWorks Application Nodes Table - HTML display, where you can view additional AppNode data.

In the **Processes totals by AppNode and App** heatmap, you can select from the available metrics to view the current status of the processes running on the selected AppNode. Available metrics include **Alert Severity**, **Alert Count**, **Created/s**, and **Average Execution**.

You can select from two different trend graphs: **System Utilization** and **Memory Utilization**. In the **System Utilization** trend graph region, you can view the CPU percentage and number of threads over a selected time range. In the **Memory Utilization** trend graph region, you can view the memory percentage and number of bytes over a selected time range.

Clicking the **Critical/Warning** link at the bottom of the display opens the Alerts Table by Component display.



Filter By:

The display might include these filtering options:

Domain: Choose a domain to show data for in the display.

AppSpace Choose an AppSpace to show data for in the display.

AppNode: Choose an AppNode to show data for in the display.

Fields and Data:

CPU % The percent (%) CPU used on the AppNode.

Active Threads The number of currently active threads for the AppNode.

Memory % The percentage of memory utilization on the AppNode.

Total Processes The total number of processes on the AppNode.

Active Processes The total number of active processes on the AppNode.

Failed Processes The total number of failed processes on the AppNode.

Heatmap Metrics Each rectangle in the heatmap represents an AppNode. Click a rectangle to drill-down and investigate in the TIBCO BusinessWorks Application Summary - HTML display.

Alert Severity The maximum level of alerts in the heatmap rectangle. Values range from **0** - **2**, as indicated in the color gradient bar, where **2** is the highest Alert Severity:

- Red indicates that one or more metrics exceeded their ALARM LEVEL threshold.
- Yellow indicates that one or more metrics exceeded their WARNING LEVEL threshold.
- Green indicates that no metrics have exceeded their alert thresholds.

Alert Count The total number of critical and warning alerts in the heatmap rectangle. The color gradient bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average alert count.

Created/s The number of processes created per second in the heatmap rectangle. The color gradient bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum creation rate in the heatmap. The middle value in the gradient bar indicates the average count.

Average Execution The average number of processes executed in the heatmap rectangle. The color gradient bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum number of processes executed in the heatmap. The middle value in the gradient bar indicates the average count.

Trend Graphs

System Utilization
Traces the sum of process metrics across all processes for all applications on the AppNode.

CPU% -- Traces he percent (%) CPU used on the AppNode.

Threads -- Traces the number of threads.

Memory Utilization
Traces the sum of process metrics across all processes for all applications on the AppNode.

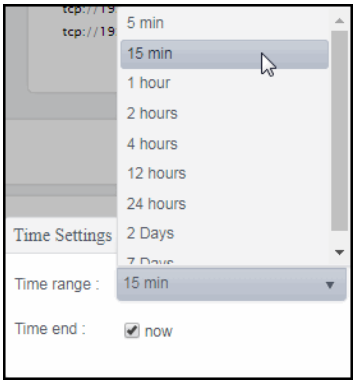
Memory Percentage -- Traces the percentage of memory used.

Bytes -- Traces the total number of bytes.

Log Scale Select to enable a logarithmic scale. Use **Log Scale** to see usage correlations for data with a wide range of values. For example, if a minority of your data is on a scale of tens, and a majority of your data is on a scale of thousands, the minority of your data is typically not visible in non-log scale graphs. **Log Scale** makes data on both scales visible by applying logarithmic values rather than actual values to the data.

Time Settings Select a time range from the drop down menu varying from **5 Minutes** to **Last 7 Days**. By default, the time range end point is the

current time.



To change the time range, deselect the **now** toggle, which displays some additional date fields. You can click the left and right arrow buttons to decrease the end time by one time period (the time selected in the **Time range** drop down) per click, or you can choose the date and time from the associated calendar and clock icons. You can also enter the date and time in the text field using the following format: **MMM dd, YYYY HH:MM:ss**. For example, Aug 21, 2018 12:24 PM. Click the **now** toggle to reset the time range end point to the current time.

State	The current status of the application. Valid values are **Running** and **Stopped.**
Created Processes	The total number of processes that have been created on the AppNode.
Critical/Warning	The number of critical and warning alerts.
Completed Processes	The total number of processes that have been completed on the AppNode.
Up Since	The number of days, hours, minutes, and seconds that the AppNode has been up and running.
Free Memory MB	The amount of available memory on the AppNode, in megabytes.
Last Update	The date and time of the last data update.

BW Application Slices - HTML

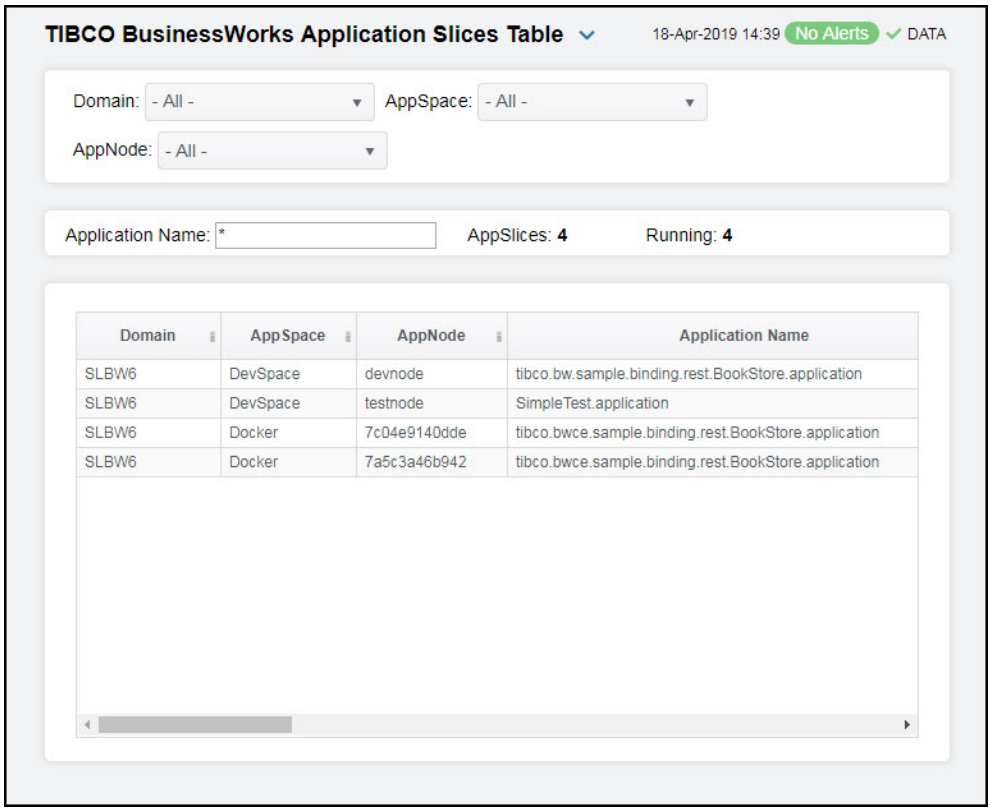
These displays present process metrics totaled by Application and AppNode for AppSlices. This is useful to see how the application is distributed and how each part of it is performing. The AppSlice is the part of an application running on a specific AppNode when the application is deployed to multiple AppNodes. Clicking **BW Application Slices** from the left/navigation menu opens the TIBCO BusinessWorks Application Slices Table - HTML display, where each row in the table displays all available metrics for the AppSlice. The options available under **BW Application Slices** are:

- **BW Application Slices Heatmap**: Opens the TIBCO BusinessWorks Application Slices Heatmap - HTML, which shows process execution metrics for all AppSlices.
- **BW Application Slice**: Opens the TIBCO BusinessWorks Application Slice Summary - HTML display, which shows current and historical metrics for a single AppSlice.

TIBCO BusinessWorks Application Slices Table - HTML

Select a domain, AppSpace, and AppNode from the drop-down menus. Each row in the table is a different AppSpace and contains all metrics available for the AppSpace. You can limit the AppSlices listed in the table by entering a value in the **Application Name** filter field. By default, all AppSlices are listed in the table.

Click a column header to sort column data in ascending or descending order. Double-click on a table row to drill-down to the TIBCO BusinessWorks Application Slice Summary - HTML display and view metrics for that particular AppSlice. Toggle between the commonly accessed **Table** and **Heatmap** displays by clicking the drop down list on the display title.



Filter By:

The display might include these filtering options:

Domain:	Choose a domain to show data for in the display.
AppSpace	Choose an AppSpace to show data for in the display.
AppNode	Choose an AppNode to show data for in the display.
Application Name Filter	Enter a string to limit data shown in the display.

Fields and Data:

AppSlices:	The total number of AppSpaces listed in the table.
Running	The total number of applications currently running in the AppSpace.

Table:
Each row in the table is a different AppNode.

Domain	The domain in which the AppSpace resides.
AppSpace	The AppSpace the AppNode is associated with.
AppNode	The name of the selected AppNode.
Name	The name of the application.
Alert Level	The most critical alert state for alerts in the row: Red indicates that one or more metrics exceeded their ALARM LEVEL threshold. Yellow indicates that one or more metrics exceeded their WARNING LEVEL threshold. Green indicates that no metrics have exceeded their alert thresholds.
Alert Count	The total number of active alerts for the AppNode.
State	The current status of the application. Valid values are **Running** and **Stopped.**
Active Processes	The number of currently active application processes.
Active/s	The rate of application processes becoming active, per second.
Created Processes	The number of application processes that have been created.
Created/s	The number of application processes created per second.
Completed Processes	The number of completed application processes.
Completed/s	The number of application processes completed per second.
Most Recent Exec Time ms	The number of seconds for the most recently executed process.
Rate Exec Time ms/s	The number of processes executed per second.
Suspended Processes	The number of suspended application processes.
Failed Processes	The number of failed application processes.
Expired	When checked (true), performance data has not been received within the time specified (in seconds) in the **Expire Time** field in the **Duration** region in the RTView Configuration Application > (**Project Name**) > **Solution Package Configuration** > **TIBCO BusinessWorks** > **DATA STORAGE** tab. The **Delete Time** field (also in the **Duration** region) allows you to define the amount of time (in seconds) in which the row will be removed from the table if there is no response.
Time Stamp	The date and time the row data was last updated.

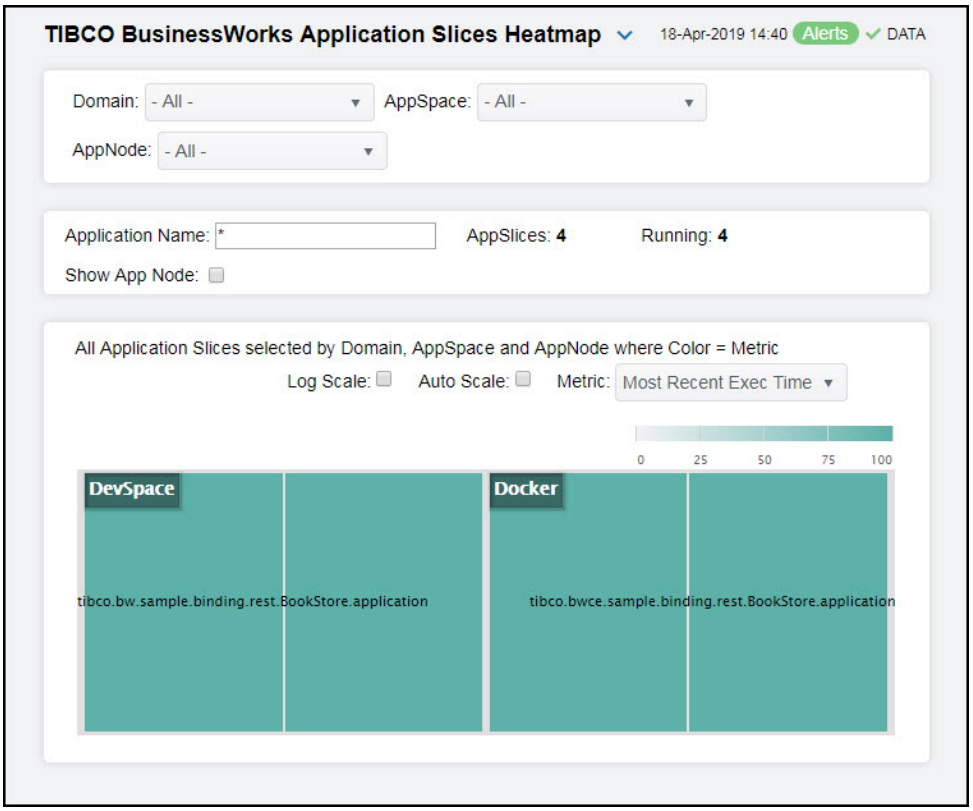
TIBCO BusinessWorks Application Slices Heatmap - HTML

Clicking **BW Application Slices Heatmap** in the left/navigation menu opens the **TIBCO BusinessWorks Application Slices Heatmap**, which allows you to view the most critical performance metrics for BusinessWorks AppSlices. Use this display to quickly identify AppSlices with high process execution numbers.

Each rectangle in the heatmap represents an AppSlice. The rectangle color indicates the process execution numbers for the AppSlice. The rectangle size represents the number of

processes created in the rectangle; a larger size is a larger value. Move your mouse over a node to display current metrics.

Choose a domain, AppSpace and AppNode from the drop-down menus. Enter a string in the **Application Name Filter** field to limit data shown in the display. Click the **Show AppNode** check-box ☑ to include or exclude labels in the heatmap. Mouse over a rectangle to see additional metrics. By default, this display shows data based on the **Active Count** metric. Select a different metric from the Metric drop down menu to display the heatmap based on that metric. Drill-down and investigate by clicking a rectangle in the heatmap to view details for the selected application in the TIBCO BusinessWorks Application Slice Summary - HTML display.



Filter By:

The display might include these filtering options:

Domain:	Choose a domain to show data for in the display.
AppSpace	Choose an AppSpace to show data for in the display.
Application Name Filter	Enter a string to limit data shown in the display.
AppNode:	Choose an AppNode to show data for in the display.

Fields and Data:

AppSlices	The number of AppSlices in the display.
Running	The total number of AppSlices currently running in the display.
Running Only	Select to show only running applications in the display.
Show App	Check to include labels in the heatmap.

Node	
Log Scale	Select to enable a logarithmic scale. Use **Log Scale** to see usage correlations for data with a wide range of values. For example, if a minority of your data is on a scale of tens, and a majority of your data is on a scale of thousands, the minority of your data is typically not visible in non-log scale graphs. **Log Scale** makes data on both scales visible by applying logarithmic values rather than actual values to the data.
Auto Scale	Select to enable auto-scaling. When auto-scaling is activated, the color gradient bar's maximum range displays the highest value. NOTE: Some metrics auto-scale automatically, even when **Auto** is not selected.
Metric	Select the metric driving the heatmap display. The default is **Alert Severity**. Each **Metric** has a color gradient bar that maps values to colors. The heatmap is organized so that each rectangle represents an AppSlice. Mouse-over any rectangle to display the current values of the metrics for the AppSlice. Click on a rectangle to drill-down to the associated TIBCO BusinessWorks Application Slice Summary - HTML display for a detailed view of metrics for that particular AppSlice.
Active Count	The total number of active processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Completed Count	The total number of completed processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum completed processes in an AppSlice in the heatmap. The middle value in the gradient bar indicates the average number of processes.
Suspended Count	The total number of suspended processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum suspended processes in an AppSlice in the heatmap. The middle value in the gradient bar indicates the average number of suspended processes.
Failed Count	The total number of failed processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum failed processes in an AppSlice in the heatmap. The middle value in the gradient bar indicates the average number of failed processes.
Created/s	The number of processes created per second in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum process creation rate in an AppSlice in the heatmap. The middle value in the gradient bar indicates the average creation rate.
Suspended /s	The number of suspended processes per second in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum process suspended rate in an AppSlice in the heatmap. The middle value in the gradient bar indicates the average suspended rate.
Failed/s	The number of failed processes per second in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum process failed rate in an AppSlice in the heatmap. The middle value in the gradient bar indicates the average failed rate.
Exec Time/ s	The process execution time per second in the heatmap rectangle. The color gradient  bar, populated by the current

	heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum execution time rate in an AppSlice in the heatmap. The middle value in the gradient bar indicates the average execution time rate.
Most Recent Exec Time	The execution time for the most recently executed process in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum execution time in an AppSlice in the heatmap. The middle value in the gradient bar indicates the average execution time.
Average Exec Time	The average execution time for all processes in the heatmap rectangle, calculated by dividing the delta execution time for the interval by the delta completed, or the number of process instances that completed in the interval. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum average execution time in the heatmap. The middle value in the gradient bar indicates the average execution time.
Average Elapsed Time	The average elapsed time for all processes in the heatmap rectangle, calculated by dividing the delta elapsed time for the interval by the delta completed, or the number of process instances that completed in the interval. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum average elapsed time in the heatmap. The middle value in the gradient bar indicates the average elapsed time.

TIBCO BusinessWorks Application Slice Summary - HTML

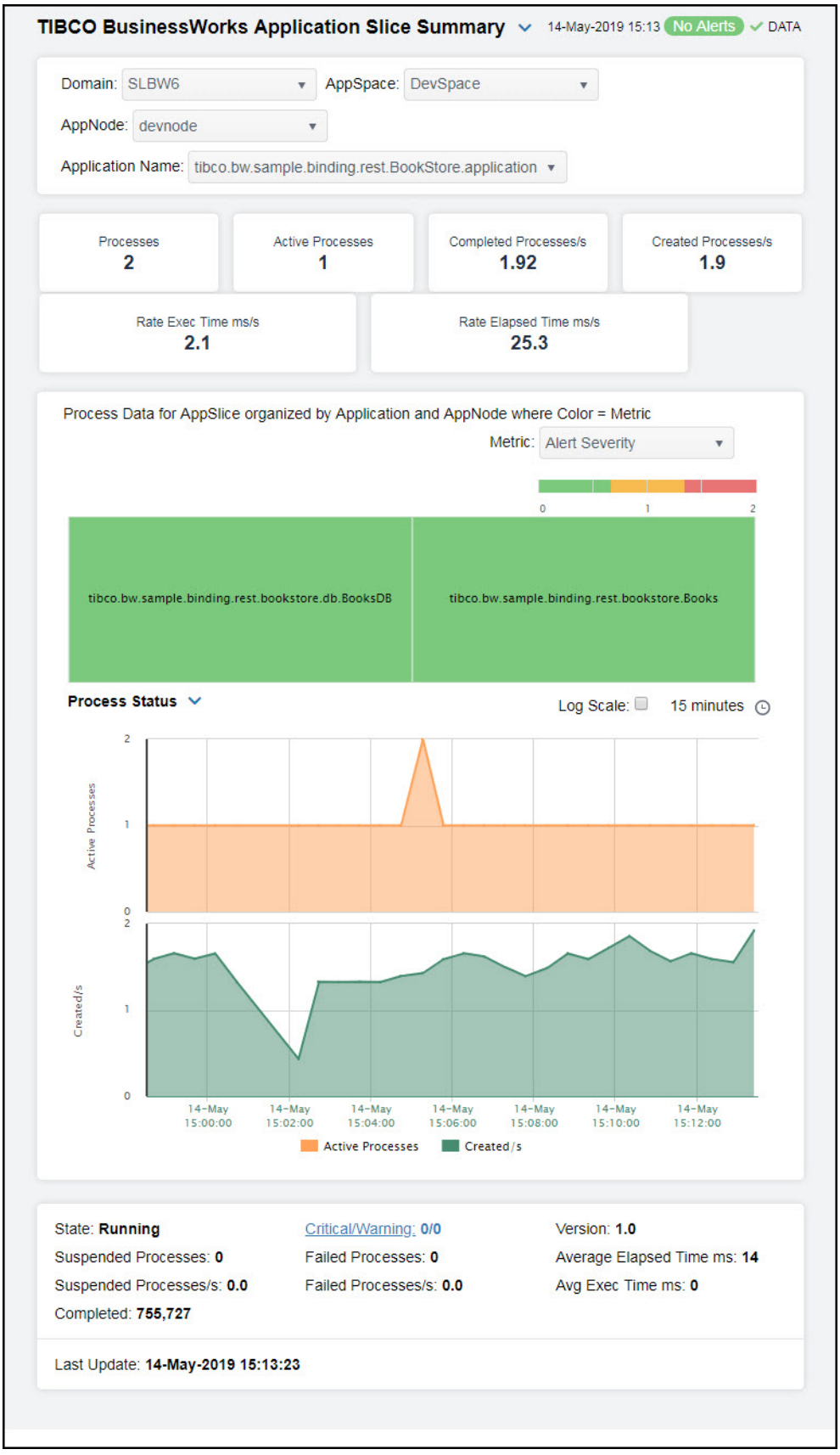
Clicking **BW Application Slice** in the left/navigation menu opens the **TIBCO BusinessWorks Application Slice Summary** display, which allows you to view current and historical utilization and performance metrics for a single BusinessWorks AppSlice. Use this display to investigate performance issues on an AppSlice level. Choose a domain, AppSpace, AppNode, and Application Name from the drop-down menus.

Clicking on the information boxes at the top of the display takes you to the TIBCO BusinessWorks Application Slices Table - HTML display, where you can view additional AppSlice data.

In the **Process Data for AppNode organized by Application** heatmap, you can select from the available metrics to view the current status of the processes running on the selected AppNode. Available metrics include **Alert Severity**, **Alert Count**, **Created/s**, and **Average Execution**. Drill-down and investigate by clicking a AppSlice in the heatmap to view details in the TIBCO BusinessWorks Application Slice Summary - HTML display.

You can select from two different trend graphs: **Process Status** and **Process Performance**. In the **Process Status** trend graph region, you can view the number active processes and number of processes created per second over a selected time range. In the **Process Performance** trend graph region, you can view the process execution rate and the elapsed time rate over a selected time range.

Clicking the **Critical/Warning** link at the bottom of the display opens the Alerts Table by Component display.


TIBCO BusinessWorks Application Slice Summary
14-May-2019 15:13
No Alerts
DATA
Domain: SLBW6
AppSpace: DevSpace
AppNode: devnode
Application Name: tibco.bw.sample.binding.rest.BookStore.application
Processes
2
Active Processes
1
Completed Processes/s
1.92
Created Processes/s
1.9
Rate Exec Time ms/s
2.1
Rate Elapsed Time ms/s
25.3
Process Data for AppSlice organized by Application and AppNode where Color = Metric
Metric: Alert Severity
0
1
2
tibco.bw.sample.binding.rest.bookstore.db.BooksDB
tibco.bw.sample.binding.rest.bookstore.Books
Process Status
Log Scale:
15 minutes
Active Processes
Created/s
14-May 15:00:00
14-May 15:02:00
14-May 15:04:00
14-May 15:06:00
14-May 15:08:00
14-May 15:10:00
14-May 15:12:00
Active Processes
Created/s
State: Running
Suspended Processes: 0
Suspended Processes/s: 0.0
Completed: 755,727
Critical/Warning: 0/0
Failed Processes: 0
Failed Processes/s: 0.0
Version: 1.0
Average Elapsed Time ms: 14
Avg Exec Time ms: 0
Last Update: 14-May-2019 15:13:23

Filter By:

The display might include these filtering options:

Domain:	Select a domain for which you want to view data in the display.
AppSpace	Select an AppSpace for which you want to view data in the display.
AppNode:	Select an AppNode for which you want to view data in the display.
Application Name:	Select an Application Name for which you want to view data in the display.

Fields and Data:

Processes	The total number of processes on the AppSlice.
Active Processes	The total number of active processes on the AppSlice.
Completed Processes/s	The rate of completed processes, per second, on the AppSlice.
Created Processes/s	The rate of created processes, per second, on the AppSlice.
Rate Exec Time ms/s	The time spent on execution of processes, in milliseconds per second, in the AppSlice.
Rate Elapsed Time ms/s	The rate of elapsed time spent for processes, in milliseconds per second.

Heatmap Metrics

Each rectangle in the heatmap represents an AppSlice. Click a rectangle to drill-down and investigate in the TIBCO BusinessWorks Process Summary - HTML display.

Alert Severity	The maximum level of alerts in the heatmap rectangle. Values range from **0** - **2**, as indicated in the color gradient  bar, where **2** is the highest Alert Severity: Red indicates that one or more metrics exceeded their ALARM LEVEL threshold. Yellow indicates that one or more metrics exceeded their WARNING LEVEL threshold. Green indicates that no metrics have exceeded their alert thresholds.
Alert Count	The total number of critical and warning alerts in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average alert count.
Created/s	The number of processes created per second in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum creation rate in the heatmap. The middle value in the gradient bar indicates the average count.
Average Execution	The average number of processes executed in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum number of processes executed in the heatmap. The middle value in the gradient bar indicates the average count.

Trend Graphs

Process Status
Traces the sum of process metrics across all processes for all applications on the AppNode.

Active Processes -- Traces the number of active processes.

Created/s -- Traces the number of processes created per second.

Process Performance
Traces the sum of process metrics across all processes for all applications on the AppNode.

Exec Time/s -- Traces the rate at which the application accumulates process execution time, in milliseconds per second

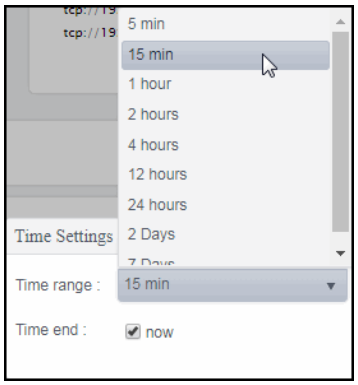
Elapsed Time/s -- Traces the rate at which the application is accumulating process elapsed time, in milliseconds per second.

Log Scale

Select to enable a logarithmic scale. Use **Log Scale** to see usage correlations for data with a wide range of values. For example, if a minority of your data is on a scale of tens, and a majority of your data is on a scale of thousands, the minority of your data is typically not visible in non-log scale graphs. **Log Scale** makes data on both scales visible by applying logarithmic values rather than actual values to the data.

Time Settings

Select a time range from the drop down menu varying from **5 Minutes** to **Last 7 Days**. By default, the time range end point is the current time.



To change the time range, deselect the **now** toggle, which displays some additional date fields. You can click the left and right arrow buttons to decrease the end time by one time period (the time selected in the **Time range** drop down) per click, or you can choose the date and time from the associated calendar and clock icons. You can also enter the date and time in the text field using the following format: **MMM dd, YYYY HH:MM:ss**. For example, Aug 21, 2018 12:24 PM. Click the **now** toggle to reset the time range end point to the current time.

State	The current status of the application. Valid values are **Running** and **Stopped.**
Suspended Processes	The total number of suspended processes.
Suspended Processes/s	The rate of processes being suspended.
Completed Processes	The total number of completed processes summed across all processes in one AppSlice of the application.
Critical/ Warning	The number of critical and warning alerts.
Failed Processes	The total number of failed processes.
Failed Processes/s	The number of failed application processes per second.
Version	The application version.
Average Elapsed Time ms	The average amount of elapsed time for processes, in milliseconds.

Average Exec Time ms	The average number of milliseconds for processes to execute.
Last Update:	The date and time the data was last updated.

BW Processes - HTML

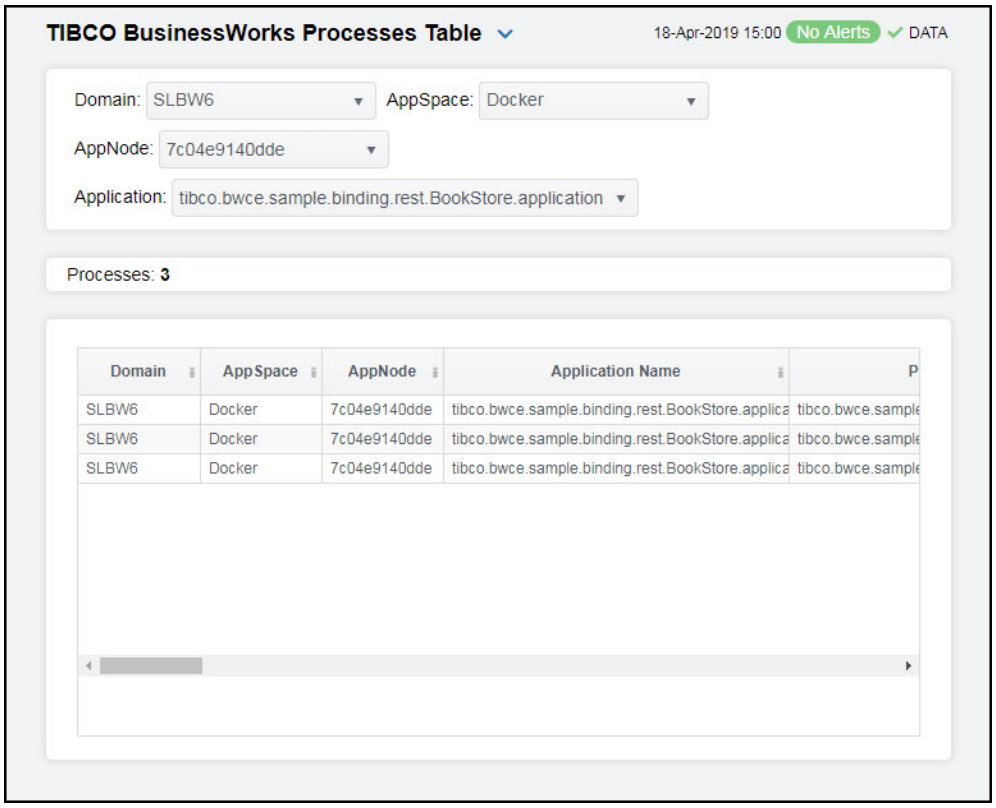
These displays present performance data for BusinessWorks processes. Use these displays to verify that individual BusinessWorks processes are executing and using resources as expected. Clicking **BW Processes** from the left/navigation menu opens the TIBCO BusinessWorks Processes Table - HTML display, where each row in the table displays all available metrics for the process. The options available under **BW Processes** are:

- **BW Processes Heatmap**: Opens the TIBCO BusinessWorks Processes Heatmap - HTML, which shows process execution metrics for all processes.
- **BW Process**: Opens the TIBCO BusinessWorks Process Summary - HTML display, which shows current and historical metrics for a single process.

TIBCO BusinessWorks Processes Table - HTML

Select a domain, AppSpace, AppNode, and Application from the drop-down menus. Each row in the table is a different process and contains all metrics available for the process.

Click a column header to sort column data in ascending or descending order. Double-click on a table row to drill-down to the TIBCO BusinessWorks Process Summary - HTML display and view metrics for that particular process. Toggle between the commonly accessed **Table** and **Heatmap** displays by clicking the drop down list on the display title.



Filter By:

The display might include these filtering options:

Domain:	Select a domain for which you want to view data in the display.
AppSpace	Select an AppSpace for which you want to view data in the display.
AppNode:	Select an AppNode for which you want to view data in the display.
Application	Select an Application for which you want to view data in the display.

Fields and Data:

Processes:	The total number of processes in the AppSpace, which are listed in the table.

Table:

Each row in the table is a different process. Column values are associated with the process.

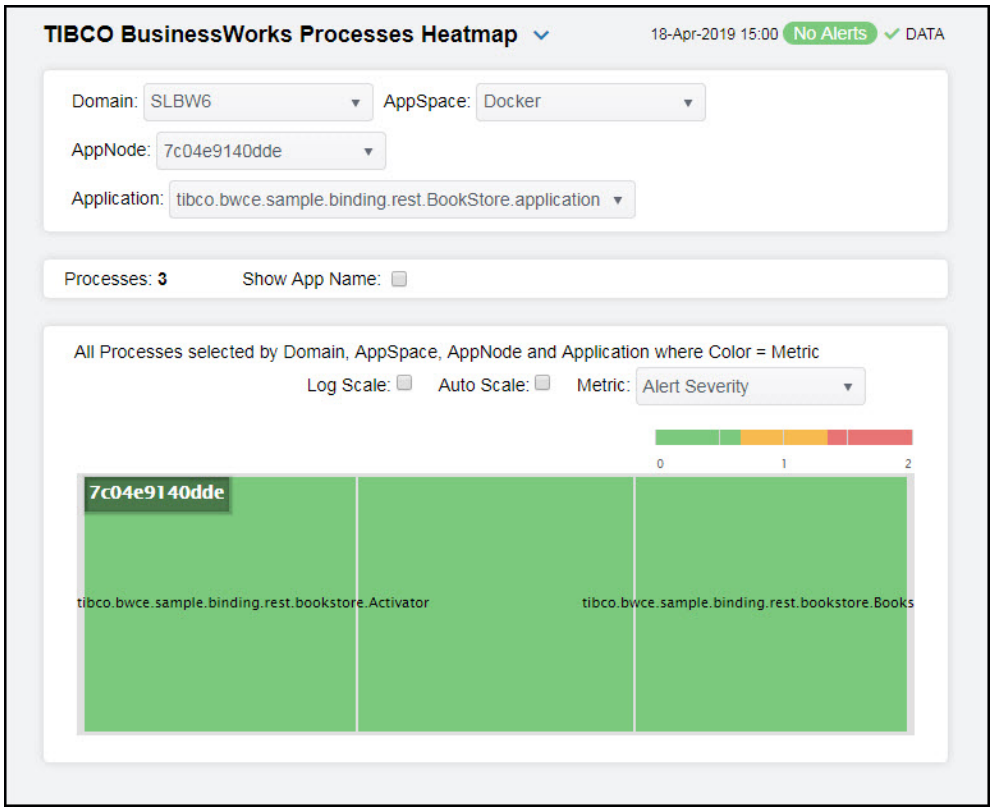
Domain	The domain in which the process resides.
AppSpace	The AppSpace in which the process resides.
AppNode	The AppSpace in which the process resides.
Application Name	The name of the application in which the process is running.
Process Name	The name of the process.
Alert Level	The most critical alert state for alerts in the row: Red indicates that one or more metrics exceeded their ALARM LEVEL threshold. Yellow indicates that one or more metrics exceeded their WARNING LEVEL threshold. Green indicates that no metrics have exceeded their alert thresholds.
Alert Count	The total number of active alerts for the process.
Total Exec Time ms	Total execution time (in milliseconds) for all successfully completed process instances.

Current Total Exec Time	Execution time accumulated during the current polling period.
Rate Exec Time ms/s	Rate execution time in milliseconds per second.
Most Recent Exec Time ms	Execution time (in milliseconds) of the most recently completed process instance.
Total Elapsed Time	Total elapsed time (in milliseconds) for all successfully completed process instances.
Current Total Elapsed Time	Elapsed time accumulated during the current polling period.
Rate Elapsed Time ms/s	Delta elapsed time, in milliseconds per second.
Most Recent Elapsed Time ms	Elapsed clock time (in milliseconds) of the most recently completed process instance.
Active	The number of currently active processes
Created	The number of processes created.
Suspended	The number of process suspensions.
Failed	The number of process failures.
Completed	The number of completed processes.
Current Active	The number of active processes since the last data update.
Active/s	The number of active processes per second.
Current Created	The number of created processes since the last data update.
Created/s	The number of created processes per second.
Current Suspended	The number of suspended processes since the last data update.
Suspended/s	The number of suspended processes per second.
Current Completed	The number of completed processes since the last data update.
Completed/s	The number of completed processes per second.
Current Failed	The number of failed processes since the last data update.
Failed/s	The number of failed processes per second.
Min Exec Time ms	Execution time (in milliseconds) of the process instance that has completed in the shortest amount of execution time.
Max Exec Time ms	Execution time (in milliseconds) of the process instance that has completed in the longest amount of execution time.
Average Exec Time ms	Average execution time (in milliseconds) for all successfully completed process instances.
Min Elapsed Time ms	Elapsed clock time (in milliseconds) of the process instance that has completed in the shortest amount of elapsed time.
Max Elapsed Time ms	Elapsed clock time (in milliseconds) of the process instance that has completed in the longest amount of elapsed time.
Average Elapsed Time ms	Average elapsed clock time (in milliseconds) for all successfully completed process instances.
Count Since Reset	The number of times the process has executed since statistics were reset.
Main Process	The name of the main process.
Version	The version of the application.
Module Name	The name of the application module.
Module Version	The version of the module.
Source	Name of RTView Data Server sending this data (or localhost).
Time Stamp	The date and time the row data was last updated.

TIBCO BusinessWorks Processes Heatmap - HTML

Clicking **BW Processes Heatmap** in the left/navigation menu opens the **TIBCO BusinessWorks Processes Heatmap**, which allows you to view the most critical BusinessWorks alerts pertaining to process creation and execution. Use this display to quickly identify processes with critical alerts.

The heatmap is organized by host with each rectangle representing a process. Move your mouse over a node to display current metrics. Choose a domain, AppSpace, AppNode, and Application from the drop-down menus. Click the **Show App Name** check-box ☑ to include or exclude labels in the heatmap. Mouse over a rectangle to see additional metrics. By default, this display shows data based on the **Active Count** metric. Select a different metric from the **Metric** drop down menu to display the heatmap based on that metric. Drill-down and investigate by clicking a rectangle in the heatmap to view details for the selected application in the TIBCO BusinessWorks Process Summary - HTML display.



Filter By:

The display might include these filtering options:

Domain: Choose a domain to show data for in the display.

AppSpace Choose an AppSpace to show data for in the display.

AppNode: Choose an AppNode to show data for in the display.

Application Choose an AppName to show data for in the display.

Fields and Data:

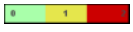
Processes: The total number of processes currently shown in the display.

Show App Name Check to display the application names in the heatmap.

Log Scale Select to enable a logarithmic scale. Use **Log Scale** to see usage correlations for data with a wide range of values. For example, if a minority of your data is on a scale of tens, and a majority of your data is on a scale of thousands, the minority of your data is typically not visible in non-log scale graphs. **Log Scale** makes data on both scales visible by applying logarithmic values rather than actual values to the data.

Auto Scale Select to enable auto-scaling. When auto-scaling is activated, the color gradient bar's maximum range displays the highest value. NOTE: Some metrics auto-scale automatically, even when **Auto Scale** is not selected.

Metric Select the metric driving the heatmap display. The default is Alert Severity. Each Metric has a color gradient bar that maps values to colors. The heatmap is organized so that each rectangle represents a process. Mouse-over any rectangle to display the current values of the metrics for the process. Click on a rectangle to drill-down to the associated TIBCO BusinessWorks Process Summary - HTML display for a detailed view of metrics for that particular process.

Alert Severity The maximum level of alerts in the heatmap rectangle. Values range from **0** - **2**, as indicated in the color gradient  bar, where **2** is the highest Alert Severity:

Red indicates that one or more metrics exceeded their ALARM LEVEL threshold.

Yellow indicates that one or more metrics exceeded their WARNING LEVEL threshold.

Green indicates that no metrics have exceeded their alert thresholds.

Alert Count The total number of critical and warning alerts in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average alert count.

Active Count The total number of active processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of active processes in the heatmap. The middle value in the gradient bar indicates the average count.

Completed Count The total number of completed processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of completed processes in the heatmap. The middle value in the gradient bar indicates the average count.

Suspended Count The total number of suspended processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of suspended processes in the heatmap. The middle value in the gradient bar indicates the average count.

Failed Count The total number of failed processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of failed

	processes in the heatmap. The middle value in the gradient bar indicates the average count.
Created / sec	The number of processes created per second in the heatmap rectangle. The color gradient bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum number of processes created in the heatmap. The middle value in the gradient bar indicates the average number of created processes.
Suspended / sec	The number of suspended processes per second in the heatmap rectangle. The color gradient bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum number of suspended processes per second in the heatmap. The middle value in the gradient bar indicates the average rate of suspended processes.
Failed / sec	The number of failed processes per second in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum rate of failed processes in the heatmap. The middle value in the gradient bar indicates the average count.
Exec Time / sec	The process execution time per second in the heatmap rectangle. The color gradient bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum execution time rate in the heatmap. The middle value in the gradient bar indicates the average execution time rate.
Most Recent Exec Time	The execution time for the most recently executed process in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the most recent execution time in the heatmap. The middle value in the gradient bar indicates the average time.
Average Exec Time	The average execution time for all processes in the heatmap rectangle, calculated by dividing the delta execution time for the interval by the delta completed, or the number of process instances that completed in the interval. The color gradient bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum average execution time in the heatmap. The middle value in the gradient bar indicates the average time.
Most Recent Elapsed Time	The elapsed time for the most recent process in the heatmap rectangle. The color gradient bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum most recent elapsed time in the heatmap. The middle value in the gradient bar indicates the average most recent elapsed time.
Average Elapsed Time	The average elapsed time for all processes in the heatmap rectangle, calculated by dividing the delta elapsed time for the interval by the delta completed, or the number of process instances that completed in the interval. The color gradient bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum average elapsed time in the heatmap. The middle value in the gradient bar indicates the average elapsed time.

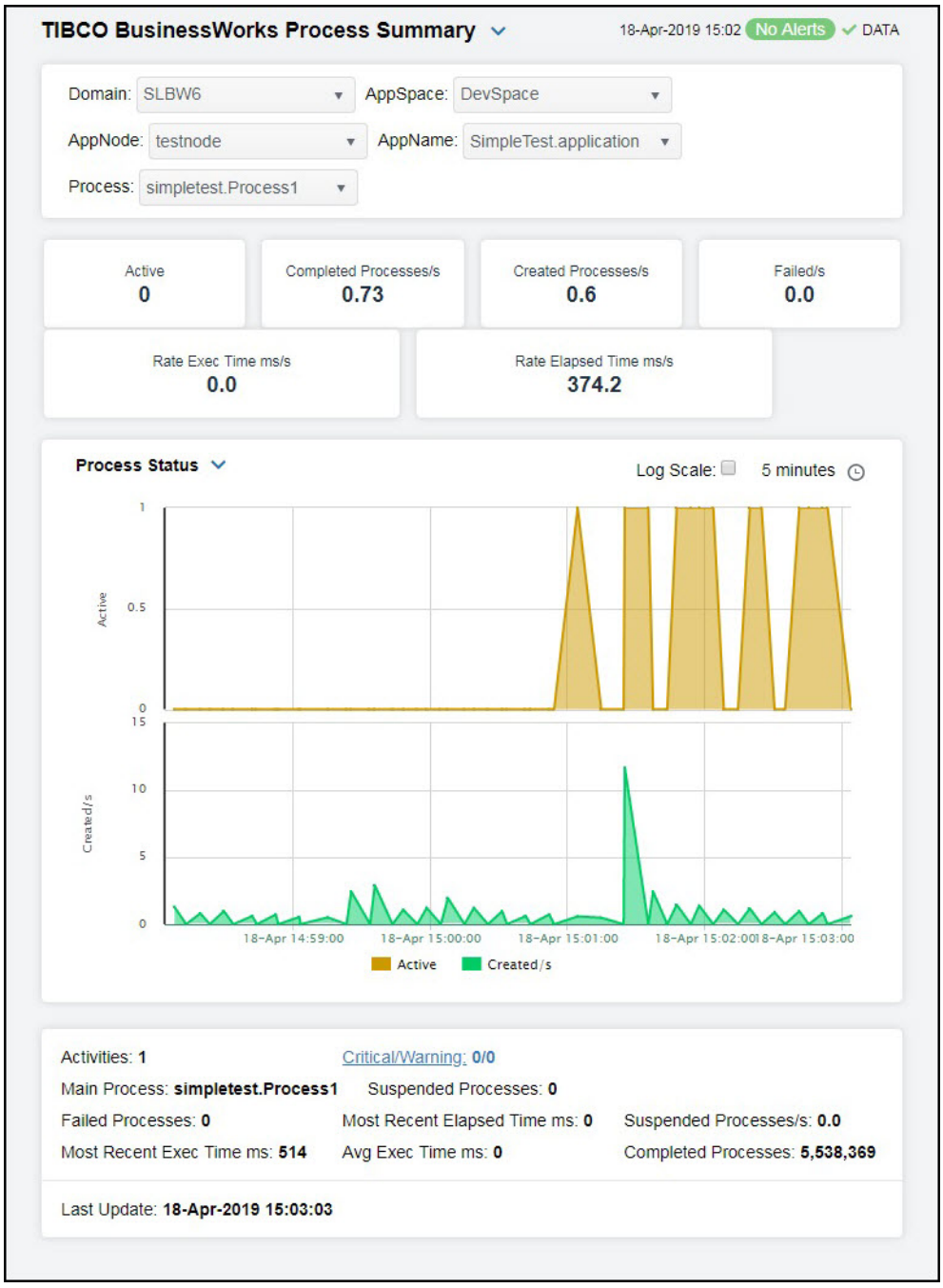
TIBCO BusinessWorks Process Summary - HTML

Clicking **BW Process** in the left/navigation menu opens the **TIBCO BusinessWorks Process Summary** display, which allows you to view current and historical execution metrics for a single BusinessWorks process. Choose a domain, AppSpace, AppNode, Application Name, and Process from the drop-down menus.

Clicking on the information boxes at the top of the display takes you to the TIBCO BusinessWorks Processes Table - HTML display, where you can view additional process data.

You can select from two different trend graphs: **Process Status** and **Process Performance**. In the **Process Status** trend graph region, you can view the number active processes and number of processes created per second over a selected time range. In the **Process Performance** trend graph region, you can view the process execution rate and the elapsed time rate over a selected time range.

Clicking the **Critical/Warning** link at the bottom of the display opens the Alerts Table by Component display.



Filter By:

The display might include these filtering options:

Domain:	Select the domain for which you want to view data in the display.
AppSpace	Select the AppSpace for which you want to view data in the display.
AppNode:	Select the AppNode for which you want to view data in the display.
AppName	Select the application for which you want to view data in the display.
Process	Select the process for which you want to view data in the display.

Fields and Data:

Active	Number of active instances for this process definition. This number is calculated using the Hawk method named GetProcesses. This method returns

information about process instances that are active at the time of update. The value here displays the current total count of all active instances discovered for this process definition. The trend below displays the same value over time.

Completed Processes/s The number of process instances completed per second.

Created Processes/s The number of process instances created per second.

Failed/s The number of errors accumulated per second.

Rate Exec Time ms/s The amount of execution time accumulated in milliseconds per second.

Rate Elapsed Time ms/s The amount of elapsed time accumulated in milliseconds per second.

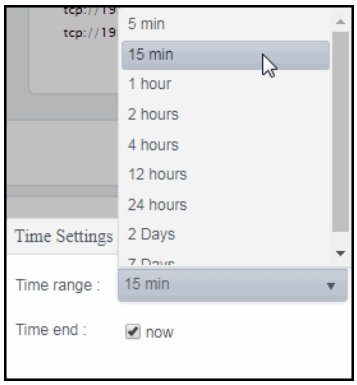
Trend Graphs

Process Status
Traces application process and activity metrics for the selected process.
Active -- Traces the number of currently active processes.
Created/s -- Traces the rate of created processes, per second.

Process Performance
Traces application process and activity metrics for the selected process.
Elapsed Time/ -- Traces the rate at which the application is accumulating process elapsed time, in milliseconds per second.
Exec Time/s -- Traces the rate at which the application is accumulating process execution time, in milliseconds per second.

Log Scale Select to enable a logarithmic scale. Use **Log Scale** to see usage correlations for data with a wide range of values. For example, if a minority of your data is on a scale of tens, and a majority of your data is on a scale of thousands, the minority of your data is typically not visible in non-log scale graphs. **Log Scale** makes data on both scales visible by applying logarithmic values rather than actual values to the data.

Time Settings Select a time range from the drop down menu varying from **5 Minutes** to **Last 7 Days**. By default, the time range end point is the current time.



To change the time range, deselect the **now** toggle, which displays some additional date fields. You can click the left and right arrow buttons to decrease the end time by one time period (the time selected in the **Time range** drop down) per click, or you can choose the date and time from the associated calendar and clock icons. You can also enter the date and time in the text field using the following format: **MMM dd, YYYY HH:MM:ss**. For example, Aug 21, 2018 12:24 PM. Click the **now** toggle to reset the time range end point to

	the current time.
Activities	The number of activities defined for the process.
Main Process	The name of the main process.
Failed Processes	The number of failed process instances.
Most Recent Exec Time ms	The most recent execution time of any process instance, in milliseconds.
Critical/Warning	The number of critical and warning alerts.
Suspended Processes	The number of suspended processes.
Most Recent Elapsed Time ms	The most recent elapsed time of any process instance, in milliseconds.
Avg Exec Time ms	The average execution time for all completed process instances, in milliseconds.
Suspended Processes/s	The rate of processes being suspended, per second.
Completed Processes	The number of completed processes.
Last Update	The date and time of the last data update.

BW Activities - HTML

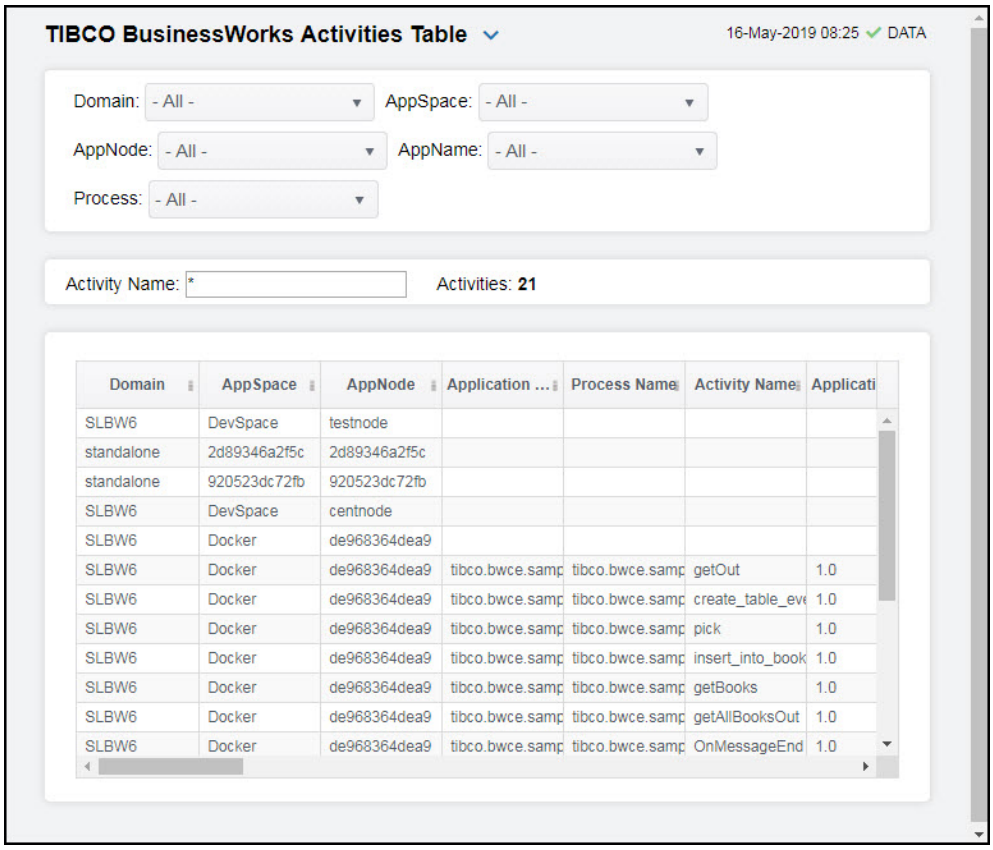
These displays present performance data for BusinessWorks activities. Use these displays to verify that individual BusinessWorks activities are executing and using resources as expected. Clicking **BW Activities** from the left/navigation menu opens the TIBCO BusinessWorks Activities Table - HTML display, where each row in the table displays all available metrics for the process. The options available under **BW Processes** are:

- **BW Activities Heatmap**: Opens the TIBCO BusinessWorks Activities Heatmap - HTML, which shows process execution metrics for all activities.
- **BW Activity**: Opens the TIBCO BusinessWorks Activity Summary - HTML display, which shows current and historical metrics for a single activity.

TIBCO BusinessWorks Activities Table - HTML

Select a domain, AppSpace, AppNode, AppName, and process from the drop-down menus. Each row in the table is a different activity and contains all metrics available for the activity.

Click a column header to sort column data in ascending or descending order. Double-click on a table row to drill-down to the TIBCO BusinessWorks Activity Summary - HTML display and view metrics for that particular activity. Toggle between the commonly accessed **Table** and **Heatmap** displays by clicking the drop down list on the display title.



Filter By:

The display might include these filtering options:

Domain:	Select the domain for which you want to view data in the display.
AppSpace	Select the AppSpace for which you want to view data in the display.
AppNode:	Select the AppNode for which you want to view data in the display.
AppName	Select the application for which you want to view data in the display.
Process	Select the process for which you want to view data in the display.
Activity Name Filter	Enter a string to limit data shown in the display.
Activities	The number of activities displayed in the heatmap as a result of the defined filters.

Table:

Each row in the table is a different process. Column values are associated with the process.

Domain	The domain in which the process resides.
AppSpace	The AppSpace in which the process resides.
AppNode	The AppSpace in which the process resides.
Application Name	The name of the application in which the process is running.
Process Name	The name of the process.
Activity Name	The name of the activity.
Application Version	The version of the application.
Execution Count	The total number of process executions.

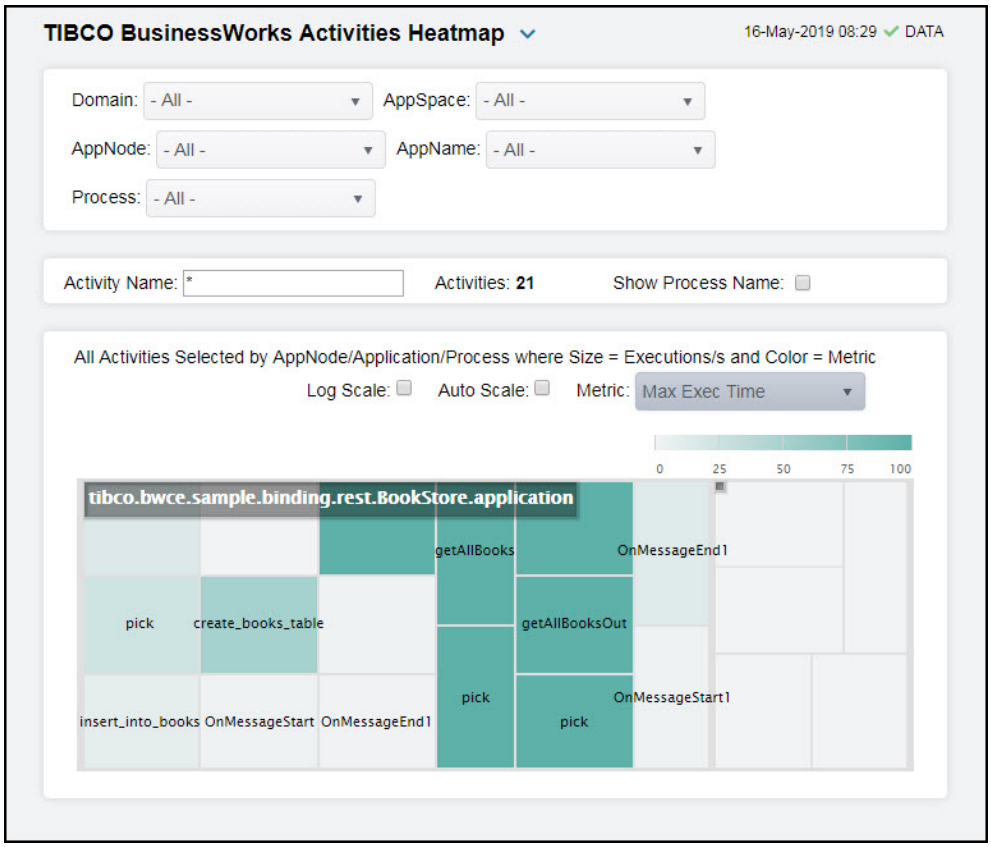
Error Count	The total number of errors accumulated.
Last Return Code	Status code (OK, DEAD, or ERROR) returned by most recent execution of this activity.
Execution Time	The amount of time used to execute processes.
Elapsed Time	The amount of elapsed time when executing processes.
Min Elapsed Time	The shortest amount of elapsed time.
Max Elapsed Time	The longest amount of elapsed time.
Min Execution Time	The shortest amount of time needed to execute a process.
Max Execution Time	The longest amount of time needed to execute a process.
Most Recent Exec Time ms	The time spent in execution since the last data update.
Most Recent Elapsed Time ms	The elapsed time accumulated since the last data update.
Exec Count Since Reset	Total number of process executions since the last data update.
Current Executions	The number of process executions since the last data update.
Execs/s	The rate of executions, per second, for the activity.
Current Elapsed Time	The elapsed time accumulated since the last data update.
Elapsed Time/s	The elapsed time spent, in milliseconds per second, for the activity.
Current Exec Time	The time spent in execution since the last data update.
Exec Time/s	The rate of executions, per second, for the activity.
Current Errors	
Errors/s	Number of errors accumulated since the last data update.
Avg Exec Time	The average execution time, in milliseconds, for the activity.
Avg Elapsed Time	The average elapsed time, in milliseconds, for the activity.
Avg Errors	The average number of errors for the activity.
Time Stamp	The date and time of the last data update.
Source	Name of RTView Data Server sending this data (or localhost).

TIBCO BusinessWorks Activities Heatmap - HTML

Clicking **BW Activities Heatmap** in the left/navigation menu opens the **TIBCO BusinessWorks Activities Heatmap**, which allows you to view the most critical BusinessWorks alerts pertaining to activity creation and execution. Use this display to quickly identify activities with critical alerts.

The heatmap is organized so that each rectangle represents a process. Move your mouse over a node to display current metrics. Click on a node to drill-down to the TIBCO BusinessWorks Activity Summary - HTML display to view specific metrics about process behavior over a specified period of time and determine which activity may be causing the bottleneck.

Choose a domain, AppSpace, AppNode, AppName, and process from the drop-down menus. Limit the activities listed in the table by specifying text in the **Activity Name** filter field. Click the **Show Process Name** check-box ☑ to include or exclude labels in the heatmap. Mouse over a rectangle to see additional metrics. By default, this display shows data based on the **Error Count** metric. Select a different metric from the **Metric** drop down menu to display the heatmap based on that metric.



Filter By:

The display might include these filtering options:

Domain:	Select the domain for which you want to view data in the display.
AppSpace	Select the AppSpace for which you want to view data in the display.
AppNode:	Select the AppNode for which you want to view data in the display.
AppName	Select the application for which you want to view data in the display.
Process	Select the process for which you want to view data in the display.
Activity Name Filter	Enter a string to limit data shown in the display.
Activities	The number of activities displayed in the heatmap as a result of the defined filters.
Show Process Names	Select this check box to display the names of the processes in their respective rectangles in the heatmap.
Log Scale	Select to enable a logarithmic scale. Use **Log Scale** to see usage correlations for data with a wide range of values. For example, if a minority of your data is on a scale of tens, and a majority of your data is on a scale of thousands, the minority of your data is typically not visible in non-log scale graphs. **Log Scale** makes

data on both scales visible by applying logarithmic values rather than actual values to the data.

Auto Scale Select to enable auto-scaling. When auto-scaling is activated, the color gradient bar's maximum range displays the highest value. NOTE: Some metrics auto-scale automatically, even when **Auto** is not selected.

Metric Select the metric driving the heatmap display. The default is **Error Count**. Each **Metric** has a color gradient bar that maps values to colors. The heatmap is organized so that each rectangle represents an activity. Mouse-over any rectangle to display the current values of the metrics for the activity. Click on a rectangle to drill-down to the associated TIBCO BusinessWorks Activity Summary - HTML display for a detailed view of metrics for that particular activity.

Alert Severity The maximum level of alerts in the heatmap rectangle. Values range from **0** - **2**, as indicated in the color gradient  bar, where **2** is the highest Alert Severity:

- Red indicates that one or more metrics exceeded their ALARM LEVEL threshold.
- Yellow indicates that one or more metrics exceeded their WARNING LEVEL threshold.
- Green indicates that no metrics have exceeded their alert thresholds.

Alert Count The total number of critical and warning alerts in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average alert count.

Error Count The total number of errors in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of errors in the heatmap. The middle value in the gradient bar indicates the average error count.

Errors/sec The number of errors per second in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of errors per second in the heatmap. The middle value in the gradient bar indicates the average errors per second count.

Exec Count The total number of executions in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of executions in the heatmap. The middle value in the gradient bar indicates the average execution count.

Exec Time/sec The process execution time per second in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average time.

Most Recent Exec Time The execution time for the most recently executed activity in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average time.

Max Exec Time The maximum execution time for all activities in the heatmap rectangle, calculated by dividing the delta execution time for the

bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average time.

TIBCO BusinessWorks Activity Summary - HTML

Clicking **BW Activity** in the left/navigation menu opens the **TIBCO BusinessWorks Activity Summary** display, which allows you to view current and historical execution metrics for a single BusinessWorks activity. Choose a domain, AppSpace, AppNode, Application Name, Process, and Activity from the drop-down menus.

Clicking on the information boxes at the top of the display takes you to the TIBCO BusinessWorks Activities Table - HTML display, where you can view additional activity data.

You can select from two different trend graphs: **Utilization** and **Performance**. In the **Utilization** trend graph region, you can view the rate of executions and average number of errors over a selected time range. In the **Performance** trend graph region, you can view the average elapsed time and the average execution time over a selected time range.

Filter By:

The display might include these filtering options:

Domain: Select the domain for which you want to view data in the display.

AppSpace Select the AppSpace for which you want to view data in the display.

AppNode: Select the AppNode for which you want to view data in the display.

AppName Select the application for which you want to view data in the display.

Process Select the process for which you want to view data in the display.

Activity Name Select the activity name for which you want to view data in the display.

Fields and Data

Executions/s The rate of executions, per second, for the activity.

Errors/s The rate of errors being generated, per second, for the activity.

Rate Exec Time ms/s Time spent in execution, in milliseconds per second, for the activity.

Avg Exec Time ms The average execution time, in milliseconds, for the activity.

Rate Elapsed Time ms/s The elapsed time spent, in milliseconds per second, for the activity.

Avg Elapsed Time ms The average elapsed time, in milliseconds, for the activity.

Trend Graphs

Utilization

Executions -- Traces the number of executions for the activity.

Average Errors -- Traces the average number of errors for the activity.

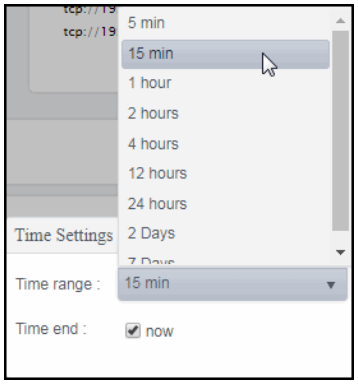
Performance

Average Elapsed Time ms -- Traces the average elapsed time, in milliseconds, for the activity.

Average Exec Time ms -- Traces the average execution time, in milliseconds, for the activity.

Log Scale Select to enable a logarithmic scale. Use **Log Scale** to see usage correlations for data with a wide range of values. For example, if a minority of your data is on a scale of tens, and a majority of your data is on a scale of thousands, the minority of your data is typically not visible in non-log scale graphs. **Log Scale** makes data on both scales visible by applying logarithmic values rather than actual values to the data.

Time Settings Select a time range from the drop down menu varying from **5 Minutes** to **Last 7 Days**. By default, the time range end point is the current time.



To change the time range, deselect the **now** toggle, which displays some additional date fields. You can click the left and right arrow buttons to decrease the end time by one time period (the time selected in the **Time range** drop down) per click, or you can choose the date and time from the associated calendar and clock icons. You can also enter the date and time in the text field using the following format: **MMM dd, YYYY**

HH:MM:ss. For example, Aug 21, 2018 12:24 PM. Click the **now** toggle to reset the time range end point to the current time.

Execution Count	The total number of process executions.
Error Count	The total number of errors accumulated.
Exec Count Since Reset	Total number of process executions since the last data update.
Avg Errors	The average number of errors for the activity.
Current Executions	The number of process executions since the last data update.
Current Errors	Number of errors accumulated since the last data update.
Min Execution Time	The shortest amount of time needed to execute a process.
Min Elapsed Time	The shortest amount of elapsed time.
Current Exec Time	The time spent in execution since the last data update.
Current Elapsed Time	The elapsed time accumulated since the last data update.
Max Execution Time	The longest amount of time needed to execute a process.
Max Elapsed Time	The longest amount of elapsed time.
Last Update	The date and time of the last data update.

TIBCO BusinessWorks 5 Monitor HTML Views/Displays

The following Views and their associated displays are in the Monitor. This section describes the Monitor displays and includes:

- TIBCO BusinessWorks 5 Overview Display: Describes the TIBCO BusinessWorks Overview display.
- BW5 Servers - HTML: The displays in this View present BusinessWorks 5.0 server performance metrics.
- "BW5 Engines - HTML": The displays in this View present BusinessWorks 5.0 engine performance metrics.
- "BW5 Processes - HTML": The displays in this View present BusinessWorks 5.0 process performance metrics.
- "BW5 Activities - HTML": The displays in this View present BusinessWorks 5.0 activity performance metrics.

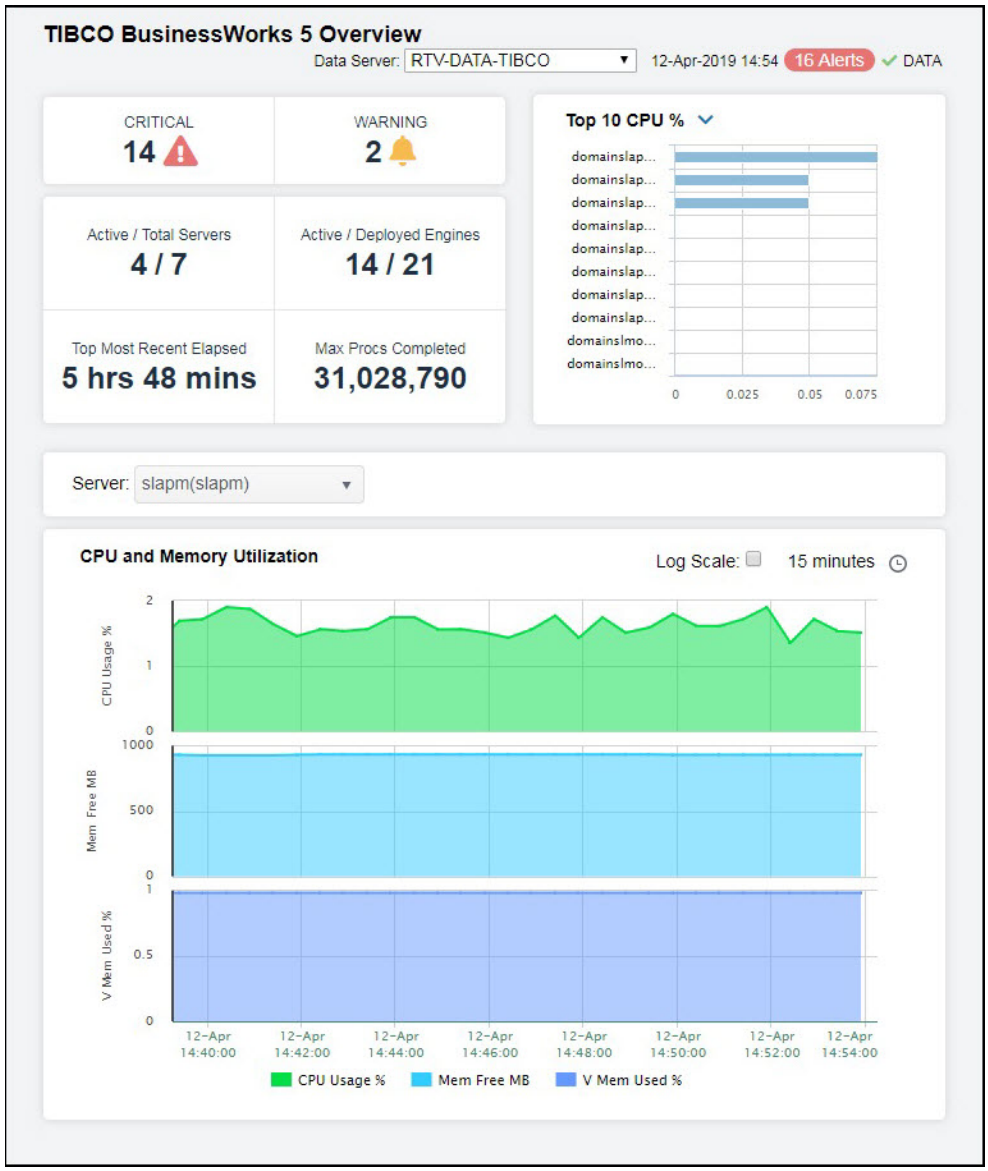
TIBCO BusinessWorks 5 Overview Display

The **TIBCO BusinessWorks 5 Overview** is the top-level display for the TIBCO Enterprise BusinessWorks 5 Monitor, which provides a good starting point for immediately getting the status of all your servers, engines, and processes on your Data Server. You can select the RTView DataServer for which you want to see data and easily view the current data for that DataServer including:

- The total number of active alerts for the selected DataServer, including the total number of critical and warning alerts.
- The number of active servers and the total number of servers.
- The number of active and deployed engines on your connected DataServer.
- The top most recent elapsed time for a process on your connected DataServer.
- The maximum number of processes completed on one engine on your connected DataServer.
- A visual list of the top 10 servers containing the highest CPU usage percentage/memory used percentage/completed processes/error rate per second on your connected DataServer.

You can hover over each region in the upper half of the Overview to see more detail. You can also drill down to see even more detail by clicking on each respective region in the Overview. For example, clicking on the alerts in the **CRITICAL** and **WARNING** alerts region opens the **Alerts Table by Components** display.

The bottom half of the display provides a CPU and memory utilization trend graph for a selected server. You can hover over the trend graph to see the values at a particular time. You can specify the time range for the trend graph and view data based on a log scale, which enables visualization on a logarithmic scale and should be used when the range in your data is very broad.



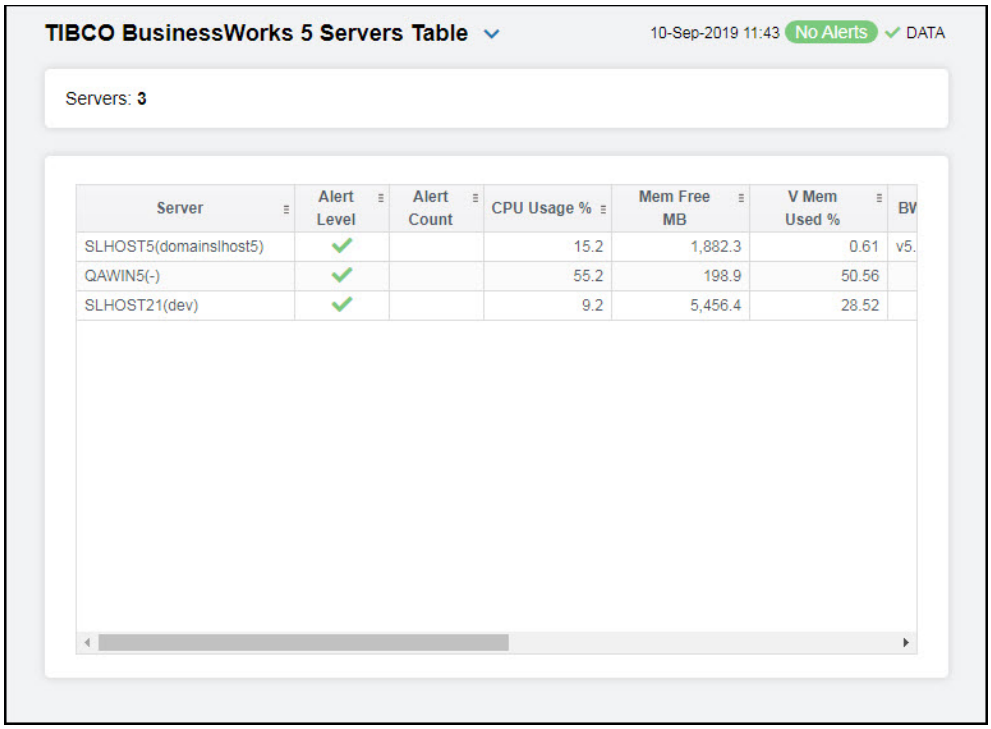
BW5 Servers - HTML

These displays present performance metrics and alert status for all BW5 servers. Clicking **BW5 Servers** from the left/navigation menu opens the TIBCO BusinessWorks 5 Servers Table - HTML display, which shows all available utilization metrics for all BW5 servers. The options available under **BW5 Servers** are:

- **All Servers Heatmap**: Opens the TIBCO BusinessWorks 5 Servers Heatmap - HTML, which shows server and alert status for all BW5 servers.
- **BW5 Server Summary**: Opens the TIBCO BusinessWorks 5 Server Summary - HTML display, which shows information for a single BW5 server.

TIBCO BusinessWorks 5 Servers Table - HTML

Investigate detailed utilization metrics for all BW servers. The **TIBCO BusinessWorks 5 Servers Table** contains all metrics available for servers, including CPU usage percentage, free memory, and percentage of virtual memory used. Each row in the table contains data for a particular server. Click a column header to sort column data in ascending or descending order. Double-click on a table row to drill-down to the TIBCO BusinessWorks 5 Server Summary - HTML display and view metrics for that particular server. Toggle between the commonly accessed **Table** and **Heatmap** displays by clicking the drop down list on the display title.



Fields and Data

Servers: The number of servers listed in the table.

Table:

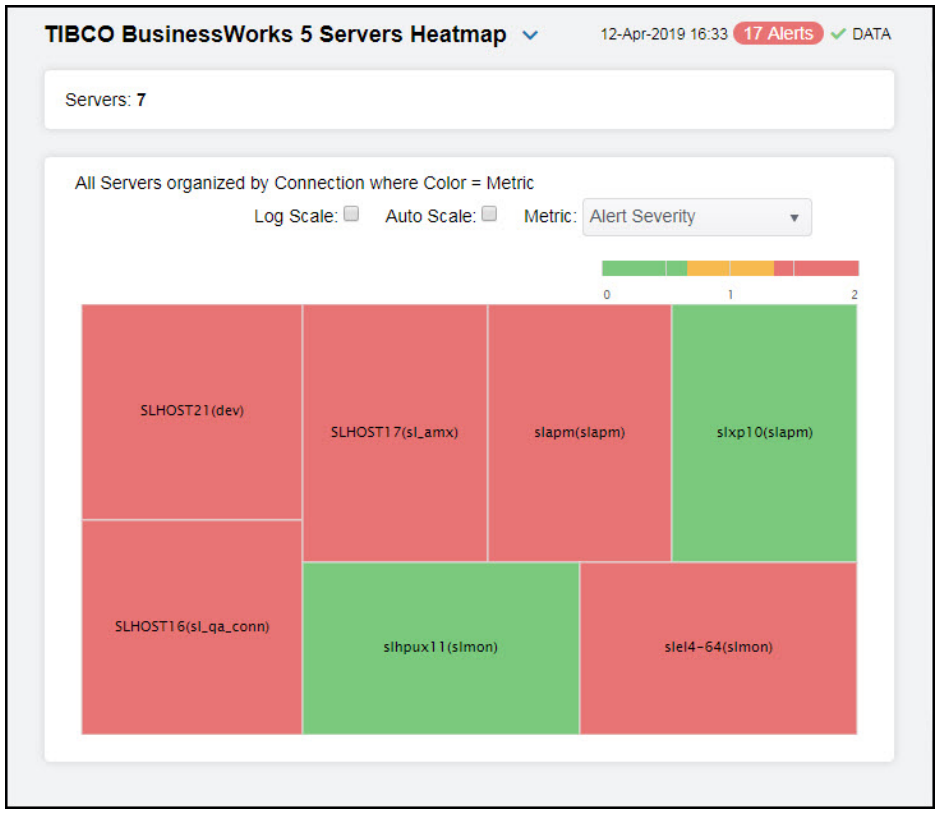
Server	Name of Server Agent.
Alert Level	The most critical alert state for alerts in the row: Red indicates that one or more metrics exceeded their ALARM LEVEL threshold. Yellow indicates that one or more metrics exceeded their WARNING LEVEL threshold. Green indicates that no metrics have exceeded their alert thresholds.
Alert Count	The total number of active alerts for the application.
CPU Usage %	Percent of server CPU in use.
Mem Free MB	Available physical memory (MB) remaining.
V Memory Used %	Percent of virtual memory used.
BW Version	The TIBCO BusinessWorks version currently in use on the server.

Source	Name of RTView Data Server sending this data (or localhost).
Deployed Engines	Total number of engines deployed on the server.
Active Engines	Number of engines currently active.
Expired	When checked, data has not been received from this host in the specified amount of time.
Time Stamp	Time this data was retrieved.

TIBCO BusinessWorks 5 Servers Heatmap - HTML

Clicking **All Servers Heatmap** in the left/navigation menu opens the **TIBCO BW5 Servers Heatmap**, which allows you to view the status and alerts of all BW5 servers. Use the **Metric** drop-down menu to view the **Alert Severity**, **Alert Count**, **CPU Used Percentage**, **Virtual Memory Used Percentage**, **Free Memory**, **Deployed Engines**, or **Active Engines**.

The heatmap is organized by host with each rectangle representing a server. The rectangle color indicates the most critical alert state. Click on a node to drill-down to the TIBCO BusinessWorks 5 Server Summary - HTML display and view metrics for a particular server. Toggle between the commonly accessed **Table** and **Heatmap** displays by clicking the drop down list on the display title. Mouse-over rectangles to view more details about host performance and status.



Fields and Data:

Servers: The total number of servers in the display.

Log Scale Select to enable a logarithmic scale. Use **Log Scale** to see usage correlations for data with a wide range of values. For example, if a minority of your data is on a scale of tens, and a majority of your data is on a scale of thousands, the minority of your data is typically not visible in non-log scale graphs. **Log Scale** makes data on both scales visible by applying logarithmic values rather than actual values to the data.

Auto Scale Select to enable auto-scaling. When auto-scaling is activated, the color gradient bar's maximum range displays the highest value. NOTE: Some metrics auto-scale automatically, even when **Auto** is not selected.

Metric Select the metric driving the heatmap display. The default is **Alert Severity**. Each **Metric** has a color gradient bar that maps values to colors. The heatmap organizes the servers by host, where each rectangle represents a server. Mouse-over any rectangle to display the current values of the metrics for the Server. Click on a rectangle to drill-down to the associated TIBCO BusinessWorks 5 Server Summary - HTML display for a detailed view of metrics for that particular server.

Alert Severity The maximum level of alerts in the heatmap rectangle. Values range from **0** - **2**, as indicated in the color gradient  bar, where **2** is the highest Alert Severity:

- Red indicates that one or more metrics exceeded their ALARM LEVEL threshold.
- Yellow indicates that one or more metrics exceeded their WARNING LEVEL threshold.
- Green indicates that no metrics have exceeded their alert thresholds.

Alert Count The total number of critical and warning alerts in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average alert count.

CPU Used% The percent (%) CPU used in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the alert threshold of **BwServersCpuUsedHigh**, which is **100**. The middle value in the gradient bar indicates the middle value of the range (the default is **50**).

V(irtual) Memory Used% The percent (%) virtual memory used in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the alert threshold of virtual memory used percentage in the heatmap. The middle value in the gradient bar indicates the middle value of the range (the default is **50**).

Free Memory The amount of free memory in the heatmap rectangle, in megabytes. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the amount of free memory in the heatmap. The middle value in the gradient bar indicates the middle value of the range.

Deployed Engines The number of deployed engines in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping.The numerical values in the gradient bar range from **0** to the number of depoloyed engines in the heatmap. The middle value in the gradient bar indicates the middle value of the range.

Active Engines The number of active engines in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar

range from **0** to the number of active engines in the heatmap. The middle value in the gradient bar indicates the middle value of the range.

TIBCO BusinessWorks 5 Server Summary - HTML

Clicking **BW5 Server Summary** in the left/navigation menu opens the **TIBCO BusinessWorks 5 Server Summary** display, which allows you to track utilization and performance metrics for specific BW5 servers. Clicking on the information boxes at the top of the display takes you to the TIBCO BusinessWorks 5 Servers Table - HTML display, the TIBCO BusinessWorks 5 Engines Table - HTML display, or the TIBCO BusinessWorks 5 Processes Table - HTML display (depending on which box you select), where you can view additional servers data. In the **CPU and Memory Utilization** trend graph region, you can view CPU usage percentage, free memory, and virtual memory used percentage over a selected time range. Clicking the **Critical/Warning** link at the bottom of the display opens the Alerts Table by Component display.

Filter By:

The display might include these filtering options:

Server: Select the server for which you want to see data.

Fields and Data

CPU Usage % Percent of server CPU in use. Click to drill-down to view additional details in the TIBCO BusinessWorks 5 Servers Table - HTML display.

Memory Free MB Available physical memory remaining (in MB). Click to drill-down to view additional details in the TIBCO BusinessWorks 5 Servers Table - HTML display.

Virtual Memory Usage % Percent of virtual memory used. Click to drill-down to view additional details in the TIBCO BusinessWorks 5 Servers Table - HTML display.

Deployed Engines Number of engines currently active. Click to drill-down to view additional details in the TIBCO BusinessWorks 5 Engines Table - HTML display.

Active Engines Shows data for the server. Click to drill-down to view additional details in the TIBCO BusinessWorks 5 Engines Table - HTML display.

Total Running Processes The number of running processes on this server across all engines. Click to drill-down to view additional details in the TIBCO BusinessWorks 5 Processes Table - HTML display.

CPU and Memory Utilization Trend Graph

CPU Usage % -- Traces the percentage of server CPU in use.

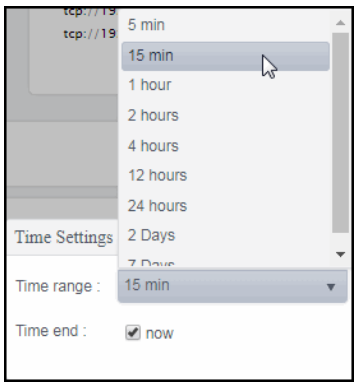
Mem Free MB -- Traces the available physical memory remaining (in MB).

V Mem Used % -- Traces the percentage of virtual memory used.

Log Scale Select to enable a logarithmic scale. Use **Log Scale** to see usage correlations for data with a wide range of values. For example, if a minority of your data is on a scale of tens, and a majority of your data is on a scale of thousands, the minority of your data is typically not visible in non-log scale graphs. **Log Scale** makes data on both scales visible by applying logarithmic values rather than actual values to the data.

