


# Hive MetaTool

- [Hive MetaTool](#)
  - [The metatool Command](#)
  - [Usage Example](#)

## Hive MetaTool

Version 0.10.0 and later

 Introduced in Hive 0.10.0. See [HIVE-3056](#) and [HIVE-3443](#).

The Hive MetaTool enables administrators to do bulk updates on the location fields in database, table, and partition records in the metastore. It provides the following functionality:

- Ability to search and replace the HDFS NN (NameNode) location in metastore records that reference the NN. One use is to transition a Hive deployment to [HDFS HA NN](#) (HDFS High Availability NameNode).
- A command line tool to execute [JDOQL](#) against the metastore. The ability to execute JDOQL against the metastore can be a useful debugging tool for both users and Hive developers.


## The `metatool` Command

The `metatool` command invokes the Hive MetaTool with these options:

| Option   | Description  |
|--|--|
| <code>-listFSRoot</code>                                     | Print the location of the current file system root (NameNode). The value is prefixed with <code>hdfs://</code> scheme.   |
| <code>-updateLocation &lt;new-loc&gt; &lt;old-loc&gt;</code> | Update records in the Hive metastore to point to a new NameNode location (file system root location). Both <i>new-loc</i> and <i>old-loc</i> should be valid URIs with valid host names and schemes. For upgrading to HDFS HA NN, <i>new-loc</i> should match the value of the <code>dfs.nameservices</code> property. The <i>old-loc</i> should match the value returned by the <code>listFSRoot</code> option.<br><br>When run with the <b>dryRun</b> option, changes are displayed but are not persisted. When run with the <b>serdePropKey/tablePropKey</b> option, <b>updateLocation</b> looks for the <i>serde-prop-key/table-prop-key</i> that is specified and updates its value if found. |
| <code>-serdePropKey &lt;serde-prop-key&gt;</code>            | Specify a SerDe property key whose value field may reference the HDFS NameNode location and hence may require an update. For example to update the Haiivvero schema URL, specify <code>schema.url</code> for this argument.<br><br>This option is valid only with the <b>updateLocation</b> option.  |
| <code>-tablePropKey &lt;table-prop-key&gt;</code>            | Specify a table property key whose value field may reference the HDFS NameNode location and hence may require an update. For example to update the Avro SerDe schema URL, specify <code>avro.schema.url</code> for this argument.<br><br>This option is valid only with the <b>updateLocation</b> option.  |
| <code>-dryRun</code>   | Perform a dry run of <b>updateLocation</b> changes. The <b>updateLocation</b> changes are displayed but not persisted.<br><br>This option is valid only with the <b>updateLocation</b> option.   |
| <code>-executeJDOQL &lt;query-string&gt;</code>              | Execute the given JDOQL query.   |
| <code>-help</code>   | Print a list of command options and their descriptions.  |

If you are unsure which version of Avro SerDe is used, pass both **serdePropKey** and **tablePropKey** arguments with their respective values for the keys to **updateLocation**.

Hive Configuration Directory

 Note that `metatool` is a command for Hive administrators and it needs direct access to the metastore RDBMS. In some environments, the RDBMS login information (especially password or the password key) is only stored on the metastore server host or in a metastore server specific config directory (with more restrictive file system permissions). In such cases, make sure you run on the metastore server machine, as the hive user, and also set the `HIVE_CONF_DIR` environment variable appropriately (for example, `export HIVE_CONF_DIR=<path to metastore server config dir>`).

## Usage Example

The following `metatool` command uses the **updateLocation**, **tablePropKey**, and **serdePropKey** options to update the NameNode location to `hdfs://localhost:9000` from `hdfs://namenode2:8020`.

```

./hive --service metatool -updateLocation hdfs://localhost:9000 hdfs://namenode2:8020 -tablePropKey avro.schema.url -serdePropKey avro.schema.url
Initializing HiveMetaTool..
15/04/22 14:18:42 INFO metastore.ObjectStore: ObjectStore, initialize called
15/04/22 14:18:42 INFO DataNucleus.Persistence: Property hive.metastore.integral.jdo.pushdown unknown - will be ignored
15/04/22 14:18:42 INFO DataNucleus.Persistence: Property datanucleus.cache.level2 unknown - will be ignored
15/04/22 14:18:43 INFO metastore.ObjectStore: Setting MetaStore object pin classes with hive.metastore.cache.pinobjtypes="Table,StorageDescriptor,SerdeInfo,Partition,Database,Type,FieldSchema,Order"
15/04/22 14:18:43 INFO DataNucleus.Datastore: The class "org.apache.hadoop.hive.metastore.model.MFieldSchema" is tagged as "embedded-only" so does not have its own datastore table.
15/04/22 14:18:43 INFO DataNucleus.Datastore: The class "org.apache.hadoop.hive.metastore.model.MOrder" is tagged as "embedded-only" so does not have its own datastore table.
15/04/22 14:18:44 INFO DataNucleus.Datastore: The class "org.apache.hadoop.hive.metastore.model.MFieldSchema" is tagged as "embedded-only" so does not have its own datastore table.
15/04/22 14:18:44 INFO DataNucleus.Datastore: The class "org.apache.hadoop.hive.metastore.model.MOrder" is tagged as "embedded-only" so does not have its own datastore table.
15/04/22 14:18:44 INFO DataNucleus.Query: Reading in results for query "org.datanucleus.store.rdbms.query.SQLQuery@0" since the connection used is closing
15/04/22 14:18:44 INFO metastore.MetaStoreDirectSql: Using direct SQL, underlying DB is MYSQL
15/04/22 14:18:44 INFO metastore.ObjectStore: Initialized ObjectStore
Looking for LOCATION_URI field in DBS table to update..
Successfully updated the following locations..
Updated 0 records in DBS table
Looking for LOCATION field in SDS table to update..
Successfully updated the following locations..
Updated 0 records in SDS table
Looking for value of avro.schema.url key in TABLE_PARAMS table to update..
Successfully updated the following locations..
Updated 0 records in TABLE_PARAMS table
Looking for value of avro.schema.url key in SD_PARAMS table to update..
Successfully updated the following locations..
Updated 0 records in SD_PARAMS table
Looking for value of avro.schema.url key in SERDE_PARAMS table to update..
Successfully updated the following locations..
Updated 0 records in SERDE_PARAMS table

```

Using metatool to read out table information (for all metastore backends).

```

HIVE_CONF_DIR=/etc/hive/conf/conf.server/ hive --service metatool -executeJDOQL 'select dbName+"."+tableName+": "+colName+"="+numDVs from org.apache.hadoop.hive.metastore.model.MTableColumnStatistics';

```

```

HIVE_CONF_DIR=/etc/hive/conf/conf.server/ hive --service metatool -executeJDOQL 'select dbName+"."+tableName+"("+partitionName+": "+colName+"="+numDVs from org.apache.hadoop.hive.metastore.model.MPartitionColumnStatistics';

```