

#QLIKSUPPORT

QlikView Services & Logs Guide

Key information about QlikView services & logs

TABLE OF CONTENTS

TABLE OF CONTENTS	1
SUMMARY	2
INTRODUCTION	2
OVERVIEW	2
THE QLIKVIEW SERVICES	3
<i>Management Service</i>	3
<i>Service Dispatcher/License Service</i>	5
<i>Distribution Service</i>	6
<i>Directory Service</i>	10
<i>Server Service</i>	11
<i>Web Server Service</i>	13
<i>Best Practices for log reviews</i>	14

SUMMARY

- QlikView Logging overview
- QlikView services and what they do
- Service by service log locations and log files and purpose of the log
- Some best practice ideas for log reviews

INTRODUCTION

This whitepaper will try to supply a full overview of all the QlikView services and their associated logs. We will also supply some details on each log and the information they hold and when it may be useful to use them in troubleshooting the environment. Lastly, we will provide some tips on how to go about reviewing the logs to spot the things you want to investigate further along with any best practice settings related to quantity of logs to keep in the environment for best performance.

Overview

QlikView has a very robust set of logs to aid in managing the environment in which it runs, but it is important to use the QlikView logs instead of relying upon the Windows Event Viewer Application logs. QlikView does write information to the Windows Event Viewer Application log, but it is just a subset of what is going on in the environment, so it is always recommended to start with the QlikView log files directly and use the Windows Event Viewer logs if the QlikView logs are not supplying the information expected. Information may be gained by browsing the logs on a regular basis or using the QlikView Governance Dashboard application to load the logs for analysis and tracking of trends. You can find information about the Governance Dashboard here: [Governance Dashboard Help](#)

The QlikView Services

Let us review the QlikView Services and what they do for us. Each service will be reviewed separately, and information and links to useful topics will be included to better aid in managing the QlikView environment.

Management Service

The QMS (QlikView Management Service) is the key service in the product, it functions as the brains of the environment keeping track of the services, where they are found and checking they are online and functioning properly. QMS also handles calls from the QlikView Administrators via the QMC (QlikView Management Console) and any QMS API (Application Programming Interface) calls coming into the environment. Once the environment is configured, the QMS being offline will have no impact on the ability of the environment to run except to the QlikView Administrators being able to get into the QMC to make changes and QMS API calls would receive no response, but everything else runs as expected.

The main Help link for the service is below:

Help link: [QlikView Management Service information](#)

The QMS has one main log in C:\ProgramData\QlikTech\ManagementService\Log by default. The location may be changed via the following setting in the QMS .exe.config file found in the installation directory chosen under \QlikView\Management Service\QVManagementService.exe.config:

```
<!-- Defaults to %PROGRAMDATA%\QlikTech\ManagementService -->  
<add key="ApplicationDataFolder" value="" />
```

The information contained in the log will depend upon the logging level selected for the service, which is set via the QMC\System\Management Service\General tab settings. The default setting is Normal, if troubleshooting an issue in the environment, it is advisable to change the level to Debug to get the added logging entries to supply clues to potential root causes.

The main thing to look for in the log are any 'Error' or 'Warning' level messages, the 'Information' messages are not going to be of much troubleshooting help, but the Errors and Warnings should supply potential clues. A useful Article regarding QMS .exe.config settings is here:

Article link: [QlikView Management Service & Distribution Service key settings](#)

A key set of files related to the QMS are in the QVPR (QlikView Publisher Repository) folder in the Application Data Folder path mentioned earlier. These xml files hold most of the configuration information for the environment. These files are backed up into a zip file daily by default, so if any corruption in the files occurs, they can be restored from the latest known good zip file to return to that backup point. Any changes made between the backup and the restore would need to be re-entered via the QMC. See the Help link below for further details:

Help link: [QlikView Management Service QVPR information](#)

Below is a Techspert Thursday presentation link on migrating the QVPR:

Techspert link: [Worry Free Migration for QlikView environments](#)

There is one more log for the QMS, and that is the Audit log which will show changes to the environment made via the QMC. The log is not helpful in that it is quite cryptic in relation to the information provided, it is mostly GUID values, so not easy to translate entries, but it may supply clues if it is necessary to track down who made a change around a specific time. See the below Help link for further information, and this setting must be activated via the .exe.config file settings for the QMS, it cannot be activated from the QMC, see the Article link below for further guidance:

Help link: [QlikView Management Service Audit Log](#)

Article link: [Enabling QMC Audit Logging](#)

There are other folders and files found in the QMS Application data folder see below for what each does:

- DistributionGroups folder: This folder holds the template to configure Publisher Groups. This is used for two purposes really, both requiring a QDS (QlikView Distribution Service) cluster setup, so if you have a single node QDS, you can ignore this completely. The first purpose is to handle unbalanced servers in the QDS cluster, i.e. the servers in the cluster do not have the same CPU/Memory configurations, and the second is to be able to pin tasks to specific QDS nodes, good for situations where you have some critical tasks you want to ensure have no issues, so you run only those tasks on one of the nodes and nothing else to guarantee no other tasks take that node down and cause problems such that the critical tasks do not run timely. See the following Help link for further information:

Help link: [QDS Publisher Groups](#)

- Publisher LEF (License Enabler File) folder: This folder holds the LEF file/information for the Publisher instance. This is a bit odd as one would think the Publisher license should go with the Distribution Service, but this is an old legacy issue created by how the services used to be many versions ago! The one thing to note about this is you can only have one Publisher license per QMS/QMC, so if you want multiple QDS services in an environment, you need to be sure the account team gets you a single Publisher license with the NUMBER_OF_CLUSTER_NODES;x;; setting in it, which will allow the multiple services. Those can then be configured as separate resources or clustered resources.
- Workorders folder: This folder holds the task information contained in the QVPR files that is then pushed over to the QDS service to store in its application data folder files. There should be a folder and an .xml file inside this folder with the same GUID value. If there are more, it is due to environment changes that have been done at some point, and the old ones can be safely removed, just be sure you keep the active ones, which should be the ones with current timestamps.
- Config.xml file: This file holds information about the services, should be no need to bother with this file.
- qvpr_machinename.ini: This is another file that should not need anything done to it, it is associated with the QVPR folder.
- reginfo.txt file: This file holds the actual Publisher license registration and would only need to be removed along with the Publisher LEF folder contents if you wanted to return to the Reload Engine and not use Publisher in an environment. Note, this will break all the tasks you have configured etc. in the environment, the QVPR cannot be downgraded once upgraded, so the two options would be to restore a backup from when you last had just the Reload Engine, or start over with a new QVPR fold

Service Dispatcher/License Service

The Service Dispatcher/License Service is a new service to the QlikView product as of the 12.40/April 2019 release. It allows for Dual Use licensing as well as the new ability to Publish QlikView documents to a Qlik Sense Hub (12.50/April 2020 release). Please see the following article link for further information.

Article link: [QlikView License Service information](#)

There is a logfile for the service found in: C:\ProgramData\QlikTech\LicenseService\Log by default. If the default location was changed, check the Services.conf file found in the installation directory chosen under \QlikView\ServiceDispatcher. The service is not necessary if you are not using the Dual License setup or planning to Publish QlikView documents to the Qlik Sense Hub. If the service is having trouble starting or seems to be causing other issues, please check the log entries for clues, and if you require help in determining the meaning of any Error or Warning level messages, please contact the Technical Support team for help but check the Knowledge Base first for existing Articles.

Distribution Service

The QDS is one of the major services in the QlikView Environment, on par with the Server Service in these two using the most resources of all the services. The QDS handles supplying the QMS information on the status of the tasks running per the schedules provided by the QlikView Administrator(s). It spawns the QlikView Batch (QVB) processes that perform the reload and distribution processing for the tasks. The other process involved in reloads is the QvConnect64 or 32 processes, and these supply the connections to the ODBC/OLE DB connectors for the QVB reload processes. See the following Help link for further details about the service:

Help link: [QlikView Distribution Service information](#)

The QDS service has many moving parts and log files. If a QDS Cluster is configured in the environment, it will result in even more folders and logs, but the cluster does share some of the files in the application data location: C:\ProgramData\QlikTech\DistributionService by default (in a clustered environment, this would be a Windows Network Share (UNC) path). If the environment does have a QDS cluster, the only file in the above path will be a config file pointing to the UNC path location for the shared application data folder and any other specific configuration changes that have been made in the QDS .exe.config file. Below is a link to an article on some of the key settings in the QDS that may need to be changed in larger/clustered environments or those with large numbers of tasks:

Article link: [QlikView Management Service & Distribution Service key settings](#)

Product Management and R&D (Research & Development) (Research & Development) have also put together a whitepaper about scaling of the QlikView Publisher, which can be found in the link below:

Article link: [Scaling QlikView Publisher](#)

The main QDS log file is the Root_date file found in the application data path in the numbered folder(s)\Log folder. Each QDS node in a cluster will have its own log folder. The Root log holds the

base service operations logging entries and should be your first place to look if a QDS service is having problems. This is the service level log.

