Business Value Dashboard

Software Version: 10.62

Administration Guide

Document Release Date: August 2017
Software Release Date: August 2017

Hewlett Packard Enterprise

Legal Notices

Warranty

The only warranties for Hewlett Packard Enterprise products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. Hewlett Packard Enterprise shall not be liable for technical or editorial errors or omissions contained herein.

The information contained herein is subject to change without notice.

Restricted Rights Legend

Confidential computer software. Valid license from Hewlett Packard Enterprise required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Copyright Notice

© 2015 - 2017 Hewlett Packard Enterprise Development LP

Trademark Notices

Adobe® and Acrobat® are trademarks of Adobe Systems Incorporated.

AMD, the AMD Arrow symbol and ATI are trademarks of Advanced Micro Devices, Inc.

Citrix® and XenDesktop® are registered trademarks of Citrix Systems, Inc. and/or one more of its subsidiaries, and may be registered in the United States Patent and Trademark Office and in other countries.

Google™ and Google Maps™ are trademarks of Google Inc.

Intel®, Itanium®, Pentium®, and Intel® Xeon® are trademarks of Intel Corporation in the U.S. and other countries.

iPad® and iPhone® are trademarks of Apple Inc.

Java is a registered trademark of Oracle and/or its affiliates.

Linux® is the registered trademark of Linus Torvalds in the U.S. and other countries.

Microsoft®, Windows®, Lync®, Windows NT®, Windows® XP, Windows Vista® and Windows Server® are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

NVIDIA® is a trademark and/or registered trademark of NVIDIA Corporation in the U.S. and other countries.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates.

Red Hat® is a registered trademark of Red Hat, Inc. in the United States and other countries.

SAP® is the trademark or registered trademark of SAP SE in Germany and in several other countries.

UNIX® is a registered trademark of The Open Group.

Contents

Get Started

Data does not have to be big to be confusing. It often comes in different formats or languages, and is spread over different locations and devices. This makes it all the harder to bring data together into one view, displaying only the latest and most important information.

HPE Business Value Dashboard (BVD) brings your data to life. Use BVD to create custom, flexible dashboards that visualize information in an appealing way and that can be accessed anywhere, anytime, from any device. Incorporate your own graphics, add color to identify status, and receive real-time updates—so you always understand the value driven by your IT environment.

BVD, color your data.

Anytime, Anywhere. BVD dashboards are real-time dashboards. You choose how often you send data to BVD; BVD displays the data with no delay. You decide where you want to view your dashboards: PC, tablet, or phone. BVD supports the major browsers. Choose your favorite!

Simple, Colorful, Flexible. Design your dashboards using Microsoft Visio. BVD provides a Visio stencil with shapes that then later become the widgets in your dashboards. The shapes include widgets for drawing charts, coloring text or values, displaying information feeds, web pages (for example, video streams), and many more. You can of course also style your dashboards according to your company's style guidelines. BVD provides sample dashboards to help you get started.

Connect. Once uploaded to BVD, you connect your widgets to the data. The BVD Manage Dashboards page makes this task simple and efficient. You can set additional widget options such as rules that determine the visibility and status colors of the widgets; or you can link widgets to other dashboards to enable drill down.

Integrate. BVD can process any kind of data as long as it is sent in JavaScript Object Notation (JSON), a language-independent, open data format. The out-of-the-box integration with OMi facilitates the integration of event and KPI status as well as metrics data. Operations Connector provides policies that automatically forward data collected from various sources to BVD. Alternatively, create your own integrations for any data source by writing an adapter for BVD. The adapter must convert the source data to JSON and send the JSON-enabled data to the BVD data receiver.

Log into BVD

You can access BVD using a supported web browser from any computer with a network connection. The level of access granted to a user depends on the user's permissions.

For a list of supported browsers, see the support matrix:

Support Matrices for Operations Center products

Download and extract the support matrix files, open the document SUMA.htm and select **Operations Manager i Business Value Dashboard** from the product list.

Tasks

This section includes:

- "How to Log Into BVD" below
- "How to Log Out of BVD" on the next page

How to Log Into BVD

1. In a browser, enter the following URL:

   https://*<external_access_host>*/bvd

   where

   <external_access_host> is the fully qualified domain name of the host which you specified as EXTERNAL_ACCESS_HOST in the install.properties file during the Container Deployment Foundation installation. Usually, this is the master node's FQDN.

2. Enter your login name and password. Initial access can be gained using the administrator user name and password as specified during the BVD configuration.

   > **Caution:** We recommend that the system superuser change this password upon first login to prevent unauthorized entry. For details on changing the user password, see "Change your name" on the next page. The login name cannot be changed.

After you log in, your login name appears on the 🧑 **Personal User Settings** menu.

How to Log Out of BVD

When you complete your session, it is recommended that you log out to prevent unauthorized entry.

Click **Log Out** on the 👤 **Personal User Settings** menu.

Change your name

The My Account dialog box enables individual users to change their name and password. In addition, users can provide their email address.

The changes made here are also applied to the user's configuration in User Management. For details, see "Manage users" on page 43.

To access

Click **My Account** on the 👤 **Personal User Settings** menu.

Tasks

How to Update My Account Information

1. In the **Email** field, enter your email address.

2. In the **Name** field, type a new display name.

3. Click **Change** to save your changes.

Create your first dashboard

Step 1: Before You Begin

1. Make sure Visio is installed on the system where you plan to create dashboards for BVD.

2. Log into BVD:

   a. Access BVD at the following URL:

      https://*<external_access_host>*/bvd

      Enter your login name and password.

   b. *Recommended.* Open the 👤 **Personal User Settings > My Account** menu and specify a new password.

3. In BVD, open 🔧 **Administration > System Settings**, and copy the **API Key**.

   This key identifies your BVD instance and must be included in the data submitted by the data senders.

   > **Note:** The System Settings page is only available to users with administrator privileges.

4. Download the following tools and samples in 🔧 **Administration > Download Tools & Samples**, and install them:

   ○ Visio Stencil (Contains the BVD widgets for Visio.)

   ○ Data Generator Sample (Contains a VBScript and a batch file to send data to the OMi sample dashboard.)

Step 2: Try the OMi Sample Dashboard

BVD provides the sample dashboard **OMi Event and KPI Status Samples** (`omi_sample`) as well as a batch and a VBScript file for sending data to the example. By following the steps below, you will become more familiar with the BVD dashboards, sending data to BVD, and connecting the data to dashboard widgets.

You could also look through the BVD Getting Started dashboards, which are based on the OMi sample dashboard and guide you through the process of modifying a Visio file, uploading the exported SVG file to BVD, and then sending data to the dashboard.

1. Run the data generator `samplegen.bat` after extracting it from `samplegen.zip`. When prompted, enter the URL of your data receiver, your API key, and the HTTP proxy and port, if required.

   > **Example**
   >
   > ```
   > Please enter the data receiver URI, e.g.
   > https://receiver.coloryourdata.io/: https://bvd.example.com/bvd-receiver
   > Please enter your BVD API Key: 47a648e9065d465012e541288b5a345e
   > Host name of your HTTP proxy server (leave empty if not required):
   > httpproxy.example.com
   > Port name of your HTTP proxy server: 8080
   > Sending sample data to ...
   > ```

   > **Note:** If you send data to BVD from an application that is **not** part of the suite container deployment (for example a classically installed OMi), define the receiver URL as follows:
   >
   > `https://<external_access_host>/bvd-receiver/api/submit/<API key>`
   >
   > If you send data to BVD from an application that is also installed as a suite container, define the receiver URL as follows:
   >
   > `http://bvd-receiver.<namespace>.svc.cluster.local:4000/bvd-receiver/api/submit/<API_key>`
   >
   > `<namespace>` is the namespace assigned to your suite deployment. You can check the namespace by accessing **SUITE > Management** in the Management Portal.

   The batch file generates random values and sends them every two seconds to the sample dashboard.

2. View the OMi sample dashboard in 🗠 **Dashboards > omi_sample**. Take a moment to explore the information displayed and watch it updating as new data arrives.
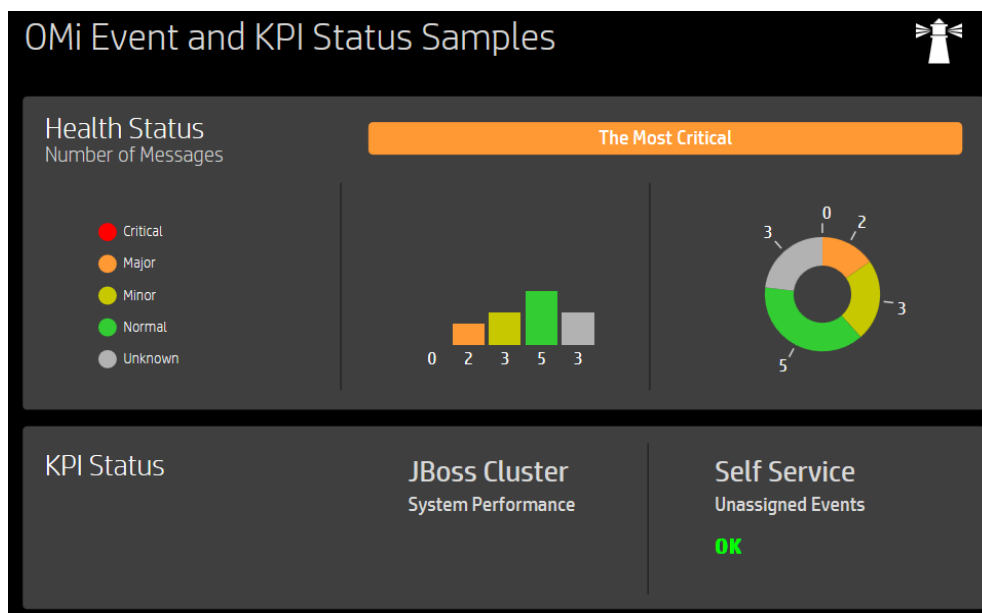
3. Configure the `omi_sample` dashboard in 🔧 **Administration > Manage Dashboards > omi_ sample**. Take a moment to look at the properties of the dashboard widgets, the data channels, fields, and rules.

    The following steps demonstrate to you how BVD maps data to widgets:

    a. Edit the text value widget below **JBoss Cluster System Performance**, and delete the existing data channel (🏷️omi🏷️kpi📦OprSample📦IT JBoss Cluster📦System Performance).

    b. Place the cursor in the **Data Channel** field and type `omi`. The editor displays a list of all data channels that match the search string `omi`. Select 🏷️omi🏷️kpi📦OprSample📦IT JBoss Cluster📦System Performance again.

    c. In the **Status Field**, click the down arrow and select `status` from the drop-down list. This step configures the text value widget to retrieve the data from the `status` field in the connected data channel.

    d. Click **Cancel** to return to the **Manage Dashboards** page.

    For more information on sending data to BVD, see "Example: Sending JSON Data to BVD" on page 27.

Step 3: Send Your Own Data to the Dashboard

Before creating your first BVD dashboard, think about the data you want to send and show. You may have numbers indicating the currently logged-in users of your online banking system, metrics indicating the response time of your web servers, or measurements of the temperature in your data centers.

BVD expects to receive your data as HTTP post requests in JavaScript Object Notation (JSON) format.

The URL should look something like this:

`https://<external_access_host>/bvd-receiver/api/submit/<API_key>/dims/<dims>` (if the application sending the data is installed classically)

`http://bvd-receiver.<namespace>.svc.cluster.local:4000/bvd-receiver/api/submit/<API_key>` (if the application sending the data is also installed as a suite container)

`<external_access_host>`

> The fully qualified domain name of the host which you specified as `EXTERNAL_ACCESS_HOST` in the `install.properties` file during the Container Deployment Foundation installation. Usually, this is the master node's FQDN.

`<namespace>`

> The namespace assigned to your suite deployment. You can check the namespace by accessing **SUITE > Management** in the Management Portal.

`<API_key>`

> Identifies your BVD instance. You can find the API key in 🔧 **Administration > System Settings**.

`<dims>`

> The names in your JSON name-value pairs. Select and combine dims that uniquely identify your data.

For more information on the URL of the BVD data receiver, see "Sending dimensions and tags in the receiver URL" on page 23.

Example URL:

```
http://bvd.example.com/bvd-
receiver/api/submit/47a648e9065d465012e541288b5a345e/dims/viewName,ciName,kp
iName/tags/omi,kpi
```

Example data channel in BVD:

Tags: omi, kpi

Dims: OprSample (=viewName), Employee Self Service (=CiName), Unassigned Events
(=kpiName)

Step 4: Design Your Own Dashboard in Visio

1. In Visio, create a new drawing. Drag and drop the BVD shapes to your drawing and then arrange and modify them as required.

   HPE recommends that you do not change the shape data of the BVD shapes in Visio itself. BVD enables you to adapt the widgets in the uploaded dashboard, which is faster and more convenient (🔧 **Administration > Manage Dashboards**).

   For an overview of the available widgets and the associated properties, see "Widgets" on page 52.

2. Save your drawing as an SVG file, making sure that the following Visio settings are selected:

   ○ Save as type: **Scalable Vector Graphics (*.svg)**

   ○ Select: **Include Visio data in the files**

   ○ Tip: Press **Ctrl+A** to select everything in the drawing. This ensures that your entire drawing is exported and not the currently selected element only.

   Alternatively, click the **Export Dashboard** button in the **Dashboard** ribbon, if you have installed the BVD **Visio Add-in**.

Step 5: Upload your Dashboard to BVD

1. In BVD, open 🔧 **Administration > Manage Dashboards** and click ➕ **Add**, select your SVG file, and then click **Upload Dashboard** to import it. The BVD dashboard editor opens and displays your uploaded dashboard.

2. You can change the properties of the dashboard itself (for example, the SVG file associated with the dashboard, the title, or the background color).

   To edit the properties of a widget, click the widget. When you click the **Data Channel** field, a drop-down list opens showing all data streams that have been received by BVD. The data you sent in Step 3 should be included in the list. Select it, adjust any of the other properties as required, and click **Save**. For more information about the widget properties, see "Widget Properties" on page 72.

3. By default, newly imported dashboards are visible in the **Dashboards** menu. To show or hide a dashboard, click the 📈 **Show in Menu/Hide from Menu** button.

4. View your dashboard by selecting it in the 📈 **Dashboards** menu. Then watch your dashboard updating as you send data over your data channels.

Integrate

BVD provides out-of-the-box integrations with HPE Software products such as HPE Operations Manager i (OMi) and HPE Operations Connector (OpsCx).

However, you can create your own integrations for any data source by writing an adapter for BVD. The adapter must convert the source data to JSON and send the JSON-enabled data to the BVD data receiver. Predefined dashboards, which are designed and configured to display the received data, then complement your integration.

This section includes:

Integrate with OMi

You can configure OMi 10.11 or later versions to send Event Status, KPI Status and Performance Dashboard data to BVD. Use the BVD Data Forwarding rules manager on the OMi server to specify which data you want to forward.

Support matrix

For a list of supported OMi versions including required hotfixes, see the support matrix:

Support Matrices for Operations Center products

Download and extract the support matrix files, open the document SUMA.htm and select **Operations Manager i Business Value Dashboard** from the product list.

Data forwarding

The following OMi data can be forwarded to BVD:

Event Status data: The event status data to be forwarded is collected from an OMi monitoring dashboard that you specify.

KPI Status data: The KPI status is collected from all CIs that are associated with a view and the KPI set that you specify. If you do not specify a KPI set, all KPIs of the chosen view are forwarded.

Performance Dashboard data: The performance dashboard data is collected from your public favorites in OMi. To forward Performance Dashboard data, save your performance dashboard charts as favorites with the **Share as Public** option enabled before including this data in a rule. For details, see "Enable Performance Dashboard Data Forwarding on the OMi Server" on the next page.

For information on how to forward this data to BVD, see "Forward OMi Data to BVD" on the next page.

This section includes:

- "BVD Data Channels for OMi Data" below

- "Enable Performance Dashboard Data Forwarding on the OMi Server" on the next page

- "Forward OMi Data to BVD" on the next page

BVD Data Channels for OMi Data

When you send BVD data to OMi by using forwarding rules, BVD creates data channels that uniquely identify the data you are sending.

Each data channel consists of tags and dimensions (dims). Tags are static labels and dimensions are names that are associated with a specific value. For more information, see "Create custom integrations" on page 22.

The data channels are structured differently depending on the data you choose to forward:

Event Status

```
<tags connected server><tags forwarding rule><dim monitoring dashboard><dim
widget label><dim widget type>
```

KPI Status

```
<tags connected server><tags forwarding rule><dim view name><dim CI name><dim
KPI name>
```

Performance Dashboard

```
<tags connected server><tags forwarding
rule><metricName><instanceName><dSName><systemName><className>
```

`<tags connected server>` are all tags that are specified when adding a BVD Connected Server. `<tags forwarding rule>` are all tags that are specified when creating a BVD Data Forwarding Rule.

Enable Performance Dashboard Data Forwarding on the OMi Server

To enable performance dashboard data forwarding to BVD, follow these steps:

1. In the Performance Perspective, select the charts that you want to forward and save them as favorites:

   a. In OMi, access **Workspaces > Operations Console > Performance Perspective**.

   b. In the **View Explorer**, select a view and then the CI for which you want enable data forwarding.

   c. In the **Performance** pane, choose a performance dashboard from the drop-down list.

   d. Click the title of the chart you want to save as favorite and click **Add to Favorite**.

   e. Choose to add the favorite to the default page, user-defined favorite page, or create a new user-defined favorite page. Click **Save**.

   f. Open the favorite in the **Performance** pane, access the menu and click **Save**. Check the **Share as Public** option and click **Save**.