Time Settings

Select a time range from the drop down menu varying from **5 Minutes** to **Last 7 Days**. By default, the time range end point is the current time.



To change the time range, deselect the **now** toggle, which displays some additional date fields. You can click the left and right arrow buttons to decrease the end time by one time period (the time selected in the **Time range** drop down) per click, or you can choose the date and time from the associated calendar and clock icons. You can also enter the date and time in the text field using the following

format: **MMM dd, YYYY HH:MM:ss**. For example, Aug 21, 2018 12:24 PM. Click the **now** toggle to reset the time range end point to the current time.

BW Version	The currently deployed version of TIBCO BusinessWorks 5.
Status	Server status: ACTIVE or EXPIRED.
Critical/Warning	The number of critical and warning alerts.
Threads	The number of threads.
Last Update	The date and time of the last data update.

BW5 Engines - HTML

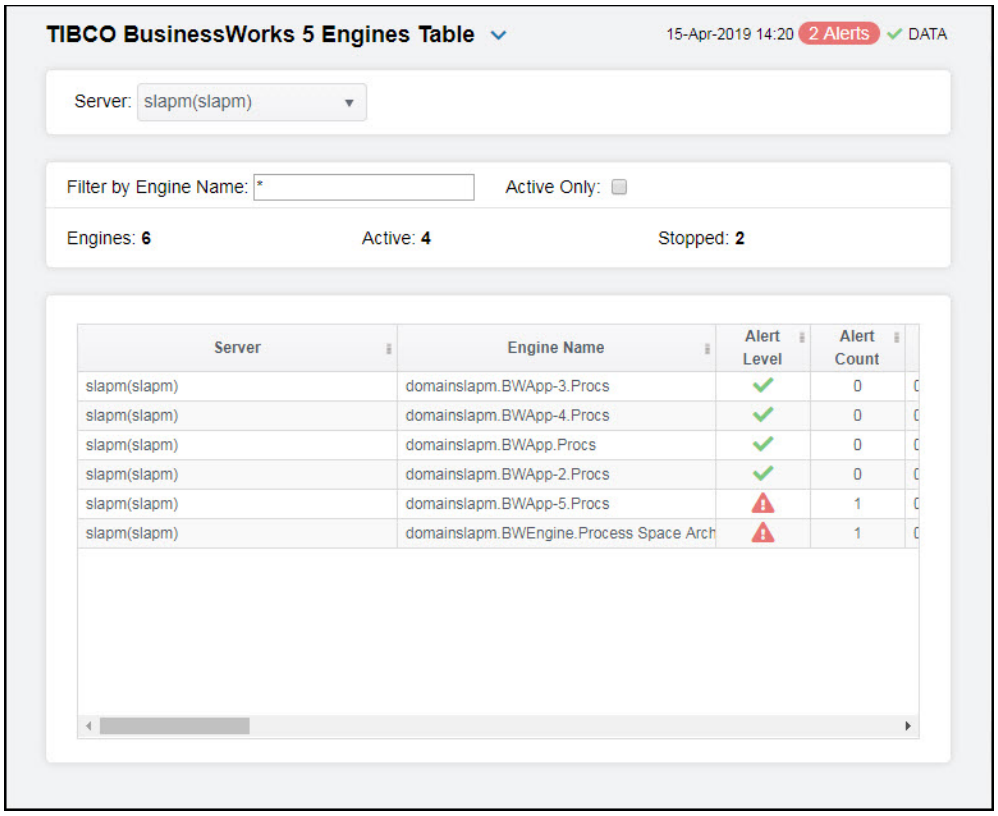
These displays present performance metrics and alert status for all BW5 engines. Clicking **BW5 Engines** from the left/navigation menu opens the TIBCO BusinessWorks 5 Engines Table - HTML display, which shows all available utilization metrics for all BW5 engines. The options available under **BW5 Engines** are:

- **All Engines Heatmap**: Opens the TIBCO BusinessWorks 5 Engines Heatmap - HTML, which shows engine and alert status for all BW5 servers.
- **BW5 Engine Summary**: Opens the TIBCO BusinessWorks 5 Engine Summary - HTML display, which shows information for a single BW5 engine.

TIBCO BusinessWorks 5 Engines Table - HTML

Investigate detailed utilization metrics for all BW engines. The **TIBCO BusinessWorks 5 Engines Table** contains all metrics available for engines, including memory usage, memory used percentage, and CPU used percentage. You can enter a string in the **Filter by Engine Name** field to show only engines in the table with names that contain the string. For example, if you enter the string Madrid, all engines with Madrid in the engine name are shown in the table. If no entry is made, all engine names are shown. For most use cases, you can enter a portion of the engine name. Each row in the table contains data for a particular engine. Click a column header to sort column data in ascending or descending order. Double-click on a table row to drill-down to the TIBCO BusinessWorks 5 Engine Summary - HTML display and view metrics for that particular engine. Toggle between the commonly accessed **Table** and **Heatmap** displays by clicking the drop down list on the display title.

Note: Metrics are made available by the Hawk microagent for the engine (for details, refer to documentation for TIBCO BusinessWorks Administration, Appendix A: TIBCO Hawk Microagent Methods).



Filter By:

The display might include these filtering options:

Server:	Select the server for which you want to show data in the display.
Filter by Engine Name:	Enter all or part of engine name to view specific engines.
Active Only	If selected, only engines with a status of ACTIVE are displayed. Otherwise, if deselected, all engines for the given Filter/Server selection are displayed.
Engines	Number of engines currently being displayed.
Active	Number of engines currently active.
Stopped	The number of stopped engines.

Table:

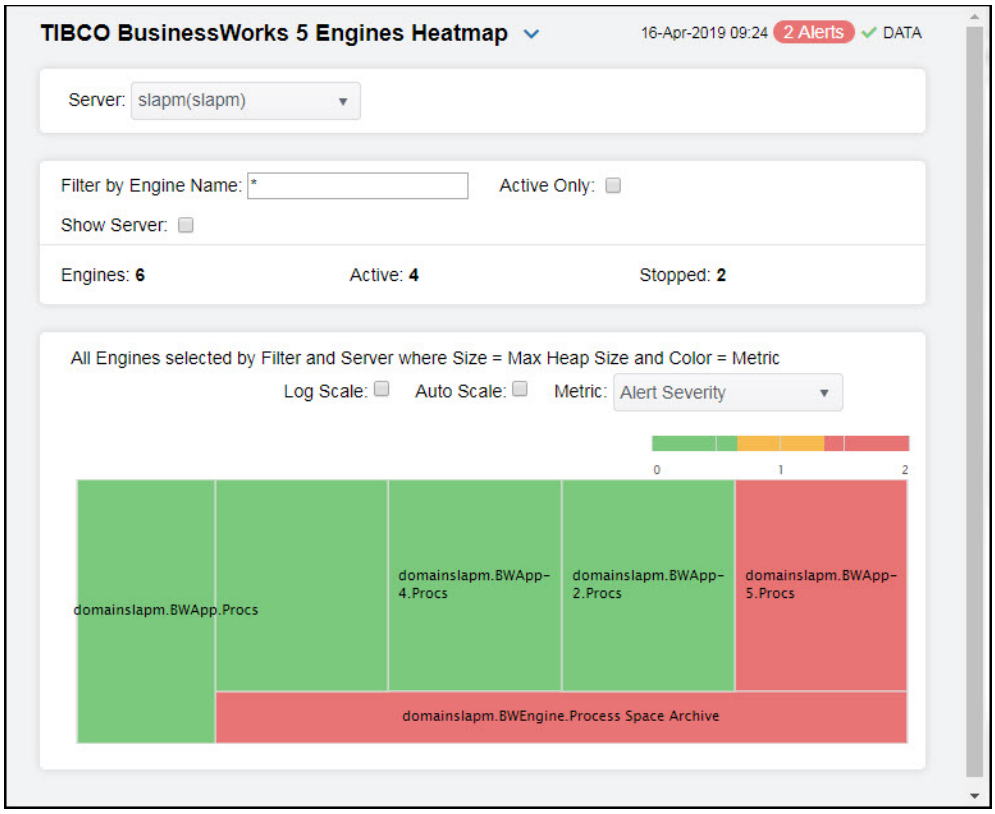
Server	The name of the Server agent.
Engine Name	The name of the engine.
Alert Level	The most critical alert state for alerts in the row: Red indicates that one or more metrics exceeded their ALARM LEVEL threshold. Yellow indicates that one or more metrics exceeded their WARNING LEVEL threshold. Green indicates that no metrics have exceeded their alert thresholds.
Alert Count	Number of current alerts
Micro Agent Instance	Unique ID of the microagent reporting the metrics.
Process ID	Process ID of engine as recognized by the server.
State	Engine status: ACTIVE, STOPPED, LIMITED, etc. (See **All Servers Grid**).

Uptime	Uptime in milliseconds since the engine was started.
Threads	Number of threads used by the engine.
CPU %	Percent of server CPU used by engine.
Running Processes	Number of running processes.
Total Bytes	Maximum heap memory this JVM has used.
Free Bytes	Amount of available memory from within the JVM.
Used Bytes	Total bytes of memory within the JVM currently used by the engine. Equal to value of: MaxBytes minus FreeBytes.
Mem Usage KB	Server memory, in KB, used by the engine.
Mem Used %	Percentage of allocated memory currently consumed by this engine from within the JVM. Equal to the value of: (100*UsedBytes) divided by MaxBytes. NOTE: Percent used is Long.
Total Errors	Total number of errors since the engine was started.
Current Errors	Current number of new errors.
Errors/s	The rate off errors occurring.
Created/s	The rate of processes being created.
Completed/s	The rate of processes being completed.
Active/s	The rate of processes becoming active.
Aborted/s	The rate of processes being aborted.
Deployment	The name of the deployment.
Domain	The name of the domain.
Max Heap Size Bytes	Maximum heap allocated to this engine for the JVM, in bytes.
BW Version	Engine project version number.
Source	Name of RTView Data Server sending this data (or localhost).
Host	Host name of server.
Time Stamp	The date and time of the last data update.

TIBCO BusinessWorks 5 Engines Heatmap - HTML

Clicking **All Engines Heatmap** in the left/navigation menu opens the **TIBCO BW5 Engines Heatmap**, which allows you to view the status and alerts of all BW5 engines. You can enter a string in the **Filter by Engine Name** field to show only engines in the heatmap with names that contain the string. For example, if you enter the string Madrid, all engines with Madrid in the engine name are shown in the table. If no entry is made, all engine names are shown. For most use cases, you can enter a portion of the engine name. Use the **Metric** drop-down menu to view the **Alert Severity**, **Alert Count**, **CPU Used Percentage**, **Memory Used Percentage**, **Running Processes**, or **Error Count**.

The heatmap is organized by host with each rectangle representing an engine. Rectangle size represents Max Heap Size and the color indicates the most critical alert state. Click on a node to drill-down to the TIBCO BusinessWorks 5 Engine Summary - HTML display and view metrics for a particular engine. Toggle between the commonly accessed **Table** and **Heatmap** displays by clicking the drop down list on the display title. Mouse-over rectangles to view more details about host performance and status.



Filter By:

The display might include these filtering options:

Server:	Select the server for which you want to show data in the display.
Filter by Engine Name:	Enter all or part of engine name to view specific engines.
Active Only	If selected, only engines with a status of ACTIVE are displayed. Otherwise, if deselected, all engines for the given Filter/Server selection are displayed.
Show Server	Select this check box to display the associated server names in the heatmap.
Engines	Number of engines currently being displayed.
Active	Number of engines currently active.
Stopped	The number of stopped engines.
Log Scale	Select to enable a logarithmic scale. Use **Log Scale** to see usage correlations for data with a wide range of values. For example, if a minority of your data is on a scale of tens, and a majority of your data is on a scale of thousands, the minority of your data is typically not visible in non-log scale graphs. **Log Scale** makes data on both scales visible by applying logarithmic values rather than actual values to the data.
Auto Scale	Select to enable auto-scaling. When auto-scaling is activated, the color gradient bar's maximum range displays the highest value. NOTE: Some metrics auto-scale automatically, even when **Auto** is not selected.
Metric	Select the metric driving the heatmap display. The default is **Alert Severity**. Each **Metric** has a color gradient bar that maps values to colors. The heatmap organizes the servers by host, where each rectangle represents an engine. Mouse-over any rectangle to display the current values of the metrics for the engine. Click on a rectangle to drill-down to the associated TIBCO BusinessWorks 5 Engine Summary - HTML display for a detailed view of metrics for that particular engine.

Alert Severity	The maximum level of alerts in the heatmap rectangle. Values range from **0** - **2**, as indicated in the color gradient  bar, where **2** is the highest Alert Severity: ● Red indicates that one or more metrics exceeded their ALARM LEVEL threshold. ● Yellow indicates that one or more metrics exceeded their WARNING LEVEL threshold. ● Green indicates that no metrics have exceeded their alert thresholds.
Alert Count	The total number of critical and warning alerts in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average alert count.
CPU Used%	The percent (%) CPU used in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the alert threshold of **BwEngineCpuUsedHigh**, which is **100**. The middle value in the gradient bar indicates the middle value of the range (the default is **50**).
Memory Used%	The percent (%) memory used in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the alert threshold of **BwEngineMemUsedHigh**, which is **100**. The middle value in the gradient bar indicates the middle value of the range (the default is **50**).
Active Processes	The number of currently active processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of active processes in the heatmap. The middle value in the gradient bar indicates the middle value of the range.
Running Processes	The number of currently running processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of running processes in the heatmap. The middle value in the gradient bar indicates the middle value of the range.
Created Processes	The number of created processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of created processes in the heatmap. The middle value in the gradient bar indicates the middle value of the range.
Created/sec	The number of created processes in the heatmap rectangle, per second. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum rate of processes created per second in the heatmap. The middle value in the gradient bar indicates the middle value of the range.
Error Count	The total number of errors in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of errors in the heatmap. The middle value in the gradient bar indicates the average alert count.

TIBCO BusinessWorks 5 Engine Summary - HTML

Clicking **BW5 Engine Summary** in the left/navigation menu opens the **TIBCO BusinessWorks 5 Engine Summary** display, which allows you to track utilization and performance metrics for specific BW5 engines and their associated processes. Clicking on the information boxes at the top of the display takes you to the TIBCO BusinessWorks 5 Engines Table - HTML display or to the TIBCO BusinessWorks 5 Processes Table - HTML display (depending on which box you select), where you can view additional data on engines and processes. In the **Processes organized by Server/Engine** heatmap, you can select from the available metrics to view the current status of the processes running on the selected engine. Available metrics include **Alert Severity**, **Alert Count**, **Created Processes**, **Completed Processes**, and **Average Execution**.

There are two options in the trend graph: **System Utilization** and **Memory Utilization**. In the **System Utilization** option on the trend graph, you can view trend data for running processes and CPU percentage over a selected time range. In the **Memory Utilization** option on the trend graph, you can view trend data for used memory (in megabytes) and current memory (in megabytes) over a selected time range.

Clicking the **Critical/Warning** link at the bottom of the display opens the Alerts Table by Component display.

Filter By:

The display might include these filtering options:

Server: Select the server for which you want to view data in the display.

Engine: Select the engine for which you want to view data in the display

Fields and Data

CPU % The percentage of CPU utilization on the engine.

Total Threads The total number of threads being executed on the engine.

Memory Used % The percentage of memory utilization on the engine.

Total Processes The total number of deployed processes on the engine.

Running Processes The total number of running processes on the engine.

Total Errors The total number of errors generated on the engine.

Heatmap Shows processes organized by Server/Engine where Size = Creation Count and Color = Average Execution. Click on a node to drill down to a specific engine.

Metric Select the metric driving the heatmap display. The default is **Alert Severity**. Each **Metric** has a color gradient bar that maps values to colors. The heatmap organizes the servers by host, where each rectangle represents an engine. Mouse-over any rectangle to display the current values of the metrics for the engine. Click on a rectangle to drill-down to the associated TIBCO BusinessWorks 5 Process Summary - HTML display for a detailed view of metrics for that particular engine.

- **Alert Severity** The maximum level of alerts in the heatmap rectangle. Values range from **0** - **2**, as indicated in the color gradient  bar, where **2** is the highest Alert Severity:
  - Red indicates that one or more metrics exceeded their ALARM LEVEL threshold.
  - Yellow indicates that one or more metrics exceeded their WARNING LEVEL threshold.
  - Green indicates that no metrics have exceeded their alert thresholds.
- **Alert Count** The total number of critical and warning alerts in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average alert count.
- **Created Processes** The number of created processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of created processes in the heatmap. The middle value in the gradient bar indicates the middle value of the range.
- **Completed Processes** The number of completed processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of created processes in the heatmap. The middle value in the gradient bar indicates the middle value of the range.
- **Avg Execution** The average exection time for processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of created processes in the heatmap. The middle value in the gradient bar indicates the middle value of the range.

Trend Graphs

System Utilization

CPU % -- Traces the CPU utilization for the engine.

Running Processes -- Traces the number of running processes on the engine.

Memory Utilization

Current Mem MB -- Traces the current memory available for the engine.

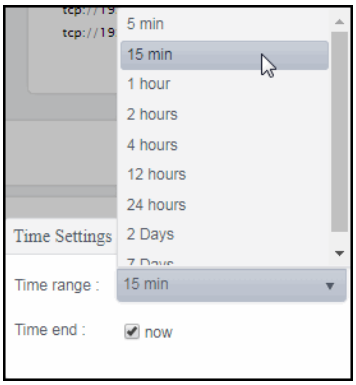
Used Mem MB -- Traces the used memory on the engine.

Log Scale

Select to enable a logarithmic scale. Use **Log Scale** to see usage correlations for data with a wide range of values. For example, if a minority of your data is on a scale of tens, and a majority of your data is on a scale of thousands, the minority of your data is typically not visible in non-log scale graphs. **Log Scale** makes data on both scales visible by applying logarithmic values rather than actual values to the data.

Time Settings

Select a time range from the drop down menu varying from **5 Minutes** to **Last 7 Days**. By default, the time range end point is the current time.



To change the time range, deselect the **now** toggle, which displays some additional date fields. You can click the left and right arrow buttons to decrease the end time by one time period (the time selected in the **Time range** drop down) per click, or you can choose the date and time from the associated calendar and clock icons. You can also enter the date and time in the text field using the following format: **MMM dd, YYYY HH:MM:ss**. For example, Aug 21, 2018 12:24 PM. Click the **now** toggle to reset the time range end point to the current time.

State

ACTIVE -- The BW microagent is providing live data and the engine is assumed active.

SUSPENDED -- This state is reported by the BW microagent.

STANDBY -- This state is reported by the BW microagent.

STOPPING -- This state is reported by the BW microagent.

STOPPED -- This state is reported by the BW microagent.

LIMITED -- Live data has been received from TIBCO, but deployment data from the custom RTView MicroAgent has not been received.

EXPIRED -- The associated server for the engine is currently in an EXPIRED state and is unavailable or stopped sending data.

Free Memory MB	The amount of free memory, in megabytes.
Total Created Processes	A BW Engine runs processes by creating instances of process definitions and making them active. A given process instance has a lifetime during which it may be suspended, swapped, queued, etc. until it is either completed or aborted. The Total value is calculated using the Hawk method named GetProcessDefinitions that returns statistics about the instances of each process definition including cumulative counts of completed, aborted, suspended, etc.
Total Aborted Processes	
Critical/Warning	The number of critical and warning alerts.
Domain	The name of the domain.
Total Completed Processes	The total number of completed processes.
Uptime	Days, hours, and minutes since the engine was started.
BW Version	
Last Update	The date and time of the last data update.

BW5 Processes - HTML

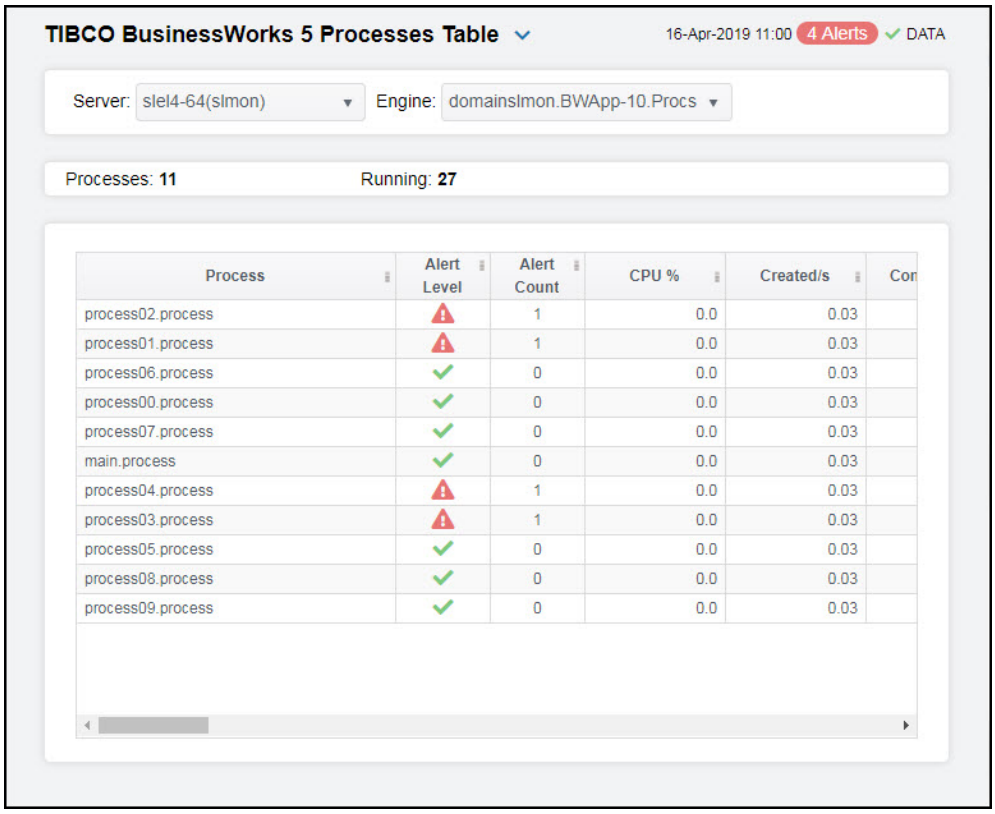
These displays present performance metrics and alert status for all BW5 processes. Clicking **BW5 Processes** from the left/navigation menu opens the TIBCO BusinessWorks 5 Processes Table - HTML display, where each row in the table displays all available metrics from the Hawk microagent for a process. The options available under **BW5 Processes** are:

- **All Processes Heatmap**: Opens the TIBCO BusinessWorks 5 Processes Heatmap - HTML, which shows process execution metrics for all BW Engines.
- **BW5 Process Summary**: Opens the TIBCO BusinessWorks 5 Process Summary - HTML display, which shows historical and current metrics for a single process, including average execution times and execution counts.

TIBCO BusinessWorks 5 Processes Table - HTML

Select a server and engine from the drop-down menus. Each row in the table is a different process. The table displays all metrics available from the Hawk microagent for a process. (Refer to documentation for TIBCO BusinessWorks Administration, see Appendix A: TIBCO Hawk Microagent Methods).

Click a column header to sort column data in ascending or descending order. Double-click on a table row to drill-down to the TIBCO BusinessWorks 5 Process Summary - HTML display and view metrics for that particular process. Toggle between the commonly accessed **Table** and **Heatmap** displays by clicking the drop down list on the display title.



Filter By:

The display might include these filtering options:

Server:	Select the server for which you want to view data in the display.
Engine:	Select the engine for which you want to view data in the display.

Processes The number of processes listed in the table.

Table:

Process	The name of the process.
Alert Level	The most critical alert state for alerts in the row: ● Red indicates that one or more metrics exceeded their ALARM LEVEL threshold. ● Yellow indicates that one or more metrics exceeded their WARNING LEVEL threshold. ● Green indicates that no metrics have exceeded their alert thresholds.
Alert Count	Number of current alerts
Total CPU	Total CPU usage in percent.
Created/s	Change in Created per second.
Completed/s	Change in Completed per second.
Created	Number of process instances created for this process definition.
Completed	Number of process instances successfully completed.
Total Exec Time ms	Total execution time (in milliseconds) for all successfully completed process instances.
Exec Time/s	Delta execution time per second.
Current Total Execs	Execution time accumulated during the current polling period.
Min Exec Time	Execution time (in milliseconds) of the process instance that has completed

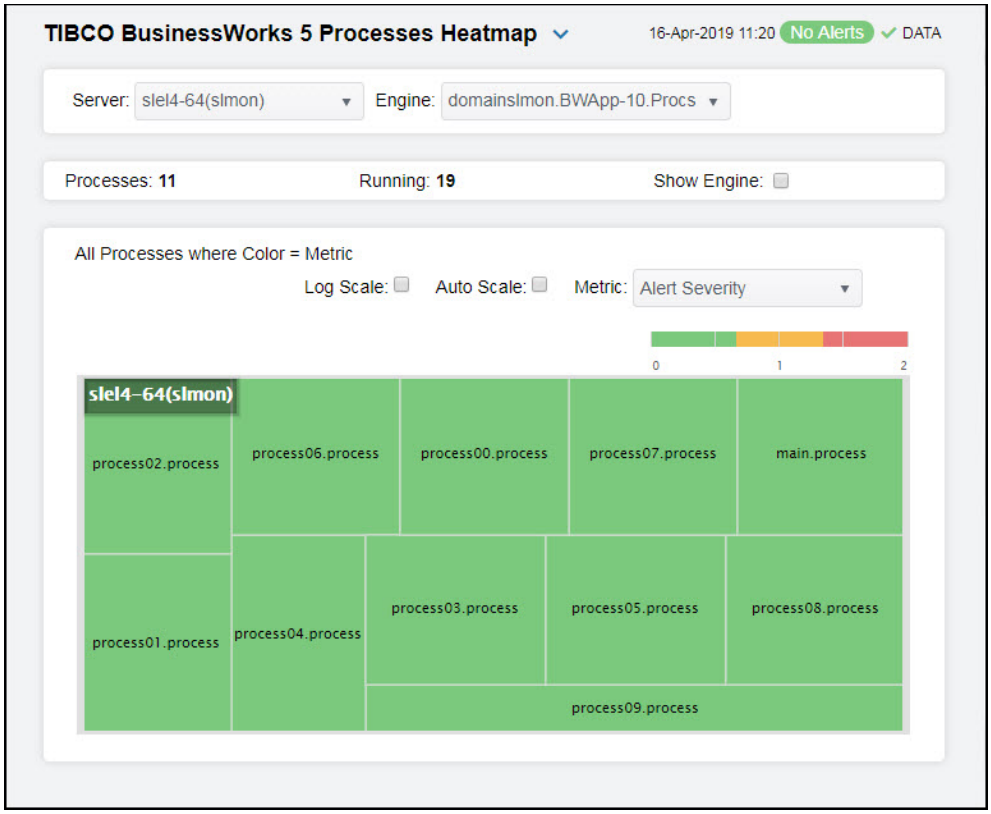
ms	in the shortest amount of execution time.
Max Exec Time ms	Execution time (in milliseconds) of the process instance that has completed in the longest amount of execution time.
Avg Exec Time ms	Average execution time (in milliseconds) for all successfully completed process instances.
Most Recent Exec Time ms	Execution time (in milliseconds) of the most recently completed process instance.
Total Elapsed Time ms	Total elapsed time (in milliseconds) for all successfully completed process instances.
Current Elapsed ms	Elapsed time accumulated during the current polling period.
Elapsed Time/s	Delta elapsed time per second.
Min Elapsed Time ms	Elapsed clock time (in milliseconds) of the process instance that has completed in the shortest amount of elapsed time.
Max Elapsed Time ms	Elapsed clock time (in milliseconds) of the process instance that has completed in the longest amount of elapsed time.
Avg Elapsed Time ms	Average elapsed clock time (in milliseconds) for all successfully completed process instances.
Most Recent Elapsed Time	Elapsed clock time (in milliseconds) of the most recently completed process instance.
Aborted	Number of times process instances have been aborted.
Current Aborted	Change in Aborted this update.
Aborted/s	Change in Aborted per second.
Queued	Number of times process instances have been queued for execution.
Current Queued	Change in Queued this update.
Queued/s	Change in Queued per second.
Suspended	Number of times process instances have been suspended.
Current Suspended	Change in Suspended this update.
Suspended/s	Change in Suspended per second.
Checkpointed	Number of times process instances have executed a checkpoint.
Current Checkpointed	Change in Checkpointed this update.
Checkpointed/s	Change in Checkpointed per second.
Swapped	Number of times process instances have been swapped to disk.
Current Swapped	Change in Swapped this update.
Swapped/s	Change in Swapped per second.
Time Since Last Update	Time since the last update.
Domain	Name of TIBCO Domain.
Starter	Name of the process starter for the process.
Micro Agent Instance	Unique ID of the microagent reporting the metrics.
Count Since Reset	Number of process instances that have completed since the last reset of the statistics.
Source	Name of RTView Data Server sending this data (or localhost).
Time Stamp	Time of last update.

TIBCO BusinessWorks 5 Processes Heatmap - HTML

Clicking **All Processes Heatmap** in the left/navigation menu opens the **TIBCO BW5 Processes Heatmap**, which allows you to view the status and alerts of all BW5 processes for

all engines or for a specific engine. Use the **Metric** drop-down menu to view processes in the heatmap by the **Alert Severity**, **Alert Count**, **CPU Used Percentage**, **Completed Processes**, **Active Processes**, **Aborted Processes**, **Suspended Processes**, **Execution Time per second**, **Created per second**, **Aborted per second**, **Suspended per second**, **Most Recent Execution Time**, **Average Execution Time**, **Most Recent Elapsed Time**, and **Average Elapsed Time**.

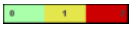
The heatmap is organized by host with each rectangle representing a process. Move your mouse over a node to display current metrics. Click on a node to drill-down to the TIBCO BusinessWorks 5 Process Summary - HTML display to view specific metrics about process behavior over a specified period of time and determine which activity in the process may be causing the bottleneck.



Filter By:

The display might include these filtering options:

Server:	Select the server for which you want to view data in the display.
Engine:	Select the engine for which you want to view data in the display.
Processes	The total number of processes in the display.
Running	Number of processes currently running.
Show Engine	Select this check box to display the names of the engines above their respective rectangles in the heatmap.
Log Scale	Select to enable a logarithmic scale. Use **Log Scale** to see usage correlations for data with a wide range of values. For example, if a minority of your data is on a scale of tens, and a majority of your data is on a scale of thousands, the minority of

	your data is typically not visible in non-log scale graphs. **Log Scale** makes data on both scales visible by applying logarithmic values rather than actual values to the data.
Auto Scale	Select to enable auto-scaling. When auto-scaling is activated, the color gradient bar's maximum range displays the highest value. NOTE: Some metrics auto-scale automatically, even when **Auto Scale** is not selected.
Metric	Select the metric driving the heatmap display. The default is Alert Severity. Each Metric has a color gradient bar that maps values to colors. The heatmap organizes the servers by host, where each rectangle represents a process. Mouse-over any rectangle to display the current values of the metrics for the process. Click on a rectangle to drill-down to the associated TIBCO BusinessWorks 5 Process Summary - HTML display for a detailed view of metrics for that particular process.
Alert Severity	The maximum level of alerts in the heatmap rectangle. Values range from **0** - **2**, as indicated in the color gradient  bar, where **2** is the highest Alert Severity: Red indicates that one or more metrics exceeded their ALARM LEVEL threshold. Yellow indicates that one or more metrics exceeded their WARNING LEVEL threshold. Green indicates that no metrics have exceeded their alert thresholds.
Alert Count	The total number of critical and warning alerts in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average alert count.
CPU Used %	The percent (%) CPU used in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the alert threshold of **BwProcessTotalCpuPercentHigh**, which is **100**. The middle value in the gradient bar indicates the middle value of the range (the default is **50**).
Completed	The total number of completed processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of completed processes in the heatmap. The middle value in the gradient bar indicates the middle value of the range.
Active	The total number of active processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of active processes in the heatmap. The middle value in the gradient bar indicates the middle value of the range.
Aborted	The total number of aborted processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of aborted processes in the heatmap. The middle value in the gradient bar indicates the middle value of the range.
Suspended	The total number of suspended processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of suspended processes in the heatmap. The middle value in the gradient bar indicates the middle value of the range.

Exec Time/s	The number of processes executed per second in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum execution rate of processes in the heatmap. The middle value in the gradient bar indicates the middle value of the range.
Created/s	The number of processes created per second in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum created rate of processes in the heatmap. The middle value in the gradient bar indicates the middle value of the range.
Aborted/s	The number of aborted processes per second in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum aborted rate of processes in the heatmap. The middle value in the gradient bar indicates the middle value of the range.
Suspended /s	The number of suspended processes per second in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum suspended rate of processes in the heatmap. The middle value in the gradient bar indicates the middle value of the range.
Most Recent Exec Time	The execution time for the most recently executed process in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the most recent execution time of processes in the heatmap. The middle value in the gradient bar indicates the middle value of the range.
Average Exec Time	The average execution time for all processes in the heatmap rectangle, calculated by dividing the delta execution time for the interval by the delta completed, or the number of process instances that completed in the interval. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the average execution time of processes in the heatmap. The middle value in the gradient bar indicates the middle value of the range.
Most Recent Elapsed Time	The elapsed time for the most recently executed process in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the most recent elapased time of processes in the heatmap. The middle value in the gradient bar indicates the middle value of the range.
Average Elapsed Time	The average elapsed time for all processes in the heatmap rectangle, calculated by dividing the delta elapsed time for the interval by the delta completed, or the number of process instances that completed in the interval. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the average elapsed time of processes in the heatmap. The middle value in the gradient bar indicates the middle value of the range.

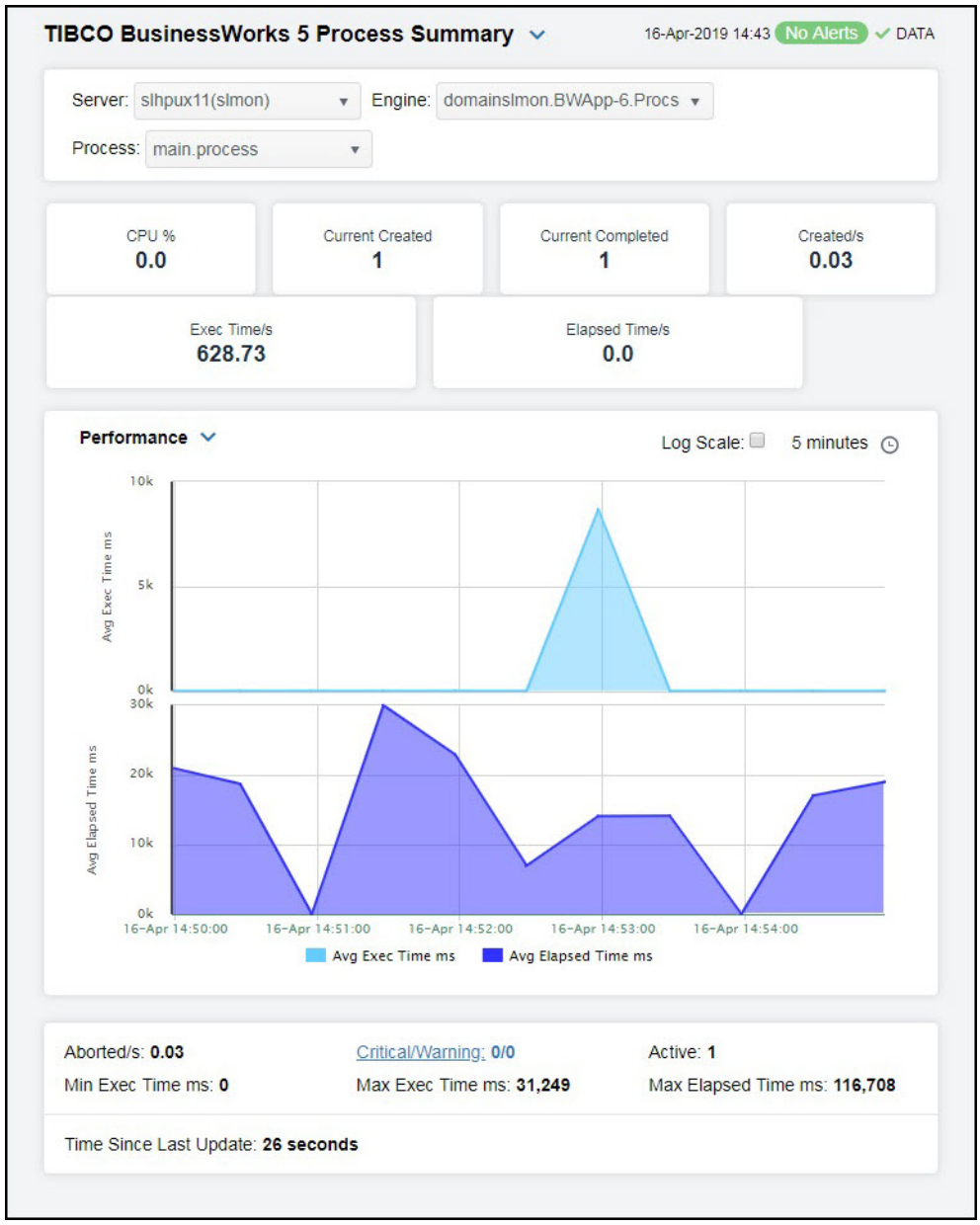
TIBCO BusinessWorks 5 Process Summary - HTML

Clicking **BW5 Process Summary** in the left/navigation menu opens the **TIBCO BusinessWorks 5 Process Summary** display, which allows you to track utilization and performance metrics for specific BW5 processes. You can select a server, engine, and process from the drop-down menus. Clicking on the information boxes at the top of the display takes

you to the TIBCO BusinessWorks 5 Processes Table - HTML display, where you can view additional data on processes.

There are two options in the trend graph: **Utilization** and **Performance**. In the **Utilization** option on the trend graph, you can view trend data for the rate of created processes and CPU percentage over a selected time range. In the **Performance** option on the trend graph, you can view trend data for used average execution time and average elapsed time over a selected time range.

Clicking the **Critical/Warning** link at the bottom of the display opens the Alerts Table by Component display.



Filter By:

The display might include these filtering options:

Server: Select the server for which you want to view data.

Engine: Select the engine for which you want to view data.

Fields and Data

CPU % The percentage of CPU used by the process.

Current Created The number of processes created since the last data update.

Current Completed The number of completed processes since the last data update.

Created/s The rate of processes created, per second.

Exec Time/s The rate of process execution time, per second.

Elapsed Time/s The rate of elapsed time, per second, for the process.

Trend Graphs

Utilization

CPU % -- Traces the CPU utilization percentage.

Created Procs/s -- Traces the number of created processes per second.

Performance

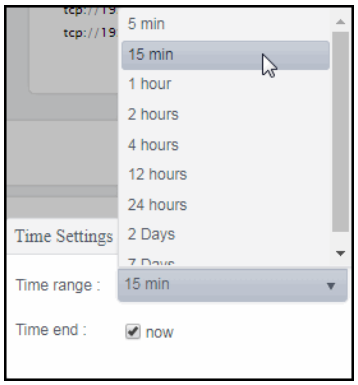
Avg Exec Time ms -- Traces the average time taken to execute processes, in milliseconds.

Avg Elapsed Time ms -- Traces the average elapsed time for processes, in milliseconds.

Log Scale Select to enable a logarithmic scale. Use **Log Scale** to see usage correlations for data with a wide range of values. For example, if a minority of your data is on a scale of tens, and a majority of your data is on a scale of thousands, the minority of your data is typically not visible in non-log scale graphs. **Log Scale** makes data on both scales visible by applying logarithmic values rather than actual values to the data.

Time Settings

Select a time range from the drop down menu varying from **5 Minutes** to **Last 7 Days**. By default, the time range end point is the current time.



To change the time range, deselect the **now** toggle, which displays some additional date fields. You can click the left and right arrow buttons to decrease the end time by one time period (the time selected in the **Time range** drop down) per click, or you can choose the date and time from the associated calendar and clock icons. You can also enter the date and time in the text field using the following

format: **MMM dd, YYYY HH:MM:ss**. For example, Aug 21, 2018 12:24 PM. Click the **now** toggle to reset the time range end point to the current time.

Abprted/s	The rate of aborted processes.
Min Exec Time ms	Shortest execution time of any process instance, in milliseconds.
Critical/Warning	The number of critical and warning alerts.
Max Exec Time ms	Longest execution time of any process instance, in milliseconds.
Active	The number of active processes.
Max Elapsed Time ms	Longest elapsed time of any process instance, in milliseconds.
Time Since Last Update	Time since the last update to file of statistics.

BW5 Activities - HTML

These displays present performance metrics and alert status for all BW5 activities. Clicking **BW5 Activities** from the left/navigation menu opens the TIBCO BusinessWorks 5 Activities Table - HTML display, where each row in the table displays all available metrics from the Hawk microagent for an activity. The options available under **BW5 Activities** are:

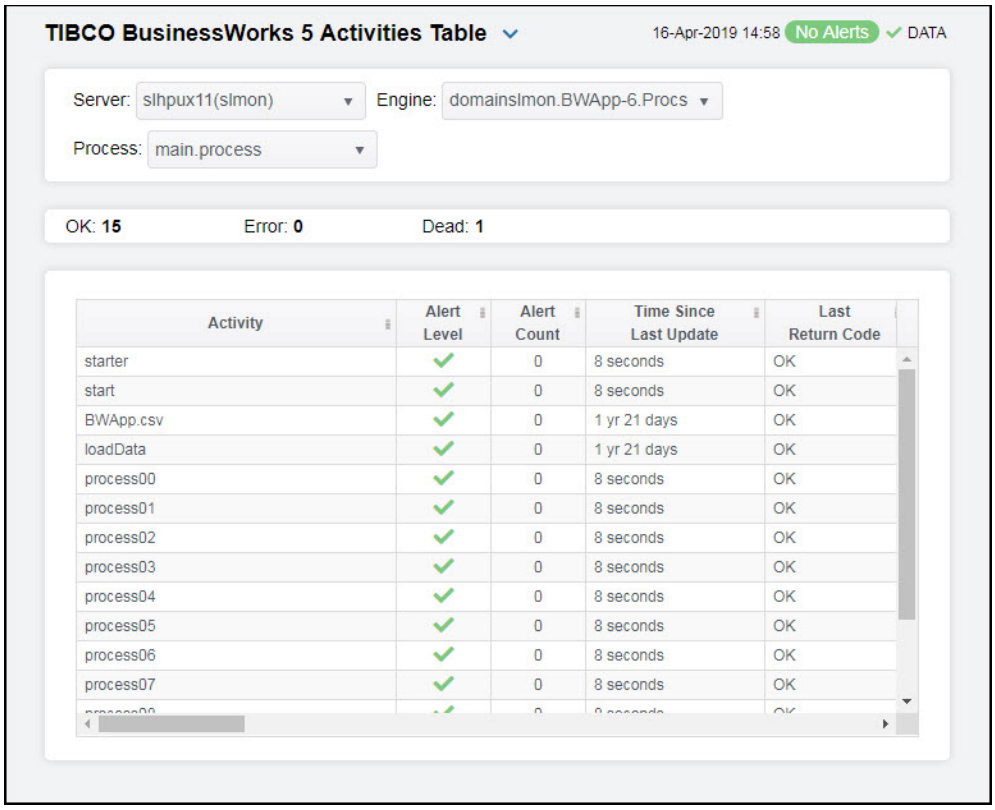
- **All Activities Heatmap**: Opens the TIBCO BusinessWorks 5 Activities Heatmap - HTML, which shows process execution metrics for all activities.
- **BW5 Activity Summary**: Opens the TIBCO BusinessWorks 5 Activity Summary - HTML display, which shows historical and current performance metrics for a single activity, including average execution times and execution counts.

TIBCO BusinessWorks 5 Activities Table - HTML

Select a server, engine, and process from the drop-down menus to see activities for the selected combination. Each row in the table is a different activity. Each table row displays all metrics available from the Hawk microagent for an activity. (Refer to documentation for TIBCO BusinessWorks Administration, see Appendix A: TIBCO Hawk Microagent Methods).

Click a column header to sort column data in ascending or descending order. Double-click on a table row to drill-down to the TIBCO BusinessWorks 5 Activity Summary - HTML display and view metrics for that particular activity. Toggle between the commonly accessed **Table** and **Heatmap** displays by clicking the drop down list on the display title.

An EXPIRED activity and the associated engine are deleted from displays when the associated server exceeds its specified threshold. Processes associated with the engine are also deleted from displays.



Filter By:

The display might include these filtering options:

Server:	Select the server for which you want to view data in the display, or select **All** to view data for all servers.
Engine:	Select the engine for which you want to view data in the display, or select **All** to view data for all engines.
Process:	Select the process for which you want to view data in the display, or select **All** to view data for all processes.

Fields and Data

OK	Number of processes with a status code of OK.
Error	Number of processes with a status code of Error.
Dead	Number of processes with a status code of Dead.

Table:

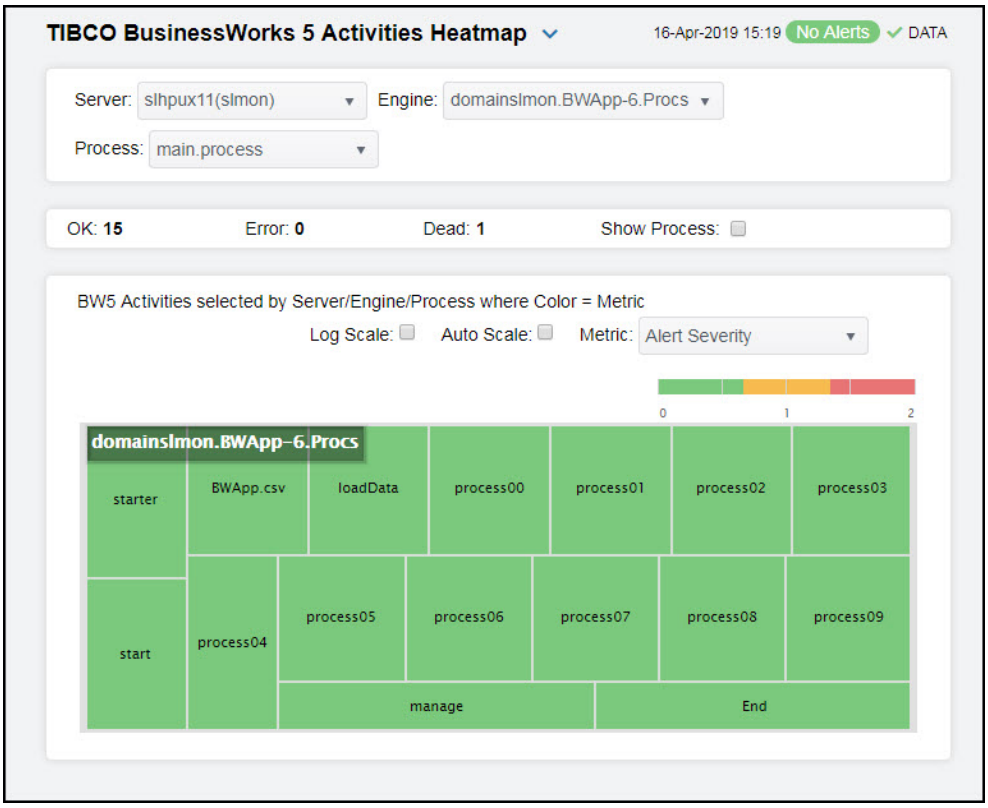
Activity	Name of activity.
Alert Level	The most critical alert state for alerts in the row: Red indicates that one or more metrics exceeded their ALARM LEVEL threshold. Yellow indicates that one or more metrics exceeded their WARNING LEVEL threshold. Green indicates that no metrics have exceeded their alert thresholds.
Alert Count	The total number of alerts for the activity.
Time Since Last Update	Time since the last update.
Last Return Code	Status code (OK, DEAD, or ERROR) returned by most recent execution of this activity.

Exec Time ms	Time (in milliseconds) used by all executions of this activity. NOTE: This does not include wait time for Sleep, Call Process, and Wait For... activities.
Current Exec Time ms	Execution time (in milliseconds) accumulated during this update cycle.
Exec Time/s	Execution time accumulated per second.
Min Exec Time ms	Time (in milliseconds) of the activity that has the shortest execution time.
Max Exec Time ms	Time (in milliseconds) of the activity that has the longest execution time.
Elapsed Time ms	Elapsed clock time (in milliseconds) used by all executions of this activity. NOTE: This does not include wait time for Sleep, Call Process, and Wait For... activities.
Current Elapsed Time ms	Change in ElapsedTime this update.
Elapsed Time/s	Change in ElapsedTime per second.
Min Elapsed Time ms	Elapsed clock time (in milliseconds) of the activity that has the shortest execution time.
Max Elapsed Time ms	Elapsed clock time (in milliseconds) of the activity that has the longest execution time.
Executions	Number of times the activity has been executed.
Executions/s	Change in ExecutionCount per second.
Current Executions	Change in ExecutionCount this update.
Errors	Total number of executions of the activity that have returned an error.
Error %	Number of errors relative to the total executions of the activity.
Errors/s	Change in ErrorCount per second.
Current Errors	Change in ErrorCount this update.
Domain	Name of TIBCO Domain.
Activity Class	Name of the class that implements the activity.
Called Process Defs	A comma-separated list of definitions called by this activity.
Tracing	• **true** Tracing is enabled for this activity. • **false** Tracing is disabled for this activity.
MicroAgentInstance	Unique ID of the microagent reporting the metrics.
Executions Since Reset	Number of times the activity has been executed since the last reset of the statistics.
Source	Name of RTView Data Server sending this data (or localhost).
Time Stamp	Time of this update.

TIBCO BusinessWorks 5 Activities Heatmap - HTML

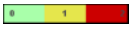
Clicking **All Activities Heatmap** in the left/navigation menu opens the **TIBCO BW5 Activities Heatmap**, which allows you to view the status and alerts of the execution times for all activities on all engines, or you can filter to look at specific servers, engines or processes. Use the **Metric** drop-down menu to view processes in the heatmap by the **Alert Severity**, **Alert Count**, **Executions**, **Errors**, **Execution Time per second**, **Error rate**, **Most Recent Execution Time**, and **Maximum Execution Time**.

The heatmap is organized by host with each rectangle representing an activity. Move your mouse over a node to display current metrics. Click on a node to drill-down to the TIBCO BusinessWorks 5 Activity Summary - HTML display to view specific metrics about process behavior over a specified period of time and determine which activity in the process may be causing the bottleneck.



Filter By:

The display might include these filtering options:

Server:	Select the server for which you want to view data in the display.
Engine:	Select the engine for which you want to view data in the display.
Process	Select from the menu to view activities running on a specific process or all processes.
OK	Number of activities that reported their Last Return Code as **OK**.
Error	Number of activities that reported their Last Return Code as **Error**.
Dead	Number of activities that reported their Last Return Code as **Dead**.
Show Process	Select this check box to display the names of the processes above their respective rectangles in the heatmap.
Log Scale	Select to enable a logarithmic scale. Use **Log Scale** to see usage correlations for data with a wide range of values. For example, if a minority of your data is on a scale of tens, and a majority of your data is on a scale of thousands, the minority of your data is typically not visible in non-log scale graphs. **Log Scale** makes data on both scales visible by applying logarithmic values rather than actual values to the data.
Auto Scale	Select to enable auto-scaling. When auto-scaling is activated, the color gradient bar's maximum range displays the highest value. NOTE: Some metrics auto-scale automatically, even when **Auto Scale** is not selected.
Metric	Select the metric driving the heatmap display. The default is **Alert Severity**. Each **Metric** has a color gradient bar that maps values to colors. The heatmap organizes the servers by host, where each rectangle represents an activity. Mouse-over any rectangle to display the current values of the metrics for the activity. Click on a rectangle to drill-down to the associated TIBCO BusinessWorks 5 Activity Summary - HTML display for a detailed view of metrics for that particular activity.
Alert Severity	The maximum level of alerts in the heatmap rectangle. Values range from **0** - **2**, as indicated in the color gradient 

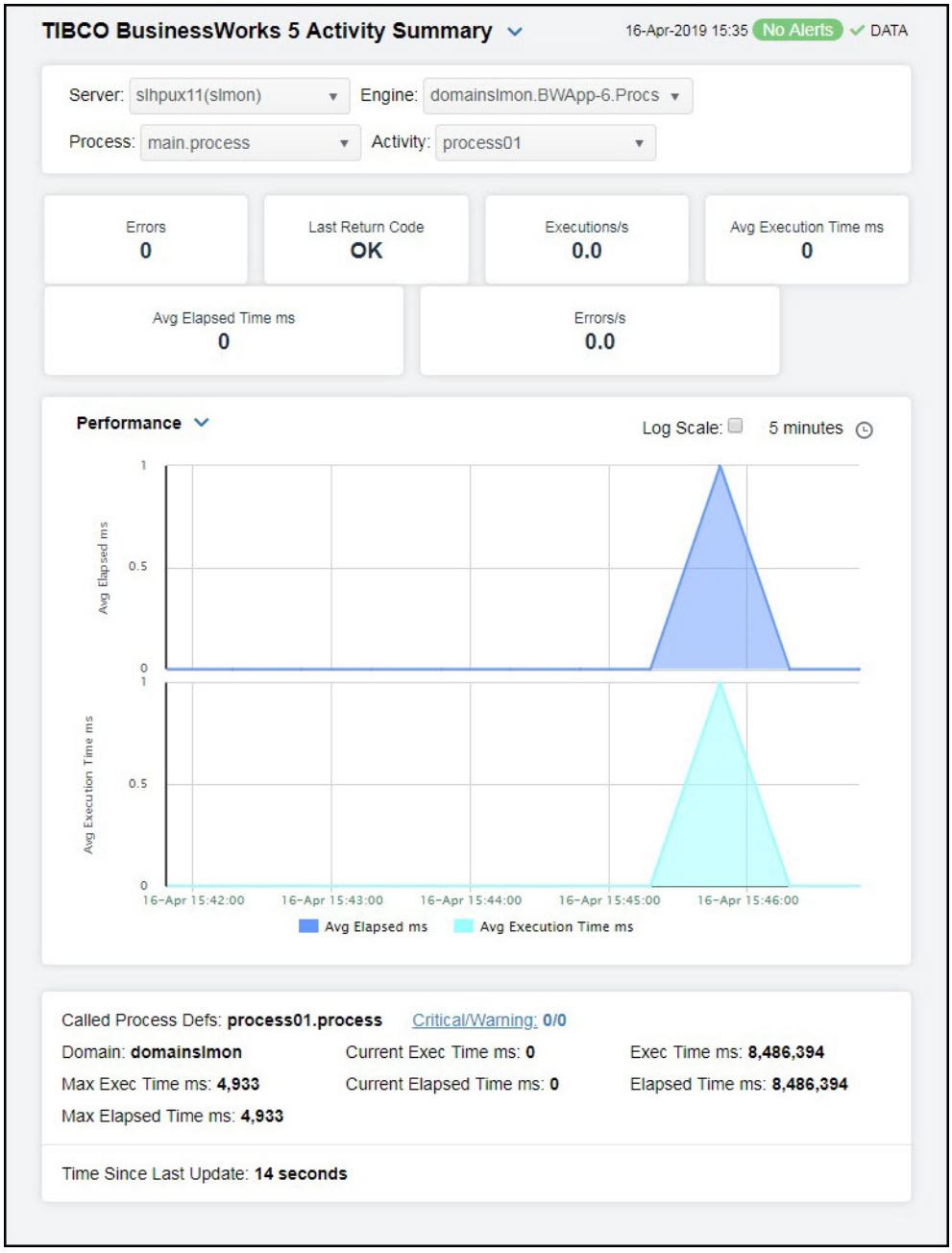
	bar, where **2** is the highest Alert Severity: ● Red indicates that one or more metrics exceeded their ALARM LEVEL threshold. ● Yellow indicates that one or more metrics exceeded their WARNING LEVEL threshold. ● Green indicates that no metrics have exceeded their alert thresholds.
Alert Count	The total number of critical and warning alerts in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average alert count.
Executions	The total number of executed processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the number of executions in the heatmap. The middle value in the gradient bar indicates the middle value of the range.
Errors	The total number of errors in the heatmap rectangle. The color gradient  bar populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the number of errors in the heatmap. The middle value in the gradient bar indicates the middle value of the range.
Exec Time/sec	The number of processes executed per second in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the rate of executions in the heatmap. The middle value in the gradient bar indicates the middle value of the range.
Errors/sec	The number of errors per second in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the rate of errors in the heatmap. The middle value in the gradient bar indicates the middle value of the range.
Most Recent Exec Time	The execution time for the most recently executed process in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the most recent execution time in the heatmap. The middle value in the gradient bar indicates the middle value of the range.
Max Exec Time	The maximum execution time for executed processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum execution time in the heatmap. The middle value in the gradient bar indicates the middle value of the range.

TIBCO BusinessWorks 5 Activity Summary - HTML

Clicking **BW5 Activity Summary** in the left/navigation menu opens the **TIBCO BusinessWorks 5 Activity Summary** display, which allows you to track utilization and performance metrics for specific BW5 activities. You can select a server, engine, process, and activity from the drop-down menus. Clicking on the information boxes at the top of the display takes you to the TIBCO BusinessWorks 5 Activities Table - HTML display, where you can view additional data on activities.

There are two options in the trend graph: **Performance** and **Success Rate and Average Failures**. In the **Performance** option on the trend graph, you can view trend data for the average elapsed time and average execution time over a selected time range. In the **Success Rate and Average Failures** option on the trend graph, you can view trend data for used execution rate and average errors over a selected time range.

Clicking the **Critical/Warning** link at the bottom of the display opens the Alerts Table by Component display.



Filter By:

The display might include these filtering options:

Server:	Select from the menu to view processes running on a specific server.
Engine:	Select from the menu to view processes running on a specific engine.
Process:	Select from the menu to view summary details for a specific process.
Activity	Select from the menu to view summary details for a specific activity.

Fields and Data

Errors	Total number of errors generated by the activity.
Last Return Code	The return code of the last execution of this activity.
Executions/s	The rate of executions, per second, for this activity.
Avg Execution Time ms	The average execution time, in milliseconds, for this activity.
Avg Elapsed Time ms	The average elapsed time, in milliseconds, for this activity.
Errors/s	The rate of errors, per second, for this activity.

Trend Graphs

Performance

Avg Elapsed Time ms -- Traces the average elapsed time, in milliseconds, for this activity.

Avg Execution Time ms -- Traces the average execution time, in milliseconds, for this activity.

Success Rate and Avg Failures

Executions/s -- Traces the rate of executions, per second, for this activity.

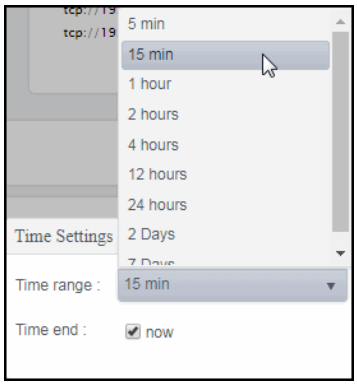
Avg Errors -- Traces the average number of errors generated by the activity.

Log Scale

Select to enable a logarithmic scale. Use **Log Scale** to see usage correlations for data with a wide range of values. For example, if a minority of your data is on a scale of tens, and a majority of your data is on a scale of thousands, the minority of your data is typically not visible in non-log scale graphs. **Log Scale** makes data on both scales visible by applying logarithmic values rather than actual values to the data.

Time Settings

Select a time range from the drop down menu varying from **5 Minutes** to **Last 7 Days**. By default, the time range end point is the current time.



To change the time range, deselect the **now** toggle, which displays some additional date fields. You can click the left and right arrow buttons to decrease the end time by one time period (the time selected in the **Time range** drop down) per click, or you can choose

	the date and time from the associated calendar and clock icons. You can also enter the date and time in the text field using the following format: **MMM dd, YYYY HH:MM:ss**. For example, Aug 21, 2018 12:24 PM. Click the **now** toggle to reset the time range end point to the current time.
Called Process Defs	The number of call process definitions for the activity.
Current Exec Time ms	The time spent, in milliseconds, in execution since the last data update.
Current Elapsed Time ms	The elapsed time, in milliseconds, accumulated since the last data update.
Critical/Warning	The number of critical and warning alerts.
Exec Time ms	The shortest amount of time needed to execute a process.
Elapsed Time ms	The shortest amount of elapsed time.
Domain	The name of the domain.
Max Exec Time ms	The longest amount of time needed to execute a process.
Max Elapsed Time ms	The longest amount of elapsed time.
Time Since Last Update	The amount of time since the last update.

Drilldowns

Displays contained under this View are only available by clicking on buttons/links in other displays. For example, clicking on the "Alerts" icon  in the upper right hand corner of the display opens the **Alerts Table by Component** display. You cannot access these displays directly via the left menu.

- **Component Alerts Table**: Associated with the Alerts Table by Component display, which can be accessed by clicking the Alerts icon  in the upper right hand corner of the display.
- **Component Alert Detail**: Associated with the Alert Detail for Component display, which can be accessed by clicking in the **Alerts Table** or by clicking  in the **Alerts Table by Component** display.

Alerts Table by Component

As an alternative to the **Alerts Table**, use the **Alerts Table by Component** to track and manage all alerts that are specifically associated with the CIs shown in a display.

You access the **Alerts Table by Component** by clicking  (the alert status icon) in the title bar of other displays. The display in which you click  is the source display.

Package provides the technology label associated with the alerts shown. For example, **Jvm**, **Tomcat** and **Host** are the technology labels for Java Virtual Machines, Tomcat applications and servers (respectively). These labels are also correlated with the RTView solution package

names (for example, the Solution Package for Host Agent). **Category** lists all alert categories related to the source display.

Use the **ACK** and **Cleared** drop-downs to filter the table by **All**, **True** or **False**.

See the **Alert Level** column icon, where:

 The alert reached its ALARM LEVEL threshold in the table row.

 The alert reached its WARNING LEVEL threshold in the table row.

To investigate, click:

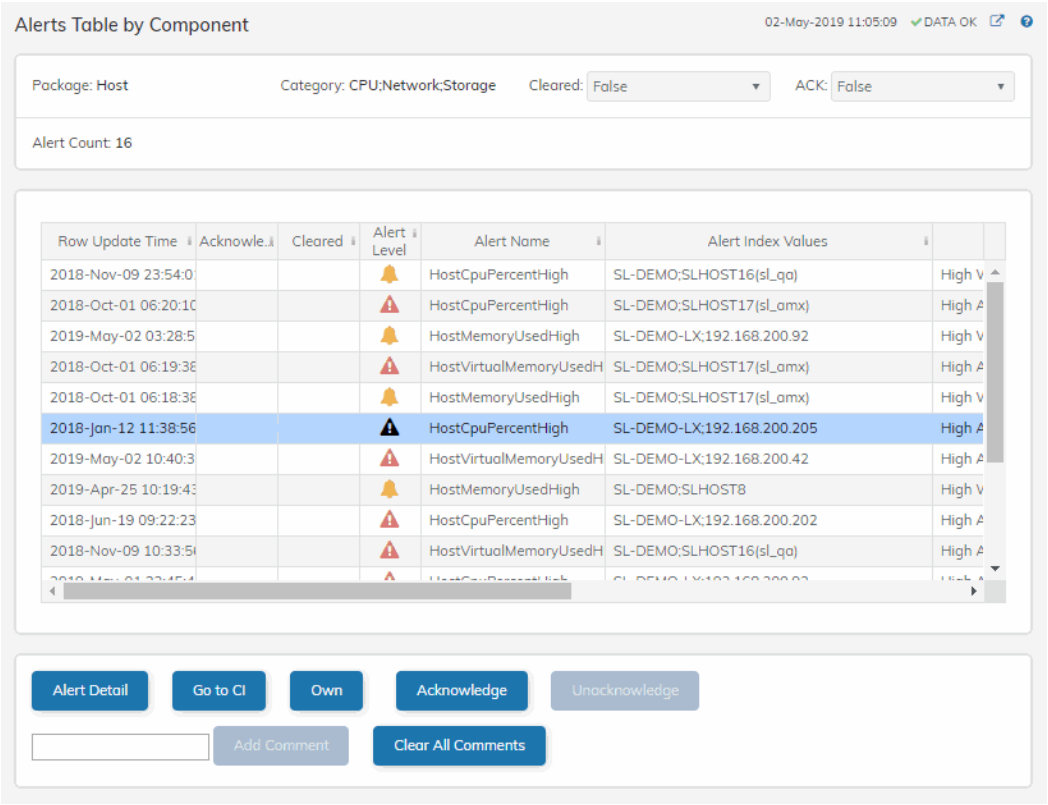
Alert Detail to open the **Alert Detail for Component** where you can see the current and historical conditions that precipitated the alert being executed.

Go to CI to open the summary display for the CI associated with the alert where you can investigate utilization metrics for the CI leading up to the alert being executed.

You can search, filter, sort and choose columns to include by clicking a column header icon (to the right of each column label) and selecting **Filter**, **Sort Ascending**, **Sort Descending** or **Columns**. Right-click on a table cell to **Export to Excel**. Use **Ctrl +** click or **Shift** + click to select multiple alerts.

With one or more alerts selected, click Own to set the alert(s) owner field, Acknowledge to acknowledge the alert(s), Unacknowledge to clear the acknowledgement on previously acknowledged alert(s), Add Comment to add a comment to the alert(s).

You must be logged in as rtvalertmgr or rtvadmin to perform the **Own**, **Ack**, **Unack**, or **Comment** actions. Otherwise, you get an error dialog.



Alert Detail for Component

Use the **Alert Detail for Component** display to investigate current and historical activity of a specific alert instance as it applies to the associated CI, and also compare against **Metric History** trends of the associated CI. A trend graph for the CI associated with the alert instance. You can hover over the trend graph to see the values at a particular time. You can specify the time range for the trend graph and view data based on a log scale, which enables visualization on a logarithmic scale and should be used when the range in your data is very broad.

Access the **Alert Detail for Component** display by clicking Details in the **Alerts Table** or Alert Detail in the **Alerts Table by Component** display.

The **Alert History** table at the bottom of the display contains a row of data for each time the alert instance was updated. See the alert **ID**, **Row Update Time**, **Cleared** status and **Reason**, **Owner** and the **Alert Level** column icon, where:

The alert reached its ALARM LEVEL threshold in the table row.

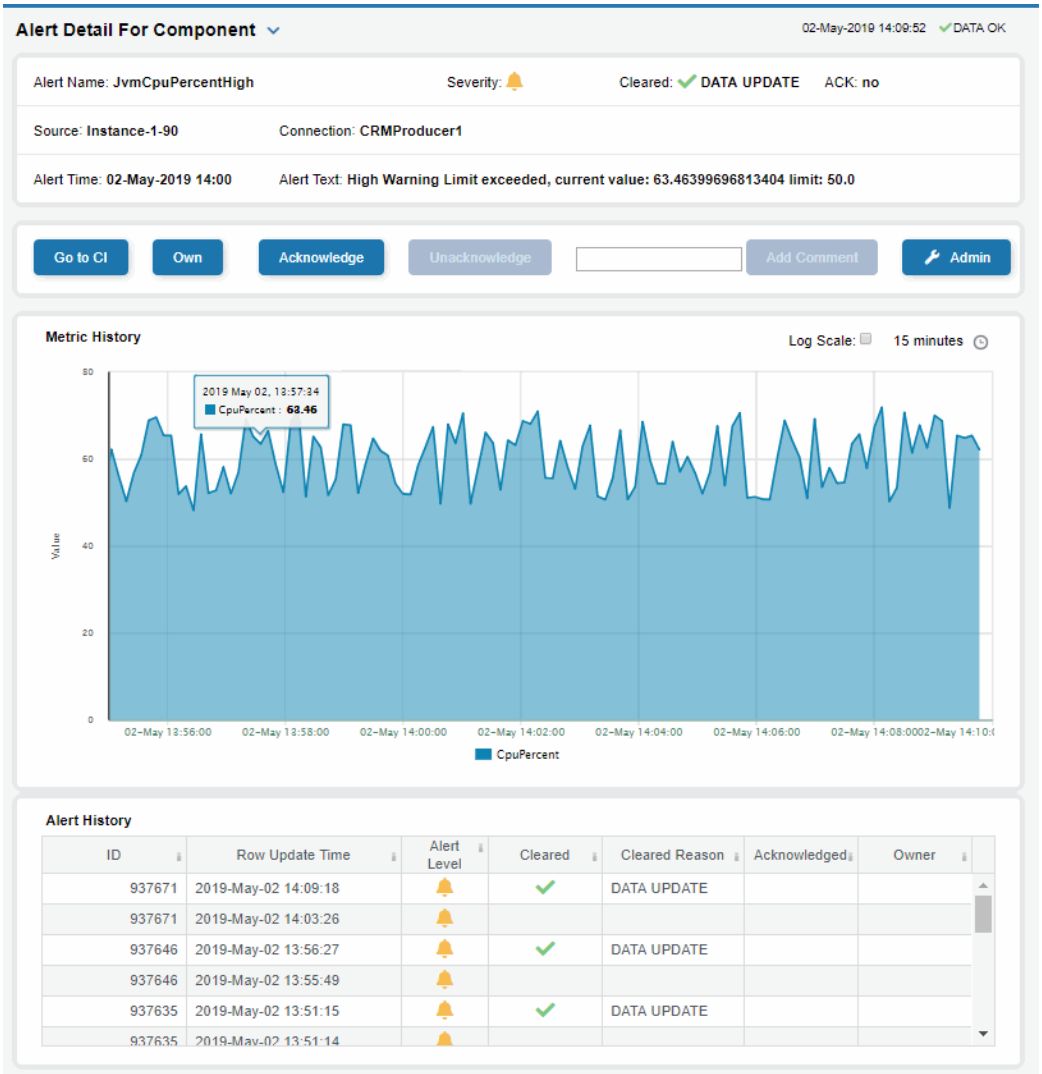
The alert reached its WARNING LEVEL threshold in the table row.

You can search, filter, sort and choose columns to include by clicking a column header icon (to the right of each column label) and selecting **Filter**, **Sort Ascending**, **Sort Descending** or **Columns**. Right-click on a table cell to **Export to Excel**. Use **Ctrl +** click or **Shift** + click to select multiple alerts.

To investigate, click:

Go to CI to see utilization conditions for the CI associated with the alert in a summary display.

Admin to open the **Alert Configuration for Component** display where you can see, modify and refine alert threshold settings for that particular alert. A trend graph traces the relevant alert metric for the CI so you can adjust thresholds in real-time.



Alerts

This display presents detailed information about all alerts that have occurred in your system. These displays present performance data for your BusinessWorks system. Displays in this

View are:

- Alerts Table

Alerts Table

Use this display to track and manage all alerts that have occurred in the system, where:

 One or more alerts exceeded their ALARM LEVEL threshold in the table row

 One or more alerts exceeded their WARNING LEVEL threshold in the table row

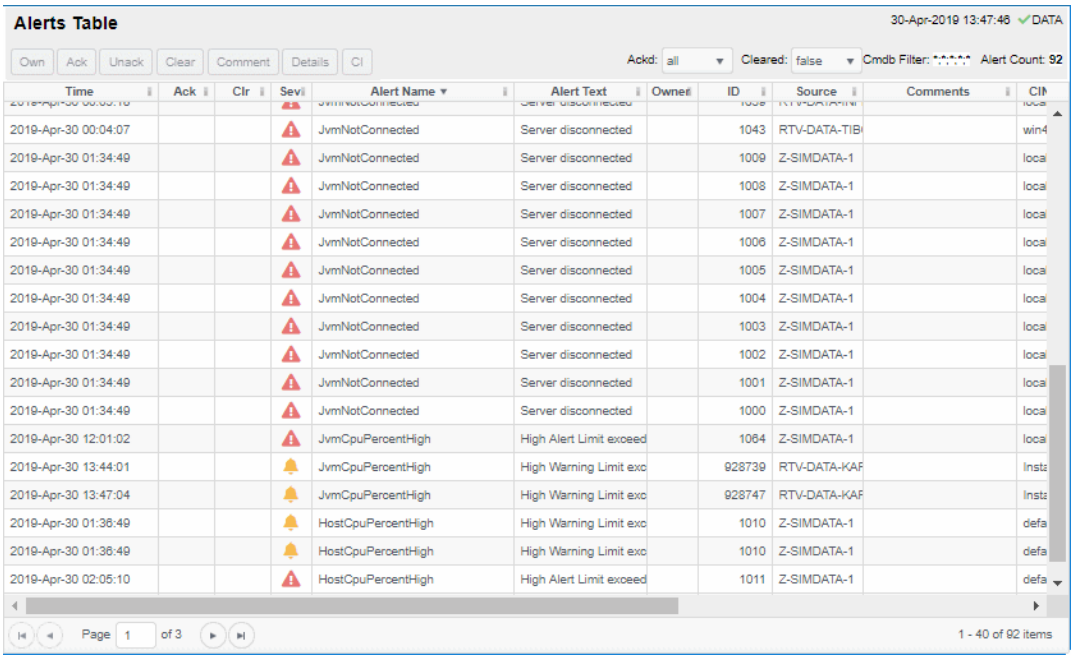
You can search, filter, sort and choose columns to include by clicking a column header icon (located to the right of each column label) and selecting **Filter**, **Sort Ascending**, **Sort Descending** or **Columns**. Use the **Ack'd** and **Cleared** drop-downs to filter the table by those columns. Right-click on a table cell to **Export to Excel** or **Copy Cell Value**. Use **Ctrl +** click or **Shift** + arrow to select multiple alerts. To investigate, select one alert and click:

 to open the **Component Alert Detail** display to get details about that particular alert instance as it specifically applies to the associated CI.

 to see utilization conditions for the CI associated with the alert during the seconds (minutes, hours or days) leading up to the alert being executed in a summary display.

With one or more alerts selected, you can click **Own** to set the alert(s) owner field, **Ack** to acknowledge the alert(s), **Unack** to clear the acknowledgement on previously acknowledged alert(s), **Clear** to set the **Cleared** flag on the selected alert(s), **Comment** to add a comment to the alert(s) and **CI** to get details about the CI associated with the alert (these buttons are enabled when you click one or more alerts).

You must be logged in as rtvalertmgr or rtvadmin to perform the **Own**, **Ack**, **Unack**, or **Comment** actions. Otherwise, you get an error dialog.



Admin

These displays enable you to set alert thresholds, observe how alerts are managed, and view internal data gathered and stored by RTView (used for troubleshooting with SL Technical Support). Displays in this View are:

- **Alert Administration**: Opens the Alerts Administration display, which displays active alerts and provides interface to modify, enable and manage alerts.
- **Alert Overrides Admin**: Associated with the Alert Overrides Administration display, which sets and modifies alert overrides. Access to this display is via the **Alert Administration** display. You cannot select this option via the left menu.
- **Alert Engine Status**: Opens the Alert Engine Admin display, which displays a table of available Data Servers and provides an interface to disable and enable the Alert Engine on a Data Server.
- **Cache Table**: Opens the Cache Table display, which allows you to view cached data that RTView is capturing and maintaining, and use this data use this for debugging with SL Technical Support.

Alerts Administration

The **Alert Administration** display allows administrators to enable/disable alerts and manage alert thresholds. The table describes the global settings for all alerts on the system.

You can set the **Delay** time (the number of seconds that must pass before an alert is triggered, where **0** sets it to immediately execute).

You can set the **Warning Level** which executes a single warning alert when the number of seconds specified here is exceeded. To set the warning to occur sooner, reduce the **Warning Level** value. To set the warning to occur later, increase the **Warning Level** value.

You can set the **Alarm Level** which executes a single alarm alert when the number of seconds specified here is exceeded. To set the alarm to occur sooner, reduce the **Alarm Level** value. To set the alarm to occur later, increase the **Alarm Level** value.

> **Note:** For low value-based alerts (an alert that executes based on a value going below a certain threshold), to set the alarm to occur sooner you increase the **Alarm Level** value. To set the alarm to occur later, reduce the **Alarm Level** value.

You can apply alert thresholds globally or as an *override*. Setting override alerts allows you to set thresholds for a subset of your resources, or for a single resource (for example, a single server). Override alerts are useful if the majority of your resources require the same threshold setting, but there are a few resources that require a different threshold setting. For example, you might not usually be concerned with execution time at a process level, but perhaps certain processes are critical. In this case, you can apply alert thresholds to each process individually. See below for instructions.

You can filter, sort and choose columns to include by clicking a column header icon (located to the right of each column label) and selecting **Filter**, **Sort Ascending**, **Sort Descending** or **Columns**. Use the **Ack'd** and **Cleared** drop-downs to filter the table by those columns. Right-click on a table cell to **Export to Excel**.

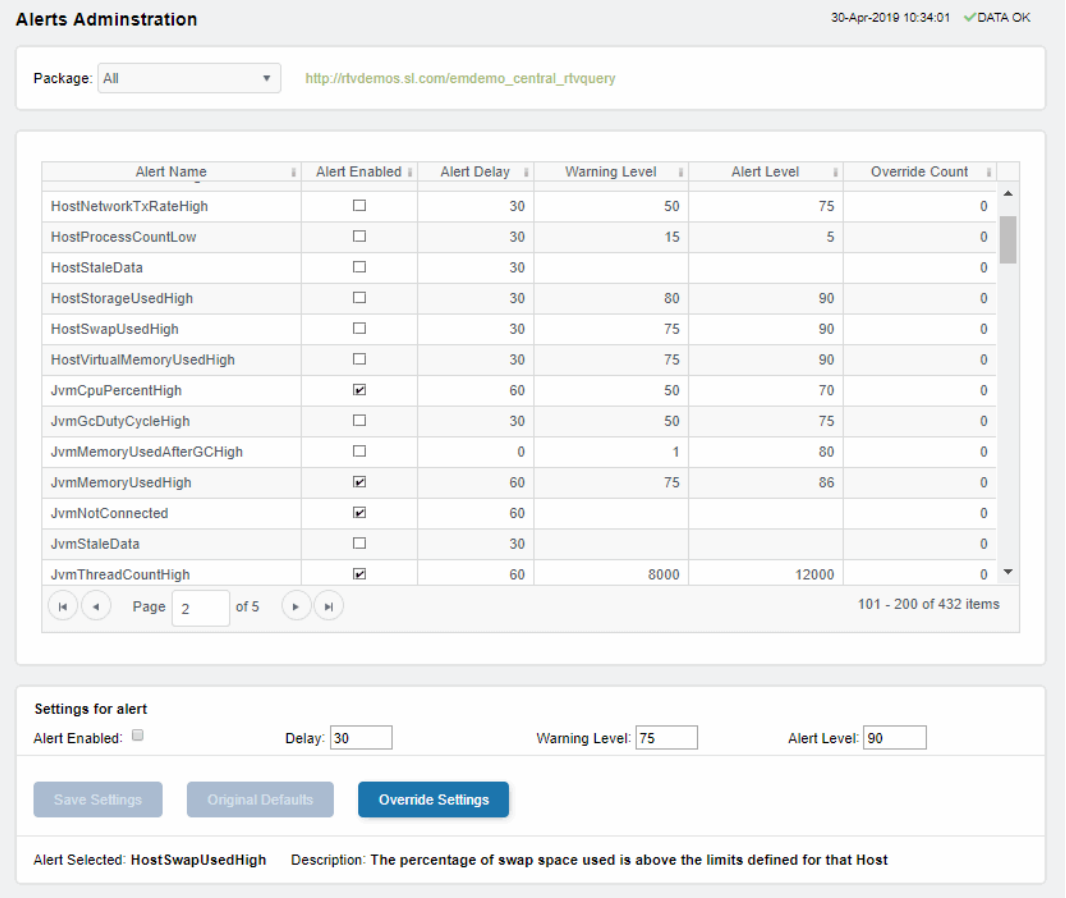
To set thresholds and enable a global alert:

Select an alert and, under **Settings for alert** (in the lower portion of the screen), modify settings for the alert **Delay**, **Warning Level** and/or **Alarm Level** and **Save Settings**. With that alert selected, check the **Alert Enabled** box under **Settings for alert** (in the lower portion of the screen) and **Save Settings**. The **Alert Enabled** box (next to the selected alert) is now checked.

You can also override the alert duration time per alert index instead of to all indexes. To override the duration for an alert index, select the alert in the **Alert Administration** display, click **Override** and edit the **Alert Delay**. For alert indexes that were overridden in a previous release (before duration override was supported) the override duration is set to **-1**, indicating that this is set to use the top level alert duration.

To set thresholds and enable an override alert:

To set an override alert, select an alert and click **Override Settings** to open the **Alert Overrides Admin** display.



Note: For more information on TIBCO RTView for TIBCO BusinessWorks alerts, see Alert Definitions

Alert Name	The name of the alert.
Alert Enabled	When checked, the alert is enabled globally.
Alert Delay	The amount of time (in seconds) that the value must be above the specified Warning Level or Alarm Level threshold before an alert is executed. **0** is for immediate execution.
Warning Level	The global warning threshold for the selected alert. When the specified value is exceeded a warning is executed.
Alert Level	The global alarm threshold for the selected alert. When the specified value is exceeded an alarm is executed.
Override Count	The number of times thresholds for this alert have been defined individually in the **Tabular Alert Administration** display. A value of: **-0** indicates that no overrides are applied to the alert. **-1** indicates that the alert does not support overrides.

