

Cloudera Release Guide



Important Notice

(c) 2010-2015 Cloudera, Inc. All rights reserved.

Cloudera, the Cloudera logo, Cloudera Impala, and any other product or service names or slogans contained in this document are trademarks of Cloudera and its suppliers or licensors, and may not be copied, imitated or used, in whole or in part, without the prior written permission of Cloudera or the applicable trademark holder.

Hadoop and the Hadoop elephant logo are trademarks of the Apache Software Foundation. All other trademarks, registered trademarks, product names and company names or logos mentioned in this document are the property of their respective owners. Reference to any products, services, processes or other information, by trade name, trademark, manufacturer, supplier or otherwise does not constitute or imply endorsement, sponsorship or recommendation thereof by us.

Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Cloudera.

Cloudera may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Cloudera, the furnishing of this document does not give you any license to these patents, trademarks copyrights, or other intellectual property. For information about patents covering Cloudera products, see <http://tiny.cloudera.com/patents>.

The information in this document is subject to change without notice. Cloudera shall not be liable for any damages resulting from technical errors or omissions which may be present in this document, or from use of this document.

Cloudera, Inc.
1001 Page Mill Road Bldg 2
Palo Alto, CA 94304
info@cloudera.com
US: 1-888-789-1488
Intl: 1-650-362-0488
www.cloudera.com

Release Information

Version: 5.3.x
Date: June 24, 2015

Table of Contents

About Cloudera Release Guide.....	4
--	----------

Release Notes.....	5
---------------------------	----------

CDH 5 Release Notes.....	5
<i>New Features in CDH 5.....</i>	<i>5</i>
<i>Incompatible Changes.....</i>	<i>49</i>
<i>Known Issues in CDH 5.....</i>	<i>69</i>
<i>Issues Fixed in CDH 5.....</i>	<i>100</i>
Cloudera Manager 5 Release Notes.....	153
<i>New Features and Changes in Cloudera Manager 5.....</i>	<i>153</i>
<i>Known Issues and Workarounds in Cloudera Manager 5.....</i>	<i>165</i>
<i>Issues Fixed in Cloudera Manager 5.....</i>	<i>173</i>
Cloudera Navigator 2 Release Notes.....	191
<i>New Features and Changes in Cloudera Navigator 2.....</i>	<i>191</i>
<i>Known Issues and Workarounds in Cloudera Navigator 2.....</i>	<i>194</i>
<i>Issues Fixed in Cloudera Navigator 2.....</i>	<i>195</i>

Version and Download Information.....	200
--	------------

Product Compatibility Matrix.....	201
--	------------

JDK Compatibility.....	201
<i>Cloudera Manager - JDK Compatibility.....</i>	<i>201</i>
<i>CDH - JDK Compatibility.....</i>	<i>202</i>
CDH and Cloudera Manager.....	203
Apache Accumulo.....	203
Backup and Disaster Recovery.....	204
Cloudera Impala.....	204
Cloudera Navigator.....	206
Cloudera Search.....	209
<i>Cloudera Search for CDH 4.....</i>	<i>209</i>
<i>Cloudera Search for CDH 5.....</i>	<i>209</i>
Apache Sentry (incubating).....	209
Apache Spark.....	210

About Cloudera Release Guide

This guide contains release and download information for installers and administrators. It includes release notes as well as information about versions and downloads. The guide also provides a release matrix that shows which major and minor release version of a product is supported with which release version of Cloudera Manager, CDH and, if applicable, Cloudera Search and Cloudera Impala.

Release Notes

CDH 5 Release Notes

This document includes the following major sections:

- [New Features in CDH 5](#) on page 5
- [Incompatible Changes](#) on page 49
- [Known Issues in CDH 5](#) on page 69
- [Issues Fixed in CDH 5](#) on page 100

For links to the detailed change lists that describe the bug fixes and improvements to all of the CDH 5 projects, see the packaging section of [CDH Version and Packaging Information](#).

New Features in CDH 5

▪ **Note:**

There is no CDH 5.2.2 release.

About Apache Hadoop MapReduce Version 1 (MRv1) and Version 2 (MRv2)

- **Important:** Cloudera recommends that you use YARN (now production-ready) with CDH 5.

- **MapReduce 2.0 (MRv2):** CDH 5 includes MapReduce 2.0 (MRv2) running on YARN. The fundamental idea of the YARN architecture is to split up the two primary responsibilities of the JobTracker — resource management and job scheduling/monitoring — into separate daemons: a global ResourceManager (RM) and per-application ApplicationMasters (AM). With MRv2, the ResourceManager (RM) and per-node NodeManagers (NM), form the data-computation framework. The ResourceManager service effectively replaces the functions of the JobTracker, and NodeManagers run on slave nodes instead of TaskTracker daemons. The per-application ApplicationMaster is, in effect, a framework-specific library and is tasked with negotiating resources from the ResourceManager and working with the NodeManager(s) to execute and monitor the tasks. For details of the new architecture, see [Apache Hadoop NextGen MapReduce \(YARN\)](#).
- **MapReduce Version 1 (MRv1):** For backward compatibility, CDH 5 continues to support the original MapReduce framework (i.e. the JobTracker and TaskTrackers), but you should begin migrating to MRv2.

▪ **Note:**

Cloudera does not support running MRv1 and YARN daemons on the same nodes at the same time.

- **Deprecated properties:**

In Hadoop 2.0.0 and later (MRv2), a number of Hadoop and HDFS properties have been deprecated. (The change dates from Hadoop 0.23.1, on which the Beta releases of CDH 4 were based). A list of deprecated properties and their replacements can be found at [Hadoop Deprecated Properties](#).

- **Note:** All of these deprecated properties continue to work in MRv1. Conversely the newmapreduce* properties listed do not work in MRv1.

What's New in CDH 5.3.5

This is a maintenance release that fixes the following issue. For details of other fixes, see [Known Issues Fixed in CDH 5.3.5](#) on page 100:

Potential job failures during YARN rolling upgrades to CDH 5.3.4

Problem: A MapReduce security fix introduced a compatibility issue that results in job failures during YARN rolling upgrades from CDH 5.3.3 to CDH 5.3.4.

Release affected: CDH 5.3.4

Release containing the fix: CDH 5.3.5

Workarounds: You can use any one of the following workarounds for this issue:

- Upgrade to CDH 5.3.5.
- Restart any jobs that might have failed during the upgrade.
- Explicitly set the version of MapReduce to be used so it is picked on a per-job basis.
 1. Update the YARN property, **MR Application Classpath** (`mapreduce.application.classpath`), either in Cloudera Manager or in the `mapred-site.xml` file. Remove all existing values and add a new entry: `<parcel-path>/lib/hadoop-mapreduce/*`, where `<parcel-path>` is the absolute path to the parcel installation. For example, the default installation path for the CDH 5.3.3 parcel would be:
`/opt/cloudera/parcels/CDH-5.3.3-1.cdh5.3.3.p0.5/lib/hadoop-mapreduce/*`.
 2. Wait until jobs submitted with the above client configuration change have run to completion.
 3. Upgrade to CDH 5.3.4.
 4. Update the **MR Application Classpath** (`mapreduce.application.classpath`) property to point to the new CDH 5.3.4 parcel.

Do *not* delete the old parcel until after all jobs submitted prior to the upgrade have finished running.

What's New in CDH 5.3.4

This is a maintenance release that fixes the following issue; for details of other important fixes, see [Known Issues Fixed in CDH 5.3.4](#) on page 101 .

- **Important:** Rolling upgrades to CDH 5.3.4 are not supported. If this is a problem, upgrade to CDH 5.3.5 (or higher) instead.

NameNode Incorrectly Reports Missing Blocks During Rolling Upgrade

Problem: During a rolling upgrade to CDH 5.3.3, the NameNode may report missing blocks after rolling multiple DataNodes. This is caused by a race condition with block reporting between the DataNode and the NameNode. No permanent data loss occurs, but data can be unavailable for up to six hours before the problem corrects itself.

Releases affected: CDH 5.0.6, 5.1.5, 5.2.5, 5.3.3

Releases containing the fix: 5.3.4

Workaround:

To avoid the problem: Cloudera advises skipping CDH 5.3.3, and upgrading to CDH 5.3.4 instead. Before upgrading to CDH 5.3.4, see [Install and Upgrade Known Issues](#) on page 69.

If you have already completed an upgrade to an affected release, or are installing a new cluster: In this case you can continue to run CDH 5.3.3, or upgrade to CDH 5.3.4, as you choose.

What's New in CDH 5.3.3

This is a maintenance release that fixes some important issues; for details, see [Known Issues Fixed in CDH 5.3.3](#) on page 103.

What's New in CDH 5.3.2

This is a maintenance release that fixes some important issues; for details, see [Known Issues Fixed in CDH 5.3.2](#) on page 104.

What's New in CDH 5.3.1

This is a maintenance release that fixes some important issues; for details, see [Known Issues Fixed in CDH 5.3.1](#) on page 107.

What's New in CDH 5.3.0

The following topics describe new features introduced in CDH 5.3.0.

Oracle JDK 8 Support

CDH 5.3 supports Oracle JDK 1.8. For important information and requirements, see [CDH 5 Requirements and Supported Versions](#) and [Upgrading to Oracle JDK 1.8](#).

Apache Hadoop HDFS

CDH 5.3 provides the following new capabilities:

- **HDFS Data At Rest Encryption** - This feature is now ready for use in production environments.

- **Important:**

Client hosts may need a more recent version of `libcrypto.so`. See [Apache Hadoop Known Issues](#) on page 71 for more information.

- **Important:** Cloudera provides two solutions:

- **Navigator Encrypt** is production ready and available to Cloudera customers licensed for Cloudera Navigator. Navigator Encrypt operates at the Linux volume level, so it can encrypt cluster data inside and outside HDFS. Consult your Cloudera account team for more information.
- **HDFS Encryption** is production ready and operates at the HDFS directory level, enabling encryption to be applied only to HDFS folders where needed.

- **S3A** - S3A is an HDFS implementation of the Simple Storage Service (S3) from Amazon Web Services. It is similar to S3N, which is the other implementation of this functionality. The key difference is that S3A relies on the officially-supported AWS Java SDK for communicating with S3, while S3N uses a best-effort-supported `jets3t` library to do the same. For a listing of the parameters, see [HADOOP-10400](#).

YARN

YARN now provides a way for long-running applications to get new delegation tokens.

Apache Flume

CDH 5.3 provides a Kafka Channel ([FLUME-2500](#)).

Apache HBase

CDH 5.3 provides `checkAndMutate(RowMutations)`, in addition to existing support for atomic `checkAndPut` as well as `checkAndDelete` operations on individual rows ([HBASE-11796](#)).

Apache Hive

- Hive can use multiple HDFS encryption zones.
- Hive-HBase integration contains many fixes and new features such as reading HBase snapshots.
- Many Hive Parquet fixes.
- Hive Server 2 can handle multiple LDAP domains for authentication.

Hue

New Features:

- Hue is re-based on Hue 3.7
- [SAML authentication](#) has been revamped.
- CDH 5.3 simplifies the task of configuring Hue to store data in an Oracle database by bundling the Oracle Install Client. For instructions, see [Hue Database](#).

Apache Oozie

- You can now update the definition and properties of an already running Coordinator. See the [documentation](#) for more information.
- A new `poll` command in the Oozie client polls a Workflow Job, Coordinator Job, Coordinator Action, or Bundle Job until it finishes. See the [documentation](#) for more information.

Apache Parquet

- [PARQUET-132](#): Add type parameter to AvroParquetInputFormat for Spark
- [PARQUET-107](#): Add option to disable summary metadata files
- [PARQUET-64](#): Add support for new type annotations (date, time, timestamp, etc.)

Cloudera Search

New Features:

- Cloudera Search includes a version of Kite 0.15.0, which includes all morphlines-related backports of all fixes and features in Kite 0.17.1. Morphlines now includes functionality that enables partially updating document as well as deleting documents. Partial updating or deleting can be completed by unique IDs or by documents that match a query. For additional information on Kite, see:
 - [Kite repository](#)
 - [Kite Release Notes](#)
 - [Kite documentation](#)
 - [Kite examples](#)
- CrunchIndexerTool now sends a commit to Solr on job success.
- Added support for deleting documents stored in Solr [by unique id](#) as well as [by query](#).

Apache Sentry (incubating)

- Sentry HDFS Plugin - Allows you to configure synchronization of Sentry privileges to HDFS ACLs for specific HDFS directories. This simplifies the process of sharing table data between Hive or Impala and other clients (such as MapReduce, Pig, Spark), by automatically updating the ACLs when a `GRANT` or `REVOKE` statement is executed. It also allows all roles and privileges to be managed in a central location (by Sentry).
- Metrics - CDH 5.3 supports metrics for the Sentry service. These metrics can be reported either through JMX or the console; configure this by setting the property `sentry.service.reporter` to `jmx` or `console`. A Sentry web server listening by default on port 51000 can expose the metrics in `json` format. Web reporting is disabled by default; enable it by setting `sentry.service.web.enable` to `true`. You can configure the port on which Sentry web server listens by means of the `sentry.service.web.port` property.

Apache Spark

- CDH Spark has been rebased on Apache Spark 1.2.0.
- Spark Streaming can now save incoming data to a WAL (write-ahead log) on HDFS, preventing any data loss on driver failure.

■ Important:

This feature is currently in Beta; Cloudera includes it in CDH Spark but does not support it.

- The YARN back end now supports dynamic allocation of executors. See <http://spark.apache.org/docs/latest/job-scheduling.html> for more information.
- Native library paths (set via Spark configuration options) are correctly propagated to executors in YARN mode ([SPARK-1719](#)).
- The Snappy codec should now work out-of-the-box on Linux distributions with older `glibc` versions such as CentOS 5.
- Spark SQL now includes the Spark Thrift Server in CDH.

▪ **Important:**

Spark SQL remains an experimental and unsupported feature in CDH.

See [Apache Spark Incompatible Changes](#) on page 68 and [Apache Spark Known Issues](#) on page 98 for additional important information.

Apache Sqoop

- **Sqoop 1:**
 - The MySQL connector now fetches on a row-by row-basis.
 - The SQL server now has *upsert* (insert or update) support ([SQOOP-1403](#)).
 - The Oracle direct connector now works with index-organized tables ([SQOOP-1632](#)). To use this capability, you must set the chunk method to `PARTITION`:

```
-Doraoop.chunk.method=PARTITION
```

- **Sqoop 2:**
 - `FROM/TO` re-factoring is now supported ([SQOOP-1367](#)).

What's New in CDH 5.2.5

This is a maintenance release that fixes some important issues; for details, see [Known Issues Fixed in CDH 5.2.5](#) on page 108.

What's New in CDH 5.2.4

This is a maintenance release that fixes some important issues; for details, see [Known Issues Fixed in CDH 5.2.4](#) on page 109.

What's New in CDH 5.2.3

This is a maintenance release that fixes some important issues; for details, see [Known Issues Fixed in CDH 5.2.3](#) on page 110.

What's New in CDH 5.2.1

This is a maintenance release that fixes the “POODLE” and Apache Hadoop Distributed Cache vulnerabilities described below. All CDH 5.2.0 users should upgrade to 5.2.1 as soon as possible.

“POODLE” Vulnerability on SSL/TLS enabled ports

The POODLE (Padding Oracle On Downgraded Legacy Encryption) attack forces the use of the obsolete SSLv3 protocol and then exploits a cryptographic flaw in SSLv3. The only solution is to disable SSLv3 entirely. This requires changes across a wide variety of components of CDH and Cloudera Manager in 5.2.0 and all earlier versions. CDH 5.2.1 provides these changes for CDH 5.2.0 deployments. For more information, see the [Cloudera Security Bulletin](#).

Apache Hadoop Distributed Cache Vulnerability

The Distributed Cache Vulnerability allows a malicious cluster user to expose private files owned by the user running the YARN NodeManager process. For more information, see the [Cloudera Security Bulletin](#).

Other Fixes

CDH 5.2.1 also fixes the following issues:

- [HADOOP-11243](#) - SSLFactory shouldn't allow SSLv3
- [HADOOP-11217](#) - Disable SSLv3 in KMS
- [HADOOP-11156](#) - DelegateToFileSystem should implement getFsStatus(final Path f).
- [HADOOP-11176](#) - KMSClientProvider authentication fails when both currentUgi and loginUgi are a proxied user
- [HDFS-7235](#) - DataNode#transferBlock should report blocks that don't exist using reportBadBlock
- [HDFS-7274](#) - Disable SSLv3 in HttpFS
- [HDFS-7391](#) - Reenable SSLv2Hello in HttpFS
- [HDFS-6781](#) - Separate HDFS commands from CommandsManual.appt.vm
- [HDFS-6831](#) - Inconsistency between 'hdfs dfsadmin' and 'hdfs dfsadmin -help'
- [HDFS-7278](#) - Add a command that allows sysadmins to manually trigger full block reports from a DN
- [YARN-2010](#) - Handle app-recovery failures gracefully
- [YARN-2588](#) - Standby RM does not transitionToActive if previous transitionToActive is failed with ZK exception.
- [YARN-2566](#) - DefaultContainerExecutor should pick a working directory randomly
- [YARN-2641](#) - Decommission nodes on -refreshNodes instead of next NM-RM heartbeat
- [MAPREDUCE-6147](#) - Support mapreduce.input.fileinputformat.split.maxsize
- [HBASE-12376](#) - HBaseAdmin leaks ZK connections if failure starting watchers (ConnectionLossException)
- [HBASE-12201](#) - Close the writers in the MOB sweep tool
- [HBASE-12220](#) - Add hedgedReads and hedgedReadWins metrics
- [HIVE-8693](#) - Separate out fair scheduler dependency from hadoop 0.23 shim
- [HIVE-8634](#) - HiveServer2 fair scheduler queue mapping doesn't handle the secondary groups rules correctly
- [HIVE-8675](#) - Increase thrift server protocol test coverage
- [HIVE-8827](#) - Remove SSLv2Hello from list of disabled protocols protocols
- [HIVE-8615](#) - beeline csv,tsv outputformat needs backward compatibility mode
- [HIVE-8627](#) - Compute stats on a table from impala caused the table to be corrupted
- [HIVE-7764](#) - Support all JDBC-HiveServer2 authentication modes on a secure cluster cluster
- [HIVE-8182](#) - beeline fails when executing multiple-line queries with trailing spaces
- [HUE-2438](#) - [core] Disable SSLv3 for Poodle vulnerability
- [IMPALA-1361: FE Exceptions with BETWEEN predicates](#)
- [IMPALA-1397: free local expr allocations in scanner threads](#)
- [IMPALA-1400: Window function insert issue \(LAG\(\) + OVER\)](#)
- [IMPALA-1401: raise MAX_PAGE_HEADER_SIZE and use scanner context to stitch together header buffer](#)
- [IMPALA-1410: accept "single character" character classes in regex functions](#)
- [IMPALA-1411: Create table as select produces incorrect results](#)
- [IMPALA-1416](#) - Queries fail with metastore exception after upgrade and compute stats
- [OOZIE-2034](#) - Disable SSLv3 (POODLEbleed vulnerability)
- [OOZIE-2063](#) - Cron syntax creates duplicate actions
- [PARQUET-107](#) - Add option to disable summary metadata aggregation after MR jobs
- [SPARK-3788](#) - Yarn dist cache code is not friendly to HDFS HA, Federation
- [SPARK-3661](#) - spark.*.memory is ignored in cluster mode
- [SPARK-3979](#) - Yarn backend's default file replication should match HDFS' default one
- [SPARK-1720](#) - use LD_LIBRARY_PATH instead of -Djava.library.path

What's New in CDH 5.2.0

- **Important:**

Upgrading to CDH 5.2.0 and later from any earlier release requires an HDFS metadata upgrade and other steps not usually required for a minor-release upgrade. See [Upgrading from an Earlier CDH 5 Release to the Latest Release](#) for more information, and be careful to follow all of the upgrade steps as instructed.

Operating System Support

CDH 5.2.0 adds support for Ubuntu Trusty (version 14.04).

- **Important:**

Installing CDH by adding a repository entails an additional step on Ubuntu Trusty, to ensure that you get the CDH version of ZooKeeper, rather than the version that is bundled with Trusty.

Apache Avro

CDH 5.2 implements Avro version 1.7.6, with backports from 1.7.7. Important changes include:

- [AVRO-1398](#): Increase default sync interval from 16k to 64k. There is a very small chance this could cause an incompatibility in some cases, but you can control the interval by setting `avro.mapred.sync.interval` in the MapReduce job configuration. For example, set it to 16000 to get the old behavior.
- [AVRO-1355](#): Record schema should reject duplicate field names. This change rejects schemas with duplicate field names. This could affect some applications, but if schemas have duplicate field names then they are unlikely to work properly in any case. The workaround is to make sure a record's field names are unique within the record.

Apache Hadoop HDFS

CDH 5.2 provides the following new capabilities:

- **HDFS Data at Rest Encryption**

- **Note:** Cloudera provides the following two solutions for data at rest encryption:

- **Navigator Encrypt** - is production ready and available for Cloudera customers licensed for Cloudera Navigator. Navigator Encrypt operates at the Linux volume level, so it can encrypt cluster data inside and outside HDFS. Talk to your Cloudera account team for more information about this capability.
- **HDFS Encryption** - included in CDH 5.2.0 operates at the HDFS folder level, enabling encryption to be applied only to HDFS folders where needed. This feature has several known limitations. Therefore, Cloudera does not currently support this feature in CDH 5.2 and it is *not* recommended for production use. If you're interested in trying the feature out, upgrade to the latest version of CDH 5.

HDFS now implements transparent, end-to-end encryption of data read from and written to HDFS by creating encryption zones. An encryption zone is a directory in HDFS with all of its contents, that is, every file and subdirectory in it, encrypted. You can use either the **KMS** or the **Key Trustee** service to store, manage, and access encryption zone keys.

HDFS now implements transparent, end-to-end encryption of data read from and written to HDFS by creating encryption zones. An encryption zone is a directory in HDFS with all of its contents, that is, every file and subdirectory in it, encrypted.

- Extended attributes: HDFS `xAttrs` allow extended attributes to be stored per file (<https://issues.apache.org/jira/browse/HDFS-2006>).
- Authentication improvements when using an HTTP proxy server.
- A new Hadoop Metrics sink that allows writing directly to Graphite.
- Specification for Hadoop Compatible Filesystem effort.
- `OfflineImageViewer` to browse an `fsimage` via the WebHDFS API.
- Supportability improvements and bug fixes to the NFS gateway.
- Modernized web UIs (HTML5 and JavaScript) for HDFS daemons.

MapReduce

CDH 5.2 provides an optimized implementation of the mapper side of the MapReduce shuffle. The optimized implementation may require tuning different from the original implementation, and so it is considered experimental and is not enabled by default.

You can select this new implementation on a per-job basis by setting the job configuration value `mapreduce.job.map.output.collector.class` to `org.apache.hadoop.mapred.native task.NativeMapOutputCollectorDelegator`, or use enable Cloudera Manager to enable it.

Some jobs which use custom writable types or comparators may not be able to take advantage of the optimized implementation.

the following new capabilities and improvements:

YARN

CDH 5.2 provides the following new capabilities and improvements:

- New features and improvements in the Fair Scheduler:
 - New features:
 - Fair Scheduler now allows setting the `fairsharePreemptionThreshold` per queue (leaf and non-leaf). This threshold is a decimal value between 0 and 1; if a queue's usage is under (`preemption-threshold * fairshare`) for a configured duration, resources from other queues are preempted to satisfy this queue's request. Set this value in `fair-scheduler.xml`. The default value is 0.5.
 - Fair Scheduler now allows setting the `fairsharePreemptionTimeout` per queue (leaf and non-leaf). For a starved queue, this timeout determines when to trigger preemption from other queues. Set this value in `fair-scheduler.xml`.
 - Fair Scheduler now shows the **Steady Fair Share** in the Web UI. The Steady Fair Share is the share of the cluster resources a particular queue or pool would get if all existing queues had running applications.
 - Improvements:
 - Fair Scheduler uses **Instantaneous Fair Share** (`fairshare` that considers only active queues) for scheduling decisions to improve the time to achieve steady state (fairshare).
 - The default for `maxAMShare` is now 0.5, meaning that only half the cluster's resources can be taken up by Application Masters. You can change this value in `fair-scheduler.xml`.
- YARN's REST APIs support submitting and killing applications.
- YARN's timeline store is integrated with Kerberos.

Apache Crunch

CDH 5.2 provides the following new capabilities:

- Improvements in [Scrunch](#), including:
 - New `join` API that matches the one in Crunch
 - New aggregation API, including support for [Algebird](#)-based aggregations
 - Built-in serialization support for all tuple types as well as case classes.

- A new module, `crunch-hive`, for reading and writing **Optimized Row Columnar** (ORC) Files with Crunch.

Apache Flume

CDH 5.2 provides the following new capabilities:

- **Kafka** Integration: Flume can now accept data from Kafka via the `KafkaSource` ([FLUME-2250](#)) and push to Kafka using the `KafkaSink` ([FLUME-2251](#)).
- Kite Sink can now write to Hive and HBase datasets ([FLUME-2463](#)).
- Flume agents can now be configured via Zookeeper (experimental, [FLUME-1491](#))
- Embedded Agents now support Interceptors ([FLUME-2426](#))
- `syslog` Sources now support configuring which fields should be kept ([FLUME-2438](#))
- File Channel replay is now much faster ([FLUME-2450](#))
- New regular-expression search-and-replace interceptor ([FLUME-2431](#))
- Backup checkpoints can be optionally compressed ([FLUME-2401](#))

Hue

CDH 5.2 provides the following new capabilities:

- New application for editing Sentry roles and Privileges on databases and tables
- Search App
- Heatmap, Tree, Leaflet widgets
- Micro-analysis of fields
- Exclusion facets
- Oozie Dashboard: bulk actions, faster display
- File Browser: drag-and-drop upload, history, ACLs edition
- Hive and Impala: LDAP pass-through, query expiration, SSL (Hive), new graphs
- Job Browser: YARN kill application button

Apache HBase

CDH 5.2 implements HBase 0.98.6, which represents a minor upgrade to HBase. This upgrade introduces new features and moves some features which were previously marked as experimental to fully supported status. For detailed information and instructions on how to use the new capabilities, see [New Features and Changes for HBase in CDH 5](#).

Apache Hive

CDH 5.2 introduces the following important changes in Hive.

- CDH 5.2 implements Hive 0.13, providing the following new capabilities:
 - Sub-queries in the `WHERE` clause
 - Common table expressions (CTE)
 - Parquet supports `timestamp`
 - HiveServer2 can be configured with a `hiverc` file that is automatically run when users connect
 - Permanent UDFs
 - HiveServer2 session and operation timeouts
 - Beeline accepts a `-i` option to initialize with a SQL file
 - New `join` syntax (implicit joins)
- As of CDH 5.2.0, you can create Avro-backed tables simply by using `STORED AS AVRO` in a DDL statement. The AvroSerDe takes care of creating the appropriate Avro schema from the Hive table schema, making it much easier to use Avro with Hive.
- Hive supports additional datatypes, as follows:
 - Hive can read `char` and `varchar` datatypes written by Hive, and `char` and `varchar` datatypes written by Impala.
 - Impala can read `char` and `varchar` datatypes written by Hive and Impala.

These new types have been enabled by expanding the supported DDL, so they are backward compatible. You can add `varchar(n)` columns by creating new tables with that type, or changing a *string* column in existing tables to `varchar`.

■ **Note:**

`char(n)` columns are not stored in a fixed-length representation, and do not improve performance (as they do in some other databases). Cloudera recommends that in most cases you use `text` or `varchar` instead.

- `DESCRIBE DATABASE` returns additional fields: `owner_name` and `owner_type`. The command will continue to behave as expected if you identify the field you're interested in by its (string) name, but could produce unexpected results if you use a numeric index to identify the field(s).

Impala

Impala in CDH 5.2.0 includes major new features such as spill-to-disk for memory-intensive queries, subquery enhancements, analytic functions, and new `CHAR` and `VARCHAR` data types. For the full feature list and more details, see [What's New in Impala](#) on page 35.

Kite

Kite is an open source set of libraries, references, tutorials, and code samples for building data-oriented systems and applications. For more information about Kite, see the [Kite SDK Development Guide](#).

Kite has been rebased to version 0.15.0 in CDH 5.2.0, from the base version 0.10.0 in CDH 5.1. `kite-morphlines` modules are backward-compatible, but this change breaks backward-compatibility for the `kite-data` API.

Kite Data

The Kite data API has had substantial updates since the version included in CDH 5.1.

Changes from 0.15.0

The Kite version in CDH 5.2 is based on 0.15.0, but includes some newer changes. Specifically, it includes support for dataset namespaces, which can be used to set the database in the Hive Metastore.

The introduction of namespaces changed the file system repository layout; now there is an additional namespace directory for datasets stored in HDFS (`repository/namespace/dataset/`). There are no compatibility problems when you use `Dataset` URIs, but all datasets created with the `DatasetRepository` API will be located in a namespace directory. This new directory level is not expected in Kite 0.15.0 or 0.16.0 and will prevent the dataset from being loaded. The work-around is to switch to using `Dataset` URIs (see below) that include the namespace component. Existing datasets will work without modification.

Except as noted above, Kite 0.15.0 in CDH 5.2 is fully backward-compatible. It can load datasets written with any previous Kite version.

Dataset URIs

Datasets are identified with a single URI, rather than a repository URI and dataset name. The dataset URI contains all the information Kite needs to determine which implementation (Hive, HBase, or HDFS) to use for the dataset, and includes both the dataset's name and a namespace.

The Kite API has been updated so that developers call methods in the `Datasets` utility class as they would use `DatasetRepository` methods. The `Datasets` methods are recommended, and the `DatasetRepository` API is deprecated.

Views

The Kite data API now allows you to select a view of the dataset by setting constraints. These constraints are used by Kite to automatically prune unnecessary partitions and filter records.

MapReduce input and output formats

The `kite-data-mapreduce` module has been added. It provides both `DatasetKeyInputFormat` and `DatasetKeyOutputFormat` that allow you to run MapReduce jobs over datasets or views. Spark is also supported by the input and output formats.

Dataset CLI tool

Kite now includes a command-line utility that can run common maintenance tasks, like creating a dataset, migrating a dataset's schema, copying from one dataset to another, and importing CSV data. It also has helpers that can create Avro schemas from data files and other Kite-related configuration.

Flume DatasetSink

The Flume `DatasetSink` has been updated for the `kite-data` API changes. It supports all previous configurations without modification.

In addition, the `DatasetSink` now supports dataset URIs with the configuration option `kite.dataset.uri`.

Apache Mahout

Mahout jobs launched from the `bin/mahout` script will now use cluster's default parameters, rather than hard-coded parameters from the library. This may change the algorithms' run-time behavior, possibly for the better. ([MAHOUT-1565](#).)

Apache Oozie

CDH 5.2 introduces the following important changes:

- A new [Hive 2 Action](#) allows Oozie to run HiveServer2 scripts. Using the Hive Action with HiveServer2 is now deprecated; you should switch to the new Hive 2 Action as soon as possible.
- The MapReduce action can now also be configured by Java code

This gives users the flexibility of using their own driver Java code for configuring the MR job, while also getting the advantages of the MapReduce action (instead of using the Java action). See the [documentation](#) for more info.
- The `PurgeService` is now able to remove completed child jobs from long running coordinator jobs
- `ALL` can now be set for `oozie.service.LiteWorkflowStoreService.user.retry.error.code.ext` to make Oozie retry actions automatically for every type of error
- All Oozie servers in an Oozie HA group now synchronize on the same randomly generated rolling secret for signing auth tokens
- You can now upgrade from CDH 4.x to CDH 5.2 and later with jobs in `RUNNING` and `SUSPENDED` states. (An upgrade from CDH 4.x to a CDH 5.x release *earlier* than CDH 5.2.0 would still require that no jobs be in either of those states).

Apache Parquet (incubating)

CDH 5.2 Parquet is rebased on Parquet 1.5 and Parquet-format 2.1.0.

Cloudera Search

New Features:

- Cloudera Search adds support for Spark indexing using the `CrunchIndexerTool`. For more information, see [Spark Indexing Reference \(CDH 5.2 or later only\)](#).
- Cloudera Search adds fault tolerance for single-shard deployments. This fault tolerance is enabled with a new `-a` option in `solrctl`, which configures shards to automatically be re-added on an existing, healthy node if the node hosting the shard become unavailable.
- Components of Cloudera Search include Kite 0.15.0. This includes all morphlines-related backports of all fixes and features in Kite 0.17.0. For additional information on Kite, see:

Release Notes

- [Kite repository](#)
- [Kite Release Notes](#)
- [Kite documentation](#)
- [Kite examples](#)
- Search adds support for multi-threaded faceting on fields. This enables parallelizing operations, allowing them to run more quickly on highly concurrent hardware. This is especially helpful in cases where faceting operations apply to large datasets over many fields.
- Search adds support for distributed pivot faceting, enabling faceting on multi-shard collections.

Apache Sentry (incubating)

CDH 5.2 introduces the following changes to Sentry.

Sentry Service:

- If you are using the database-backed Sentry service, upgrading from CDH 5.1 to CDH 5.2 will require a schema upgrade.
- **Hive SQL Syntax:**
 - `GRANT` and `REVOKE` statements have been expanded to include `WITH GRANT OPTION`, thus allowing you to delegate granting and revoking privileges.
 - The `SHOW GRANT ROLE` command has been updated to allow non-admin users to list grants for roles that are currently assigned to them.
 - The `SHOW ROLE GRANT GROUP <groupName>` command has been updated to allow non-admin users that are part of the group specified by `<groupName>` to list all roles assigned to this group.

Apache Spark

CDH 5.2 Spark is rebased on Apache Spark/Streaming 1.1 and provides the following new capabilities:

- Stability and performance improvements.
- New sort-based shuffle implementation (disabled by default).
- Better performance monitoring through the Spark UI.
- Support for arbitrary Hadoop `InputFormats` in PySpark.
- Improved Yarn support with several bug fixes.

Apache Sqoop

CDH 5.2 Sqoop 1 is rebased on Sqoop 1.4.5 and includes the following changes:

- Mainframe connector added.
- Parquet support added.

There are no changes for Sqoop 2.

What's New in CDH 5.1.5

This is a maintenance release that fixes some important issues; for details, see [Known Issues Fixed in CDH 5.1.5](#) on page 114.

What's New in CDH 5.1.4

This is a maintenance release that fixes the “POODLE” Apache Hadoop Distributed Cache vulnerabilities described below. All CDH 5.1.x users should upgrade to 5.1.4 as soon as possible.

