



Log column details

Revision 20210304

NOTE

This document is confidential and proprietary of **Denodo Technologies**.
No part of this document may be reproduced in any form by any means without prior written authorization of **Denodo Technologies**.

Copyright © 2022
Denodo Technologies Proprietary and Confidential

1.1

1.2 **GOAL**

This document describes the content of the VDP logs and Denodo Monitor logs.

1.3 **VDP LOGS**

Denodo Virtual DataPort uses the Apache-log4j log system that allows registering log messages with different levels of granularity. It can be configured using the log configuration files.

The Denodo Platform logs are stored in the following path `DENODO_HOME/logs/`. There is a folder for the logs of each installed module.

For instance, the logs for the VDP Server are located in the `DENODO_HOME/logs/vdp` folder and the logs for the VDP Administration Tool are in the `DENODO_HOME/logs/vdp-admin` folder.

The purpose of this section is to explain the log column details for VDP Logs.

1.3.1 **Most relevant information in VDP logs**

- The `vdp.log` file contains general logging events from the Virtual DataPort Server. Usually, it is worth looking into this file when any issue arises while working with VDP.
- The `vdp-admin.log` file contains general logging events from the Virtual DataPort Administration Tool.
- The `vdp-requests.log` file contains the VQL requests. It can be used to check all the statements received by the server.
- The `vdp-queries.log` file contains the VQL requests with more detailed information. When the information provided by the `vdp-requests.log` is not enough, this file can be checked in order to know for instance which user has executed which query, the status of the execution or the total duration of the query.
- The `vdp-cache.log` file contains cache accesses in Virtual DataPort.

1.3.2 **Vdp.log and vdp-admin.log column description**

The `vdp.log` and `vdp-admin.log` files contain the following columns:

- **Elapsed Milliseconds:** Number of milliseconds elapsed from the moment the logging began.
- **Thread Name:** Name of the thread that generated the logging event.
- **Error Priority:** Priority of the logging event, which can take one of the following values: TRACE, DEBUG, INFO, WARN, ERROR and FATAL.
- **Timestamp:** Date when the logging event occurred.
- **Category:** Category of the logging event.
- **Nested Diagnostic Content:** Contextual information of the thread that generated the logging event.
- **Message:** Description of the logging event.

- **Error Trace:** Stack trace of the thread that generated the logging event.

Note that if TRACE error level is enabled, additional columns can be added.

1.3.3 Vdp-requests.log columns description

The vdp-requests.log file logs all the requests processed by the Server. It contains the following columns:

- **Elapsed Milliseconds:** Number of milliseconds elapsed from the moment the logging began.
- **Thread Name:** Name of the thread that generated the logging event.
- **Error Priority:** Priority of the logging event, which can take one of the following values: TRACE, DEBUG, INFO, WARN, ERROR and FATAL.
- **Timestamp:** Date when the logging event occurred.
- **Category:** Category of the logging event.
- **Nested Diagnostic Content:** Contextual information of the thread that generated the logging event.
- **VQL:** Contains the VQL executed in the request.

1.3.4 Vdp-queries.log columns description

The vdp-queries.log file contains the following columns:

- **Elapsed Milliseconds:** Number of milliseconds elapsed from the moment the logging began.
- **Thread Name:** Name of the thread that generated the logging event.
- **Timestamp:** Date when the logging event occurred.
- **Nested Diagnostic Content:** Contextual information of the thread that generated the logging event.
- **Message:** Description of the logging event.
- **Database name:** Name of the target database of the query.
- **UserName:** User that launched the query.
- **NotificationType:** Type of logged notification (start Request or end Request)
- **SessionId:** Session identifier.
- **StartTime:** Start time of the query.
- **EndTime:** Finish time of the query.
- **Duration:** Duration of the query.
- **Waiting Time:** Time the query was waiting to be executed.
- **Number of Rows:** Number of rows returned by the query.
- **State:** Status of the query.
- **Completed:** Boolean showing if the query was completed.
- **Cache:** Boolean showing if the query used the cache or not.
- **Query:** VQL of the query.
- **TotalRequests:** Total number of queries executed since the server was booted..
- JDBC time columns:
 - **TotalJDBCConnectionTime:** The total time spent to open all JDBC connections established during the execution of the query
 - **PartialJDBCConnectionTime:** List of time values spent to open each JDBC connection that was established during the execution of the query.
 - **TotalJDBCResponseTime:** The total time spent to receive the first row