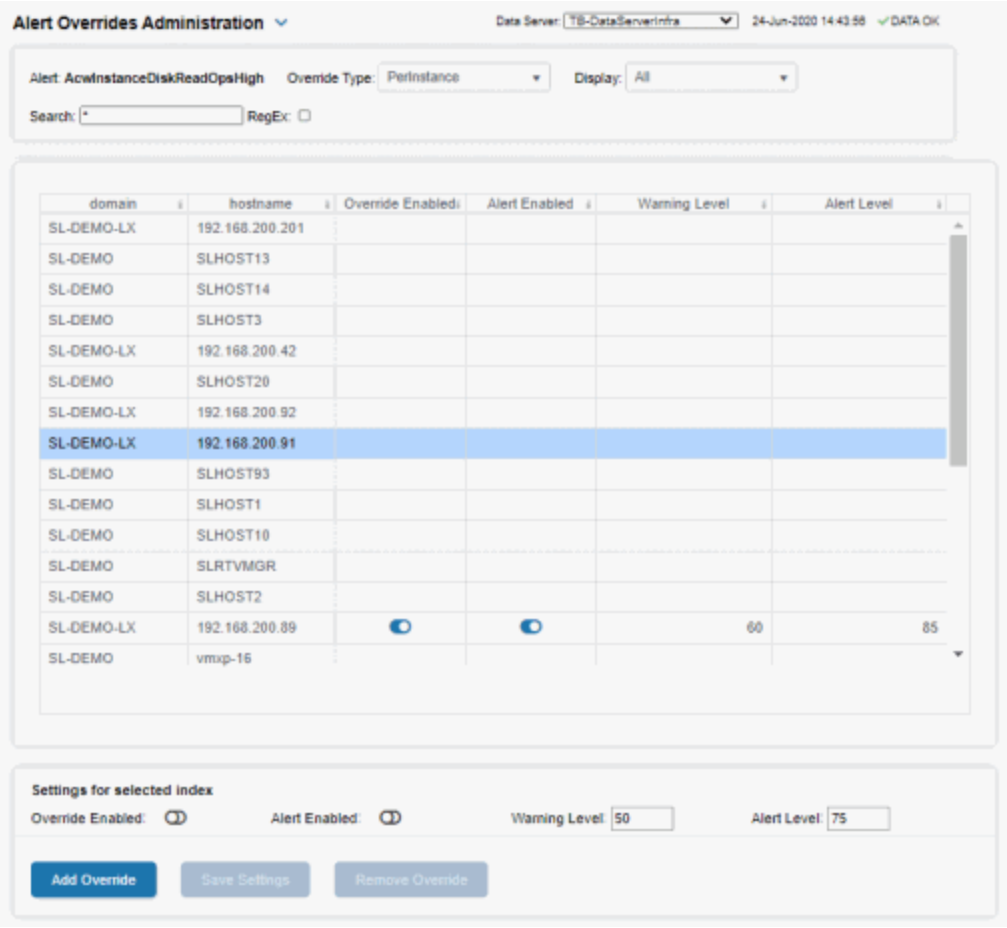
Settings for alert
Select an alert in the table to use the following options:

Alert Enabled	Check / uncheck this box to enable or disable the selected alert globally.
Delay	Enter the amount of time (in seconds) that the value must be above the specified Warning Level or Alarm Level threshold before the selected alert is executed. **0** is for immediate execution.
Warning Level	Enter the global warning threshold for the selected alert. When the specified value is exceeded a warning is executed. To set the warning to occur sooner, reduce the Warning Level value. To set the warning to occur later, increase the Warning Level value.
Alert Level	Enter the global alarm threshold for the selected alert. When the

	specified value is exceeded an alarm is executed. To set the alarm to occur sooner, reduce the Alarm Level value. To set the warning to occur later, increase the Alarm Level value. NOTE: For low value-based alerts (such as **EmsQueuesConsumerCountLow**), to set the alarm to occur sooner, increase the Alarm Level value. To set the alarm to occur later, reduce the Alarm Level value.
Save Settings	Click to apply alert settings for the selected alert.
Original Defaults	Click to revert to original alert settings for the selected alert.
Override Settings	Click to set an alert override in the **Alert Overrides Admin** display on the selected alert.

Alert Overrides Administration

Administrators use this display to override the alert settings defined in the **Alert Administration** display. To access this display, select an alert in the **Alert Administration** display and choose **Override Settings**.



The table lists all the possible overrides that can be defined for the alert you selected from the **Alert Administration** display. Each row in the table represents a different resource or group of resources that can be overridden. When the four last columns are blank, that means the resource has not been overridden, and the default settings for the alert apply. Otherwise, columns describe whether the alert

is enabled, if the override itself is enabled, the overridden alert thresholds and the overridden duration for each row.

Use the **Override Type** drop-down menu to switch the list to a specific type of override (the options for this menu vary according to the alert type), and use the **Display** drop-down menu to list **All** resources, **Overridden** resources or **Free** resources.

You can also enter a pattern or regular expression in the **Search** string to limit the list.

The **RegEx** checkbox indicates whether the text you entered is treated as a search pattern or as a regular expression. Multiple rows can be selected to create/edit/remove many overrides simultaneously.

You can filter, sort and choose columns to include by clicking a column header icon (located to the right of each column label) and selecting **Filter**, **SortAscending**, **Sort Descending** or **Columns**. Use the **Display** drop-down to filter the table to show **All** resources, only resources with the **Overridden** alert applied or **Free** resources (to show only resources without the alert override applied). Right-click on a table cell to **Export to Excel** or **Copy Cell Value**.

To set overrides:

Select an **Override Type** from the drop-down menu (depending on the alert, there might be only one type) and then select one or more rows from the table. Under **Settings for selected index** (in the lower portion of the screen), modify settings for the **Override Enabled**, **Alert Enabled**, **Alert Delay**, **Warning Level** and/or **Alarm Level**, then click **Add Override**. The table updates with your new settings.

To alter overrides:

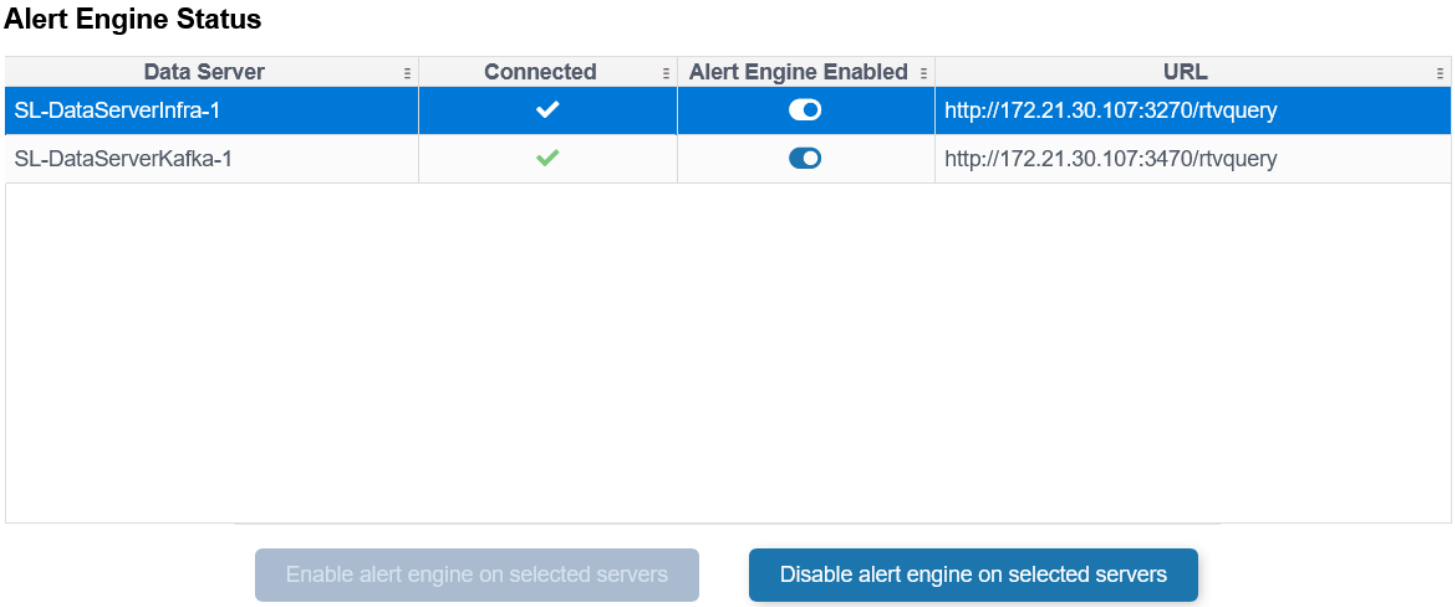
To alter existing overrides with new settings, select them from the table, set all properties under **Settings for selected index** as desired, then click **Save Settings**. To clear existing overrides, select one or more rows, then click **Remove Override**.

> **Note:** You can override alert and warning levels without overriding duration by setting it to **-1**.

For alert indexes that were overridden in a previous release (before duration override was supported) the override duration is set to **-1**, indicating that this is set to use the top level alert duration.

Alert Engine Admin

This display allows you to enable and disable the alert engine(s) of your Data Server(s) on a per-server basis. This display requires administrator privileges.



The **Alert Engine Status** table lists Data Servers that are connected to your deployment. The **Connected** column will display whether or not the Data Server is presently connected. If the Data Server is connected, the **Alert Engine Enabled** column will display whether the alert engine for that Data Server is enabled or not.

Disabling the alert engine on a Data Server clears all existing alerts on that server. In the Data Server's RtvAlertTable cache, the "Cleared Reason" column will show MANUAL for each alert that was cleared as a result of disabling the alert engine. No new alerts will be generated by that server until its alert engine is re-enabled..

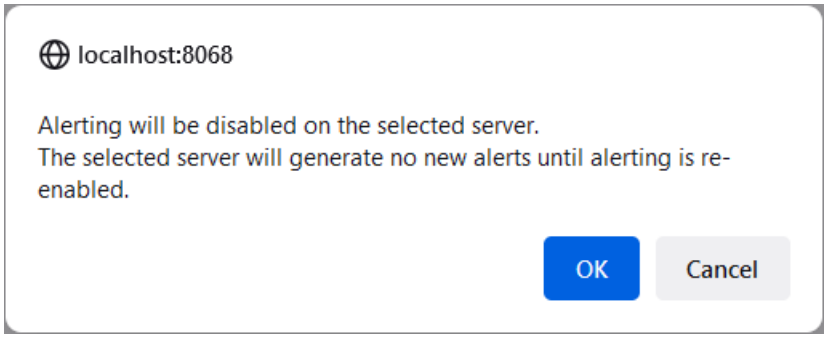
By default the alert engine is enabled for a Data Server. When a Data Server is restarted, its alert engine is always re-enabled.

Disable Alert Engine

Select one or more Data Servers in the **Alert Engine Status** table, then click 

.

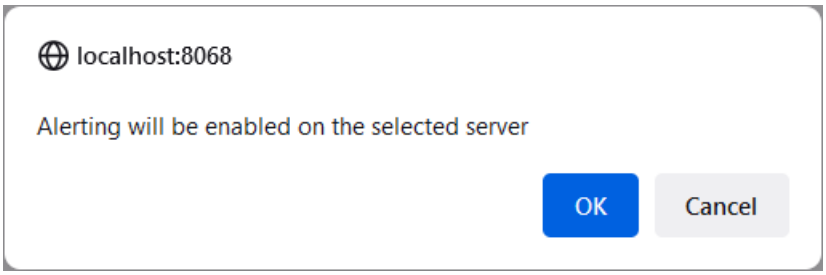
A confirmation dialog box will display. Click **OK** to continue and disable the selected alert engine(s).



Enable Alert Engine

Select one or more Data Servers in the **Alert Engine Status** table, then click Enable alert engine on selected servers.

A confirmation dialog box will display. Click **OK** to continue and enable the selected alert engine(s).



Warning! On Firefox, the enable/disable confirmation dialogs may display a checkbox with the text "Don't allow HOST:8068 to prompt you again". This is from the browser, not from RTView. Do not check that box, as it will prevent the display from working. If it is accidentally selected, clear the Firefox browser cache (**Options -> History -> Clear recent history ...**)

Cache Table

View the raw data that RTView is capturing and maintaining to investigate utilization and capacity metrics, as well as connection details, for caches on a data server.

Select a **Data Server** from the drop-down menu. The upper table contains a row of data for each cache on the selected data server. You can see the current number of **Rows** and **Columns** in each table and the amount of **Memory** used. You can also find out the cache **Table** type of which there are five:

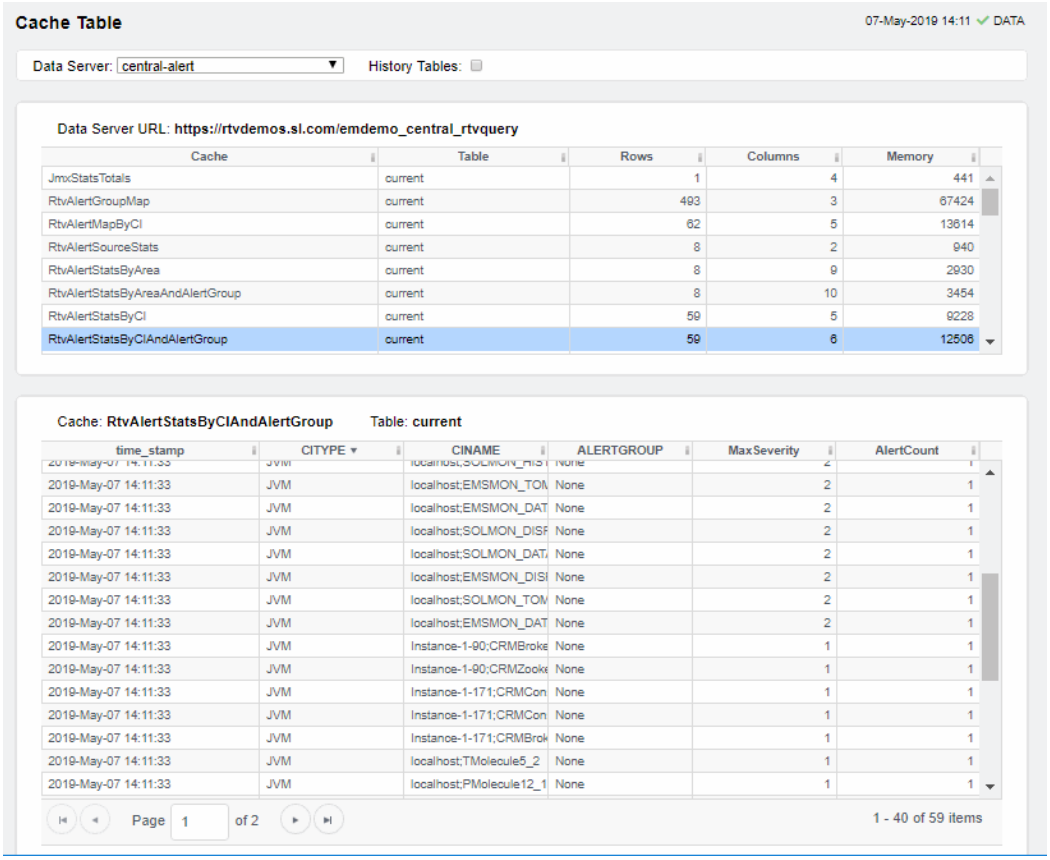
- **current** tables show the most recently received values for each index.
- **current_condensed** tables are current tables with primary compaction configured.
- **history** tables show the historical values for each index.
- **history_condensed** tables are history tables with primary compaction configured.
- **history_combo** tables are history tables with primary compaction configured, and which is also configured to store rows of recent raw data followed by rows of older condensed data.

Select a cache to see connection utilization details for that cache in the lower table. The lower table shows the contents of the selected cache table. Available columns vary by cache. For example, a JVM cache table might provide **BootClassPath** and **InputArgument** columns, and a Tomcat cache might provide **RateAccess** and **cacheMaxSize** columns.

You can search, filter, sort and choose columns to include by clicking a column header icon (to the right of each column label) and selecting **Filter**, **Sort Ascending**, **Sort Descending** or **Columns**. Or just click a column header to sort.

Right-click on a table cell to **Export to Excel** or **Copy Cell Value**. Use **Ctrl +** click or **Shift** + click to select multiple alerts. Use **History Tables** to include / exclude history tables in the table. Right-click on a table cell to **Export to Excel** or **Copy Cell Value**.

This low-level option can be useful to identify the source of the problem when the displays are not showing the expected data. Use this data for debugging and troubleshooting with Technical Support.


Cache Table
07-May-2019 14:11 DATA
Data Server: central-alert
History Tables:
Data Server URL: https://rtvdemos.sl.com/emdemo_central_rtvquery
Cache Table Rows Columns Memory
JmxStatsTotals current 1 4 441
RtvAlertGroupMap current 493 3 67424
RtvAlertMapByCI current 62 5 13614
RtvAlertSourceStats current 8 2 940
RtvAlertStatsByArea current 8 9 2930
RtvAlertStatsByAreaAndAlertGroup current 8 10 3454
RtvAlertStatsByCI current 59 5 9228
RtvAlertStatsByCIAndAlertGroup current 59 6 12506
Cache: RtvAlertStatsByCIAndAlertGroup Table: current
time_stamp CITYPE CINAME ALERTGROUP MaxSeverity AlertCount
2019-May-07 14:11:33 JVM localhost;EMSMON_TOM None 2 1
2019-May-07 14:11:33 JVM localhost;EMSMON_DAT None 2 1
2019-May-07 14:11:33 JVM localhost;SOLMON_DISF None 2 1
2019-May-07 14:11:33 JVM localhost;SOLMON_DAT/ None 2 1
2019-May-07 14:11:33 JVM localhost;EMSMON_DISI None 2 1
2019-May-07 14:11:33 JVM localhost;SOLMON_TOM None 2 1
2019-May-07 14:11:33 JVM localhost;EMSMON_DAT None 2 1
2019-May-07 14:11:33 JVM Instance-1-90;CRMBroke None 1 1
2019-May-07 14:11:33 JVM Instance-1-90;CRMZooke None 1 1
2019-May-07 14:11:33 JVM Instance-1-171;CRMCon None 1 1
2019-May-07 14:11:33 JVM Instance-1-171;CRMCon None 1 1
2019-May-07 14:11:33 JVM Instance-1-171;CRMBrok None 1 1
2019-May-07 14:11:33 JVM localhost;TMolecule5_2 None 1 1
2019-May-07 14:11:33 JVM localhost;PMolecule12_1 None 1 1
Page 1 of 2
1 - 40 of 59 items

CHAPTER 6 Using the Monitor

This section describes Monitor features, graphs and functionality as well as Monitor displays. This section includes:

- Overview: Describes the Monitor features and functionality.
- BW Applications: The displays in this View present BusinessWorks application performance metrics.
- BW Containers: The displays in this View present BusinessWorks container performance metrics.
- BW AppNodes: The displays in this View present BusinessWorks AppNode performance metrics.
- BW AppSlices: The displays in this View present BusinessWorks AppSlice performance metrics.
- BW Processes: The displays in this View present BusinessWorks process performance metrics.
- BW5 Engines: The displays in this View present BusinessWorks 5.0 engine performance metrics.
- BW5 Processes: The displays in this View present BusinessWorks 5.0 process performance metrics.
- BW5 Activities: The displays in this View present BusinessWorks 5.0 activity performance metrics.
- BW5 Servers: The displays in this View present BusinessWorks 5.0 server performance metrics.
- Hawk Views: Describes displays for TIBCO® BusinessWorks Hawk data.
- Alert Views: The display in this View presents the status of all alerts across all BW Servers, and allows you to track, manage and assign alerts.
- Administration: The displays in this View enable you to set global alerts and override alerts. You can also view internal data gathered and stored by RTView (used for troubleshooting with SL Technical Support).
- RTView Servers: The displays in this View enable you to view performance metrics gathered by RTView, and monitor all RTView Servers.

Overview

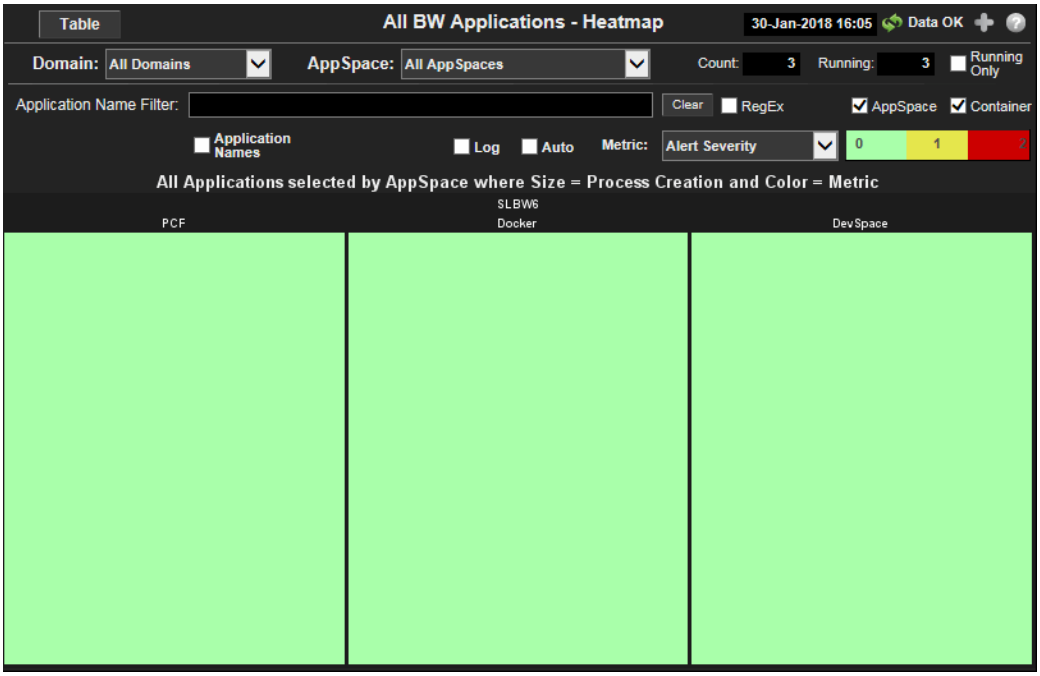
This section describes the main Monitor features, how to read Monitor objects, GUI functionality and navigation. This section includes:

- Monitor Main Display: Describes the Monitor display that opens by default as well as the navigation tree.
- Heatmaps: Describes how to read heatmaps and heatmap functionality.
- Tables: Describes how to read tables and table functionality.
- Trend Graphs: Describes how to read trend graphs and trend graph functionality.
- Using the Monitor: Describes the top layer of the title bar shared by Monitor displays.
- Context Menu: Describes right-click popup menu in the Monitor.

- Multiple Windows: Describes opening multiple windows in the Monitor.
- Export Report: Describes how to export reports from the Monitor.

Monitor Main Display

The **All Applications Heatmap** is the default display of the BW Monitor (if you are only running TIBCO BusinessWorks™ 5, the default display is the All Engines Heatmap). This color-coded heatmap provides a good starting point for immediately getting the status of all your BW applications. The following figure illustrates the Monitor.



Note: It takes about 60 seconds after either a BusinessWorks engine or the Monitor Data Server is started for data to initially appear in Monitor displays. By default, data is collected and displays are refreshed every 30 seconds.

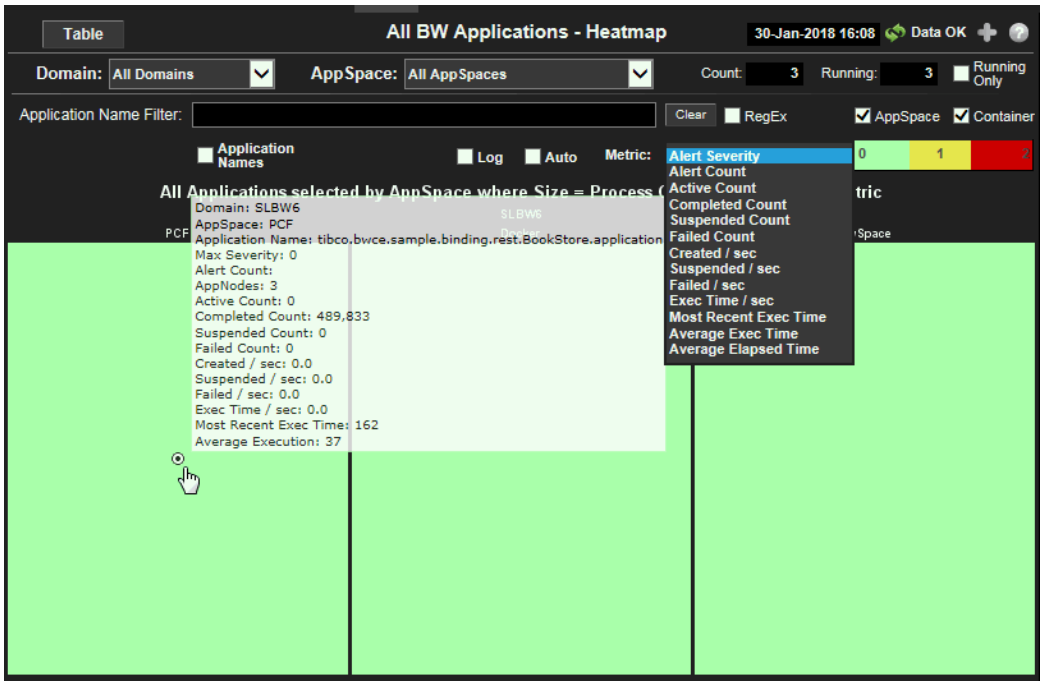
Navigation Tree

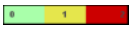
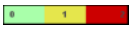
The BW Monitor navigation tree (in the left panel) is organized by *Views.* Each View features performance data for a type of system resource. Typically, the performance data is shown in a tabular, heatmap and summary display for each View.

Heatmaps

Heatmaps organize your BW resources (servers, processes, and so forth) into rectangles and use color to highlight the most critical values in each. Heatmaps enable you to view various metrics in the same heatmap using drop-down menus. Each metric has a color gradient bar that maps relative values to colors. In most heatmaps, the rectangle size represents the number of resources in the rectangle; a larger size is a larger value. Heatmaps include drop-down menus to filter data by. The filtering options vary among heatmaps.

For example, each rectangle in the **All Applications Heatmap** represents an application, where color is representative of the selected **Metric**.



The **Metric** drop-down menu in this heatmap contains options to show **Alert Severity**, **Alert Count**, as well as other metrics. Menu options vary according to the data populating the heatmap. **Alert Severity** is selected and its corresponding color gradient  bar is shown. **Alert Severity** is the maximum level of alerts in the heatmap rectangle. Values range from **0** - **2**, as indicated in the color gradient  bar, where **2** is the highest **Alert Severity**:

- Red indicates that one or more resources associated with that application currently has an alert in an alarm state.
- Yellow indicates that one or more resources associated with that application currently have an alert in a warning state.
- Green indicates that no resources associated with that application have alerts in a warning or alarm state.

In most heatmaps, you can also drill-down to a *Summary* display containing detailed data for the resource (in this case, you drill-down to detailed data for the selected application in the **Single Application Summary** display). You can also open a new window and then drill-down. The drill-down opens a display that contains relevant and more detailed data.

Mouse-over

The mouse-over functionality provides additional detailed data in an over imposed pop-up window when you mouse-over a heatmap. The following figure illustrates mouse-over functionality in a heatmap object.

Log Scale

Typically, heat maps provide the Log Scale option, which enables visualization on a logarithmic scale. This option should be used when the range in your data is very broad. For example, if you have data that ranges from the tens to the thousands, then data in the range of tens will be neglected visually if you do not check this option. This option makes data on both extreme ranges visible by using the logarithmic of the values rather than the actual values.

Tables

Monitor tables contain the same data that is shown in the heatmap in the same View. Tables provide you a text and numeric view of the data shown in that heatmap, and additional data not included the heatmap. For example, the **All Applications Table** display (shown below) shows the same data as the **All Applications Heatmap** display (shown previously).

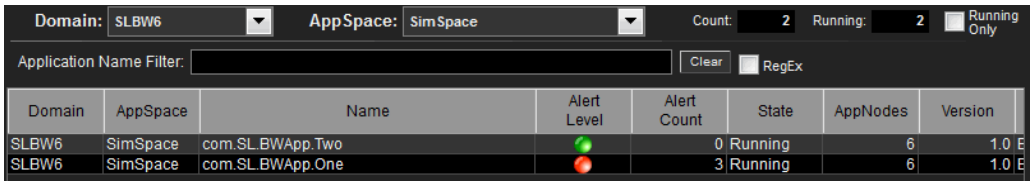


Table rows also sometimes use color to indicate the current most critical alert state for all resources associated with a given row. For example, the color coding is typically as follows:

Red indicates that one or more resources associated with that application currently has an alert in an alarm state.

Yellow indicates that one or more resources associated with that application currently have an alert in a warning state.

Green indicates that no resources associated with that application currently have an alert in a warning or alarm state.

Tables support advanced HTML interactive features such as sorting on multiple columns, filtering on multiple columns, column resizing, column reordering, and hiding columns. Many of these features are accessed from the column menu, shown in the screen shot above, which you open by clicking on the menu icon in a column's header.

Some tables in the **Components** tab gray out rows when they’re in an expired state. A row is expired when data has not been received within the time specified in the solution package that is hosting the data.

Also see:

- Multiple Column Sorting

- Column Visibility
- Column Filtering
- Column Locking
- Column Reordering
- Saving Settings
- Row Paging
- Row Color Code
- Row Keyboard Selection

Multiple Column Sorting

Click on a column header to sort the table by that column. On the first click, the column is sorted in ascending order (smallest value at the top), on the second click the sort is in descending order, and on the third click, the column is returned to its original unsorted state. A sort on a string column is case-insensitive.

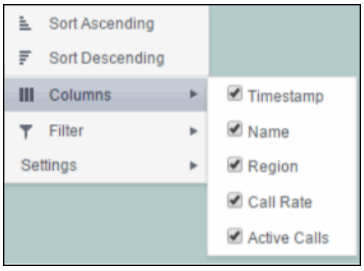
To sort multiple columns, click on the column header for each column you want to sort. The sorting is performed in the order that the column headers were clicked. Multiple column sorting is a very useful feature, but can also cause confusion if you intend to sort on a single column, but forget to "unsort" any previously selected sort columns first. You should check for the up/down sort icon in other column headers if a sort gives unexpected results.

The grid's row selection is cleared if the sort is changed or if columns are resized or reordered.

Column sorting is reflected in an export to HTML and Excel.

Column Visibility

You can hide or show columns in the table by clicking on any column's menu icon, and choosing **Columns** from the menu. This opens a submenu with a check box for each column that toggles the visibility of the column. All columns in the data table appear in the Columns menu, even those that are initially hidden.



The leftmost column (the row header column) cannot be hidden.

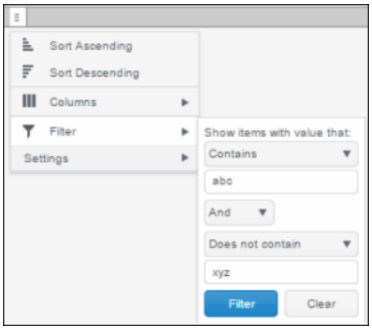
Column visibility changes are NOT reflected in an export to HTML and Excel.

Column Filtering

You can create a filter on any column. If filters are created on multiple columns, then only the rows that pass all of the filters are displayed. That is, if there are multiple filters they are logically "ANDed" together to produce the final result.

The background of a column's menu icon changes to white to indicate that a filter is defined on that column. This is intended to remind you which columns are filtered.

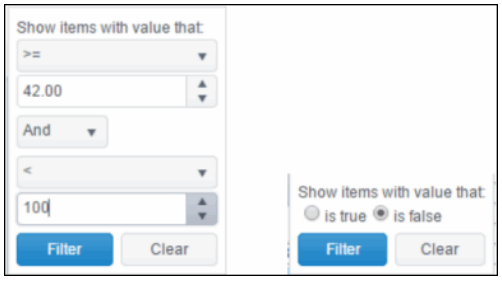
You can configure a filter on any column by clicking on the column's menu icon and choosing **Filter** from the menu. This opens the **Column Filter** dialog:



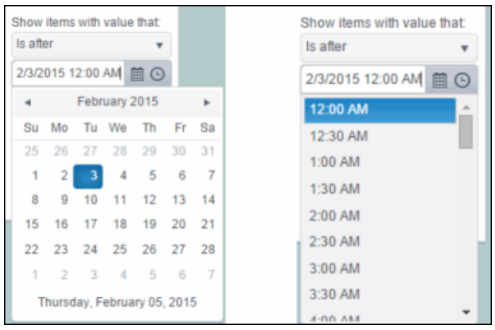
Options in the **Column Filter** dialog vary according to the data type of the selected column:

- **String columns**: You can enter a filter string such as "abc" and, from the dropdown list, select the operator (equal to, not equal to, starts with, contains, etc) to be used when comparing the filter string to each string in the column. All of the filter comparisons on strings are case-insensitive. You can optionally enter a second filter string (e.g. "xyz") and specify if an AND or OR combination should be used to combine the first and second filter results on the column.
- **Numeric columns**: You can enter numeric filter values and select arithmetic comparison operators, (=, !=, >, >=, <, <=). You can optionally enter a second filter value and comparison operator, and specify if an AND or OR combination should be used to combine the first and second filter results.
- **Boolean columns**: You simply select whether matching items should be true or false.

The numeric and boolean filter dialogs are shown below.



- **Date columns**: You can select a date and time and choose whether matching items should have a timestamp that is the same as, before, or after the filter time. The date is selected by clicking on the calendar icon and picking a date from a calendar dialog. The time is selected by clicking on the time icon and picking a time from a dropdown list:



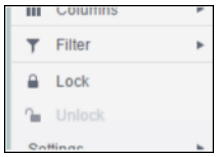
Alternatively, a date and time can be typed into the edit box. The strings shown in a date column are formatted by the Display Server using its time zone. But if a filter is specified on a date column, the date and time for the filter are computed using the client system's time zone. This can be confusing if the Display Server and client are in different time zones.

Data updates to the grid are suspended while the filter menu is opened. The updates are applied when the menu is closed.

Column filtering is reflected in an export to HTML and Excel.

Column Locking

The leftmost column is "locked" in position, meaning that it does not scroll horizontally with the other columns in the table. If the row header is enabled, then two items labeled **Lock** and **Unlock** appear in the column menu. These can be used to add or remove additional columns from the non-scrolling row header area.



If the row header is enabled, at least one column must remain locked.

Column locking is NOT reflected in an export to HTML and Excel.

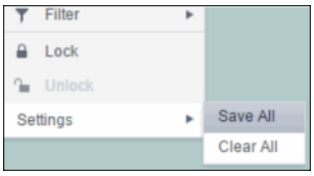
Column Reordering

You can reorder the grid columns by dragging and dropping a column's header into another position. Dragging a column into or out of the row header area (the leftmost columns) is equivalent to locking or unlocking the column.

Column reordering is NOT reflected in an export to HTML and Excel.

Saving Settings

You can permanently save all of the custom settings made to the grid, including filtering, sorting, column size (width), column order, column visibility, and column locking. This is done by opening any column menu, clicking **Settings**, and then clicking **Save All**:



The grid's settings are written as an item in the browser's local storage. The item's value is a string containing the grid's settings. The item uses a unique key comprised of the URL path name, the display name, and the table's RTView object name. If the Thin Client's login feature is enabled, the key will also include the username and role, so different settings can be saved for each user and role for a grid on any given display, in the same browser and host.

If you save the grid settings and navigate away from the display or close the browser, then the next time you return to the display in the same browser the settings are retrieved from the browser's local storage and applied to the grid. The browser's local storage items are persistent, so the grid settings are preserved if the browser is closed and reopened or if the host system is restarted.

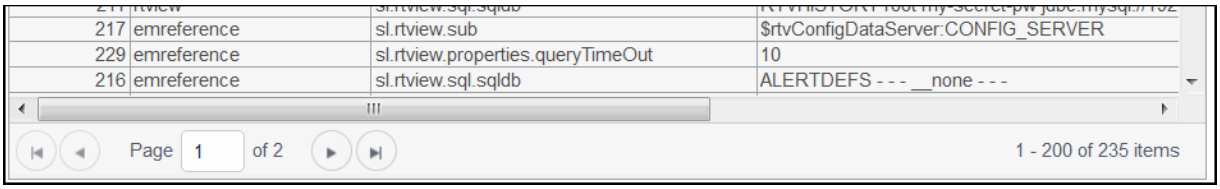
Note that each browser has its own local storage on each host. The local storage items are not shared between browsers on the same host or on different hosts. So, if a user logs in as Joe with **role = admin**, in Internet Explorer on host H1, and saves grid settings for display X, then those grid settings are restored each time a user logs in as Joe, role admin, on host H1 and opens display X in Internet Explorer. But if all the same is true except that the browser is Chrome, then the settings saved in Internet Explorer are not applied. Or if the user is Joe and role is admin and the browser is IE and the display is X, but the host system is H2 not H1, then the grid settings saved on H1 are not applied.

Revert Table Settings

You can delete the grid's item from local storage by clicking **Settings> Clear All** in any column menu. This permanently deletes the saved settings for the grid and returns the grid to the state defined in the display file.

Row Paging

If the data table contains more than one 200 rows, page controls appear at the bottom of the grid.



Row Color Code

Table rows sometimes use color to indicate the current most critical alert state for all CIs associated with the row. In this example, the **Severity Level** column is sorted in descending order (from high to low values).

The yellow row color indicates that one or more alerts exceeded their warning threshold for one or more CIs associated with the Service. The red row color indicates that one or more alerts exceeded their critical threshold for the CI associated with the Service (in this case there is a single CI). To summarize:

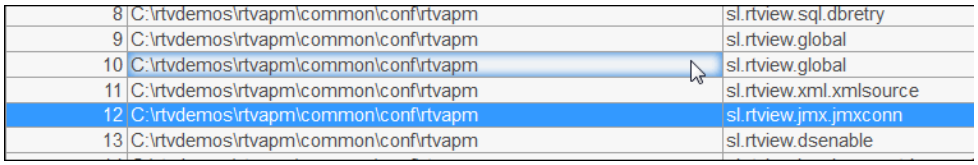
Row Color Code:

Tables with colored rows indicate the following:

- Red indicates that one or more alerts exceeded their ALARM LEVEL threshold in the table row.
- Yellow indicates that one or more alerts exceeded their WARNING LEVEL threshold in the table row.
- Green indicates that no alerts exceeded their WARNING or ALARM LEVEL threshold in the table row.

Row Keyboard Selection

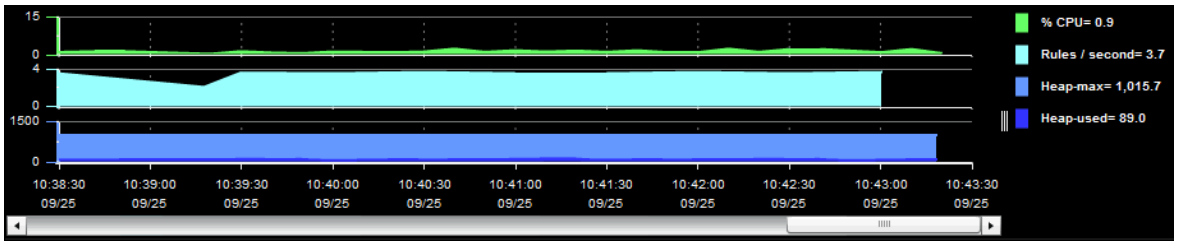
You can use the mouse to select a row and use the arrow keys to change the focus (highlighted) row, but to select the focus row, you must then press the space bar.



Trend Graphs

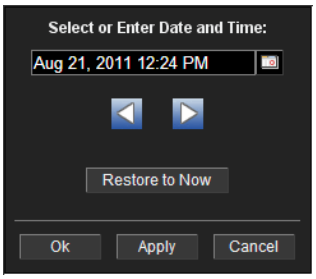
Monitor trend graphs enable you to view and compare performance metrics over time. You can use trend graphs to assess utilization and performance trends.

For example, the following figure illustrates a typical Monitor trend graph.



Time Range

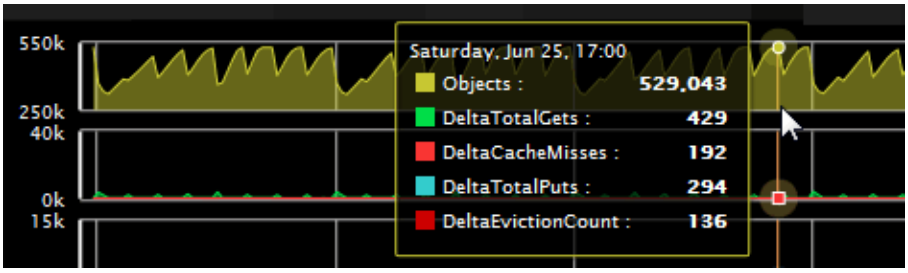
Select a time range from the drop down menu varying from **2 Minutes** to **Last 7 Days**, or display **All Data**. By default, the time range end point is the current time.



To change the time range click Open Calendar , choose the date and time, then click **OK**. Or enter the date and time in the text field using the following format: **MMM dd, YYYY HH:MM:ss**. For example, Aug 21, 2011 12:24 PM. Click **Apply**. Use the Navigation Arrows to move forward or backward one time period (the time period selected from the Time Range drop-down menu). Click **Restore to Now** to reset the time range end point to the current time.

Mouse-over

The mouse-over functionality provides additional detailed data in an over imposed pop-up window when you mouse-over trend graphs. The following figure illustrates mouse-over functionality. In this example, when you mouse-over a single dot, or data point, a pop-up window shows data for that data point.



Log Scale

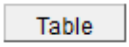
Typically, trend graphs provide the Log Scale option. Log Scale enables you to see usage correlations for data with a wide range of values. For example, if a minority of your data is on a scale of tens, and a majority of your data is on a scale of thousands, the minority of your data is typically not visible in non-log scale graphs. Log Scale makes data on both scales visible by applying logarithmic values rather than actual values to the data.

Title Bar Functionality

Displays share the same top layer in the title bar, as shown and described below.

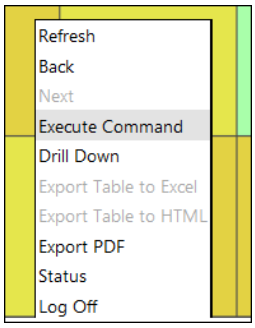
The following table describes the functionality in the display title bar.

	Opens the previous display.

	Opens the display that is up one level.
	Navigates to a display that is most commonly accessed from the current display. The target display differs among displays.
	Navigates to displays that are most commonly accessed from the current display. The drop-down menu options differ among displays.
	Opens the Alerts Table display in a new window.
	The current date and time. If the time is incorrect, this might indicate that RTView stopped running. When the date and time is correct and the **Data OK** indicator is green, this is a strong indication that the platform is receiving current and valid data.
	The data connection state. Red indicates the data source is disconnected (for example, if the Data Server is not receiving data, or if the Display Server does not receive data from the Data Server, this will be red). Green indicates the data source is connected. When the date and time is correct and the **Data OK** indicator is green, this is a strong indication that the platform is receiving current and valid data.
	The number of items currently in the display.
	Opens an instance of the same display in a new window. Each window operates independently, allowing you to switch views, navigate to other displays in RTView EM, and compare server performance data.
	Opens the online help page for the current display.

Context Menu

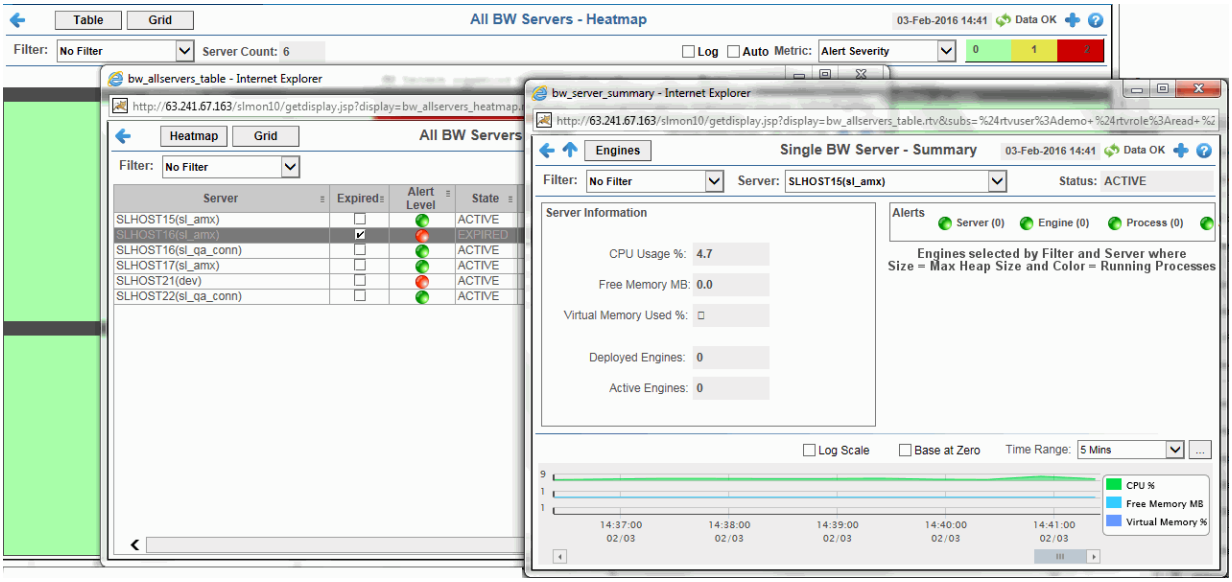
Typically, you can right-click on displays to open a popup menu. By default, options include **Refresh**, **Back**, **Next**, **Execute Command**, **Drill Down**, **Export Table to Excel**, **Export Table to HTML**, **Export PDF**, **Status** and **Log Off**. The following figure illustrates the popup menu in a heatmap.



For details about exporting a PDF report, see "Export Report" on page 209.

Multiple Windows

The following illustrates the use of Open New Window .

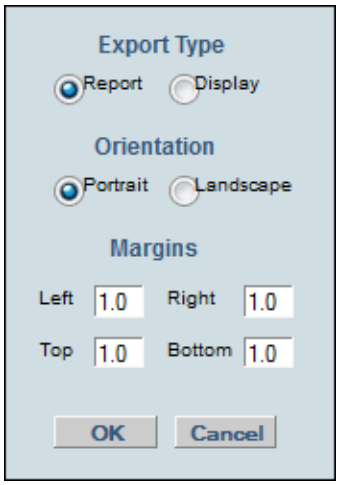


Export Report

You can quickly export reports for displays, or for tables and grid objects in a display, to a PDF file.

To generate a report for a display:

Right-click on the display and select **Export PDF**. The **Export to PDF** dialog opens.



Set the margins and choose the **Export Type**:

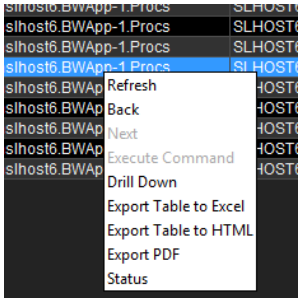
- **Report**: Generates an image of the display on the first page, followed by at least one page for each table or object grid in the display. As many pages as are necessary to show all the data in each table or object grid are included in the report. This enables you to view all data in a table or object grid that you otherwise must use a scrollbar to

see. If there are no tables or object grids in your display, you only get a image of the display.

- **Display**: Generates an image of the display in PDF format.Choose the page orientation (**Portrait** or **Landscape**), set the page margins and click **OK**. The report opens in a new window.

To generate a report for a table or grid object in a display:

Right-click on the table or grid object and choose **Export PDF**, **Export Table to Excel** or **Export Table to HTML**.



BusinessWorks Monitor Views/Displays

The following Views and their associated displays are in the Monitor. This section describes the Monitor displays and includes:

- BW Applications: The displays in this View present BusinessWorks application performance metrics.
- BW Containers: The displays in this View present BusinessWorks container performance metrics.
- BW AppNodes: The displays in this View present BusinessWorks AppNode performance metrics.
- BW AppSlices: The displays in this View present BusinessWorks AppSlice performance metrics.
- BW Processes: The displays in this View present BusinessWorks process performance metrics.
- BW5 Engines: The displays in this View present BusinessWorks 5.0 engine performance metrics.
- BW5 Processes: The displays in this View present BusinessWorks 5.0 process performance metrics.
- BW5 Activities: The displays in this View present BusinessWorks 5.0 activity performance metrics.
- BW5 Servers: The displays in this View present BusinessWorks 5.0 server performance metrics.
- Hawk Views: Describes displays for TIBCO® BusinessWorks Hawk data.
- Alert Views: The display in this View presents the status of all alerts across all BW Servers, and allows you to track, manage and assign alerts.
- Administration: The displays in this View enable you to set global alerts and override alerts. You can also view internal data gathered and stored by RTView (used for troubleshooting with SL Technical Support).

- RTView Servers: The displays in this View enable you to view performance metrics gathered by RTView, and monitor all RTView Servers.

BW Applications

These displays present process performance data for your BusinessWorks applications and AppSpaces across BusinessWorks Domains. Process metrics are totaled by application. Use these displays to monitor critical alerts for all your BusinessWorks applications, and investigate those alerts in lower-level displays. Displays in this View are:

- "BW All Applications Heatmap" on page 211: A color-coded heatmap view of selected application performance metrics.
- "BW All Applications Table" on page 214: A tabular view of all available application performance data in this BusinessWorks View.
- "BW Single Application Summary" on page 216: Current and historical metrics for a single application.

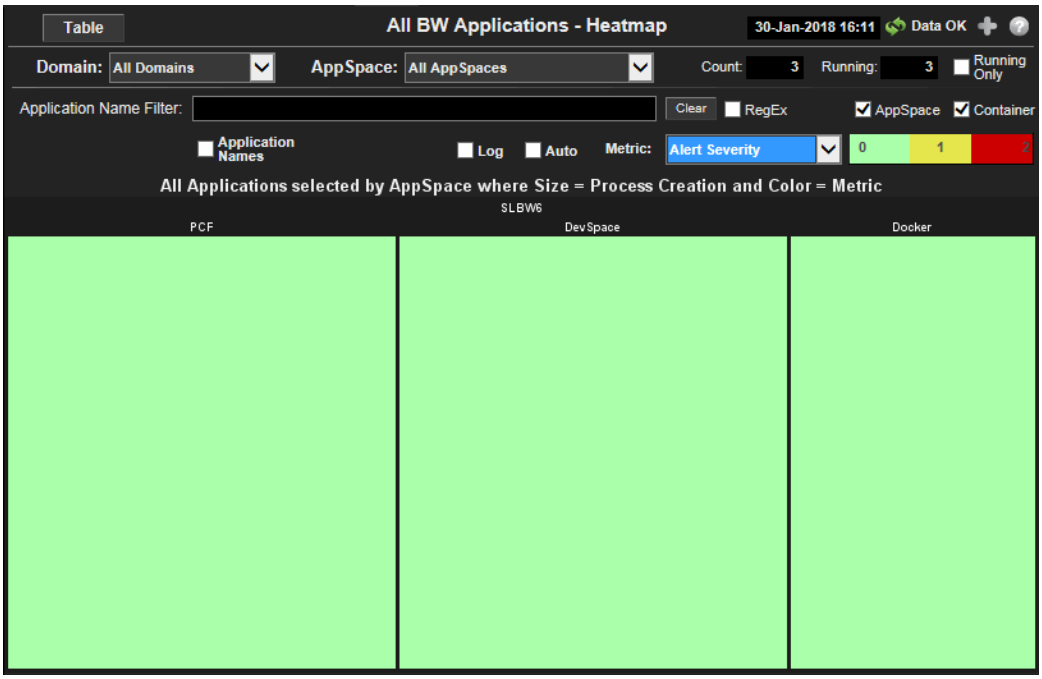
BW All Applications Heatmap

View the most critical BusinessWorks application alert states pertaining to process creation and execution for all nodes on which the applications are deployed. Use this display to quickly identify applications with critical alerts.

Each rectangle in the heatmap represents an application. The rectangle color indicates the most critical alert state associated with the application. The rectangle size represents process creation across applications; a larger size is a larger value.

Choose a domain and AppSpace from the drop-down menus. Choose a different metric to display from the **Metric** drop-down menu. Enter a string in the **Application Name Filter** field to limit data shown in the display. Use the **Application Names** check-box ☑ to include or exclude labels in the heatmap. Mouse over a rectangle to see additional metrics. By default, this display shows **Alert Severity**.

Drill-down and investigate an application by clicking a rectangle in the heatmap to view details in the BW Single Application Summary display.



Title Bar (possible features are):

- Open the previous and upper display.
- Open an instance of this display in a new window.
- Open the online help page for this display.
- Menu, Table open commonly accessed displays.
- 6,047 The number of items currently in the display.
- Data OK Data connection state. Red indicates the Data Server is not receiving data or the Display Server is not receiving data from the Data Server. Green indicates the data source is connected.
- 23-Mar-2017 12:04 Current date and time. Incorrect time might indicate the Monitor stopped running. Correct time and green **Data OK** icon is a strong indication that data is current and valid.
- Open the **Alert Views - RTView Alerts Table** display.

Filter By:

The display might include these filtering options:

Domain:	Select the domain for which you want to view data in the display.
AppSpace	Select the AppSpace for which you want to view data in the display.
Application Name Filter	Enter a string (all or part of a application name) to filter the data shown in the display. If you enter part of an application name, you must enter "*****" before and/or after the string. For example, if you have an application named AppNameOne, you could filter using *Name*, *NameOne, or AppName*.
Clear	Clears the **Application Name Filter** entries from the display.
RegEx	Toggles the **Application Name Filter** to accept Regular Expressions for filtering. For example, if your application name is AppNameOne and this option was toggled on, you could enter "Name" (without using "*****"to display the application in the heatmap).
AppSpace	When selected, those AppNodes deployed in an AppSpace display in the heatmap.
Container	When selected, those AppNodes deployed in a container display in the heatmap.
Application Names	Check to include labels in the heatmap.

Fields and Data:

Count:	The total number of AppSpaces currently shown in the display.

Running	The total number of AppSpaces currently running in the display.
Running Only	Select to show only running applications in the display.
Log	Select to enable a logarithmic scale. Use **Log Scale** to see usage correlations for data with a wide range of values. For example, if a minority of your data is on a scale of tens, and a majority of your data is on a scale of thousands, the minority of your data is typically not visible in non-log scale graphs. **Log Scale** makes data on both scales visible by applying logarithmic values rather than actual values to the data.
Auto	Select to enable auto-scaling. When auto-scaling is activated, the color gradient bar's maximum range displays the highest value. NOTE: Some metrics auto-scale automatically, even when **Auto** is not selected.
Metric	Choose a metric to view in the display.

Alert Severity	The maximum level of alerts in the heatmap rectangle. Values range from **0** - **2**, as indicated in the color gradient  bar, where **2** is the highest Alert Severity: Red indicates that one or more metrics exceeded their ALARM LEVEL threshold. Yellow indicates that one or more metrics exceeded their WARNING LEVEL threshold. Green indicates that no metrics have exceeded their alert thresholds.
Alert Count	The total number of critical and warning alerts in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average alert count.
Active Count	The total number of active processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Completed Count	The total number of completed processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Suspended Count	The total number of suspended processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Failed Count	The total number of failed processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Created / sec	The number of processes created per second in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Suspended / sec	The number of suspended processes per second in the heatmap

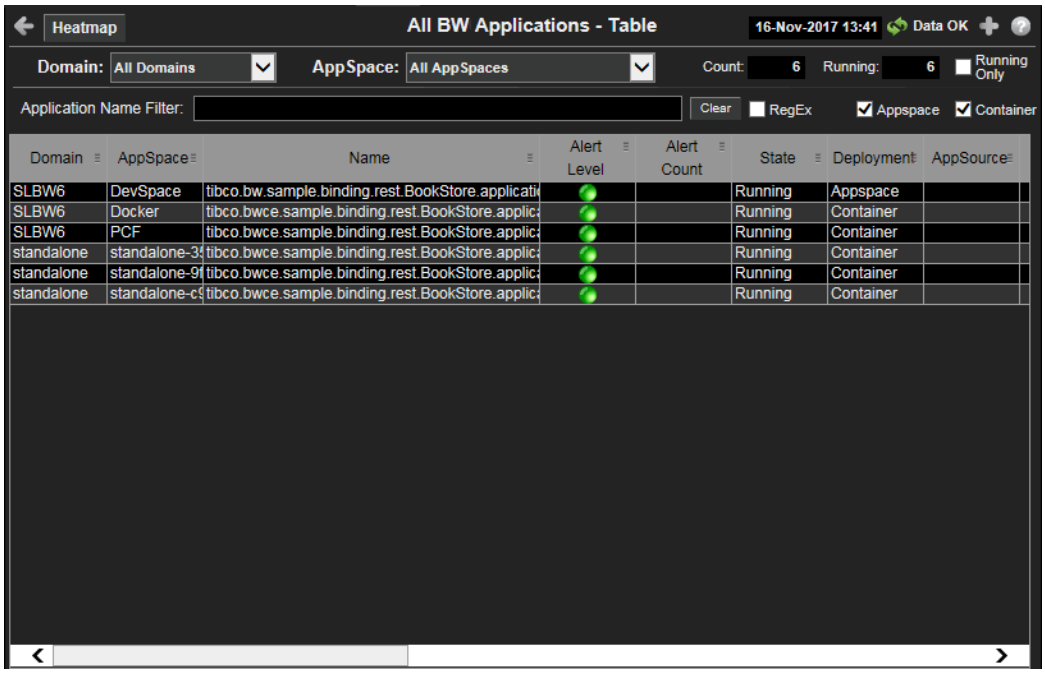
	bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Failed / sec	The number of failed processes per second in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Exec Time / sec	The process execution time per second in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Most Recent Exec Time	The execution time for the most recently executed process in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Average Exec Time	The average execution time for all processes in the heatmap rectangle, calculated by dividing the delta execution time for the interval by the delta completed, or the number of process instances that completed in the interval. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Average Elapsed Time	The average elapsed time for all processes in the heatmap rectangle, calculated by dividing the delta elapsed time for the interval by the delta completed, or the number of process instances that completed in the interval. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.

BW All Applications Table

View BusinessWorks data shown in the BW All Applications Heatmap, and additional details, in a tabular format.

Each row in the table is an application. Choose a domain and AppSpace from the drop-down menus. Click a column header to sort column data in numerical or alphabetical order. Enter a string in the **Application Name Filter** field to limit data shown in the display.

Drill-down and investigate by clicking a row to view details for the selected application in the BW Single Application Summary display.



Title Bar (possible features are):

Open the previous and upper display.

Open an instance of this display in a new window.

Open the online help page for this display.

Menu, Table open commonly accessed displays.

6,047 The number of items currently in the display.

Data OK Data connection state. Red indicates the Data Server is not receiving data or the Display Server is not receiving data from the Data Server. Green indicates the data source is connected.

23-Mar-2017 12:04 Current date and time. Incorrect time might indicate the Monitor stopped running. Correct time and green **Data OK** icon is a strong indication that data is current and valid.

Open the **Alert Views - RTView Alerts Table** display.

Filter By:

The display might include these filtering options:

Domain:	Choose a domain to show data for in the display.
AppSpace	Choose an AppSpace to show data for in the display.
Application Name Filter	Enter a string (all or part of a application name) to filter the data shown in the display. If you enter part of an application name, you must enter "*****" before and/or after the string. For example, if you have an application named AppNameOne, you could filter using *Name*, *NameOne, or AppName*.
Clear	Clears the **Application Name Filter** entries from the display.
RegEx	Toggles the **Application Name Filter** to accept Regular Expressions for filtering. For example, if your application name is AppNameOne and this option was toggled on, you could enter "Name" (without using "*****"to display the application in the table).
AppSpace	When selected, those AppNodes deployed in an AppSpace display in the table.
Container	When selected, those AppNodes deployed in a container display in the table.

Fields and Data:

Count:	The total number of applications in the AppSpace.
Running	The total number of applications currently running in the AppSpace.
Running Only	Select to show only running applications in the display.

Table:

Each row in the table is a different application.

Domain	The domain in which the application resides.
AppSpace	The AppSpace in which the application resides.
Name	The name of the application.
Alert Level	The most critical alert state for alerts in the row: Red indicates that one or more metrics exceeded their ALARM LEVEL threshold. Yellow indicates that one or more metrics exceeded their WARNING LEVEL threshold. Green indicates that no metrics have exceeded their alert thresholds.
Alert Count	The total number of active alerts for the application.
State	The current status of the application. Valid values are **Running** and **Stopped.**
AppNodes	The total number of AppNodes associated with the application.
Active Processes	The number of currently active application processes.
Suspended Processes	The number of suspended application processes.
Failed Processes	The number of failed application processes.
Completed Processes	The number of completed application processes.
Created /sec	The number of application processes created per second.
Suspended /sec	The number of application process suspensions per second.
Failed /sec	The number of application process failures per second.
Exec Time/sec	The number of processes executed per second.
Recent Exec Time	The number of seconds for the most recently executed process.
Average Exec Time	The average number of seconds for all processes to execute.
Version	The application version.
Module	The application module.
Shared Module	The shared module, if any.
Time Stamp	The date and time the row data was last updated.
Source	Name of RTView Data Server sending this data (or localhost).
Expired	When checked, data has not been received from this host in the specified amount of time.

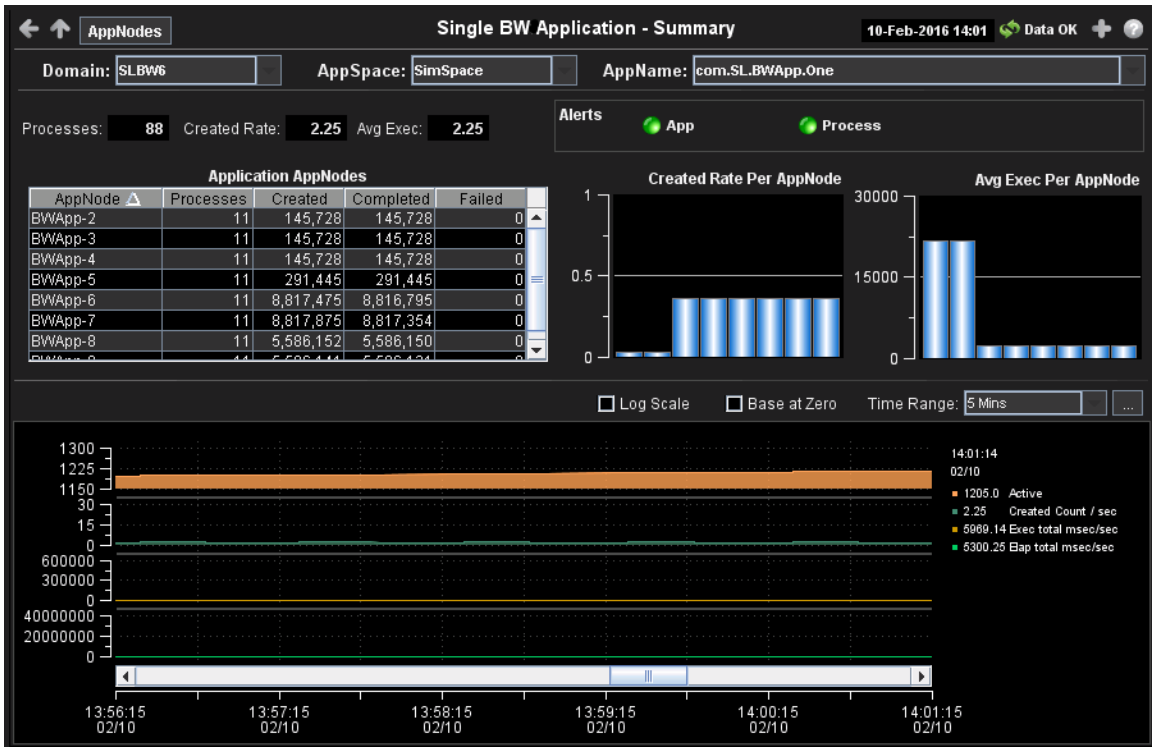
BW Single Application Summary

View current and historical metrics for a single BusinessWorks application across multiple nodes. Use this display to investigate performance issues of application AppNodes within an AppSpace. Use this display to view all available data for each AppNode by Domain and AppSpace.

This display includes a list of AppNodes with their host names and memory metrics, bar graphs per AppNode for process creation and execution, and trend graphs of process creation and execution metrics.

The summary display also shows the AppNodes of the deployment and process metrics totaled by AppNode. This is useful to see the deployment and load balancing of the Application in current and historical time.

Choose a domain, AppSpace and Application from the drop-down menus. Drill-down and investigate by clicking an AppNode in the table to view details in the BW Single AppNode Summary display.



Title Bar (possible features are):

Open the previous and upper display.

Open an instance of this display in a new window.

Open the online help page for this display.

Menu, Table open commonly accessed displays.

6,047 The number of items currently in the display.

Data OK Data connection state. Red indicates the Data Server is not receiving data or the Display Server is not receiving data from the Data Server. Green indicates the data source is connected.

23-Mar-2017 12:04 Current date and time. Incorrect time might indicate the Monitor stopped running. Correct time and green **Data OK** icon is a strong indication that data is current and valid.

Open the **Alert Views - RTView Alerts Table** display.

Filter By:

The display might include these filtering options:

Domain: Select the domain for which you want to view data in the display.

AppSpace Choose the AppSpace for which you want to view data in the display.

AppName: Choose the AppName for which you want to view data in the display.

Fields and Data:

Processes: The number of processes currently running for the selected application.

Created Rate: The number of processes created per second for the selected application.

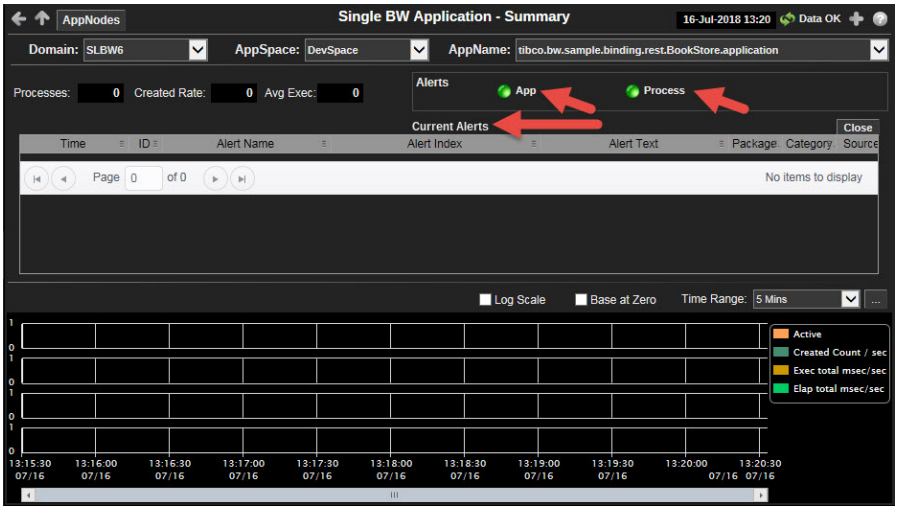
Avg Exec: The average number of seconds for processes to execute for the selected application.

Alerts

Indicates the greatest severity level and the number of open **App** and **Process** alerts for the selected application. Values range from **0** to **2**, where **2** is the greatest Severity:

- One or more alerts exceeded their ALARM LEVEL threshold.
- One or more alerts exceeded their WARNING LEVEL threshold.
- No alert thresholds have been exceeded.

Click on the alert indicator to display a table listing the current alerts for the selected application. Click the **Close** button (for the current alerts table) to close the table.



Application Appnodes Table

Each row in the table is a different AppNode. Column values describe processes for the selected application on that AppNode. Click a row to view AppNode details in the BW Single AppNode Summary display.

AppNode	The name of the AppNode.
Deployment	Describes the application's AppNode deployment type: **AppSpace** indicates that the AppNodes are AppSpace deployments. **Container** indicates that the AppNodes are Container deployments. **Hybrid** indicates that the AppNodes are a combination of AppSpace and Container deployments.
CPU%	The percent CPU utilization on the AppNode.
Memory	The percent memory utilization on the AppNode.

Created Rate Per AppNode Bar Graph

The bar graph shows the current process creation rate per AppNode. Click to drill-down and investigate in the BW Single AppSlice Summary display.

AvgExec Per AppNode Bar Graph

The bar graph shows the current average process execution rate per AppNode for the selected application. Click to drill-down and investigate in the BW Single AppSlice Summary display.

Trend Graphs

Traces the sum of process metrics across all processes in all slices of the selected application.

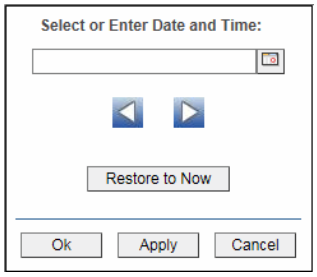
Active	Traces the number of currently active application processes.
Created Count / sec	Traces the number of created application processes.
Exec total msec/sec	Traces the rate at which the application is accumulating process execution time, in milliseconds per second.
Elap total msec/sec	Traces the rate at which the application accumulates process elapsed time, in milliseconds per second.

Log Scale	Select to enable a logarithmic scale. Use **Log Scale** to see usage

correlations for data with a wide range of values. For example, if a minority of your data is on a scale of tens, and a majority of your data is on a scale of thousands, the minority of your data is typically not visible in non-log scale graphs. **Log Scale** makes data on both scales visible by applying logarithmic values rather than actual values to the data.

Base at Zero Select to use zero (**0**) as the Y axis minimum for all graph traces.

Time Range Select a time range from the drop down menu varying from **2 Minutes** to **Last 7 Days**, or display **All Data**. To specify a time range, click Calendar .



By default, the time range end point is the current time. To change the time range end point, click Calendar and select a date and time from the calendar or enter the date and time in the text field using the following format: **MMM dd, YYYY HH:MM**. For example, **Aug 21, 2011 12:24 PM**.

Use the navigation arrows to move forward or backward one time period. NOTE: The time period is determined by your selection from the **Time Range** drop-down menu.

Click **Restore to Now** to reset the time range end point to the current time.

BW Containers

These displays present process performance data for your BusinessWorks containers across BusinessWorks Domains. Process metrics are totaled by container. Use these displays to monitor critical alerts for all your BusinessWorks containers, and investigate those alerts in lower-level displays. Displays in this View are:

- "All Containers Heatmap" on page 219: A color-coded heatmap view of selected container performance metrics.
- "All Containers Table" on page 222: A tabular view of all available container performance data in this BusinessWorks View.
- "Single Container Summary" on page 224: Current and historical metrics for a single container.

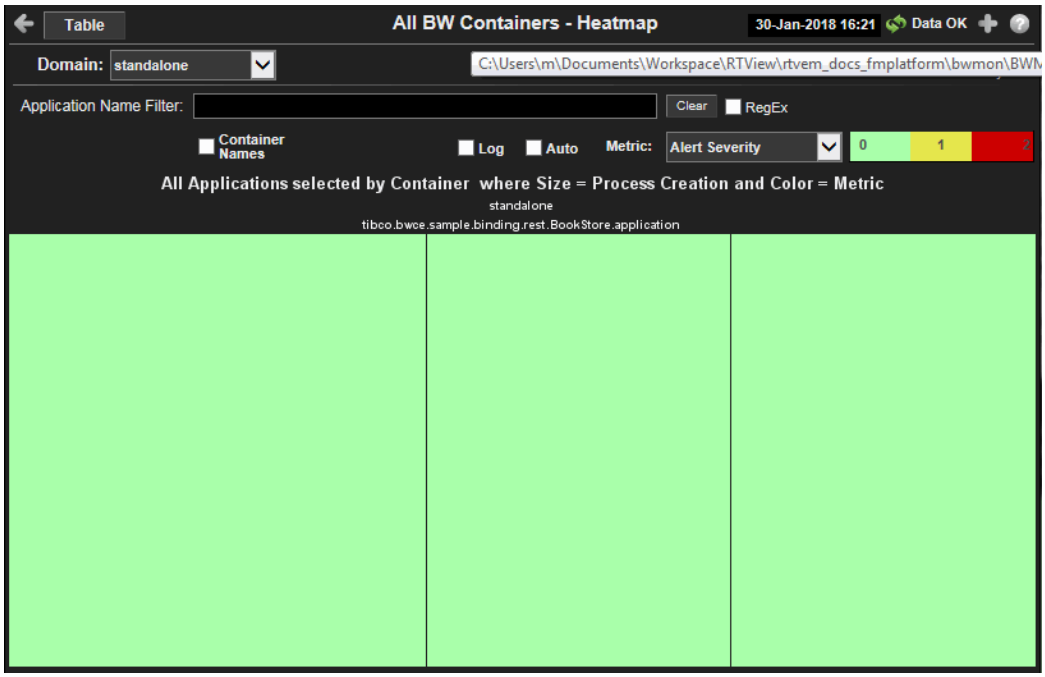
All Containers Heatmap

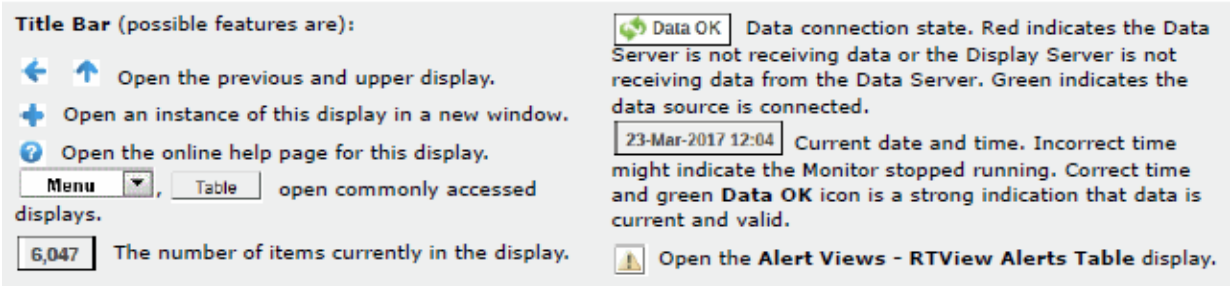
View the most critical BusinessWorks container alert states pertaining to process creation and execution for all nodes on which the containers are deployed. Use this display to quickly identify containers with critical alerts.

Each rectangle in the heatmap represents a container. The rectangle color indicates the most critical alert state associated with the container. The rectangle size represents process creation across containers; a larger size is a larger value.

Choose a domain and AppSpace from the drop-down menus containing the containers for which you want to view metrics. By default, this display shows the heatmap based on the **Alert Severity** metric, but you can select a different metric from the **Metric** drop-down menu to view the heatmap based on the selected metric. To view data shown for a specific container (s) in the display, enter a string in the **Container Name Filter** field. Use the **Container Names** check-box ☑ to include or exclude labels in the heatmap. You can mouse over a rectangle to see additional metrics.

Drill-down and investigate a container by clicking a rectangle in the heatmap, which opens the details for the selected container in the Single Container Summary display.





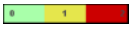
Filter By:

The display might include these filtering options:

Domain:	Select the domain for which you want to view data in the display.
Application Name Filter	Enter a string (all or part of an application name) to filter the data shown in the display. If you enter part of a application name, you must enter "*****" before and/or after the string. For example, if you have an application named AppNameOne, you could filter using *Name*, *NameOne, or AppName*. You can also enable the **RegEx** toggle to just enter a portion of the application name.

Clear	Clears the **Application Name Filter** entries from the display.
RegEx	Toggles the **Application Name Filter** to accept Regular Expressions for filtering. For example, if your application name is AppNameOne and this option was toggled on, you could enter "Name" (without using "*"to display the application in the heatmap).
Container Names	Check to include container name labels in the heatmap.

Fields and Data:

Count:	The total number of containers currently shown in the display.
Running	The total number of containers currently running in the display.
Running Only	Select to show only running containers in the display.
Log	Select to enable a logarithmic scale. Use **Log Scale** to see usage correlations for data with a wide range of values. For example, if a minority of your data is on a scale of tens, and a majority of your data is on a scale of thousands, the minority of your data is typically not visible in non-log scale graphs. **Log Scale** makes data on both scales visible by applying logarithmic values rather than actual values to the data.
Auto	Select to enable auto-scaling. When auto-scaling is activated, the color gradient bar's maximum range displays the highest value. NOTE: Some metrics auto-scale automatically, even when **Auto** is not selected.
Metric	Choose a metric to view in the display.
Alert Severity	The maximum level of alerts in the heatmap rectangle. Values range from **0** - **2**, as indicated in the color gradient  bar, where **2** is the highest Alert Severity: Red indicates that one or more metrics exceeded their ALARM LEVEL threshold. Yellow indicates that one or more metrics exceeded their WARNING LEVEL threshold. Green indicates that no metrics have exceeded their alert thresholds.
Alert Count	The total number of critical and warning alerts in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average alert count.
Active Count	The total number of active processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Completed Count	The total number of completed processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Suspended Count	The total number of suspended processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Failed Count	The total number of failed processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts

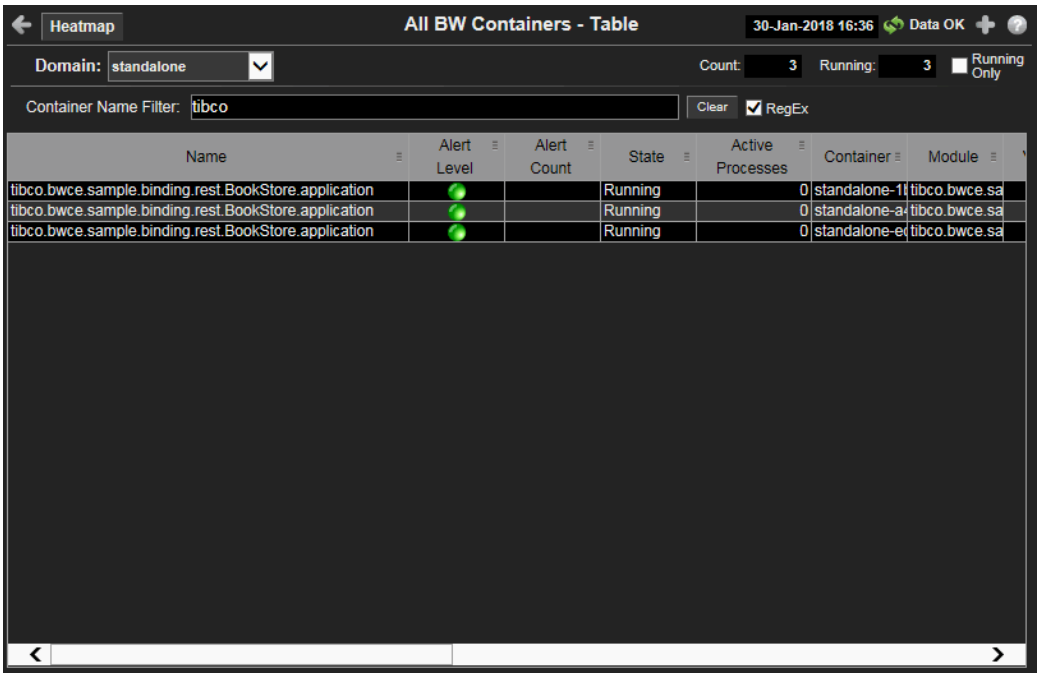
	in the heatmap. The middle value in the gradient bar indicates the average count.
Created / sec	The number of processes created per second in the heatmap rectangle. The color gradient bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Suspended / sec	The number of suspended processes per second in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Failed / sec	The number of failed processes per second in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Exec Time / sec	The process execution time per second in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Most Recent Exec Time	The execution time for the most recently executed process in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Average Exec Time	The average execution time for all processes in the heatmap rectangle, calculated by dividing the delta execution time for the interval by the delta completed, or the number of process instances that completed in the interval. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Average Elapsed Time	The average elapsed time for all processes in the heatmap rectangle, calculated by dividing the delta elapsed time for the interval by the delta completed, or the number of process instances that completed in the interval. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.

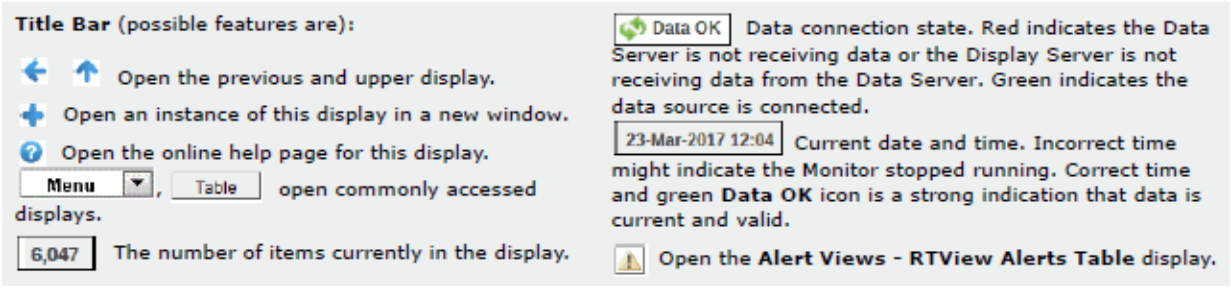
All Containers Table

This display provides a view of the most critical BusinessWorks container alert states pertaining to process creation and execution for all nodes on which the containers are deployed in a tabular format. Use this display to quickly identify containers with critical alerts. Each row in the table is a container in the selected domain.

Select a domain from the drop-down menu to view associated containers and, optionally, enter a string in the **Container Name Filter** field to further limit the list of containers shown in the display. You can click a column header to sort column data in numerical or alphabetical order.

To view additional details for a specific container, drill-down and investigate by clicking the row in the table for the desired container, which opens the Single Container Summary display.





Filter By:

The display might include these filtering options:

Domain:	Choose a domain to show data for in the display.
Container Name Filter	Enter a string (all or part of a container name) to filter the data shown in the display. If you enter part of an container name, you must enter "*****" before and/or after the string. For example, if you have a container named ContNameOne, you could filter using *Name*, *NameOne, or ContName*.
Clear	Clears the **Container Name Filter** entries from the display.
RegEx	Toggles the **Container Name Filter** to accept Regular Expressions for filtering. For example, if your application name is ContNameOne and this option was toggled on, you could enter "Name" (without using "*****"to display the container in the table).

Fields and Data:

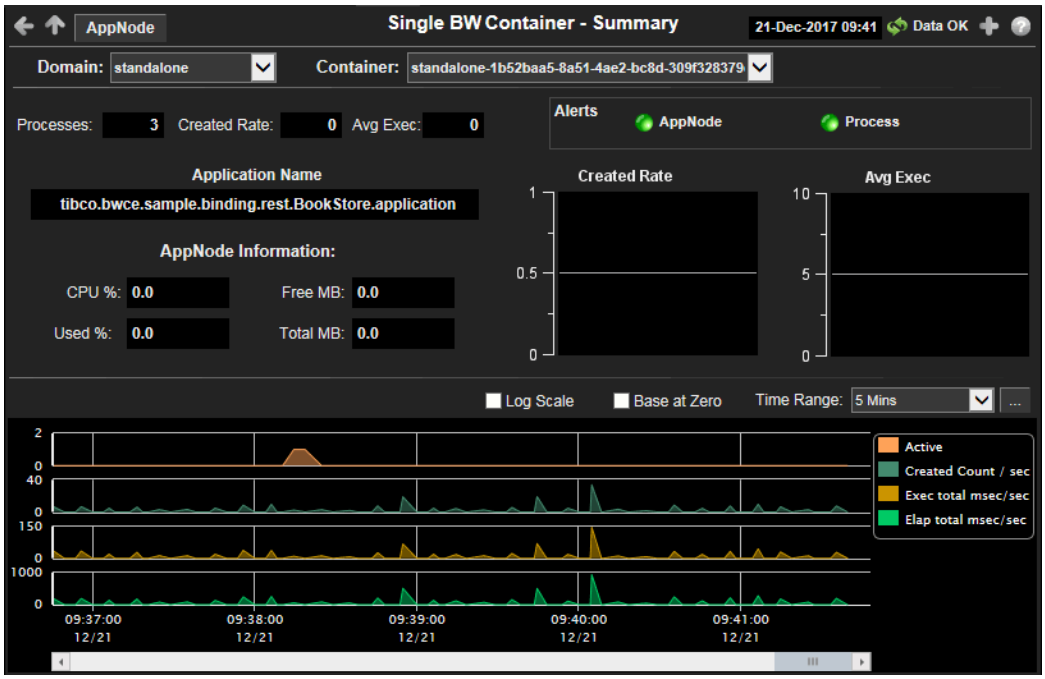
Count:	The total number of containers listed in the table.
Running	The total number of containers that are currently running.