“POODLE” Vulnerability on SSL/TLS enabled ports

The POODLE (Padding Oracle On Downgraded Legacy Encryption) attack takes advantage of a cryptographic flaw in the obsolete SSLv3 protocol, after first forcing the use of that protocol. The only solution is to disable SSLv3 entirely. This requires changes across a wide variety of components of CDH and Cloudera Manager in all current versions. CDH 5.1.4 provides these changes for CDH 5.1.x deployments.

For more information, see the [Cloudera Security Bulletin](#).

Apache Hadoop Distributed Cache Vulnerability

The Distributed Cache Vulnerability allows a malicious cluster user to expose private files owned by the user running the YARN NodeManager process. For more information, see the [Cloudera Security Bulletin](#).

Other Fixes

CDH 5.1.4 also fixes the following issues:

- [DATAFU-68](#) - SampleByKey can throw NullPointerException
- [HADOOP-11243](#) - SSLFactory shouldn't allow SSLv3
- [HADOOP-11156](#) - DelegateToFileSystem should implement getFsStatus(final Path f).
- [HDFS-7391](#) - Reenable SSLv2Hello in HttpFS
- [HDFS-7235](#) - DataNode#transferBlock should report blocks that don't exist using reportBadBlock
- [HDFS-7274](#) - Disable SSLv3 in HttpFS
- [HDFS-7005](#) - DFS input streams do not timeout
- [HDFS-6376](#) - Distcp data between two HA clusters requires another configuration
- [HDFS-6621](#) - Hadoop Balancer prematurely exits iterations
- [YARN-2273](#) - NPE in ContinuousScheduling thread when we lose a node
- [YARN-2566](#) - DefaultContainerExecutor should pick a working directory randomly
- [YARN-2588](#) - Standby RM does not transitionToActive if previous transitionToActive is failed with ZK exception.
- [YARN-2641](#) - Decommission nodes on -refreshNodes instead of next NM-RM heartbeat
- [YARN-2608](#) - FairScheduler: Potential deadlocks in loading alloc files and clock access
- [HBASE-12376](#) - HBaseAdmin leaks ZK connections if failure starting watchers (ConnectionLossException)
- [HBASE-12366](#) - Add login code to HBase Canary tool
- [HBASE-12098](#) - User granted namespace table create permissions can't create a table
- [HBASE-12087](#) - [0.98] Changing the default setting of hbase.security.access.early_out to true
- [HBASE-11896](#) - LoadIncrementalHFiles fails in secure mode if the namespace is specified
- [HBASE-12054](#) - bad state after NamespaceUpgrade with reserved table names
- [HBASE-12460](#) - Moving Chore to hbase-common module
- [HIVE-5643](#) - ZooKeeperHiveLockManager.getQuorumServers incorrectly appends the custom zk port to quorum hosts
- [HIVE-8675](#) - Increase thrift server protocol test coverage
- [HIVE-8827](#) - Remove SSLv2Hello from list of disabled protocols
- [HIVE-8182](#) - beeline fails when executing multiple-line queries with trailing spaces
- [HIVE-8330](#) - HiveResultSet.findColumn() parameters are case sensitive
- [HIVE-5994](#) - ORC RLEv2 encodes wrongly for large negative BIGINTs (64 bits)
- [HIVE-7629](#) - Problem in SMB Joins between two Parquet tables
- [HIVE-6670](#) - ClassNotFound with Serde
- [HIVE-6409](#) - FileOutputCommitterContainer::commitJob() cancels delegation tokens too early.
- [HIVE-7647](#) - Beeline does not honor --headerInterval and --color when executing with \
- [HIVE-7441](#) - Custom partition scheme gets rewritten with hive scheme upon concatenate
- [HIVE-5871](#) - Use multiple-characters as field delimiter
- [HIVE-1363](#) - SHOW TABLE EXTENDED LIKE command does not strip single/double quotes
- [HIVE-5989](#) - Hive metastore authorization check is not threadsafe
- [HUE-2438](#) - [core] Disable SSLv3 for Poodle vulnerability
- [HUE-2291](#) - [oozie] Faster dashboard display
- [IMPALA-1334](#) - Impala does not map principals to lowercase, affecting Sentry authorisation
- [IMPALA-1251](#) - High-offset queries hang
- [IMPALA-1338](#) - HDFS does not return all ACLs in getAclStatus()
- [IMPALA-1279](#) - Impala does not employ ACLs when checking path permissions for LOAD and INSERT

- [OOZIE-2034](#) - Disable SSLv3 (POODLEbleed vulnerability)
- [OOZIE-2063](#) - Cron syntax creates duplicate actions
- [SENTRY-428](#) - Sentry service should periodically renew the server kerberos ticket
- [SENTRY-431](#) - Sentry db provider client should attempt to refresh kerberos ticket before connection
- [SPARK-3606](#) - Spark-on-Yarn AmlpFilter does not work with Yarn HA

What's New in CDH 5.1.3

This is a maintenance release that fixes the following issues:

- [HADOOP-11035](#) - distcp on mr1(branch-1) fails with NPE using a short relative source path.
- [HBASE-10188](#) - Hide ServerName constructor
- [HBASE-10012](#) - Hide ServerName constructor
- [HBASE-11349](#) - [Thrift] support authentication/impersonation
- [HBASE-11446](#) - Reduce the frequency of RNG calls in SecureWALCellCodec#EncryptedKvEncoder
- [HBASE-11457](#) - Increment HFile block encoding IVs accounting for cipher's internal use
- [HBASE-11474](#) - [Thrift2] support authentication/impersonation
- [HBASE-11565](#) - Stale connection could stay for a while
- [HBASE-11627](#) - RegionSplitter's rollingSplit terminated with "/" by zero", and the _balancedSplit file was not deleted properly
- [HBASE-11788](#) - hbase is not deleting the cell when a Put with a KeyValue, KeyValue.Type.Delete is submitted
- [HBASE-11828](#) - Callers of ServerName.valueOf should use equals and not ==
- [HDFS-4257](#) - The ReplaceDatanodeOnFailure policies could have a forgiving option
- [HDFS-6776](#) - Using distcp to copy data between insecure and secure cluster via webhdfs doesn't work
- [HDFS-6908](#) - Incorrect snapshot directory diff generated by snapshot deletion
- [HUE-2247](#) - [Impala] Support pass-through LDAP authentication
- [HUE-2295](#) - [librdbs] External oracle DB connection is broken due to a typo
- [HUE-2273](#) - [desktop] Blacklisting apps with existing document will break home page
- [HUE-2318](#) - [desktop] Documents shared with write group permissions are not editable
- [HIVE-5087](#) - Rename npath UDF to matchpath
- [HIVE-6820](#) - HiveServer(2) ignores HIVE_OPTS
- [HIVE-7635](#) - Query having same aggregate functions but different case throws IndexOutOfBoundsException
- [IMPALA-958](#) - Excessively long query plan serialization time in FE when querying huge tables
- [IMPALA-1091](#) - Improve TScanRangeLocation struct and associated code
- [OOZIE-1989](#) - NPE during a rerun with forks
- [YARN-1458](#) - FairScheduler: Zero weight can lead to livelock

What's New in CDH 5.1.2

▪ **Note:**

There is no CDH 5.1.1 release. This skip in the CDH 5.x sequence allows the CDH and CM components of Cloudera Enterprise 5.1.2 to have consistent numbering.

This is a maintenance release that fixes the following issues:

- [FLUME-2438](#)
- [HBASE-11052](#)
- [HBASE-11143](#)
- [HBASE-11609](#)
- [HDFS-6114](#)
- [HDFS-6640](#)
- [HDFS-6703](#)

- [HDFS-6788](#)
- [HDFS-6825](#)
- [HUE-2211](#)
- [HUE-2223](#)
- [HUE-2232](#)
- [HIVE-5515](#)
- [HIVE-6495](#)
- [HIVE-7450](#)
- [IMPALA-1093](#)
- [IMPALA-1107](#)
- [IMPALA-1131](#)
- [IMPALA-1142](#)
- [IMPALA-1149](#)
- [MAPREDUCE-5966](#)
- [MAPREDUCE-5979](#)
- [MAPREDUCE-6012](#)
- [OOZIE-1920](#)
- [PARQUET-19](#)
- [SENTRY-363](#)
- [YARN-2273](#)
- [YARN-2274](#)
- [YARN-2313](#)
- [YARN-2352](#)
- [YARN-2359](#)

What's New in CDH 5.1.0

Operating System Support

CDH 5.1 adds support for version 6.5 of RHEL and related platforms.

Apache Crunch

- CDH 5.1.0 implements Crunch 0.10.0.

Apache Flume

- CDH 5.1.0 implements Flume 1.5.0.

Apache Hadoop

HDFS

POSIX Access Control Lists: As of CDH 5.1, HDFS supports POSIX Access Control Lists (ACLs), an addition to the traditional POSIX permissions model already supported. ACLs provide fine-grained control of permissions for HDFS files by providing a way to set different permissions for specific named users or named groups.

NFS Gateway Improvements: CDH 5.1 makes the following improvements to the HDFS NFS gateway capability:

- Subdirectory mounts :
 - Previously, clients could mount only the HDFS root directory.
 - As of CDH 5.1, a single mount point, configured via the `nfs.export.point` property in `hdfs-site.xml` on the NFS gateway node, is available to clients.
- Improved support for Kerberized clusters ([HDFS-5898](#)):
 - Previously the NFS Gateway could connect to a secure cluster, but didn't support logging in from a keytab.
 - As of CDH 5.1, set the `nfs.kerberos.principal` and `nfs.keytab.file` properties in `hdfs-site.xml` to allow users to log in from a keytab.

- Support for port monitoring ([HDFS-6406](#)):
 - Previously, the NFS Gateway would always accept connections from any client.
 - As of CDH 5.1, set `nfs.port.monitoring.disabled` to `false` in `hdfs-site.xml` to allow connections only from privileged ports (those with root access).
- Static uid/gid mapping for NFS clients that are not in synch with the NFS Gateway ([HDFS-6435](#)):
 - NFS sends UIDs and GIDs over the network from client to server, meaning that the UIDs and GIDs must be in synch between clients and server machines in order for users and groups to be set appropriately for file access and file creation; this is usually but not always the case.
 - As of CDH 5.1, you can configure a static UID/GID mapping file, by default `/etc/nfs.map`.
 - You can change the default (to use a different file path) by means of the `nfs.static.mapping.file` property in `hdfs-site.xml`.
 - The following sample entries illustrate the format of the file:

```
uid 10 100 # Map the remote UID 10 the local UID 100
gid 11 101 # Map the remote GID 11 to the local GID 101
```

- Hadoop portmap, or insecure system portmap, no longer required:
 - Many supported OS have portmap bugs detailed [here](#).
 - CDH 5.1 allows you to circumvent the problems by starting the NFS gateway as root, whether you install CDH from packages or parcels.

▪ **Note:**

After initially registering with the system portmap as root, the NFS Gateway drops privileges and runs as a regular user.

- Cloudera Manager starts the gateway as root by default.

- Support for AIX NFS clients ([HDFS-6549](#)):
 - To deploy AIX NFS clients, set `nfs.aix.compatibility.mode.enabled` to `true` in `hdfs-site.xml`.
 - This enables code that handles bugs in the AIX implementation of NFS.

MapReduce and YARN

YARN with Impala supports Dynamic Prioritization.

Apache HBase

- CDH 5.1.0 implements HBase 0.98.
- As of CDH 5.1.0, HBase fully supports BucketCache, which was introduced as an experimental feature in CDH 5 Beta 1.
- HBase now supports access control for `EXEC` permissions.
- CDH 5.1.0 HBase introduces a reverse scan API; allowing you to scan a table in reverse.
- You can now run a MapReduce job over a snapshot from HBase, rather than being limited to live data.
- A new stateless streaming scanner is available over the REST API.
- The `delete*` methods of the Delete class of the HBase Client API now use the timestamp from the constructor, the same behavior as the Put class. (In HBase versions before CDH 5.1, the `delete*` methods ignored the constructor's timestamp, and used the value of `HConstants.LATEST_TIMESTAMP`. This behavior was different from the behavior of the `add()` methods of the Put class.)
- The `SnapshotInfo` tool has been enhanced in the following ways:
 - A new option, `-list-snapshots`, has been added to the `SnapshotInfo` command. This option allows you to list snapshots on either a local or remote server.

- You can now pass the `-size-in-bytes` flag to print the size of snapshot files in bytes rather than the default human-readable format.
- The size of each snapshot file in bytes is checked against the size reported in the manifest, and if the two sizes differ, the tool reports the file as corrupt.
- A new `-target` option for `ExportSnapshot` allows you to specify a different name for the target cluster from the snapshot name on the source cluster.

In addition, Cloudera has fixed some binary incompatibilities between HBase 0.96 and 0.98. As a result, the incompatibilities introduced by [HBASE-10452](#) and [HBASE-10339](#) *do not* affect CDH 5.1 HBase, as explained below:

- HBASE-10452 introduced a new exception and error message in `setTimeStamp()`, for an extremely unlikely event when where getting a `TimeRange` could fail because of an integer overflow. CDH 5.1 suppresses the new exception to retain compatibility with HBase 0.96, but logs the error.
- HBASE-10339 contained code which inadvertently changed the signatures of the `getFamilyMap` method. CDH 5.1 restores these signatures to those used in HBase 0.96, to retain compatibility.

Apache Hive

- Permission inheritance fixes
- Support for decimal computation, and for reading and writing decimal-format data from and to Parquet and Avro

Hue

CDH 5.1.0 implements Hue 3.6.

New Features:

- Search App v2:
 - 100% Dynamic dashboard
 - Drag-and-Drop dashboard builder
 - Text, Timeline, Pie, Line, Bar, Map, Filters, Grid and HTML widgets
 - Solr Index creation wizard (from a file)
- Ability to view compressed Snappy, Avro and Parquet files
- Impala HA
- Close Impala and Hive sessions queries and commands

Apache Mahout

- CDH 5.1.0 implements Mahout 0.9.

See also [Apache Mahout Incompatible Changes](#) on page 65.

Apache Oozie

- You can now submit Sqoop jobs from the Oozie command line.
- `LAST_ONLY` execution mode now works correctly ([OOZIE-1319](#)).

Cloudera Search

New Features:

- A Quick Start script that automates using Search to query data from the Enron Email dataset. The script downloads the data, expands it, moves it to HDFS, indexes, and pushes the results live. The documentation now also includes a companion quick start guide, which describes the tasks the script completes, as well as customization options.
- `solrctl` now has built-in support for schema-less Solr.
- Sentry-based document-level security for role-based access control of a collection. Document-level access control associates authorization tokens with each document in the collection, enabling granting Sentry roles access to sets of documents in a collection.

Release Notes

- Cloudera Search includes a version of Kite 0.10.0, which includes all morphlines-related backports of all fixes and features in Kite 0.15.0. For additional information on Kite, see:
 - [Kite repository](#)
 - [Kite Release Notes](#)
 - [Kite documentation](#)
 - [Kite examples](#)
- Support for the Parquet file format is included with this version of Kite 0.10.0.
- Inclusion of hbase-indexer-1.5.1, a new version of the Lily HBase Indexer. This new version of the indexer includes the 0.10.0 version of Kite mentioned above. This 0.10.0 version of Kite includes the backports and fixes included in Kite 0.15.0.

Apache Sentry (incubating)

- CDH 5.1.0 implements Sentry 1.2. This includes a database-backed Sentry service which uses the more traditional GRANT/REVOKE statements instead of the previous policy file approach making it easier to maintain and modify privileges.
- Revised authorization privilege model for Hive and Impala.

Apache Spark

- CDH 5.1.0 implements Spark 1.0.
- The `spark-submit` command abstracts across the variety of deployment modes that Spark supports and takes care of assembling the classpath for you.
- Application History Server (SparkHistoryServer) improves monitoring capabilities.
- You can launch PySpark applications against YARN clusters. PySpark currently only works in YARN Client mode.

Other improvements include:

- Streaming integration with Kerberos
- Addition of more algorithms to MLlib (Sparse Vector Support)
- Improvements to Avro integration
- Spark SQL alpha release (new SQL engine). Spark SQL allows you to run SQL statements inside a Spark application that manipulate and produce RDDs.

■ **Note:**

Because of its immaturity and alpha status, Cloudera does not currently offer commercial support for Spark SQL, but bundles it with our distribution so that you can try it out.

- Authentication of all Spark communications

What's New in CDH 5.0.6

This is a maintenance release that fixes some important issues; for details, see [Known Issues Fixed in CDH 5.0.6](#) on page 117.

What's New in CDH 5.0.5

This is a maintenance release that fixes the “POODLE” and Apache Hadoop Distributed Cache vulnerabilities described below. All CDH 5.0.x users should upgrade to 5.0.5 as soon as possible.

“POODLE” Vulnerability on SSL/TLS enabled ports

The POODLE (Padding Oracle On Downgraded Legacy Encryption) attack takes advantage of a cryptographic flaw in the obsolete SSLv3 protocol, after first forcing the use of that protocol. The only solution is to disable SSLv3 entirely. This requires changes across a wide variety of components of CDH and Cloudera Manager in all current versions. CDH 5.0.5 provides these changes for CDH 5.0.x deployments.

For more information, see the [Cloudera Security Bulletin](#).

Apache Hadoop Distributed Cache Vulnerability

The Distributed Cache Vulnerability allows a malicious cluster user to expose private files owned by the user running the YARN NodeManager process. For more information, see the [Cloudera Security Bulletin](#).

Other Fixes

- [HADOOP-11243](#) - SSLFactory shouldn't allow SSLv3
- [HDFS-7274](#) - Disable SSLv3 in HttpFS
- [HDFS-7391](#) - Reenable SSLv2Hello in HttpFS
- [HBASE-12376](#) - HBaseAdmin leaks ZK connections if failure starting watchers (ConnectionLossException)
- [HIVE-8675](#) - Increase thrift server protocol test coverage
- [HIVE-8827](#) - Remove SSLv2Hello from list of disabled protocols protocols
- [HUE-2438](#) - [core] Disable SSLv3 for Poodle vulnerability
- [OOZIE-2034](#) - Disable SSLv3 (POODLEbleed vulnerability)
- [OOZIE-2063](#) - Cron syntax creates duplicate actions

What's New in CDH 5.0.4

This is a maintenance release that fixes the following issues:

- [FLUME-2438](#)
- [HBASE-11609](#)
- [HDFS-6044](#)
- [HDFS-6529](#)
- [HDFS-6618](#)
- [HDFS-6622](#)
- [HDFS-6640](#)
- [HDFS-6647](#)
- [HDFS-6703](#)
- [HDFS-6788](#)
- [HIVE-5515](#)
- [HIVE-7459](#)
- [HUE-2166](#)
- [HUE-2249](#)
- [IMPALA-1019](#)
- [MAPREDUCE-5966](#)
- [MAPREDUCE-5979](#)
- [OOZIE-1920](#)
- [PARQUET-19](#)
- [SPARK-1930](#)
- [YARN-1550](#)
- [YARN-2061](#)
- [YARN-2132](#)

What's New in CDH 5.0.3

This is a maintenance release that fixes the following issues:

- [FLUME-2245](#)
- [FLUME-2416](#)
- [HBASE-10871](#)
- [HDFS-5891](#)
- [HDFS-6021](#)

- [HDFS-6077](#)
- [HDFS-6340](#)
- [HDFS-6475](#)
- [HDFS-6510](#)
- [HDFS-6527](#)
- [HDFS-6563](#)
- [HUE-1928](#)
- [HUE-2184](#)
- [HUE-2085](#)
- [HUE-2192](#)
- [HUE-2193](#)
- [OOZIE-1621](#)
- [OOZIE-1890](#)
- [OOZIE-1907](#)
- [SOLR-5593](#)
- [SOLR-5915](#)
- [SOLR-6161](#)
- [YARN-1550](#)
- [YARN-2155](#)

What's New in CDH 5.0.2

This is a maintenance release that fixes the following issues:

- [HADOOP-10556](#)
- [HADOOP-10638](#)
- [HADOOP-10639](#)
- [HADOOP-10658](#)
- [HBASE-6690](#)
- [HBASE-10312](#)
- [HBASE-10371](#)
- [HDFS-6326](#)
- [HIVE-5380](#)
- [HIVE-6913](#)
- [PIG-3677](#)
- [YARN-2073](#)

What's New in CDH 5.0.1

This is a maintenance release that fixes the following issues:

- [HADOOP-10442](#) - Group look-up can cause segmentation fault when a certain JNI-based mapping module is used.
- [HADOOP-10456](#) - Bug in `Configuration.java` exposed by Spark (`ConcurrentModificationException`)
- [HDFS-5064](#) - Standby checkpoints should not block concurrent readers
- [HDFS-6039](#) - Uploading a File under a Dir with default ACLs throws "Duplicated ACLFeature"
- [HDFS-6094](#) - The same block can be counted twice towards safe mode threshold
- [HDFS-6231](#) - `DFSClient` hangs infinitely if using hedged reads and all eligible `DataNodes` die
- [HIVE-6495](#) - `TableDesc.getDeserializer()` should use correct classloader when calling `Class.forName()`
- [HIVE-6575](#) - `select *` fails on parquet table with map data type
- [HIVE-6648](#) - Fixed permission inheritance for multi-partitioned tables
- [HIVE-6740](#) - Fixed addition of Avro JARs to classpath
- [HUE-2061](#) - Task logs are not retrieved if containers not on the same host

- [OOZIE-1794](#) - java-opts and java-opt in the Java action don't always work properly in YARN
- [SOLR-5608](#) - Frequently reproducible failures in CollectionsAPIDistributedZkTest#testDistribSearch
- [YARN-1924](#) - STATE_STORE_OP_FAILED happens when ZKRMStateStore tries to update app(attempt) before storing it

Enabling SSL in CDH 5: Enabling HTTPS communication in CDH 5 requires extra configuration properties to be added to [YARN](#) (yarn-site.xml and mapred-site.xml) and [HDFS](#) (hdfs-site.xml), in addition to the existing configuration settings described [here](#).

What's New in CDH 5.0.0

Apache Hadoop

HDFS

New Features:

- [HDFS-5776](#)- Hedged reads in HDFS for improved HBase MTTR.
- [HDFS-4685](#)- Implementation of extended file access control lists in HDFS.

Notable Bug Fixes:

- [HDFS-5339](#) - WebHDFS URI does not accept logical nameservices when security is enabled.
- [HDFS-5898](#) - Allow NFS gateway to login/relogin from its Kerberos keytab.
- [HDFS-5921](#) - "Browse filesystem" on the Namenode UI doesn't work if any directory has the sticky bit set.
- HDFS and Hive replication between different Kerberos realms now works.
- [HDFS-5922](#) - DataNode heartbeat thread can get stuck in a tight loop.

MapReduce & YARN

New Feature:

- FairScheduler supports moving running applications between queries.

Notable Bug Fixes:

- Several critical fixes to stabilize ResourceManager HA - Web UI, unmanaged ApplicationMasters and secure-cluster support.
- Support for large values of mapreduce.task.io.sort.mb.
- JobHistory Server has information on failed MapReduce jobs.

Apache HBase

New Features:

- [HBASE-10436](#)- Restore RegionServer lists removed from HBase 0.96.0 JMX.

Many of the metrics exposed in CDH 4/0.94 were removed with the refactorization of metrics in CDH 5/0.96. This patch restores the availability of the lists of live and dead RegionServers. In 0.94 this was a large nested structure as shown below, which included the RegionServer lists and metrics from each region.

```
{
  "name" : "hadoop:service=Master,name=Master",
  "modelerType" : "org.apache.hadoop.hbase.master.MXBeanImpl",
  "ZookeeperQuorum" : "localhost:2181",
  ....
  "RegionsInTransition" : [ ],
  "RegionServers" : [ {
    "key" : "localhost,48346,1390857257246",
    "value" : {
      "load" : 2,
    },
  },
  ....
}
```

CDH 5 Beta 1 and Beta 2 did not contain this list; they only displayed counts of the number of live and dead RegionServers. As of CDH 5.0.0, this list is now presented in a semi-colon separated field as follows:

```
{
  "name" : "Hadoop:service=HBase,name=Master,sub=Server",
  "modelerType" : "Master,sub=Server",
  "tag.Context" : "master",
  "tag.liveRegionServers" : "localhost,56196,1391992019130",
  "tag.deadRegionServers" :

  "localhost,40010,1391035309673;localhost,41408,1391990380724;localhost,38682,1390950017735",
  ...
}
```

- Assorted usability and compatibility improvements as well as improvements to exporting snapshots.

Apache Flume

New Feature:

- The HBase Sink now supports coalescing multiple Increment RPCs into one ([FLUME-2338](#)).

Changed Behavior:

- File Channel Write timeout has been removed and the configuration parameter is now ignored ([FLUME-2307](#)).
- Syslog UDP source can now accept larger messages ([FLUME-2130](#)).
- AsyncHBase Sink is now fully functional ([FLUME-2334](#)).
- Use standard lookup to find queue/topic in JMS Source ([FLUME-2311](#)).

Notable Bug Fixes:

- Deadlock fixed in Dataset sink ([FLUME-2320](#)).
- FileChannel Dual Checkpoint Backup Thread is now released on application stop ([FLUME-2328](#)).
- Spool Dir source now checks interrupt flag before writing to channel ([FLUME-2283](#)).
- Morphline sink increments `eventDrainAttemptCount` when it takes event from channel ([FLUME-2323](#)).
- Bucketwriter now permanently closed only on idle and roll timeouts ([FLUME-2325](#)).
- `BucketWriter#close` now cancels `idleFuture` ([FLUME-2305](#)).

Apache Oozie

As of CDH 5.0.0 Oozie includes a glob pattern feature ([OOZIE-1471](#)), allowing you do a move of wild cards in the FS Action. For example:

```
<fs name="archive-files">
  <move source="hdfs://namenode/output/*"
  target="hdfs://namenode/archive" />
  <ok to="next"/>
  <error to="fail"/>
</fs>
```

By default, up to 1000 files can be matched; you can change this default by means of the `oozie.action.fs.glob.max` parameter.

Cloudera Search

- Cloudera Search includes a version of Kite 0.10.0, which includes backports of all fixes and features in Kite 0.12.0. For additional information on Kite, see:
 - [Kite repository](#)
 - [Kite Release Notes](#)
 - [Kite documentation](#)
 - [Kite examples](#)

What's New in CDH 5 Beta 2

Apache Crunch

The Apache Crunch™ project develops and supports Java APIs that simplify the process of creating data pipelines on top of Apache Hadoop. The Crunch APIs are modeled after FlumeJava (PDF), which is the library that Google uses for building data pipelines on top of their own implementation of MapReduce. For more information and installation instructions, see [Crunch Installation](#).

Apache DataFu

- Upgraded from version 0.4 to 1.1.0 (this upgrade is not backward compatible).
- New features include UDFS SHA, SimpleRandomSample, COALESCE, ReservoirSample, EmptyBagToNullFields, and many others.

Apache Flume

- [FLUME-2294](#) - Added a new sink to write Kite datasets.
- [FLUME-2056](#) - Spooling Directory Source can now only pass the name of the file in the event headers.
- [FLUME-2155](#) - File Channel is indexed during replay to improve replay performance for faster startup.
- [FLUME-2217](#) - Syslog Sources can optionally preserve all syslog headers in the message body.
- [FLUME-2052](#) - Spooling Directory Source can now replace or ignore malformed characters in input files.

Apache Hadoop

HDFS

New Features/Improvements:

- As of CDH 5 Beta 2, you can upgrade HDFS with high availability (HA) enabled, if you are using Quorum-based storage. (Quorum-based storage is the only method available in CDH 5; NFS shared storage is not supported.) For upgrade instructions, see [Upgrading from CDH 4 to CDH 5](#).
- [HDFS-4949](#) - CDH 5 Beta 2 supports [Configuring Centralized Cache Management in HDFS](#).
- As of CDH 5 Beta 2, you can configure an NFSv3 gateway that allows any NFSv3-compatible client to mount HDFS as a file system on the client's local file system. For more information and instructions, see [Configuring an NFSv3 Gateway](#).
- [HDFS-5709](#) - Improve upgrade with existing files and directories named `.snapshot`.

Major Bug Fixes:

- [HDFS-5449](#) - Fix WebHDFS compatibility break.
- [HDFS-5671](#) - Fix socket leak in `DFSInputStream#getBlockReader`.
- [HDFS-5353](#) - Short circuit reads fail when `dfs.encrypt.data.transfer` is enabled.
- [HDFS-5438](#) - Flaws in block report processing can cause data loss.

Changed Behavior:

- As of CDH 5 Beta 2, in order for the NameNode to start up on a secure cluster, you should have the `dfs.web.authentication.kerberos.principal` property defined in `hdfs-site.xml`. This has been documented in the [CDH 5 Security Guide](#). For clusters managed by Cloudera Manager, you do not need to explicitly define this property.
- [HDFS-5037](#) - Active NameNode should trigger its own edit log rolls. Clients will now retry for a configurable period when encountering a NameNode in Safe Mode.
- The default behavior of the `mkdir` command has changed. As of CDH 5 Beta 2, if the parent folder does not exist, the `-p` switch must be explicitly mentioned otherwise the command fails.

MapReduce (MRv1 and YARN)

- Fair Scheduler (in YARN and MRv1) now supports advance configuration to automatically place applications in queues.
- MapReduce now supports running multiple reducers in `uber` mode and in local job runner.

Apache HBase

- **Online Schema Change** is now a supported feature.
- **Online Region Merge** is now a supported feature.
- **Namespaces:** CDH 5 Beta 2 includes the namespaces feature which enables different sets of tables to be administered by different administrative users. All upgraded tables will live in the default "hbase" namespace. Administrators may create new namespaces and create tables users with rights to the namespace may administer permissions on the tables within the namespace.
- There have been several improvements to HBase's **mean time to recovery** (mttr) in the face of Master or RegionServer failures.
 - Distributed log splitting has matured, and is always activated. The option to use the old slower splitting mechanism no longer exists.
 - Failure detection time has been improved. New notifications are now sent when RegionServers or Masters fail which triggers corrective action quickly.
 - The Meta table has a dedicated write ahead log which enables faster recovery region recovery if the RegionServer serving meta goes down.
- The **Region Balancer** has been significantly updated to take more load attributes into account.
- Added **TableSnapshotInputFormat** and **TableSnapshotScanner** to perform scans over HBase table snapshots from the client side, bypassing the HBase servers. The former configures a MapReduce job, while the latter does a single client-side scan over snapshot files. Can also be used with offline HBase with in-place or exported snapshot files.
- The KeyValue API has been deprecated for applications in favor of the **Cell** interface. Users upgrading to HBase 0.96 may still use KeyValue by future upgrades may remove the class or parts of its functionality. Users are encouraged to update their applications to use the new Cell interface.
- Currently Experimental features:
 - **Distributed log replay:** This mechanism allows for faster recovery from RegionServer failures but has one special case where it will violate ACID guarantees. Cloudera does not currently recommend activating this feature.
 - **Bucket cache:** This is an offheap caching mechanism that use extra RAM and block devices (such as flash drives) to greatly increase the read caching capabilities provided by the BlockCache. Cloudera does not currently recommend activating this feature.
 - **Favored nodes:** This feature enables HBase to better control where its data is written to in HDFS in order to better preserve performance after a failure. This is disabled currently because it doesn't interact well with the HBase Balancer or HDFS Balancer. Cloudera does not currently recommend activating this feature.

See this [blog post](#) for more details.

Apache Hive

New Features:

- Improved JDBC specification coverage:
 - Improvements to `getDatabaseMajorVersion()`, `getDatabaseMinorVersion()` APIs ([HIVE-3181](#))
 - Added JDBC support for new datatypes: Char ([HIVE-5683](#)), Decimal ([HIVE-5355](#)) and Varchar ([HIVE-5209](#))
 - You can now specify the database for a session in the HiveServer2 connection URL ([HIVE-4256](#))
- Encrypted communication between the Hive Server and Clients. This includes SSL encryption for non-Kerberos connections to HiveServer2 ([HIVE-5351](#)).
- A native Parquet SerDe is now available as part of the CDH 5 Beta 2 package. Users can directly create a Parquet format table without any external package dependency.

Changed Behavior:

- [HIVE-4256](#) - With Sentry enabled, the `use <database>` command is now executed as part of the connection to HiveServer2. Hence, a user with no privileges to access a database will not be allowed to connect to HiveServer2.

Hue

- Hue has been upgraded to version 3.5.0.
- Impala and Hive Editor are now one-page apps. The Editor, Progress, Table list and Results are all on the same page
- Result graphing for the Hive and Impala Editors.
- Editor and Dashboard for Oozie SLA, crontab and credentials.
- The Sqoop2 app supports autocomplete of database and table names/fields.
- [DBQuery App](#): MySQL and PostgreSQL Query Editors.
- New Search feature: [Graphical facets](#)
- Integrate external Web applications in any language. See this [blog post](#) for more details.
- Create Hive tables and load quoted CSV data. Tutorial available [here](#).
- Submit any Oozie jobs directly from HDFS. Tutorial available [here](#)
- New [SAML backend](#) enables single sign-on (SSO) with Hue.

Apache Oozie

- Oozie now supports cron-style scheduling capability.
- Oozie now supports High Availability with security.

Apache Pig

- AvroStorage rewritten for better performance, and moved from piggybank to core Pig
- ASSERT, IN, and CASE operators added
- ParquetStorage added for integration with Parquet

Cloudera Search

- The Cloudera CDK has been renamed and updated to Kite version 0.11.0. For additional information on Kite, see:
 - [Kite repository](#)
 - [Kite Release Notes](#)
 - [Kite documentation](#)
 - [Kite examples](#)

Apache Spark (incubating)

Spark is a fast, general engine for large-scale data processing. For installation and configuration instructions, see [Spark Installation](#).

Apache Sqoop

Sqoop 2 has been upgraded from version 1.99.2 to 1.99.3.

What's New in CDH 5 Beta 1

- [Oracle JDK 7 Support](#) on page 30
- [Apache Flume](#) on page 30
- [Apache HBase](#) on page 31
- [Apache HDFS](#)
- [Hue](#)
- [Apache Hive and HCatalog](#) on page 34
- [Cloudera Impala](#) on page 34
- [Llama](#) on page 34
- [Apache Mahout](#) on page 34
- [MapReduce v2 \(YARN\)](#) on page 30
- [Apache Oozie](#) on page 35
- [Cloudera Search](#) on page 35

- [Apache Sentry \(incubating\)](#) on page 35
- [Apache Sqoop](#) on page 35

Oracle JDK 7 Support

- CDH 5 supports Oracle JDK 1.7 and supports users running applications compiled with JDK 1.7. For CDH 5 Beta 1 the certified version is JDK 1.7.0_25. Cloudera has tested this version across all components.
- CDH 5 does not support JDK 1.6; you must install JDK 1.7, as instructed [here](#).