- in all the JDBC wrappers executed during the execution of the query
- **PartialJDBCResponseTime:** List of time values spent to receive the first row on each JDBC wrappers executed during the execution of the query.
 - **FeedCache:** Boolean showing if the query updated the cache or not.
 - **CachedResultsViewNames:** Names of the views whose data was obtained from the cache.
 - **TotalCacheResultCheckTime:** The total time spent to check if the data of the view using the Partial cache mode is stored in the cache database
 - **PartialCacheResultCheckTime:** List of time values spent to check if the data are stored in cache database, when each view using the Partial cache mode is executed.
 - **JDBCConnectionIds:** List of the identifiers of the JDBC connections used during the execution of the query.
 - **RequestType:** Type of operation performed by the query.
 - **Elements:** Elements used by the query.
 - **UserAgent:** Client user agent that performed the query.
 - **AccessInterface :** Type of interface used by the client.
 - **ClientIP:** The ip of the client.
 - **TransactionID:** The identifier for the transaction

1.3.5 Vdp-cache.log columns description

The vdp-cache.log file logs cache events. It contains the following columns:

- **Elapsed Milliseconds:** Number of milliseconds elapsed from the moment the logging began.
- **Thread Name:** Name of the thread that generated the logging event.
- **Timestamp:** Date when the logging event occurred.
- **Nested Diagnostic Content:** Contextual information of the thread that generated the logging event.
- **Message:** Description of the logging event.

1.4 DENODO MONITOR LOGS DESCRIPTION

The Denodo Monitor is a tool included in the Denodo Platform that logs information about the Denodo servers.

The purpose of this section is to explain the log column details for Denodo Monitor Logs.

1.4.1 Most relevant information in Denodo Monitor logs

- The processes.log contains information about all the running processes in the local computer such as the PID and memory usage of the process.
- The sockets.log contains information of the active connections of the local computer such as the protocol, local address and PID.
- The vdp-threads.log collects information such as the CPU usage for every thread in the server, it can be useful to detect queries monopolizing the

resources or blocked threads.

- The `vdp-queries.log` collects information about the queries executed in the server such as the query itself, the username of the user who sent the query and the execution status.
- The `vdp-loadcacheprocesses.log` collects information about the load cache processes such as the name of the view cached or the number of tuples loaded.
- The `vdp-connections.log` collects information about the connections established with VDP such as the login name, the name of the client app and the client IP.
- The `vdp-datasources.log` collects information about the usage of the data sources such as the number of requests per data source.

Note that the information from these log files can be combined, for instance, you can match `vdp-connections.log` with `vdp-queries.log` by the field `SessionId` to discover from which IP was sent some query.

1.4.2 Processes monitor

This monitor logs information about all the running processes in the computer such as the PID and memory usage of each process. The output is logged to `denodo-monitor/logs/processes.log`. It contains the following columns:

- **Image Name:** The name of the process or the executable file running the process.
- **PID:** The process ID.
- **Session Name:** Name of the session that executes the process.
- **Session#:** Number of the session.
- **Mem Usage:** Amount of memory used by the process in KB.
- **Status:** Gives the current status of the process as "Running", "Not Responding", or "Unknown".
- **User Name:** User account under which the process is running.
- **CPU Time:** Total amount of CPU time used by the process since its start.
- **Window Title:** Windows display name of the process if it exists.

1.4.3 Sockets monitor

It logs information of the active connections in the computer such as the protocol, local address and PID. The output is logged to `denodo-monitor/logs/sockets.log`. It contains the number of UDP and TCP sockets classifying the last ones by their status. It also logs the active connections with the following columns:

- **Proto:** Protocol used in the connection.
- **Local Address:** Ip address and port of the local computer.
- **Foreign Address:** Ip address and port of the remote computer where the socket is connected to.
- **State:** Status of the connection.
- **PID:** Process ID

1.4.4 Resources Monitor

This monitor shows the busy resources for an application. The output is logged to `denodo-monitor/logs/<Denodo server name>-resources.log`. It collects information about the resources usage such as the memory size, number of threads and opened connections, garbage collector and CPU usage.

The log includes the following columns:

- **Server Name:** Logical name of the logged server. Determined by the name of the monitor in the `ConfigurationParameters.properties`
- **Host:** Host of the VDP server.
- **Port:** Port of the VDP server.
- **Date:** Date of the log message
- Columns indicating the **memory usage** of the JVM that executes the monitored server. The following columns contain the size in bytes of each space of the heap. For each space, there are two values: the amount of bytes currently used by the space and its total size.
 - **Perm Gen (Denodo 6):** Data about the permanent generation space.
 - **Metaspace (Denodo 7 and 8):** Data about the metaspace (Java 8+).
 - **(Par) Survivor Space:** Data about the survivor space.
 - **CMS Old Gen:**
 - **PS Old Gen (Denodo 7):** Data about the old generation space.
 - **Tenured Gen (Denodo 6):** Data about the tenured generation space.
 - **Eden Space:** Data about the Eden space.
 - **Code Cache:** Data about the Code Cache space.
 - **Heap Memory Usage:** Sum of the memory used/total size of the Eden Space, Survivor Space and Tenured Generation space.
 - **Non Heap Memory Usage:** Sum of the memory used/total size of the Permanent Generation and the Code Cache spaces.
- Columns indicating **classes information**:
 - **Loaded Classes Count:** Number of Java classes that are currently loaded.
 - **Total Loaded Classes:** Total number of Java classes loaded since the Server was started.
- Columns indicating **threads information**:
 - **Thread Count:** Current number of threads.
 - **Peak Thread Count:** Peak number of threads since the Server was started.
- VDP information (this is filled only when the monitor is connected to a VDP server)
 - **VDP Total Conn:** Number of connections opened since the VDP server was started.
 - **VDP Active Conn:** Number of currently active connections to the VDP Server.
 - **VDP Active Requests:** Number of request that are currently being executed.
 - **VDP Waiting Requests:** Number of queued requests.
 - **VDP Total Mem:** Total amount of memory currently committed by the Server. That is, memory that is currently reserved by the Java process of the Server.
 - **VDP Max Mem:** Maximum amount of memory that the Server will attempt to use.
- **CPU%:** CPU Percentage currently used.

- Information about the Garbage Collection process. The name of these columns depends on the Garbage Collector that the Java Virtual Machine is using:
 - **GC_CC:XXX**: total number of collections performed by the collector XXX since the Server was started.
 - **GC_CT:XXX**: accumulated time in milliseconds spent by the collector XXX in collection.
 - **- GC%**: current CPU usage dedicated to garbage collection (percentage)

When the server slows down this is a good place to start looking at. This behaviour could be affecting the server because of the memory/CPU configuration or the number of requests received by the server.

The resources log can be used in combination to the queries monitor, for instance to check the memory consumption by any query or to discover which are those queries being executed that reach the memory limit.

1.4.5 Threads Monitor

The threads monitor logs the threads launched by a Denodo server and how much CPU time they consume.

Its log file is stored in `denodo-monitor/conf/<Denodo server name>-threads.log`

This monitor shows the % of execution time of each thread between the intervals of monitored time (**CpuTime**). In VDP server, it is possible with the name of the thread to identify which query being executed is consuming CPU in an abnormal way. Hence, it is useful to find those queries monopolizing the resources or blocked threads causing an inconsistent behaviour of the server.

Frequently, this log file shows that the Cache Maintenance Task is enabled for several databases sharing the cache database, although the recommendation is to run the `CLEAN_CACHE_DATABASE` stored procedure from a Scheduler job.

1.4.6 VDP Queries Monitor

It logs all the requests sent to a VDP server. It also can be configured to log only the ones targeted to a specific VDP database of that server (property `'vdpqueries.jmx.database'` of `'conf/ConfigurationParameters.properties'`)

Its log file is stored in `denodo-monitor/conf/vdp.queries.log` and contains the following columns:

- **ServerName**: Logical name of the logged server. Determined by the name of the monitor in the `ConfigurationParameters.properties`
- **Host**: Host of the VDP server.
- **Port**: Port of the VDP server.
- **Id**: Query number.
- **Database**: Name of the target database of the query.
- **UserName**: User that launched the query
- **NotificationType**: Type of logged notification (start Request or end Request)
- **SessionId**: Session identifier.
- **StartTime**: Start time of the query.

- **EndTime:** Finish time of the query.
- **Duration:** Duration of the query.
- **WaitingTime:** Time the query was waiting to be executed.
- **NumRows:** Number of rows returned by the query.
- **State:** Status of the query.
- **Completed:** boolean showing if the query was completed.
- **Cache:** boolean showing if the query used the cache or not.
- **Query:** VQL of the query.
- **RequestType:** Type of operation performed by the query.
- **Elements:** Elements used by the query.
- **UserAgent:** Client user agent that performed the query.
- **AccessInterface:** Type of interface used by the client.
- **ClientIP:** The ip of the client.
- **TransactionId:** The id of the transaction.
- **WebServiceName (Denodo 7):** Name of the involved web service.