Forward OMi Data to BVD

> **Important:** Before you create BVD data forwarding rules in OMi, access

Administration > Setup and Maintenance > Connected Servers to add your BVD instance as a Connected Server. For details on which information to enter, see the *OMi Help*.

To forward OMi data to BVD, access **Administration > Setup and Maintenance > BVD Data Forwarding**.

1. In the right pane, click **Create**. Alternatively, click ✳ **New**.

2. In the **General** section, complete the following information:

   a. Enter a display name and (optional) a description for the forwarding rule.

   b. *Optional*. Enter a comma-separated list of tags.

      Data channels can be tagged to separate data from incoming streams and to create more specific data channels. For example, if you have separate OMi servers for different regions and you want separate dashboards for each region, you can add a tag that identifies the region in which this OMi server is located.

      The tags you enter will be added to the data channel after the tags specified for the BVD Connected Server. For details, see "BVD Data Channels for OMi Data" on page 15.

3. In the **Target BVD Server** section, choose one or multiple Connected Servers from the drop-down list. This list shows all Connected Servers of the type BVD and Alias. If your BVD connection is not listed in the drop-down, access **Administration > Setup and Maintenance > Connected Servers** to ensure that the server connection was set up correctly.

4. *Optional*. In the **Event Status** section, choose one or multiple monitoring dashboards from which you want to forward data to BVD.

   > **Caution:** If the monitoring dashboard name is changed, the data channel is **not** updated. Instead, a new data channel with the changed monitoring dashboard name is created. Widgets that use the old data channel in BVD will not receive data from OMi anymore and need to be updated to the new data channel.

5. *Optional*. In the **KPI Status** section, choose the one or multiple views from which you want to forward KPI status data. Click 🔧 next to the view name to choose specific KPIs. If no individual KPIs are selected, KPI status data will be forwarded for all CIs that are associated with the chosen view.

6. *Optional*. In the **Performance Dashboard Data** section, choose one or multiple public favorites of your performance dashboards for which you want to forward data to BVD. For information on how to add performance dashboards as favorites, see "Enable Performance Dashboard Data Forwarding on the OMi Server" on the previous page.

7. *Optional*. Clear the check box **Activate after save** if you want the status of the rule to be inactive after clicking **Save**. You can activate the rule at a later point in time.

8. Click **Save** to save the BVD data forwarding rule.

Integrate with OpsCx

To forward performance data from Operations Connector (OpsCx) to BVD, create a Data Forwarding policy in Operations Connector. The policy must specify your BVD instance as a target and contains rules that specify which data is sent to BVD.

Learn More

Support matrix

For a list of supported Operations Connector versions, see the support matrix:

Support Matrices for Operations Center products

Download and extract the support matrix files, open the document SUMA.htm and select **Operations Manager i Business Value Dashboard** from the product list.

Tasks

This section includes:

- "Configuring data forwarding in Operations Connector" below

- "Forwarding data using an HTTP proxy" on page 21

- "Forwarding data using HTTPS" on page 21

Configuring data forwarding in Operations Connector

1. In BVD, open **Administration > System Settings**, and copy the **API Key**.

This key identifies your BVD instance and must be included in the data submitted by the data senders.

> **Note:** The System Settings page is only available to users with administrator privileges.

2. In Operations Connector, click ✳ **Create** in the toolbar. Then click **Forwarding >** 📊 **Data Forwarding**.

3. In the **Properties** page, define information that is related to the policy itself (for example, the name and description of the policy).

4. In the **Targets** page, set up your BVD instance as a target to which the data is forwarded:

    a. In the target list, click ✳ **New Item** above the **Name** column to add a new target.

    b. Enter the name of the BVD target and a description.

    c. Enter the URL of your BVD data receiver. The URL must include your API key and dimensions or tags (or both dimensions and tags) to create unique data channels for your metrics data. BVD expects to receive your data as HTTP post requests in JavaScript Object Notation (JSON) format.

       The URL should look something like this:

       `https://<external_access_host>:<port>/bvd-receiver/api/submit/<API_key>/dims/<dims>`

       `<external_access_host>`

       The fully qualified domain name of the host which you specified as `EXTERNAL_ACCESS_HOST` in the `install.properties` file during the Container Deployment Foundation installation. Usually, this is the master node's FQDN.

       `<port>`

       The port must be specified as part of the URL, even for the standard HTTP or HTTPS port. Default: 80 (HTTP) or 443 (HTTPS)

       > **Example**: If your target is `https://bvd1.example.net/bvd-receiver/api/submit/...`, add the standard HTTPS port to the URL:
       > `https://bvd1.example.net:443/bvd-receiver/api/submit/...`

       `<API_key>`

       Identifies your BVD instance. You can find the API key in 🔧 **Administration > System Settings**.

<dims>

The names in your JSON name-value pairs. Select and combine dims that uniquely identify your data.

For more information on the URL of the BVD data receiver, see "Sending dimensions and tags in the receiver URL" on page 23.

> **Example**
>
> You can add the following metrics attributes as dimensions to the URL:
>
> `dims/dataDomain,relatedCi,name,metricClass,node`
>
> This results in the following data channel:
>
> ⬡HPBsmIntOEM ⬡oem.example.com/oracle_database/oemdb
>
> ⬡Average Active Sessions ⬡Throughput ⬡oem.example.com

For information on the metrics attributes, see the Operations Connector documentation.

d. Select **JSON** as **Wire format**.

e. Make sure **Use Guaranteed Delivery** is cleared. If you select this check box, Operations Connector attempts to resend data that could not be delivered earlier and BVD may receive outdated data.

5. Configure data forwarding rules:

○ To forward metrics data after policy rules are applied, select the **Metric** page and configure metric data forwarding rules.

Metric forwarding rules define what a Data Forwarding policy should do in response to specific metric data. Each rule consists of a condition and of settings for the data generated by the policy. The settings enable you to configure what data Operations Connector forwards to which target.

○ To forward structured input data, that is, data gathered by a policy before the policy rules are applied, select the **Structured Input** page and configure structured input data forwarding rules.

Structured input data rules define what a Data Forwarding policy should do in response to specific structured input data. Each rule consists of a condition and of settings for the data generated by the policy. The settings enable you to configure what data Operations Connector forwards to which target.

Discarding rules have a higher precedence than forwarding rules.

6. Click **Save and Close** to save the policy and close the editor.

7. Download the root CA certificates from the BVD CA and import them to the Operations Connector trusted root certificate store. For details, see "Forwarding data using HTTPS" below.

8. Activate the Data Forwarding policy in Operations Connector. Operations Connector starts sending data to BVD.

   In BVD, connect your widgets to the data channels opened up by Operations Connector.

Forwarding data using an HTTP proxy

If you must redirect the connection from Operations Connector to BVD through an HTTP proxy, configure the Operations Connector system as follows:

1. Edit the XPL configuration file:

   ```
   ovconfchg -edit
   ```

2. Add the following lines:

   ```
   [bbc.http]
   PROXY=<proxy_hostname>:<proxy_port>+(<included hosts>)-(<excluded hosts>)
   ```

   Replace `<included_hosts>` with a comma-separated list of hostnames or IP addresses to which the proxy enables communication. Replace `<excluded hosts>` with a comma-separated list of hostnames or IP addresses to which the proxy cannot connect. Asterisks (*) are wild cards in hostnames and IP addresses. Both `<included_hosts>` and `<excluded hosts>` are optional.

   To specify multiple proxies, separate each proxy with a semicolon (;). The first suitable proxy in the list takes precedence.

   > **Example**
   >
   > ```
   > [bbc.http]
   > PROXY=myproxy.example.com:8888+(mybvd.example.com)
   > ```

3. Restart the Operations Connector process that forwards the data:

   ```
   ovc -restart opcgeni
   ```

Forwarding data using HTTPS

By default, Operations Connector only trusts the OMi server as certificate authority (CA). For Operations Connector to trust the CA that issued the BVD certificate, you must download the root CA certificates from the BVD CA and import them to Operations Connector trusted root certificate store.

1. *Only for SaaS deployments*. Download the following certificates from https://certs.godaddy.com/repository/:

   ○ GoDaddy Class 2 Certification Authority Root Certificate

   ○ GoDaddy Class 2 Certification Authority Root Certificate - G2

   ○ GoDaddy Secure Server Certificate (Intermediate Certificate) - G2

2. *Only for container or onPrem deployments*. Obtain the root CA certificates from the root and any intermediate authorities that issued the BVD server certificate.

3. On the Operations Connector system, import the certificates to the trusted root certificate store:

   ```
   ovcert –importtrusted –file <certificate_file>
   ```

4. *Optional.* On the Operations Connector system, run the following command to verify that the trust has been established:

   ```
   bbcutil -ping https://<BVD_server>:<port>
   ```

   > **Example**
   >
   > ```
   > bbcutil -ping https://bvdserver.example.com:12225
   > ```

   The following output indicates that the HTTPS connection is successful.

   ```
   https://<BVD_server>:<port>:
   (bbc-288) status=eServiceError coreID= bbcV= appN= appV= conn=0 time=471 ms
   ```

   `eServiceError` is normal and indicates that there is no BBC service on the BVD system. If the output includes eSSLError (or similar), the HTTPS connection was not successful, possibly because the trust has not been established correctly.

Create custom integrations

BVD expects to receive your data as HTTP post requests in JavaScript Object Notation (JSON) format.

HPE recommends that your JSON input contains flat data, consisting of name-value pairs. If you must send nested data, BVD automatically flattens the data (see also "Nested JSON data" on page 26). You can also send JSON data in arrays. This enables you to send multiple data objects in a single web service call (see also "JSON data arrays" on page 25).

See also "Example: Sending JSON Data to BVD" on page 27.

Learn More

This section includes:

Sending dimensions and tags in the receiver URL

The BVD receiver URL should look something like this:

- URL with dimensions only:

  ```
  https://<external_access_host>/bvd-receiver/api/submit/<API key>/dims/<dims>
  [,<dims=value>]
  ```

- URL with tags only:

  ```
  https://<external_access_host>/bvd-receiver/api/submit/<API key>/tags/<tags>
  ```

- URL with both dimensions and tags:

  ```
  https://<external_access_host>/bvd-receiver/api/submit/<API key>/dims/<dims>
  [,<dims=value>]/tags/<tags>
  ```

  The tags can also precede the dims:

  ```
  https://<external_access_host>/bvd-receiver/api/submit/<API
  key>/tags/<tags>/dims/<dims>[,<dims=value>]
  ```

If the application sending the data is also installed as a suite container, define the receiver URL as follows:

```
http://bvd-receiver.<namespace>.svc.cluster.local:4000/bvd-
receiver/api/submit/<API_key>
```

```
<external_access_host>
```

The fully qualified domain name of the host which you specified as `EXTERNAL_ACCESS_HOST` in the `install.properties` file during the Container Deployment Foundation installation. Usually, this is the master node's FQDN.

`<namespace>`

The namespace assigned to your suite deployment. You can check the namespace by accessing **SUITE > Management** in the Management Portal.

`<API_key>`

Identifies your BVD instance. You can find the API key in 🔧 **Administration > System Settings**.

`<tags>`

Static labels that you can attach to your data to create more specific data channels.

`<dims>`

The names in your JSON name-value pairs. Select and combine dimensions (dims) that uniquely identify your data.

`<dims=value>`

The names and values in your JSON name-value pairs. Directly assign values with names to improve data identification. Use this option, for example, if you have separate servers of the same data source for different locations and you want separate dashboards for each location. These name-value pairs do not have to be part of the JSON input. If they are, the values in the JSON input will be overwritten by the values in the URL.

Sending dims and tags as HTTP parameters

You can also submit the dims and tags as HTTP parameters of the URL.

> **Example**
>
> ```
> https://bvd.example.com/bvd-
> receiver/api/submit/47a648e9065d465012e541288b5a345e?dims=viewName,ciName,kpi
> Name,location=nyc&tags=omi,kpi
> ```

Sending dims and tags in the receiver URL and as HTTP parameters

You can combine the receiver URL and HTTP parameters to send dims and tags. Define the dims and tags as part of the URL path first, then add additional dims and tags as HTTP parameters.

Example

```
https://bvd.example.com/bvd-
receiver/api/submit/47a648e9065d465012e541288b5a345e/dims/viewName,ciName=abc
/tags/omi,kpi?dims=kpiName,location=nyc&tags=bvd
```

However, if the same dimension or tag is specified more than once, the value of the last query parameter overwrites the values of the previous parameters. The value of the last query parameter appears multiple times as data channel.

Example

```
https://bvd.example.com/bvd-
receiver/api/submit/47a648e9065d465012e541288b5a345e/dims/location=boston?dim
s=location=nyc&tags=east&dims=location=atlanta
```

In this example, the dim `location` will have the value `atlanta`. Because dimensions are accumulated, the value `atlanta` appears three times as data channel.

JSON data arrays

You can submit multiple JSON objects in a single web service call by adding them to an array.

Array:

```
[
   {
      a: 1,
      b: 2
   },
   {
      c: 3,
      d: 4
   }
]
```

Nested JSON data

If the input contains nested data, BVD automatically flattens the data by renaming nested name-value pairs to include the names of the parent elements, separated by slashes (/), for example:

Nested JSON data:

```
{
    a: 1,
    b: 2,
    c: {
        x: 6,
        y: 7
    }
}
```

**Flattened JSON data:**

```
{
    a: 1,
    b: 2,
    c/x: 6,
    c/y: 7
}
```

Data storage

BVD stores only a specific number of data records per channel. The records are only kept if they are related to a widget. The BVD data aging process scans the database every 30 minutes to identify and automatically delete records that exceed the configured maximum age.

Example: Sending JSON Data to BVD

Sending Data From Data Center East
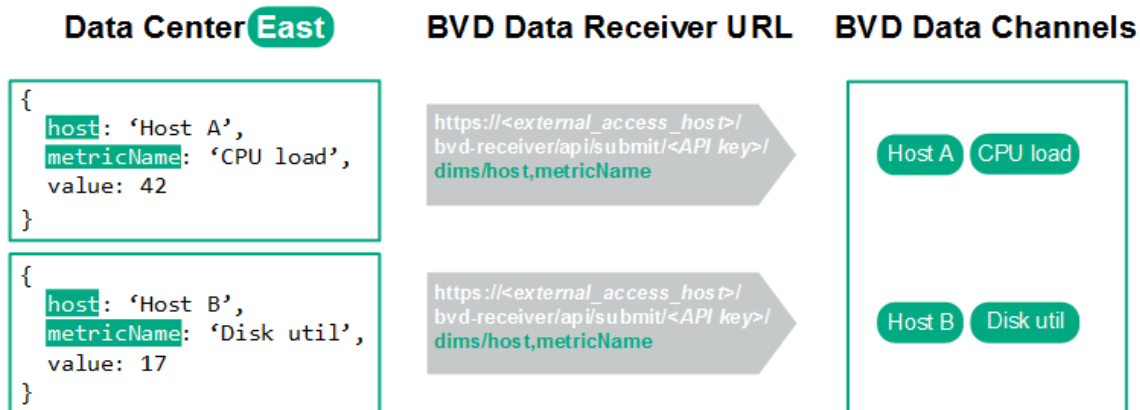
In this example, Data Center East sends two sets of JSON data to the BVD receiver. In both sets, the data fields `host` and `metricName` uniquely identify the value. The fields are therefore selected as dimensions (dims) and included in the URL. Once received by the BVD server, the JSON data creates two data channels:

⊡Host A ⊡CPU load and ⊡Host B ⊡Disk util.

Lessons learned: Pick the fields in your data that uniquely identify the values you want to send to BVD and include the fields as dimensions in the HTTP post request.

Note: If you send data to BVD from an application that is **not** part of the suite container deployment (for example a classically installed OMi), define the receiver URL as follows:

`https://<external_access_host>/bvd-receiver/api/submit/<API key>`
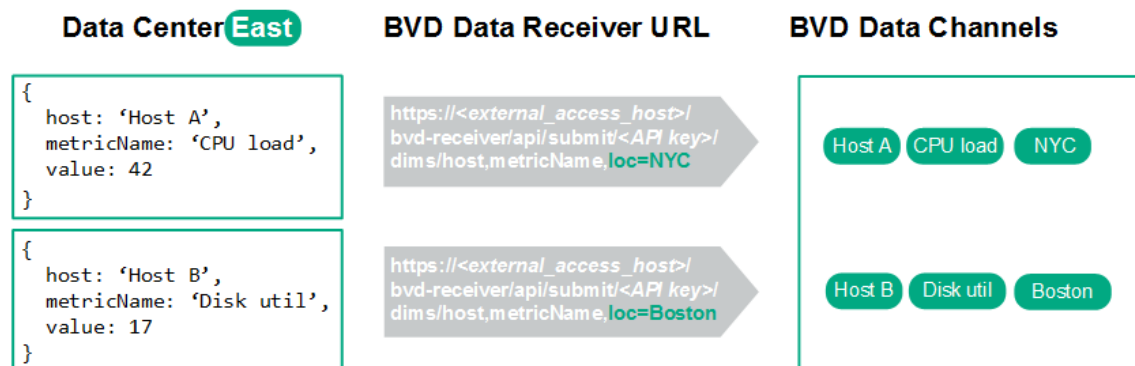
If you send data to BVD from an application that is also installed as a suite container, define the receiver URL as follows:

`http://bvd-receiver.<namespace>.svc.cluster.local:4000/bvd-receiver/api/submit/<API_key>`

`<namespace>` is the namespace assigned to your suite deployment. You can check the namespace by accessing **SUITE > Management** in the Management Portal.