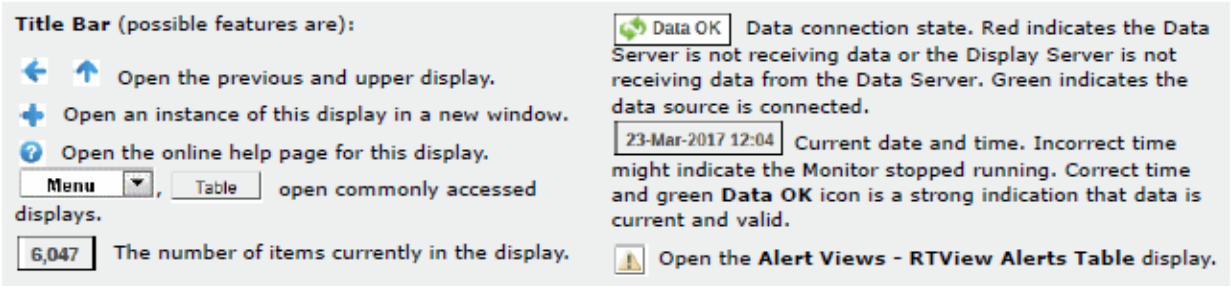
Running Only	Select to show only running containers in the display.

Table:
Each row in the table is a different application.

Name	The name of the container.
Alert Level	The most critical alert state for alerts in the row: Red indicates that one or more metrics exceeded their ALARM LEVEL threshold. Yellow indicates that one or more metrics exceeded their WARNING LEVEL threshold. Green indicates that no metrics have exceeded their alert thresholds.
Alert Count	The total number of active alerts in the container.
State	The current status of the application. Valid values are **Running** and **Stopped.**
Active Processes	The number of currently active processes in the container.
Container	The name of the container.
Module	The name of the container module.
Version	The version of the container.
Average Exec Time	The average number of seconds for all processes to execute.
Completed Processes	The number of completed processes in the container.
Expired	When checked, data has not been received from this host in the specified amount of time.
Failed Processes	The number of failed processes in the container.
Recent Exec Time	The number of seconds for the most recently executed process.
Created /sec	The number of processes created per second in the container.
Failed /sec	The number of process failures per second in the container.
Suspended /sec	The number of process suspensions per second in the container.
Exec Time/sec	The number of processes executed per second in the container.
Suspended Processes	The number of suspended application processes in the container.
Time Stamp	The date and time the row data was last updated.
Source	Name of RTView Data Server sending this data (or localhost).

Single Container Summary





Filter By:

The display might include these filtering options:

Domain: Select the domain for which you want to view data in the display.

Container Choose the container for which you want to view data in the display.

Fields and Data:

Processes: The number of processes currently running on the selected container.

Created Rate: The number of processes created per second on the selected container.

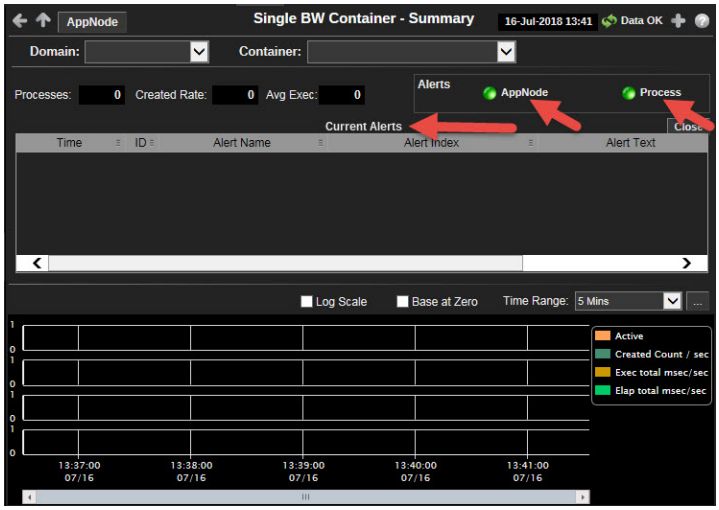
Avg Exec: The average number of seconds for processes to execute on the selected container.

Alerts

Indicates the greatest severity level and the number of open **AppNode** and **Process** alerts for the selected container. Values range from **0** to **2**, where **2** is the greatest Severity:

- One or more alerts exceeded their ALARM LEVEL threshold.
- One or more alerts exceeded their WARNING LEVEL threshold.
- No alert thresholds have been exceeded.

Click on the alert indicator to display a table listing the current alerts for the selected container. Click the **Close** button (for the current alerts table) to close the table.



Application Name	The name of the application running on the container.	
AppNode Information		
	CPU %	The percentage of CPU used by the AppNode.
	Used %	The percentage of memory used by the AppNode.
	Free MB	The amount of free memory, in megabytes.
	Total MB	The total amount of used and free memory, in megabytes.

Created Rate Bar Graph

The bar graph shows the current process creation rate per AppNode. Click to drill-down and investigate in the BW Single AppSlice Summary display.

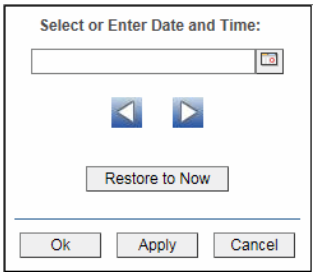
AvgExec Bar Graph

The bar graph shows the current average process execution rate per AppNode for the selected application. Click to drill-down and investigate in the BW Single AppSlice Summary display.

Trend Graphs

Traces the sum of process metrics across all processes in all slices of the selected container.

	Active	Traces the number of currently active application processes on the container.
	Created Count / sec	Traces the number of created application processes on the container.
	Exec total msec/sec	Traces the rate at which the application is accumulating process execution time, in milliseconds per second, on the container.
	Elap total msec/sec	Traces the rate at which the application accumulates process elapsed time, in milliseconds per second, on the container.
Log Scale	Select to enable a logarithmic scale. Use **Log Scale** to see usage correlations for data with a wide range of values. For example, if a minority of your data is on a scale of tens, and a majority of your data is on a scale of thousands, the minority of your data is typically not visible in non-log scale graphs. **Log Scale** makes data on both scales visible by applying logarithmic values rather than actual values to the data.	
Base at Zero	Select to use zero (**0**) as the Y axis minimum for all graph traces.	
Time Range	Select a time range from the drop down menu varying from **2 Minutes** to **Last 7 Days**, or display **All Data**. To specify a time range, click Calendar .	



By default, the time range end point is the current time. To change the time range end point, click Calendar and select a date and time from the calendar or enter the date and time in the text field using the following format: **MMM dd, YYYY HH:MM**. For example, **Aug 21, 2011 12:24 PM**.

Use the navigation arrows to move forward or backward one time period. NOTE: The time period is determined by your selection from the **Time Range** drop-down menu.

Click **Restore to Now** to reset the time range end point to the current time.

BW AppNodes

These displays present internal JVM memory and host CPU utilization for BusinessWorks AppNodes and their resources. This is useful because the AppNode performance is dependent on both internal and external factors and they sometimes interact. Displays in this View are:

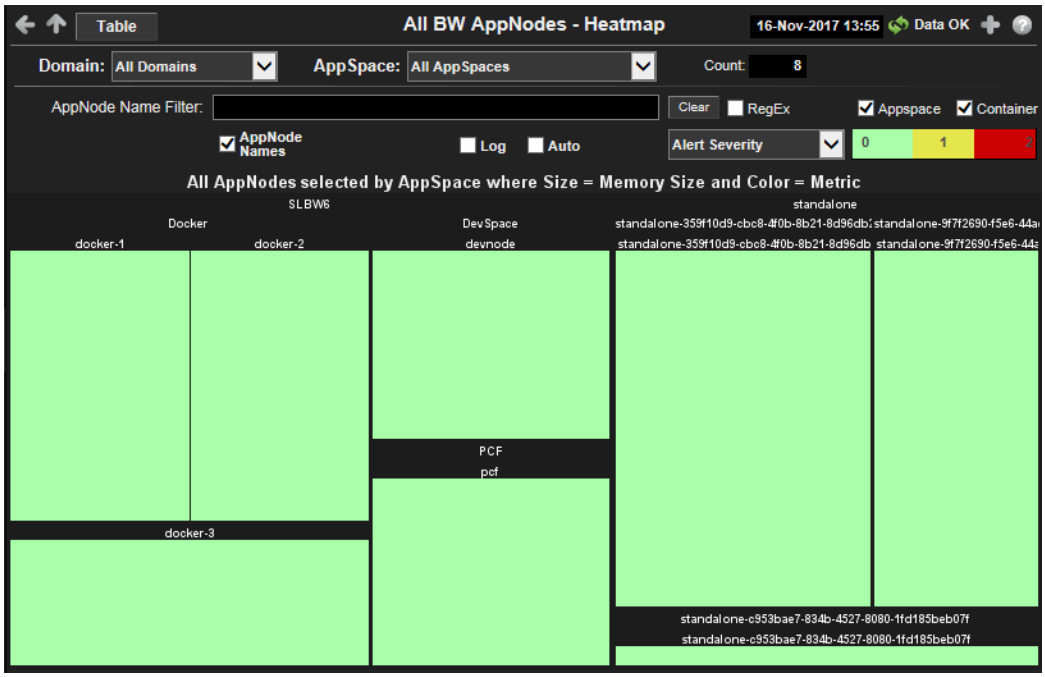
- "BW All AppNodes Heatmap" on page 227: A color-coded heatmap view of utilization metrics.
- "BW All AppNodes Table" on page 229: A tabular view of all available utilization data in this BusinessWorks View.
- "BW Single AppNode Summary" on page 231: Current and historical metrics for a single AppNode.

BW All AppNodes Heatmap

View the most critical JVM memory and host resource utilization for BusinessWorks AppNodes. Use this display to quickly identify AppNodes with critical alerts.

Each rectangle in the heatmap represents an AppNode. The rectangle color indicates the most critical alert state associated with the AppNode. The rectangle size represents the maximum memory used in the rectangle; a larger size is a larger value.

Choose a domain and AppSpace from the drop-down menus. Choose a different metric to display from the **Metric** drop-down menu. Enter a string in the **AppNode Name Filter** field to limit data shown in the display. Use the **AppNode Names** check-box ☑ to include or exclude labels in the heatmap. Mouse over a rectangle to see additional metrics. By default, this display shows **Alert Severity**. Drill-down and investigate by clicking a rectangle in the heatmap to view details for the selected application in the BW Single AppNode Summary display.



Title Bar (possible features are):

Open the previous and upper display.

Open an instance of this display in a new window.

Open the online help page for this display.

Menu, Table open commonly accessed displays.

6,047 The number of items currently in the display.

Data OK Data connection state. Red indicates the Data Server is not receiving data or the Display Server is not receiving data from the Data Server. Green indicates the data source is connected.

23-Mar-2017 12:04 Current date and time. Incorrect time might indicate the Monitor stopped running. Correct time and green **Data OK** icon is a strong indication that data is current and valid.

Open the **Alert Views - RTView Alerts Table** display.

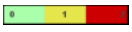
Filter By:

The display might include these filtering options:

Domain:	Choose a domain to show data for in the display.
AppSpace	Choose an AppSpace to show data for in the display.
Count:	The total number of AppNodes in the AppSpace.
AppNode Name Filter	Enter a string to limit data shown in the display.
Clear	Clears the **Application Name Filter** entries from the display.
RegEx	Toggles the **Search Text** field to accept Regular Expressions for filtering.
AppSpace	When selected, those AppNodes deployed in an AppSpace display in the heatmap.
Container	When selected, those AppNodes deployed in a container display in the heatmap.
AppNode Names	Check to include labels in the heatmap.
Log	Select to enable a logarithmic scale. Use **Log Scale** to see usage correlations for data with a wide range of values. For example, if a minority of your data is on a scale of tens, and a majority of your data is on a scale of thousands, the minority of your data is typically not visible in non-log scale graphs. **Log Scale** makes data on both scales visible by applying logarithmic values rather than actual values to the data.
Auto	Select to enable auto-scaling. When auto-scaling is activated, the color gradient

bar's maximum range displays the highest value. NOTE: Some metrics auto-scale automatically, even when **Auto** is not selected.

Metric Choose a metric to view in the display.

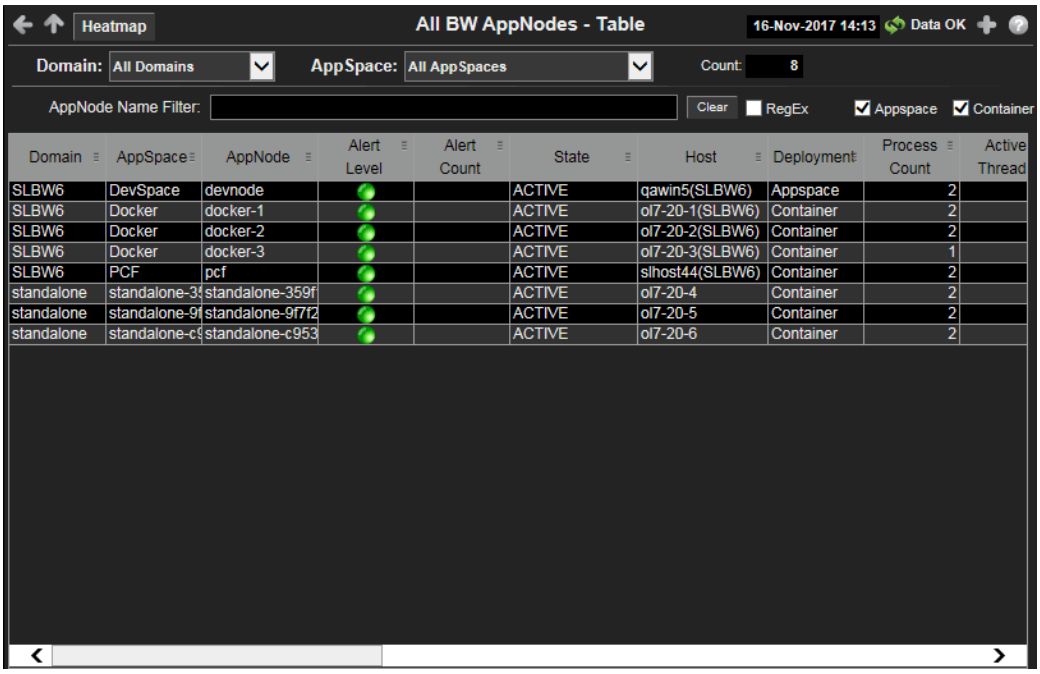
Alert Severity	The maximum level of alerts in the heatmap rectangle. Values range from **0** - **2**, as indicated in the color gradient  bar, where **2** is the highest Alert Severity: Red indicates that one or more metrics exceeded their ALARM LEVEL threshold. Yellow indicates that one or more metrics exceeded their WARNING LEVEL threshold. Green indicates that no metrics have exceeded their alert thresholds.
Alert Count	The total number of critical and warning alerts in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average alert count.
CPU Used%	The percent (%) CPU used in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Memory Used%	The percent (%) memory used in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Active Processes	The number of currently active processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Created Processes	The number of processes created in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Created/sec	The number of processes created per second in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.

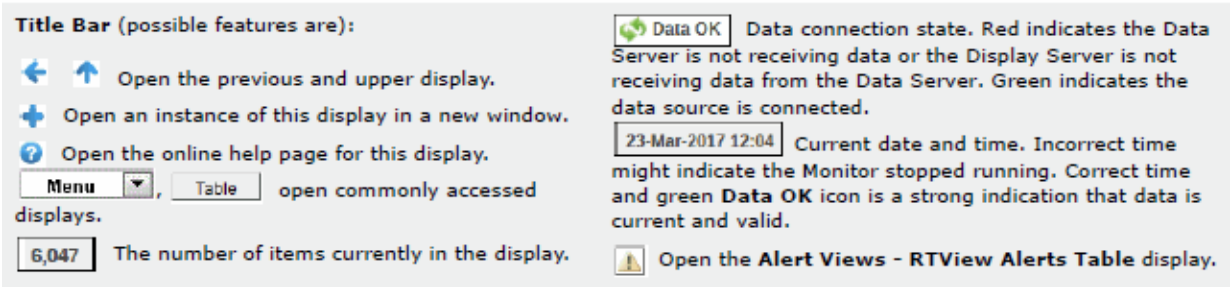
BW All AppNodes Table

View BusinessWorks data shown in the BW All AppNodes Heatmap display, and additional details, in a tabular format. Use this display to view all available data for each AppNode by Domain and AppSpace.

Each row in the table is an AppNode. Choose a domain and AppSpace from the drop-down menus. Click a column header to sort column data in numerical or alphabetical order. Enter a string in the **Application Name Filter** field to limit data shown in the display.

Drill-down and investigate by clicking a row to view details for the selected AppNode in the BW Single AppNode Summary display.





Filter By:

The display might include these filtering options:

- **Domain:** Choose a domain to show data for in the display.
- **AppSpace** Choose an AppSpace to show data for in the display.
- **Count:** The total number of rows in the table.
- **AppNode Name Filter** Enter a string to limit data shown in the display.
  - **Clear** Clears the **Application Name Filter** entries from the display.
- **RegEx** Toggles the **Search Text** field to accept Regular Expressions for filtering.
- **AppSpace** When selected, those AppNodes deployed in an AppSpace display in the AppNodes table.
- **Container** When selected, those AppNodes deployed in a container display in the AppNodes table.

Count: The total number of rows in the table.

Table:

Column values describe the AppNode.

- **Domain** The domain in which the AppNode resides.

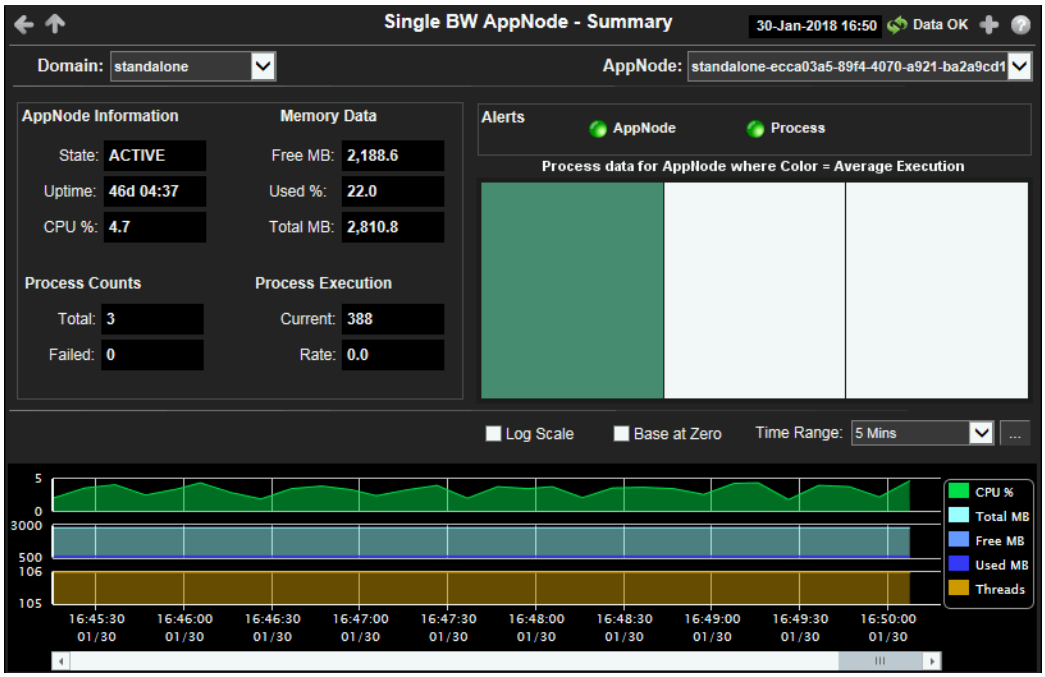
AppSpace	The AppSpace in which the AppNode resides.
AppNode	The name of the AppNode.
Alert Level	The most critical alert state for alerts in the row: Red indicates that one or more metrics exceeded their ALARM LEVEL threshold. Yellow indicates that one or more metrics exceeded their WARNING LEVEL threshold. Green indicates that no metrics have exceeded their alert thresholds.
Alert Count	The total number of active alerts for the AppNode.
Host	The host on which the AppNode resides.
Process Count	The number of processes running.
Active Threads	The number of currently active threads.
Total Memory	The total amount of used and free memory, in megabytes.
Used Memory	The amount of used memory, in megabytes.
Free Memory	The amount of free memory, in megabytes.
Used Memory%	The percent (%) used memory.
Used CPU%	The percent (%) used CPU.
System Process ID	A unique string identifier for the process.
Up Since	The date and time the AppNode was last started.
Active Processes	The number of currently active processes.
Suspended Processes	The number of suspended application processes.
Failed Processes	The number of failed application processes.
Completed Processes	The number of completed application processes.
Created /sec	The number of application processes created per second.
Suspended /sec	The number of application processes suspended per second.
Failed /sec	The number of failed application processes per second.
Exec Time /sec	The number of application processes executed per second.
Recent Exec Time	The number of seconds for the most recently executed process.
Average Exec Time	The average number of seconds for all processes to execute.
Time Stamp	The date and time the row data was last updated.
Source	Name of RTView Data Server sending this data (or localhost).

BW Single AppNode Summary

View current and historical utilization and performance metrics for a single BusinessWorks AppNode. Use this display to investigate performance issues on an AppNode.

This display includes a heatmap showing most critical alerts pertaining to process execution, and trend graphs tracing CPU utilization and thread count.

Choose a domain, AppSpace and AppNode from the drop-down menus. Use the **Time-Range** to "zoom-in" or "zoom-out" on a specific time frame in the trend graph. Drill-down and investigate by clicking an AppNode in the table to view details in the BW Single AppNode Summary display.



Title Bar (possible features are):

Open the previous and upper display.

Open an instance of this display in a new window.

Open the online help page for this display.

Menu, Table open commonly accessed displays.

6,047 The number of items currently in the display.

Data OK Data connection state. Red indicates the Data Server is not receiving data or the Display Server is not receiving data from the Data Server. Green indicates the data source is connected.

23-Mar-2017 12:04 Current date and time. Incorrect time might indicate the Monitor stopped running. Correct time and green **Data OK** icon is a strong indication that data is current and valid.

Open the **Alert Views - RTView Alerts Table** display.

Filter By:

The display might include these filtering options:

Domain:	Choose a domain to show data for in the display.
AppSpace	Choose an AppSpace to show data for in the display.
AppNode:	Choose an AppNode to show data for in the display.

Fields and Data:

AppNode Information	**Uptime:**	The number of days, hours and minutes since the AppNode started.
	CPU%	The percent (%) CPU used on the AppNode.
	Threads:	The number of currently active threads for the AppNode.
Memory Data	**Free:**	The amount of available memory on the AppNode.
	Used%	The percent (%) memory used on the AppNode.
	Total	The total amount of memory on the AppNode.
Process Counts	**Total:**	The number of currently active processes for the AppNode.
	Failed:	The number of failed processes for the AppNode.
Process Execution	**Current**	The number of processes executed by the AppNode.

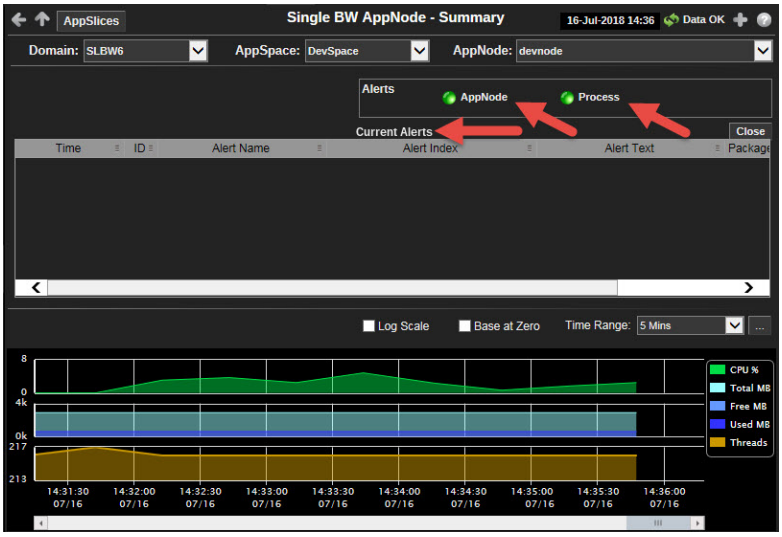
Rate: The number of processes executed per second.

Indicates the greatest severity level and the number of open **AppNode** and **Process** alerts for the selected AppNode. Values range from **0** to **2**, where **2** is the greatest Severity:

- One or more alerts exceeded their ALARM LEVEL threshold.
- One or more alerts exceeded their WARNING LEVEL threshold.
- No alert thresholds have been exceeded.

Click on the alert indicator to display a table listing the current alerts for the selected AppNode. Click the **Close** button (for the current alerts table) to close the table.

Alerts



Heatmap

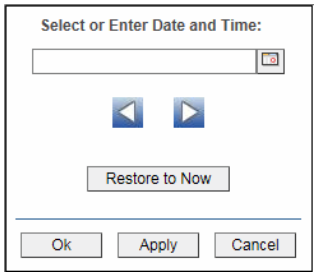
Each rectangle in the heatmap represents an AppSlice. The rectangle color indicates the most critical **Average Exec Time** alert state associated with the AppSlice. The rectangle size represents the maximum number of processes executed in the rectangle; a larger size is a larger value. Click a rectangle to drill-down and investigate in the BW Single AppSlice Summary display.

Trend Graphs

Traces the sum of process metrics across all processes for all applications on the AppNode.

- **CPU%**: The percent (%) CPU used on the AppNode.
- **Total MB**: The amount of memory used.
- **Free MB**: The amount of available memory.
- **Used MB**: The amount of used memory.
- **Threads**: The number of threads.

Log Scale	Select to enable a logarithmic scale. Use **Log Scale** to see usage correlations for data with a wide range of values. For example, if a minority of your data is on a scale of tens, and a majority of your data is on a scale of thousands, the minority of your data is typically not visible in non-log scale graphs. **Log Scale** makes data on both scales visible by applying logarithmic values rather than actual values to the data.
Base at Zero	Select to use zero (**0**) as the Y axis minimum for all graph traces.
Time Range	Select a time range from the drop down menu varying from **2 Minutes** to **Last 7 Days**, or display **All Data**. To specify a time range, click Calendar .



By default, the time range end point is the current time. To change the time range end point, click Calendar and select a date and time from the calendar or enter the date and time in the text field using the following format: **MMM dd, YYYY HH:MM**. For example, **Aug 21, 2011 12:24 PM**.

Use the navigation arrows to move forward or backward one time period. NOTE: The time period is determined by your selection from the **Time Range** drop-down menu.

Click **Restore to Now** to reset the time range end point to the current time.

BW AppSlices

These displays present process metrics totaled by Application and AppNode for AppSlices. This is useful to see how the application is distributed and how each part of it is performing. The AppSlice is the part of an application running on a specific AppNode when the application is deployed to multiple AppNodes. Displays in this View are:

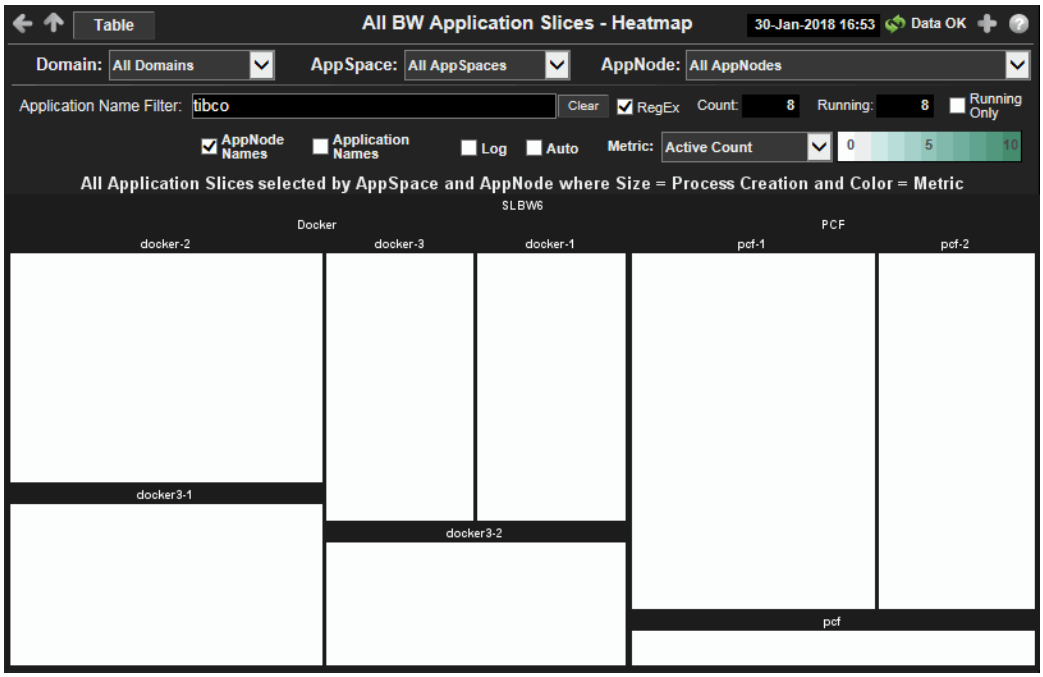
- "BW All AppSlices Heatmap" on page 234: A color-coded heatmap view of process creation and execution metrics.
- "BW All AppSlices Table" on page 237: A tabular view of all available data in this BusinessWorks View.
- "BW Single AppSlice Summary" on page 239: Current and historical metrics for a single AppSlice.

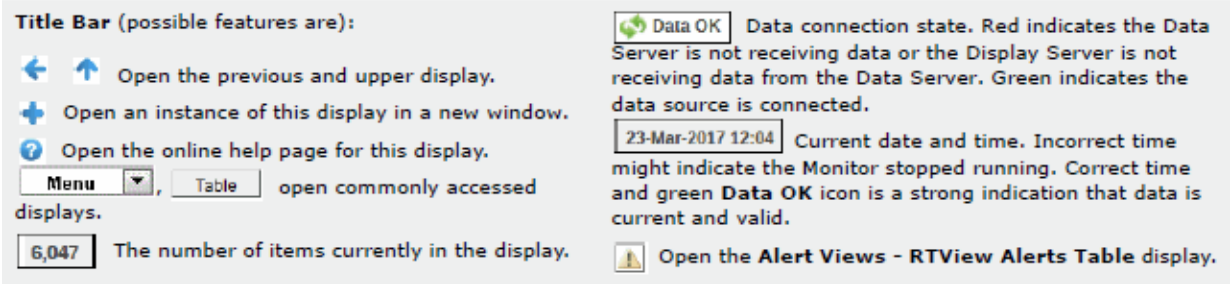
BW All AppSlices Heatmap

View the most critical performance metrics for BusinessWorks AppSlices. Use this display to quickly identify AppSlices with high process execution numbers.

Each rectangle in the heatmap represents an AppSlice. The rectangle color indicates the process execution numbers for the AppSlice. The rectangle size represents the number of processes created in the rectangle; a larger size is a larger value.

Choose a domain, AppSpace and AppNode from the drop-down menus. Choose a different metric to display from the **Metric** drop-down menu. Enter a string in the **Application Name Filter** field to limit data shown in the display. Use the **AppNode Names** and **Application Names** check-boxes ☑ to include or exclude labels in the heatmap. Mouse over a rectangle to see additional metrics. By default, this display shows **Active Count**. Drill-down and investigate by clicking a rectangle in the heatmap to view details for the selected application in the BW Single AppSlice Summary display.





Filter By:

The display might include these filtering options:

- **Domain:** Choose a domain to show data for in the display.
- **AppSpace** Choose an AppSpace to show data for in the display.
- **AppNode:** Choose an AppNode to show data for in the display.

Fields and Data:

- **Application Name Filter** Enter a string to limit data shown in the display.
  - **Clear** Clears the **Application Name Filter** entries from the display.
- **RegEx** Toggles the **Search Text** field to accept Regular Expressions for filtering.
- **Count** The number of AppNodes in the display.
- **Running** The total number of AppSpaces currently running in the display.
- **Running Only** Select to show only running applications in the display.
- **AppNode Names** Check to include labels in the heatmap.
- **Application Names** Check to include labels in the heatmap.
- **Log** Select to enable a logarithmic scale. Use **Log Scale** to see usage correlations for

data with a wide range of values. For example, if a minority of your data is on a scale of tens, and a majority of your data is on a scale of thousands, the minority of your data is typically not visible in non-log scale graphs. **Log Scale** makes data on both scales visible by applying logarithmic values rather than actual values to the data.

Auto Select to enable auto-scaling. When auto-scaling is activated, the color gradient bar's maximum range displays the highest value. NOTE: Some metrics auto-scale automatically, even when **Auto** is not selected.

Metric Choose a metric to view in the display.

Active Count The total number of active processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.

Completed Count The total number of completed processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.

Suspended Count The total number of suspended processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.

Failed Count The total number of failed processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.

Created / sec The number of processes created per second in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.

Suspended / sec The number of suspended processes per second in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.

Failed / sec The number of failed processes per second in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.

Exec Time / sec The process execution time per second in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.

Most Recent Exec Time The execution time for the most recently executed process in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The

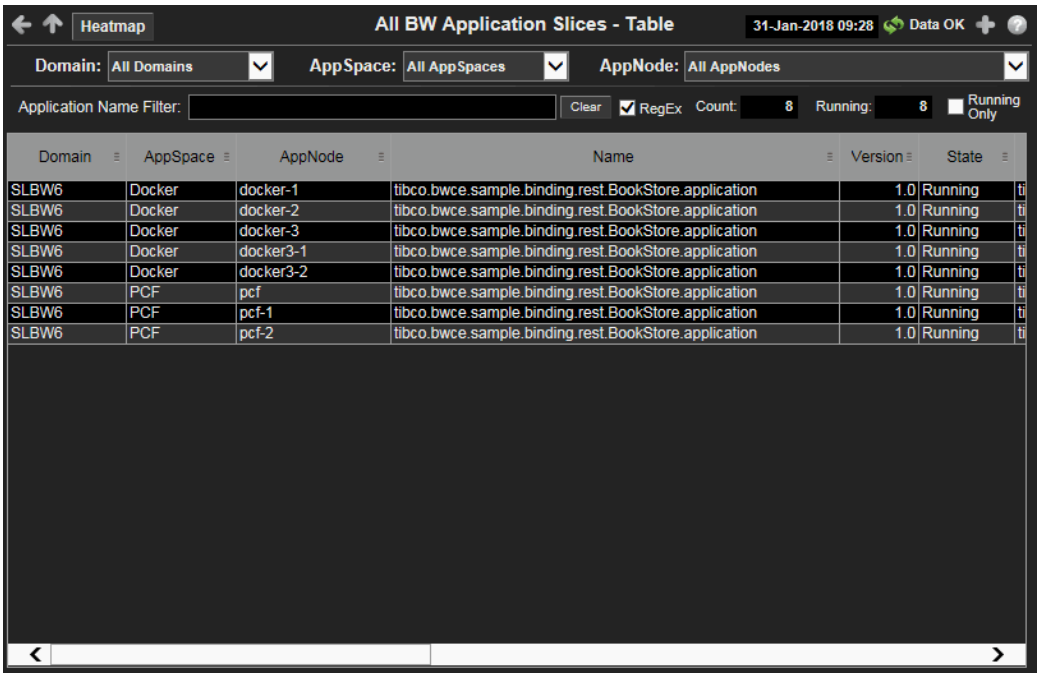
	numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Average Exec Time	The average execution time for all processes in the heatmap rectangle, calculated by dividing the delta execution time for the interval by the delta completed, or the number of process instances that completed in the interval. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Average Elapsed Time	The average elapsed time for all processes in the heatmap rectangle, calculated by dividing the delta elapsed time for the interval by the delta completed, or the number of process instances that completed in the interval. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.

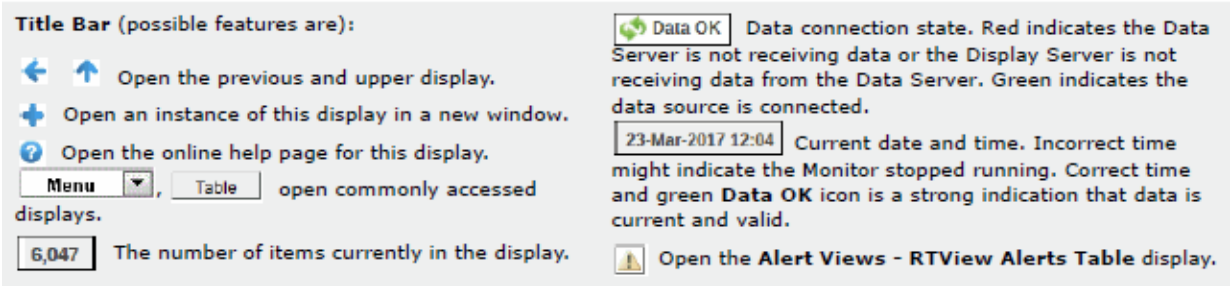
BW All AppSlices Table

View BusinessWorks data shown in the BW All AppSlices Heatmap, and additional details, in a tabular format.

Each row in the table is an AppSlice. Choose a domain (or **All Domains**), an AppSpace (or **All AppSpaces**) and an AppNode (or **All AppNodes**) from the drop-down menus. Click a column header to sort column data in numerical or alphabetical order. Enter a string in the **Application Name Filter** field to limit data shown in the display.

Drill-down and investigate by clicking a row to view details in the BW Single AppSlice Summary display.





Filter By:

The display might include these filtering options:

Domain:	Choose a domain to show data for in the display.
AppSpace	Choose an AppSpace to show data for in the display.
AppNode	Choose an AppNode to show data for in the display.
Application Name Filter	Enter a string to limit data shown in the display.
Clear	Clears the **Application Name Filter** entries from the display.
RegEx	Toggles the **Application Name Filter** to accept Regular Expressions for filtering.

Fields and Data:

Count:	The total number of rows in the table.
Running	The total number of applications currently running in the AppSpace.
Running Only	Select to show only running applications in the display.

Table:

Each row in the table is a different AppNode.

Domain	The domain in which the AppSpace resides.
AppSpace	The AppSpace the AppNode is associated with.
AppNode	The name of the selected AppNode.
Name	The name of the application.
Version	The application version.
State	The current status of the application. Valid values are **Running** and **Stopped**.
Module	The application module.
Shared Module	The shared module, if any.
Active Processes	The number of currently active application processes.
Suspended Processes	The number of suspended application processes.
Failed Processes	The number of failed application processes.
Completed Processes	The number of completed application processes.
Created /sec	The number of application processes created per second.
Suspended /sec	The number of application process suspensions per second.
Failed /sec	The number of application process failures per second.
Exec Time /sec	The number of processes executed per second.
Recent Exec Time /sec	The number of seconds for the most recently executed process.

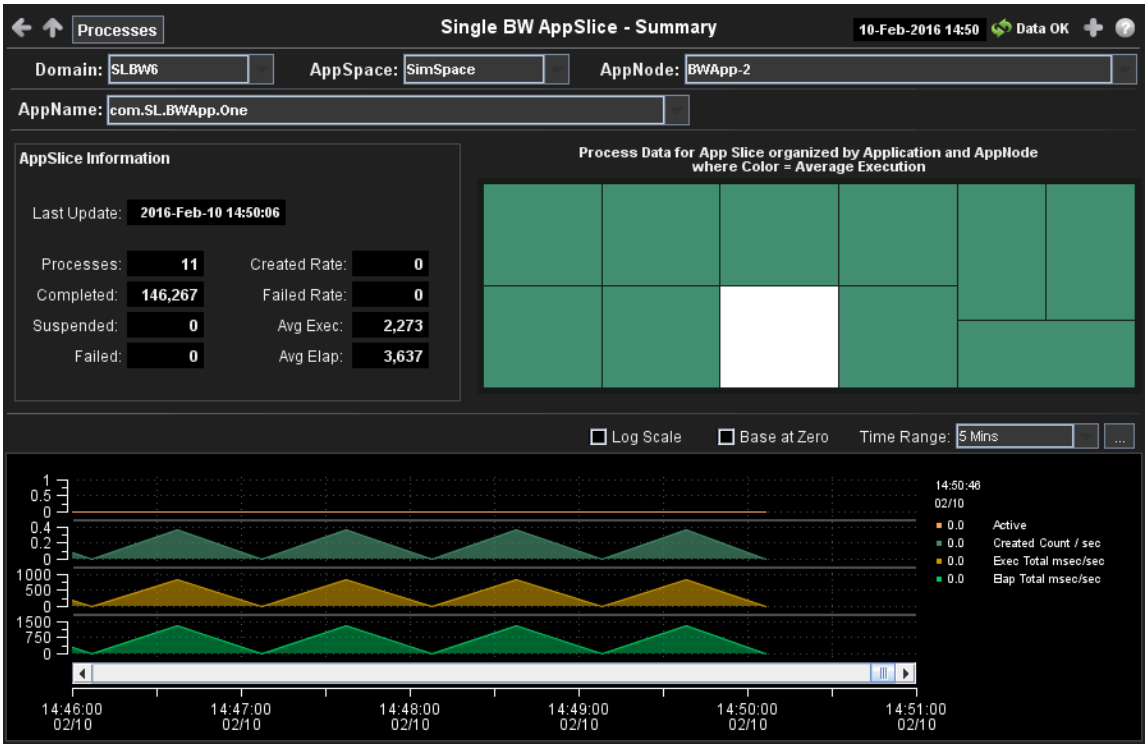
Average Exec Time	The average number of seconds for all processes to execute.
Time Stamp	The date and time the row data was last updated.
Source	Name of RTView Data Server sending this data (or localhost).

BW Single AppSlice Summary

View current and historical utilization and performance metrics for a single BusinessWorks AppSlice. Use this display to investigate performance issues on an AppSlice level.

This display includes a heatmap showing the most critical process execution alerts for AppSlices on the selected application, and trend graphs tracing process execution times.

Choose a domain, AppSpace, AppNode and AppNode from the drop-down menus. Use the **Time-Range** to "zoom-in" or "zoom-out" on a specific time frame in the trend graph. Drill-down and investigate by clicking a process in the heatmap to view details in the BW Single Process Summary display.



Title Bar (possible features are):

Open the previous and upper display.

Open an instance of this display in a new window.

Open the online help page for this display.

Menu, Table open commonly accessed displays.

6,047 The number of items currently in the display.

Data OK Data connection state. Red indicates the Data Server is not receiving data or the Display Server is not receiving data from the Data Server. Green indicates the data source is connected.

23-Mar-2017 12:04 Current date and time. Incorrect time might indicate the Monitor stopped running. Correct time and green **Data OK** icon is a strong indication that data is current and valid.

Open the **Alert Views - RTView Alerts Table** display.

Filter By:

The display might include these filtering options:

Domain: Choose a domain to show data for in the display.

AppSpace Choose an AppSpace to show data for in the display.

AppNode: Choose an AppNode to show data for in the display.

AppName: Choose an AppName to show data for in the display.

Fields and Data:

AppSlice Information

Last Update: The date and time the data was last updated.

Processes The number of active processes.

Completed: The total number of completed processes summed across all processes in one AppSlice of the application.

Suspended: The total number of suspended processes

Failed: The total number of failed processes

Created Rate: The number of application processes created per second.

Failed Rate: The number of failed application processes per second.

Avg Exec: The average number of seconds for processes to execute.

Avg Elap: The average amount of elapsed time for processes, in seconds.

Heatmap

Each rectangle in the heatmap represents one process in an AppSlice. The rectangle color indicates the most critical **Average Exec Time** alert state associated with the AppSlice. The rectangle size represents the processes execution time in the rectangle; a larger size is a larger value. Click a rectangle to drill-down and investigate in the BW Single Process Summary display.

Trend Graphs

Traces the sum across all processes in one AppSlice of the application.

- **Active**: Traces the number of active processes.
- **Created Count**: Traces the number of processes created.
- **Exec Total msec/sec**: Traces the rate at which the application accumulates process execution time, in milliseconds per second.
- **Elap Total msec/sec**: Traces the rate at which the application is accumulating process elapsed time, in milliseconds per second.

Log Scale

Select to enable a logarithmic scale. Use **Log Scale** to see usage correlations for data with a wide range of values. For example, if a minority of your data is on a scale of tens, and a majority of your data is on a scale of thousands, the minority of your data is typically not visible in non-log scale graphs. **Log Scale** makes data on both scales visible by applying logarithmic values rather than actual values to the data.

Base at Zero Select to use zero (**0**) as the Y axis minimum for all graph traces.

Time Range

Select a time range from the drop down menu varying from **2 Minutes** to **Last 7 Days**, or display **All Data**. To specify a time range, click Calendar .

By default, the time range end point is the current time. To change the time range end point, click Calendar and select a date and time from the calendar or enter the date and time in the text field using the following format: **MMM dd, YYYY HH:MM**. For example, **Aug 21, 2011 12:24 PM**.

Use the navigation arrows to move forward or backward one time period.
NOTE: The time period is determined by your selection from the **Time Range** drop-down menu.

Click **Restore to Now** to reset the time range end point to the current time.

BW Processes

These displays present performance data for BusinessWorks processes. Use these displays to verify that individual BusinessWorks processes are executing and using resources as expected. Displays in this View are:

- "BW All Processes Heatmap" on page 241: A color-coded heatmap view of selected process performance metrics.
- "BW All Processes Table" on page 244: A tabular view of all available process performance data in this BusinessWorks View.
- "BW Single Process Summary" on page 247: Current and historical metrics for a single process.

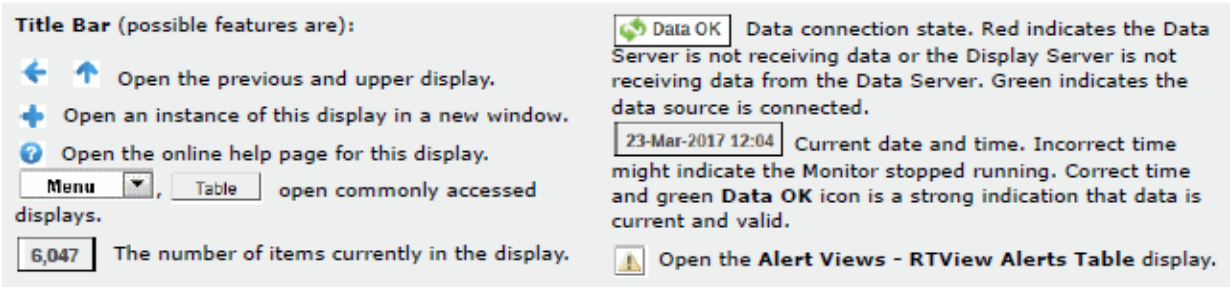
BW All Processes Heatmap

View the most critical BusinessWorks alerts pertaining to process creation and execution. Use this display to quickly identify processes with critical alerts.

Each rectangle in the heatmap represents a process. The rectangle color indicates the most critical alert state associated with the processes (the rectangle size is uniform for all processes.)

Choose a domain, applications, AppNode and AppSpace from the drop-down menus. Choose a different metric to display from the **Metric** drop-down menu. Enter a string in the **Process Name Filter** field to limit data shown in the display. Use the **Application Names** and **Process Names** check-boxes ☑ to include or exclude labels in the heatmap. Mouse over a rectangle to see additional metrics. By default, this display shows **Alert Severity**.

Drill-down and investigate by clicking a rectangle in the heatmap to view details in the BW Single Process Summary display.



Filter By:

The display might include these filtering options:

- **Domain:** Choose a domain to show data for in the display.
- **AppSpace** Choose an AppSpace to show data for in the display.
- **AppNode:** Choose an AppNode to show data for in the display.
- **AppName** Choose an AppName to show data for in the display.
- **Count:** The total number of processes currently shown in the display.

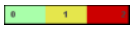
Fields and Data:

- **Process Name Filter** Enter a string to limit data shown in the display.
  - **Clear** Clears the **Processes Name Filter** entries from the display.
- **RegEx** Toggles the **Processes Name Filter** to accept Regular Expressions for filtering.
- **Application Names** Check to include labels in the heatmap.
- **Process Names** Check to include labels in the heatmap.
- **Log** Select to enable a logarithmic scale. Use **Log Scale** to see usage correlations for

data with a wide range of values. For example, if a minority of your data is on a scale of tens, and a majority of your data is on a scale of thousands, the minority of your data is typically not visible in non-log scale graphs. **Log Scale** makes data on both scales visible by applying logarithmic values rather than actual values to the data.

Auto Select to enable auto-scaling. When auto-scaling is activated, the color gradient bar's maximum range displays the highest value. NOTE: Some metrics auto-scale automatically, even when **Auto** is not selected.

Metric Choose a metric to view in the display.

Alert Severity The maximum level of alerts in the heatmap rectangle. Values range from **0** - **2**, as indicated in the color gradient  bar, where **2** is the highest Alert Severity:

- Red indicates that one or more metrics exceeded their ALARM LEVEL threshold.
- Yellow indicates that one or more metrics exceeded their WARNING LEVEL threshold.
- Green indicates that no metrics have exceeded their alert thresholds.

Alert Count The total number of critical and warning alerts in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average alert count.

Active Count The total number of active processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.

Completed Count The total number of completed processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.

Suspended Count The total number of suspended processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.

Failed Count The total number of failed processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.

Created / sec The number of processes created per second in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.

Suspended / sec The number of suspended processes per second in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar

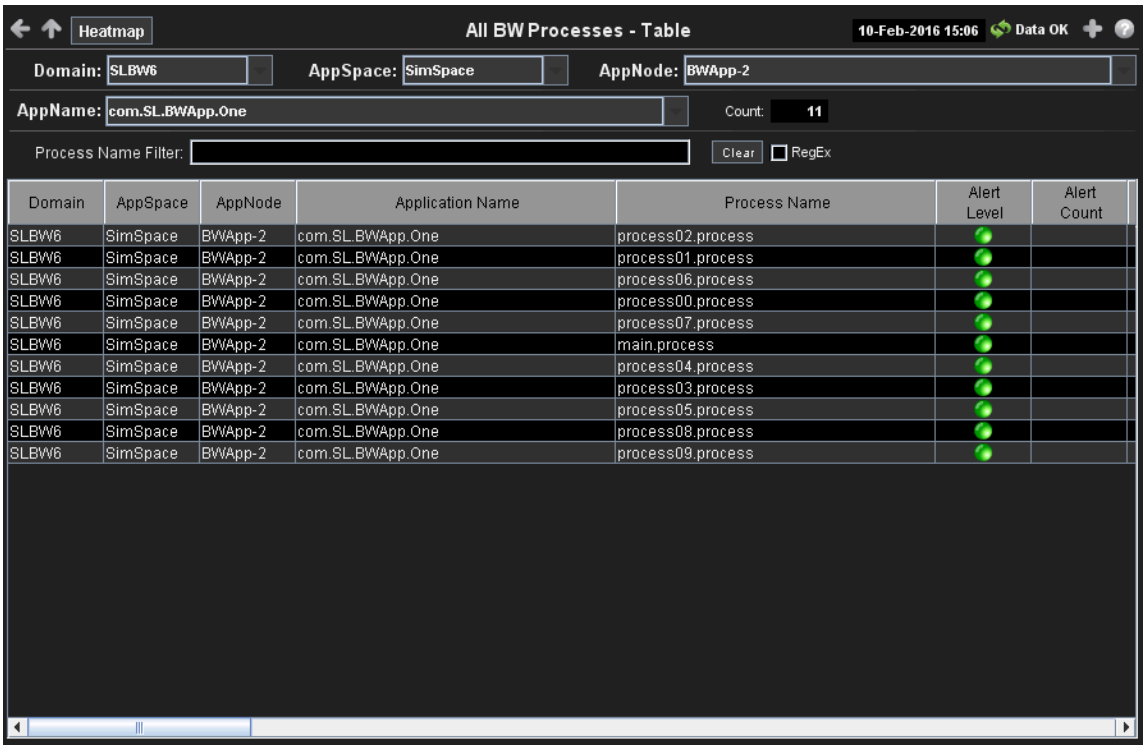
	indicates the average count.
Failed / sec	The number of failed processes per second in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Exec Time / sec	The process execution time per second in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Most Recent Exec Time	The execution time for the most recently executed process in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Average Exec Time	The average execution time for all processes in the heatmap rectangle, calculated by dividing the delta execution time for the interval by the delta completed, or the number of process instances that completed in the interval. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Most Recent Elapsed Time	The elapsed time for the most recent process in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Average Elapsed Time	The average elapsed time for all processes in the heatmap rectangle, calculated by dividing the delta elapsed time for the interval by the delta completed, or the number of process instances that completed in the interval. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.

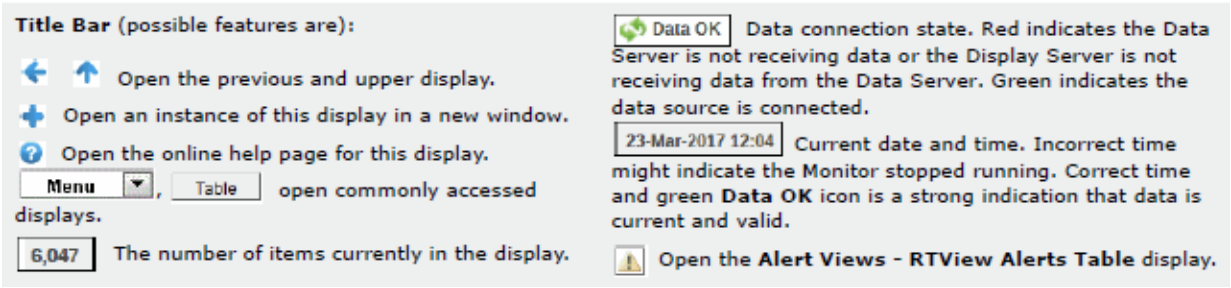
BW All Processes Table

View BusinessWorks data shown in the BW All Applications Heatmap, and additional details, in a tabular format.

Each row in the table is a process. Choose a domain, applications, AppNode and AppSpace from the drop-down menus. Click a column header to sort column data in numerical or alphabetical order. Enter a string in the **Process Name Filter** field to limit data shown in the display.

Drill-down and investigate by clicking a row to view details for the selected process in the BW Single Process Summary display





Filter By:

The display might include these filtering options:

- **Domain:** Choose a domain to show data for in the display.
- **AppSpace** Choose an AppSpace to show data for in the display.
- **AppNode:** Choose an AppNode to show data for in the display.
- **AppName** Choose an AppName to show data for in the display.

Fields and Data:

- **Count:** The total number of processes in the AppSpace.
- **Process Name Filter** Enter a string to limit data shown in the display.
  - **Clear** Clears the **Application Name Filter** entries from the display.
- **RegEx** Toggles the **Application Name Filter** to accept Regular Expressions for filtering.

Table:

Each row in the table is a different AppSlice. Column values are associated with the process.

- **Domain** The domain in which the process resides.
- **AppSpace** The AppSpace in which the process resides.

AppNode	The AppSpace in which the process resides.
Application Name	The name of the application in which the process is running.
Process Name	The name of the process.
Alert Level	The most critical alert state for alerts in the row: Red indicates that one or more metrics exceeded their ALARM LEVEL threshold. Yellow indicates that one or more metrics exceeded their WARNING LEVEL threshold. Green indicates that no metrics have exceeded their alert thresholds.
Alert Count	The total number of active alerts for the process.
Total Exec Time	Total execution time (in milliseconds) for all successfully completed process instances.
Delta Exec Time	Execution time accumulated during the current polling period.
Exec Time/sec	Delta execution time per second.
Recent Exec Time	Execution time (in milliseconds) of the most recently completed process instance.
Total Elapsed Time	Total elapsed time (in milliseconds) for all successfully completed process instances.
Delta Elapsed Time	Elapsed time accumulated during the current polling period.
Elapsed Time/sec	Delta elapsed time per second.
Recent Elapsed Time	Elapsed clock time (in milliseconds) of the most recently completed process instance.
Active	The number of currently active processes
Created	The number of processes created.
Suspended	The number of process suspensions.
Failed	The number of process failures.
Completed	The number of completed processes.
Delta Active	The number of active processes since the last data update.
Active/sec	The number of active processes per second.
Delta Created	The number of created processes since the last data update.
Created/sec	The number of created processes per second.
Delta Suspended	The number of suspended processes since the last data update.
Suspended/sec	The number of suspended processes per second.
Delta Completed	The number of completed processes since the last data update.
Completed/sec	The number of completed processes per second.
Delta Failed	The number of failed processes since the last data update.
Failed/sec	The number of failed processes per second.
Min Exec Time	Execution time (in milliseconds) of the process instance that has completed in the shortest amount of execution time.
Max Exec Time	Execution time (in milliseconds) of the process instance that has completed in the longest amount of execution time.
Average Exec Time	Average execution time (in milliseconds) for all successfully completed process instances.
Min Elapsed Time	Elapsed clock time (in milliseconds) of the process instance that has completed in the shortest amount of elapsed time.
Max Elapsed Time	Elapsed clock time (in milliseconds) of the process instance that has completed in the longest amount of elapsed time.
Average Elapsed Time	Average elapsed clock time (in milliseconds) for all successfully completed process instances.

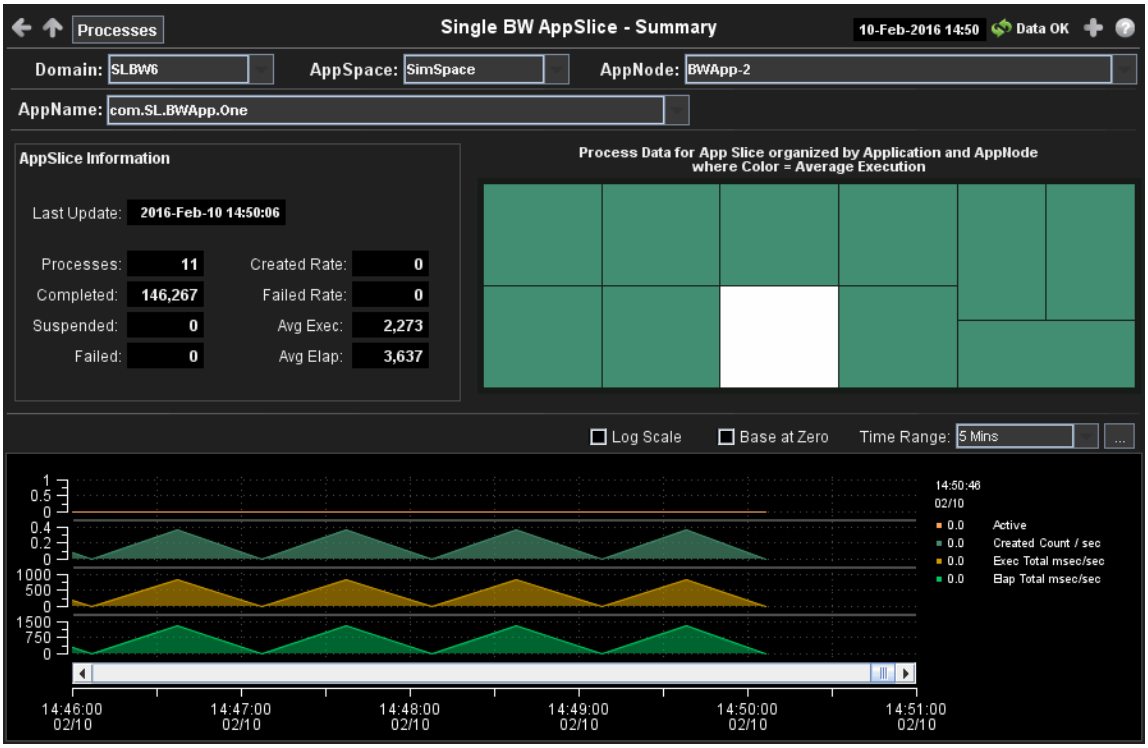
Count Since Reset	The number of times the process has executed since statistics were reset.
Main Process	The name of the main process.
Application Version	The application version.
Module Name	The application module.
Module Version	The module version.
Time Stamp	The date and time the row data was last updated.
Source	Name of RTView Data Server sending this data (or localhost).

BW Single Process Summary

View current and historical execution metrics for a single BusinessWorks process. Use this display to investigate performance issues for a process.

This display includes trend graphs tracing process and activity execution counts and times.

Choose a domain, application, AppNode and AppSpace from the drop-down menus. Use the **Time-Range** to "zoom-in" or "zoom-out" on a specific time frame in the trend graph.



Title Bar (possible features are):

Open the previous and upper display.

Open an instance of this display in a new window.

Open the online help page for this display.

Menu, Table open commonly accessed displays.

6,047 The number of items currently in the display.

Data OK Data connection state. Red indicates the Data Server is not receiving data or the Display Server is not receiving data from the Data Server. Green indicates the data source is connected.

23-Mar-2017 12:04 Current date and time. Incorrect time might indicate the Monitor stopped running. Correct time and green **Data OK** icon is a strong indication that data is current and valid.

Open the **Alert Views - RTView Alerts Table** display.

Filter By:

The display might include these filtering options:

Domain:	Choose a domain to show data for in the display.
AppSpace	Choose an AppSpace to show data for in the display.
AppNode:	Choose an AppNode to show data for in the display.
AppName	Choose an application to show data for in the display.
Process	Choose a process to show data for in the display.

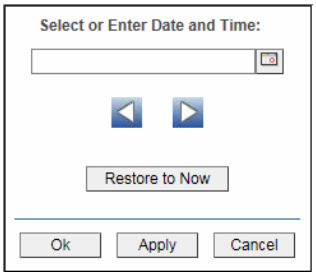
Fields and Data:

Activity Count:		The number of activities defined for the process.
Main Process:		The name of the main process.
Active		Number of active instances for this process definition. This number is calculated using the Hawk method named GetProcesses. This method returns information about process instances that are active at the time of update. The value here displays the current total count of all active instances discovered for this process definition. The trend below displays the same value over time.
Active/sec		The number of currently active application processes per second.
Created	**Total**	The number of process instances created for this process definition.
	Current	The number of process instances created this update cycle.
	Rate	The number of process instances created per second.
Completed	**Total**	The number of process instances completed for this process definition.
	Current	The number of process instances completed this update cycle.
	Rate	The number of process instances completed per second.
Errors	**Total**	The number of errors accumulated by all process instances.
	Current	The number of errors accumulated this update cycle.
	Rate	The number of errors accumulated per second.
Execution	**Min**	The shortest execution time of any process instance, in milliseconds.
	Max	The longest execution time of any process instance, in milliseconds.
	Average	The average execution time for all completed process instances, in milliseconds.
	Current	The amount of time accumulated this update cycle.
	Rate	The amount of time accumulated per second.
Elapsed	**Min**	The shortest elapsed time of any process instance, in milliseconds.
	Max	The longest elapsed time of any process instance, in milliseconds.
	Average	The average elapsed time for all completed process instances, in milliseconds.
	Current	The amount of elapsed time accumulated this update cycle.
	Rate	The amount of elapsed time accumulated per second.

Trend Graphs

Traces application process and activity metrics for the selected process.

- **Active Count**: Traces the number of currently active processes.
- **Created Count:** Traces the number of created processes.
- **Process Elapsed Time/sec**: Traces the rate at which the application is accumulating process elapsed time, in milliseconds per second.
- **Process Exec Time/sec**: Traces the rate at which the application is accumulating process execution time, in milliseconds per second.
- **All Activities Exec Count/sec**: Traces the number of executed activities per second.
- **All Activities Exec Time/sec**: Traces the amount of execution time for executed activities per second.

Log Scale	Select to enable a logarithmic scale. Use **Log Scale** to see usage correlations for data with a wide range of values. For example, if a minority of your data is on a scale of tens, and a majority of your data is on a scale of thousands, the minority of your data is typically not visible in non-log scale graphs. **Log Scale** makes data on both scales visible by applying logarithmic values rather than actual values to the data.
Base at Zero	Select to use zero (**0**) as the Y axis minimum for all graph traces.
Time Range	Select a time range from the drop down menu varying from **2 Minutes** to **Last 7 Days**, or display **All Data**. To specify a time range, click Calendar .  By default, the time range end point is the current time. To change the time range end point, click Calendar and select a date and time from the calendar or enter the date and time in the text field using the following format: **MMM dd, YYYY HH:MM**. For example, **Aug 21, 2011 12:24 PM**. Use the navigation arrows to move forward or backward one time period. NOTE: The time period is determined by your selection from the **Time Range** drop-down menu. Click **Restore to Now** to reset the time range end point to the current time.

BW5 Engines

These displays present performance metrics for BW5 Engines. Displays in this View are:

- "All Engines Heatmap " on page 249:Performance metrics of CPU and memory utilization for all BW Engines.
- "All Engines Table " on page 252: Available metrics from the Hawk microagent for each BW Engine.
- "All Engines Grid " on page 254: Displays the main health metrics and a single trend graph per engine, summarizing the status of each BW Engine.
- "Single Engine Summary " on page 256: Detailed performance metrics and alert status for a single BW Engine.

All Engines Heatmap

Quick view of BW5 Engines status for the selected **Filter** and **Server**. Each rectangle in the heatmap represents an engine. Rectangle size represents Max Heap Size and the color represents the most severe value in the heatmap rectangle is shown for the selected Metric. By default, the maximum **Alert Severity** is shown:

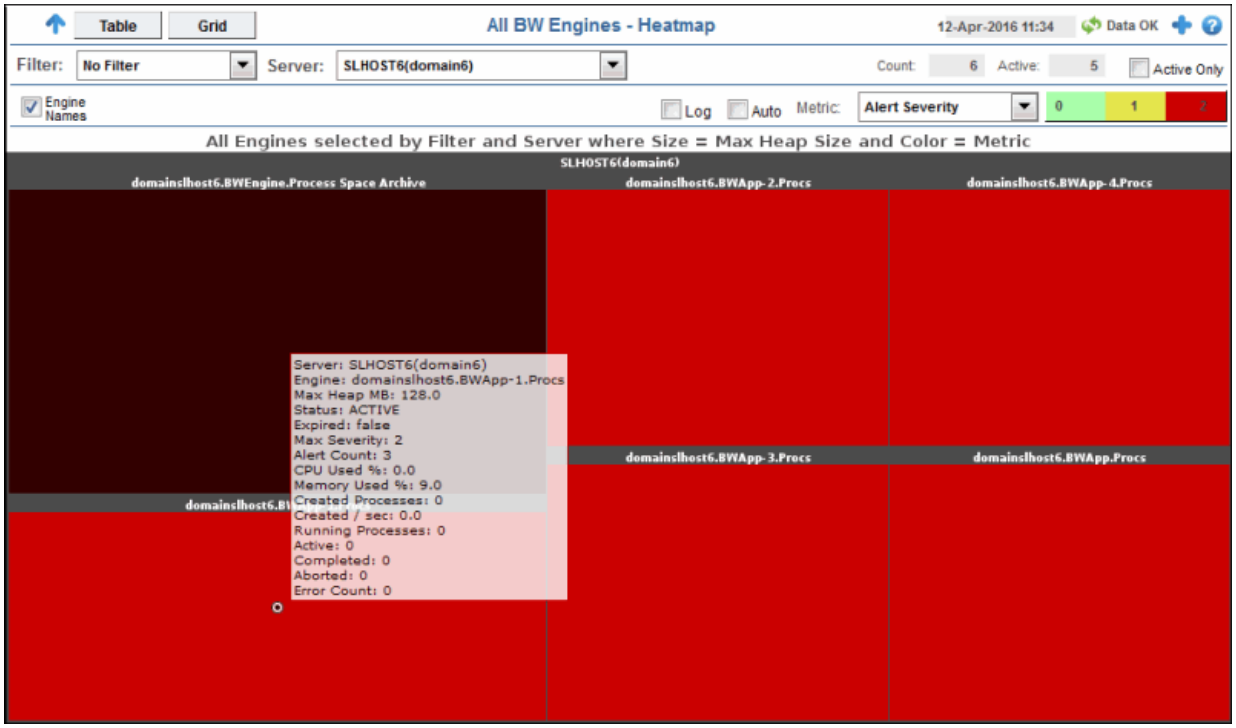
Values range from **0** - **2**, as indicated in the color gradient bar, where **2** is the highest Alert Severity:

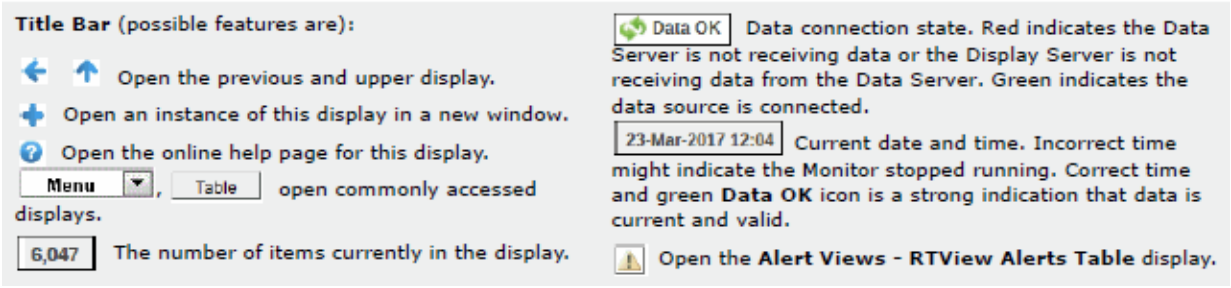
- Red indicates that one or more metrics exceeded their ALARM LEVEL threshold.
- Yellow indicates that one or more metrics exceeded their WARNING LEVEL threshold.
- Green indicates that no metrics have exceeded their alert thresholds.

Mouseover to see the following performance metrics:

- **Server**: Server agent name.
- **Engine**: Engine name.
- **Max Heap MB**: Maximum heap allocated to this engine for the JVM.
- **Status**: ACTIVE, STOPPED or LIMITED.
- **Expired**: When checked, data has not been received from this host in the specified amount of time.
- **Alert Count**: Number of current alerts
- **CPU Used %**: Percent of server CPU used by engine.
- **Memory Used %**: Percentage of allocated memory currently consumed by this engine from within the JVM. Equal to the value of: (100*UsedBytes) divided by MaxBytes. NOTE: Percent used is Long.
- **Created Processes**: The total number of processes created.
- **Created / sec**: The number of processes created per second.
- **Running Processes**: The number of currently running processes.
- **Active**: The number of currently active processes.
- **Completed**: The total number of completed processes.
- **Aborted**: The total number of aborted processes.
- **Error Count**: The total number of errors.

Click on a node to drill down to the Single Engine Summary display to look at number of processes running, threads, history of memory utilization and other performance metrics for a specific engine. Mouse-over nodes to view details about engine performance and status.





Filter By:

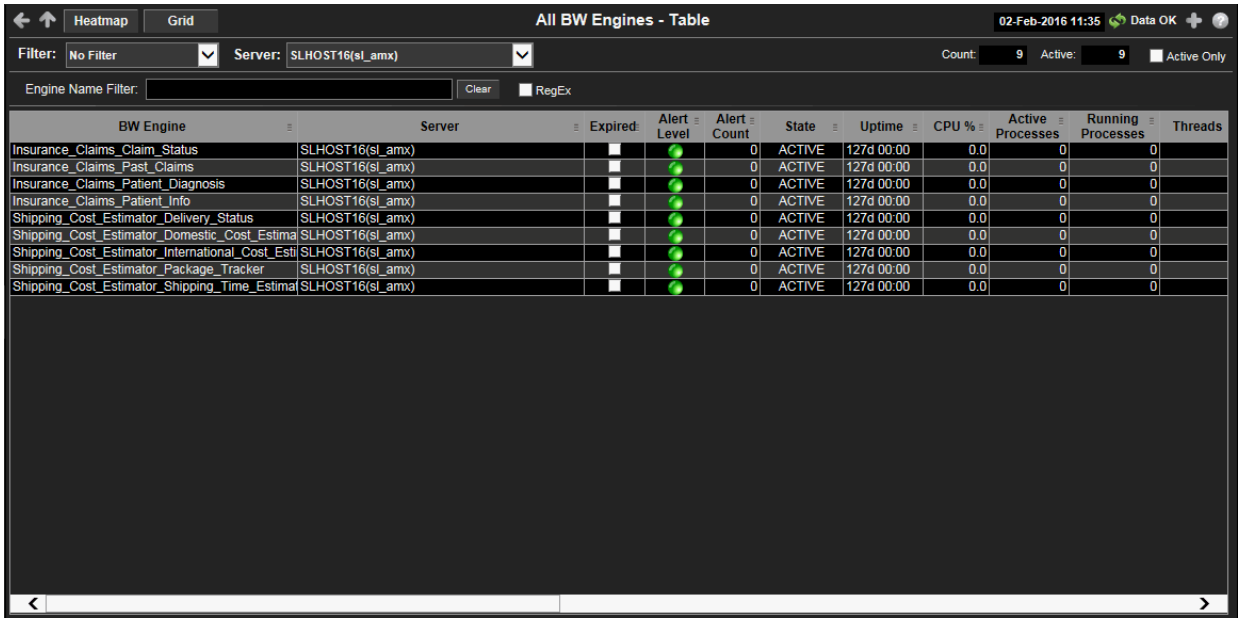
The display might include these filtering options:

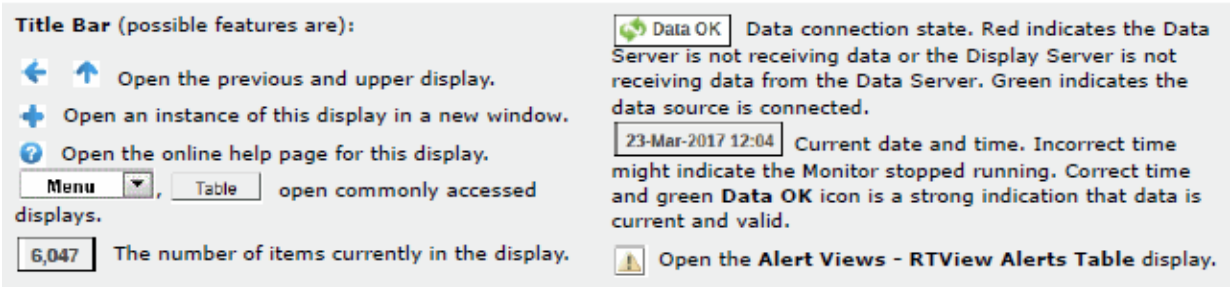
Filter:	Choose a filter to show data for in the display. By default, the **Filter:** drop-down menu only contains the **No Filter** option. To create your own filtering options, see **Creating Customized Filters** in the User's Guide.
Server:	Choose a server to show data for in the display.
Count:	The total number of engines in the display.
Active	Number of engines currently active.
Active Only	If selected, only engines with a status of ACTIVE are displayed. Otherwise, if deselected, all engines for the given Filter/Server selection are displayed.
Engine Names	Select this check box to display the names of the engines above their respective rectangles in the heatmap.
Log	Select to enable a logarithmic scale. Use **Log Scale** to see usage correlations for data with a wide range of values. For example, if a minority of your data is on a scale of tens, and a majority of your data is on a scale of thousands, the minority of your data is typically not visible in non-log scale graphs. **Log Scale** makes data on both scales visible by applying logarithmic values rather than actual values to the data.
Auto	Select to enable auto-scaling. When auto-scaling is activated, the color gradient bar's maximum range displays the highest value. NOTE: Some metrics auto-scale automatically, even when **Auto** is not selected.
Metric	Choose a metric to view in the display.
Alert Severity	The maximum level of alerts in the heatmap rectangle. Values range from **0** - **2**, as indicated in the color gradient bar, where **2** is the highest Alert Severity: Red indicates that one or more metrics exceeded their ALARM LEVEL threshold. Yellow indicates that one or more metrics exceeded their WARNING LEVEL threshold. Green indicates that no metrics have exceeded their alert thresholds.
Alert Count	The total number of critical and warning alerts in the heatmap rectangle. The color gradient bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average alert count.
CPU Used%	The percent (%) CPU used in the heatmap rectangle. The color gradient bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Memory Used%	The percent (%) memory used in the heatmap rectangle. The color gradient bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar

	range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Active Processes	The number of currently active processes in the heatmap rectangle. The color gradient bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Running Processes	The number of currently running processes in the heatmap rectangle. The color gradient bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Created Processes	The number of created processes in the heatmap rectangle. The color gradient bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Created/sec	The number of created processes in the heatmap rectangle, per second. The color gradient bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Error Count	The total number of errors in the heatmap rectangle. The color gradient bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average alert count.

All Engines Table

Each row in the table is an engine. Metrics are made available by the Hawk microagent for the engine (for details, refer to documentation for TIBCO BusinessWorks Administration, Appendix A: TIBCO Hawk Microagent Methods). Click on an row to drill down to the Single Engine Summary display.





Filter By:

The display might include these filtering options:

Filter:	Choose a filter to show data for in the display. By default, the **Filter:** drop-down menu only contains the **No Filter** option. To create your own filtering options, see **Creating Customized Filters** in the User's Guide.
Server:	Choose a server to show data for in the display.
Count	Number of engines currently being displayed.
Active	Number of engines currently active.
Active Only	If selected, only engines with a status of ACTIVE are displayed. Otherwise, if deselected, all engines for the given Filter/Server selection are displayed.
Engine Name Filter	Enter all or part of engine name to view specific engines. NOTE: Wild card characters are supported.
Clear	Removes Engine Name Filter and all engines for the given Filter/Server selection are displayed.
RegEx	If selected, the specified Engine Name Filter will be interpreted as a full Regular Expression rather than a simple wildcard.

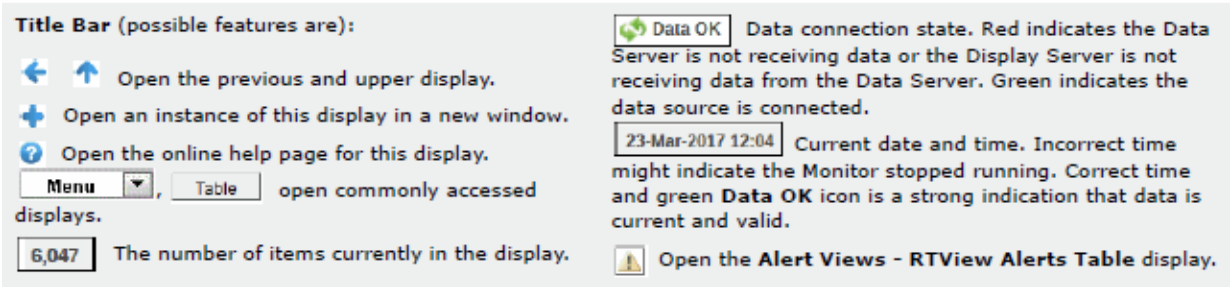
Table:

BW Engine	BW Engine name.
Server	Server agent name.
Expired	When checked, data has not been received from this host in the specified amount of time.
Alert Level	The most critical alert state for alerts in the row: Red indicates that one or more metrics exceeded their ALARM LEVEL threshold. Yellow indicates that one or more metrics exceeded their WARNING LEVEL threshold. Green indicates that no metrics have exceeded their alert thresholds.
Alert Count	Number of current alerts
State	Engine status: ACTIVE, STOPPED, LIMITED, etc. (See **All Servers Grid**).
Uptime	Uptime in milliseconds since the engine was started.
CPU %	Percent of server CPU used by engine.
Active Processes	Number of active processes calculated each update period using data returned by the Hawk method GetProcesses. **Note:** This column will display **NaN** or **Not Available** for any engine whose status is **STOPPED**.
Running Processes	Number of running processes.
Threads	Number of threads used by the engine.
Memory Used%	Percentage of allocated memory currently consumed by this engine from within the JVM. Equal to the value of: (100*UsedBytes) divided by MaxBytes. NOTE: Percent used is Long.

Max Heap Size	Maximum heap allocated to this engine for the JVM.
Total Bytes	Maximum heap memory this JVM has used.
Used Bytes	Total bytes of memory within the JVM currently used by the engine. Equal to value of: MaxBytes minus FreeBytes.
Free Bytes	Amount of available memory from within the JVM.
Mem Usage KBytes	Server memory in KB used by engine.
Errors	Total number of errors since the engine was started.
Delta Errors	Current number of new errors.
Errors/sec	Error rate per second.
Created Processes	The total number of processes that were created.
Completed Processes	The total number of processes that were completed.
Aborted Processes	The total number of processes that were aborted.
Process ID	Process ID of engine as recognized by the server.
Micro Agent Instance	Unique ID of the microagent reporting the metrics.
Deployment	Name of Deployment.
Domain	Name of Domain.
BW Version	The TIBCO BusinessWorks version currently in use on the server.
Source	Name of RTView Data Server sending this data (or localhost).
Time Stamp	Time of last update.
Process Name	Name of the BW Engine process on the server. Note: This information is not displayed in the table but is present in "raw" cache data.
Host	Host name of server. Note: This information is not displayed in the table but is present in "raw" cache data.
Adapter Name	Name of adapter. Note: This information is not displayed in the table but is present in "raw" cache data.
Instance ID	Instance ID name of the engine. Note: This information is not displayed in the table but is present in "raw" cache data.
Version	Engine project version number. Note: This information is not displayed in the table but is present in "raw" cache data.

All Engines Grid

Displays the main health metrics and a single trend graph per engine, summarizing the status of each BW5 Engine. Click on an engine icon to drill down to the Single Engine Summary display.



