



Reference Manual

Version 8.5 for Microsoft Windows®

First Edition, Lund, Sweden, June 2008

Authored by QlikTech International AB PJB/CEN

Tableau EX1009

IPR2024-01388

U.S. Patent No. 9,712,595

Copyright © 1994-2008 Qlik®Tech International AB, Sweden.

Under international copyright laws, neither the documentation nor the software may be copied, photocopied, reproduced, translated or reduced to any electronic medium or machine-readable form, in whole or in part, without the prior written permission of QlikTech International AB, except in the manner described in the software agreement.

Qlik®Tech and Qlik®View are registered trademarks of QlikTech International AB.

Microsoft, MS-DOS, Windows, Windows NT, Windows 2000, Windows 2003, Windows XP, Windows Vista, SQL Server, FoxPro, ~~Excel~~, Access and MS Query are trademarks of Microsoft Corporation.

IBM, AS/400 and PowerPC are trademarks of International Business Machines Corporation.

Borland, Paradox, Delphi and dBASE are trademarks of Borland International.

ORACLE and SQL*Net are trademarks of Oracle Corporation.

MacOS is a trademark of Apple Corporation.

CONTENT

- 1 INTRODUCTION 7**
 - 1.1 About this documentation 7
 - 1.2 Documentation outline 8
- 2 VERSION HISTORY 9**
 - 2.1 What's new in version 8.0 9
 - 2.2 What's new in version 8.01 12
 - 2.3 What's new in version 8.2 13
 - 2.4 What's New in version 8.5 16
 - 2.5 Migration Considerations 18
- 3 QLIKVIEW SERVER - QVS 23**
 - 3.1 System Requirements 23
 - 3.2 Upgrading the QlikView Server 26
 - 3.3 Installing the QlikView Server 28
 - 3.4 Completing the Installation 33
 - 3.5 Register 34
 - 3.6 Start Service 35
 - 3.7 Set Default Security in MS IIS 35
 - 3.8 Testing the QlikView Server Installation with the Sample pages 35
- 4 QVS LICENSING 43**
 - 4.1 Client Access Licenses (CALs) 43
 - 4.2 Types of CALS 43
 - 4.3 Combining different types of CALs 44
 - 4.4 Cluster Licensing 44
 - 4.5 QlikView Server Small Business (SB) Edition 44
- 5 SERVER MANAGEMENT CONSOLE PARAMETERS 47**
 - 5.1 The QlikView Management Console 47
- 6 QLIKVIEW SERVER 8 SECURITY SET-UP 81**
 - 6.1 Communication encryption 81
 - 6.2 File system security on server 81
 - 6.3 File system security vs. QlikView Section Access security 85
 - 6.4 Security Configurations 86
- 7 QLIKVIEW SERVER FUNCTIONAL ARCHITECTURE 93**
 - 7.1 QlikView Server – Client Communication 93
 - 7.2 QlikView Server Tunnel 97

8	SERVER LOGGING	101
8.1	Logging from QlikView Server	101
8.2	The Session log	101
8.3	The Performance log	103
8.4	The Event log	105
9	QLIKVIEW SERVER BUILT-IN HTTP WEB SERVER	107
9.1	Starting the QVS built-in http web server	107
9.2	Displaying sample pages with the QVS built-in http web server	107
10	SERVER REPOSITORY FOR SHARED OBJECTS	111
10.1	Types of Objects Available for Sharing	111
10.2	Settings required for Collaboration	111
11	DOCUMENT METADATA SERVICE (DMS)	115
12	QLIKVIEW SERVER LOAD SHARING (CLUSTERING)	117
13	SUMMARY OF QLIKVIEW CLIENTS	121
14	ABOUT THE SAMPLE PAGES	123
14.1	Sample HTML Pages	123
14.2	QVS AccessPoint	125
15	QLIKVIEW JAVA CLIENT	127
15.1	Java Client	127
15.2	Display of application	127
15.3	Collaboration - Shared Objects	128
15.4	HTML Pages	128
15.5	Using custom HTML to invoke applet (Advanced)	128
16	QLIKVIEW JAVA OBJECTS CLIENT	131
16.1	Java Objects Client	131
16.2	QlikView Page Generator for Java Objects	132
16.3	Objects definition	133
16.4	Applying QlikView Java Objects to a web page	133
16.5	Language and bookmark settings	134
17	QLIKVIEW ANALYZER FOR INTERNET EXPLORER	137
17.1	Plug-in Client	137
17.2	Collaboration - Shared Objects	138
17.3	Deployment of QVA for IE	139
17.4	Customizing settings for QVA for IE	139
18	QLIKX - PUBLISHING SEPARATE SHEET OBJECTS FROM THE QVA FOR IE PLUG-IN CLIENT 141	
18.1	Technical description of the QlikX concept	141

18.2	Limitations	141
18.3	Getting it to work	141
18.4	QlikView Page Generator for QlikX	142
18.5	Capabilities, differences and limitations	145
18.6	Complete sample page	145
19	QLIKVIEW INSTALLED WINDOWS CLIENTS	149
19.1	Locally installed Windows Client	149
19.2	Open in Server	149
20	THE QLIKVIEW AJAX ZERO-FOOTPRINT CLIENT (ZFC)	153
20.1	General	153
20.2	QlikView Page Generator for AJAX ZFC	154
20.3	Collaboration - Shared Objects	156
20.4	Getting it to work	156
20.5	Using section access with QlikView AJAX ZFC	158
20.6	Capabilities, differences and limitations	162
20.7	ASP timeouts for very large QlikView documents	164
21	CONSIDERATIONS WHEN DEVELOPING DOCUMENTS FOR USE WITH QLIKVIEW SERVER 165	
21.1	General	165
21.2	General limitations when working from clients	165
21.3	Limitations in Java clients layout functionality	165
21.4	Font Mapping in Java clients	166
21.5	Performance Considerations	166
21.6	Considerations when using macros in QV documents on the QlikView Server	167
21.7	Security and Access restriction	170
22	JAVA APPLET API	175
22.1	General provisions for the API	175
22.2	Documentation and examples	175
23	HTML PAGE SCRIPT API	177
23.1	General Specifications for the script API	177
23.2	General provisions	177
23.3	Error handling	177
23.4	Provisions for examples	177
23.5	QlikView Java/QlikView Java Objects script API function summary	178
23.6	Function descriptions	179
1	APPENDIX A - QLIKVIEW JAVA FAQ	197
2	APPENDIX B - QLIKVIEW PLUG-IN FAQ	199

3 APPENDIX C - MS WINDOWS INSTALLER - MSI PACKAGES 205

4 APPENDIX D - DEPLOYING MSI PACKAGES WITH GROUP POLICIES 213

5 APPENDIX E - QLIKVIEW SERVER SCRIPT API FOR DMS AND CALs 221

PART I: QLIKVIEW SERVER

- **Introduction**
- **Version History**
- **System Requirements**
- **Installation**
- **Control Panels**
- **Security set-up**
- **Functional Architecture and Tunneling**
- **Server Repository**
- **Document Metadata Service (DMS)**
- **Load Sharing (Cluster environments)**

1 INTRODUCTION

1.1 About this documentation

General

QlikView Server provides a platform for hosting, distributing, and sharing QlikView information over the Internet/Intranet. QlikView Server is tightly integrated with QlikView and QlikView Publisher to deliver a seamless suite of data analysis technology to end users. The server component of QlikView Server is the centerpiece of this technology, supplying a robust, centrally managed, QlikView document community, connecting multiple users, client types, documents, and objects within a secure and safe environment.

There are multiple client types available to connect to QlikView Server. There are the installed Windows clients - QlikView Developer, Professional, Analyzer+, and Analyzer. There is a Sun Java client that can be implemented as either a full or object based client (Java Client and Java Objects Client) for QlikView analysis in a standard web browser. There is also an ActiveX Internet Explorer plug-in client which can also be implemented as either a full or object based client (QlikView Analyzer for Internet Explorer Client – QVA for IE - and QlikX Objects Client) for analysis in an Internet Explorer browser. Finally, there is an AJAX Zero-Footprint Client (ZFC) that provides QlikView Objects support in a standard browser without requiring client side installation. Nothing apart from a standard web browser needs to be installed on the client machine.

With the installed QlikView (exe) clients and the QVA for IE ActiveX plug-in client, whole QlikView documents can be shown with complete sheet layout and more or less 100% fidelity to how the document would look if opened as a local qvw file in QlikView. These clients can be used to create and maintain new sheet objects that can be shared with other users of the document through QlikView Server. As of version 8.5, the AJAX Client can also be used to create and maintain shared objects. QlikView user collaboration is a major theme starting with QlikView version 8, and will continue to be a focal point of development as QlikView Server continues to evolve.

The QlikView Objects Clients (Java Objects, QlikX, and AJAX) are based on the concept of placing individual sheet objects from a QlikView document in an HTML environment. This gives the web designer the freedom of placing single objects in arbitrary places on a web site and integrating them

with text and other HTML code. As of QlikView version 8, the basic HTML code to display these objects in any of the object clients can be automatically generated through QlikView.

This documentation provides the necessary steps to complete the installation of the QlikView Server, test your installation, and share your QlikView documents. The documentation also describes how to configure and monitor QlikView Server through the Management Console, how to connect to QlikView Server and its documents through different clients, and how to access the data in QlikView documents via an API.

Programming languages

The examples in this documentation are written in HTML and ASP using JavaScript. For the server-side, Microsoft JScript is used and for the client-side standard JavaScript is used. This document is however not a manual for these scripting languages. The reader is expected to possess a basic knowledge of HTML, ASP and JavaScript programming.

Style coding

In this documentation all menu commands and dialog options are shown in **Arial bold**. All file names and paths are shown in **Courier Bold**. Sample code is shown in **Courier** and **Courier Bold**.

1.2 Documentation outline

Part I of this document describes the QlikView Server component, how to install it, configure it, start it and monitor it.

Part II describes the different client types. It also provides some examples of how to deploy QlikView documents.

Part III provides general API specifications for the QlikView Java and QlikView Java Objects clients for QlikView Server.

2 VERSION HISTORY

2.1 What's new in version 8.0

One Product Suite

With the release of QlikView 8 we want to emphasize the merging of QlikView, QlikView Server and QlikView Publisher into one integrated product suite. All components now share the same version number. A number of features have been added to make the components feel as one integrated unit.

New QlikView Server Management Console from inside QlikView

By entering a URL in the Locations folder of User Preferences the QlikView Server administrator can have a shortcut to the new HTML based QlikView Server Management Console from the QlikView menu. The Management Console will appear as an HTML window inside QlikView.

Direct Access to QlikView Server Document Folders

Folder locations for QlikView Server documents can be defined in the **Locations** page of **User Preferences** in QlikView.

Page Generator for Object Based Clients

A page generator for the object based clients (AJAX zero-footprint, Java Objects and QlikX) has been integrated into QlikView. The wizard driven generator will take any QlikView document and produce a corresponding HTML page layout for each sheet in the document, customized for any of the client types listed above. The resulting page(s) can be previewed in an HTML window inside QlikView. The pages produced may be used directly or edited further in any HTML design tool.

QlikView Server and Clients

New features in QlikView Server extend the functionality of the server component and enhance the user experience for the clients.

Graceful document refresh

When using QlikView as a client to QlikView Server, it is now possible to refresh your session with new data (as the server document is updated) without losing session or selection state. This is accomplished through a combination of settings in QlikView Server and in the QlikView Document.

New Java clients

The QlikView Java clients have been completely re-written to make full use of the possibilities in Sun Java 1.5. The result is a client close to pixel-perfect in behavior compared to the IE plug-in client. New features include better font-handling, support for transparent objects, color gradients, and object borders with rounded corners. Additional compatibility is expected as version 8 matures. The following is a list of enhancements found in 8.0:

- Slider/calendar objects are supported
- Bookmark objects are supported
- Line/Arrow objects are supported in the QlikView Objects Java client
- Gauges and pictures in table cells are supported
- Transparent or semi-transparent backgrounds in charts are supported
- Rounded corners on sheet object borders are supported
- Borders are available in QlikView Java Objects
- Advanced sheet object borders are supported

Improved QlikView AJAX Zero-Footprint Client

The work continues to expand the functionality of the AJAX client and also to improve performance. Better perceived performance is expected mainly to asynchronous update of straight tables, table boxes and list boxes. Other new functionality includes support for line/arrow objects, transparent sheet objects (requires Internet Explorer 7) and improved search in list boxes. Finally, support has been added for sorting straight tables and table boxes by clicking in column headers.

New QlikView Server Management Console

The new QlikView server Management Console is fully HTML based. It communicates directly with QlikView Server via a new web service interface. In addition to most of the controls of the legacy exe control panel, it includes monitoring of various aspects of the QlikView Server, e.g. active users, active sessions, memory usage etc. This new control panel will provide the platform for future planned enhancements to the control and monitoring of QlikView Server.

Built-in http server

A basic (trimmed down to the specific needs of QlikView Server and QlikView Publisher) http server has been added to QlikView Server. When running QlikView Server on Windows XP/Windows Server 2003 or later, it will provide an alternative to MS IIS.

Collaboration

One of the major design goals for QlikView 8 is to enhance user collaboration during the QlikView experience. QlikView Server will be the centerpiece of the shared infrastructure required to support this design goal.

Server bookmarks and shared server bookmarks

Bookmarks created when working with documents on QlikView Server can now be stored in a repository file, located in the same folder as the document on the server. Your personal server bookmarks will be available for roaming from any computer where you work as an authenticated user. These bookmarks can always be found in the My Server Bookmarks page of the More Bookmarks dialog. Furthermore, you may flag any of your personal server bookmarks as shared to other users. That means that they will appear on the Shared Server Bookmarks page of the More Bookmarks dialog. Other users' shared bookmarks will be available to you in the same manner. This functionality will be available to Enterprise, Professional and Analyzer users alike, but for now not from the Java and AJAX zero-footprint clients.

User objects and shared objects

It is now possible for an end-user to add sheet objects when working with a document on QlikView Server. Objects may be created in the usual manner but will be stored in a repository file, located in the same folder as the document on the server. Your personal sheet objects will be available for roaming from any computer where you work as an authenticated user. These objects can always be found in the My Server Objects page of the Server Objects dialog (Layout menu). Furthermore, you may flag any of your personal server objects as shared to other users. That means that they will appear on the Shared Server Objects page of the Server Objects dialog. Other users' shared objects will be available to you in the same manner. This functionality will be available to Enterprise and Professional users with exe or IE plug-in client only.

Shared reports

Similar to server sheet objects there will also be the possibility to define personal server reports and share those with other users. Again, this functionality will be available to Enterprise and Professional users with exe or IE plug-in client only.

Mail bookmark - click to invoke

When you find something worth sharing in a QlikView Server document, you may, with just a single command from the menu, send a snapshot to another user. A temporary server bookmark (including layout status informa-

tion) will be created on the server. A mail with a qvp URL encoding the bookmark reference will be generated. You just need to enter recipient name and any text you like to go with the link. When the recipient clicks on the link, the QlikView Server document will be opened and your selections and layout status reapplied. This functionality, of course, relies on the recipient having access to the document on the server and access to relevant parts of data and layout based on security restrictions.

General features

Field level security in section access

The QlikView script section access has been amended with syntax to suppress one or more fields based on log-in identity. Suppressed fields in these documents are completely unavailable to the client.

2.2 What's new in version 8.01

Improved QlikView AJAX Zero-Footprint Client

The AJAX Zero-Footprint Client continues to be enhanced.

- Multi boxes and pivot tables have been optimized for asynchronous update, resulting in better performance characteristics.
- Table styles, such as cell colors, borders etc are now correctly rendered as they would appear in regular QlikView.
- Print and Send to Excel are now supported commands. These are accessed via caption icons.
- Support for caption icons in general has been expanded, and support has been added for almost all caption icons available in regular QlikView.

All these features are automatically supported by the AJAX ZFC page generator.

General optimizations

A number of performance optimizations have been added. Of special significance in this release is improved multi-threading of the internal memory manager. This should lead to significantly higher utilization of multiple processor cores in environments with many users and documents.

2.3 What's new in version 8.2

Extended management options in QlikView Server

The new Document Metadata Service (DMS) in QlikView Server adds a whole new range of management options in the QVS Management Console. When using the DMS (optional) the QVS becomes aware of available documents in a whole new way. The new possibilities include:

- ability to have a document automatically loaded whenever the QVS service starts
- ability to schedule loading and unloading of a document
- ability to limit the number of sessions towards a document

Support for third-party (non-Windows) authentication

QlikView Server now contains provisions for accepting third-party authentication via a trusted web server. QlikView Server can accept and handle tickets from a web-server and thereby recognize an identity authenticated by that web server. The actual authentication is done completely outside the bounds of the QlikView software. The use of third-party authentication requires the use of extended authorization via the DMS (see below).

Extended (DMS) authorization mode

QlikView Server has traditionally relied on Windows NTFS file security for authorizing a user to access a QlikView document. Whereas this behavior remains the default, QlikView 8.2 also introduces an extended authorization mode, based on the new Document Metadata Service (DMS). This mode is compulsory, whenever using third-party authentication (see above). In this mode, the QlikView Server can make use of the Directory Service Connector (DSC) installed with QlikView Publisher (hence also a third-party Directory Service Provider) to resolve group memberships for authenticated users. With the use of extended authorization it also becomes possible to control access to a document for a specified user or user group to specified time intervals.

Improved support for clustered server environments

The support for several QlikView Servers has been improved. Provided that the QVSs share the same document location, there is now complete support for collaboration objects working seamlessly over all servers. The license model has also been extended, allowing several servers in a cluster to share the same CALs.

Session timeouts

Settings have been added to force the termination of server sessions after a specified timeout.

Borrow CAL from server

Users of the Windows QlikView Analyzer (non registered) client can use the credentials of a named CAL on a QlikView Server to open documents locally (Analyzer+ capability). This ability expires 14 days after the last connection to the QlikView server using the named CAL, but is renewed every time a server connection occurs.

Tighter integration between QVS and QV Publisher

The integration between QlikView Publisher and QlikView Server has been further strengthened. QlikView Publisher now recognizes a QlikView Server as a resource. QlikView Publisher also has been extended to communicate with the Document metadata Service (DMS) of the QlikView Server to support operation in conjunction with third-party authentication and extended authorization mode.

AJAX ZFC generation by QlikView Publisher

The new version of QlikView Publisher has the ability to automatically generate AJAX ZFC pages for distributed documents, thus making an AJAX ZFC alternative available via QlikView AccessPoint.

Improved AJAX Zero-Footprint Client

The AJAX Zero-Footprint Client has been further improved. New and improved features include:

- completely new multi box with the same look as in regular QlikView
- support for slider/calendar object
- support for bookmarks (for authenticated users)
- support for drill-down and cyclic groups
- support for cyclic expressions
- support for fast type-change in charts
- support for show conditions on sheets
- support for fixed number of lines in tables
- support for drop-down select in tables
- sort indicators in tables
- selection indicators (LEDs) in tables

- Clear and lock icons in current selections object
- improved layout pixel fidelity to Windows client
- interactive search feedback
- support for Safari 3.0 browser
- support for dynamic label colors (using the caption settings in the document)
- For handcrafted pages, you may now disable inlinestyling

Improved Java Client

The Java Client has been further improved. New and improved features include:

- improved performance
- support for slider object
- tool tips in charts
- support for all relevant caption buttons in all sheet objects
- context menus on cycle and fast type-change buttons
- support for grid style multi box
- fast type change between bitmap charts and table charts
- improved handling of pop-up windows

Improved GUI for collaboration sheet objects

A new collaboration object pane has been added for better overview and easier handling of personal and shared server sheet objects. The pane features informative icons with hover tool tips, giving you information about the available objects. Objects can be dragged from the pane onto the sheet.

Convert user bookmarks to server bookmarks

A utility has been added to convert all old user bookmarks for a specific document to server bookmarks stored in the server repository. This action is recommended for easier maintenance of bookmarks.

Convert user reports to server reports

A utility has been added to convert all old user reports for a specific document to server reports stored in the server repository. This action is recommended for easier maintenance of reports.

2.4 What's New in version 8.5

Improved performance and memory utilization

QlikView and QlikView server have been further optimized for better performance and better memory utilization in multi-user environments.

Scripting interface to DMS

Syntax has been added for scripted access to QlikView Server DMS. This will allow batch update of DMS and CAL data in QlikView Server. For more information, refer to the QlikView Server Script API in the appendix section of this manual.

Further Improvements to the AJAX Zero-Footprint Client

The AJAX Zero-Footprint Client continues to be improved. New and improved features include:

Collaboration object creation from AJAX client

Server settings permitting, it is now possible to create user charts and list boxes from the AJAX clients. This is the first step towards the goal of providing full collaboration object support in this environment.

Anonymous user bookmarks

Previously only authenticated users could create bookmarks via the AJAX client. With this release that capability is extended to all users, authenticated or anonymous.

Context menus

Right-click context menus have been added to the AJAX client. More or less all options found in the thicker client are now also available to the AJAX user.

Toolbar

It is now possible to add a toolbar with the most common commands to the AJAX client application. The option to include the toolbar is available in the Client Page Generator dialog.

Upgraded sheet object drawing

Major improvements have been made in the visual presentation of tables and Current Selection boxes. Support has been added for displaying bitmap buttons and aqua style buttons.

Upgraded sheet tabs

The drawing of sheet tabs has been improved. Support for selection beacons has been added.

Support for input fields

Input fields can now be edited via the AJAX client.

Full support for fast type change in charts

Fast type change now also works between bitmap charts and table charts.

Expression search

Support has been added for search in straight table expression columns.

Frequency in list boxes

Support has been added for the frequency column in list boxes.

Export to CSV

Export to csv has been added as an option for selected objects via their context menus.

Further Improvements to the Java Client

The Java Client continues to be improved. New and improved features include:

Java Beans integration

The Java client now consists of Java beans which can be used for server side and/or client side integration with other Java applications.

Stand-alone capabilities

The Java client can now be run as a stand-alone application outside the browser environment. For this to be useful, an open in server dialog has been added, so that the client can connect to a QlikView Server and open documents. Furthermore the client and QlikView Server 8.5 and later will have cross version compatibility just as the QlikView IE plug-in and the installed QlikView clients.

Graphical updates

A number of improvements have been made in the graphical presentation of the Java client, so that it now supports most features found in the thick clients. Features include an improved tab row, support for the new caption/border style, and support for rounded corners on sheet objects.

Improved tables

Tables in the Java client now support pivoting and moving of columns. Additionally, support has been added for displaying gauges in table cells.

Improved export option

Support for export to csv and copy to clipboard has been added.

Open files from info button

It is now possible to open files by means of an info button.

Simplified licensing

Several licensing changes have been introduced to simplify and expand the licensing options for QlikView Server.

Same price for 32- and 64-bit

The price difference between 32- and 64-bit QlikView and QlikView Server has been removed. All QlikView users now have the choice to use the full power of 64-bit QlikView without extra cost.

User object creation from QlikView Analyzer

The previous requirement for a QlikView Professional or QlikView Enterprise (Developer) license key in order to create personal and shared sheet objects in a QlikView server document has been removed. If you have a valid CAL and are an authenticated user, you may create and share your own sheet objects (server settings permitting).

QlikView Server Editions: Small Business Edition and Enterprise Edition

The QlikView Server Small Business (SB) Edition has been introduced as a new option for small businesses looking for a limited set of functionality in QlikView Server. For more information, refer to the section titled QVS Licensing. There will also be notations throughout this document denoting differences between the editions.

2.5 Migration Considerations

The following considerations apply when migrating QlikView Server from version 7.x to version 8.x, and within 8.x where applicable.

Generate AJAX Pages in 8.5

The AJAX client has undergone major restructuring and extension in version 8.5. AJAX pages generated with previous versions of QlikView should be re-generated with QlikView 8.5 before being used with QlikView Server 8.5.

Legacy Control Panel no longer supported in 8.5

The legacy QlikView Server Control Panel (QVSControl.exe) is no longer supported as of version 8.5. Configuration and communication with the QlikView Server service should be handled through the Management Console for QlikView Server.

New directory structure in 8.5

Some of the default installed paths for QlikView Server components have been changed in 8.5 to provide better Windows compatibility. Primarily, this affects the default QvsDocuments folder and the default sample web pages location.

New directory paths to QlikView AJAX ZFC

The default installed paths to the QlikView AJAX Zero-Footprint client have been changed to more accurately reflect the name of the client.

The default path to the client has changed from:

C:\Program Files\QlikView\Server\QvClients\QvZfp

To the path:

C:\Program Files\QlikView\Server\QvClients\QvAJAXzfc

The virtual directory name for this location is changed to **QvAJAXzfc** for both IIS and the new QVS http server. This requires that the document connector code in existing AJAX HTML must be updated to reflect the new path. Refer to the section titled “The QlikView AJAX Zero-Footprint Client (ZFC)” for more information.

The default path to the AJAX web pages has changed from:

C:\Program Files\QlikView\Examples\QvWebpages\Zfp

To the path (for QVS 8.0 – 8.2:

C:\Program Files\QlikView\Examples\QvWebpages\AJAXzfc

To the path (for QVS 8.5+):

C:\Program Files\QlikView\Webpages\AJAXzfc

Replacement of QlikView AJAX ZFC qvxmlayer (asp, dll)

Avq.htc in 8 has moved from Http-GET to Http-POST (for AJAX standard). This also means that **qvlayer.dll** and **qvxmlayer.asp** are replaced by **QvsComRemote.dll** (common for all http-calls to QlikView Server) and **QvsViewClient.asp** (dedicated to the ZFC protocol). This change will affect existing coding for ZFC applications. See the section titled “The QlikView AJAX Zero-Footprint Client (ZFC)” for more information.

Single JAR file for Java

The Java archive files (*.jar) have been integrated into a single file (**QlikView.jar**) with multiple entry points to support the specific function required.

New Class ID for QlikX

The Class ID has been changed for version 8 of QlikX (plug-in objects).

The Class ID for the document connector for version 7.0 to 7.50 was:

6E1BAAF6-ECB9-4505-86C1-5D04467B02CC

The Class ID for the document connector for version 7.51 to 7.52 was:

C966E50C-534E-44d2-84EE-52AA5C501F84

For version 8, the Class ID for the document connector is:

74DC8438-E36A-40a0-B750-4E2257FA2E41

The Class ID for the object display for version 7.0 to 7.52 was:

2B098AA4-1FBD-4d84-8AE3-E95A74371022

For version 8, the Class ID for the object display is:

54D93DE9-7DA5-4477-A3DA-869ADB41CBBB

Refer to the section titled “QlikX - Publishing separate sheet objects from the QVA for IE plug-in client” for more information.

Cross-release compatibility

QlikView Server version 8 will work seamlessly with QlikView 6 or QlikView 7 clients, and QlikView Server 6/7 with QlikView 8 clients. Feature limitations based on version will, of course, apply.

QlikView Server version 8 is not compatible with QlikView 4 or QlikView 5 clients. QlikView Server (QlikWeb) version 4 and 5 are not compatible with QlikView 8 clients.

Control Panel settings in Registry

QlikView Server version 8 will inherit the same control panel settings from version 7, since it will use the same Windows Registry location:

[HKEY_LOCAL_MACHINE\SOFTWARE\QlikTech\QlikViewServer\Settings 7]

Support for native IA-64 (Itanium) dropped in version 8

QlikView Server version 8 does not ship in a native IA-64 (Itanium) version. The 32-bit version will run on Itanium based machines with Windows, but

performance can be expected to be poor. QlikView Server version 7 is recommended for native 64-bit Itanium support.

New Sample Pages in HTML

QlikView Server version 8 will include sample pages in HTML. The legacy ASP sample pages have been phased out as of 8.5.

QlikView Server Service automatically installed

The QlikView Server version 8 installation will automatically perform the Install Service function for the QlikView Server Windows service. It is recommended to set the proper Recovery option for the QlikView Server service so that the service will restart automatically if it encounters an error. See the Installation section of this document for further information.

Multiple QVS Installations sharing the same license are considered Clustered

If multiple QlikView Server installations share the same license, they will automatically be considered as clustered. Note that this configuration will affect Networks where unauthorized license sharing between test and production environments was configured.

3 QLIKVIEW SERVER - QVS

3.1 System Requirements

In order to successfully install and run the QlikView Server, the following basic requirements must be met by the system:

Hardware and Software

You need a computer with

- 1 GHz (x86 processor) or 1.4 GHz (x64 processor). 2 GHz or faster, with several cores/processors recommended.
- graphics hardware supporting color XGA resolution or better. QlikView Server will use the color settings of the Windows server where it runs when sending charts and other graphics to the client. For best results, the color palette on the Windows server should be set to at least 65,536 colors (16 bit).
- a monitor with a resolution of at least 1024 x 768.
- a mouse or an equivalent pointing device supported by MS Windows.
- (optional) a DVD drive for DVD-based install media only.
- a hard disk with at least 250 MB of free disk space.
- enough primary memory (RAM). 512MB RAM minimum (1 GB RAM or more recommended on x86 systems, 4 GB or more on x64 systems). The server's capacity to publish QlikView documents and the number of users who concurrently can connect to it are strongly related to the amount of RAM available.
- An http server for providing QlikView Java, Java Objects Client and AJAX ZFC solutions to end users (e.g. MS Internet Information Server or the built-in QVS http server). Microsoft IIS or the built-in http server is required when using tunneling, external authentication or NT security with Java clients and whenever the AJAX Zero-Footprint client is used.
- TCP/IP Network.
- Microsoft .NET 2.0.
- Paging File Size defined as 4 GB or greater. For x64 systems, recommendation is 1.5 times the amount of RAM.

Actual requirements will vary, based on system configurations. It is recommended that you work with your local QlikView representative to configure an appropriate hardware platform for your QlikView Server requirements.

Operating System

You need one of the following operating systems:

- Microsoft® Windows 2000™; or
- Microsoft® Windows Server 2000™; or
- Microsoft® Windows Server 2003™; or
- Microsoft® Windows Server 2008™
full support with QvsHttpServer, limited support* with Microsoft Internet Information Server 7; or
- Microsoft® Windows XP™
- Microsoft® Windows Vista™

Note that the built-in QVS http server requires Windows XP or Windows Server 2003 or later.

*Limited support includes:

- Windows Authentication must be enabled on the webpage level for the Qvs-Management to function properly when accessing via the browser (no issues when access via shortcut)
- Listing folders/files with NFS rights set, are not properly listed.
- Tunneling does not work.

Additional Requirements for QlikView Server x64

- Processor supporting the x64 architecture (includes AMD Athlon 64, AMD Opteron 64, Intel Xeon EM64T and Intel Pentium with x64 extension)
- Minimum 1 GB RAM. 4 GB RAM or more recommended.
- Microsoft® Windows XP Professional x64 Edition™; or
- Microsoft® Windows Server 2003 x64 Edition™
- Microsoft® Windows Vista x64 Edition™, or
- Microsoft® Windows Server 2008™, x64. See above for IIS 7 limitations.

Client requirements for installed exe clients

- See reference manual for QlikView

Client requirements for plug-in (QVA for IE)

In addition to the server requirements above, the client must be running a compatible web browser. Client requirements are as follows:

- Microsoft Internet Explorer 6 or higher.
and
- Microsoft® Windows 2000™; or
- Microsoft® Windows Server 2000 ™; or
- Microsoft® Windows Server 2003 ™; or
- Microsoft® Windows Server 2008 ™; or
- Microsoft® Windows XP? ; or
- Microsoft® Windows Vista?

Client requirements for Java clients

In addition to the server requirements above, the client must be running a compatible web browser. Client requirements are as follows:

- The Java Runtime Environment (JRE) must be one of the following:
 - Sun Java 1.5 or later
 - Apple Java 1.5 or later
 - Java 1.5 or later (Sun Java compatible)
- Java and Java Script must be enabled in the web browser.

QlikView Java and QlikView Java Objects clients have been successfully tested with Java enabled versions of the following browsers:

- Under MS Windows:
 - Microsoft Internet Explorer 6 and 7 with Sun JRE.
 - Firefox 1.0.6 with Sun JRE
- Under MacOS X (tested on v. 10.4 "Tiger" only):
 - Firefox 1.0.6 or later

QlikView Java and QlikView Java Objects clients will most probably run on many other environments, including various UNIX versions but this has not been verified by QlikTech R&D. As the number of possible combinations of operating system versions, browser versions and JRE versions is very large, QlikTech cannot guarantee correct operation with all possible set-ups. If some specific combination would be found to suffer from problems, we encourage customers to report back, so that better coverage can be achieved in future releases of QlikView.

Client requirements for AJAX Zero-Footprint Clients

In addition to the server requirements above, the client must be running a compatible web browser.

Client requirements are as follows:

- Under MS Windows:
 - Microsoft Internet Explorer 6 or later
 - Firefox 1.0.6 or later
- Under Linux (tested on Ubuntu Linux only):
 - Netscape Navigator 7.2 or later
 - Firefox 1.0.6 or later
- Under MacOS X (tested on v. 10.4 "Tiger" only):
 - Netscape Navigator 7.2 or later
 - Firefox 1.0.6 or later

Other Mozilla-based browsers should work and the QlikView AJAX ZFC will most probably run on many other environments, including various UNIX versions but this has not been verified by QlikTech R&D. As the number of possible combinations of operating system versions and browser versions is very large, QlikTech cannot guarantee correct operation with all possible set-ups. If some specific combination would be found to suffer from problems, we encourage customers to report back, so that better coverage can be achieved in future releases of QlikView.

3.2 Upgrading the QlikView Server

If you are installing QlikView Server for the first time on a server, you may skip this section, and proceed to "Installing the QlikView Server" for installation instruction. If you already have QlikView Server installed on a server, and would like to upgrade to a more current release, then follow the instructions in this section.

Whether you are upgrading QlikView Server to a new release or a new version, it is helpful to be aware of a few basic practices that will help to insure a successful transition to a new level.

- Utilize the sample HTML pages as examples to create your own code with your own naming conventions. Upgrading to a new release or version will typically overlay existing sample files with new ones. So, if you are not careful, you could lose your work.
- Always be sure to read the ReadMe documentation, if available, prior to installing an upgrade. This will have the most current information available to help you perform a successful migration.

- Be sure you have backup media of the current software, as well as your custom code for back out procedures, if necessary.
- QlikView Server must be stopped to perform an upgrade, so it is best to schedule this procedure for an off time.
- Registration (licensing) information and Settings will be saved by default when the QlikView Server program is removed. They will then be applied to any subsequent install of QlikView Server on that system.

TIP: Document folder information is not preserved, so if you have changed the default locations for document folders through the Management Console, you will have to reenter this information once the new version/release is installed.

Upgrading to a new release of QlikView Server will generally require an **UNINSTALL** of the old release and **INSTALL** of the new release.

For the uninstall of QlikView Server, be sure to perform the following steps prior to running a Windows Remove Program procedure:

- 1 Verify that backup media exists for the current release of QlikView Server and backup all current files associated with QlikView Server (HTML pages, QlikView documents, licensing file, QlikView Server version 8 .share files, etc.)
- 2 If you are already running version 8 of QlikView Server, use the QlikView Server Management Console **Users** tab to determine if there are any active users linked to QlikView Server. You may wish to send out a broadcast message to notify users that the service will be stopping.
- 3 Stop the QVS service through the QVS Management Console (or Windows Services dialog). If using the Server Management Console, on the **Install** page of the dialog, click on the **Stop** button.
- 4 Uninstall the QVS service through the Server Management Console. On the **Install** page of the dialog, click on the **Uninstall** button. Note that this only uninstalls the Windows Service.
- 5 Perform the Windows Remove Program procedure against QlikView Server from the Windows Control Panel.

For client program updates, if applicable, be sure the client computer has no open QlikView Server sessions before applying the update.

Now you are ready to install the new release of QlikView Server. Follow the instructions in the next section.

3.3 Installing the QlikView Server

The QlikView Server installation can be performed off CD media or from a disk file. To install QlikView Server, insert the CD in a drive accessible from the target server hardware.

TIP: It is recommended to install QlikView Server after the web server software (e.g. MS IIS) has been installed and configured.

TIP: If the required Microsoft .NET 2.0 Framework is not installed, it will be included as part of the QlikView Server installation process.