Sending Additional Data From Data Center East

Data Center East

```
{
  host: 'Host A',
  metricName: 'CPU load',
  value: 42
}
```

```
{
  host: 'Host B',
  metricName: 'Disk util',
  value: 17
}
```

BVD Data Receiver URL

```
https://<external_access_host>/
bvd-receiver/api/submit/<API key>/
dims/host,metricName,loc=NYC
```

```
https://<external_access_host>/
bvd-receiver/api/submit/<API key>/
dims/host,metricName,loc=Boston
```

BVD Data Channels

Host A   CPU load   NYC

Host B   Disk util   Boston

The primary location of Data Center East is in New York City, with backup servers located in Boston. Both locations send the same set of JSON data. To differentiate the data from the two locations without modifying the JSON data, you can add an additional dimension `loc` with the corresponding value to the URL. The modified URL updates the data channels to
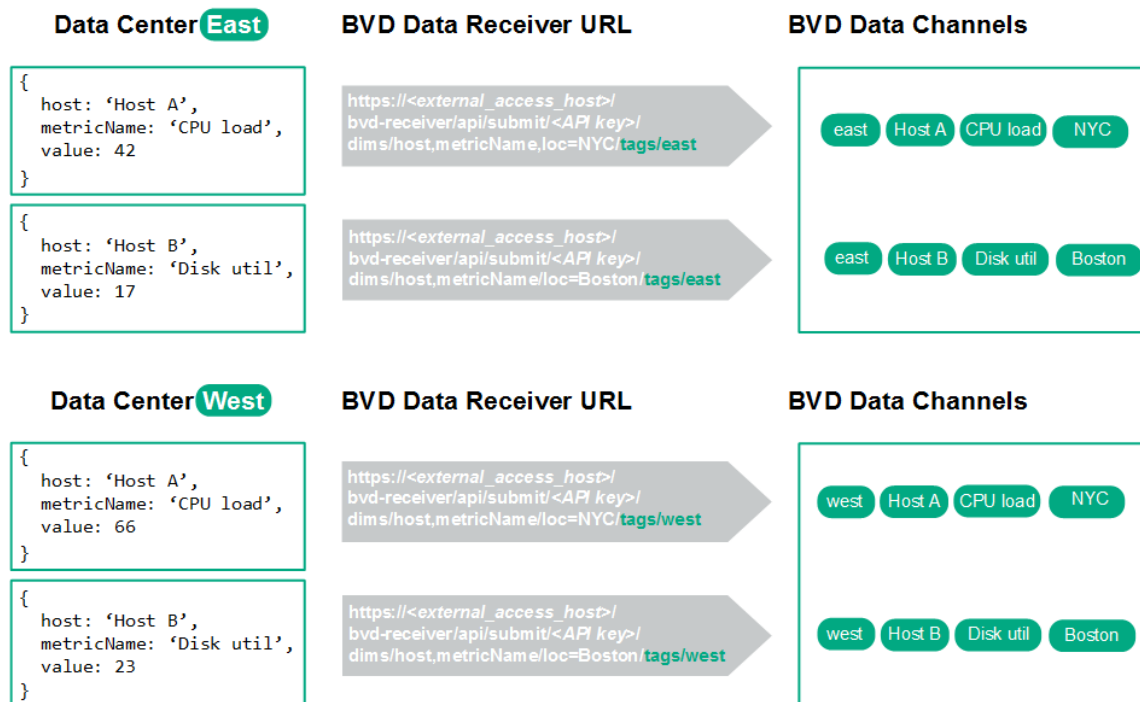
Host A CPU load NYC and Host B Disk util Boston.

In this example, we added the dimension `loc` to the URL.

Lessons learned: Directly assign values to your dimensions by adding dim=value pairs to the HTTP post request.

Sending Data From Data Center West

Data Center East

```
{
  host: 'Host A',
  metricName: 'CPU load',
  value: 42
}
```

```
{
  host: 'Host B',
  metricName: 'Disk util',
  value: 17
}
```

BVD Data Receiver URL

https://<external_access_host>/
bvd-receiver/api/submit/<API key>/
dims/host,metricName,loc=NYC/**tags/east**

https://<external_access_host>/
bvd-receiver/api/submit/<API key>/
dims/host,metricName/loc=Boston/**tags/east**

BVD Data Channels

east Host A CPU load NYC

east Host B Disk util Boston

Data Center West

```
{
  host: 'Host A',
  metricName: 'CPU load',
  value: 66
}
```

```
{
  host: 'Host B',
  metricName: 'Disk util',
  value: 23
}
```

BVD Data Receiver URL

https://<external_access_host>/
bvd-receiver/api/submit/<API key>/
dims/host,metricName,loc=NYC/**tags/west**

https://<external_access_host>/
bvd-receiver/api/submit/<API key>/
dims/host,metricName/loc=Boston/**tags/west**

BVD Data Channels

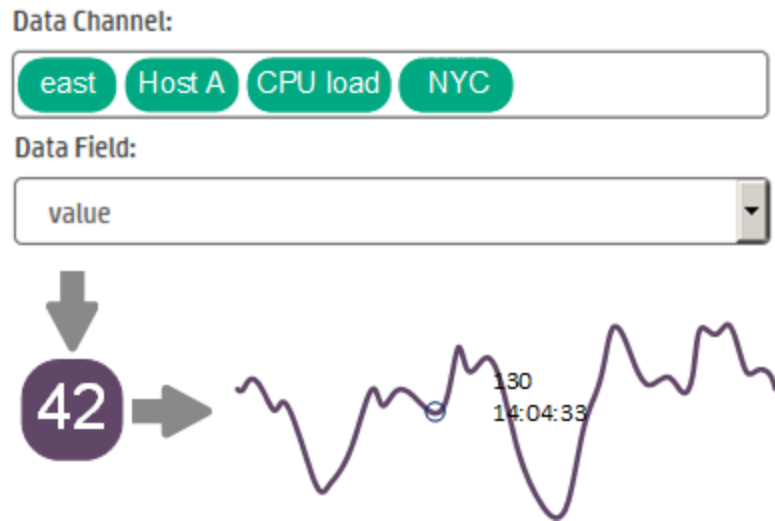west Host A CPU load NYC

west Host B Disk util Boston

A second data center, Data Center West, starts sending data similar to the JSON data sent by Data Center East. The data from Data Center West uses the same data channels as the data from East. To distinguish the data from the two centers, you must add the origin to the data. You can do this by adding tags to the URL. Tags are static labels that you can attach to your data to create more specific data channels.

In this example, we added the tags east and west to the URL. The tags precede the dims in the data channels.

Lessons learned: Attach tags to your data to create specific data channels.

Associating Data Channels with Widgets

Once BVD has received the data, it creates the corresponding data channels. You can then associate a data channel with your widget in the widget's properties. In this example, the data channel ⬦east ⬡Host A ⬡CPU load ⬡NYC has been selected for the sparkline widget.

By default, the widget consumes data from the `value` data field. In this example, the current value is 42. If the field that holds the values you are interested in has a different name (for example, metricVal), select that name in the Data Field property of the widget.

Lessons learned: Connect your data to a widget by selecting the corresponding data channel in the widget's properties.

Administer

This section describes the administrator tasks that you can perform in BVD.

Modify the system settings

The System Settings page displays your API key and enables you to change the look and feel of your dashboards and BVD using a custom style sheet.

To access

🔧 **Administration > System Settings**

Learn More

API Key

The API key connects your BVD data to your BVD instance. The key must be included in all of you data streams and ensures that your data arrives in your dashboards.

You can request a new API key, for example if your old key has been compromised. To request a new key, click **Request New API Key** in the System Settings page. After you confirm your decision, BVD displays your new key immediately. Remember to update your data senders to include your new API key.

API Key

44c355f0b6ef11e6bb615bbf36c84ab0

Requesting a new API key will invalid your existing one. In consequence you have to adjust all your data senders to use this new API key instead of the old one!

Request new API key

Style Customization

You can change the look and feel of BVD, for example, the color of the masthead. To do so, inspect the HTML element that you want to change, type the new style in the **Style customization** field and click **Upload changed style**.

To reset the style to the default, clear the contents in the **Style customization** field and click **Upload changed style**.

See also "Use Custom Fonts in Your Dashboards" below for detailed instructions on how to choose a web font other than Google Fonts for your dashboards.

Do It Yourself

Use Custom Fonts in Your Dashboards

BVD dashboards by default use fonts served by Google at http://www.google.com/fonts. To use custom fonts for your dashboards, do the following:

1. Use your preferred fonts in the Visio drawings of your dashboards, open the exported SVG file in a text editor, and then search for the string `font`. In SVG files, fonts are declared in CSS definitions similar to the following:

```
<style type="text/css">
<![CDATA[
    .st1 {fill:none;stroke:none;stroke-linecap:round;stroke-linejoin:round;stroke-width:0.75}
    .st2 {fill:#000000;font-family:Arial;font-size:3em}
    .st3 {fill:#000000;font-family:Courier New;font-size:3em}
    .st4 {fill:#000000;font-family:Metric;font-size:3em}
    .st5 {font-size:1em}
    .st6 {fill:none;fill-rule:evenodd;font-size:12px;overflow:visible;stroke-linecap:square;stroke-miterlimit:3}
]]>
</style>
```

   Take note of the font family name, for example `Metric`.

2. Make sure your custom fonts are installed on a publicly accessible web server.

   The web server must allow access to the font resources from other sites. To do so, it must include a properly configured `Access-Control-Allow-Origin` header in each reply to a font request.

3. Create a custom CSS font definition in BVD:

   Open 🔧 **Administration > System Settings** and type your CSS definition in the **Style customization** field, then click **Upload changed style**.

   > **Example**
   >
   > ```
   > @font-face {
   >     font-family: "Metric";
   >     src: url("https://www.example.com/fonts/metric.woff") format("woff");
   > }
   > ```

Manage dashboards

The **Manage Dashboards** page lists all available dashboards, and enables you to upload, delete, download, edit, or show or hide dashboards in the dashboards menu.

To access

⚒ **Administration > Manage Dashboards**

Learn More

This section includes:

- "Dashboard Icons" on the next page

- "Uploading Dashboards" on the next page

- "Downloading Dashboards" on page 35

- "Categorizing Dashboards" on page 35

- "Changing the Title of a Dashboard" on page 36

- "Selecting Widgets for Editing" on page 37

- "Using Dashboards as Templates" on page 38

Dashboard Icons

Use these buttons to manage your dashboards as follows:

	Upload a dashboard from an SVG file. The maximum dashboard size is 20 MB.
🗑	Delete a dashboard from BVD.
⬇	Download a dashboard to an SVG file.

> **Note:** The ⬇ **Download** button is not available when working with BVD on mobile devices.

✏	Edit a dashboard in BVD.
📈	Show or hide a dashboard in the 📈 **Dashboards** menu. This action does not impact permissions to view dashboards granted through roles. To change view permissions, see "Manage users" on page 43.

> **Tip:** On mobile devices, click the buttons twice to trigger their actions.

The following icons notify you of missing or incorrect configurations when editing a dashboard:

⚠	Not all required fields of the widget have been completed; for example, the data channel has not been selected yet. You can save dashboards that have warnings.
❌	The widget configuration contains errors (for example, incorrect color values, incorrect HTTP protocol in web page widgets, and so on). You must first resolve the errors before the dashboard can be saved.
✳	The widget has unsaved changes.

Uploading Dashboards

You can upload an SVG file as often as you want. Edit the dashboard, and in the dashboard properties click ⬆ **Replace Dashboard**. The maximum dashboard size is 20 MB. If the file already exists, BVD replaces the previous version of the file with the newer version and opens the associated dashboard for you to edit. The newer version does not overwrite existing dashboard properties or widgets, only new widgets are added.

Deleting widgets from an SVG file only deletes the widgets from the dashboard. The configuration of the deleted widgets remains in the database so that you can easily re-upload a previous version of an SVG file to reactivate the previously deleted widgets.

Downloading Dashboards

When you upload an SVG file created from Visio, the SVG file usually does not have any data channels configured. This task is best done in the BVD Manage Dashboards UI.

To obtain a version of your SVG files with the data channels configured, you can download the dashboards from BVD by using the ⬇ **Download** button in the Manage Dashboards UI. The download creates an SVG file with the original name.
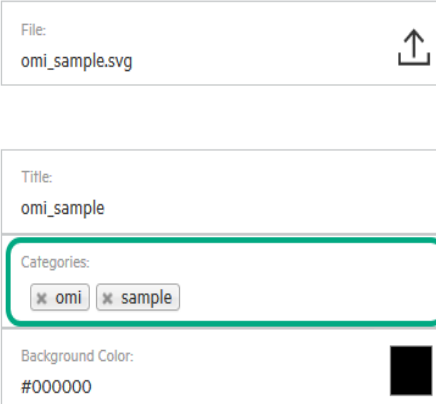
You can then transfer this file to another BVD instance and upload it there, or store it in a safe location for backup purposes.

> **Note:** The downloaded SVG file does not include the assigned categories.

Categorizing Dashboards

To be able to grant dashboard permissions to a user, a dashboard must have one or more categories assigned. Categories are arbitrary text strings that you enter in the Categories field in the dashboard properties. Once entered, a category can be selected and added to any dashboard.
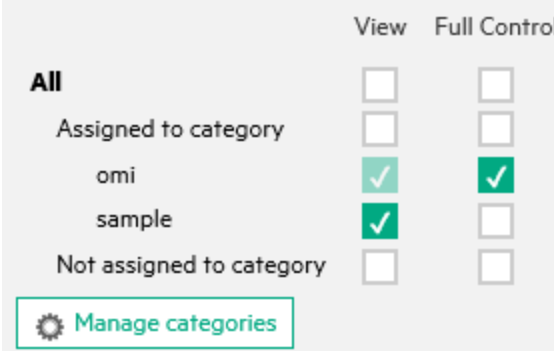
Dashboard Properties

File:
omi_sample.svg ⬆

Title:
omi_sample

Categories:
× omi   × sample

Background Color:
#000000

When you create a user role, you can assign view or full control permissions to a dashboard category. This restricts a user's access to dashboards of the specified category. See also "How to Set Permissions" on page 49.

	View	Full Control
All	☐	☐
Assigned to category	☐	☐
omi	☑	☑
sample	☑	☐
Not assigned to category	☐	☐

⚙ Manage categories

Changing the Title of a Dashboard

When you upload an SVG file, the file name automatically becomes the title of the dashboard. This behavior can make the 📈 **Dashboards** menu look very technical:

Consider changing the title of your dashboards in the dashboard's properties:

🔧 **Administration > Manage Dashboards >** ✏️ **Edit**

In the **Title** field, enter a descriptive name.

> **Note:** The following characters are not allowed in the dashboard title: ! [ ] / \ = + ; ? * % &

Dashboards ✕

Filter Dashboards 🔍

🏠 2-AO_world

🏠 omi_sample

Dashboard Properties

File:
omi_sample.svg ⬆️

Title:
omi_sample

Categories:
✕ omi   ✕ sample
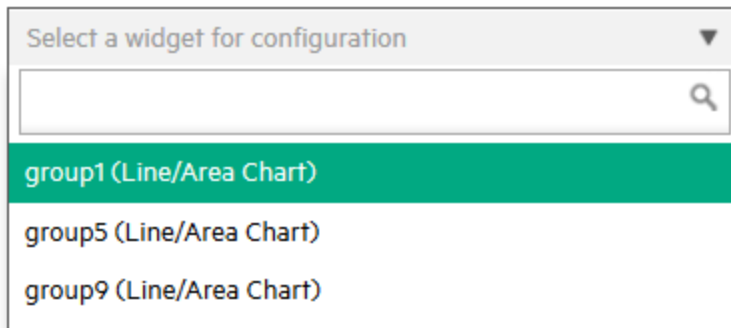
Background Color:
#000000 ⬛

Selecting Widgets for Editing

To edit the properties of a widget, edit the dashboard that contains the widget and then click the widget. The properties are displayed to the right of the dashboard for you to edit. When done, click **Apply** or **Save**.

If your dashboard contains widgets that are stacked on top of each other so that you cannot access the ones below, do one of the following:

- **Two widgets.** If there are only two widgets stacked on top of each other, click the top one once and once again. The second click selects the one below.

- **More than two widgets.** If you have a stack of more than two widgets, go the dashboard properties, and under **Widgets** click **Select a widget for configuration** to open a list of available widgets:

Widgets

Select a widget for configuration ▼
🔍
group1 (Line/Area Chart)
group5 (Line/Area Chart)
group9 (Line/Area Chart)

Select the widget you want to edit to access the widget's properties.

Using Dashboards as Templates

Use existing dashboards as templates in order to create multiple instances that use the same widgets but different data.