Filter By:

The display might include these filtering options:

Filter:	Choose a filter to show data for in the display. By default, the **Filter:** drop-down menu only contains the **No Filter** option. To create your own filtering options, see **Creating Customized Filters** in the User's Guide.
Server:	Choose a server to show data for in the display.
Count	Number of engines currently being displayed.
Active	Number of engines currently active.
Active Only	Toggle this setting to display active servers or all servers.
Time Range	Choose a time range. Also sets range for the **Single Engine Summary** display. Options are: **All Data**, **2 Mins**, **5 Mins**, **20 Mins**, **1 Hour**, **2 Hours**, **4 Hours**, **8 Hours**, **24 Hours**, **2 Days and 7 Days**.

Fields and Data

Engine Name	Name of the engine.
Status	Indicates the current state of the engine: • **ACTIVE** Indicates the BW microagent is providing live data and the engine is assumed active. • **SUSPENDED** This state is reported by the BW microagent. • **STANDBY** This state is reported by the BW microagent. • **STOPPING** This state is reported by the BW microagent.

- **STOPPED** This state is reported by the BW microagent.
- **LIMITED** Live data has been received from TIBCO, but deployment data from the custom RTView microagent has not been received.
- **EXPIRED** Indicates the server associated with the engine is unavailable or stopped sending data.

CPU Usage% Percent of server CPU in use.

Mem Usage% Available physical memory (MB) remaining.

Error Rate Number of errors accumulated per second.

Total Processes Number of process definitions for this engine.

Active Processes Number of process instances currently active.

Trend Graphs

Traces data for the server.

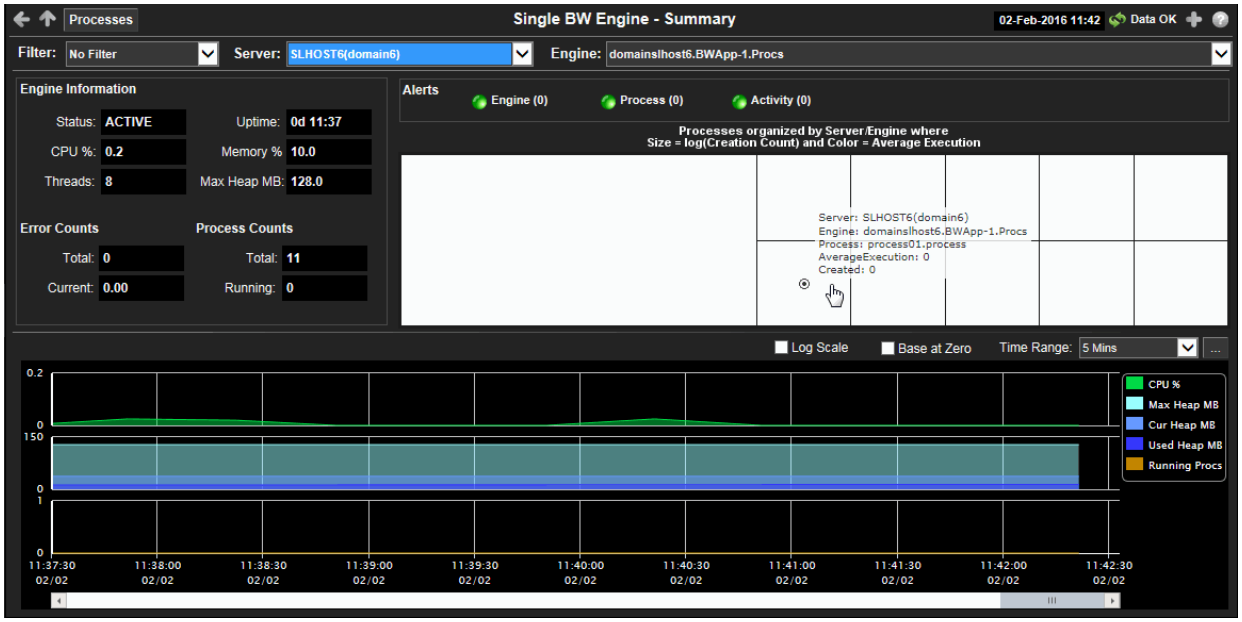
CPU Traces percent of server CPU in use.

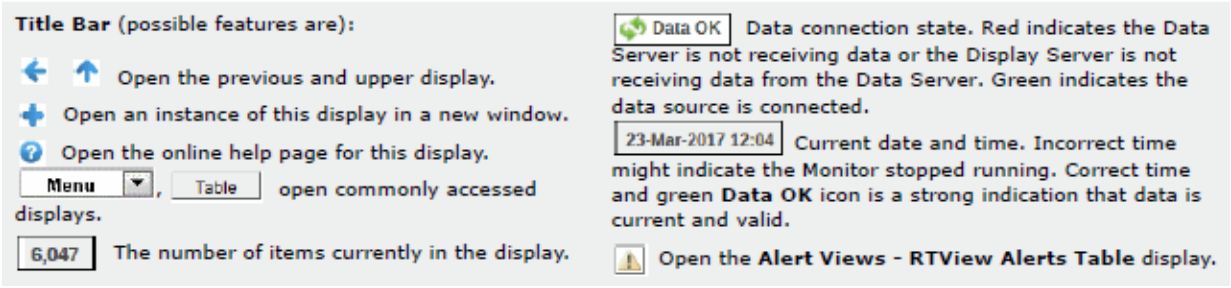
MEM Traces available physical memory remaining.

PROCS Traces total number of active processes.

Single Engine Summary

Several views show historical and current performance metrics for a single engine, including the number of processes running, threads, history of memory utilization, and trend graphs of memory utilization. In this display, when an engine is **Stopped** the engine name is appended with **(X)**, the background color is light red and Uptime is zero.





Note: Clicking the **Processes** button in the Title Bar takes you to the All Processes Heatmap . Clicking the **JVM** button, which is automatically enabled when a JMX connection is defined for the engine, takes you to the **JVM CPU/Mem Summary** display. See **Enable Monitoring Via JMX** for more information on enabling a JMX connection.

Filter By:

The display might include these filtering options:

Filter:	Choose a filter to show data for in the display. By default, the **Filter:** drop-down menu only contains the **No Filter** option. To create your own filtering options, see **Creating Customized Filters** in the User's Guide.
Server:	Choose a server to show data for in the display.
Engine:	Choose an engine to show data for in the display. An engine is not running when the engine name is appended with **(X)**.

Fields and Data

Engine Information

Status	**ACTIVE**	The BW microagent is providing live data and the engine is assumed active.
	SUSPENDED	This state is reported by the BW microagent.
	STANDBY	This state is reported by the BW microagent.
	STOPPING	This state is reported by the BW microagent.
	STOPPED	This state is reported by the BW microagent.
	LIMITED	Live data has been received from TIBCO, but deployment data from the custom RTView MicroAgent has not been received.
	EXPIRED	The associated server for the engine is currently in an EXPIRED state and is unavailable or stopped sending data.
Uptime		Days hours and minutes since the engine was started.
CPU%		Percent of server CPU used by engine.
Memory%		Available physical memory remaining (in MB).
Threads		Number of threads used by this engine
Max Heap MB		Maximum heap allocated to this engine for the JVM.

Error Counts

Total	Total errors accumulated by this engine.
Current	

Process Counts

Total	A BW Engine runs processes by creating instances of process definitions and making them active. A given process instance has a lifetime during which it may be suspended, swapped, queued, etc. until it is either completed or aborted. The Total value is calculated using the Hawk method named GetProcessDefinitions that returns statistics about the instances of each process definition including cumulative counts of completed, aborted, suspended, etc.

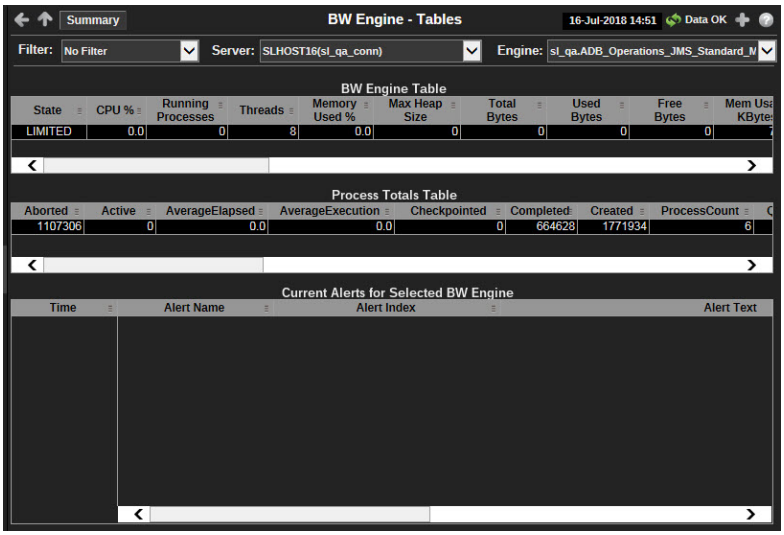
Running	Total number of running process instances. This number is calculated using the Hawk method named GetProcessCount. It is displayed in the Monitor Engines Table as RunningProcesses. The trend below displays the same value over time as Running Procs.

Alerts

Indicates the greatest severity level and the number of open **Engine**, **Process**, and **Activity** alerts for the selected engine. Values range from **0** to **2**, where **2** is the greatest Severity:

- One or more alerts exceeded their ALARM LEVEL threshold.
- One or more alerts exceeded their WARNING LEVEL threshold.
- No alert thresholds have been exceeded.

Click on the alert indicator to display the **BW Engine - Tables** display, which contains a table listing the current alerts for the selected engine.



Engine	Number of engine alerts and the most critical alert state for the engine: Red indicates that one or more metrics exceeded their ALARM LEVEL threshold. Yellow indicates that one or more metrics exceeded their WARNING LEVEL threshold. Green indicates that no metrics have exceeded their alert thresholds.
Process	Number of process alerts and the most critical alert state for the engine: Red indicates that one or more metrics exceeded their ALARM LEVEL threshold. Yellow indicates that one or more metrics exceeded their WARNING LEVEL threshold. Green indicates that no metrics have exceeded their alert thresholds.
Activity	Number of activity alerts and the most critical alert state for the engine: Red indicates that one or more metrics exceeded their ALARM LEVEL threshold. Yellow indicates that one or more metrics exceeded their WARNING LEVEL threshold. Green indicates that no metrics have exceeded their alert

thresholds.

Heatmap

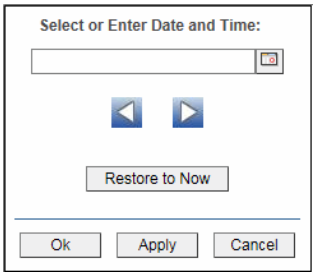
Shows processes organized by Server/Engine where Size = Creation Count and Color = Average Execution. Click on a node to drill down to a specific engine.

Trend Graphs

Log Scale — Select to enable a logarithmic scale. Use **Log Scale** to see usage correlations for data with a wide range of values. For example, if a minority of your data is on a scale of tens, and a majority of your data is on a scale of thousands, the minority of your data is typically not visible in non-log scale graphs. **Log Scale** makes data on both scales visible by applying logarithmic values rather than actual values to the data.

Base at Zero — Select to use zero (**0**) as the Y axis minimum for all graph traces.

Time Range — Select a time range from the drop down menu varying from **2 Minutes** to **Last 7 Days**, or display **All Data**. To specify a time range, click Calendar .



By default, the time range end point is the current time. To change the time range end point, click Calendar and select a date and time from the calendar or enter the date and time in the text field using the following format: **MMM dd, YYYY HH:MM**. For example, **Aug 21, 2011 12:24 PM**.

Use the navigation arrows to move forward or backward one time period. NOTE: The time period is determined by your selection from the **Time Range** drop-down menu.

Click **Restore to Now** to reset the time range end point to the current time.

BW5 Processes

These displays present performance metrics for BW5 processes. Displays in this View are:

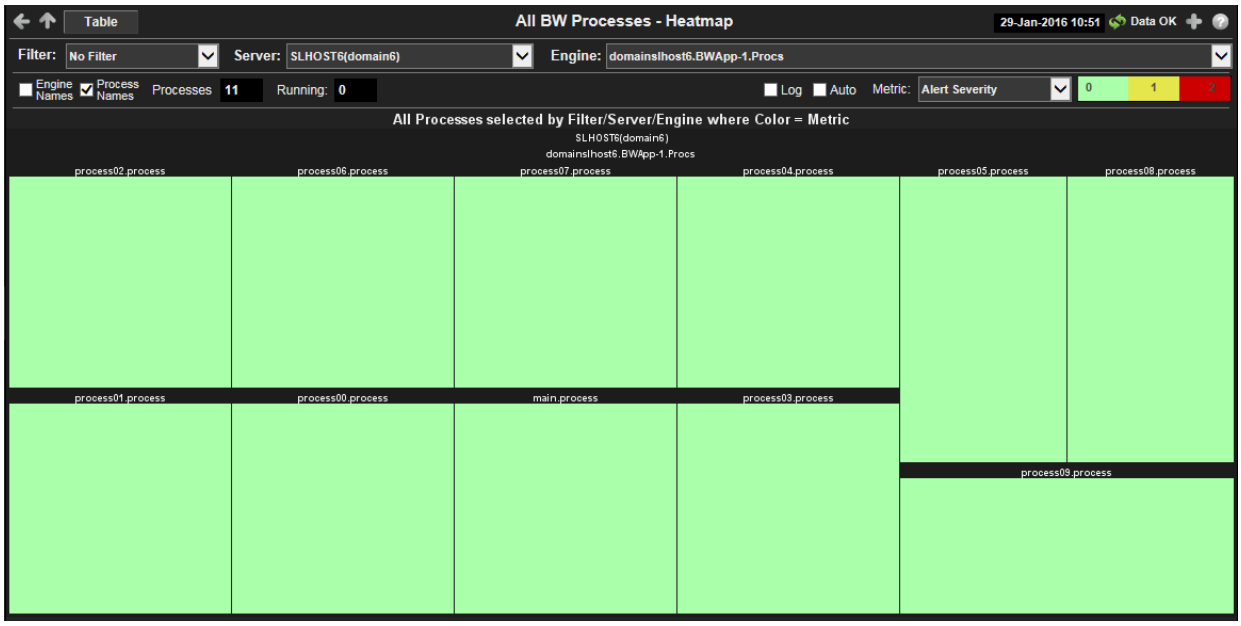
- "All Processes Heatmap " on page 259: Displays process execution metrics for all BW Engines.
- "All Processes Table " on page 262: Each row in the table displays all available metrics from the Hawk microagent for a process.
- "Single Process Summary" on page 265: Several views show historical and current metrics for a single process, including average execution times and execution counts.

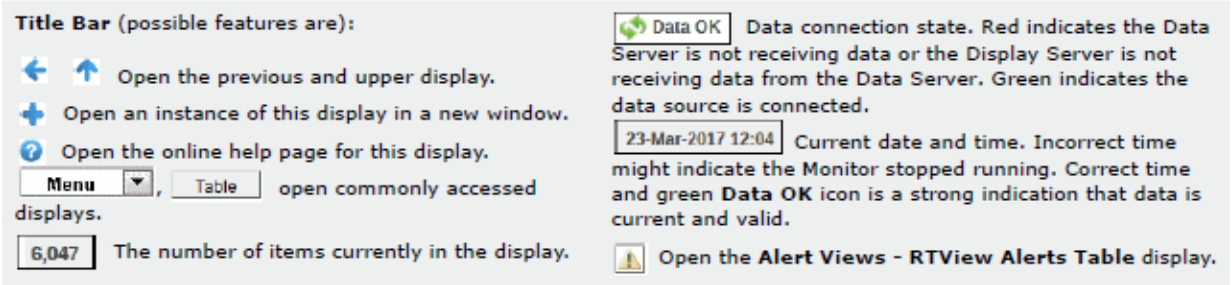
All Processes Heatmap

Summary view of processes can show the execution times for all processes on all engines or you can filter to look at specific servers or engines. Each rectangle (node) in the heatmap represents a process. Move your mouse over a node to display current metrics. Click on a node to drill-down to the Single Process Summary display to view specific metrics about process behavior over a specified period of time and determine which activity in the process may be causing the bottleneck.

An engine is not running when the engine name is appended with **(X)**.

Mouse-over any node to display the current values for the metric selected from the **Metric** drop-down menu.





Filter By:

The display might include these filtering options:

Filter:	Choose a filter to show data for in the display. By default, the **Filter:** drop-down menu only contains the **No Filter** option. To create your own filtering options, see **Creating Customized Filters** in the User's Guide.
Server:	Choose a server to show data for in the display.
Engine:	Choose an engine to show data for in the display. An engine is not running when the engine name is appended with **(X)**.
Engine Names	Select this check box to display the names of the engines above their respective rectangles in the heatmap.
Process Names	Select this check box to display the names of the processes above their respective rectangles in the heatmap.
Processes	The total number of processes in the display.
Running	Number of processes currently running.
Log	Select to enable a logarithmic scale. Use **Log Scale** to see usage correlations for data with a wide range of values. For example, if a minority of your data is on a scale of tens, and a majority of your data is on a scale of thousands, the minority of

your data is typically not visible in non-log scale graphs. **Log Scale** makes data on both scales visible by applying logarithmic values rather than actual values to the data.

Auto Select to enable auto-scaling. When auto-scaling is activated, the color gradient bar's maximum range displays the highest value. NOTE: Some metrics auto-scale automatically, even when **Auto** is not selected.

Metric Choose a metric to view in the display.

Alert Severity The maximum level of alerts in the heatmap rectangle. Values range from **0** - **2**, as indicated in the color gradient  bar, where **2** is the highest Alert Severity:

- Red indicates that one or more metrics exceeded their ALARM LEVEL threshold.
- Yellow indicates that one or more metrics exceeded their WARNING LEVEL threshold.
- Green indicates that no metrics have exceeded their alert thresholds.

Alert Count The total number of critical and warning alerts in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average alert count.

Completed Count The total number of completed processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.

Active Count The total number of active processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.

Aborted Count The total number of aborted processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.

Suspended Count The total number of suspended processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.

Exec Time / sec The number of processes executed per second in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.

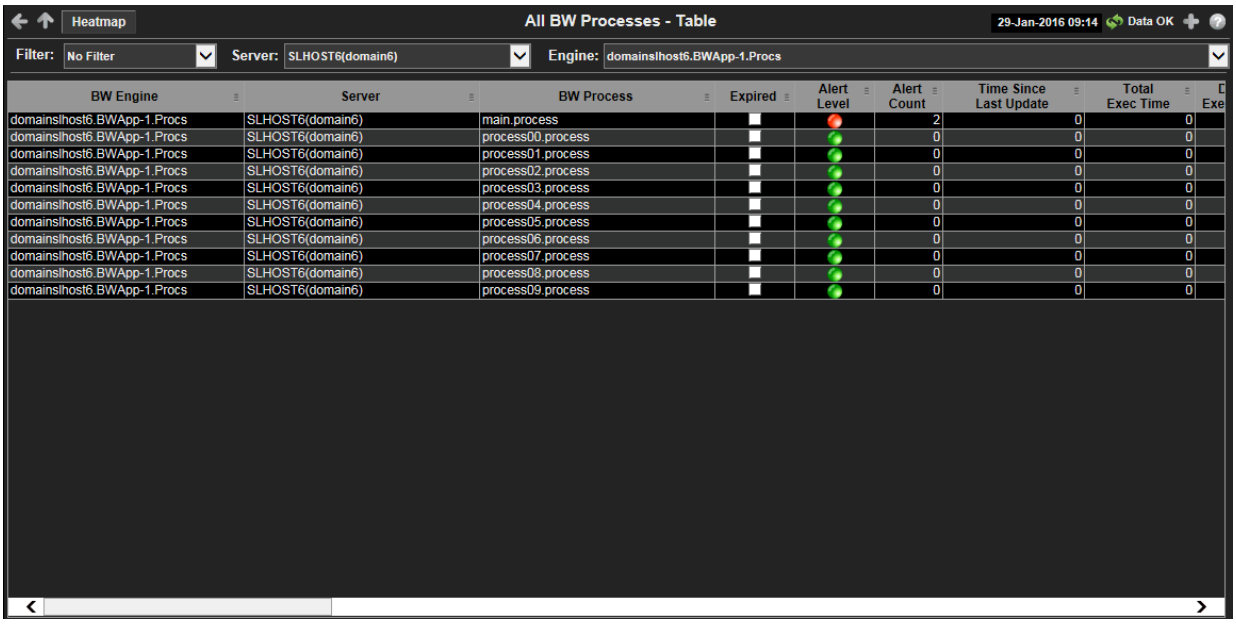
Created / sec The number of processes created per second in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.

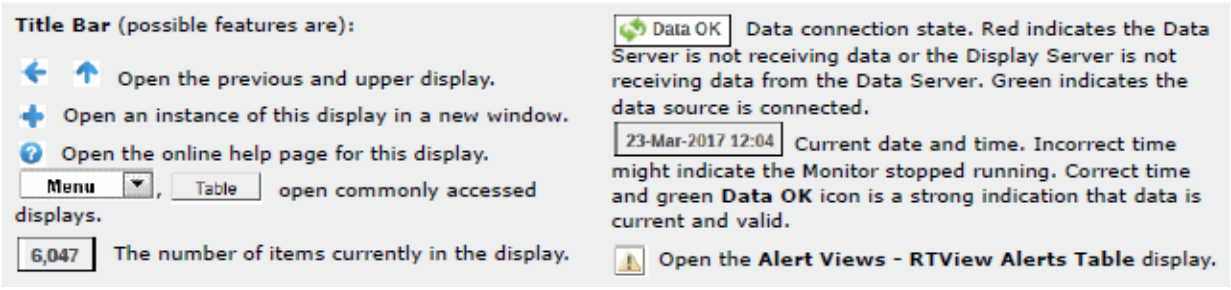
Aborted / sec	The number of aborted processes per second in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Suspended / sec	The number of suspended processes per second in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Most Recent Exec Time	The execution time for the most recently executed process in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Average Exec Time	The average execution time for all processes in the heatmap rectangle, calculated by dividing the delta execution time for the interval by the delta completed, or the number of process instances that completed in the interval. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Most Recent Elapsed Time	The elapsed time for the most recently executed process in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Average Elapsed Time	The average elapsed time for all processes in the heatmap rectangle, calculated by dividing the delta elapsed time for the interval by the delta completed, or the number of process instances that completed in the interval. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.

All Processes Table

Select a server and engine from the drop-down menus. Each row in the table is a different engine. The table displays all metrics available from the Hawk microagent for an engine. (Refer to documentation for TIBCO BusinessWorks Administration, see Appendix A: TIBCO Hawk Microagent Methods).

Click on a row in the table to drill down to the Single Engine Summary display.





Filter By:

The display might include these filtering options:

Filter:	Choose a filter to show data for in the display. By default, the **Filter:** drop-down menu only contains the **No Filter** option. To create your own filtering options, see **Create Customized Filters** for more information.
Server:	Choose a server to show data for in the display.
Engine:	Choose an engine to show data for in the display. An engine is not running when the engine name is appended with **(X)**.

Table:

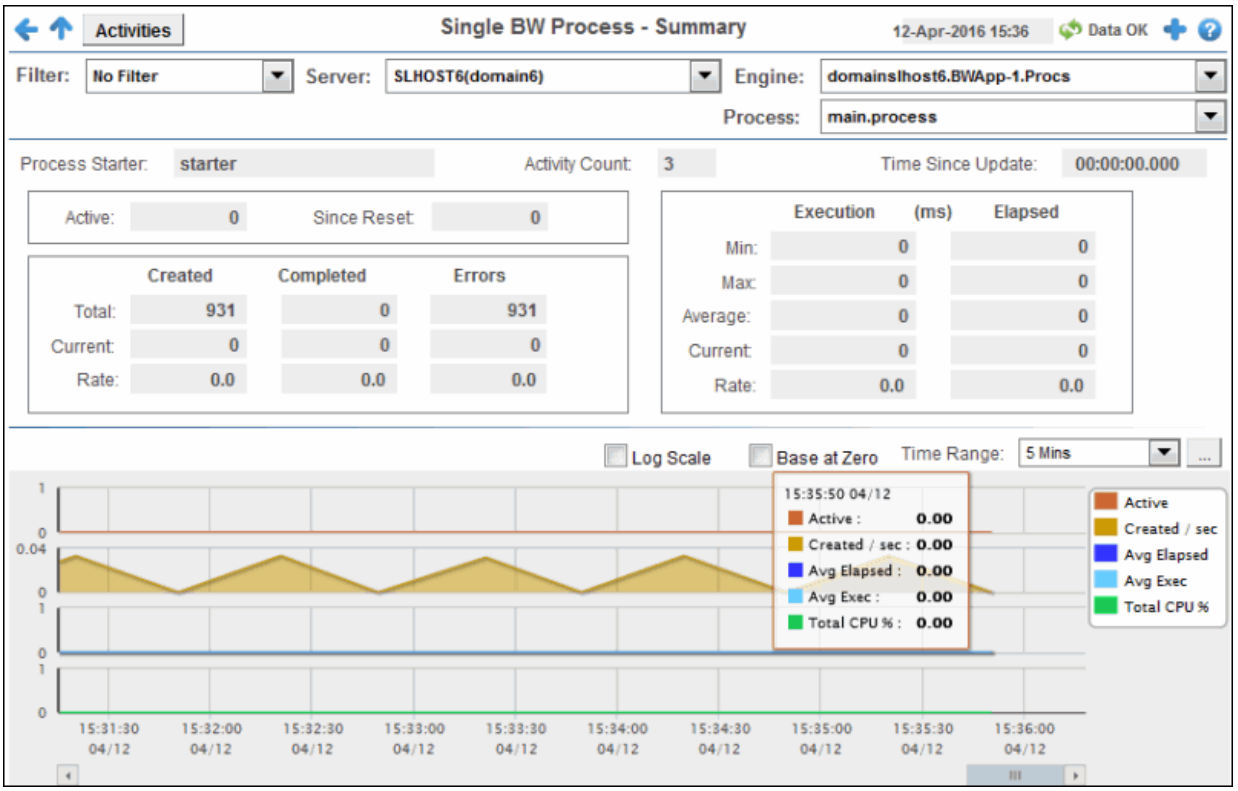
BW Engine	BW Engine name.
Server	Server agent name.
BW Process	The name of the process.
Expired	When checked, data has not been received from this host in the specified amount of time.
Alert Level	The most critical alert state for alerts in the row: Red indicates that one or more metrics exceeded their ALARM LEVEL threshold. Yellow indicates that one or more metrics exceeded their WARNING LEVEL threshold. Green indicates that no metrics have exceeded their alert thresholds.

Alert Count	Number of current alerts
Active	Number of active processes.
Total CPU	Total CPU usage in percent.
Created/sec	Change in Created per second.
Completed/sec	Change in Completed per second.
Delta Created	Change in Created this update.
Delta Completed	Change in Completed this update.
Created	Number of process instances created for this process definition.
Completed	Number of process instances successfully completed.
Total Exec Time	Total execution time (in milliseconds) for all successfully completed process instances.
Delta Exec Time	Execution time accumulated during the current polling period.
Exec Time/sec	Delta execution time per second.
Min Exec Time	Execution time (in milliseconds) of the process instance that has completed in the shortest amount of execution time.
Max Exec Time	Execution time (in milliseconds) of the process instance that has completed in the longest amount of execution time.
Average Exec Time	Average execution time (in milliseconds) for all successfully completed process instances.
Recent Exec Time	Execution time (in milliseconds) of the most recently completed process instance.
Total Elapsed Time	Total elapsed time (in milliseconds) for all successfully completed process instances.
Delta Elapsed Time	Elapsed time accumulated during the current polling period.
Elapsed Time/sec	Delta elapsed time per second.
Min Elapsed Time	Elapsed clock time (in milliseconds) of the process instance that has completed in the shortest amount of elapsed time.
Max Elapsed Time	Elapsed clock time (in milliseconds) of the process instance that has completed in the longest amount of elapsed time.
Average Elapsed Time	Average elapsed clock time (in milliseconds) for all successfully completed process instances.
Recent Elapsed Time	Elapsed clock time (in milliseconds) of the most recently completed process instance.
Aborted	Number of times process instances have been aborted.
Delta Aborted	Change in Aborted this update.
Aborted/sec	Change in Aborted per second.
Queued	Number of times process instances have been queued for execution.
Delta Queued	Change in Queued this update.
Queued/sec	Change in Queued per second.
Suspended	Number of times process instances have been suspended.
Delta Suspended	Change in Suspended this update.
Suspended/sec	Change in Suspended per second.
Checkpointed	Number of times process instances have executed a checkpoint.
Delta Checkpointed	Change in Checkpointed this update.
Checkpointed/ sec	Change in Checkpointed per second.
Swapped	Number of times process instances have been swapped to disk.
Delta Swapped	Change in Swapped this update.
Swapped/sec	Change in Swapped per second.
Time Since Last	Time since the last update.

Update	
Domain	Name of TIBCO Domain.
Starter	Name of the process starter for the process.
MicroAgent Instance	Unique ID of the microagent reporting the metrics.
CountSince Reset	Number of process instances that have completed since the last reset of the statistics.
Source	Name of RTView Data Server sending this data (or localhost).
Time Stamp	Time of last update.

Single Process Summary

Detailed performance metrics and alert status for a single BW process. Select a server, engine and process from the drop-down menus. The background color of the display is red when the selected engine is stopped.



Title Bar (possible features are):

Open the previous and upper display.

Open an instance of this display in a new window.

Open the online help page for this display.

Menu, Table open commonly accessed displays.

6,047 The number of items currently in the display.

Data OK Data connection state. Red indicates the Data Server is not receiving data or the Display Server is not receiving data from the Data Server. Green indicates the data source is connected.

23-Mar-2017 12:04 Current date and time. Incorrect time might indicate the Monitor stopped running. Correct time and green **Data OK** icon is a strong indication that data is current and valid.

Open the **Alert Views - RTView Alerts Table** display.

Filter By:

The display might include these filtering options:

Filter:	Choose a filter to show data for in the display. By default, the **Filter:** drop-down menu only contains the **No Filter** option. To create your own filtering options, see **Creating Customized Filters** in the User's Guide.
Server:	Choose a server to see metrics for.
Engine:	Choose a server to see metrics for. An engine is not running when the engine name is appended with **(X)**.
Process:	Choose a process to see metrics for.
Process Starter	Name of the process starter for the process.
Activity Count	Number of activities defined for this process.
Time Since Update	Time since the last update to file of statistics.
Active	Number of active instances for this process definition. This number is calculated using the Hawk method named GetProcesses. This method returns information about process instances that are active at the time of update. The value here displays the current total count of all active instances discovered for this process definition. The trend below displays the same value over time.
Since Reset	Number of activity executions that have completed since the last reset of the statistics. This is the number retrieved from the Hawk method named GetProcessDefinition which returns ExecutionCountSinceReset.

Execution Counts

Most recent execution counts for this process.

Created	**Total**	Number of process instances created for this process definition.
	Current	Number of process instances created this update cycle.
	Rate	Number of process instances created per second.
Completed	**Total**	Number of process instances successfully completed.
	Current	Number of process instances successfully completed this update cycle.
	Rate	Number of process instances successfully completed per second.
Errors	**Total**	Number of errors accumulated by all process instances.
	Current	Number of errors accumulated this update cycle.
	Rate	Number of errors accumulated per second.

Execution (ms) Elapsed

Execution and elapsed times in milliseconds for this process.

Min	Shortest time of any process instance.
Max	Longest time of any process instance.
Average	Average time across all successfully completed process instances.
Current	Time accumulated this update cycle.
Rate	Time accumulated per second.

Trend Graphs

- **Active**: Traces the number of currently active processes.
- **Created / sec**: Traces the number of created processes per second.
- **Avg Elapsed:** Traces the average number of elapsed processes. This value is calculated by dividing the delta elapsed time for the interval by the delta completed, or the number of process instances that completed in the interval.
- **Avg Exec**: Traces the average number of executed processes. This value is calculated by dividing the delta executed time for the interval by the delta completed, or the number of process instances that completed in the interval.
- **Total CPU %**: Traces CPU utilization by processes, in percent.

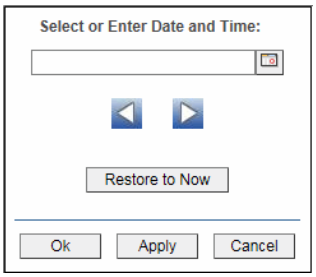
Log Scale	Select to enable a logarithmic scale. Use **Log Scale** to see usage correlations for data with a wide range of values. For example, if a minority of your data is on a scale of tens, and a majority of your data is on a scale of thousands, the minority of your data is typically not visible in non-log scale graphs. **Log Scale** makes data on both scales visible by applying logarithmic values rather than actual values to the data.

Base at Zero

Select to use zero (**0**) as the Y axis minimum for all graph traces.

Time Range

Select a time range from the drop down menu varying from **2 Minutes** to **Last 7 Days**, or display **All Data**. To specify a time range, click Calendar .



By default, the time range end point is the current time. To change the time range end point, click Calendar and select a date and time from the calendar or enter the date and time in the text field using the following format: **MMM dd, YYYY HH:MM**. For example, **Aug 21, 2011 12:24 PM**.

Use the navigation arrows to move forward or backward one time period. NOTE: The time period is determined by your selection from the **Time Range** drop-down menu.

Click **Restore to Now** to reset the time range end point to the current time.

BW5 Activities

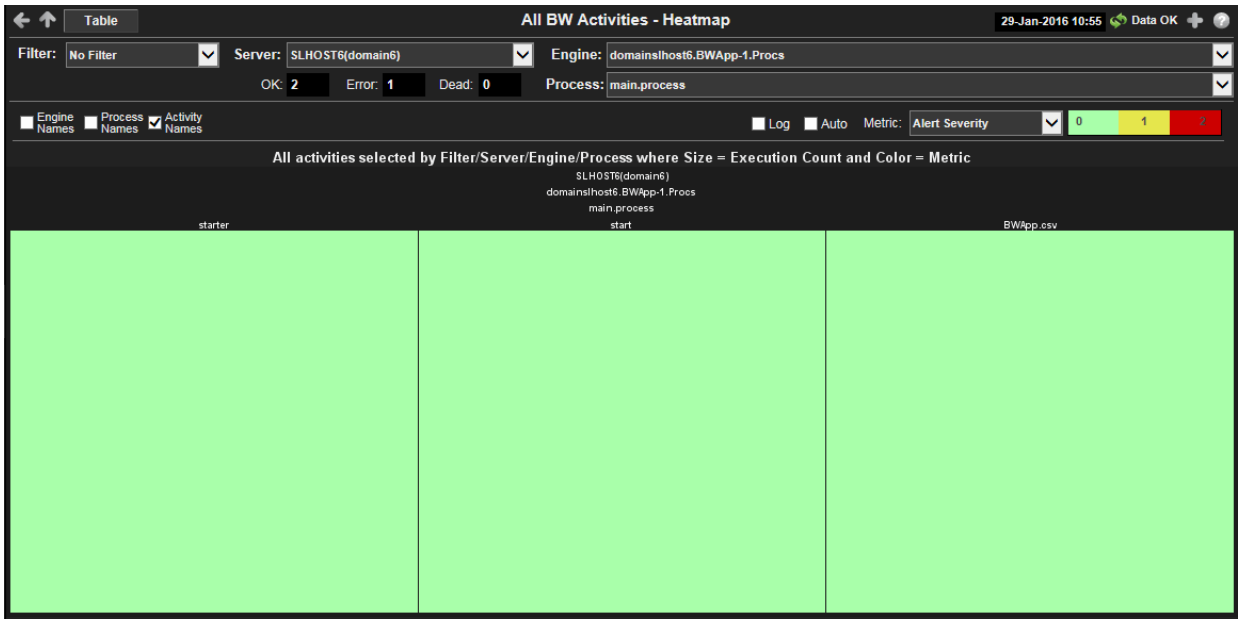
These displays present performance metrics for BW5 activities. Displays in this View are:

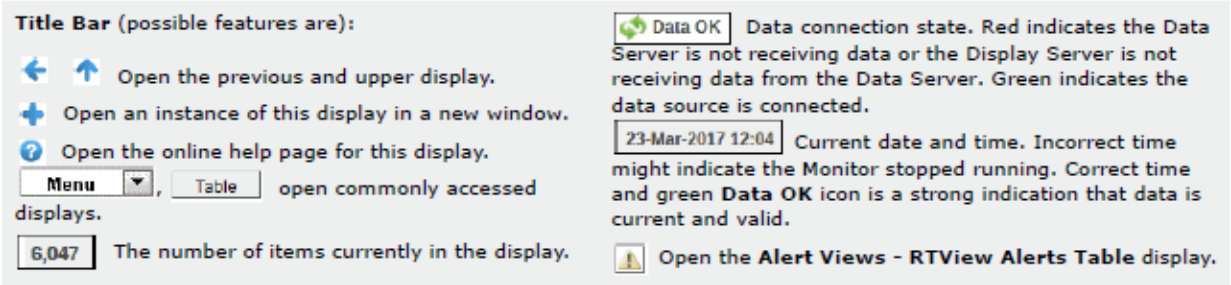
- "All Activities Heatmap " on page 267: Displays execution performance metrics for all BW activities.
- "All Activities Table " on page 270: Each row in the table displays all available metrics from the Hawk microagent for an activity.
- "Single Activity Summary" on page 272: Historical and current performance metrics for a single activity, including average execution times and execution counts.

All Activities Heatmap

Summary view of activities shows the execution times for all activities on all engines, or you can filter to look at specific servers, engines or processes. An engine is not running when the engine name is appended with **(X)**.

Move your mouse over a node to display current metrics. Click on a node to drill down to the Single Activity Summary display to view specific metrics about activity behavior over a specified period of time.





Filter By:

The display might include these filtering options:

Filter:	Choose a filter to show data for in the display. By default, the **Filter:** drop-down menu only contains the **No Filter** option. To create your own filtering options, see **Creating Customized Filters** in the User's Guide.
Server:	Choose a server to show data for in the display.
Engine:	Choose an engine to show data for in the display. An engine is not running when the engine name is appended with **(X)**.
Process	Select from the menu to view activities running on a specific process or all processes.
OK	Number of activities that reported their Last Return Code as **OK**.
Error	Number of activities that reported their Last Return Code as **Error**.
Dead	Number of activities that reported their Last Return Code as **Dead**.
Engine Names	Select this check box to display the names of the engines above their respective rectangles in the heatmap.
Process Names	Select this check box to display the names of the processes above their respective rectangles in the heatmap.
Activity Names	Select this check box to display the names of the activities above their respective rectangles in the heatmap.
Log	Select to enable a logarithmic scale. Use **Log Scale** to see usage correlations for data with a wide range of values. For example, if a minority of your data is on a scale of tens, and a majority of your data is on a scale of thousands, the

minority of your data is typically not visible in non-log scale graphs. **Log Scale** makes data on both scales visible by applying logarithmic values rather than actual values to the data.

Auto Select to enable auto-scaling. When auto-scaling is activated, the color gradient bar's maximum range displays the highest value. NOTE: Some metrics auto-scale automatically, even when **Auto** is not selected.

Metric Choose a metric to view in the display.

- **Alert Severity** The maximum level of alerts in the heatmap rectangle. Values range from **0** - **2**, as indicated in the color gradient  bar, where **2** is the highest Alert Severity:
  - Red indicates that one or more metrics exceeded their ALARM LEVEL threshold.
  - Yellow indicates that one or more metrics exceeded their WARNING LEVEL threshold.
  - Green indicates that no metrics have exceeded their alert thresholds.
- **Alert Count** The total number of critical and warning alerts in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average alert count.
- **Exec Count** The total number of executed processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
- **Error Count** The total number of errors in the heatmap rectangle. The color gradient  bar populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
- **Exec Time / sec** The number of processes executed per second in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
- **Errors / sec** The number of errors per second in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
- **Most Recent Exec Time** The execution time for the most recently executed process in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
- **Max Exec Time** The maximum execution time for executed processes in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.

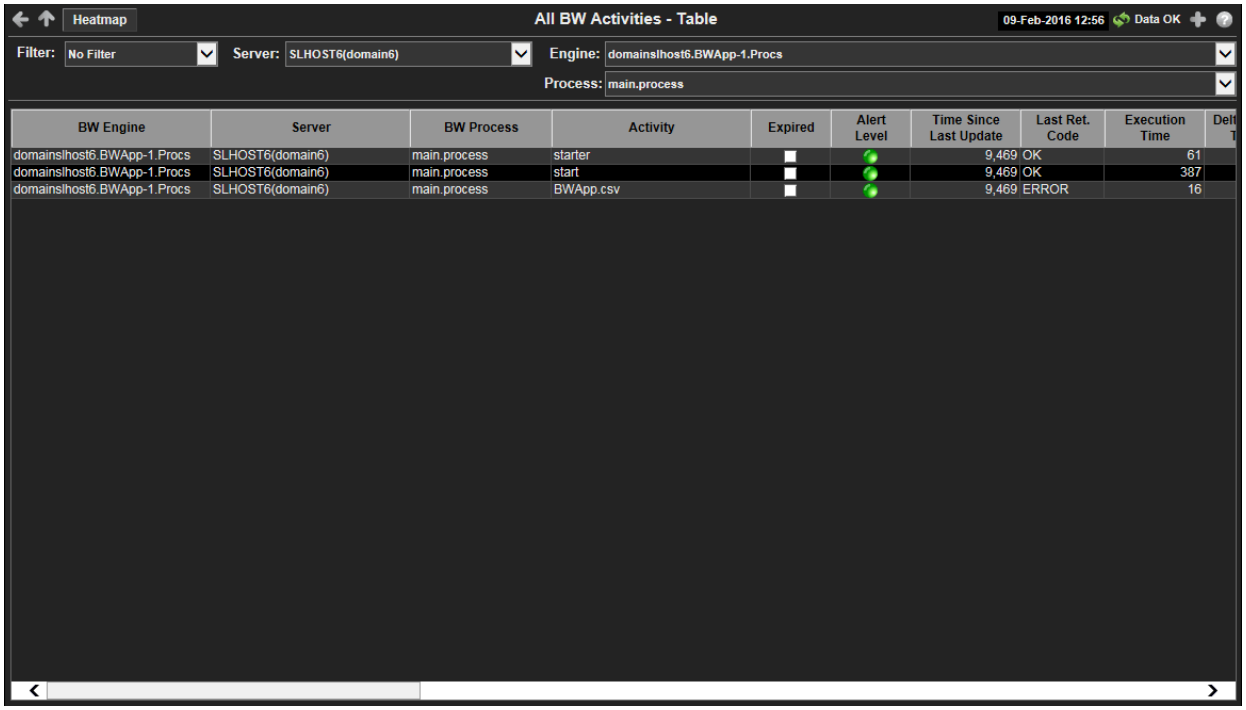
All Activities Table

Select a server, engine and process from the drop-down menus. Each row in the table displays all metrics available from the Hawk microagent for an activity. (Refer to documentation for TIBCO BusinessWorks Administration, see Appendix A: TIBCO Hawk Microagent Methods).

Click on a row in the table to drill down to the Single Activity Summary display to view specific metrics about activity behavior over a specified period of time.

When the background/foreground color of a row changes color, the associated engine for the activity is currently in an EXPIRED state. An engine is EXPIRED when the associated server is unavailable or stopped sending data.

An EXPIRED activity and the associated engine are deleted from displays when the associated server exceeds its specified threshold. Processes associated with the engine are also deleted from displays.



Title Bar (possible features are):

Open the previous and upper display.

Open an instance of this display in a new window.

Open the online help page for this display.

Menu, Table open commonly accessed displays.

6,047 The number of items currently in the display.

Data OK Data connection state. Red indicates the Data Server is not receiving data or the Display Server is not receiving data from the Data Server. Green indicates the data source is connected.

23-Mar-2017 12:04 Current date and time. Incorrect time might indicate the Monitor stopped running. Correct time and green **Data OK** icon is a strong indication that data is current and valid.

Open the **Alert Views - RTView Alerts Table** display.

Filter By:

The display might include these filtering options:

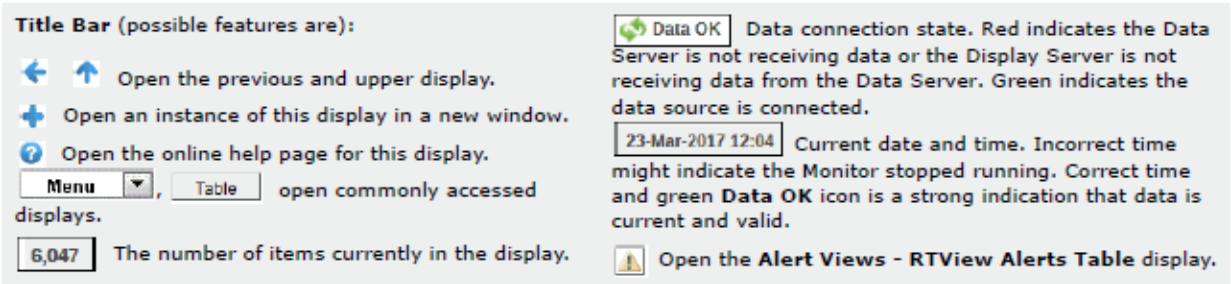
Filter: Choose a filter to show data for in the display. By default, the **Filter:** drop-down menu only contains the **No Filter** option. To

	create your own filtering options, see **Creating Customized Filters** in the User's Guide.
Server:	Choose a server to show data for in the display.
Engine:	Select from the menu to view activities running on a specific engine or all engines. An engine is not running when the engine name is appended with **(X)**.
Process:	Select from the menu to view activities running on a specific process or all processes.
Table:	
BW Engine	Name of BW Engine.
Server	Name of Server agent.
BW Process	Name of the BW engine Process on the Server.
Activity	Name of activity.
Expired	When checked, data has not been received from this host in the specified amount of time.
Alert Level	The most critical alert state for alerts in the row: Red indicates that one or more metrics exceeded their ALARM LEVEL threshold. Yellow indicates that one or more metrics exceeded their WARNING LEVEL threshold. Green indicates that no metrics have exceeded their alert thresholds.
Time Since Last Update	Time since the last update.
Last Ret(urn) Code	Status code (OK DEAD ERROR) returned by most recent execution of this activity.
Execution Time	Time (in milliseconds) used by all executions of this activity. NOTE: This does not include wait time for Sleep, Call Process, and Wait For... activities.
Delta Exec(ution) Time	Execution time accumulated this update cycle.
Exec(ution) Time /sec	Execution time accumulated per second.
Min Exec(ution) Time	Time (in milliseconds) of the activity that has the shortest execution time.
Max Exec(ution) Time	Time (in milliseconds) of the activity that has the longest execution time.
Elapsed Time	Elapsed clock time (in milliseconds) used by all executions of this activity. NOTE: This does not include wait time for Sleep, Call Process, and Wait For... activities.
Delta Elapsed Time	Change in ElapsedTime this update.
Elapsed Time/sec	Change in ElapsedTime per second.
Min Elapsed Time	Elapsed clock time (in milliseconds) of the activity that has the shortest execution time.
Max Elapsed Time	Elapsed clock time (in milliseconds) of the activity that has the longest execution time.
Executions	Number of times the activity has been executed.
Delta Exec(ution)	Change in ExecutionCount this update.
Executions/sec	Change in ExecutionCount per second.
Errors	Total number of executions of the activity that have returned an error.
Delta Errors	Change in ErrorCount this update.
Errors/sec	Change in ErrorCount per second.
Domain	Name of TIBCO Domain.
ActivityClass	Name of the class that implements the activity.

CalledProcessDefs	A comma-separated list of definitions called by this activity.
Tracing	• **true** Tracing is enabled for this activity. • **false** Tracing is disabled for this activity.
MicroAgentInstance	Unique ID of the microagent reporting the metrics.
ExecutionCountSinceReset	Number of times the activity has been executed since the last reset of the statistics.
Source	Name of RTView Data Server sending this data (or localhost).
Time Stamp	Time of this update.

Single Activity Summary

Detailed performance metrics and alert status for a single BW activity. In this display, when an engine associated with the activity is **Stopped** the engine name is appended with **(X)** and the background color is light red.



Filter By:

The display might include these filtering options:

Filter:	Choose a filter to show data for in the display. By default, the **Filter:** drop-down menu only contains the **No Filter** option. To create your own filtering options, see **Creating Customized Filters** in the User's Guide.
Server:	Select from the menu to view processes running on a specific server.

Engine:	Select from the menu to view processes running on a specific engine. An engine is not running when the engine name is appended with **(X)**.
Process:	Select from the menu to view summary details for a specific process.
Activity	Select from the menu to view summary details for a specific activity.
Class	Name of the activity class.
Last Return Code	Last return code reported from this activity.
Time Since Update	Time since the last update.

Execution Counts

Most recent execution counts for this activity.

Total	Number of times the activity has been executed.
Since Reset	Number of times the activity has been executed since the last Hawk reset of the statistics.
Current	Change in ExecutionCount this update.
Rate	Change in Execution Count per second.

Error Counts

Most recent error counts for this activity.

Total	Number of errors accumulated by all activities.
Average	Average number of errors accumulated by all activities.
Current	Number of errors accumulated this update cycle.
Rate	Number of errors accumulated per second.

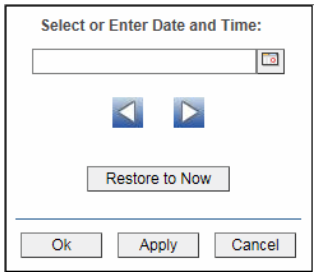
Execution (ms) Elapsed

Execution and elapsed times in milliseconds for this activity.

Min	Shortest time of any activity instance.
Max	Longest time of any activity instance.
Average	Average time across all successfully completed activity instance.
Current	Time accumulated this update cycle.
Rate	Time accumulated per second.

Trend Graphs

Log Scale	Select to enable a logarithmic scale. Use **Log Scale** to see usage correlations for data with a wide range of values. For example, if a minority of your data is on a scale of tens, and a majority of your data is on a scale of thousands, the minority of your data is typically not visible in non-log scale graphs. **Log Scale** makes data on both scales visible by applying logarithmic values rather than actual values to the data.
Base at Zero	Select to use zero (**0**) as the Y axis minimum for all graph traces.
Time Range	Select a time range from the drop down menu varying from **2 Minutes** to **Last 7 Days**, or display **All Data**. To specify a time range, click Calendar .



By default, the time range end point is the current time. To change the time range end point, click Calendar and select a date and time from the calendar or enter the date and time in the text field using the following format: **MMM dd, YYYY HH:MM**. For example, **Aug 21, 2011 12:24 PM**.

Use the navigation arrows to move forward or backward one time period.
NOTE: The time period is determined by your selection from the **Time Range** drop-down menu.

Click **Restore to Now** to reset the time range end point to the current time.

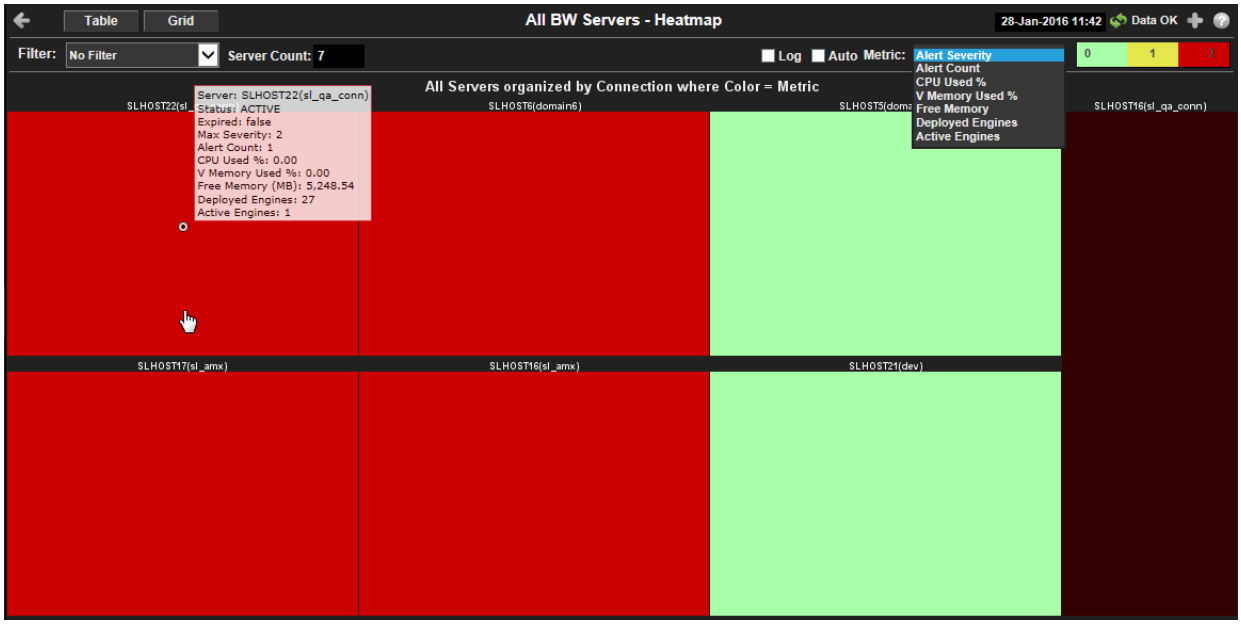
BW5 Servers

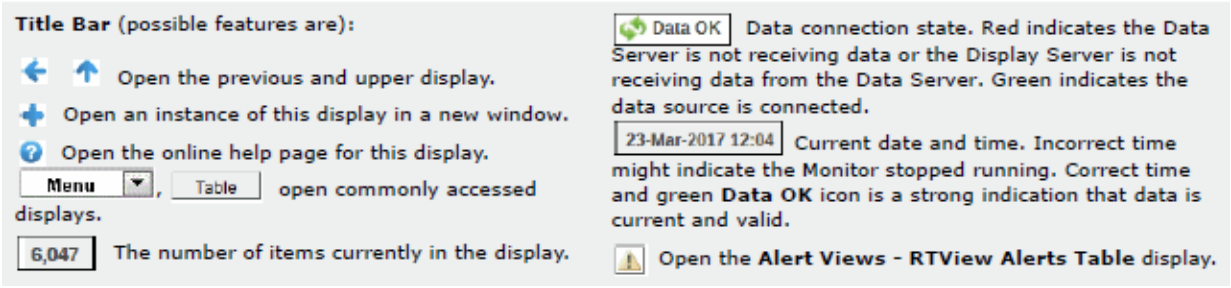
These displays present performance data for your BusinessWorks system. Displays in this View are:

All Servers Heatmap

Quick view of BW Servers status determined by selected Filter, organized by Connection (host) and where color equals the selected Metric. Each rectangle (node) in the heatmap represents a server.

Click on a node to drill down to the Single Server Summary display and view metrics for a particular server. Mouse-over any node to display the current values for the metric selected from the Metric drop-down menu.





Filter By:

The display might include these filtering options:

Filter: Choose a filter to limit data shown in the display. By default, the **Filter:** drop-down menu only contains the **No Filter** option. To create your own filtering options, see **Creating Customized Filters** in the User's Guide.

Server Count: The total number of servers in the display.

Log Select to enable a logarithmic scale. Use **Log Scale** to see usage correlations for data with a wide range of values. For example, if a minority of your data is on a scale of tens, and a majority of your data is on a scale of thousands, the minority of your data is typically not visible in non-log scale graphs. **Log Scale** makes data on both scales visible by applying logarithmic values rather than actual values to the data.

Auto Select to enable auto-scaling. When auto-scaling is activated, the color gradient bar's maximum range displays the highest value. NOTE: Some metrics auto-scale automatically, even when **Auto** is not selected.

Metric Choose a metric to view in the display.

Alert Severity The maximum level of alerts in the heatmap rectangle. Values range from **0** - **2**, as indicated in the color gradient bar, where **2** is the highest Alert Severity:

- Red indicates that one or more metrics exceeded their ALARM LEVEL threshold.
- Yellow indicates that one or more metrics exceeded their WARNING LEVEL threshold.
- Green indicates that no metrics have exceeded their alert thresholds.

Alert Count The total number of critical and warning alerts in the heatmap rectangle. The color gradient bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average alert count.

CPU Used% The percent (%) CPU used in the heatmap rectangle. The color gradient bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.

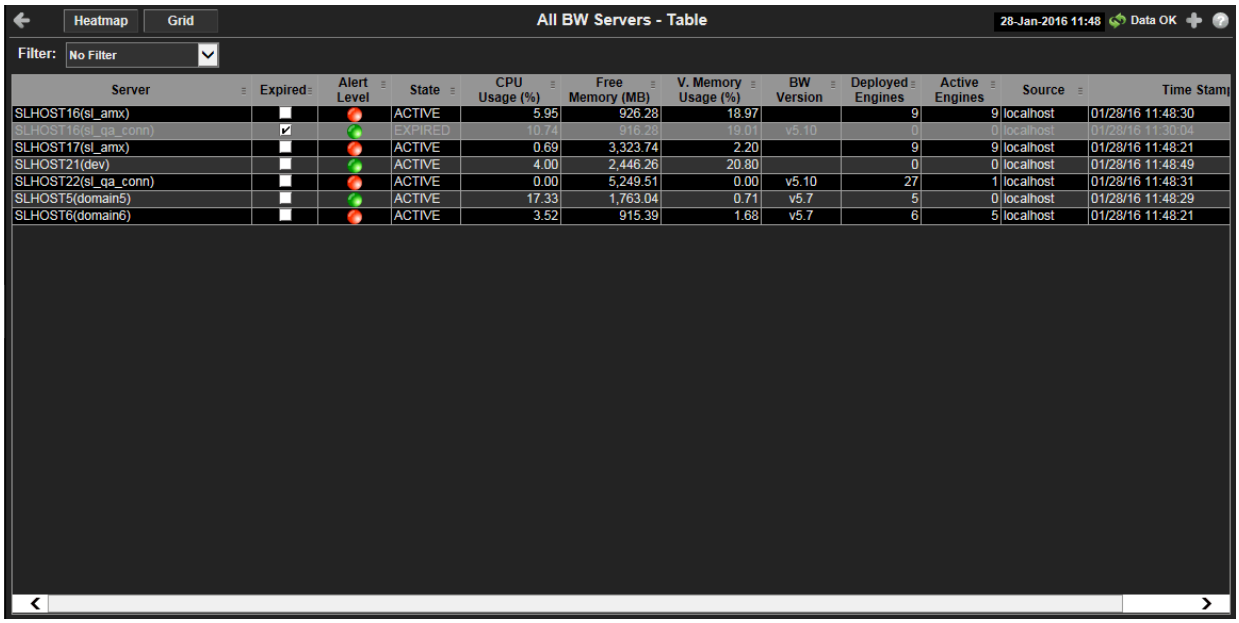
V(irtual) Memory Used% The percent (%) virtual memory used in the heatmap rectangle. The color gradient bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.

Free Memory The amount of free memory in the heatmap rectangle, in megabytes. The color gradient bar, populated by the current heatmap, shows the value/color mapping. The numerical

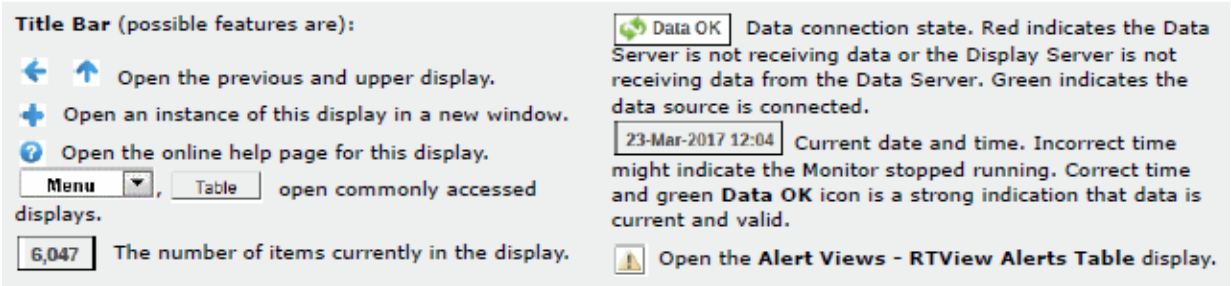
	values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Deployed Engines	The number of deployed engines in the heatmap rectangle. The color gradient bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
Active Engines	The number of active engines in the heatmap rectangle. The color gradient bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.

All Servers Table

This table provides a list view of utilization metrics for all BW servers (represented in the All Servers Heatmap). Each row in the table contains data for a particular server. Click a column header to sort column data in numerical or alphabetical order. Click on a table row to drill down to the Single Server Summary display and view metrics for that particular server.



Server	Expired	Alert Level	State	CPU Usage (%)	Free Memory (MB)	V. Memory Usage (%)	BW Version	Deployed Engines	Active Engines	Source	Time Stamp
SLHOST16(sl_amx)	☐		ACTIVE	5.95	926.28	18.97		9	9	localhost	01/28/16 11:48:30
SLHOST16(sl_qa_conn)	☑		EXPIRED	10.74	916.28	19.01	v5.10	0	0	localhost	01/28/16 11:30:04
SLHOST17(sl_amx)	☐		ACTIVE	0.69	3,323.74	2.20		9	9	localhost	01/28/16 11:48:21
SLHOST21(dev)	☐		ACTIVE	4.00	2,446.26	20.80		0	0	localhost	01/28/16 11:48:49
SLHOST22(sl_qa_conn)	☐		ACTIVE	0.00	5,249.51	0.00	v5.10	27	1	localhost	01/28/16 11:48:31
SLHOST5(domain5)	☐		ACTIVE	17.33	1,763.04	0.71	v5.7	5	0	localhost	01/28/16 11:48:29
SLHOST6(domain6)	☐		ACTIVE	3.52	915.39	1.68	v5.7	6	5	localhost	01/28/16 11:48:21



Filter By:

The display might include these filtering options:

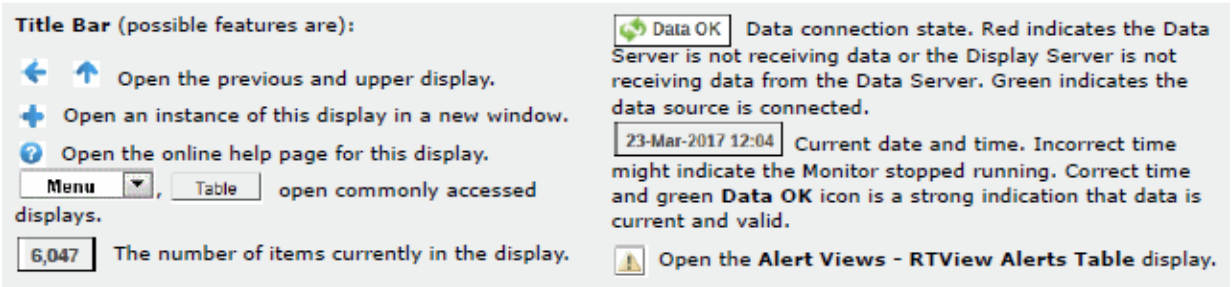
Filter:	Choose a filter to show data for in the display. By default, the **Filter:** drop-down menu only contains the **No Filter** option. To create your own filtering options, see **Creating Customized Filters** in the User's Guide.

Table:

Server	Name of Server Agent.
Expired	When checked, data has not been received from this host in the specified amount of time.
Alert Level	The most critical alert state for alerts in the row: Red indicates that one or more metrics exceeded their ALARM LEVEL threshold. Yellow indicates that one or more metrics exceeded their WARNING LEVEL threshold. Green indicates that no metrics have exceeded their alert thresholds.
State	The current status of the application. Valid values are **Running** and **Stopped**.
CPU Usage (%)	Percent of server CPU in use.
Free Memory (MB)	Available physical memory (MB) remaining.
V. Memory Usage (%)	Percent of virtual memory used.
BW Version	The TIBCO BusinessWorks version currently in use on the server.
Deployed Engines	Total number of engines deployed on the server.
Active Engines	Number of engines currently active.
Source	Name of RTView Data Server sending this data (or localhost).
Time Stamp	Time this data was retrieved.

All Servers Grid

This grid provides a list view of utilization metrics for all BW servers (represented in the All Servers Heatmap). Track and view in parallel the general performance of all BW servers. Click on a node to drill down to the Single Server Summary display and view detailed metrics for that particular server.



Filter By:

The display might include these filtering options:

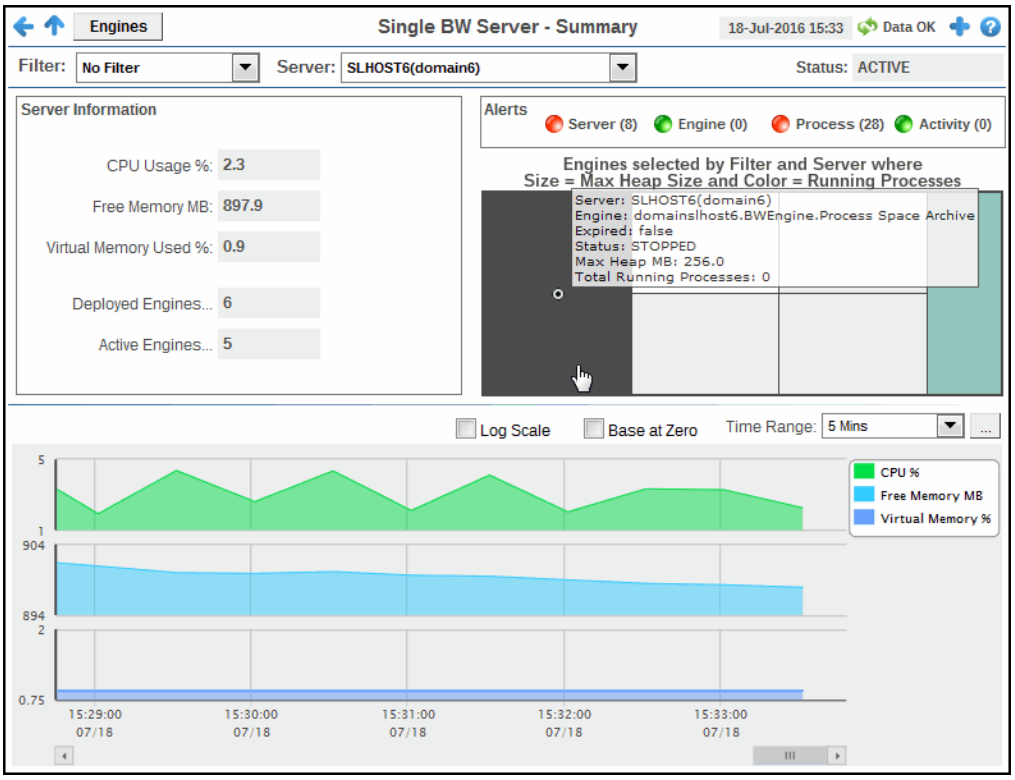
Filter:	Choose a filter to show data for in the display. By default, the **Filter:** drop-down menu only contains the **No Filter** option. To create your own filtering options, see **Creating Customized Filters** in the User's Guide.
Time Range	Choose a time range to show data for in the display. Options are: **All Data**, **2 Mins**, **5 Mins**, **20 Mins**, **1 Hour, 2 Hours**, **4 Hours**, **8 Hours**, **24 Hours**, **2 Days and 7 Days**.

Fields and Data

Server Name	Name of the server.	
CPU Usage%	Percent of server CPU in use.	
Free Memory	Available physical memory (MB) remaining.	
Virtual Mem Used%	Percent of virtual memory used.	
State	Server status: ACTIVE or EXPIRED.	
Deployed Engines	Total number of engines deployed on the server.	
Active Engines	Number of engines currently active.	
Trend Graphs	Shows data for the server.	
	CPU	Traces percent of server CPU in use.
	MEM	Traces available physical memory remaining.
	VMEM	Traces the percent of virtual memory used.

Single Server Summary

Detailed performance metrics and alert status for a single BW server. Click on any alert indicator to drill down to the **BW Server - Tables** display to view current alerts for the selected server.



Title Bar (possible features are):

Open the previous and upper display.

Open an instance of this display in a new window.

Open the online help page for this display.

Menu, Table open commonly accessed displays.

6,047 The number of items currently in the display.

Data OK Data connection state. Red indicates the Data Server is not receiving data or the Display Server is not receiving data from the Data Server. Green indicates the data source is connected.

23-Mar-2017 12:04 Current date and time. Incorrect time might indicate the Monitor stopped running. Correct time and green **Data OK** icon is a strong indication that data is current and valid.

Open the **Alert Views - RTView Alerts Table** display.

Filter By:

The display might include these filtering options:

Filter:	Choose a filter to show data for in the display. By default, the **Filter:** drop-down menu only contains the **No Filter** option. To create your own filtering options, see **Creating Customized Filters** in the User's Guide.
Server:	Choose a server to see metrics for.
Status	Server status: ACTIVE or EXPIRED.

Server Information

CPU Usage (%)	Percent of server CPU in use. Values are traced in trend graph (below).
Free Memory (MB)	Available physical memory remaining (in MB). Values are traced in trend graph (below).
V. Memory Usage (%)	Percent of virtual memory used. Values are traced in trend graph (below).
Deployed Engines	Number of engines currently active. Click to drill-down to details for deployed and active engines in the "All Engines Table " on page 252 display.

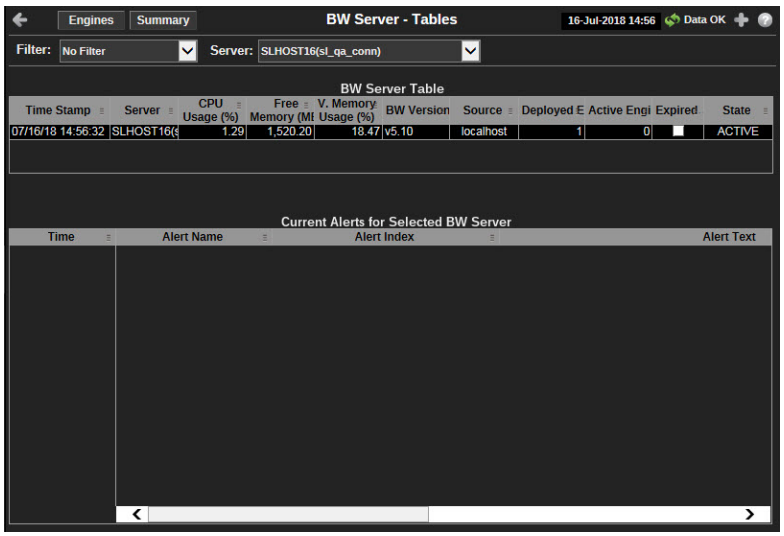
Active Engines	Shows data for the server. Click to drill-down to details for active engines in the "All Engines Table " on page 252 display.

Alerts

Indicates the greatest severity level and the number of open **Server**, **Engine**, **Process**, and **Activity** alerts for the selected server. Values range from **0** to **2**, where **2** is the greatest Severity:

- One or more alerts exceeded their ALARM LEVEL threshold.
- One or more alerts exceeded their WARNING LEVEL threshold.
- No alert thresholds have been exceeded.

Click on the alert indicator to display the **BW Server - Tables** display, which contains a table listing the current alerts for the selected engine.



Heatmap

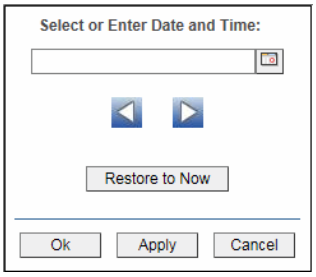
Engines selected by Filter and Server, where the heatmap rectangle size = Max Heap Size and the heatmap rectangle color = Running Processes. Dark green is the highest value for the metric shown). Click on a node to drill down to a specific engine:

- Red indicates that the engine is expired.
- Gray indicates that the engine is stopped.

Trend Graphs

Traces **CPU %**, **Free Memory MB** and **Virtual Memory %**.

Log Scale	Select to enable a logarithmic scale. Use **Log Scale** to see usage correlations for data with a wide range of values. For example, if a minority of your data is on a scale of tens, and a majority of your data is on a scale of thousands, the minority of your data is typically not visible in non-log scale graphs. **Log Scale** makes data on both scales visible by applying logarithmic values rather than actual values to the data.
Base at Zero	Select to use zero (**0**) as the Y axis minimum for all graph traces.
Time Range	Select a time range from the drop down menu varying from **2 Minutes** to **Last 7 Days**, or display **All Data**. To specify a time range, click Calendar .



By default, the time range end point is the current time. To change the time range end point, click Calendar ... and select a date and time from the calendar or enter the date and time in the text field using the following format: **MMM dd, YYYY HH:MM**. For example, **Aug 21, 2011 12:24 PM**.

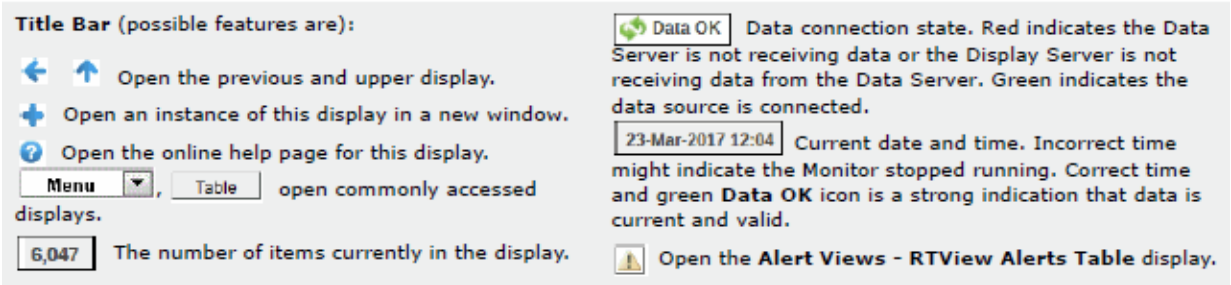
Use the navigation arrows to move forward or backward one time period. NOTE: The time period is determined by your selection from the **Time Range** drop-down menu.

Click **Restore to Now** to reset the time range end point to the current time.

Server Processes

Detailed information about operating system processes of a single BW Server. The heatmap shows server processes selected by Filter and Server, where the rectangle size equals memory usage and the rectangle color equals CPU percent usage.

NOTE: By default, this display is not enabled. For details, see **Enable BW Servers**.



Filter By:

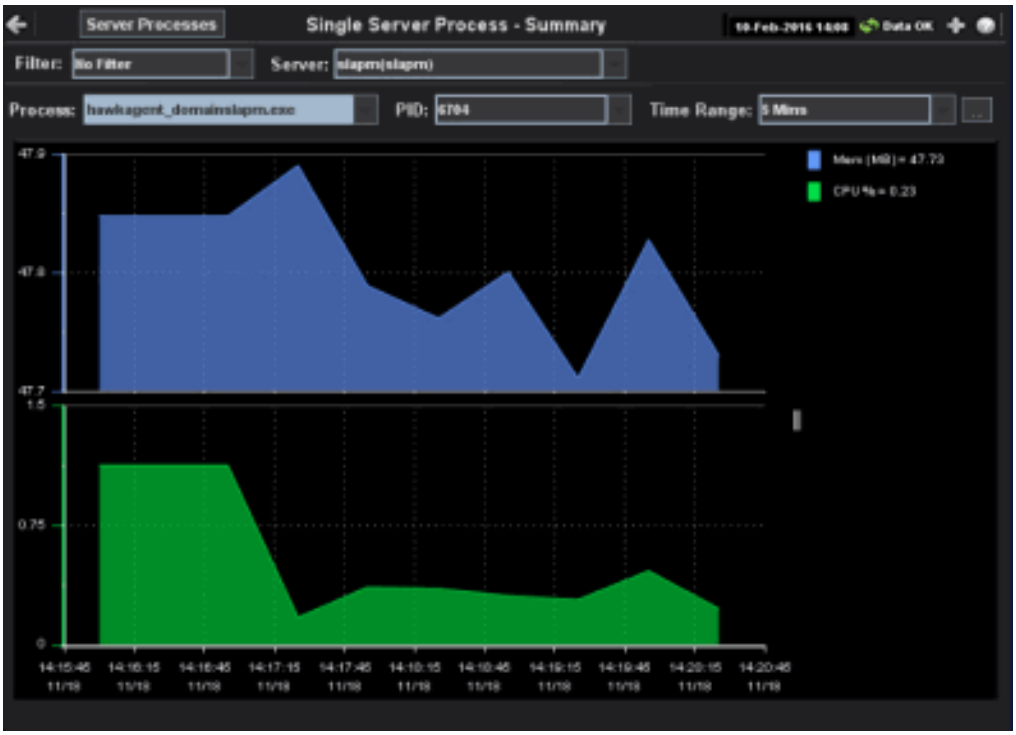
The display might include these filtering options:

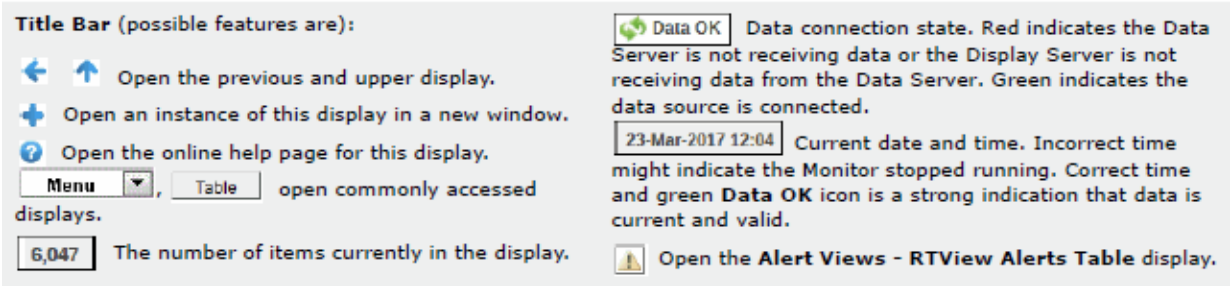
Filter:	Choose a filter to show data for in the display. By default, the **Filter:** drop-down menu only contains the **No Filter** option. To create your own filtering options, see **Creating Customized Filters** in the User's Guide.
Server:	Choose a server to see metrics for.

Single Server Process - Summary

Detailed information about a single operating system process running on a single BW Server.

NOTE: By default, this display is not enabled. For details, see **Enable BW Servers**.





Filter By:

The display might include these filtering options:

Filter: Choose a filter to show data for in the display. By default, the **Filter:** drop-down menu only contains the **No Filter** option. To create your own filtering options, see **Creating Customized Filters** in the User's Guide.

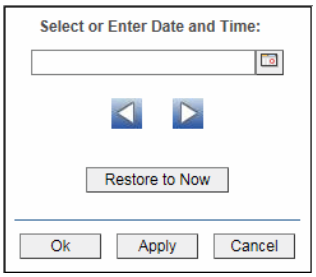
Server: Choose a server to see metrics for.

Process: Choose a server process.

PID: Choose a server PID.

Time Range

Select a time range from the drop down menu varying from **2 Minutes** to **Last 7 Days**, or display **All Data**. To specify a time range, click Calendar .



By default, the time range end point is the current time. To change the time range end point, click Calendar and select a date and time from the calendar or enter the date and time in the text field using the following format: **MMM dd, YYYY HH:MM**. For example, **Aug 21, 2011 12:24 PM**.

Use the navigation arrows to move forward or backward one time period. NOTE: The time period is determined by your selection from the **Time Range** drop-down menu.

Click **Restore to Now** to reset the time range end point to the current time.

Hawk Views

These displays present performance data for your TIBCO BusinessWorks Hawk hosts. Use these displays to monitor performance and utilization for all your TIBCO Hawk hosts and agents.

Displays in this View are:

- "All Hosts Heatmap" on page 284
- "All Hosts Table" on page 285
- "All Hosts Grid" on page 287
- "All Host Processes Table" on page 289

- "Host Summary" on page 290
- "All Hawk Agents Table" on page 293

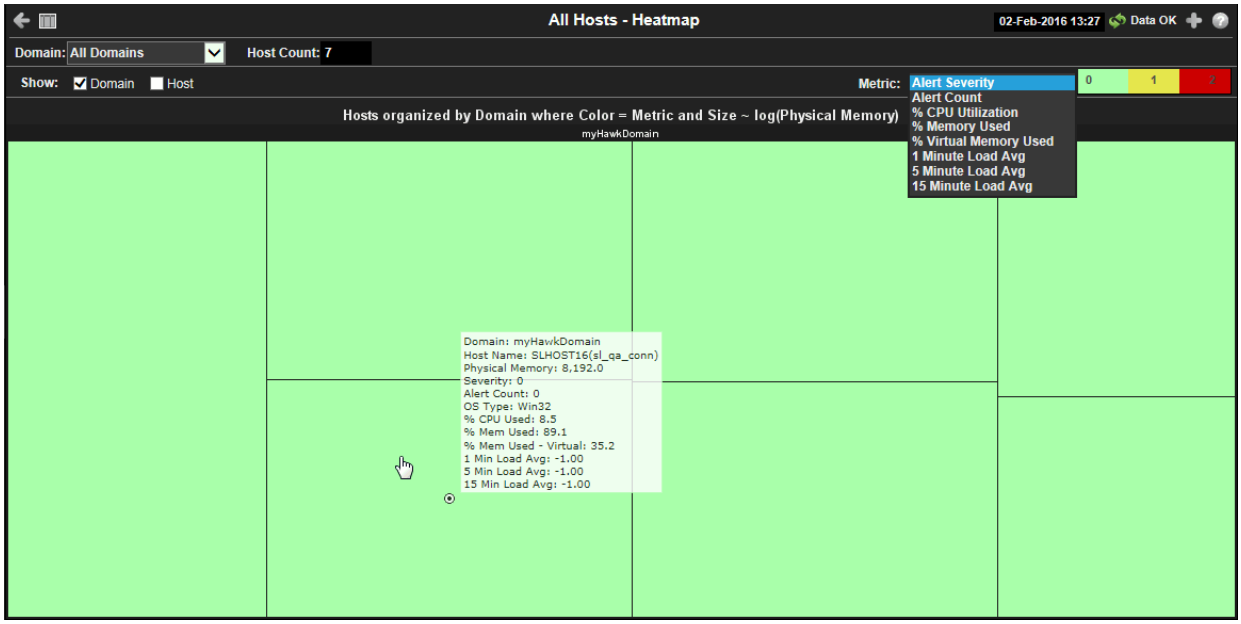
All Hosts Heatmap

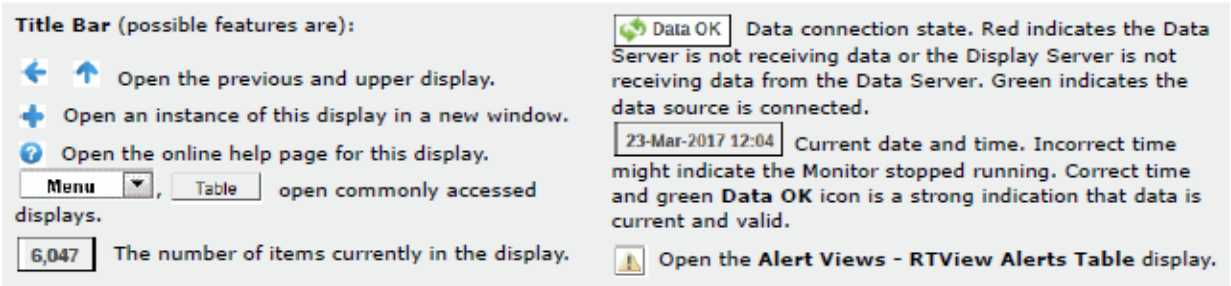
View the most critical alert states pertaining to your hosts. Use this display to quickly identify hosts with critical alerts.

