



Hewlett Packard
Enterprise

HPE Vertica VHist ETL Overview

HPE Vertica Analytic Database

May, 2016

Legal Notices

Warranty

The only warranties for Hewlett Packard Enterprise products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. Hewlett Packard Enterprise shall not be liable for technical or editorial errors or omissions contained herein.

The information contained herein is subject to change without notice.

Restricted Rights Legend

Confidential computer software. Valid license from Hewlett Packard Enterprise required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Copyright Notice

© Copyright 2016 Hewlett Packard Enterprise Development L.P.

Trademark Notices

Adobe® is a trademark of Adobe Systems Incorporated.

Microsoft® and Windows® are U.S. registered trademarks of Microsoft Corporation.

UNIX® is a registered trademark of The Open Group.

This product includes an interface of the 'zlib' general purpose compression library, which is Copyright © 1995-2002 Jean-loup Gailly and Mark Adler.

Table of Contents

What is VHist?	4
About this Document.....	4
Overview VHist ETL.....	4
VHist ETL Steps.....	5
Source-to- Stage ETL Process Flow.....	6
Stage-to-Star ETL Process Flow.....	7
About VHist Incremental Loads.....	8
VHist Fact Tables.....	8
Overview VHist Fact Tables.....	8
USER_SESSIONS_FACT	8
QUERY_PROFILES_FACT	12
LOAD_STREAMS_FACT	15
DISK_STORAGE_FACT	18
PROJECTION_STORAGE_FACT	20
RESOURCE_REJECTIONS_FACT	22
PROJECTION_USAGE_FACT.....	24
SYSTEM_RESOURCE_USAGE_FACT.....	26
VHist Dimension Tables	28
Overview VHist Dimension Tables.....	28
DATE_DIM.....	28
TIME_DIM.....	28
BATCH_DIM.....	29
NODE_DIM.....	29
USER_DIM.....	30
Find More Information.....	32
Contact Us.....	32

What is VHist?

VHist (Vertica History) is a sample star schema that is populated with information about system processes, including user sessions, cluster management, memory management, disk activity, and I/O. A VHist ETL application loads the data from Vertica system tables into the VHist data warehouse.

The data in Vertica system tables is dynamic, reflecting the constantly-changing state of the system. However, when the data is captured in the VHist star schema, it is suitable for querying, allowing DBAs to monitor system activity over time.

Vertica ETL QuickStart applications are based on VHist. The ETL QuickStarts are sample apps created using different ETL products from Vertica technology partners. The QuickStarts are available for download on the [HPE Big Data Marketplace](#).

VHist and the Vertica ETL QuickStarts are freely available for demonstration and educational purposes. They are not governed by any license or support agreements and are not suitable for deployment in production environments.

About this Document

This document describes the VHist schema and provides an overview of the ETL processes that populate the VHist dimension tables and fact tables.

This document is included in the download package with each ETL QuickStart. You can also find this document posted in the [Vertica Knowledge Base](#) on the [HPE Developer Community](#).

Overview VHist ETL

ETL processes extract data from Vertica system tables in the V_CATALOG and V_MONITOR schemas and load it into the dimension tables and fact tables in the VHist schema.

VHist has five dimension tables. ETL processes populate two of them with data from the USERS and NODES tables in V_CATALOG. SQL scripts generate the DATE, TIME, and BATCH dimension tables without ETL processing. For details, see [VHist Dimension Tables](#).

VHist has eight fact tables. ETL processes populate each of them with data from a single source table in V_MONITOR. ETL for six of the fact tables is based on the source timestamp. ETL for the other two occurs at a user-defined frequency. For details, see [VHist Fact Tables](#).

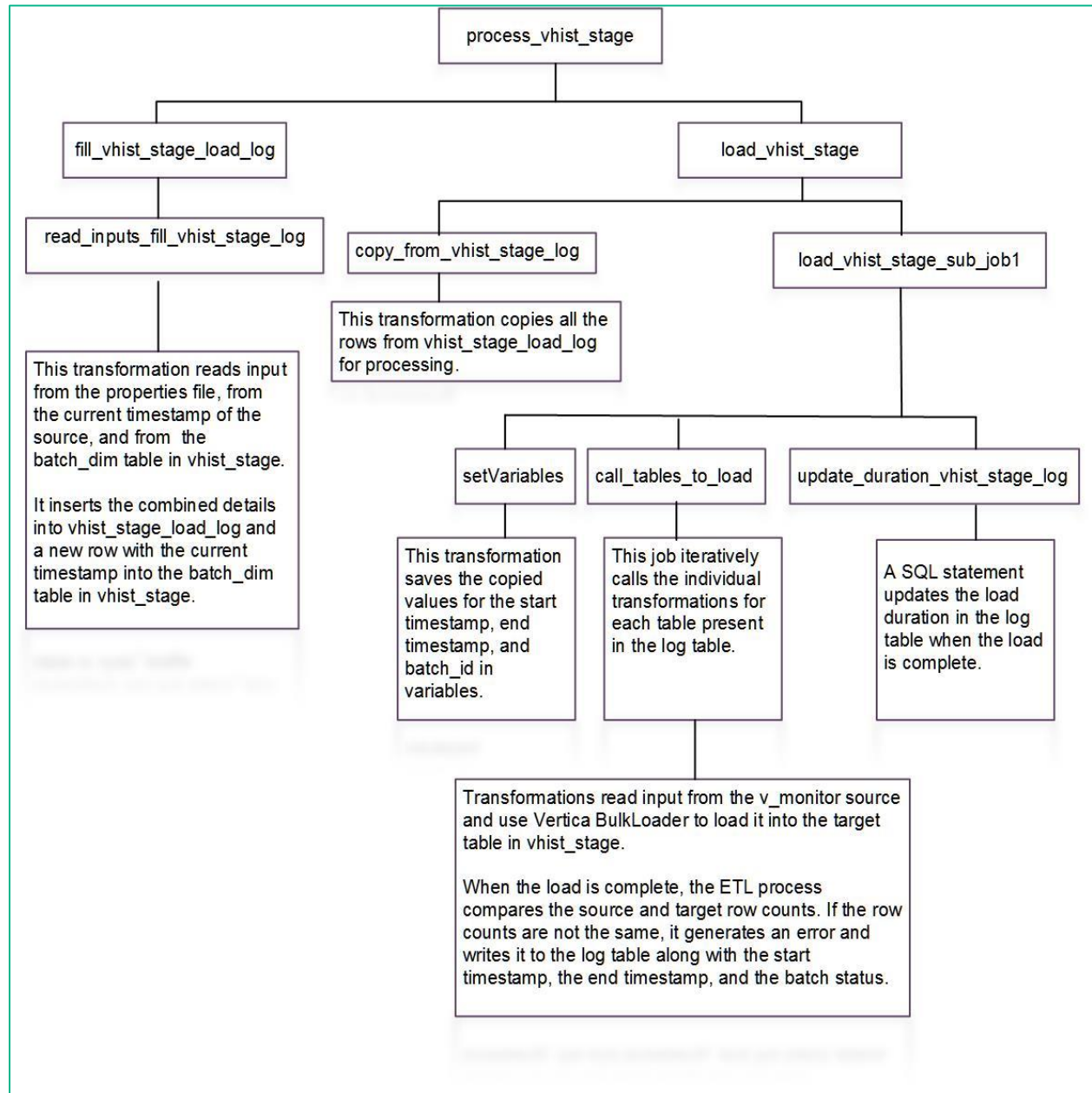
VHist ETL Steps

Populating VHist is a two-step process:

1. Source to stage—The source tables are loaded into staging tables. Each staging table is a replica of its source table, with one difference: the addition of BATCH_ID, an integer column that holds the identity of the ETL batch load. (See [Source-to-Stage ETL Process Flow](#).)
2. Stage to star—The staging tables are transformed into a star schema. (See [Stage-to-Star ETL Process Flow](#).)