Templates consist of the original dashboard and a number of individual instances, which can be viewed in the ⤢ **Dashboards** menu.

Follow these steps to convert a dashboard into a template:

1. Access the properties of the dashboard you want to use as a template:

    🔧 **Administration > Manage Dashboards >** ✏ **Edit**

2. Click on the dimension that you want to use as a variable for the instances, for example ⬡dim.

3. Click **Yes** in the resulting dialog to open the Template Manager.

    The original dashboard is automatically added as the first instance of the template. For more information, see "Manage templates" on the next page.

Example

You have a dashboard with a widget that uses the data channel ⬦bvd ⬦kpi ⬘nyc. You have two more data channels that, compared to the first data channel, differ only in the dimension **location**:

⬦bvd ⬦kpi ⬘boston

⬦bvd ⬦kpi ⬘atlanta

Instead of creating a new dashboard for every location, click the dimension ⬘nyc in the existing dashboard and click **Yes** in the resulting dialog. This converts the dashboard into a template using the dimension ⬘nyc as variable. The original dashboard is automatically added as the first instance. Rename the instance and click **Save**.
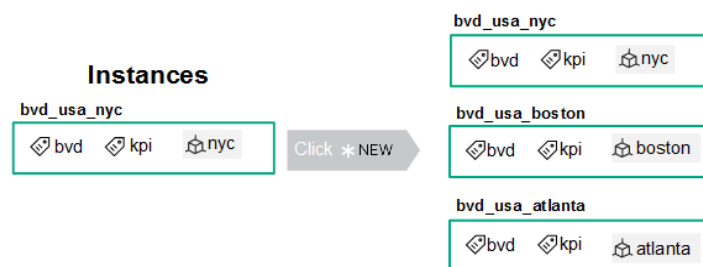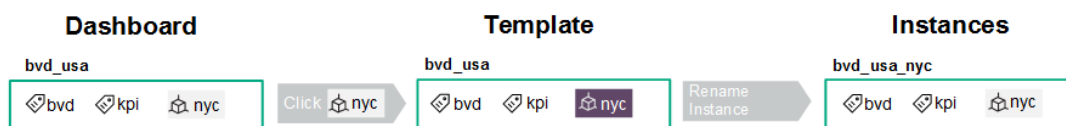
After this, create one instance of the template for every location by choosing different values for the variable **location**.

This results in three individual dashboard instances. The dashboards show different data for the widget that is using the data channel.

Manage templates

The **Template Manager** enables you to edit, download, and delete dashboard templates. You can add new instances, or edit and delete existing ones.

To access

⚒ **Administration > Manage Dashboards**

Choose a dashboard, click 🖉 **Edit**, then select the dimension that you want to use as a variable for your instances, for example 🔲 dim. Click **Yes** in the resulting dialog.

Alternatively, choose an already created template and click 🖉 **Edit**.

Learn More

This section includes:

- "Licensing for Templates and Instances" below
- "Template and Instance Icons" on the next page

Licensing for Templates and Instances

The number of dashboards you are entitled to create depends on the suite license you have obtained.

Each instance counts as one dashboard. The template itself is not counted.

Template and Instance Icons

Use these buttons to manage your dashboard templates as follows:

🗑 Delete the template and all its instances from BVD.

> **Caution:** Deleting the template will also delete all instances of the template. To only delete a single instance, use the delete button in the Instances pane.

⬇ Download the template to an SVG file.

✎ Edit the template.

Use these buttons to manage your instances as follows:

✳ NEW Add an instance of the template.

🗑 Delete an instance of the template. Click **undo** after the deletion to restore the deleted instance.

📈 Show or hide the instance in the 📈 **Dashboards** menu. This action does not impact permissions to view dashboards granted through roles. To change view permissions, see "Manage users" on page 43.

Tasks

This section includes:

- "How to Add New Instances" on the next page

- "How to Edit Instances" on the next page

- "How to Modify Templates" on page 43

How to Add New Instances

After creating the dashboard template, the original dashboard is automatically added as the first instance of the template.

Add additional instances by clicking ✳**New** in the middle pane. On the right pane, you can edit the title of the dashboard (recommended), the categories, and variables.

If you do not edit the values of the variables, the widgets will show the default values of the template.

For more information, see "How to Edit Instances" below.

How to Edit Instances

Click an instance in the middle pane to open the properties on the right.

Edit any of the following properties:

Title
Recommended. Change the title of the instance in order to differentiate between the similar dashboards.

Categories
Instances inherit the categories that are assigned to the template, but can also have individual categories assigned to them. Remove or add new categories to restrict a user's access to dashboards of a specified category. For more information, see "Categorizing Dashboards" on page 35.

Variables
The Variables section displays the dimensions you selected as variables when creating the template, and the available values of the dimensions. Change the values to create individual instances. By default, the placeholder `Variables from template` indicates that no value is assigned and the widget shows the default values of the template. To add more variables, see "How to Modify Templates" on the next page.
If you remove all variables, the instances will be deleted and the template will be converted back into a dashboard.

You can use template variables in URLs and embedded web pages. For more information, see "Web Page Widgets" on page 64.

> **Tip:** If recently added variables and/or values are not displayed, select another instance in the right pane, then reselect the instance you want to modify.

How to Modify Templates

Editing Templates

Click the ✎ **Edit** button in the left pane.

Edit any of the general Dashboard Properties, or select a widget for editing.

To add more variables to your template, select a widget for editing. Click on the dimensions you want to use as variables and save the template.

Saving the modified dashboard template will apply these changes to all instances.

Make sure to assign different values of the variables to each instance. If no values are assigned, the instances will show the default values of the template.

Replacing Templates

You can upload an SVG file as often as you want. Click the ✎ **Edit** button in the left pane, and in the Dashboard Properties click ⬆ **Replace Dashboard**. Upload a new file with the same name.

Alternatively, click ➕ in the **Manage Dashboards** page and upload a new file with the same name.

You can choose to replace the SVG only or replace both the SVG and the instances:

Click **Replace SVG** to replace the previous version of the template with the newer version while preserving the instances. The newer version does not overwrite existing dashboard properties or widgets, only new widgets are added. The changes of the new dashboard template are applied to all instances.
Click **Replace Template** to replace the previous version of the template with the newer version while removing the previous instances. The newer version overwrites existing dashboard properties and widgets.

Manage users

Use the **User Management** page to create users and groups and to define roles and permissions. To reduce the effort and complexity involved in configuring roles for individual users in BVD, permissions are granted only through roles.

You can specify roles either by assigning them to a group (so that all members of the group are assigned the same roles) or by assigning roles to a user directly. The user and group configuration depends on the functions the user or group of users fulfill, as well as the tasks that they perform.

For a suggested workflow and overview of the steps involved in setting up users, groups, and roles, see "Best Practices" below.

For a step-by-step description of the tasks you can perform through user management, see "Tasks" on page 47.

To access

🔧 **Administration > User Management**

Learn More

This section includes:

- "Best Practices" below
- "Roles and Permissions" on the next page
- "Permissions for Templates and Instances" on the next page
- "Users and Groups" on page 46
- "Super-Admin User" on page 46
- "LDAP Users" on page 47

Best Practices

Below is a set of planning best-practices and a suggested workflow for setting up user management in BVD:

1. Before you configure users, groups, and roles, you should map out the required roles and their relevant permissions, as well as the users and groups you intend to assign the roles to. For example, enter the following information in an Excel sheet:

    a. A list of users who are to access BVD, edit, and create dashboards. Gather appropriate user details such as user names, logins, and initial passwords. Although not needed to define users, at this stage it might be useful to also collect user contact information such as email addresses.

    b. If multiple users require similar permissions, create a list of groups, and the users that should belong to each group.

    c. The appropriate permissions for each role. To aid in this process, review the Permissions

Reference section to learn about dashboard categories and resources for which permissions can be granted. For details, see "Permissions Reference" on page 50.

2. Create roles and assign relevant permissions.

   For step-by-step instructions on how to create and configure a role, see "How to Create Roles" on page 48.

3. Create groups and grant them the appropriate roles.

   For step-by-step instructions on how to create a group, see "How to Create Groups" on page 48.

4. Create users, grant them the appropriate roles and place them in the appropriate groups.

   For step-by-step instructions on how to create a user, see "How to Create Users" on page 47.

Roles and Permissions

BVD enables you to fine-tune permissions management by applying permissions within roles. Permissions enable you to restrict the scope of a role. You can assign roles to users and groups enabling access to specific areas of BVD.

Permissions consist of resources, for example, **Dashboards**, to which operations, for example **View**, are applied.

- When the **View** operation is one of the resource's available operations and you select one of the other available operations, the **View** operation is also automatically selected.

- The **Full Control** operation automatically includes all operations available on the resource. When applied, the other operations are automatically selected.

For a list of available resources and descriptions of operations in BVD, see "Permissions Reference" on page 50.

Permissions for Templates and Instances

For dashboard templates, you can either assign **View** or **Full Control** permission for the template itself, or assign permissions only for specific instances.

This depends on the categories you assign to the template and each instance.

If you grant a user or group **Full Control** or **View** permission for a category assigned to the template, the user or group will have these permissions also for every instance of the template.

If you grant a user or group **Full Control** or **View** permission for a category assigned to an instance, the user will have these permissions only for that instance.

Note that granting a user or group **Full Control** permission for a category assigned to an instance, but not for a category assigned to the template itself, has the same effect as granting them **View** permission. Users or groups are only able to manage the instance if they have **Full Control** permission for a category assigned to the template.

For information on how to set permissions, see "How to Set Permissions" on page 49.

Users and Groups

- **Groups.** Groups make managing roles more efficient; instead of assigning roles to each user one at a time, you can group users who are assigned the same roles into a single unit.

- **User Types.** BVD supports different user types:

  Users with the appropriate dashboard permissions can view, edit, or create dashboards and instances. They can download tools and samples to aid in the development of dashboards and the integration of data. Users with edit permissions can view the **Error Notifications** icon on the menu bar. This icon is useful when binding dashboards to data sources. Clicking the icon displays the error list.

  Administrator users (users with super-admin permissions) have the following capabilities:

  - Full control and view on all dashboards.

  - Manage users, groups, and roles, including the creation and deletion of dashboard categories.

  - Apply custom style sheets and reset API keys in the BVD system settings.

Super-Admin User

One built-in super-admin user is defined for every installation of BVD. The login and the password for this account are specified during the BVD configuration.

The built-in super-admin is not listed among the users in user management. If you have logged in as the super-admin, you can change the user's information, including password and contact information in the **My Account** page in the ⚇ **Personal User Settings** menu.

You can apply **Super-Admin** permissions to other users in the system. These super-admin users can be modified in user management. For information on how to grant super-admin status to a user, see "How to Create Users" on the next page.

> **Note:** Super-admins have all permissions assigned and are the only user type that can work with

> user management and system settings.

LDAP Users

To obtain more user management capabilities and security, HPE recommends using external LDAP user management. You can apply the **LDAP User** type when creating or editing users to manually configure them as LDAP users. LDAP users will be authenticated against the chosen LDAP server. For information on how to configure BVD to work with LDAP, review the suite online help.

Tasks

This section includes:

- "How to Create Users" below

- "How to Create Groups" on the next page

- "How to Create Roles" on the next page

- "How to Set Permissions" on page 49

How to Create Users

1. Click **Create New User** on the user management screen.

   Alternatively, access the **Manage Users** pane and select **New User**.

2. In the Properties section, enter the required user name, login, password, and optional email.

3. *Optional.* If LDAP is enabled, you can select the **LDAP User** check box to mark the user as a manually-created LDAP user. For information on LDAP, see LDAP Authentication and Mappings.

4. Select the groups the user will be a member of.

5. Assign roles or (*optional*) set the user as a Super-admin with all permissions assigned. When finished, click **Create User**. For information on the Super-admin user type, see "Super-Admin User" on the previous page.

The user's basic information, as well as groups they belong to and roles assigned and inherited from groups, now appears on the right of the **Manage Users** pane when selecting the user.

To delete existing users, select **Manage Users**, select one or more users you want to delete, and click the Delete User or Users button.

> **Tip:** In the Manage Users pane (🔧 **Administration > User Management**), you can click the **Deactivate** button to apply the **Inactive User** flag to users who are set to be temporarily inactive, for example, while the administrator is setting up roles and groups or if the user is on vacation or leave. Inactive users cannot log on to BVD.

How to Create Groups

1. Click **Create New Group** on the user management screen.

   Alternatively, access the **Manage Groups** pane and select **New Group**.

2. In the Properties section, enter the required group name and optional description.

3. *Optional.* If LDAP is enabled, you can search for and select the LDAP groups that are mapped to the BVD group. For information on LDAP, see LDAP Authentication and Mappings.

4. Add group members and assign roles in the relevant sections of the group editor. When finished, select **Create Group**.

   > **Tip:** You can select multiple users to add to the group by clicking the **...** button next to the **Add user...** field. The **Select Users** editor opens. Select a single user, and then hold down the **Ctrl** key while you click other users that you want to select.
   >
   > When working with BVD on a mobile device, make sure to click a user twice to select it.

The group's basic information, including roles assigned, now appears on the right of the Manage Groups pane when selecting the group. You can also navigate between groups in the group's information pane.

To delete existing groups go to the **Manage Groups** pane, select one or more groups you want to delete, and click the Delete Group or Groups button.

How to Create Roles

1. Click **Create New Role** on the user management screen.

   Alternatively, access the **Manage Roles** pane and select **New Role**.

2. In the Properties section, enter the required role name and optional description.

3. In **Permissions**, you can select the relevant category and set the appropriate permissions by checking the related boxes, or use the drop-down menus on each permission summary.

   For a list of permissions and related descriptions see "Permissions Reference" on page 50.

For a detailed task on setting permissions, see "How to Set Permissions" below.

If you have already created users or groups, you can assign them during role creation or editing. When finished, select **Create Role**.

To delete existing roles, select **Manage Roles**, select one or more roles you want to delete, and click the Delete Role or Roles button.

> **Note:** When there are more than 1000 users in the system, the user search box is no longer displayed. Click the **...** button to access the user list and search filter dialog box.

How to Set Permissions

In this task, you set or modify the permissions in BVD roles. Users require permissions to view, edit, or create BVD dashboards, templates, and instances.

1. In the **User Management** page or in the **Manage Roles** page, create or edit the role you want to configure.

2. Scroll to the permissions section in the **Create Role** or **Edit Role** page.

3. Expand **Dashboards > Categories** and assign an operation to one or more dashboard categories:

   - **All.** To grant permissions to all dashboards and instances regardless of the category, click **View** or **Full Control** for **All**.

   - **Assigned to category.** To grant permissions to all dashboards and instances with any category, click **View** or **Full Control** for **Assigned to category**.

     Alternatively, to grant access to individual dashboard categories, select the appropriate category.

   - **Not assigned to category.** To grant permissions to all dashboards and instances without a category, click **View** or **Full Control** for **Not assigned to category**.

4. Click **Manage Categories** to add new categories to the permissions list. You can also delete categories from the list if they are no longer needed.

> **Note:** Changes to a user account are only fully available after the user whose account was changed logs in again to BVD.

Permissions Reference

Dashboards

Categories	All	Grants view or full control permission for all dashboards (with or without categories assigned).	
		Full control on **All** grants the following additional permissions:	
		• Upload dashboards.	
		• Convert dashboards into templates.	
		• Delete dashboards, templates, and instances.	
		• Create and delete categories, and assign categories to dashboards and instances.	
		• Download tools and samples from the 🔧 **Administration** menu.	
	Assigned to category <category> **Not assigned to category**	View	Grants permission to:
			• View dashboards and instances in the 📈 **Dashboards** menu.
			• Make a dashboard the default dashboard.
		Full Control	• Edit dashboards in the **Manage Dashboards** page.
			• Show or hide dashboards in the 📈 **Dashboards** menu.
			• Download dashboards.
			• Convert dashboards into templates.
			• Edit, download, and delete templates.
			• Edit and delete instances.

Download tools and samples

BVD provides an extensive set of tools and samples to help you develop dashboards and integrate data:

Visio Stencil (required)

`BusinessValadmueDashboardStencil.zip`. Contains the BVD widgets for Visio. Start Visio; in the **Shapes** window, click **More Shapes > Open Stencil** and select the extracted BVD stencil file.

You can also save the stencil file to **C:\Users\<user name>\My Documents\My Shapes** folder and then access the BVD stencil in Visio in **More Shapes > My Shapes**.

Visio Dashboard Template (useful)

`Dashboard_Template.zip`. Contains a Visio template with an embedded macro. The macro exports your Visio drawing to SVG each time you save it. The SVG file has the same name as the Visio drawing and is stored in the same folder.

You can add the BVD template to Visio by specifying the template file name in **Visio Options > Save > Default personal templates location**.

Visio Add-in (useful)

`VisioAddIn.zip`. Contains an installer for a Visio ribbon called **Dashboard**. The button **Export Dashboard** in the ribbon exports your Visio drawing to an SVG file with the same name and in the same folder as the drawing itself.

Make sure you have administrative privileges when installing the Visio Add-in.

Data Generator Sample (interesting)

`samplegen.zip`. Contains a VBScript and a batch file to send data to the OMi sample dashboard. The batch file prompts you for the URL of your data receiver, your API key, and the HTTP proxy and port.