Apache Flume

New Features:

- [FLUME-2190](#) - Includes a new Twitter Source that feeds off the Twitter firehose
- [FLUME-2109](#) - HTTP Source now supports HTTPS
- Flume now auto-detects Cloudera Search dependencies.

Apache Hadoop HDFS

New Features:

- [HDFS-4953](#): Enable HDFS local reads via mmap.
- [HDFS-2802](#): Support for RW/RO snapshots in HDFS. See: `hadoop-hdfs-project/hadoop-hdfs/src/site/apt/HdfsNfsGateway.appt.vm`
- [HDFS-4750](#): Support NFSv3 interface to HDFS.
- [HDFS-4817](#): Make HDFS advisory caching configurable on a per-file basis.
- [HDFS-3601](#): Add BlockPlacementPolicyWithNodeGroup to support block placement with 4-layer network topology.
- [HDFS-5122](#): Support failover and retry in WebHdfsFileSystem for NN HA.
- [HDFS-4772](#) / [HDFS-5043](#): Add number of children (of a directory) in HdfsFileStatus.
- [HDFS-4434](#): Provide a mapping from INodeId to INode. See: `/.reserved/.inodes/<INODE_NUMBER>`
- [HDFS-2576](#): Enhances the DistributedFileSystem's Create API so that clients can specify favored DataNodes for a file's blocks.

Changed Features:

- [HDFS-4659](#): Support setting execution bit for regular files.
 - **Impact:** In CDH 5, files copied out of `copyToLocal` may now have the executable bit set if it was set when they were created or copied into HDFS.
- [HDFS-4594](#): WebHDFS open sets Content-Length header to what is specified by length parameter rather than how much data is actually returned.
 - **Impact:** In CDH 5, Content-Length header will contain the number of bytes actually returned, rather than the request length.

Changed Behavior:

- [HDFS-4645](#): Move from randomly generated block ID to sequentially generated block ID.
- [HDFS-4451](#): HDFS balancer command returns exit code 1 on success instead of 0.

MapReduce v2 (YARN)

New Features:

- **ResourceManager High Availability:** YARN now allows you to use multiple ResourceManagers so that there is no single point of failure. In-flight jobs are recovered without re-running completed tasks.
- Monitoring and enforcing memory and CPU-based resource utilization using `cgroups`.
- **Continuous Scheduling:** This feature decouples scheduling from the node heartbeats for improved performance in large clusters.

Changed Feature:

- **ResourceManager Restart:** Persistent implementations of the RMStateStore (filesystem-based and ZooKeeper-based) allow recovery of in-flight jobs.

Apache HBase

Summary of New Features

- Support for Hadoop 2.0
- Improved MTTR (meta first recovery, distributed log replay)
- Improved compatibility and upgradeability (ProtoBuf serialization format)
- Namespaces added for administrative domains
- Snapshots (ported to 0.94 / CDH4.2)
- Online region merge mechanisms added
- Major security and functional improvements made for the REST proxy server

Administrative Features

ProtoBuf: All of the serialization that goes across the wire between servers written to and read by HBase file formats have been converted to extensible Protobuf encodings. This breaks compatibility with previous versions but should make future extensions less likely to break compatibility in these areas. This feature is enabled by default.

- [HBASE-5305](#): Improve cross-version compatibility and upgradeability.
- [HBASE-7898](#): Serializing cells over RPC.

Namespaces: Namespaces is a new feature that groups tables into different administrative domains. An admin can be only given rights to act upon a particular namespace. This feature is enabled by default and requires file system layout changes that must be completed during upgrade.

- [HBASE-8015](#): Added support for namespaces.

MTTR Improvements: Mean time to recovery has greatly improved.

- [HBASE-7590](#): "Costless" notifications from master to rs/clients.
- [HBASE-7213](#) / [HBASE-8631](#): New `.meta` suffix to separate HLog file / Recover Meta before other regions in case of server crash.
- [HBASE-7006](#): Distributed log replay (Caveat).
- [HBASE-9116](#): Adds a view/edit tool for favored node mappings for regions (incomplete, likely a dot version).

Metrics: There are several new metrics and a new naming convention for metrics in HBase. This also includes metrics for each region.

- [HBASE-3614](#): Per region metrics.
- [HBASE-4050](#): Rationalize metrics; Update HBase metrics framework to metrics2.

Miscellaneous:

- [HBASE-7403](#): HBase online region merge.
- Shell improvements; tables list to be more well-rounded.
- [HBASE-5953](#): Expose the current state of the balancerSwitch.
- [HBASE-5934](#): Add the ability for Performance Evaluation to set table compression.
- [HBASE-6135](#): New Web UI.
- [HBASE-8148](#): Allow IPC to bind to a specific address (also 0.94.7)
- [HBASE-5498](#): Secure Bulk Load (also 0.94.5)

Backup and Disaster Recovery Features

Replication: Several critical bug fixes.

Release Notes

- [HBASE-9373](#): Replication has been hardened.
- [HBASE-9158](#): Serious bug in cyclic replication.
- [HBASE-8737](#): Changes to the replication RPC to use cell blocks.

Snapshots: HBase table snapshots were backported to 0.94.x. There are some incompatibilities between the implementation released in CDH 4 with that in CDH 5.

- [HBASE-7290](#): Online snapshots (backported to 0.94.x).
- [HBASE-8352](#): Rename snapshots folder from `.snapshot` to `.hbase-snapshots` (Incompatible change).

Copy table:

- [HBASE-8609](#): Add startRow-stopRow options to the CopyTable.

Import:

- [HBASE-7702](#): Add filtering to import jobs.

HBase Proxies

The REST server now supports Hadoop authentication and authorization mechanisms. The Avro gateway has been removed while the Thrift2 proxy has made progress but is not complete. However, it has been included as a preview feature.

REST:

- [HBASE-9347](#): Support for specifying filter in REST server requests.
- [HBASE-7803](#): Support caching on scan.
- [HBASE-7757](#): Add Web UI for Thrift and REST servers.
- [HBASE-5050](#): SPNEGO-based authentication.
- [HBASE-8661](#): Support REST over HTTPS.
- [HBASE-8662](#): Support for impersonation.
- [HBASE-7986](#): [REST] Make HTablePool size configurable.

Thrift:

- [HBASE-5879](#): Enable JMX metrics collection for the Thrift proxy.

Thrift2: Ongoing efforts to match Thrift and REST functionality. (Incomplete, only a preview feature)

Avro:

- [HBASE-5948](#): Avro gateway removed.

Stability Features

There have been several bug fixes, test fixes and configuration default changes that greatly increase our confidence in the stability of the 0.96.0 release. The main improvement comes from the use of a systematic fault-injection framework.

- [HBASE-7721](#) Atomic multi-row mutations in META
- Integration testing
- [HBASE-7977](#) - TableLocks
- [HBASE-7898](#) many flaky tests hardened

Performance Features

Several features have been added to improve throughput and performance characteristics of HBase and its clients.

- **Warning:**

Currently the 0.95.2/CDH 5 beta 1 release will suffer performance degradation when over 40 nodes are used when compared to CDH 4.

Throughput:

- [HBASE-4676](#): Prefix compression / tree encoding.
- [HBASE-8334](#): Essential column families on by default (filtering optimization).
- [HBASE-5074](#) / [HBASE-8322](#): Re-enable HBase checksums by default.
- [HBASE-6466](#): Enable multi-threaded memstore flush
- [HBASE-6783](#): Make short circuit read the default.

Predictable Performance:

- [HBASE-5959](#): Added a Stochastic LoadBalancer
- [HBASE-7842](#): Exploring compactor.
- [HBASE-7236](#): Add per-table/per-cf configuration via metadata
- [HBASE-8163](#): MemStoreChunkPool: Improvement for Java GC
- [HBASE-4391](#)[HBASE-6567](#): Mlock / Memory locking improvements (less disk swap).
- [HBASE-4391](#): Bucket cache (untested)

Miscellaneous:

- [HBASE-6870](#): Improvement to HTable coprocessorExec scan performance.

Developer Features

These features are to aid application developers or for major changes that will enable future minor version improvements.

- [HBASE-9121](#): HTrace updates.
- [HBASE-8375](#): Durability setting per table.
 - [HBASE-7801](#) Deferred sync for WAL logs (0.94.7 and later)
- [HBASE-7897](#): Tags supported in cell interface (for future security features).
- [HBASE-5937](#): Refactor HLog into interface (allows for new HLogs in 0.96.x).
- [HBASE-4336](#): Modularization of POM / Multiple jars (many follow-ons, [HBASE-7898](#)).
- [HBASE-8224](#): Publish -hadoop1 and -hadoop2 versioned jars to Maven (CDH published jars are assumed -hadoop2).
- [HBASE-9164](#): Move towards Cell interface in client instead of KeyValue.
- [HBASE-7898](#): Serializing cells over RPC.
- [HBASE-7725](#): Add ability to create custom compaction request.

Hue

New Features:

- With the Sqoop 2 application, data from databases can be easily exported or imported into HDFS in a scalable manner. The Job Wizard hides the complexity of creating Sqoop jobs and the dashboard offers live progress and log access.
- Zookeeper App: Navigate and browse the Znode hierarchy and content of a Zookeeper cluster. Znodes can be added, deleted and edited. Multi-clusters are supported and various statistics are available for them.
- The Hue Shell application has been removed and replaced by the Pig Editor, HBase Browser and the Sqoop 1 apps.
- Python 2.6 is required.
- Beeswax daemon has been replaced by HiveServer2.
- CDH 5 Hue will only work with HiveServer2 from CDH 5. No support for impersonation.

Hue also includes the following changed features (Updated to upstream version 3.0.0):

- [\[HUE-897\]](#) - [core] Redesign of the overall layout
- [\[HUE-1521\]](#) - [core] Improve JobTracker High Availability
- [\[HUE-1493\]](#) - [beeswax] Replace the Beeswax server with HiveServer2
- [\[HUE-1474\]](#) - [core] Upgrade Django backend version from 1.2 to 1.4
- [\[HUE-1506\]](#) - [search] Impersonation support added
- [\[HUE-1475\]](#) - [core] Switch back from the Spawning web server
- [\[HUE-917\]](#) - Support SAML based authentication to enable single sign-on (SSO)

From master:

- [\[HUE-950\]](#) - [core] Improvements to the document model
- [\[HUE-1595\]](#) - Integrate Metastore data into Hive and Impala Query UIs
- [\[HUE-1275\]](#) - [metastore] Show Metastore table details
- [\[HUE-1622\]](#) - [core] Mini tour added to Hue home page

Apache Hive and HCatalog

New Features (Updated to upstream version 0.11.0):

- [\[HIVE-446\]](#) - Implement TRUNCATE for table data
- [\[HIVE-896\]](#) - Add LEAD/LAG/FIRST/LAST analytical windowing functions to Hive
- [\[HIVE-2693\]](#) - Add DECIMAL data type
- [\[HIVE-3834\]](#) - Support ALTER VIEW AS SELECT in Hive

Performance improvements (from 0.12):

- [\[HIVE-3764\]](#) - Support metastore version consistency check
- [\[HIVE-305\]](#) - Port Hadoop streaming process's counters/status reporters to Hive Transforms
- [\[HIVE-1402\]](#) - Add parallel ORDER BY to Hive
- [\[HIVE-2206\]](#) - Add a new optimizer for query correlation discovery and optimization
- [\[HIVE-2517\]](#) - Support GROUP BY on struct type
- [\[HIVE-2655\]](#) - Ability to define functions in HQL
- [\[HIVE-4911\]](#) - Enable QOP configuration for HiveServer2 Thrift transport

Cloudera Impala

Cloudera Impala 1.2.0 is now available as part of CDH 5. For more details on Impala, refer the [Impala Documentation](#).

Llama

Llama is a system that mediates resource management between Cloudera Impala and Hadoop YARN. Llama enables Impala to reserve, use, and release resource allocations in a Hadoop cluster. Llama is only required if resource management is enabled in Impala.

See [Managing the Impala Llama ApplicationMaster](#) for more information.

Apache Mahout

New Features (Updated to Mahout 0.8):

- Numerous performance improvements to Vector and Matrix implementations, APIs and their iterators (see also [MAHOUT-1192](#), [MAHOUT-1202](#))
- Numerous performance improvements to the recommender implementations (see also [MAHOUT-1272](#), [MAHOUT-1035](#), [MAHOUT-1042](#), [MAHOUT-1151](#), [MAHOUT-1166](#), [MAHOUT-1167](#), [MAHOUT-1169](#), [MAHOUT-1205](#), [MAHOUT-1264](#))
- [MAHOUT-1088](#): Support for biased item-based recommender.
- [MAHOUT-1089](#): SGD matrix factorization for rating prediction with user and item biases.
- [MAHOUT-1106](#): Support for SVD++

- [MAHOUT-944](#): Support for converting one or more Lucene storage indexes to SequenceFiles as well as an upgrade of the supported Lucene version to Lucene 4.3.
- [MAHOUT-1154](#) and related: New streaming k-means implementation that offers online (and fast) clustering.
- [MAHOUT-833](#): Make conversion to SequenceFiles Map-Reduce. 'seqdirectory' can now be run as a MapReduce job.
- [MAHOUT-1052](#): Add an option to MinHashDriver that specifies the dimension of vector to hash (indexes or values).
- [MAHOUT-884](#): Matrix concatenate utility; presently only concatenates two matrices.

Apache Oozie

New Features:

- Updated to Oozie 4.0.0.
- **High Availability:** Multiple Oozie servers can now be utilized to provide an HA Oozie service as well as provide horizontal scalability. See upstream [documentation](#) for more details.
- **HCatalog Integration:** HCatalog table partitions can now be used as data dependencies in coordinators. See upstream [documentation](#) for more details. .
- **SLA Monitoring:** Oozie can now actively monitor SLA-sensitive jobs and send out notifications for SLA meets and misses. SLA information is also now available through a new SLA tab in the Oozie Web UI, JMS messages, and a REST API. See upstream [documentation](#).
- **JMS Notifications:** Oozie can now publish notifications to a JMS Provider about job status changes and SLA events. See upstream [documentation](#).
- The FileSystem action can now use glob patterns for file paths when doing move, delete, chmod, and chgrp.

Cloudera Search

Cloudera Search 1.0.0 is now available as part of CDH 5. For more details on Search see the [Search documentation](#).

The Cloudera Development Kit (CDK) is a set of libraries and tools that can be used with Search and other CDH components to build jobs/systems on top of the Hadoop ecosystem. See the [CDK Documentation](#) and [Release Notes](#) for more details.

- **Note:** An existing dependency, Apache Tika, has been upgraded to version 1.4.

Apache Sentry (incubating)

CDH 5 Beta 1 includes the first upstream release of Apache Sentry, `sentry-1.2.0-incubating`.

Apache Sqoop

CDH 5 Sqoop 1 has been rebased on Apache Sqoop 1.4.4.

What's New in Impala

This release of Impala contains the following changes and enhancements from previous releases.

New Features in Impala Version 2.1.4 / CDH 5.3.4

No new features. This point release is exclusively a bug fix release. Because CDH 5.3.5 does not include any code changes for Impala, Impala 2.1.4 is included with both CDH 5.3.4 and 5.3.5.

- **Note:** This Impala maintenance release is only available as part of CDH 5, not under CDH 4.

New Features in Impala Version 2.1.3 / CDH 5.3.3

No new features. This point release is exclusively a bug fix release.

- **Note:** Impala 2.1.3 is available as part of CDH 5.3.3, not under CDH 4.

New Features in Impala Version 2.1.2 / CDH 5.3.2

No new features. This point release is exclusively a bug fix release.

- **Note:** Impala 2.1.2 is available as part of CDH 5.3.2, not under CDH 4.

New Features in Impala Version 2.1.1 / CDH 5.3.1

No new features. This point release is exclusively a bug fix release.

New Features in Impala Version 2.1.0 / CDH 5.3.0

This release contains the following enhancements to query performance and system scalability:

- Impala can now collect statistics for individual partitions in a partitioned table, rather than processing the entire table for each `COMPUTE STATS` statement. This feature is known as incremental statistics, and is controlled by the `COMPUTE INCREMENTAL STATS` syntax. (You can still use the original `COMPUTE STATS` statement for nonpartitioned tables or partitioned tables that are unchanging or whose contents are entirely replaced all at once.) See [COMPUTE STATS Statement](#) and [How Impala Uses Statistics for Query Optimization](#) for details.
- Optimization for small queries lets Impala process queries that process very few rows without the unnecessary overhead of parallelizing and generating native code. Reducing this overhead lets Impala clear small queries quickly, keeping YARN resources and admission control slots available for data-intensive queries. The number of rows considered to be a “small” query is controlled by the `EXEC_SINGLE_NODE_ROWS_THRESHOLD` query option. See [EXEC_SINGLE_NODE_ROWS_THRESHOLD Query Option](#) for details.
- An enhancement to the statestore component lets it transmit heartbeat information independently of broadcasting metadata updates. This optimization improves reliability of health checking on large clusters with many tables and partitions.
- The memory requirement for querying gzip-compressed text is reduced. Now Impala decompresses the data as it is read, rather than reading the entire gzipped file and decompressing it in memory.

New Features in Impala Version 2.0.4 / CDH 5.2.5

No new features. This point release is exclusively a bug fix release.

- **Note:** Impala 2.0.4 is available as part of CDH 5.2.5, not under CDH 4.

New Features in Impala Version 2.0.3 / CDH 5.2.4

No new features. This point release is exclusively a bug fix release.

- **Note:** Impala 2.0.3 is available as part of CDH 5.2.4, not under CDH 4.

New Features in Impala Version 2.0.2 / CDH 5.2.3

No new features. This point release is exclusively a bug fix release.

- **Note:** Impala 2.0.2 is available as part of CDH 5.2.3, not under CDH 4.

New Features in Impala Version 2.0.1 / CDH 5.2.1

No new features. This point release is exclusively a bug fix release.

New Features in Impala Version 2.0.0 / CDH 5.2.0

The following are the major new features in Impala 2.0. This major release, available both with CDH 5.2 and for CDH 4, contains improvements to performance, scalability, security, and SQL syntax.

- Queries with joins or aggregation functions involving high volumes of data can now use temporary work areas on disk, reducing the chance of failure due to out-of-memory errors. When the required memory for

the intermediate result set exceeds the amount available on a particular node, the query automatically uses a temporary work area on disk. This “spill to disk” mechanism is similar to the `ORDER BY` improvement from Impala 1.4. For details, see [SQL Operations that Spill to Disk](#).

- Subquery enhancements:
 - Subqueries are now allowed in the `WHERE` clause, for example with the `IN` operator.
 - The `EXISTS` and `NOT EXISTS` operators are available. They are always used in conjunction with subqueries.
 - The `IN` and `NOT IN` queries can now operate on the result set from a subquery, not just a hardcoded list of values.
 - Uncorrelated subqueries let you compare against one or more values for equality, `IN`, and `EXISTS` comparisons. For example, you might use `WHERE` clauses such as `WHERE column = (SELECT MAX(some_other_column) FROM table)` or `WHERE column IN (SELECT some_other_column FROM table WHERE conditions)`.
 - Correlated subqueries let you cross-reference values from the outer query block and the subquery.
 - Scalar subqueries let you substitute the result of single-value aggregate functions such as `MAX()`, `MIN()`, `COUNT()`, or `AVG()`, where you would normally use a numeric value in a `WHERE` clause.

For details about subqueries, see [Subqueries](#). For information about new and improved operators, see [EXISTS Operator](#) and [IN Operator](#).

- Analytic functions such as `RANK()`, `LAG()`, `LEAD()`, and `FIRST_VALUE()` let you analyze sequences of rows with flexible ordering and grouping. Existing aggregate functions such as `MAX()`, `SUM()`, and `COUNT()` can also be used in an analytic context. See [Impala Analytic Functions](#) for details. See [Impala Aggregate Functions](#) for enhancements to existing aggregate functions.
- New data types provide greater compatibility with source code from traditional database systems:
 - `VARCHAR` is like the `STRING` data type, but with a maximum length. See [VARCHAR Data Type \(CDH 5.2 or higher only\)](#) for details.
 - `CHAR` is like the `STRING` data type, but with a precise length. Short values are padded with spaces on the right. See [CHAR Data Type \(CDH 5.2 or higher only\)](#) for details.
- Security enhancements:
 - Formerly, Impala was restricted to using either Kerberos or LDAP / Active Directory authentication within a cluster. Now, Impala can freely accept either kind of authentication request, allowing you to set up some hosts with Kerberos authentication and others with LDAP or Active Directory. See [Using Multiple Authentication Methods with Impala](#) for details.
 - `GRANT` statement. See [GRANT Statement \(CDH 5.2 or higher only\)](#) for details.
 - `REVOKE` statement. See [REVOKE Statement \(CDH 5.2 or higher only\)](#) for details.
 - `CREATE ROLE` statement. See [CREATE ROLE Statement \(CDH 5.2 or higher only\)](#) for details.
 - `DROP ROLE` statement. See [DROP ROLE Statement \(CDH 5.2 or higher only\)](#) for details.
 - `SHOW ROLES` and `SHOW ROLE GRANT` statements. See [SHOW Statement](#) for details.
 - To complement the HDFS encryption feature, a new Impala configuration option, `--disk_spill_encryption` secures sensitive data from being observed or tampered with when temporarily stored on disk.

The new security-related SQL statements work along with the Sentry authorization framework. See [Enabling Sentry Authorization for Impala](#) for details.

- Impala can now read compressed text files compressed by gzip, bzip, or Snappy. These files do not require any special table settings to work in an Impala text table. Impala recognizes the compression type automatically based on file extensions of `.gz`, `.bz2`, and `.snappy` respectively. These types of compressed text files are intended for convenience with existing ETL pipelines. Their non-splittable nature means they are not optimal for high-performance parallel queries. See [Using gzip, bzip2, or Snappy-Compressed Text Files](#) for details.

- Query hints can now use comment notation, `/* +hint_name */` or `-- +hint_name`, at the same places in the query where the hints enclosed by `[]` are recognized. This enhancement makes it easier to reuse Impala queries on other database systems. See [Hints](#) for details.
- A new query option, `QUERY_TIMEOUT_S`, lets you specify a timeout period in seconds for individual queries.

The working of the `--idle_query_timeout` configuration option is extended. If no `QUERY_OPTION_S` query option is in effect, `--idle_query_timeout` works the same as before, setting the timeout interval. When the `QUERY_OPTION_S` query option is specified, its maximum value is capped by the value of the `--idle_query_timeout` option.

That is, the system administrator sets the default and maximum timeout through the `--idle_query_timeout` startup option, and then individual users or applications can set a lower timeout value if desired through the `QUERY_TIMEOUT_S` query option. See [Setting Timeout Periods for Daemons, Queries, and Sessions](#) and [QUERY_TIMEOUT_S Query Option](#) for details.

- New functions `VAR_SAMP()` and `VAR_POP()` are aliases for the existing `VARIANCE_SAMP()` and `VARIANCE_POP()` functions.
- A new date and time function, `DATE_PART()`, provides similar functionality to `EXTRACT()`. You can also call the `EXTRACT()` function using the SQL-99 syntax, `EXTRACT(unit FROM timestamp)`. These enhancements simplify the porting process for date-related code from other systems. See [Impala Date and Time Functions](#) for details.
- New approximation features provide a fast way to get results when absolute precision is not required:
 - The `APPX_COUNT_DISTINCT` query option lets Impala rewrite `COUNT(DISTINCT)` calls to use `NDV()` instead, which speeds up the operation and allows multiple `COUNT(DISTINCT)` operations in a single query. See [APPX_COUNT_DISTINCT Query Option](#) for details.

The `APPX_MEDIAN()` aggregate function produces an estimate for the median value of a column by using sampling. See [APPX_MEDIAN Function](#) for details.

- Impala now supports a `DECODE()` function. This function works as a shorthand for a `CASE()` expression, and improves compatibility with SQL code containing vendor extensions. See [Impala Conditional Functions](#) for details.
- The `STDDEV()`, `STDDEV_POP()`, `STDDEV_SAMP()`, `VARIANCE()`, `VARIANCE_POP()`, `VARIANCE_SAMP()`, and `NDV()` aggregate functions now all return `DOUBLE` results rather than `STRING`. Formerly, you were required to `CAST()` the result to a numeric type before using it in arithmetic operations.
- The default settings for Parquet block size, and the associated `PARQUET_FILE_SIZE` query option, are changed. Now, Impala writes Parquet files with a size of 256 MB and an HDFS block size of 256 MB. Previously, Impala attempted to write Parquet files with a size of 1 GB and an HDFS block size of 1 GB. In practice, Impala used a conservative estimate of the disk space needed for each Parquet block, leading to files that were typically 512 MB anyway. Thus, this change will make the file size more accurate if you specify a value for the `PARQUET_FILE_SIZE` query option. It also reduces the amount of memory reserved during `INSERT` into Parquet tables, potentially avoiding out-of-memory errors and improving scalability when inserting data into Parquet tables.
- Anti-joins are now supported, expressed using the `LEFT ANTI JOIN` and `RIGHT ANTI JOIN` clauses. These clauses returns results from one table that have no match in the other table. You might use this type of join in the same sorts of use cases as the `NOT EXISTS` and `NOT IN` operators. See [Joins](#) for details.
- The `SET` command in `impala-shell` has been promoted to a real SQL statement. You can now set query options such as `PARQUET_FILE_SIZE`, `MEM_LIMIT`, and `SYNC_DDL` within JDBC, ODBC, or any other kind of application that submits SQL without going through the `impala-shell` interpreter. See [SET Statement](#) for details.
- The `impala-shell` interpreter now reads settings from an optional configuration file, named `$HOME/.impalarc` by default. See [impala-shell Configuration Options](#) for details.

- The library used for regular expression parsing has changed from Boost to Google RE2. This implementation change adds support for non-greedy matches using the `. * ?` notation. This and other changes in the way regular expressions are interpreted means you might need to re-test queries that use functions such as `regexp_extract()` or `regexp_replace()`, or operators such as `REGEXP` or `RLIKE`. See [Cloudera Impala Incompatible Changes](#) on page 57 for those details.

New Features in Impala Version 1.4.4 / CDH 5.1.5

No new features. This point release is exclusively a bug fix release.

- **Note:** Impala 1.4.4 is available as part of CDH 5.1.5, not under CDH 4.

New Features in Impala Version 1.4.3 / CDH 5.1.4

No new features. This point release is exclusively a bug fix release for an SSL security issue.

- **Note:** Impala 1.4.3 is available as part of CDH 5.1.4, and under CDH 4.

New Features in Impala Version 1.4.2 / CDH 5.1.3

Impala 1.4.2 is purely a bug-fix release. It does not include any new features.

- **Note:** Impala 1.4.2 is only available as part of CDH 5.1.3, not under CDH 4.

New Features in Impala Version 1.4.1 / CDH 5.1.2

Impala 1.4.1 is purely a bug-fix release. It does not include any new features.

New Features in Impala Version 1.4.0 / CDH 5.1.0

- The `DECIMAL` data type lets you store fixed-precision values, for working with currency or other fractional values where it is important to represent values exactly and avoid rounding errors. This feature includes enhancements to built-in functions, numeric literals, and arithmetic expressions.
- On CDH 5, Impala can take advantage of the HDFS caching feature to “pin” entire tables or individual partitions in memory, to speed up queries on frequently accessed data and reduce the CPU overhead of memory-to-memory copying. When HDFS files are cached in memory, Impala can read the cached data without any disk reads, and without making an additional copy of the data in memory. Other Hadoop components that read the same data files also experience a performance benefit.
- Impala can now use Sentry-based authorization based either on the original policy file, or on rules defined by `GRANT` and `REVOKE` statements issued through Hive.
- For interoperability with Parquet files created through other Hadoop components, such as Pig or MapReduce jobs, you can create an Impala table that automatically sets up the column definitions based on the layout of an existing Parquet data file.
- `ORDER BY` queries no longer require a `LIMIT` clause. If the size of the result set to be sorted exceeds the memory available to Impala, Impala uses a temporary work space on disk to perform the sort operation.
- LDAP connections can be secured through either SSL or TLS.
- The following new built-in scalar and aggregate functions are available:
 - A new built-in function, `EXTRACT()`, returns one date or time field from a `TIMESTAMP` value.
 - A new built-in function, `TRUNC()`, truncates date/time values to a particular granularity, such as year, month, day, hour, and so on.
 - `ADD_MONTHS()` built-in function, an alias for the existing `MONTHS_ADD()` function.
 - A new built-in function, `ROUND()`, rounds `DECIMAL` values to a specified number of fractional digits.

- Several built-in aggregate functions for computing properties for statistical distributions: `STDDEV()`, `STDDEV_SAMP()`, `STDDEV_POP()`, `VARIANCE()`, `VARIANCE_SAMP()`, and `VARIANCE_POP()`.
 - Several new built-in functions, such as `MAX_INT()`, `MIN_SMALLINT()`, and so on, let you conveniently check whether data values are in an expected range. You might be able to switch a column to a smaller type, saving memory during processing.
 - New built-in functions, `IS_INF()` and `IS_NAN()`, check for the special values infinity and “not a number”. These values could be specified as `inf` or `nan` in text data files, or be produced by certain arithmetic expressions.
- The `SHOW PARTITIONS` statement displays information about the structure of a partitioned table.
 - New configuration options for the `impalad` daemon let you specify initial memory usage for all queries. The initial resource requests handled by Llama and YARN can be expanded later if needed, avoiding unnecessary over-allocation and reducing the chance of out-of-memory conditions.
 - Impala can take advantage of the Llama high availability feature in CDH 5.1, for improved reliability of resource management through YARN.
 - The Impala `CREATE TABLE` statement now has a `STORED AS AVRO` clause, allowing you to create Avro tables through Impala.
 - New `impalad` configuration options let you fine-tune the calculations Impala makes to estimate resource requirements for each query. These options can help avoid problems due to overconsumption due to too-low estimates, or underutilization due to too-high estimates.
 - A new `SUMMARY` command in the `impala-shell` interpreter provides a high-level summary of the work performed at each stage of the explain plan. The summary is also included in output from the `PROFILE` command.
 - Performance improvements for the `COMPUTE STATS` statement:
 - The `NDV` function is speeded up through native code generation.
 - Because the `NULL` count is not currently used by the Impala query planner, in Impala 1.4.0 and higher, `COMPUTE STATS` does not count the `NULL` values for each column. (The `#Nulls` field of the stats table is left as `-1`, signifying that the value is unknown.)
 - Performance improvements for partition pruning. This feature reduces the time spent in query planning, for partitioned tables with thousands of partitions. Previously, Impala typically queried tables with up to approximately 3000 partitions. With the performance improvement in partition pruning, now Impala can comfortably handle tables with tens of thousands of partitions.
 - The documentation provides additional guidance for planning tasks.
 - The `impala-shell` interpreter now supports UTF-8 characters for input and output. You can control whether `impala-shell` ignores invalid Unicode code points through the `--strict_unicode` option. (Although this option is removed in Impala 2.0.)

New Features in Impala Version 1.3.3 / CDH 5.0.5

No new features. This point release is exclusively a bug fix release for an SSL security issue.

- **Note:** Impala 1.3.3 is only available as part of CDH 5.0.5, not under CDH 4.

New Features in Impala Version 1.3.2 / CDH 5.0.4

No new features. This point release is exclusively a bug fix release for the IMPALA-1019 issue related to HDFS caching.

- **Note:** Impala 1.3.2 is only available as part of CDH 5.0.4, not under CDH 4.

New Features in Impala Version 1.3.1 / CDH 5.0.3

This point release is primarily a vehicle to deliver bug fixes. Any new features are minor changes resulting from fixes for performance, reliability, or usability issues.

Because 1.3.1 is the first 1.3.x release for CDH 4, if you are on CDH 4, also consult [New Features in Impala Version 1.3.0 / CDH 5.0.0](#) on page 41 for more features that are new to you.

■ **Note:**

- The Impala 1.3.1 release is available for both CDH 4 and CDH 5. This is the first release in the 1.3.x series for CDH 4.

- A new `impalad` startup option, `--insert_inherit_permissions`, causes Impala `INSERT` statements to create each new partition with the same HDFS permissions as its parent directory. By default, `INSERT` statements create directories for new partitions using default HDFS permissions. See [INSERT Statement](#) for examples of `INSERT` statements for partitioned tables.
- The `SHOW FUNCTIONS` statement now displays the return type of each function, in addition to the types of its arguments. See [SHOW Statement](#) for examples.
- You can now specify the clause `FIELDS TERMINATED BY '\0'` with a `CREATE TABLE` statement to use text data files that use ASCII 0 (nul) characters as a delimiter. See [Using Text Data Files with Impala Tables](#) for details.
- In Impala 1.3.1 and higher, the `REGEXP` and `RLIKE` operators now match a regular expression string that occurs anywhere inside the target string, the same as if the regular expression was enclosed on each side by `.*`. See [REGEXP Operator](#) for examples. Previously, these operators only succeeded when the regular expression matched the entire target string. This change improves compatibility with the regular expression support for popular database systems. There is no change to the behavior of the `regexp_extract()` and `regexp_replace()` built-in functions.

New Features in Impala Version 1.3.0 / CDH 5.0.0

■ **Note:**

- The Impala 1.3.1 release is available for both CDH 4 and CDH 5. This is the first release in the 1.3.x series for CDH 4.

- The admission control feature lets you control and prioritize the volume and resource consumption of concurrent queries. This mechanism reduces spikes in resource usage, helping Impala to run alongside other kinds of workloads on a busy cluster. It also provides more user-friendly conflict resolution when multiple memory-intensive queries are submitted concurrently, avoiding resource contention that formerly resulted in out-of-memory errors. See [Admission Control and Query Queuing](#) for details.
- Enhanced `EXPLAIN` plans provide more detail in an easier-to-read format. Now there are four levels of verbosity: the `EXPLAIN_LEVEL` option can be set from 0 (most concise) to 3 (most verbose). See [EXPLAIN Statement](#) for syntax and [Understanding Impala Query Performance - EXPLAIN Plans and Query Profiles](#) for usage information.
- The `TIMESTAMP` data type accepts more kinds of input string formats through the `UNIX_TIMESTAMP` function, and produces more varieties of string formats through the `FROM_UNIXTIME` function. The documentation now also lists more functions for date arithmetic, used for adding and subtracting `INTERVAL` expressions from `TIMESTAMP` values. See [Impala Date and Time Functions](#) for details.
- New conditional functions, `NULLIF()`, `NULLIFZERO()`, and `ZEROIFNULL()`, simplify porting SQL containing vendor extensions to Impala. See [Impala Conditional Functions](#) for details.
- New utility function, `CURRENT_DATABASE()`. See [Impala Miscellaneous Functions](#) for details.