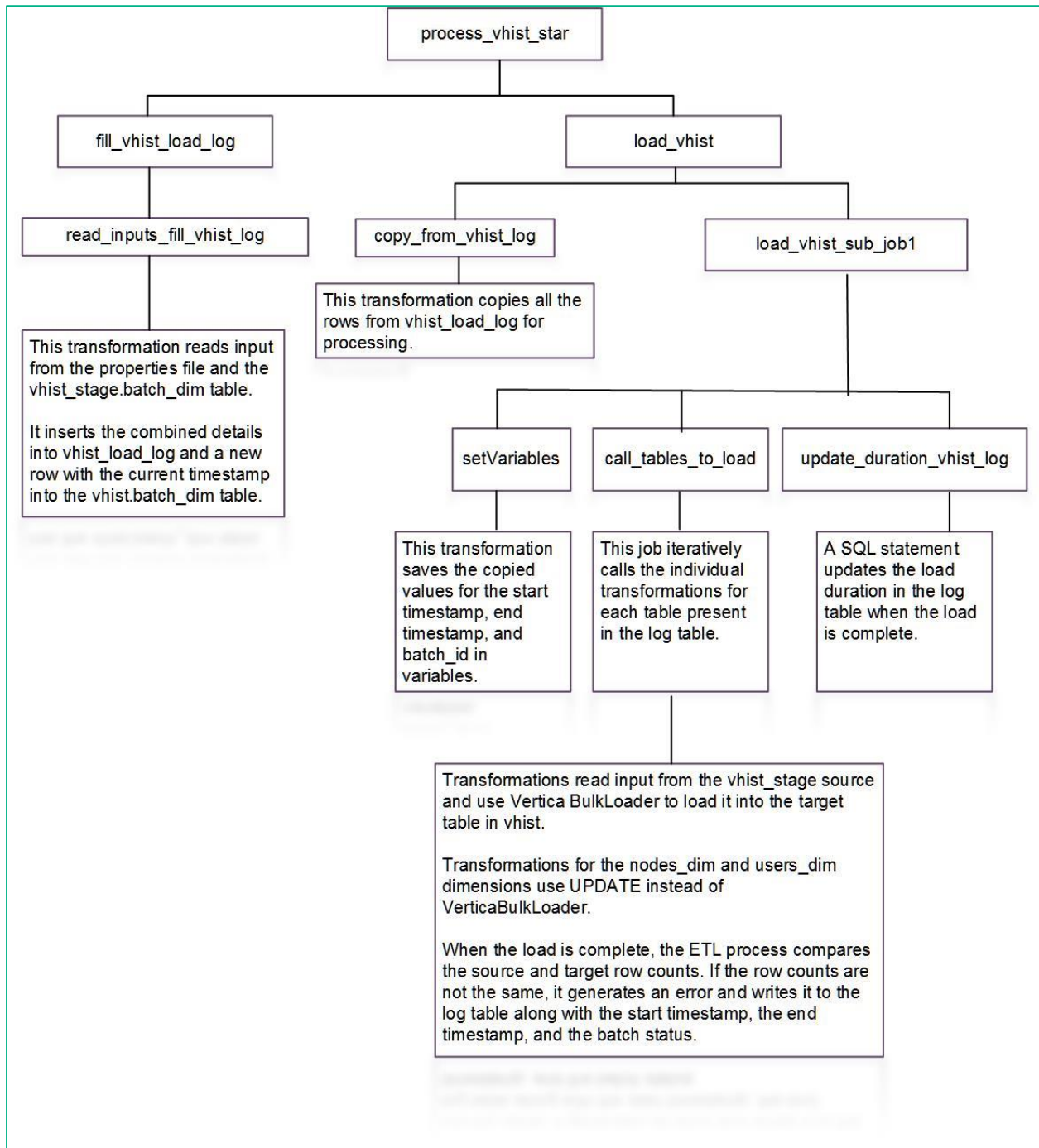
Source-to- Stage ETL Process Flow

This diagram summarizes the ETL process flow for creating vhist_stage from Vertica system tables.



Stage-to-Star ETL Process Flow

This diagram summarizes the ETL process flow for creating the VHist star schema from the vhist_stage staging tables.



About VHist Incremental Loads

Incremental loads cause the VHist data warehouse to accumulate system data over time, creating a rich store of data for analysis. However, you should take care when repeatedly refreshing the data warehouse, especially when scheduling the ETL job to run at regular intervals.

Keep the following in mind:

- The size of the VHist data warehouse could potentially grow very large with repeated loads, and the size of VHist counts towards the size limit specified in your Vertica license.

NOTE: The Vertica Community Edition allows up to one terabyte of free storage. If you already have a licensed version of Vertica, you could build the VHist warehouse using the Community Edition on a separate cluster.

- There could be gaps in some of the data loaded into VHist, because the source system tables are not flushed uniformly. Vertica system tables are actually views of underlying tables that are flushed at different intervals.

NOTE: The VHist ETL process generates log tables that capture detailed information about the timing of each load and the amount of data loaded into each target table. To avoid running incremental loads more often than is necessary, try starting with daily loads and review the results. If there are gaps in the results, decrease the interval between loads until you find an optimal balance. The VHist log tables are listed as follows:

```
vhist_stage.vhist_stage_load_log
vhist.vhist_load_log
```

VHist Fact Tables

Overview VHist Fact Tables

V_MONITOR Source	VHist Target	ETL Based On:
USER_SESSIONS	USER_SESSIONS_FACT	Timestamp
QUERY_PROFILES	QUERY_PROFILES_FACT	Timestamp
LOAD_STREAMS	LOAD_STREAMS_FACT	Timestamp
RESOURCE_REJECTIONS	RESOURCE_REJECTIONS_FACT	Timestamp
DISK_STORAGE	DISK_STORAGE_FACT	Polling
PROJECTION_STORAGE	PROJECTION_STORAGE_FACT	Polling
PROJECTION_USAGE	PROJECTION_USAGE_FACT	Timestamp
SYSTEM_RESOURCE_USAGE	SYSTEM_RESOURCE_USAGE_FACT	Timestamp

For details about V_MONITOR system tables, see [V_MONITOR Schema](#) in the *Vertica SQL Reference Manual*.

USER_SESSIONS_FACT

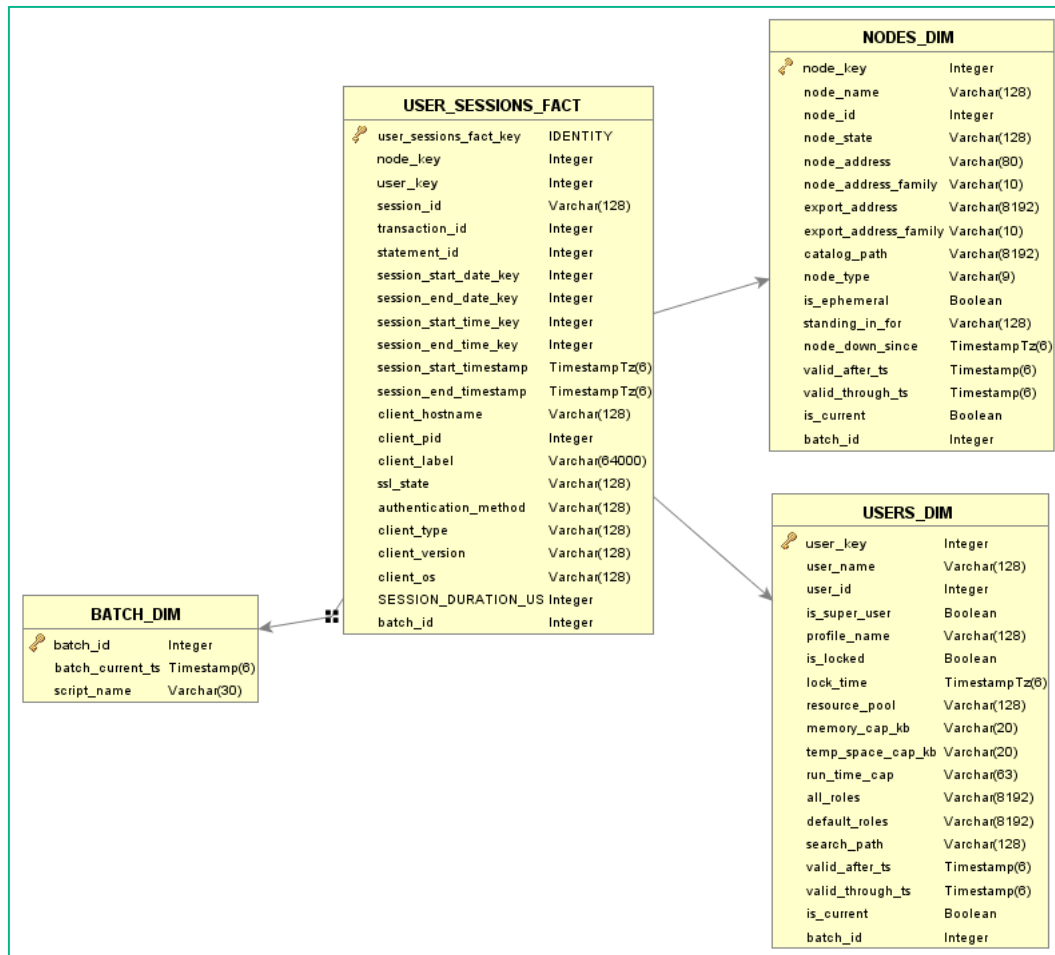
USER_SESSIONS_FACT contains data about user sessions. Each row describes a completed user session.

ETL for USER_SESSIONS_FACT is based on the value of SESSION_END_TIMESTAMP in the source and includes only the rows for sessions that are no longer active.