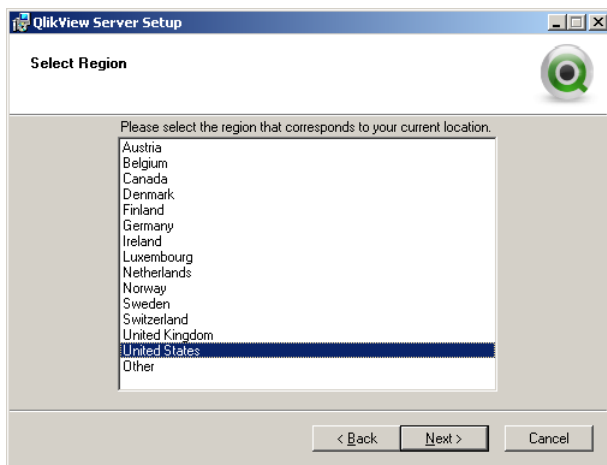
TIP: It is recommended to not move folder locations after QVS installation is complete, since many settings are dependent on their initial file location. If you wish to change the location of QVS after it is installed, this should be done through an uninstall/install process.

If the DVD does not auto-run, or if you are installing from a different media, then execute **d:\AvsX64SetupRedist.exe**, where *d* is the logical drive or path for the target server installation media. Note that 32-bit version have a slightly different naming convention.

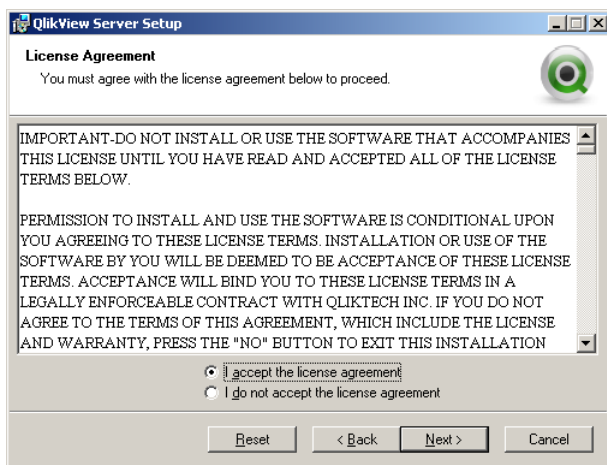
- 1 If the CD does not auto-run, or if you are installing from a different media, then execute **d:\ QVS_SETUP.exe** or **d:\QvsSetupRedist.exe** where d is the logical drive or path for the target server installation media. Note that x64 versions have a slightly different naming convention.



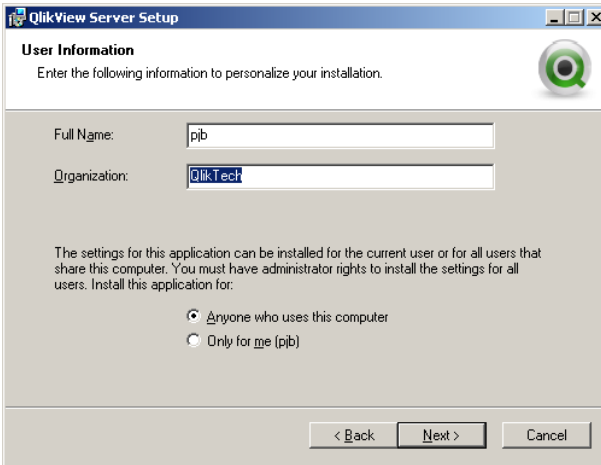
2 Click **Next** to continue the QlikView Server setup.



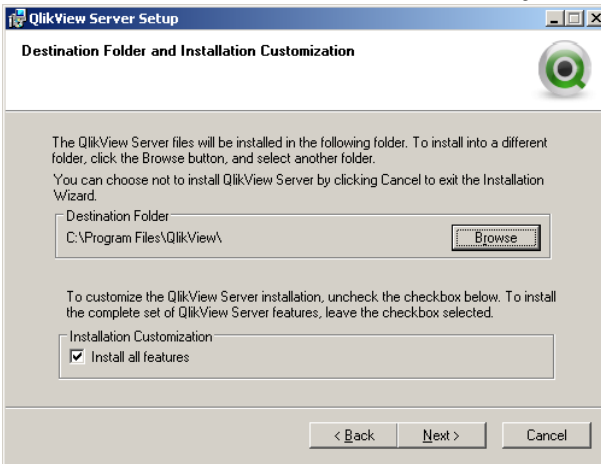
3 Select the region for the local location of the server.



- 4 Read the license agreement, and continue by selecting **I accept the license agreement**, and then click **Next**.



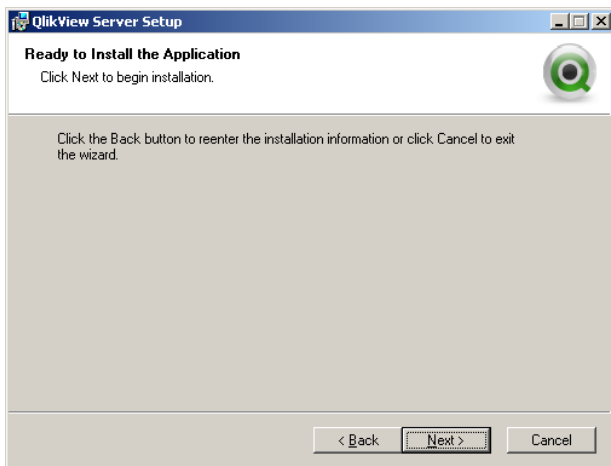
- 5 Enter the user information for QlikView Server.



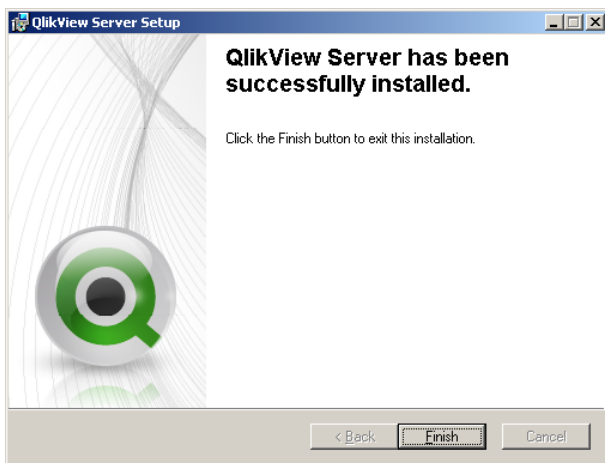
- 6 All files will be installed under the specified folder location. The QlikView Server executable (**qvs.exe**) file will be located in **C:\Program Files\QlikView\Server** by default. If you would like to change the root folder location of the installed files, click **Browse** to specify the preferred location.

- 7 For full functionality, including the example web pages and documents, leave the **Install all features** checkbox selected. This will perform a complete installation (recommended) of the QlikView Server software. To control exactly what is installed, uncheck the checkbox. Continue by clicking **Next**.

TIP: It is recommended to install the QlikView Server example web pages and documents to use to test the installation for new environments.



- 8 Click **Next** to install the application.



- 9 Once the Installation is complete, click **Finish**.

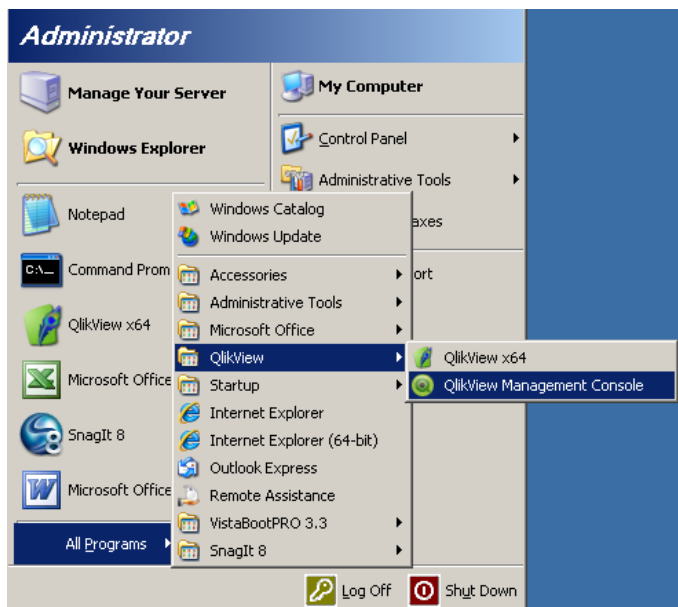
The QlikView Server is now installed on your computer. In order to activate it, you need to complete a few further steps.

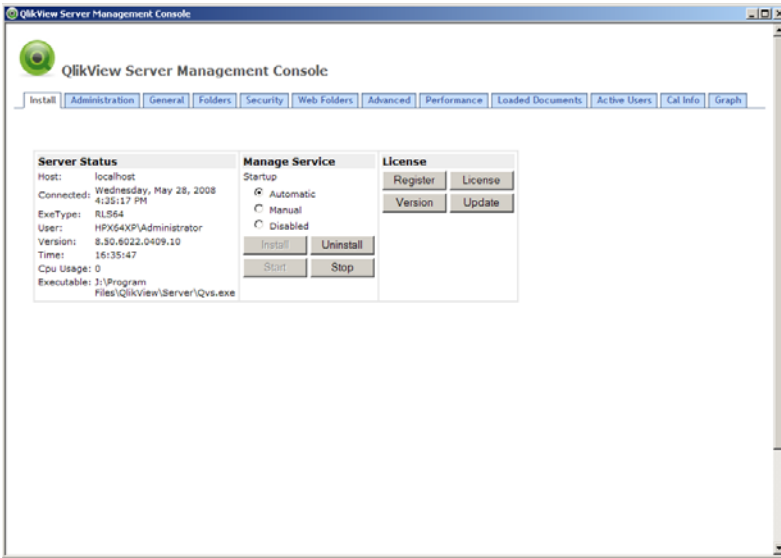
3.4 Completing the Installation

After successfully installing QlikView Server, you must complete the following steps to activate it:

- 1 Register
- 2 Start Service

These steps must be completed after the installation using the QlikView Server Management Console (found under **Programs, QlikView** in the **Start** menu). The topics below explain how to use the Server Management Console.





3.5 Register

Registration authenticates your copy of QlikView Server and allows it to run on your computer. In order to register, you must have a valid Serial Number and Control Number issued by your vendor. If you do not have both a Serial Number and a Control Number, contact your vendor.

TIP: Make sure you enter the serial and control number for QlikView Server. This will be a different license than, e.g., QlikView Developer.

In the QlikView Server Management Console (**Install** page), click on the **Register** button. Enter the Serial Number and Control Number assigned to your copy of the QlikView Server. You should also enter your name and company in the fields provided.

The License Enabler File (lef.txt) for QlikView Server will be automatically written to **C:\Documents and Settings\All Users\Application Data\QlikTech**. If for any reason, the LEF information cannot be accessed through the Internet from your server, you can obtain this information from your vendor, and copy the entire **LEF.txt** file to this location, or paste the LEF data using the **License** button on the **Management Console Install** page. Contact your vendor for specific instructions.

3.6 Start Service

Once the QlikView Server has been registered and installed as a Windows service, it can be started. If the QlikView Server is set for automatic startup, reboot-ing the operating system will start the QlikView Server service. To manually start the service, either the Windows Services control panel or the QlikView Server Management Console can be used.

In the QlikView Server Management Console (**Install** page), click on the **Start** button to activate the QlikView Server service. The service will start and the **Stop** button will become enabled. No dialog will appear indicating successful startup of the service. The **Stop** button can be used to manually stop the QlikView Server service if necessary.

TIP: If the service does not start, and you receive the message “The service did not start in a timely manner” verify that you have a valid License Enabler File (lef.txt), and that the date contained in the LEF has not expired. If you need assistance with licensing, please contact your vendor.

3.7 Set Default Security in MS IIS

If you are using MS IIS as your web server in a Windows Server 2003 or greater environment, be sure to check the following default security settings to insure proper operation of the QlikView Server sample pages, as well as extended functions (e.g. QVS Tunnel).

- Enable ASP Pages
- Enable ASP.NET 2.0
- Add QVSTunnel.dll as a Web Service Extension if you wish to utilize the QVS Tunnel extension (see Chapter 7 for more information on the QVS Tunnel extension)

3.8 Testing the QlikView Server Installation with the Sample pages

When the above steps have been completed, the QlikView Server is installed and running. To test the installation to verify proper operation of the server, you can use the provided sample pages.

If you performed a Complete installation or Custom, and included the QlikView Server example web pages and documents, several sample HTML pages were loaded onto your server during installation in order to allow you to verify proper operation. To access these web pages, the directory where the pages were loaded must be accessible to your web server.

The default folder location for the QlikView Server sample pages is:

C:\Program Files\QlikView\Webpages

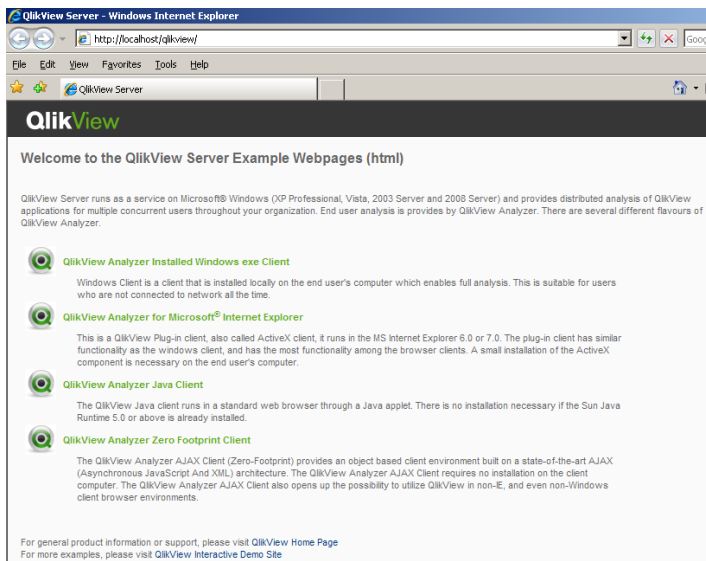
This location has a URL of **http://<server>/QlikView/**

If you cannot display the sample web pages, verify that there exists a virtual directory named **QlikView** defined in your web server, and pointing to the default folder location for the QlikView Server sample pages noted above.

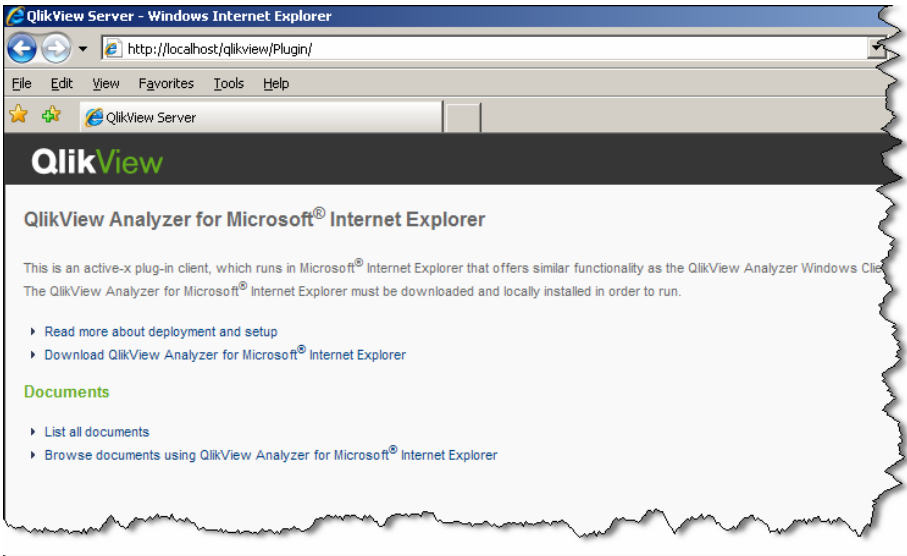
If you installed to a different location or have chosen a different virtual directory name for your installation, then substitute your directory as described below.

The following steps should be completed on the server where QlikView Server was installed:

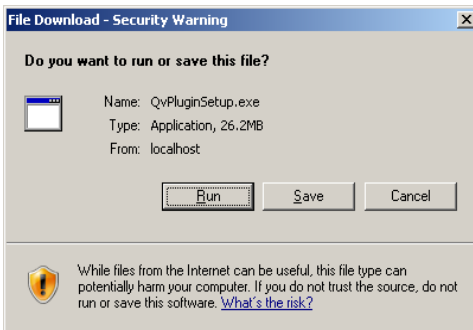
- 1 Open your web browser.
- 2 Connect to the sample QlikView Server pages by entering the following URL: **http://localhost/QlikView.**



3 Select QlikView Analyzer for Microsoft® Internet Explorer.



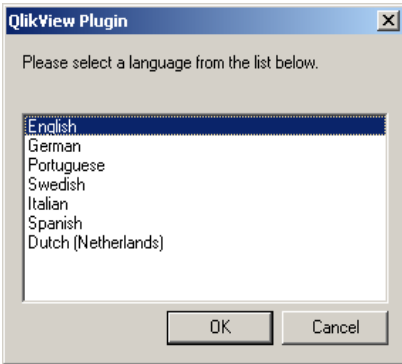
4 Select Download QlikView Analyzer for Microsoft® Internet Explorer.



5 Click **Run** to execute the **QvPuginSetup.exe** file.



6 Click **Run** again.

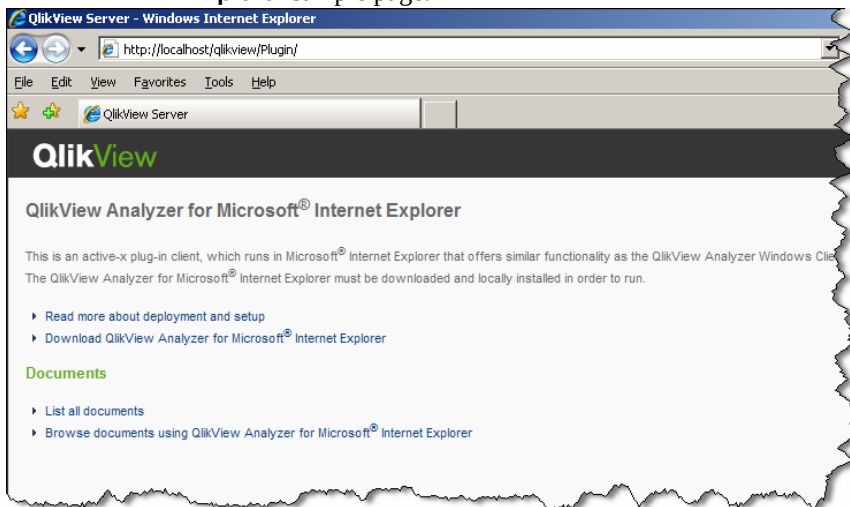


7 Select the appropriate language

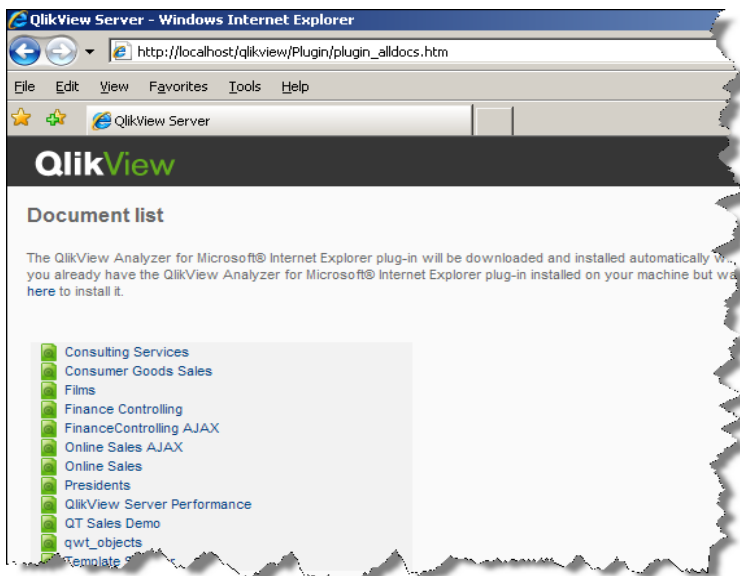


8 Install the QlikView Plugin. You will be asked to specify your location, accept the license agreement, and accept the default location.

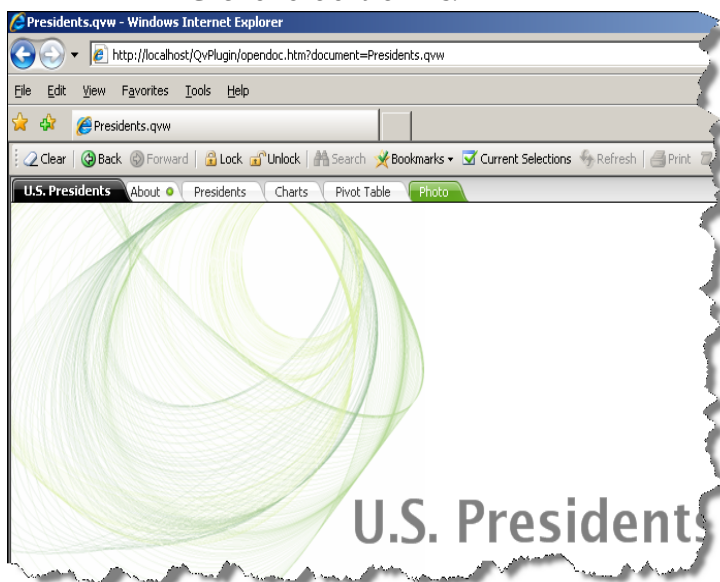
- 9 Once the **QlikView Analyzer for Microsoft® Internet Explorer** is installed, return to the **QlikView Analyzer for Microsoft® Internet Explorer** sample page.



- 10 Select **List all documents** from the sample page.



- 11 You will now see a list of links to sample QlikView documents. Click on one of the links.



- 12 The **QlikView Analyzer for Microsoft® Internet Explorer** client should open in your Internet Explorer web browser. Within a few seconds the toolbar, the tabbed sheets and other sheet objects should appear in the browser window.

TIP: If the QlikView document does not open after clicking on the link, the user may have to add the server/Fully Qualified Domain Name to the Trusted sites security zone in the browser settings.

- 13 If the application displays in your web browser, then you have successfully installed QlikView Server.
- 14 If you would like to test another client, simply select a different client choice on the default sample page. Note that the QlikView Analyzer Windows Client will also require a download and install. To use the Java or AJAX clients, you may have to modify some settings in your web server (e.g. IIS or the built-in QVS http server) to disallow anonymous access. See the section titled “QlikView Server 8 security setup” in this manual for more information.

4 QVS LICENSING

4.1 Client Access Licenses (CALs)

In order to connect to a QlikView Server each client needs a Client Access License (CAL). The CALs are purchased with QlikView Server and tied to the QlikView Server serial number. A CAL is never transferred to a client, but a client uses a CAL when connected to a specific QlikView Server. CALs are thus not transferable between different instances of QlikView Server. If a user is required to work with documents residing on several instances of QlikView Server, a separate CAL is needed at each of the QlikView Servers.

4.2 Types of CALS

There are three different types of CALs available:

- **User CAL** – Access is based on the user identity or machine identity. Any number of concurrent sessions from one user on one machine at a time is allowed. In order to utilize a user CAL, the client user must be identifiable either via user name (Windows or Novell) or with a unique machine ID. IP Address is not a valid form of identification for a User CAL. User name identification and machine name identification cannot be mixed on the same instance of QlikView Server. Note that user name identification requires Windows authentication on Java and Zero-Footprint clients, since machine name identification is not possible from these clients.
- **Session CAL** – Each Session CAL allows one user to access one QlikView document, i.e. to have one session at a time. Anonymous users are allowed, no identification of the client user is necessary.
- **Usage CAL** – Each Usage CAL gives the right to initiate one session (single document) per running 28-day period. The session may last a maximum of one hour. Any activity after the first hour has expired will count as a new session (albeit without visible interruption). No identification of the client user is necessary.

TIP: CALs are used for purposes of licensing only and they have nothing to do with user authentication for data access purposes.

4.3 Combining different types of CALs

A given instance of QlikView Server can carry any combination of the CAL types listed above. When different CAL types are combined, the CAL options will be used as follows:

- 1 If there is a dedicated user CAL for the connecting client, that will be used.
- 2 If there is an available session CAL, that will be used.
- 3 If there is an available usage CAL, that will be used.
- 4 If none of the above, access will be denied.

4.4 Cluster Licensing

A special type of license is available to allow multiple QlikView Server installations to share the same license serial number, and support shared CALs. These servers are automatically considered as clustered. Note that this configuration will affect Networks where unauthorized license sharing between test and production environments has been configured.

4.5 QlikView Server Small Business (SB) Edition

QlikView Server Small Business (SB) Edition is available for customers looking for a QlikView Server specifically designed to support a smaller organization. While QlikView Server Enterprise Edition will contain all features and functionality of the full QlikView Server product, the SB Edition will be more focused in the features that will be offered. Upgrade from SB to Enterprise will be controlled through licensing, so the migration process can be easily accomplished.

QlikView Server Enterprise Edition	QlikView Server Small Business Edition
Designed to support large numbers of users and integrate into enterprise environments	Designed for small numbers of users
End user QlikView UI development is allowed on the server	QlikView UI is read-only on the server
Integrates to 3rd party authentication for security and access control	Windows Active Directory only for security and access control
Users can be shared across servers for load-balancing, test environments, and fail-over	Single server deployments only

QlikView Server Enterprise Edition	QlikView Server Small Business Edition
Named, session and usage CALs. Allows anonymous (typically external) users	Only named CALs. No anonymous users
AJAX client which makes it much easier to maintain large user communities	No AJAX client (Java, plug-in, stand-alone only)
No user limits	Limited to 25 users

5 SERVER MANAGEMENT CONSOLE PARAMETERS

5.1 The QlikView Management Console

The QlikView Server Management Console can be utilized to set and change all parameters controlling the QlikView Server operation. In addition, the QVS Management Console provides the ability to monitor key performance indicators, document usage and user connection status of the QlikView Server. You can open the QVS Management Console from the **Start menu**, under **Programs ... QlikView**. The default location for the program is:

C:\Program Files\QlikView\MgmtConsole\QvsMC.hta

If you are managing multiple servers in a clustered environment, you may connect to individual servers by specifying a parameter on the shortcut command line as follows:

C:\Program Files\QlikView\MgmtConsole\QvsMC.hta <server>

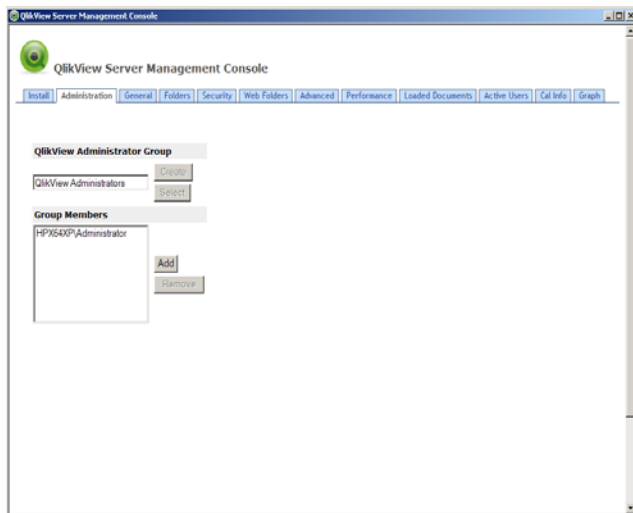
This will allow you to monitor activity in a specific server. Note that any settings changed through the Management Console to a server in a clustered environment will change all other servers in the cluster as well. Any requirement to restart a server based on applying changes will restart all servers in the cluster.

You can also open the QVS Management Console in either a standard web browser or in a local QlikView. If you open the QVS Management Console through the web or QlikView, you will not have access to the **Install** and **Graph** pages.

Prior to utilizing the functions in the QlikView Server Management Console, you must register as an authorized user in the QlikView Administrators group. You can do this either through the program shortcut or through the web dialog at the default address:

http://<server>/qvsmanagement/Default.htm

Note that prior to adding an authorized user, only the Install page in the QVS Management Console will be available..



The Administration page of the QVS Management Console.

Add one or more authorized users that you wish to grant access rights to the QlikView Server management console. Click on the **Add** button.



You will be prompted for a user name. Enter the user name in the following format:

<Domain>\<user>

You may add as many authorized users as you wish.

TIP: The QlikView Administrators group and/or users in that group can also be added through standard Windows user management dialogs.

The QVS management virtual directory in IIS (**qvsmangement**) should be set to disallow anonymous access, with integrated authentication set.

Once the users have been added, you may set the default location address for the QVS Management Console in QlikView for each of those users.

To open in QlikView, set the URL for the QVS Management Console in **Settings ... User Preferences ... Locations**. Modify the location for Server Control Panel (URL).

The default URL of the QVS Management Console is:

http://<server>/qvsmanagement/Default.html

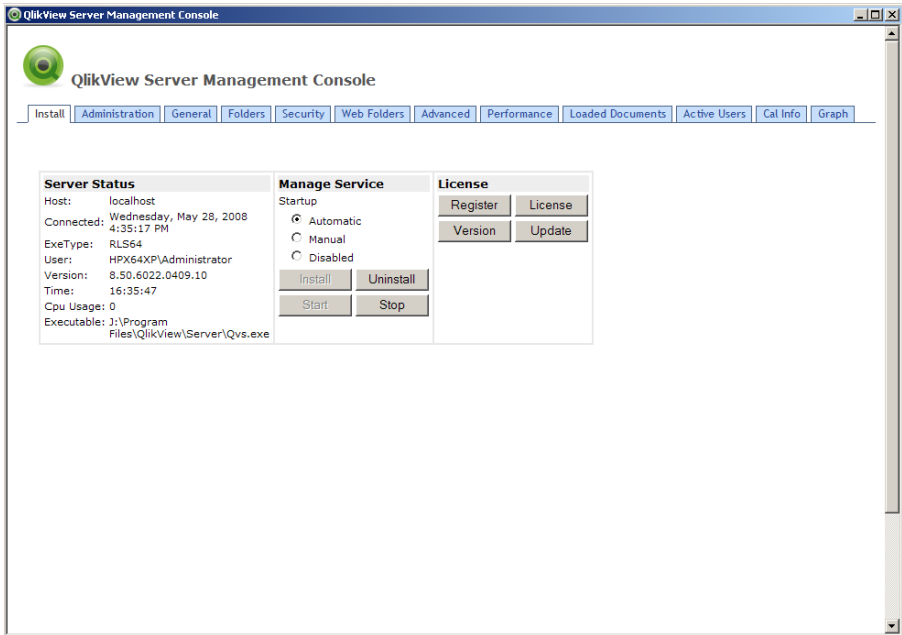
Once set, this location will be referenced when selecting **Open QlikView Server Control Panel** off the **File** menu in QlikView.

TIP: QlikView Publisher Enterprise will require communications with QlikView Server, e.g., for DMS data, so be sure to add the Publisher Administrator user to the QlikView Administrators Group.

Since the QVS Management Console communicates directly with QlikView Server, it is not possible to access the settings available on the Install tab of the Control Panel dialog, since they could affect the operation of the QlikView Server you are communicating with. For example, you must use the Server Control Panel to Register the QlikView Server, and Install and Uninstall the QlikView Server as a Windows service.

For explanations of the common individual settings available through the Server Control Panel and the QVS Management Console, please refer to the previous chapter.

Install page



The Install page allows direct control of the QlikView Server service. This page is only available through the Program shortcut (HTA).

Server Status

This area provides information about the current status of the QlikView Server as seen from the HTA

- **Host** is the QVS-machine that the HTA is connected to.
- **Connected** is the time when the HTA established the connection
- **ExeType** is the type of server (Release, Debug 32 or 64 bit)
- **User** is the user as seen by the server
- **Version** is the installed version of QlikView Server
- **Time** is current server time (also acts as a heartbeat)
- **Cpu Usage** is current server cpu usage (also acts as a heartbeat)
- **Executable** the actual path to the server (as seen from the HTA)

TIP: The Executable path must be valid for the operation of the Manage Service buttons.

Manage Service

Startup

Set the QlikView Server startup mode: **Automatic**, **Manual**, or **Disabled**.

Automatic startup mode instructs the QlikView Server to start each time the computer is restarted. Manual allows the administrator to manually start the QlikView Server when needed; it will not automatically start upon reboot. Disabled prevents the QlikView Server from being started even manually.

Install

The QlikView Server runs as a service on Windows Operating Systems. The default installation procedure will automatically install the QlikView Server service. If for any reason you need to install the service manually, you can use this button.

Uninstall

If at any time you wish to remove the QlikView Server service from Windows, click the **Stop** button, then click on the **Uninstall** button. This button does not remove any files from your system. Instead, it merely removes it from Windows's list of available services.

Start

Once the QlikView Server has been registered and installed as a Windows service, it can be started. If the QlikView Server is set for Automatic startup, rebooting the computer will start the QlikView Server service. To manually start the service, either the Windows control panel or this button in the QVS Management Console can be used.

In the QVS Management Console, click on the **Start** button to activate the QlikView Server service. The service will start and the **Stop** button will become enabled. No dialog will appear indicating successful startup of the service.

Stop

The Stop button can be used to manually stop the QlikView Server service if necessary.

License

Register

Registration authenticates your copy of QlikView Server and allows it to run on your computer. In order to register,

you must have a valid Serial Number and Control Number issued by your vendor. If you do not have both a Serial Number and a Control Number for QlikView server, contact your vendor.

In the QVS Management Console , click on the **Register** button. Enter the Serial Number and Control Number assigned to your copy of QlikView Server. You should also enter your name and company name in the fields provided. Once you accept this dialog, a License Enabler File (LEF) will be obtained through the Internet. If for some reason this information cannot be accessed, registration will not be completed successfully. For manual validation procedures, contact your vendor.

License

Opens the License Enabler File Editor. License enabling information is normally downloaded automatically from QlikTech. Should this not work, e.g. because of lack of Internet access, the license enabling information may be entered here manually. For manual validation procedures, contact your vendor.

Version

Shows the version and build number of the QlikView Server software.

Update

Launches QlikView Update to check the QlikTech web site for available patches to QlikView Server.

General page

QlikView Server Management Console

Install Administration General Folders Security Web Folders Advanced Performance Loaded Documents Active Users Cal Info Graph

Sessions

Maximum Number of Concurrent Sessions

Possible Session Timeout (seconds)
(Server may terminate inactive session and re-use slot if needed for other user)

Maximum Inactive Session Time (seconds)
(0 means no limit)

Maximum Total Session Time (seconds)
(0 means no limit)

Documents

Document Timeout (minutes)

☒ Allow only one copy of document in memory

☐ Disallow moving and sizing objects

☒ Use Regional Settings for Send to Excel (for Java Clients)

☒ Allow Server Collaboration

☐ Allow Anonymous Server Bookmark

☐ Allow Document Upload

☐ Allow Document Download

Apply

Logging

☒ Enable Session Logging

☒ Enable Performance Logging every minutes

☒ Enable Event Logging

Log Folder

Split Files
☒ Never
☐ Daily
☐ Weekly
☐ Monthly
☐ Yearly

Verbosity
☒ Low
☐ Medium
☐ High

The General page of the QVS Management Console.

Maximum Number of Concurrent Sessions

The **Maximum Number of Concurrent Sessions** parameter specifies the maximum number of user sessions allowed to be connected to the QlikView Server at one time. A new user session is generated for each document that a user opens on the QlikView Server. This setting is unrelated to CAL specifications.

The value under **Maximum Number of Concurrent Sessions** works in connection with the **Possible Session Timeout** parameter discussed next.

When a user attempts to open any QlikView document, the QlikView Server will compare the number of active sessions against the **Maximum Number of Concurrent Sessions** limit. If the maximum number of sessions is already reached, the active sessions are checked to see if any of them have timed out. If a session has been inactive for the specified timeout period, it will close down and the new user session request will be granted. If the maximum number of

sessions is reached and none have timed out, the new user session is refused.

This setting is provided as a way for the system administrator to prevent an excessive number of users on one server to cause general congestion and unacceptable response times. It has nothing to do with licensing restrictions.