- Integration with the YARN resource management framework. Only available in combination with CDH 5. This feature makes use of the underlying YARN service, plus an additional service (Llama) that coordinates requests to YARN for Impala resources, so that the Impala query only proceeds when all requested resources are available. See [Integrated Resource Management with YARN](#) for full details.

On the Impala side, this feature involves some new startup options for the `impalad` daemon:

- `-enable_rm`
- `-llama_host`
- `-llama_port`
- `-llama_callback_port`
- `-cgroup_hierarchy_path`

For details of these startup options, see [Modifying Impala Startup Options](#).

This feature also involves several new or changed query options that you can set through the `impala-shell` interpreter and apply within a specific session:

- `MEM_LIMIT`: the function of this existing option changes when Impala resource management is enabled.
- `REQUEST_POOL`: a new option. (Renamed to `RESOURCE_POOL` in Impala 1.3.0.)
- `V_CPU_CORES`: a new option.
- `RESERVATION_REQUEST_TIMEOUT`: a new option.

For details of these query options, see [impala-shell Query Options for Resource Management](#).

New Features in Impala Version 1.2.4

- **Note:** Impala 1.2.4 works with CDH 4. It is primarily a bug fix release for Impala 1.2.3, plus some performance enhancements for the catalog server to minimize startup and DDL wait times for Impala deployments with large numbers of databases, tables, and partitions.

- On Impala startup, the metadata loading and synchronization mechanism has been improved and optimized, to give more responsiveness when starting Impala on a system with a large number of databases, tables, or partitions. The initial metadata loading happens in the background, allowing queries to be run before the entire process is finished. When a query refers to a table whose metadata is not yet loaded, the query waits until the metadata for that table is loaded, and the load operation for that table is prioritized to happen first.
- Formerly, if you created a new table in Hive, you had to issue the `INVALIDATE METADATA` statement (with no table name) which was an expensive operation that reloaded metadata for all tables. Impala did not recognize the name of the Hive-created table, so you could not do `INVALIDATE METADATA new_table` to get the metadata for just that one table. Now, when you issue `INVALIDATE METADATA table_name`, Impala checks to see if that name represents a table created in Hive, and if so recognizes the new table and loads the metadata for it. Additionally, if the new table is in a database that was newly created in Hive, Impala also recognizes the new database.
- If you issue `INVALIDATE METADATA table_name` and the table has been dropped through Hive, Impala will recognize that the table no longer exists.
- New startup options let you control the parallelism of the metadata loading during startup for the `catalogd` daemon:
 - `--load_catalog_in_background` makes Impala load and cache metadata using background threads after startup. It is `true` by default. Previously, a system with a large number of databases, tables, or partitions could be unresponsive or even time out during startup.
 - `--num_metadata_loading_threads` determines how much parallelism Impala devotes to loading metadata in the background. The default is 16. You might increase this value for systems with huge numbers of databases, tables, or partitions. You might lower this value for busy systems that are CPU-constrained due to jobs from components other than Impala.

New Features in Impala Version 1.2.3

- **Note:** Impala 1.2.3 works with CDH 4 and with CDH 5 beta 2. The resource management feature requires CDH 5 beta.

Impala 1.2.3 contains exactly the same feature set as Impala 1.2.2. Its only difference is one additional fix for compatibility with Parquet files generated outside of Impala by components such as Hive, Pig, or MapReduce. If you are upgrading from Impala 1.2.1 or earlier, see [New Features in Impala Version 1.2.2](#) on page 43 for the latest added features.

New Features in Impala Version 1.2.2

- **Note:** Impala 1.2.2 works with CDH 4. Its feature set is a superset of features in the Impala 1.2.0 beta, with the exception of resource management, which relies on CDH 5.

Impala 1.2.2 includes new features for performance, security, and flexibility. The major enhancements over 1.2.1 are performance related, primarily for join queries.

New user-visible features include:

- Join order optimizations. This highly valuable feature automatically distributes and parallelizes the work for a join query to minimize disk I/O and network traffic. The automatic optimization reduces the need to use query hints or to rewrite join queries with the tables in a specific order based on size or cardinality. The new `COMPUTE STATS` statement gathers statistical information about each table that is crucial for enabling the join optimizations. See [Performance Considerations for Join Queries](#) for details.
- `COMPUTE STATS` statement to collect both table statistics and column statistics with a single statement. Intended to be more comprehensive, efficient, and reliable than the corresponding Hive `ANALYZE TABLE` statement, which collects statistics in multiple phases through MapReduce jobs. These statistics are important for query planning for join queries, queries on partitioned tables, and other types of data-intensive operations. For optimal planning of join queries, you need to collect statistics for each table involved in the join. See [COMPUTE STATS Statement](#) for details.
- Reordering of tables in a join query can be overridden by the `STRAIGHT_JOIN` operator, allowing you to fine-tune the planning of the join query if necessary, by using the original technique of ordering the joined tables in descending order of size. See [Overriding Join Reordering with STRAIGHT_JOIN](#) for details.
- The `CROSS JOIN` clause in the `SELECT` statement to allow Cartesian products in queries, that is, joins without an equality comparison between columns in both tables. Because such queries must be carefully checked to avoid accidental overconsumption of memory, you must use the `CROSS JOIN` operator to explicitly select this kind of join. See [Cross Joins and Cartesian Products with the CROSS JOIN Operator](#) for examples.
- The `ALTER TABLE` statement has new clauses that let you fine-tune table statistics. You can use this technique as a less-expensive way to update specific statistics, in case the statistics become stale, or to experiment with the effects of different data distributions on query planning.
- LDAP username/password authentication in JDBC/ODBC. See [Enabling LDAP Authentication for Impala](#) for details.
- [GROUP_CONCAT\(\)](#) aggregate function to concatenate column values across all rows of a result set.
- The `INSERT` statement now accepts hints, `[SHUFFLE]` and `[NOSHUFFLE]`, to influence the way work is redistributed during `INSERT...SELECT` operations. The hints are primarily useful for inserting into partitioned Parquet tables, where using the `[SHUFFLE]` hint can avoid problems due to memory consumption and simultaneous open files in HDFS, by collecting all the new data for each partition on a specific node.
- Several built-in functions and operators are now overloaded for more numeric data types, to reduce the requirement to use `CAST()` for type coercion in `INSERT` statements. For example, the expression `2+2` in an `INSERT` statement formerly produced a `BIGINT` result, requiring a `CAST()` to be stored in an `INT` variable. Now, addition, subtraction, and multiplication only produce a result that is one step “bigger” than their

arguments, and numeric and conditional functions can return `SMALLINT`, `FLOAT`, and other smaller types rather than always `BIGINT` or `DOUBLE`.

- New `fnv_hash()` built-in function for constructing hashed values. See [Impala Mathematical Functions](#) for details.
- The clause `STORED AS PARQUET` is accepted as an equivalent for `STORED AS PARQUETFILE`. This more concise form is recommended for new code.

Because Impala 1.2.2 builds on a number of features introduced in 1.2.1, if you are upgrading from an older 1.1.x release straight to 1.2.2, also review [New Features in Impala Version 1.2.1](#) on page 44 to see features such as the `SHOW TABLE STATS` and `SHOW COLUMN STATS` statements, and user-defined functions (UDFs).

New Features in Impala Version 1.2.1

- **Note:** Impala 1.2.1 works with CDH 4. Its feature set is a superset of features in the Impala 1.2.0 beta, with the exception of resource management, which relies on CDH 5.

Impala 1.2.1 includes new features for security, performance, and flexibility.

New user-visible features include:

- `SHOW TABLE STATS table_name` and `SHOW COLUMN STATS table_name` statements, to verify that statistics are available and to see the values used during query planning.
- `CREATE TABLE AS SELECT` syntax, to create a new table and transfer data into it in a single operation.
- `OFFSET` clause, for use with the `ORDER BY` and `LIMIT` clauses to produce “paged” result sets such as items 1-10, then 11-20, and so on.
- `NULLS FIRST` and `NULLS LAST` clauses to ensure consistent placement of `NULL` values in `ORDER BY` queries.
- New [built-in functions](#): `least()`, `greatest()`, `initcap()`.
- New aggregate function: `ndv()`, a fast alternative to `COUNT(DISTINCT col)` returning an approximate result.
- The `LIMIT` clause can now accept a numeric expression as an argument, rather than only a literal constant.
- The `SHOW CREATE TABLE` statement displays the end result of all the `CREATE TABLE` and `ALTER TABLE` statements for a particular table. You can use the output to produce a simplified setup script for a schema.
- The `--idle_query_timeout` and `--idle_session_timeout` options for `impalad` control the time intervals after which idle queries are cancelled, and idle sessions expire. See [Setting Timeout Periods for Daemons, Queries, and Sessions](#) for details.
- User-defined functions (UDFs). This feature lets you transform data in very flexible ways, which is important when using Impala as part of an ETL or ELT pipeline. Prior to Impala 1.2, using UDFs required switching into Hive. Impala 1.2 can run scalar UDFs and user-defined aggregate functions (UDAs). Impala can run high-performance functions written in C++, or you can reuse existing Hive functions written in Java.

You create UDFs through the `CREATE FUNCTION` statement and drop them through the `DROP FUNCTION` statement. See [Impala User-Defined Functions \(UDFs\)](#) for instructions about coding, building, and deploying UDFs, and [CREATE FUNCTION Statement](#) and [DROP FUNCTION Statement](#) for related SQL syntax.

- A new service automatically propagates changes to table data and metadata made by one Impala node, sending the new or updated metadata to all the other Impala nodes. The automatic synchronization mechanism eliminates the need to use the `INVALIDATE METADATA` and `REFRESH` statements after issuing Impala statements such as `CREATE TABLE`, `ALTER TABLE`, `DROP TABLE`, `INSERT`, and `LOAD DATA`.

For even more precise synchronization, you can enable the [SYNC DDL](#) query option before issuing a DDL, `INSERT`, or `LOAD DATA` statement. This option causes the statement to wait, returning only after the catalog service has broadcast the applicable changes to all Impala nodes in the cluster.

■ **Note:**

Because the catalog service only monitors operations performed through Impala, `INVALIDATE METADATA` and `REFRESH` are still needed on the Impala side after creating new tables or loading data through the Hive shell or by manipulating data files directly in HDFS. Because the catalog service broadcasts the result of the `REFRESH` and `INVALIDATE METADATA` statements to all Impala nodes, when you do need to use those statements, you can do so a single time rather than on every Impala node.

This service is implemented by the `catalogd` daemon. See [The Impala Catalog Service](#) for details.

- `CREATE TABLE ... AS SELECT` syntax, to create a table and copy data into it in a single operation. See [CREATE TABLE Statement](#) for details.
- The `CREATE TABLE` and `ALTER TABLE` statements have new clauses `TBLPROPERTIES` and `WITH SERDEPROPERTIES`. The `TBLPROPERTIES` clause lets you associate arbitrary items of metadata with a particular table as key-value pairs. The `WITH SERDEPROPERTIES` clause lets you specify the serializer/deserializer (SerDes) classes that read and write data for a table; although Impala does not make use of these properties, sometimes particular values are needed for Hive compatibility. See [CREATE TABLE Statement](#) and [ALTER TABLE Statement](#) for details.
- Impersonation support lets you authorize certain OS users associated with applications (for example, `hue`), to submit requests using the credentials of other users. Only available in combination with CDH 5. See [Configuring Per-User Access for Hue](#) for details.
- Enhancements to `EXPLAIN` output. In particular, when you enable the new `EXPLAIN_LEVEL` query option, the `EXPLAIN` and `PROFILE` statements produce more verbose output showing estimated resource requirements and whether table and column statistics are available for the applicable tables and columns. See [EXPLAIN Statement](#) for details.
- `SHOW CREATE TABLE` summarizes the effects of the original `CREATE TABLE` statement and any subsequent `ALTER TABLE` statements, giving you a `CREATE TABLE` statement that will re-create the current structure and layout for a table.
- The `LIMIT` clause for queries now accepts an arithmetic expression, in addition to numeric literals.

New Features in Impala Version 1.2.0 (Beta)

- **Note:** The Impala 1.2.0 beta release only works in combination with the beta version of CDH 5. The Impala 1.2.0 software is bundled together with the CDH 5 beta 1 download.

The Impala 1.2.0 beta includes new features for security, performance, and flexibility.

New user-visible features include:

- User-defined functions (UDFs). This feature lets you transform data in very flexible ways, which is important when using Impala as part of an ETL or ELT pipeline. Prior to Impala 1.2, using UDFs required switching into Hive. Impala 1.2 can run scalar UDFs and user-defined aggregate functions (UDAs). Impala can run high-performance functions written in C++, or you can reuse existing Hive functions written in Java. You create UDFs through the `CREATE FUNCTION` statement and drop them through the `DROP FUNCTION` statement. See [Impala User-Defined Functions \(UDFs\)](#) for instructions about coding, building, and deploying UDFs, and [CREATE FUNCTION Statement](#) and [DROP FUNCTION Statement](#) for related SQL syntax.
- A new service automatically propagates changes to table data and metadata made by one Impala node, sending the new or updated metadata to all the other Impala nodes. The automatic synchronization mechanism eliminates the need to use the `INVALIDATE METADATA` and `REFRESH` statements after issuing Impala statements such as `CREATE TABLE`, `ALTER TABLE`, `DROP TABLE`, `INSERT`, and `LOAD DATA`.

■ **Note:**

Because this service only monitors operations performed through Impala, `INVALIDATE METADATA` and `REFRESH` are still needed on the Impala side after creating new tables or loading data through the Hive shell or by manipulating data files directly in HDFS. Because the catalog service broadcasts the result of the `REFRESH` and `INVALIDATE METADATA` statements to all Impala nodes, when you do need to use those statements, you can do so a single time rather than on every Impala node.

This service is implemented by the `catalogd` daemon. See [The Impala Catalog Service](#) for details.

- Integration with the YARN resource management framework. Only available in combination with CDH 5. This feature makes use of the underlying YARN service, plus an additional service (Llama) that coordinates requests to YARN for Impala resources, so that the Impala query only proceeds when all requested resources are available. See [Integrated Resource Management with YARN](#) for full details.

On the Impala side, this feature involves some new startup options for the `impalad` daemon:

- `-enable_rm`
- `-llama_host`
- `-llama_port`
- `-llama_callback_port`
- `-cgroup_hierarchy_path`

For details of these startup options, see [Modifying Impala Startup Options](#).

This feature also involves several new or changed query options that you can set through the `impala-shell` interpreter and apply within a specific session:

- `MEM_LIMIT`: the function of this existing option changes when Impala resource management is enabled.
- `YARN_POOL`: a new option. (Renamed to `RESOURCE_POOL` in Impala 1.3.0.)
- `V_CPU_CORES`: a new option.
- `RESERVATION_REQUEST_TIMEOUT`: a new option.

For details of these query options, see [impala-shell Query Options for Resource Management](#).

- `CREATE TABLE ... AS SELECT` syntax, to create a table and copy data into it in a single operation. See [CREATE TABLE Statement](#) for details.
- The `CREATE TABLE` and `ALTER TABLE` statements have a new `TBLPROPERTIES` clause that lets you associate arbitrary items of metadata with a particular table as key-value pairs. See [CREATE TABLE Statement](#) and [ALTER TABLE Statement](#) for details.
- Impersonation support lets you authorize certain OS users associated with applications (for example, `hue`), to submit requests using the credentials of other users. Only available in combination with CDH 5. See [Configuring Per-User Access for Hue](#) for details.
- Enhancements to `EXPLAIN` output. In particular, when you enable the new `EXPLAIN_LEVEL` query option, the `EXPLAIN` and `PROFILE` statements produce more verbose output showing estimated resource requirements and whether table and column statistics are available for the applicable tables and columns. See [EXPLAIN Statement](#) for details.

New Features in Impala Version 1.1.1

Impala 1.1.1 includes new features for security and stability.

New user-visible features include:

- Additional security feature: auditing. New startup options for `impalad` let you capture information about Impala queries that succeed or are blocked due to insufficient privileges. To take full advantage of this feature with Cloudera Manager, upgrade to Cloudera Manager 4.7 or higher. For details, see [Overview of Impala Security](#).

- Parquet data files generated by Impala 1.1.1 are now compatible with the Parquet support in Hive. See [Cloudera Impala Incompatible Changes](#) on page 57 for the procedure to update older Impala-created Parquet files to be compatible with the Hive Parquet support.
- Additional improvements to stability and resource utilization for Impala queries.
- Additional enhancements for compatibility with existing file formats.

New Features in Impala Version 1.1

Impala 1.1 includes new features for security, performance, and usability.

New user-visible features include:

- Extensive new security features, built on top of the Sentry open source project. Impala now supports fine-grained authorization based on roles. A policy file determines which privileges on which schema objects (servers, databases, tables, and HDFS paths) are available to users based on their membership in groups. By assigning privileges for views, you can control access to table data at the column level. For details, see [Overview of Impala Security](#).
- Impala 1.1 works with Cloudera Manager 4.6 or higher. To use Cloudera Manager to manage authorization for the Impala web UI (the web pages served from port 25000 by default), use Cloudera Manager 4.6.2 or higher.
- Impala can now create, alter, drop, and query views. Views provide a flexible way to set up simple aliases for complex queries; hide query details from applications and users; and simplify maintenance as you rename or reorganize databases, tables, and columns. See the overview section [Views](#) and the statements [CREATE VIEW Statement](#), [ALTER VIEW Statement](#), and [DROP VIEW Statement](#).
- Performance is improved through a number of automatic optimizations. Resource consumption is also reduced for Impala queries. These improvements apply broadly across all kinds of workloads and file formats. The major areas of performance enhancement include:
 - Improved disk and thread scheduling, which applies to all queries.
 - Improved hash join and aggregation performance, which applies to queries with large build tables or a large number of groups.
 - Dictionary encoding with Parquet, which applies to Parquet tables with short string columns.
 - Improved performance on systems with SSDs, which applies to all queries and file formats.
- Some new built-in functions are implemented: [translate\(\)](#) to substitute characters within strings, [user\(\)](#) to check the login ID of the connected user.
- The new `WITH` clause for `SELECT` statements lets you simplify complicated queries in a way similar to creating a view. The effects of the `WITH` clause only last for the duration of one query, unlike views, which are persistent schema objects that can be used by multiple sessions or applications. See [WITH Clause](#).
- An enhancement to `DESCRIBE` statement, `DESCRIBE FORMATTED table_name`, displays more detailed information about the table. This information includes the file format, location, delimiter, ownership, external or internal, creation and access times, and partitions. The information is returned as a result set that can be interpreted and used by a management or monitoring application. See [DESCRIBE Statement](#).
- You can now insert a subset of columns for a table, with other columns being left as all `NULL` values. Or you can specify the columns in any order in the destination table, rather than having to match the order of the corresponding columns in the source. `VALUES` clause. This feature is known as “column permutation”. See [INSERT Statement](#).
- The new `LOAD DATA` statement lets you load data into a table directly from an HDFS data file. This technique lets you minimize the number of steps in your ETL process, and provides more flexibility. For example, you can bring data into an Impala table in one step. Formerly, you might have created an external table where the data files are not entirely under your control, or copied the data files to Impala data directories manually, or loaded the original data into one table and then used the `INSERT` statement to copy it to a new table with a different file format, partitioning scheme, and so on. See [LOAD DATA Statement](#).
- Improvements to Impala-HBase integration:
 - New query options for HBase performance: [HBASE_CACHE_BLOCKS](#) and [HBASE_CACHING](#).
 - Support for binary data types in HBase tables. See [Supported Data Types for HBase Columns](#) for details.

- You can issue `REFRESH` as a SQL statement through any of the programming interfaces that Impala supports. `REFRESH` formerly had to be issued as a command through the `impala-shell` interpreter, and was not available through a JDBC or ODBC API call. As part of this change, the functionality of the `REFRESH` statement is divided between two statements. In Impala 1.1, `REFRESH` requires a table name argument and immediately reloads the metadata; the new `INVALIDATE METADATA` statement works the same as the Impala 1.0 `REFRESH` did: the table name argument is optional, and the metadata for one or all tables is marked as stale, but not actually reloaded until the table is queried. When you create a new table in the Hive shell or through a different Impala node, you must enter `INVALIDATE METADATA` with no table parameter before you can see the new table in `impala-shell`. See [REFRESH Statement](#) and [INVALIDATE METADATA Statement](#).

New Features in Impala Version 1.0.1

The primary enhancements in Impala 1.0.1 are internal, for compatibility with the new Cloudera Manager 4.6 release. Try out the new **Impala Query Monitoring** feature in Cloudera Manager 4.6, which requires Impala 1.0.1.

New user-visible features include:

- The `VALUES` clause lets you `INSERT` one or more rows using literals, function return values, or other expressions. For performance and scalability, you should still use `INSERT ... SELECT` for bringing large quantities of data into an Impala table. The `VALUES` clause is a convenient way to set up small tables, particularly for initial testing of SQL features that do not require large amounts of data. See [VALUES Clause](#) for details.
- The `-B` and `-o` options of the `impala-shell` command can turn query results into delimited text files and store them in an output file. The plain text results are useful for using with other Hadoop components or Unix tools. In benchmark tests, it is also faster to produce plain rather than pretty-printed results, and write to a file rather than to the screen, giving a more accurate picture of the actual query time.
- Several bug fixes. See [Issues Fixed in the 1.0.1 Release](#) on page 144 for details.

New Features in Impala Version 1.0

This version has multiple performance improvements and adds the following functionality:

- Several bug fixes. See [Issues Fixed in the 1.0 GA Release](#) on page 145.
- [ALTER TABLE](#) statement.
- [Hints](#) to allow specifying a particular join strategy.
- [REFRESH](#) for a single table.
- Dynamic resource management, allowing high concurrency for Impala queries.

New Features in Version 0.7 of the Cloudera Impala Beta Release

This version has multiple performance improvements and adds the following functionality:

- Several bug fixes. See [Issues Fixed in Version 0.7 of the Beta Release](#) on page 148.
- Support for the Parquet file format. For more information on file formats, see [How Impala Works with Hadoop File Formats](#).
- Added support for Avro.
- Support for the memory limits. For more information, see the example on modifying memory limits in [Modifying Impala Startup Options](#).
- Bigger and faster joins through the addition of partitioned joins to the already supported broadcast joins.
- Fully distributed aggregations.
- Fully distributed top-n computation.
- Support for creating and altering tables.
- Support for `GROUP BY` with floats and doubles.

In this version, both CDH 4.1 and 4.2 are supported, but due to performance improvements added, we highly recommend you use CDH 4.2 or higher to see the full benefit. If you are using Cloudera Manager, version 4.5 is required.

New Features in Version 0.6 of the Cloudera Impala Beta Release

- Several bug fixes. See [Issues Fixed in Version 0.6 of the Beta Release](#) on page 149.
- Added support for Impala on SUSE and Debian/Ubuntu. Impala is now supported on:

- RHEL5.7/6.2 and Centos5.7/6.2
- SUSE 11 with Service Pack 1 or higher
- Ubuntu 10.04/12.04 and Debian 6.03
- Cloudera Manager 4.5 and CDH 4.2 support Impala 0.6.
- Support for the RCFile file format. For more information on file formats, see [Understanding File Formats](#).

New Features in Version 0.5 of the Cloudera Impala Beta Release

- Several bug fixes. See [Issues Fixed in Version 0.5 of the Beta Release](#) on page 150.
- Added support for a JDBC driver that allows you to access Impala from a Java client. To use this feature, follow the instructions in [Configuring Impala to Work with JDBC](#) to install the JDBC driver JARs on the client machine and modify the CLASSPATH on the client to include the JARs.

New Features in Version 0.4 of the Cloudera Impala Beta Release

- Several bug fixes. See [Issues Fixed in Version 0.4 of the Beta Release](#) on page 151.
- Added support for Impala on RHEL5.7/Centos5.7. Impala is now supported on RHEL5.7/6.2 and Centos5.7/6.2.
- Cloudera Manager 4.1.3 supports Impala 0.4.
- The Impala debug webserver now has the ability to serve static files from `${IMPALA_HOME}/www`. This can be disabled by setting `--enable_webserver_doc_root=false` on the command line. As a result, Impala now uses the Twitter Bootstrap library to style its debug webpages, and the `/queries` page now tracks the last 25 queries run by each Impala daemon.
- Additional metrics available on the Impala Debug Webpage.

New Features in Version 0.3 of the Cloudera Impala Beta Release

- Several bug fixes. See [Issues Fixed in Version 0.3 of the Beta Release](#) on page 151.
- The `state-store-service` binary has been renamed `statestored`.
- The location of the Impala configuration files has changed from the `/usr/lib/impala/conf` directory to the `/etc/impala/conf` directory.

New Features in Version 0.2 of the Cloudera Impala Beta Release

- Several bug fixes. See [Issues Fixed in Version 0.2 of the Beta Release](#) on page 152.
- **Added Default Query Options** Default query options override all default QueryOption values when starting `impalad`. The format is:

```
-default_query_options='key=value;key=value'
```

Incompatible Changes

Important:

For changes in operating-system support, and other major requirements, see [CDH 5 Requirements and Supported Versions](#).

Apache Avro Incompatible Changes

See [Apache Avro](#) on page 11 section of [What's New in CDH 5.2.0](#) on page 11 for two changes that could possibly affect you when you upgrade to CDH 5.2.0.

Apache Crunch Incompatible Changes

The following changes introduced in CDH 5.2 are not backward compatible:

- The `MemPipeline` now checks to ensure that any `DoFns` that are passed to it are serializable. This is designed to catch non-serializable `DoFns` during testing.

- Scala's `Iterable` has been replaced by `TraversableOnce` inside Scrunch `flatMap` functions in order to support functions that return iterators.

Apache DataFu Incompatible Changes

- Upgraded from version 0.4 to 1.1.0 (this upgrade is not backwards compatible).
- Removed `ApplyQuantiles`, `AliasBagFields`.
- Renamed package `datafu.pig.numbers` to `datafu.pig.random`.
- Renamed package `datafu.pig.bag.sets` to `datafu.pig.sets`.
- Renamed `TimeCount` to `SessionCount`, moved to `datafu.pig.sessions`.

Apache Flume Incompatible Changes

There are no incompatible changes at this point.

Apache Hadoop Incompatible Changes HDFS

The following incompatible changes have been introduced in CDH 5:

- The `getSnapshottableDirListing()` method returns `null` when there are no snapshottable directories. This is a change from CDH 5 Beta 2 where the method returns an empty array instead.
- [HDFS-5138](#) - The `-finalize` NameNode startup option has been removed. To finalize an in-progress upgrade, you should instead use the `hdfs dfsadmin -finalizeUpgrade` command while your NameNode is running, or while both NameNodes are running in a High Availability setup.
- [HDFS-2832](#) - The HDFS internal layout version has changed between CDH 5 Beta 1 and CDH 5 Beta 2, so a file system upgrade is required to move an existing Beta 1 cluster to Beta 2.
- [HDFS-4997](#) - `libhdfs` functions now return correct error codes in `errno` in case of an error, instead of always returning 255.
- [HDFS-4451](#): HDFS balancer command returns exit code 0 on success instead of 1.
- [HDFS-4659](#): Support setting execution bit for regular files.
 - **Impact:** In CDH 5, files copied out of `copyToLocal` may now have the executable bit set if it was set when they were created or copied into HDFS.
- [HDFS-4594](#): WebHDFS open sets Content-Length header to what is specified by length parameter rather than how much data is actually returned.
 - **Impact:** In CDH 5, Content-Length header will contain the number of bytes actually returned, rather than the request length.
- [HADOOP-10020](#): Disable symlinks temporarily.
- Files named `.snapshot` or `.reserved` must not exist within HDFS.

Change in High-Availability Support

In CDH 5, the only high-availability (HA) implementation is Quorum-based storage; shared storage using NFS is no longer supported.

MapReduce

- **Important:** There is no separate tarball for MRv1. Instead, the MRv1 binaries, examples, etc., are delivered in the Hadoop tarball itself. The scripts for running MRv1 are in the `bin-mapreduce1` directory in the tarball, and the MRv1 examples are in the `examples-mapreduce1` directory. You need to do some additional configuration; follow the directions below.

To use MRv1 from a tarball installation, proceed as follows:

1. Extract the files from the tarball.

- **Note:** In the steps that follow, *install_dir* is the name of the directory into which you extracted the files.

2. Create a symbolic link as follows:

```
ln -s install_dir/bin-mapreduce1 install_dir/share/hadoop/mapreduce1/bin
```

3. Create a second symbolic link as follows:

```
ln -s install_dir/etc/hadoop-mapreduce1 install_dir/share/hadoop/mapreduce1/conf
```

4. Set the `HADOOP_HOME` and `HADOOP_CONF_DIR` environment variables in your execution environment as follows:

```
$ export HADOOP_HOME=install_dir/share/hadoop/mapreduce1
$ export HADOOP_CONF_DIR=$HADOOP_HOME/conf
```

5. Copy your existing `start-dfs.sh` and `stop-dfs.sh` scripts to *install_dir/bin-mapreduce1*
6. For convenience, add *install_dir/bin* to the `PATH` variable in your execution environment.

Apache MapReduce 2.0 (YARN) Incompatible Changes

The following incompatible changes occurred for Apache MapReduce 2.0 (YARN) between CDH 4.x and CDH 5 Beta 2:

- The `CATALINA_BASE` variable no longer determines whether a component is configured for YARN or MRv1. Use the `alternatives` command instead, and make sure `CATALINA_BASE` is not set.
- [YARN-1288](#) - YARN Fair Scheduler ACL change. Root queue defaults to everybody, and other queues default to nobody.
- YARN High Availability configurations have changed. Configuration keys have been renamed among other changes.
- The `YARN_HOME` property has been changed to `HADOOP_YARN_HOME`.
- Note the following changes to configuration properties in `yarn-site.xml`:
 - The value of `yarn.nodemanager.aux-services` should be changed from `mapreduce.shuffle` to `mapreduce_shuffle`.
 - `yarn.nodemanager.aux-services.mapreduce.shuffle.class` has been renamed to `yarn.nodemanager.aux-services.mapreduce_shuffle.class`
 - `yarn.resourcemanager.resourcemanager.connect.max.wait.secs` has been renamed to `yarn.resourcemanager.connect.max-wait.secs`
 - `yarn.resourcemanager.resourcemanager.connect.retry_interval.secs` has been renamed to `yarn.resourcemanager.connect.retry-interval.secs`
 - `yarn.resourcemanager.am.max-retries` is renamed to `yarn.resourcemanager.am.max-attempts`
 - The `YARN_HOME` environment variable used in the `yarn.application.classpath` has been renamed to `HADOOP_YARN_HOME`. Make sure you include `$HADOOP_YARN_HOME/*`, `$HADOOP_YARN_HOME/lib/*` in the classpath.
- A CDH 4 client cannot be used against a CDH 5 cluster and vice-versa. Note that YARN in CDH 4 is experimental, and suffers from the following major incompatibilities.
 - Almost all of the proto files have been renamed.
 - Several user-facing APIs have been modified as part of an API stabilization effort.

Apache HBase Incompatible Changes

Compatibility Notes for CDH 5

This section contains information that is relevant for all releases within the CDH 5 family. See the sections below for information which pertains to specific releases within CDH 5. If you are upgrading through more than one version (for instance, from CDH 5.0 to CDH 5.2), read the sections for each version, as most of the information listed applies to the given version and newer releases.

General Notes

- Rolling upgrades from CDH 4 to CDH 5 are not possible because existing CDH 4 HBase clients cannot make requests to CDH 5 servers and CDH 5 HBase clients cannot make requests to CDH 4 servers. Replication between CDH 4 and CDH 5 is not currently supported. Exposed JMX metrics in CDH 4 have been refactored and some have been removed.
- The upgrade from CDH 4 HBase to CDH 5 HBase is irreversible and requires HBase to be shutdown completely.
- As of CDH4.2, the default Split Policy changed from `ConstantSizeRegionSplitPolicy` to `IncreasingToUpperBoundRegionSplitPolicy` (ITUBRSP). This affects upgrades from CDH 4.1 or earlier to CDH 5.
- `FilterBase` no longer implements `Writable`. This means that you do not need to implement `readFields()` and `write()` methods when writing your own custom fields. Instead, put this logic into the `toByteArray` and `parseFrom` methods. See [this page](#) for an example.
- The default number of retained cell versions is reduced from 3 to 1. To increase the number of versions, you can specify the `VERSIONS` option at table creation or by altering existing tables. Starting with CDH 5.2, you can specify a global default number of versions, which will be applied to all newly created tables where the number of versions is not otherwise specified, by setting `hbase.column.max.version` to the desired number of versions in `hbase-site.xml`.
- In CDH 5 prior to 5.1.3, a Put submitted with a `KeyValue`, `KeyValue.Type.Delete` does not delete the cell. This is different from the behavior in CDH 4. In CDH 5.1.3, this behavior is changed, so that a Put submitted with a `KeyValue`, `KeyValue.Type.Delete` does delete the cell. This fix is provided in [HBASE-11788](#).

Developer API Changes

- The set of exposed APIs has been solidified. If you are using APIs outside of the [user API](#), we cannot guarantee compatibility with future minor versions.
- CDH 5 introduces a new layout for HBase build artifacts and requires POM changes if you use Maven, or JAR changes otherwise.

Previously, in CDH 4 you only needed to add a dependency for the HBase JAR:

```
<dependency>
  <groupId> org.apache.hbase </groupId>
  <artifactId> hbase </artifactId>
  <optional> true </optional>
</dependency>
```

Now, when building against CDH 5 you will need to add a dependency for the `hbase-client` JAR. The `hbase` module continues to exist as a convenient top-level wrapper for existing clients, and it pulls in all the sub-modules automatically. But it is only a simple wrapper, so its repository directory will carry no actual jars.

```
<dependency>
  <groupId>org.apache.hbase</groupId>
  <artifactId>hbase-client</artifactId>
  <version>${hbase.version}</version>
</dependency>
```

If your code uses the HBase minicluster, you can pull in the `hbase-testing-util` dependency:

```
<dependency>
  <groupId>org.apache.hbase</groupId>
  <artifactId>hbase-testing-util</artifactId>
  <version>${cdh.hbase.version}</version>
</dependency>
```

If you need to obtain all HBase JARs required to build a project, copy them from the CDH installation directory (typically `/usr/lib/hbase` for an RPM install, or `/opt/cloudera/parcels/CDH/lib/hbase` if you install using Parcels), or from the [CDH 5 HBase tarballs](#). However, for building client applications, Cloudera recommends using build tools such as Maven, rather than manually referencing JARs.