Source Table

The source table for V HIST.USER_SESSIONS_FACT is V_MONITOR.USER_SESSIONS. For details about the source table, see [USER_SESSIONS](#) in the *Vertica SQL Reference Manual*.

USER_SESSIONS_FACT Overview



USER_SESSIONS_FACT Columns

Most columns come from a subset of the columns in V_MONITOR.USER_SESSIONS.

The primary keys, USER_SESSIONS_FACT_KEY and SESSION_ID, are not guaranteed unique.

Column Name	Data Type	Description	Use in VHist
USER_SESSIONS_FACT_KEY	INTEGER	Identifier for this session. This identifier is unique within the cluster at any point in time but can be reused when the session closes.	primary key
NODE_KEY	INTEGER	A unique numeric surrogate key assigned by Vertica sequence. Identifies the node for a particular date range.	dimension
USER_KEY	INTEGER	A unique numeric surrogate key assigned by Vertica sequence. Identifies the user for a particular date range.	dimension
SESSION_ID	VARCHAR	Identifier for this session. This identifier is unique within the cluster at any point in time but can be reused when the session closes.	primary key
TRANSACTION_ID	VARCHAR	Identifier for the transaction within the session, if any. If a session is active but no transaction has begun, TRANSACTION_ID returns NULL.	degenerate dimension
STATEMENT_ID	VARCHAR	Unique numeric ID for the currently-running statement. NULL indicates that no statement is currently being processed. The combination of TRANSACTION_ID and STATEMENT_ID uniquely identifies a statement within a session.	degenerate dimension
SESSION_START_DATE_KEY	INTEGER	Date at beginning of history interval.	dimension
SESSION_END_DATE_KEY	INTEGER	Date of session at end of history interval.	dimension
SESSION_START_DATE	DATE	Date at beginning of history interval.	dimension
SESSION_END_DATE	DATE	Date of session at end of history interval.	dimension
SESSION_START_TIME_OF_DAY	TIME(0)	Time of session at beginning of history interval.	dimension
SESSION_END_TIME_OF_DAY	TIME(0)	Time of session at end of history interval.	dimension
SESSION_START_TIMESTAMP	TIMESTAMPZ	Value of session at beginning of history interval.	degenerate dimension
SESSION_END_TIMESTAMP	TIMESTAMPZ	Value of session at end of history interval.	degenerate dimension
CLIENT_HOSTNAME	VARCHAR	IP address of the client system.	degenerate dimension
CLIENT_PID	INTEGER	Linux process identifier of the client process that issued this connection. Note: The client process could be on a different machine from the server.	degenerate dimension
CLIENT_LABEL	VARCHAR	User-specified label for the client connection that can be set when using ODBC. See "Label/SessionLabel" in <i>Data Source Name (DSN) Connection Parameters</i> in the Vertica documentation.	degenerate dimension
SSL_STATE	VARCHAR	Indicates if Vertica used Secure Socket Layer (SSL) for a particular session. Possible values are:	degenerate dimension

		<p>None – Vertica did not use SSL.</p> <p>Server – The client authenticates the server.</p> <p>Mutual – The server and the client authenticated each other.</p> <p>See SSL Authentication in the Vertica documentation.</p>	
AUTHENTICATION_METHOD	VARCHAR	<p>Type of client authentication used for a particular session, if known. Possible values are:</p> <p>Unknown</p> <p>Trust</p> <p>Reject</p> <p>Kerberos</p> <p>Password</p> <p>MD5</p> <p>LDAP</p> <p>Kerberos-GSS</p> <p>Ident</p> <p>See Security and Authentication and Implementing Client Authentication in the Vertica documentation.</p>	degenerate dimension
CLIENT_TYPE	VARCHAR	<p>The type of client from which the connection was made. Possible client type values:</p> <p>ADO.NET Drive</p> <p>ODBC Driver</p> <p>JDBC Driver</p> <p>sql</p>	degenerate dimension
CLIENT_VERSION	VARCHAR	Returns the client version.	degenerate dimension
CLIENT_OS	VARCHAR	Returns the client operating system.	degenerate dimension
SESSION_DURATION_US	NUMERIC(18,0)	The duration of the session in microseconds.	measure
BATCH_ID	INTEGER	Identifies the ETL batch load.	dimension

QUERY_PROFILES_FACT

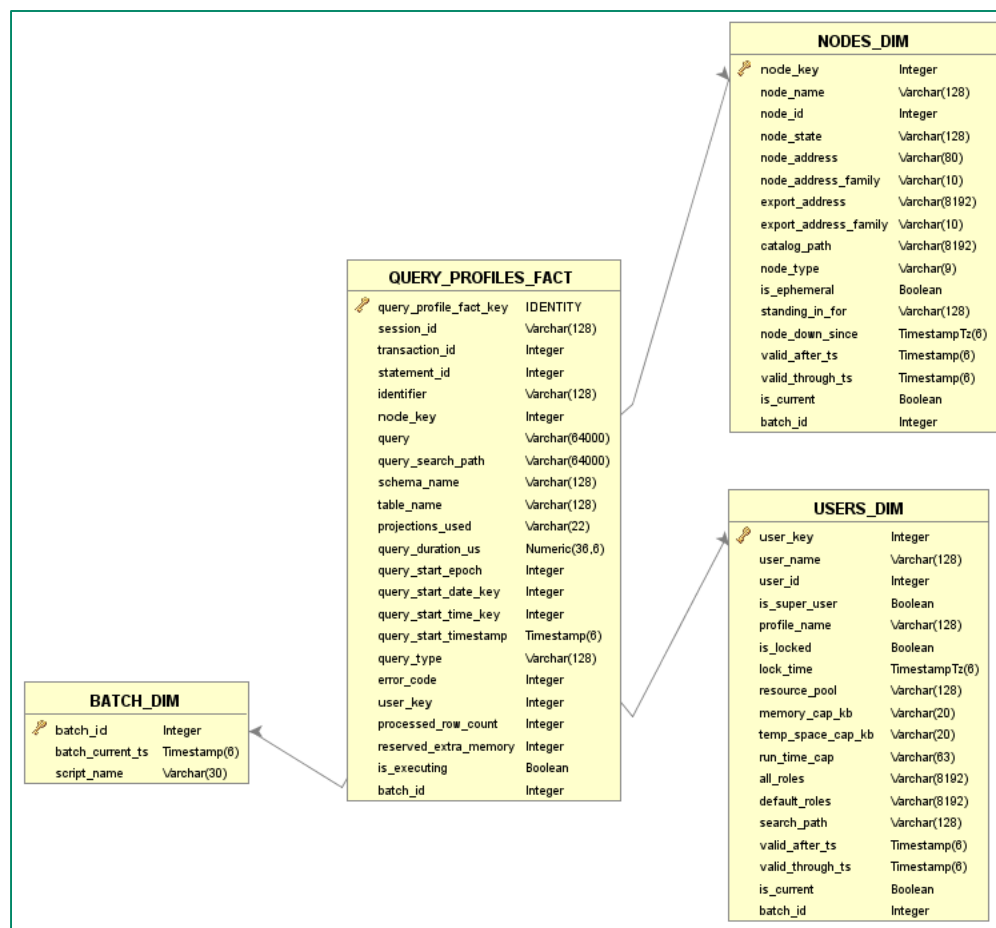
QUERY_PROFILES_FACT contains data about queries. Each row describes a completed query.

ETL for QUERY_PROFILES_FACT is based on the value of QUERY_START in the source and includes only the rows for queries that are no longer executing.

Source Table

The source table for VHist.QUERY_PROFILES_FACT is V_MONITOR. QUERY_PROFILES. For details about the source table, see [QUERY_PROFILES](#) in the *Vertica SQL Reference Manual*.

QUERY_PROFILES_FACT OVERVIEW



QUERY_PROFILES_FACT Columns

Most columns come from a subset of the columns in V_MONITOR.QUERY_PROFILES.

The primary key, QUERY_PROFILE_FACT_KEY, is driven by a Vertica sequence defined at the database level.

