QlikView

Web Parts

Version 10 for Microsoft Windows®
First Edition, Lund, Sweden, October 2010
Authored by QlikTech International AB

Content

1 What's New in Version 10

The following list outlines changes and improvements in QlikView Web Parts for SharePoint version 10:

- The new installer simplifies the process of installing and configuring QlikView Web Parts for SharePoint. Manual steps have been eliminated.
- The Datasource Web Part is no longer needed making page layout more intuitive. The QlikView document to use is defined as just another property of any QlikView Web Part.
- A single unified Web Part for all objects makes adding QlikView Web Parts even easier and prevents clutter the Web Part Gallery.
- Simplified properties for QlikView Web Parts makes configuring each QlikView Web Part easier.
- The back end architecture has been altered to take advantage of the services traditionally supplied by the QlikView AccessPoint. This means that:
    - QlikView Web Parts for SharePoint can take advantage of QlikView load balancing when operating against clustered QlikView Servers.
    - QlikView Web Parts for SharePoint can more easily avoid the multi-hop authentication problem inherent in NTLM when the QlikView Server is on a different machine from SharePoint.

2 Version Compatibility

Due to the architectural changes needed to support the new features in version 10 and to allow us to extend the QlikView Web Parts for SharePoint product into the future beyond version 10, QlikView Web Part for SharePoint version 10 is not compatible with any earlier release of QlikView Server (i.e. QlikView Server version 9, 8.5, etc.). Likewise, SharePoint websites that contain version 9 or earlier QlikView Web Parts are not compatible with QlikView Server version 10.

3 Installing QlikView Web Parts

Requirements

1. QlikView Server 10. The Server does not need to be installed on the same machine, but must be reachable over the network.
    1. The QlikView Server must be configured to serve QlikView Ajax content via either the QlikView WebServer or Microsoft IIS (see the QlikView Server Reference manual for more information, also see the QvAjaxZfcPath setting below).
    2. The QlikView Server must be licensed to allow QlikView Web Parts for SharePoint. To verify if a given QlikView server is licensed for QlikView Web Parts for SharePoint usage, verify that the License Enabler File for the QlikView Server has a line that states Web Parts;YES;; - for more information, contact you QlikView Account Executive.
2. Because QlikView Web Parts for SharePoint acts as a fully featured client to a QlikView Server and exposes full QlikView functionality to the end user, each SharePoint user that will consume QlikView content via the QlikView Web Parts for SharePoint must have the appropriate licensing on the QlikView Server (see your QlikView account manager for details).
3. One of the following (installed on the same machine as QlikView Web Parts for SharePoint):
    - Windows SharePoint Services 3.0
    - MOSS2007
    - SharePoint Foundation 2010
    - SharePoint Standard 2010
    - SharePoint Enterprise 2010

Note! Throughout this manual Windows Sharepoint Services (wss) 3.0 is used.

Installation

1. Start the installation program, **`QlikViewWeb Parts_x64.exe`** or **`QlikViewWeb Parts_x86`**.
2. The installation unpacks the files and computes the space needed for the installation. A welcome screen is then displayed. Click **Next** to continue.
3. You now come to the software license agreement. Read it, and click **I accept the terms of the license agreement**, then click **Next**.
4. In the **Customer Information** dialog you specify for whom the installation will be personalized. Click **Next** to continue.
5. The **Destination Folder** dialog shows the default installation path of the program, click **Change** if you want to enter an alternate installation path, click **Next** to accept the path.
6. In the **Default Values** dialog, you enter the path to the **`QvAjaxZfc`** virtual directory on you web server, either QlikView Web Server or Microsoft IIS. Click **Test url** to confirm the path to the directory.
7. The **SharePoint** dialog lets you enter the url to your SharePoint web application where you wish to use the QlikView Web Part. Make sure to enable the **Use Proxy** setting if you have SharePoint and the QvAjaxZfc folder on different computers. If you want an automatic installation, you must check the **Install** check box. Click on the question mark to read more about the stsadmin commands that are executed during an automatic installation.

8. You are now ready to install the program. Click **Next** to install the files.
9. After installation is completed, you can look at the readme file for information on how to configure QlikView Web Parts manually, look at the installation log or just click **Finish** to complete the process.
10. Once the installation is complete, you can continue with the manual installation of the web part.