- CDH 5 introduces support for addressing cells with an empty column qualifier (a string of 0 bytes in length), but not all edge services handle that scenario correctly. In some cases, attempting to address a cell at [*rowkey*, *fam*] results in interaction with the entire column family, rather than the empty column qualifier.

Users of the HBase Shell, MapReduce, REST, and Thrift must use *family* instead of *family:* (notice the omitted ":"), to interact with an entire column family, rather than an empty column qualifier. Including the ":" will be interpreted as an interaction with the empty qualifier in the *family* column family.

API Removals

- [HBASE-7315/HBASE-7263](#) - Row lock user API has been removed.
- [HBASE-6706](#) - Removed total order partitioner.

Operator API Changes

- Many of the default configurations from CDH 4 in `hbase-default.xml` have been changed to new values in CDH 5. See [HBASE-8450](#) for a complete list of changes.
- [HBASE-6553](#) - Removed Avro Gateway. This feature was less robust and not used as much as the Thrift gateways. It has been removed upstream.
- HBase provides a metrics framework based on JMX beans. Between HBase 0.94 and 0.96, the metrics framework underwent many changes. Some beans were added and removed, some metrics were moved from one bean to another, and some metrics were renamed or removed. Click [here](#) to download the CSV spreadsheet which provides a mapping.

User API Changes

- The HBase User API (Get, Put, Result, Scanner etc; see [Apache HBase API documentation](#)) has evolved and attempts have been made to make sure the HBase Clients are source code compatible and thus should recompile without needing any source code modifications. This cannot be guaranteed however, since with the conversion to ProtoBufs, some relatively obscure APIs have been removed. Rudimentary efforts have also been made to preserve recompile compatibility with advanced APIs such as Filters and Coprocessors. These advanced APIs are still evolving and our guarantees for API compatibility are weaker here.
- As of 0.96, the User API has been marked and all attempts at compatibility in future versions will be made. A version of the javadoc that only contains the User API can be found [here](#).
- Other changes to CDH 5 HBase that require the upgrade include:
 - [HBASE-8015](#): The HBase Namespaces feature has changed HBase's HDFS file layout.
 - [HBASE-4451](#): Renamed ZooKeeper nodes.
 - [HBASE-3171](#): The `META` table in CDH 4 has been renamed to be `hbase:meta`. Similarly the ACL table has been renamed to `hbase:acl`. The `.ROOT` table has been removed.
 - [HBASE-8352](#): HBase snapshots are now saved to the `/<hbase>/hbase-snapshot` dir instead of the `/snapshot` dir. This should be handled before upgrading HDFS.
 - [HBASE-7660](#): Removed support for HFile V1. All internal HBase files in the HFile v1 format must be converted to the HFile v2 format.
 - [HBASE-6170/HBASE-8909](#) - The `hbase.regionserver.lease.period` configuration parameter has been deprecated. Use `hbase.client.scanner.timeout.period` instead.

- The behavior of the filter `MUST_PASS_ALL` changed between CDH 4 and CDH 5. In CDH 4, a `FilterList` with the default `MUST_PASS_ALL` operator return all rows (not filtering the results). In CDH 5, no results are returned when the `FilterList` is empty with the `MUST_PASS_ALL` operator. To continue using the CDH 4 behavior, modify your code to use the `scan.setLoadColumnFamiliesOnDemand(false);` method.

Compatibility Notes for CDH 5.3

- The `Put` class no longer implements `Writable`. Instead, you can change the definition to `org.apache.hadoop.hbase.io.Put` if you have only Puts, or `org.apache.hadoop.hbase.io.PutDelete` if you have a mix of Puts, Gets, and Deletes.

Compatibility Notes for CDH 5.2

- In HBase in CDH 5.1, the default value for `hbase.security.access.early_out` was set to `false`. In CDH 5.2, the default value has been changed to `true`, to maintain consistency with the behavior in CDH 4. When set to `true`, if a user is not granted access to a column family qualifier, the `AccessController` immediately throws an `AccessDeniedException`. This change to the default behavior will affect users who enabled HFile version 3 and the `AccessController` coprocessor in CDH 5.1, and then upgrade to CDH 5.2. In this case, if you prefer `hbase.security.access.early_out` to be disabled, explicitly set it to `false` in `hbase-site.xml`.
- Starting with CDH 5.2, you can specify a global default number of versions, which will be applied to all newly created tables where the number of versions is not otherwise specified, by setting `hbase.column.max.version` to the desired number of versions in `hbase-site.xml`.
- HBase in CDH 5.2 differs from Apache HBase 0.98.6 in that CDH does not include [HBASE-11546](#), which provides ZooKeeper-less region assignment. CDH omits this feature because it is an incompatible change that prevents an upgraded cluster from being rolled back to a previous version.

Developer Interface Changes

- HBase 0.98.5 removed `ClientSmallScanner` from the public API. HBase in CDH 5.2 restores the constructor to maintain backward compatibility, but in future releases of HBase, this class will no longer be public. You should change your code to use the `Scan.setSmall(true)` method instead.

Compatibility Notes for CDH 5.1

General Notes

- [HBASE-8218](#) changes `AggregationClient` by replacing the `byte[] tablename` parameters with `HTable table`. This means that coprocessors compiled against CDH 5.0.x won't run or compile in CDH 5.1 and later.
- In CDH 5.1 and later, `delete*` methods of the `Delete` class of the HBase Client API use the timestamp from the constructor, the same behavior as the `Put` class. (In previous versions, the `delete*` methods ignored the constructor's timestamp, and used the value of `HConstants.LATEST_TIMESTAMP`. This behavior was different from the behavior of the `add()` methods of the `Put` class.) See [HBASE-10964](#).
- In CDH 5 prior to 5.1.3, a `Put` submitted with a `KeyValue, KeyValue.Type.Delete` does not delete the cell. This is different from the behavior in CDH 4. In CDH 5.1.3, this behavior is changed, so that a `Put` submitted with a `KeyValue, KeyValue.Type.Delete` does delete the cell. This fix is provided in [HBASE-11788](#).
- In CDH 5.1 and newer, HBase introduces a new snapshot format ([HBASE-7987](#)). A snapshot created in HBase 0.98 cannot be read by HBase 0.96. HBase 0.98 can read snapshots produced in previous versions of HBase, and no conversion is necessary.
- In CDH 5.1, the default value for `hbase.security.access.early_out` was changed from `true` to `false`. A setting of `true` means that if a user is not granted access to a column family qualifier, the `AccessController` immediately throws an `AccessDeniedException`. *This behavior change was reverted for CDH 5.2.*

Developer Interface Changes

- HTablePool is no longer supported in CDH 5.1 and later. The HConnection object is the replacement. You create the connection once and pass it around, as with the old table pool.

```
HConnection connection = HConnectionManager.createConnection(config);
HTableInterface table = connection.getTable(tableName);
table.put(put);
table.close();
connection.close();
```

You can set the `hbase.hconnection.threads.max` property in `hbase-site.xml` to control the pool size or you can pass an `ExecutorService` to `HConnectionManager.createConnection()`.

```
ExecutorService pool = ...;
HConnection connection = HConnectionManager.createConnection(conf, pool);
```

Compatibility Notes for CDH 5 Beta Releases

Warning:

CDH 5 Beta 1 and Beta 2 are not intended for production use, and have been superseded by official releases in the CDH 5 family.

The HBase client from CDH 5 Beta 1 is not wire compatible with CDH 5 Beta 2 because of changes introduced in [HBASE-9612](#). As a consequence, CDH 5 Beta 1 users will not be able to execute a rolling upgrade to CDH 5 Beta 2 (or later). This patch unifies the way the HBase clients make requests and simplifies the internals, but breaks wire compatibility. Developers may need to recompile applications built upon the CDH 5 Beta 1 API.

As of CDH 5 Beta 1 (HBase 0.95), the value of `hbase.regionserver.checksum.verify` defaults to `true`; in earlier releases the default is `false`.

API Removals

- See [API Differences between CDH 4.5 and CDH 5 Beta 2](#).

Compatibility between CDH Beta and Apache HBase Releases

- Apache HBase 0.95.2 is not wire compatible with CDH 5 Beta 1 HBase 0.95.2.
- Apache HBase 0.96.x should be wire compatible with CDH 5 Beta 2 HBase 0.96.1.1.

Apache Hive Incompatible Changes

- **Note:** As of CDH 5, HCatalog is part of Apache Hive; incompatible changes in HCatalog are included below.

Metastore schema upgrade: CDH 5.2.0 includes Hive version 0.13.1. Upgrading from an earlier Hive version to Hive 0.13.1 or later requires a metastore schema upgrade.

Warning:

You must upgrade the metastore schema before starting the new version of Hive. Failure to do so may result in metastore corruption. See [Upgrading Hive](#).

CDH 5 includes a new offline tool called `schematool`; Cloudera recommends you use this tool to upgrade your metastore schema. See [Upgrade the Metastore Schema](#) for more information.

Hive upgrade: Upgrading Hive from CDH 4 to CDH 5, or from an earlier CDH 5 release to CDH 5.2 or later, requires several manual steps. Follow the upgrade guide closely. See [Upgrading Hive](#).

Incompatible changes between CDH 4 and CDH 5:

- The CDH 4 JDBC client is not compatible with CDH 5 HiveServer2. JDBC applications connecting to the CDH 5 HiveServer2 will require the CDH 5 JDBC client driver.
- JDBC applications will require the newer CDH 5 JDBC packages in order to connect to HiveServer2. You do not need to recompile applications for this change.
- Because of security and concurrency issues, the original Hive server (HiveServer1) and the Hive command-line interface (CLI) are deprecated in current versions of CDH 5 and will be removed in a future release. Cloudera strongly encourages you to migrate to [HiveServer2](#) and [Beeline](#) as soon as possible.
- CDH 5 Hue will not work with HiveServer2 from CDH 4.
- The `npath` function has been removed.
- Cloudera recommends that custom ObjectInspectors created for use with custom SerDes have a no-argument constructor in addition to their normal constructors, for serialization purposes. See [HIVE-5380](#) for more details.
- The SerDe interface has been changed which requires the custom SerDe modules to be reworked.
- The decimal data type format has changed as of CDH 5 Beta 2 and is not compatible with CDH 4.
- From CDH 5 Beta 2 onwards, the Parquet SerDe is part of the Hive package. The SerDe class name has changed as a result. However, there is a wrapper class for backward compatibility, so any existing Hive tables created with the Parquet SerDe will continue to work with CDH 5 Beta 2 and later Hive versions.

Incompatible changes between any earlier CDH version and CDH 5.3.x:

- CDH **5.2.0** and later clients cannot communicate with CDH 5.1.x and earlier servers. This means that you must upgrade the server before the clients.
- As of CDH **5.2.0**, `DESCRIBE DATABASE` returns additional fields: `owner_name` and `owner_type`. The command will continue to behave as expected if you identify the field you're interested in by its (string) name, but could produce unexpected results if you use a numeric index to identify the field(s).
- CDH **5.2.0** implements [HIVE-6248](#), which includes some backward-incompatible changes to the HCatalog API.
- The CDH **5.2** Hive JDBC driver is not wire-compatible with the CDH 5.1 version of HiveServer2. Make sure you upgrade Hive clients and all other Hive hosts in tandem: the server first, and then the clients.
- HiveServer 1 is deprecated in CDH **5.3**, and will be removed in a future release of CDH. Users of HiveServer 1 should upgrade to [HiveServer 2](#) as soon as possible.
- `org.apache.hcatalog` is deprecated in CDH **5.3**. All client-facing classes were moved from `org.apache.hcatalog` to `org.apache.hive.hcatalog` as of CDH 5.0 and the deprecated classes in `org.apache.hcatalog` will be removed altogether in a future release. If you are still using `org.apache.hcatalog`, you should move to `org.apache.hive.hcatalog` immediately.
- **Date partition columns:** as of Hive version 13, implemented in CDH 5.2, Hive validates the format of dates in partition columns, if they are stored as dates. A partition column with a date in invalid form can neither be used nor dropped once you upgrade to CDH 5.2 or later. To avoid this problem, do one of the following:
 - Fix any invalid dates before you upgrade. Hive expects dates in partition columns to be in the form `YYYY-MM-DD`.
 - Store dates in partition columns as strings or integers.

You can use the following SQL query to find any partition-column values stored as dates:

```
SELECT "DBS"."NAME", "TBLS"."TBL_NAME", "PARTITION_KEY_VALS"."PART_KEY_VAL"
FROM "PARTITION_KEY_VALS"
  INNER JOIN "PARTITIONS" ON "PARTITION_KEY_VALS"."PART_ID" = "PARTITIONS"."PART_ID"

  INNER JOIN "PARTITION_KEYS" ON "PARTITION_KEYS"."TBL_ID" = "PARTITIONS"."TBL_ID"

  INNER JOIN "TBLS" ON "TBLS"."TBL_ID" = "PARTITIONS"."TBL_ID"
  INNER JOIN "DBS" ON "DBS"."DB_ID" = "TBLS"."DB_ID"
  AND "PARTITION_KEYS"."INTEGER_IDX" = "PARTITION_KEY_VALS"."INTEGER_IDX"
  AND "PARTITION_KEYS"."PKEY_TYPE" = 'date';
```


Hue Incompatible Changes

- In [HUE-1859](#), the LDAP synchronization backend was moved to a generic middleware. If your code uses `DesktopSynchronizationBackendBase`, you will need to create your own middleware, and extend the new `LdapSynchronizationMiddleware`. Put that new custom middleware class in the `middleware=` line of the `[desktop]` section of `hue.ini`. The following example uses a middleware called `desktop.auth.backend.my_middleware`.

```
[desktop]
...
# Comma-separated list of Django middleware classes to use.
# See https://docs.djangoproject.com/en/1.4/ref/middleware/ for more details on
# middlewares in Django.
middleware=desktop.auth.backend.LdapSynchronizationBackend,desktop.auth.backend.my_middleware
...
```

- [HUE-1658 \[oozie\]](#) Hue depends on [OOZIE-1306](#) which is in CDH 5 Beta 2 but has not been included in any other release yet. Set the following backward compatibility flag to false to use the old frequency number/unit representation instead of the new crontab.

```
enable_cron_scheduling = false
```

- Hue 3.0.0 was a major revision of Hue. The user interface changed significantly.
- CDH 5 Hue will only work with the default Python version of the operating system on which it is being installed. For example, on RHEL/CentOS 6 you will need Python 2.6 to start Hue.

■ **Note:** RHEL 5 and CentOS 5 users will have to download Python 2.6 from the EPEL repository.

- The Beeswax daemon has been replaced by HiveServer2. Hue should therefore point to a running HiveServer2. This change involves removing the Beeswaxd code entirely and the following major updates to the `[beeswax]` section of the Hue configuration file, `hue.ini`.

```
[beeswax]
# Host where Hive server Thrift daemon is running.
# If Kerberos security is enabled, use fully-qualified domain name (FQDN).
## hive_server_host=<FQDN of Hive Server>

# Port where HiveServer2 Thrift server runs on.
## hive_server_port=10000
```

- Search bind authentication is now used by default instead of direct bind. To revert to the previous settings, use the new `search_bind_authentication` configuration property.

```
[desktop]
[[ldap]]
search_bind_authentication=false
```

- The Hue Shell app has been removed completely. This includes removing both the Shell app code and the `[shell]` section from `hue.ini`.
- YARN should be used by default.

Cloudera Impala Incompatible Changes

The Impala version covered by this documentation library contains the following incompatible changes. These are things such as file format changes, removed features, or changes to implementation, default configuration, dependencies, or prerequisites that could cause issues during or after an Impala upgrade.

Even added SQL statements or clauses can produce incompatibilities, if you have databases, tables, or columns whose names conflict with the new keywords.

Incompatible Changes Introduced in Cloudera Impala 2.1.4 / CDH 5.3.4

No incompatible changes. Because CDH 5.3.5 does not include any code changes for Impala, Impala 2.1.4 is included with both CDH 5.3.4 and 5.3.5.

- **Note:** This Impala maintenance release is only available as part of CDH 5, not under CDH 4.

Incompatible Changes Introduced in Cloudera Impala 2.1.3 / CDH 5.3.3

No incompatible changes.

- **Note:** Impala 2.1.3 is available as part of CDH 5.3.3, not under CDH 4.

Incompatible Changes Introduced in Cloudera Impala 2.1.2 / CDH 5.3.2

No incompatible changes.

- **Note:** Impala 2.1.2 is available as part of CDH 5.3.2, not under CDH 4.

Incompatible Changes Introduced in Cloudera Impala 2.1.1 / CDH 5.3.1

No incompatible changes.

Incompatible Changes Introduced in Cloudera Impala 2.1.0 / CDH 5.3.0

Changes to Prerequisites

Currently, Impala 2.1.x does not function on CPUs without the SSE4.1 instruction set. This minimum CPU requirement is higher than in previous versions, which relied on the older SSSE3 instruction set. Check the CPU level of the hosts in your cluster before upgrading to Impala 2.1.x or CDH 5.3.x.

Changes to Output Format

The “small query” optimization feature introduces some new information in the `EXPLAIN` plan, which you might need to account for if you parse the text of the plan output.

New Reserved Words

New SQL syntax introduces additional reserved words: `FOR`, `GRANT`, `REVOKE`, `ROLE`, `ROLES`, `INCREMENTAL`.

Incompatible Changes Introduced in Cloudera Impala 2.0.4 / CDH 5.2.5

No incompatible changes.

- **Note:** Impala 2.0.4 is available as part of CDH 5.2.5, not under CDH 4.

Incompatible Changes Introduced in Cloudera Impala 2.0.3 / CDH 5.2.4

- **Note:** Impala 2.0.3 is available as part of CDH 5.2.4, not under CDH 4.

Incompatible Changes Introduced in Cloudera Impala 2.0.2 / CDH 5.2.3

No incompatible changes.

- **Note:** Impala 2.0.2 is available as part of CDH 5.2.3, not under CDH 4.

Incompatible Changes Introduced in Cloudera Impala 2.0.1 / CDH 5.2.1

- The `INSERT` statement has always left behind a hidden work directory inside the data directory of the table. Formerly, this hidden work directory was named `.impala_insert_staging`. In Impala 2.0.1 and later, this directory name is changed to `_impala_insert_staging`. (While HDFS tools are expected to treat names

beginning either with underscore and dot as hidden, in practice names beginning with an underscore are more widely supported.) If you have any scripts, cleanup jobs, and so on that rely on the name of this work directory, adjust them to use the new name.

- The `abs()` function now takes a broader range of numeric types as arguments, and the return type is the same as the argument type.
- Shorthand notation for character classes in regular expressions, such as `\d` for digit, are now available again in regular expression operators and functions such as `regexp_extract()` and `regexp_replace()`. Some other differences in regular expression behavior remain between Impala 1.x and Impala 2.x releases. See [Incompatible Changes Introduced in Cloudera Impala 2.0.0 / CDH 5.2.0](#) on page 59 for details.

Incompatible Changes Introduced in Cloudera Impala 2.0.0 / CDH 5.2.0

Changes to Prerequisites

Currently, Impala 2.0.x does not function on CPUs without the SSE4.1 instruction set. This minimum CPU requirement is higher than in previous versions, which relied on the older SSSE3 instruction set. Check the CPU level of the hosts in your cluster before upgrading to Impala 2.0.x or CDH 5.2.x.

Changes to Query Syntax

The new syntax where query hints are allowed in comments causes some changes in the way comments are parsed in the `impala-shell` interpreter. Previously, you could end a `--` comment line with a semicolon and `impala-shell` would treat that as a no-op statement. Now, a comment line ending with a semicolon is passed as an empty statement to the Impala daemon, where it is flagged as an error.

Impala 2.0 and later uses a different support library for regular expression parsing than in earlier Impala versions. Now, Impala uses the [Google RE2 library](#) rather than Boost for evaluating regular expressions. This implementation change causes some differences in the allowed regular expression syntax, and in the way certain regex operators are interpreted. The following are some of the major differences (not necessarily a complete list):

- `. * ?` notation for non-greedy matches is now supported, where it was not in earlier Impala releases.
- By default, `^` and `$` now match only begin/end of buffer, not begin/end of each line. This behavior can be overridden in the regex itself using the `m` flag.
- By default, `.` does not match newline. This behavior can be overridden in the regex itself using the `s` flag.
- `\z` is not supported.
- `<` and `>` for start of word and end of word are not supported.
- Lookahead and lookbehind are not supported.
- Shorthand notation for character classes, such as `\d` for digit, is not recognized. (This restriction is lifted in Impala 2.0.1, which restores the shorthand notation.)

Changes to Output Format

In Impala 2.0 and later, `user()` returns the the full Kerberos principal string, such as `user@example.com`, in a Kerberized environment.

The changed format for the user name in secure environments is also reflected where the user name is displayed in the output of the `PROFILE` command.

In the output from `SHOW FUNCTIONS`, `SHOW AGGREGATE FUNCTIONS`, and `SHOW ANALYTIC FUNCTIONS`, arguments and return types of arbitrary `DECIMAL` scale and precision are represented as `DECIMAL(*,*)`. Formerly, these items were displayed as `DECIMAL(-1,-1)`.

Changes to Query Options

The `PARQUET_COMPRESSION_CODEC` query option has been replaced by the `COMPRESSION_CODEC` query option. See [COMPRESSION_CODEC Query Option](#) for details.

Changes to Configuration Options

The meaning of the `--idle_query_timeout` configuration option is changed, to accommodate the new `QUERY_TIMEOUT_S` query option. Rather than setting an absolute timeout period that applies to all queries, it now sets a maximum timeout period, which can be adjusted downward for individual queries by specifying a value for the `QUERY_TIMEOUT_S` query option. In sessions where no `QUERY_TIMEOUT_S` query option is specified, the `--idle_query_timeout` timeout period applies the same as in earlier versions.

The `--strict_unicode` option of `impala-shell` was removed. To avoid problems with Unicode values in `impala-shell`, define the following locale setting before running `impala-shell`:

```
export LC_CTYPE=en_US.UTF-8
```

New Reserved Words

Some new SQL syntax requires the addition of new reserved words: `ANTI`, `ANALYTIC`, `OVER`, `PRECEDING`, `UNBOUNDED`, `FOLLOWING`, `CURRENT`, `ROWS`, `RANGE`, `CHAR`, `VARCHAR`.

Changes to Data Files

The default Parquet block size for Impala is changed from 1 GB to 256 MB. This change could have implications for the sizes of Parquet files produced by `INSERT` and `CREATE TABLE AS SELECT` statements.

Although older Impala releases typically produced files that were smaller than the old default size of 1 GB, now the file size matches more closely whatever value is specified for the `PARQUET_FILE_SIZE` query option. Thus, if you use a non-default value for this setting, the output files could be larger than before. They still might be somewhat smaller than the specified value, because Impala makes conservative estimates about the space needed to represent each column as it encodes the data.

When you do not specify an explicit value for the `PARQUET_FILE_SIZE` query option, Impala tries to keep the file size within the 256 MB default size, but Impala might adjust the file size to be somewhat larger if needed to accommodate the layout for *wide* tables, that is, tables with hundreds or thousands of columns.

This change is unlikely to affect memory usage while writing Parquet files, because Impala does not pre-allocate the memory needed to hold the entire Parquet block.

Incompatible Changes Introduced in Cloudera Impala 1.4.4 / CDH 5.1.5

No incompatible changes.

- **Note:** Impala 1.4.4 is available as part of CDH 5.1.5, not under CDH 4.

Incompatible Changes Introduced in Cloudera Impala 1.4.3 / CDH 5.1.4

No incompatible changes. The SSL security fix does not require any change in the way you interact with Impala.

- **Note:** Impala 1.4.3 is available as part of CDH 5.1.4, and under CDH 4.

Incompatible Changes Introduced in Cloudera Impala 1.4.2 / CDH 5.1.3

None. Impala 1.4.2 is purely a bug-fix release. It does not include any incompatible changes.

- **Note:** Impala 1.4.2 is only available as part of CDH 5.1.3, not under CDH 4.

Incompatible Changes Introduced in Cloudera Impala 1.4.1 / CDH 5.1.2

None. Impala 1.4.1 is purely a bug-fix release. It does not include any incompatible changes.

Incompatible Changes Introduced in Cloudera Impala 1.4.0 / CDH 5.1.0

- There is a slight change to required security privileges in the Sentry framework. To create a new object, now you need the `ALL` privilege on the parent object. For example, to create a new table, view, or function requires having the `ALL` privilege on the database containing the new object.
- With the ability of `ORDER BY` queries to process unlimited amounts of data with no `LIMIT` clause, the query options `DEFAULT_ORDER_BY_LIMIT` and `ABORT_ON_DEFAULT_LIMIT_EXCEEDED` are now deprecated and have no effect.
- There are some changes to the list of reserved words. The following keywords are new:
 - `API_VERSION`
 - `BINARY`
 - `CACHED`
 - `CLASS`
 - `PARTITIONS`
 - `PRODUCED`
 - `UNCACHED`

The following were formerly reserved keywords, but are no longer reserved:

- `COUNT`
- `GROUP_CONCAT`
- `NDV`
- `SUM`
- The fix for issue [IMPALA-973](#) changes the behavior of the `INVALIDATE METADATA` statement regarding nonexistent tables. In Impala 1.4.0 and higher, the statement returns an error if the specified table is not in the metastore database at all. It completes successfully if the specified table is in the metastore database but not yet recognized by Impala, for example if the table was created through Hive. Formerly, you could issue this statement for a completely nonexistent table, with no error.

Incompatible Changes Introduced in Cloudera Impala 1.3.3 / CDH 5.0.5

No incompatible changes. The SSL security fix does not require any change in the way you interact with Impala.

- **Note:** Impala 1.3.3 is only available as part of CDH 5.0.5, not under CDH 4.

Incompatible Changes Introduced in Cloudera Impala 1.3.2 / CDH 5.0.4

With the fix for [IMPALA-1019](#), you can use HDFS caching for files that are accessed by Impala.

- **Note:** Impala 1.3.2 is only available as part of CDH 5.0.4, not under CDH 4.

Incompatible Changes Introduced in Cloudera Impala 1.3.1 / CDH 5.0.3

- In Impala 1.3.1 and higher, the `REGEXP` and `RLIKE` operators now match a regular expression string that occurs anywhere inside the target string, the same as if the regular expression was enclosed on each side by `.*`. See [REGEXP Operator](#) for examples. Previously, these operators only succeeded when the regular expression matched the entire target string. This change improves compatibility with the regular expression support for popular database systems. There is no change to the behavior of the `regexp_extract()` and `regexp_replace()` built-in functions.
- The result set for the `SHOW FUNCTIONS` statement includes a new first column, with the data type of the return value.

Incompatible Changes Introduced in Cloudera Impala 1.3.0 / CDH 5.0.0

- The `EXPLAIN_LEVEL` query option now accepts numeric options from 0 (most concise) to 3 (most verbose), rather than only 0 or 1. If you formerly used `SET EXPLAIN_LEVEL=1` to get detailed explain plans, switch to `SET EXPLAIN_LEVEL=3`. If you used the mnemonic keyword (`SET EXPLAIN_LEVEL=verbose`), you do not need to change your code because now level 3 corresponds to `verbose`.
- The keyword `DECIMAL` is now a reserved word. If you have any databases, tables, columns, or other objects already named `DECIMAL`, quote any references to them using backticks (```) to avoid name conflicts with the keyword.

■ **Note:** Although the `DECIMAL` keyword is a reserved word, currently Impala does not support `DECIMAL` as a data type for columns.

- The query option named `YARN_POOL` during the CDH 5 beta period is now named `REQUEST_POOL` to reflect its broader use with the Impala admission control feature.
- There are some changes to the list of reserved words.
 - The names of aggregate functions are no longer reserved words, so you can have databases, tables, columns, or other objects named `AVG`, `MIN`, and so on without any name conflicts.
 - The internal function names `DISTINCTPC` and `DISTINCTPCSA` are no longer reserved words, although `DISTINCT` is still a reserved word.
 - The keywords `CLOSE_FN` and `PREPARE_FN` are now reserved words.
- The HDFS property `dfs.client.file-block-storage-locations.timeout` was renamed to `dfs.client.file-block-storage-locations.timeout.millis`, to emphasize that the unit of measure is milliseconds, not seconds. Impala requires a timeout of at least 10 seconds, making the minimum value for this setting 10000. On systems not managed by Cloudera Manager, you might need to edit the `hdfs-site.xml` file in the Impala configuration directory for the new name and minimum value.

Incompatible Changes Introduced in Cloudera Impala 1.2.4

There are no incompatible changes introduced in Impala 1.2.4.

Previously, after creating a table in Hive, you had to issue the `INVALIDATE METADATA` statement with no table name, a potentially expensive operation on clusters with many databases, tables, and partitions. Starting in Impala 1.2.4, you can issue the statement `INVALIDATE METADATA table_name` for a table newly created through Hive. Loading the metadata for only this one table is faster and involves less network overhead. Therefore, you might revisit your setup DDL scripts to add the table name to `INVALIDATE METADATA` statements, in cases where you create and populate the tables through Hive before querying them through Impala.

Incompatible Changes Introduced in Cloudera Impala 1.2.3

Because the feature set of Impala 1.2.3 is identical to Impala 1.2.2, there are no new incompatible changes. See [Incompatible Changes Introduced in Cloudera Impala 1.2.2](#) on page 62 if you are upgrading from Impala 1.2.1 or 1.1.x.

Incompatible Changes Introduced in Cloudera Impala 1.2.2

The following changes to SQL syntax and semantics in Impala 1.2.2 could require updates to your SQL code, or schema objects such as tables or views:

- With the addition of the `CROSS JOIN` keyword, you might need to rewrite any queries that refer to a table named `CROSS` or use the name `CROSS` as a table alias:

```
-- Formerly, 'cross' in this query was an alias for t1
-- and it was a normal join query.
-- In 1.2.2 and higher, CROSS JOIN is a keyword, so 'cross'
-- is not interpreted as a table alias, and the query
```

```
-- uses the special CROSS JOIN processing rather than a
-- regular join.
select * from t1 cross join t2...

-- Now if CROSS is used in other context such as a table or column name,
-- use backticks to escape it.
create table `cross` (x int);
select * from `cross`;
```

- Formerly, a `DROP DATABASE` statement in Impala would not remove the top-level HDFS directory for that database. The `DROP DATABASE` has been enhanced to remove that directory. (You still need to drop all the tables inside the database first; this change only applies to the top-level directory for the entire database.)
- The keyword `PARQUET` is introduced as a synonym for `PARQUETFILE` in the `CREATE TABLE` and `ALTER TABLE` statements, because that is the common name for the file format. (As opposed to `SequenceFile` and `RCFile` where the “File” suffix is part of the name.) Documentation examples have been changed to prefer the new shorter keyword. The `PARQUETFILE` keyword is still available for backward compatibility with older Impala versions.
- New overloads are available for several operators and built-in functions, allowing you to insert their result values into smaller numeric columns such as `INT`, `SMALLINT`, `TINYINT`, and `FLOAT` without using a `CAST()` call. If you remove the `CAST()` calls from `INSERT` statements, those statements might not work with earlier versions of Impala.

Because many users are likely to upgrade straight from Impala 1.x to Impala 1.2.2, also read [Incompatible Changes Introduced in Cloudera Impala 1.2.1](#) on page 63 for things to note about upgrading to Impala 1.2.x in general.

In a Cloudera Manager environment, the catalog service is not recognized or managed by Cloudera Manager versions prior to 4.8. Cloudera Manager 4.8 and higher require the catalog service to be present for Impala. Therefore, if you upgrade to Cloudera Manager 4.8 or higher, you must also upgrade Impala to 1.2.1 or higher. Likewise, if you upgrade Impala to 1.2.1 or higher, you must also upgrade Cloudera Manager to 4.8 or higher.

Incompatible Changes Introduced in Cloudera Impala 1.2.1

The following changes to SQL syntax and semantics in Impala 1.2.1 could require updates to your SQL code, or schema objects such as tables or views:

- In Impala 1.2.1 and higher, all `NULL` values come at the end of the result set for `ORDER BY ... ASC` queries, and at the beginning of the result set for `ORDER BY ... DESC` queries. In effect, `NULL` is considered greater than all other values for sorting purposes. The original Impala behavior always put `NULL` values at the end, even for `ORDER BY ... DESC` queries. The new behavior in Impala 1.2.1 makes Impala more compatible with other popular database systems. In Impala 1.2.1 and higher, you can override or specify the sorting behavior for `NULL` by adding the clause `NULLS FIRST` or `NULLS LAST` at the end of the `ORDER BY` clause.

Impala 1.2.1 goes along with CDH 4.5 and Cloudera Manager 4.8. If you used the beta version Impala 1.2.0 that came with the beta of CDH 5, Impala 1.2.1 includes all the features of Impala 1.2.0 except for resource management, which relies on the YARN framework from CDH 5.

The new `catalogd` service might require changes to any user-written scripts that stop, start, or restart Impala services, install or upgrade Impala packages, or issue `REFRESH` or `INVALIDATE METADATA` statements:

- See [Impala Installation](#), [Upgrading Impala](#) and [Starting Impala](#), for usage information for the `catalogd` daemon.
- The `REFRESH` and `INVALIDATE METADATA` statements are no longer needed when the `CREATE TABLE`, `INSERT`, or other table-changing or data-changing operation is performed through Impala. These statements are still needed if such operations are done through Hive or by manipulating data files directly in HDFS, but in those cases the statements only need to be issued on one Impala node rather than on all nodes. See [REFRESH Statement](#) and [INVALIDATE METADATA Statement](#) for the latest usage information for those statements.
- See [The Impala Catalog Service](#) for background information on the `catalogd` service.

In a Cloudera Manager environment, the catalog service is not recognized or managed by Cloudera Manager versions prior to 4.8. Cloudera Manager 4.8 and higher require the catalog service to be present for Impala. Therefore, if you upgrade to Cloudera Manager 4.8 or higher, you must also upgrade Impala to 1.2.1 or higher. Likewise, if you upgrade Impala to 1.2.1 or higher, you must also upgrade Cloudera Manager to 4.8 or higher.

Incompatible Changes Introduced in Cloudera Impala 1.2.0 (Beta)

There are no incompatible changes to SQL syntax in Impala 1.2.0 (beta).

Because Impala 1.2.0 is bundled with the CDH 5 beta download and depends on specific levels of Apache Hadoop components supplied with CDH 5, you can only install it in combination with the CDH 5 beta.