Column Name	Data Type	Description	Use in VHist
QUERY_PROFILE_FACT_KEY	INTEGER	The identification key of the table. It is unique for the table and generated by a sequence.	primary key
SESSION_ID	VARCHAR	The identification of the session for which profiling information is captured. This identifier is unique within the cluster at any point in time but can be reused when the session closes.	degenerate dimension
TRANSACTION_ID	INTEGER	An identifier for the transaction within the session if any; otherwise NULL.	degenerate dimension
STATEMENT_ID	INTEGER	Unique numeric ID for the currently-running statement. NULL indicates that no statement is currently being processed. The combination of TRANSACTION_ID, STATEMENT_ID uniquely identifies a statement within a session.	degenerate dimension
QUERY_ID	VARCHAR	A string to identify the query in system tables.	degenerate dimension
NODE_KEY	INTEGER	A unique numeric surrogate key, identifies the node for a particular date range Assigned by Vertica sequence.	dimension
QUERY	VARCHAR	The query string used for the query.	degenerate dimension
QUERY_SEARCH_PATH	VARCHAR	A list of schemas in which to look for tables.	degenerate dimension
SCHEMA_NAME	VARCHAR	The schema name in which the query is being profiled.	degenerate dimension
TABLE_NAME	VARCHAR	The table name in the query being profiled.	degenerate dimension
PROJECTIONS_USED	VARCHAR	The projections used in the query.	degenerate dimension
QUERY_DURATION_US	NUMERIC(18,0)	The duration of the query in microseconds.	measure
QUERY_START_EPOCH	INTEGER	The epoch number at the start of the given query.	degenerate dimension
QUERY_START_DATE_KEY	INTEGER	Date of query execution.	dimension
QUERY_START_DATE	DATE	Date of query execution.	dimension
QUERY_START_TIME_OF_DAY	TIME(0)	Time (of day) of query execution.	dimension
QUERY_START_TIMESTAMP	TIMESTAMP	The Linux system time stamp of query execution in a format that can be used as a DATE/TIME expression.	degenerate dimension
QUERY_TYPE	VARCHAR	One of these values: INSERT SELECT UPDAT DELETE UTILITY UNKNOWN	degenerate dimension

ERROR_CODE	INTEGER	The return error code for the query.	degenerate dimension
USER_KEY	INTEGER	A unique numeric surrogate key assigned by Vertica sequence. Identifies the user for a particular date range.	dimension
PROCESSED_ROW_COUNT	INTEGER	The number of rows returned by the query.	measure
RESERVED_EXTRA_MEMORY	INTEGER	The amount of extra memory, in bytes, reserved for the query. Extra memory is the amount of memory reserved for the plan but not assigned to a particular operator. This is the memory from which unbounded operators pull first. If operators acquire all of the extra memory, then the plan must go back to the Resource Manager for more memory.	measure
BATCH_ID	INTEGER	Identifies the ETL batch load.	dimension

LOAD_STREAMS_FACT

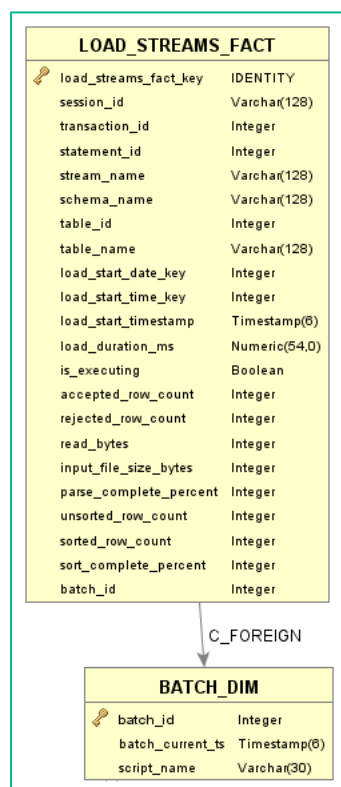
LOAD_STREAMS_FACT contains data about load streams on each node. Each row describes a completed load.

ETL for LOAD_STREAMS_FACT is based on the value of LOAD_START in the source and includes the rows for completed loads only.

Source Table

The source table for VHist.LOAD_STREAMS_FACT is V_MONITOR.LOAD_STREAMS. For details about the source table, see [LOAD_STREAMS](#) in the Vertica SQL Reference Manual.

LOAD_STREAMS_FACT Overview



LOAD_STREAMS_FACT Columns

Most columns come from a subset of the columns in V_MONITOR.LOAD_STREAMS.

The primary key, `LOAD_STREAMS_FACT_KEY`, is driven by a Vertica sequence defined at the database level.

Column Name	Data Type	Description	Use in VHist
<code>LOAD_STREAMS_FACT_KEY</code>	INTEGER	The identification key of the table. It is unique for the table and generated by a sequence.	primary key
<code>SESSION_ID</code>	VARCHAR	Identifier of the session for which HP Vertica captures load stream information. This identifier is unique within the cluster for the current session, but can be reused in a subsequent session.	degenerate dimension
<code>TRANSACTION_ID</code>	INTEGER	Identifier for the transaction within a session. If a session is active but no transaction has begun, this is NULL.	degenerate dimension
<code>STATEMENT_ID</code>	INTEGER	Unique numeric ID for the currently-running statement. NULL indicates that no statement is currently being processed. The combination of <code>TRANSACTION_ID</code> , <code>STATEMENT_ID</code> uniquely identifies a statement within a session.	degenerate dimension
<code>STREAM_NAME</code>	VARCHAR	Load stream identifier. If the user does not supply a specific name, the <code>STREAM_NAME</code> default value is <code>tablenameID</code> , where <code>tablename</code> is the table into which data is being loaded, and <code>ID</code> is an integer value guaranteed to be unique within the current session on a node. The <code>LOAD_STREAMS</code> system table includes stream names for every COPY statement that takes more than 1-second to run. The 1-second duration includes the time to plan and execute the statement.	degenerate dimension
<code>SCHEMA_NAME</code>	VARCHAR	Schema name for which load stream information is listed. Lets you identify two streams that are targeted at tables with the same name in different schemas.	degenerate dimension
<code>TABLE_ID</code>	INTEGER	A unique numeric ID assigned by the Vertica catalog that identifies the table.	degenerate dimension
<code>TABLE_NAME</code>	VARCHAR	Name of the table being loaded.	degenerate dimension
<code>LOAD_START_DATE_KEY</code>	INTEGER	Linux system time when the load started.	dimension
<code>LOAD_START_DATE</code>	DATE	Linux system time when the load started.	dimension
<code>LOAD_START_TIME_OF_DAY</code>	TIME(0)	Linux system time when the load started.	dimension
<code>LOAD_START_TIMESTAMP</code>	VARCHAR	Linux system time when the load started.	degenerate dimension
<code>LOAD_DURATION_MS</code>	NUMERIC(54,0)	Duration of the load stream in milliseconds.	measure
<code>ACCEPTED_ROW_COUNT</code>	INTEGER	Number of rows loaded.	measure
<code>REJECTED_ROW_COUNT</code>	INTEGER	Number of rows rejected.	measure
<code>READ_BYTES</code>	INTEGER	Number of bytes read from the input file.	measure
<code>INPUT_FILE_SIZE_BYTES</code>	INTEGER	Size of the input file in bytes. Note: When using STDIN as input, the input file size is zero (0).	measure
<code>PARSE_COMPLETE_PERCENT</code>	INTEGER	Percent of rows from the input file that have been loaded.	measure

UNSORTED_ROW_COUNT	INTEGER	Cumulative number rows not sorted across all projections. Note: UNSORTED_ROW_COUNT could be greater than ACCEPTED_ROW_COUNT because data is copied and sorted for every projection in the target table.	measure
SORTED_ROW_COUNT	INTEGER	Cumulative number of rows sorted across all projections.	measure
SORT_COMPLETE_PERCENT	INTEGER	Percent of rows from the input file that have been sorted.	measure
BATCH_ID	INTEGER	Identifies the ETL batch load.	dimension