1.4.7 VDP Cache Monitor

It logs the activity of the processes that load the Virtual DataPort cache. The output is logged to `denodo-monitor/logs/vdp-loadcacheprocesses.log`. It contains the following columns:

- **SessionId:** Session identifier.
- **ServerName:** Logical name of the logged server. Determined by the name of the monitor in the `ConfigurationParameters.properties`
- **Host:** Host of the VDP server.
- **Port:** Port of the VDP server.
- **NotificationType:** Type of logged notification (start Request or end Request)
- **NotificationTimestamp:** Timestamp of the logged notification.
- **Id:** Unique identifier of the cache load process.
- **QueryPatternId:** Identifier of the query pattern.
- **DatabaseName:** Name of the target database of the query.
- **ViewName:** Name of the view in VDP.
- **ViewId:** The internal name of the view.
- **SqlViewName:** SQL name of the view
- **ProjectedFields:** Names of the projected fields in the query.
- **NumConditions:** Number of conditions in the query.
- **VDPConditionList:** Conditions list.
- **CacheStatus:** The current configured cache mode for the view, possible values are OFF / PARTIAL / PARTIAL EXACT / PARTIAL PRELOAD / PARTIAL EXACT PRELOAD.
- **TtlStatusInCache:** The TTL configuration type for the cache, possible values:
 - CUSTOM - If the TTL was other than default value.
 - DEFAULT - If the TTL is default
 - NOEXPIRE - If the TTL for the cache is set not to expire.
- **TtlInCache:** Time to live in cache.
- **QueryPatternState:** Status of the query pattern.
- **Exception:** Possible exceptions.
- **NumOfInsertedRows:** Number of rows inserted.
- **NumOfReceivedRows:** Number of rows received by the query.
- **Storage cache times:**
 - **StartQueryPatternStorageTime** - Time when the QueryPattern

- started to be stored in the Cache Metadata table
- **EndQueryPatternStorageTime** - Time when the QueryPattern completed to be stored in the Cache Metadata table
- **QueryPatternStorageTime** - The time elapsed for the query pattern to be stored.
- **StartCachedResultMetadataStorageTime** - The start time of the cache metadata update process, this includes, Query Pattern update, Cache Status update, TTL update etc.
- **EndCachedResultMetadataStorageTime** - The end time of the cache metadata update process, this includes, Query Pattern update, Cache Status update, TTL update etc.
- **CachedResultMetadataStorageTime** - The elapsed time for storing the Cache Metadata
- **StartDataStorageTime** - Time when the the cached data started to be stored in the underlying cache table
- **EndDataStorageTime** - Time when the the cached data completed to be stored in the underlying cache table
- **DataStorageTime** - The time elapsed for the cached data to be stored in the underlying cache table .
- **Truncateddata:** "*projectedfields*", "*exception*" and "*vdpconditionslist*" fields carry long texts and the Denodo Monitor by default takes 4000 characters. In case if these fields exceed 4000 characters this field will be set as 1 to indicate to there was an information loss.

1.4.8 VDP Connections Monitor

It logs information about opened and closed connections. The output is logged to `denodo-monitor/logs/vdp-connections.log`. It contains the following columns:

- **ServerName:** Logical name of the logged server. Determined by the name of the monitor in the ConfigurationParameters.properties
- **Host:** Host of the VDP server.
- **Port:** Port of the VDP server.
- **NotificationType:** Type of logged notification (start Request or end Request)
- **ConnectionId:** Identifier of the connection.
- **ConnectionStartTime:** Start time of the connection.
- **ConnectionEndTime:** End time of the connection.
- **ClientIP:** Ip of the client that performed the connection.
- **UserAgent:** Client user agent.
- **AccessInterface:** Type of access interface of the connection.
- **SessionId:** Session identifier.
- **SessionStartTime:** Session start time.
- **SessionEndTime:** Session end time.
- **Login:** User login name.
- **DatabaseName:** Name of the target database.
- **WebServiceName:** Name of the involved web service.
- **JMSQueueName:** Name of the JMS queue.
- **IntermediateClientIP:** IP address where the service is running (only for SOAP, REST and the global RESTful Web service).

1.4.9 VDP Data Sources Monitor

It logs information about the data sources of the VDP server. The output is logged to `denodo-monitor/logs/vdp-datasources.log`. It contains the following columns:

- **Date:** Date when logging event occurred.
- **DatabaseName:** Name of the database the data source belongs to
- **DataSourceType:** Type of the data source.
- **DataSourceName:** Name of the data source.
- **ActiveRequests:** Number of active requests being currently executed on the data source.
- **NumRequests:** Number of total requests to this data source since the launch of the Virtual DataPort server.
- **MaxActive:** Maximum active requests to this data source.
- **NumActive:** Number of connections established with the data source that are being used to execute a query. An empty value means there is no pool.
- **NumIdle:** Number of inactive connections in the pool. An empty value means there is no pool.
- **PingStatus (Denodo 7 and 8):** Status of the data source after a ping request, which can take one of the following values: UP, DOWN and TIMEOUT.
- **PingExecutionTime (Denodo 7 and 8):** Moment at which the ping was performed.
- **PingDuration (Denodo 7 and 8):** Response time in milliseconds of the data source to a ping request.
- **PingDownCause (Denodo 7 and 8):** In case the data source is DOWN, the error captured when the ping was performed.

References

[Storing Denodo Monitor logs in an external Database](#)