

Understanding the Relationship between QlikView 10 Extension Objects and the QlikView 10 Workbench

Overview

The introduction in QlikView version 10 of Extension Objects has changed the ways that the QlikView Workbench product will be used. This document will discuss these changes, describe the benefits of the new architecture, and clarify the usage of the QlikView Workbench product in QlikView 10.

Discussion

Historically, the QlikView Workbench version 9 and prior was mainly used to accomplish two very different things, namely:

- 1) **Deployment:** QlikView Workbench 9 and prior allowed deploying content via ASP .Net websites or web applications.
- 2) **Custom Visualizations and Mash-ups:** QlikView Workbench9 and prior allowed the creation of custom visualizations and allowed a developer to have this custom QlikView content interact with other web based content.

However it is important to note that in order to get the ability to do #2 (custom visualizations and mash-ups), in version 9 and earlier it was a requirement to do #1 (deploy via an .ASP .Net website or web application). This was a significant limitation to those customers who were happy with deploying QlikView through the traditional means (i.e. the plug-in client or the AJAX client) but still wanted custom visualizations.

For example, many customers wanted the ability to show custom visualizations in their QlikView applications, but lacked either the technical infrastructure or desire to change their entire QlikView deployment over to a .Net based website.

This issue was solved in QlikView 10. In QlikView 10 we introduced Extension Objects. These are normal QlikView objects that render in the QlikView AJAX client (as well as the plug-in and QlikView Desktop when running in WebView mode). The only difference is that they give a developer an entry point to add their own custom rendering code (using our new JavaScript API to interact with the QlikView AJAX engine). Thus you can now display custom visualizations in the Native QlikView clients without the requirement to deploy via a .Net website.

That means that now in QlikView 10 and beyond, the QlikView Workbench is really only used for use # 1 above (because you want or need to deploy via a .Net website or web application). The net effect is that the QlikView Workbench will be needed somewhat less often in version 10 than it was in v9, because the use case of showing custom visualizations is now covered by the 'core' QlikView product.

As far as creating QlikView 10 extension objects, the only tool you really need is a text editor (e.g. notepad). However, in the QlikView 10 Workbench we have added a Visual Studio® template project to make building QlikView extension objects easier. The use of this template (and thus the requirement to have access to the QlikView Workbench) is in no way mandatory. As stated above, it is entirely possible to create QlikView extension objects without this template. In fact many great QlikView extensions have been created using notepad.

However, the new template does make this task easier, so some customers may still want to use the QlikView workbench, not for deploying via a .Net website, but rather solely for the use of this extension object template.

Summary

I Want To...	Using QlikView 9	Using QlikView 10
Deploy Standard QlikView Objects Embedded in an ASP.net website or web application	Use the QlikView 9 Workbench.	Use the QlikView 10 Workbench.
Use a Visualization Not Provided by Standard QlikView Objects	Use the QlikView 9 Workbench, but the limitation is that the project must be deployed via an ASP .Net website or web application.	Use a QlikView 10 Extension Object (Can be deployed via IE Plug-in in WebView mode, QlikView AJAX Client, or in a QlikView Workbench website).
Use a Microsoft Visual Studio® project template to help me get started quickly and easily building Extension Objects.	N/A - Extension Objects do not exist in V9.	Use the Extension Object template provided in QlikView Workbench 10.