DISK_STORAGE_FACT

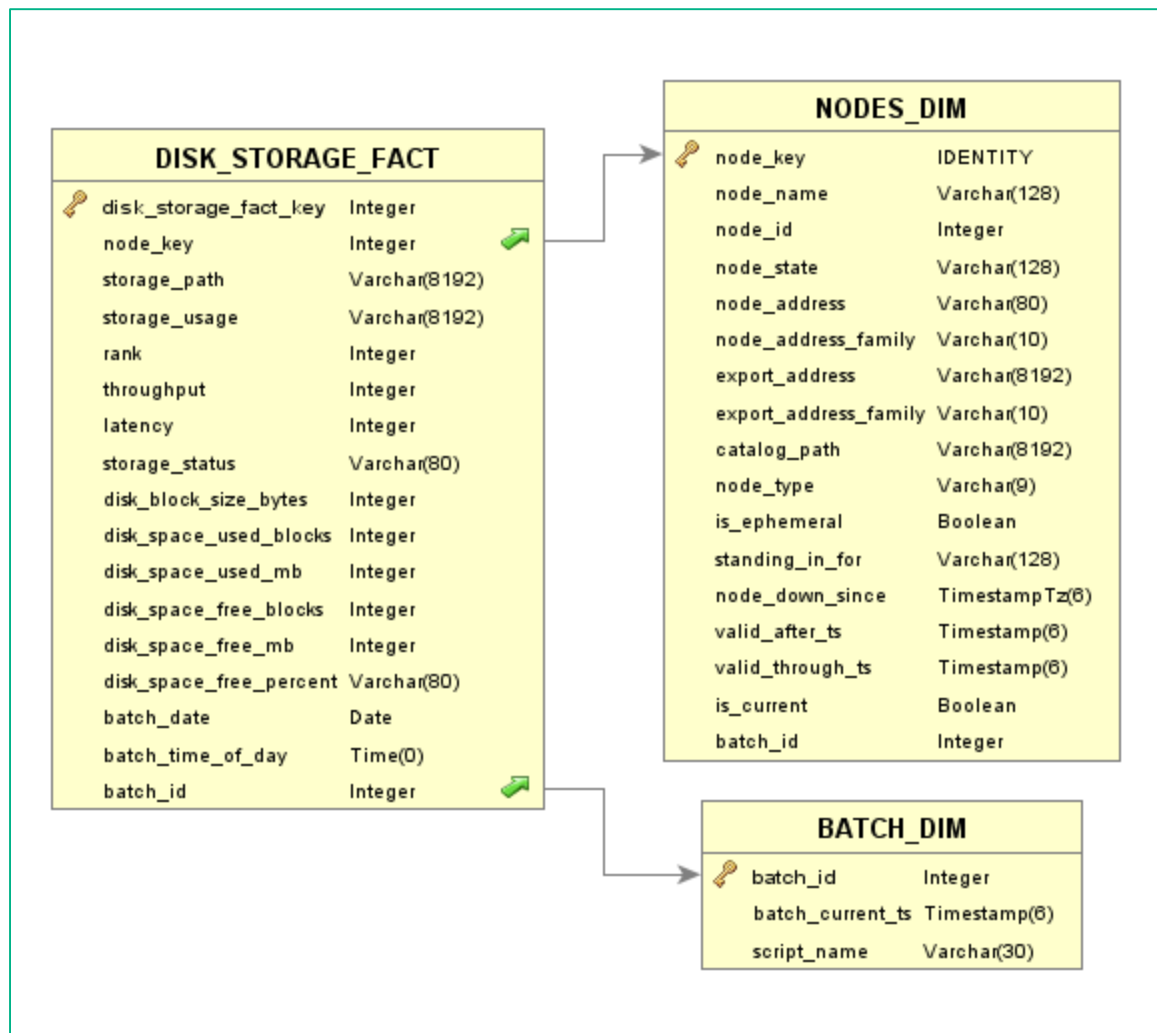
DISK_STORAGE_FACT contains data about disk storage used by the database on each node. Each row contains a snapshot of disk storage at a point in time equal to the timestamp of the batch load job.

ETL for DISK_STORAGE_FACT is based on polling of the source table at a user-defined frequency. The batch timestamp is parsed into a date and a time and inserted into the fact tables for easy reporting.

Source Table

The source table for VHist.DISK_STORAGE_FACT is V_MONITOR.DISK_STORAGE. For details about the source table, see [DISK_STORAGE](#) in the *Vertica SQL Reference Manual*.

DISK_STORAGE_FACT Overview



DISK_STORAGE_FACT Columns

Most columns come from a subset of the columns in V_MONITOR.DISK_STORAGE.

The primary key, DISK_STORAGE_FACT_KEY, is driven by a Vertica sequence defined at the database level.

Column Name	Data Type	Description	Use in VHist
DISK_STORAGE_FACT_KEY	INTEGER	The identification key of the table, unique for the table and generated by a sequence.	primary key
NODE_KEY	INTEGER	A unique numeric surrogate key, assigned by a Vertica sequence that identifies the node for a particular date range.	dimension
STORAGE_PATH	VARCHAR	The path where the storage location is mounted.	degenerate dimension
STORAGE_USAGE	VARCHAR	The type of information stored in the location. Values are: DATA TEMP DATA, TEMP USER	degenerate dimension
RANK	INTEGER	The rank assigned to the storage location based on its performance. See Managing Storage Locations in the Vertica documentation.	measure
THROUGHPUT	INTEGER	The measure of a storage location's performance in MB/sec. 1/throughput is the time taken to read 1MB of data.	measure
LATENCY	INTEGER	The measure of a storage location's performance in seeks per second. 1/latency is the time taken to seek to the data.	measure
STORAGE_STATUS	VARCHAR	The status of the storage location: ACTIVE or RETIRED	degenerate dimension
DISK_BLOCK_SIZE_BYTES	INTEGER	The block size of the disk in bytes.	measure
DISK_SPACE_USED_BLOCKS	INTEGER	The number of disk blocks in use.	measure
DISK_SPACE_USED_MB	INTEGER	The number of megabytes of disk storage in use.	measure
DISK_SPACE_FREE_BLOCKS	INTEGER	The number of free disk blocks available.	measure
DISK_SPACE_FREE_MB	INTEGER	The number of megabytes of free storage available.	measure
DISK_SPACE_FREE_PERCENT	INTEGER	The percentage of free disk space remaining.	measure
BATCH_DATE	DATE	The date of the batch	dimension
BATCH_TIME_OF_DAY	TIME(O)	The time of the batch.	dimension
BATCH_ID	INTEGER	Identifies the ETL batch load.	dimension

PROJECTION_STORAGE_FACT

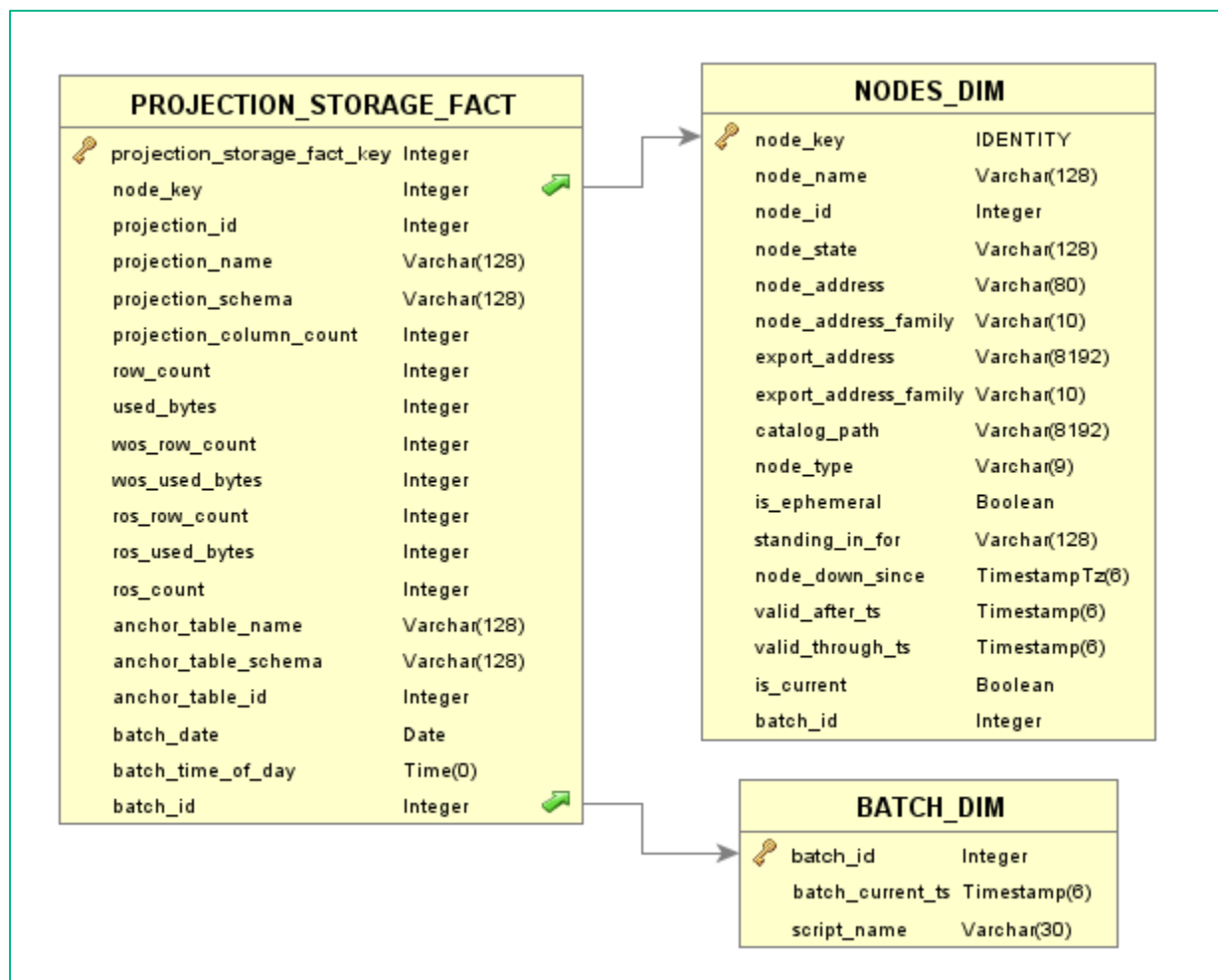
PROJECTION_STORAGE_FACT contains data about disk storage used by each projection on each node. Each row is a snapshot of projection storage at a point in time equal to the timestamp of the batch load job.