Possible Session Timeout (seconds)

A session begins when a user opens a QlikView document with the server. When the session has had no activity for the specified number of seconds, it is eligible to be closed if a new user requests to start a session and the **Maximum Number of Concurrent Sessions** has been reached. Refer to the **Maximum Number of Concurrent Users** parameter described above for more information.

Maximum Inactive Session Time (seconds)

A session begins when a user opens a QlikView document with the server. If this setting is non-zero, and the session has had no activity for the specified number of seconds, it will automatically be terminated by QlikView Server. This setting is also available on a Document level. You can find it on the Server tab of the Document Properties dialog. If specified in both locations, the more restrictive setting will be used.

Maximum Total Session Time (seconds)

A session begins when a user opens a QlikView document with the server. If this setting is non-zero, all sessions will be limited to the maximum number of seconds as specified. Once the time limit is reached, the session will automatically be terminated by QlikView Server. This setting is also available on a Document level. You can find it on the Server tab of the Document Properties dialog. If specified in both locations, the more restrictive setting will be used.

Document Timeout (minutes)

A document is a QVW file opened by the QlikView Server. Open documents take up valuable system resources (i.e. RAM) and should not be allowed to remain open when not in use. However, if documents are closed too quickly, the user may see longer delay times when accessing the document while the server reopens it.

The **Document Timeout** value allows you to control for how long a document will be allowed to be unused before the QlikView Server closes the document and reclaims the resources.

TIP: As of release 8.20, document loading and unloading can be controlled directly through settings in the Document Metadata Services (DMS) dialogs available on the Security tab of the QlikView Server Management Console.

Allow Only One Copy of a Document in Memory

If the status of a currently open document is changed due to a reload or changes in the layout, etc., QlikView Server will either open a new instance of the document or replace the currently opened instance. If the checkbox **Allow Only One Copy of a Document in Memory** is selected, the client session document will be forced to update whenever changes have been made in the original document on the QlikView Server. This setting will conserve memory resources on the server, but could cause forced session updates or disconnect sessions for connected users. This setting works in collaboration with the graceful document refresh settings **Refresh Mode when Document is Updated on Server** and **Client Refresh Initiation Mode**, both located on the Server tab in the QlikView Document Properties dialog. If this setting is checked, old data will not be kept on the server.

This setting replaces the **Forced Document Update** setting found in previous versions.

Disallow moving and sizing of objects (for Java clients)

This setting applies only to Java clients, and may be used to override object security settings in the QlikView document. Note that this setting will not affect the ability of Java client users to change certain column sizes in Multi Box and table/chart objects.

Use Regional Settings for Send to Excel (for Java clients)

If this setting is checked, the Regional Number Settings of the operating system will be used when exporting data from a QlikView object to an Excel file from the Java clients. If not checked the number formatting of the exported QlikView object will be used in the export.

Allow Server Collaboration

QlikView Server Collaboration for Bookmarks, Objects and Reports allows sharing of objects between users. Make sure this setting is checked if you want to allow sharing of objects. This setting works in association with the QlikView Server Collaboration settings **Allow Server Bookmarks**, **Allow Server Objects** and **Allow Server Reports** located on the **Server** tab in the **QlikView Document Properties** dialog.

Allow Anonymous Server Bookmark

If this setting is checked, anonymous users will be allowed to create bookmarks. The machine ID of the client will be used for ownership. The client must allow persistent cookies to be created.

Allow Document Upload

If checked, the **Allow Document Upload** setting will allow new or updated documents to be uploaded to QlikView Server through the Publisher Access Point. QlikView Server must be defined as a resource in Publisher.

Allow Document Download

If checked, the **Allow Document Download** setting will allow documents to be downloaded through the Publisher Access Point.

Logging

This group contains settings for session, performance, and event logging.

Enable Session Logging

Mark this check box to enable detailed session logging from QlikView Server. The file will be called Session-Stats.log.

Enable Performance Logging every n minutes

Mark this check box to enable performance logging from QlikView Server. The file will be called Performance.log. The logging interval can be set between one minute and 24 hours (1440 minutes).

Enable Event Logging

Mark this check box to enable mirroring to a log file of entries from QlikView Server to the Windows event log. The file will be called Events.log.

Log Folder

Here you may specify the folder in which QlikView server will create log files. The default is C:\Documents and Settings\All Users\Application Data\QlikTech\QVS.

Split Files

Here you can specify a time interval after which a new log file will be generated. Useful in order to avoid very large log files. Old files are not deleted. Log file names will have date information appended to them to identify intervals.

Verbosity

Use this setting to control how much information will be written to the log files.

Folders page

QlikView Server Management Console

Install Administration General **Folders** Security Web Folders Advanced Performance Loaded Documents Active Users Cal Info Graph

Document Folder

Path
C:\Documents and Settings\All Users\Application Data\QlikTech\Qvs\

Mounted Folders

Name	Path	Browsable

Add Remove Edit

☐ Use Separate Document Folder for Java

Path

Mounted Folders (Java)

Name	Path	Browsable

Add Remove Edit

Apply Applying Changes will Restart QlikView Server

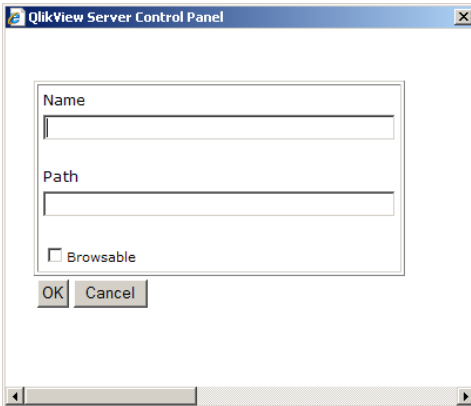
Document Folder

Specify where documents to be accessed from any QlikView client are located. This will typically reflect the default document location. Documents may also reside in subfolders to this folder. Windows file security applies for all access by a client to document folders and files, unless DMS Authorization mode is used. The default location of the Document folder will be different for Windows Server 2008 and Windows Vista operating systems. It will be located under the **C:\ProgramData\QlikTech\QVS\Documents** folder

Mounted Folders

Here you specify folders, physically located outside the QlikView Server document folder, which should functionally appear to QlikView Server as if they were located inside. A mounted folder may contain subfolders to any level.

Click Add to add a mounted folder.



Name Logical name of mounted folder as seen from QlikView Server.

Path Path to mounted folder.

Browsable

Mark this check box if the mounted folder and its contents should be browsable from the **Open in Server** dialog of QlikView.

TIP: To specify a document in a mounted folder location in web pages, make sure to use the folder name followed by a forward slash, then the document name.

(e.g. **Pubdocs/Sales.qvw**)

Select a mounted folder in the list and click on **Remove** to delete the mount (the physical folder and its contents are not affected). Select a mounted folder in the list and click on **Edit** to reopen the **Edit Mount** dialog and change properties of a mount.

Windows file security applies for all access by a client to the contents of a mounted folder.

Use Separate Document Folder for Java

Mark this checkbox should you prefer to have a separate document folder for the Java clients. Specify the folder location in the edit box below. Mounted folders for Java client access can be specified under **Mounted Folders (Java)**.

Security(Communication) page

QlikView Server Management Console

Install Administration General **Folders** Security Web Folders Advanced Performance Loaded Documents Active Users Cal Info Graph

Authentication

Windows Client ☒ **Separate Java Settings**

☐ Always Anonymous ☐ Always Anonymous
☒ Allow Anonymous ☒ Allow Anonymous
☐ Prohibit Anonymous ☐ Prohibit Anonymous

Anonymous Account

☐ On Domain
☒ On Local Computer

Document Metadata Service (DMS)

Document Management

☐ Document Auto Load
 Polling interval: (in minutes)

Authorization

☒ NTFS Authorization (Windows controls file access)
☐ DMS Authorization (QlikView controls file access)
 Directory Services Connector URL:

Communication Ports

Standard Ports **Tunneling Port**

Misc

☒ Compress Network Traffic
☐ Allow admin using name and password

Applying Changes will Restart QlikView Server

Authentication

In this group you select whether the QlikView server should use Windows authentication when possible. It is possible to force anonymous communication (**Always Anonymous**), force authentication (**Prohibit Anonymous**) or to use authentication whenever possible (**Allow Anonymous**). **Allow Anonymous** is the default.

Make sure that this setting is consistent with any security settings that may be specified in the web server virtual directories (e.g. if IIS allows Anonymous, but QlikView Server does not, the client user will get an error message when trying to open the application through the virtual directory).

It is possible to use separate settings for Java clients by marking the **Separate Java Settings** check box to the right.

Communication Ports

The QlikView client communicates with the QlikView Server over one or more TCP ports. The default port is 4747, and this port should work for most environments. If you have a firewall, you can either open Port 4747 for QlikView sessions (e.g. for public Inter-

net use) or tunnel QlikView via http (typically Port 80) by using the QVSTunnel.dll that is installed in the Scripts virtual directory during installation.

Multiple ports can be specified. QlikView Server will try to initiate communication with the client over each port in the order they are listed until a working port is found. This is usually only required for Internet sessions where your users may be behind a firewall outside of your control. If the user's firewall has blocked communication over your default port, QlikView Server will continue with the other listed ports until an unblocked port is found.

For additional information on the QVS Tunnel extension, please refer to Chapter 7.

Tunneler port

Specifies the local server port used for tunneled communication.

Compress

Mark this check box if large packages should be compressed in communication between client and server. It is recommended to uncheck this setting in high bandwidth environments, since the compression routines could require more resource than sending large packages of data over a high bandwidth network.

Allow admin using name and password

This setting is used by Publisher if it is running in a separate Active Directory so that name and password can be used to connect to the QVS service.

Document Metadata Service (DMS)

Document Metadata Service (DMS) is a thread running in the QlikView Server service. There are two distinct and separate functions provided by DMS. The first function (**Document Management**) is used to control automated document loading and unloading. The second function (**Authorization**) is used to control the authorization mode that QlikView Server will run in. There are 2 choices for Server authorization mode. NTFS Authorization mode that QlikView Server has traditionally run in is the default selection. Alternatively, DMS Authorization mode can be used to enable third party (non-Windows) authenticated users to access documents through QlikView Server, as well as specification of specific access rights for individual or anonymous users, along with access days and times.

TIP: Syntax has been added for scripted access to QlikView Server DMS. This will allow batch update of DMS and CAL data in QlikView Server. For more information, refer to the QlikView Server Script API in the appendix section of this manual.

Document Management

Document Auto Load

Check this box to enable automated document loading and unloading. Document settings are configured in the QlikView Server Document Metadata – Webpage Dialog. Refer to the description of this dialog below.

Polling interval (in minutes)

Specify the interval (in minutes) that QlikView Server will use to poll the DMS configuration for changes. The default setting is 10 minutes.

Authorization

Choose one of the options in this section to determine the authorization mode that QlikView Server will use when authorizing access to documents. Traditionally, QlikView Server has utilized **NTFS Authorization**, where the Windows Operating System controls access to files for users and groups through NTFS security settings. This is the default authorization mode for QlikView Server.

Alternately, choose **DMS Authorization** to utilize the QlikView Server DMS facility to authorize access to documents for users and groups. The QlikView Publisher Directory Services Connector (DSC) must be accessible in order to resolve Group membership. Document settings are configured in the QlikView Server Document Metadata – Webpage Dialog. Refer to the description of this dialog below.

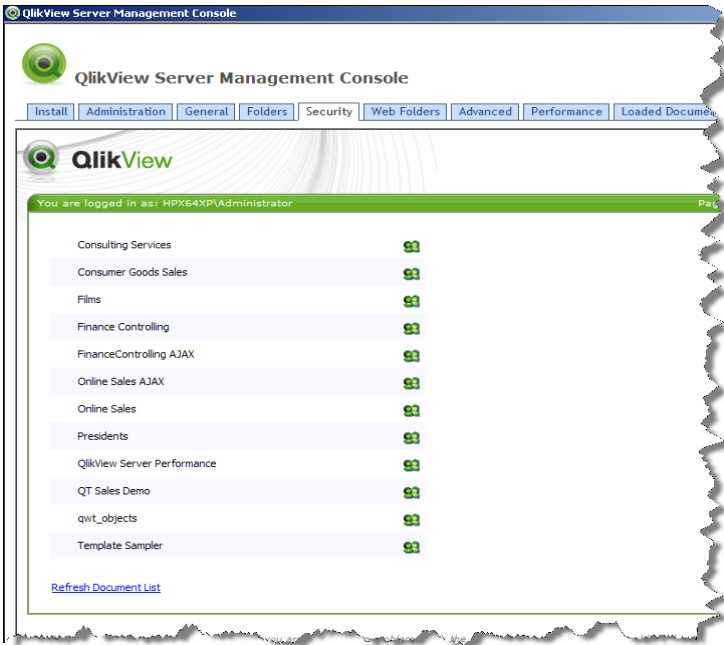
Directory Services Connector URL:

Enter the URL address for the QlikView Publisher Directory Services Connector (DSC). This must contain a valid address for the active DSC in order to resolve Group membership when using DMS Authorization.

Edit Document Metadata

Click the **Edit Document Metadata** button to open the QlikView Server Document Metadata – Webpage Dialog in a new browser window. Here you can specify the docu-

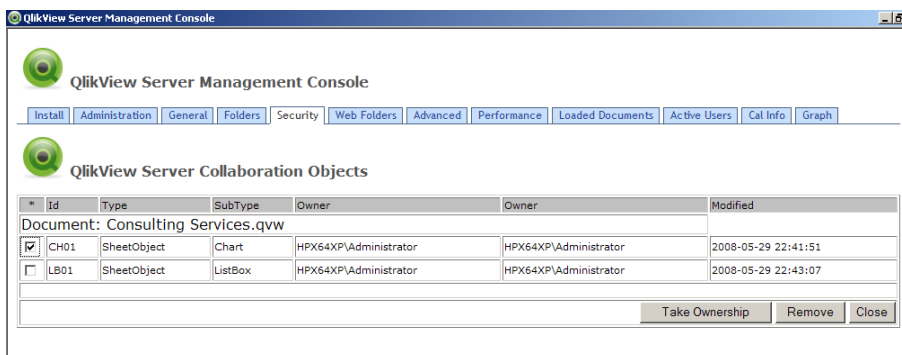
ment level settings for Document Management and DMS Authorization.



DMS Document Selection Page

This page displays the QlikView Server Accesspoint, allowing you to select the document you would like to configure for Document Management, DMS Authorization, and/or Accesspoint settings.

From this page, you can also configure shared collaboration objects by clicking on the icon to the right of the document name.



DMS Server Collaboration Objects Page

This page allows you to manage server collaboration objects by document. Click on the checkbox next to the **object ID** to select that object for management. The **Take Ownership** button may be used to take ownership of the object in order to view/change object properties. Access to object properties is through an appropriate client. Use this option with care, since ownership cannot be given back to the original owner of the object once this is initiated. Use the **Remove** button to delete the selected objects. The **Close** button may be used when the Management Console is opened in a web browser to close

the window. Program users may use the **Back** button, or simply click on another page tab.

The screenshot shows the QlikView Server Management Console window. The title bar reads 'QlikView Server Management Console'. Below the title bar is a navigation bar with tabs: 'Install', 'Administration', 'General', 'Folders', 'Security', 'Web Folders', 'Advanced', 'Performance', 'Loaded Documents', 'Active Users', 'Cal Info', and 'Graph'. The 'Administration' tab is selected. The main content area is titled 'QlikView Server Document Metadata'. Below this, it says 'Document: Consulting Services.qvw'. There are three main sections: 'Document Management', 'Document Authorization', and 'Accesspoint settings'. The 'Document Management' section has 'PreLoaded' options: 'Never' (selected), 'Always', and 'Restricted'. It also has 'Max Open Sessions' and 'Sessions' fields. The 'Document Authorization' section has a 'User/Group' field with a dropdown arrow, and 'Add' and 'Remove' buttons. The 'Accesspoint settings' section has 'Category', 'Source', 'Last Update', and 'Next Update' fields, an 'EventDriven' checkbox, and a 'Clients' section with checkboxes for 'Plugin', 'Java', 'Ajax', and 'Download'. There is also an 'Ajax Url' field and a 'Generate Pages' button. At the bottom, there is a 'Document Attributes' section with 'Add' and 'Delete' buttons, and a table with 'Name' and 'Value' columns.

Document Metadata – Webpage Dialog

This dialog provides the ability to configure Document Management, DMS Authorization, and Accesspoint settings for the selected document.

Document Management

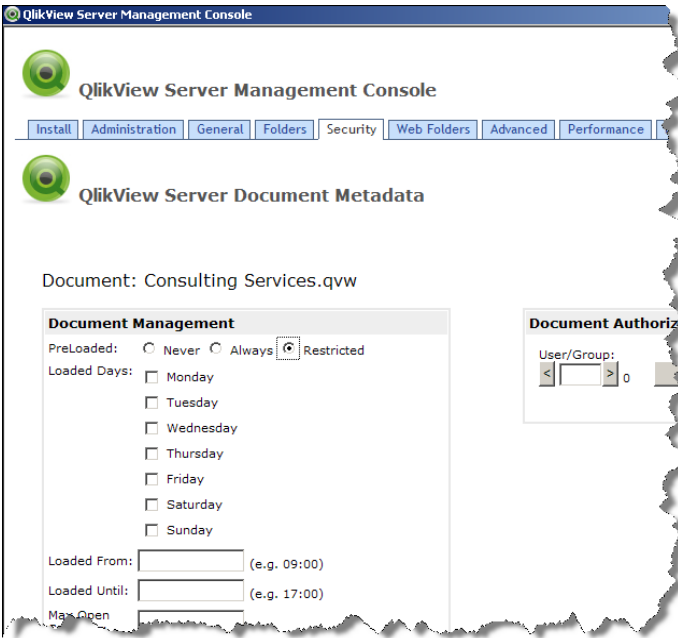
This section is used to configure document management settings for the selected QlikView document. Choose one of the options for **PreLoaded** as follows:

Never: The document will never be loaded automatically through DMS. Standard loading techniques, based on user requests and Document Timeout settings will apply.

Always: The document will always be loaded into Server memory.

Restricted: The document will be loaded automatically through DMS, based on specific day of the week and time restric-

tions. If this option is selected, additional settings will become visible.



If **PreLoaded: Restricted** is selected, you may choose specific days of the week to automatically load the document and the times of the day to load and unload. All times are Server local times (in 24 hour format).

Use the **Max Open Sessions** parameter to limit the number of open sessions allowed at a time for this document.

Accesspoint settings

This section is used to configure the Accesspoint settings for the document.

Category is the Category grouping for this document.

Source is used to cluster documents with the same source document.

Last Update is the Last Update timestamp for this document.

Next Update is the Next Update timestamp for this document.

Clients is the list of client types for analysis available for this document. Use Download for offline analysis.

Ajax Url is the URL address of the default page for an existing AJAX ZFC client page for this document. The address is taken from the **Web Folders** page of the Management console. Use the **Generate Pages** button to generate the AJAX html pages to the specified location.

Document Attributes

Document Attributes are name value pairs that can be used for categorizing the document.

Document Authorization

This section is used to configure document authorization settings for the selected QlikView document. These settings will take effect once **DMS Authorization** is selected as the Authorization Method for this server. Only users specified in this configuration will be allowed access to the document once **DMS Authorization** is selected.

The screenshot displays the QlikView Server Management Console interface. At the top, a navigation bar includes tabs for 'Install', 'Administration', 'General', 'Folders', 'Security', 'Web Folders', 'Advanced', 'Performance', 'Loaded Documents', 'Active Users', and 'Cal Info'. The 'Web Folders' tab is currently selected. Below the navigation bar, the 'QlikView Server Document Metadata' section is visible. The document being managed is 'Consulting Services.qvw'. The interface is divided into two main panels: 'Document Management' and 'Document Authorization'. The 'Document Management' panel includes settings for 'PreLoaded' (set to 'Never'), 'Max Open Sessions' (set to 1), 'Accesspoint settings' (with fields for Category, Source, Last Update, Next Update, and a checkbox for 'EventDriven'), and 'Clients' (with checkboxes for Plugin, Java, Ajax, and Download). The 'Ajax Url' field is empty, and the 'Generate Pages' button is visible. The 'Document Authorization' panel shows 'User/Group' settings (with a list containing '0' and '1', and 'Add'/'Remove' buttons), 'User' settings (with 'Anonymous' checked), 'Access' settings (with 'Always' and 'Restricted' radio buttons), 'WeekDays' (with checkboxes for Monday through Sunday), and 'From'/'Until' time ranges (set to '(e.g. 09:00)' and '(e.g. 17:00)'). At the bottom, there is a 'Document Attributes' table with columns 'Name' and 'Value', and 'Add'/'Delete' buttons. The 'OK' and 'Cancel' buttons are located at the bottom right of the 'Document Authorization' panel.

QlikView Server Management Console

QlikView Server Management Console

Install Administration General Folders Security Web Folders Advanced Performance Loaded Documents Active Users Cal Info

QlikView Server Document Metadata

Document: Consulting Services.qvw

Document Management

PreLoaded: ☒ Never ☐ Always ☐ Restricted

Max Open Sessions:

Accesspoint settings

Category:

Source:

Last Update:

Next Update: ☐ EventDriven

Clients:

☐ Plugin

☐ Java

☐ Ajax

☐ Download

Ajax Url: **Generate Pages**

Document Authorization

User/Group: **Add** **Remove**

User: ☒ Anonymous

Access: ☐ Always ☒ Restricted

WeekDays:

☐ Monday

☐ Tuesday

☐ Wednesday

☐ Thursday

☐ Friday

☐ Saturday

☐ Sunday

From: (e.g. 09:00)

Until: (e.g. 17:00)

Document Attributes **Add** **Delete**

Name	Value
------	-------

OK **Cancel**

To add an authorized user/group, click on the **Add** button. To remove an existing authorized user/group, click on the **Remove** button.

The User/Group can be either Anonymous or named. Group names may be used, but access to the QlikView Publisher Directory Services Connector (DSC) will be required to resolve the Group.

Access can be granted as **Always** for no time restrictions, or **Restricted** to limit access to this document to specific days of the week as well as times. All times are Server local times (in 24 hour format).

QlikView Publisher Developer is the recommended method for maintaining the DMS Authorization Repository. Publisher will be granted priority for writing this information to DMS, so any entries entered manually through the QlikView Server Management Console dialogs will potentially be over-written if they also exist in the Publisher Repository.

Web Folders page (formerly Printing page)

QlikView Server Management Console

Install Administration General Folders Security Web Folders Advanced Performance Loaded Documents Active Users Cal Info Graph

Printing

URL
http://HPX64XP/QvPrint

Folder
J:\Documents and Settings\All Users\Application Data\QlikTech\Qvs\

Timeout (seconds)
300

Ajax Pages

URL
J/QvsDocuments

Folder
C:\Program Files\QlikView\Examples\QvsDocuments

Apply

The Web Folders page, formerly designated the Printing page, allows configuration to support printing and export of Java and AJAX Client chart objects. The locations on this page will direct chart and table object print and exports to a temporary HTML page, stored on the server that can be viewed and printed through the client's web browser. This page also can be used to specify the default location for generated AJAX pages.

In order to create an HTML version of an object for printing in the Java client, simply right-click the object displayed in the client and choose **Export HTML for print** from the popup menu or click on the toolbar icons for Print or Export to Excel once the object has focus. QlikView Server will create an HTML version of the selected object and store the page under a unique name in a directory. QlikView Server will then open a new browser window and open the URL containing the newly created page.

In order for the **Export HTML for print** feature to be enabled in the Java client, the URL and Folder locations for the temporary print files must be specified in the Management Console.

The **Folder** and **URL** addresses should resolve to the same location.

URL

Use the **URL** field to specify the complete URL where exported HTML pages will be accessible to your users. QlikView server will create a temporary HTML page for the object exported, and automatically open a new browser window at the appropriate URL

For example, during installation, a virtual directory named **QvPrint** was created under the Default Web Site directory. The default Print folder would then be accessible as:

http://Myserver/QvPrint

Enter the complete URL where the folder specified in the Folder field can be accessed. If you specified a different folder in the Folder path, your URL may be different from that shown above.

TIP: The server address portion of the Printing URL may have to be changed for printing to function properly in certain configurations. For example, using an IP address or www.Myserver.com may be required for correct name resolution.

Folder

Use the **Folder** field to specify the complete path to a folder on the server where the temporary export HTML pages will be created. When a user requests the **Export HTML for print** feature, a uniquely named HTML page will be created in this folder. QlikView Server will automatically launch a new browser window and open the HTML page for the user. The path must be readable by all users.

For example, the default installation location for the Print folder will be the following directory on your system:

C:\Documents and Settings\All Users\Application Data\QlikTech\Qvs\QvPrint

If you installed to a different location, you may need to substitute a different path to the Print folder. You may also choose a different folder if desired. Enter the complete path into the Folder field.

Timeout (seconds)

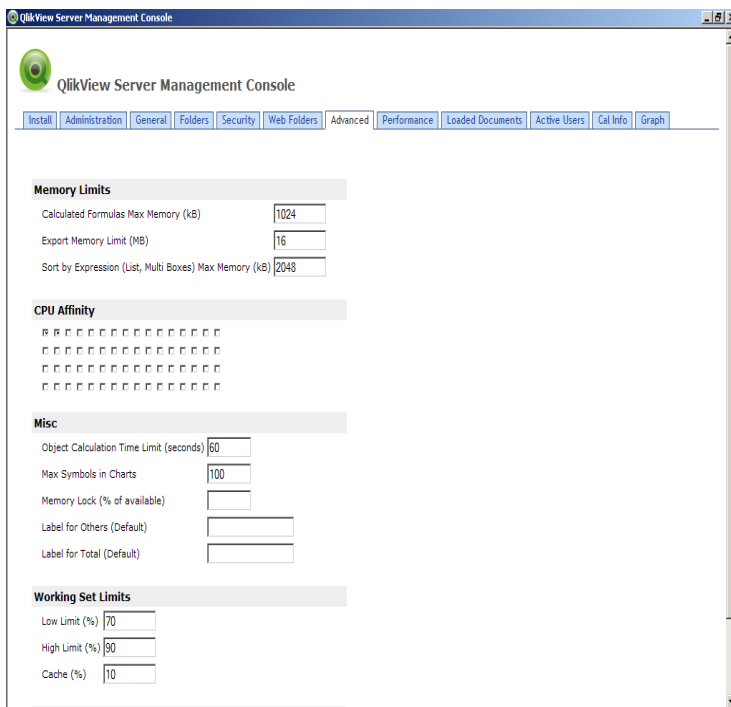
Use the **Timeout** field to specify a timeout value in seconds after which temporary HTML export files will be deleted from the server. Be sure to specify the timeout value high enough to enable temporary pages to transfer completely to the client's browser before the page is deleted.

Ajax Pages

Here you may specify the URL and folder addresses to define the default location of generated AJAX pages, if using the **Generate Pages** button in the **DMS** dialog.

The **Apply** button will set any changes, but will not restart the QlikView Server service.

Advanced page



Calculated formulas

Note This setting is over-ridden by the automatic memory limiter settings since version 7.5x

The **Calculated Formulas** setting specifies the memory allocation for the evaluation of calculated formulas on the server.

Export Memory Limit

The **Export Memory Limit** setting specifies the memory allocation that QlikView server may temporarily use when exporting data. Setting this value too high may cause the server to run out of memory.

Sort by Expression (List, Multi Boxes)

Note This setting is over-ridden by the automatic memory limiter settings since version 7.5x

The **Sort by Expression** setting specifies the memory allocation for the calculation of sort expressions on the server.

CPU Affinity

In cases when QlikView Server is run on hardware with more processors than supported by your QVS license, you may deselect the use of specific processors. QlikView Server will normally automatically select the processors to use and this setting needs to be changed only when you wish to override that choice.

Max Symbols in Charts

Here you can specify a maximum number of symbols to plot in one chart.

Object Calculation Time Limit

The **Object Calculation Time Limit** setting specifies the maximum amount of time, in seconds, that QlikView Server will attempt to calculate a chart object.

Memory Lock (% of available)

Note	This setting is only available on 64-bit systems
-------------	--

When working with very large documents (Gigabyte size) on certain 64-bit systems it may be necessary to lock an amount of the available physical RAM for use by QlikView Server. By doing this you may prevent sudden swapping of the document, which will cause the service to appear to freeze for up to several minutes. This setting should be used with care and never exceed 90. Make sure not to run other RAM-intensive applications (e.g. DBMS systems, QlikView Publisher or QlikView Developer) on the same machine when working with large documents on the QlikView Server.

Label for Others

Here you can specify a default label for Others in bar charts and pie charts.

Label for Total

Here you can specify a default label for Totals in bar charts, pivot tables and straight tables.

Working Set Limits

Use these settings to control the QlikView Server task (qvs.exe) memory working set size.

Low Limit (%)

The minimum working set size for the process. The virtual memory manager attempts to keep at least this much memory resident in the process whenever the process is active.

High Limit (%)

The maximum working set size for the process. The virtual memory manager attempts to keep no more than this much memory resident in the process whenever the process is active and available memory is low.

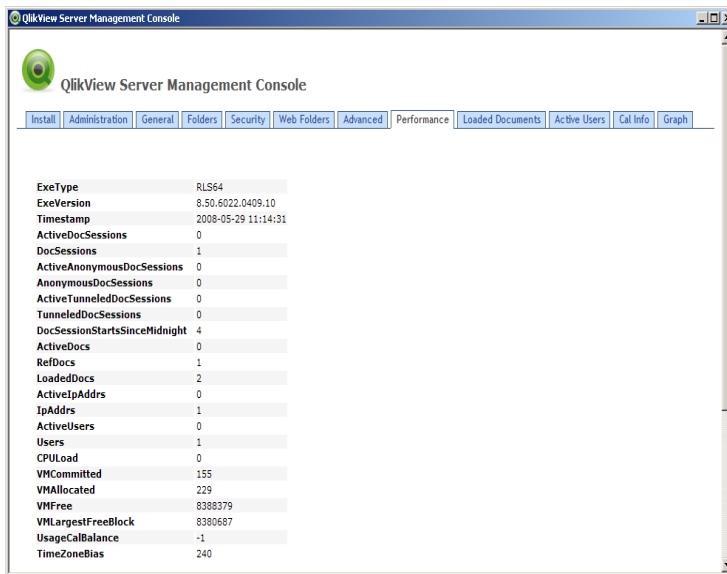
Cache (%)

Controls the MINIMUM allowed amount memory used for caching purposes.

Client licensing Info

Currently not used.

Performance page



ExeType	RLS64
ExeVersion	8.50.6022.0409.10
TimeStamp	2008-05-29 11:14:31
ActiveDocSessions	0
DocSessions	1
ActiveAnonymousDocSessions	0
AnonymousDocSessions	0
ActiveTunneledDocSessions	0
TunneledDocSessions	0
DocSessionStartsSinceMidnight	4
ActiveDocs	0
RefDocs	1
LoadedDocs	2
ActiveIpAddrs	0
IpAddrs	1
ActiveUsers	0
Users	1
CPUload	0
VMCommitted	155
VMAllocated	229
VMFree	8388379
VMLargestFreeBlock	8380687
UsageCalBalance	-1
TimezoneBias	240

All parameter settings on the Performance page of the QVS Management Console reflect current performance measures captured in QlikView Server. The interval for

performance measures is set on the Install tab of the Server Control Panel dialog. The default interval is 5 minutes.

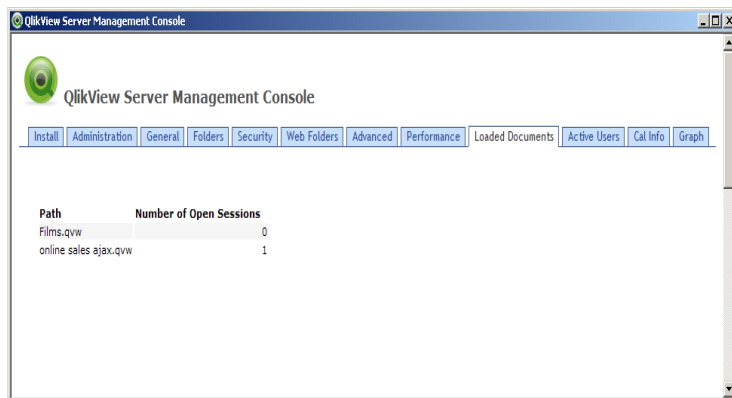
Performance field	Explanation
Exe Type	Type of QlikView Server build. Example: 'RLS32'=32-bit release build.
Exe Version	Full version number of QlikView Server. Example: '8.50.6022.0409.10'
Timestamp	Date and Time of performance measures.
ActiveDocSessions	Number of document sessions* that have shown activity during the interval and still exist at the end of the interval.
ActiveAnonymousDocSessions	Number of document sessions* with anonymous user that have shown activity during the interval and still exist at the end of the interval.
AnonymousDocSessions	Total number of document sessions* with anonymous user that exist at the end of the interval.
ActiveTunneledDocSessions	Number of document sessions* with tunneled connection that have shown activity during the interval and still exist at the end of the interval.
TunneledDocSessions	Total number of document sessions* with tunneled connection that exist at the end of the interval.
DocSessionStartsSinceMidnight	Number of document sessions* that have been initiated during the current day (as defined by the server).
ActiveDocs	Number of documents loaded at the end of the interval in which there has been user activity during the interval.
RefDocs	Number of documents loaded at the end of the interval for which there is a session at the end of the interval.
LoadedDocs	Total number of documents loaded at the end of the interval.

Performance field	Explanation
ActiveIpAddr	Number of distinct IP-addresses that have been active during the interval and still exist at the end of the interval. Note that tunneled sessions and multiple users originating from the same IP cannot be distinguished.
IpAddr	Total number of distinct IP-addresses connected at the end of the interval. Note that tunneled sessions and multiple users originating from the same IP cannot be distinguished.
ActiveUsers	Number of distinct NT users that have been active during the interval and still exist at the end of the interval. Note that anonymous users cannot be distinguished here.
Users	Total number of distinct NT users connected at the end of the interval. Note that anonymous users cannot be distinguished here.
CPUload	Average CPU load from QlikView Server during interval.
VMCommitted	Size in MB of virtual memory actually used by QlikView Server at the end of the interval. This number is part of VMAllocated(MB) and should not exceed the size of the physical memory in order to avoid unacceptable response times.
VMAllocated	Size in MB of virtual memory allocated by QlikView Server at the end of the interval**
VMFree	Size in MB of unallocated virtual memory available to QlikView Server**
VMLargestFreedBlock	Size in MB of the largest contiguous block of unallocated virtual memory available to QlikView Server. This number is part of VMFREE(MB).
UsageCalBalance	Usage CALs remaining. '-1' denotes no Usage CALs exist.
TimeZoneBias	Server time offset from GMT (in minutes)

* one user + one document = one document session.

** VMAllocated(MB)+VMFree(MB)=total maximum virtual memory space available to the QlikView Server process.

Loaded Documents page



The Loaded Documents page of the QVS Management Console.

The Loaded Documents page of the QVS Management Console can be used to monitor current active document session activity in QlikView Server.

Path

The QlikView Server Folder path and document name of the active document(s). Loaded documents with no connected user sessions are not listed here. For example:

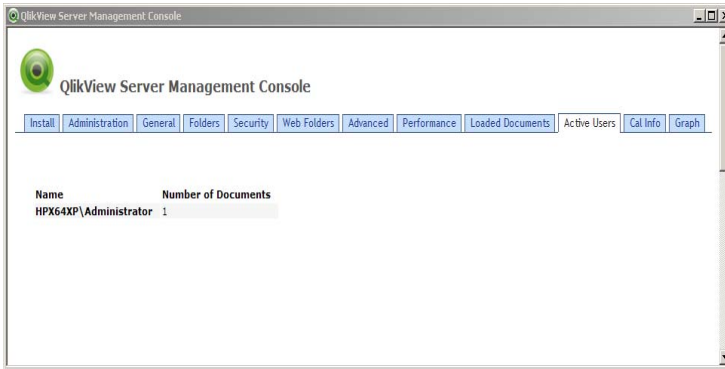
Additional Docs\Presidents.qvw

Number of Users

The number of user sessions connected to the active document.

TIP: You may have to refresh this page manually to view current activity.

Active Users page



The Active Users page of the QVS Management Console can be used to monitor current user session activity in QlikView Server.