The other primary logs are the Task logs, those are also written to the same numbered folders\Log folder mentioned above in the application data path but are in the dated folder and then a specific task folder that will be in the format of 'Timestamp – Task Name.qvw.' There may be up to three files in these folders:

- TaskLog.txt: This file is the primary task log for just the task in question, it supplies details of the task processing and should be used as a starting point to troubleshoot a specific task that is not running properly.
- DocumentLog.txt: This file is the script log, particularly useful if you need to troubleshoot failure during the load process.

Article link: [How to enable Script/Document Logging](#)

- DistributionReport.xml: This file will confirm your distribution setup of the users/groups to which the document is being distributed if you did specify specific users/groups. If you chose All Authenticated, it would note that too. It does show some of the other settings being pushed to the Server service about the document settings that should be setup server-side.

There are a few other log files in the numbered folder\Log folder:

- Date.txt: This file, if it exists, will have minimal information in it unless there is a problem in the environment. It is mostly just a summary logfile of the QDS service, so not a bad place to check too if the service itself is not working as expected.
- LoadBalancer_date.txt: This file holds information regarding whether that node is experience overload conditions as related to the two following .exe.config settings in the QDS service:
 <!-- If the CPU usage % goes above this value the machine is considered overloaded and no new tasks will be started. -->
 <add key="CpuOverloadLimit" value="75"/>
 <!-- If the memory usage % goes above this value the machine is considered overloaded and no new tasks will be started. -->
 <add key="MemoryOverloadLimit" value="90"/>

- `Workorder_date.txt`: This file has information about the task synchronization with the QMS service, so good to check if you are having sync issues between the QMS and QDS services, which manifest themselves by way of tasks mysteriously running sometimes and not others or the next scheduled dates appearing/disappearing in the QMC\Status view.

Now let us go back to the main application data folder to discuss the ones the QDS service is using to keep track of everything it is doing.

Folders:

- `EDXResult`: This folder holds the EDX result files created for every task. The Help says: Until the task is completed, this file holds the status of the EDX task. When the execution is finished, it holds the result (success/fail) and the task started as a result (if any). These are created for all tasks so the number of files can increase exponentially, and it may be wise to decrease the following `.exe.config` setting to a much smaller number than the default 30 days.


```
<!-- The number of days to keep EDX Result Files -->
<add key="NbrOfDaysToKeepEDXResultFiles" value="30"/>
```
- `Repeat Variables`: This folder coincides with the `EDXResult` to a degree in that it is tied to the use of 'Script Parameters' section in the Reload tab of a Source Document Task. The files in here are temporary and should be removed automatically. It is simply a repository for the QDS to place the variable parameters temporarily while the task is being run, once the task completes, the file should be removed as well.
- `TaskExecutionHistory`: This folder holds a separate Task File for every task that exists in the environment, how many entries are inside the Task File depends upon how often the Task is run and how many days of logging are set in the `.exe.config` settings for the QDS service. These files supply the information to the QMS to populate the Task History are of the Status\Tasks\Show Task Details area and the Task History tab there.
- `TaskLogIndex`: This folder is self-explanatory in that it holds the index/links for the Tasks to their associated Task Histories.
- `TaskProgress`: I was not able to find any documentation on this folder, nor was I able to ever see anything populate to it, so the best guidance I can provide on this one is it should be empty like the Repeat Variables folder.

- **TaskResults:** This folder holds the detailed results of the last execution of the associated task, check the TaskExecutionHistory folder and task file there for the full history of the task.
- **Tasks:** This folder holds an .xml file for every task that has been configured in the environment. It is important to note that this folder may hold more task files than you have tasks showing in the QMC, as by default, the QDS does not drop tasks that are removed in the QMC. For this to happen, you need to add the EnableAdvancedFilePurge setting to the QDS .exe.config file settings per the above Article: QlikView Management Service & Distribution Service Key Settings.
- **Triggers:** This folder holds all the associated Trigger files for the Tasks which have been configured in the environment. There may be multiple Triggers for some Tasks, so the number of files may not match the number of Tasks, and if a Task has no Trigger configured, then it could be less in that case.