Post Installation Manual Setup

All files necessary for a manual setup are copied to **C:\Program Files\QlikView\WebParts** during the installation. The solutions file, **QlikViewWebPartsSolution.wsp**, can either be run manually via stsadmin or with the command file, **Install.cmd**, located in the **Setup** folder. The command file runs all the necessary stsadmin commands for the SharePoint web application you selected during installation. You find the contents of the solution file in the **Source files** sub folder. If you want to make changes to the configuration, you must edit the web.config file for the web application, you should also edit webconfig.FDC32F77-D5E3-4010-BFA6-C4EEA3CED089.xml in **C:\Program Files\Common Files\Microsoft Shared\Web Server Extensions\14\CONFIG**. The path may differ slightly depending on which version of SharePoint you are using. The webconfig file in **C:\Program Files\Common Files\Microsoft Shared\Web Server Extensions\14\CONFIG** is a template used when creating new web applications. If you want to update an already existing web application, you can run the stsadmin command **-o copyappincontent**.

The command file adds the following section to the web.config file:

<QlikViewWebParts>

<General>

<add key="QvAjaxZfcPath" value="http://RD-CENTEST1/QvAjaxZfc/" />

<add key="Proxy" value"" />

</General>

</QlikViewWebParts>

QvAjaxZfcPath is used as a default value for the QvAjaxZfcPath property when you add the first QvObject to a page.

Proxy is used if you are running the SharePoint and QvAjaxZfc virtual folder on different computers. If you want to enable the proxy in the future, you must set the value to **/_layouts/Proxy.aspx**.

To install QlikView Web Parts for SharePoint to another web application you can use a copy of **Install.cmd** and edit the **url** parameter for the SharePoint web application.

The **Setup** sub folder also contains files for uninstalling, **Uninstall.cmd**, and upgrading, **Upgrade.cmd**, the QlikView Web Part for SharePoint.

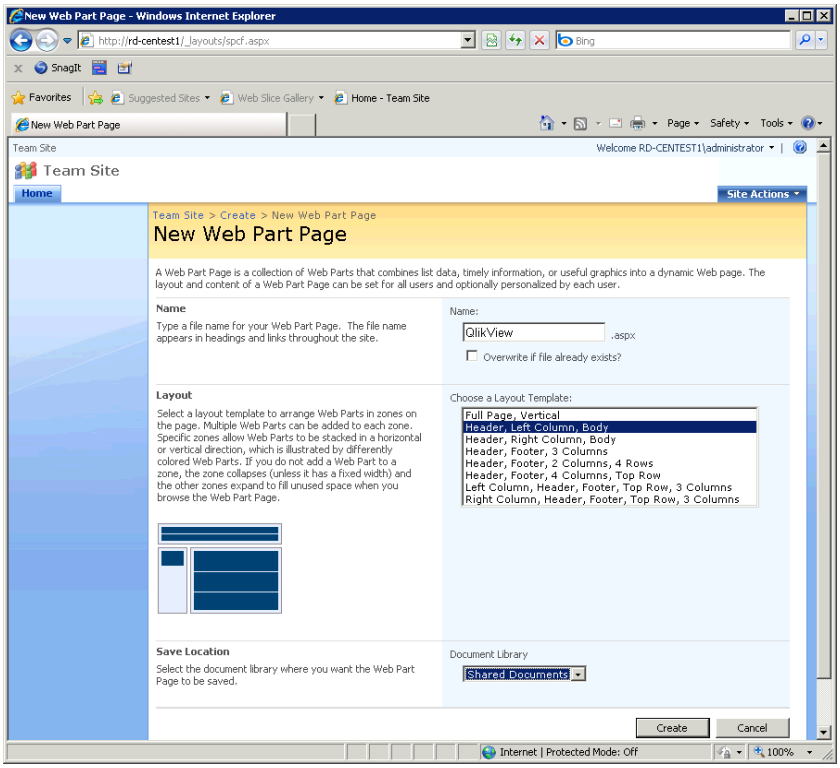
Using Proxy

If you do not have your AccessPoint (the QvAjaxZfx virtual directory) on the same computer as your QlikView WorkBench site, you must use a proxy to avoid cross-site scripting issues. You can use Proxy.aspx for asp.net sites. You can also create your own proxy . When you create your own proxy the following conditions must be met:

- Server request are coming ("escaped") in client request query string "u"
- Cookies are copied from client request to server request
- Headers are copied from server response to client response
- Server response is copied in binary form to client response.