Name

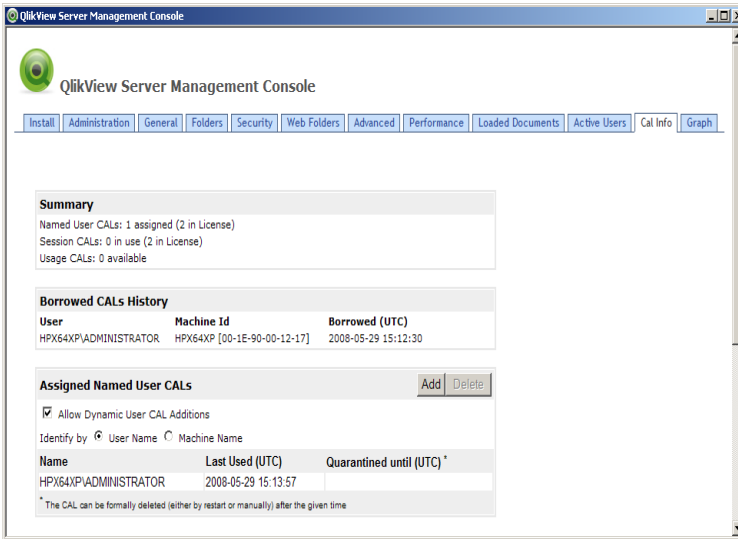
The qualified user name with either machine name or domain and user name of the currently connected user(s). For anonymous users on a session or usage CAL, the anonymous user name for the domain or machine will be listed. For example:

<domain>\QVUser

Number of Documents

The number of documents currently open in a user session for this user.

Cal Info page



The Cal Info page of the QVS Management Console can be used to display information about Client Access License (CAL) usage. This functionality replaces the Manage User CALs button, formerly available on the Install tab of the legacy Control Panel.

Summary

The usage by type of CAL and number of CALs defined in the LEF is displayed.

Borrowed CALs History

This section will list current information about Borrowed CAL activity. A Borrowed CAL is used by clients who connect to QlikView Server and may borrow a license to open the downloaded server document for 14 days.

Assigned Named User CALs

The current allocation of Named User CALs is displayed. Named User CALs can be either automatically allocated or manually added by clicking on the **Add** button, if there is a licensed Named User CAL slot available. Note that the allocation of Named User CALs does not imply security Authentication. Allocation of Named User CALs is for QlikView Server licensing, not operating system authentication.

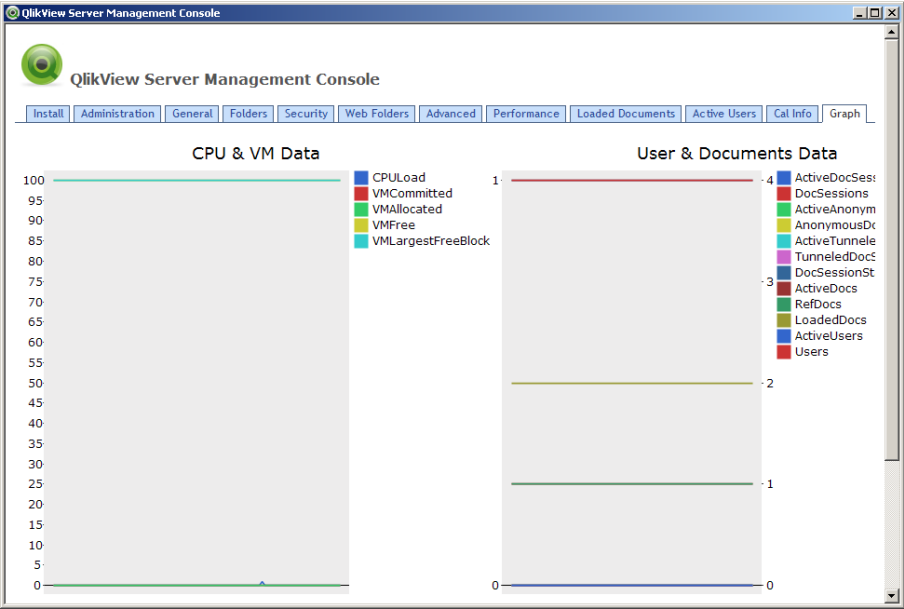
If the **Allow Dynamic User CAL Additions** is checked, new named user CALs will automatically be granted to users connecting to this QlikView Server for the first time, as long as there are available Named User CALs to assign.

In the **Identify** by group you decide whether named users should be identified via identified **User Name** or via **Machine Name** (actually machine name + MAC address). It is possible to change this setting at any time but it is strongly recommended to use one mode consistently with a given QlikView Server. If changed during operation, the same user can take up two CALs, one based on user name and one on machine name.

To delete an allocated Named User CAL, highlight the entry, and click on the **Delete** button. If the CAL is not currently in use, it will be deleted immediately. If the CAL is currently being used, it will be marked for deletion, and not allow new sessions for user access through this CAL, but will still occupy an allocated spot for a 24 hour period. During this period, you may Undelete by highlighting the entry and clicking the **Undelete** button. After the 24 hour period, the entry will be deleted, and no longer available.

TIP: Maintenance of Named User CALs no longer requires a restart of the QlikView Server service.

Graph page



The Graph page displays graphical performance data for the connected QlikView Server. This page is only available through the Program shortcut.

This page will automatically refresh.

6 QLIKVIEW SERVER 8 SECURITY SET-UP

6.1 Communication encryption

All communication between QlikView Server and Windows based clients (QlikView Developer, Professional, Analyzer and plug-in) is encrypted. QlikView Server will attempt to establish 128-bit encryption based on the RSA algorithm when a client connects. The level of encryption may however be lowered if the operating system of the client computer does not support this strength of encryption (notably certain older Windows 98 installations).

Communication between QlikView Server and the Java based clients (QlikView Java client and QlikView Objects Java client) is scrambled but not encrypted. Encrypted communication can however be achieved by using the QlikView Server Tunnel and the https protocol.

Secure communication between QlikView Server and the AJAX client depends on http or https. Between the web browser and QVS, it depends on IIS or the QVS http server. Only the QVS http server can (and will by default) encrypt communication starting with version 8.5.

6.2 File system security on server

If the legacy NTFS Authorization mode is set on the Security page in the QlikView Server Management Console, QlikView Server will only make qvw documents available to a connecting client if the client has an identity with Operating System file access rights to that document. Each client must be given read permissions to both document folders and the documents themselves through a Windows security level. See below for details regarding anonymous clients.

Document and folder permissions are set on the **Security** page of the **Properties** dialog for documents and folders respectively. These settings are made entirely in the operating system and not from QlikView or QlikView Server.

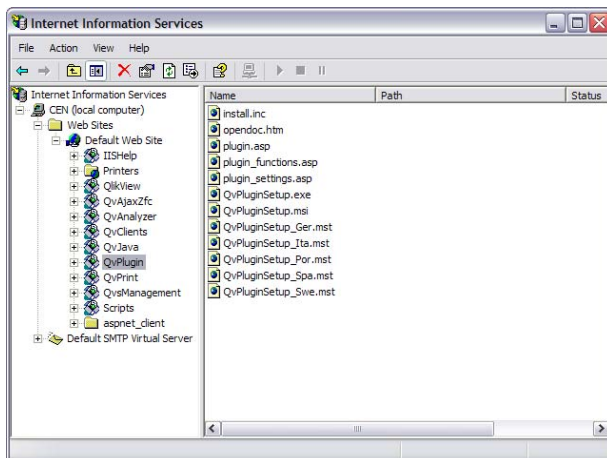
TIP: Make sure to review the effective permissions after changing permission settings on a file or folder level. In complex environments it is not uncommon that conflicting permission settings cause access to be more or less restricted than intended.

For DMS Authorization mode information and settings, refer to the sections titled Document Metadata Service (DMS) and the Security page description for the QlikView Server Management Console.

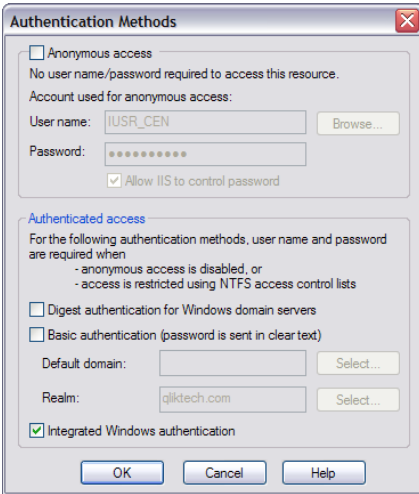
Setting up IIS for Windows authentication with QVA for IE and QV AJAX ZFC

In order for the QVA for IE plug-in to be able to work with Windows authentication when the client computer is not on the same domain as the QVS, the following steps have to be performed:

- 1 Under **Administrative Tools** in the Windows Control Panel open **Internet Information Server**. Then expand your web site in the tree control.



- 2 Right-click on **QVPlugin** and select **Properties**. Navigate to the **Directory Security** tab and click on **Edit**.



- 3 Make sure that **Anonymous Access** is deselected and that at least one of the check boxes under **Authenticated Access** is selected.

Setting up IIS for Windows authentication with QlikView AJAX ZFC

In order for the QlikView AJAX ZFC to be able to work with Windows authentication you must perform the same procedure as the one described for QVA for IE above, but set properties for **QvAjaxZfc** instead. This step is required if only named user CALs are licensed.

While it is not a requirement, you may also want to set the virtual directory subfolder **QlikView\QvAjaxZfc** to Windows authentication as well to set the access rights to the html pages.

Setting up IIS for Windows authentication with QlikView Java client

In order for the QlikView Java client to be able to work with Windows authentication you must perform the same procedure as the one described for QVA for IE above, but set properties for the subfolder QlikView/Java instead. This step is required if only named user CALs are licensed.

The anonymous user account

When the QlikView Server is started for the first time on a machine an account will be created for anonymous users. The account will be named IQVS_name where name is the name of the machine in the local network.

If the machine in question is a domain server, the anonymous account will be created as a domain account or it will be a local machine account.

Each folder and file that should be available from anonymous clients must be given read privileges to the anonymous account.

Note It is important to start QlikView Server and thereby let it create the anonymous account before any attempt is made to grant privileges. You must not try to create the anonymous account yourself!

Connection pseudo-URLs

When connecting to QlikView Server from Windows clients, either via the **Open in Server** dialog or via link files, the identity to be used is specified via the pseudo-URL document address.

The syntax is:

```
qvp://[[username]@]servername [:(port | protocol)] /  
[documentname.qvw][?paramname=paramvalue{&param-  
name=paramvalue}]
```

where

username is a Windows user ID

servername is the name of a server running QlikView Server

documentname is the name of the QlikView document (excluding qvw extension)

port (e.g. 4749) can be used to specify a specific port used by the server

protocol (e.g. http) can be used to specify tunneling protocol

paramname := (USERID | XUSERID | PASSWORD | XPASSWORD |
MACRO |

IIS_AUTHENTICATE)

USERID denotes a section access userID in clear text

XUSERID denotes a scrambled section access userID

PASSWORD denotes a section access password in clear text

XPASSWORD denotes a scrambled section access password

MACRO denotes the name of a macro to be run when the document is opened

(only one macro allowed)

IIS_AUTHENTICATE denotes a single-use key (40 hex characters) for IIS integrated authentication.

paramvalue is a valid value for each parameter.

@ without **username** denotes anonymous identity.

If user identity is omitted altogether, the logged in Windows identity is assumed.

Examples:

qvp://www.qliktech.com/AcmeStores.qvw

qvp://@www.qliktech.com/AcmeStores.qvw

qvp://john.doe@www.qliktech.com/AcmeStores.qvw

qvp://www.qliktech.com:http/AcmeStores.qvw

qvp://www.qliktech.com/AcmeStores.qvw?USERID=JOHN&PASSWORD=ABC123

qvp://www.qliktech.com/AcmeStores.qvw?MACRO=Mymacro

TIP: Internet Explorer 7 does not support @ or : in the URL in order to prevent spoofing of URLs. To specify these characters in the URL, you need to URL-encode them.

Use %3A for: and %40 for @.

6.3 File system security vs. QlikView Section Access security

NTFS Authorization or DMS Authorization mode file system security only controls which documents a client is allowed to see in the file tree and attempt to open. The documents may of course contain a script section access which further prevents or limits the client's access to the content of the document once opened.

The QlikView Windows clients and Java clients will prompt the user for section access USERID and PASSWORD when required. When using section access with QlikView USERID and PASSWORD in connection with QlikView AJAX ZFC it is necessary to make your own provisions for entering them and then pass them to the QlikView AJAX ZFC by means of URL parameters (see special sub-section titled "Using section access with QlikView AJAX ZFC" for details).

6.4 Security Configurations

There are, of course, many configuration choices available for a QlikView Server implementation. As of release 8.2, however, many new options are possible. This section will attempt to describe some of those options as examples of possible configurations.

Authentication vs. Authorization

Authentication: "Who is this user?"

The main way of authenticating a user should be

- an Operating System logon (Windows, Novell, etc), or
- any Web logon using a Directory Service.

Either way, it is made by non-QlikTech software.

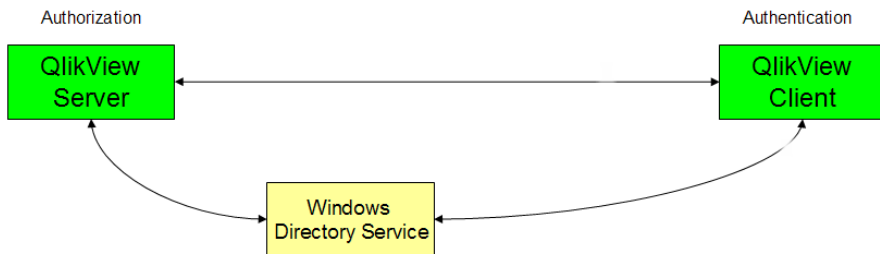
An additional authentication can be made by QlikView through the fields USER, PASSWORD or SERIAL in a Section Access of the load script.

Authorization: "What data is this user allowed to see?"

- 1 If the QlikView Server runs in **DMS Authorization** mode, the authorization is handled through the DMS thread on the QlikView Server.
- 2 If the QlikView Server runs in **NTFS Authorization** mode (legacy mode), the authorization is handled by the Windows NTFS file system. This requires that the authentication is made through Windows.
- 3 If a Custom User logon has been made, the authorization can be handled by the Publisher. This should be avoided, however, since the DMS Authorization mode above is safer.
- 4 In both modes, an additional access limitation can be defined in the Section Access of the script using e.g. NTNAME. This is handled by QlikView.

Client Side Authentication

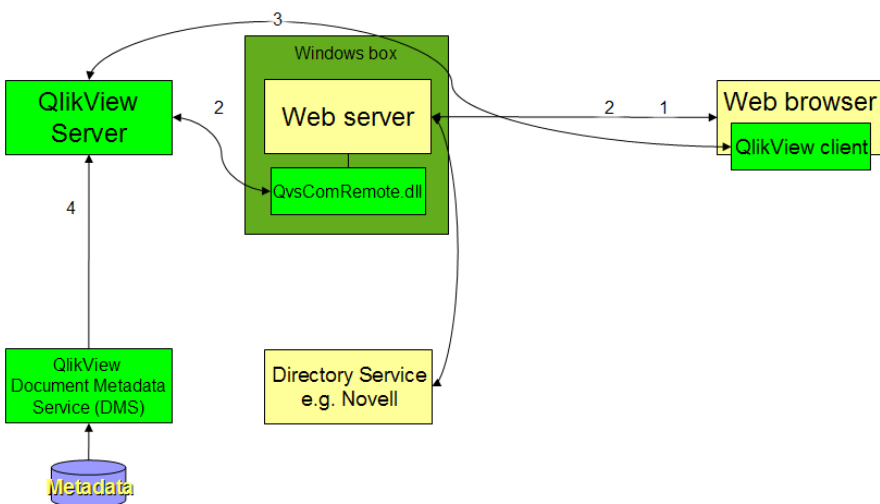
The QlikView Server is within the same Domain as the client, and a Windows Directory Service is available.



The user is already authenticated when the QlikView client is started. As the client clicks on a QlikView link, a request is sent to the QVS with the user credentials. The QlikView Server checks with the Domain Controller if the ticket is valid. The QlikView Server uses NTFS and Section Access to see if the user is allowed to see the document.

Server Side Authentication – Windows Web Server

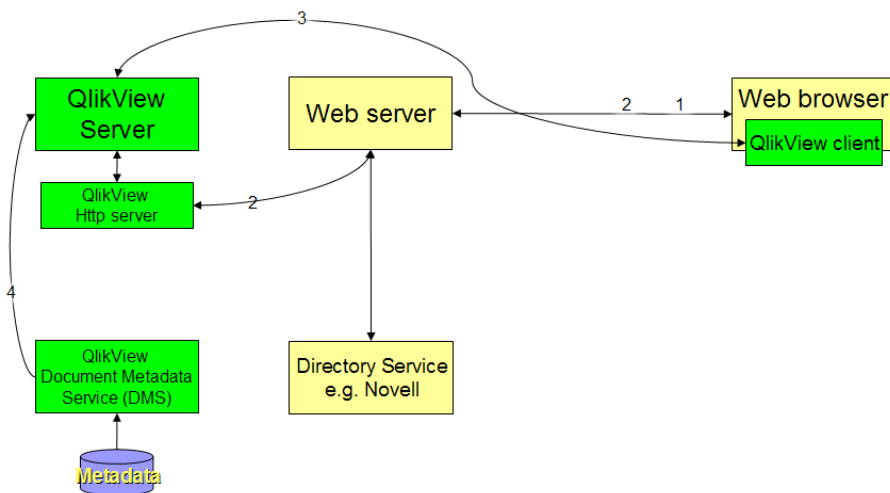
The QlikView Server will issue a ticket for authenticating a user through QvsComRemote.dll. If the user presents a valid ticket when requesting a session, the access is granted based on the user's authorization to open a document.



- 1 The client makes a call to a web server. The web server must already have an authentication system in place. Either background authentication or a log-on screen.
- 2 As the client clicks on a QlikView link, the web server uses **QvsComRemote.dll** to send a request to the QVS with the user name and gets a ticket in return.
- 3 The client launches a QlikView client that sends a request including the ticket to the QVS.
- 4 The QVS trusts the web server and thus "knows" who the user is. QVS checks with the DMS if the user is allowed to see the document.

Server Side Authentication – Non Windows Web Server

The QlikView Server will issue a ticket for authenticating a user through the QVS http web server. If the user presents a valid ticket when requesting a session, the access is granted based on the user's authorization to open a document.



- 1 The client makes a call to a web server. The web server must already have an authentication system in place. Either background authentication or a log-on screen.
- 2 As the client clicks on a QlikView link, the web server makes a web service call to the QlikView Http server with a request to the QVS with the user name and gets a ticket in return.

- 3 The client launches a QlikView client that sends a request including the ticket to the QVS.
- 4 The QVS trusts the web server and thus "knows" who the user is. QVS checks with the DMS if the user is allowed to see the document.

Server Side Authentication – Get Ticket Process

QlikView Server does not authenticate the user; it authenticates the process asking for a connection. There are two methods that can be used for authentication, Negotiated Authentication and Ticket Authentication. Once the connection is established QVS make no distinction between how the authentication was done.

Authenticated Names are required for any task where a user name is required, other than CAL assignments, which use a simple, best guess procedure.

Negotiated Authentication: This authentication will be used as the authenticated user if:

- The connection is marked as admin.

Or if all the below apply:

- The server is not set to 'Always anonymous'
- The connection is not done by ticket
- The authenticated user isn't considered as equal to anonymous (e.g. USR_...)

Negotiated Authentication will attempt to use Kerberos, but if that is unsuccessful, NTLM will be used.

Ticket Authentication: The alternate method to get an authenticated user is through ticket. See examples below for additional information.

- QVS is passed a username from a trusted source (in QVS Admin Group) and QVS trusts that authentication has happened elsewhere.
- Or, Tickets can also be obtained 'For Me' i.e. for the actual user of the asking process

Client Usage:

- The Windows client can use tickets (via QVP url) or negotiate authentication
- The Java client must pass the ticket in the applet-tag <param name="user-name" value="ticket" />

-
- The AJAX client must use the userid parameter, e.g. **http://localhost/salesdemo/AJAXzfc/?userid=510EA55C2DB723DC04C16C6FB3CDAB24F3390792**

Get Ticket examples:

There are two ways of requesting a ticket from QlikView Server, to be used in different Single Sign On (SSO) scenarios:

GetTicketForMe This will require that you are an authenticated Windows user and will generate a ticket only valid for yourself. The function takes no parameters.

To try it out type the following in a web-browser:

```
http://webhost/qvajaxzfc/  
qvsviewclient.asp?cmd=<Global%20method='GetTicketForMe'%20/>
```

In programming (ASP/VBScript), use the following:

```
set ntsecurity =  
Server.CreateObject("QVSRemote.Client")  
ntsecurity.AdminConnect "localhost"  
ticket = ntsecurity.execute("<Global  
method='GetTicketForMe' />")
```

The response will be an XML-document in the following format:

```
<Global>  
<_retval_>  
40 char hex  
</_retval_>  
</Global>
```

GetTicket This function will generate a ticket for any UserId submitted to the function.

Note Only members of the local QlikView Administrators group can retrieve a ticket. If not part of the group the function will return **<Error />**. See below for other options

Note QVS must be running in DMS-mode to be able to manage **GetTicket**. If running in NTFS-mode, no ticket will be given (No error will be returned, only an empty _retval_)

In programming (ASP/VBScript), try the following:

```
set ntsecurity = CreateObject ("QVSRemote.Client")
ntsecurity.AdminConnect "localhost"
ticket = ntsecurity.Execute("<Global
method='GetTicket'><UserId>User</UserId></Global>")
msgbox ticket
```

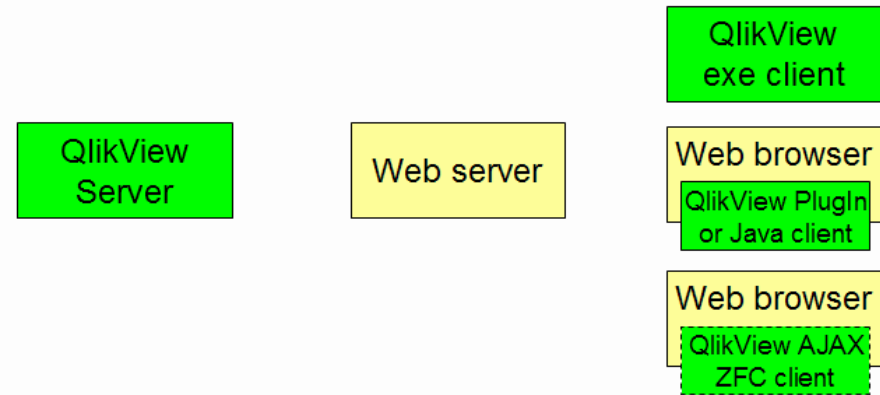
The UserID is retrieved from any other trusted authentication source.

If QVS and IIS is installed on different machines, replace "localhost" with the IP/DNS name for the QlikView Server.

7 QLIKVIEW SERVER FUNCTIONAL ARCHITECTURE

7.1 QlikView Server – Client Communication

The QlikView Server – Client communication architecture requires three primary processes, which must be able to communicate with each other in a consistent and secure manner. This interaction can potentially involve multiple computers and multiple network connections, as well as other subordinate processes.



The three primary processes are:

- 1 The **QlikView Server (QVS)**, which provides QlikView functionality to the client. The machine that is hosting this service must be running in a Microsoft Windows Operating System (refer to System Requirements section at the beginning of this document).
- 2 The **Client**, running in a web browser or a QlikView shell that provides a container for the client code. The client communicates with QlikView Server either directly or through the Web Server to provide the QlikView interface and functionality to the end user.
- 3 The **Web Server** (or HTTP), running an HTTP server, which can be used to serve up the HTML web page to the client, assist with authentication of the user, and enable communication between the client and QlikView Server.

In the simplest scenario, all three processes can be running on a single machine, with a single user. The complexity of this relationship can increase quickly, however, as separate machines, Internet connections, multiple firewalls, and multiple Web Servers are introduced. Finally, multiple users who require security authentication and authorization from a myriad of Directory Services are added, and a QlikView Server – Client communication architecture can become quite involved.

There are, of course, a large number of possible network configurations that QlikView Server can participate in, but there are a few considerations to keep in mind regardless of the final configuration:

- QlikView Server runs as a Windows OS Service only
- At least one network communication path must exist between the QlikView Server and the Client
- The authentication of the Client user must be performed either through Windows Authentication, QlikView Authentication (section access), or any third party system that can authenticate the user.

QlikView Server Functional Description

There will be one QlikView Server process per logical computer, which must be running a Windows Operating System. QlikView Server can run as a 32-bit or 64-bit process (OS and hardware dependent). The QlikView Server process can be identified as qvs.exe.

Client Access License (CAL)

All client access to QlikView Server must be licensed. This is accomplished through the use of Client Access Licenses (CALs) linked to the specific instance of the QlikView Server through the LEF file. In this context, it is important to understand the definitions of anonymous user and authenticated user.

Anonymous user – an unidentified or unknown user (any user).

There is no authentication for anonymous users, they can be anyone.

Authenticated user – an identified user whose identity can be verified.

Authenticated Windows OS user (e.g. NTNAME, NT User, NTDOMAINSID)

* Authenticated non-Windows user

Authenticated QlikView user (e.g. section access: USERID, PASSWORD)

- * Authenticated third party (build partner) user
- * As of release 8.2+

The type of CAL will affect how users are allowed to connect to QlikView Server, based on the Client type and Authentication settings in the Web Server and/or QlikView Server.

A SESSION CAL allows any user – authenticated or anonymous – to connect to QlikView Server. There is no limit on the number of sessions over time, but only one session at any time is allowed per session CAL.

A USAGE CAL allows any user – authenticated or anonymous – to connect to QlikView Server. Each usage CAL is linked to a specific user – document session, and can only be initiated once per 28 day cycle.

A USER CAL allows only specific users (or machines) to connect to QlikView Server. The Java and AJAX Clients will not allow anonymous OS users or machine names as user CALs. Therefore, anonymous access should be disabled in the Web Server (or in QlikView Server if using the QVS http web server) when using these clients.

Client Functional Description

QlikView Server can support three categories of Clients:

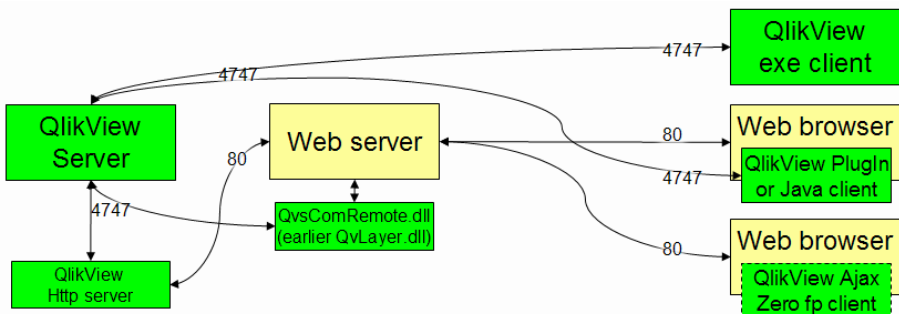
- 1 **Windows Clients** – this includes the QlikView program variants of QlikView Analyzer, Analyzer+, Professional, and Developer. This category also includes the Internet Explorer plug-in ActiveX client running as a full window or object only (QlikX). All Windows Clients require installation with Administrator level rights. QlikView Analyzer+, Professional, and Developer require licensing on the Client machine in addition to the QlikView Server CAL.
- 2 **Java Clients** – this includes Sun/Apple Java running in a browser window, with either a full window or objects support. Java Clients require installation of the Java applet, although this installation is handled automatically through the Web Server, and does not require user intervention. No Client side licensing is required.
- 3 **AJAX (ZFC) Clients** – this includes the AJAX Client, which supports HTML objects only. No Client side installation or licensing is required.

Client Communication to QlikView Server

Windows Clients can communicate directly with QlikView Server, using QVP (QlikView Protocol) and, typically, port 4747. These clients do not require a Web Server to establish and maintain a connection with QlikView Server.

Java Clients can communicate directly with QlikView Server, typically, over port 4747, but they must first establish a connection with the Web Server (e.g. IIS or the QVS http web server) serving the page containing the Java applet, typically, over port 80 (http).

AJAX (ZFC) Clients can never communicate directly with QlikView Server. They must establish and maintain a connection through a Web Server (e.g. IIS or the QVS http web server). This is typically accomplished through port 80 (http).



Web Server Functional Description

Traditionally, the standard web server in a QlikView Server configuration has been Microsoft Internet Information Services (IIS). As of QlikView version 8, however, an alternative solution is offered, and is included with the QlikView Server installation. This is the QVS http web server. This web server can act as a stand alone service, but is not configured to handle asp pages.

Other web servers can be utilized in a QlikView Server environment, but there are some restrictions. If the other web server is able to direct traffic to the QVS http web server (running on the same machine as QlikView Server), the possibilities are many, including the configuration with the other web server running under a non-Windows OS. If the other web server must utilize a local QlikView Server dll (QvComRemote.dll) to communicate with QlikView Server (e.g. for tunneling), then the other web server must be running under a Windows OS.

Web Server on Separate Machine from QlikView Server

If the IIS or QVS http web server is running on a separate machine from the QlikView Server, you will need to configure the location of the QlikView Server, and optionally, the port, to allow the web server to locate the QlikView Server. The configuration requirement will vary, based on which web server you are using.

IIS web server

Edit the file **qvclients\settings.js** to point to the QlikView Server, and optionally, the port. Change the vars **qvshost** and **qvport** to match your environment, and remove the comments. The **qvsviewclient.asp** is configured to include the **settings.js** code, but you will need to remove the comment tags.

QVS http web server

Edit the file **httpserver\config.xml** to point to the QlikView Server. Change the tags **qvshost** and **qvstunnel** to match your environment.

7.2 QlikView Server Tunnel

If the standard communication port to QlikView Server (4747) is blocked in any way (typically by a firewall limitation), the Windows and Java Clients will attempt to re-route their connection through port 80 (http). This connection path must then include the QvsComRemote.dll or the QVS http server so that the QlikView Tunnel communication can be established. All communication through the QVS Tunnel must include the secure communication packet, so this will significantly increase the network traffic (along with response times) required between the QlikView Server and the client.

The QlikView Tunnel is installed into the Web Server process and allows the QlikView Client to be tunneled over the HTTP protocol to the HTTP process and then forwarded onwards to the QVS process.

When there is a requirement for the HTTP process to run on a third machine (perhaps since it is not a Microsoft Windows server) but communication between the Client and the HTTP machine is restricted, then the setup is similar. The HTTP machine having a Tunnel installed to redirect the QlikView Client protocol on the QVS machine. Communications between the QVS and HTTP cannot be restricted in any way.

Finally, if the HTTP process must run on a third machine and communication between the Client and HTTP machine is not restricted in any way, then another process can come into play. This is a TCP/IP Redirector (or Redirect) that runs on the

HTTP machine. It is required because (in the case of Java) the Client applet can only connect to the machine that served the web page containing the applet. The redirect process accepts the connection from the applet for the QlikView Client protocol and forwards it onto the actual QVS machine. The Redirect process may be a separate program, part of the Operating System of the HTTP machine or even a function of the firewall/proxy system in use between the HTTP machine and the Client machine. All that matters is that both the machine name and the IP address of the Redirect is the same as the HTTP machine.

For Tunneling on a Windows Server using IIS or the QVS built-in http server

If you have defined a path to the QVSTunnel.dll in the Control Panel, you have the possibility to tunnel the communication between the server and the client. The dll-file is by default copied to the following directory during installation:

C:\Program Files\QlikView\Server\QvTunnel

A virtual directory is set in IIS and the QVS built-in http server configuration as:

Scripts

If the client cannot connect via the default TCP connection, the client will by default try to connect via http (Port 80).

QVS HTTP:

Edit the Config.xml file to specify the location of the <QvsHost> and <QvsTunnel>. <QvsHost> is used in all non-tunnel-cases and <QvsTunnel> when tunnelling is requested. It is thus possible to have one Qvs handling all non-tunneling and another handling tunnelling. Note that if you omit <QvsTunnel> the HttpServer will NOT support QVS tunnel.

```
<?xml version="1.0" encoding="utf-8" ?>
  <Config>
    <QvsHost>HIC-HP</QvsHost>
    <QvsTunnel>HIC-HP</QvsTunnel>
    <Url>http://HIC-HP/</Url>
```

Microsoft IIS:

Two entries are required in the registry:

[HKEY_LOCAL_MACHINE\SOFTWARE\QlikTech\QlikTunnel]

"QVSPort"=dword:000012a6

"QVSServer"="QvsHost"

The QVSPort entry should already exist, but the QVSServer must be added manually.

Tunneling from Windows clients

Tunneling from Windows clients is achieved by adding :http as protocol in the pseudo-URL describing the server or document address.

QlikView Tunnel Test Procedure

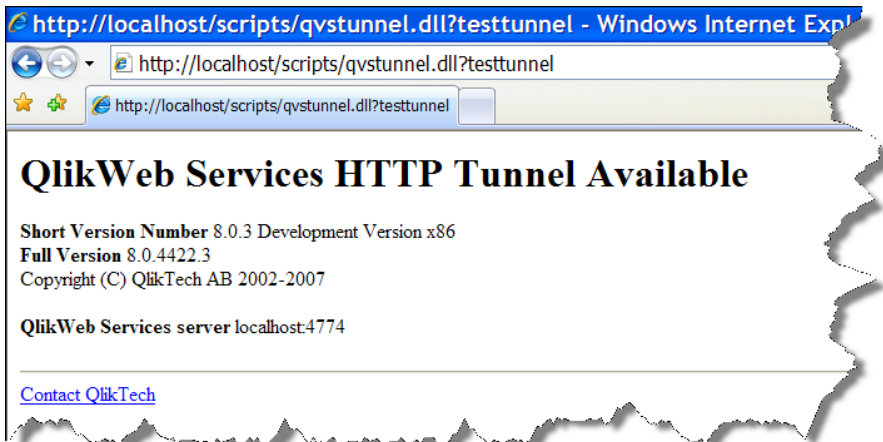
You can test the QlikView Tunnel by entering the following URL from a Client browser window:

http://Server/scripts/qvstunnel.dll?testtunnel

Where

Server is the Web Server name or address

If the QlikView Tunnel is set up correctly, the webpage should return with:



8 SERVER LOGGING

8.1 Logging from QlikView Server

Alerts from QlikView Server will appear in the Windows event log.

More detailed logs for sessions can be found in the logging directory specified on the **General** page of the QlikView Server Management Console. The default location is **C:\Documents and Settings\All Users\Application Data\Qlik-Tech\QVS**.

Log files can be set to split (create new) daily, weekly, monthly, yearly or never. Performance log intervals can be set from 1 minute and higher.

A QlikView document designed to load data from the log files and support analysis is provided on the default installation of QlikView Server. This file is named **QvServerPerformance.qvw** and is located in the QvsDocuments folder.

8.2 The Session log

The session log is updated each time a session ends. A session is defined as a single user connected to a single document. The file name of the session log is **Sessions*.log**, where * reflects the server name and the split interval. Each entry of the session log will contain the following fields.