Files:

- **Config_servername.xml:** This file holds any changes made to the default configuration of the QDS .exe.config file, in a QDS clustered environment, the file will exist locally on each node.
- **Configuration.xml:** This file holds the base configuration information for the QDS services including information about the other QlikView services in the environment.
- **LoadBalance.xml:** This file is used in a QDS clustered environment to determine which node to start a task upon. The file updates approximate every 4 seconds, so if tasks are starting within the same 4 second window, it is most probable they will all end up on the same node in that case and putting some added space in between task start times will be needed to get things to balance better between the nodes.
- **MasterConfigurationNotification.xml:** This file is a list of configuration notification files and is used to keep QDS nodes synchronized and to notify QDS nodes of configuration changes.
- **MasterTaskExecutionHistoryNotification.xml:** This file keeps track of the TaskExecutionHistoryNotificaiton.xml file and the generation, so each QDS node in a cluster stays in synchronization.
- **MasterTaskNotification.xml:** This file keeps track of the TaskNotification.xml file and the generation, so each QDS node in a cluster stays in synchronization.

- MasterTriggerNotification.xml: This file keeps track of the TriggerNotification.xml file and the generation, so each QDS node in the cluster stays in synchronization.
- NodeInformation.xml: This file holds all other QDS node data that is not used by the load balancer.
- Registration.xml: This file holds the information for each QDS node in the cluster, so it is possible to find which node number equates to which specific node in the cluster. This file can end up with 'orphan' entries if you remove and add multiple nodes, so it may be necessary to rebuild in some cases to get everything back in proper synchronization. This would require a rebuild of all the Application data files to be sure everything stays synchronized.
- The following three files are used in a QDS cluster to keep the nodes in synchronization. It is especially important all the notification files exist on a fast disk in a clustered environment. See the NotificationFolder setting in the Article: QlikView Management Service & Distribution Service Key Settings above.
 - TaskExecutionHistoryNotification.xml
 - TaskNotification.xml
 - TriggerNotification.xml

Below are a couple of other Help topics related to some added logging that can be activated for the QDS services to gain added insights into how the services are performing:

Help link: [QlikView Distribution Service Task Performance Summary Log](#)

Help link: [QlikView Distribution Service Reload Performance Log](#)

Here are two more articles on activating more logging for the QVB (QlikView Batch) processes and the QvConnect services that may be useful when troubleshooting task failures:

Article link: [Enable QVB Logging](#)

Article link: [QvConnect32/64 logging](#)

Directory Service

The DSC (Directory Service Connector) is a simple service, for most environments, it is simply supplying a means to lookup users and groups from an internal domain via an LDAP connection to the

LDAP servers. This is used when adding users/groups to distribution tasks in Publisher, when adding users to Named and Document CALs (Client Access Licenses), and anywhere else in the QMC a search for users and/or groups is possible.

There are two exceptions to the above, the first is if an environment deploys the Custom Directory DSP (Directory Service Provider), as in this case, the Directory Service is involved in performing both Authentication and Authorization requests. This configuration is normally used by smaller organizations as having more than a couple to few hundred users/groups configured can start to cause performance issues, so it is really meant for small environments that do not have their own LDAP directories or third party directories. The Custom Directory entries are stored in the Application Data folder C:\ProgramData\QlikTech\DirectoryServiceConnector by default, in the CustomDirectoryData.xml file.

The other exception is when the QlikView Server Service is running in DMS (Document Meta Data) (Document Meta Data) Security mode. When this is the case, the DSC is again involved in performing the Authorization lookups for the QlikView Server. The information is stored by the QlikView Server in its .meta files that have the same name as their associated .qvw files.

The DSC has a single log to supply information about the requested lookups of users and groups, which may be helpful in troubleshooting situations where the DSC is being requested to get the Group memberships for Users coming into the environment. This is mostly employed when third-party authentication Single-Sign-on (SSO) systems are employed. The log can confirm whether the query is getting sent and what is being returned. There is one key note here, the DSC will only supply group lookups for the domain in which the User belongs, so you want to be sure you are not trying to use a cross-domain group setup in these situations, as it will not work.

The Resources folder in the DSC Application Data folder path, holds one record for every DSP that is configured in the DSC resource.

Below is the main Help site link for the DSC:

Help link: [QlikView Directory Service information](#)

Server Service

The QVS (QlikView Server Service) is the other main service in the QlikView product, it supplies the means for the Users to view their applications without the need for major resources on their local client machines. It can also handle Authorization requests if it is running the DMS security mode and it may handle distribution requests from the QDS if the distribution task has been configured as a Server

Distribution. When the QDS distributes to a Folder or via Email, the QDS handles the entire distribution. See the below link for more information on the QVS:

Help link: [QlikView Server Service information](#)

The following article explains in further detail the memory and CPU usage of the service:

Article link: [QIX Engine Memory Management & CPU Utilization](#)

The QVS Session logs record every user session that is started, but the log entry is only written at the end of the session, so it cannot be used in a real-time situation to try to see active sessions. The QMC\Status\QVS Statistics area is where you can see the active information. The Session logs have some useful information such as the Exit reason for the session, which application the user was using, the client they were using and the version of that client. It is good to check the Session logs whenever a user reports an incident to see if there is anything in the log to supply a clue as to what may have happened to the user's session. See the below link for more information.