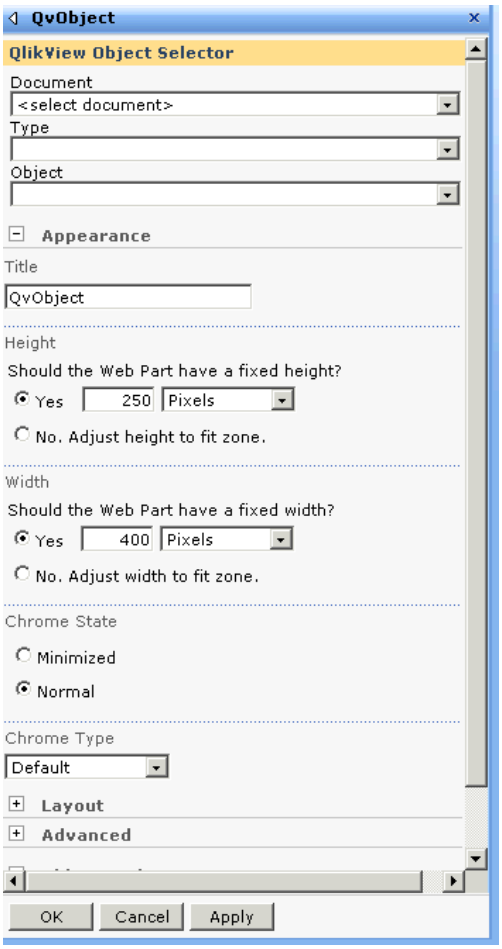
4 Creating a Web Part Page

To create a web part page with QlikView Web Parts, choose **Site Actions** and **Create**. Next, you must choose the layout of your page and set a name for it. Then click **Create**. You now see the page and all the web parts on it.



Click on one of the zones and you will see the following dialog, where you should choose **QvObject**.

Click **Add** and you will be returned to your web parts page where you can edit the properties of the web part. Click on the link in the object to set the properties. The **QlikView Object Selector** will open in a pane to the right.



The **QlikView Object Selector** contains the following settings:

Document	Select a document present on the QlikView Server to connect to. Note that the list is limited to the documents that the user is authorized to see by QlikView Server (See the QlikView Server Reference manual).
Type	Select an object type from the QlikView document. This will filter the list for **Object**.
Object	Select an object from the QlikView document. Note that the list is limited to the objects that the user is authorized to see by QlikView Server (See the QlikView Server Reference manual).

Note! If you cannot see any documents in the **Document** setting, you may need to configure the QvAjaxZfcPath below or you may have the wrong path in the **QvAjaxZfcPath** setting below.

The only other properties in the selector that are relevant to the QlikView object are those under **QlikView Object** and **QlikView Settings**:

InlineStyle

Some styles are set using a stylesheet created by the QlikView Server's Ajax engine. These styles can be overridden in your own custom stylesheet. However, other styles are added to the inline HTML generated by the QlikView Server Ajax engine. These styles cannot normally be overridden in your own custom stlyesheet. However, changing this setting from the default **True** to **False** .will allow these inline styles to be overridden in your custom stylesheet. The styles that are provided inline by the QlikView Server Ajax engine are styles regarding fonts, borders and colors. Read more about the inline styles in "Appendix A" on page 18

Tag

Define you own tag for the object that can be used in, for instance JavaScript. This can be used for customizing a QlikView Web Part, marking a QlikView Web Part for special action, distinguishing between one QlikView Web Part and another at run time, etc. This information can be used to better integrate your QlikView Web Parts with other SharePoint content. Note: The tag is added as an attribute of the QlikView Web Part's **`div`** tag. Here is an example of a function that retries an array of tag attributes from all the QlikView Web Parts on a webpage:

```
<script type="text/javascript">
        GetAllQvObjectsByTag = function(tag) {
                var tagObjs = [];
                var divs = document.getElementsByTagName("div");
                for (var i = 0; i < divs.length; i++) {
                        if (divs[i].getAttribute("Tag") == tag) {
                                tagObjs.push(divs[i]);
                        }
                }
                return tagObjs;
        }
</script>
```