Sample OMi Status Dashboard (interesting)

`omi_sample.zip`. Contains the Visio drawing of the OMi sample dashboard. The dashboard is the target of the data sent by the data generator VBScript. You can use it as the basis for creating your own OMi event and KPI status dashboards.

AO-Bank Sample Dashboards (useful)

`ao-bank.zip`. Contains the Visio files for the AO-Bank sample dashboards, which you can view at

the BVD demo.

AO-Cellular Sample Dashboards (useful)

`ao-cellular.zip`. Contains the Visio files for a sample cellular provider.

Use

This section provides more advanced details about learning how to use BVD. It includes reference information, procedures, and tips that you may find useful.

Read this section when you are comfortable with the basic configuration steps, and refer back to it when necessary. The information in this section is organized as a reference.

Widgets

The topics in this section describe all available dashboard widgets. You can create the widgets from the following BVD shapes in Visio:
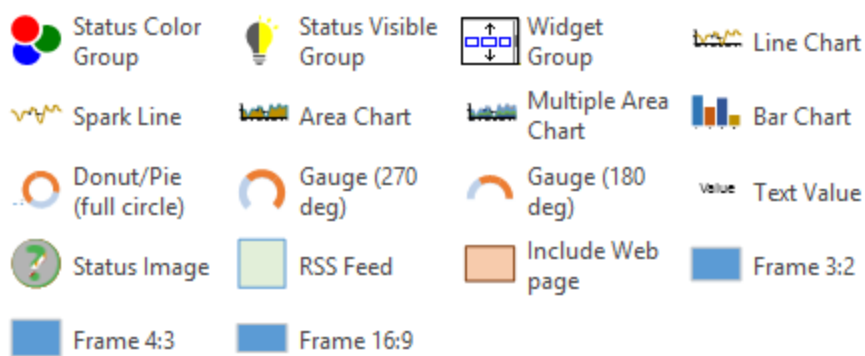
You can edit the properties of a widget in Visio (by editing the Shape Data) or in BVD (by editing a dashboard in Manage Dashboards). We recommend that you make your changes in BVD because the dashboard editor simplifies this task by offering values in drop-down lists for you to choose.

Some advanced operations can only be done in Visio itself. See "Tips and Tricks" on page 88 for details.

You can upload an SVG file as often as you want. If the file already exists, BVD replaces the previous version of the file with the newer version and opens the associated dashboard for you to edit. The newer version does not overwrite existing dashboard properties or widgets, only new widgets are added.

Rule Operators

Rules use operators to compare current and given values. The available operators are:

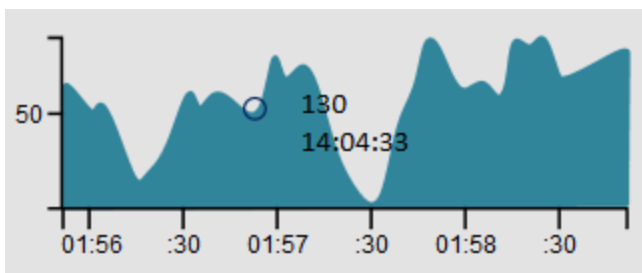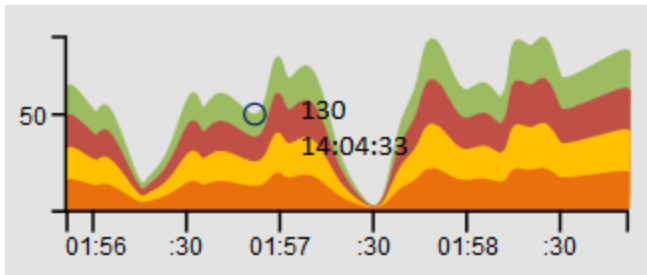|	
==	equal
<	less than
>	greater than
<=	less than or equal
>=	greater than or equal
!=	not equal

Available widgets:

Area and Multiple Area Chart Widgets

Area charts are used to display quantitative data. They are based on line charts.

If you have only one set of data to display, use a simple **Area Chart** widget:

To display multiple sets of data, use a **Multiple Area Chart** widget:

You can customize the x- and y-axes by changing the font and color in Visio.

See also:

- "Data Channel" on page 76

- "Data Field" on page 77

- "Min Value" on page 83

- "Max Value" on page 82

- "Chart Autoscale" on page 72

- "Mouse Over" on page 84

- "Show Chart Numbers" on page 85

- "Chart Colors" on page 73

- "Chart Period" on page 74

- "Number Format" on page 84

- "Visibility Rule" on page 87

- "Hyperlink" on page 79

Bar Chart Widgets

Use the **Bar Chart** widget to display a bar chart. Bar charts are useful when you want to compare multiple values from the same data channel. Each bar in a bar chart corresponds to a field in the data received over the associated data channel.

The **Data Field** and the **Chart Colors** properties refer to the bars using a numbering scheme. Bar number 1 is the leftmost bar with numbering continuing to the right:
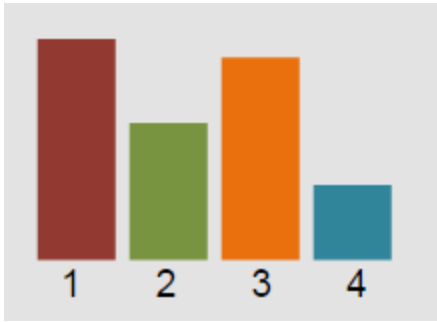
For details on how to set the colors of the individual bars, see "Chart Colors" on page 73.

To make room for larger labels, you can rotate the bar chart labels in Visio.

The **Angle** property in Visio determines the orientation of the label. A default bar chart label has a start angle of 0 (zero). To rotate the labels, set the angle of the first label to the angle at which you want all labels to appear. You can do this either by clicking on the label and rotating it or by changing the value of the label's **Angle** property:
Access the tab **View**. Click **Task Panes > Size & Position**. Click on the first label of the bar chart and adjust the **Angle** property.

For example, to rotate the labels of the following bar chart, set the angle of the first label to -45 degrees. As a result, all other labels will be rotated to the same degree in BVD.
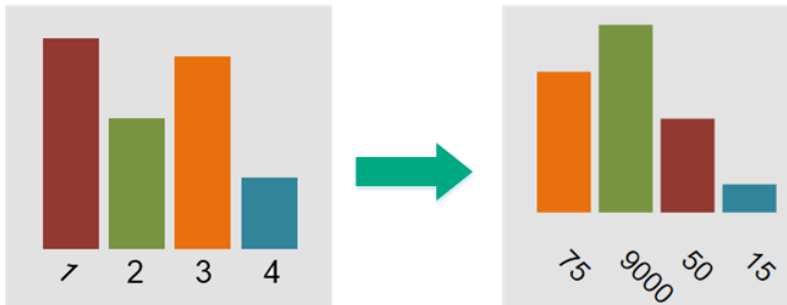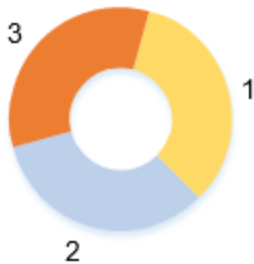

See also:

- "Data Channel" on page 76

- "Data Field" on page 77

- "Max Value" on page 82

- "Chart Autoscale" on page 72

- "Show Chart Numbers" on page 85

- "Chart Colors" on page 73

- "Number Format" on page 84

- "Visibility Rule" on page 87

- "Hyperlink" on page 79

Donut/Pie Widgets

Use the **Donut/Pie** widget to display a donut or pie chart. Each slice in a donut/pie widget corresponds to a field in the data received over the associated data channel.

The **Data Field** and **Chart Colors** properties refer to the donut/pie slices using a numbering scheme. Slice number 1 is the top right slice with numbering continuing clockwise:

In comparison to the Donut Chart widget, you can modify the hole size of the Donut/Pie widget directly in Visio, by using the yellow marker that appears when you click on the chart.

Right-click the widget and select **Hide Labels** if you do not want the widget to be displayed with labels for the received data.
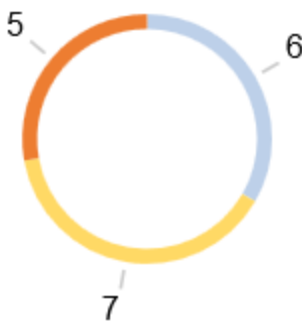
> **Note:** To also modify the start angle and donut size in Visio, use a Gauge widget.

See also:

Feed Widgets

The **Feed** widget enables you to display information feeds, similar to well-known RSS feeds. Feeds must be sent as JSON-encoded data. The data channel must include the field's `title` and `link` in order to display the feed. New feed items are always added to the top of the list.

The Feed widget displays the following items:

- The creation time stamp of the item.

- The title field as text (retrieved from the `title` data field).

- The link field as hyperlink (retrieved from the `link` data field).

Example feed data:

```
{
    "time":1437633749317,
    "type":"test",
    "title":"Tests show UK Quran manuscript is among world's oldest",
    "link":"http://rss.cnn.com/~r/rss/cnn_latest/~3/HgufPus_pOs/index.html"
}
```

Note: Do not rotate Feed widgets. This breaks the widget functionality.

You can style the displayed feed items by providing a custom CSS definition in the BVD System Settings. For more information, see "Modify the system settings" on page 31.

Example

```
.feedItem .ts {display: none;}
.feedItem .even {background-color: #262627;}
.feedItem a:hover {text-decoration: none;}
.feedItem section {hight:30px; padding: 3px; margin: 0}
```

See also:

- "Data Channel" on page 76

- "Feed Max Items" on page 79

- "Transparent Background" on page 87

Frame Widgets

BVD offers the following Frame shapes:

- Frame 3:2

- Frame 4:3

- Frame 16:9

The shapes do not have any shape data defined. Their purpose is to help you lay out your dashboards based on predefined ratios.

Gauge Widgets

Use the **Gauge** widget to display a gauge chart. By default, gauge widgets are available with a 180 or 270 degrees start angle. Each slice in a gauge chart corresponds to a field in the data received over the associated data channel.

The **Data Field** and the **Chart Colors** properties refer to the gauge slices using a numbering scheme. Slice number 1 is the top right slice with numbering continuing clockwise:
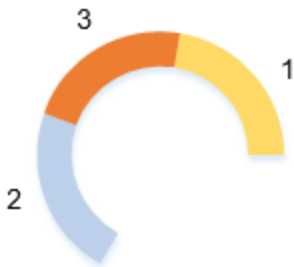
You can modify the start and end angle, donut size, and hole size of the gauge widget directly in Visio, by using the yellow markers that appear when you click on the chart.

Right-click the widget and deselect **Hide Labels** if you want the widget to be displayed with labels for the received data.
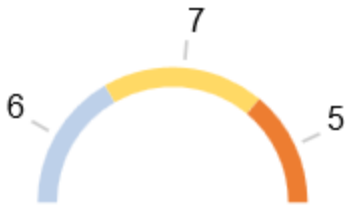
See also:

- "Data Channel" on page 76

- "Data Field" on page 77

- "Max Value" on page 82

- "Chart Colors" on page 73

- "Number Format" on page 84

- "Visibility Rule" on page 87

- "Hyperlink" on page 79

Line Chart and Sparkline Widgets

Use the **Line Chart** widget to show a line chart with axes and coordinates:
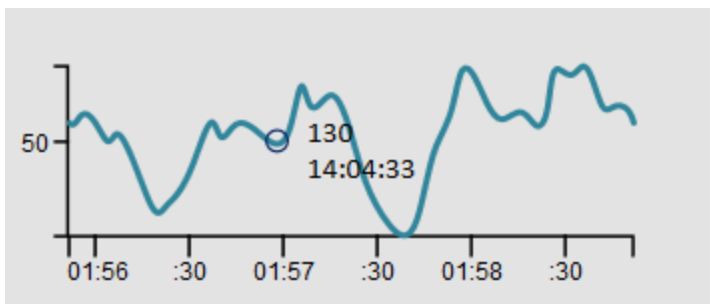
The **Sparkline** widget creates a sparkline chart. Typically, a sparkline is a very small chart, shown without axes or coordinates:
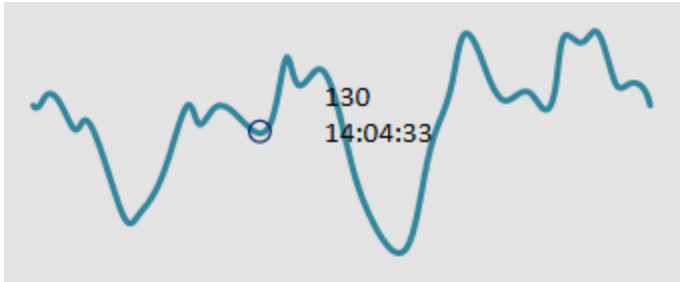
> **Tip:** Position sparklines next to an absolute number to provide you with a quick reference to the data trend.

You can change the style of the line in Visio.

See also:

- "Data Channel" on page 76

- "Data Field" on page 77

- "Min Value" on page 83

- "Max Value" on page 82

- "Chart Autoscale" on page 72

- "Mouse Over" on page 84

- "Chart Period" on page 74

- "Number Format" on page 84

- "Visibility Rule" on page 87

- "Hyperlink" on page 79

Status Color Group

Group the **Status Color Group** shape with other shapes in Visio to make them change color depending on the values received. You can define the values that cause a color change using the Coloring Rule property.

Use the group widget when you want to change the color of non-BVD shapes or to color a large number of shapes. Then it is easier to group the shapes with the Status Color Group and set the coloring rule in the group widget.

The Status Color Group widget can color the following SVG elements: `<path>`, `<rect>`, `<ellipse>`, `<circle>`, and `<polygon>`.

See also:

- "Data Channel" on page 76

- "Coloring Rule" on page 75

- "Visibility Rule" on page 87

- "Hyperlink" on page 79

Status Image Widgets

Use the Status Image widget to display an image depending on the value received for the Status Field or depending on a rule.

The Status Image shape is a group of shapes. Each shape in the group is an image, and each image has the shape data Switch Value defined. For example, the "error" image in the BVD Status Image shape has the Switch Value "red". BVD then selects the image to display depending on the values received in the data channel. You can choose the data field to use as input by selecting it in the Status Field (default: `status`), or you can define an image selection rule. For example, if the result of the image selection rule is "red", the "error" image is displayed.
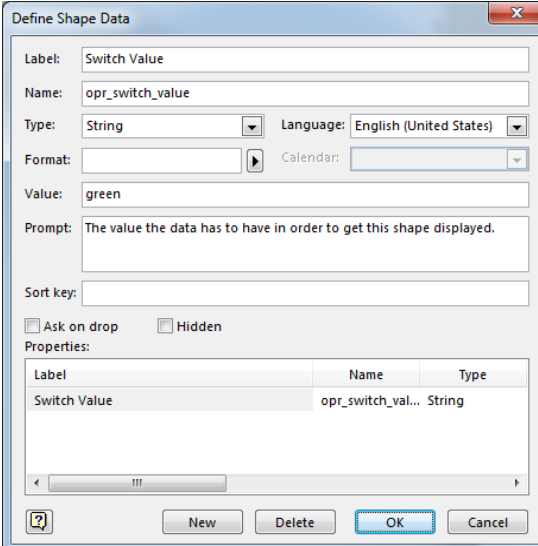
You can take a look at the Status Image shape in Visio by right-clicking the shape and selecting **Group > Open Status Image**.

> **Caution:** Never ungroup the Status Image shape.

The shape consists of the following four images:

Image Name	Switch Value	Image
ok	green	✅
warning	yellow	⚠️
error	red	❌
unknown	grey	❓

Do it yourself - Create your own Status Image

1. Make sure Visio is running in developer mode:

   File > Options > Advanced > Run in developer mode

2. Drag the BVD Status Image shape to your drawing.

3. Right-click the Status Image shape and select **Group > Open Status Image**.

   You are now inside the shape. You can start editing, deleting, or adding shapes as required.

4. If you add a new shape, you must add the shape data **Switch Value** to the shape:

   a. Right-click the added shape and select **Data > Define Shape Data**.

   b. Add a property with the name `opr_switch_value` of the type `String`.

   c. Set the value of the property to a value of the Status Field or one that can be selected by an Image Selection Rule.

After uploading the exported SVG file to BVD, select the data channel for your Status Image widget. Then either select a data field for the Status Field property or configure an image selection rule. You

also need to set a default value for situations when no value is available or the value is not one of the defined switch values.

See also:

- "Data Channel" on page 76

- "Status Field" on page 86

- "Default Value" on page 78

- "Image Selection Rule" on page 81

- "Visibility Rule" on page 87

- "Hyperlink" on page 79

Status Visible Group

Group the invisible **Status Visible Group** Visio shape with other shapes to show or hide the widgets depending on the result of the rule defined in the **Visibility Rule** property.

> **Tip:** Use the group widget when you want to show or hide non-BVD shapes, or to show or hide a large number of shapes. Then it is easier to group the shapes with the Status Visible Group and set the visibility rule in the group widget. You can also set the visibility of a widget based on its **Visibility Rule** property. Use this property to show or hide individual widgets.

See also:

- "Data Channel" on page 76

- "Visibility Rule" on page 87

- "Hyperlink" on page 79

Text Value Widgets

Use the Text Value widget to display values or to color text. Text attributes like font, alignment, and rotation are not updated.