ETL for PROJECTION_STORAGE_FACT is based on polling of the source table at a user-defined frequency. The batch timestamp is parsed into a date and a time and inserted into the fact tables for easy reporting.

Source Table

The source table for VHist.PROJECTION_STORAGE_FACT is V_MONITOR.PROJECTION_STORAGE. For details about the source table, see [PROJECTION_STORAGE](#) in the *Vertica SQL Reference Manual*.

PROJECTION_STORAGE_FACT Overview



PROJECTION_STORAGE_FACT Columns

Most columns in PROJECTION_STORAGE_FACT come from a subset of the columns in the source table.

The primary key, PROJECTION_STORAGE_FACT_KEY, is driven by a Vertica sequence defined at the database level.

Column Name	Data Type	Description	Use in VHist
PROJECTION_STORAGE_FACT_KEY	INTEGER	The identification key of the table. Unique for the table and generated by a sequence.	primary key
BATCH_ID	INTEGER	ETL load batch ID. Use this to get start and end times for load and point-in-time.	dimension
NODE_KEY	INTEGER	A unique numeric surrogate key assigned by Vertica sequence. Identifies the node for a particular date range.	dimension
PROJECTION_ID	VARCHAR	A unique numeric ID assigned by the Vertica catalog. Identifies the projection.	dimension
PROJECTION_NAME	VARCHAR	The name of the projection for which information is listed.	dimension
PROJECTION_SCHEMA	VARCHAR	The name of the schema associated with the projection.	degenerate dimension
PROJECTION_COLUMN_COUNT	INTEGER	The number of columns in the projection.	measure
ROW_COUNT	INTEGER	The number of rows in the table's projections, including any rows marked for deletion.	measure
USED_BYTES	INTEGER	The number of bytes of disk storage used by the projection.	measure
WOS_ROW_COUNT	INTEGER	The number of WOS rows in the projection.	measure
WOS_USED_BYTES	INTEGER	The number of WOS bytes in the projection.	measure
ROS_ROW_COUNT	INTEGER	The number of ROS rows in the projection.	measure
ROS_USED_BYTES	INTEGER	The number of ROS bytes in the projection.	measure
ROS_COUNT	INTEGER	The number of ROS containers in the projection.	measure
ANCHOR_TABLE_NAME	VARCHAR	The associated table name for which information is listed.	dimension
ANCHOR_TABLE_SCHEMA	VARCHAR	The associated table schema for which information is listed.	dimension
ANCHOR_TABLE_ID	INTEGER	Identifier of the anchor table.	degenerate dimension
BATCH_DATE	DATE	The date of the batch	dimension
BATCH_TIME_OF_DAY	TIME(O)	The time of the batch.	dimension
BATCH_ID	INTEGER	Identifies the ETL batch load.	dimension

RESOURCE_REJECTIONS_FACT

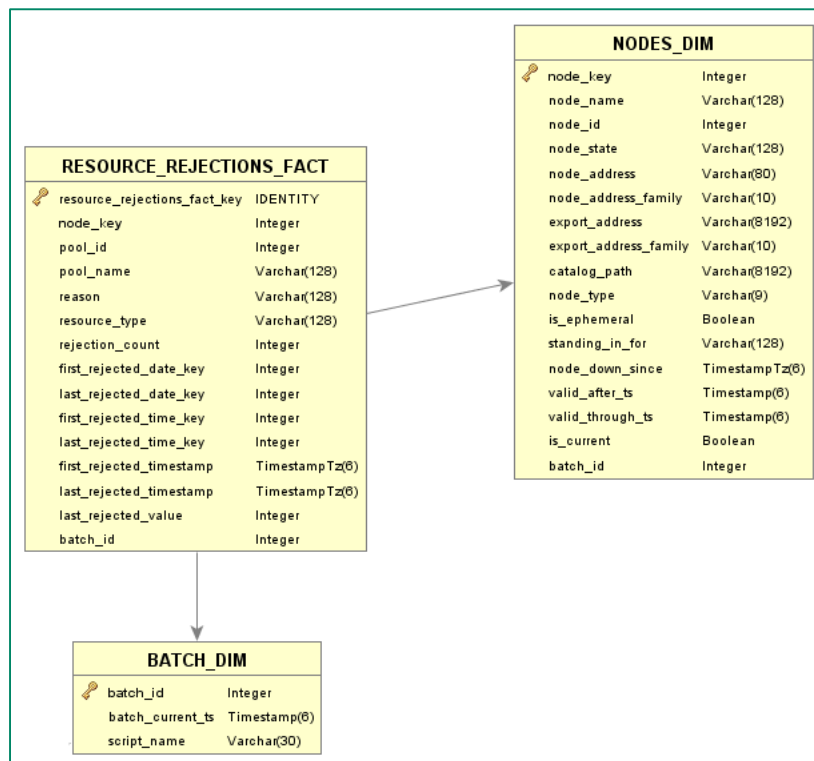
RESOURCE_REJECTIONS_FACT contains data about resource requests that have been rejected by the Resource Manager. Each row describes a rejected request for resources.

ETL for RESOURCE_REJECTIONS_FACT is based on the value of LAST_REJECTED_TIMESTAMP in the source and includes only the rows for rejections that are no longer executing.

Source Table

The source table for VHIST.RESOURCE_REJECTIONS_FACT is V_MONITOR.RESOURCE_REJECTIONS. For details about the source table, see [RESOURCE_REJECTIONS](#) in the *Vertica SQL Reference Manual*.

RESOURCE_REJECTIONS_FACT Overview



RESOURCE_REJECTIONS_FACT Columns

Most columns come from a subset of the columns in V_MONITOR.RESOURCE_REJECTIONS.

The primary key, RESOURCE_REJECTIONS_FACT_KEY, is driven by a Vertica sequence defined at the database level.

Column Name	Data Type	Description	Use in VHist
RESOURCE_REJECTIONS_FACT_KEY	INTEGER	The identification key of the table. Unique for the table and generated by a sequence.	primary key
NODE_KEY	INTEGER	A unique numeric surrogate key assigned by Vertica sequence. Identifies the node for a particular date range.	dimension
POOL_ID	INTEGER	A unique numeric ID, assigned by the Vertica catalog. Identifies the resource pool.	degenerate dimension
POOL_NAME	VARCHAR	The name of the resource pool.	degenerate dimension
REASON	VARCHAR	The reason for rejecting this request.	degenerate dimension
RESOURCE_TYPE	VARCHAR	Memory, threads, file handles or execution slots.	degenerate dimension D
REJECTION_COUNT	INTEGER	Number of requests rejected due to specified reason and RESOURCE_TYPE.	measure
FIRST_REJECTED_DATE_KEY	INTEGER	The time of the first rejection.	dimension
LAST_REJECTED_DATE_KEY	INTEGER	The time of the last rejection for this pool.	dimension
FIRST_REJECTED_DATE	DATE	The time of the first rejection.	degenerate dimension
LAST_REJECTED_DATE	DATE	The time of the last rejection for this pool.	degenerate dimension
FIRST_REJECTED_TIME_OF_DAY	TIME(0)	The time of the first rejection.	degenerate dimension
LAST_REJECTED_TIME_OF_DAY	TIME(0)	The time of the last rejection for this pool.	degenerate dimension
FIRST_REJECTED_TIMESTAMP	TIMESTAMPTZ	The time of the first rejection.	degenerate dimension
LAST_REJECTED_TIMESTAMP	TIMESTAMPTZ	The time of the last rejection for this pool.	degenerate dimension
LAST_REJECTED_VALUE	INTEGER	The amount of the specific resource requested by the last rejection.	MEASURE
BATCH_ID	INTEGER	Identifies the ETL batch load.	dimension