QvAjaxZfcPath

This is the path to the **QvAjaxZfc** virtual directory on the web server that is serving your QlikView Ajax content. For example, use the default **/QvAjaxZfc/** when the QvAjaxZfc is hosted on the same web server that is running SharePoint. However, it may be more common in production environments to have a dedicated SharePoint server and have the QlikView Server virtual directories on another server, in that case use an address like **http://{MyQvWebServer}/QvAjaxZfc/** where {MyQvWebServer} is the DNS name or IP address of the web server serving the QlikView Ajax content. If you are unsure what this path is, contact your QlikView Server administrator. For more information, please see the section "QlikView Web Parts for SharePoint and the QvAjaxZfc Virtual Directory" on page 14 below.

CustomIcons

Enter substitute images for caption icons. The syntax is **icon code:icon url**. The following icons can be replaced by custom icons. Custom icons must use relative paths. Separate icons with comma if more than one.

The icon code precedes the name of the icon:

LS - Lock Excluded

US - Unlock Selected

CA - Clear All

CA.Disabled - Clear All Disabled

SE - Select Excluded

SP - Select Possible

SA - Select All

SEARCH - Search

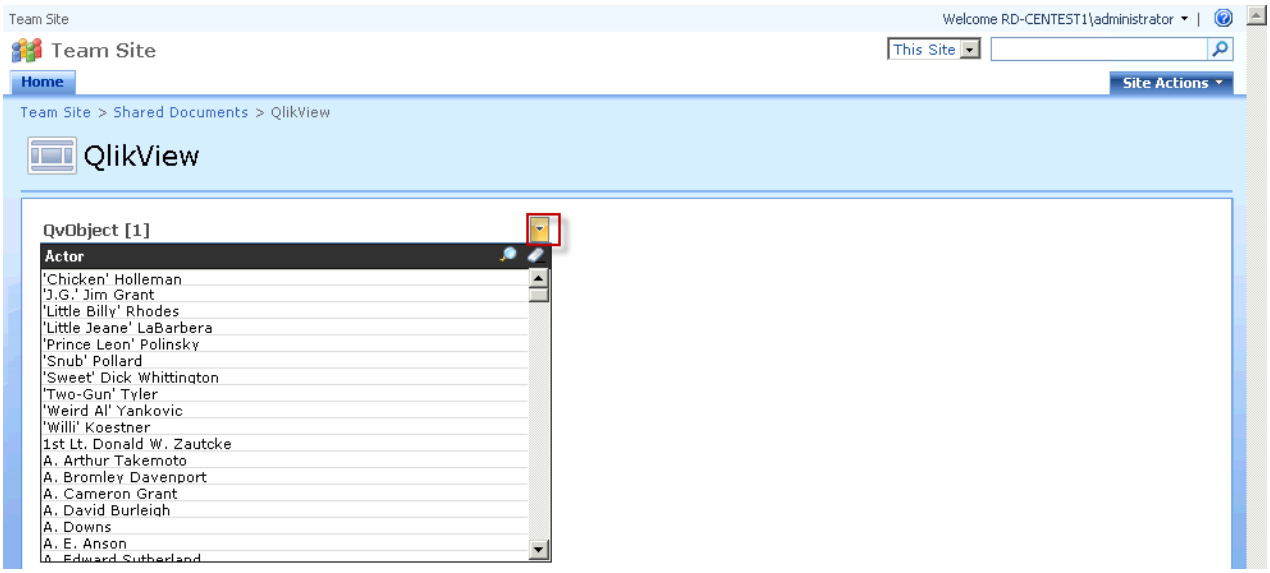
XL - Send to Excel

CD - Clear

PR - Print

HELP - Help

To modify an already created web part, you must click on the small arrow in caption of the object.