For example, you could use the text value widget to display the current temperature in your store in New York City. As temperature measurements arrive, the number changes reflecting the current temperature measured in the store. Additionally, you could configure the text value widget to change the color of the value displayed depending on the current temperature:

Temperature in NYC: 25

See also:

- "Data Channel" on page 76

- "Data Field" on page 77

- "Number Format" on page 84

- "Coloring Rule" on page 75

- "Visibility Rule" on page 87

- "Hyperlink" on page 79

Web Page Widgets

Web Page widgets enable you to show web pages in a dashboard. For example, you can include web pages that stream television, video, or audio.

The website must be accessible via HTTPS because the BVD UI is also only accessible via HTTPS. However, if the website sends an X-Frame-Options HTTP header restricting the embedding of this page to same origin, the browser will not display this web page within the dashboard.

| **Note:** Do not rotate Web Page widgets. This breaks the widget functionality.

You can add variables to the URL, either based on the data fields in the data channel, or based on the variables from a dashboard template. Variables are replaced with their assigned value when a dashboard or an instance is viewed.

See also:

- "URL" on page 87

Widget Groups

Use the **Widget Group** to display a dynamic list of individual widgets within one widget container. Every widget that is grouped within a widget group receives data from the same data channel. The JSON data that is sent via this data channel must be an array of objects that contain the list items. The widget group works like a list that is dynamically populated with list items based on the data that is being sent.

> **Note:** Adding Feed widgets, Web Page widgets and Frames to widget groups is not supported.
> Other than that, all widget types as well as non-widget shapes can be added to widget groups.

The display of the widgets inside the widget group is limited by the size of the widget group box. If the content of the widgets is larger than the space in the box, the content is cut off. For example, if you have a Text Field Widget with a long value inside the widget group, the text is cut off if the text is longer than the space available within the widget group box.

You can modify the color of the box and the scroll bar individually. The vertical scroll bar appears if the length of the content that is displayed in the widget group exceeds the size of the box.

To add widgets to the widget group, put another widget on top of the widget group, for example the Text Value Widget. Then select both widgets, right-click and select **Group** > **Group**.

Learn more

Send JSON data

The data that is being fed into the widget group via the data channel property must be defined in JSON data and contain an array of values similar to the following:

```
{

	"items" : [
	{
		"id": "1",
		"value": "content1"
	},
	{
		"id": "2",
		"value": "content2"
	},
```

```
      {
            "id": "3",
            "value": "content3"
      }
   ]

}
```

The JSON data must contain an array of values called `items`. The array elements must be JSON objects with a property called `id` that uniquely identifies the single instances of the widgets inside the group.

Data updates must always contain data for all instances to be displayed. If an ID is missing in the updated data, the corresponding widget instance is removed. If a new ID is added in a data update, a new widget instance is added.

Widget Groups with multiple widgets or shapes

A Widget Group can contain more than one widget or shape. Note the following when working with complex widget groups:

- You must group the widgets/shapes inside the group before grouping them with the widget group. For example, if you have two widgets and a basic rectangle shape inside a group widget, group them as follows:
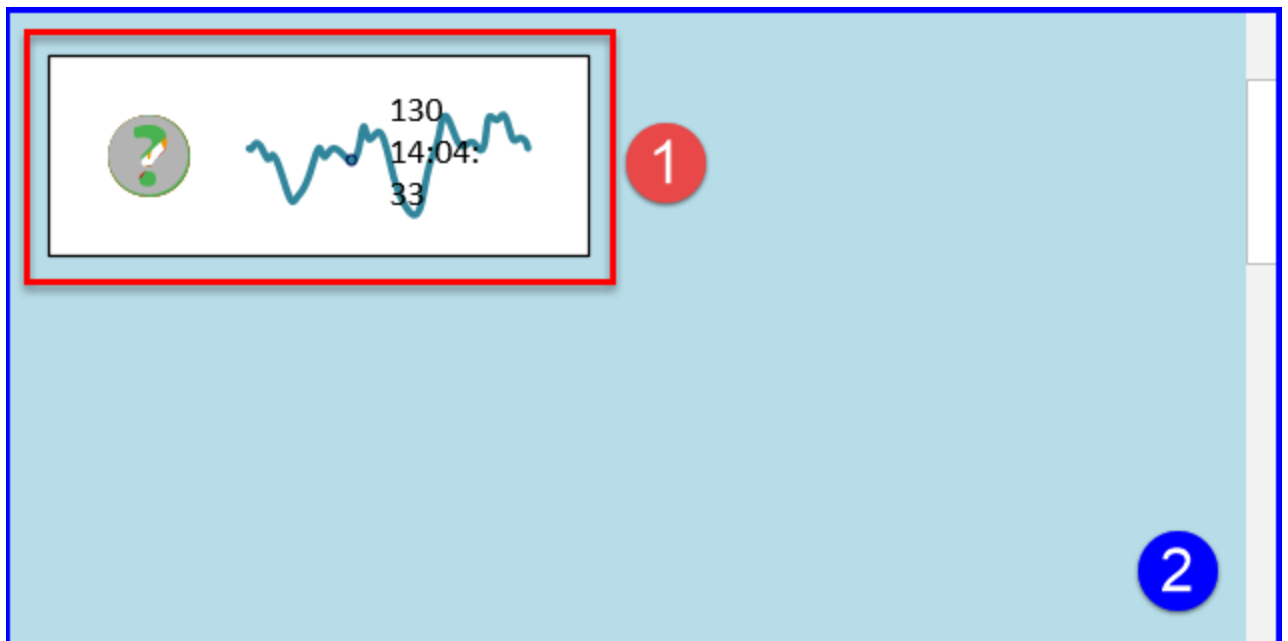
First, you select all widgets and shapes in (1), right-click and select **Group** > **Group**. Then you select the grouped widgets/shapes and the widget group box (2), right-click and again select **Group** > **Group**.

- Widgets inside the group are sorted in the same way as the data in the JSON array. You can control the display order by changing the order of the values in the array.

- In BVD, you can select the widgets inside the widget group and change their properties. However, you cannot change the data channel for those widgets as the channel is set by the group. If you hover over the data channel of the widget group, the latest received values are displayed in a tooltip.

Examples

Example (simple): adding a Text Value Widget to a widget group

As an example, create a dashboard that displays the values of multiple dimensions in a widget group.

1. In Visio, create a new drawing and drag the Widget Group from the BVD stencil into the free space.
2. *Optional*. Change the color of the widget group box and scroll bar.
3. Drag the Text Value Widget on top of the Widget Group. Select both widgets, right-click and select **Group** > **Group**.

   Note that you only have to add one Text Value Widget into the Widget Group. The widget will be dynamically replicated inside the widget group as often as required to display all of the data being sent via the data channel.
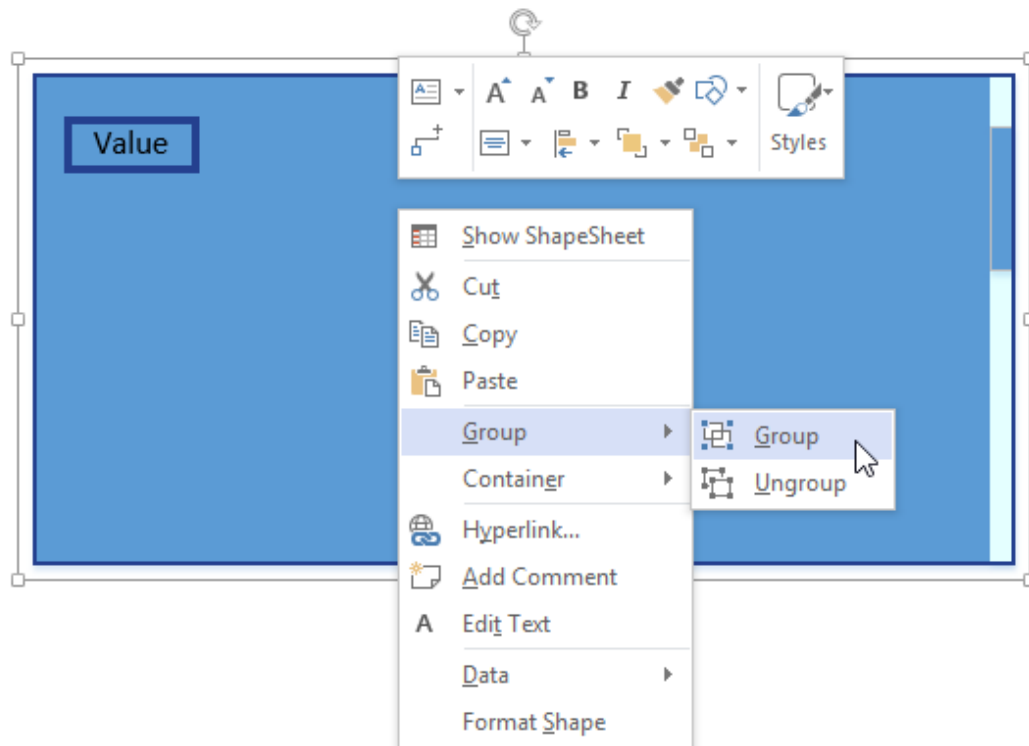
4. Export the dashboard (or manually save it as an SVG file), then upload the new dashboard to BVD in the Manage Dashboards page.

5. Send nested JSON data to BVD. For this example, we will use the following JSON data:

```
{

    "items" : [
    {
            "id": "1",
            "value": "content1"
    },
    {
            "id": "2",
            "value": "content2"
    },

    {
            "id": "3",
            "value": "content3"
    },

    {
```

```
                                "id": "4",
                                "value": "content4"
                        },
                        {
                                "id": "5",
                                "value": "content5"
                        },

                        {
                                "id": "5",
                                "value": "content5"
                        },

                        {
                                "id": "6",
                                "value": "content6"
                        },
                        {
                                "id": "7",
                                "value": "content7"
                        },

                        {
                                "id": "8",
                                "value": "content8"
                        },

                        {
                                "id": "9",
                                "value": "content9"
                        }

                ]

        }
```

Send the data to the receiver URL as described in "Create custom integrations" on page 22.

6. In BVD, edit the dashboard you uploaded earlier. Select the widget group, and choose the data channel that contains the items property.

7. *Optional*. Select the Text Widget inside the widget group to change its properties. Note that you cannot change the data channel for widgets inside a widget group as the channel is set by the group.

8. Click **Save**.

9. Select and view the dashboard in the 📈 **Dashboards** menu. The widget group displays all of the list items one after the other. You can use the scroll bar to see all of the values.

10. *Optional*. To see how the widget group displays the content dynamically, add additional list items to the JSON data, and delete or modify existing list items.

Example (complex): adding multiple widgets/shapes to a widget group

As a bit more complex example, create a dashboard that displays the values in a group containing a Text Widget, a Status Image Widget, and a Spark Line Widget.

1. In Visio, create a new drawing and drag the Widget Group from the BVD stencil into the free space.

2. *Optional*. Change the color of the widget group box and scroll bar.

3. Drag and drop a rectangle basic shape into the Widget Group and adjust its size. Then add a Text Value Widget, a Status Image Widget, and a Spark Line Widget on top of the frame. Select the three widgets and the rectangle, right-click and select **Group** > **Group**.

> **Note:** It is important where in the widget group you place your widgets and shapes. The space between the grouped widgets/shapes and the right side of the widget group box stays the same in the final BVD dashboard. Additionally, the space between the grouped widgets/shapes and the top of the widget group box defines the space between the repeated elements in the final dashboard.
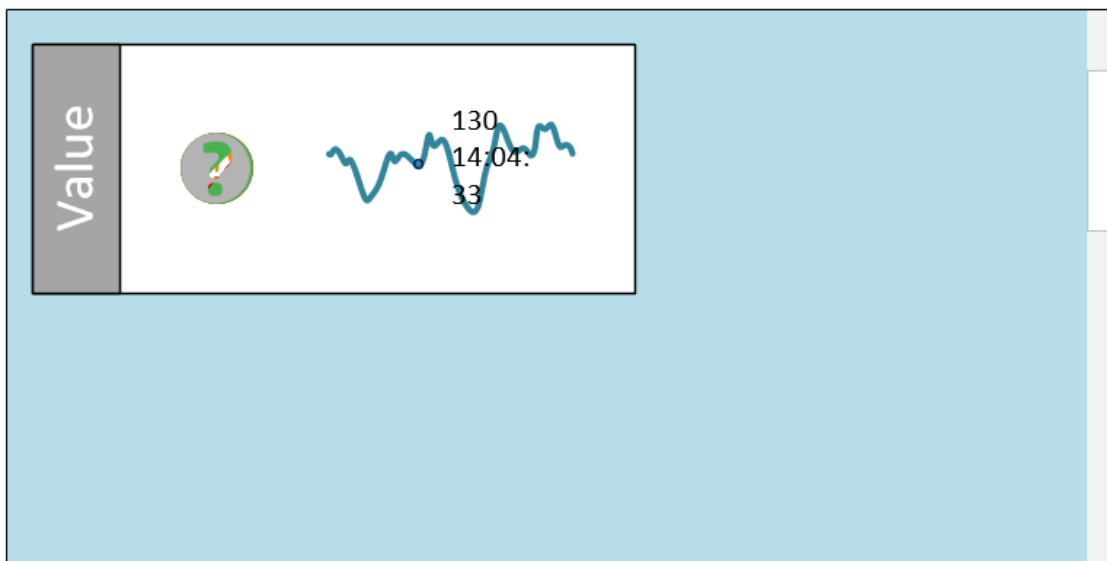
4. Select the grouped widgets and the widget group box, and again select **Group** > **Group**.

5. Export the dashboard (or manually save it as an SVG file), then upload the new dashboard to BVD

in the Manage Dashboards page.

6. Send nested JSON data to BVD. For this example, we will use the following JSON data:

```
{
    "items" : [
    {
            "id": "1",
            "value": "content1"
    },
    {
            "id": "2",
            "value": "content2"
    },

    {
            "id": "3",
            "value": "content3"
    },

    {
            "id": "4",
            "value": "content4"
    },
    {
            "id": "5",
            "value": "content5"
    },

    {
            "id": "5",
            "value": "content5"
    },

    {
            "id": "6",
            "value": "content6"
    },
    {
            "id": "7",
            "value": "content7"
    },

    {
            "id": "8",
            "value": "content8"
    },

    {
```

```
            "id": "9",
            "value": "content9"
      }
    ]
  }
```

Send the data to the receiver URL as described in "Create custom integrations" on page 22.

7. In BVD, edit the dashboard you uploaded earlier. Select the widget group, and choose the data channel that contains the items property.

8. Select the widgets inside the widget group to change their properties. For the Status or Data Field, select the data you want the widget to display. Note that you cannot change the data channel for widgets inside a widget group as the channel is set by the group.

9. Click **Save**.

10. Select and view the dashboard in the ⬈ **Dashboards** menu.

11. *Optional*. To see how the widget group displays the content dynamically, add additional list items to the JSON data, and delete or modify existing list items.

See also:

- "Data Channel" on page 76

Widget Properties

The topics in this section describe all available widget properties.

Chart Autoscale

Chart Autoscale automatically scales the maximum values and, if available, minimum values.

Bar charts only. If you set Max Value to 0, the bar with the highest value will be shown in full height; the height of all other bars is shown relative to the highest bar.

Default: not selected

Examples

The following three sparklines have the same size and show the same data. The scaling, however, differs because of different min and max values or Autoscale:

Chart Colors

The **Chart Colors** property enables you to set colors for your charts. Provide a semicolon-separated list of RGB color codes in hexadecimal notation.

You can choose the coloring of the first four colors by changing the colors in Visio. However, the colors you specify in the Chart Colors property override the colors defined in Visio.

If you specify only one color, BVD uses this color for the first data field, and uses the Visio-defined colors for data fields two to four. If more than four data fields but no colors are defined, BVD randomly chooses a color for data fields five and higher.

Default: not defined

Example Donut Chart

The donut chart in the OMi sample dashboard uses the following color codes:

`FF0000;FF9933;C8C800;33CC33;B2B2B2`

They color the donut chart like this:
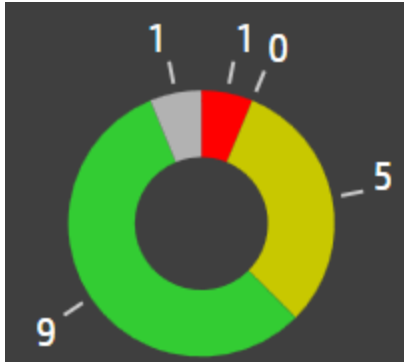
Chart Period

Use the **Chart Period** property to set the period of time (in minutes) for which the widget will display data.

BVD stores only the last 500 data records received from the data senders. To be able to display data, you must align the chart period with the rate data is received. For example, if BVD receives one value per day, the maximum chart period would be 720,000 minutes (500 days). If data is received every 15 minutes, the maximum chart period would be 7,500 minutes (5.2 days).