Help link: [QlikView Server Session Log](#)

The QVS Performance log supplies interval reporting on the QVS service, the most useful information in this log is to quickly see if there was an unplanned QVS restart and to spot if the Server was using most of the resources available to the server when that restart occurred. The Performance log is a quick place to check when having stability issues to see if the environment is running in the normal parameters or if something has changed in relation to those normal numbers. See the below link for more information.

Help link: [QlikView Server Performance Log](#)

The QVS Event log is the primary log of the QVS service, it is the main service log for the QVS service and will report everything the QVS service is doing if the log verbosity is set to the High option. That choice will include all log information messaging except Debug level messages, the QVS log verbosity must be manually set to a value of 900 to obtain those messages in the QVS Settings.ini file. The QVS Event log is where to start looking if the Server service is having issues. The Low verbosity supplies Error level messages and Medium adds Warning messages. Notice messages may appear on both of those too, but they are not likely to be helpful. It is the Information level messages that supply context to the Warning and Error messages in most cases. If you find Internal Inconsistency messaging, those are going to relate service crashes, and it will be necessary to open a support case if you cannot find any Articles about the specific message/symptoms experienced. See the below link for more information,

and the Article link for increasing the logging level to 'Debug.' See the below help link for further information:

Help link: [QlikView Server Event Log](#)

Some useful troubleshooting articles related to log entries that may be encountered:

Article link: [Enable Debug logging level in Server service](#)

Article link: [AAALR / Disable New Row Applicator](#)

Article link: [QVS Event log fno\(x\) error list](#)

The Server Side Extension log is only active if the environment has installed QlikView Extensions running. See the below help link for further information.

Help link: [QlikView Server Server Side Extension Log](#)

The QIX (Qlik Indexing Engine) performance log supplies detailed information on the QIX Engine performance. By default, the QIX Performance log file is disabled. See the following help and article links for further information:

Help link: [QlikView Server QIX \(Qlik Indexing Engine\) Performance Log](#)

Article link: [Qix Performance Log](#)

The end-user audit log holds information on user selections, including cleared selections, activated sheets, application of bookmarks, accessed reports, and maximized objects. See the following help and article links for further details:

Help link: [QlikView Server End User Audit Log](#)

Article link: [Audit Logging](#)

Web Server Service

The Web Server Service QVWS (QlikView Web Server) performs one of two functions depending upon which web server is being used in the environment. If the QlikView Web Server Service is being used, the QVWS is a fully functional web server based upon the Microsoft IIS (Internet Information Services)

redistributable engine. If the QlikView Settings Service is being used, the QVWS is supplying an interface to send IIS configuration information and collect setting information in the IIS Web Server. The log file for the service holds useful information related to the client to web server communications such as authentication request/response information, the documents to be shown in the AccessPoint view and issues related to communication with the Server service. See the following help link for further information:

Help link: [QlikView Web Server Service information](#)

Best Practices for log reviews

The first step to troubleshooting is to understand which log files to use, and then to ensure the logging level is set to the highest level possible if you are trying to troubleshoot an active problem, for intermittent issues, this can be a bit tricky. With the log level verbosity at the highest level, there will be three main message types, Information, Warning and Error. In some cases, there may also be Debug and Notice messages. It is best to use something like NotePad++ to load the logs into and then search on 'Warning' or 'Error' to quickly see those messages, and if there is something of interest, that record can be clicked to take you to the full log location to search the surrounding messages to determine if there is something to point to a cause of the issue or not. It is good to search the Support Knowledge area and Community for messages to see if there are already existing articles or posts that may supply further clues.



About Qlik

Qlik's vision is a data-literate world, one where everyone can use data to improve decision-making and solve their most challenging problems. Only Qlik offers end-to-end, real-time data integration and analytics solutions that help organizations access and transform all their data into value. Qlik helps companies lead with data to see more deeply into customer behavior, reinvent business processes, discover new revenue streams, and balance risk and reward. Qlik does business in more than 100 countries and serves over 50,000 customers around the world.

[qlik.com](https://www.qlik.com)

© 2020 QlikTech International AB. All rights reserved. All company and/or product names may be trade names, trademarks and/or registered trademarks of the respective owners with which they are associated.