Session Log filed	Explanation
Exe Type	Type of QlikView Server build Example: 'RLS32' = 32-bit release build
Exe Version	Full version number of QlikView Server. Example: '8.0.4366.0409.10'
Timestamp	Date and Time when log entry was created
Document	QlikView document accessed
Document Timestamp	File timestamp of document accessed
QlikView User	QlikView section access UserID (if used)

Session Log filed	Explanation
Exit Reason	Reason for session termination: ‘Socket closed’= client induced termination. ‘LRU’=terminated as Least Recently Used in favor of new user ‘Shutdown’=server induced termination for other causes Additional values exist, but should normally not occur.
Session Start	Time when session was started
Session Duration	Duration of session in hours:minutes:seconds
CPU Spent (s)	CPU-seconds spent by session
Bytes Received	Bytes received by server during session
Bytes Sent	Bytes sent by server during session
Calls	Number of QlikView calls during session (bidirectional)
Selections	Number of QlikView selections made during session
Authenticated user	Authenticated Windows NT UserID (if it exists)
Identifying user	User identification for client
Client machine identification	Machine identification for client
Serial number	Serial number of QlikView client (QVA+, QVP or QVE installed client only)
Client Type	Type of client used ‘Windows exe’=Windows client ‘Java’=Java client
Secure Protocol	‘On’ when encrypted communication is used (typically Windows clients=. ‘Off’ when non-encrypted communication is used (typically Java clients)
Tunnel Protocol	‘Tunnel’ when QVS tunnel communication is used.
Server Port	Port used by server.
Client Address	Client IP number
Client Port	Client port

Session Log filed	Explanation
Experienced Performance	A measure indicating how large portion of the CPU power that could theoretically be utilized by the session's needs that was actually available to it. The closer you get to the (theoretical and unachievable) maximum value of 100 the less the session had to wait for other sessions, server overhead etc. The value will vary between different types of documents. If you get consistent low readings in this column or if you get low readings for a specific document at certain times of the day, you should consider expanding server capacity.
Cal Type	Type of Client Access License used 'User'=User CAL 'Session'=Session CAL 'Usage'=Usage CAL
Cal Usage Count	Count of Usage CALs

8.3 The Performance log

The performance log is updated at an interval set on the **Install** page of the QlikView Server Control Panel . The default interval is 5 minutes. Additional entries are made whenever the server is started or stopped. The file name of the session log is **Performance* . log**, where * reflects the server name and the split interval. Each entry of the log will contain the following fields.

Performance Log field	Explanation
Exe Type	Type of QlikView Server build Example: 'RLS32' = 32-bit release build
Exe Version	Full version number of QlikView Server Example: '8.0.4366.0409.10'
Timestamp	Date and Time when log entry was created
EntryType	Type of entry. 'Server starting' denotes startup. 'Normal' denotes normal interval log entry. 'Server shutting down' denotes shutdown
ActiveDocSessions	Number of document sessions* that have shown activity during the interval and still exist at the end of the interval
DocSessions	Total number of document sessions* that exist at the end of the interval

Performance Log field	Explanation
ActiveAnonymousDocSessions	Number of document sessions* with anonymous user that have shown activity during the interval and still exist at the end of the interval
AnonymousDocSessions	Total number of document sessions* with anonymous user that exist at the end of the interval
ActiveTunneledDocSessions	Number of document sessions* with tunneled connection that have shown activity during the interval and still exist at the end of the interval
TunneledDocSessions	Total number of document sessions* with tunneled connection that exist at the end of the interval
DocSessionStarts	Number of document sessions* that have been initiated during the interval
ActiveDocs	Number of documents loaded at the end of the interval in which there has been user activity during the interval
RefDocs	Number of documents loaded at the end of the interval for which there is a session at the end of the interval
LoadedDocs	Total number of documents loaded at the end of the interval
DocLoads	Number of new documents loaded during the interval
DocLoadFails	Number of documents that have failed to load during the interval
Calls	Total number of calls to QlikView Server during interval
Selections	Number of selection calls during interval
ActiveIpAddr	Number of distinct IP-addresses that have been active during the interval and still exist at the end of the interval. Note that tunneled sessions and multiple users originating from the same IP cannot be distinguished
IpAddr	Total number of distinct IP-addresses connected at the end of the interval. Note that tunneled sessions and multiple users originating from the same IP cannot be distinguished
ActiveUsers	Number of distinct NT users that have been active during the interval and still exist at the end of the interval. Note that anonymous users cannot be distinguished here

Performance Log field	Explanation
Users	Total number of distinct NT users connected at the end of the interval. Note that anonymous users cannot be distinguished here
CPUload	Average CPU load from QlikView Server during interval
VMAllocated(MB)	Size in MB of virtual memory allocated by QlikView Server at the end of the interval **
VMCommitted(MB)	Size in MB of virtual memory actually used by QlikView Server at the end of the interval. This number is part of VMAllocated(MB) and should not exceed the size of the physical memory in order to avoid unacceptable response times
VMFree(MB)	Size in MB of unallocated virtual memory available to QlikView Server **
VMLargestFreeBlock(MB)	Size in MB of the largest contiguous block of unallocated virtual memory available to QlikView Server. This number is part of VMFree(MB)
UsageCalBalance	'-1.00' denotes no Usage CALs exist

* one user + one document = one document session

**VMAllocated(MB)+ VMFree(MB) = total maximum virtual memory space available to the QlikView Server process.

8.4 The Event log

The event log is updated each time a log entry is made to the Windows event log from QlikView Server. The information stored is a mirror of the information written to the Windows event log. The file name of the event log is Events*.log, where * reflects the server name and the split interval. Each entry of the log will contain the following fields.

Event Log field	Explanation
Timestamp	Date and Time when log entry was created
SeverityID	Unique ID of severity level 1 = Error 2 = Warning 4 = Information
EventID	Unique ID for the type of event

Event Log field	Explanation
Severity	Severity level of event Error Information Warning
Message	Description of the event

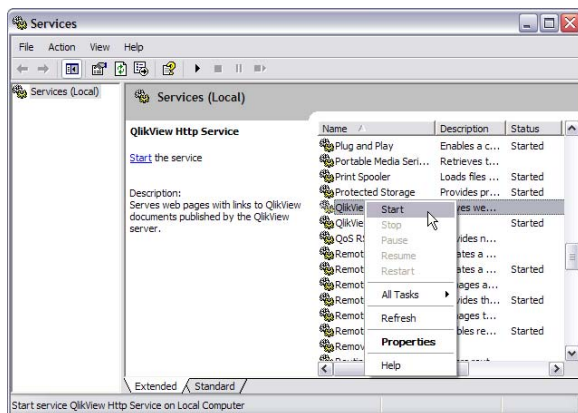
9 QLIKVIEW SERVER BUILT-IN HTTP WEB SERVER

An http web server is available with QlikView Server. When running QlikView Server on Windows XP/Windows Server 2003 or later, the built-in http web server can provide an alternative to MS IIS.

9.1 Starting the QVS built-in http web server

The built-in http web server is installed as a Windows service during a default Complete installation of QlikView Server. To start the server, use the Windows Services dialog. This is accessible from the Windows Control Panel ... Administrative Tools, or from **Start .. Run ... services.msc**

In the Services dialog, scroll down to find the **QlikView Http Service** entry and select it. You can either right-click, and select **Start**, or click on the **Start the Service** link to the left.



Right-click on the service, and select **Start**.

9.2 Displaying sample pages with the QVS built-in http web server

The built-in web server supports the HTML sample pages distributed with QlikView Server, as well as tunneling, Windows Authentication and external authentication.

This web server will not support .asp pages. It can be used in place of IIS for the operation of QlikView Server.

Configuring the QVS Http Service

You may configure the web server through a configuration file located in the following location:

C:\Program Files\QlikView\Server\HttpServer\config.xml

The QVS Http Service uses the QvsNetRemote.dll to support encryption and machine-id. This file is located in the same location as the config.xml file.

The **config.xml** file contains a commented out section to simplify the usage of common but non-default options.

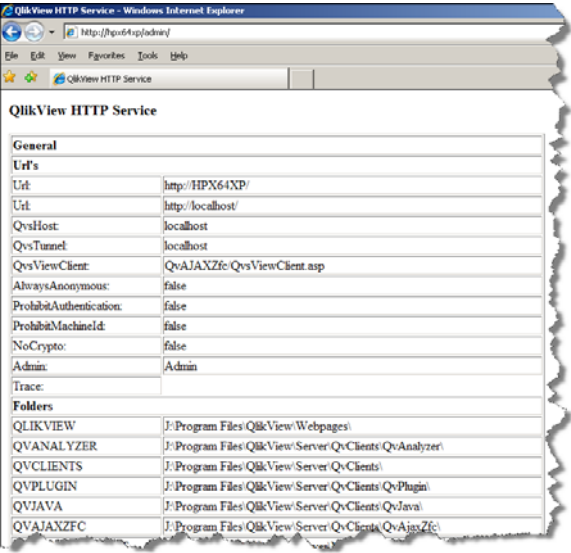
As of 8.5, new configuration options are available:

- Multiple **<Url>** tags to let the service listen to more than one Url. The default **config.xml** comes with two tags **<Url>http://localhost</Url>** is added
- A simplified syntax is possible for **QvsTunnel** and **AlwaysAnonymous** - **mouse<QvsTunnel/>** to allow tunneling to the same QVS as already specified in **<QvsHost>**
<AlwaysAnonymous/> instead of
<AlwaysAnonymous>true</AlwaysAnonymous>

New (or previously undocumented) keywords are as follows. To utilize these functions, copy or move the tag to an uncommented section in the config.

- **<ProhibitMachineId/>** prohibit sending machineid. This will effectively exclude the usage of anonymous bookmark (a new collaboration feature in 8.5)
- **<ProhibitAuthentication/>** prohibit any authentication (including access to the Management Console) through the QVS Http Service
- **<NoCrypto/>** prohibit the use of encryption between the QVS Http Service and the QVS. This is useful when encryption, for some reason is not applicable.

- **<Admin/>** make it possible to see the current configuration through a web browser (e.g. **http://localhost/Admin/**) The URL will work only through the first **<Url>** tag present.



10 SERVER REPOSITORY FOR SHARED OBJECTS

10.1 Types of Objects Available for Sharing

There are multiple objects available for user collaboration and sharing through QlikView Server.

- Bookmarks
- Sheet objects, including Charts
- Reports

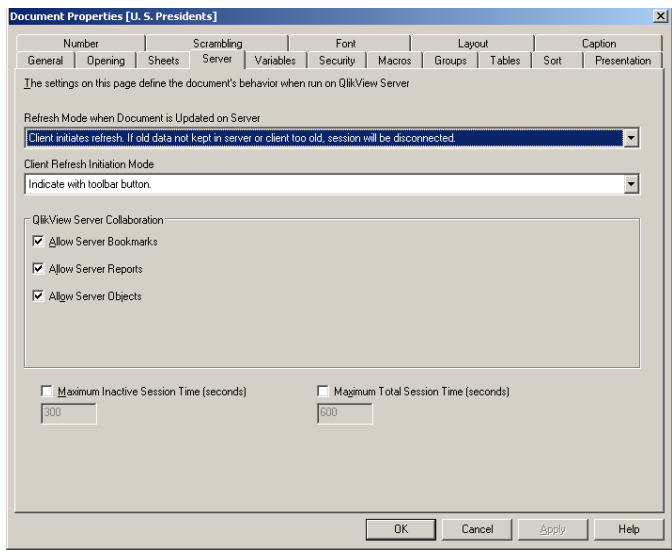
Each of these objects may be defined as a user object, available to the authenticated user, regardless of access method or location, or a shared object, available to all users of the document through QlikView Server.

10.2 Settings required for Collaboration

Client based bookmark, object and report creation is limited as follows:

- Server and client are QlikView version 8 or later
- Windows clients (QlikView or plug-in) only
- User authentication is required
- User must be licensed to at least Professional Level on the machine being used

In order for QlikView document objects to be enabled for sharing, the document must be set to allow Server Collaboration on each of the object types. This is the default setting for documents in QlikView 8.



The **Server** page of the QlikView Document Properties dialog.

Allow Server Bookmarks

This check box must be enabled, if remote clients are to be allowed to create and share bookmarks with this document on the QlikView Server.

Allow Server Objects

This check box must be enabled, if remote clients are to be allowed to create and share sheet objects with this document on the QlikView Server.

Allow Server Reports

This check box must be enabled, if remote clients are to be allowed to create and share reports with this document on the QlikView Server

TIP: For more information on Server Collaboration settings in QlikView, please consult the QlikView Reference Manual.

In addition, QlikView Server must be set to **Allow Server Collaboration**. Set this on the **General** page of the Server Management Console . If the Server is set to **disallow**, this setting will override the Document settings for all documents on that server.

Once QlikView Server is enable for collaboration, and any of the QlikView Server Collaboration settings are checked, and the document is opened in QlikView Server, a special database file will be created and maintained in the same location as the QlikView document. The file will have the same name as the QlikView document, but will have a file extension of .Shared.

For example:

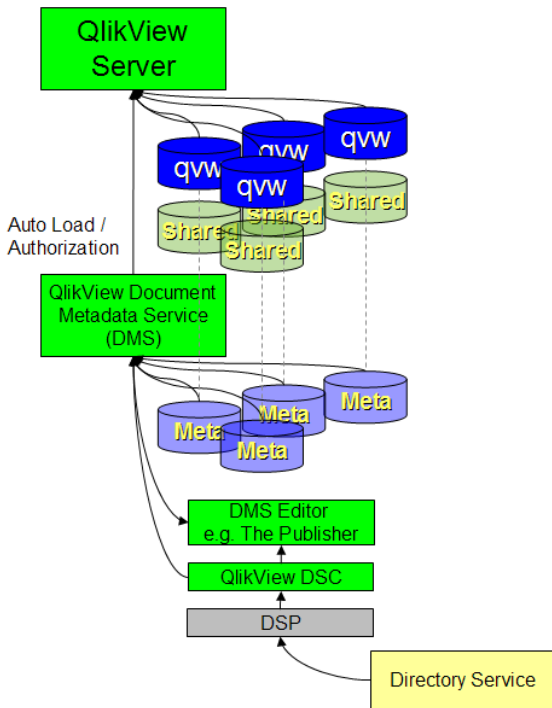
QlikView document: **Presidents.qvw**

QlikView Server share file: **PRESIDENTS.QVW.Shared**

If the name of the QlikView document is changed for any reason, you will have to manually rename the .Shared file to match before opening the newly named QlikView document in QlikView Server. This will preserve the shared objects attached to the document.

11 DOCUMENT METADATA SERVICE (DMS)

Document Metadata Service (DMS) is a thread running in the QlikView Server service. There are two distinct and separate functions provided by DMS. The first function is used to control automated document loading and unloading. The second function used to control the authorization mode that QlikView Server will run in. DMS Authorization mode can be used to enable third party (non-Windows) authenticated users to access documents through QlikView Server, as well as specification of specific access rights for individual or anonymous users, along with access days and times.

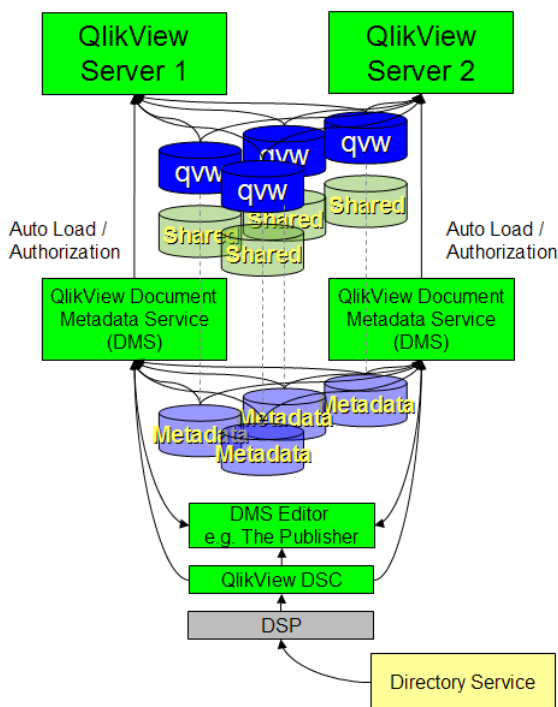


The DMS will utilize *.meta files stored alongside the QlikView documents to contain the automatic load and user/group authorization data. The *.meta files will have the same base filename of the QlikView document, with the .meta extension added, e.g. Presidents.qvw.Meta

The QlikView Server DMS will work in conjunction with the QlikView Publisher Enterprise Directory Services Connector (DSC) and Directory Services Provider (DSP) to populate, maintain, and enforce the document/user authorization configuration. Note that Groups can only be resolved through the use of the QlikView Publisher Enterprise DSC and DSP. The DMS Repository can also be maintained through the DMS dialogs available on the Security (Communication) page in the QlikView Server Management Console, as well as the Script API, which is documented in the Appendix Chapters. Note that the QlikView Publisher Enterprise DSC, if enabled, will have write priority over the DMS manual entry dialogs.

12 QLIKVIEW SERVER LOAD SHARING (CLUSTERING)

QlikView Server will support load sharing of documents across multiple physical or logical computers. This sharing includes the ability to share in real time, information about collaboration objects, automated document loading and unloading (through DMS), and user license CALs. Special licensing is available to enable multiple server instances to share the same license number.



In order to utilize load sharing between multiple QlikView Servers, all document and support files must be shared between the servers. In other words, all servers should point to the same physical location for the files. In addition to the file types described in the diagram above, QlikView Server will create and maintain additional files to store load sharing data. These files will have a file type extension of .pgo (Persistent Group Object), and they will be located in the same folder as the QlikView documents.

Operating System Load Balance or Failover configurations are external to the QlikView Server load sharing configuration, and QlikView Server has no control over those systems.

Server configuration settings are shared between all clustered QVS, and can be maintained through the QlikView Server Management Console connected to any of the clustered QVS. Performance of a particular QVS system can be monitored through the Management Console by connecting to that system.

Since DMS data is shared among the QVS, any automated document load/unload procedures are performed on all QVS. DMS Authorization is, of course, also shared among all clustered QVS.

PART II: CLIENTS

- Client overview
- Sample HTML
- HTML pages
- QlikView Java Client
- QlikView Java Objects Client
- QlikView installed Windows Client
- QlikView Analyzer for Internet Explorer
- QlikView AJAX Zero-Footprint Client (ZFC)

13 SUMMARY OF QLIKVIEW CLIENTS

A QlikView client is required for display and usage of an existing QlikView document (.qvw file). While the QlikView Server is responsible for opening, hosting, and calculating the document, the clients are required for user interaction and presenting the document and its objects.

The choice of which client or set of clients to use is entirely dependent on the customer's environment and preferences. Client choices range from a fully installed QlikView Developer to a (no installation required) AJAX Zero-Footprint Client (ZFC). Any combination of client types is allowed, as long as the proper licensing CALs are available (see Section QVS Licensing) on the server license. In general, any QlikView document may be displayed with any client, although additional HTML code must be created for the Object clients, including AJAX, to display and interact with the objects within a particular document. In addition, certain considerations must be taken into account when deploying a QlikView document with QlikView Server and its clients. See Section Considerations when developing documents for use with QlikView Server for information on limitations when using a QlikView client to interface with a QlikView document.

The following table provides a brief summary description of client choices.

QlikView Client	Description
Java Client	Full image client utilizing Sun Java in web browser. No installation or client side licensing required.
Java Objects Client	Object only client utilizing Sun Java in web browser. Requires web page design or QlikView Client Generator. No installation or client side licensing required.
Windows Installed Client	Full image desktop installed QlikView (Developer / Professional / Analyzer / Analyzer+). Installation and client (except for Analyzer) licensing required.
Analyzer for Internet Explorer	Full image ActiveX plug in for Internet Explorer web browser. Installation required. No client side licensing required.
QlikX Analyzer for Internet Explorer	Object only ActiveX plug in for Internet Explorer web browser. Requires web page design or QlikView Client Generator. Installation required. No client side licensing required.
AJAX Zero-Footprint Client (ZFC)	Object only Dynamic HTML client utilizing AJAX architecture in web browser. Requires web page design or QlikView Client Generator. No installation or client side licensing required.

The table below shows the different client varieties and some of their most important positioning properties.

Clients for publishing QlikView documents	Clients for building web applications with QlikView data and logic	Layout fidelity and functionality	Demands on client environment and install bandwidth
Installed EXE clients (QVE, QVP, QVA)	-	High ↓ Low	High
QVA for IE (plug-in)	QlikX:s (part of plug-in)		
QVA Java client	QVA Java Objects client		Medium
-	AJAX Zero Footprint Client (ZFC)		Low

The leftmost column in the table contains client variants where entire QlikView documents with sheets and layout can be presented without the need for web page design.

The second column shows client variants which require web page design.

14 ABOUT THE SAMPLE PAGES

14.1 Sample HTML Pages

If you selected **Complete** for Installation Type (default) or **Custom**, and included the QlikView Server example web pages and documents, several sample HTML pages were loaded onto your server during installation in order to allow you to verify proper operation. To access these web pages, the directory where the pages were loaded must be accessible to your web server.

The default folder location for the QlikView Server sample HTML pages is:
C:\Program Files\QlikView\Webpages

This location has a URL of **http://<server>/QlikView/**

If you cannot display the sample web pages, verify that there exists a virtual directory named **QlikView** defined in your web server, and pointing to the default folder location for the QlikView Server sample HTML pages noted above.

The QlikView Server directory has a default start page that links to subdirectories for the different client alternatives:

- QlikView Analyzer Windows Client
- QlikView Analyzer for Microsoft Internet Explorer (plug-in) Client
- QlikView Analyzer Java Client
- QlikView Analyzer Java Objects Client
- QlikView Analyzer Zero Footprint client (AJAX)

From this start page the user will be guided through the different client alternatives that the QlikView Server installation provides.

These locations are also used as the default locations for pages created by the Object Client Page Generators.

TIP: If you modify any of the default pages that were installed by QlikView Server, be sure to either keep backups of the new pages, rename the new pages, or change the location of the new pages. Failure to do so could result in losing the modifications when upgrading or re-installing QlikView Server.

showdocument.htm (Java clients only)

The HTML file named `showdocument.asp` can be used to show a QlikView document using the QlikView Java applet. This file utilizes JavaScript functions found in **/QvJava/CodeLib/build_java.js**

To choose a QlikView document and display sizes the page needs three parameters:

Document	- the QlikView file to show
Width	- the width of the applet in pixels
Height	- the height of the applet in pixels

There are also optional parameters for use in the query string for the client's network settings:

isapi	- Disable standard TCP connection and forces http tunneling
Port	Specify the communication port to QVS
Tunnelport	Specify the communication port between QVS and QVS tunnel

Example of calling `showdocument.htm`

```
<a href='showdocument.htm?width=1000&height=700&document=Presidents.qvw'> Presidents</a>
```

Plugin_showdocument.htm (QVA for IE plug-in only)

The HTML file named **plugin.asp** can be used to initiate the Connect to Server dialog and show a QlikView document using the QlikView Analyzer for Internet Explorer.

alldocs.htm (Java clients only)

The **alldocs.htm** displays a list of all available QlikView documents available. This uses functions available in **QvDocList.js**, located in the scripts subdirectory of the Webpages location.

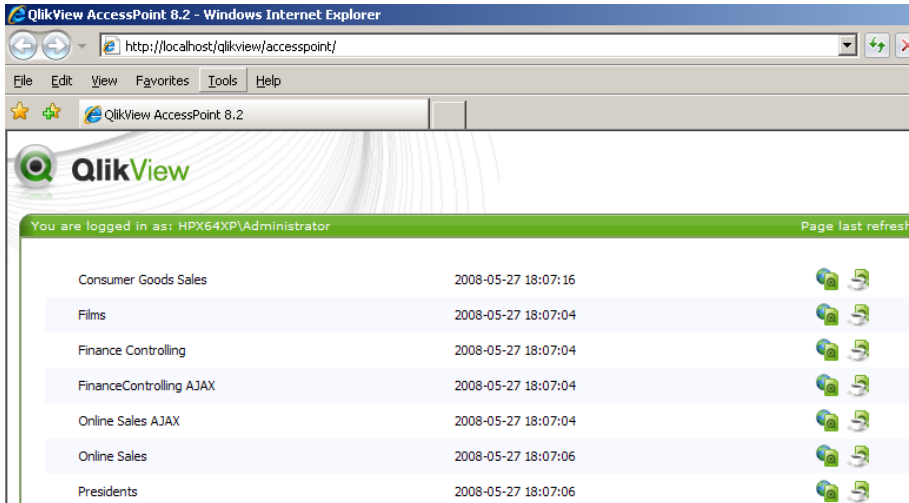
plugin_alldocs.htm (QVA for IE plug-in only)

The **alldocs.htm** displays a list of all available QlikView documents available. This uses functions available in **QvDocList.js**, located in the scripts subdirectory of the Webpages location.

14.2 QVS AccessPoint

If you selected **Complete** for Installation Type (default) or **Custom**, and included the QlikView Server example web pages and documents, a QlikView Server AccessPoint was set up under the default Webpages directory. Enter the following URL to open the AccessPoint:

http://<server>/QlikView/AccessPoint



From this page, the user may open documents with the choice of several different cli-

ents.

15 QLIKVIEW JAVA CLIENT

15.1 Java Client

The QlikView Java client is a Java applet which appears as an integrated part of the browser window. The Java Client may also be initiated in a stand alone Java window, without the use of a browser. Refer to the Java sample pages for an example of this.

The QlikView menu bar is not available, but a standard toolbar is available. Sheet and sheet object context menus are available where applicable.



QlikView Java client in Internet Explorer.

15.2 Display of application

You may create your own custom applications and load them onto the QlikView Server. You must create the application using QlikView and save the application as a QVW file.

To display the application on the QlikView Server just copy the QlikView QVW file to the QlikView Server application directory.

The default location is:

C:\Documents and Settings\All Users\Application Data\Qlik-Tech\QVS\Documents

You may have chosen a different directory during QlikView Server installation. Also, Java documents are allowed to have a separate dedicated folder.

Note The QVW file must be copied to the directory specified as **Document Folder** in the Management Console. This is not the same directory where the ASP pages are located. This directory should be secure from normal user access, and does not need to be readable by the web server (e.g. IIS.)

15.3 Collaboration - Shared Objects

Collaboration - shared objects - is not currently supported in the Java Client.

15.4 HTML Pages

You can customize the sample pages provided, or create your own. The web page can contain any normal HTML tag, but it must have room to display the Java applet window for the QlikView Java Client.

15.5 Using custom HTML to invoke applet (Advanced)

For those experienced in HTML programming there is an option to skip the sample pages altogether and write custom HTML to publish an application with the QlikView Java applet.

Basic example

A typical applet definition could look like :

```
<applet code='com.qt.qw.QlikApplet.class'
  archive='/QvJava/QlikView.jar' height=700 width=1000
  MAYSCRIPT>
  <param name='DOCUMENT' value='Presidents.qvw'>
<param name='PORT' value='4747'>
<param name='ISAPI' value='/scripts/QVSTunnel.dll'>
  <param name='HEIGHT' value='700'>
  <param name='WIDTH' value='1000'>
  <param name='LANGUAGE' value='English'>
</applet>
```

On the applet tag, you can change the WIDTH and HEIGHT values to specify the size of the applet window that is drawn in the web browser. The parameters passed to the applet control various aspects of the application. The DOCUMENT parameter value is the name of the QVW file that should be

loaded in the applet window.

The PORT parameter value must match the port specified in the QVS Control Panel.

This is the TCP port that the client and server will communicate over.

The ISAPI parameter value specifies the location of the tunneler dll, if such is to be used.

The HEIGHT and WIDTH parameter values tell the QlikView Java client application the

dimensions of the applet window available for use. Normally these values should match the values specified in the applet tag.

The LANGUAGE parameter value specifies the language of context menus in the client. Valid values are :

- "English"
- "Swedish" or "Svenska"

By adding an optional parameter <param name='HIDETOOLBAR' value=''> the client toolbar will be suppressed.

Example with Windows authentication

A minimal example with Windows authentication enabled could look like follows:

```
<%@ Language=JavaScript %>
<html>
  <%
    // Authenticate using QVSNTsecurity.dll
    var s = Server.CreateObject("QVSNTSecurity.QlikNTLia-
    son");
    var username_qlikweb = s.LogValidatedSes-
    sion(Request.ServerVariables("REMOTE_ADDR"));
  %>
```

```
<applet code='QlikWeb.class' name='QlikDocument'
id='QlikDocument' archive='/QvJava/QlikWeb.jar'
height=700 width=1000 mayscript>
  <param name='DOCUMENT' value='presidents.qvw'>
  <param name='USERNAME' value='<% =
  username_qlikweb %>'>
  <param name='PORT' value='4747'>
  <param name='ISAPI' value='/scripts/QvsTun-
  nel.dll'>
  <param name='HEIGHT' value='700'>
  <param name='WIDTH' value='1000'>
  <param name='LANGUAGE' value='English'>
</applet>
<script language='javascript'></script>
</html>
```

On the applet tag, you can change the **WIDTH** and **HEIGHT** values to specify the size of the applet window that is drawn in the web browser. The parameters passed to the applet control various aspects of the application.

The **DOCUMENT** parameter value is the name of the QVW file that should be loaded in the applet window.

The **PORT** parameter value must match the port specified in the QVS Management Console.

This is the TCP port that the client and server will communicate over.

The **ISAPI** parameter value specifies the location of the tunneler dll, if such is to be used.

The **HEIGHT** and **WIDTH** parameter values tell the QlikView Java client application the dimensions of the applet window available for use. Normally these values should match the values specified in the applet tag.

The **LANGUAGE** parameter value specifies the language of context menus in the client. Valid values are :

- **English**
- **Swedish** or **Svenska**

By adding an optional parameter **<param name='HIDETOOLBAR' value=''>** the client toolbar will be suppressed.

16 QLIKVIEW JAVA OBJECTS CLIENT

16.1 Java Objects Client

The QlikView Java Objects client consists of so called applets, one for each sheet object, which can be inserted at any position on an html page. Web pages containing Java Objects linked to a QlikView document may either be created through the QlikView **Objects Client Page Generator**, or manually created. This section describes the requirements and process for manual creation and/or maintenance of a Java Objects web page.

The QlikView menu bar and toolbars are not available in the Java Objects client. Sheet object context menus are however available where applicable.

vin - pris :: bedömning :: mat :: druva :: detaljer :: info enskilt vin ::

Typ	Vin	Land	Pris (75cl)	AoM betyg
☐ Rött	4 You Rivaner	Argentina	<60	☐ Fynd
☐ Vitt	35 South Carmenère	Australien	60-79	☐ Mer än prisvärt
☐ Rose	35 South Chardonnay	Bulgarien	80-99	☐ Prisvärt
☐ Bubbel	iSil	Chile	100-149	☐ Ej prisvärt
	A Mano Primitivo	Cypern	150-250	

vinlista :: butiklista ::

Dina val stämmer för närvarande in på 2483 viner

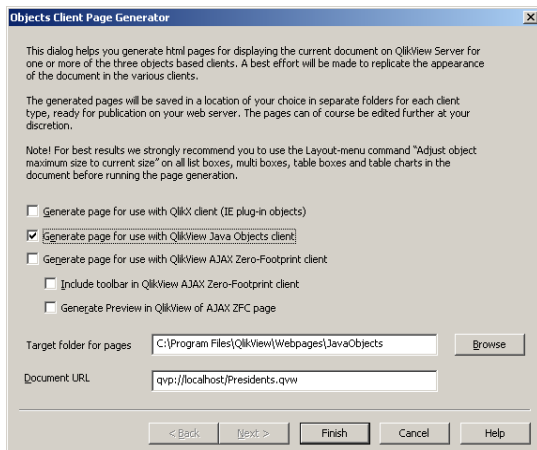
ArtikelNr	Vin	Volym	Pris
2000	Trapiche Cabernet Sauvignon	750	67,00
2001	Voorspoed Reserve Chardonnay	750	79,00
2003	Douglas Green Pinotage	750	69,00
2004	Bellingham Shiraz	750	89,00
2007	Zonnebloem Shiraz	750	79,00
2008	Veenwouden Classic	750	219,00
2009	Cuvée des Montagnards Blanc	750	59,00
2011	Côtes-du-Rhône	750	95,00
2011	Côtes-du-Rhône	1500	195,00
2012	Rijk's Chenin Blanc	750	139,00
2013	KWV Chêne	750	59,00
2014	Chinon Domaine de la Perrière	750	105,00
2020	Zonnebloem Sauvignon Blanc	750	65,00

A = Grundsoriment för alla butiker, B = Beställnings Sortimentet

QlikView Java Objects client in Internet Explorer.

16.2 QlikView Page Generator for Java Objects

The Objects Client Page Generator tool in QlikView 8+, located off the Tools menu, can be used to generate QlikView Java Objects pages based on existing QlikView documents. This is the recommended way to quickly and easily develop the QlikView Java Objects pages for your documents. The following sections in this chapter can then be used to modify those pages for those wishing to further customize the presentation.



The Objects Client Page Generator for Java Objects client.

Target folder for pages

Specifies the folder where the generated pages will be saved. Each page variant (for different client types) will be saved in a separate sub-folder in the specified location. The target folder must exist for the **Finish** button to be enabled.

Document URL

Specifies the target document URL on the QlikView Server where the pages are to be run. This setting is necessary for the Java Objects and QlikX pages to work properly.

Adjust Object Maximum Size to Current Size

Some types of sheet objects, like list boxes, multi boxes and tables, may have a larger maximum size than what is currently utilized. This allows e.g. a table to expand on the sheet when more data is added. This operation resets the maximum object size to the currently used size for all selected objects on the active sheet. You may use CTRL+A to select all sheet objects on the sheet. It is recommended to always perform this operation on all list boxes, multi boxes, table boxes and table charts in the document before using the **Objects Client Page Generator wizard**.

16.3 Objects definition

General

The option to access QlikView files through Objects gives the web designer more freedom to use the space of the HTML page more efficiently.

The solution builds on having one applet communicating with the QVS and the specified QlikView file. In turn this “master applet” communicates with several applets representing each specific QlikView object you want to display.

The solution uses the unique object ID given to each object in a QlikView document. This ID can be found in QlikView, on the **General** page in the **Properties** dialog of the sheet object. Instead of displaying the whole file as described above, the web designer can choose where on the HTML page he wants to place the object. This object can be a listbox, a graph, a button etc.

Programming languages

The examples below are written in HTML and ASP using JavaScript. This documentation is however not a manual for these scripting languages. The reader is therefore supposed to have some knowledge of HTML and ASP programming.

16.4 Applying QlikView Java Objects to a web page

The setDocument function

In order to connect the client page to the right QlikView document (qvw file) on the server, the setDocument function has to be applied. Below you find an example on how to do this.

```
<% setDocument("qwt_objects.qvw") %>
```

The addObject function

In a QlikView application, every sheet object has a unique ID. You find the ObjectID on the General tab in the Properties dialog of the sheet object.

When you apply a sheet object to the page, you call the addObject function including parameters for QlikView Document Name, ObjectID and the height and width of the actual applets that are going to display the different QlikView sheet objects. Below you find an example of how to do this.

```
<% addObject("qwt_objects.qvw", "LB01", 140, 80) %>
```

Sample QlikView Java Objects code

Below you see the sample code for a complete working QlikView Java Objects page. The sheet objects are positioned on the page using a standard HTML table.

```
<!--#INCLUDE Virtual="/QvJava/Servers/QlikWebObjects/Qlik-
Web.asp"-->
<html>
<head>
    <title>.: QlikWeb Objects :.</title>
</head>
<body>
<% setDocument("qwt_objects.qvw") %>
<table border="0" cellpadding="0" cellspacing="0">
    <tr>
        <td>
            <% addOb-
            ject("qwt_objects.qvw", "LB01", 140, 80) %>
        </td>
        <td>
            <% addOb-
            ject("qwt_objects.qvw", "LB02", 250, 100) %>
        </td>
    </tr>
    <tr>
        <td>
            <% addOb-
            ject("qwt_objects.qvw", "LB03", 140, 140) %>
        </td>
        <td>
            <% addOb-
            ject("qwt_objects.qvw", "CH01", 250, 150) %>
        </td>
    </tr>
</table>
</body>
</html>
```

16.5 Language and bookmark settings

If you use the following functions before the setDocument function you will be able to either change the language of the session (menus and error messages) or set what bookmark that should be used as an initial selection. The functions are:

setLanguage(lang)

setStartBookmark(bookmarkname)

Possible languages are English and Swedish

The following example shows how these functions are used by the Java clients. By calling these functions the bookmark used at the start of the session will be "quarter1". The session language will be set to Swedish.

```
<% setStartBookmark ("quarter1"); %>  
<% setLanguage ("swedish"); %>  
<% setDocument("sales.qvw"); %>
```

17 QLIKVIEW ANALYZER FOR INTERNET EXPLORER

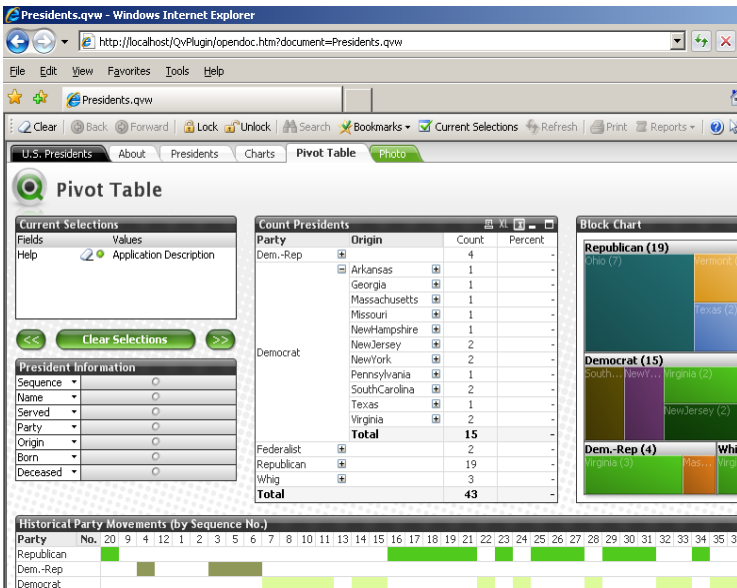
17.1 Plug-in Client

A plug-in is a program hosted by and running inside a web browser. Typically it consists of an ActiveX component with ocx as file extension. Acrobat Reader is a common example of a plug-in that computer users should be familiar with. QVA for IE installs and operates along exactly the same principles.