Each rectangle in the heatmap represents a host. The rectangle color indicates the most critical alert state associated with the host for the selected **Metric**. The rectangle size represents the amount of physical memory present on the host; a larger size is a larger value.

Choose a domain or **All Domains** from the **Domain** drop-down menu to filter data shown in the display. Choose a different metric to display from the **Metric** drop-down menu. Mouse over a rectangle to see additional metrics. By default, this display shows **Alert Severity**.

Drill-down and investigate a host by clicking a rectangle in the heatmap to view details in the **Host Summary** display.



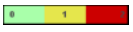


Filter By:

The display might include these filtering options:

Domain:	Choose a domain to show data for in the display. Domain names are specified when your administrator configures your Data Server to collect Hawk data, and applies to all host data collected from Hawk by that Data Server.

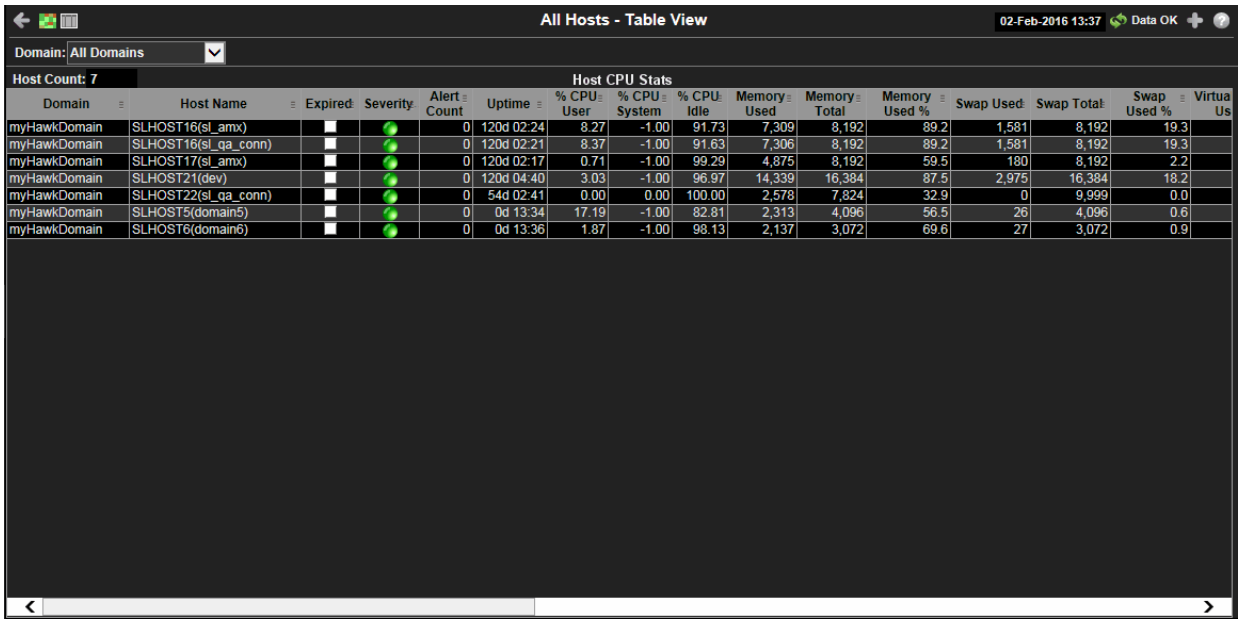
Fields and Data:

Host Count:		The total number of hosts currently shown in the display.
Show:	**Domain**	When selected, includes the Domain name in the display.
	Host	When selected, includes the Host name in the display.
Metric		Choose a metric to view in the display.
	Alert Severity	The maximum level of alerts in the heatmap rectangle. Values range from **0** - **2**, as indicated in the color gradient  bar, where **2** is the highest Alert Severity: ● Red indicates that one or more metrics exceeded their ALARM LEVEL threshold. ● Yellow indicates that one or more metrics exceeded their WARNING LEVEL threshold. ● Green indicates that no metrics have exceeded their alert thresholds.
	Alert Count	The total number of critical and warning alerts in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average alert count.
	% CPU Utilization	The percent of CPU used in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
	% Memory Used	The percent of memory used in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
	% Virtual Memory Used	The percent of virtual memory used in the heatmap rectangle. The color gradient  bar, populated by the current heatmap, shows the value/color mapping. The numerical values in the gradient bar range from **0** to the maximum count of alerts in the heatmap. The middle value in the gradient bar indicates the average count.
	1 Minute Load Avg	The average number of processes running over 1 minute.
	5 Minute Load Avg	The average number of processes running over 5 minutes.
	15 Minute Load Avg	The average number of processes running over 15 minutes.

All Hosts Table

View host utilization data in a tabular format. Use this display to see all available data for this View.

Each row in the table is a different host. Choose a domain or **All Domains** from the **Domain** drop-down menu. Click a column header to sort column data in numerical or alphabetical order. Drill-down and investigate by clicking a row to view details for the selected application in the **Host Summary** display.



Title Bar (possible features are):

Open the previous and upper display.

Open an instance of this display in a new window.

Open the online help page for this display.

Menu, Table open commonly accessed displays.

6,047 The number of items currently in the display.

Data OK Data connection state. Red indicates the Data Server is not receiving data or the Display Server is not receiving data from the Data Server. Green indicates the data source is connected.

23-Mar-2017 12:04 Current date and time. Incorrect time might indicate the Monitor stopped running. Correct time and green **Data OK** icon is a strong indication that data is current and valid.

Open the **Alert Views - RTView Alerts Table** display.

Filter By:

The display might include these filtering options:

Domain: Choose a domain to show data for in the display.

Fields and Data:

Host Count: The total number of hosts in the table.

Table:

Each row in the table is a different host.

Domain	The domain in which the host resides. Domain names are specified when your administrator configures your Data Server to collect Hawk data, and applies to all host data collected from Hawk by that Data Server.
Host Name	The name of the host.
Expired	When checked, data has not been received from this host in the specified amount of time. The host will be removed from the Monitor in the specified amount of time. The default setting is **60** seconds.
Severity	The maximum level of alerts in the row. Values range from **0** - **2**, as indicated in the color gradient bar, where **2** is the highest Alert Severity: Red indicates that one or more metrics exceeded their ALARM LEVEL threshold. Yellow indicates that one or more metrics exceeded their WARNING LEVEL threshold. Green indicates that no metrics exceeded their alert thresholds.

Alert Count	The total number of active alerts associated with the host.
Uptime	The amount of time the application has been running, in the following format: **0d 00:00 <days>d <hours>:<minutes>:<seconds>** For example: **10d 08:41:38**
% CPU Used	The amount of CPU used, in percent.
% CPU System	The amount of CPU used, in percent.
% CPU Idle	The amount of CPU not used, in percent.
Memory Used	The amount of memory, in megabytes, currently used.
Memory Total	The total amount of memory, in megabytes.
Memory Used%	The amount of memory used, in percent.
Swap Used	The amount of swap space, in megabytes, currently used.
Swap Total	The total amount of swap space, in megabytes.
Swap Used %	The amount of swap space used, in percent.
Virtual Mem (ory) Used	The amount of virtual memory currently used, in megabytes.
Virtual Mem (ory) Total	The total amount of virtual memory, in megabytes.
Virtual Mem (ory) Used%	The amount of virtual memory used, in percent.
Load Avg 1 Minute	The average number of processes running over 1 minute.
Load Avg 5 Minute	The average number of processes running over 5 minutes.
Load Avg 15 Minute	The average number of processes running over 15 minutes.
OS Type	The type of operating system (for example, Linux, HP-UX, Windows 2003).
OS Description	The name of the operating system.
OS Version	The operating system version.
CPU Model	The CPU model.
# CPUs	The number of node connections.
Agent Type	The type of agent from which the data was collected: **HOSTMON** (a SL Host Agent), **Hawk**, **WMI** or **SNMP**.
Agent Class	The specific version of the agent software.
Source	The name of the SL Data Server where the host data was collected.
Timestamp	The date and time the data was last updated.

All Hosts Grid

This grid provides a list view of utilization metrics for all hosts. Use this display to track and view in parallel the general performance of your hosts. Drill down and investigate by clicking a host to view details in the **Host Summary** display.

Title Bar (possible features are):

Open the previous and upper display.

Open an instance of this display in a new window.

Open the online help page for this display.

Menu, Table open commonly accessed displays.

6,047 The number of items currently in the display.

Data OK Data connection state. Red indicates the Data Server is not receiving data or the Display Server is not receiving data from the Data Server. Green indicates the data source is connected.

23-Mar-2017 12:04 Current date and time. Incorrect time might indicate the Monitor stopped running. Correct time and green **Data OK** icon is a strong indication that data is current and valid.

Open the **Alert Views - RTView Alerts Table** display.

Filter By:

The display might include these filtering options:

Domain:	Choose a domain to show data for in the display. Domain names are specified when your administrator configures your Data Server to collect Hawk data, and applies to all host data collected from Hawk by that Data Server.
Host Count	Displays the number of hosts (including expired hosts) listed in the display.
Time Range:	Choose a time range to show data for in the display. Options are: **All Data**, **2 Mins**, **5 Mins**, **20 Mins**, **1 Hour, 2 Hours**, **4 Hours**, **8 Hours**, **24 Hours**, **2 Days and 7 Days**.

Grid

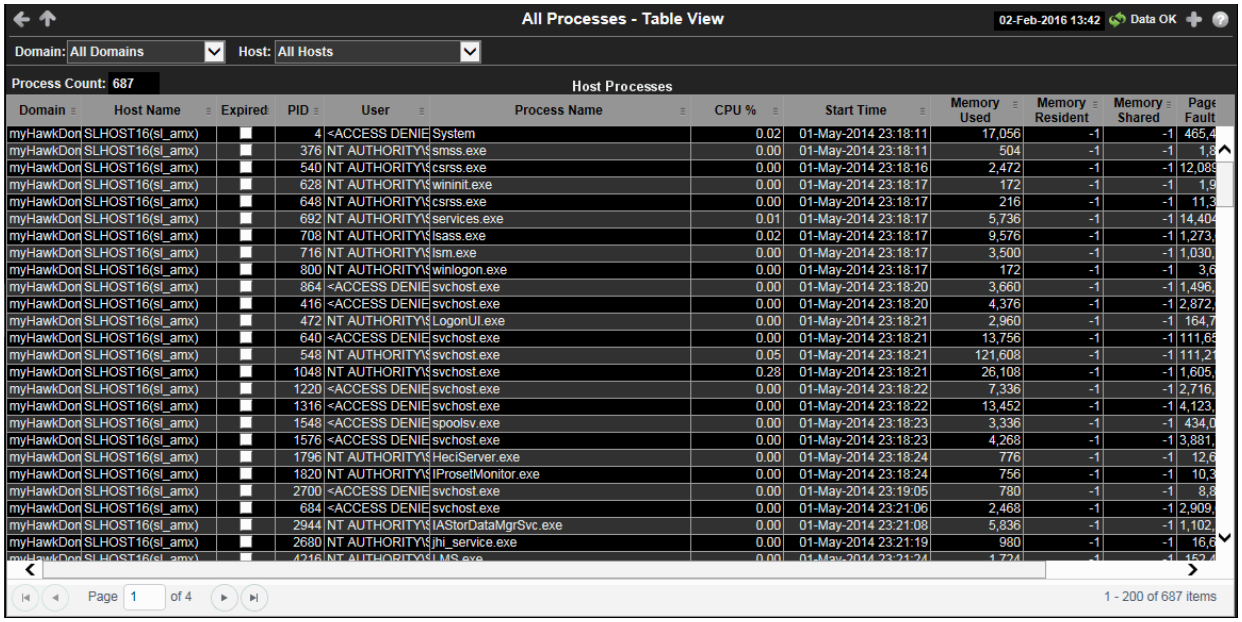
Utilization data shown for hosts in the selected domain.

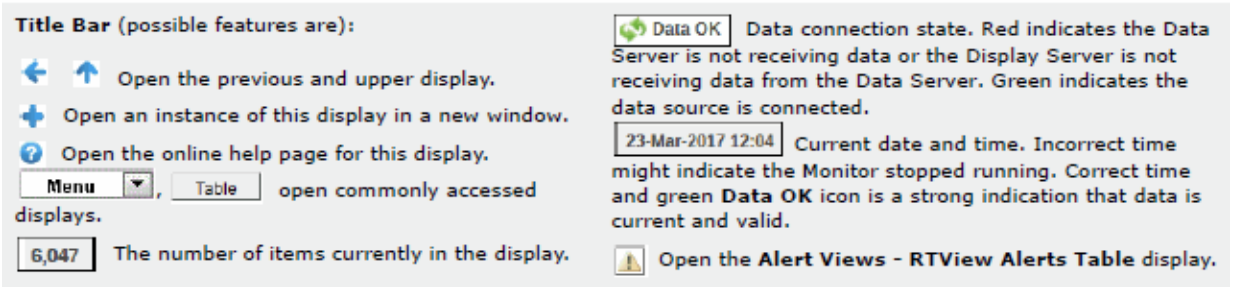
Host Name		The name of the host.
OS Type		The name of the operating system.
Uptime		The amount of time (days, hours, seconds) the operating system has been running.
Phys Mem		The amount of physical memory used, in megabytes.
Virtual Mem		The amount of virtual memory used, in megabytes.
Load Avg	**1**	The average number of processes running over 1 minute.
	5	The average number of processes running over 5 minutes.

	15	The average number of processes running over 15 minutes.
CPU Usage	The bar graph shows the amount of CPU currently used.	
VMem Usage	The bar graph shows the amount of virtual memory currently used.	
Trend Graphs		
	CPU	Traces the amount of CPU currently used.
	VM Usage	Traces the amount of virtual memory currently used.
	Rx KB/s	Traces the amount data currently being received per second.
	Tx KB/s	Traces the amount data currently being transmitted per second.

All Host Processes Table

View host utilization data in a tabular format. Use this display to see all available data for this View. Each row in the table is a different host. Choose a domain or **All Domains** and a host or **All Hosts** from the drop-down menus. Click a column header to sort column data in numerical or alphabetical order. Drill-down and investigate by clicking a row to view details for the selected application in the **Host Summary** display.





Filter By:
The display might include these filtering options:

Domain:	Choose a domain to show data for in the display. Domain names are specified when your administrator configures your Data Server to collect Hawk data, and applies to all host data collected from Hawk by that Data Server.
Host:	Choose a host to show data for in the display.

Fields and Data:

Process Count:	The total number of processes in the table.

Table:
Each row in the table is a different host.

Domain	The domain in which the host resides.
Host Name	The name of the host.
Expired	When checked, data has not been received from this host in the specified amount of time. The host will be removed from the Monitor in the specified amount of time. The default setting is 60 seconds.
PID	The process ID.
User	The user name.
Process Name	The name of the process.
CPU%	The amount of CPU used, in percent.
Start Time	The host start time, in the following format: **0d 00:00 <days>d <hours>:<minutes>:<seconds>** For example: **10d 08:41:38**
Memory Used	The amount of memory currently used, in megabytes.
Memory Resident	The amount of memory currently used by the process that resides in physical memory and is not paged out. Set to **-1** when the data is not available from an agent. (Hawk does not provide this data.)
Memory Shared	The amount of physical memory that is shared with other processes. Set to **-1** when the data is not available from an agent. (Hawk does not provide this data.)
Page Faults	The number of page faults.
Page Faults /sec	The number of page faults per second.
Timestamp	The date and time the data was last updated.

Host Summary

This display provides a detailed view of utilization metrics for a single server.

Title Bar (possible features are):

Open the previous and upper display.

Open an instance of this display in a new window.

Open the online help page for this display.

Menu, Table open commonly accessed displays.

6,047 The number of items currently in the display.

Data OK Data connection state. Red indicates the Data Server is not receiving data or the Display Server is not receiving data from the Data Server. Green indicates the data source is connected.

23-Mar-2017 12:04 Current date and time. Incorrect time might indicate the Monitor stopped running. Correct time and green **Data OK** icon is a strong indication that data is current and valid.

Open the **Alert Views - RTView Alerts Table** display.

Filter By:

The display might include these filtering options:

Domain:	Choose a domain to show data for in the display. Domain names are specified when your administrator configures your Data Server to collect Hawk data, and applies to all host data collected from Hawk by that Data Server.
Host:	Choose a host to show data for in the display.
Expired	When checked, data has not been received from this host in the specified amount of time. The host will be removed from the Monitor in the specified amount of time. The default setting is **60** seconds.
Last Update	The time the display was last updated.

Fields and Data:

Data describes the selected host except where noted.

OS:	The operating system.	
Version:	The operating system version.	
Uptime:	The number of days, hours and minutes since started.	
	#CPUs	The number of node connections.
CPU Type:	The type of CPU.	
%CPU	**User**	The amount of CPU used by the user, in percent.
	System	The amount of CPU used by the system, in percent.
	Idle	The amount of CPU that is not used, in percent.
Physical	**Used**	The amount of physical memory used, in kilobytes.

Memory		
	Total(MB)	The amount of physical memory available, in kilobytes.
	%Used	The amount of physical memory used, in percent.
Virtual Memory	**Used**	The amount of virtual memory used, in kilobytes.
	Total(MB)	The amount of virtual memory available, in kilobytes.
	%Used	The amount of virtual memory used, in percent.
Processes	The number of processes running.	
Load Avg:	**1 Min**	The average number of processes running over 1 minute.
	5 Min	The average number of processes running over 5 minutes.
	15 Min	The average number of processes running over 15 minutes.
Storage	**File System**	The amount of storage space used for the file system, in kilobytes.
	Mount Point	The name used by the operating system to mount and provide an entry point to other storage volumes.
	%Used	The amount of storage space used, in percent.
Network	**ifName**	The name assigned to the network interface by the operating system.
	RxKB/s	The amount of network data received per second, in kilobytes.
	TxKB/s	The amount of network data transmitted per second, in kilobytes.

Trend Graphs
Traces metrics for the selected host.

- **CPU% Used**: The amount of CPU used, in percent.
- **Mem Total:** The amount of available memory, in kilobytes.
- **Mem Used**: The amount of memory used, in kilobytes.
- **Net Rx KB/s**: The amount of network data received per second, in kilobytes.
- **Net Tx KB/s**: The amount of network data transmitted per second, in kilobytes.

Log Scale

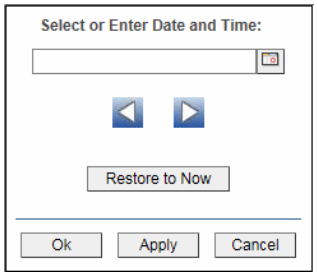
Select to enable a logarithmic scale. Use **Log Scale** to see usage correlations for data with a wide range of values. For example, if a minority of your data is on a scale of tens, and a majority of your data is on a scale of thousands, the minority of your data is typically not visible in non-log scale graphs. **Log Scale** makes data on both scales visible by applying logarithmic values rather than actual values to the data.

Base at Zero

Select to use zero (**0**) as the Y axis minimum for all graph traces.

Time Range

Select a time range from the drop down menu varying from **2 Minutes** to **Last 7 Days**, or display **All Data**. To specify a time range, click Calendar ... .



By default, the time range end point is the current time. To change the time range end point, click Calendar ... and select a date and time from the calendar or enter the date and time in the text field using the following format: **MMM dd, YYYY HH:MM**. For example, **Aug 21, 2011 12:24 PM**.

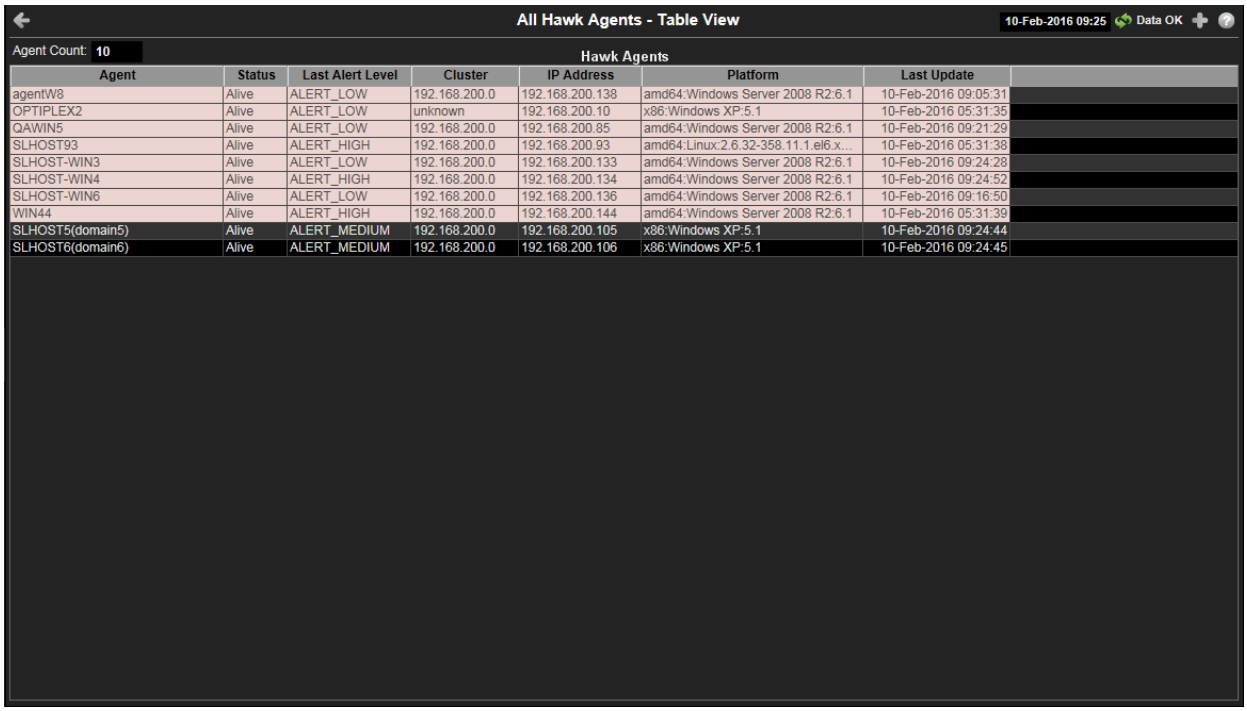
Use the navigation arrows to move forward or backward one time period. NOTE: The time period is determined by your selection from the **Time Range** drop-down menu.

Click **Restore to Now** to reset the time range end point to the current

time.

All Hawk Agents Table

This table provides a list of TIBCO BusinessWorks Hawk agents as well as network connectivity details about each agent.



Agent	Status	Last Alert Level	Cluster	IP Address	Platform	Last Update
agentW8	Alive	ALERT_LOW	192.168.200.0	192.168.200.138	amd64:Windows Server 2008 R2:6.1	10-Feb-2016 09:05:31
OPTIPLEX2	Alive	ALERT_LOW	unknown	192.168.200.10	x86:Windows XP:5.1	10-Feb-2016 05:31:35
QAWIN5	Alive	ALERT_LOW	192.168.200.0	192.168.200.85	amd64:Windows Server 2008 R2:6.1	10-Feb-2016 09:21:29
SLHOST93	Alive	ALERT_HIGH	192.168.200.0	192.168.200.93	amd64:Linux:2.6.32-358.11.1.el6.x...	10-Feb-2016 05:31:38
SLHOST-WIN3	Alive	ALERT_LOW	192.168.200.0	192.168.200.133	amd64:Windows Server 2008 R2:6.1	10-Feb-2016 09:24:28
SLHOST-WIN4	Alive	ALERT_HIGH	192.168.200.0	192.168.200.134	amd64:Windows Server 2008 R2:6.1	10-Feb-2016 09:24:52
SLHOST-WIN6	Alive	ALERT_LOW	192.168.200.0	192.168.200.136	amd64:Windows Server 2008 R2:6.1	10-Feb-2016 09:16:50
WIN44	Alive	ALERT_HIGH	192.168.200.0	192.168.200.144	amd64:Windows Server 2008 R2:6.1	10-Feb-2016 05:31:39
SLHOST5(domain5)	Alive	ALERT_MEDIUM	192.168.200.0	192.168.200.105	x86:Windows XP:5.1	10-Feb-2016 09:24:44
SLHOST6(domain6)	Alive	ALERT_MEDIUM	192.168.200.0	192.168.200.106	x86:Windows XP:5.1	10-Feb-2016 09:24:45

Title Bar (possible features are):

Open the previous and upper display.

Open an instance of this display in a new window.

Open the online help page for this display.

Menu, Table open commonly accessed displays.

6,047 The number of items currently in the display.

Data OK Data connection state. Red indicates the Data Server is not receiving data or the Display Server is not receiving data from the Data Server. Green indicates the data source is connected.

23-Mar-2017 12:04 Current date and time. Incorrect time might indicate the Monitor stopped running. Correct time and green **Data OK** icon is a strong indication that data is current and valid.

Open the **Alert Views - RTView Alerts Table** display.

Fields and Data:

Agent Count: The total number of agents in the table.

Table:

Each row in the table is a different agent.

Domain	The domain in which the host resides. Domain names are specified when your administrator configures your Data Server to collect Hawk data, and applies to all host data collected from Hawk by that Data Server.
Agent	The name for the agent which is composed of the hostname and Hawk domain (in parenthesis). Agent names which do not contain an explicit Hawk domain are members of the "default" domain.
Status	The agent status, either **Alive** or **Expired**.

Last Alert Level	The most recent and most critical alert level.
Cluster	The IP address of the cluster to which this agent belongs.
IP Address	The IP subnet address for the group of machines to which this agent belongs.
Platform	The physical CPU class and operating system version.
Last Update	The date and time the row data was last updated.

Alert Views

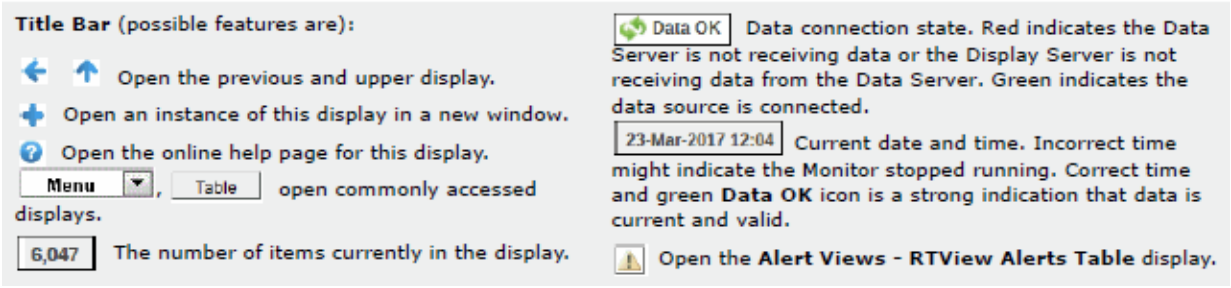
These displays present detailed information about all alerts that have occurred in your system. These displays present performance data for your BusinessWorks system. Displays in this View are:

- "Alert Detail Table" on page 294: Time ordered list of all alerts that have occurred in the system.
- "Hawk Alerts Table " on page 297: Summary of alerts delivered to Hawk agent according to its active rulebases.

Alert Detail Table

Use this display to track and manage all alerts that have occurred in the system, add comments, acknowledge or assign Owners to alerts.

Each row in the table is a different active alert. Select one or more rows, right-click and choose **Alert** to see all actions that you can perform on the selected alert(s). Choose **Alert / Set Filter Field** to apply the selected cell data to the **Field Filter** and **Search Text** fields. Or enter filter criteria directly in the **Field Filter** and **Search Text** fields. Click **Clear** to clear the **Field Filter** and **Search Text** fields. Click Sort to order column data.



Row Color Code:

Tables with colored rows indicate the following:

- Red indicates that one or more alerts exceeded their ALARM LEVEL threshold in the table row.
- Yellow indicates that one or more alerts exceeded their WARNING LEVEL threshold in the table row.
- Green indicates that no alerts exceeded their WARNING or ALARM LEVEL threshold in the table row.

Fields and Data

This display includes:

Alert Name Filter	Select from a list of alert types or select All Alert Types. Filters limit display content and drop down menu selections to only those items that pass through the selected filter's criteria. Therefore if no items match the filter, you may see nothing in a given display and may not have any options available in the drop-down menu(s). **NOTE:** Filter selection is disabled on drill down summary displays.
Show Critical Alerts Only	If selected, only currently critical alerts are shown in the table. Otherwise, all active alerts are shown in the table.
Show Cleared Alerts	If selected, cleared alerts are shown in the table.
Alert Text Filter	Enter all or part of the Alert Text to view specific alerts. For example, High selects and displays all alerts that include High in the Alert Text. **NOTE:** Wild card characters are supported.
Owner Filter	Select the alert **Owner** to show alerts for in the table. **All** — Shows alerts for all Owners in the table: **Not Owned** and **Owned By Me** alerts. **Not Owned** — Shows only alerts without Owners in the table. **Owned By Me** — Shows only alerts for the current user in the table.
Show Acknowledged Alerts	If selected, acknowledged alerts are shown in the table.
Total	Total number of alerts.
Critical	Number of critical alerts.
Warning	Total number of alerts that are currently in a warning state.
Alert Settings Conn OK	The Alert Server connection state: Disconnected. Connected.

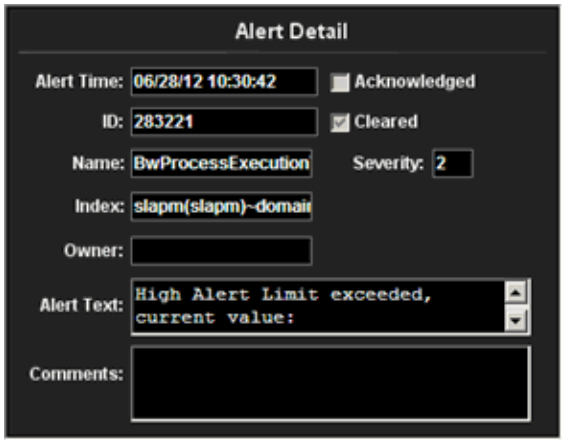
Alerts Table

This table lists all active alerts for the current filters.

Time	The time (Java format) that the alert was activated.
ID	A unique string identifier assigned to each activated alert.
Clr'd	When checked, this typically indicates that the alert has been resolved. An alert is automatically cleared when the value being monitored no longer in the alert threshold.
Ack'd	When checked, this typically indicates that the alert is being addressed.
Owner	The named owner assigned by the administrator.
Alert Name	The name of the alert. For a list of all alerts, see Alert Administration.
Alert Index	The IP address and port number for the source (application, server, and so forth) associated with the alert.
Alert Text	Descriptive text about the alert.
Severity	The severity of the alert: **0 = Normal** **1** = Warning / Yellow **2 = Alarm / Red** The color for the alert severity is shown by the row in the alert table.
Source	Name of RTView Data Server sending this data (or localhost).

Selected Alerts Lists the alerts selected in the table.

Acknowledge One Alert	Select one alert from the Current Alerts table and click to acknowledge.
Acknowledge Multiple Alerts	Select one or more alerts from the Current Alerts table and click to acknowledge.
Set Owner and Comments	Select one or more alerts from the Current Alerts table and click to open the Set Owner and Comments dialog. 
See Details	Select an alert from the Current Alerts table and click to open the Set Owner and Comments dialog.



Hawk Alerts Table

If you have designated Hawk Alerts for your BW Engines, you can view a summary of alerts delivered to a Hawk agent according to its active rulebases. To enable Hawk Alerts to be displayed throughout the Monitor, go to the Alert Administration display.

To filter unwanted alerts out of the Hawk cache data, enter the following into the **sample.properties** file (located in the project settings directory you created):

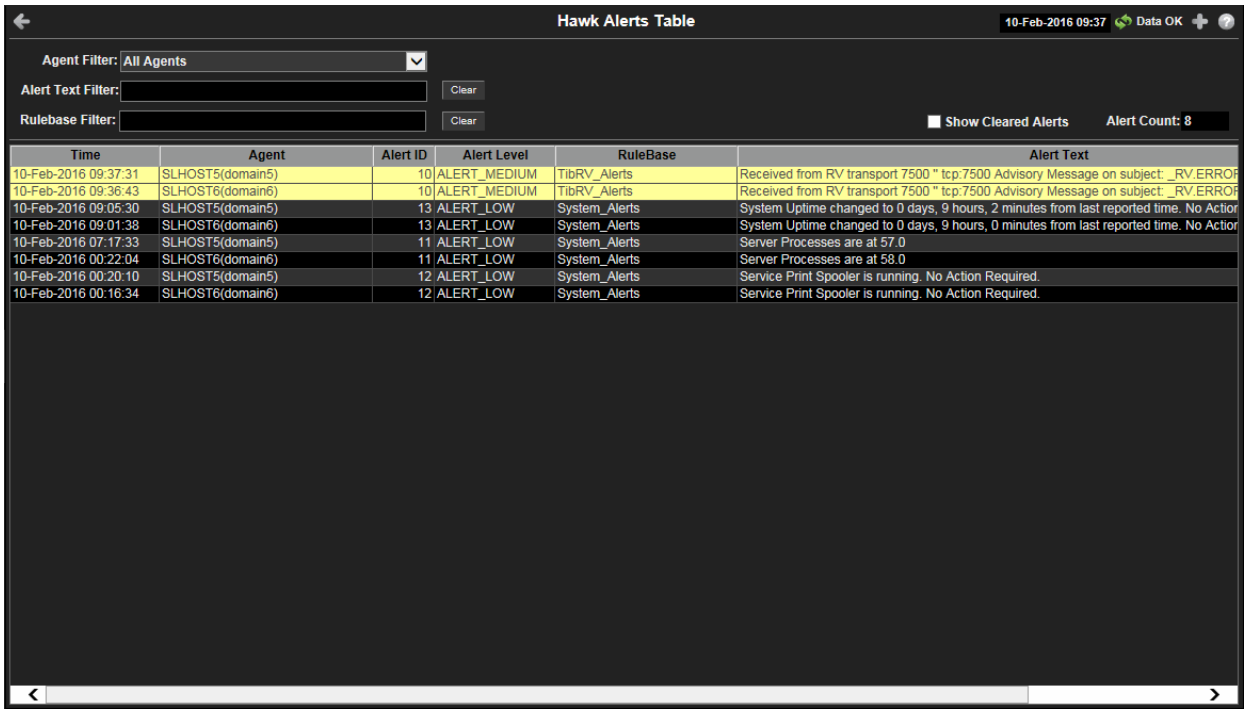
sl.rtview.sub=$hawkAlertTextFilterOut:AlertText

For example, to filter out all Hawk Alerts in which the AlertText contains Source you would enter the following:

sl.rtview.sub=$hawkAlertTextFilterOut:Source

The default time to remove cleared Hawk Alerts from the table is **3600** seconds. To adjust this setting, edit the following in **sample.properties**:

sl.rtview.sub=$hawkAlertTextFilterOut:3600



Title Bar (possible features are):

Open the previous and upper display.

Open an instance of this display in a new window.

Open the online help page for this display.

Menu, Table open commonly accessed displays.

6,047 The number of items currently in the display.

Data OK Data connection state. Red indicates the Data Server is not receiving data or the Display Server is not receiving data from the Data Server. Green indicates the data source is connected.

23-Mar-2017 12:04 Current date and time. Incorrect time might indicate the Monitor stopped running. Correct time and green **Data OK** icon is a strong indication that data is current and valid.

Open the **Alert Views - RTView Alerts Table** display.

Filter By:

The display might include these filtering options:

Agent Filter: Select from a list of agents or select All. This filter limits items in this table to contain only those Hawk Alerts from a particular Agent. Therefore if there are no alerts from the selected Agent, there may be nothing displayed in this table.

Alert Text Filter: Enter all or part of the Alert Text to view alerts on specific agents. For example, High selects and displays all alerts that include High in the Alert Text. NOTE: Wild card characters are supported.

Clear Removes Alert Text Filter and all agents for the selected **Agent Filter** are displayed.

Rulebase Filter: Enter all or part of the Rulebase to view alerts for specific agents. For example, System selects and displays all alerts that include System in the RuleBase column. NOTE: Wild card characters are supported.

Clear Removes Rulebase Filter and all agents for the selected **Agent Filter** are displayed.

Show Cleared Alerts If selected, both active and cleared alerts are shown in the table. Otherwise, only currently active alerts are shown in the table.

Alert Count: Number of alerts currently being displayed. NOTE: Alerts filtered out from this display are not removed from the data and therefore will still be included in other alert counts and displays throughout the Monitor.

Hawk Alerts Table:
Displays last 100 Hawk Alerts received from all BW Engine agents.

Time	Time (Java format) that the alert was activated.
Agent	Name of the agent that posted the alert.
AlertID	A unique string identifier assigned to each activated alert.
Alert Level	Rulebase state of the alert, where the values ALERT-LOW, ALERT-MEDIUM, ALERT-HIGH correspond to the TIBCO Hawk Console API.
RuleBase	Rulebase that posted the alert.
AlertText	Text for the alert.
Cleared	If selected, the alert has cleared.

Administration

These displays enable you to set alert thresholds, observe how alerts are managed, and view internal data gathered and stored by RTView (used for troubleshooting with SL Technical Support). Displays in this View are:

- "Alert Administration" on page 299: Displays active alerts and provides interface to modify and manage alerts.
- "RTView Cache Tables" on page 304: View cached data that RTView is capturing and maintaining, and use this data use this for debugging with SL Technical Support.
- "RTView Agent Administration" on page 306: Display information about RTView Agent data servers.

Alert Administration

Set global or override alert thresholds. Alert settings are global by default.

The table describes the global settings for all alerts on the system. To filter the alerts listed in the table, enter a string in the **Alert Filter** field and press **<enter>** or click elsewhere in the display. Filters are case sensitive and no wildcard characters are needed for partial strings. For example, if you enter Server in the **Alert Filter** field, it filters the table to show only alerts with **Server** in the name. Choose **Clear** to clear the filter.

Global Thresholds

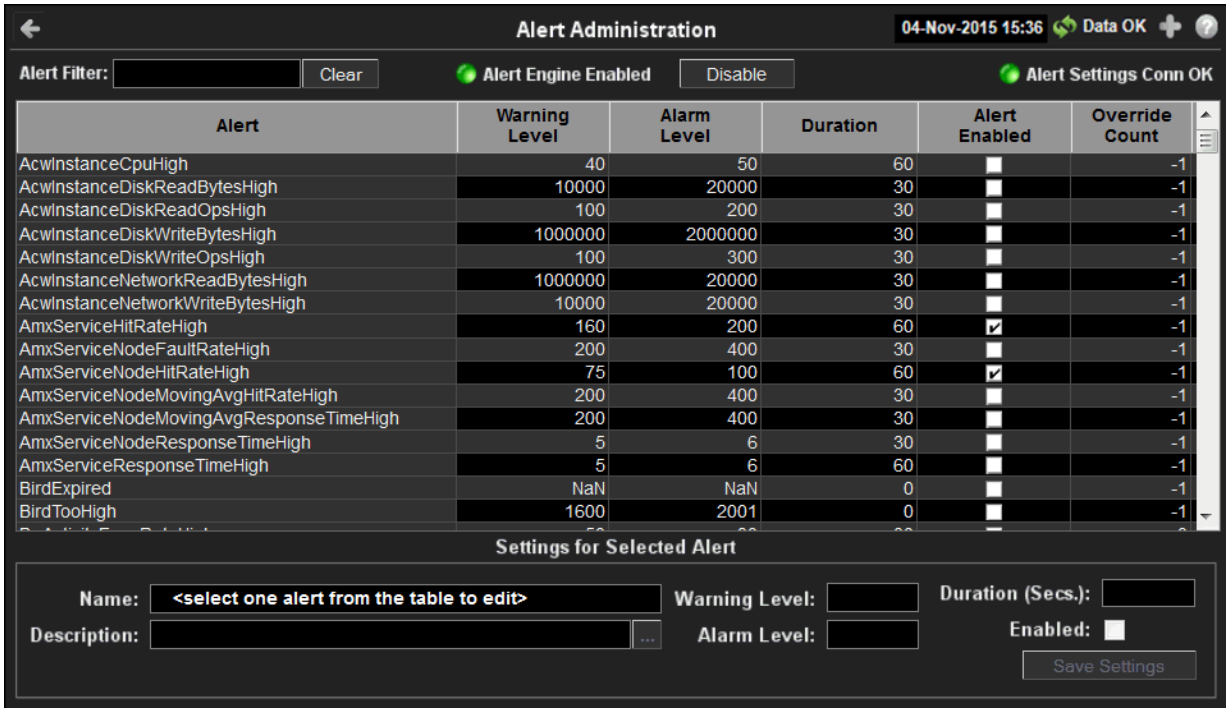
To set a global alert, select an alert from the **Active Alert Table**. The name of the selected alert populates the **Settings for Selected Alert Name** field. Edit the **Settings for Selected Alert** and click **Save Settings** when finished.

The manner in which global alerts are applied depends on the Solution Package. For example, the EMS Monitor Solution Package has queue alerts, topic alerts and server alerts. When a queue alert is applied globally, it is applied to all queues on all servers. Likewise, a server alert applies to all servers, and a topic alert applies to all topics on all servers.

Override Thresholds

Setting override alerts allows you to set thresholds for a single resource (for example, a single server). Override alerts are useful if the majority of your alerts require the same threshold setting, but there are other alerts that require a different threshold setting. For example, you might not usually be concerned with execution time at a process level, but perhaps certain processes are critical. In this case, you can apply alert thresholds to each process individually.

To apply an individual alert you Index the Monitored Instance or resource. The Index Types available are determined by the Solution Package installed. For example, the EMS Monitor package lets you set an alert for a specific *topic* on a specific *server* (such as the PerServerTopic Index option), rather than for all topics on all servers.



Title Bar (possible features are):

Open the previous and upper display.

Open an instance of this display in a new window.

Open the online help page for this display.

Menu, Table open commonly accessed displays.

6,047 The number of items currently in the display.

Data OK Data connection state. Red indicates the Data Server is not receiving data or the Display Server is not receiving data from the Data Server. Green indicates the data source is connected.

23-Mar-2017 12:04 Current date and time. Incorrect time might indicate the Monitor stopped running. Correct time and green **Data OK** icon is a strong indication that data is current and valid.

Open the **Alert Views - RTView Alerts Table** display.

Fields and Data
This display includes:

Alert Filter	Enter the (case-sensitive) string to filter the table by the **Alert** table column value. **NOTE:** Partial strings can be used without wildcard characters. Press **<enter>** or click elsewhere in the display to apply the filter.
Clear	Clears the **Alert Filter** entry.
Alert Engine Enabled	Alerting is disabled. Alerting is enabled (by default).
Disable	Suspends all alerting.
Alert Settings Conn OK	The Alert Server connection state:

Disconnected.

Connected.

Active Alert Table

This table describes the global settings for all alerts on the system. Select an alert. The name of the selected alert populates the **Settings for Selected Alert Name** field (in the lower panel). Edit **Settings for Selected Alert** fields and click **Save Settings**.

NOTE: To filter the alerts shown in the table by Solution Package, use the **$rtvAlertPackageMask** substitution.

Alert	The name of the alert.
Warning Level	The global warning threshold for the selected alert. When the specified value is exceeded a warning is executed.
Alarm Level	The global alarm threshold for the selected alert. When the specified value is exceeded an alarm is executed.
Duration (Secs)	The amount of time (in seconds) that the value must be above the specified Warning Level or Alarm Level threshold before an alert is executed. **0** is for immediate execution.
Alert Enabled	When checked, the alert is enabled globally.
Override Count	The number of times thresholds for this alert have been defined individually in the **Tabular Alert Administration** display.

Settings for Selected Alert

To view or edit global settings, select an alert from the **Active Alert Table**. Edit the **Settings for Selected Alert** fields and click **Save Settings** when finished.

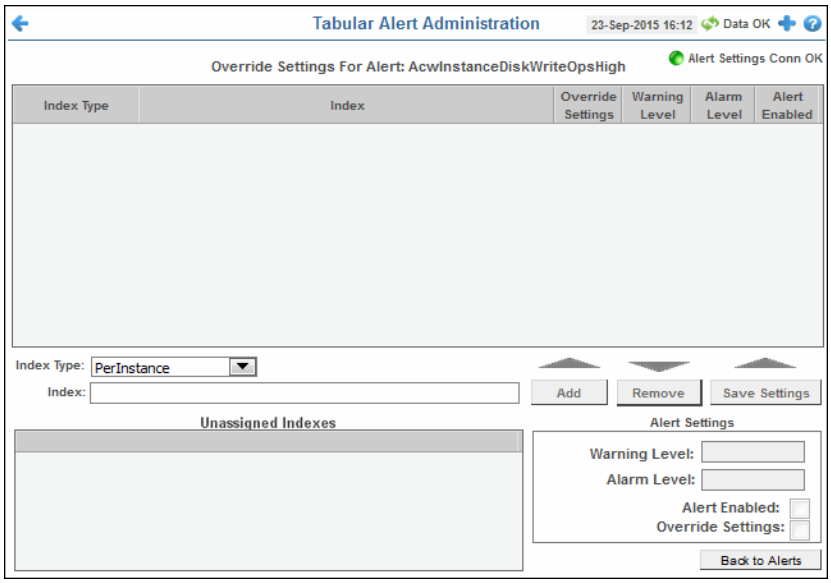
To set override alerts, click on **Override Settings** to open the **Tabular Alert Administration** display.

Name	The name of the alert selected in the **Active Alert Table**.
Description	Description of the selected alert. Click Calendar  for more detail.
Warning Level	Set the Global warning threshold for the selected alert. When the specified value is exceeded a warning is executed. To set the warning to occur sooner, reduce the Warning Level value. To set the warning to occur later, increase the Warning Level value. **NOTE:** For low value-based alerts (such as **EmsQueuesConsumerCountLow**), to set the warning to occur sooner, increase the Warning Level value. To set the warning to occur later, reduce the Warning Level value.
Alarm Level	Set the Global alarm threshold for the selected alert. When the specified value is exceeded an alarm is executed. To set the alarm to occur sooner, reduce the Alarm Level value. To set the warning to occur later, increase the Alarm Level value. **NOTE:** For low value-based alerts (such as **EmsQueuesConsumerCountLow**), to set the alarm to occur sooner, increase the Alarm Level value. To set the alarm to occur later, reduce the Alarm Level value.
Duration	Set the amount of time (in seconds) that the value must be above the specified Warning Level or Alarm Level threshold before an alert is executed. **0** is for immediate execution. This setting is global.
Enabled	Check to enable alert globally.
Save Settings	Click to apply alert settings.
Override Settings	Click to open the **Tabular Alert Administration** display to set override alerts on the selected alert.

Tabular Alert Administration

Set override alerts (override global alert settings). This display opens when you select an alert in the **Alert Administration** display and then select **Override Settings**.

For step-by-step instructions setting thresholds for individual alerts, see **Setting Override Alerts**..



Fields and Data
This display includes:

Alert Settings Conn OK
The connection state.
- No servers are found.
- One or more servers are delivering data.

Override Settings For Alert:(name)

This table lists and describes alerts that have override settings for the selected alert. Select a row to edit alert thresholds. The selected item appears in the Index field. Edit settings in the Alert Settings fields, then click Save Settings.

Index Type
Select the type of alert index to show in the Values table. Options in this drop-down menu are populated by the type of alert selected, which are determined by the Package installed. For example, with the EMS Monitor package the following Index Types are available:
- PerServer: Alert settings are applied to a specific server.
- PerQueue: Alert settings are applied to the queue on each server that has the queue defined.
- PerServerQueue: Alert settings are applied to a single queue on a specific server.
- PerTopic: Alert settings are applied to the topic on each server that has the topic defined.
- PerServerTopic: Alert settings are applied to a single topic on a specific server.

Index
The value of the index column.

Override Settings
When checked, the override settings are applied.

	Alert Enabled	When checked, the alert is enabled.
Index Type		Select the index type. The index type specifies how to apply alert settings. For example, to a queue (topic or JVM, and so forth) across all servers, or to a queue on a single server. NOTE: Options in this drop-down menu are populated by the type of alert selected from the Alert Administration display. Index Types available depend on the Package installed.
Index		The selected index column to be edited. This field is populated by the selection made in the **Unassigned Indexes** table.
Unassigned Indexes		This table lists all possible indexes corresponding to the Index Type chosen in the drop-down list. Select a row to apply individual alert thresholds. The selected item appears in the **Index** field. Edit settings in the **Alert Settings** fields, then click **Add**.
Add		Click to add changes made in **Alert Settings**, then click **OK** to confirm.
Remove		Click to remove an alert selected in the **Index Alert Settings** table, then click **OK** to confirm.
Save Settings		Click to save changes made to alert settings.
Alert Settings		
Select a topic, server or queue from the **Unassigned Indexes** table and edit the following settings.		
	Warning Level	Set the warning threshold for the selected alert. When the specified value is exceeded a warning is executed. To set the warning to occur sooner, reduce the Warning Level value. To set the warning to occur later, increase the Warning Level value. NOTE: For low value-based alerts (such as **EmsQueuesConsumerCountLow**), to set the warning to occur sooner, increase the Warning Level value. To set the warning to occur later, reduce the Warning Level value. **Click Save Settings to save settings.**
	Alarm Level	Set the alarm threshold for the selected alert. When the specified value is exceeded an alarm is executed. To set the alarm to occur sooner, reduce the Alarm Level value. To set the warning to occur later, increase the Alarm Level value. NOTE: For low value-based alerts (such as **EmsQueuesConsumerCountLow**), to set the alarm to occur sooner, increase the Alarm Level value. To set the alarm to occur later, reduce the Alarm Level value. Click **Save Settings** to save settings.
	Alert Enabled	Check to enable the alert, then click **Save Settings**.
	Override Settings	Check to enable override global setting, then click **Save Settings**.
Back to Alerts		Returns to the **Administration** - **Alert Administration** display.

Setting Override Alerts

Perform the following steps to set an override alert. Index Types available depend on the Solution Package installed. In this example, we use the EMS Monitor Package to illustrate.

> **Note:** To turn on an alert, both **Alert Enabled** and **Levels Enabled** must be selected.

To turn on/off, change threshold settings, enable/disable or remove an alert on a single resource:

1. In the **Alert Administration** display, select a tabular alert in the **Active Alert Table** and click **Override Settings**. The **Tabular Alert Administration** display opens.

> **Note:** Alerts that do not support overrides have a value of **-1** for the **Override Count** column and the **Override Settings** option is not present when you select such an alert.

2. In the **Tabular Alert Administration** display, select the Index type from the **Index Type** drop-down menu (options are populated by the type of alert you previously selected). For example, with the EMS Monitor package, select PerServerQueue, PerServerTopic or PerServer. NOTE: If you select PerServerQueue or PerServerTopic, the alert settings are applied to the queue or topic on a single server.

3. In the **Unassigned Indexes** table, select the item you want to apply an override alert setting to, click **Add** and **OK** in the confirmation dialog. After a few moments the override setting appears in the **AlertLevels** table.

4. Select the item in the **AlertLevels** table.

5. In the Alert Settings panel (lower right), if needed, modify the Warning Level and Alarm Level settings.

6. In the **Alert Settings** panel, set the following as appropriate.
   - To turn on the alert for this index with the given thresholds:

   Alert Enabled Select this option.

   Override Settings Select this option.

   NOTE: To turn on an alert, both **Alert Enabled** and **Override Settings** must be selected.
   - To turn off the alert for only this index (global alert thresholds will no longer apply to this index):

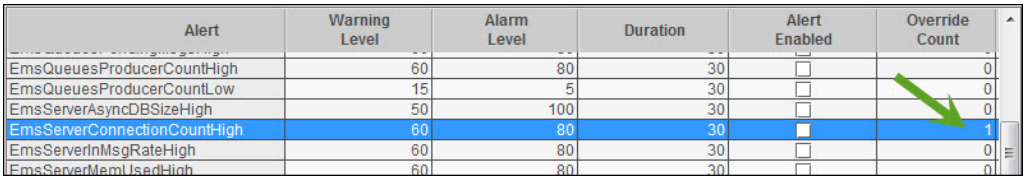
   Alert Enabled Deselect this option.

   Override Settings Select this option.
   - To no longer evaluate this indexed alert and revert to global settings (or, optionally, Remove it if it is never to be used again):

   Alert Enabled Not used.

   Override Settings Deselect this option.

7. Click **Save Settings**. In a few moments the modifications are updated and a new record appears in the **AlertLevels** table. For example, in the following figure, the EmsServerConnectionCountHigh alert has a new override applied. New overrides increment the alert **Override Count** in the **ALERTLEVELS** table.

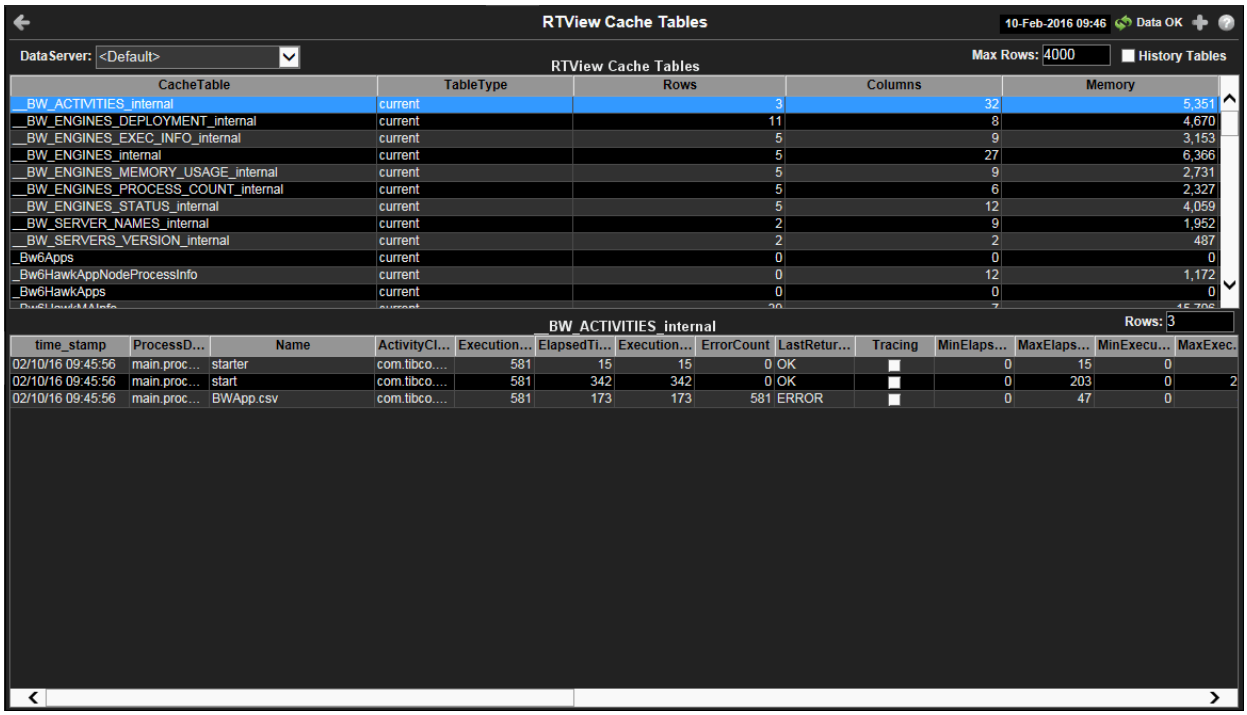


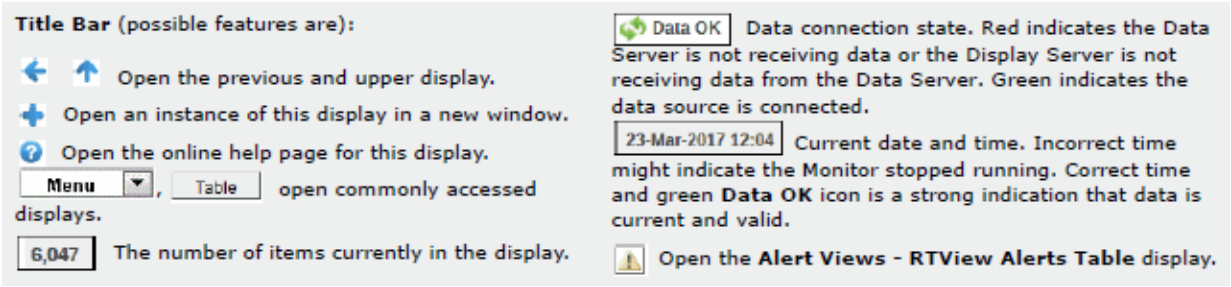
Alert	Warning Level	Alarm Level	Duration	Alert Enabled	Override Count
EmsQueuesProducerCountHigh	60	80	30	☐	0
EmsQueuesProducerCountLow	15	5	30	☐	0
EmsServerAsyncDBSizeHigh	50	100	30	☐	0
EmsServerConnectionCountHigh	60	80	30	☐	1
EmsServerInMsgRateHigh	60	80	30	☐	0
EmsServerMemUsedHigh	60	80	30	☐	0

RTView Cache Tables

View data that RTView is capturing and maintaining. Drill down and view details of RTView Cache Tables. Use this data for debugging. This display is typically used for troubleshooting with Technical Support.

Choose a cache table from the upper table to see cached data.





DataServer Select a data server from the drop down menu.

Max Rows Enter the maximum number of rows to display in RTView Cache Tables.

History Tables Select to include all defined history tables in RTView Cache Tables.

RTView Cache Tables

This table lists and describes all defined RTView Cache Tables for your system. Cache tables gather Monitor data and are the source that populate the Monitor displays.

NOTE: When you click on a row in RTView Cache Tables a supplemental table will appear that gives more detail on the selected Cache Table.

CacheTable The name of the cache table.

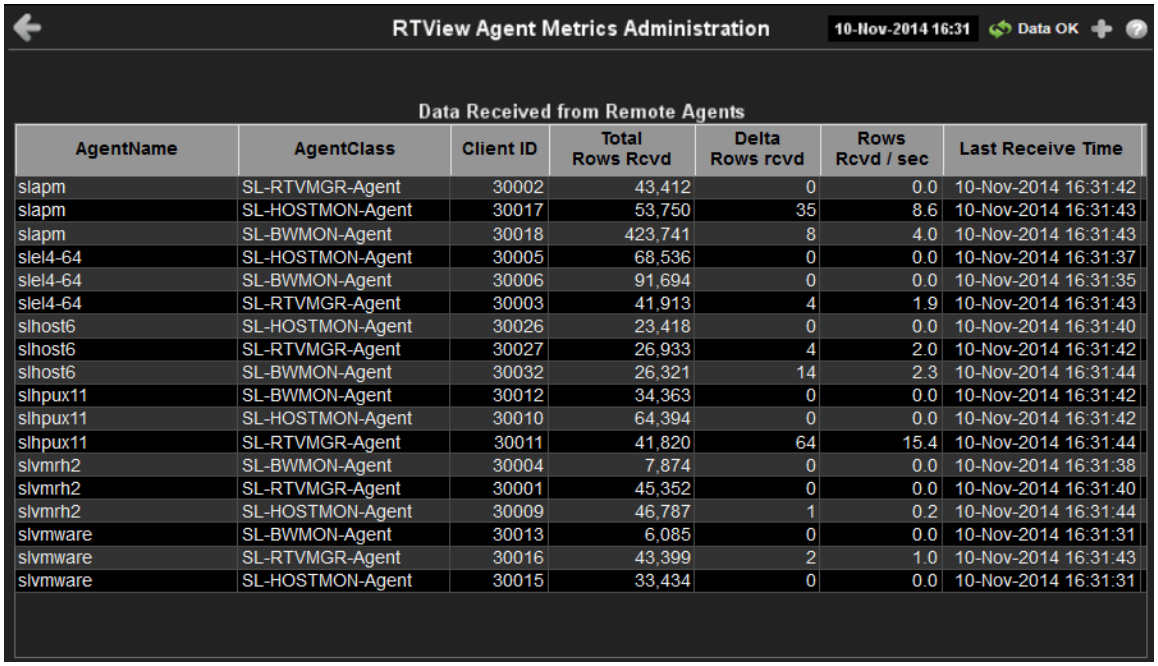
TableType The type of cache table:

current	Current table which shows the current values for each index.
current_condensed	Current table with primary compaction configured.
history	History table.
history_condensed	History table with primary compaction configured.

Rows	Number of rows currently in the table.
Columns	Number of columns currently in the table.
Memory	Amount of space, in bytes, used by the table.

RTView Agent Administration

Verify when agent metrics were last queried by the Monitor. The data in this display is predominantly used for debugging by Technical Support.



AgentName	AgentClass	Client ID	Total Rows Rcvd	Delta Rows rcvd	Rows Rcvd / sec	Last Receive Time
slapm	SL-RTVMGR-Agent	30002	43,412	0	0.0	10-Nov-2014 16:31:42
slapm	SL-HOSTMON-Agent	30017	53,750	35	8.6	10-Nov-2014 16:31:43
slapm	SL-BWMON-Agent	30018	423,741	8	4.0	10-Nov-2014 16:31:43
slel4-64	SL-HOSTMON-Agent	30005	68,536	0	0.0	10-Nov-2014 16:31:37
slel4-64	SL-BWMON-Agent	30006	91,694	0	0.0	10-Nov-2014 16:31:35
slel4-64	SL-RTVMGR-Agent	30003	41,913	4	1.9	10-Nov-2014 16:31:43
slhost6	SL-HOSTMON-Agent	30026	23,418	0	0.0	10-Nov-2014 16:31:40
slhost6	SL-RTVMGR-Agent	30027	26,933	4	2.0	10-Nov-2014 16:31:42
slhost6	SL-BWMON-Agent	30032	26,321	14	2.3	10-Nov-2014 16:31:44
slhpux11	SL-BWMON-Agent	30012	34,363	0	0.0	10-Nov-2014 16:31:42
slhpux11	SL-HOSTMON-Agent	30010	64,394	0	0.0	10-Nov-2014 16:31:42
slhpux11	SL-RTVMGR-Agent	30011	41,820	64	15.4	10-Nov-2014 16:31:44
slvmrh2	SL-BWMON-Agent	30004	7,874	0	0.0	10-Nov-2014 16:31:38
slvmrh2	SL-RTVMGR-Agent	30001	45,352	0	0.0	10-Nov-2014 16:31:40
slvmrh2	SL-HOSTMON-Agent	30009	46,787	1	0.2	10-Nov-2014 16:31:44
slvmware	SL-BWMON-Agent	30013	6,085	0	0.0	10-Nov-2014 16:31:31
slvmware	SL-RTVMGR-Agent	30016	43,399	2	1.0	10-Nov-2014 16:31:43
slvmware	SL-HOSTMON-Agent	30015	33,434	0	0.0	10-Nov-2014 16:31:31

Title Bar (possible features are):

Open the previous and upper display.

Open an instance of this display in a new window.

Open the online help page for this display.

Menu, Table open commonly accessed displays.

6,047 The number of items currently in the display.

Data OK Data connection state. Red indicates the Data Server is not receiving data or the Display Server is not receiving data from the Data Server. Green indicates the data source is connected.

23-Mar-2017 12:04 Current date and time. Incorrect time might indicate the Monitor stopped running. Correct time and green **Data OK** icon is a strong indication that data is current and valid.

Open the **Alert Views - RTView Alerts Table** display.

Data Received from Remote Agents Table	
AgentName	Name of the agent.
AgentClass	Class of the agent.
Client ID	Unique client identifier.
Total Rows Rcvd	Total number of rows of data received.
Rows Rcvd/sec	Number of rows of data received per second.

Last Receive Time	Last time data was received from the agent.

RTView Servers

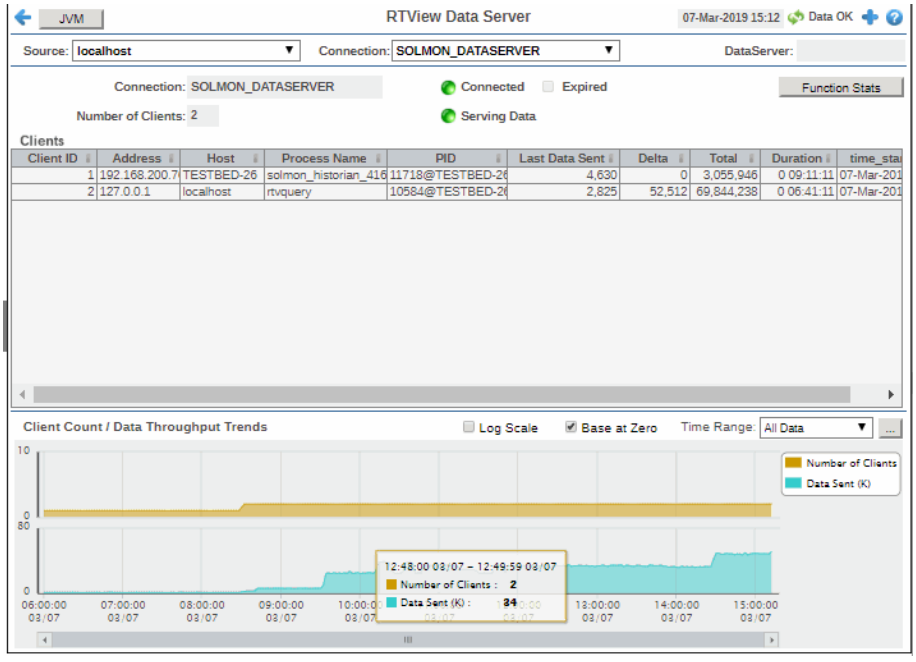
These displays present performance data for all RTView Servers. Displays in this View are:

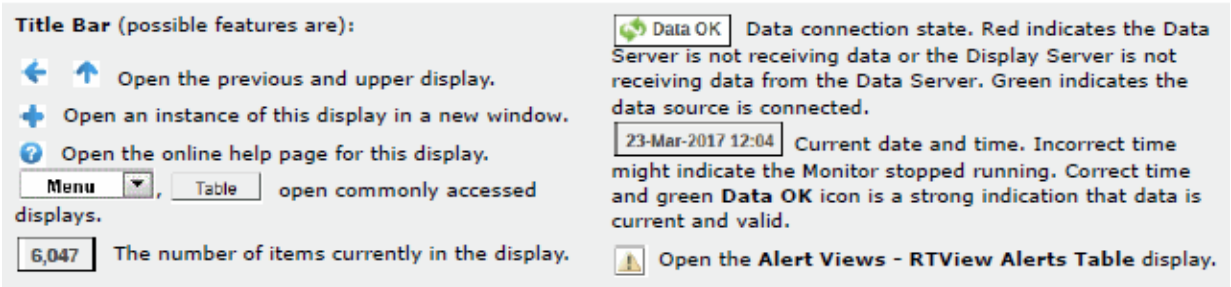
- "Data Server Metrics" on page 307: Shows metrics for RTView Data Servers.
- "Display Server Metrics" on page 309: Shows metrics for RTView Display Servers.
- "Historian Servers" on page 311: Shows metrics for RTView Historian Servers.
- "Tomcat Server Summary" on page 312: Shows metrics for Tomcat application sessions, including Tomcat hosting and connection details.
- "Tomcat Modules Summary" on page 315: Shows metrics for Tomcat application modules and utilization details.
- "JVM CPU/Mem Summary" on page 317: Shows Java Virtual Machine memory and CPU usage, JVM system information, application performance metrics, and input arguments for a single connection.
- "JVM Mem Pool Trends" on page 320: Shows Java Virtual Machine heap and non-heap memory usage for a single connection.
- "JVM Mem GC Trends" on page 323: Shows Java Virtual Machine garbage collection memory usage for a single connection.
- "JVM System Properties" on page 325: Shows Java Virtual Machine input arguments and system properties for a single connection.
- "Version Info" on page 326: Shows version information for all connected RTView applications.
- "About" on page 328: Shows the Monitor version and all available data sources.

Data Server Metrics

Track data transfer metrics for RTView Data Servers, client count and throughput trends.

Use the available drop-down menus or right-click to filter data shown in the display.





Source	Select the type of connection to the RTView Server.
Connection	Select an RTView Server from the drop-down menu. Names can be modified in the RTView Server configuration properties file.
Connection	The connection selected from the **Connection** drop-down menu.
Number of Clients	The number of clients currently server on this Data Server.
Connected	The Data Server connection state: Disconnected. Connected.
Serving Data	The Data Server is not currently serving data. The Data Server is currently serving data.
Expired	This server has been marked as expired after no activity.
Function Stats	Opens the **RTView Function Stats** display which shows detailed performance statistics for RTView functions in the selected Data Server. This button is only enabled if the RTVMGR has a JMX connection defined for the selected Data Server.

Clients

This table describes all clients on the selected server.

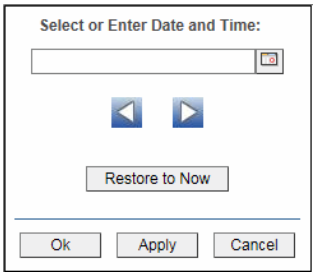
Address	The client IP address.
Client ID	The unique client identifier.
Duration	The amount of time for this client session. Format: **dd HH:MM:SS** **<days> <hours>:<minutes>:<seconds>** **For example:** **10d 08:41:38**
Host	The client host name.
Last Data Sent	The amount of data, in bytes, last sent to the client.
Delta	The amount of data, in bytes, sent since the last update.
Total	The total amount of data, in bytes, sent to the client.
TIME_STAMP	The date and time this row of data was last updated.

Client Count / Data Throughput Trends

Shows throughput metrics for all clients on the selected server.

Log Scale	Enable to use a logarithmic scale for the Y axis. Use Log Scale to see usage correlations for data with a wide range of values. For example, if a minority of your data is on a scale of tens, and a majority of your data is on a scale of thousands, the minority of your data is typically not visible in non-log scale graphs. Log Scale makes data on both scales visible by applying logarithmic values rather than actual values to the data.
Base at Zero	Use zero as the Y axis minimum for all graph traces.
Time Range	Select a time range from the drop down menu varying from **2 Minutes** to **Last 7 Days**, or display **All Data**. To specify a time

range, click Calendar .



By default, the time range end point is the current time. To change the time range end point, click Calendar and select a date and time from the calendar or enter the date and time in the text field using the following format: **MMM dd, YYYY HH:MM**. For example, **Aug 21, 2011 12:24 PM**.

Use the navigation arrows to move forward or backward one time period. NOTE: The time period is determined by your selection from the **Time Range** drop-down menu.

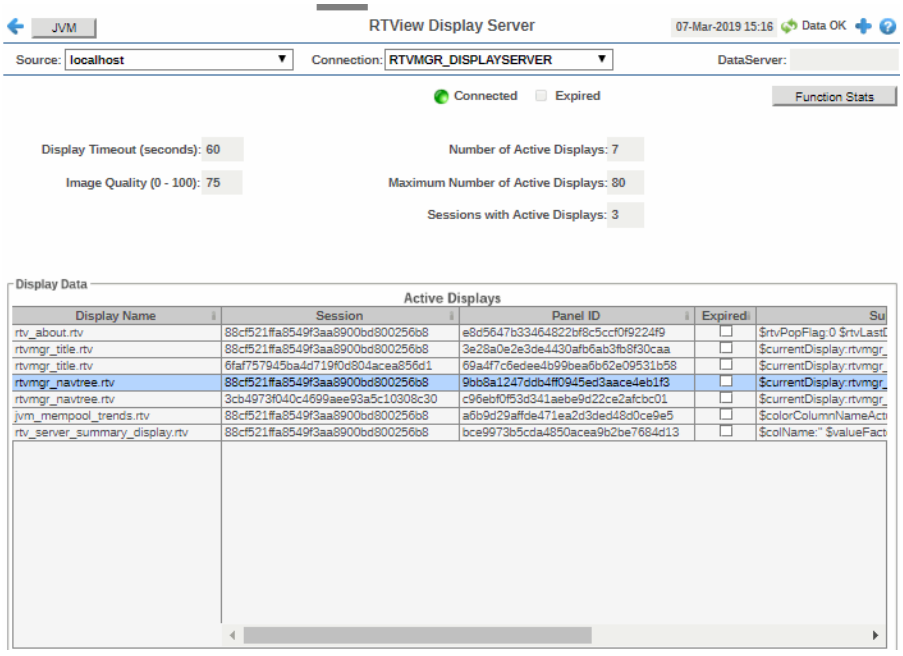
Click **Restore to Now** to reset the time range end point to the current time.

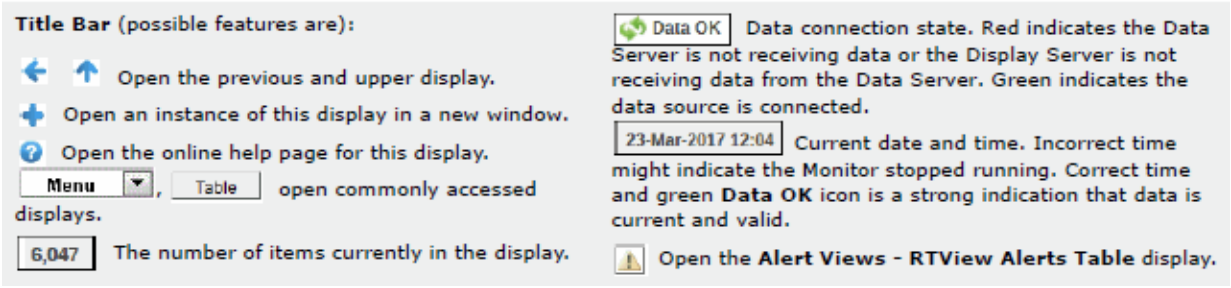
Number of Clients	Traces the number of clients being served by the Data Server.
Data Sent	Traces the total amount of data, in Kilobytes, sent to all clients.

Display Server Metrics

Track display utilization metrics for RTView Display Servers.

Use the available drop-down menus or right-click to filter data shown in the display.





Fields and Data
This display includes:

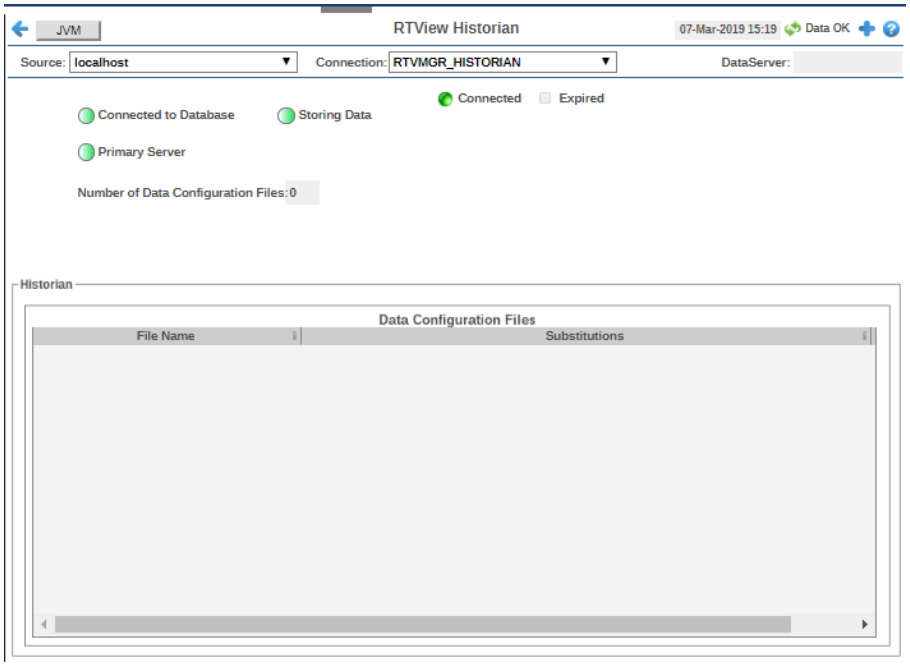
Source	Select the type of connection to the RTView Server.
Connection	Select an RTView Server from the drop-down menu. Names can be modified in the RTView Server configuration properties file.
Connected	The Display Server connection state: Disconnected. Connected.
Expired	This server has been marked as expired after no activity.
Function Stats	Opens the **RTView Function Stats** display which shows detailed performance statistics for RTView functions in the selected Display Server. This button is only enabled if the RTVMGR has a JMX connection defined for the selected Display Server.
Display Timeout (seconds)	The amount of time, in seconds, that a display can be kept in memory after the Display Servlet has stopped requesting it. The default is **60** seconds (to allow faster load time when switching between displays).
Image Quality (0-100)	A value between **0** and **100**, which controls the quality of the generated images. If the value is **100**, the Display Server outputs the highest quality image with the lowest compression. If the value is **0**, the Display Server outputs the lowest quality image using the highest compression. The default is **75**.
Number of Active Displays	The total number of displays currently being viewed by a user.
Maximum Number of Active Displays	The maximum number of displays kept in memory. The default is **20** (to optimize memory used by the Display Server).
Sessions with Active Displays	Number of clients accessing the Display Server.
Display Data / Active Displays	

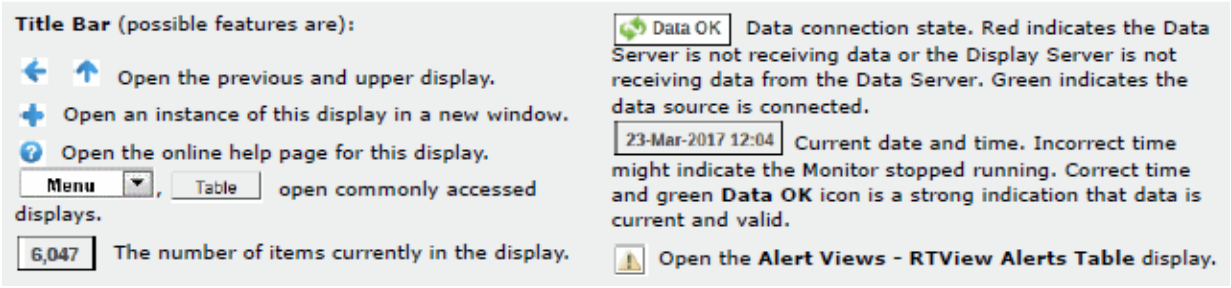
Display Name	The name of the currently open display.
Session	A unique string identifier assigned to each session.
Panel ID	A unique string identifier assigned to each panel. The Display Server loads each display requested by each client into a panel. This ID can be useful in troubleshooting.
Substitutions	Lists the substitutions used for the display.
Last Ref	The amount of time that has elapsed since the display was last requested by a client.
ID	The client ID.
Preloaded	When checked, indicates that the display (**.rtv**) file is configured in the **DISPLAYSERVER.ini** file to be preloaded. The **history_config** option is used to configure display preloading. Preloading a display makes data immediately available. Preloaded displays are not unloaded unless the Display Server is restarted or the display cache is cleared via JMX. This option can be used multiple times to specify multiple displays to preload.

Historian Servers

Track the status of RTView Historian Servers and data configuration file usage. View the caches that are archived by the Historian application, substitution variables associated with the history cache configuration file, as well as the history cache status. You can also stop and start the Historian, and purge data.

Use the available drop-down menus or right-click to filter data shown in the display.





Fields and Data
This display includes:

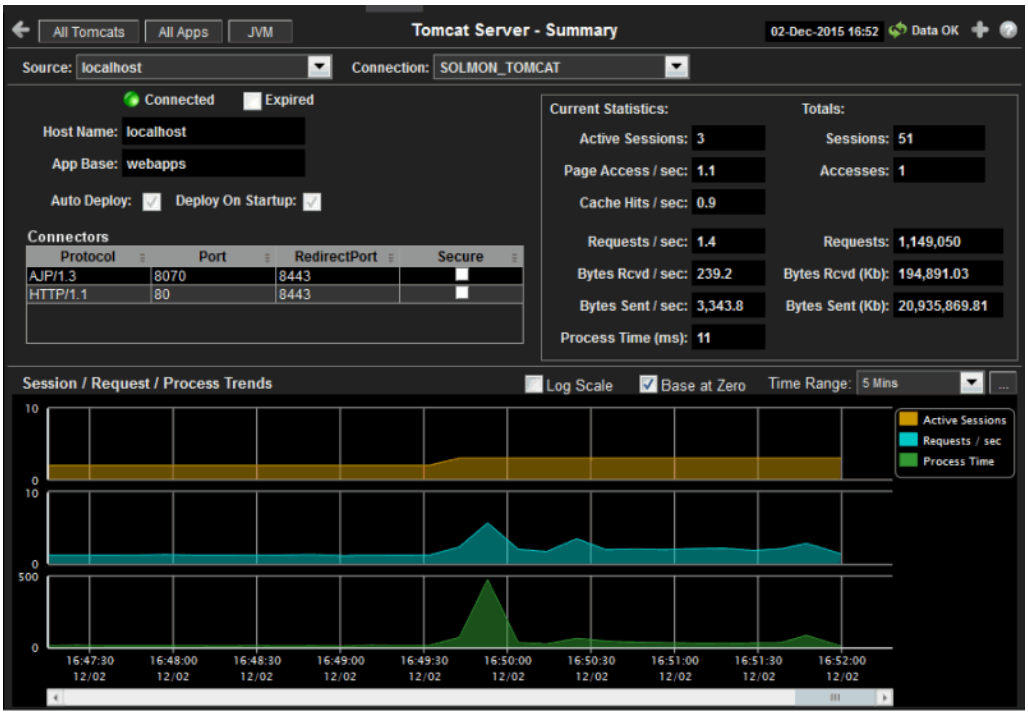
Source	Select the type of connection to the RTView Server.
Connection	Select an RTView Server from the drop-down menu. Names can be modified in the RTView Server configuration properties file.
Connected	The Historian Server connection state: ● Disconnected. ● Connected.
Expired	This server has been marked as expired after no activity.
Connected to Database	The Historian Server database connection state: ● Disconnected. ● Connected.
Primary Server	When green, indicates that this Historian, when used within a group of Historians, is the primary group member. If the primary member fails or shuts down, the standby member with the highest priority becomes the primary group member. When red, indicates that the Historian is a secondary server. The Historian Server member state: ● The Historian Server is a secondary group member. ● This Historian is the primary group member.
Number of Data Configuration Files	The number of configuration files that are used by the history cache.