Default: 10 minutes

Examples

The following two sparklines have the same size and show the same data. The upper line shows the data for the last 10 minutes, the lower line only the last minute.
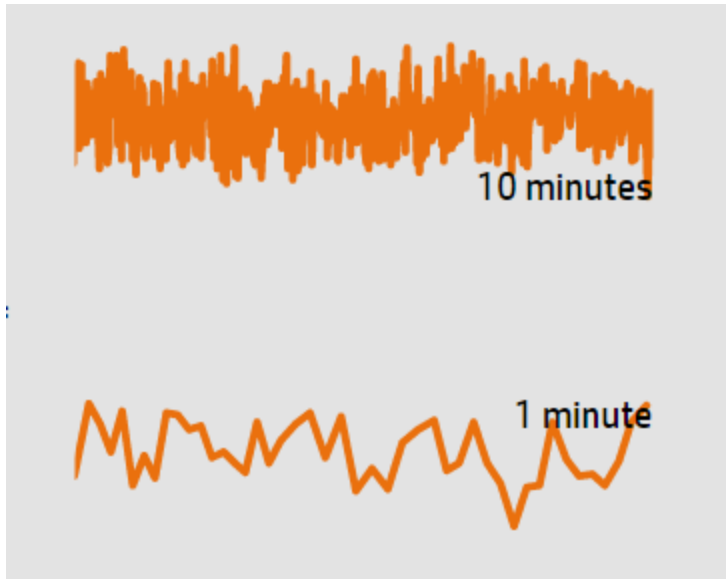
Coloring Rule

The **Coloring Rule** property enables you to determine the color to display depending on the outcome of a rule.

You can add multiple coloring rules separated by semicolons:

Rule format: `<rule>;<rule>;...`

Coloring rules can contain a value only; for example, the value `#008000` in a coloring rule colors the widget green. Rules can also contain conditions that must be matched; for example, `#008000:temperature<60` colors a widget green when the value of the data field `temperature` is less than 60.

Rules are evaluated from left to right. When one condition is matched, no additional rules will be evaluated. If no rule matches, the default is applied; therefore, when you define a set of rules, always insert the default as the last rule.

Coloring rules have the following format:

`<color>[:<property><operator><value>]`

`<color>`

RGB color code; for example, #000000.

`<property>`

The name of the data property to use to calculate the color.

`<operator>`

The operation that is used to compare the current value of the property with the given value. For a list of operators, see "Rule Operators" on page 53.

`<value>`

The value the operator works on.

Example

```
#AABBCC:temperature<60;#7FFF00:temperature<30;#00ff00
```

If the value of the data field `temperature` is less than 60, the color #AABBCC is used. If the value is less than 30, the color #7FFF00 is used. In all other cases, #00ff00 is used as the color.

Data Channel

The **Data Channel** property enables you to select the data stream over which you want to feed data into the widget. Before you can select a data stream, you must send data to BVD, addressed to the data channel you want to select for your widget.

Example

To show the rise and fall of the temperature in your store in New York City, select the following data channel for your widget:

The data received includes the following:

```
type: Temperature
element: Monitor
value: 20.9
status: #008000
```

Data Field

Select one or more data fields in the data that BVD received through the data channel to display in your widget.

Single Data Field

Sparkline, line, and area charts as well as status image and text value widgets support only one data field.

For single data fields, BVD assumes that the data includes the data field `value` and uses the values received for `value` in the widget.

Default: `value`

> **Example Line Chart**
>
> The temperature data BVD is receiving from your New York City store already happens to include the `value` data field, so you do not need to select it. Otherwise select the data field with your temperature values.
>
>

Multiple Data Fields

Multiple area, bar, and donut charts support multiple data fields. Select a data field for each area, bar, or slice in the chart. If you select more than four data fields, BVD automatically chooses the colors of the additional fields. For details, see "Chart Colors" on page 73.

Donut charts only. If you select only one data field for a donut chart, a second one will be automatically generated based on the max value. For details, see "Max Value" on page 82.

> **Example Donut Chart**
>
> The donut chart in the OMi sample dashboard displays the five data fields representing the number of events per severity, resulting in a donut with five slices:
>
> Data Field:
>
> | × numberOfCritical | × numberOfMajor |
> | × numberOfMinor | × numberOfNormal |
> | × numberOfUnknown |
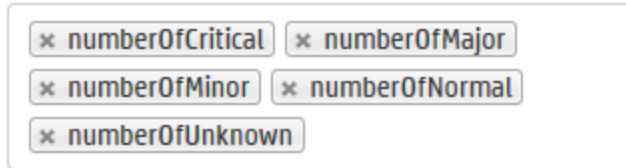
Default Value

The value set in **Default Value** is used when the data in the Status data field is empty or is not one of the defined switch values. For example, if `status: blue` or if `status: <empty>`, then the selected default value will be used by the Status Image widget.

You can only select one of the switch values defined in the Status Image widget.

The switch values of the BVD default Status Image widget are grey, red, green, and yellow.

Default: grey

Donut Size

The **Donut Size** property configures the size of a donut chart within the 360 degrees of the circle of the donut chart. By default, a donut completes a full circle (360 degrees). To create a gauge-like semicircle, set the donut size to 180 degrees.

Default: 360

Donut Hole Size

The **Donut Hole Size** property configures the size (in percent) of the hole in the middle of a donut chart. To create a pie chart, set the donut hole size to 0 percent.

Default: 35

Feed Max Items

The **Feed Max Items** property sets the maximum number of items to display in the Feed widget.

Hyperlink

The **Hyperlink** property enables you to link a widget to another dashboard or to a URL. When a user then clicks the widget, the linked dashboard or URL opens. Linked dashboards open in the current browser window, while external URLs open in a new window.

You can add variables to the URL in order to link widgets to specific content based on the data channel. Use the values assigned to the data fields, or use the variables from a dashboard template. You can also combine both types of variables in one URL.

Data fields as variables

Use the data fields of your received data as variables in the URL. Specify the data field names as variables. The data field names are replaced with their associated values when the dashboard is viewed. You can add data field variables to the URL in the format `#{dataField_name}`.

`https://example.com/#{dataField_name}`

When the value associated with the variable changes, the URL changes accordingly.

> **Tip:** You can specify only the data field name, for example `#{url}`, if the data field has a complete URL as value, for example `https://example.com`.

Example

A company wants to link a widget to their website, searching for the host name specified in the data channel:

⬡east ⬡Host A ⬡CPU load ⬡NYC

They specify the following URL, which later replaces the query value with the value of the dimension `host`:

`https://example.com/q=#{host}`

When viewing the dashboard, the dimension `host` is replaced with its assigned value in the URL:

`https://example.com/q=Host+A`

If the variable is not defined, the variable expression in the URL is replaced with an empty string.

Template variables

You can add variables from dashboard templates to the URL. Variables are replaced with their assigned value when an instance is viewed. You can add variables to the URL in the format `${variable}`.

Example

A company uses a URL in a dashboard template to show the floor plan of their company location. They use different instances for each location and want to show the floor plan specific to each location.

The following URL replaces the query value with the value of the variable `location` that is assigned to the instance:

`https://example.com/office-locations/floor-plans?site=${location}`

When viewing the instance with the assigned value `Atlanta`, the floor plan for the company location in Atlanta is shown. The `location` variable is replaced with `Atlanta` in the URL:

`https://example.com/office-locations/floor-plans?site=Atlanta`

If the variable is not defined or the template is converted back into a dashboard, the variable expression in the URL is replaced with an empty string.

For more information on templates and variables, see "Manage templates" on page 39.

Image Selection Rule

The **Image Selection Rule** property enables you to determine the image to display depending on the outcome of a rule.

> **Note:** Image selection rules override the values received for the Status Field.

You can add multiple rules separated by semicolons:

Rule format: `<rule>;<rule>;...`

Rules can contain a value only; for example, the value `green` in an image selection rule selects the switch value `green`. Rules can also contain conditions that must be matched; for example, `green:statusColor==verde` selects the switch value `green` when the value of the data field `statusColor` is `verde`.

Rules are evaluated from left to right. When one condition is matched, no additional rules will be evaluated. If no rule matches, the default is applied; therefore, when you define a set of rules, always insert the default as the last rule.

Image selection rules have the following format:

`<switch value>[:<property><operator><value>]`

`<switch value>`

Switch value assigned to an image in the Status Image shape. The switch values of the default Status Image shape are `green`, `yellow`, `red`, and `grey`.

`<property>`

The name of the data property to use to calculate the color.

`<operator>`

The operation that is used to compare the current value of the property with the given value. For a list of operators, see "Rule Operators" on page 53.

`<value>`

The value the operator works on.

> **Example**
>
> ```
> green:statusColor==verde;yellow:statusColor==amarillo;red:statusColor==rojo;g
> rey
> ```
>
> If the value of the data field statusColor is `verde`, the image with the switch value `green` is selected. The value `amarillo` selects the image `yellow`, `rojo` selects `red`, and the image `grey` is displayed in all other cases.

Max Value

Max Value in Bar Charts

Set **Max Value** to set the maximum value the chart should display.

Default: 100

Max Value in Donut Charts

Set **Max Value** to set a maximum value for the chart. Setting the Max Value property is only relevant for donut charts with only one data field. If two or more fields are selected, Max Value is disabled.

Default: 100

> **Example Donut Chart**
>
> If the current value of the data field is 30 and Max Value is set to 100, the donut chart will display two slices: one slice with the value 30 and the other slice with the value 70.

## Max Value in Line, Sparkline, Area, and Multiple Area Charts

Line, sparkline, area, and multiple area charts have a min and a max value property. Set Min Value and Max Value to adjust the range of data displayed in the chart. Data outside this range is cut from the chart.

If Chart Autoscale is used, the min and max values are ignored.

Default: 100

Examples

The following three sparklines have the same size and show the same data. The scaling, however, differs because of different min and max values or Autoscale:
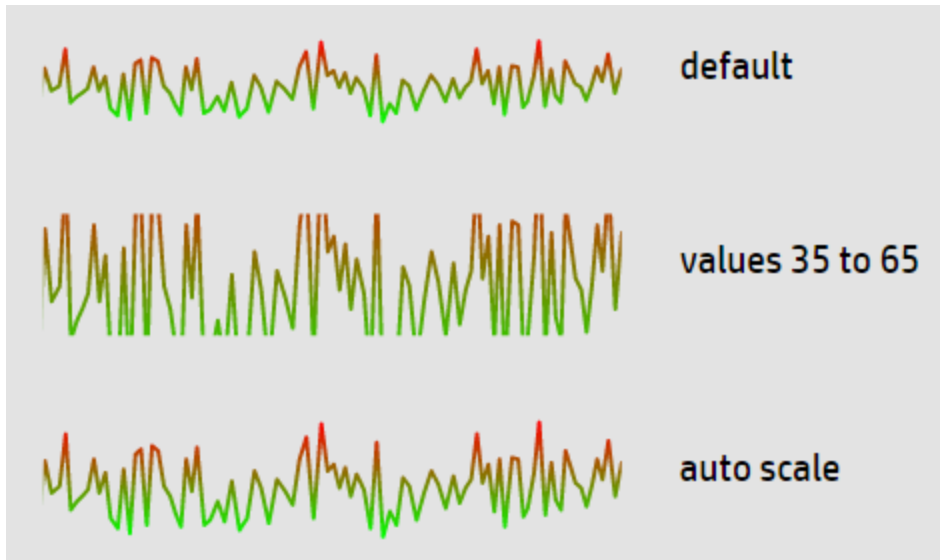

Min Value

Set **Min Value** and **Max Value** to adjust the range of data displayed in the chart. Data outside this range is cut from the chart. If Chart Autoscale is used, the min and max values are ignored.

Default: 0

Examples

The following three sparklines have the same size and show the same data. The scaling, however, differs because of different min and max values or Autoscale:
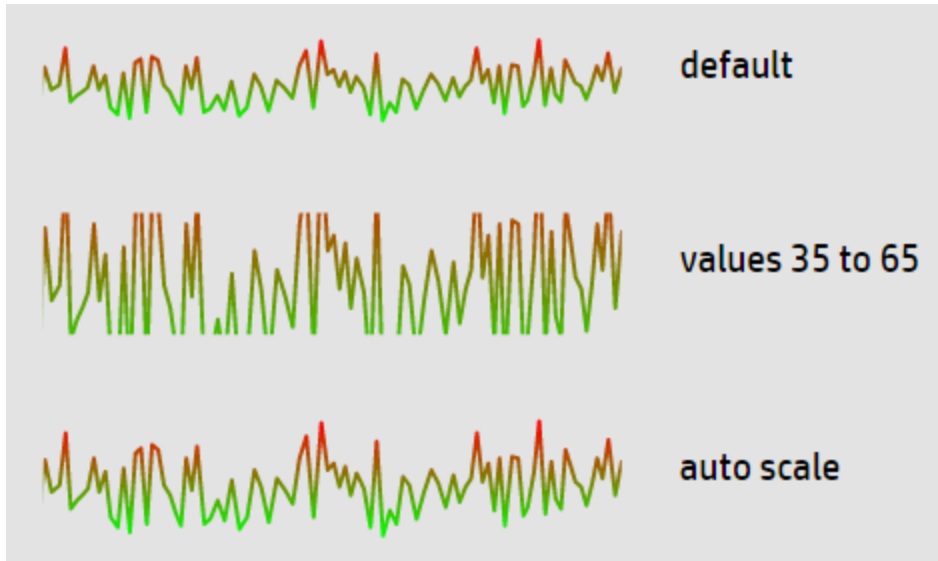
Mouse Over

The **Mouse Over** property enables you to enable or disable the mouse over tooltip. The tooltip offers detailed information on the current data point. The indicator and the tooltip font can be customized in Visio.

Default: selected

Number Format

Use the **Number Format** property to format or manipulate the values displayed below the charts.

Number format works as documented here: http://numeraljs.com/

Number format respects the current locale for formatting.

> **Example**
>
> `'$0,0.00'`
>
> This example changes the number 1000.234 to the string $1,000.23.

Reverse Order of Data Fields

Use the **Reverse the Display Order of the Data Fields** property to change the display order of the donut slices (or data fields) from clockwise to counterclockwise. This property is useful when your donut chart starts at an angle contrary to the natural reading direction.

For example, to create the following gauge-like semicircle donut, set the start angle to 90 and the donut size to 180. The first slice starts at 90 degrees with the data sources organized clockwise. To change the order of the slices to counterclockwise, click the **Reverse the Display Order of the Data Fields** check box in the donut widget properties.
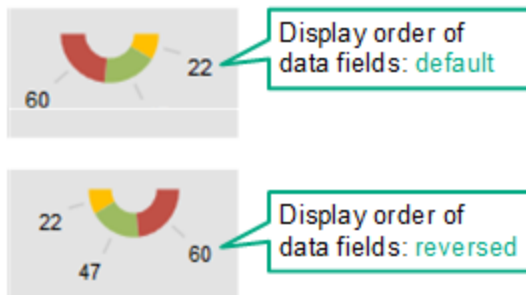
Default: not selected

Show Chart Numbers

The **Show Chart Numbers** property has the following effect:

- **Bar and donut charts.** Shows or hides the numbers.

  You can customize the formatting of the numbers (for example, change color or font) by reformatting the number "1" in Visio.

- **Line, area, and multiple area charts.** Shows or hides x- and y-axes. See also "Showing or hiding

You can customize the x- and y-axes by changing the font and color in Visio.

Default: selected

> **Tip:** Use the **Number Format** property to format or manipulate the values.

Start Angle

The **Start Angle** property determines the location of the first slice in a donut chart. By default, the first slice (or data field) starts at 0 (zero) degrees within the 360 degrees of the circle of the donut chart.

Default: 0 (zero)

Status Field

The **Status Field** property enables you to select a data field in the received data that contains a switch value for the Status Image widget.

By default, BVD assumes that the data includes the data field `status` and uses the values received for `status` to update the color.

> **Note:** Image selection rules override the values received for the Status Field.

Default value: `status` (providing switch values)

> **Example**
>
> The temperature data BVD is receiving from your New York City store already happens to include the `status` data field, so you do not need to select it. Otherwise select the data field that contains your switch values.
>
> Data Field:
>
> type
> element
> value
> status

Transparent Background

Select **Transparent Background** to hide the placeholder shape and show the feed as a transparent overlay.

URL

Specifies the **URL** of the web page you want to show in this widget.

> **Example**
>
> `https://softwaresupport.hpe.com/group/softwaresupport/search-result?keyword=`

For more information, see "Hyperlink" on page 79.

Visibility Rule

The **Visibility Rule** property enables you show or hide the widget based on the outcome of a rule.

> **Tip:** You can also use the Status Visible Group widget to show or hide a widget. Use the group widget when you want to show or hide non-BVD shapes, or to show or hide a large number of shapes. Then it is easier to group the shapes with the Status Visible Group and set the visibility

rule in the group widget. See also "Status Visible Group" on page 63.

Visibility rules have the following format:

```
<property><operator><value>
```

```
<property>
```
The name of the data property to use to calculate the visibility.

```
<operator>
```
The operation that is used to compare the current value of the property with the given value. For a list of operators, see "Rule Operators" on page 53.

```
<value>
```
The value the operator works on.

Example

```
errors>=10
```

If the value of the data field `errors` is greater than or equal to ten, the widget is shown in the dashboard. In all other cases, the widget is hidden.

Tips and Tricks

This section includes:

- "Font usage" on the next page

- "Exporting Visio drawings to SVG" on the next page

- "Showing or hiding x- and y-axes" on page 90

- "Displaying small fonts in Firefox" on page 90

- "Displaying horizontal or vertical lines with gradient line color" on page 90

- "Showing widget tooltips even if the widget is overlaid by another shape" on page 91

- "Inserting Twitter feeds in a dashboard" on page 92

- "Linking dashboards" on page 93

- "Improving loading time of dashboards with raster graphics" on page 93

Font usage

For BVD to be able to render the text as designed in Visio, you must make the fonts used in Visio available to the web browser where you view the dashboards. If the web browser does not have access to the fonts, the system default fonts are used.