QlikView Analyzer for Internet Explorer is a freely downloadable program and can easily be distributed throughout the company via the sample HTML pages provided with the installation of QlikView Server.

The QVA for IE client appear as an integrated part of the MS Internet Explorer window. No QlikView menu bar is available, but most of the QlikView toolbar functions are available. Sheet and sheet object context menus are available where applicable.

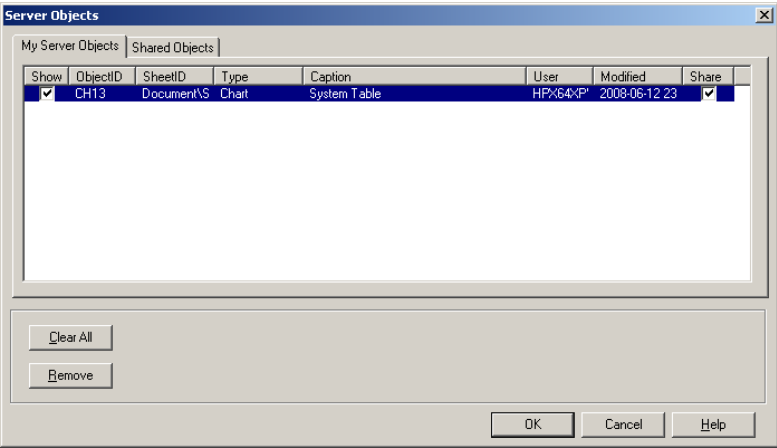
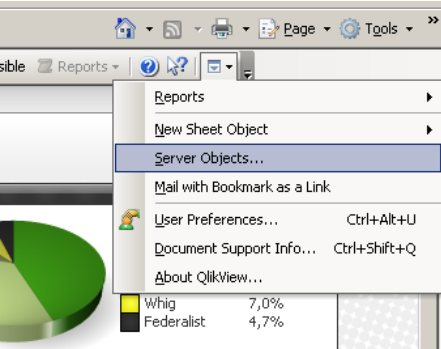
The functionality of QVA for IE is otherwise identical to that of the unregistered version of QlikView Analyzer. Local files are not accessible from QVA for IE..



QlikView plug-in client in Internet Explorer.

17.2 Collaboration - Shared Objects

Collaboration – shared objects – is supported in the Plugin Client for authenticated users. Sheet objects may be created, moved, and sized. Use the standard right-click menu and select **New Sheet Object**. Reports can be created and existing reports can be edited. New objects, reports and bookmarks may be shared with other users through the **Server Objects** dialog. Locate this option off the Menu Options toolbar icon.



Use the **Share** option to share the selected objects with other server users. Access to the object through this dialog will move to the **Shared Objects** tab.

17.3 Deployment of QVA for IE

There are three basic ways to get the QVA for IE plug-in installed on a client computer :

Simple install link on web page

Clicking on a **download and install** link on a web page starts the installation. This is a very common approach used for many commercial plug-ins.

It is necessary that the user installing the plug-in has install privileges on their own machine in order for this approach to work.

Automatic install from asp page

Clicking on a link to a QlikView document on an HTML page encoded to detect the presence of QVA for IE and install if not present starts the installation. The sample HTML pages supplied with QlikView Server contain such links.

It is necessary that the user installing the plug-in has install privileges on their own machine in order for this approach to work.

Pushing out client via group policy

In organizations where individual users are not allowed to install new software on their computers, the system administrator may create a so called "group policy" and "shoot out" the QVA for IE clients to any set of computers in the directory.

17.4 Customizing settings for QVA for IE

The appearance and behavior of the QVA for IE client can be changed via a couple of special commands, as follows.

Selecting toolbars

The client comes with a predefined toolbar for navigation and bookmarks. The toolbar may be customized by clicking on the **Toolbar Options** icon located to the right of the toolbar.

Setting User Preferences

Use the **Show Menu** toolbar icon to display the **User Preferences** selection. This will allow setting user preferences, such as language and printing preferences. For a detailed explanation of each of the available commands, please refer to the QlikView Reference Manual.

18 QLIKX - PUBLISHING SEPARATE SHEET OBJECTS FROM THE QVA FOR IE PLUG-IN CLIENT

18.1 Technical description of the QlikX concept

The QVA for IE plug-in has the ability to support the display of individual sheet objects on a web page, without the surrounding sheet environment. Sheet objects from different QVS documents can be displayed on the same page. Web pages that are to utilize this functionality may either be created through the QlikView **Objects Client Page Generator**, or manually created. This section describes the requirements and process for manual creation and/or maintenance of a QlikX web page.

18.2 Limitations

The following conditions must be met for the QlikXs to work:

- MS Internet Explorer version 6 or later must be installed on the client computer
- The QlikView Analyzer for Internet Explorer plug-in must be installed on the client computer

18.3 Getting it to work

This section provides an outline of the steps necessary to build a web page with QlikX objects. It is assumed that the reader possesses a general knowledge of HTML.

Infrastructure

Server environment

QlikView Server must be of version 7 or later.

Web page components

HTML web page

The basic HTML web page(s) defining the client typically reside in a directory somewhere under the web server's wwwroot, either directly, or indirectly through the QlikView virtual directory. The pages could contain any

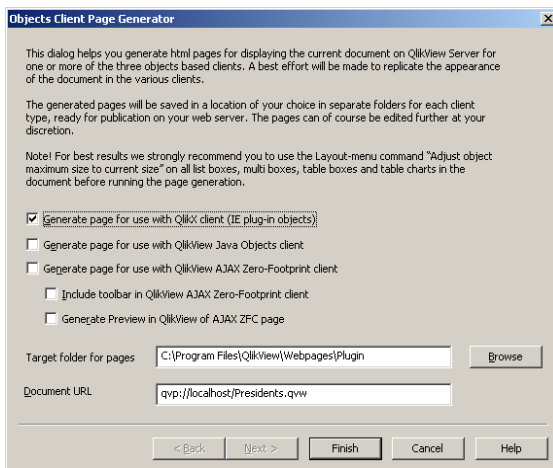
standard HTML code. The details of how to present QlikX sheet objects is described in the next chapter.

Plug-in

The QlikView Analyzer for Internet Explorer plug-in must be installed on the client computer.

18.4 QlikView Page Generator for QlikX

The Objects Client Page Generator tool in QlikView 8+, located off the Tools menu, can be used to generate QlikView QlikX pages based on existing QlikView documents. This is the recommended way to quickly and easily develop the QlikView QlikX object pages for your documents. The following sections in this chapter can then be used to modify those pages for those wishing to further customize the presentation.



The Objects Client Page Generator for QlikX client.

Target folder for pages

Specifies the folder where the generated pages will be saved. Each page variant (for different client types) will be saved in a separate sub-folder in the specified location. The target folder must exist for the **Finish** button to be enabled.

Document URL

Specifies the target document URL on the QlikView Server where the pages are to be run. This setting is necessary for the Java Objects and QlikX pages to work properly.

Adjust Object Maximum Size to Current Size

Some types of sheet objects, like list boxes, multi boxes and tables, may have a larger maximum size than what is currently utilized. This allows e.g. a table to expand on the sheet when more data is added. This operation resets the maximum object size to the currently used size for all selected objects on the active sheet. You may use CTRL+A to select all sheet objects on the sheet. It is recommended to always perform this operation on all list boxes, multi boxes, table boxes and table charts in the document before using the **Objects Client Page Generator wizard**.

Web page design with QlikX objects

This section describes the HTML code necessary to publish QlikView sheet objects as QlikXs.

Document connector code

The <body> tag of the html page must contain a chunk of code connecting the page to a QlikView document running on the QlikView Server. The code should look like follows:

```
<OBJECT
  id="QVPID1"
  height="1"
  width="1"
  classid="CLSID:74DC8438-E36A-40a0-B750-4E2257FA2E41">
  <PARAM
    NAME="SRC"
    VALUE="qvp://www.server.xyz/MyQVApp.qvw">
  <PARAM
    NAME="AutoScan"
    VALUE="-1">
</OBJECT>
```

The value of the VALUE property in the first parameter must be a complete qvp URL pointing at a QlikView document on a QlikView Server.

The value of the id property on the first line is the connection id of the document. You may have connections to multiple QlikView documents on a single HTML page, but the connection id must be unique for each new document. You may choose connection id at random, but we recommend following the convention laid out in this document (QVPIDn).

This connection code must reside in the HTML page before any references to sheet objects in the document in question (see below).

Incompatibility notice! The class IDs for the document connector and object display in QlikX for QlikView version 8 has been

changed from previous versions. It is thus necessary to change all HTML and/or ASP pages when upgrading to plug-in version 8.0 or later. It is not possible to use the same web page for pre- and post 8.0 clients using QlikX objects. See the Migration section in this document for more information.

Sheet object display code

For each sheet object to be shown another chunk of code must be added. The code should look as follows:

```
<OBJECT
  id="Qlix1"
  height="133"
  width="100"
  classid="CLSID:54D93DE9-7DA5-4477-A3DA-869ADB41CBBB" >
  <PARAM NAME="ObjectID" VALUE="LB38">
  <PARAM NAME="QvPluginID" VALUE="QVPID1">
</OBJECT>
```

The value of the VALUE property in the second parameter is the connection id of the document.

The value of the VALUE property in the first parameter is the id of the sheet object you want to show. The sheet object must reside on the opening sheet of the source document.

The value of the height property is the height in pixels of the rectangle containing the sheet object in the HTML page.

The value of the width property is the height in pixels of the rectangle containing the sheet object in the HTML page.

Automation access to QlikX objects

The <head> tag of the HTML page could contain scripting code referencing a specific QlikView document, identified by its connector id. The code could look as follows:

```
<head>
  <script type="text/javascript">
  function ClearAll()
  {
    QVPID1.ActiveDocument.ClearAll();
  }
  </script>
</head>
```

The QVPID1.ActiveDocument gives you an Automation handle to the connected QlikView document.

18.5 Capabilities, differences and limitations

This section describes some of the technical differences and limitations with QlikX in relation to other QlikView clients.

Sheet Objects supported

All sheet objects apart from Custom Objects are supported

QlikView entities with partial support

The following QlikView entities currently have partial support:

- Sheets (there is no direct connection available to the sheets in the QVW document, but it is very easy to create the same functionality using HTML frames and tabs).

The following QlikView entities currently have no counterpart in the QlikX environment but may appear in some form in future versions:

- Alerts
- Reports

Navigational differences

There are a number of significant differences in the GUI facing users of QlikX compared to users of other types of QlikView clients.

- There is no menu bar
- There is no toolbar

Print/Export

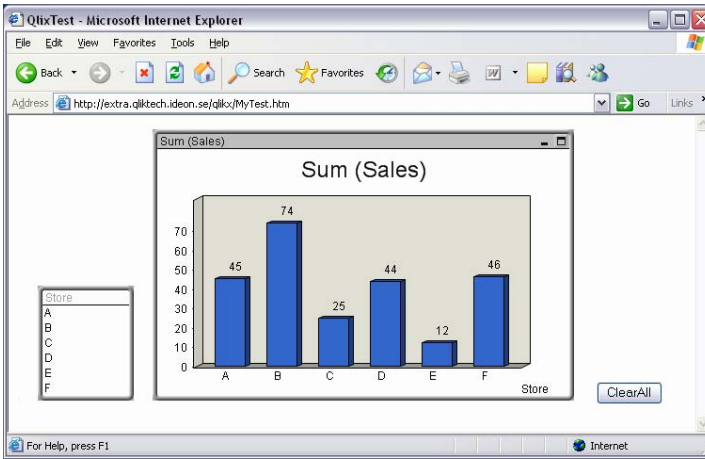
Print and export work exactly as in normal QlikView. Right-click on the object and select Print...

18.6 Complete sample page

Below is a complete HTML page for displaying one list box and one chart with QlikX. Minimum HTML formatting applied.


```
type="button"
value="ClearAll"
onclick="ClearAll()"
id=button2
name=button2>
</body>
</html>
```

The result of the HTML code above can be seen below:



The QlikX example in Microsoft Internet Explorer.

19 QLIKVIEW INSTALLED WINDOWS CLIENTS

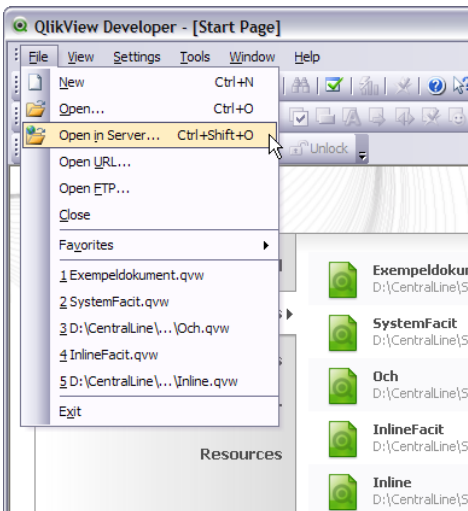
19.1 Locally installed Windows Client

With QlikView Analyzer, Professional or Developer installed on your machine, you may open any application on QlikView Server which you have access rights to.

QlikView Analyzer is a freely downloadable program and can easily be distributed throughout the company via the sample asp pages provided with the installation of QlikView Server.

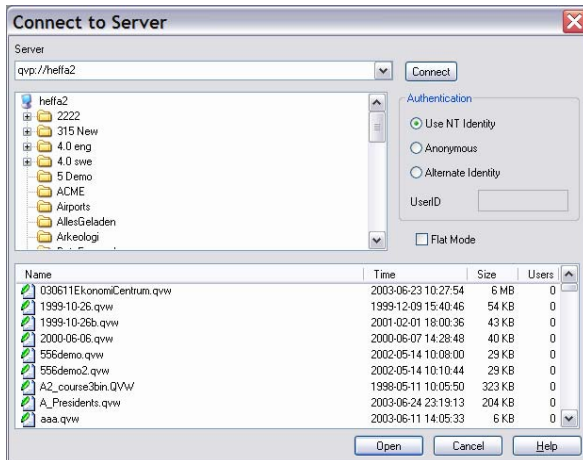
19.2 Open in Server

Once QlikView has been installed, the end user can choose to open QlikView documents through the **Open in Server** command in the **File** menu.



The Open in Server menu item.

This command will provide the end user with all the possible QlikView documents provided and which QlikView Servers are available. By clicking in the **Connect to Server** dialog the application will open and the analysis session can begin.



The Connect to Server dialog in QlikView.

Connection pseudo-URLs

When connecting to QlikView Server from Windows clients, either via the Open in Server dialog or via link files, the identity to be used is specified via the pseudo-URL document address. The syntax is:

qvp://[[username]@]servername [:(port | protocol)] / [documentname.qvw][?paramname=paramvalue{¶mname=paramvalue}]

where

username is a Windows user ID

servername is the name of a server running QlikView Server

documentname is the name of the QlikView document (excluding qvw extension)

port (e.g. 4749) can be used to specify a specific port used by the server

protocol (e.g. http) can be used to specify tunneling protocol

paramname := (USERID | XUSERID | PASSWORD | XPASSWORD | MACRO | IIS_AUTHENTICATE)

USERID denotes a section access userID in clear text. This parameter is also utilized to pass a **Ticket** value during the **Get Ticket** process.

XUSERID denotes a scrambled section access userID

PASSWORD denotes a section access password in clear text

XPASSWORD denotes a scrambled section access password

MACRO denotes the name of a macro to be run when the document is opened (only one macro allowed)

IIS_AUTHENTICATE denotes a single-use key (40 hex characters) for IIS integrated authentication.

paramvalue is a valid value for each parameter.

@ without username denotes anonymous identity.

If user identity is omitted altogether, the logged in Windows identity is assumed.

Examples:

qvp://www.qliktech.com/AcmeStores.qvw

qvp://@www.qliktech.com/AcmeStores.qvw

qvp://john.doe@www.qliktech.com/AcmeStores.qvw

qvp://www.qliktech.com:http/AcmeStores.qvw

qvp://www.qliktech.com/AcmeStores.qvw?USERID=JOHN&PASSWORD=ABC123

qvp://www.qliktech.com/AcmeStores.qvw?MACRO=Mymacro

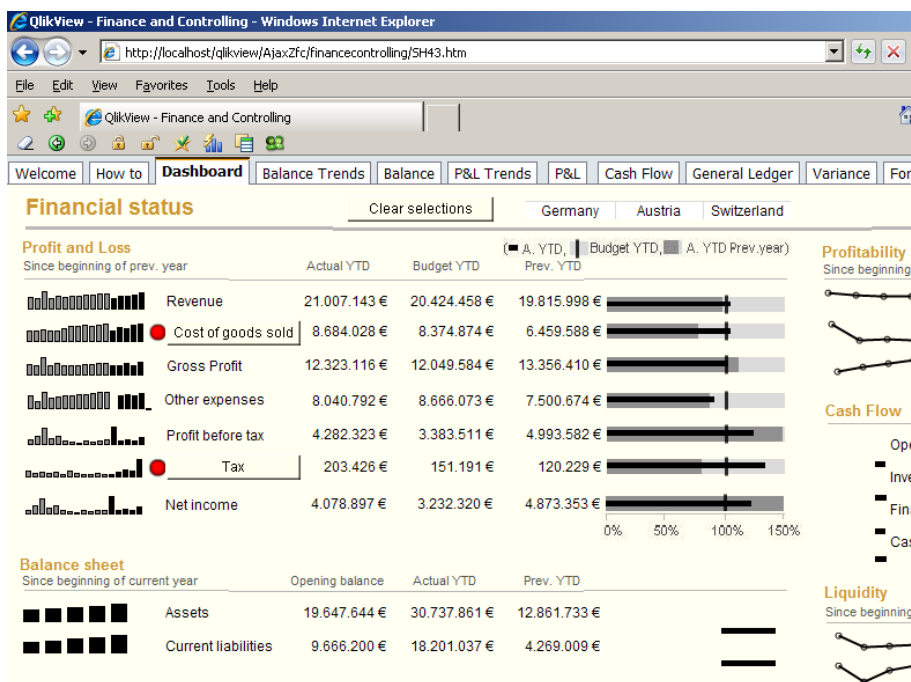
TIP: Internet Explorer 7 does not support @ or : in the URL in order to prevent spooling of URLs. To specify these characters in the URL, you need to URL-encode them.

Use %3A for: and %40 for @.

20 THE QLIKVIEW AJAX ZERO-FOOTPRINT CLIENT (ZFC)

20.1 General

The QlikView AJAX Zero-Footprint Client (ZFC) provides an object based client environment built on a state-of-the-art AJAX (Asynchronous JavaScript And XML) architecture. The QlikView AJAX ZFC requires no installation on the client computer, but some HTML/ASP development is required to display and interact with objects from a QlikView document hosted by QlikView Server. The QlikView AJAX ZFC also opens up the possibility to utilize QlikView in non-IE, and even non-Windows client browser environments.



QlikView AJAX client with the toolbar option

Basic description of the QlikView AJAX ZFC

One of the main advantages of the AJAX architecture is the inherent asynchronous update capability to provide quick, incremental updates to the user

interface, without requiring a browser page refresh. The QlikView AJAX ZFC provides the environment for the QlikView Server to produce and send Dynamic HTML (DHTML) pages and XML data to the browser running on the client computer and also receive feedback from the user clicking in those pages. DHTML is basically HTML with scripting. Nothing is installed on the client computer.

QlikView AJAX ZFC is based on the component AVQ.HTC, which is part of Winsider AB's "Visual Value"™ framework. The "Visual Value"™ framework is a data modeling and presentation framework that allows location independent presentation and manipulation of data with advanced business logic rules and constraints. QlikTech has licensed this component for use with QlikView Server.

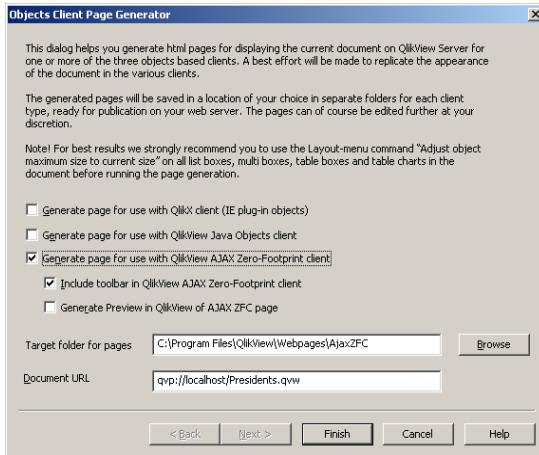
Limitations (short version)

- The client requires one of the following browser types:
 - MS Internet Explorer version 6 or later
 - Browsers based on the Mozilla engine version 1.0.6 or later (e.g. Firefox)
- Different browsers may render the same page slightly differently.
- This is a way to build web pages featuring one or more QlikView sheet objects. The standard QlikView Sheet Tab is not supported, but separate QlikView sheets may be emulated through the use of multiple HTML pages. Selection state in the source QlikView document will hold throughout the same browser session.
- A few basic asp examples and style sheets are provided with a Complete installation of QlikView Server. The default location of these resources is **C:\Program Files\QlikView\Webpages\Ajaxzfc**.
- Most, but not all types of QlikView sheet objects are supported. See the end of this chapter for details.
- The looks of the QlikView sheet objects will in many cases be slightly different from standard QlikView, as e.g. all grid objects (list boxes, tables etc) are generated in HTML.

20.2 QlikView Page Generator for AJAX ZFC

The Objects Client Page Generator tool in QlikView 8+, located off the Tools menu, can be used to generate QlikView AJAX ZFC pages based on existing QlikView documents. This is the recommended way to quickly and easily develop the QlikView AJAX ZFC pages for your documents. The following sections in this chapter can

then be used to modify those pages for those wishing to further customize the presentation.



The Objects Client Page Generator for AJAX Zero-Footprint client.

Target folder for pages

Specifies the folder where the generated pages will be saved. Each page variant (for different client types) will be saved in a separate sub-folder in the specified location. The target folder must exist for the **Finish** button to be enabled.

Document URL

Specifies the target document URL on the QlikView Server where the pages are to be run. This setting is necessary for the Java Objects, QlikX and AJAX pages to work properly.

Include toolbar in QlikView AJAX Zero-Footprint Client

This option is checked by default, and will include a basic navigation toolbar in the generated pages.

You can optionally choose to view a preview of the generated pages in this instance of QlikView. Check the box for **Generate Preview in QlikView of AJAX ZFC page**.

Adjust Object Maximum Size to Current Size

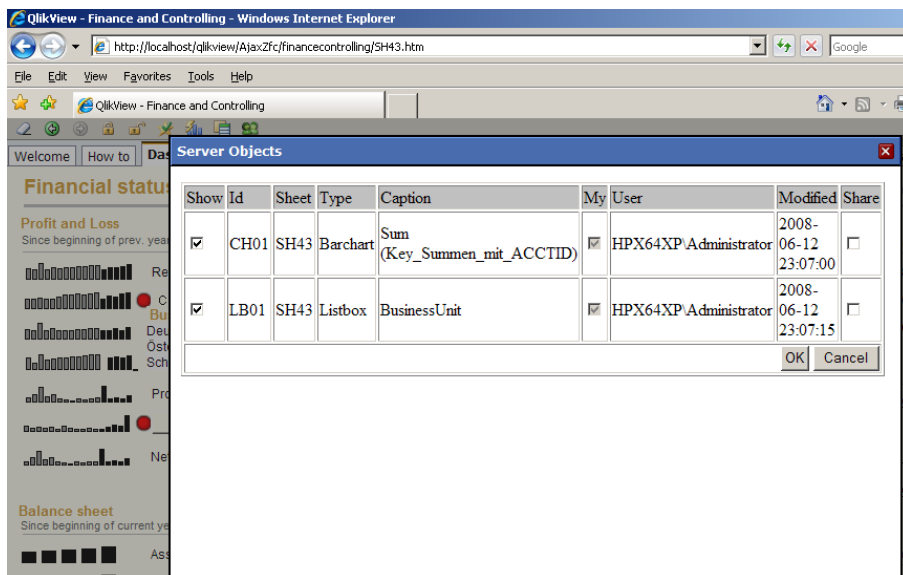
Some types of sheet objects, like list boxes, multi boxes and tables, may have a larger maximum size than what is currently utilized. This allows e.g. a table to expand on the sheet when more data is added. This operation resets the maximum object size to the currently used size for all selected objects on the active sheet. You may use CTRL+A to select all sheet objects on the

sheet. it is recommended to always perform this operation on all list boxes, multi boxes, table boxes and table charts in the document before using the **Objects Client Page Generator wizard**.

20.3 Collaboration - Shared Objects

Collaboration – shared objects – is supported in the AJAX Client for authenticated users. Charts and listboxes may be created, moved, and sized. Use the toolbar provided through the AJAX page generator to allow these options for users. New charts and listboxes may be shared with other users through the **Server Objects** dialog. Locate this option on the toolbar.

Bookmarks may be added, but not deleted. Remember to add a Bookmark sheet object in your source document prior to using the page generator, since there is no ability to implement a bookmark from the toolbar.



Use the **Share** option to share the selected objects with other server users.

20.4 Getting it to work

This section provides an outline of the steps necessary to build a web page with QlikView AJAX ZFC. All files reside or are referenced from the QlikView Server. No client side file installation or maintenance is required.

It is expected that the reader has at least a general knowledge of HTML coding.

Infrastructure

Summary table for file locations

QlikView AJAX ZFC Required File	Default location on QlikView Server
default.htm*	C:\Program Files\QlikView\Webpages\AjaxZfc\basic
QvsComRemote.dll	C:\Program Files\QlikView\Server
QvsViewClient.htm	C:\Program Files\QlikView\Server\QvClients\QvAjaxZfc
Style sheets	
JavaScript	C:\Program Files\QlikView\Server\QvClients\QvAjaxZfc\htc
Support files	

* required only as an example

Server environment

The current version of QlikView AJAX ZFC requires a Microsoft Internet Information Services (MS IIS) web server, version 6 or later, or the QVS built-in http web server.

If the web server does not reside on the same physical machine as the QlikView Server, this must be specified in the QvsViewClient.asp page. Update the code:

```
var host = 'localhost';
```

to:

```
var host = '<name of server to use>'
```

QlikView Server must be version 7 or later.

Required dll

The dll file **QvsComRemote.dll** must reside in the same directory as QlikView Server and be registered. This is normally taken care of by the QlikView Server Installation program.

Web page components

HTML web page

The basic HTML web page(s) utilized by the ZFC client reside in a default location under the standard QlikView virtual directory (path = **C:\Program**

Files\QlikView\Examples\QvsHtmlWeb-

pages\AjaxZfc – for HTML, or **C:\Program**

Files\QlikView\Examples\QvWebpages\AjaxZfc –

for ASP). The pages contain standard HTML code, along with ZFC specific references to QlikView documents and objects. If you perform a Complete Installation of QlikView server, a few examples are provided, including a basic example (**default.htm**).

ASP page

The file **QvsViewClient.asp** must reside in the QvAjaxZfc virtual directory (path = **C:\Program Files\QlikView\Server\QvClients\QvAjaxZfc**).

20.5 Using section access with QlikView AJAX ZFC

This section describes how to use section access with QlikView USERID and PASSWORD together with QlikView AJAX ZFC. This functionality requires the use of QlikView Server 7.52 or later.

URL parameters for section access

The QlikView AJAX ZFC has no provisions for displaying log-in dialogs for QlikView section access USERID and PASSWORD. It is possible, however, to send section access USERID and/or PASSWORD as URL parameters. The syntax is:

http:// pagename [?paramname=paramvalue{¶mname=paramvalue}]

where

pagename is a URL pointing at the HTML page defining the QlikView AJAX ZFC document.

paramname := (USERID | XUSERID | PASSWORD | XPASSWORD)

USERID denotes a section access USERID in clear text

XUSERID denotes a scrambled section access USERID

PASSWORD denotes a section access password in clear text

XPASSWORD denotes a scrambled section access password

paramvalue is a valid value for each parameter.

Example:

http://www.santa.gov/QVpresentplanner.htm?USERID=Santa&PASSWORD=dec25

Note There is a known bug in QlikView Server 7.52 that requires the XUSERID and XPASSWORD parameter names to be specified as lower case (as xuserid and xpassword). This has been corrected in version 8 to be case insensitive.

Scrambling passwords with QvScrambler.dll

In order to convert unscrambled user IDs and passwords into the scrambled versions XUSERID and XPASSWORD, the QvScrambler.dll is provided by the QlikView Server installation. An example using the QvScrambler.dll follows:

```
<html>
<head>
  <title>Extended Security Demo</title>
</head>
<body bgcolor="#eaeaea" >
<%
  QVSName = Request.ServerVariables ("SERVER_NAME")
  userid = Request("userid")
  if userid="" then userid = "ADMIN" end if
  password = ""
  if userid="ADMIN" then password = "ADMIN" end if
  if userid="HAKAN" then password = "WOLGE" end if
  if userid="LARS" then password = "ANDERSSON" end if
  if Request("client_type")<>" then
    redirect_url = ""
    if Request("client_type")="plugin" then redirect_url =
"qvp://" & QVSName & "/StationsAndLines.qvw?IIS_AUTHENTICATE="
    if Request("client_type")="zfp" then redirect_url = "Station-
sAndLines.html?userid="
    if Request("client_type")="java" then redirect_url = "Sta-
tionsAndLines.java.asp?userid="
  set ntsecurity = Server.CreateObject ("QVSNTSecurity.QlikNTLia-
son")
    token = ntsecurity.LogCustomValidatedSession
(Request.ServerVariables ("REMOTE_ADDR"), userid)
  Response.Redirect (redirect_url & token)
end if

set scrambler = Server.CreateObject ("QvScrambler.ScrambleUn-
scramble")
xuserid = scrambler.Scramble (userid)
xpassword = ""
if not password="" then xpassword = scrambler.Scramble (pass-
word) end if
```

```

%>
<form name="input" action="StationsAndLines.asp"
method="get">
  Username:
  <select name="userid" onchange="input.submit()"
  value="<%=userid%>">
    <option value="ADMIN" <%if userid="ADMIN" then
Response.Write("selected") end if%> >ADMIN</option>
    <option value="HAKAN" <%if userid="HAKAN" then
Response.Write("selected") end if%> >HAKAN</option>
    <option value="LARS" <%if userid="LARS" then
Response.Write("selected") end if%> >LARS</option>
    <option value="BAKERLOO" <%if userid="BAKERLOO" then
Response.Write("selected") end if%> >BAKERLOO</option>
    <option value="CENTRAL" <%if userid="CENTRAL" then
Response.Write("selected") end if%> >CENTRAL</option>
    <option value="CIRCLE" <%if userid="CIRCLE" then
Response.Write("selected") end if%> >CIRCLE</option>
    <option value="DISTRICT" <%if userid="DISTRICT" then
Response.Write("selected") end if%> >DISTRICT</option>
    <option value="EAST" <%if userid="EAST" then
Response.Write("selected") end if%> >EAST</option>
    <option value="HAMMERSMITH" <%if userid="HAMMERSMITH" then
Response.Write("selected") end if%> >HAMMERSMITH</option>
    <option value="JUBILEE" <%if userid="JUBILEE" then
Response.Write("selected") end if%> >JUBILEE</option>
    <option value="METROPOLITAN" <%if userid="METROPOLITAN"
then Response.Write("selected") end if%> >METROPOLITAN</option>
    <option value="NORTHERN" <%if userid="NORTHERN" then
Response.Write("selected") end if%> >NORTHERN</option>
    <option value="PICCADILLY" <%if userid="PICCADILLY" then
Response.Write("selected") end if%> >PICCADILLY</option>
    <option value="VICTORIA" <%if userid="VICTORIA" then
Response.Write("selected") end if%> >VICTORIA</option>
    <option value="WATERLOO" <%if userid="WATERLOO" then
Response.Write("selected") end if%> >WATERLOO</option>
  </select>
</form>
<p>
  <a href="qvp://<%=QVSName%>/StationsAndLines.qvw?use-
rid=<%=userid%><%if password<>" then Response.Write("&pass-
word=" & password) end if%>">
    Open plug-in with userid <%=userid%><%if password<>" then
Response.Write(" and password " & password) end if%>
  </a>
</p>
<p>
  <a href="qvp://<%=QVSName%>/StationsAndLines.qvw?xuse-
rid=<%=xuserid%><%if xpassword<>" then Response.Write("&xpass-
word=" & xpassword) end if%>">
    Open plug-in with xuserid <%=xuserid%><%if xpassword<>" then
Response.Write(" and xpassword " & xpassword) end if%>
  </a>
</p>
<p>
  <a href="StationsAndLines.asp?userid=<%=use-

```

```

rid%>&client_type=plugin">
    Open plug-in using "QVSNTSecurity.QlikNTLiason" with userid
<%=userid%>
</a>
</p>
<p>
    <a href="StationsAndLines.java.asp?userid=<%=userid%><%if
password<>"" then Response.Write("&password=" & password) end
if%>">
        Open QlikWeb with userid <%=userid%><%if password<>"" then
Response.Write(" and password " & password) end if%>
    </a>
</p>
<p>
    <a href="StationsAndLines.java.asp?xuserid=<%=xuserid%><%if
xpassword<>"" then Response.Write("&xpassword=" & xpassword)
end if%>">
        Open QlikWeb with xuserid <%=xuserid%><%if xpassword<>""
then Response.Write(" and xpassword " & xpassword) end if%>
    </a>
</p>
<p>
    <a href="StationsAndLines.asp?userid=<%=use-
rid%>&client_type=java">
        Open QlikWeb using "QVSNTSecurity.QlikNTLiason" with userid
<%=userid%>
    </a>
</p>
<p>
    <a href="StationsAndLines.html?userid=<%=userid%><%if pass-
word<>"" then Response.Write("&password=" & password) end
if%>">
        Open ZFP with userid <%=userid%><%if password<>"" then
Response.Write(" and password " & password) end if%>
    </a>
</p>
<p>
    <a href="StationsAndLines.html?xuserid=<%=xuserid%><%if
xpassword<>"" then Response.Write("&xpassword=" & xpassword)
end if%>">
        Open ZFP with xuserid <%=xuserid%><%if xpassword<>"" then
Response.Write(" and xpassword " & xpassword) end if%>
    </a>
</p>
<p>
    <a href="StationsAndLines.asp?userid=<%=use-
rid%>&token_type=userid_token&client_type=zfp">
        Open ZFP using "QVSNTSecurity.QlikNTLiason" with userid
<%=userid%>
    </a>
</p>
</body>
</html>

```

20.6 Capabilities, differences and limitations

This section describes some of the technical differences and limitations with QlikView AJAX ZFC in relation to other QlikView Server clients.