The new `catalogd` service might require changes to any user-written scripts that stop, start, or restart Impala services, install or upgrade Impala packages, or issue `REFRESH` or `INVALIDATE METADATA` statements:

- See [Impala Installation](#), [Upgrading Impala](#) and [Starting Impala](#), for usage information for the `catalogd` daemon.
- The `REFRESH` and `INVALIDATE METADATA` statements are no longer needed when the `CREATE TABLE`, `INSERT`, or other table-changing or data-changing operation is performed through Impala. These statements are still needed if such operations are done through Hive or by manipulating data files directly in HDFS, but in those cases the statements only need to be issued on one Impala node rather than on all nodes. See [REFRESH Statement](#) and [INVALIDATE METADATA Statement](#) for the latest usage information for those statements.
- See [The Impala Catalog Service](#) for background information on the `catalogd` service.

The new resource management feature interacts with both YARN and Llama services, which are available in CDH 5. These services are set up for you automatically in a Cloudera Manager (CM) environment. For information about setting up the YARN and Llama services, see the instructions for [YARN](#) and [Llama](#) in the *CDH 5 Documentation*.

Incompatible Changes Introduced in Cloudera Impala 1.1.1

There are no incompatible changes in Impala 1.1.1.

Previously, it was not possible to create Parquet data through Impala and reuse that table within Hive. Now that Parquet support is available for Hive 10, reusing existing Impala Parquet data files in Hive requires updating the table metadata. Use the following command if you are already running Impala 1.1.1:

```
ALTER TABLE table_name SET FILEFORMAT PARQUETFILE;
```

If you are running a level of Impala that is older than 1.1.1, do the metadata update through Hive:

```
ALTER TABLE table_name SET SERDE 'parquet.hive.serde.ParquetHiveSerDe';
ALTER TABLE table_name SET FILEFORMAT
  INPUTFORMAT "parquet.hive.DeprecatedParquetInputFormat"
  OUTPUTFORMAT "parquet.hive.DeprecatedParquetOutputFormat";
```

Impala 1.1.1 and higher can reuse Parquet data files created by Hive, without any action required.

As usual, make sure to upgrade the `impala-lzo-cdh4` package to the latest level at the same time as you upgrade the Impala server.

Incompatible Change Introduced in Cloudera Impala 1.1

- The `REFRESH` statement now requires a table name; in Impala 1.0, the table name was optional. This syntax change is part of the internal rework to make `REFRESH` a true Impala SQL statement so that it can be called through the JDBC and ODBC APIs. `REFRESH` now reloads the metadata immediately, rather than marking it for update the next time any affected table is accessed. The previous behavior, where omitting the table name caused a refresh of the entire Impala metadata catalog, is available through the new `INVALIDATE METADATA` statement. `INVALIDATE METADATA` can be specified with a table name to affect a single table, or without a table name to affect the entire metadata catalog; the relevant metadata is reloaded the next time

it is requested during the processing for a SQL statement. See [REFRESH Statement](#) and [INVALIDATE METADATA Statement](#) for the latest details about these statements.

Incompatible Changes Introduced in Cloudera Impala 1.0

- If you use LZO-compressed text files, when you upgrade Impala to version 1.0, also update the `impala-lzo-cdh4` to the latest level. See [Using LZO-Compressed Text Files](#) for details.
- Cloudera Manager 4.5.2 and higher only supports Impala 1.0 and higher, and vice versa. If you upgrade to Impala 1.0 or higher managed by Cloudera Manager, you must also upgrade Cloudera Manager to version 4.5.2 or higher. If you upgrade from an earlier version of Cloudera Manager, and were using Impala, you must also upgrade Impala to version 1.0 or higher. The beta versions of Impala are no longer supported as of the release of Impala 1.0.

Incompatible Change Introduced in Version 0.7 of the Cloudera Impala Beta Release

- The defaults for the `-nn` and `-nn_port` flags have changed and are now read from `core-site.xml`. Impala prints the values of `-nn` and `-nn_port` to the log when it starts. The ability to set `-nn` and `-nn_port` on the command line is deprecated in 0.7 and may be removed in Impala 0.8.

Incompatible Change Introduced in Version 0.6 of the Cloudera Impala Beta Release

- Cloudera Manager 4.5 supports only version 0.6 of the Cloudera Impala Beta Release. It does not support the earlier beta versions. If you upgrade your Cloudera Manager installation, you must also upgrade Impala to beta version 0.6. If you upgrade Impala to beta version 0.6, you must upgrade Cloudera Manager to 4.5.

Incompatible Change Introduced in Version 0.4 of the Cloudera Impala Beta Release

- Cloudera Manager 4.1.3 supports only version 0.4 of the Cloudera Impala Beta Release. It does not support the earlier beta versions. If you upgrade your Cloudera Manager installation, you must also upgrade Impala to beta version 0.4. If you upgrade Impala to beta version 0.4, you must upgrade Cloudera Manager to 4.1.3.

Incompatible Change Introduced in Version 0.3 of the Cloudera Impala Beta Release

- Cloudera Manager 4.1.2 supports only version 0.3 of the Cloudera Impala Beta Release. It does not support the earlier beta versions. If you upgrade your Cloudera Manager installation, you must also upgrade Impala to beta version 0.3. If you upgrade Impala to beta version 0.3, you must upgrade Cloudera Manager to 4.1.2.

Llama Incompatible Changes

The following changes have been made in the Llama API to help solve synchronization problems:

- As of CDH 5.1, the Reserve API requires you to provide the `reservationId` as part of the request. In previous releases, the `reservationId` was auto-generated and returned to the user.

The `TLlamaAMReservationRequest` has an additional field called `reservation_id` which needs to be initialized to a UUID value. If you do not set this field, the request will result in an error with the error code set to `ErrorCode.RESERVATION_NO_ID_PROVIDED`.

- The Expand API now requires you to provide the `expansionId` as part of the request. In previous releases, the `expansionId` was auto-generated and returned to the user.

The `TLlamaAMReservationExpansionRequest` has an additional field called `expansion_id` which needs to be initialized to a UUID value. If you do not set this field, the request will result in an error with the error code set to `ErrorCode.EXPANSION_NO_EXPANSION_ID_PROVIDED`.

Apache Mahout Incompatible Changes

CDH 5.1 introduces the following incompatible changes.

Minor changes in behavior:

- [MAHOUT-1368](#)

The `org.apache.mahout.math.stats.OnlineSummarizer` algorithm has changed, potentially leading to different results.

- [MAHOUT-1392](#)

The streaming `KMeans` implementation has changed in how it outputs centroids when run outside of a Hadoop cluster.

- [MAHOUT-1565](#)

Major Developer API changes

- [MAHOUT-1296](#):

The following implementations have been removed:

- Clustering:
 - `DirichletMeanShift`
 - `MinHash`
 - `Eigencuts` in `o.a.m.clustering.spectral.eigencuts`
- Classification:
 - `WinnnowPerceptron`
 - `Frequent Pattern Mining`
- Collaborative Filtering:
 - All recommenders in `o.a.m.cf.taste.impl.recommender.knn`
 - `TreeClusteringRecommender` in `o.a.m.cf.taste.impl.recommender`
 - `SlopeOne` implementations in `o.a.m.cf.taste.hadoop.slopeone` and `o.a.m.cf.taste.impl.recommender.slopeone`
 - Distributed pseudo recommender in `o.a.m.cf.taste.hadoop.pseudo`

- [MAHOUT-1362](#):

The `examples/bin/build-reuters.sh` script has been removed.

Minor Developer API changes

- [MAHOUT-1280](#):

Some SSVD support code, such as `UpperTriangularMatrix`, has been moved to `mahout-math`.

- [MAHOUT-1363](#):

Scala-related math code has been moved into an `org.apache.mahout.math.scalabindings` sub-package.

Minor packaging changes

- [MAHOUT-1382](#):

Move to Guava 16.

- [MAHOUT-1364](#):

Update to require Lucene 4.6.1.

Apache Oozie Incompatible Changes

The following incompatible changes occurred between CDH 4 and CDH 5:

- [OOZIE-1680](#) - By default, at submission time, Oozie will now reject any coordinators whose frequency is faster than 5 minutes. This check can be disabled by setting the

`oozie.service.coord.check.maximum.frequency` property to `false` in `oozie-site.xml`; however, Cloudera does not recommended you disable this check or submit coordinators with frequencies greater than 5 minutes. Doing so can lead to unintended behavior and additional system stress.

- The procedure to install the Oozie Sharelib has changed. See [Configuring Oozie](#) for instructions.
- The Oozie Sharelib should be updated to the one provided with the CDH 5 package. See [Configuring Oozie](#).
- The Oozie database schema has changed and must be upgraded. See [Configuring Oozie](#) for more details. To configure Oozie using Cloudera Manager see [The Oozie Service](#).
- An Oozie client running CDH 4 will not work with an Oozie server running CDH 5 when obtaining coordinator job information. Make sure you update all the Oozie clients ([OOZIE-1482](#)).
- In CDH 4, subworkflows inherit all JAR files from their parent workflow by default. In CDH 5, this has changed so that subworkflows do not inherit JAR files by default, because the latter is actually the correct behavior. Cloudera recommends that you rework workflows and subworkflows to remove any reliance on inheriting JAR files from a parent workflow. However, setting `oozie.wf.subworkflow.classpath.inheritance` in `job.properties` or `oozie.subworkflow.classpath.inheritance` to `true` in `oozie-site.xml` will restore the old behavior. For more details, see the [Sub-workflow Action documentation](#)

As of CDH 5.2.0, a new [Hive 2 Action](#) allows Oozie to run HiveServer2 scripts. Using the Hive Action with HiveServer2 is now deprecated; you should switch to the new Hive 2 Action as soon as possible.

Apache Pig Incompatible Changes

- Apache Pig has been upgraded from version 0.11 to 0.12.
- A custom UDF must return the schema as a tuple with exactly one field.
- Added the `IN`, `CASE` and `Assert` keywords; They can't be used as variables or UDF names any more.

Cloudera Search Incompatible Changes

Incompatible changes between Cloudera Search for CDH 5.2 and Cloudera Search for CDH 5.3

Some packaging changes were made that have consequences for `CrunchIndexerTool` start-up scripts. If those startup scripts include the following line:

```
export myDriverJar=$(find $myDriverJarDir -maxdepth 1 -name \
'*.jar' ! -name '*-job.jar' ! -name '*-sources.jar')
```

That line in those scripts should be changed as follows:

```
export myDriverJar=$(find $myDriverJarDir -maxdepth 1 -name \
'search-crunch-*.jar' ! -name '*-job.jar' ! -name '*-sources.jar')
```

Incompatible changes between Cloudera Search for CDH 5 beta 2 and older versions of Cloudera Search:

The following incompatible changes occurred between Cloudera Search for CDH 5 beta 2 and older versions of Cloudera Search including both earlier versions of Cloudera Search for CDH 5 and Cloudera Search 1.x:

- Supported values for the `--reducers` option of the `MapReduceIndexer` tool change with the release of Search for CDH 5 beta 2. To use one reducer per output shard, 0 is used in Search 1.x and Search for CDH 5 beta 1. With the release of Search for CDH 5 beta 2, -2 is used for one reducer per output shard. Because of this change, commands using `--reducers 0` that were written for previous Search releases do not continue to work in the same way after upgrading to Search for CDH 5 beta 2. After upgrading to Search for CDH 5 beta 2, using `--reducers 0` results in an exception stating that zero is an illegal value.

Apache Sentry (incubating) Incompatible Changes

- CDH 5.1 introduces a new privilege model in Sentry. This introduces a backward incompatible change for Impala. Creating a new object now requires the `ALL` privilege on the parent object. For example, creating a database now requires server-level privileges (previously needed database-level) and creating a table requires database-level privileges (previously needed table-level).

- Upgrading Sentry from a release **earlier than CHD 5.2 to CDH 5.2 or later** entails a schema upgrade to the Sentry database; for more information see [or](#) .
- As of CDH 5.3, `MSCK REPAIR TABLE` now requires `ALL` privileges for the table (previously this statement required `ALL` privileges on the parent database for the table).

Apache Spark Incompatible Changes

- As of CDH 5.1, before you can run Spark in standalone mode, you must set the `spark.master` property in `/etc/spark/conf/spark-defaults.conf`, as follows:

```
spark.master=spark://MASTER_IP:MASTER_PORT
```

where `MASTER_IP` is the IP address of the host the Spark master is running on and `MASTER_PORT` is the port.

This setting means that all jobs will run in standalone mode by default; you can override the default on the command line.

- The CDH 5.1 release of Spark includes changes that will enable Spark to avoid breaking compatibility in the future. As a result, most applications will require a recompile to run against Spark 1.0, and some will require changes in source code. The details are as follows:
 - There are two changes in the core Scala API:
 - The `cogroup` and `groupByKey` operators now return `Iterators` over their values instead of `Seqs`. This change means that the set of values corresponding to a particular key need not all reside in memory at the same time.
 - `SparkContext.jarOfClass` now returns `Option[String]` instead of `Seq[String]`.
 - Spark's Java APIs have been updated to accommodate Java 8 lambdas. See [Migrating from pre-1.0 Versions of Spark](#) for more information.

■ **Note:**

CDH 5.1 does not support Java 8, which is [supported](#) as of CDH 5.3.

- If you have [uploaded the Spark assembly JAR file to HDFS](#), you must upload the new version of the file each time you upgrade Spark to a new minor CDH release (for example, any CDH 5.2.x or 5.3.x release, including 5.2.0 and 5.3.0) . You may also need to modify the configured path for the file; see the next bullet below.
- As of CDH 5.2, the configured paths for `spark.eventLog.dir`, `spark.history.fs.logDirectory`, and the `SPARK_JAR` environment variable have changed in a way that may not be backward-compatible. By default, those paths now refer to the local filesystem. To make sure everything works as before, modify the paths as following:
 - For HDFS, if this is not a federated cluster, prepend `hdfs:` to the path.
 - For HDFS in a federated cluster, prepend `viewfs:` to the path.Alternatively, you can prepend the value of `fs.defaultFS`, set in `core-site.xml` in the HDFS configuration.
- The following changes introduced in CDH 5.2 may affect existing applications:
 - The default for I/O compression is now Snappy (changed from LZF).
 - PySpark now performs external spilling during aggregations.
- As of CDH 5.2, the following Spark-related artifacts are no longer published as part of the Cloudera repository:
 - `spark-assembly`: The `spark-assembly jar` is used internally by Spark distributions when executing Spark applications and should not be referenced directly. Instead, projects should add dependencies for those parts of the Spark project that are being used, for example, `spark-core`.
 - `spark-yarn`
 - `spark-tools`

- spark-examples
- spark-repl
- Spark 1.2, on which CDH 5.3 is based, does not expose a transitive dependency on the Guava library. As a result, projects that use Guava but don't explicitly add it as a dependency will need to be modified: the dependency must be added to the project and also packaged with the job.
- The CDH 5.3 version of Spark 1.2 differs from the Apache Spark 1.2 release in using Akka version 2.2.3, the version used by Spark 1.1 and CDH 5.2. Apache Spark 1.2 uses Akka version 2.3.4.

Apache Sqoop Incompatible Changes

Upgrading Sqoop 2 from an earlier release to CDH 5.2.0 and later entails a schema upgrade to the repository database; see [Upgrading Sqoop 2 from an Earlier CDH 5 Release](#).

Apache Whirr Incompatible Changes

There are currently no incompatible changes.

Apache ZooKeeper Incompatible Changes

There are no known incompatible changes between CDH 4 and CDH 5.

Known Issues in CDH 5

Performance Known Issues

- **Important:** For best practices, and solutions to known performance problems, see [Improving Performance](#).

Install and Upgrade Known Issues

Potential job failures during YARN rolling upgrades to CDH 5.3.4

Problem: A MapReduce security fix introduced a compatibility issue that results in job failures during YARN rolling upgrades from CDH 5.3.3 to CDH 5.3.4.

Release affected: CDH 5.3.4

Release containing the fix: CDH 5.3.5

Workarounds: You can use any one of the following workarounds for this issue:

- Upgrade to CDH 5.3.5.
 - Restart any jobs that might have failed during the upgrade.
 - Explicitly set the version of MapReduce to be used so it is picked on a per-job basis.
1. Update the YARN property, **MR Application Classpath** (`mapreduce.application.classpath`), either in Cloudera Manager or in the `mapred-site.xml` file. Remove all existing values and add a new entry: `<parcel-path>/lib/hadoop-mapreduce/*`, where `<parcel-path>` is the absolute path to the parcel installation. For example, the default installation path for the CDH 5.3.3 parcel would be: `/opt/cloudera/parcels/CDH-5.3.3-1.cd5.3.3.p0.5/lib/hadoop-mapreduce/*`.
 2. Wait until jobs submitted with the above client configuration change have run to completion.
 3. Upgrade to CDH 5.3.4.
 4. Update the **MR Application Classpath** (`mapreduce.application.classpath`) property to point to the new CDH 5.3.4 parcel.

Do *not* delete the old parcel until after all jobs submitted prior to the upgrade have finished running.

NameNode Incorrectly Reports Missing Blocks During Rolling Upgrade

Problem: During a rolling upgrade to CDH 5.3.3, the NameNode may report missing blocks after rolling multiple DataNodes. This is caused by a race condition with block reporting between the DataNode and the NameNode. No permanent data loss occurs, but data can be unavailable for up to six hours before the problem corrects itself.

Releases affected: CDH 5.0.6, 5.1.5, 5.2.5, 5.3.3

Releases containing the fix: 5.3.4

Workaround:

To avoid the problem: Cloudera advises skipping CDH 5.3.3, and upgrading to CDH 5.3.4 instead. Before upgrading to CDH 5.3.4, see [Install and Upgrade Known Issues](#) on page 69.

If you have already completed an upgrade to an affected release, or are installing a new cluster: In this case you can continue to run CDH 5.3.3, or upgrade to CDH 5.3.4, as you choose.

— Extra step needed on Ubuntu Trusty if you add the Cloudera repository

If you install or upgrade CDH on Ubuntu Trusty using the command line, and add the Cloudera repository yourself (rather than using the "1-click Install" method) you need to perform an additional step to ensure that you get the CDH version of ZooKeeper, rather than the version that is bundled with Trusty. See [Steps to Install CDH 5 Manually](#).

— Upgrading `hadoop-kms` from 5.2.x and 5.3.x releases fails on SLES

Upgrading `hadoop-kms` fails on SLES when you try to upgrade an existing version from 5.2.x releases earlier than 5.2.4, and from 5.3.x releases earlier than 5.3.2. For details and troubleshooting instructions, see [Troubleshooting: upgrading hadoop-kms from 5.2.x and 5.3.x releases on SLES](#).

— No in-place upgrade to CDH 5 from CDH 4

Cloudera fully supports upgrade from Cloudera Enterprise 4.x and CDH 4.x to Cloudera Enterprise 5.0. Upgrade requires uninstalling the CDH 4 packages before installing CDH 5 packages. See the [CDH 5 upgrade documentation](#) for instructions.

— No upgrade directly from CDH 3 to CDH 5

You must upgrade to CDH 4, then to CDH 5. See the [CDH 4 documentation](#) for instructions on upgrading from CDH 3 to CDH 4.

— Upgrading to CDH 5.2 or later requires an HDFS upgrade

Upgrading to CDH 5.2 or later requires an HDFS Upgrade, and other steps not usually required for a minor-release upgrade. See [Upgrading from an Earlier CDH 5 Release to the Latest Release](#) for further information.

— Upgrading from CDH 4 requires an HDFS upgrade

Upgrading from CDH 4 requires an HDFS Upgrade. See [Upgrading from CDH 4 to CDH 5](#) for further information.

After upgrading from a release earlier than CDH 4.6, you may see reports of corrupted files

Some older versions of CDH do not handle DataNodes with a large number of blocks correctly. The problem exists on versions 4.6, 4.7, 4.8, 5.0, and 5.1. The symptom is that the NameNode Web UI and the `fsck` command incorrectly report missing blocks, even when those blocks are present.

The cause of the problem is that if the DataNode attempts to send a block report that is larger than the maximum RPC buffer size, the NameNode rejects the report. This prevents the NameNode from becoming aware of the blocks on the affected DataNodes. The maximum buffer size is controlled by the `ipc.maximum.data.length` property, which defaults to 64 MB.

This problem does not exist in CDH 4.5 and earlier because there is no maximum RPC buffer size in these versions. Starting in CDH 5.2, DataNodes now send individual block reports for each storage volume, which mitigates the problem.

Bug: [HADOOP-9676](#)

Severity: Medium

Workaround: Immediately after upgrading, increase the value of `ipc.maximum.data.length`; Cloudera recommends doubling the default value, from 64 MB to 128 MB:

```
<property>
  <name>ipc.maximum.data.length</name>
  <value>134217728</value>
</property>
```

- In a Cloudera Manager installation, set this property in the `hdfs_service_config_safety_valve`.
- In a command-line-only installation, add and set this property in `core-site.xml`.

After setting `ipc.maximum.data.length`, restart the NameNode(s).

—Must build native libraries when installing from tarballs

When installing Hadoop from Cloudera tarballs, you must build your own native libraries. The tarballs do not include libraries that are built for the different distributions and architectures.

Apache Flume Known Issues

— Hive sink support

Flume does not provide a native sink that stores the data that can be directly consumed by Hive.

Bug: [FLUME-1008](#)

Severity: Medium

Workaround: None

— Fast Replay does not work with encrypted File Channel

If an encrypted file channel is set to use fast replay, the replay will fail and the channel will fail to start.

Bug: [FLUME-1885](#)

Severity: Low

Workaround: Disable fast replay for the encrypted channel by setting `use-fast-replay` to false.

Apache Hadoop Known Issues

— Deprecated Properties

In Hadoop 2.0.0 and later, a number of Hadoop and HDFS properties have been deprecated. (The change dates from Hadoop 0.23.1, on which the Beta releases of CDH 4 were based). A list of deprecated properties and their replacements can be found at

<http://archive.cloudera.com/cdh5/cdh/5/hadoop/hadoop-project-dist/hadoop-common/DeprecatedProperties.html>.

HDFS

— Upgrade Requires an HDFS Upgrade

Upgrading from any release earlier than CDH 5.2.0 to CDH 5.2.0 or later requires an HDFS Upgrade.

— Optimizing HDFS Encryption at Rest Requires Newer openssl Library on Some Systems

CDH 5.3 implements the **Advanced Encryption Standard New Instructions** (AES-NI), which provide substantial performance improvements. To get these improvements, you need a recent version of `libcrypto.so` on HDFS and MapReduce client hosts -- that is, any host from which you originate HDFS or MapReduce requests. Many OS versions have an older version of the library that does not support AES-NI.

See [HDFS Data At Rest Encryption](#) in the *Encryption* section of the *Cloudera Security* guide for instructions for obtaining the right version.

— Other HDFS Encryption Known Issues

Potentially Incorrect Initialization Vector Calculation in HDFS Encryption

A mathematical error in the calculation of the Initialization Vector (IV) for encryption and decryption in HDFS could cause data to appear corrupted when read. The IV is a 16-byte value input to encryption and decryption ciphers. The calculation of the IV implemented in HDFS was found to be subtly different from that used by Java and OpenSSL cryptographic routines. The result is that data could possibly appear to be corrupted when it is read from a file inside an Encryption Zone.

Fortunately, the probability of this occurring is extremely small. For example, the maximum size of a file in HDFS is 64 TB. This enormous file would have a 1-in-4- million chance of hitting this condition. A more typically sized file of 1 GB would have a roughly 1-in-274-billion chance of hitting the condition.

Severity: Low

Workaround: If you are using the experimental HDFS encryption feature in CDH 5.2, upgrade to CDH 5.3 and verify the integrity of all files inside an Encryption Zone.

— Solr, Oozie and HttpFS fail when KMS and SSL are enabled using self-signed certificates

When the KMS service is added and SSL is enabled, Solr, Oozie and HttpFS are not automatically configured to trust the KMS's self-signed certificate and you might see the following error.

```
org.apache.oozie.service.AuthorizationException: E0501: Could not perform authorization
operation,
sun.security.validator.ValidatorException: PKIX path building failed:
sun.security.provider.certpath.SunCertPathBuilderException:
unable to find valid certification path to requested target
```

Severity: Medium

Workaround: You must explicitly load the relevant truststore with the KMS certificate to allow these services to communicate with the KMS.

Solr, Oozie: Add the following arguments to their environment safety valve so as to load the truststore with the required KMS certificate.

```
CATALINA_OPTS="-Djavax.net.ssl.trustStore=/etc/path-to-truststore.jks
-Djavax.net.ssl.trustStorePassword=<password>"
```

HttpFS: Add the following arguments to the **Java Configuration Options for HttpFS** property.

```
-Djavax.net.ssl.trustStore=/etc/path-to-truststore.jks
-Djavax.net.ssl.trustStorePassword=<password>
```

— DistCp between unencrypted and encrypted locations fails

By default, DistCp compares checksums provided by the filesystem to verify that data was successfully copied to the destination. However, when copying between unencrypted and encrypted locations, the filesystem checksums will not match since the underlying block data is different.

Severity: Low

Workaround: Specify the `-skipcrccheck` and `-update` distcp flags to avoid verifying checksums.

— Cannot move encrypted files to trash

With HDFS encryption enabled, you cannot move encrypted files or directories to the trash directory.

Bug: [HDFS-6767](#)

Severity: Low

Workaround: To remove encrypted files/directories, use the following command with the `-skipTrash` flag specified to bypass trash.

```
rm -r -skipTrash /testdir
```

— *If you install CDH using packages, HDFS NFS gateway works out of the box only on RHEL-compatible systems*

Because of a bug in native versions of `portmap/rpcbind`, the HDFS NFS gateway does not work out of the box on SLES, Ubuntu, or Debian systems if you install CDH from the command-line, using packages. It does work on [supported versions](#) of RHEL-compatible systems on which `rpcbind-0.2.0-10.el6` or later is installed, and it does work if you use Cloudera Manager to install CDH, or if you start the gateway as root.

Bug: [731542](#) (Red Hat), [823364](#) (SLES), [594880](#) (Debian)

Severity: High

Workarounds and caveats:

- On Red Hat and similar systems, make sure `rpcbind-0.2.0-10.el6` or later is installed.
- On SLES, Debian, and Ubuntu systems, do one of the following:
 - Install CDH using Cloudera Manager; *or*
 - As of CDH 5.1, start the NFS gateway as root; *or*
 - [Start the NFS gateway without using packages](#); *or*
 - You can use the gateway by running `rpcbind` in insecure mode, using the `-i` option, but keep in mind that this allows anyone from a remote host to bind to the portmap.

— *No error when changing permission to 777 on .snapshot directory*

Snapshots are read-only; running `chmod 777` on the `.snapshots` directory does not change this, but does not produce an error (though other illegal operations do).

Bug: [HDFS-4981](#)

Severity: Low

Workaround: None

— *Snapshot operations are not supported by ViewFileSystem*

Bug: None

Severity: Low

Workaround: None

— *Snapshots do not retain directories' quotas settings*

Bug: [HDFS-4897](#)

Severity: Medium

Workaround: None

— *NameNode cannot use wildcard address in a secure cluster*

In a secure cluster, you cannot use a wildcard for the NameNode's RPC or HTTP bind address. For example, `dfs.namenode.http-address` must be a real, routable address and port, not `0.0.0.0.<port>`. This should affect you only if you are running a secure cluster *and* your NameNode needs to bind to multiple local addresses.

Bug: [HDFS-4448](#)

Severity: Medium

Workaround: None

— *Permissions for `dfs.namenode.name.dir` incorrectly set.*

Hadoop daemons should set permissions for the `dfs.namenode.name.dir` (or `dfs.name.dir`) directories to `drwx-----` (700), but in fact these permissions are set to the file-system default, usually `drwxr-xr-x` (755).

Bug: [HDFS-2470](#)

Severity: Low

Workaround: Use `chmod` to set permissions to 700. See [Configuring Local Storage Directories for Use by HDFS](#) for more information and instructions.

— *`hadoop fsck -move` does not work in a cluster with host-based Kerberos*

Bug: None

Severity: Low

Workaround: Use `hadoop fsck -delete`

— *HttpFS cannot get delegation token without prior authenticated request.*

A request to obtain a delegation token cannot initiate an SPNEGO authentication sequence; it must be accompanied by an authentication cookie from a prior SPNEGO authentication sequence.

Bug: [HDFS-3988](#)

Severity: Low

Workaround: Make another WebHDFS request (such as `GETHOMEDIR`) to initiate an SPNEGO authentication sequence and then make the delegation token request.

— *DistCp does not work between a secure cluster and an insecure cluster in some cases*

See the upstream bug reports for details.

Bugs: [HDFS-7037](#), [HADOOP-10016](#), [HADOOP-8828](#)

Severity: High

Workaround: None

— *Using DistCp with Hftp on a secure cluster using SPNEGO requires that the `dfs.https.port` property be configured*

In order to DistCp using Hftp from a secure cluster using SPNEGO, you must configure the `dfs.https.port` property on the client to use the HTTP port (50070 by default).

Bug: [HDFS-3983](#)

Severity: Low

Workaround: Configure `dfs.https.port` to use the HTTP port on the client

— *Non-HA DFS Clients do not attempt reconnects*

This problem means that streams cannot survive a NameNode restart or network interruption that lasts longer than the time it takes to write a block.

Bug: [HDFS-4389](#)

— *Offline Image Viewer (OIV) tool regression: missing Delimited outputs.*

Bugs: [HDFS-6673](#), [HDFS-5952](#)

Severity: Medium

Workaround: Set up `dfs.namenode.legacy-oiv-image.dir` to an appropriate directory on the secondary NameNode (or standby NameNode in an HA configuration), and use `hdfs oiv_legacy` to process the legacy format of the OIV `fsimage`.

— *The active NameNode will not accept an fsimage sent from the standby during rolling upgrade*

The result is that the NameNodes fail to checkpoint until the upgrade is finalized.

■ **Note:**

Rolling upgrade is supported only for clusters managed by Cloudera Manager; you cannot do a rolling upgrade in a command-line-only deployment.

Bug: [HDFS-7185](#)

Severity: Medium

Workaround: None.

— *On a DataNode with a large number of blocks, the block report may exceed the maximum RPC buffer size*

Bug: None

Workaround: Increase the value `ipc.maximum.data.length` in `hdfs-site.xml`:

```
<property>
  <name>ipc.maximum.data.length</name>
  <value>268435456</value>
</property>
```

MapReduce

— *Starting an unmanaged ApplicationMaster may fail*

Starting a custom Unmanaged ApplicationMaster may fail due to a race in getting the necessary tokens.

Bug: [YARN-1577](#)

Severity: Low

Workaround: Try to get the tokens again; the custom unmanaged ApplicationMaster should be able to fetch the necessary tokens and start successfully.

— *Job movement between queues does not persist across ResourceManager restart*

CDH 5 adds the capability to move a submitted application to a different scheduler queue. This queue placement is not persisted across ResourceManager restart or failover, which resumes the application in the original queue.

Bug: [YARN-1558](#)

Severity: Medium

Workaround: After ResourceManager restart, re-issue previously issued move requests.

— *No JobTracker becomes active if both JobTrackers are migrated to other hosts*

If JobTrackers in an High Availability configuration are shut down, migrated to new hosts, then restarted, no JobTracker becomes active. The logs show a `Mismatched address` exception.

Bug: None

Severity: Low

Workaround: After shutting down the JobTrackers on the original hosts, and before starting them on the new hosts, delete the ZooKeeper state using the following command:

```
$ zkCli.sh rmr /hadoop-ha/<logical name>
```

— *Hadoop Pipes may not be usable in an MRv1 Hadoop installation done through tarballs*

Under MRv1, MapReduce's C++ interface, Hadoop Pipes, may not be usable with a Hadoop installation done through tarballs unless you build the C++ code on the operating system you are using.

Bug: None

Severity: Medium

Workaround: Build the C++ code on the operating system you are using. The C++ code is present under `src/c++` in the tarball.

— *Task-completed percentage may be reported as slightly under 100% in the web UI, even when all of a job's tasks have successfully completed.*

Bug: None

Severity: Low

Workaround: None

— *Spurious warning in MRv1 jobs*

The `mapreduce.client.genericoptionsparser.used` property is not correctly checked by `JobClient` and this leads to a spurious warning.

Bug: None

Severity: Low

Workaround: MapReduce jobs using `GenericOptionsParser` or implementing `Tool` can remove the warning by setting this property to true.

— *Oozie workflows will not be recovered in the event of a JobTracker failover on a secure cluster*

Delegation tokens created by clients (via `JobClient#getDelegationToken()`) do not persist when the JobTracker [fails over](#). This limitation means that Oozie workflows will not be recovered successfully in the event of a failover on a secure cluster.

Bug: None

Severity: Medium

Workaround: Re-submit the workflow.

— *Encrypted shuffle in MRv2 does not work if used with `LinuxContainerExecutor` and encrypted web UIs.*

In MRv2, if the `LinuxContainerExecutor` is used (usually as part of Kerberos security), and `hadoop.ssl.enabled` is set to `true` (See [Configuring Encrypted Shuffle, Encrypted Web UIs, and Encrypted HDFS Transport](#)), then the encrypted shuffle does not work and the submitted job fails.

Bug: [MAPREDUCE-4669](#)

Severity: Medium

Workaround: Use encrypted shuffle with Kerberos security without encrypted web UIs, or use encrypted shuffle with encrypted web UIs without Kerberos security.

— *Link from `ResourceManager` to `Application Master` does not work when the Web UI over HTTPS feature is enabled.*

In MRv2 (YARN), if `hadoop.ssl.enabled` is set to `true` (use HTTPS for web UIs), then the link from the `ResourceManager` to the running MapReduce Application Master fails with an HTTP Error 500 because of a PKIX exception.

A job can still be run successfully, and, when it finishes, the link to the job history does work.

Bug: [YARN-113](#)

Severity: Low

Workaround: Don't use encrypted web UIs.

— *Hadoop client JARs don't provide all the classes needed for clean compilation of client code*

The compile does succeed, but you may see warnings as in the following example:

```
$ javac -cp '/usr/lib/hadoop/client/*' -d wordcount_classes WordCount.java
org/apache/hadoop/fs/Path.class(org/apache/hadoop/fs:Path.class): warning: Cannot find
annotation method 'value()'
in type 'org.apache.hadoop.classification.InterfaceAudience.LimitedPrivate': class file
for org.apache.hadoop.classification.InterfaceAudience not found
1 warning
```

- **Note:** This means that the example at the bottom of the page on managing Hadoop API dependencies (see "Using the CDH 4 Maven Repository" under [CDH Version and Packaging Information](#)) will produce a similar warning.