PROJECTION_USAGE_FACT

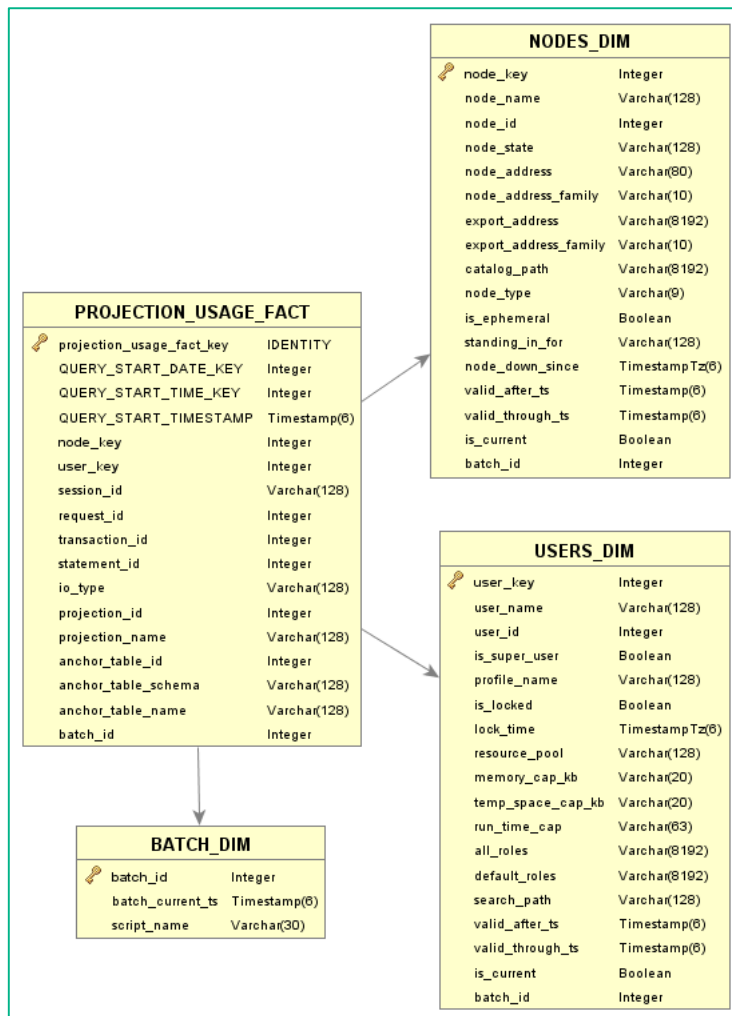
PROJECTION_USAGE_FACT contains data about projections that Vertica uses in processing queries. Each row describes the projection usage for a query.

ETL for PROJECTION_USAGE_FACT is based on the value of QUERY_START_TIMESTAMP in the source and includes only the rows for queries that are no longer executing.

Source Table

The source table for VHist.PROJECTION_USAGE_FACT is V_MONITOR.PROJECTION_USAGE. For details about the source table, see [PROJECTION_USAGE](#) in the *Vertica SQL Reference Manual*.

PROJECTION_USAGE_FACT Overview



PROJECTION_USAGE_FACT Columns

Most columns come from a subset of the columns in V_MONITOR.PROJECTION_USAGE.

The primary key is driven by a Vertica sequence defined at the database level.

Column Name	Data Type	Description	Use in VHist
PROJECTION_USAGE_FACT_KEY	INTEGER	The identification key of the table. Unique for the table and generated by a sequence.	primary key
QUERY_START_DATE_KEY	INTEGER	Value of query at beginning of history interval.	dimension
QUERY_START_DATE	DATE	Value of query at beginning of history interval.	degenerate dimension
QUERY_START_TIME_OF_DAY	TIME(O)	Value of query at beginning of history interval.	degenerate dimension
QUERY_START_TIMESTAMP	TIMESTAMP	Value of query at beginning of history interval.	degenerate dimension
NODE_KEY	INTEGER	A unique numeric surrogate key assigned by Vertica sequence. Identifies the node for a particular date range.	dimension
USER_KEY	INTEGER	A unique numeric surrogate key assigned by Vertica sequence. Identifies the user for a particular date range.	dimension
SESSION_ID	VARCHAR	Identifier for this session. This identifier is unique within the cluster at any point in time but can be reused when the session closes.	degenerate dimension
REQUEST_ID	INTEGER	Unique identifier of the query request in the user session.	degenerate dimension
TRANSACTION_ID	INTEGER	Identifier for the transaction within the session, if any; otherwise NULL.	degenerate dimension
STATEMENT_ID	INTEGER	Unique numeric ID for the currently-running statement. NULL indicates that no statement is currently being processed. The combination of TRANSACTION_ID, STATEMENT_ID, and REQUEST_ID uniquely identifies a statement within a session.	degenerate dimension
IO_TYPE	VARCHAR	Input/output.	measure
PROJECTION_ID	INTEGER	Unique numeric ID assigned by the Vertica catalog. Identifies the projection.	degenerate dimension
PROJECTION_NAME	VARCHAR	Projection name for which information is listed.	degenerate dimension
ANCHOR_TABLE_ID	INTEGER	Unique numeric ID assigned by the Vertica catalog. Identifies the anchor table.	degenerate dimension
ANCHOR_TABLE_SCHEMA	VARCHAR	Name of the schema that contains the anchor table.	degenerate dimension
ANCHOR_TABLE_NAME	VARCHAR	Name of the projection's associated anchor table.	degenerate dimension
BATCH_ID	INTEGER	Identifies the ETL batch load.	dimension

SYSTEM_RESOURCE_USAGE_FACT

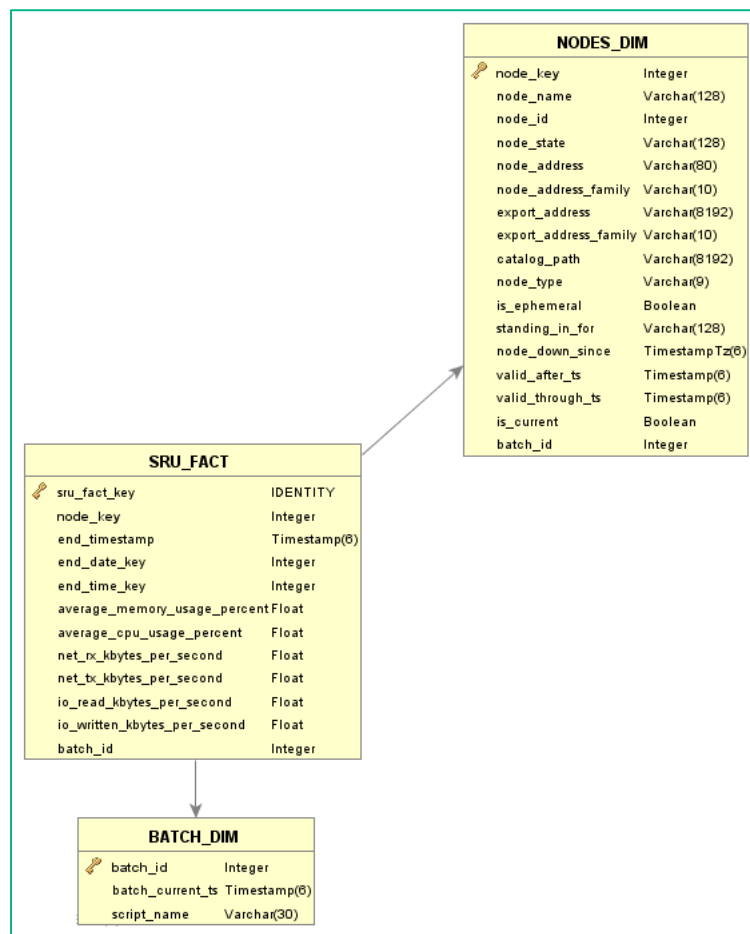
SYSTEM_RESOURCE_USAGE_FACT provides statistics about system resources. Each row contains statistics about memory, CPU, I/O, and disk activity for a given time interval.

ETL for SYSTEM_RESOURCE_USAGE_FACT is based on the value of END_TIME in the source and includes only the rows for statistics that have completed computing.

Source Table

The source table for VHist.SYSTEM_RESOURCE_USAGE_FACT is V_MONITOR.SYSTEM_RESOURCE_USAGE. For details about the source table, see [SYSTEM_RESOURCE_USAGE](#) in the *Vertica SQL Reference Manual*.

SYSTEM_RESOURCE_USAGE_FACT Overview



SYSTEM_RESOURCE_USAGE_FACT Columns

Most columns come from a subset of the columns in V_MONITOR.SYSTEM_RESOURCE_USAGE.

The primary key is driven by a Vertica sequence defined at the database level.

