Home

- Apache Hive
- Hive Documentation
    - General Information about Hive
    - User Documentation
    - Administrator Documentation
    - HCatalog and WebHCat Documentation
    - Resources for Contributors
- Hive Versions and Branches

Apache Hive

The **Apache Hive**^{TM} data warehouse software facilitates querying and managing large datasets residing in distributed storage. Built on top of **Apache Hadoop**^{TM}, it provides

- Tools to enable easy data extract/transform/load (ETL)
- A mechanism to impose structure on a variety of data formats
- Access to files stored either directly in **Apache HDFS**^{TM} or in other data storage systems such as **Apache HBase**^{TM}
- Query execution via MapReduce

Hive defines a simple SQL-like query language, called QL, that enables users familiar with SQL to query the data. At the same time, this language also allows programmers who are familiar with the MapReduce framework to be able to plug in their custom mappers and reducers to perform more sophisticated analysis that may not be supported by the built-in capabilities of the language. QL can also be extended with custom scalar functions (UDF's), aggregations (UDAF's), and table functions (UDTF's).

Hive does not mandate read or written data be in the "Hive format" — there is no such thing. Hive works equally well on Thrift, control delimited, or your specialized data formats. Please see File Formats and Hive SerDe in the Developer Guide for details.

Hive is not designed for OLTP workloads and does not offer real-time queries or row-level updates. It is best used for batch jobs over large sets of append-only data (like web logs). What Hive values most are scalability (scale out with more machines added dynamically to the Hadoop cluster), extensibility (with MapReduce framework and UDF/UDAF/UDTF), fault-tolerance, and loose-coupling with its input formats.

Components of Hive include HCatalog and WebHCat.

- **HCatalog** is a component of Hive. It is a table and storage management layer for Hadoop that enables users with different data processing tools — including Pig and MapReduce — to more easily read and write data on the grid.
- **WebHCat** provides a service that you can use to run Hadoop MapReduce (or YARN), Pig, Hive jobs or perform Hive metadata operations using an HTTP (REST style) interface.

Hive Documentation

The links below provide access to the Apache Hive wiki documents. This list is not complete, but you can navigate through these wiki pages to find additional documents. For more information, please see the official Hive website.

General Information about Hive

- Getting Started
- Books about Hive
- Presentations and Papers about Hive
- Sites and Applications Powered by Hive
- Related Projects
- FAQ
- Hive Users Mailing List
- Hive IRC Channel: `#hive` on irc.freenode.net
- About This Wiki

User Documentation

- Hive Tutorial
- HiveQL Language Manual: Commands, CLIs, File Formats, Data Types,

DDL (create/drop/alter/truncate/show/describe), Statistics (analyze), Indexes, Archiving,
DML (load/insert/update/delete, import/export, explain plan),
Queries (select), Operators and UDFs, Locks, Authorization

- Procedural Language: Hive HPL/SQL
- Unit testing HQL
- Hive Configuration Properties
- Hive Clients
    - Hive Client (JDBC, ODBC, Thrift)
    - HiveServer2 Client and Beeline
- Hive Web Interface
- Hive SerDes: Avro SerDe, Parquet SerDe, CSV SerDe, JSON SerDe
- Hive Accumulo Integration
- Hive HBase Integration
- Hive Transactions, Streaming Data Ingest, and Streaming Mutation API

Administrator Documentation

- Installing Hive
- Configuring Hive
- Setting Up Metastore
- Setting Up Hive Web Interface
- Setting Up Hive Server (JDBC, ODBC, Thrift, HiveServer2)
- Hive Replication
- Hive on Amazon Web Services
- Hive on Amazon Elastic MapReduce
- Hive on Spark: Getting Started

HCatalog and WebHCat Documentation

- HCatalog
- WebHCat (Templeton)

Resources for Contributors

- How to Contribute
- Hive Contributors Meetings
- Hive Developer Docs
    - Hive Developer Guide (code organization, compile and run Hive, unit tests, debug, pluggable interfaces)
    - Hive Developer FAQ (move files, build Hive, test Hive, MiniDriver and Beeline tests)
    - Plugin Developer Kit
    - Writing UDTFs
    - Hive APIs Overview
- Hive Testing Docs
    - FAQ: Testing
    - Developer Guide: Unit Tests
    - Unit Test Parallel Execution
    - Tips for Adding New Tests
    - Hive PTest2 Infrastructure
    - Hive PreCommit Patch Testing
    - MiniDriver and Beeline Tests
- Hive Performance
- Hive Architecture Overview
- Hive Design Docs: Completed; In Progress; Proposed; Incomplete, Abandoned, Other
- Roadmap/Call to Add More Features
- Full-Text Search over All Hive Resources
- How to edit the website
- Becoming a Committer
- How to Commit
- How to Release
- Project Bylaws

Hive Versions and Branches

Recent versions of Hive are available on the Downloads page of the Hive website. For each version, the page provides the release date and a link to the change log. If you want a change log for an earlier version (or a development branch), use the Configure Release Notes page.

The Apache Hive JIRA keeps track of changes to Hive code, documentation, infrastructure, etc. The version number or branch for each resolved JIRA issue is shown in the "Fix Version/s" field in the Details section at the top of the issue page. For example, HIVE-5107 has a fix version of 0.13.0.

Sometimes a version number changes before the release. When that happens the original number might still be found in the JIRA, wiki, and mailing list discussions. For example:

Release Number	Original Number
1.0.0	0.14.1
1.1.0	0.15.0

More information about Hive branches is available in How to Contribute: Understanding Hive Branches.

Apache Hive, Apache Hadoop, Apache HBase, Apache HDFS, Apache, the Apache feather logo, and the Apache Hive project logo are trademarks of The Apache Software Foundation.