Historian / Data Configuration Files

File Name	The name of the history cache configuration file.
Substitutions	Lists the substitutions specified in the history cache configuration file.

Tomcat Server Summary

Track the performance of one Tomcat Server and get Tomcat hosting and connection details. You can drill down to this display from the Servers table for detailed information and historical trends for a specific server. he trends include Active Sessions, Requests per Sec, and Process Time.



Title Bar (possible features are):

Open the previous and upper display.

Open an instance of this display in a new window.

Open the online help page for this display.

Menu, Table open commonly accessed displays.

6,047 The number of items currently in the display.

Data OK Data connection state. Red indicates the Data Server is not receiving data or the Display Server is not receiving data from the Data Server. Green indicates the data source is connected.

23-Mar-2017 12:04 Current date and time. Incorrect time might indicate the Monitor stopped running. Correct time and green **Data OK** icon is a strong indication that data is current and valid.

Open the **Alert Views - RTView Alerts Table** display.

Fields and Data

This display includes:

Source	Select the host where the Tomcat Server is running.
Connection	Select a Tomcat Server from the drop-down menu.
Connected	The Tomcat Server connection state: Disconnected. Connected.
Expired	When checked, this server is expired due to inactivity.
Host Name	The name of the host where the application resides.
App Base	The directory in which Tomcat modules are installed.
Auto Deploy	When checked, indicates that the Tomcat option, automatic application deployment, is enabled. NOTE: This Tomcat option is set using the **autoDeploy** property in the **server.xml** file, located in the Tomcat **conf** directory. **autoDeploy=true** enables the option.
Deploy On Startup	When checked, indicates that the option to deploy the application on Tomcat startup is enabled. NOTE: This Tomcat option is set using the **deployOnStartup** property in the **server.xml** file, located in the Tomcat **conf** directory. When enabled (**deployOnStartup=true**), applications from the host are automatically deployed.

Connectors

This table shows Tomcat application connection information.

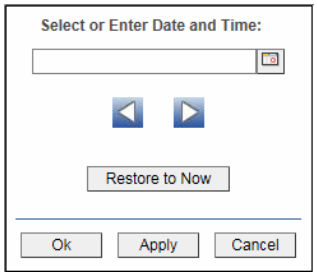
Protocol	The protocol used by the Tomcat application on the host.
Port	The port number used by the Tomcat application on the host.
RedirectPort	The redirect port number used by the Tomcat application on the host.
Secure	When checked, specifies that the Tomcat application uses a secure connection on the host.

Current Statistics / Totals

Active Sessions	The number of clients currently in session with the servlet.
Sessions	The total number of client sessions since the server was started.
Page Access / sec	The number of times pages are accessed, per second.
Accesses	The total number of page accesses since the server was started.
Cache Hits / sec	The number of times the cache is accessed, per second.
Requests / sec	The number of requests received, per second.
Requests	The total number of requests since the server was started.
Bytes Rcvd / sec	The number of bytes received, per second.
Bytes Rcvd (Kb)	The number of kilobytes received since the server was started.
Bytes Sent / sec	The number of bytes sent, per second.
Bytes Sent (Kb)	The total number of kilobytes sent since the server was started.
Process Time	The amount of time, in milliseconds, for the servlet to process client requests.

Session / Request / Process Trends

Shows metrics for the selected server.

Log Scale	Select to enable a logarithmic scale. Use Log Scale to see usage correlations for data with a wide range of values. For example, if a minority of your data is on a scale of tens, and a majority of your data is on a scale of thousands, the minority of your data is typically not visible in non-log scale graphs. Log Scale makes data on both scales visible by applying logarithmic values rather than actual values to the data.
Base at Zero	Use zero as the Y axis minimum for all graph traces.
Time Range	Select a time range from the drop down menu varying from **2 Minutes** to **Last 7 Days**, or display **All Data**. To specify a time range, click Calendar .  By default, the time range end point is the current time. To change the time range end point, click Calendar and select a date and time from the calendar or enter the date and time in the text field using the following format: **MMM dd, YYYY HH:MM**. For example, **Aug 21, 2011 12:24 PM**. Use the navigation arrows to move forward or backward

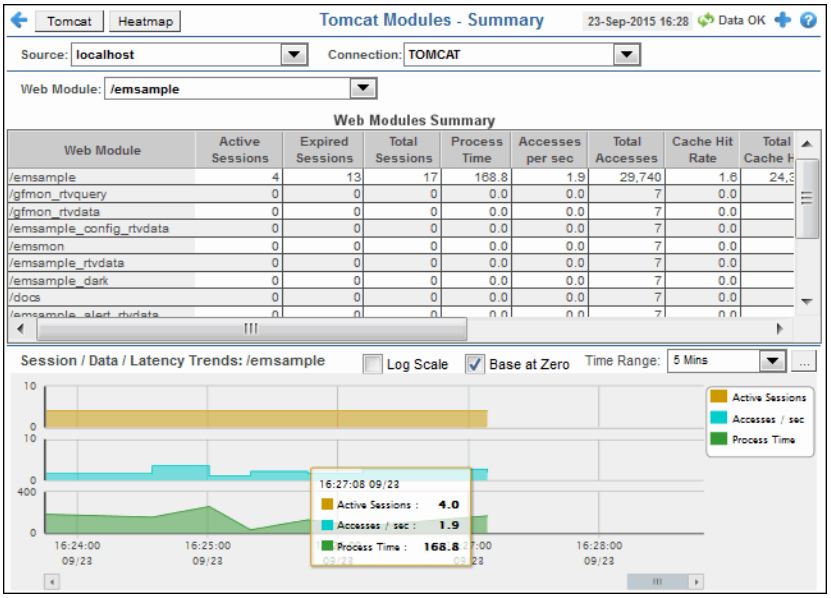
	one time period. NOTE: The time period is determined by your selection from the **Time Range** drop-down menu. Click **Restore to Now** to reset the time range end point to the current time.
Active Sessions	Traces the number of currently active client sessions.
Requests /sec	Traces the number of requests received, per second.
Process Time	Traces the average amount of time, in milliseconds, to process requests.

Tomcat Modules Summary

Track the performance of all web application modules in a server and view utilization details. The table summarizes the sessions, accesses, cache hit and so forth, for all installed web modules. Each row in the table is a different web application module. The row color for inactive modules is dark red. Select a web application module to view metrics in the trend graph.

Use this data to verify response times of your Web application modules.

Use the available drop-down menus or right-click to filter data shown in the display.



Title Bar (possible features are):

- Open the previous and upper display.
- Open an instance of this display in a new window.
- Open the online help page for this display.
- Menu, Table open commonly accessed displays.
- 6,047 The number of items currently in the display.
- Data OK Data connection state. Red indicates the Data Server is not receiving data or the Display Server is not receiving data from the Data Server. Green indicates the data source is connected.
- 23-Mar-2017 12:04 Current date and time. Incorrect time might indicate the Monitor stopped running. Correct time and green **Data OK** icon is a strong indication that data is current and valid.
- Open the **Alert Views - RTView Alerts Table** display.

Fields and Data
This display includes:

Source	Select the host where the Tomcat Server is running.
Connection	Select a Tomcat Server from the drop-down menu. This menu is populated by the selected Source.
Web Module	Select a Web module from the drop-down menu. This menu is populated by the selected Connection. The Web Module you select populates the trend graphs.

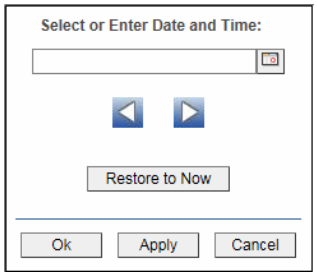
Web Module Summary

Web Module	The name of the Web module.
Sessions Active	The number of currently active client sessions.
Sessions Total	The total number of client sessions since the application was started.
Sessions Expired	The total number of client sessions that expired since the application was started.
Accesses per sec	The number of times pages are accessed, per second.
Accesses Total	The total number of times pages have been accessed since the application was started.
Bytes Rcvd per sec	The number of bytes received per second.
Bytes Rcvd Total	The total number of bytes received since the application was started.
Bytes Sent per sec	The number of bytes sent per second.
Bytes Sent Total	The total number of bytes sent since the application was started.
Cache Hit Rate	The number of times the cache is accessed, per second.
Requests per sec	The number of requests received, per second.
Requests Total	The total number of requests received since the application was started.
Process Time	The average amount of time, in milliseconds, to process requests.
Error Count	The number of errors occurred since the application was started.
appBase	The directory in which Tomcat is installed.
Expired	When checked, this connection is expired due to inactivity.
time_stamp	The date and time this row of data was last updated. Format: **MM/DD/YY HH:MM:SS** **<month>/ <day>/<year> <hours>:<minutes>:<seconds>**

Session/Data/Latency Trends

Shows metrics for the selected Web module. The Web module can be selected from the **Web Module** drop-down menu or the **Web Modules Summary** table.

Log Scale	Select to enable a logarithmic scale. Use Log Scale to see usage correlations for data with a wide range of values. For example, if a minority of your data is on a scale of tens, and a majority of your data is on a scale of thousands, the minority of your data is typically not visible in non-log scale graphs. Log Scale makes data on both scales visible by applying logarithmic values rather than actual values to the data.
Base at Zero	Use zero as the Y axis minimum for all graph traces.
Time Range	Select a time range from the drop down menu varying from **2 Minutes** to **Last 7 Days**, or display **All Data**. To specify a time range, click Calendar .



By default, the time range end point is the current time. To change the time range end point, click Calendar and select a date and time from the calendar or enter the date and time in the text field using the following format: **MMM dd, YYYY HH:MM**. For example, **Aug 21, 2011 12:24 PM**.

Use the navigation arrows to move forward or backward one time period. NOTE: The time period is determined by your selection from the **Time Range** drop-down menu.

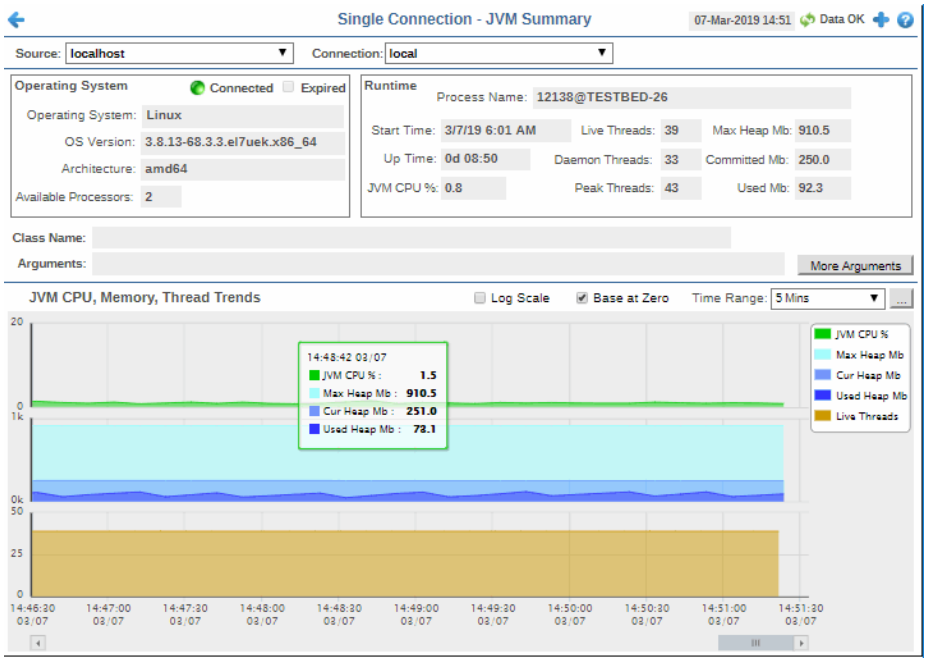
Click **Restore to Now** to reset the time range end point to the current time.

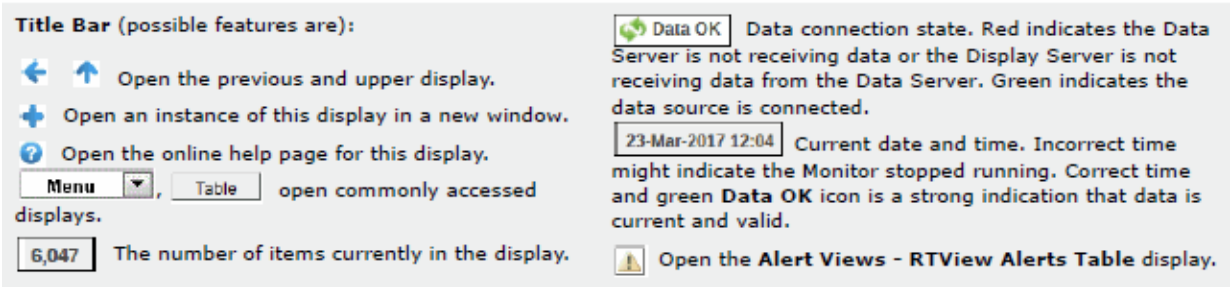
Active Sessions	Traces the number of currently active client sessions.
Accesses /sec	Traces the number of times pages are accessed, per second.
Process Time	Traces the average amount of time, in milliseconds, to process requests.

JVM CPU/Mem Summary

Track JVM memory and CPU usage, get JVM system information, application performance metrics, and input arguments for a single connection. Verify whether the memory usage has reached a plateau. Or, if usage is getting close to the limit, determine whether to allocate more memory.

Use the available drop-down menus or right-click to filter data shown in the display.





Fields and Data
This display includes:

Source Select the type of connection to the RTView Server.

Connection Select an RTView Server from the drop-down menu. Names can be modified in the RTView Server configuration properties file.

Operating System

Displays data pertaining to the operating system running on the host on which the JVM resides.

Connected	The data connection state: Disconnected. Connected.
Expired	When checked, this server is expired due to inactivity.
Operating System	The name of the operating system running on the host on which the JVM resides.
OS Version	The operating system version.
Architecture	The ISA used by the processor.
Available Processors	The total number of processors available to the JVM.

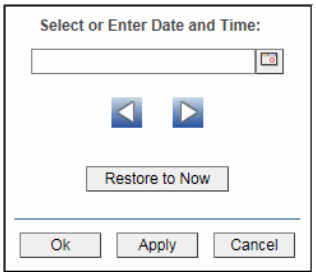
Runtime

Process Name	Name of the process.
Start Time	The date and time that the application started running.
Up Time	The amount of time the application has been running, in the following format: **0d 00:00** **<days>d <hours>:<minutes>:<seconds>** For example: **10d 08:41:38**
JVM CPU %	The amount of CPU usage by the JVM, in percent.
Live Threads	The total number of live threads.
Daemon Threads	The total number of live daemon threads.
Peak Threads	The total number of peak live threads since the JVM started or the peak was reset.
Max Heap Mb	The maximum amount of memory used for memory management by the application in the time range specified. This value may change or be undefined. NOTE: A memory allocation can fail if the JVM attempts to set the **Used** memory allocation to a value greater than the **Committed** memory allocation, even if the amount for **Used** memory is less than or equal to the *Maximum* memory allocation (for example, when the system is low on virtual memory).
Committed Mb	The amount of memory, in megabytes, guaranteed to be available for use by the JVM. The amount of committed memory can be a fixed or variable size. If set to be a variable size, the amount of committed memory can change over time, as the JVM may release memory to the system. This means that the amount allocated for **Committed** memory could be less than the amount initially allocated. **Committed** memory will always be greater than or equal to the amount allocated for **Used** memory.
Used Mb	The amount of memory currently used by the application. Memory used includes the memory occupied by all objects including both reachable and unreachable objects.
Class Name	Class name used for JVM.
Arguments	The arguments used to start the application.
More Arguments	Additional arguments used to start the application.

JVM CPU, Memory, Thread Trends

Shows JVM metrics for the selected server.

Log Scale	Enable to use a logarithmic scale for the Y axis. Use Log Scale to see usage correlations for data with a wide range of values. For example, if a minority of your data is on a scale of tens, and a majority of your data is on a scale of thousands, the minority of your data is typically not visible in non-log scale graphs. Log Scale makes data on both scales visible by applying logarithmic values rather than actual values to the data.
Base at Zero	Use zero as the Y axis minimum for all graph traces.
Time Range	Select a time range from the drop down menu varying from **2 Minutes** to **Last 7 Days**, or display **All Data**. To specify a time range, click Calendar .



By default, the time range end point is the current time. To change the time range end point, click Calendar and select a date and time from the calendar or enter the date and time in the text field using the following format: **MMM dd, YYYY HH:MM**. For example, **Aug 21, 2011 12:24 PM**.

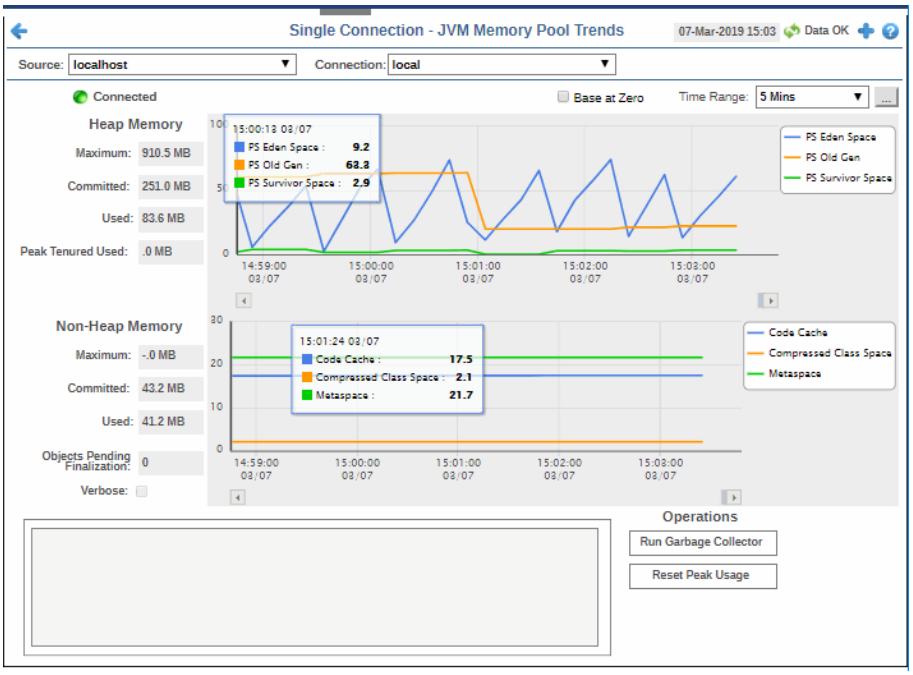
Use the navigation arrows to move forward or backward one time period. NOTE: The time period is determined by your selection from the **Time Range** drop-down menu.

Click **Restore to Now** to reset the time range end point to the current time.

JVM CPU %	Traces the amount of memory, in percent, used by the JVM in the time range specified.
Max Heap Mb	Traces the maximum amount of memory used for memory management by the application in the time range specified. This value may change or be undefined. NOTE: A memory allocation can fail if the JVM attempts to set the **Used** memory allocation to a value greater than the **Committed** memory allocation, even if the amount for **Used** memory is less than or equal to the **Maximum** memory allocation (for example, when the system is low on virtual memory).
Cur Heap Mb	Traces the current amount of memory, in megabytes, used for memory management by the application in the time range specified.
Used Heap Mb	Traces the memory currently used by the application.
Live Threads	Traces the total number of currently active threads in the time range specified.

JVM Mem Pool Trends

Track JVM heap and non-heap memory usage for a single connection. Use the available drop-down menus or right-click to filter data shown in the display.



Title Bar (possible features are):

Open the previous and upper display.

Open an instance of this display in a new window.

Open the online help page for this display.

Menu, Table open commonly accessed displays.

6,047 The number of items currently in the display.

Data OK Data connection state. Red indicates the Data Server is not receiving data or the Display Server is not receiving data from the Data Server. Green indicates the data source is connected.

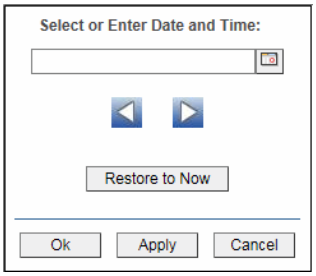
23-Mar-2017 12:04 Current date and time. Incorrect time might indicate the Monitor stopped running. Correct time and green **Data OK** icon is a strong indication that data is current and valid.

Open the **Alert Views - RTView Alerts Table** display.

Fields and Data

This display includes:

Source	Select the type of connection to the RTView Server.
Connection	Select an RTView Server from the drop-down menu. Names can be modified in the RTView Server configuration properties file.
	The data connection state:
Connected	Disconnected.
	Connected.
Base at Zero	Use zero as the Y axis minimum for all graph traces.
Time Range	Select a time range from the drop down menu varying from **2 Minutes** to **Last 7 Days**, or display **All Data**. To specify a time range, click Calendar .



By default, the time range end point is the current time. To change the time range end point, click Calendar and select a date and time from the calendar or enter the date and time in the text field using the following format: **MMM dd, YYYY HH:MM**. For example, **Aug 21, 2011 12:24 PM**.

Use the navigation arrows to move forward or backward one time period. NOTE: The time period is determined by your selection from the **Time Range** drop-down menu.

Click **Restore to Now** to reset the time range end point to the current time.

Heap Memory

Maximum	The maximum amount of memory used, in megabytes, for memory management by the application in the time range specified. This value may change or be undefined. NOTE: A memory allocation can fail if the JVM attempts to set the **Used** memory allocation to a value greater than the **Committed** memory allocation, even if the amount for **Used** memory is less than or equal to the **Maximum** memory allocation (for example, when the system is low on virtual memory).
Committed	The amount of memory, in megabytes, guaranteed to be available for use by the JVM. The amount of committed memory can be a fixed or variable size. If set to be a variable size, the amount of committed memory can change over time, as the JVM may release memory to the system. This means that the amount allocated for **Committed** memory could be less than the amount initially allocated. **Committed** memory will always be greater than or equal to the amount allocated for **Used** memory.
Used	The amount of memory, in megabytes, currently used by the application. Memory used includes the memory occupied by all objects including both reachable and unreachable objects.
Peak Tenured Used	The amount of memory, in megabytes, used by tenured JVM objects in the time range specified. Tenured refers to JVM objects contained in a pool that holds objects that have avoided garbage collection and reside in the survivor space. Peak tenured refers to the maximum value of the tenured memory over a specified period of time.
Eden Space	Traces the amount of memory used by the JVM eden pool in the time range specified. Eden refers to the JVM eden pool, which is used to initially allocate memory for most objects.
Survivor Space	Traces the amount of memory used by the JVM survivor pool in the time range specified. The JVM survivor pool holds objects that survive the eden space garbage collection.
Tenured Gen	Traces the amount of memory used by tenured JVM objects in the time range specified. Tenured refers to JVM objects contained in a pool that holds objects that have avoided garbage collection and reside in the survivor space. Peak tenured refers to the maximum value of the tenured memory over a specified period of time.

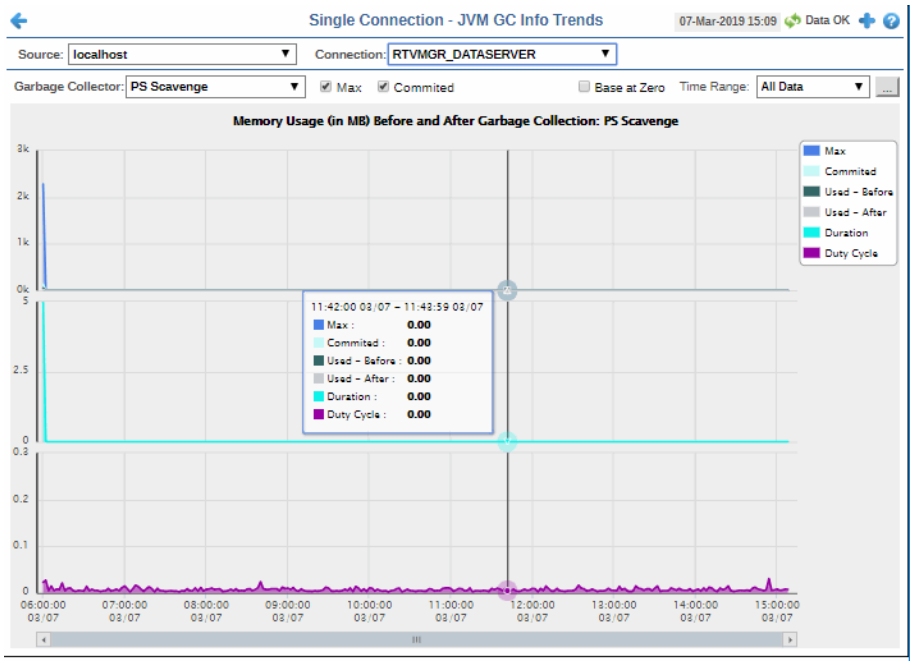
Non-Heap Memory

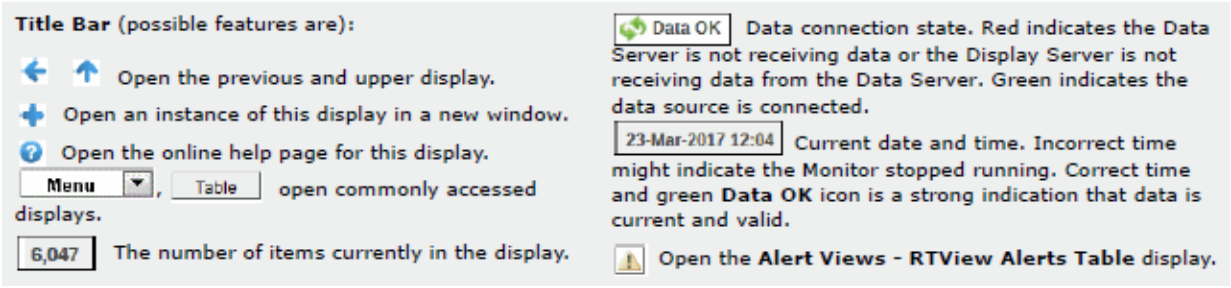
Maximum	The maximum amount of memory, in megabytes, used for JVM non-heap memory management by the application in the time range specified.
Committed	The amount of memory, in megabytes, guaranteed to be available

		for use by JVM non-heap memory management. The amount of committed memory can be a fixed or variable size. If set to be a variable size, it can change over time, as the JVM may release memory to the system. This means that the amount allocated for **Committed** memory could **be** less than the amount initially allocated. Committed memory will always be greater than or equal to the amount allocated for **Used** memory.
	Used	The amount of memory, in megabytes, currently used by the application. Memory used includes the memory occupied by all objects including both reachable and unreachable objects.
	Objects Pending Finalization	The value of the **MemoryMXBean ObjectPendingFinalizationCount** attribute.
	Verbose	The value of the **MemoryMXBean Verbose** attribute.
	Code Cache	Traces the amount of non-heap memory used in the JVM for compilation and storage of native code.
	Perm Gen	Traces the amount of memory used by the pool containing reflective data of the virtual machine, such as class and method objects. With JVMs that use class data sharing, this generation is divided into read-only and read-write areas.
Operations		
	Run Garbage Collector	Performs garbage collection on the selected server.
	Reset Peak Usage	Clears peak usage on the selected server.

JVM Mem GC Trends

Track JVM garbage collection memory usage for a single connection. Use the available drop-down menus or right-click to filter data shown in the display.





Fields and Data
This display includes:

Source Select the type of connection to the RTView Server.

Connection Select an RTView Server from the drop-down menu. Names can be modified in the RTView Server configuration properties file.

Garbage Collector Select a garbage collection method: **Copy** or **MarkSweepCompact**.

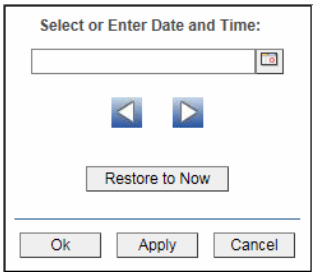
Max Shows the maximum amount of memory used for JVM garbage collection in the time range specified.

Committed Shows the amount of memory guaranteed to be available for use by JVM non-heap memory management. The amount of committed memory can be a fixed or variable size. If set to be a variable size, it can change over time, as the JVM may release memory to the system. This means that the amount allocated for **Committed** memory could be less than the amount initially allocated. **Committed** memory will always be greater than or equal to the amount allocated for **Used** memory.

Base at Zero Use zero as the Y axis minimum for all graph traces.

Time Range

Select a time range from the drop down menu varying from **2 Minutes** to **Last 7 Days**, or display **All Data**. To specify a time range, click Calendar .



By default, the time range end point is the current time. To change the time range end point, click Calendar and select a date and time from the calendar or enter the date and time in the text field using the following format: **MMM dd, YYYY HH:MM**. For example, **Aug 21, 2011 12:24 PM**.

Use the navigation arrows to move forward or backward one time period. NOTE: The time period is determined by your selection from the **Time Range** drop-down menu.

Click **Restore to Now** to reset the time range end point to the current time.

Memory Usage (in MB) Before and After Garbage Collection

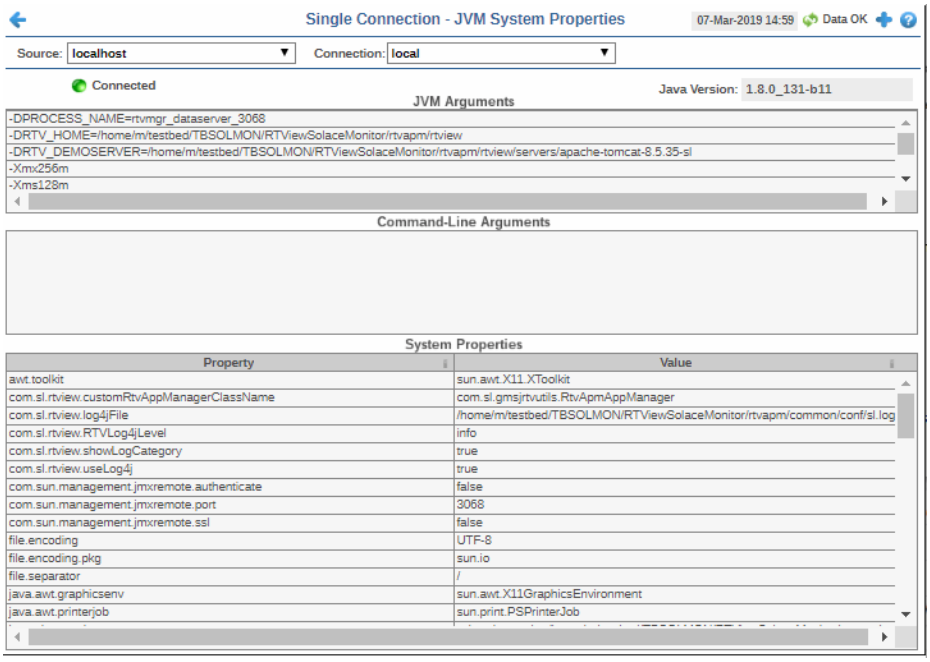
Maximum Traces the maximum amount of memory used by garbage collection in the time range specified. This value may change or be undefined.

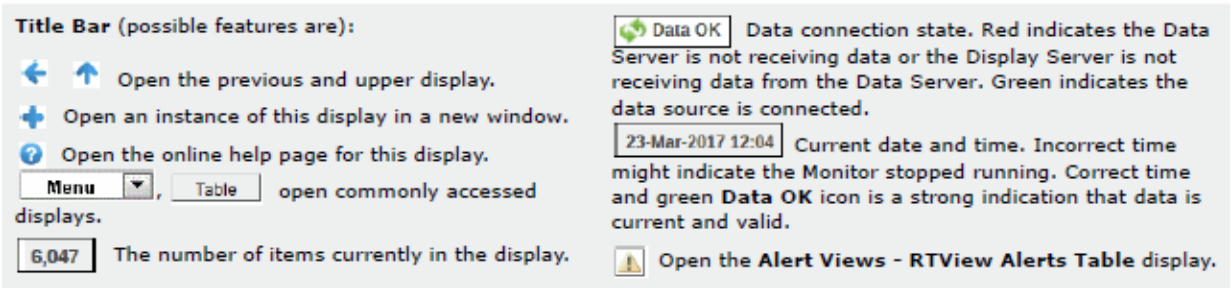
NOTE: A memory allocation can fail if the JVM attempts to set the **Used** memory allocation to a value greater than the **Committed** memory allocation, even if the amount for **Used** memory is less than or equal to the **Maximum** memory allocation (for example,

	when the system is low on virtual memory).
Committed	Traces the amount of memory guaranteed to be available for use by the JVM. The amount of committed memory can be a fixed or variable size. If set to be a variable size, the amount of committed memory can change over time, as the JVM may release memory to the system. This means that the amount allocated for **Committed** memory could be less than the amount initially allocated. **Committed** memory will always be greater than or equal to the amount allocated for **Used** memory.
Used - Before	Traces the amount of memory used before the last garbage collection.
Used - After	Traces the amount of memory used after the last garbage collection.
Duration	The duration, in seconds, of garbage collection.
Duty Cycle	The percentage of time that the application spends in garbage collection.

JVM System Properties

Track JVM input arguments and system properties for a single connection. Use the available drop-down menus or right-click to filter data shown in the display.





Fields and Data
This display includes:

Source	Select the type of connection to the RTView Server.
Connection	Select an RTView Server from the drop-down menu. Names can be modified in the RTView Server configuration properties file.
	The data connection state:
Connected	Disconnected.
	Connected.
Java Version	The Java version running on the selected server.
JVM Arguments	The JVM arguments in the **RuntimeMXBean InputArguments** attribute.
Command Line Arguments	Arguments used to start the application.

System Properties
This table lists and describes system property settings.

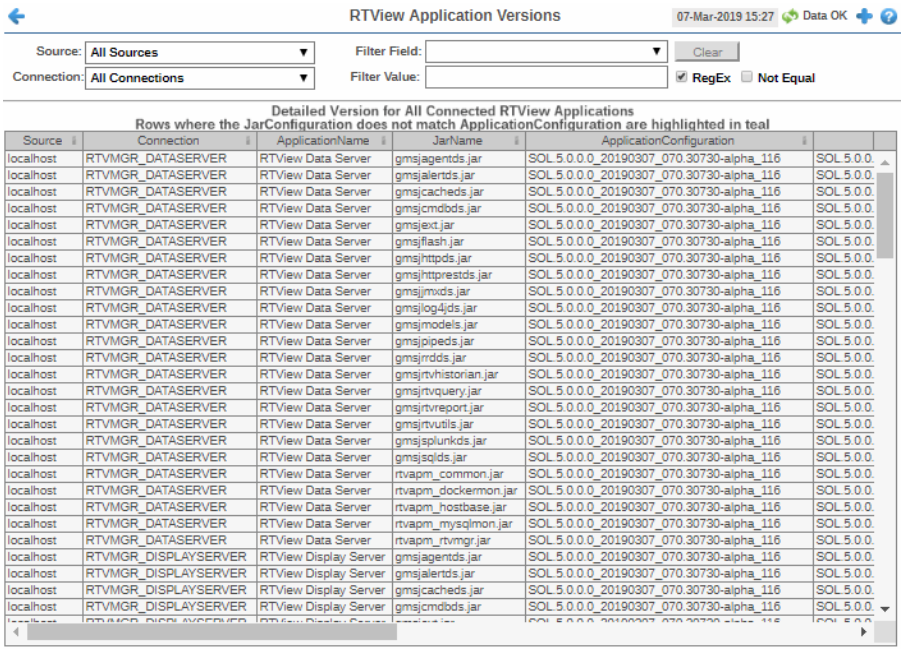
Property	Name of the property.
Value	Current value of the property.

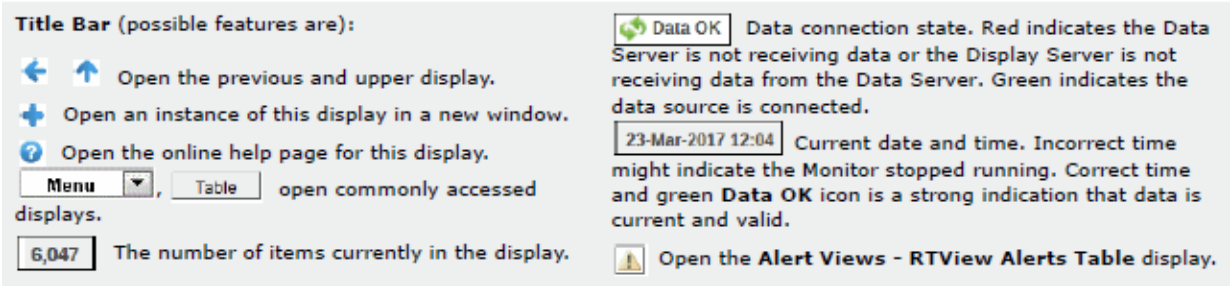
Version Info

This display provides detailed version information for all of the connected RTView applications. You can view specific applications by filtering data using the **Source**, **Connection**, **Filter Field**, and **Filter Value** fields at the top of the display. This display provides valuable information about the version of each jar that is used in each connected RTView application that can be used to help Technical Support when issues arise. Rows in the table where the **JarConfiguration** does not match the **ApplicationConfiguration** are highlighted in teal.

> **Note:** RTView applications running versions previous to this enhancement will only have one row in the table and will display "version info not supported in this version" in the **ApplicationConfiguration** column.

Use the available drop-down menus or right-click to filter data shown in the display.



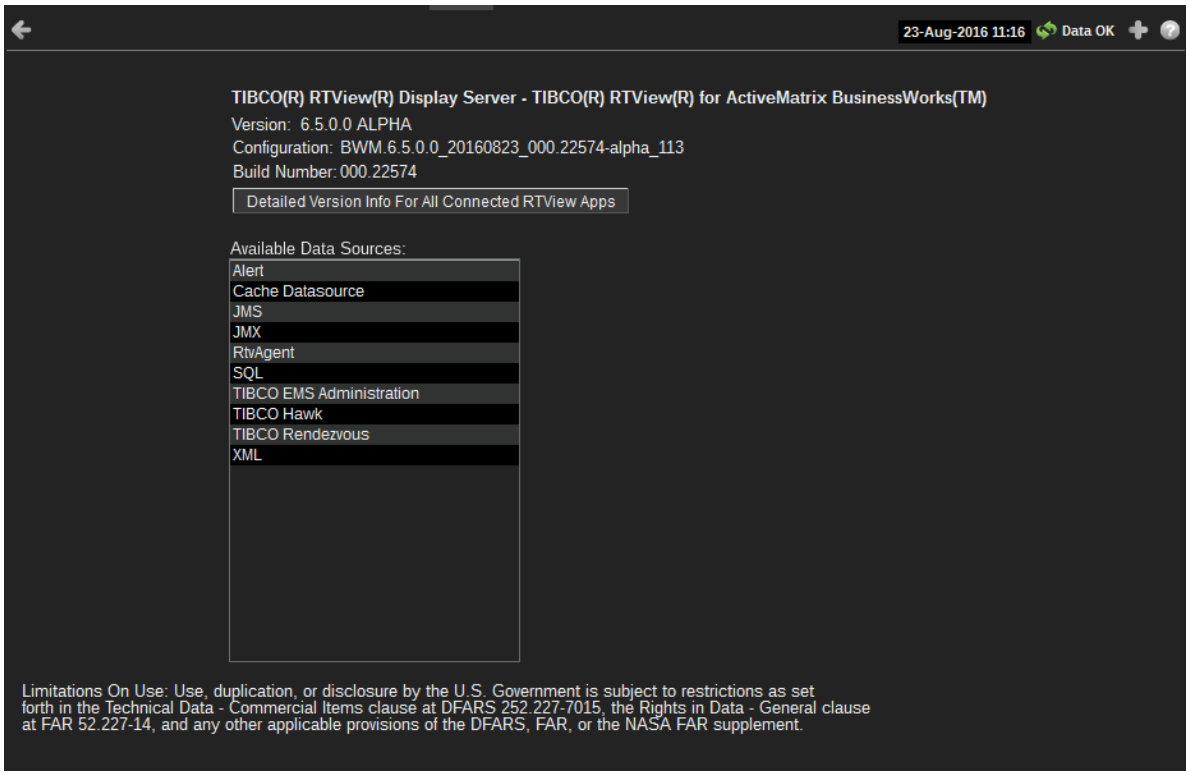


Fields and Data
This display includes:

Source	Select a filter value for the **Source** column.
Connection	Select a filter value for the **Connection** column.
Filter Field	Select a table column from the drop-down menu to perform a search in: **ApplicationName, JarName, ApplicationConfiguration, JarConfiguration, JarVersionNumber, JarVersionDate, JarReleaseDate,** and **JarMicroVersion**. Filters limit display content and drop-down menu selections to only those items that pass through the selected filter's criteria. If no items match the filter, you might have zero search results (an empty table). Double-clicking on a specific field in the table will populate this field with the selected field's content. For example, double-clicking on the **DataServerName** field in one of the rows displays the entire field's content into this field.
Clear	Clears entries in the **Filter Field** display list, **Filter Value** field, and **Not Equal** check box.
Filter Value	Enter the (case-sensitive) string to search for in the selected **Filter Field**.
RegEx	Select this check box to use the **Filter Value** as a regular expression when filtering. When selected, the **Not Equal** check box displays.
Not Equal	Works in conjunction with the **RegEx** field. Selecting this check box searches for values in the specified **Filter Field** that are NOT equal to the value defined in the **Filter Value** field. For example, if the **Filter Field** specified is **JarMicroVersion**, the **Filter Value** is specified as **317,** and this check box is selected, then only those rows containing **JarMicroVersion** fields NOT EQUAL to **317** will display. This field is only enabled when the **RegEx** check box is checked.
Source	The name of the source of the RTVMGR.
Connection	Lists the name of the JMX connection to the RTView application.
Application Name	Lists the name of the application.
JarName	Lists the name of the jar used in the connected application.
Application Configuration	Lists the configuration string of the application. This string contains the main application version that corresponds to the version information printed to the console at startup.
JarConfiguration	Lists the configuration string for the jar.
JarVersionNumber	Lists the version number for the jar.
JarVersionDate	Lists the version date for the jar.
JarReleaseType	Lists the release type for the jar.
JarMicroVersion	Lists the micro version for the jar.
Expired	When checked, this connection is expired due to inactivity.
time_stamp	The time at which the information in the current row was last received.
DataServerName	The name of the RTVMGR data server connection.

About

This display shows details about the Monitor version and data sources available to your system.



CHAPTER 7 Third-Party Reports

This section describes third-party reports for BusinessWorks 6, including system requirements, configuration steps, and the available parameters within each report.

This section includes:

TIBCO Spotfire Reports

Currently, there is one TIBCO Spotfire report that is provided with the Monitor, the **BusinessWorks Engines Metrics Report**. This section includes:

System Requirements

This section describes the minimum system requirements necessary to use these reports.

- **TIBCO BusinessWorks 6**
- **TIBCO Spotfire**

Version 7.0 for Oracle and MySQL reports

- **Clients**

Microsoft Windows 64-bit

- **Databases Supported**

Oracle (version 11G) and MySQL (version 5.6)

Configuring Spotfire Reports

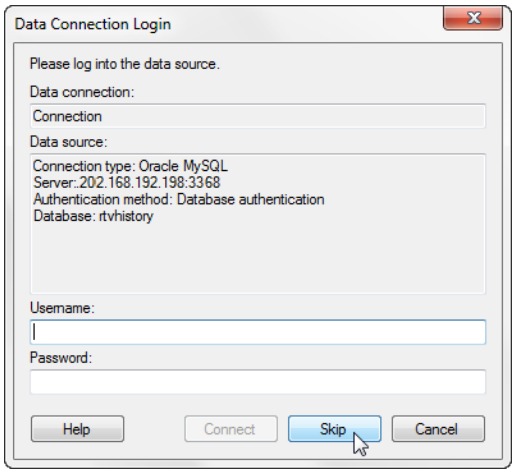
You can configure the **BusinessWorks Engines Metrics Report** using Oracle or MySQL:

MySQL Report Configuration

This section describes how to configure the **BusinessWorks Engines Metrics Report** using MySQL.

1. Open the **bw_engines_mysql.dxp** Spotfire Analysis file in the **TIB_rtview-bw/projects/reports/Spotfire** directory that was created during the Monitor installation.

   The **Data Connection Login** window opens.



2. Click **Skip** (there is no need to log in at this point). Because there is no data, Spotfire might display a "Missing Data" dialog. Click **OK** to dismiss it.

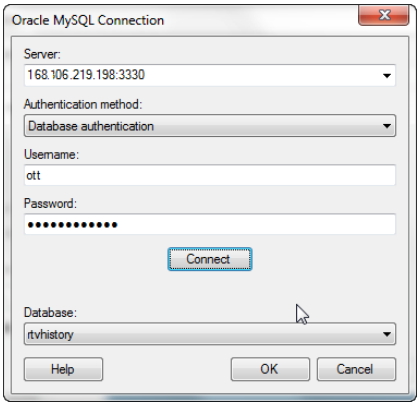
The **TIBCO Spotfire** dashboard opens.

3. Click **File/Replace Data Table**.

The **Replace Data Table** window opens.

4. Click **Select** (associated with the **Select source to replace with** field) and choose **Oracle MySQL**.

The **Oracle MySQL Connection** window opens.

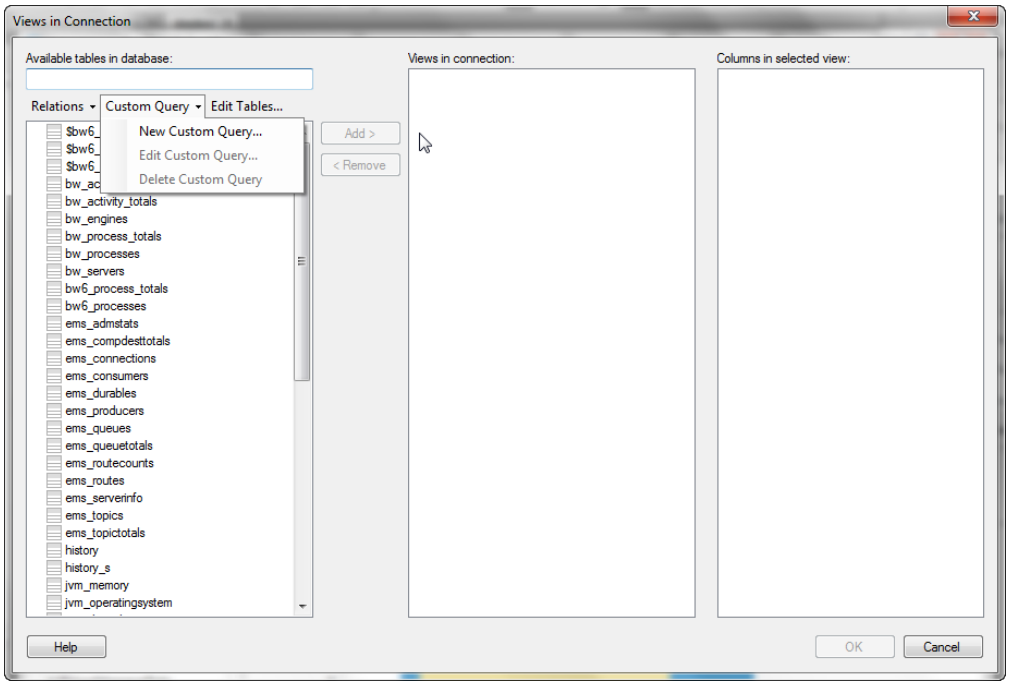


5. Enter the **Server**, **Username** and **Password**, select **Database authentication** as the **Authentication Method**, and click **Connect**.

The **Database** drop down menu should populate.

6. Select **rtvhistory** from the **Database** drop down menu and click **OK**.

The **Views in Connection** window opens.



7. Select the **Custom Query** drop down list and choose **New Custom Query**.

The **Custom Query** window opens.

8. Enter the desired name (a name that is meaningful for you) in the **Query_name** field.

9. Open **bw_engines_mysql.txt** file and copy/paste the SQL code from the file into the **Custom Query** window **Query** field. Click **Verify**.

Note: This step is required because the database contains data that has been compacted as well as data that has not yet been compacted. The SQL code compacts the data that has not been compacted and adds the newly compacted data to the already compacted data so that all the "bucket" values are the same. For example, let's say the compacted data is compacted so that the oldest data is contained in 15 minute buckets, but the more recent data is contained in 5 or 10 minute buckets. The SQL code takes the data contained in the 5 and 10 minute buckets and compacts it into 15 minute buckets so that all the data is consistently bucketed.

After the SQL is verified, the column names display in the **Result Columns** tab.

10. Click **OK** on the **Custom Query** window.

The new query (for example, **BW_engines_mysql)** should display in the list of **Custom queries** on the **Views in Connection** window.

11. Select your new custom query and click **Add**.

Your new custom query should display in the **Views in connection** region and the query's associated columns should display in the **Columns in selected view** region.

12. Click **OK** on the **Views in Connection** window.

    The **Replace Data Table** window displays.

13. Select the **Import data table** radio button and click **OK**.

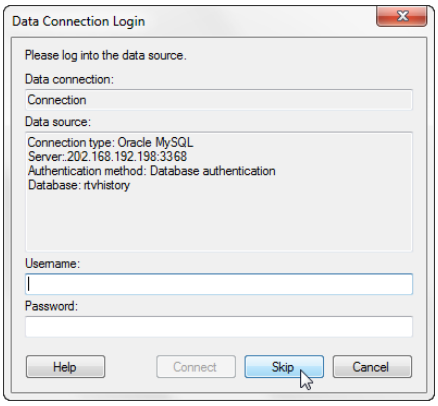
    Your data should display in TIBCO Spotfire.

Oracle Report Configuration

This section describes how to configure the **BusinessWorks Engines Metrics Report** using Oracle.

1. Open the **bw_engines_sql.dxp** Spotfire Analysis file in the **TIB_rtview-bw/projects/reports/Spotfire** directory that was created during the Monitor installation.

   The **Data Connection Login** window displays.



2. Click **Skip** (there is no need to log in at this point). Because there is no data, Spotfire might display a "Missing Data" dialog. Click **OK** to dismiss it.

   The **TIBCO Spotfire** dashboard opens.

3. Click **File> Replace Data Table**.

   The **Replace Data Table** window opens.

4. Click **Select** (associated with the **Select source to replace with** field) and select **Oracle**.

   The **Oracle Connection** window opens.

5. Enter the **Server**, select **SID** in the **Connect using** drop down (and enter **orcl** in the associated field if not defaulted), select **Oracle authentication** as the **Authentication Method**, enter the **Username** and **Password**, and click **Connect**.

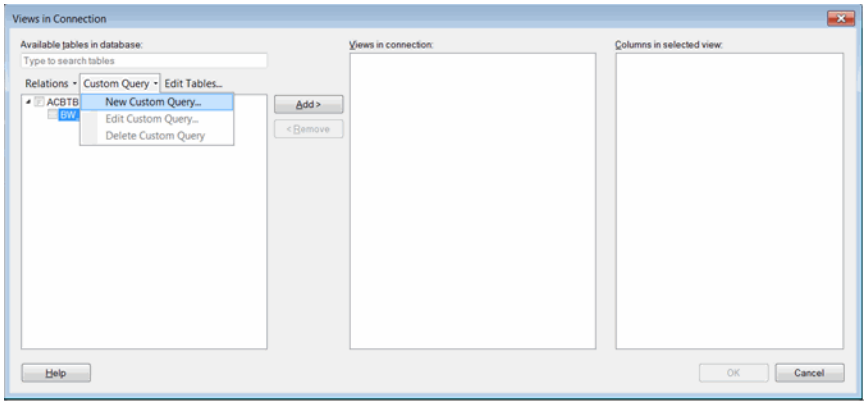
   The **Select Database Tables** window opens.

6. Select **BW_ENGINES** from the **Available Tables** list and click **Add**.

   The table displays in the **Selected tables** region.

7. Click **OK**.

   The **Views in Connection** window displays with the selected table listed in the **Available tables in the database** region.



8. Select the **BW_ENGINES** table from the list and click **Custom Query / New Query**.

   The **Custom Query** window opens.

9. Enter the desired name (a name that is meaningful for you) into the **Query_name** field.

10. , Open the **bw_engines_sql.txt** file and copy/paste the SQL code from the file into the **Custom Query** window **Query** field.

11. Click **Verify**.

Note: Depending on your environment, you might need to add your server name prefix in front of the **bw_engines** table name, as shown in the screen shot, above.

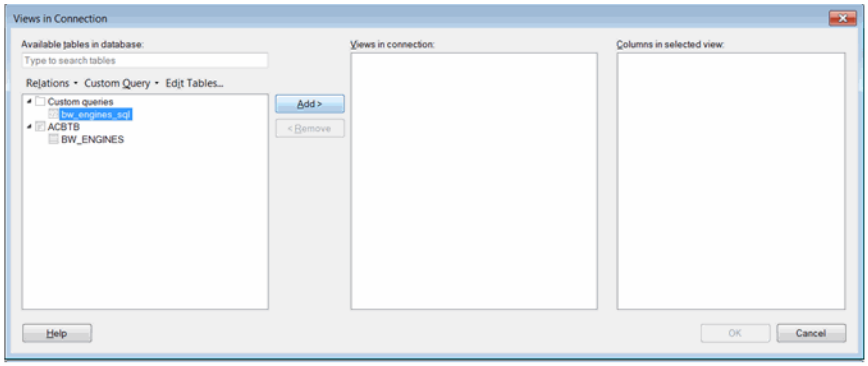
Note: This step is required because the database contains data that has been compacted as well as data that has not yet been compacted. The SQL code compacts the data that has not been compacted and adds the newly compacted data to the already compacted data so that all

the "bucket" values are the same. For example, let's say the compacted data is compacted so that the oldest data is contained in 15 minute buckets, but the more recent data is contained in 5 or 10 minute buckets. The SQL code takes the data contained in the 5 and 10 minute buckets and compacts it into 15 minute buckets so that all the data is consistently bucketed.

After the SQL script is verified, the column names display in the **Result Columns** tab.

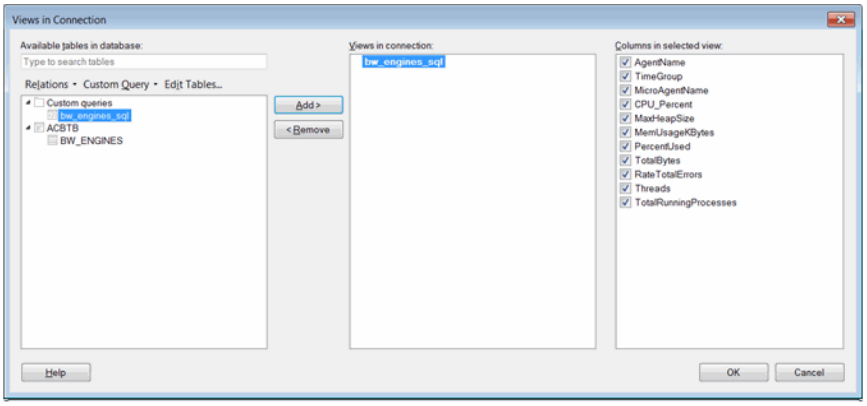
12. Click **OK**.

The new query displays under **Custom queries** in the **Available tables in database** list on the **Views in Connection** window.



13. Select your newly added query/view and click **Add**.

The new query displays in the **Views in connection** list and the associated columns display in the **Columns in selected view** region.

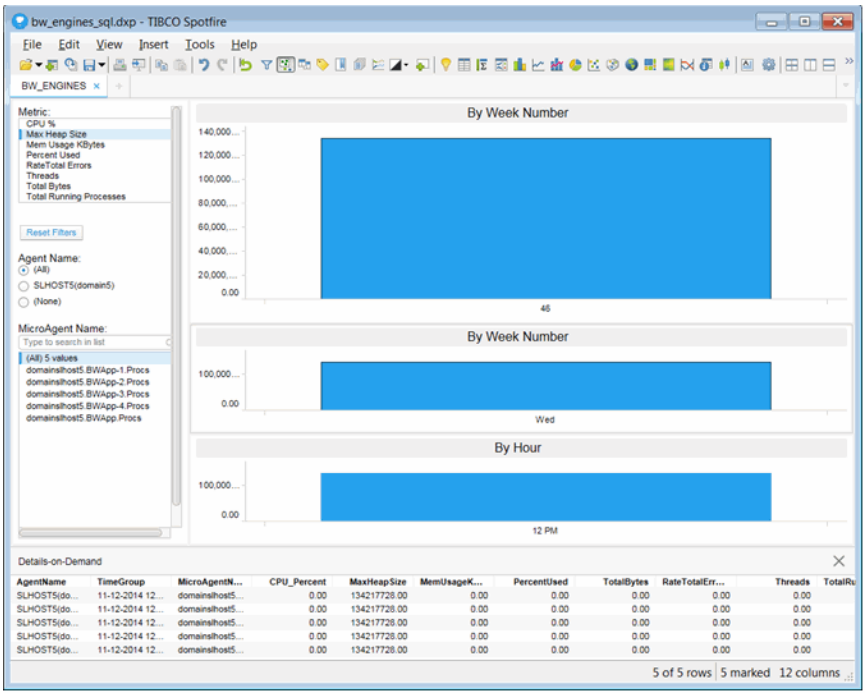


14. Click **OK**.

The **Replace Data Table** window opens.

15. Select **Import data table** as the **Load Method** and click **OK**.

Your report should display in the TIBCO Spotfire dashboard.



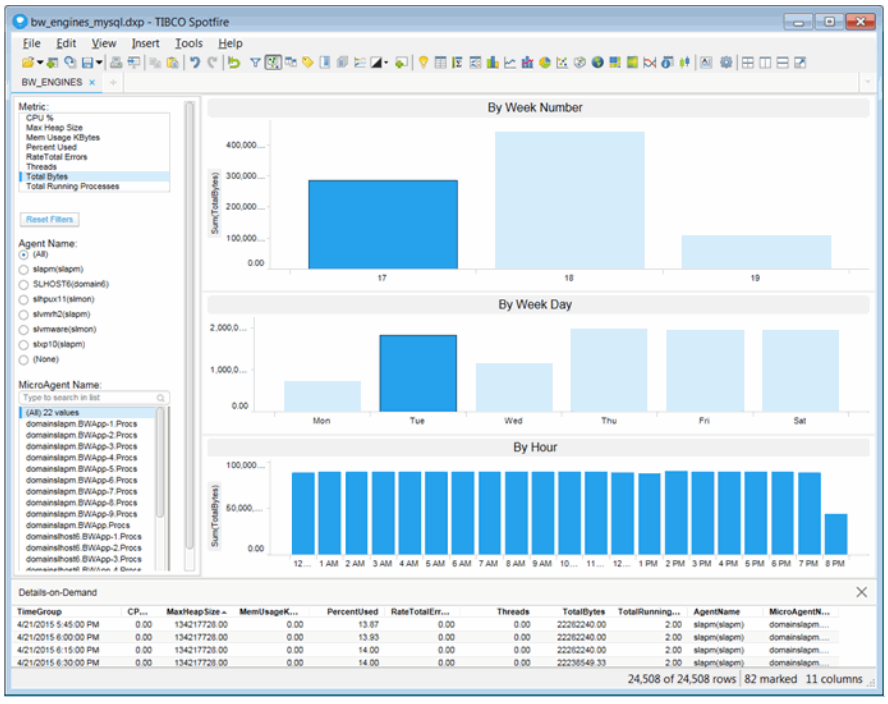
Reports

The following reports are available:

- "BusinessWorks Engines Metrics Report" on page 335

BusinessWorks Engines Metrics Report

The **BusinessWorks Engines Message Metrics Report** allows you to details for various metrics for one or more selected servers.



Metrics and Data
This report includes:

Metric	Lists the metrics available for the report.	
	CPU%	The percent (%) of server CPU used by the engine.
	Max Heap Size	The maximum amount of heap memory allocated to this engine for the JVM.
	Mem Usage Kbytes	The amount of server memory, in kilobytes, used by the engine.
	Percent Used	The percent (%) of allocated memory currently consumed by this engine from within the JVM. Equal to the value of: (**100*UsedBytes**) divided by **MaxBytes**.
	Rate Total Errors	The number of errors per second.
	Threads	The number of threads used by this engine.
	Total Bytes	The total bytes of memory within the JVM currently used by the engine.
	Total Running Processes	The number of running processes.
Reset All Filters	Resets the Agent Name and MicroAgent Name filters.	
Agent Name	The name of the TIBCO Hawk agent.	
MicroAgent Name	The name of the TIBCO Hawk MicroAgent.	
By Week Number	Displays the averages (for the Rate metrics) or sums (for the Count metrics) for the selected server(s) for each week. You can hover over each week to view the exact counts or rates for that week. Clicking on a particular week displays data for each day for that particular week in the **By Week Day** region.	
By Week Day	Displays the averages (for the Rate metrics) or sums (for the Count metrics) for each day in the selected week. Hovering over a particular day displays the exact sum or average for that day. Clicking on a particular day populates data for each hour in the **By Hour** region.	
By Hour	Displays the averages (for the Rate metrics) or sums (for the Count metrics) for each hour in the selected day. Hovering over a particular hour displays the exact sum or average for that hour. Clicking on a particular hour updates the **TimeBucket** information in the **Details-on-Demand** window.	

APPENDIX A Monitor Scripts

This section describes Monitor scripts and the **rtvservers.dat** configuration file. This section includes:

Scripts

This section describes scripts that are available for the Monitor.

The following scripts are available when used from an initialized command window. The scripts can be executed from a Windows Command Prompt or UNIX terminal window. On Windows, you can type the commands as described in this section. On UNIX systems, you must add **.sh** to each command. For example, **rtvapm_init.sh**.

These instructions assume use of a BASH or a BASH-compliant shell. The following scripts are available when used from an initialized command window. The scripts can be executed from a Windows Command Prompt or UNIX terminal window. On Windows, you can type the commands as described in this section. On UNIX systems, you must add **.sh** to each command. For example, **rtvapm_init.sh**. Also on UNIX systems, it is a requirement that the installation directory path not contain spaces.

These instructions assume use of a BASH or a BASH-compliant shell.

Script Name	Description
my_alert_actions.bat/sh	Sample script to define actions for alerts. Location: The project directory. Format: **my_alert_actions** (Append **.sh** on UNIX)
rtv_setup.bat/sh	Initializes a command prompt or terminal window. Location: **<installation directory>/bin** This script must be executed in the directory in which it resides. Format: **rtv_setup** (Append **.sh** on UNIX)
rtvapm_init.bat/sh	Initializes a command window. Location: **rtvapm** This script must be executed in the directory in which it resides. Format: **rtvapm_init** (Append **.sh** on UNIX)

start_cmd.bat	Starts an initialized Command Prompt window on Windows. Location: **<installation directory>/bin** This script must be executed in the directory in which it resides. You can also execute the script by double-clicking in an Explorer window.

start_rtv.bat/sh	Starts processes in an RTView configuration as specified in the **rtvservers.dat** configuration file.
	Location:
	rtvapm/common/bin
	This script must be executed in the project directory (the directory containing the **rtvservers.dat** file). This script requires **rtvapm_init.bat/sh** be executed first.
	An RTView configuration might include a Data Server or Display Server, an Historian and a Central Server Database. **start_rtv** only attempts to start processes it detects are not running. The action can be applied to all RTView configurations, a single RTView configuration or a single process in an RTView configuration.
	Before starting an RTView server, this script detects port conflicts caused by another server. If the conflict is caused by another RTView server, it returns a message identifying that server by its **rtvapm**. For example:
	`...start_rtv.bat: another dataserver running with JMX port 3268 under C:\rtview\RTViewDataServer\rtvapm`
	If the port conflict is caused by a non-RTView process, it returns a message similar to this, for example:
	`...start_rtv.bat: JMX port 3268 in use by PID 1234`
	In both cases the script includes this advice:
	`Warning: server not started, port conflict`

	To avoid port conflicts, run your start script with the **-portprefix:** command line argument to change the first two (**2**) digits of all your server ports. To persist these port changes, change the port prefix in the RTView Configuration Application or use the **-saveportprefix** command line argument. Additional arguments can be included on the command line in which case they are passed to every server specified by the command. Additional arguments can also be included in the **rtvservers.dat** file, in which case they are only applied to the specific server in whose command they are included. **Note:** If you use the **-properties** or **-propfilter** argument with **start_rtv**, you should also use them with **status_rtv** and **stop_rtv**. Those commands use the JMX ports defined for the server, and if any of the properties specified by **-properties** or **-propfilter** arguments change those ports, subsequent commands will be unable to find the server unless also given those properties.
	–console (or **–c**) - Start the processes with a command window (which is useful for testing).

	When used without arguments, this script returns usage information and a list of available configurations. For example, **start_rtv** returns:
	`Usage: start_rtv config or 'all' [server or 'all'] [args...]`
	`Available configs:`
	`default`
	`dataserver`
	`historian`
	`displayserver`
	`database`
	`sender`
	`dataserver`

	all Starts all RTView configurations that are specified in the **rtvservers.dat** file. **all** applies the action to all RTView configurations specified in the **rtvservers.dat** file (and corresponding servers or clients specified in each configuration). **Note:** When multiple configurations are specified in the **rtvservers.dat** file and they have different project settings directory locations, the **all** argument processes all the configurations. However, if the configurations have the same project settings directory locations, the **all** argument processes only the first configuration as the others are considered alternative configurations. Example: **start_rtv all** **(Append .sh on UNIX)**
	[Configuration Name] Starts a single RTView configuration specified in the **rtvservers.dat** file: **start_rtv [Configuration Name]** **(Append .sh on UNIX)** Configuration Name is the RTView configuration name specified in the **rtvservers.dat** file. The action applies to all servers or clients specified in the configuration. Example: **start_rtv web_deployment** **(Append .sh on UNIX)**
	[Server Name] Starts a single process in an RTView configuration specified in the **rtvservers.dat** file: **start_rtv [Configuration Name] [Server Name]** **(Append .sh on UNIX)** Server Name is the name of a server or client member in the configuration. For example, **dataserver**, **displayserver**, **historian** and **database**. The action applies only to that server or client in the configuration. Example: **start_rtv web_deployment dataserver**

	(Append .sh on UNIX)
	Use With Secured JMX Ports This script works with RTView servers whose JMX ports are secured with either a username and password, or with SSL. You provide the scripts with the necessary credential information and the scripts manage authentication with the server. There are two ways that you can provide credential information to the scripts: via command-line arguments and via properties placed in any property file that is used by the server. **Securing with username and password** • To secure with a username and password via command-line, use the arguments as follows: **-jmxuser:...** **-jmxpass:...** • To secure with a username and password in a property file, use the properties as follows: **sl.rtview.jmxremote.username=...** **sl.rtview.jmxremote.password=....** **Securing with SSL** To secure with SSL, you provide the client KeyStore and TrustStore locations and their corresponding passwords. • To secure with SSL via command-line, use the arguments as follows: **-sslkeystore:...** **-sslkeystorepass:...** **-ssltruststore:...** **-ssltruststorepass:...** • To secure with SSL in a property file, use the properties as follows: **sl.rtview.ssl.client.keyStore=...** **sl.rtview.ssl.client.keyStorePassword=...** **sl.rtview.ssl.client.trustStore=...** **sl.rtview.ssl.client.trustStorePassword=....** **Password Encryption** To encrypt the passwords in your properties files, use the command-line tool "encode_string", for example: **encode_string encoder2 password** This will give you an encrypted value for "password" that you can use in your properties.
start_server.bat/sh	Starts the RTView DataServer. Location: **<installation directory>** This script must be executed in the directory in which it resides. You can also execute the script by double-clicking in an Explorer window. Format: **start_server** (Append **.sh** on UNIX)
start_servers.bat/sh	Starts the RTViewCentral servers.

	Location: **<installation directory>/bin** This script must be executed in the directory in which it resides. You can also execute the script by double-clicking in an Explorer window. Format: **start_servers** (Append **.sh** on UNIX)
start_tomcat.bat/sh	Starts Apache Tomcat. Location: **<installation directory>/bin** This script must be executed in the directory in which it resides. You can also execute the script by double-clicking in an Explorer window. Format: **start_tomcat** (Append **.sh** on UNIX)
status_rtv.bat/sh	Returns the status of all RTView configurations that are specified in the **rtvservers.dat** configuration file. Location: **rtvapm/common/bin** This script must be executed in the project directory (the directory containing the **rtvservers.dat** file). This script requires **rtvapm_init.bat/sh** be executed first. This action uses defined JMX ports. An RTView configuration might include a Data Server, a Display Server or Viewer, an Historian and a Central Server Database. **status_rtv** only attempts to start processes it detects are not running. The action can be applied to all RTView configurations, a single RTView configuration or a single process in an RTView configuration. Additional arguments can be included on the command line in which case they are passed to every server specified by the command. Additional arguments can also be included in the **rtvservers.dat** file, in which case they are only applied to the specific server in whose command they are included. Note that if you use **-properties** or **-propfilter** arguments with **start_rtv**, you should also use them with **status_rtv** and **stop_rtv**. Those commands use the JMX ports defined for the server, and if any of the properties specified by **-properties** or **-propfilter** arguments change those ports, subsequent commands will be unable to find the server unless also given those properties.
	all Returns the status of all RTView configurations specified in the **rtvservers.dat** file. **Note:** When multiple configurations are specified in the **rtvservers.dat** file and they have different project settings directory locations, the **all** argument processes all the configurations. However, if the configurations have the same project settings directory locations, the **all** argument processes only the first configuration as the others are considered alternative configurations. Example: **status_rtv all** (Append **.sh** on UNIX)
	[Configuration Name] Returns the status of a single RTView configuration specified in the **rtvservers.dat** file:

	status_rtv [Configuration Name] **(Append .sh on UNIX)** Configuration Name is the RTView configuration name specified in the **rtvservers.dat** file. The action applies to all servers or clients specified in the configuration. Example: **status_rtv web_deployment** (Append **.sh** on UNIX)
	[Server Name] Returns the status of a single process in an RTView configuration specified in the **rtvservers.dat** file: **status_rtv [Configuration Name] [Server Name]** **(Append .sh on UNIX)** Server Name is the name of a server or client member in the configuration. For example, **dataserver**, **displayserver**, **historian** and **database**. The action applies only to that server or client in the configuration. Example: **status_rtv web_deployment dataserver** **(Append .sh on UNIX)**
	Use With Secured JMX Ports This script works with RTView servers whose JMX ports are secured with either a username and password, or with SSL. You provide the scripts with the necessary credential information and the scripts manage authentication with the server. There are two ways that you can provide credential information to the scripts: via command-line arguments and via properties placed in any property file that is used by the server. **Securing with username and password** • To secure with a username and password via command-line, use the arguments as follows: **-jmxuser:...** **-jmxpass:...** • To secure with a username and password in a property file, use the properties as follows: **sl.rtview.jmxremote.username=...** **sl.rtview.jmxremote.password=....** **Securing with SSL** To secure with SSL, you provide the client KeyStore and TrustStore locations and their corresponding passwords. • To secure with SSL via command-line, use the arguments as follows: **-sslkeystore:...** **-sslkeystorepass:...** **-ssltruststore:...** **-ssltruststorepass:...** • To secure with SSL in a property file, use the properties as follows: **sl.rtview.ssl.client.keyStore=...** **sl.rtview.ssl.client.keyStorePassword=...** **sl.rtview.ssl.client.trustStore=...** **sl.rtview.ssl.client.trustStorePassword=....**

	Password Encryption To encrypt the passwords in your properties files, use the command-line tool "encode_string", for example: **encode_string encoder2 password** This will give you an encrypted value for "password" that you can use in your properties.
status_server.bat/sh	Returns the status of the RTView DataServer. Location: **<installation directory>** This script must be executed in the project directory (the directory containing the **rtvservers.dat** file). Format: **status_server** (Append **.sh** on UNIX)
status_servers.bat/sh	Returns the status of the RTViewCentral servers (as well as the Solace PubSub+ Monitor in RTViewSolaceMonitor). Location: **<installation directory>/bin** This script must be executed in the project directory (the directory containing the **rtvservers.dat** file). Format: **status_servers** (Append **.sh** on UNIX)
stop_rtv.bat/sh	Stops processes in an RTView configuration as specified in the **rtvservers.dat** configuration file. Location: **rtvapm/common/bin** This script must be executed in the project directory (the directory containing the **rtvservers.dat** file). This script requires **rtvapm_init.bat**/**sh** be executed first. This action uses defined JMX ports. An RTView configuration might include a Data Server or a Display Server, an Historian and a Central Server Database. **stop_rtv** only attempts to start processes it detects are not running. The action can be applied to all RTView configurations, a single RTView configuration or a single process in an RTView configuration. Additional arguments can be included on the command line in which case they are passed to every server specified by the command. Additional arguments can also be included in the **rtvservers.dat** file, in which case they are only applied to the specific server in whose command they are included. Note that if you use **-properties** or **-propfilter** arguments with **start_rtv**, you should also use them with **status_rtv** and **stop_rtv**. Those commands use the JMX ports defined for the server, and if any of the properties specified by **-properties** or **-propfilter** arguments change those ports, subsequent commands will be unable to find the server unless also given those properties. Location: **project directory** **This script must be executed in the project directory (the directory containing the rtvservers.dat file).**
	all Stops all RTView configurations that are specified in the **rtvservers.dat** file. **all** applies the action to all RTView

	configurations specified in the **rtvservers.dat** file (and corresponding servers or clients specified in each configuration). **Note:** When multiple configurations are specified in the **rtvservers.dat** file and they have different project settings directory locations, the **all** argument processes all the configurations. However, if the configurations have the same project settings directory locations, the **all** argument processes only the first configuration as the others are considered alternative configurations. Example: **stop_rtv all** **(Append .sh on UNIX)**
	[Configuration Name] Stops a single RTView configuration specified in the **rtvservers.dat** file: **stop_rtv [Configuration Name]** **(Append .sh on UNIX)** Configuration Name is the RTView configuration name specified in the **rtvservers.dat** file. The action applies to all servers or clients specified in the configuration. Example: **stop_rtv web_deployment** **(Append .sh on UNIX)**
	[Server Name] Stops a single process in an RTView configuration specified in the **rtvservers.dat** file: **stop_rtv [Configuration Name] [Server Name]** **(Append .sh on UNIX)** **Server Name** is the name of a server or client member in the configuration. For example, **dataserver**, **displayserver**, **historian** and **database**. The action applies only to that server or client in the configuration. Example: **stop_rtv web_deployment dataserver** **(Append .sh on UNIX)**
	Use With Secured JMX Ports This script works with RTView servers whose JMX ports are secured with either a username and password, or with SSL. You provide the scripts with the necessary credential information and the scripts manage authentication with the server. There are two ways that you can provide credential information to the scripts: via command-line arguments and via properties placed in any property file that is used by the server. **Securing with username and password** • To secure with a username and password via command-line, use the arguments as follows: **-jmxuser:...** **-jmxpass:...** • To secure with a username and password in a property file, use the properties as follows: **sl.rtview.jmxremote.username=...** **sl.rtview.jmxremote.password=....** **Securing with SSL** To secure with SSL, you provide the client KeyStore and

	TrustStore locations and their corresponding passwords. • To secure with SSL via command-line, use the arguments as follows: **-sslkeystore:...** **-sslkeystorepass:...** **-ssltruststore:...** **-ssltruststorepass:...** • To secure with SSL in a property file, use the properties as follows: **sl.rtview.ssl.client.keyStore=...** **sl.rtview.ssl.client.keyStorePassword=...** **sl.rtview.ssl.client.trustStore=...** **sl.rtview.ssl.client.trustStorePassword=....** **Password Encryption** To encrypt the passwords in your properties files, use the command-line tool "encode_string", for example: **encode_string encoder2 password** This will give you an encrypted value for "password" that you can use in your properties.
stop_server.bat/sh	Stops the RTView DataServer. Location: **<installation directory>** This script must be executed in the directory in which it resides. Format: **stop_server** (Append **.sh** on UNIX)
stop_servers.bat/sh	Stops the RTViewCentral servers. Location: **<installation directory>/bin** This script must be executed in the directory in which it resides. You can also execute the script by double-clicking in an Explorer window. Format: **stop_servers** (Append **.sh** on UNIX)
stop_tomcat.bat/sh	Stops Apache Tomcat. Location: **<installation directory>/bin** This script must be executed in the directory in which it resides. Format: **start_tomcat** (Append **.sh** on UNIX)
update_wars.bat/sh	Creates/updates the primary Monitor servlets. Location: **<installation directory>/projects/rtview-server** This script must be executed in the directory in which it resides. This script requires **rtvapm_init.bat**/**sh** be executed first.

	Format: **update_wars.sh [appname [host [portprefix]]]** For example: **update_wars.sh my-appname my-hostname 99** The name, host, and portprefix are declared in variables at the top of the script for easy editing, and can be passed into the scripts on the command-line. **-secure** Use the "-secure" argument to update the rtvquery war with security enabled. You can use **?** or **help** to get a usage message. For example: **update_wars.sh help** You can edit other variables at the top of the scripts to set properties for high-availability (HA). **Set HA_HOST** to the hostname of the backup data server. **Set HA_DISPLAYHOST** to the hostname of the backup display server. **Set HA_FAILBACK** to true to automatically reconnect to the primary display server.

rtvservers.dat

This section describes the **rtvservers.dat** configuration file which is used to manage your TIBCO® RTView® for TIBCO BusinessWorks™ deployment and RTView processes. This section includes:

- Single Configuration File
- Multiple Configuration File

The **rtvservers.dat** text file contains one or more RTView configurations. An RTView configuration is a group of servers that should be started together. For example, the configuration might include any of the following: a Data Server, Historian, HSQLDB database, and a Display Server (for a Web Deployment). The **rtvservers.dat** file is used when the following scripts are executed:

- start_rtv Starts RTView processes specified in the **rtvservers.dat** file.
- stop_rtv Stops the RTView processes specified in the **rtvservers.dat** file.
- status_rtv Returns status information for RTView processes specified in the **rtvservers.dat** file.

Single Configuration File

The following **rtvservers.dat** file, located in your project directory, contains a single RTView configuration, named **default**.

```
default . dataserver rundata
default . historian runhist -ds
```

```
default . displayserver rundisp -ds
default . database rundb
```

Note: The last line in the **rtvservers.dat** file must end with a new line, or be followed by a blank line.

In this example, to start the **default** configuration type: **start_rtv default** or **start_rtv all**. To start a single server in the configuration, type **start_rtv <Configuration Name> <Server Name>**. For example: **start_rtv default displayserver**.

Each line has the following format consisting of four fields:

<Configuration Name> <Project Settings Directory Location> <Property Filter Identifying the Server> <Command>

<Configuration Name>	The name of the RTView configuration (**default** in this example).
<Project Settings Directory Location>	The RTView project settings directory location, relative to the location of the **rtvservers.dat** file (**.**, the current directory, in this example).
<Property Filter Identifying the Server>	The property filter that identifies the server, which is the property filter under which the server's JMX port is defined. By default, this is the server name, such as **dataserver**, **displayserver** and **historian**.
<Command>	The script used to start the process. Valid values are: • rundata: Starts the Data Server. • runhist: Starts the Historian. • rundisp: Starts the Display Server. • rundb: Starts the HSQLDB Database.

Multiple Configuration File

When multiple configurations are specified in the **rtvservers.dat** file and they have different project settings directory locations, the **all** argument processes all the configurations. However, if the configurations have the same project settings directory locations, the all argument processes only the first configuration as the others are considered alternative configurations. Alternative configurations allow you to alternate between two configurations for a single TIBCO® RTView® for TIBCO BusinessWorks™ deployment.

For example, the following **rtvservers.dat** file contains two configurations, **bwmon** and **emsmon**. Note that the project settings directory locations differ (**./bwmon** and **./emsmon**, respectively).

```
bwmon ./bwmon dataserver rundata
bwmon ./bwmon historian runhist -ds
bwmon ./bwmon displayserver rundisp -ds

emsmon ./emsmon dataserver rundata
emsmon ./emsmon historian runhist -ds
emsmon ./emsmon displayserver rundisp –ds
```

Because the project settings directory locations differ, you can use type **start_rtv all** to start both configurations. To start only the bwmon configuration, type: **start_rtv bwmon**. To start

a single server in the **bwmon** configuration, type **start_rtv <Configuration Name> <Server Name>**. For example: **start_rtv bemon displayserver**.

APPENDIX B Alert Definitions

This section describes alerts for TIBCO BusinessWorks and their default settings.