Team Site
Welcome RD-CENTEST1\administrator
Team Site
This Site
Home
Site Actions
Team Site > Shared Documents > QlikView
QlikView
QvObject [1]
Actor
'Chicken' Holleman
'J.G.' Jim Grant
'Little Billy' Rhodes
'Little Jeane' LaBarbera
'Prince Leon' Polinsky
'Snub' Pollard
'Sweet' Dick Whittington
'Two-Gun' Tyler
'Weird Al' Yankovic
'Willi' Koestner
1st Lt. Donald W. Zautcke
A. Arthur Takemoto
A. Bromley Davenport
A. Cameron Grant
A. David Burleigh
A. Downs
A. E. Anson

5 QlikView Web Parts for SharePoint and the QvAjaxZfc Virtual Directory

QlikView Web Parts for SharePoint exposes QlikView Ajax content the same as is served up in the QlikView Ajax client. Both the QlikView Ajax Client and QlikView Web Parts for SharePoint rely on the QvAjaxZfc virtual directory to serve the QlikView Ajax content. This directory is a virtual directory on a webserver (either the QlikView WebServer or IIS) and is installed as part of the QlikView Server install package (see the QlikView Server Reference Manual for more information).

This virtual directory can, but does not need to be hosted on the same web server as SharePoint. This leads to a very flexible deployment scenario with many options. Here are a list of options and some pros and cons of each option:

Scenario 1:

Box 1: Both SharePoint and QvAjaxZfc hosted on the same IIS

Box 2: QlikView Server

Pros: Prevents any multi-hop authentication issues, only a single web server

Cons: May not be suitable where SharePoint administrator wants SharePoint dedicated to portal only and all content processing must happen on other machines.

Notes: Good when QVS will be dedicated to serving content to SharePoint, otherwise non-SharePoint users consuming vanilla QlikView content will use some resources on the SharePoint IIS machine. Also requires that SharePoint administrator allows processing QlikView Ajax content on the IIS that is running SharePoint.

Scenario 2:

Box 1: SharePoint on IIS

Box 2: QlikView Server and QvAjaxZfc (hosted on either QlikView Web Server or IIS)

Pros: Dedicated SharePoint box

Cons: May encounter muti-hop authentication issues where NTLM is used between client and SharePoint (see next section). Must use proxy.

Notes: Good when the QVS will service both SharePoint users and other vanilla QlikView client users. Off-loads all QlikView Ajax content processing off of the SharePoint box. See next section for more information on the multi-hop authentication issue.

Scenario 3:

Box 1: SharePoint hosted on IIS

Box 2: QvAjaxZfc hosted on either IIS or the QlikView WebServer

Box 3: QlikView Server

Pros: Both QVS and SharePoint have their own dedicated boxes.

Cons: Requires three machines. May encounter muti-hop authentication issues where NTLM is used between client and SharePoint (see next section). Must use proxy.

Notes: Only recommended in very high volume environments when heavy loads are expected on both the QVS and SharePoint machines and therefore they both require their own dedicated machines.

Scenario 4:

Box 1: SharePoint and QvAjaxZfc both hosted on IIS and QlikView Server

Pros: Requires only one machine.

Cons: QlikView Server is not on a dedicated machine. Not recommended by QlikTech.

Notes: QlikTech recommends that QlikView Server always be on a dedicated machine. This configuration may have QlikView Server and SharePoint fighting for resources (i.e. RAM, CPU time) and should be avoided except for demo, development, or evaluation environments where no real load will be placed on the system.

Other Notes:

It is possible to have SharePoint hosted on IIS and QvAjaxZfc hosted on QlikView Web Server both on the same box (either using port sharing or running on separate ports), but this configuration would usually not make much sense. However, it is possible it could prove useful in an environment with highly specialized requirements.

6 QlikView Web Parts for SharePoint and Multi-hop Authentication

The issue of multi-hop authentication is not QlikView specific. It is a known limitation of Microsofts NTLM authentication mechanism. The basic issue is that NTLM, due to its very nature, can only achieve one hop. When this one hop is from a client to a server (for example from an end user's machine to a SQL Server machine) then there is only one hop and there is no problem. However, when you add an intermediary server, such as a portal server (i.e. SharePoint) you have two hops (for example from the end user's machine to the SharePoint machine, and from the SharePoint machine to the SQL Server). What happens in this case is the user's identity gets moved from their client machine to the SharePoint server via NTLM, but NTLM does not support a second hop (so the user in our example comes into the SQL Server not as an authenticated user, but as anonymous).

When using QlikView Web Parts for SharePoint is possible to encounter the multi-hop authentication issue when the QvAjaxZfc directory is on a different machine from SharePoint (Scenarios 2 and 3 in the section above). In this case the user will come into the QlikView Server as anonymous even though the user is authenticated as far as SharePoint (and the IIS SharePoint is running on) is concerned.