Sheet Objects supported

The following types of sheet objects are currently supported to a smaller or larger extent in the QlikView AJAX ZFC:

- List box
- Multi box
- Table Box
- Straight Table
- Pivot Table
- Bitmap charts
- Bookmark Object
- Current Selections Box
- Input Box
- Button
- Text Object
- Line/arrow Object
- Statistics box
- Slider/Calendar object

The following types of sheet objects are not supported in the QlikView AJAX ZFC and are very unlikely to ever be supported:

- Custom Object

QlikView entities with partial support

The following QlikView entities currently have partial support in the QlikView AJAX ZFC environment:

- Sheets (there is no direct connection available to the sheets in the QVW document, but it is very easy to create the same functionality using HTML frames and tabs). User selection state is maintained throughout a single browser session.

Other QlikView entities not supported

The following QlikView entities currently have no counterpart in the QlikView AJAX ZFC environment but may appear in some form in future versions:

- Alerts
- Reports

Navigational differences

There are a number of significant differences in the GUI facing users of QlikView AJAX ZFC compared to users of other types of QlikView clients.

- There are no menus
- There are no right-click menus or functionality
- There is no tool-bar

Print/Export

The following general comments can be made with regard to print and export of objects and data when using the QlikView AJAX ZFC

Functionality supported includes:

- The HTML page can be printed by using the Print command in MS Internet Explorer. The usual rules and limitations regarding MS IE printing apply.
- Chart images can be copied to clipboard or saved as any other pieces of graphics on a web page.
- Button-driven server-side export providing results in a new MS Internet Explorer window. Same functionality as can be found in the existing QlikView Java clients.

Right-Click functionality is not supported, but caption icons can be utilized in object captions

APIs and integration

QlikView AJAX ZFC builds on HTML. This implies certain limitations with regard to programmatic access and integration

- Automation APIs cannot be used
- No real client side APIs are available. It may be possible in the future to access data stored client-side in QlikView AJAX ZFC elements via scripting.
- Note however that

-
- Server side macros can be executed via buttons
 - The HTML elements of QlikView AJAX ZFC can co-exist with all other types of web page components on a common HTML page. That includes other QlikView clients (e.g. QVA for IE plug-in, including QlikX components) and all types of ActiveX controls. Interaction between QlikView AJAX ZFC and the other components will however be limited.

20.7 ASP timeouts for very large QlikView documents

When using the QlikView AJAX ZFC with large QlikView documents, the asp code might sometimes require that you increase the asp timeout. This can be made in two ways, either programmatically or by customizing the IIS.

- By setting the `Server.ScriptTimeout` property in your code, such as:
`<%Server.ScriptTimeout = 180 %>`, where the numeric value is the number of seconds that the current script will be executed.
- To set the timeout in the IIS, open the **IIS Management Console**, open **Properties** for the folder containing the asp code, go to the **Directory** or **Virtual Directory** page (depends on what type of folder you use), press the **Configuration** button to open the **Application Configuration** dialog, go to the **Options** page where you find the edit box for the **ASP Script Timeout**.

21 CONSIDERATIONS WHEN DEVELOPING DOCUMENTS FOR USE WITH QLIKVIEW SERVER

21.1 General

The QlikView Server will read all qvw files created by QlikView 5.0 and later. Files with older file formats must be converted to a newer before used with the QlikView Server.

21.2 General limitations when working from clients

Regardless which client you use there are some things you cannot do when working with a document opened through QlikView server:

- You cannot create new documents
- You cannot access or execute the script
- You cannot add sheets.
- You cannot access sheet or sheet object properties of non-shared objects
- You cannot access or change the macro module
- You cannot save changes to the document, beyond shared objects

21.3 Limitations in Java clients layout functionality

Some features in QlikView 8 are not supported in the Java Clients. Apart the limitations mentioned above there are a number of layout features currently not supported in the Java clients:

- Chart animations not supported
- Custom objects are not supported
- Pictures in list box cells are not supported
- Dynamic (calculated) text on sheet tabs not available

-
- Dynamic (calculated) color on sheet tabs not available
 - It is not possible to Copy or Paste sheet objects
 - It is not possible to access the QlikView sheet objects via ODBC
 - No Scroll control for predefined values in input box
 - Multi line text wrapping cannot be used in multi boxes
 - Launch/Export buttons have no function
 - Info returning other items than text and URL's are not supported
 - Use of VBScript is restricted. (See 9.5).

21.4 Font Mapping in Java clients

Font mapping in the Java Clients has improved dramatically over previous version, but there may still exist minor differences in the way sheet objects are pre-sented on the screen in standard QlikView and in the Java client. This is mainly due to the fact that the rendering of the screen objects are not handled by Windows in the Java environment. Windows and Java may handle font sizes slightly differently in terms of actual size on screen and when the size has to be rounded.

TIP: It is strongly recommended that you always test any application in the QlikView Server together with the browser(s) that you intend to use, before commissioning.

21.5 Performance Considerations

Response time and memory usage both on the server and on the client can depend heavily on how your application is designed. Below are a few tips to bear in mind when building your application:

- 1 Each open chart and table will require extra memory in the server for each user connected. Minimized charts/tables as well as charts/tables not on the active sheet do not require extra memory. Each open chart/table must also be recalculated on the server after each change in the logical state, which could prolong response times. Therefore avoid opening charts and tables when it is not necessary, or try to spread them over several sheets.
- 2 Large bitmap pictures in sheet backgrounds or text objects may cause delays, as they have to be transferred from the server to the client. The bitmaps are compressed during transfer and are only transferred once per session, but nevertheless can cause substantial amount of data to be moved. Consider using colored sheet back-

ground and text objects with opaque background for better performance. If you have to use bitmap sheet backgrounds, it is more efficient to apply one background on application level than assigning the same background to each sheet.

- 3 Very small (e.g. 1x1 pixel) tiled background bitmaps can greatly reduce performance on the client, as they have to be rendered repeatedly by the Java Virtual Machine. Rather use medium size (e.g. 20x20 pixel) tiles or colored sheet background.

21.6 Considerations when using macros in QV documents on the QlikView Server

Macros on QlikView Server

QlikView Server can execute macros in QlikView documents. Due to client-server environment some considerations must be taken when using macros.

Macro trigger limitations

Macro triggers may or may not work with the various client types. The table below summarizes the situation.



Trigger will work as usual.



Trigger is not to be used as they lack meaning or may cause unexpected results.



Trigger lacks meaning with this type of client.

Trigger	Windows clients	Java client	Java Objects client
Document.OnAnySelect			
Document.OnClose			
Document.OnOpen			
Document.OnPostReduceData			
Document.OnPostReload			

Trigger	Windows clients	Java client	Java Objects client
Document.OnToolbarMacro1			
Document.OnToolbarMacro2			
Document.OnToolbarMacro3			
Field.OnSelect			
Field.OnChange			
Field.OnLock			
Field.OnUnlock			
Sheet.OnActivateSheet			
Sheet.OnLeaveSheet			
SheetObject.OnActivate			
SheetObject.OnDeActivate			
Variable.OnChange			
Variable.OnInput			

Macro functionality limitations

The type of functionalities that will normally work well in macros in the QlikView Server environment are summarized in the table below:



Will work as usual.



Will not work.



May work.

Type of macro operation	Windows clients	Java client	Java Objects client
Logical operations such as Clear Selections or selections of field values.			
Operations changing variable values			
Layout operations acting on the properties of sheets and sheet objects, including minimization/restoration of sheet objects.			
Operations changing document or user settings			
Creation of sheets or sheet objects			
All operations related to the script, including Reload			
Data reduction operations, e.g. ReduceData			
Operations such as Save and Open document			

VBScript functions

VBScript functions defined in the module of a QlikView document will generally work well on QlikView Server. General functionality limitations as defined in the previous section apply.

Server-side export

It is possible to export the content of any chart to a text file on the server through the use of macro automation. This feature has to be added in the QlikView document and then the Macro can be executed from the QlikView Server directly in the web browser.

Below you will find a sample of VB Script code used in the Edit Module in QlikView for making the export possible. The macro is then connected to a trigger, for example, a button.

```
Sub ServerExport
  Set st = ActiveDocument.GetSheetObject("CH20")
  st.ServerSideExport "C:\Program
Files\QlikView\Server\CH20export.skv" , ";" , 0
End Sub
```

As the button is clicked, the contents of the Straight Table CH20 can be exported as a text file to the specified directory on the QlikView Server machine.

21.7 Security and Access restriction

It is important that information is distributed only to those with access rights to it. Since QlikView makes the previously cumbersome process of retrieving information a very simple task, it is obvious that an access restriction mechanism is necessary. This can be done when creating the application. For more information on this, see the QlikView reference manual.

When a QlikView document is published on a QlikView Server, it is just as important to protect confidential data and prevent unauthorized access as it is when running the standard QlikView. QlikView Server therefore supports all security features built into QlikView. In addition, it is possible to use the Windows NT file security or DMS authorization to control access to specific documents. This can be done also without using any of the QlikView built-in security features.

QlikView File Security

QlikView and QlikView server can be instructed to test for a number of user properties before letting the user open a document. (For more information on this, see the QlikView reference manual.) Possible fields are:

USERID: A field that should contain a valid user ID. The client will open a dialog asking for user ID.

PASSWORD: A field that should contain a valid password. The client will open a dialog asking for Password.

SERIAL: A field that should contain a number corresponding to the QlikView serial number. The server will check the Server Serial Number and compare that to the valid ones listed in the QlikView document. In other words, the Serial Number check is a very blunt security instrument for documents that are to be published: Any client will be allowed to open a document on the valid server.

NTNAME: A field that should contain a string corresponding to a Windows NT Domain user name or group name. The server will ask the operating sys-

tem if the user named in the QlikView document corresponds to the user logged on, or to a group to which the logged on user belongs.

NTDOMAINSID: A field that should contain a string corresponding to a Windows NT Domain SID. The server will ask the operating system if the Windows NT Domain named in the QlikView document corresponds to the Domain to which the logged on user belongs.

NTSID: A field that should contain a Windows NT SID. The server will ask the operating system if the NT SID named in the QlikView document corresponds to the NT SID of the user logged on, or to a group to which the logged on user belongs.

Only if all the required tests are passed, will the user be able to open the document.

The Windows File security described above and the three last tests are of course very similar. The difference is that for the former, the security information is fetched from the file system, whereas for the latter, it is fetched from within the QlikView file. Both are, however, compared to the logon information from the operating system.

Dynamic Data reduction

QlikView Web Server supports a feature that can hide some part of the data (field values) in a document from the user based on the login information. Only the information that the user has the right to see will be shown.

The selection of values to be shown is controlled by means of having one or more fields with common names in the data and in the access restriction data. Only the field values logically connected to the user logged on will be shown.

This is a very efficient way to hide information to which the user does not have access.

For more information on how to build such an application, see the QlikView reference manual.

PART III: JAVA CLIENT APIs

- **Java applet API**
- **HTML page script API**

22 JAVA APPLET API

22.1 General provisions for the API

General

The Java applet API exposes a large sub-set of the functionality of the QlikView Automation API for access from other Java applets on a web page. Compared to the script API (see next chapter below) this API offers considerably richer functionality and better performance.

Disclaimer

The Java applet API was introduced with version 7.51 of QlikView Server. Not all parts have been tested in detail. It is likely that some classes will be added to and deleted from the API in future releases.

22.2 Documentation and examples

- An HTML documentation of all available classes and methods can be found in the folder **C:\Program files\QlikView\Documentation\Java API** after installing QlikView Server 7.51 or later.
- A small Java code example accessing the API can be found in the same folder.

23 HTML PAGE SCRIPT API

23.1 General Specifications for the script API

General

This part of the documentation provides a systematic description of the API (Application Programming Interface) of the full document Java applet (QlikView Java Client) and of the object Java applet (QlikView Java Objects Client).

Programming languages

The examples in this documentation are written in ASP, HTML and JavaScript. This documentation is, however, not a manual for these scripting languages. The reader is expected to possess a basic knowledge of ASP, HTML and JavaScript programming.

Style coding

In this documentation all menu commands and dialog options are shown in **Arial bold**. All file names and paths are shown in **Courier Bold**. Sample code is shown in **Courier Bold**.

23.2 General provisions

- All functions block until the operation has been performed.
- All parameters must be strings in order to guarantee proper results in all browsers. This means that e.g. a column index 3 should be stated as "3".

23.3 Error handling

On any error the Int functions return -1 (minus one) and the string functions return the zero length String "" (which is often translated into null by JavaScript).

23.4 Provisions for examples

All examples under the function descriptions below assume the following:

- The current document contains a QlikView Java Objects master applet named QlikDocument.

23.5 QlikView Java / QlikView Java Objects script API function summary

Member	Comment
clearAll	Clears all the selections for this application.
clickButton	Activates a button.
getCaption	Returns the caption of any specified object.
getCellValue	Returns the contents of a specified cell in a table.
getColCount	Returns the number of columns for table boxes and straight tables and the number of data columns for pivot tables.
getColLabel	Returns the label for a specified column in a table box or straight table.
getLabel	Returns the label for a specified column in a table box or straight table and for a specified column and row in a pivot table.
getLabelColCount	Returns the number of label columns to the left of the data columns in a pivot table.
getLabelRowCount	Returns the number of label rows above the data rows in a pivot table.
getListState	Returns the state of a specific row in a list box or multi box field.
getListValue	Returns the value from a specific row in a list box or multi box field.
getOptionalCount	Returns the number of optional values in a list box or multi box field.
getRowCount	Returns the number of data rows for table boxes, straight tables and pivot tables.
getSelectedCount	Returns the number of selected values in a list box or multi box field.
getStateCount	Returns the number of values that match the given state in a list box or multi box field.
getStateValues	Returns the values that match the given state in a list box or multi box field.
getTextValue	Returns the current text from a text object.
Redo	Reapplies a previously undone selection action.
runMacro	Executes a macro on the QlikView document at the server.
searchSelect	Makes a selection in a list box or multi box field based on a search string.
searchToggle	Toggles a selection in a list box or multi box field based on a search string.

Member	Comment
Undo	Undoes the last selection action.

23.6 Function descriptions

clearAll

Clears all the selections for this application

ParameterNo	Parameter	Type	Direction	Comment
return value	-	-	-	No return value
1	locked	String	in	If non-zero, all locked selections will be cleared too

Example of use :

```
<FORM id=DocFrm name=DocFrm>
  <INPUT type=button id=ClearAllBtn name=ClearAllBtn
    value=ClearAll onclick="doClearAll()">
</FORM>
<SCRIPT LANGUAGE=JavaScript>
function doClearAll()
{
  document.QlikDocument.clearAll("0");
}
</SCRIPT>
```

clickButton

Activates a button.

ParameterNo	Parameter	Type	Direction	Comment
return value	-	string	-	URL to load if the button's activation should result in the loading of a new URL
1	buttonId	string	in	QlikView Object ID of button

Example of use :

```
<FORM id=ButtonFrm name=ButtonFrm>
  <INPUT type=button id=ButtonBtn name=ButtonBtn value=But-
    ton onclick="doButton ()">
  <INPUT type=text size=20 maxlength=20 id=ButtonCell
    name=ButtonCell readonly>
</FORM>
```

```

<SCRIPT LANGUAGE=JavaScript>
function doButton()
{
    var frm = document.ButtonFrm;
    frm.ButtonCell.value = document.QlikDocument.clickBut-
        ton("BU01");
}
</SCRIPT>

```

getCaption

Returns the caption of any specified object.

ParameterNo	Parameter	Type	Direction	Comment
return value	-	string	-	Value of caption returned
1	objID	string	in	QlikView Object ID

Example of use :

```

<FORM id=frm name=frm>
<input type=button value="get caption" onClick="getCap()">
    <input type=text name=result>
</FORM>
<SCRIPT language=javascript>
function getCap()
{
    frm.result.value = document.QlikDocument.getCap-
        tion("LB01");
}
</SCRIPT>

```

getCellValue

Returns the contents of a specified data cell in a table.

ParameterNo	Parameter	Type	Direction	Comment
return value	-	string	-	Text contents of data cell returned
1	gridID	string	in	QlikView Object ID of table box, straight table or pivot table
2	col	string	in	A number indicating column index (0 based)
3	row	string	in	A number indicating row index (0 based)

Example of use :

```

<FORM id=TableFrm name=TableFrm>
  Column: <INPUT type=text size=5 maxlength=5 id=TableCell-
  Col name=TableCellCol value=1>
  Row: <INPUT type=text size=5 maxlength=5 id=TableCellRow
  name=TableCellRow value=1>
  <INPUT type=button id=TableCellBtn name=TableCellBtn
  value=Cell onclick="doTableCell()">
  <INPUT type=text size=20 maxlength=20 id=TableCell
  name=TableCell readonly>
</FORM>
<SCRIPT LANGUAGE=JavaScript>
function doTableCell()
{
  var frm = document.TableFrm;
  frm.TableCell.value = document.QlikDocument.getCell-
  Value("TB01",
  frm.TableCellCol.value, frm.TableCellRow.value);
}
</SCRIPT>

```

getColCount

Returns the number of columns for table boxes and straight tables and the number of data columns for pivot tables.

ParameterNo	Parameter	Type	Direction	Comment
return value	-	integer	-	Number of data columns returned
1	gridID	string	in	QlikView Object ID of table box, straight table or pivot table

Example of use :

```

<FORM id=TableFrm name=TableFrm>
  <INPUT type=button id=TableRowBtn name=TableColBtn
  value=Cols onclick="doTableCols()">
  <INPUT type=text size=5 maxlength=5 id=TableCols
  name=TableCols readonly>
</FORM>
<SCRIPT LANGUAGE=JavaScript>
function doTableCols()
{
  document.TableFrm.TableCols.value = document.QlikDocu-
  ment.getColCount("TB01");
}
</SCRIPT>

```

getColLabel

Returns the label for a specified column in a table box or straight table.

ParameterNo	Parameter	Type	Direction	Comment
return value	-	string	-	Column label returned

ParameterNo	Parameter	Type	Direction	Comment
1	tableID	string	in	QlikView Object ID of table box, straight table or pivot table
2	col	string	in	A number indicating column index (0 based)

Example of use :

```

<FORM id=StraightFrm name=StraightFrm>
  Column: <INPUT type=text size=5 maxlength=5 id=Straight-
    LabelCol name=StraightLabelCol value=1>
  <INPUT type=button id=StraightLabelBtn name=StraightLa-
    belBtn value=Label onclick="doStraightLabel()">
  <INPUT type=text size=20 maxlength=20 id=StraightLabel
    name=StraightLabel readonly>
</FORM>
<SCRIPT LANGUAGE=JavaScript>
function doStraightLabel()
{
  var frm = document.StraightFrm;
  frm.StraightLabel.value =
  document.QlikDocument.getColLabel("CH01", frm.Straight-
    LabelCol.value);
}
</SCRIPT>

```

getLabel

Returns the label for a specified column in a table box or straight table and for a specified column and row in a pivot table.

ParameterNo	Parameter	Type	Direction	Comment
return value	-	string	-	Column label returned
1	tableID	string	in	QlikView Object ID of table box, straight table or pivot table
2	col	string	in	A number indicating column index (0 based)
3	row	string	in	A number indicating row index (0 based)

Example of use :

```

<FORM id=PivotFrm name=PivotFrm>

```

```

Column: <INPUT type=text size=5 maxLength=5 id=PivotLabelCol
        name=PivotLabelCol value=0>
Row: <INPUT type=text size=5 maxLength=5 id=PivotLabelRow
      name= PivotLabelRow value=0>
<INPUT type=button id=PivotLabelBtn name=PivotLabelBtn
      value=Label onclick="doPivotLabel()">
<INPUT type=text size=20 maxLength=20 id=PivotLabel
      name=PivotLabel readonly>
</FORM>
<SCRIPT LANGUAGE=JavaScript>
function doPivotLabel() {
    var frm = document.PivotFrm;
    frm.PivotLabel.value =
    document.QlikDocument.getLabel("CH01", frm.PivotLabel-
    Col.value, frm.PivotLabelRow.value);
}
</SCRIPT>

```

getLabelColCount

Returns the number of label columns to the left of the data columns in a pivot table. If used on a straight table or table box always returns 0.

ParameterNo	Parameter	Type	Direction	Comment
return value	-	integer	-	Number of label columns returned
1	gridID	string		QlikView Object ID of table box, straight table or pivot table

Example of use :

```

<FORM id=PivotFrm name=PivotFrm>
  <INPUT type=button id=PivotLabelColBtn name=PivotLabel-
    ColBtn
    value=Label onclick="doPivotLabelColCount()">
  <INPUT type=text size=20 maxLength=20 id=LabelColCount
    name=LabelColCount readonly>
</FORM>
<SCRIPT LANGUAGE=JavaScript>
function doPivotLabelColCount()
{
    var frm = document.PivotFrm;
    frm.LabelColCount.value =
    document.QlikDocument.getLabelColCount("CH01");
}
</SCRIPT>

```

getLabelRowCount

Returns the number of label rows above the data rows in a pivot table. If used on a straight table or table box always returns 1.

ParameterNo	Parameter	Type	Direction	Comment
return value	-	integer	-	Number of data rows returned
1	gridID	string		QlikView Object ID of table box, straight table or pivot table

Example of use :

```
<FORM id=PivotFrm name=PivotFrm>
  <INPUT type=button id=PivotLabelRowBtn name=PivotLabel-
    RowBtn
    value=Label onclick="doPivotLabelRowCount()">
  <INPUT type=text size=20 maxlength=20 id=LabelRowCount
    name=LabelRowCount readonly>
</FORM>
<SCRIPT LANGUAGE=JavaScript>
function doPivotLabelRowCount()
{
  var frm = document.PivotFrm;
  frm.LabelRowCount.value =
  document.QlikDocument.getLabelRowCount("CH01");
}
</SCRIPT>
```

getListState

Returns the state of a specific row in a list box or multi box field.

ParameterNo	Parameter	Type	Direction	Comment
return value	-	int	-	Field state returned
1	listID	string	in	Either the QlikView Object ID of a list box (e.g. "LB01") or the QlikView Object ID of a multi box followed by a comma and a numeric index (zero-based) indicating a specific field row in the multi box (e.g. "MB01,1")
2	index	string	in	Number indicating the index (0 based) of the requested field value in the field shown in the list box

Example of use :

```
<FORM id=ListFrm name=ListFrm>
```

```

State: <INPUT type=text size=5 maxlength=1 id=GetNState
      name=GetNState value=1>
<INPUT type=button id=GetStateBtn name=GetStateBtn
value=Get onclick="doGetNthState()">
<INPUT type=text size=20 maxlength=20 id=GetNthState
name=GetNthState readonly>
</FORM>
<SCRIPT LANGUAGE=JavaScript>
function doGetNthState()
{
  var frm = document.ListFrm;
  frm.GetNthState.value = document.QlikDocument.getList-
    State("LB05",
    frm.GetNState.value);
}
</SCRIPT>

```

getListValue

Returns the field value from a specific row in a list box or multi box field.

ParameterNo	Parameter	Type	Direction	Comment
return value	-	string	-	Field value returned
1	listID	string	in	Either the QlikView Object ID of a list box (e.g. "LB01") or the QlikView Object ID of a multi box followed by a comma and a numeric index (zero-based) indicating a specific field row in the multi box (e.g. "MB01,1")
2	index	string	in	Number indicating the index (0 based) of the requested field value in the field shown in the list box

Example of use :

```

<FORM id=ListFrm name=ListFrm>
  Row: <INPUT type=text size=5 maxlength=5 id=GetN
      name=GetN value=1>

```

```

        <INPUT type=button id=GetBtn name=GetBtn value=Get
            onclick="doGetNth()">
        <INPUT type=text size=20 maxlength=20 id=GetNth name=Get-
            Nth readonly>
</FORM>
<SCRIPT LANGUAGE=JavaScript>
function doGetNth()
{
    var frm = document.ListFrm;
    frm.GetNth.value = document.QlikDocument.getList-
        Value("LB05", frm.GetN.value);
}
</SCRIPT>

```

getOptionalCount

Returns the number of optional values in a list box or multi box field.

ParameterNo	Parameter	Type	Direction	Comment
return value	-	integer	-	Number of optional field values returned
1	listID	string	in	Either the QlikView Object ID of a list box (e.g. "LB01") or the QlikView Object ID of a multi box followed by a comma and a numeric index (zero-based) indicating a specific field row in the multi box (e.g. "MB01,1")

Example of use :

```

<FORM id=ListFrm name=ListFrm>
    <INPUT type=button id=OptionalBtn name=OptionalBtn
        value=Optional onclick="doGetOptional()">
    <INPUT type=text size=5 maxlength=5 id=GetCount name=Get-
        Count readonly>
</FORM>
<SCRIPT LANGUAGE=JavaScript>
function doGetOptional()
{
    document.ListFrm.GetCount.value = document.QlikDocu-
        ment.getOptionalCount("LB05");
}
</SCRIPT>

```

getRowCount

Returns the number of field values for a field in a list box or multi box and the number of data rows for a table box, straight table and pivot tables.

ParameterNo	Parameter	Type	Direction	Comment
return value	-	integer	-	Number of data rows in table returned
1	objectID	string	in	Either the QlikView Object ID of a list box , table box, straight table or pivot table (e.g. "LB01") or the QlikView Object ID of a multi box followed by a comma and a numeric index (zero-based) indicating a specific field row in the multi box (e.g. "MB01,1")

Examples of use :

```

<FORM id=ListFrm name=ListFrm>
  <INPUT type=button id=CardinalBtn name=CardinalBtn
    value=Cardinal onclick="doGetCardinal()">
  <INPUT type=text size=5 maxlength=5 id=GetCount name=Get-
    Count readonly>
</FORM>
<SCRIPT LANGUAGE=JavaScript>
function doGetCardinal()
{
  document.ListFrm.GetCount.value = document.QlikDocu-
    ment.getRowCount("LB05");
}
</SCRIPT>
<FORM id=StraightFrm name=StraightFrm>
  <INPUT type=button id=StraightRowsBtn name=StraightRows-
    Btn value=Label onclick="doStraightRows()">
  <INPUT type=text size=20 maxlength=20 id=StraightRows
    name=StraightRows readonly>
</FORM>
<SCRIPT LANGUAGE=JavaScript>
function doStraightRows()
{
  document.StraightFrm.StraightRows.value = document.Qlik-
    Document.getRowCount("CH01");
}
</SCRIPT>

```

getSelectedCount

Returns the number of selected values in a list box or multi box field.

ParameterNo	Parameter	Type	Direction	Comment
return value	-	integer	-	Number of selected field values returned
1	ListID	string	in	Either the QlikView Object ID of a list box (e.g. "LB01") or the QlikView Object ID of a multi box followed by a comma and a numeric index (zero-based) indicating a specific field row in the multi box (e.g. "MB01,1")

Example of use :

```
<FORM id=ListFrm name=ListFrm>
  <INPUT type=button id=SelectedBtn name=SelectedBtn
    value=Selected onclick="doGetSelected()">
  <INPUT type=text size=5 maxlength=5 id=GetCount name=Get-
    Count readonly>
</FORM>
<SCRIPT LANGUAGE=JavaScript>
function doGetSelected()
{
  document.ListFrm.GetCount.value = document.QlikDocu-
    ment.getSelectedCount("LB05");
}
</SCRIPT>
```

getStateCount

Returns the number of values with the given state in a list box or multi box field.

ParameterNo	Parameter	Type	Direction	Comment
return value	-	integer	-	Number of field values with this state returned
1	ListID	string	in	Either the QlikView Object ID of a list box (e.g. "LB01") or the QlikView Object ID of a multi box followed by a comma and a numeric index (zero-based) indicating a specific field row in the multi box (e.g. "MB01,1")
2	state	String	in	A number representing the state to be counted

Example of use :

```

<FORM id=ListFrm name=ListFrm>
  State: <INPUT type=text size=5 maxlength=1 id=GetCount-
    State
      name=GetCountState value=1>
  <INPUT type=button id=StateCountBtn name=StateCountBtn
    value=Count onclick="doGetStateCount()">
  <INPUT type=text size=5 maxlength=5 id=GetStateCount
    name=GetStateCount readonly>
</FORM>
<SCRIPT LANGUAGE=JavaScript>
function doGetStateCount()
{
  var frm = document.ListFrm;
  frm.GetStateCount.value = document.QlikDocument.get-
    StateCount("LB05",
      frm.GetCountState.value);
}
</SCRIPT>

```

getStateValues

Returns the number of values with the given state in a list box or multi box field.

ParameterNo	Parameter	Type	Direction	Comment
return value	-	String	-	String containing all the values that have this state value. They are separated by a pipe (“ ”) character.
1	ListID	string	In	Either the QlikView Object ID of a list box (e.g. “LB01”) or the QlikView Object ID of a multi box followed by a comma and a numeric index (zero-based) indicating a specific field row in the multi box (e.g. “MB01,1”)
2	state	String	In	A number representing the state to be counted

Example of use :

```

<FORM id=ListFrm name=ListFrm>

```

```

State: <INPUT type=text size=5 maxlength=1 id=GetVal-
uesState
      name=GetValuesState value=1>
<INPUT type=button id=StateValuesBtn name=StateValuesBtn
      value=Values onclick="doGetStateValues()">
<INPUT type=text size=5 maxlength=5 id=Values name=Values
      readonly>
</FORM>
<SCRIPT LANGUAGE=JavaScript>
function doGetStateValues()
{
  var frm = document.ListFrm;
  frm.Values.value = document.QlikDocument.getStateVal-
ues("LB05",
  frm.GetValuesState.value);
}
</SCRIPT>

```

getTextValue

Returns the current text from a text object.

ParameterNo	Parameter	Type	Direction	Comment
return value	-	String	-	Text in text object returned
1	textID	String	in	The QlikView Object ID of a text object
(e.g. "TX01")				

Example of use :

```

<FORM id=ListFrm name=ListFrm>
  <INPUT type=text size=10 maxlength=10 id=MyTxt name=MyTxt
  value="">
</FORM>
<SCRIPT LANGUAGE=JavaScript>
function getText()
{
  document.ListFrm.MyTxt.value = document.QlikDocu-
ment.getTextValue("TX02");
}
</SCRIPT>

```

redo

Reapplies a previously undone selection action.

ParameterNo	Parameter	Type	Direction	Comment
return value	-	-	-	No return value

Example of use :

```

<FORM id=ListFrm name=ListFrm>

```

```

        <INPUT type=button id=RedoBtn name=RedoBtn value=Redo
            onclick="doRedo()">
</FORM>
<SCRIPT LANGUAGE=JavaScript>
function doRedo()
{
    document.QlikDocument.redo();
}
</SCRIPT>

```

runMacro

Executes a macro defined in the QlikView document macro module.

ParameterNo	Parameter	Type	Direction	Comment
return value	-	-	-	No return value
1	MacroName	string	in	QlikView macro name

Example of use :

```

<FORM id=ListFrm name=ListFrm>
    <INPUT type=button id=MacroBtn name=MacroBtn
        value=MyMacro onclick="doMacro()">
</FORM>
<SCRIPT LANGUAGE=JavaScript>
function doMacro()
{
    document.QlikDocument.runMacro("MyMacro");
}
</SCRIPT>

```

searchSelect

Makes a selection in a list box or multi box field based on a search string.

ParameterNo	Parameter	Type	Direction	Comment
return value	-	-	-	No return value
1	listID	String	in	Either the QlikView Object ID of a list box (e.g. "LB01") or the QlikView Object ID of a multi box followed by a comma and a numeric index (zero-based) indicating a specific field row in the multi box (e.g. "MB01,1")
2	pattern	String	in	A text matching pattern (e.g. "**{*}")

Example of use :

```

<FORM id=ListFrm name=ListFrm>

```

```

        <INPUT type=text size=10 maxlength=10 id=SearchTxt
        name=SearchTxt value="*t*">
        <INPUT type=button id=SelectBtn name=SelectBtn
        value=Select onclick="doSelect()">
</FORM>
<SCRIPT LANGUAGE=JavaScript>
function doSelect()
{
    document.QlikDocument.searchSelect("LB05",
    document.ListFrm.SearchTxt.value);
}
</SCRIPT>

```

searchToggle

Toggles a selection in a list box or multi box field based on a search string.

ParameterNo	Parameter	Type	Direction	Comment
return value	-	-	-	No return value
1	listID	string	in	Either the QlikView Object ID of a list box (e.g. "LB01") or the QlikView Object ID of a multi box followed by a comma and a numeric index (zero-based) indicating a specific field row in the multi box (e.g. "MB01,1")
2	pattern	string	in	A text matching pattern (e.g. "*t*")

Example of use :

```

<FORM id=ListFrm name=ListFrm>
    <INPUT type=text size=10 maxlength=10 id=SearchTxt
    name=SearchTxt value="*t*">
    <INPUT type=button id=ToggleBtn name=ToggleBtn value=Toggle
    onclick="doToggle()">
</FORM>
<SCRIPT LANGUAGE=JavaScript>
function doToggle()
{
    document.QlikDocument.searchToggle("LB05",
    document.ListFrm.SearchTxt.value);
}
</SCRIPT>

```

undo

Undoes the last selection action.

ParameterNo	Parameter	Type	Direction	Comment
return value	-	-	-	No return value

Example of use :

```
<FORM id=ListFrm name=ListFrm>
  <INPUT type=button id=UndoBtn name=UndoBtn value=Undo
    onclick="doUndo()">
</FORM>
<SCRIPT LANGUAGE=JavaScript>
function doUndo()
{
  document.QlikDocument.undo();
}
</SCRIPT>
```

APPENDICES

- **QlikView Java FAQ**
- **QlikView plug-in FAQ**
- **MS Windows Installer - MSI Packages**
- **Deploying MSI Packages with Group Policies**
- **QlikView Server Script API**

1 APPENDIX A - QLIKVIEW JAVA FAQ

This appendix attempts to answer some of the most common questions with regard to the QlikView Java client solutions.

Java basics

Q: Is all Java the same?

A: No, apart from different versions of Java (1.5, 1.6 etc) there are several different uses of the Java language. One should at least differentiate between:

- JavaScript which is a scripting language for web pages etc
- Java applets (e.g. QlikView Java clients) which are programs run inside a Java

Runtime Environment in a browser.

- Java programs which are self-contained applications written in Java as opposed to e.g. VB or VC++.

Q: What is a Java Runtime Environment (JRE)?

A: The JRE is a plug-in to an internet browser which creates a "sandbox" for running applets written in Java inside the browser. The applets have no access to the machine on which they are running outside the JRE.

Java versions

Q: Which Java versions are compatible with QlikView Java clients?

A: Sun Java 1.5 or 1.6

QlikTech Java support

Q: Will all new features in QlikView be implemented in the QlikView Java clients?

A: Some will and some will not, depending on technical considerations.

-
- Q: What does QlikTech recommend users who for some reasons do not want to deploy or continue using the QlikView Java client solutions?
- A: QlikView Analyzer or QlikView Analyzer for IE (plug-in) or QlikView AJAX ZFC

Common technical issues

- Q: Sometimes when using the QlikView Java clients I get scrollbars where I should not have them. Why does a layout created in QlikView not look exactly the same in a QlikView Java client,?
- A: The JVM is a separate entity from the hosting computer. It has its own graphics library and its own fonts, which differ from those used under Windows. Slight differences in font sizes and rounding errors can make a scrollbar to go on or off compared to how it looks under Windows.
- Q: Isn't the behavior described in the previous question a bug which could be fixed by changing the program code?
- A: Unfortunately not.
- Q: Why can I not print/export directly from the QlikView Java client but have to go via an intermediate html page?
- A: The JVM is a "sand-box" which for security reasons has no contact with the resources of the host computer. Therefore local printers and disks are not reachable from the Java applet.
- Q: Is communication between QlikView Server and QlikView Java clients encrypted?
- A: Not currently, unless you use https tunneling.

2 APPENDIX B - QLIKVIEW PLUG-IN FAQ

This appendix attempts to answer some of the most common questions with regard to the QlikView plug-in client, QlikView Analyzer for Internet Explorer.

Plug-in basics

Q: What is a plug-in?

A: A program hosted by and running inside a web browser. Typically-can ActiveX component with ocx as file extension.

QlikView Plug-in specifics

Q: What version of QlikView is provided as plug-in?

A: Professional level functionality for server collaboration.

Q: Does QlikView Analyzer for IE require a serial number?

A: No.

Q: Can QlikView Analyzer for IE be registered with a serial number?

A: No.

Q: Can QlikView Analyzer for IE open local files?

A: No.