Alert	Warning Level	Alarm Level	Duration	Enabled
BW6AppErrorState BW6 application status is not Running or Stopped (status is Impaired, AppError or StartFailed) Index Type: PerApp Metric: State	NaN	NaN	30	FALSE
Bw6AppExpired BW6 application expired due to application inactivity. Index Type: PerApp Metric: Stopped	NaN	NaN	30	FALSE
Bw6AppNodeCpuUsedHigh BW6 AppNode CPU usage exceeded limit. CPU Usage is the CPU time in use by all processes expressed as a percentage of the total CPU time available. Index Type: PerAppNode Metric: CPU Usage%	50	80	30	FALSE
Bw6AppNodeMemUsedHigh BW6 AppNode memory usage exceeded limit. Memory usage is the percentage of total JVM memory currently consumed by this appnode. Index Type: PerAppNode Metric: Memory Usage%	50	80	30	FALSE
Bw6AppNodeStopped BW6 AppNode stopped purposefully (for example, an administrator stopped the AppNode process). Index Type: PerAppNode Metric: State	NaN	NaN	10	FALSE
Bw6AppNodeUnreachable BW6 AppNode stopped abnormally (for example, the AppNode process crashed). Index Type: PerAppNode Metric: State	NaN	NaN	10	FALSE
Bw6AppProcessCreatedRateHigh BW6 Process created rate for application exceeded limit. Index Type: PerApp Metric: App Created Rate	50	80	30	FALSE
Bw6AppProcessElapsedTimeHigh BW6 Process delta elapsed time rate of increase for	200	400	30	FALSE

application exceeded limit. Index Type: PerApp Metric: App Elapsed Rate				
Bw6AppProcessExecutionTimeHigh BW6 Process delta execution time rate of increase for application exceeded limit. Index Type: PerApp Metric: App Execution Rate	200	400	30	FALSE
Bw6AppProcessFailedRateHigh BW6 Process failed rate for application exceeded limit. Index Type: PerApp Metric: App Failed Rate	50	80	30	FALSE
Bw6AppStopped BW6 application stopped. Index Type: PerApp Metric: Stopped	NaN	NaN	30	FALSE
Bw6ProcessActivityErrorRateHigh BW6 Process error rate exceeded limit. Index Type: PerProcess Metric: Process Failed Rate	50	80	30	FALSE
Bw6ProcessCreatedRateHigh BW6 Process error rate exceeded limit. Index Type: PerProcess Metric: Process Failed Rate	50	80	30	FALSE
Bw6ProcessElapsedTimeHigh BW6 Process delta elapsed time rate of increase exceeded limit. Index Type: PerProcess Metric: Delta Exec Rate	200	400	30	FALSE
Bw6ProcessExecutionTimeHigh BW6 Process delta execution time rate of increase exceeded limit. Index Type: PerProcess Metric: Delta Time Rate	200	400	30	FALSE
Bw6ProcessFailedRateHigh BW6 Process suspended rate exceeded limit. Index Type: PerProcess Metric: Suspended Rate	50	80	30	FALSE
Bw6ProcessHung The delta elapsed time is greater than zero but the delta execution time is zero. Index Type: PerProcess Metric: Hung/Not Hung	NaN	NaN	10	FALSE
Bw6ProcessSuspendRateHigh BW6 Process failed rate exceeded limit. Index Type: PerProcess Metric: Failed Rate	50	80	30	FALSE

BwActivityErrorRateHigh BW5 Activity error rate exceeded limit. The rate is calculated by taking the delta of total error returns in this update period and dividing by the length of the period. Index Type: PerActivity Metric: RateErrorCount	50	80	30	FALSE
BwActivityExecutionTimeHigh BW5 Activity execution time rate of increase exceeded limit The rate is calculated by taking the delta of total execution time in this update period and dividing by the length of the period. Index Type: PerActivity Metric: RateExecutionTime	200	400	30	FALSE
BwEngineCpuUsedHigh BW Engine CPU usage (% of total) exceeded limit. CPU Usage is the CPU time used by the BW engine expressed as a percentage of the total CPU time available. Index Type: PerEngine Metric: CPU Usage%	50	80	30	FALSE
BwEngineMemUsedHigh BW Engine memory usage (% of total) exceeded limit. Memory usage is the percentage of total JVM memory currently consumed by this engine. Index Type: PerEngine Metric: PercentUsed	50	80	30	FALSE
BwEngineStopped BW Engine has stopped running. Index Type: PerEngine Metric: Stopped	NaN	NaN	30	FALSE
BwEngineUnreachable BW engine stopped abnormally. Index Type: PerEngine Metric: State	NaN	NaN	30	FALSE
BwProcessAbortRateHigh BW Process aborted rate exceeded limit. The rate is calculated by taking the delta of total aborts in this update period and dividing by the length of the period. Index Type: PerProcess Metric: RateAborted	50	80	30	FALSE
BwProcessAvgElapsedTimeHigh BW Process Average Elapsed Time exceeded limit. Value is calculated by dividing the delta elapsed time for the interval by the delta completed, or the number of process instances that completed in the interval. Index Type: PerProcess Metric: Process Avg Elapsed Time	100	200	30	FALSE
BwProcessAvgExecutionTimeHigh BW Process average execution time exceeded limit. Index Type: PerProcess Metric: AverageExecution	0	0	0	FALSE
BwProcessCreatedRateHigh	100	200	30	FALSE

BW Process creation rate exceeded limit. The rate is calculated by taking the number of process instances created in the interval and dividing by the length of the interval in seconds. Index Type: PerProcess Metric: Processes Created/sec				
BwProcessCreatedRateLow BW Process creation rate per second went below limit. Index Type: PerProcess Metric: App Created Rate	0	0	0	FALSE
BwProcessElapsedTimeHigh BW Process elapsed time rate of increase exceeded limit. The rate is calculated by taking the delta of total elapsed time in this update period and dividing by the length of the period. Index Type: PerProcess Metric: RateTotalElapsed	50	80	30	FALSE
BwProcessExecutionTimeHigh BW Process execution time rate of increase exceeded limit. The rate is calculated by taking the delta of total execution time in this update period and dividing by the length of the period. Index Type: PerProcess Metric: RateTotalExecution	50	80	30	FALSE
BwProcessHung The delta elapsed time is greater than zero but the delta execution time is zero. Index Type: PerProcess Metric: Hung/Not Hung	NaN	NaN	10	FALSE
BwProcessSuspendRateHigh BW Process suspended rate exceeded limit. The rate is calculated by taking the delta of total suspends in this update period and dividing by the length of the period. Index Type: PerProcess Metric: RateSuspended	50	80	30	FALSE
BwProcessTotalCpuPercentHigh BW Process CPU percent utilization exceeded limit. This is the percent CPU used by all process instances executing over the interval. Index Type: PerProcess Metric: Process Total CPU Percent	50	75	30	FALSE
BwServerCpuUsedHigh BW Server CPU usage (% of total) exceeded limit. CPU Usage is the CPU time in use by all processes expressed as a percentage of the total CPU time available. Index Type: PerServer Metric: CPU Usage%	60	85	30	FALSE
BwServerFreeMemLow BW Server free memory available (in megabytes) is below limit. Free memory means available physical (RAM) memory. Index Type: PerServer	15	5	30	FALSE

Metric: Memory Free Mbytes				
BwServerInactive BW Server has become inactive. The period of time specified by the substitution variable $bwserverExpirationTime has passed since data was last received from the server. Index Type: PerServer Metric: Expired	NaN	NaN	30	FALSE
BwServerMemUsedHigh BW Server memory usage (% of total) exceeded limit. Memory usage is the virtual memory in use expressed as a percentage of the available virtual memory. The meaning of available virtual memory is system-dependent: on Windows it refers to pagefile space; on Unix systems it refers to swap space. Index Type: PerServer Metric: Virtual Memory Used%	50	80	30	FALSE
HawkAlert Display Hawk alerts throughout the Monitor. To enable Hawk Alerts to be included in alert counts and displayed throughout the Monitor, scroll down to **HawkAlert** in the **Active Alert Table** and select the **Alert Enabled** checkbox. It is possible to filter unwanted alerts from the cache data so that those alerts are not included throughout the Monitor. To filter unwanted alerts out of the Hawk cache data, enter the following into the **sample.properties** file (located in the project directory you created). NOTE: Unwanted alerts are filtered out according to the AlertText. **sl.rtview.sub=$hawkAlertTextFilterOut:AlertText** For example, to filter out all Hawk Alerts in which the AlertText contains **Source** you would enter the following: **sl.rtview.sub=$hawkAlertTextFilterOut:Source** The default time to remove cleared Hawk Alerts from the table is **3600** seconds. To adjust this setting, edit the following in **sample.properties**: **sl.rtview.sub=$hawkAlertTextFilterOut:3600** Index Type: PerServer Metric: Hawk	NaN	NaN	-1	TRUE
JvmCpuPercentHigh The percentage of CPU that has been reached by the JVM is above the limit. Index Type: PerJVM Metric: CpuPercent	50	75	30	FALSE
JvmGcDutyCycleHigh The duty cycle is out the upper limit. Index Type: PerGC Metric: DutyCycle	50	75	30	FALSE
JvmMemoryUsedHigh The memory used out the upper limit Index Type: PerJVM	50	75	30	FALSE

Metric: MemoryUsedPercent				
JvmNotConnected The JVM in not connected. Index Type: PerJVM Metric: Connected	NaN	NaN	30	FALSE
JvmStaleData Cut in reception from that JVM. Index Type: PerJVM Metric: Expired	NaN	NaN	30	FALSE

APPENDIX C Limitations

This section includes:

TIBCO BusinessWorks

Business Works 5.7.1 Engine Status

The BW Engine microagent has a method **GetExecInfo** that includes a field called **Status**, which may have the following values:

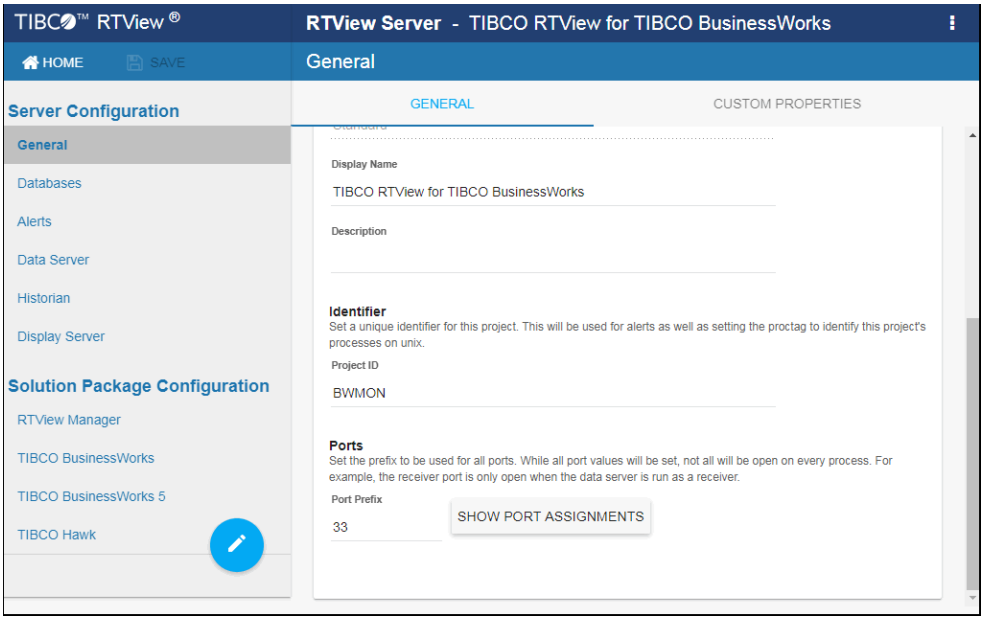
- ACTIVE
- SUSPENDED
- STANDBY
- STOPPING
- STOPPED

In Business Works 5.7.1 (but not earlier or later versions) this method fails to return any data and, in some cases when the Monitor starts, it may not know an engine's exact status. For example, if an engine is deployed but not active it could be SUSPENDED or STOPPED, or else it could be ACTIVE or STOPPING. In these cases the Monitor sets the status to UNKNOWN. An UNKNOWN status will be resolved once the engine is stopped and restarted; henceforth the status will display as STOPPED or ACTIVE.

BWSE Components

- JVM memory metrics are available for BWSE components running in AMX 3.x environments only.
- The BW Version column in the All Engines Table display is blank for BWSE components.
- The Deployment column in the All Engines Table display is UNKNOWN for BWSE components. This is because the AMX environment controls in which node or nodes a BWSE component is running, therefore the concept of "deployment" in traditional BusinessWorks does not apply.
- BWSE components only appear in the All Engines Table display when they are running in a node.

APPENDIX D RTView Configuration Application



The RTView Configuration Application is a tool that you can use to help configure the Monitor by defining various properties and connections via an easy-to-use interface. The RTView Configuration Application consists of three different sections: the top-level **Projects** page, the **Server Configuration** view, and the **Solution Package Configuration** view. This section will provide high-level definitions for each option within each view. More detailed descriptions on how this tool can be used to set up the Monitor can be found in the Configuration chapter, as well as in the Quick Start chapter.

This section contains the following:

Accessing the RTView Configuration Application

There are two ways you can access the RTView Configuration Application:

- Via URL
- Via a Button in the Monitor (if "Login" is enabled)

Via URL

You can access the RTView Configuration Application via URL by performing the following steps:

1. Download and extract the BWMON compressed .zip file.
2. Enabling Login in the Monitor.
3. Set the **JAVA_HOME** environment variable.
4. Run **start_server** (**.bat** or **.sh**) from your project directory to start all servers.
5. Open a browser and enter **http://localhost:3370/rtview-bwmon-rtvadmin**.

See Quick Start for additional details.

> **Note:** Once you have finished making changes in the RTView Configuration Application, you must restart your data server for your changes to take place in the Monitor.

Via a Button in the Monitor

> **Note:** "Login" must be enabled and you must log in with administrator privileges in order for the RTView Configuration Application button to display in the Monitor. See Enabling Login in the Monitor for more information. The RTView Configuration Application button is only available in the Display Server version of the User Interface. The button will not display in the HTML User Interface.

You can access the RTView Configuration Application via a button in the Monitor by performing the following steps:

1. Download and extract the BWMON compressed .zip file.
2. Enabling Login in the Monitor.
3. Set the **JAVA_HOME** environment variable.
4. Run **start_server** (**.bat** or **.sh**) from your project directory to start all servers.
5. Open a browser and enter **http://localhost:3370/rtview-bwmon-classic**.
   The Monitor displays.
6. Click the  in the Monitor to open the RTView Configuration Application.

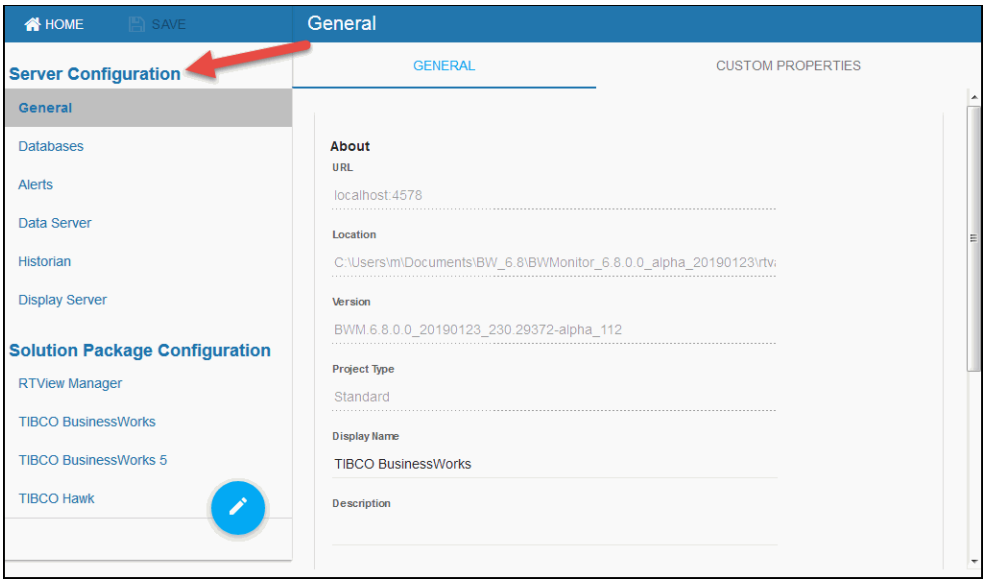
See Quick Start for additional details.

Note: Once you have finished making changes in the RTView Configuration Application, you must restart your data server for your changes to take place in the Monitor.

Projects Page

The Projects Page lists the project(s) in your project directory. Click the project to access the Configuration Views.

Server Configuration View

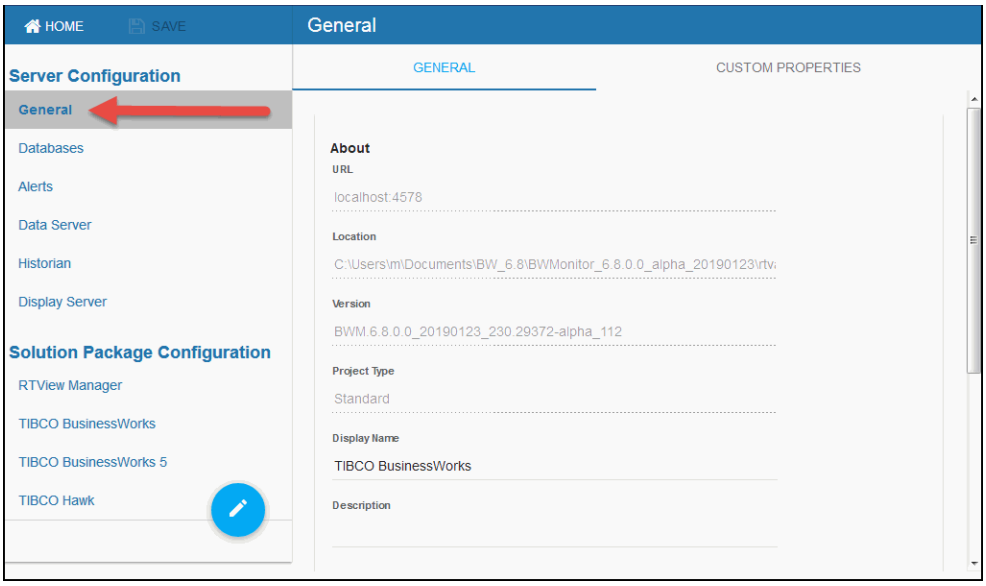


The **Server Configuration** View provides options that allow you to modify the default settings for the project including the project name and default port, define the alert threshold database connection and alert notification settings, define custom properties, define data server properties, define display server properties, and define the historian database connection and other historian properties. This section contains the following:

- General
- Databases
- Alerts
- Security
- Data Server
- Historian
- Display Server

Tip: Gray text shows the default setting for the field which you can edit. To return to the default setting, delete the text you entered.

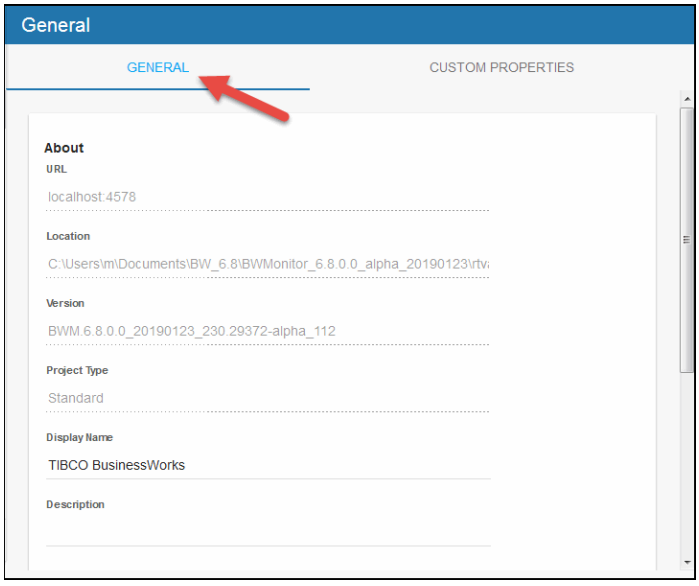
General



The **General** option consists of three different tabs that allow you to define the values for the project, specify the port, define the alert threshold database connection and other alert properties, and define any custom properties you might need to create. The three different tabs are:

- General Tab
- Custom Properties Tab

General Tab



This tab contains the following regions:

About

URL: Displays the URL used to the connect to the server. This field cannot be edited.

Location: Displays the project directory location (path). This field cannot be edited.

Version: Displays the current version of TIBCO BusinessWorks installed. This field cannot be edited.

Project Type: Displays the type of project (Standard, Sender, or ConfigClient). This field cannot be edited.

Display Name: Displays the default name for the project and displays on the Home/**RTView Project Configuration** (top level) page. This field can be edited.

Description: Optionally specify a description that will display on the Home/**RTView Project Configuration** (top level) page.

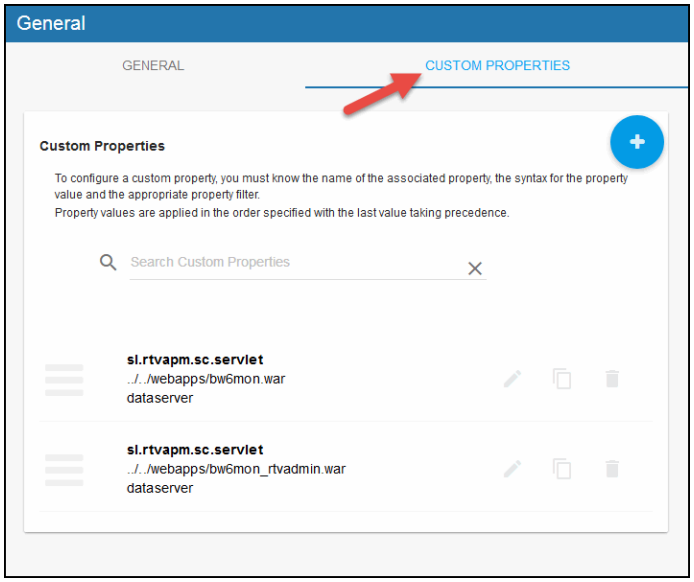
Identifier

Project ID: Displays a default unique identifier for the project, which you can modify.

Ports

Port Prefix: Displays the default port prefix (first two numbers used for the port) that will be used for all ports, which you can modify. The latter two numbers in the port are predefined and cannot be modified. Click **Show Port Assignments** to view the Port Assignments.

Custom Properties Tab



The **Custom Properties** tab allows you to create custom properties. Property values are applied in the order specified with the last value taking precedence. To create properties you need the name of the associated property, the syntax for the property value, and the appropriate property filter. Click the + icon to open the **Add Property** dialog, which has the following fields:

Name: (Required) The name of the associated property.

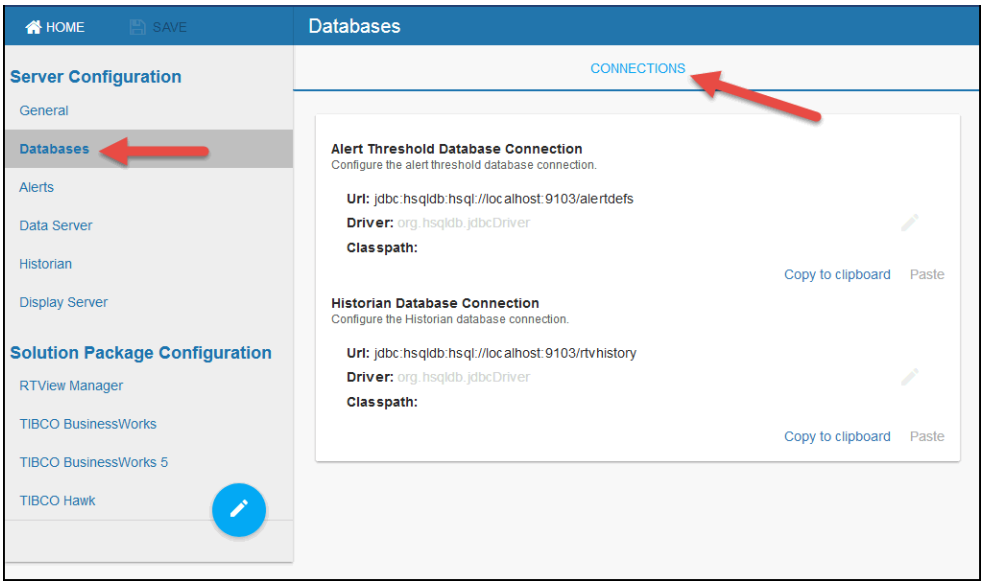
Value: (Optional) The value for the associated property (using the correct syntax).

Filter: (Optional) The filter for the associated property.

Comment: (Optional) Enter useful details about the property and its behavior for yourself and other users.

See Configure Alert Notifications for an example of when you can use **Custom Properties**.

Databases



The **Databases** option consists of the **Connections** tab that allows you to define Alert Threshold Database and Historian Database connections. The **Connections** tab contains the following regions:

Alert Threshold Database Connection

If you want to use local alert threshold database connection, add the connection information where:

URL: The complete URL for the database connection.

Driver: The full name for the driver.

Classpath: The complete classpath for the jar location.

Username: The username is used when creating the connection. This field is optional.

Password: This password is used when creating the connection. This field is optional. By default, the password entered is hidden. Click the icon to view the password text.

Run Queries Concurrently: When selected, database queries are run concurrently.

Historian Database Connection

URL: The complete URL for the database connection.

Driver: The full name for the driver.

Classpath: The complete classpath for the jar location.

Username: The username is used when creating the connection. This field is optional.

Password: This password is used when creating the connection. This field is optional. By default, the password entered is hidden. Click the icon to view the password text.

Run Queries Concurrently: When selected, database queries are run concurrently.

Alerts

The Alerts option consists of the Alerts tab and the History tab, which allow you to define the alert and history properties. Alert and Historian database connections are set up using the Databases option. The following tabs are available:

- Alerts Tab
- History Tab

Alerts Tab

This tab contains the following regions:

Notifications

- **Enable Alert Notifications**: Selecting this toggle enables alert notifications to be sent.
- **Notification Platform**: Select the platform type (**UNIX** or **Windows**).

Alert Event Options

- **Notify on New Alerts**: A notification is executed every time a new alert is created.
- **Nofity on First Severity Change**: A notification is executed the first time the **Severity** changes for each alert.
- **Nofity on Cleared Alerts**: A notification is executed every time an alert is cleared.
- **Periodically Renotify on Unacknowledged Alerts**: Enter the **Renotification Interval** (number of seconds). A notification is executed for each unacknowledged alert per the interval you specify here. If the Renotification Interval is greater than **0** and no actions are defined, the **New Alerts** action will be used for renotifications.

Selecting the  button next to each of the Alert Event Options displays the following options:

 Run a Script	This alert notification action executes the following script in the **TIB_rtview-ems/projects/rtview-server** directory: • **my_alert_actions.bat/sh** – New and First Severity Change • **my_alert_actions.cleared.bat/sh** – Cleared • **my_alert_actions.renotify.bat/sh** – Periodically Renotify This action can only be added once per notification type. In addition to selecting this action in the Configuration Application, you must also modify the appropriate script to execute the actions for your notification. This script has access to the following fields from the alert: **Alert Name**, **Alert Index**, **ID**, **Alert Text** and **Severity**.

This alert notification action allows you to implement your alert notification actions using Java code. It executes the **my_alert_notification.$domainName.$alertNotifyType.$alertNotifyCol** command in your Custom Command Handler and passes the row from the alert table that corresponds to the alert.

Execute Java Code

This action can only be added once per notification type. In addition to selecting this action the Configuration Application you must also modify the custom command handler to execute the actions for your notification. A sample custom command handler is included under **projects/custom**. It prints the alert notification to the console. You will modify this command handler to implement your own notification actions.

Make the following entries:

- **Custom Command Handler Class Name**: Enter the fully qualified name of the Custom Command Handler class. This defaults to the sample Custom Command Handler in the **TIB_rtview-ems/projects/custom** directory.
- **Custom Command Handler Jar**: Enter the path and name of the jar containing the Custom Command Handler class. The path may be absolute or relative to the location of data server. This defaults to the sample Custom Command Handler in the **TIB_rtview-ems/projects/custom** directory.

Note that if you can only have one custom command handler per Data Server, so changing these settings for one notification event will change them for the rest of the notification events.

Send Email

This alert notification action sends an email. This action can be added multiple times per notification type. No additional setup is required beyond filling in the **Send Email** dialog in the Configuration Application.

Make the following entries:

- **SMTP Host**: The SMTP host address. This is required. Consult your administrator.
- **SMTP Port**: The SMTP port number. This is required. Consult your administrator.
- **From**: The email address to which to send the email. This is required.
- **To**: The email address to which to send the email. This is required and may contain multiple entries.
- **Subject**: The subject for the email. This is required. You can include the value from any column in the alert table in your subject. Click the **Show More** link at the bottom of the dialog to see the alert column values you can use in the **Subject**.
- **Body**: The body of the email. This is optional. Click the **Show More** link at the bottom of the dialog to see the alert column values you can use in the **Subject**.

- **User**: The user name for the account from which you are sending the email. This is optional.
- **Password**: The password for the account from which you are sending the email. This is optional.

Send SNMP Trap

This alert notification action sends an SNMP Trap as described in **rtvapm/common/lib/SL-RTVIEW-EM-MIB.txt**. This action can be added multiple times per notification type. No additional setup is required beyond filling in the **Send Email** dialog in the Configuration Application

Make the following entries:

- **Trap Type**: Select the SNMP version of the trap. This is required.
- **Destination Address**: The system name or IP address of the receiving system. This is required.
- **Destination Port**: The UDP port on the receiving system. This is required.
- **Community Name**: (This field is visible when **Trap Type v2/v3** is selected.) The SNMP v2 Community Name string. This is required.

Run Command String

This alert notification action executes a specified command. This action can be added multiple times per notification type. Make the following entry:

Command String: Enter the command string for any command supported by RTView Classic. To enter a command string, you must know the correct syntax for the command. Contact Technical Support for assistance on syntax. You can include the value from any column in the alert table using the syntax in the Show More link at the bottom of the dialog.

Conditional Filter

This alert notification action alert allows you to execute different actions for different alerts based on information in the alert. For example, you can configure EMS alerts to send emails to your EMS team and Solace alerts to send emails to your Solace team. This action can be added multiple times per notification type.

To create a condition, make the following entries:

- **Alert Field**: Select an alert field: **Alert Name**, **Alert Index**, **Category, CI Name, Owner, Package, Primary Service** or **Severity**. This is required. Note that **CI Name** and **Primary Service** fields are for RTViewCentral only.
- **Operator**: Select one - **EQUALS**, **DOES NOT EQUAL**, **STARTS WITH**, **ENDS WITH** or **CONTAINS**. This is required.
- **Value**: Enter the value to which to compare the Alert Field. Cannot contain wildcard characters. This is required.
- **Action(s)**: Select one or more actions to execute when this condition is met.

Persistence

Persist Alerts: When enabled, saves alerts to the database for high availability purposes.

History Tab

This tab contains the following region:

History

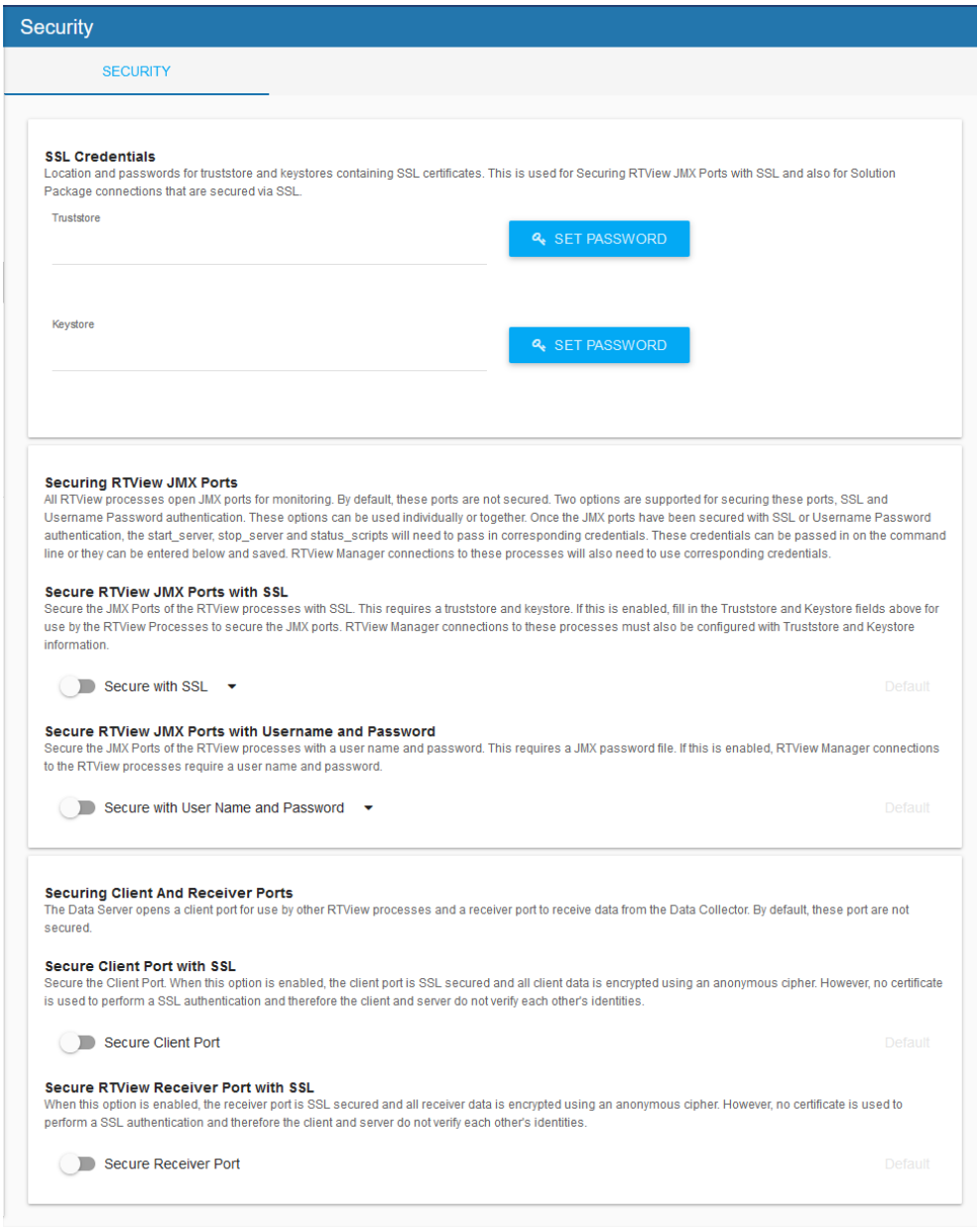
Store Alert History: Toggle to enable/disable **Store Alert History** to store alerts in the history database. This value is used in the **Alerts Table** (which makes it easier to find the alerts).

History Table Name Prefix: This field allows you to define a prefix that will be added to the database table names so that the Monitor can differentiate history data between data servers when you have multiple data servers with corresponding Historians using the same solution package(s) and database. In this case, each Historian needs to save to a different table, otherwise the corresponding data server will load metrics from both Historians on startup. Once you have defined the **History Table Name Prefix**, you will need to create the corresponding tables in your database as follows:

- Locate the .sql template for your database under **TIB_rtview-bw/rtvapm/bwmon/dbconfig** and make a copy of it
- Add the value you entered for the **History Table Name Prefix** to the beginning of all table names in the copied .sql template
- Use the copied .sql template to create the tables in your database

Security

All RTView processes (Data Server, Historian, Display Server) open JMX ports for monitoring which, by default, are not secured. The **Security** tab allows you secure these ports as well as specify credentials needed to connect to SSL secured servers from RTView's Solution Packages.



SSL Credentials

This region allows you to specify the path to the **Truststore** and **Keystore** files (and their associated passwords) that contain the SSL credentials needed to secure the RTView JMX Ports and/or access any SSL secured servers associated with RTView's Solution Packages. This is required if the **Secure with SSL** option is enabled (see below for details).

Optional: To obscure the credentials of the truststore and keystore in the output of the **ps** and **jps** commands, add the following custom property to each Data Server on which SSL Credentials have been configured:

```
Name: sl.rtview.jvm
Value: -Drtv.hidesslprops=true
Comment: hide ssl properties in ps/jps output
```

Securing RTView JMX Ports

This region provides a couple of options for securing the JMX ports that are opened by the RTView processes: **Secure with SSL** and/or **Secure with Username and Password**.

Secure with SSL

When toggled on, this option secures the JMX ports for the RTView processes with SSL. When the JMX ports are SSL secured, an SSL handshake is performed between the client and the server when the client attempts to connect. For this handshake, the client must provide a certificate the server trusts, and the server must provide a certificate the client trusts. A Keystore file contains the application's certificate and private key and a Truststore file contains the application's trusted certificates. These files are created by running the Java keytool on the command line. When this option is enabled, you must specify the path to the server's Truststore and Keystore files (and their associated passwords) in the **SSL Credentials** region (see above).

The **start_server**, **stop_server**, and **status_server** scripts are all connected to the JMX Ports of the RTView processes to execute operations and get status. If the JMX ports have been secured with SSL, these scripts need the path and passwords for the truststore and keystore files containing the client credentials in order to connect. You can either pass these in on the command line each time you run (**-sslkeystore:clientkeystore.jks-sslkeystorepass:clientkeystorepass-ssltruststore:clienttruststore.jks-ssltruststorepass:clienttruststorepass**) or you can fill in the fields under **SSL Credentials for RTView Scripts**.

The RTView Manager application also connects to the JMX Ports of the RTView processes in order to monitor them. If you are using the RTView Manager and the JMX ports have been secured with SSL, you must fill in the **SSL Credentials** on the **Security** tab of the RTView Manager Configuration Application to specify the path the truststore and keystore files containing the client credentials.

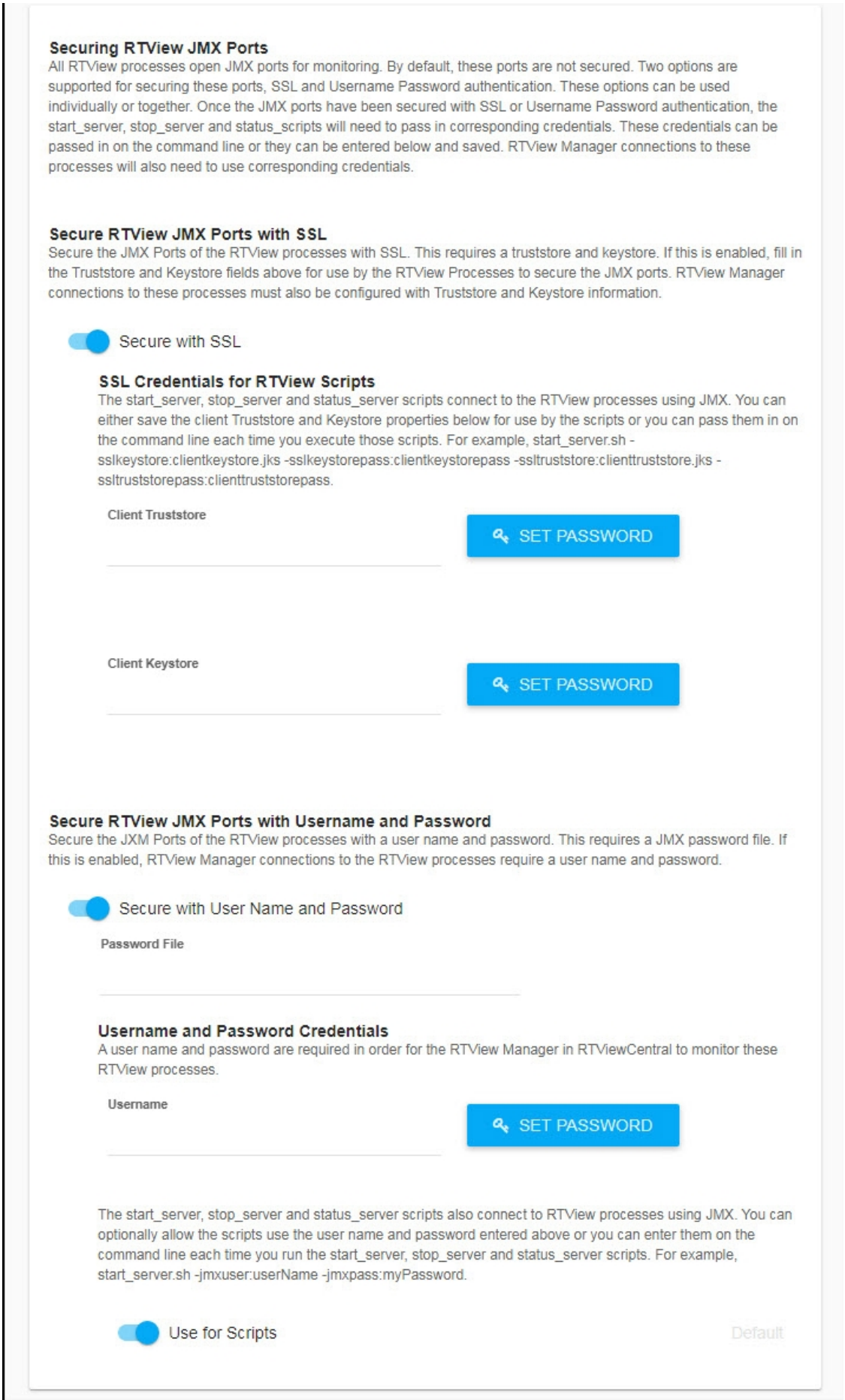
Secure with Username and Password

This region allows you to secure the JMX ports for RTView processes with a username/password. This can be used in addition to Securing with SSL (see above). If this option is enabled, you must specify the path to a JMX password file containing all valid user names and passwords.

Important! A JMX password file must be read-only to the owner. See Java documentation for details on the creation of a JMX remote password file.

The **start_server**, **stop_server**, and **status_server** scripts are all connected to the JMX Ports of the RTView processes to execute operations and get status. If the JMX ports have been secured with a username and password, the scripts need a valid user name and password in order to connect. You can either pass these into the command line each time you run (**-jmxuser:userName-jmxpass:myPassword**) or you can fill in the **Username and Password Credentials** and enable the **Use for Scripts** toggle.

The RTView Manager application also connects to the JMX Ports of the RTView processes in order to monitor them. If you are using the RTView Manager in RTViewCentral and the JMX ports have been secured with username and password, you must fill in the **Username and Password Credentials** that the RTView Manager should use to connect. If you are using the RTView Manager in a deliverable other than RTViewCentral, you will need to fill in the user name and password in the connection to this RTViewDataServer in the RTView Manager Configuration Application.


Securing RTView JMX Ports
All RTView processes open JMX ports for monitoring. By default, these ports are not secured. Two options are supported for securing these ports, SSL and Username Password authentication. These options can be used individually or together. Once the JMX ports have been secured with SSL or Username Password authentication, the start_server, stop_server and status_scripts will need to pass in corresponding credentials. These credentials can be passed in on the command line or they can be entered below and saved. RTView Manager connections to these processes will also need to use corresponding credentials.
Secure RTView JMX Ports with SSL
Secure the JMX Ports of the RTView processes with SSL. This requires a truststore and keystore. If this is enabled, fill in the Truststore and Keystore fields above for use by the RTView Processes to secure the JMX ports. RTView Manager connections to these processes must also be configured with Truststore and Keystore information.
Secure with SSL
SSL Credentials for RTView Scripts
The start_server, stop_server and status_server scripts connect to the RTView processes using JMX. You can either save the client Truststore and Keystore properties below for use by the scripts or you can pass them in on the command line each time you execute those scripts. For example, start_server.sh -sslkeystore:clientkeystore.jks -sslkeystorepass:clientkeystorepass -ssltruststore:clienttruststore.jks -ssltruststorepass:clienttruststorepass.
Client Truststore
SET PASSWORD
Client Keystore
SET PASSWORD
Secure RTView JMX Ports with Username and Password
Secure the JXM Ports of the RTView processes with a user name and password. This requires a JMX password file. If this is enabled, RTView Manager connections to the RTView processes require a user name and password.
Secure with User Name and Password
Password File
Username and Password Credentials
A user name and password are required in order for the RTView Manager in RTViewCentral to monitor these RTView processes.
Username
SET PASSWORD
The start_server, stop_server and status_server scripts also connect to RTView processes using JMX. You can optionally allow the scripts use the user name and password entered above or you can enter them on the command line each time you run the start_server, stop_server and status_server scripts. For example, start_server.sh -jmxuser:userName -jmxpass:myPassword.
Use for Scripts
Default

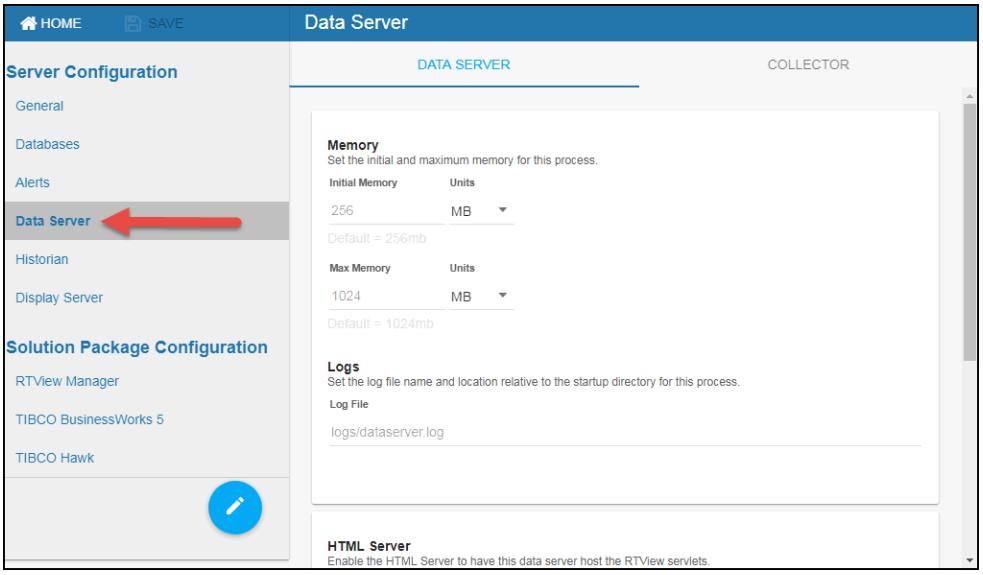
Secure Client and Receiver Ports with SSL

The Data Server opens a client port for use by other RTView processes and a receiver port to receive data from the Data Collector. By default, these port are not secured.

When **Secure Client Port with SSL** is enabled, the client port is SSL secured and all client data is encrypted using an anonymous cipher. However, no certificate is used to perform a SSL authentication and therefore the client and server do not verify each other's identities.

When **Secure RTView Receiver Port** is enabled, the receiver port is SSL secured and all receiver data is encrypted using an anonymous cipher. However, no certificate is used to perform a SSL authentication and therefore the client and server do not verify each other's identities.

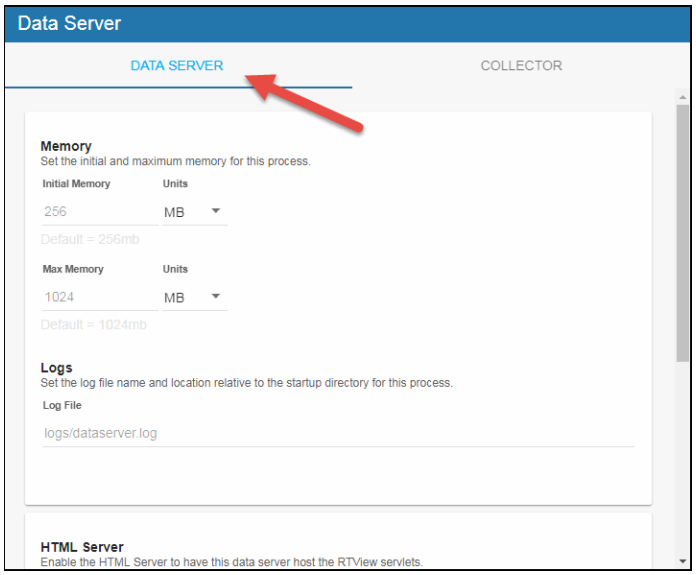
Data Server



This section describes the Data Server Configuration settings. There are two tabs available:

- Data Server Tab
- Collector Tab

Data Server Tab



This tab contains the following:

Memory: Set the initial memory and maximum memory for the Data Server process. Specify a number followed by a unit. Units are k (kilobyte), m (megabyte), g (gigabyte). If no unit is used, the number is assumed to be bytes. **Note**: Use caution when you change the memory allocation. If the memory allocation is too small the server might crash during startup and if too large the server might eventually exceed the available CPU/memory and fail.

Initial Memory: The initial amount of memory to allocate for this process.

Max Memory: The maximum amount of memory to allocate for this process.

Logs

Log File: The log file name and location relative to the startup directory for this process. In the **Log File** field, use the following format: **<directory name>/<log file name>**.

For example, logs/dataserver.log.

HTML Server

HTML Server Enabled: Enable the Eclipse Jetty HTML Server in the Data Server. If enabled, Eclipse Jetty will host the RTView Servlets at **http://localhost:XX70**, where **XX** is the port prefix specified on the **Server Configuration** > **General** > **GENERAL** tab. **Note**: You cannot disable this option if the RTView Configuration Application is being hosted by Eclipse Jetty in the Data Server. All RTView Servlets hosted by Eclipse Jetty are automatically configured with the correct Data Server port at runtime. The following RTView Servlets are hosted in Eclipse Jetty:

rtview-bwmon-classic

rtview-bwmon-rtvadmin

rtvadmin

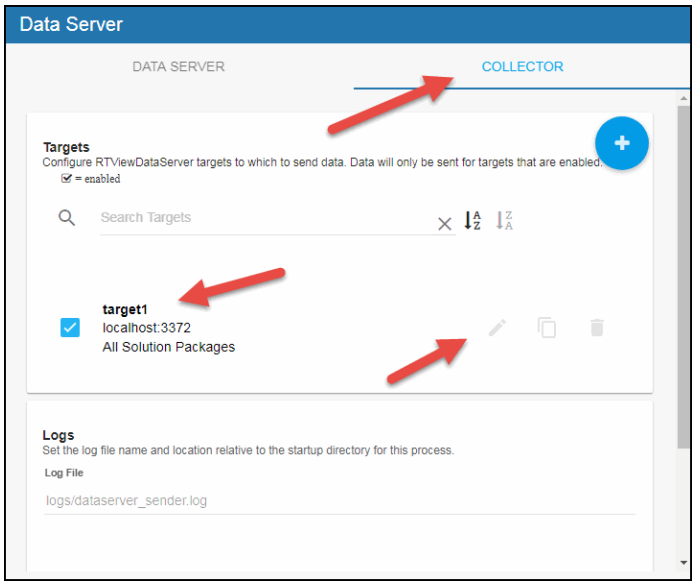
rtvdata

rtvquery

rtvagent

rtvpost

Collector Tab



This tab contains the following:

Targets: If you only have one target, you can edit the default target by clicking the "pencil" icon next to the **target1**. You can specify multiple targets by adding them one at a time. All fields on the **Add Target** dialog are required. Click the + icon to open the **Add Target** dialog, which has the following fields:

ID: A unique name for the target.

URL: Specify the URL for the receiver. The url can be **host:port** (for example, somehost:3372) or an **http url** for the rtvagent servlet on the receiver. For example, if you were using Tomcat, you would use: **http://somehost:8068/bwmon-rtvagent**. If you were using Jetty, you would use: **http://somehost:3370/rtvagent**.

Targets: Select the **All solution packages** option.

Enabled: Select this check box to enable the target.

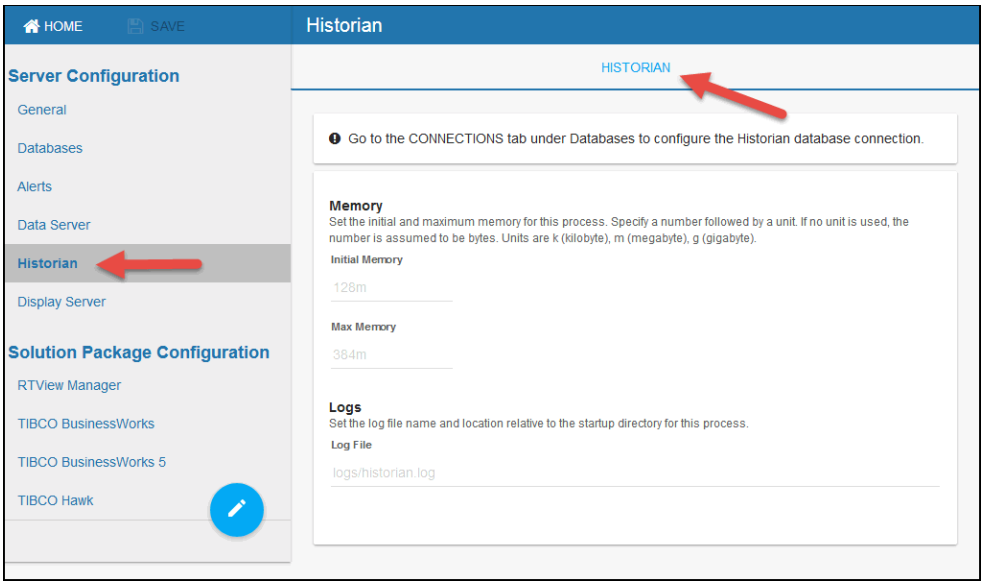
Logs

Log File: The log file name and full path.

Identifier

Name: A unique name for the sender data server, which is typically your machine's name.

Historian



The **Historian** option consists of the **Historian** tab, which allows you to define the history properties. Historian database connections are set up using the Databases option. This option contains the following regions:

Memory: Set the initial memory and maximum memory for the Historian process. Specify a number followed by a unit. Units are k (kilobyte), m (megabyte), g (gigabyte). If no unit is used, the number is assumed to be bytes. **Note**: Use caution when you change the memory allocation. If the memory allocation is too small the server might crash during startup and if too large the server might eventually exceed the available CPU/memory and fail.

Initial Memory: The initial amount of memory to allocate for this process.

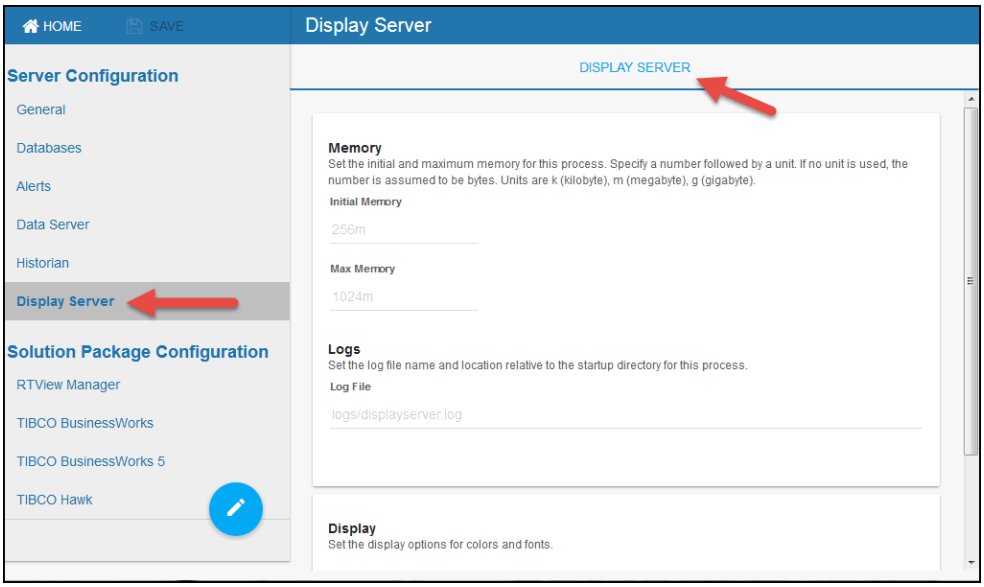
Max Memory: The maximum amount of memory to allocate for this process.

Logs

Log File: The log file name and location relative to the startup directory for this process. In the **Log File** field, use the following format: **<directory name>/<log file name>**.

For example, logs/historian.log.

Display Server



This tab contains the following regions:

Memory: Set the initial memory and maximum memory for the Display Server process. Specify a number followed by a unit. Units are k (kilobyte), m (megabyte), g (gigabyte). If no unit is used, the number is assumed to be bytes. **Note**: Use caution when you change the memory allocation. If the memory allocation is too small the server might crash during startup and if too large the server might eventually exceed the available CPU/memory and fail.

Initial Memory: The initial amount of memory to allocate for this process.

Max Memory: The maximum amount of memory to allocate for this process.

Logs

Log File: The log file name and location relative to the startup directory for this process. In the **Log File** field, use the following format: **<directory name>/<log file name>**.

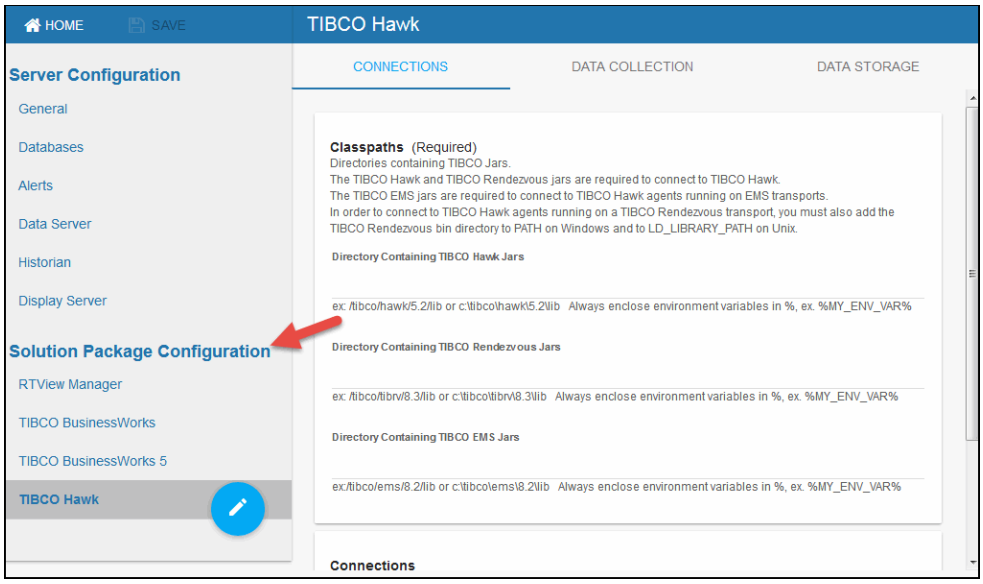
For example, logs/displayserver.log.

Display

Use Light Stylesheet: By default, RTView displays appear with a black background. Select this option for all RTView displays to appear with a white background.

Enable Cross Platform Fonts: Selecting this option prevents LINUX users from seeing inconsistently aligned labels in displays. This option should only be applied to Display Servers on Linux AND only if the text size and alignment issue is observed in the Thin Client. Otherwise, it can cause unnecessary overhead or unwanted changes to the appearance of text in RTView displays.

Solution Package Configuration View



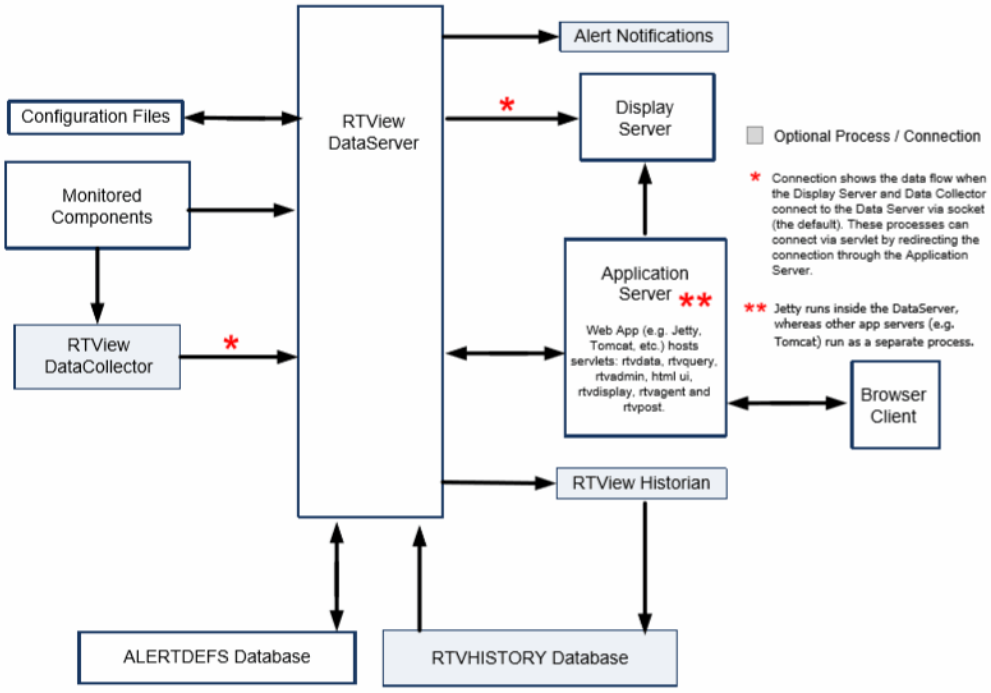
The **Solution Package Configuration** View provides options that allow you to modify the default settings for the project including data collection and data storage properties for the RTView Servers, define the classpaths and connections for Hawk, and define the data collection and data storage properties for your configured version(s) of TIBCO BusinessWorks (version 5 and/or version 6). Descriptions for all of the properties for these options, as they pertain to TIBCO BusinessWorks, are explained in detail in the Configuration chapter. You can also view the basic steps to get TIBCO BusinessWorks up and running in the Quick Start chapter.

APPENDIX E Security Configuration

This section provides details for securing a direct connection RTView deployment. This section contains:

Introduction

The following diagram shows how data flows through the RTView deployment. The Data Server connects to the Monitored Components to collect metric data which it stores in local caches. The Data Server uses the collected data to generate alerts based on enabled threshold settings in the ALERDEFS database. If the user has (optionally) defined alert notifications, the Data Server also executes them.



In cases where the data collection needs to be distributed, one or more Data Collectors can be deployed to connect to the Monitored Components and forward the collected data to the Data Server.

The HTML UI and Display Server (thin client) are browser-based user interfaces that show metric and alert data from the Data Server and also allow the user to enable, disable and set thresholds on alerts.

The Historian is an optional process that stores historical metric and alert data to the RTVHISTORY database. When the Historian is enabled, the Data Server queries historical data from the RTVHISTORY on startup to populate in-memory history and also any time the Display Server or HTML UI request history data that is older than the data in the in-memory history.

The Configuration Application is a browser based application for configuring the RTView processes. It connects to the Data Server to read and write Configuration Files.

The next sections provide a more detailed description of each process.

Port Settings

This document describes port settings for the following TIBCO applications, where the **XX** prefix is replaced with the following:

- For TIBCO EMS, replace **XX** with **31**
- For TIBCO BusinessWorks5, replace **XX** with **33**
- For TIBCO BusinessWorks6, replace **XX** with **45**
- For TIBCO BusinessEvents, replace **XX** with **32**

Data Server

The Data Server gathers and caches the data from the applications being monitored and also hosts the alerts for that data. Because the Data Server can exist behind firewalls, it simplifies and strengthens the secured delivery of information to clients beyond the firewall. The Data Server serves metric and alert data to the Display Server and Historian via socket on port **XX78** and receives data via socket from the optional Data Collector on port **XX72**. It also serves metrics and alert data to the HTML UI via the rtvquery servlet which connects via socket on port **XX78**.

The Historian and Display Server run in the same directory as the Data Server, while the optional Data Collector(s) typically run in a different directory or a different system. By default, socket connections to the Data Server are unsecured. The Data Server supports secure socket connections (SSL) with or without certificates. It also supports client whitelist and blacklist. Secure socket and client whitelist/blacklist configuration are described in the *RTView Core User's Guide* under Deployment/Data Server/Security.

The Display Server can optionally be configured to connect to the Data Server via the rtvdata servlet instead of the socket. In this case, the rtvdata servlet connects to the Data Server via socket on port **XX78**. While the rtvdata servlet cannot be configured for authentication, Tomcat access filters can be used to restrict access. The rtvdata servlet will connect via secure socket if the Data Server is configured for SSL sockets.

The HTML UI connects to the Data Server via the rtvquery servlet. See HTML UI in this document for information on how to enable authentication in the HTML IU and rtvquery servlets. The rtvquery servlet will connect via secure socket if the Data Server is configured for SSL sockets.

The Data Collector can optionally be configured to send data to the Data Server via the rtvagent servlet instead of the socket. In this case, the rtvagent servlet connects to the Data Server via socket on port **XX72**. While the rtvagent servlet cannot be configured for authentication, Tomcat access filters can be used to restrict access. The rtvagent servlet will connect via secure socket if the Data Server is configured for SSL sockets.

The Configuration Application connects to the Data Server via the rtvadmin servlet to read and write properties files. The rtvadmin servlet connects to the Data Server via socket on port **XX78**. See Configuration Application in this document for information about servlet authentication. The rtvadmin servlet will connect via secure socket if the Data Server is configured for SSL sockets.

If the Historian is enabled, the Data Server connects to the RTVHISTORY database on startup to read initial cache history data and if the thin client or HTML UI request history data older than the in memory cache history. It also connects to the ALERTDEFS database to query and set alert thresholds. See Database in this document for more information.

The Data Server optionally executes alert notifications based on user settings. Since the notification actions are user defined, security must be determined by the user.

The Data Server opens a JMX port on **XX68** to enable monitoring. By default, the JMX port is not secured. See RTView Processes under Monitored Components for information on securing this connection.

By default, the Data Server runs a Jetty process which hosts all of the RTView servlets and accepts HTTP client requests on port **XX70**. You can optionally configure Jetty to use HTTPS instead of HTTP.

Also see Port Settings.

Display Server (thin client)

The Classic user interface deployment, the thin client, is implemented using the Display Server. The Display Server consists of two parts, the Display Server application and the Display Servlet (**rtview-<sp>-classic.war**, where **<sp>** is replaced with **emsmon**, **bwmon**, **bw6mon** or **tbemon**). The Display Server application is generally installed on a dedicated platform. It loads displays from the file system and queries data from the Data Server that it passes on to the Display Servlet via a socket. The Display Servlet runs on an application server (like Tomcat or Jetty). Browser clients connect to the Display Servlet using HTTP or HTTPS (depending on the Application Server configuration). This process opens a JMX port on **XX79** to enable monitoring. By default, the JMX port is not secured. See RTView Processes under Monitored Components for information on securing this connection.

Also see Port Settings.

For the thin client, we support our **users**/**role.xml** login (which can be customized to integrate with LDAP) as well as SSO using BASIC or DIGEST HTTP authentication. This is described in the *RTView Core User's Guide* under Role-based Security and also under Deployment/Browser Deployment/Display Server.

> **Note:** The external libraries used for PDF export capability, iText and iTextAsian, can trigger security warnings due to a High Severity CVE (http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-9096) which "might allow remote attackers to conduct XML external entity attacks via a crafted PDF".
>
> These libraries do not pose a genuine security risk. The only PDFs that are handled by iText in RTView are those generated from customer displays, and therefor there is no mechanism to exploit the vulnerability with a "crafted PDF".
>
> Users who do not need to use the PDF export or reporting capability, and who would like to remove any library that raises alarms in a security scan, can remove gmsjextpdf.jar from the lib directory.

HTML UI

The new user interface is implemented in HTML and is deployed as a servlet, **rtview-<sp>mon** (where **<sp>** is replaced with **emsmon**, **bwmon**, **bw6mon** or **tbemon**), which is configured by default to use BASIC HTTP authentication. Browser clients connect via HTTP or HTTPS depending on the Application Server configuration. It should be used with HTTPS since BASIC authentication does not encrypt user credentials. The HTML UI connects to the Data Server via the rtvquery servlet. See Data Server for information on securing the connection between the rtvquery servlet and the Data Server. By default, the rtvquery servlet is not configured for authentication, but can be modified to require BASIC HTTP authentication as follows (this should be used with HTTPS since BASIC authentication does not encrypt user credentials):

1. Extract the web.xml file from the rtvquery servlet as follows:

   jar -xf rtview-<sp>-rtvquery.war WEB-INF/web.xml

   where <sp> is replaced with emsmon, bwmon, bw6mon or tbemon
2. Open **WEB-INF/web.xml** in a text editor and uncomment the security section.
3. Pack the modified **web.xml** file back into the rtvquery servlet as follows:

   jar -uf rtview-<sp>-rtvquery.war WEB-INF/web.xml

   where <sp> is replaced with emsmon, bwmon, bw6mon or tbemon

After you enable BASIC HTTP authentication in the rtvquery servlet, you will also need to modify the HTML UI to pass in credentials:

1. Extract the **setup.js** file from r**tview-<sp>.war** as follows:

   jar -xf rtview-<sp>.war setup.js

   where <sp> is replaced with emsmon, bwmon, bw6mon or tbemon
2. Open **setup.js** in a text editor and remove the **//** from the beginning of the **authValueC** line: **//authValueC: 'Basic ' + btoa('rtvuser:rtvuser')**
3. Pack the modified **setup.js** file back into the HTML UI servlet as follows:

   jar -uf rtview-<sp>.war setup.js

   where **<sp>** is replaced with **emsmon**, **bwmon**, **bw6mon** or **tbemon**

Data Collectors

This process is optional and is used to distribute connections to Monitored Components Data Collectors intead of having the Data Server connect to each component to be monitored directly. This process collects data from Monitored Components and forwards it to the Data Server via socket or the rtvagent servlet. See Data Server for information about securing the connection between the Data Collector and Data Server. This process does not keep history or process alerts - those are handed in the Data Server. While the Data Collector typically does not have data clients, it accepts data requests via socket on port **XX76** which can be secured as described in the Data Server section. It runs Jetty on port **XX70** and also opens JMX on port **XX66** for monitoring. By default, the JMX port is not secured. See RTView Processes under Monitored Components for information on securing this connection.

Also see Port Settings.

Configuration Application

The Configuration Application connects to the Data Server via the rtvadmin servlet which is configured with BASIC HTTP authentication. It should be run on HTTPS since Basic Authentication does not encrypt user credentials. Passwords saved by the configuration application are scrambled except in the case where they are added in the **CUSTOM PROPERTIES** section. See Data Server for information about securing the connection between the Configuration Application and Data Server.

Configuration Files

Configuration (**.properties** and **.properties.json**) files are stored on the file system and read by all RTView processes to control configuration. Additionally, the Configuration Application writes these files, scrambling all connection and database passwords. Passwords entered in the **CUSTOM PROPERTIES** tab are not scrambled.

Historian

The Historian connects to the Data Server via socket and saves cache history to a database via JDBC. This process is optional and the user can configure which data will be saved. See Data Server for information about securing the connection between the Historian and Data Server. See Database for information about the connection between the Historian and the database.

This process opens JMX port **XX67** for monitoring. By default, the JMX port is not secured. See RTView Processes under Monitored Components for information on securing this connection.

Also see Port Settings.

Database

The ALERTDEFS database stores alert threshold information and optionally alert persistence information. The Data Server connects to the ALERTDEFS database to query thresholds and also to set thresholds when the user interacts with the **Alert Administration** page in the user interface. The RTVHISTORY database stores cache data (if the Historian is enabled). The Historian connects to the RTVHISTORY database to insert cache history data and to perform data compaction. The Data Server connects to the RTVHISTORY database on startup to load initial history into the caches and also when the user interface asks for history data older than what is contained in the in-memory history caches.

By default, the Data Server and Historian connect to the HSQLDB database that is included with RTView using an unsecured JDBC connection. See the HSQLDB documentation for information on configuring it for secure JDBC connections. Alternately, you can use your own database and secure the JDBC connection according to the documentation for that database.

Application Servers

By default, the Data Server runs a Jetty process which hosts all of the RTView servlets and accepts HTTP client requests on port **XX70**. You can optionally configure Jetty to use HTTPS instead of HTTP. This will require you to provide a certificate for your domain.

Also see Port Settings.

When you have a certificate, do the following in the Configuration Application in the **Data Server** tab:

1. Turn on the **Use HTTPS** toggle.
2. Set the **Keystore File** to the keystore file name (including the path) that contains the certificate for your domain.
3. Optionally enter the **Keystore Password** and **Key Manager Password** if they are required for your keystore.
4. **Save** your configuration and restart the data server.

The Configuration Application and HTML UI use BASIC HTTP authentication and require the following roles which are preconfigured. You can modify the user names and passwords (but not the roles) in **RTVAPM_HOME/common/lib/ext/jetty/rtvadmin-users.xml**:

- rtvadmin
- rtvuser
- rtvalertmgr

Jetty does not limit the number of failed login attempts which leaves it open to brute force attacks. If this is a concern, you should deploy with Tomcat or another Application Server.

You can optionally use Tomcat or another Application Server in addtion to or instead of the Jetty process that comes with RTView. To deploy your servlets to your application server, go into the **RTVAPM__HOME/<sp>/projects/sample** directory (where **<sp>** is replaced with **emsmon**, **bwmon**, **bw6mon** or **tbemon)** and run **update_wars.bat** or **update_wars.sh**. Copy all of the generated war files to the **webapps** directory in your application server.

Tomcat and most other Application Servers can be configured for HTTPS. This will require you to provide a certificate for your domain. Follow the application server instructions to enable HTTPS.

Additionally, Tomcat access filters can be configured to restrict access according to the remote client host or address. Tomcat also has a feature named LockOut Realm to protect against brute force login attacks. After 5 successive login attempts for a given username with invalid password, then all logins for that username are rejected for the next 5 minutes. The LockOut Realm parameters are configurable. See Apache Tomcat documentation for more information.

You will need to add the following roles to your Application Server for use with the Configuration Application and HTML UI authentication. For Tomcat, users and roles are defined in **conf\tomcat-users.xml**:

- rtvadmin
- rtvuser
- rtvalertmgr

You can optionally disable Jetty in the Data Server when using Tomcat or another Application Server. To disable Jetty, you must access the Configuration Application from Tomcat or another Application Server. In the Configuration Application, go to the **Data Server** tab and do the following:

- Turn off the **HTML Server Enabled** toggle.
- **Save** your configuration and restart.

Monitored Components

Monitored Components are the processes that the Data Server and Data Collector connect to in order to request metric data. Some examples of Monitored Components are EMS Servers, Oracle Databases and RTView Processes. Connections to Monitored Components are made through application-specific APIs, so the options for securing these connections differ based on the Monitored Component.

This section contains:

- TIBCO BusinessEvents
- TIBCO BusinessWorks 5
- TIBCO BusinessWorks 6
- TIBCO EMS Server
- TIBCO Hawk
- RTView Manager
- RTView Processes

TIBCO BusinessEvents

The Data Server connects to BusinessEvents using JMX. BusinessEvents does not support secure JMX connections.

TIBCO BusinessWorks 5

The Data Server connects to TIBCO BusinessWorks 5 using TIBCO Hawk. See TIBCO Hawk for information about securing those connections. Additional server metrics can optionally be collected via JMX using the RTView Manager . See the TIBCO BusinessWorks 5 documentation for enabling JMX and securing it in your TIBCO BusinessWorks engine. See RTView Manager in

this document for information on making secure connections to JMX. For BWSE engines, RTView collects AMX Node data via JMS messages which are hosted on an EMS Server. The EMS Server can be configured to require a user name and password which the user enters in the Configuration Application when you define the Connection to that Server. Additionally, the EMS Server can be configured to use SSL. In this case, the user must implement a subclass of the GmsRtViewJmsDsSSLHandler to return a Map of the required SSL parameters per connection. This is described in the *RTView Core User's Guide* under **RTView Data Sources/JMS Data Source/Application Options - JMS/JMS Connections Tab/JMS SSL Parameters**.

TIBCO BusinessWorks 6

The Data Server connects to BusinessWorks 6 either using TIBCO Hawk or via the OSGI plugin. See the TIBCO Hawk section for information about securing TIBCO Hawk connections. When using the OSGI plugin, data is sent via socket to the Data Server on port **XX72**. By default, this socket is not secure, but the data will be sent via secure socket if the Data Server is configured for SSL sockets.

Also see Port Settings.

TIBCO EMS Server

The Data Server connects to EMS Servers using TIBCO's TibjmsAdmin API. The EMS Server can be configured to require a user name and password which the user enters in the Configuration Application/**EMS Server Connection** dialog when you define the connection to that EMS Server. Additionally, the EMS Server can be configured to use SSL. In this case, the user must implement a subclass of the GmsRtViewTibJmsSSLHandler to return a Map of the required SSL parameters per connection. This is described in the *RTView Core User's Guide* under **RTView Data Sources/TIBCO EMS Administration Data Source/Application Options - TIBCO EMS/TIBCO EMS Servers Tab/TIBCO EMS Administration SSL Parameters**.

TIBCO Hawk

TIBCO Hawk is used to gather metrics for both BusinessWorks 5 (required) and BusinessWorks 6. The Data Server connects to TIBCO Hawk via TIBCO's TIBHawkConsole API. The TIBCO Hawk installation may either be configured to run on a Rendezvous transport (rvd) or an EMS Transport. In the case of Rendezvous transports, no secure connection options are supported. In the case of EMS transports, the TIBCO Hawk agent can be configured to require a user name and password which the user enters in the Configuration Application **TIBCO Hawk Connection** dialog when adding a connection to that agent. The EMS transport can also be configured for SSL. In this case, the user must implement a subclass of the GmsRtViewHawkCustomSSLHandlerto return a Map of the required SSL parameters per connection. This is described in the *RTView Core User's Guide* under **RTView Data Sources/TIBCO Hawk Data Source/Application Options - TIBCO Hawk/TIBCO Hawk SSL Parameters**.

RTView Manager

The RTView Manager connects to processes via JMX. A process that opens a JMX port can be configured to require a user name and password which the user enters in the Configuration Application **RTView Manager Connection** dialog when you define the connection to that process. A process that opens a JMX port can also be configured to require SSL. To connect to

SSL JMX, in the **CUSTOM PROPERTIES** tab, add the following properties replacing the values with the appropriate values for your SSL configuration:

1. Set key store:
   a. **Property Name**: sl.rtview.jvm
   b. **Propety Value**: -Djavax.net.ssl.keyStore=client_keystore.jks
2. Set keystore password:
   a. **Property Name**: sl.rtview.jvm
   b. **Property Value**: -Djavax.net.ssl.keyStorePassword=mypassword
3. Set truststore
   a. **Property Name**: sl.rtview.jvm
   b. **Property Value**: sl.rtview.jvm=-Djavax.net.ssl.trustStore=client_truststore.jks
4. Set truststore password
   a. **Property Name**: sl.rtview.jvm
   b. **Property Value**: sl.rtview.jvm=-Djavax.net.ssl.trustStorePassword=mypassword
5. **Save** your changes and **Restart Servers**.

RTView Processes

The Data Server, Data Collector, Historian and Display Server all open JMX ports for monitoring. By default, these JMX ports are unsecured, but they can be secured either by user name and password or by SSL. See RTView Manager in this document for instructions on connecting to secure JMX. Note that the **start**/**stop**/**status_rtv** scripts use JMX to communicate with the RTView processes. If you secure the JMX ports on the RTView processes, you will need to pass additional arguments into the **start**/**stop**/**status_rtv** scripts as described in the *RTView Enterprise Monitor Configuration Guide*.

To secure with user name and password:

1. Setup a password file in **JRE_HOME/lib/management** using the **jmxremote.password.template** file as a template as described in the Oracle docs (for example, you might refer to Using File-Based Password Authentication here: https://-docs.oracle.com/javase/8/docs/technotes/guides/management/agent.html).
2. Start the Data Server for the project you want to secure and open the Configuration Application. In the **CUSTOM PROPERTIES** tab, add the following properties:
   - Disable ssl

   Property Name: sl.rtview.jvm

   Property Value: -Dcom.sun.management.jmxremote.ssl=false
   - Enable authentication:

   Property Name: sl.rtview.jvm

   Property Value: -Dcom.sun.management.jmxremote.authenticate=true
   - Set password file:

   Property Name: sl.rtview.jvm

   Property Value: -Dcom.sun.management.password.file=jmxremote.properties (or the name and path to a different password file)
3. **Save** your changes and **Restart Servers**. The JMX Port on all RTView Servers for that project will now require a user name and password.

To secure with SSL:

If you do not already have a key pair and certificate setup on your server, follow the instructions in the Oracle docs (for example, you might refer to Using SSL/To Setup SSL, Step 1 here: https://docs.oracle.com/javase/8/docs/technotes/guides/management/agent.html).

1. Start the Data Server for the project you want to secure and open the Configuration Application. In the **CUSTOM PROPERTIES** tab, add the following properties replaceing the values with the appropriate values for your SSL configuration:
   - Enable ssl

   Property Name: sl.rtview.jvm

   Property Value: -Dcom.sun.management.jmxremote.ssl=true
   - Set key store:

   Property Name: sl.rtview.jvm

   Propety Value: -Djavax.net.ssl.keyStore=server_keystore.jks
   - Set keystore password:

   Property Name: sl.rtview.jvm

   Property Value: -Djavax.net.ssl.keyStorePassword=mypassword
   - Set truststore

   Property Name: sl.rtview.jvm

   Property Value: sl.rtview.jvm=-Djavax.net.ssl.trustStore=server_truststore.jks
   - Set truststore password

   Property Name: sl.rtview.jvm

   Property Value: sl.rtview.jvm=-Djavax.net.ssl.trustStorePassword=mypassword
2. **Save** your changes and **Restart Servers**. The JMX Port on all RTView Servers for that project will now require SSL.

Security Summary

Security options per RTView process are included in the section for each component above. This section provides a summary of security options for the entire deployment organized by priority.

This section contains:

Secure Installation Location - High Priority

The RTView installation and Application Server should be run in a secure location to ensure displays and configuration files are secure and access-restricted.

Login and Servlet Authentication - High Priority

- **HTML UI** - By default, the HTML UI is configured with BASIC HTTP authentication which should use HTTPS since BASIC authentication does not encrypt user credentials. The HTML UI connects to the Data Server via the rtvquery servlet. The rtvquery servlet does not have authentication enabled by default. See the HTML UI section in this document for information on enabling authentication in the rtvquery servlet.
- **Display Server** - By default, the Display Server authentication is disabled. See the Display Server (thin client) section in this document for information on enabling authentication for the Display Server.
- **Configuration Application** - By default, the Configuration Application is configured with BASIC HTTP authentication which should use HTTPS since BASIC authentication does not encrypt user credentials.

Application Server Security - High Priority

It is highly recommended that you configure your Application Server to use HTTPS as described in the Application Servers section of this document. The RTView servlets that support HTTP authentication all use BASIC authentication which does not encrypt user credentials.

It is highly recommended that you change the user credentials in your Application Server for the rtvadmin, rtvuser and rtvalertmgr roles since the default credentials are documented and publicly available.

Secure Connections between RTView Processes - Medium-to-Low Priority*

The Historian, Data Server, Data Collector, rtvquery servlet, rtvdata servlet, rtvadmin servlet and rtvagent servlet all connect to the Data Server via socket which is unsecured by default. The Data Server supports secure socket connections (SSL) with or without certificates. It also supports client whitelist and blacklist. Secure socket and client whitelist/blacklist configuration are described in the RTView Core User's Guide under **Deployment/Data Server/Security**.

Secure Connections to Monitored Components - Medium-to-Low Priority*

The Data Server uses component specific API's to connect to Monitored Components. See the Monitored Components section in this document for information on how to secure these connections.

Secure Connections to Databases - Medium-to-Low Priority*

The Data Server and Historian both create database connections using JDBC. See the Database section in this document for information on securing JDBC connections to your database.

*If Secured Installation Location has been met, these are lower priority.