For example, if you use the Windows font Calibri in Visio, and then view your dashboard in a browser on a Linux system, the browser will substitute Calibri with a Linux system font because Calibri is not installed.

To enable platform-independent text rendering, use Google Fonts when designing your dashboard drawings in Visio. BVD then directs the browser to load the fonts from http://www.google.com/fonts when displaying a BVD dashboard.

You can also use custom fonts but you must set up a publicly accessible web server that serves the fonts and specify a CSS definition for your custom font in the System Settings page. See "Use Custom Fonts in Your Dashboards" on page 32 for details.

Exporting Visio drawings to SVG

When you save a Visio drawing as an SVG file make sure that the following settings are selected:

- Save as type: **Scalable Vector Graphics (*.svg)**

- Select: **Include Visio data in the files**

- Tip: Press **Ctrl+A** to select everything in the drawing. This ensures that your entire drawing is exported and not the currently selected element only.

Alternatively, click the **Export Dashboard** button in the **Dashboard** ribbon, if you have installed the BVD **Visio Add-in**.

Showing or hiding x- and y-axes

Line, area, and multiple area charts by default show x- and y-axes. You can hide the axes by clearing the **Show Chart Numbers** check box in the widget properties.

The availability of the check box is controlled by the Visio shape data **Show Chart Numbers**, which is by default set to TRUE. If you change this to FALSE and then re-import the exported SVG file, the check box is removed from the widget properties. To re-enable the check box, change the setting to TRUE in Visio and re-import the drawing to BVD.

Displaying small fonts in Firefox

Firefox displays small fonts in SVGs larger than their intended size. To work around this problem, make the original SVG file bigger and let the browser scale it down.
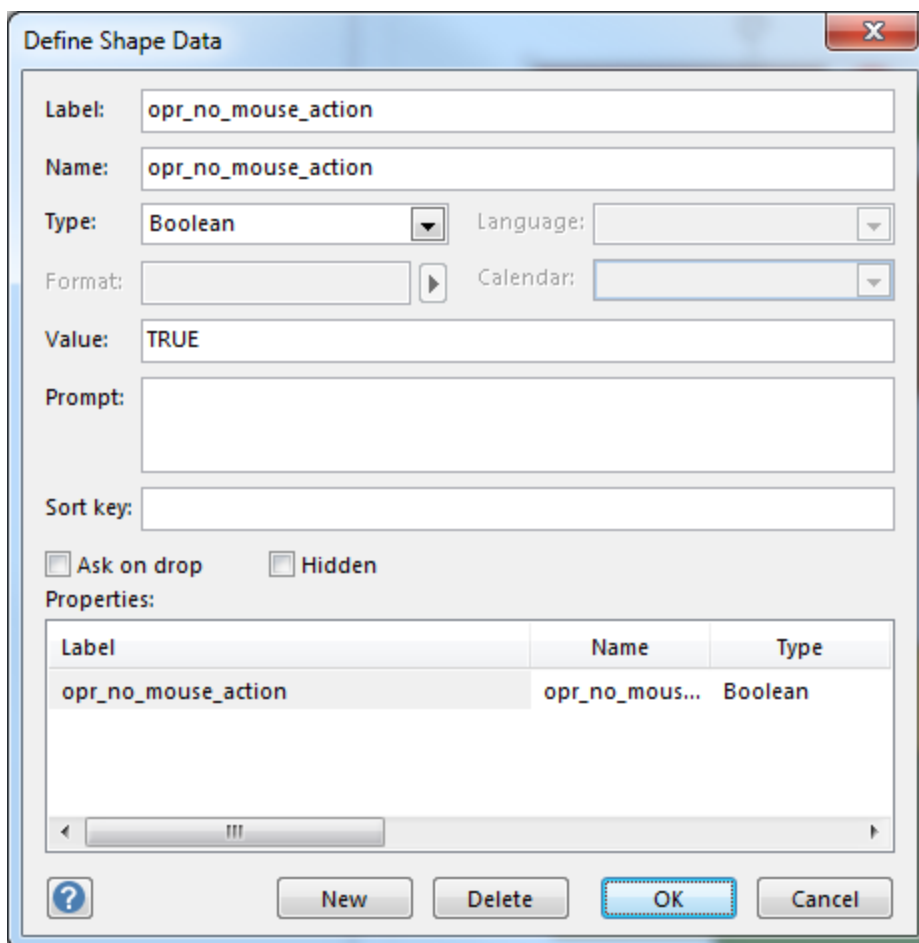
Displaying horizontal or vertical lines with gradient line color

SVG files do not display horizontal or vertical lines with gradient line color. The lines need to deviate from being horizontal or vertical.

Showing widget tooltips even if the widget is overlaid by another shape

> **Note:** To add data to a shape, Visio must be running in developer mode: **File > Options > Advanced > Run in developer mode**.

If you have placed a Visio shape on top of a BVD shape, you can configure the obscuring shape to display the BVD tooltips by adding the shape data **opr_no_mouse_action**:

Set the value of **opr_no_mouse_action** to TRUE. This makes the obscuring shape transparent to the mouse and enables the BVD widget to display tooltips on mouse over.

Inserting Twitter feeds in a dashboard

Although the name suggests it, Twitter feeds cannot be inserted directly in a dashboard using the Feed widget. You would first need to convert the tweets to JSON format and then send the converted tweets to BVD.

The steps below describe an alternative method to include tweets using the Web Page widget:

1. *Prerequisite.* You need a web server that is configured to serve HTML files and allows the inclusion of its pages into the BVD page (X-Frame-Option HTTP Header).

2. Place an HTML file with the following content on the web server:

```
<html>
    <head>
        <style>
            iframe {
                    height: 100%;
                    }
        </style>
</head>
<body style="margin:0">
</body>
```

3. Place the HTML snippet provided by Twitter between the body tags of the HTML file.

4. Add a Web Page widget to your dashboard and set its URL property to the URL of the page located on your web server. For details, see "Web Page Widgets" on page 64.

Linking dashboards

You can link dashboards by inserting any widget and selecting the target dashboard in the **Hyperlink** property. When a user clicks the widget, the linked dashboard opens and replaces the current dashboard in the browser.

If the link should be a simple button, without status updates, use the Text Value widget as follows:

1. Insert a Text Value widget in your Visio drawing, change the default text "Value" to what will be your link, and style the widget as desired. Export the drawing to SVG and then upload the SVG file to BVD.

2. In BVD, edit the Text Value widget:

   a. Do not select a **Data Channel**. This will cause an error, which you can ignore.

   b. Use the **Hyperlink** drop-down list to select the dashboard you want to link to.

3. Save your changes to the dashboard. Then view the dashboard and test the link.

For details, see "Text Value Widgets" on page 63.

Improving loading time of dashboards with raster graphics

Raster graphics images in dashboards increase the size and therefore the loading time of the dashboards. To reduce the size of the images, compress them in Visio before saving your drawing to SVG. In Visio, select the image, then click **Format > Compress Picture**. Increasing the compression reduces the file size but also the quality of the image.

Troubleshoot

This section includes:

- "Enabling additional logging" on the next page

- "Browser limitations" on page 95

- "Verifying metrics forwarding from OMi to BVD" on page 96

- "Notification messages" on page 97

- "Dashboards and Templates" on page 100

Enabling additional logging

BVD log files are located on the configured NFS share, in the directory `bvd/var/bvd/log`.

Available log types are `access`, `audit`, `bvd` and `error`. The logs follow the naming schema `<log type>-<pod name>.log`

- The `access` logs contain information about all requests processed by the server (for example HTML files or graphics). This data can then be statistically analyzed and summarized by another program.

- The `audit` logs contain information about successful and failed user logins.

- The `bvd` logs contain debug information for more detailed troubleshooting.

By default, BVD only logs errors and auditing information.

You can enable additional logging for the BVD receiver, controller, and/or web server. To enable additional logging, complete the following steps:

> **Caution:** Enabling access logging for the receiver will also log the API keys. This might impact the security of BVD as every user with access to the log files will be able to see the API keys.

1. Launch the Management Portal from a supported web browser:

   https://*<external_access_host>*:5443

   <external_access_host> is the fully qualified domain name of the host which you specified as `EXTERNAL_ACCESS_HOST` in the `install.properties` file during the Container Deployment Foundation installation. Usually, this is the master node's FQDN.

2. Access **RESOURCES** > ***<OpsBridge namespace>*** > **Workloads** > **Deployments**

3. For either `bvd-receiver-deployment` or `bvd-www-deployment`, click ⋮ **Actions** > **View/edit YAML**.

4. In the search field of the new window, type `debug` to find the DEBUG entry.

5. Set the value of the DEBUG variable to the log type you are interested in, for example `bvd:error*`

6. Click **UPDATE**.

If BVD is installed with the internal PostgreSQL, you can also view the Redis and PostgreSQL logs using kubectl:

```
kubectl get pods -n <opsbridge namespace>
kubectl logs <pod name> -c bvd-redis -n <opsbridge namespace>
kubectl logs <pod name> -c bvd-postgres -n <opsbridge namespace>
```

Note that directly after the installation of BVD some errors are expected to be logged by the BVD processes and PostgreSQL. They should stop after a few minutes, once BVD is up and running correctly.

Browser limitations

- **Internet Explorer only:** Uploading a dashboard locks the SVG file on disk

  In Internet Explorer 10 or 11 browsers, the upload of SVG files locks the files in the file system. For example, after uploading an SVG file in Internet Explorer, when you then click **Apply** or **Save** in the Manage Dashboards page and try to re-export to the file from Visio, the Visio changes are not saved to the SVG file.

  This is a known problem in Internet Explorer 10 and 11:
  https://connect.microsoft.com/IE/feedback/details/817183/ie10-and-ie11-keep-files-locked-after-uploading

  Workaround: After saving the uploaded dashboard, refresh the Manage Dashboards page, for example by pressing **F5**.

- **Chrome only:** It is not possible to download a dashboard that includes a picture

  In Chrome browsers, downloading a dashboard from the **Administration > Manage Dashboards** page is not working. This affects dashboards that include images of more than 2MB.

  Workaround: Chrome has a limitation of 2MB for data URI downloads:
  https://code.google.com/p/chromium/issues/detail?id=44820

Verifying metrics forwarding from OMi to BVD

Metrics Forwarding from Performance Graphing in OMi 9.2x and 10.0x:

1. In OMi, open Infrastructure Settings:

   Admin > Platform > Setup and Maintenance > Infrastructure Settings

   In the **Applications** list, select **Performance Graphing**.

2. Set the option **Trace Level** to 2.

3. Access the `ovpmtrace.0.txt` file available at the following location:

   Windows: `%ovdatadir%\shared\server\log`

   Linux: `/var/opt/OV/shared/server/log`

4. The log file contains trace messages that indicate that Performance Graphing is forwarding the data to the endpoint.

   The following are samples from the log file:

   ```
   com.hp.pm.core.configuration.PostDataTask:run() -> JSON data to post ...
   com.hp.pm.core.configuration.PostDataTask:postDashboardData() -> Post data  to
   service dashboard endpoint is success
   ```

Metrics Forwarding from Performance Dashboard in OMi 10.1x:

1. In OMi, open Infrastructure Settings:

   Administration > Setup and Maintenance > Infrastructure Settings

   In the **Applications** list, select **Performance Dashboard**.

2. Access the `bvd.log` file available at the following location:

   Windows: `<OMi_HOME>\log\pmi`

   Linux: `/opt/HP/BSM/log/pmi`

3. The log file contains trace messages that indicate that Performance Dashboard is forwarding the data to the endpoint.

   The following are samples (trace level set to INFO) from the log file:

   ```
   com.hp.pm.core.configuration.bvd.PostDataTask:postDashboardData()
   -> BVD - Post data to endpoint is success
   ```

Notification messages

BVD displays notification messages if there are issues with the created dashboards. The following sections describe the messages and solutions.

The data received from the server does not contain {0} but {1} values

Widgets: Bar Chart, Donut Chart

The chart is configured to display {0} values. However, the data sent by the server only contains {1} values. Check the data sent by the server and make sure that it contains all required data fields. Open **Administration > Manage Dashboards** to verify the configured data fields and the data sent by the server in the data channel selector.

There is no valid data channel set

Widgets: Bar Chart, Status Color Group, Donut Chart, Feed, Status Images, Sparkline / Multiple Area Chart, Text Value, Status Visibility Group

There is no data channel set for this widget. Therefore it will not receive any data to display. Open **Administration > Manage Dashboards** to configure a data channel for the widget.

There are no 'path' or 'rect' elements for the dashboard item '{0}'

Widgets: Status Color Group

The Status Color Group is coloring lines and areas of the grouped shapes. If these do not contain lines and areas, coloring will fail with this error. Open the dashboard in Visio and make sure that valid shapes are grouped with this Status Color Group.

The property '{0}' used in the coloring rule is not part of the data. Please check the syntax of the rule. (Dashboard item '{1}')

Widgets: All widgets with coloring/image selection rules

Open **Administration > Manage Dashboards** and verify the rule's syntax: the rule should only use data fields that are included in the data sent by the server.

The coloring rule '{0}' is not valid. Please check the syntax of the rule. (Dashboard item '{1}')

Widgets: All widgets with coloring/image selection rules

Open 🔧 **Administration > Manage Dashboards** and verify that the rule adheres to the format described in "Coloring Rule" on page 75.

The data received from the server is missing either the 'link' or the 'title' field (Feed '{0}')

Widgets: Feed

Make sure that the data sent by the server contains the data fields "link" and "title".

The data received from the server does not contain a status (Status images '{0}')

Widgets: Status Image

The status field configured in 🔧 **Administration> Manage Dashboards** is not part of the data sent by the server. Make sure that the server sends the correct data or the correct status field is chosen.

The data received from the server does not contain a property called '{0}' (Spark line '{1}')

Widgets: Sparkline / Multiple Area Chart

The data field configured in 🔧 **Administration> Manage Dashboards** is not part of the data sent by the server. Make sure that the server sends the correct data or the correct data field is chosen.

There is no valid URL for the dashboard item '{0}'

Widgets: Web Page

There is no URL set for this widget. Therefore it will not display any data. Open 🔧 **Administration> Manage Dashboards** to configure a URL for the dashboard item.

The property '{0}' used in the visibility rule is not part of the data. Please check the syntax of the rule. (Dashboard item '{1}')

Widgets: All widgets with visibility rules

Open 🔧 **Administration> Manage Dashboards** and verify the rule's syntax: the rule should only use data fields that are included in the data sent by the server.

The visibility rule '{0}' is not valid. Please check the syntax of the rule. (Dashboard item '{1}')

Widgets: All widgets with visibility rules

Open 🔧 **Administration> Manage Dashboards** and verify that the rule adheres to the format described in "Visibility Rule" on page 87.

Unable to calculate a color with the data received from the server (Dashboard item '{0}')

Widgets: All widgets with coloring rules

The value sent by the server, together with the provided coloring rule, did not result in a color. Open 🔧 **Administration> Manage Dashboards** to verify the accuracy of your coloring rule and provide a default color as the last entry. For details on defining coloring rules, see "Coloring Rule" on page 75.

Widget type '{0}' is not supported

Widgets: Widget Group

The widget type you placed in your Widget Group is not supported. Not supported widget types are Feed and Web Page widgets.
Edit the dashboard or template in Visio to make sure that a supported widget type has been grouped with the widget group. For more information, see "Widget Groups" on page 64.

Dashboards and Templates

Find solutions and workarounds for dashboard or template limitations that are not covered by the notification messages. For details, see "Notification messages" on page 97.

Visio produces blank SVG files or shapes are missing in the export

Description

When editing a dashboard in Visio, the dashboard is blank or shapes are missing after it is exported as SVG file.

Solution

Visio supports styles for shapes which the SVG standard does not. Having applied one of these styles to a shape can lead to missing shapes or empty dashboards after exporting to SVG. This happens, for example, to all shapes with shadows.

Remove the unsupported style or review the Microsoft Visio documentation for possible workarounds.

Template: values are not available in the instances' Variables section

Description

After uploading a template and sending the required JSON data to BVD, the instances' Variables drop-down list displays the dimension name, but not the available values.

> **Note:** This limitation has been fixed for BVD versions 10.62 and later.

Solution

The tags and dims are not saved for widgets that contain variables. Do the following to reselect the data channel:

1. In the Manage Dashboards menu, hover over the template that does not display the values and click ✎ **Edit**.

2. In the Template Manager, click ✎ **Edit** in the left pane.

3. Select the widget that should display the values of the dimension you want to use as variable for your instances.

4. On the right side, select the corresponding data channel from the drop-down list and click the dimension you want to use as variable. The dimension is highlighted.

5. Click **Save**.

6. In the Template Manager, select any of your instances. The Variable section now lists the available values of your variables. Assign them as intended and click **Save**.

Send documentation feedback

If you have comments about this document, you can contact the documentation team by email. If an email client is configured on this system, click the link above and an email window opens with the following information in the subject line:

Feedback on Administration Guide (Business Value Dashboard 10.62)

Just add your feedback to the email and click send.

If no email client is available, copy the information above to a new message in a web mail client, and send your feedback to ovdoc-asm@hpe.com.

We appreciate your feedback!