Bug:

Severity: Low

Workaround: None

— *The ulimits setting in /etc/security/limits.conf is applied to the wrong user if security is enabled.*

Bug: <https://issues.apache.org/jira/browse/DAEMON-192>

Severity: Low

Anticipated Resolution: None

Workaround: To increase the `ulimits` applied to `DataNodes`, you must change the `ulimit` settings for the root user, not the `hdfs` user.

— *Must set `yarn.resourcemanager.scheduler.address` to routable host:port when submitting a job from the ResourceManager*

When you submit a job from the ResourceManager, `yarn.resourcemanager.scheduler.address` must be set to a real, routable address, not the wildcard 0.0.0.0.

Bug: None

Severity: Low

Workaround: Set the address, in the form `host:port`, either in the client-side configuration, or on the command line when you submit the job.

— *Amazon S3 copy may time out*

The Amazon S3 filesystem does not support renaming files, and performs a copy operation instead. If the file to be moved is very large, the operation can time out because S3 does not report progress to the TaskTracker during the operation.

Bug: [MAPREDUCE-972](#)

Severity: Low

Workaround: Use `-Dmapred.task.timeout=15000000` to increase the MR task timeout.

Task Controller Changed from `DefaultTaskController` to `LinuxTaskController`

In CDH 5, the MapReduce task controller is changed from `DefaultTaskController` to `LinuxTaskController`. The new task controller has different directory ownership requirements which can cause jobs to fail. You can switch back to `DefaultTaskController` by adding the following to the MapReduce Advanced Configuration Snippet if you use Cloudera Manager, or directly to `mapred-default.xml` otherwise.

```
<property>
  <name>mapreduce.tasktracker.taskcontroller</name>
```

```
<value>org.apache.hadoop.mapred.DefaultTaskController</value>
</property>
```

—Out-of-memory errors may occur with Oracle JDK 1.8

The total JVM memory footprint for JDK8 can be larger than that of JDK7 in some cases. This may result in out-of-memory errors.

Bug: None

Severity: Medium

Workaround: Increase max default heap size (`-Xmx`). In the case of MapReduce, for example, increase **Reduce Task Maximum Heap Size** in Cloudera Manager (`mapred.reduce.child.java.opts`, or `mapreduce.reduce.java.opts` for YARN) to avoid out-of-memory errors during the shuffle phase.

Apache HBase Known Issues

— Some New Features in HBase 0.98 Not Supported in CDH 5.3

The following features, introduced upstream in HBase 0.98, are not supported in CDH 5.3:

- Visibility labels
- Transparent server-side encryption
- Stripe compaction
- Distributed log replay

For more information, see [New Features and Changes for HBase in CDH 5](#).

— HBase moves to Protoc 2.5.0

This change may cause JAR conflicts with applications that have older versions of `protobuf` in their Java classpath.

Bug: None

Severity: Medium

Workaround: Update applications to use Protoc 2.5.0.

— Write performance may be a little slower in CDH 5 than in CDH 4

Bug: None

Severity: Low

Workaround: None, but see [Checksums](#) in the HBase section of the *Cloudera Installation and Upgrade* guide.

Must explicitly add permissions for `owner` users before upgrading from 4.1. *x*

In CDH 4.1. *x*, an HBase table could have an owner. The `owner` user had full administrative permissions on the table (`RWXCA`). These permissions were implicit (that is, they were not stored explicitly in the HBase `acl` table), but the code checked them when determining if a user could perform an operation.

The `owner` construct was removed as of CDH 4.2.0, and the code now relies exclusively on entries in the `acl` table. Since table owners do not have an entry in this table, their permissions are removed on upgrade from CDH 4.1. *x* to CDH 4.2.0 or later.

Bug: None

Severity: Medium

Anticipated Resolution: None; use workaround

Workaround: Add permissions for owner users before upgrading from CDH 4.1. x. You can automate the task of making the owner users' implicit permissions explicit, using code similar to the following. (Note that this snippet is intended only to give you an idea of how to proceed; it may not compile and run as it stands.)

```
PERMISSIONS = 'RWXCA'

tables.each do |t|
  table_name = t.getNameAsString
  owner = t.getOwnerString
  LOG.warn( "Granting " + owner + " with
    " + PERMISSIONS + " for
    table " + table_name)
  user_permission = UserPermission.new(owner.to_java_bytes, table_name.to_java_bytes,
                                         nil, nil, PERMISSIONS.to_java_bytes)
  protocol.grant(user_permission)
end
```

— [Change in default splitting policy from ConstantSizeRegionSplitPolicy to IncreasingToUpperBoundRegionSplitPolicy may create too many splits](#)

This affects you only if you are upgrading from CDH 4.1 or earlier.

Split size is the number of regions that are on this server that all are part of the same table, squared, times the region flush size *or* the maximum region split size, whichever is smaller. For example, if the flush size is 128MB, then on first flush we will split, making two regions that will split when their size is $2 * 2 * 128\text{MB} = 512\text{MB}$. If one of these regions splits, there are three regions and now the split size is $3 * 3 * 128\text{MB} = 1152\text{MB}$, and so on until we reach the configured maximum file size, and then from then, we'll use that.

This new default policy could create many splits if you have many tables in your cluster.

This default split size has also changed - from 64MB to 128MB; and the region eventual split size, `hbase.hregion.max.filesize`, is now 10GB (it was 1GB).

Bug: None

Severity: Medium

Anticipated Resolution: None; use workaround

Workaround: If find you are getting too many splits, either go back to the old split policy or increase the `hbase.hregion.memstore.flush.size`.

— [In a cluster where the HBase directory in HDFS is encrypted, an IOException can occur if the BulkLoad staging directory is not in the same encryption zone as the HBase root directory.](#)

If you have encrypted the HBase root directory (`hbase.rootdir`) and you attempt a BulkLoad where the staging directory is in a different encryption zone from the HBase root directory, you may encounter errors such as:

```
org.apache.hadoop.ipc.RemoteException(java.io.IOException):
/tmp/output/f/5237a8430561409bb641507f0c531448 can't be moved into an encryption zone.
```

Bug: None

Anticipated Resolution: None; use workaround

Severity: Medium

Workaround: Configure `hbase.bulkload.staging.dir` to point to a location within the same encryption zone as the HBase root directory.

— [In a non-secure cluster, MapReduce over HBase does not properly handle splits in the BulkLoad case](#)

You may see errors because of:

- missing permissions on the directory that contains the files to bulk load
- missing ACL rights for the table/families

Bug: None

Anticipated Resolution: None; use workaround

Severity: Medium

Workaround: In a non-secure cluster, execute `BulkLoad` as the `hbase` user.

- **Note:** For important information about configuration that is required for `BulkLoad` in a secure cluster as of CDH 4.3, see the [Apache HBase Incompatible Changes](#) on page 52 subsection under Incompatible Changes in these Release Notes.

— Pluggable compaction and scan policies via coprocessors (HBASE-6427) not supported

Cloudera does not provide support for user-provided custom coprocessors.

Bug: [HBASE-6427](#)

Severity: Low

Workaround: None

— Custom constraints coprocessors (HBASE-4605) not supported

The constraints coprocessor feature provides a framework for constraints and requires you to add your own custom code. Cloudera does not support user-provided custom code, and hence does not support this feature.

Bug: [HBASE-4605](#)

Severity: Low

Workaround: None

— Pluggable split key policy (HBASE-5304) not supported

Cloudera supports the two split policies that are supplied and tested: `ConstantSizeSplitPolicy` and `PrefixSplitKeyPolicy`. The code also provides a mechanism for custom policies that are specified by adding a class name to the `HTableDescriptor`. Custom code added via this mechanism must be provided by the user. Cloudera does not support user-provided custom code, and hence does not support this feature.

Bug: [HBASE-5304](#)

Severity: Low

Workaround: None

— HBase may not tolerate HDFS root directory changes

While HBase is running, do not stop the HDFS instance running under it and restart it again with a different root directory for HBase.

Bug: None

Severity: Medium

Workaround: None

— AccessController postOperation problems in asynchronous operations

When security and Access Control are enabled, the following problems occur:

- If a `Delete Table` fails for a reason other than missing permissions, the access rights are removed but the table may still exist and may be used again.
- If `hbaseAdmin.modifyTable()` is used to delete column families, the rights are not removed from the Access Control List (ACL) table. The `postOperation` is implemented only for `postDeleteColumn()`.
- If `Create Table` fails, full rights for that table persist for the user who attempted to create it. If another user later succeeds in creating the table, the user who made the failed attempt still has the full rights.

Bug: [HBASE-6992](#)

Severity: Medium

Workaround: None

— [Native library not included in tarballs](#)

The native library that enables Region Server page pinning on Linux is not included in tarballs. This could impair performance if you install HBase from tarballs.

Bug: None

Severity: Low

Workaround: None

— [Out Of Memory Exception in Thrift Server Without Framed Transport](#)

Without framed transport enabled, the Thrift server attempts to allocate a large amount of memory when it receives an invalid packet. This is a separate issue from [HBASE-11052](#), which has already been addressed.

Bug: [THRIFT-2660](#)

Severity: Low

Workaround: Enable framed transport on the Thrift server and clients if possible. Otherwise, increase the heap size to 8 GB so that the large memory allocation can succeed when an invalid packet is received.

Apache Hive Known Issues

- **Note:** As of CDH 5, HCatalog is part of Apache Hive; HCatalog known issues are included [below](#).

— [UDF infile\(\) does not accept arguments of type CHAR or VARCHAR](#)

Bug: [HIVE-6637](#)

Severity: Low

Workaround: Cast the argument to type String.

— [UDF printf\(\) does not accept arguments of type CHAR or VARCHAR](#)

Bug: [HIVE-6620](#)

Severity: Low

Workaround: Cast the argument to type String.

— [Hive's Decimal type cannot be stored in Parquet and Avro](#)

Tables containing decimal columns can't use Parquet or the Avro storage engine.

Bug: [HIVE-6367](#) and [HIVE-5823](#)

Severity: Low

Workaround: Use a different file format.

— [Hive creates an invalid table if you specify more than one partition with `alter table`](#)

Hive (in all known versions from 0.7) allows you to configure multiple partitions with a single `alter table` command, but the configuration it creates is invalid for both Hive and Impala.

Bug: None

Severity: Medium

Resolution: Use workaround.

Workaround:

Correct results can be obtained by configuring each partition with its own `alter table` command in either Hive or Impala. For example, the following:

```
ALTER TABLE page_view ADD PARTITION (dt='2008-08-08', country='us') location
'/path/to/us/part080808' PARTITION
(dt='2008-08-09', country='us') location '/path/to/us/part080809';
```

should be replaced with:

```
ALTER TABLE page_view ADD PARTITION (dt='2008-08-08', country='us') location
'/path/to/us/part080808';
ALTER TABLE page_view ADD PARTITION (dt='2008-08-09', country='us') location
'/path/to/us/part080809';
```

— PostgreSQL 9.0+ requires additional configuration

The Hive metastore will not start if you use a version of PostgreSQL later than 9.0 in the default configuration. You will see output similar to this in the log:

```
Caused by: javax.jdo.JDODataStoreException: Error executing JDOQL query
"SELECT "THIS"."TBL_NAME" AS NUCORDER0 FROM "TBLS" "THIS" LEFT OUTER JOIN "DBS"
"THIS_DATABASE_NAME" ON "THIS"."DB_ID" = "THIS_DATABASE_NAME"."DB_ID"
WHERE "THIS_DATABASE_NAME"."NAME" = ? AND (LOWER("THIS"."TBL_NAME") LIKE ? ESCAPE '\\\
) ORDER BY NUCORDER0 " : ERROR: invalid escape string
Hint: Escape string must be empty or one character..
NestedThrowables:
org.postgresql.util.PSQLException: ERROR: invalid escape string
Hint: Escape string must be empty or one character.
at
org.datanucleus.jdo.NucleusJDOHelper.getJDOExceptionForNucleusException(NucleusJDOHelper.java:313)

at org.datanucleus.jdo.JDOQuery.execute(JDOQuery.java:252)
at org.apache.hadoop.hive.metastore.ObjectStore.getTables(ObjectStore.java:759)
... 28 more
Caused by: org.postgresql.util.PSQLException: ERROR: invalid escape string
Hint: Escape string must be empty or one character.
at
org.postgresql.core.v3.QueryExecutorImpl.receiveErrorResponse(QueryExecutorImpl.java:2096)

at org.postgresql.core.v3.QueryExecutorImpl.processResults(QueryExecutorImpl.java:1829)

at org.postgresql.core.v3.QueryExecutorImpl.execute(QueryExecutorImpl.java:257)
at org.postgresql.jdbc2.AbstractJdbc2Statement.execute(AbstractJdbc2Statement.java:510)

at
org.postgresql.jdbc2.AbstractJdbc2Statement.executeWithFlags(AbstractJdbc2Statement.java:386)

at
org.postgresql.jdbc2.AbstractJdbc2Statement.executeQuery(AbstractJdbc2Statement.java:271)

at
org.apache.commons.dbcp.DelegatingPreparedStatement.executeQuery(DelegatingPreparedStatement.java:96)

at
org.apache.commons.dbcp.DelegatingPreparedStatement.executeQuery(DelegatingPreparedStatement.java:96)

at
org.datanucleus.store.rdbms.SQLController.executeStatementQuery(SQLController.java:457)
at org.datanucleus.store.rdbms.query.legacy.SQLEvaluator.evaluate(SQLEvaluator.java:123)

at
org.datanucleus.store.rdbms.query.legacy.JDOQLQuery.performExecute(JDOQLQuery.java:288)

at org.datanucleus.store.query.Query.executeQuery(Query.java:1657)
at org.datanucleus.store.rdbms.query.legacy.JDOQLQuery.executeQuery(JDOQLQuery.java:245)

at org.datanucleus.store.query.Query.executeWithArray(Query.java:1499)
at org.datanucleus.jdo.JDOQuery.execute(JDOQuery.java:243)
... 29 more
```

The problem is caused by a backward-incompatible change in the default value of the `standard_conforming_strings` property. Versions up to PostgreSQL 9.0 defaulted to `off`, but starting with version 9.0 the default is `on`.

Bug: None

Severity: Low

Resolution: Use workaround.

Workaround: As the administrator user, use the following command to turn `standard_conforming_strings` off:

```
ALTER DATABASE <hive_db_name> SET standard_conforming_strings = off;
```

— Queries spawned from MapReduce jobs in MRv1 fail if `mapreduce.framework.name` is set to `yarn`

Queries spawned from MapReduce jobs fail in MRv1 with a null pointer exception (NPE) if `/etc/mapred/conf/mapred-site.xml` has the following:

```
<property>
  <name>mapreduce.framework.name</name>
  <value>yarn</value>
</property>
```

Bug: None

Severity: High

Resolution: Use workaround

Workaround: Remove the `mapreduce.framework.name` property from `mapred-site.xml`.

— Commands run against an Oracle backed Metastore may fail

Commands run against an Oracle-backed Metastore fail with error:

```
javax.jdo.JDODataStoreException Incompatible data type for column TBLS.VIEW_EXPANDED_TEXT
: was CLOB (datastore),
but type expected was LONGVARCHAR (metadata). Please check that the type in the datastore
and the type specified in the MetaData are consistent.
```

This error may occur if the metastore is run on top of an Oracle database with the configuration property `datanucleus.validateColumns` set to `true`.

Bug: None

Severity: Low

Workaround: Set `datanucleus.validateColumns=false` in the `hive-site.xml` configuration file.

— Hive, Pig, and Sqoop 1 fail in MRv1 tarball installation because `/usr/bin/hbase` sets `HADOOP_MAPRED_HOME` to MR2

This problem affects tarball installations only.

Bug: None

Severity: High

Resolution: Use workaround

Workaround: If you are using MRv1, edit the following line in `/etc/default/hadoop` from

```
export HADOOP_MAPRED_HOME=/usr/lib/hadoop-mapreduce
```

to

```
export HADOOP_MAPRED_HOME=/usr/lib/hadoop-0.20-mapreduce
```

In addition, `/usr/lib/hadoop-mapreduce` must not exist in `HADOOP_CLASSPATH`.

— Hive Web Interface not supported

Cloudera no longer supports the Hive Web Interface because of inconsistent upstream maintenance of this project.

Bug: [DISTRO-77](#)

Severity: Low

Resolution: Use workaround

Workaround: Use Hue and Beeswax instead of the Hive Web Interface.

— Hive may need additional configuration to make it work in an Federated HDFS cluster

Hive jobs normally move data from a temporary directory to a warehouse directory during execution. Hive uses `/tmp` as its temporary directory by default, and users usually configure `/user/hive/warehouse/` as the warehouse directory. Under Federated HDFS, `/tmp` and `/user` are configured as ViewFS mount tables, and so the Hive job will actually try to move data between two ViewFS mount tables. Federated HDFS does not support this, and the job will fail with the following error:

```
Failed with exception Renames across Mount points not supported
```

Bug: None

Severity: Low

Resolution: No software fix planned; use the workaround.

Workaround: Modify `/etc/hive/conf/hive-site.xml` to allow the temporary directory and warehouse directory to use the same ViewFS mount table. For example, if the warehouse directory is `/user/hive/warehouse`, add the following property to `/etc/hive/conf/hive-site.xml` so both directories use the ViewFS mount table for `/user`.

```
<property>
  <name>hive.exec.scratchdir</name>
  <value>/user/${user.name}/tmp</value>
</property>
```

— Cannot create archive partitions with external HAR (HTTP Archive) tables

`ALTER TABLE ... ARCHIVE PARTITION` is not supported on external tables.

Bug: None

Severity: Low

Workaround: None

— Setting `hive.optimize.skewjoin` to true causes long running queries to fail

Bug: None

Severity: Low

Workaround: None

— JDBC - `executeUpdate` does not return the number of rows modified

Contrary to the documentation, method `executeUpdate` always returns zero.

Severity: Low

Workaround: None

—Hive Auth (Grant/Revoke/Show Grant) statements do not support fully qualified table names (`default.tab1`)

Bug: None

Severity: Low

Workaround: Switch to the database before granting privileges on the table.

—Object types `Server` and `URI` are not supported in `"SHOW GRANT ROLE roleName on OBJECT objectName"`

Bug: None

Severity: Low

Workaround: Use `SHOW GRANT ROLE roleName` to list all privileges granted to the role.

HCatalog Known Issues

- **Note:** As of CDH 5, HCatalog is part of Apache Hive.

—Hive's `DECIMAL` data type cannot be mapped to Pig via HCatalog

HCatalog does recognize the `DECIMAL` data type.

Bug: none

Severity: Low

Workaround: None

—Job submission using WebHCatalog might not work correctly

Bug: none

Severity: Low

Resolution: Use workaround.

Workaround: Cloudera recommends using the Oozie REST interface to submit jobs, as it's a more mature and capable tool.

—WebHCatalog does not work in a Kerberos-secured Federated cluster

Bug: none

Severity: Low

Resolution: None planned.

Workaround:None

Hue Known Issues

—Installing Hue on Debian/Ubuntu may require manual restart of service

When you install or upgrade Hue from packages, the system tries to restart the Hue service in between updating the apps. This causes Hue to start with an interface that could be missing some apps.

Bug: None

Severity: Low

Resolution: Use workaround.

Workaround: Use the following command to restart Hue once installation/upgrade is complete.

```
$ sudo service hue restart
```

— Sqoop Application is not supported

When you open up the Sqoop Editor, the page shows an error. The Sqoop 2 service is currently under heavy development and the Hue App has not yet been updated to reflect all of the changes.

Bug: None

Severity: Medium

Workaround: None.

— Hue does not support the Spark App

Hue does not currently support the Spark App.

— Migrations to MySQL fail if multiple Hue users have the same name but different upper/lower case letters

Bug: None

Severity: Medium

Workaround: None.

Hue hangs or fails because SQLite database is overloaded, returning "database is locked"

Hue can hang or fail because the SQLite database is overloaded, returning `database is locked`.

Bug: None

Severity: Medium

Workaround: Do one of the following:

- Increase the timeout setting in `[desktop][[database]]` in the Hue configuration file, *OR*
- Use a different database.

— Importing Hue data to MySQL can cause columns to be truncated on import

Importing Hue data to MySQL can cause columns to be truncated on import, displaying `Warning: Data truncated for column 'name' at row 1`

Bug: None

Severity: Medium

Workaround: In the `/etc/my.cnf` file, configure the database operation to fail rather than truncate data:

```
[mysqld]
sql_mode=STRICT_ALL_TABLES
```

Cloudera Impala Known Issues

The following sections describe known issues and workarounds in Impala.

For issues fixed in various Impala releases, see [Cloudera Impala Fixed Issues](#) on page 124.

Known Issues in the Current Production Release (Impala 2.1.x)

These known issues affect the current release. Any workarounds are listed here. The bug links take you to the Impala issues site, where you can see the diagnosis and whether a fix is in the pipeline.

Impala requires Parquet column metadata in same order as the schema definition

Impala could read columns in incorrect order from Parquet files created by other components. Some files created using external Parquet libraries could contain column metadata written in a different order than the actual columns within the file.

Severity: High

Resolution: The Parquet libraries used by other components, and the Parquet spec itself, are being updated to match Impala behavior as part of the issue [PARQUET-188](#).

invalid tuple_idx when combining INSERT INTO with analytic subquery

An `INSERT ... SELECT` statement could encounter an error if the `SELECT` portion included an analytic function call.

Bug: [IMPALA-1737](#)

Severity: Major

Workaround: Rewrite the statement as a `CREATE TABLE AS SELECT` statement.

CPU requirement for SSE4.1

Currently, Impala 2.0.x and 2.1.x do not function on CPUs without the SSE4.1 instruction set. This minimum CPU requirement is higher than in previous versions, which relied on the older SSSE3 instruction set. Check the CPU level of the hosts in your cluster before upgrading to Impala 2.0.x or 2.1.x, or CDH 5.2.x or CDH 5.3.x.

impala processes should cleanup their own old log files

Because Impala log files are not automatically deleted, you could potentially encounter disk space issues due to log file growth.

Bug: [IMPALA-377](#)

Severity: High

Workaround: Set up manual log rotation using your Linux tool or technique of choice. See [Rotating Impala Logs](#) for details.

Can't update stats manually via alter table after upgrading to CDH 5.2

Bug: [IMPALA-1420](#)

Severity: High

Workaround: On CDH 5.2, when adjusting table statistics manually by setting the `numRows`, you must also enable the Boolean property `STATS_GENERATED_VIA_STATS_TASK`. For example, use a statement like the following to set both properties with a single `ALTER TABLE` statement:

```
ALTER TABLE table_name SET TBLPROPERTIES('numRows'='new_value',
'STATS_GENERATED_VIA_STATS_TASK' = 'true');
```

Resolution: The underlying cause is the issue [HIVE-8648](#) that affects the metastore in Hive 0.13. The workaround is only needed until the fix for this issue is incorporated into a CDH release.

Memory leak using zlib on CentOS6 (and possibly other platforms)

Unreleased memory could accumulate as more and more queries are run. The cause is thought to be a bug in version 1.2.3 of the `zlib` library, which is used in CentOS 6.4 and possibly other Linux releases. Impala uses this library internally to compress query profiles.

Bug: [IMPALA-1194](#)

Severity: High

Resolution: Under investigation

Memory Limit Exceeded Error when running with multiple clients

Out-of-memory errors could occur if multiple concurrent queries utilize the “spill to disk” mechanism because of memory pressure from other queries.

Bug: [IMPALA-1385](#)

Severity: High

Workaround: Either run such queries concurrently using a mechanism such as admission control, or reduce the memory limit for each query so that the spilling operation is triggered sooner. For example, if two queries are encountering this issue when running with `MEM_LIMIT=4g`, reduce the memory limit for each query by half, to 2 GB.

ORDER BY rand() does not work.

Because the value for `rand()` is computed early in a query, using an `ORDER BY` expression involving a call to `rand()` does not actually randomize the results.

Bug: [IMPALA-397](#)

Severity: High

Impala BE cannot parse Avro schema that contains a trailing semi-colon

If an Avro table has a schema definition with a trailing semicolon, Impala encounters an error when the table is queried.

Bug: [IMPALA-1024](#)

Severity: High

Process mem limit does not account for the JVM's memory usage

Some memory allocated by the JVM used internally by Impala is not counted against the memory limit for the `impalad` daemon.

Bug: [IMPALA-691](#)

Severity: High

Workaround: To monitor overall memory usage, use the `top` command, or add the memory figures in the Impala web UI `/memz` tab to JVM memory usage shown on the `/metrics` tab.

Impala Parser issue when using fully qualified table names that start with a number.

A fully qualified table name starting with a number could cause a parsing error. In a name such as `db.571_market`, the decimal point followed by digits is interpreted as a floating-point number.

Bug: [IMPALA-941](#)

Severity: High

Workaround: Surround each part of the fully qualified name with backticks (```).

CatalogServer should not require HBase to be up to reload its metadata

If HBase is unavailable during Impala startup or after an `INVALIDATE METADATA` statement, the `catalogd` daemon could go into an error loop, making Impala unresponsive.

Bug: [IMPALA-788](#)

Severity: High

Workaround: For systems not managed by Cloudera Manager, add the following settings to `/etc/impala/conf/hbase-site.xml`:

```
<property>
  <name>hbase.client.retries.number</name>
  <value>3</value>
</property>
<property>
  <name>hbase.rpc.timeout</name>
  <value>3000</value>
</property>
```


Currently, Cloudera Manager does not have an Impala-only override for HBase settings, so any HBase configuration change you make through Cloudera Manager would take affect for all HBase applications. Therefore, this change is not recommended on systems managed by Cloudera Manager.

Kerberos tickets must be renewable

In a Kerberos environment, the `impalad` daemon might not start if Kerberos tickets are not renewable.

Workaround: Configure your KDC to allow tickets to be renewed, and configure `krb5.conf` to request renewable tickets.

Avro Scanner fails to parse some schemas

Querying certain Avro tables could cause a crash or return no rows, even though Impala could `DESCRIBE` the table.

Bug: [IMPALA-635](#)

Severity: High

Workaround: Swap the order of the fields in the schema specification. For example, `["null", "string"]` instead of `["string", "null"]`.

Resolution: Not allowing this syntax agrees with the Avro specification, so it may still cause an error even when the crashing issue is resolved.

Configuration needed for Flume to be compatible with Impala

For compatibility with Impala, the value for the Flume HDFS Sink `hdfs.writeFormat` must be set to `Text`, rather than its default value of `Writable`. The `hdfs.writeFormat` setting must be changed to `Text` before creating data files with Flume; otherwise, those files cannot be read by either Impala or Hive.

Severity: High

Resolution: This information has been requested to be added to the upstream Flume documentation.

Impala does not support running on clusters with federated namespaces

Impala does not support running on clusters with federated namespaces. The `impalad` process will not start on a node running such a filesystem based on the `org.apache.hadoop.fs.viewfs.ViewFs` class.

Bug: [IMPALA-77](#)

Severity: Undetermined

Anticipated Resolution: Limitation

Workaround: Use standard HDFS on all Impala nodes.

Deviation from Hive behavior: Out of range values float/double values are returned as maximum allowed value of type (Hive returns NULL)

Impala behavior differs from Hive with respect to out of range float/double values. Out of range values are returned as maximum allowed value of type (Hive returns `NULL`).

Severity: Low

Workaround: None

Deviation from Hive behavior: Impala does not do implicit casts between string and numeric and boolean types.

Severity: Low

Anticipated Resolution: None

Workaround: Use explicit casts.

If Hue and Impala are installed on the same host, and if you configure Hue Beeswax in CDH 4.1 to execute Impala queries, Beeswax cannot list Hive tables and shows an error on Beeswax startup.

Hue requires Beeswaxd to be running in order to list the Hive tables. Because of a port conflict bug in Hue in CDH4.1, when Hue and Impala are installed on the same host, an error page is displayed when you start the Beeswax application, and when you open the **Tables** page in Beeswax.

Severity: High

Anticipated Resolution: Fixed in an upcoming CDH4 release

Workarounds: Choose one of the following workarounds (but only one):

- Install Hue and Impala on different hosts. *OR*
- Upgrade to CDH4.1.2 and add the following property in the `beeswax` section of the `/etc/hue/hue.ini` configuration file:

```
beeswax_meta_server_only=9004
```

OR

- If you are using CDH4.1.1 and you want to install Hue and Impala on the same host, change the code in this file:

```
/usr/share/hue/apps/beeswax/src/beeswax/management/commands/beeswax_server.py
```

Replace line 66:

```
str(beeswax.conf.BEESWAX_SERVER_PORT.get()),
```

With this line:

```
'8004',
```

Beeswaxd will then use port 8004.

■ **Note:**

If you used Cloudera Manager to install Impala, refer to the Cloudera Manager release notes for information about using an equivalent workaround by specifying the `beeswax_meta_server_only=9004` configuration value in the options field for Hue. In Cloudera Manager 4, these fields are labelled **Safety Valve**; in Cloudera Manager 5, they are called **Advanced Configuration Snippet**.

Impala should tolerate bad locale settings

If the `LC_*` environment variables specify an unsupported locale, Impala does not start.

Bug: [IMPALA-532](#)

Severity: Low

Workaround: Add `LC_ALL="C"` to the environment settings for both the Impala daemon and the Statestore daemon. See [Modifying Impala Startup Options](#) for details about modifying these environment settings.

Resolution: Fixing this issue would require an upgrade to Boost 1.47 in the Impala distribution.

Log Level 3 Not Recommended for Impala

The extensive logging produced by log level 3 can cause serious performance overhead and capacity issues.

Severity: Low

Workaround: Reduce the log level to its default value of 1, that is, `GLOG_v=1`. See [Setting Logging Levels](#) for details about the effects of setting different logging levels.

Apache Oozie Known Issues

— [Oozie can't submit jobs to a secure MRv2 cluster if the Job History Server is down](#)

After trying to submit the job (by default, three times), Oozie will `SUSPEND` the workflow automatically.

Severity: Low

Workaround: When you bring up the Job History Server, use the `resume` command to tell Oozie to continue the workflow from where it left off.

— [An Oozie server works either with a Hadoop MRv1 cluster or a Hadoop YARN cluster, not both](#)

Severity: Low

Anticipated Resolution: None planned

Workaround: Use two different Oozie servers

[When you use Hive Server 2 from Oozie, Oozie won't collect or print out the Hadoop Job IDs of any jobs launched by Hive Server 2](#)

Bug: None

Severity: Low

Workaround: You can get the Hadoop IDs from the Resource Manager or JobTracker.

Apache Parquet (incubating) Known Issues

— [Parquet file writes run out of memory if \(number of partitions\) times \(block size\) exceeds available memory](#)

The Parquet output writer allocates one block for each table partition it is processing and writes partitions in parallel. The MapReduce or YARN task will run out of memory if (number of partitions) times (Parquet block size) is greater than the available memory.

Bug: None

Severity: Medium

Workaround: None; if necessary, reduce the number of partitions in the table.

— [parquet-thrift cannot read Parquet data written by Hive](#)

`parquet-thrift` cannot read Parquet data written by Hive, and `parquet-avro` will show an additional record level in lists named `array_element`.

Bug: [PARQUET-113](#)

Severity: Medium

Workaround: None; arrays written by `parquet-avro` or `parquet-thrift` cannot currently be read by `parquet-hive`.

Apache Pig Known Issues

— [Hive, Pig, and Sqoop 1 fail in MRv1 tarball installation because /usr/bin/hbase sets HADOOP_MAPRED_HOME to MRv2](#)

This problem affects tarball installations only.

Bug: None

Severity: High

Resolution: Use workaround.

Workaround: If you are using MRv1, edit the following line in `/etc/default/hadoop` from

```
export HADOOP_MAPRED_HOME=/usr/lib/hadoop-mapreduce
```

to

```
export HADOOP_MAPRED_HOME=/usr/lib/hadoop-0.20-mapreduce
```

In addition, `/usr/lib/hadoop-mapreduce` must not exist in `HADOOP_CLASSPATH`.

[Pig fails to read Parquet file \(created with Hive\) with a complex field if schema not specified explicitly](#)

Bug: None

Severity: Low

Workaround: Provide the schema of the fields in the `LOAD` statement.

Cloudera Search Known Issues

The current release includes the following known limitations:

— [Solr, Oozie and HttpFS fail when KMS and SSL are enabled using self-signed certificates](#)

When the KMS service is added and SSL is enabled, Solr, Oozie and HttpFS are not automatically configured to trust the KMS's self-signed certificate and you might see the following error.

```
org.apache.oozie.service.AuthorizationException: E0501: Could not perform authorization operation,
sun.security.validator.ValidatorException: PKIX path building failed:
sun.security.provider.certpath.SunCertPathBuilderException:
unable to find valid certification path to requested target
```

Severity: Medium

Workaround: You must explicitly load the relevant truststore with the KMS certificate to allow these services to communicate with the KMS.

Solr, Oozie: Add the following arguments to their environment safety valve so as to load the truststore with the required KMS certificate.

```
CATALINA_OPTS="-Djavax.net.ssl.trustStore=/etc/path-to-truststore.jks
-Djavax.net.ssl.trustStorePassword=<password>"
```

HttpFS: Add the following arguments to the **Java Configuration Options for HttpFS** property.

```
-Djavax.net.ssl.trustStore=/etc/path-to-truststore.jks
-Djavax.net.ssl.trustStorePassword=<password>
```

— [CrunchIndexerTool which includes Spark indexer requires specific input file format specifications](#)

If the `--input-file-format` option is specified with `CrunchIndexerTool` then its argument must be `text`, `avro`, or `avroParquet`, rather than a fully qualified class name.

Severity: Low

— [Previously deleted empty shards may reappear after restarting the leader host](#)

It is possible to be in the process of deleting a collection when hosts are shut down. In such a case, when hosts are restarted, some shards from the deleted collection may still exist, but be empty.

Severity: Low

Workaround: To delete these empty shards, manually delete the folder matching the shard. On the hosts on which the shards exist, remove folders under `/var/lib/solr/` that match the collection and shard. For example, if you had an empty shard 1 and empty shard 2 in a collection called `MyCollection`, you might delete all folders matching `/var/lib/solr/MyCollection{1,2}_replica*/`.

— The `quickstart.sh` file does not validate ZooKeeper and the NameNode on some operating systems

The `quickstart.sh` file uses the `timeout` function to determine if ZooKeeper and the NameNode are available. To ensure this check can be complete as intended, the `quickstart.sh` determines if the operating system on which the script is running supports `timeout`. If the script detects that the operating system does not support `timeout`, the script continues without checking if the NameNode and ZooKeeper are available. If your environment is configured properly or you are using an operating system that supports `timeout`, this issue does not apply.