In order to use a configuration that has multiple hops (i.e. either Scenario 2 or 3 from the section above) it is necessary to avoid NTLM and configure Kerberos instead. When using Kerberos it also must be configured to allow delegation (i.e. the second hop). This procedure is well documented in numerous places, for more information please see the Microsoft Developers Network (MSDN) or consult you Windows administrator.

It is important to note that when the QvAjaxZfc directory is on the same machine as SharePoint (i.e. Scenario 1 from above) the multi-hop issue does not arise even though there are still multiple hops and it seems like you would also see this issue in this scenario. This is due to the fact that the QvAjaxZfc will use QlikView specific authentication instead of NTLM for the hop back to the QVS. This avoids the multi-hop issue altogether. It should be noted that placing the QvAjaxZfc directory on the SharePoint machine will almost certainly take less configuration effort than configuring Kerberos to allow delegation so this method is recommended where possible.

7 Troubleshooting

The QlikView Web Part does not appear in the SharePoint site's Web Part Gallery:

- Verify you are looking at the web site you specified when you installed the QlikView Web Parts for SharePoint.
- If you did not check the **Install** check box, you must install the web part manually. See "Post Installation Manual Setup" on page 7
- Check the SharePoint logs for information about the stsadmin commands that were executed during the installation, C:\Program Files\Common Files\Microsoft Shared\Web Server Extensions\14\LOGS. The path may vary depending on your version of SharePoint.

The QlikView Web Part appears in the Gallery but when it is added to the SharePoint web site the pull down menu is empty in the Document property of the Web Part:

- Verify you can access the standard Ajax client from a browser on the same machine as the SharePoint web site. If this does not work, it is an issue with the QlikView Server. Please consult your QlikView Server Administrator or the QlikView Server Reference Manual for more information.
- Verify that the QvAjaxZfcPath property is set correctly in your Web Part.
- Verify that the current user is being transferred to the QlikView server correctly. Check the QlikView server logs. Also it may be helpful to set up a document authorized to be access by the anonymous account on the QVS (this is done differently for NTFS authorization and DMS authorization – see the QlikView Server Reference Manual for details). If the document now appears in the menu then there is an authentication issue between the SharePoint server and the QVS. Typically a SharePoint site will be using Windows Integrated Authentication so look for something that may be preventing this from functioning. Consider the multi-hop authentication issue inherent in NTLM (see your Windows administrator for more information).

The Web Part can be added correctly and the QlikView Document property can be set, but trying to set the QlikView Object property throws an error.

- Typically this indicates that the QlikView document in question has section access that requires entering a user ID and password. The workaround here is to have the QlikView Document administrator disable the section access while the Web Part is added to the web page and re-enable it again afterwards.

8 Appendix A

Inline Styles

Some styles are set using a stylesheet created by the QlikView Server's Ajax engine. These styles can be overridden in your own custom stylesheet. However other styles are added to the inline HTML generated by the QlikView Server Ajax engine. These styles cannot normally be overridden in your own custom stlyesheet. However, changing this setting from the default True to False will allow these inline styles to be overridden in your custom stylesheet.

The styles that are provided inline by the QlikView Server Ajax engine are:

fontfamily, fontsize, fontstyle, fontweight, textalign, verticalalign, textdecoration, paddingTop, paddingLeft, paddingRight, paddingBottom, color

background-color, color, text-align, font-style, font-weight, text-decoration, font-size, border-bottom, border-top, border-left, border-right

MozBorderRadiusTopleft, MozBorderRadiusTopright, MozBorderRadiusBottomleft, MozBorderRadiusBottomright

WebkitBorderTopLeftRadius, WebkitBorderTopRightRadius, WebkitBorderBottomLeftRadius, WebkitBorderBottomRightRadius

Logging

Errors are automatically logged to QlikView WebParts Errors.txt, located in your Share Point application directory. If you want to extend the logging, you can add the value **`LogFile`** in **`web.config`**:

```
<QlikViewWebParts>
<General>
<add key="QvAjaxZfcPath" value="http://RD-TSG-MOSS/QvAjaxZfc/" />
<add key="Proxy" value="/_layouts/proxy.aspx"/>
<add key="LogFile" value="/wplog.txt"/>
</General>
</QlikViewWebParts>
```