Q: What can I do with QlikView Analyzer for IE?

A: Almost everything you can do with QlikView Professional except edit script and open local files.

Q: Will the QV plug-in work with other browsers than MS IE?

A: No.

Q: Are their plans to make the QV plug-in work with e.g. Netscape?

-
- A: No, not at the moment.
- Q: Is the plug-in and the standard QV OCX the same thing?
- A: Kind of but not quite. The actual OCX is the same, but the shell each program is run under differs.
- Q: Will QVA for IE work under MacOS?
- A: No, this is Windows only!
- Q: Will QVA for IE work under Linux?
- A: No, this is Windows only!
- Q: Will QVA for IE work under Windows 98?
- A: No, Windows 2000 is the minimum requirement.

Installation and deployment issues

- Q: How and when is the QVA for IE plug-in installed?
- A: There are three basic ways/cases :
- 1 Clicking on a "download and install" link on a web page.
 - 2 Clicking on a link to a QlikView document on an HTML page encoded to detect the presence of QVA for IE and install if not present. The sample HTML pages supplied with QlikView Server will contain such links.
 - 3 In organizations where individual users are not allowed to install new software on their computers, the sysadmin may create a so called "group policy" and "shoot out" the QVA for IE client to any set of computers in the directory.
- Q: I get a message saying only administrators may install. What is wrong?
- A: You do not have the necessary permissions to install new software on your computer (see case 3 above). Contact your sysadmin.
- Q: Are there template ASP pages supplied with QVS with provisions for listing files from directories with links that invoke QVA for IE?
- A: Yes!

Q: What is written to disk when I install?

A: The following:

1 **QlikOCX.ocx** under **\Program Files\Qlikview\QlikOcx**

This is the plug-in OCX control.

2 ***.chm** under **\Program Files\Qlikview\QlikOcx**

These are the QlikView html help files in different languages.

3 **qvp.dll** under **\Program Files\Qlikview\QvProtocol**

This is the definition of the qvp protocol

4 **QvRes*.dll** under **\Program Files\Qlikview\QlikOcx**

These are the QlikView support files in different languages.

Q: How much hard disk space is taken up by the QVA for IE client?

A: Approximately 30 MB when installed. MS Installer may need twice as much for caching during installation

Q: What is the size of the download when installing?

A: Approximately 30 MB depending on msi package used.

Q: Is any part of the QVW documents accessed from QVA for IE saved on disk?

A: No, not unless you make an export of data.

Common technical issues

Q: Does the layout look exactly the same when opened in QVA for IE as when opened in a standard installed QV?

A: YES!

-
- Q: Are all layout goodies available in standard QV available in QVA for IE?
- A: YES!
- Q: Can I print from QVA for IE directly to my local printer?
- A: YES!
- Q: Can I export and send to Excel directly from QVA for IE?
- A: YES!
- Q: Is communication between QVA for IE and QVS encrypted?
- A: YES! All communication is encrypted with a 128 bit key.
- Q: How is the performance of QVA for IE compared to QlikView Java clients?
- A: Faster!
- Q: Is the QVW document data transferred to the client computer RAM when using QVA for IE?
- A: No, just as with the QlikView Java clients the data mainly stays on the server. Only what is needed to draw the screen is transferred. However, when you print and export data locally, more data will of course need to be transferred.
- Q: Is the client computer's CPU used for the AQL logic and chart calculations when using QVA for IE?
- A: No, the client machine just takes care of rendering all sheets and sheet objects and of course receiving the user's mouse clicks.
- Q: Is the load on the QVS different when using QVA for IE compared to when using QlikView Java clients?
- A: Yes, it will be slightly lower as QVA for IE (as well as the installed exe Windows clients) render all charts by itself.

Advanced technical issues

Q: Can I have multiple instances of QVA for IE running on a single client machine?

A: Yes

Q: Can I achieve the same "single object per window" design that is possible with QlikView Java Objects when using QVA for IE?

A: This client is referred to as QVA for IE plug-in - QlikX

3 APPENDIX C - MS WINDOWS INSTALLER - MSI PACKAGES

General description

When Microsoft deployed the Office 2000 in 1999 they used the first version of the MSI packaging technique. The development team behind Office 2000 is also the team behind the MSI technique. They wanted to create a safer and more reliable installation technique than the one they used for Office 97.

Microsoft discovered early that this technique was not only applicable to Office 2000, but to many other software products.

MS Windows installer consists of two parts:

- 1 The Windows Installer service, this is a service running on the operating system. This service consists of the following five files: **MSIEXEC.EXE**, **MSI.DLL**, **MSIHND.DLL**, **MSIMSG.DLL**, and **MSI-SIP.DLL**. They are located in the system32 folder.
- 2 The MSI package (*.msi file), this is the file that the software vendor distributes. It consists of a database describing the installation and one or several *.CAB files containing the files to be installed.

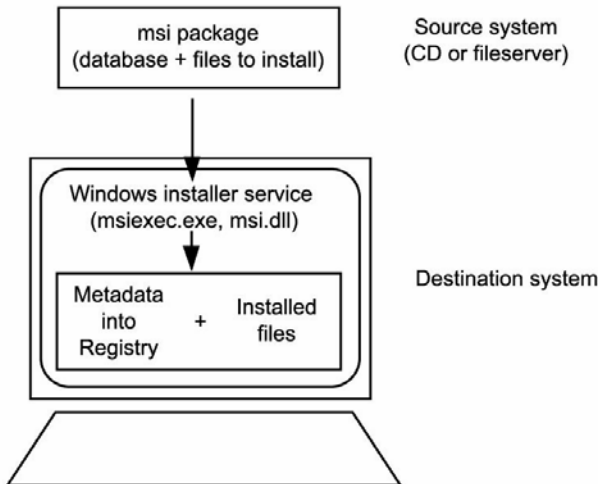
What makes the msi packages so much better than the old exe packages?

- It is a standardized procedure that all msi packages work according to the msi rules
- Extended possibility to repair and support installed applications
- Runtime files e.g. ms dll files. The runtime files are packaged and installed by so called merge modules and developed by the vendor
- Administrative installation, specially adapted to installations in a locked down environment
- Built-in rules for overwriting files
- If an installation is aborted, the rollback functionality restores the computer to the exact same state it had before the installation was executed.

The MSI installation package

An MSI file is a combination of a database and a package with the files to be installed. The database holds all the information the MS Windows Installer service needs to know about how to install, configure and uninstall an application.

During the installation process metadata about what was installed, where it was installed and from where it was installed, is saved onto the destination system. This metadata is saved in the registry and in the folder <windows>Installer and is used to administer self-repair, configuring of features, uninstall etc. Fig 1 below illustrates the relationship between the service on the destination system and the msi package.



How Windows installer works

When the Windows Installer service executes the msi package directly, i.e. from a CD, it starts by copying the msi file onto the destination systems <temp> folder from where the installation starts.

In the opening phase, the Acquisition-phase, all necessary system information is gathered, the database tables analyzed and a script that describes the following installation steps is generated. Next is the Execution-phase, it copies the files onto the system, rollback (if activated) caches its files in the hidden folder config.msi and finally all metadata is saved into the registry and to the msi file in <windows>\Installer.

When changing profile, repair and uninstall, this metadata is used. Every time an msi package is executed, Windows installer investigates what rights it shall execute with. Normally you can't install an msi package from an account with guest or user rights; it is however possible for a system administrator to give either the Windows Installer service or the msi package itself specific rights on a locked down account.

MSI packages in a locked down environment

Most computers today are configured with user accounts without rights to place files into certain folders or to edit the registry information. A common problem is therefore to run an installation that needs to add information to the registry and/or place files into locked down folders. The Windows Installer service does very thorough controls on the account that is to install a specific package. It is however possible for a system administrator to give either the whole Windows Installer service or a specific installation package rights to execute with administrator rights on a locked down account. This is called installing with elevated privileges.

Giving all MSI packages elevated privileges

On Windows NT4, 2000 and XP it is possible for a system administrator to specify a number of local policies for the Windows Installer service. These policies specify what the system is allowed to do with any Msi installations.

The policies are written in the registry on the following locations:

- Machine policies: **HKEY_LOCAL_MACHINE\Software\Policies\Microsoft\Windows\Installer**
- User policies: **HKEY_CURRENT_USER\Software\Policies\Microsoft\Windows\Installer**

There are several different policies that the system administrator can specify; this document will only handle the policy for giving the Windows Installer service elevated privileges. For more information about policies, see Microsoft developer network homepage, msdn.microsoft.com.

The policy that handles the privileges for Windows installer is called: **AlwaysInstallElevated**.

Below you find an excerpt from the Microsoft documentation about this policy:

AlwaysInstallElevated

On Microsoft® Windows NT® 4.0 or Microsoft Windows® 2000, an administrator can set the **AlwaysInstallElevated** policy for both per-user and per-machine installations. Note that this method can open the computer to a security risk because when this policy has been set a non-administrator user might run installations with elevated privileges and access secure locations on the computer, such as the **SystemFolder** or **HKLM** registry key. If the application is installed per-machine while the **AlwaysInstallElevated** policy is set, the product is treated as managed. In this case, the application can still perform a repair with elevated privileges if the policy is

removed. If the application is installed per-user while the AlwaysInstallElevated policy is set, the application is unable to perform a repair if the policy is removed.

To always install a package with elevated (system) privileges, set the AlwaysInstallElevated value to "1" under both of the following registry keys.

HKEY_CURRENT_USER\Software\Policies\Microsoft\Windows\Installer

HKEY_LOCAL_MACHINE\Software\Policies\Microsoft\Windows\Installer

If the AlwaysInstallElevated value is not set to "1" under both of the previous registry keys, the installer uses elevated privileges to install managed applications and uses the current user's privilege level for non-managed applications.

Giving a specific MSI package elevated privileges

It is possible for a system administrator to assign elevated privileges to a specific msi package (the package becomes "managed"), either by advertising the package or by installing it with the property ALLUSERS set to 1.

The Windows Installer service then gets special privileges for that specific msi package to install to locked down folders and writing to the registry.

Note It is not possible for a user with non-administrative rights to give an msi package elevated privileges, a system administrator must initially do this.

There are three different ways to give an msi package elevated privileges on the destination computer.

- 1 The system administrator advertises or installs the package for the ALL_USERS profile.
 - Advertising the package means that the non-administrator user can execute the msi package as a normal installation when the administrator has advertised the package. To advertise a package the system administrator must execute the msi package accordingly to the command line below:
msiexec /jm "myinstallation.msi"
 - Installing with the property ALLUSERS=1, means that the installed application becomes available for any user on the computer; each user is able to perform repair and/or modifications to the installed package. To install an msi package to the ALL_USER profile the system administrator must execute the package with the following command line:
msiexec /i "myinstallation.msi" ALLUSERS=1 /qb

- 2 It is possible to programmatically advertise an installation per user and at the same time define to be executed with elevated privileges. This is however a very unusual way to assign an installation with elevated privileges and it will not be described any further in this document.
- 3 The administrator defines an installation package with Group Policies Objects and then assigns the package to the user via Active Directory. Clients that run Windows 2000 or XP and are logged on to a Windows Server 2000/2003 domain can then run the installation with elevated privileges. An NT4 client can be complemented to support some parts of the Active Directory-technique.

Windows file protection

The information below is retrieved directly from the Windows installer SDK.

Windows File Protection on Windows 2000 and Windows XP

Windows Installer adheres to Windows File Protection (WFP) when installing essential system files on Microsoft Windows 2000. If a protected system file is modified by an unattended installation of an application, WFP restores the file to the verified file version. For more information, see Windows File Protection in the Platform SDK.

Windows Installer never attempts to install or replace a protected file. When the InstallFiles action or any other action scheduled before InstallFiles attempts to install a file protected on Windows 2000, the installer calls WFP with a request to install or replace the protected file. The installer requests the file installation from WFP immediately after executing the InstallFiles action. Windows File Protection installs or replaces the file on the user's system with a cached version of the protected file. Note that this does not guarantee that the version of the file installed from the cache is the version required by the application. After WFP has installed the file, the installer determines whether this version matches the version in the package. If the file version in the package is greater than the installed version, the installer informs the user that it cannot update the system and that an update of the operating system may be required for the application.

If any action sequenced after InstallFiles attempts to install or replace a protected file not already installed on the system, the installer cannot call WFP to install the file. In this case, the installer

informs the user that it cannot update the system and that an update of the operating system may be required for the application.

The installer also checks with Windows File Protection when removing files and never attempts to remove protected system files.

Component Key Files Protected by Windows File Protection

When the installer attempts to install a component's key file on Windows 2000, it first calls Windows File Protection to determine if the key file is protected. When the key file of a component is protected by WFP, and that key file is already installed, the installer updates the component only if the version of the key file in the package is greater than the installed version. If the installation package specifies that a component be installed, and the key file of the component is not currently installed, then regardless of whether the key file is protected the installer installs the component. Once any component having a key file protected by WFP is installed, it is permanently installed, and the installer never removes or replaces the component.

Note that if a Windows Installer component contains a WFP file, this file must be specified as the key path for the component.

Installation of Assemblies

Windows File Protection (WFP) for assemblies differs from Windows File Protection for Windows 2000 and Windows XP system files.

WFP protects Windows 2000 and Windows XP system files by detecting attempts to replace protected system files. This protection is triggered after WFP receives a directory change notification for a file in a protected directory. When WFP receives this notification, it determines which file has changed. If the file is protected, WFP looks up the file signature in a static catalog file to determine if the new file is the correct version. If the file version is not correct, the system replaces the file with the correct version from either the cache or distribution media.

In contrast, WFP of assemblies is dynamic. WFP is extended to files as they are added to the shared side-by-

side assembly cache. If an assembly becomes corrupted, WFP will request that the installer replace the file. Windows Installer may or may not be able to replace the file depending on whether the source package is accessible. If the source package is inaccessible, WFP will put up a dialog box stating that it is unable to restore the file.

Note that Windows File Protection protects unmanaged shared side-by-side assemblies, installed in %windir%\winsxs. Unmanaged private assemblies, installed in the application directory, are not protected by WFP. Managed global assemblies installed in the application directory or %windir%\assembly\gac are not protected by WFP.

Windows File Protection on Windows Millennium Edition

Support for Windows File Protection on Microsoft® Windows® Millennium Edition (Windows Me) is provided with Windows Installer version 1.2. Windows Me implements its own version of system file protection using catalogs that describe protected files. To completely protect a file on Windows Me, the file must be listed by Windows File Protection and there must be a catalog with version information. To install a protected file, update the version information for the file with a catalog. Windows File Protection catalogs are provided to developers to enable their applications to update system components.

Note	that if a Windows Installer component contains a WFP file, this file must be specified as the KeyPath for the component.
-------------	--

The InstallSFPCatalogFile action queries the Component table, File table, FileSFPCatalog table, and SFPCatalog table for catalogs and files. Each protected file in a catalog that can be installed separately should be put into a separate component. A catalog is associated with a protected file and is installed only if the component controlling the file is set for local installation or is already locally installed and is being repaired. The msidbComponentAttributesPermanent bit flag must be included for the component in the Attributes column of the Component Table. This prevents the removal of the catalog and protected file if the component is subsequently removed. The entry in the KeyPath column of the Component Table must exactly match the path specified by the provider of the Windows File Protection catalog. If the paths do not match, the file is not protected.

The InstallSFPCatalogFile action, FileSFPCatalog table, and SFP-Catalog table are not available with Windows Installer versions 1.0 or 1.1.

Useful links

Additional information can be found at:

msdn.microsoft.com/library (General Microsoft and Windows information)

www.desktopengineer.com (Questions&Answers, click MSI4Admins)

www.appdeploy.com (General information about application deployment)

www.wise.com (The author of the development tool: "Wise for windows installer")

Sources for this appendix

- "Microsoft developer network"

- Course documentation "Wise för programutvecklare"

4 APPENDIX D - DEPLOYING MSI PACKAGES WITH GROUP POLICIES

General

A common issue today is how to deploy applications in a network environment where the users have limited rights and how to deploy applications to a specific group of users. This document will shortly describe how to deploy Microsoft's Windows Installer (.msi) packages with group policies in an Active Directory environment.

Note Deploying software with group policies is only supported by workstations running Windows 2000, XP Professional or 2003 Server.

The QlikView .msi packages also require version 2.0 or higher of the Windows Installer service to be installed on the destination workstations.

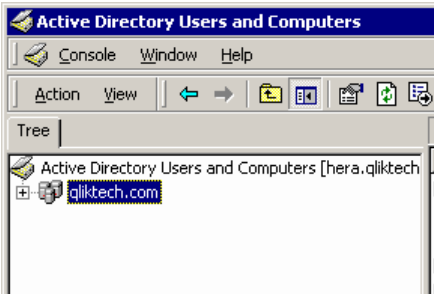
Deploying the MSI Package

When you have obtained your .msi file it must be placed in a folder shared on the network. Make sure that all users and/or computers, that will install the application, have read access to that folder. When the package is made accessible to these users and/or computers you are ready to create the Group policy object that will advertise the installation package. See section 1.3 for further information about advertising.

The package can be advertised for each user or each computer. Use the "User Configuration/Software Settings" container to advertise per user. Use the "Computer Configuration/Software Settings" container to advertise it per computer. Both containers are located in the Group Policy Object editor.

If the package is advertised per user, you can either assign or publish it. A package that is advertised per computer can only be published.

To publish a package per user means that it is listed (advertised) in the “Add programs from your network”-list in the “Add/Remove programs” dialog, see figure below.



Each user must then click the **Add** button to complete the installation.

To publish a package per computer means that the package is installed and accessible for all users on that computer the next time the computer is rebooted.

An advertised package that is assigned is also listed in the “Add programs from your network” list and can be added from there. This option also offers a few more ways to activate the installation package:

- Shortcuts, if the installation package adds any shortcuts, to desktop and/or start menu, these are added and the installation package can be executed by clicking on any of these.
- File association, the installation program is executed when the user tries to open a file that is associated with the advertised application.
- There are a few more ways to execute the installation when it is advertised as assigned but they are not applicable to any QlikView installations and therefore fall out of the scope for this documentation.

Note Executing the installation from shortcuts or via a file association is not applicable to the “QlikView Analyzer for Internet Explorer”-installation package, since it doesn’t add any shortcuts or file associations. Therefore it is not recommended to advertise QlikView installation packages with the assign option.

Advertising

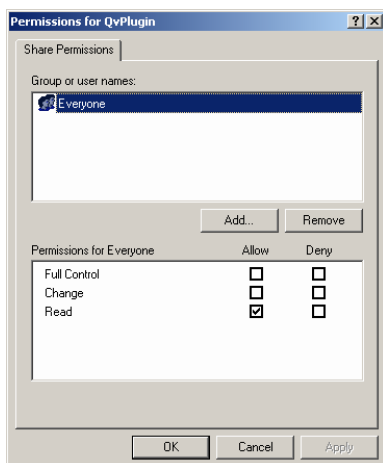
To advertise means that the administrator gives the installation package permission to execute on an account with locked down permissions.

When the package is advertised, there are so called “entry points” loaded onto the destination system. Entry points are typically shortcuts, file associations, listing in the Add/Remove programs dialog etc.

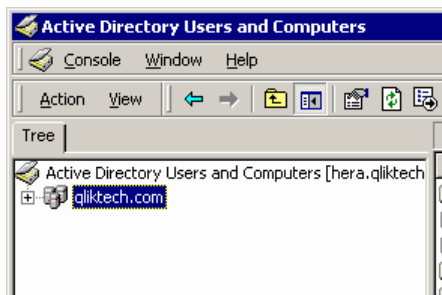
Step-by-step guide

This section provides a brief step-by-step guide for creating a group policy for the advertising of QlikView Analyzer for Internet Explorer .msi package on a number of machines in the Active Directory.

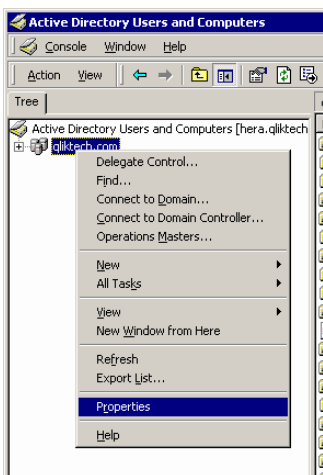
For more details on creating group policies we refer to the wealth of published literature in this field.



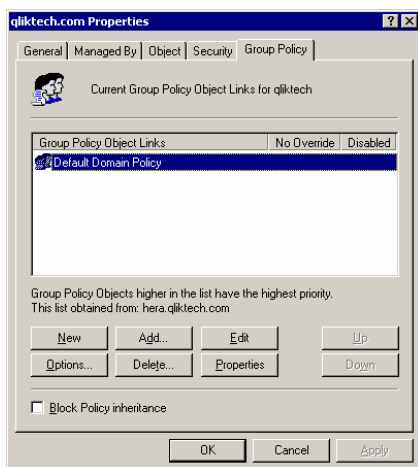
- 1 Browse to the folder containing the .msi package. Share the folder to the network users with permission to install the package.



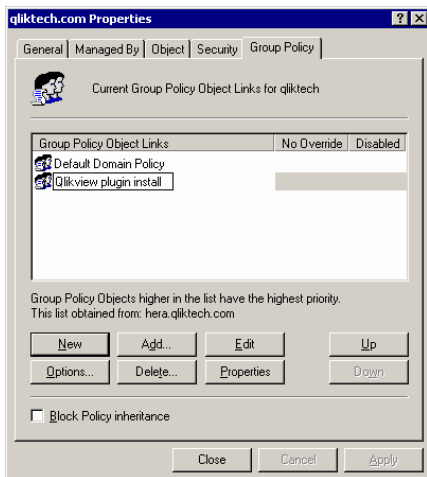
- 2 Open **Active Directory Users and Computers** and highlight the **Organizational Unit (OU)** where you want to deploy the package.



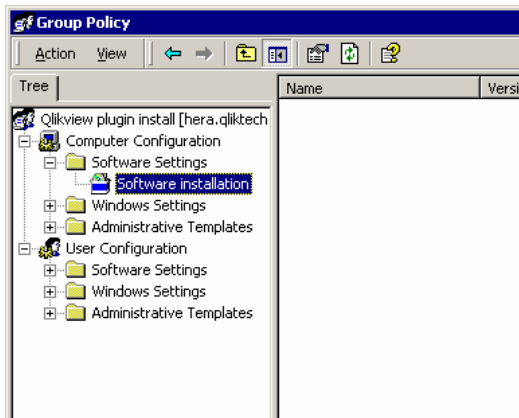
- 3 Right-click and choose **Properties**.



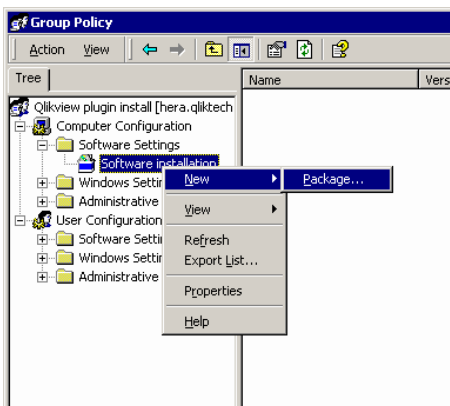
- 4 Go to the **Group Policy** tab, click **New** and give it an appropriate name.



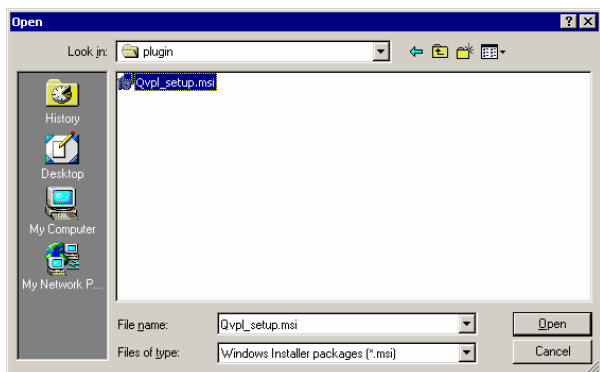
- 5 Highlight the new group policy object and press **Edit**.



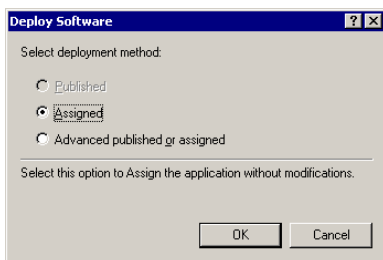
- 6 Expand to **Computer Configuration/Software Settings** or **User Configuration/Software Settings** depending on how you want to deploy the package. We select **Computer Configuration** and then highlight **Software installation**.



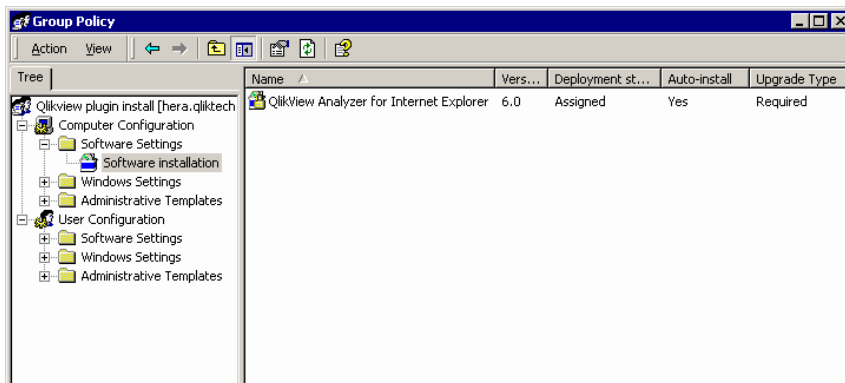
- 7 Right-click **Software installation** and choose **New -> Package**. A pop-up window is shown asking where to locate the installation package.



- 8 Find the installation package, select it and press **Open** (in this case **qvpl_setup.msi**).



- 9 Select the deployment method **Assigned** and press the **OK** button. Since we selected to apply the installation to the **Computer configuration** in item #6, it is only possible to use the **Assigned** deployment method, see section 1.2 for further information.



- 10 The deployment rule is now ready for use. All the machines in this Operational Unit (OU) get this deployment automatically. What actually happens is that when a computer is rebooted the installation program is executed so that any user who logs on to a computer in that OU, will be able to run the installed program. The rule can be applied to many different OU's.

5 APPENDIX E - QLIKVIEW SERVER SCRIPT API FOR DMS AND CALS

The Document Metadata and the CAL data in QlikView Server can be administered through an AJAX interface. This section describes the xml needed to perform these tasks. It also describes the JavaScript library **QvcLibrary.js** and how to communicate with QVS directly from a .NET language.

The XML Syntax

The first command must be:

```
<update mark="" view="" level="0" stamp=""
kind="TTT_obj" ident="new:AAA" />
```

TTT should be replaced by **MetaServiceData** or **CalInfoTab**. AAA is empty for **CalInfoTab** and the document path for MetaServiceData.

The answer should look like:

```
<result>
  <object name="TTT" ident="III" stamp="MMM"
session="SSS" ...>
  ...
</object>
  <message text="" />
</result>
```

Start to check that the text attribute of the message-tag is empty (or missing). If not, it contains the error text. Also validate that the name-attribute of the object tag is equal to the MMM you supplied. If not, the operation failed. If ok, save the ident (III), the stamp (MMM) and the session (SSS). These are needed in the next calls.

The following commands look like:

```
<update mark="" view="" level="0" stamp="MMM"
kind="TTT_obj" ident="III" session="SSS">
  <set name="NNN" ... />
  ...
</update>
```

In most cases one call is enough. To check that all went ok look in the result that the text attribute of the message-tag is empty (or missing) and that the session attribute hasn't changed.

In the case of **MetaServiceData** the last set must be:

```
<set name=".CreateFromActiveObject" action=""/>
```

In case of **CalInfoTab** the commands take effect immediately.

The most interesting commands for **CalInfoTab** are:

```
<set name="CalInfoTab.AddNamedCal"  
text="Domain\User"/>  
<set name="CalInfoTab.DeleteNamedCal"  
text="Domain\User"/>
```

Among the most interesting commands for MetaServiceData are related to authorisation:

```
<set name="MetaServiceData.AccessList.RemoveAll"  
action=""/>  
<set name="MetaServiceData.AccessList.Append"  
action=""/>  
<set name="MetaServiceData.AccessList.Selected.User"  
text="Domain\User"/>  
<set name="MetaServiceData.AccessList.Selected.Allow"  
text="Restricted"/>  
<set name="MetaServiceData.AccessList.Selected.Days"  
count=" Monday"/>  
<set name="MetaServiceData.AccessList.Selected.From"  
text="9.00"/>  
<set name="MetaServiceData.AccessList.Selected.Until"  
text="17.00"/>
```

The JavaScript Library

The file **QvcLibrary.js** contains the class **QvcObject**. All errors are issued as exceptions.

CalInfoTab example (add ten new users):

```
var calinfo = new QvcObject("CalInfoTab",  
true, true);  
calinfo.Connect("");  
for(var ix = 0; ix < 10; ++ix) {  
    calinfo.SetText("AddNamedCal", "User_" + ix);  
}  
calinfo.Commit();
```

MetaServiceData example (Set access right for two users of a document)

```
var dms = new QvcObject("MetaServiceData", true);
    dms.Connect("", "xmas.qvw");
    dms.RemoveAll("AccessList");
    dms.Append("AccessList");
    dms.SetText("AccessList.Selected.User",
"XMas\\Santa");
    dms.Append("AccessList");
    dms.SetText("AccessList.Selected.User",
"XMas\\Brownie");
    dms.SetText("AccessList.Selected.Allow",
"Restricted");

dms.Include("AccessList.Selected.Restricted.Days",
"Monday");

dms.SetText("AccessList.Selected.Restricted.From",
"9");

dms.SetText("AccessList.Selected.Restricted.Until",
"17");

    dms.Commit();
```

Communicate with QVS from a .NET Language

The component to use is **QvsPureRemote.dll**. The class is **QvClient** in the namespace **QlikTech.NetClient**. To connect use the following call:

```
QvClient client = new QvClient("localhost",
QvClient.Mode.Admin);
```

Replace localhost with server where the QVS resides.

Construct the XML as a string (here named 'request').

Make the call:

```
string data = client.Execute(request);
```

Parse the XML-text (here contained in **data**);

As always the calling process must be running as (or impersonate) a user existing in the "QlikView Administrators" group on the QVS machine.

The CalInfoTab Reference

AddNamedCal

```
<set name="CalInfoTab.AddNamedCal"
text="Domain\User"/>
```

DeleteNamedCal

```
<set name="CalInfoTab.DeleteNamedCal"
text="Domain\User"/>
```

UndeleteNamedCal

```
<set name="CalInfoTab.UndeleteNamedCal"  
text="Domain\User"/>
```

AllowDynamicNamedCals (0=false, 1=true)

```
<set name="CalInfoTab.AllowDynamicNamedCals"  
value="1"/>
```

NamedCalMode (0=User Name, 1=Machine Id)

```
<set name="CalInfoTab.DeleteNamedCal" text="0"/>
```

The MetaServiceData Reference

The following controls loading of document

AutoLoadMode (Never, Always, Restricted)

```
<set name="MetaServiceData.AutoLoadMode"  
text="Never"/>
```

AutoLoadRestricted.From (timeofday in format 17:45)

```
<set name="MetaServiceData.AccessList.Selected.From"  
text="9.00"/>
```

AutoLoadRestricted.Until (timeofday in format 17:45)

```
<set name="MetaServiceData.AccessList.Selected.Until"  
text="17.00"/>
```

AutoLoadRestricted.Days (Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, Sunday) prefix space to add minus (-) to remove

```
<set name="MetaServiceData.AccessList.Selected.Days"  
count=" Monday"/>  
<set name="MetaServiceData.AccessList.Selected.Days"  
count="-Sunday"/>
```

MaxOpenSessions (>= 0, empty string means not limitations)

```
<set name="MetaServiceData.MaxOpenSessions" text="3"/>
```

The following are used by e.g. accesspoint

Category (any text)

```
<set name="MetaServiceData.Category" text="Sales  
Document"/>
```

LastUpdate (empty or time in format 2008-12-24 13:56:00)

```
<set name="MetaServiceData.LastUpdate" text="2008-12-
24 13:56:00"/>

    NextUpdate (empty or time in format 2008-12-24
    13:56:00)

<set name="MetaServiceData.NextUpdate" text="2008-12-
24 13:56:00"/>

    EventDriven (0=false, 1=true)

<set name="MetaServiceData.EventDriven" value="1"/>

    SourceName (any text)

<set name="MetaServiceData.SourceName"
text="lastyear2.qvw"/>

    Clients (set of Plugin, Java, Ajax, Download) pre-
    fix space to add minus (-) to remove

<set name="MetaServiceData.Clients" count=" Ajax"/>

    AjaxUrl (an url, or empty)" mode="enabled" value="/
QvsDocuments/Sörens Filer/test1_ajax" text="/Qvs-
Documents/Sörens Filer/test1_ajax"/>

<set name="MetaServiceData.AjaxUrl" text="/QvsDocs/
Sales/lastyear_ajax"/>
```

The following are the access data (used when access control is set to DMS mode)

```
    AccessList (the list of control records)

    AccessList.RemoveAll

<set name="MetaServiceData.AccessList.RemoveAll"
action=""/>

    AccessList.Current (the zero based index of the
    selected reocrd (to be manipulated through Access-
    List.Selected)

<set name="MetaServiceData.AccessList.Current"
text="1"/>

    AccessList.Remove (remove the current record)

<set name="MetaServiceData.AccessList.Remove"
action=""/>

    AccessList.Append (append a new record at the end
    and make it current)
```

```
<set name="MetaServiceData.AccessList.Append"
action=""/>
```

AccessList.Selected.Anonymous (0=false, 1=true)

```
<set
name="MetaServiceData.AccessList.Selected.Anonymous"
value="1"/>
```

AccessList.Selected.User

```
<set name="MetaServiceData.AccessList.Selected.User"
text="Acme\Bob"/>
```

AccessList.Selected.Allow (Always, Restricted)

```
<set name="MetaServiceData.AccessList.Selected.Allow"
text="Restricted"/>
```

AccessList.Selected.Restricted (see Auto-LoadRestricted)

INDEX

A

addObject function	133
AJAX Zero-Footprint Client (ZFC)	153
alldocs.asp	124
Authentication vs. Authorization	86

B

Built-in http Web Server	107
--------------------------------	-----

C

clearAll	179
clickButton	179
Client Access Licenses (CALs)	43
Client Side Authentication	87
Cluster Licensing	44
Clustering	117
Communication encryption	81

D

Document Metadata Service (DMS) ...	115
Dynamic Data reduction	171

E

Event log	105
-----------------	-----

F

File Security	170
File system security on server	81
Functional Architecture	93
functions.asp	127

G

getCaption	180
getCellValue	180
getColCount	181
getColLabel	181
getLabel	182
getLabelColCount	183
getLabelRowCount	184
getListState	184
getListValue	185

getOptionalCount	186
getRowCount	187
getSelectedCount	188
getStateCount	188
getStateValues	189
getTextValue	190

H

HTML page script API	177
----------------------------	-----

J

Java applet API	175
Java Client	127
Java Objects Client	131

L

Licensing	43
-----------------	----

M

Macro functionality limitations	168
Macro trigger limitations	167
Macros on QlikView Server	167

P

Page Generator for AJAX ZFC	154
Page Generator for Java Objects	132
Page Generator for QlikX	142
Performance log	103
Plug-in Client	137
plugin_alldocs.asp	124
plugin_functions.asp	127
Plugin.asp	124

Q

QlikView clients	121
QlikView Management Console	47
QlikView Server Load Sharing	117
QlikX	141

R

redo	190
------------	-----

runMacro	191
----------------	-----

S

Sample Pages	123
Script API for DMS and CAL	221
searchSelect	191
searchToggle	192
Security Configurations	86
Server logging	101
Server Side Authentication – Get Ticket Process	89
Server Side Authentication – Non Windows Web Server	88
Server Side Authentication – Windows Web Server	87
Server Tunnel	97
session log	101
setDocument function	133
showdocument.asp	124
Small Business (SB) Edition	44
System Requirements	23

U

undo	192
------------	-----

V

VBScript functions	169
--------------------------	-----

W

Windows clients	149
-----------------------	-----