Column Name	Data Type	Description	Use in VHist
SRU_FACT_KEY	INTEGER	The identification key of the table. Unique for the table and generated by a sequence.	primary key
NODE_KEY	INTEGER	A unique numeric surrogate key assigned by Vertica sequence. Identifies the node for a particular date range.	dimension
END_DATE_KEY	INTEGER	End date of the history interval.	dimension
END_DATE	DATE	End date of the history interval.	degenerate dimension
END_TIME_OF_DAY	TIME(0)	End time-of-day of the history interval.	dimension
END_TIMESTAMP	TIMESTAMP	End time stamp of the history interval.	degenerate dimension
AVERAGE_MEMORY_USAGE_PERCENT	FLOAT	Average memory usage in percent of total memory (0-100) during the history interval.	measure
AVERAGE_CPU_USAGE_PERCENT	FLOAT	Average CPU usage in percent of total CPU time (0-100) during the history interval.	measure
NET_RX_KBYTES_PER_SECOND	FLOAT	Average number of kilobytes received from network (incoming) per second during the history interval.	measure
NET_TX_KBYTES_PER_SECOND	FLOAT	Average number of kilobytes transmitting to network (outgoing) per second during the history interval.	measure
IO_READ_KBYTES_PER_SECOND	FLOAT	Disk I/O average number of kilobytes read from disk per second during the history interval.	measure
IO_WRITTEN_KBYTES_PER_SECOND	FLOAT	Average number of kilobytes written to disk per second during the history interval.	measure
BATCH_ID	INTEGER	Identifies the ETL batch load.	dimension

VHist Dimension Tables

Overview VHist Dimension Tables

Dimension	Description	Created by:
DATE_DIM	Contains date data	SQL script
TIME_DIM	Contains timestamp data	SQL script
BATCH_DIM	Contains information about the ETL batch jobs	SQL Script
NODE_DIM	Contains data loaded from V_CATALOG.NODES	ETL
USER_DIM	Contains data loaded from V_CATALOG.USERS	ETL

DATE_DIM

DATE_DIM is a system-generated date dimension.

DATE_DIM Dimension Table Columns

Column Name	Data Type	Description
DATE_KEY	INTEGER	Primary key. Not Null.
DATE	DATE	Date in standard date format from ISO 8601. Looks like this: ""19991231"".
WEEKDAY_NUM	INTEGER	Ordinal number for the day or the week, where Monday is 1 and Sunday is 7.
WEEKDAY	CHAR(9)	Name of the day of the week.
WEEK_NUM	INTEGER	Ordinal number for the week in the month, where the first week is 1 and the last week is 4 or 5.
MONTH_NUM	INTEGER	Ordinal number for the week in the year, where January is 1 and December is 12
MONTH	CHAR(9)	Name of the month.
QUARTILE_NUM	INTEGER	Ordinal number for the quarter in the year, where Jan-Mar is 1 and Oct-Dec is 4.
YEAR	INTEGER	Name of the year.

TIME_DIM

TIME_DIM is a system-generated time dimension.

TIME_DIM Dimension Table Columns

Column Name	Data Type	Description
TIME_KEY	INTEGER	Primary key. Not Null.
TIME_OF_DAY	TIME	Date in standard date format from ISO 8601. Looks like this: ""19991231"".
SECOND_NUM	INTEGER	Ordinal number for the second.
MINUTE_NUM	INTEGER	Ordinal number for the minute.
HOURL_NUM	INTEGER	Ordinal number for the hour.
SHIFT_NUM	INTEGER	Ordinal number for the shift.

BATCH_DIM

BATCH_DIM is a system-generated dimension that contains information about the batch jobs handling the ETL for VHist.

BATCH_DIM Dimension Table Columns

Column Name	Data Type	Description
BATCH_ID	INTEGER	A unique numeric ID assigned by the VHist loading processes, which identifies the ETL batch.
BATCH_CURRENT_TS	TIMESTAMP(6)	The timestamp of the batch
SCRIPT_NAME	VARCHAR(30)	The name of the batch processing script. Usually <code>source_to_stage</code> or <code>stage_to_star</code>

NODE_DIM

NODE_DIM contains data about the nodes in the database.

Source table

The source table for VHIST.NODE_DIM is V_CATALOG.NODES. For details about the source table, see [NODES](#) in the *Vertica SQL Reference Manual*.

NODE_DIM Dimension Table Columns

Column Name	Data Type	Description	VHist
NODE_KEY	INTEGER	A unique numeric surrogate key assigned by Vertica sequence. Identifies the node for a particular date range.	surrogate key
NODE_NAME	VARCHAR(128)	The name of the node.	attribute (SCD-2)
NODE_ID	INTEGER	A numeric ID, assigned by the Vertica catalog, which uniquely identifies the node.	attribute (object identifier)
NODE_STATE	VARCHAR(128)	The node's current state (up, down, recovering, etc.).	attribute (SCD-2)
NODE_ADDRESS	VARCHAR(80)	The host address of the node.	attribute (SCD-2)
NODE_ADDRESS_FAMILY	VARCHAR(10)	The IP Version of the NODE_ADDRESS. For example, ipv4.	attribute (SCD-2)
EXPORT_ADDRESS	VARCHAR(8192)	The IP address of the node (on the public network) used for import/export operations and native load-balancing.	attribute (SCD-2)
EXPORT_ADDRESS_FAMILY	VARCHAR(10)	The IP Version of the EXPORT_ADDRESS. For example, ipv4.	attribute (SCD-2)
CATALOG_PATH	VARCHAR(8192)	The absolute path to the catalog on the node.	attribute (SCD-2)

NODE_TYPE	VARCHAR(9)	The type of the node. For more information on the types of nodes, refer to Node Types in the Vertica documentation.	attribute (SCD-2)
IS_EPHEMERAL	BOOLEAN	(Deprecated) True if this node has been marked as ephemeral in preparation for removing it from the cluster.	attribute (SCD-2)
STANDING_IN_FOR	VARCHAR(128)	The name of the node that this node is currently replacing.	attribute (SCD-2)
NODE_DOWN_SINCE	TIMESTAMPZ	The amount of time that the replaced node has been unavailable.	attribute (SCD-2)
VALID_AFTER_TS	timestamp	This row is valid starting after this timestamp.	attribute - DW metadata
VALID_THROUGH_TS	timestamp	This row is valid ending at this timestamp.	attribute - DW metadata
IS_CURRENT	boolean	Is this row the currently active row for this node_id?	attribute - DW metadata
NODE_DOWN_SINCE	INTEGER	A unique numeric surrogate key assigned by Vertica sequence. Identifies the node for a particular date range.	surrogate key

USER_DIM

USER_DIM contains data about the users in the database.

Source table

The source table for VHist.USER_DIM is V_CATALOG.USERS. For details about the source table, see [USERS](#) in the *Vertica SQL Reference Manual*.

USER_DIM Dimension Table Columns

Column Name	Data Type	Description	VHist
USER_KEY	INTEGER	A unique numeric surrogate key assigned by Vertica sequence. Identifies the user for a particular date range.	surrogate key
USER_ID	INTEGER	A unique numeric ID assigned by the Vertica catalog. Identifies the user.	attribute (object identifier)
USER_NAME	VARCHAR	The user name for which information is listed.	attribute (SCD-2)
IS_SUPER_USER	BOOLEAN	Indicates whether the current user is super user: true or false.	attribute (SCD-2)
PROFILE_NAME	VARCHAR	The name of the profile to which the user is assigned. The profile controls the user's password policy.	attribute (SCD-2)
IS_LOCKED	BOOLEAN	Whether the user's account is locked. A locked user cannot log into the system.	attribute (SCD-2)
LOCK_TIME	DATETIME	When the user's account was locked. Used to determine when to automatically unlock the account, if the user's profile has a <code>PASSWORD_LOCK_TIME</code> parameter set.	attribute (SCD-2)

RESOURCE_POOL	VARCHAR	The resource pool to which the user is assigned.	attribute (SCD-2)
MEMORY_CAP_KB	VARCHAR	The maximum amount of memory in kilobytes that a query run by the user can consume.	attribute (SCD-2)
TEMP_SPACE_CAP_KB	VARCHAR	The maximum amount of temporary disk space in kilobytes that a query run by the user can consume.	attribute (SCD-2)
RUN_TIME_CAP	VARCHAR	The maximum amount of time any of the user's queries is allowed to run.	attribute (SCD-2)
ALL_ROLES	VARCHAR	Roles assigned to the user. An asterisk in ALL_ROLES output means role granted WITH ADMIN OPTION. See About Database Roles in the Vertica documentation.	attribute (SCD-2)
DEFAULT_ROLES	VARCHAR	Default roles assigned to user. An asterisk in DEFAULT_ROLES output means role granted WITH ADMIN OPTION. See Default Roles for Database Users in the Vertica documentation.	attribute (SCD-2)
SEARCH_PATH	VARCHAR	Sets the default schema search path for the user. See Setting Search Paths in the Vertica documentation.	attribute (SCD-2)
VALID_AFTER_TS	TIMESTAMP	This row is valid starting after this timestamp.	attribute – data warehouse metadata
VALID_THROUGH_TS	TIMESTAMP	This row is valid ending at this timestamp.	attribute – data warehouse metadata

Find More Information

Subject	Link
Vertica	https://my.vertica.com/
Vertica system tables	Vertica System Tables
Vertica documentation	Vertica Documentation
Vertica Developer Community	HPE Developer Community – Big Data and Analytics
Vertica Knowledge Base	Vertica Knowledge Base
HPE Big Data Marketplace	HPE Big Data Marketplace

Contact Us

We welcome your feedback. If you have questions, comments, or suggestions about VHist, please contact us by clicking the **Contact developer** button on the [HPE Big Data Marketplace](#) download page for any of the ETL QuickStarts.