Severity: Low

Workaround: This issue only occurs in some operating systems. If `timeout` is not available, a warning is displayed, but the `quickstart` continues and final validation is always done by the MapReduce jobs and Solr commands that are run by the `quickstart`.

— Field value class guessing and Automatic schema field addition are not supported with the `MapReduceIndexerTool` nor the `HBaseMapReduceIndexerTool`

The `MapReduceIndexerTool` and the `HBaseMapReduceIndexerTool` can be used with a Managed Schema created via NRT indexing of documents or via the Solr Schema API. However, neither tool supports adding fields automatically to the schema during ingest.

Severity: Medium

Workaround: Define the schema before running the `MapReduceIndexerTool` or `HBaseMapReduceIndexerTool`. In non-schemaless mode, define in the schema using the `schema.xml` file. In schemaless mode, either define the schema using the Solr Schema API or index sample documents using NRT indexing before invoking the tools. In either case, Cloudera recommends that you verify that the schema is what you expect using the `List Fields` API command.

— The “Browse” and “Spell” Request Handlers are not enabled in schemaless mode

The “Browse” and “Spell” Request Handlers require certain fields be present in the schema. Since those fields cannot be guaranteed to exist in a Schemaless setup, the “Browse” and “Spell” Request Handlers are not enabled by default.

Severity: Low

Workaround: If you require the “Browse” and “Spell” Request Handlers, add them to the `solrconfig.xml` configuration file. Generate a non-schemaless configuration to see the usual settings and modify the required fields to fit your schema.

— Using Solr with Sentry may consume more memory than required

The sentry-enabled `solrconfig.xml.secure` configuration file does not enable the `hdfs` global block cache. This does not cause correctness issues, but it can greatly increase the amount of memory that Solr requires.

Severity: Medium

Workaround: Enable the `hdfs` global block cache, by adding the following line to `solrconfig.xml.secure` under the `directoryFactory` element:

```
<str name="solr.hdfs.blockcache.global">${solr.hdfs.blockcache.global: true}</str>
```

— Enabling blockcache writing may result in unusable indexes

It is possible to create indexes with `solr.hdfs.blockcache.write.enabled` set to `true`. Such indexes may appear corrupt to readers, and reading these indexes may irrecoverably corrupt indexes. Blockcache writing is disabled by default.

Severity: Medium

Workaround: Do not enable blockcache writing.

— Solr fails to start when Trusted Realms are added for Solr into Cloudera Manager

Cloudera Manager generates name rules with spaces as a result of entries in the Trusted Realms, which do not work with Solr. This causes Solr to not start.

Severity: Medium

Workaround: Do not use the Trusted Realm field for Solr in Cloudera Manager. To write your own name rule mapping, add an environment variable `SOLR_AUTHENTICATION_KERBEROS_NAME_RULES` with the mapping. See the [Cloudera Manager Security Guide](#) for more information.

— Lily HBase batch indexer jobs fail to launch

A symptom of this issue is an exception similar to the following:

```
Exception in thread "main" java.lang.IllegalAccessException: class
com.google.protobuf.ZeroCopyLiteralByteString cannot access its superclass
com.google.protobuf.LiteralByteString
    at java.lang.ClassLoader.defineClass1(Native Method)
    at java.lang.ClassLoader.defineClass(ClassLoader.java:792)
    at java.security.SecureClassLoader.defineClass(SecureClassLoader.java:142)
    at java.net.URLClassLoader.defineClass(URLClassLoader.java:449)
    at java.net.URLClassLoader.access$100(URLClassLoader.java:71)
    at java.net.URLClassLoader$1.run(URLClassLoader.java:361)
    at java.net.URLClassLoader$1.run(URLClassLoader.java:355)
    at java.security.AccessController.doPrivileged(Native Method)
    at java.net.URLClassLoader.findClass(URLClassLoader.java:354)
    at java.lang.ClassLoader.loadClass(ClassLoader.java:424)
    at java.lang.ClassLoader.loadClass(ClassLoader.java:357)
    at org.apache.hadoop.hbase.protobuf.ProtobufUtil.toScan(ProtobufUtil.java:818)
    at
org.apache.hadoop.hbase.mapreduce.TableMapReduceUtil.convertScanToString(TableMapReduceUtil.java:433)
    at
org.apache.hadoop.hbase.mapreduce.TableMapReduceUtil.initTableMapperJob(TableMapReduceUtil.java:186)
    at
org.apache.hadoop.hbase.mapreduce.TableMapReduceUtil.initTableMapperJob(TableMapReduceUtil.java:147)
    at
org.apache.hadoop.hbase.mapreduce.TableMapReduceUtil.initTableMapperJob(TableMapReduceUtil.java:270)
    at
org.apache.hadoop.hbase.mapreduce.TableMapReduceUtil.initTableMapperJob(TableMapReduceUtil.java:100)
    at
com.ngdata.hbaseindexer.mr.HBaseMapReduceIndexerTool.run(HBaseMapReduceIndexerTool.java:124)
    at
com.ngdata.hbaseindexer.mr.HBaseMapReduceIndexerTool.run(HBaseMapReduceIndexerTool.java:64)
    at org.apache.hadoop.util.ToolRunner.run(ToolRunner.java:70)
    at
com.ngdata.hbaseindexer.mr.HBaseMapReduceIndexerTool.main(HBaseMapReduceIndexerTool.java:51)
    at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
    at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:57)
    at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
```

```
at java.lang.reflect.Method.invoke(Method.java:606)
at org.apache.hadoop.util.RunJar.main(RunJar.java:212)
```

This is because of an optimization introduced in [HBASE-9867](#) that inadvertently introduced a classloader dependency. In order to satisfy the new classloader requirements, `hbase-protocol.jar` must be included in Hadoop's classpath. This can be resolved on a per-job launch basis by including it in the `HADOOP_CLASSPATH` environment variable when you submit the job.

Severity: High

Workaround: Run the following command before issuing Lily HBase MapReduce jobs. Replace the `.jar` file names and filepaths as appropriate.

```
$ export HADOOP_CLASSPATH=/path/to/hbase-protocol.jar; hadoop jar <MyJob>.jar
<MyJobMainClass>
```

— [Users may receive limited error messages on requests in Sentry-protected environment.](#)

Users submit requests which are received by a host. The host that receives the request may be different from the host with the relevant information. In such a case, Solr forwards the request to the appropriate host. Once the correct host receives the request, Sentry may deny access.

Because the request was forwarded, available information may be limited. In such a case, the user's client display the error message `Server returned HTTP response code: 401 for URL:` followed by the Solr machine reporting the error.

Severity: Low

Workaround: For complete error information, review the contents of the Solr logs on the machine reporting the error.

— [Users with insufficient Solr permissions may receive a "Page Loading" message from the Solr Web Admin UI](#)

Users who are not authorized to use the Solr Admin UI are not given page explaining that access is denied, and instead receive a web page that never finishes loading.

Severity: Low

Workaround: None

— [Mapper-only HBase batch indexer fails if configured to use security.](#)

Attempts to complete an HBase batch indexing jobs fail when Kerberos authentication is enabled and reducers is set to 0.

Workaround: Either disable Kerberos authentication or use one or more reducers.

— [Spark indexer fails if configured to use security.](#)

Spark indexing jobs fail when Kerberos authentication is enabled.

Workaround: Disable Kerberos authentication or use another indexer.

— [Using MapReduceIndexerTool or HBaseMapReduceIndexerTool multiple times may produce duplicate entries in a collection.](#)

Repeatedly running the `MapReduceIndexerTool` on the same set of input files can result in duplicate entries in the Solr collection. This occurs because the tool can only insert documents and cannot update or delete existing Solr documents.

Severity: Medium

Workaround: To avoid this issue, use HBaseMapReduceIndexerTool with zero reducers. This must be done without Kerberos.

— Deleting collections may fail if hosts are unavailable.

It is possible to delete a collection when hosts that host some of the collection are unavailable. After such a deletion, if the previously unavailable hosts are brought back online, the deleted collection may be restored.

Severity: Low

Workaround: Ensure all hosts are online before deleting collections.

— Lily HBase Indexer is slow to index new data after restart.

After restarting the Lily HBase Indexer, you can add data to one of the HBase tables. There may be a delay of a few minutes before this newly added data appears in Solr. This delay only occurs with a first HBase addition after a restart. Similar subsequent additions are not subject to this delay.

Severity: Low

Workaround: None

— Some configurations for Lily HBase Indexers cannot be modified after initial creation.

Newly created Lily HBase Indexers define their configuration using the properties in `/etc/hbase-solr/conf/hbase-indexer-site.xml`. Therefore, if the properties in the `hbase-indexer-site.xml` file are incorrectly defined, new indexers do not work properly. Even after correcting the contents of `hbase-indexer-site.xml` and restarting the indexer service, old, incorrect content persists. This continues to create non-functioning indexers.

Severity: Medium

Workaround:

- **Warning:** This workaround involves completing destructive operations that delete all of your other Lily HBase Indexers.

To resolve this issue:

1. Connect to each machine running the Lily HBase Indexer service using the NGdata and stop the indexer:

```
service hbase-solr-indexer stop
```

- **Note:** You may need to stop the service on multiple machines.

2. For each indexer machine, modify the `/etc/hbase-solr/conf/hbase-indexer-site.xml` file to include valid settings.
3. Connect to the ZooKeeper machine, invoke the ZooKeeper CLI, and remove all contents of the `/ngdata` chroot:

```
$ /usr/lib/zookeeper/bin/zkCli.sh  
[zk: localhost:2181( CONNECTED) 0] rmr /ngdata
```

4. Connect to each indexer machine and restart the indexer service.

```
service hbase-solr-indexer start
```

After restarting the client services, ZooKeeper is updated with the correct information stored on the updated clients.

— Saving search results is not supported in this release.

This version of Cloudera Search does not support the ability to save search results.

Severity: Low

Workaround: None

— HDFS Federation is not supported in this release.

This version of Cloudera Search does not support HDFS Federation.

Severity: Low

Workaround: None

— Block Cache Metrics are not supported in this release.

This version of Cloudera Search does not support block cache metrics.

Severity: Low

Workaround: None

— Shard splitting support is experimental.

Shard splitting was added with the recent release of Solr 4.4. Cloudera anticipates shard splitting to function as expected with Cloudera Search, but this interaction has not been thoroughly tested. Therefore, Cloudera cannot guarantee issues will not arise when shard splitting is used with Search.

Severity: Low

Workaround: Use shard splitting for test and development purposes, but be aware of the risks of using shard splitting in production environments. To avoid using shard splitting, use the source data to create a new index with a new sharding count by re-indexing the data to a new collection. You can enable this using the MapReduceIndexerTool.

— User with `update` access to the administrative collection can elevate the access.

Users are granted access to collections. Access to several collections can be simplified by aliasing a set of collections. Creating an alias requires `update` access to the administrative collection. Any user with `update` access to the administrative collection is granted `query` access to all collections in the resulting alias. This is true even if the user with `update` access to the administrative collection otherwise would be unable to `query` the other collections that have been aliased.

Severity: Medium

Workaround: None. Mitigate the risk by limiting the users with `update` access to the administrative collection.

Apache Sentry (incubating) Known Issues

— `INSERT OVERWRITE LOCAL` fails if you use only the Linux pathname

Bug: None

Severity: Low

Workaround: Prefix the path of the local file with `file://` when using `INSERT OVERWRITE LOCAL`.

— `INSERT OVERWRITE` and `CREATE EXTERNAL` commands fail because of HDFS URI permissions

When you use Sentry to secure Hive, and use HDFS URIs in a HiveQL statement, the query will fail with an HDFS permissions error unless you specify the NameNode and port.

Bug: None

Severity: Low

Workaround: Specify the NameNode and port, where applicable, in the URI; for example specify `hdfs://nn-uri:port/user/warehouse/hive/tab` rather than simply `/user/warehouse/hive/tab`. In a high-availability deployment, specify the value of `FS.defaultFS`.

—Hive Auth (Grant/Revoke/Show Grant) statements do not support fully qualified table names (`default.tab1`)

Bug: None

Severity: Low

Workaround: Switch to the database before granting privileges on the table.

—Object types `Server` and `URI` are not supported in `SHOW GRANT ROLE roleName on OBJECT objectName`

Bug: None

Severity: Low

Workaround: Use `SHOW GRANT ROLE roleName` to list all privileges granted to the role.

Apache Spark Known Issues

—Spark SQL, MLlib, and GraphX not currently supported

Cloudera does not currently offer commercial support for Spark SQL, MLlib, or GraphX because of their immaturity.

—Spark Streaming recovery not currently supported

Cloudera does not currently support Spark Streaming recovery (fault-tolerant processing) because of its immaturity, but bundles it with CDH so that you can try it out.

—Spark standalone mode does not work on secure clusters

Bug: None

Severity: Low

Resolution: Use workaround.

Workaround: On secure clusters, run Spark applications on YARN.

—Spark's sort-based shuffle is affected by a kernel bug

Spark's sort-based shuffle is affected by a kernel bug (<http://git.kernel.org/cgit/linux/kernel/git/torvalds/linux.git/commit/?id=2cb4b05e7647891b46b91c07c9a60304803d1688>). The kernel bug was fixed in RHEL/CentOS 6.2.

■ **Note:**

CDH defaults to hash-based shuffle.

Bug: [SPARK-3948](#)

—Spark not automatically picking up `hive-site.xml`

When you run Spark SQL on Yarn, the client `hive-site.xml` does not get picked up automatically by `spark-submit`.

Bug: [SPARK-2669](#)

Severity: Low

Workaround: Do one of the following, depending on which mode you are running in:

- If you are running in `yarn-client` mode, set `HADOOP_CONF_DIR` to `/etc/hive/conf/` (or the directory where your `hive-site.xml` is located).
- If you are running in `yarn-cluster` mode, the easiest thing to do is to add `--files=/etc/hive/conf/hive-site.xml` (or the path for your `hive-site.xml`) to your `spark-submit` script.

— Spark does not support Scala 2.11

CDH does not currently support Spark on Scala 2.11 because it is binary incompatible, and also not yet full-featured.

Apache Sqoop Known Issues

— MySQL JDBC driver shipped with CentOS 6 systems does not work with Sqoop

CentOS 6 systems currently ship with version 5.1.17 of the MySQL JDBC driver. This version does not work correctly with Sqoop.

Bug: None

Severity: Medium

Resolution: Use workaround.

Workaround: Install version 5.1.31 of the JDBC driver, following directions in (Sqoop 1) or (Sqoop 2).

Sqoop 1

— *Hive, Pig, and Sqoop 1 fail in MRv1 tarball installation because /usr/bin/hbase sets HADOOP_MAPRED_HOME to MR2*

This problem affects tarball installations only.

Bug: None

Severity: High

Resolution: Use workaround.

Workaround: If you are using MRv1, edit the following line in /etc/default/hadoop from

```
export HADOOP_MAPRED_HOME=/usr/lib/hadoop-mapreduce
```

to

```
export HADOOP_MAPRED_HOME=/usr/lib/hadoop-0.20-mapreduce
```

In addition, /usr/lib/hadoop-mapreduce must not exist in HADOOP_CLASSPATH.

— Sqoop import into Hive Causes a Null Pointer Exception (NPE)

Bug: None

Severity: High

Workaround: Import the data into HDFS via Sqoop first and then import it into Hive from HDFS.

Sqoop 2

— *Sqoop 2 client cannot be used with a different version of the Sqoop 2 server*

The Sqoop 2 client and server must be running the same CDH version.

Bug: None

Severity: Low

Workaround: Make sure all Sqoop 2 components are running the same version of CDH.

— Sqoop 2 works only in a non-secure cluster

Bug: None

Severity: Low

Workaround: Deploy Sqoop 2 only in a non-secure cluster.

Apache ZooKeeper Known Issues

— The ZooKeeper server cannot be migrated from version 3.4 to 3.3, then back to 3.4, without user intervention.

Upgrading from 3.3 to 3.4 is supported, as is downgrading from 3.4 to 3.3. However, moving from 3.4 to 3.3 and back to 3.4 will fail. 3.4 is checking the `datadir` for `acceptedEpoch` and `currentEpoch` files and comparing these against the snapshot and log files contained in the same directory. These epoch files are new in 3.4.

As a result: 1) Upgrading from 3.3 to 3.4 is fine - the `*Epoch` files don't exist, and the server creates them. 2) Downgrading from 3.4 to 3.3 is also fine as version 3.3 ignores the `*Epoch` files. 3) Going from 3.4 to 3.3 then back to 3.4 fails because 3.4 sees invalid `*Epoch` files in the `datadir`; 3.3 will have ignored them, applying changes to the snapshot and log files without updating the `*Epoch` files.

Bug: [ZOOKEEPER-1149](#)

Severity: Low

Anticipated Resolution: See workaround

Workaround: Delete the `*Epoch` files if this situation occurs — the version 3.4 server will recreate them as in case 1) above.

Issues Fixed in CDH 5

- **Note:** For links to the detailed change lists that describe the bug fixes and improvements to all of the CDH 5 projects, including bug-fix reports for the corresponding upstream Apache projects, see the packaging section of [CDH Version and Packaging Information](#).

Known Issues Fixed in CDH 5.3.5

This is a maintenance release that fixes the following issues:

Potential job failures during YARN rolling upgrades to CDH 5.3.4

Problem: A MapReduce security fix introduced a compatibility issue that results in job failures during YARN rolling upgrades from CDH 5.3.3 to CDH 5.3.4.

Release affected: CDH 5.3.4

Release containing the fix: CDH 5.3.5

Workarounds: You can use any one of the following workarounds for this issue:

- Upgrade to CDH 5.3.5.
 - Restart any jobs that might have failed during the upgrade.
 - Explicitly set the version of MapReduce to be used so it is picked on a per-job basis.
1. Update the YARN property, **MR Application Classpath** (`mapreduce.application.classpath`), either in Cloudera Manager or in the `mapred-site.xml` file. Remove all existing values and add a new entry: `<parcel-path>/lib/hadoop-mapreduce/*`, where `<parcel-path>` is the absolute path to the parcel installation. For example, the default installation path for the CDH 5.3.3 parcel would be: `/opt/cloudera/parcels/CDH-5.3.3-1.cdh5.3.3.p0.5/lib/hadoop-mapreduce/*`.
 2. Wait until jobs submitted with the above client configuration change have run to completion.
 3. Upgrade to CDH 5.3.4.
 4. Update the **MR Application Classpath** (`mapreduce.application.classpath`) property to point to the new CDH 5.3.4 parcel.

Do *not* delete the old parcel until after all jobs submitted prior to the upgrade have finished running.

Upstream Issues Fixed

The following upstream issue has been fixed in CDH 5.3.5:

- [YARN-3811](#) - NodeManager restarts could lead to application failures

Known Issues Fixed in CDH 5.3.4

NameNode Incorrectly Reports Missing Blocks During Rolling Upgrade

Problem: During a rolling upgrade to CDH 5.3.3, the NameNode may report missing blocks after rolling multiple DataNodes. This is caused by a race condition with block reporting between the DataNode and the NameNode. No permanent data loss occurs, but data can be unavailable for up to six hours before the problem corrects itself.

Releases affected: CDH 5.0.6, 5.1.5, 5.2.5, 5.3.3

Releases containing the fix: 5.3.4

Workaround:

To avoid the problem: Cloudera advises skipping CDH 5.3.3, and upgrading to CDH 5.3.4 instead. Before upgrading to CDH 5.3.4, see [Install and Upgrade Known Issues](#) on page 69.

If you have already completed an upgrade to an affected release, or are installing a new cluster: In this case you can continue to run CDH 5.3.3, or upgrade to CDH 5.3.4, as you choose.

Upstream Issues Fixed

The following upstream issues are fixed in CDH 5.3.4:

- [HDFS-7980](#) - Incremental BlockReport will dramatically slow down the startup of a namenode
- [HDFS-8380](#) - Always call addStoredBlock on blocks which have been shifted from one storage to another
- [HDFS-7645](#) - Rolling upgrade is restoring blocks from trash multiple times
- [HDFS-7869](#) - Inconsistency in the return information while performing rolling upgrade
- [HDFS-7340](#) - make rollingUpgrade start/finalize idempotent
- [HDFS-7312](#) - Update DistCp v1 to optionally not use tmp location (branch-1 only)
- [HDFS-7530](#) - Allow renaming of encryption zone roots
- [HDFS-7587](#) - Edit log corruption can happen if append fails with a quota violation
- [YARN-3485](#) - FairScheduler headroom calculation doesn't consider maxResources for Fifo and FairShare policies
- [YARN-3491](#) - PublicLocalizer#addResource is too slow.
- [YARN-3021](#) - YARN's delegation-token handling disallows certain trust setups to operate properly over DistCp
- [YARN-3241](#) - FairScheduler handles "invalid" queue names inconsistently
- [YARN-3022](#) - Expose Container resource information from NodeManager for monitoring
- [YARN-2984](#) - Metrics for container's actual memory usage
- [YARN-3465](#) - Use LinkedHashMap to preserve order of resource requests
- [MAPREDUCE-6339](#) - Job history file is not flushed correctly because isTimerActive flag is not set true when flushTimerTask is scheduled.
- [MAPREDUCE-5710](#) - Backport MAPREDUCE-1305 to branch-1
- [MAPREDUCE-6238](#) - MR2 can't run local jobs with -libjars command options which is a regression from MR1
- [MAPREDUCE-6076](#) - Zero map split input length combine with none zero map split input length may cause MR1 job hung sometimes.
- [HBASE-13374](#) - Small scanners (with particular configurations) do not return all rows
- [HBASE-13269](#) - Limit result array preallocation to avoid OOME with large scan caching values
- [HBASE-13422](#) - remove use of StandardCharsets in 0.98
- [HBASE-13335](#) - Update ClientSmallScanner and ClientSmallReversedScanner
- [HBASE-13262](#) - ResultScanner doesn't return all rows in Scan
- [HIVE-10646](#) - ColumnValue does not handle NULL_TYPE
- [HIVE-10453](#) - HS2 leaking open file descriptors when using UDFs
- [HIVE-9655](#) - Dynamic partition table insertion error

- [HIVE-10452](#) - Followup fix for HIVE-10202 to restrict it for script mode.
- [HIVE-10312](#) - SASL.QOP in JDBC URL is ignored for Delegation token Authentication
- [HIVE-10202](#) - Beeline outputs prompt+query on standard output when used in non-interactive mode
- [HIVE-10087](#) - Beeline's --silent option should suppress query from being echoed when running with -f option
- [HIVE-10085](#) - Lateral view on top of a view throws RuntimeException
- [HIVE-2828](#) - make timestamp accessible in the hbase KeyValue
- [HUE-2741](#) - [home] Hide the document move dialog
- [HUE-2732](#) - Hue isn't correctly doing add_column migrations with non-blank defaults
- [HUE-2513](#) - [fb] File list column sorting is broken
- [IMPALA-1519](#) - Fix wrapping of exprs via a TupleIsNullPredicate with analytics
- [IMPALA-1952](#) - Expand parsing of decimals to include scientific notation
- [IMPALA-1860](#) - INSERT/CTAS evaluates and applies constant predicates.
- [IMPALA-1900](#) - Assign predicates below analytic functions with a compatible partition by clause
- [IMPALA-1376](#) - Split up Planner into multiple classes.
- [IMPALA-1888](#) - FIRST_VALUE may produce incorrect results with preceding windows
- [IMPALA-1559](#) - FIRST_VALUE rewrite fn type might not match slot type
- [IMPALA-1808](#) - AnalyticEvalNode cannot handle partition/order by exprs with NaN
- [IMPALA-1562](#) - AnalyticEvalNode not properly handling nullable tuples
- [OOZIE-2063](#) - Cron syntax creates duplicate actions
- [OOZIE-2218](#) - META-INF directories in the war file have 777 permissions
- [OOZIE-1878](#) - Can't execute dryrun on the CLI
- [SENTRY-696](#) - Improve Metastoreplugin Cache Initialization time
- [SENTRY-703](#) - Calls to add_partition fail when passed a Partition object with a null location
- [SENTRY-408](#) - The URI permission should support more filesystem prefixes
- [SENTRY-598](#) - Hive binding should support enforcing URI privilege for transforms
- [SOLR-7478](#) - UpdateLog#close shutdown it's executor with interrupts before running close, preventing a clean close.
- [SOLR-7437](#) - Make HDFS transaction log replication factor configurable.
- [SOLR-7338](#) - A reloaded core will never register itself as active after a ZK session expiration
- [SOLR-7370](#) - FSHDFSUtils#recoverFileLease tries to recover the lease every one second after the first four second wait.
- [SPARK-6578](#) - Outbound channel in network library is not thread-safe, can lead to fetch failures
- [SQOOP-2343](#) - AsyncSqlRecordWriter stucks if any exception is thrown out in its close method
- [SQOOP-2286](#) - Ensure Sqoop generates valid avro column names
- [SQOOP-2283](#) - Support usage of --exec and --password-alias
- [SQOOP-2281](#) - Set overwrite on kite dataset
- [SQOOP-2282](#) - Add validation check for --hive-import and --append
- [SQOOP-2257](#) - Parquet target for imports with Hive overwrite option does not work
- [ZOOKEEPER-2146](#) - BinaryInputArchive readString should check length before allocating memory
- [ZOOKEEPER-2149](#) - Logging of client address when socket connection established

Published Known Issues Fixed

As a result of the fixes described above, the following issue, previously published in [Known Issues in CDH 5](#) on page 69, is also fixed.

—*Executing oozie job -config properties file -dryrun fails because of a code defect in argument parsing*

Bug: [OOZIE-1878](#)

Severity: Low

Workaround: None.

Known Issues Fixed in CDH 5.3.3

Upstream Issues Fixed

The following upstream issues are fixed in CDH 5.3.3:

- [HADOOP-11722](#) - Some Instances of Services using ZKDelegationTokenSecretManager go down when old token cannot be deleted
- [HADOOP-11469](#) - KMS should skip default.key.acl and whitelist.key.acl when loading key acl
- [HADOOP-11710](#) - Make CryptoOutputStream behave like DFSOutputStream wrt synchronization
- [HADOOP-11674](#) - oneByteBuf in CryptoInputStream and CryptoOutputStream should be non static
- [HADOOP-11445](#) - Bzip2Codec: Data block is skipped when position of newly created stream is equal to start of split
- [HADOOP-11620](#) - Add support for load balancing across a group of KMS for HA
- [HDFS-6830](#) - BlockInfo.addStorage fails when DN changes the storage for a block replica
- [HDFS-7961](#) - Trigger full block report after hot swapping disk
- [HDFS-7960](#) - The full block report should prune zombie storages even if they're not empty
- [HDFS-7575](#) - Upgrade should generate a unique storage ID for each volume
- [HDFS-7596](#) - NameNode should prune dead storages from storageMap
- [HDFS-7579](#) - Improve log reporting during block report rpc failure
- [HDFS-7208](#) - NN doesn't schedule replication when a DN storage fails
- [HDFS-6899](#) - Allow changing MiniDFScluster volumes per DN and capacity per volume
- [HDFS-6878](#) - Change MiniDFScluster to support StorageType configuration for individual directories
- [HDFS-6678](#) - MiniDFScluster may still be partially running after initialization fails.
- [HDFS-7961](#) - Trigger full block report after hot swapping disk
- [HDFS-7960](#) - The full block report should prune zombie storages even if they're not empty
- [YARN-3351](#) - AppMaster tracking URL is broken in HA
- [YARN-3242](#) - Asynchrony in ZK-close can lead to ZKRMStateStore watcher receiving events for old client
- [YARN-2865](#) - Application recovery continuously fails with "Application with id already present. Cannot duplicate"
- [MAPREDUCE-6275](#) - Race condition in FileOutputCommitter v2 for user-specified task output subdirs
- [MAPREDUCE-4815](#) - Speed up FileOutputCommitter#commitJob for many output files
- [HBASE-13131](#) - ReplicationAdmin leaks connections if there's an error in the constructor
- [HIVE-10086](#) - Hive throws error when accessing Parquet file schema using field name match
- [HIVE-10098](#) - HS2 local task for map join fails in KMS encrypted cluster
- [HIVE-7426](#) - ClassCastException: ...IntWritable cannot be cast to ...Text involving ql.udf.generic.GenericUDFBasePad.evaluate
- [HIVE-7737](#) - Hive logs full exception for table not found
- [HIVE-9749](#) - ObjectStore schema verification logic is incorrect
- [HIVE-9788](#) - Make double quote optional in tsv/csv/dsv output
- [HIVE-9755](#) - Hive built-in "ngram" UDAF fails when a mapper has no matches.
- [HIVE-9770](#) - Beeline ignores --showHeader for non-tabular output formats i.e csv,tsv,dsv
- [HIVE-8688](#) - serialized plan OutputStream is not being closed
- [HIVE-9716](#) - Map job fails when table's LOCATION does not have scheme
- [HIVE-5857](#) - Reduce tasks do not work in uber mode in YARN
- [HIVE-8938](#) - Compiler should save the transform URI as input entity
- [HUE-2569](#) - [home] Delete project is broken
- [HUE-2529](#) - Increase the character limit of 'Name' Textfield in Useradmin Ldap Sync Groups
- [HUE-2506](#) - [search] Marker map does not display with HTML widget
- [HUE-1663](#) - [core] Option to either follow or not LDAP referrals for auth
- [HUE-2198](#) - [core] Reduce noise such as "handle_other(): Mutual authentication unavailable on 200 response"
- [SENTRY-683](#) - HDFS service client should ensure the kerberos ticket validity before new service connection
- [SENTRY-654](#) - Calls to append_partition fail when Sentry is enabled

- [SENTRY-664](#) - After Namenode is restarted, Path updates remain unsynched
- [SENTRY-665](#) - PathsUpdate.parsePath needs to handle special characters
- [SENTRY-652](#) - Sentry fails to parse spaces when HDFS ACL sync enabled
- [SOLR-7092](#) - Stop the HDFS lease recovery retries on HdfsTransactionLog on close and try to avoid lease recovery on closed files.
- [SOLR-7141](#) - RecoveryStrategy: Raise time that we wait for any updates from the leader before they saw the recovery state to have finished.
- [SOLR-7113](#) - Multiple calls to UpdateLog#init is not thread safe with respect to the HDFS FileSystem client object usage.
- [SOLR-7134](#) - Replication can still cause index corruption.
- [SQOOP-1764](#) - Numeric Overflow when getting extent map
- [IMPALA-1658](#) - Add compatibility flag for Hive-Parquet-Timestamps
- [IMPALA-1820](#) - Start with small pages for hash tables during repartitioning
- [IMPALA-1897](#) - Fixes for old hash join and agg
- [IMPALA-1894](#) - Fix old aggregation node hash table cleanup
- [IMPALA-1863](#) - Avoid deadlock across fragment instances
- [IMPALA-1915](#) - Fix query hang in BufferedBlockMgr:FindBlock()
- [IMPALA-1890](#) - Fixing a race between ~BufferedBlockMgr() and the WriteComplete() call
- [IMPALA-1738](#) - Use snprintf() instead of lexical_cast() in float-to-string casts
- [IMPALA-1865](#) - Fix partition spilling cleanup when new stream OOMs
- [IMPALA-1835](#) - Keep the fragment alive for TransmitData()
- [IMPALA-1805](#) - Impala's ACLs check do not consider all group ACLs, only checked first one.
- [IMPALA-1794](#) - Fix infinite loop opening or closing file with invalid metadata
- [IMPALA-1801](#) - external-data-source-executor leaking global jni refs
- [IMPALA-1712](#) - Unexpected remote bytes read counter was not being reset properly
- [IMPALA-1636](#) - Generalize index-based partition pruning to allow constant expressions

Published Known Issues Fixed

As a result of the above fixes, the following issues, previously published as [Known Issues in CDH 5](#) on page 69, are also fixed.

— *After upgrade from a release earlier than CDH 5.2.0, storage IDs may no longer be unique*

As of CDH 5.2, each storage volume on a DataNode should have its own unique `storageID`, but in clusters upgraded from CDH 4, or CDH 5 releases earlier than CDH 5.2.0, each volume on a given DataNode shares the same `storageID`, because the HDFS upgrade does not properly update the IDs to reflect the new naming scheme. This causes problems with load balancing. The problem affects only clusters upgraded from CDH 5.1.x and earlier to CDH 5.2 or later. Clusters that are new as of CDH 5.2.0 or later do not have the problem.

Bug: [HDFS-7575](#)

Severity: Medium

Workaround: Upgrade to a later or patched version of CDH.

Known Issues Fixed in CDH 5.3.2

Upstream Issues Fixed

The following upstream issues are fixed in CDH 5.3.2:

- [AVRO-1630](#) - Creating Builder from instance loses data
- [AVRO-1628](#) - Add Schema.createUnion(Schema... type)
- [AVRO-1539](#) - Add FileSystem-based FsInput Constructor
- [AVRO-1623](#) - GenericData#validate() of enum: IndexOutOfBoundsException
- [AVRO-1614](#) - Always getting a value...
- [AVRO-1592](#) - Java keyword as an enum constant in Avro schema file causes deserialization to fail.

- [AVRO-1619](#) - Generate better JavaDoc
- [AVRO-1622](#) - Add missing license headers
- [AVRO-1604](#) - ReflectData.AllowNull fails to generate schemas when @Nullable is present.
- [AVRO-1407](#) - NettyTransceiver can cause a infinite loop when slow to connect
- [AVRO-834](#) - Data File corruption recovery tool
- [AVRO-1596](#) - Cannot read past corrupted block in Avro data file
- [HADOOP-11350](#) - The size of header buffer of HttpServer is too small when HTTPS is enabled
- [HDFS-7707](#) - Edit log corruption due to delayed block removal again
- [HDFS-7718](#) - Store KeyProvider in ClientContext to avoid leaking key provider threads when using FileContext
- [HDFS-6425](#) - Large postponedMisreplicatedBlocks has impact on blockReport latency
- [HDFS-7560](#) - ACLs removed by removeDefaultAcl() will be back after NameNode restart/failover
- [HDFS-7513](#) - HDFS inotify: add defaultBlockSize to CreateEvent
- [HDFS-7158](#) - Reduce the memory usage of WebImageViewer
- [HDFS-7497](#) - Inconsistent report of decommissioning DataNodes between dfsadmin and NameNode webui
- [HDFS-6917](#) - Add an hdfs debug command to validate blocks, call recoverlease, etc.
- [HDFS-6779](#) - Add missing version subcommand for hdfs
- [YARN-2697](#) - RMAuthenticationHandler is no longer useful
- [YARN-2656](#) - RM web services authentication filter should add support for proxy user
- [YARN-3082](#) - Non thread safe access to systemCredentials in NodeHeartbeatResponse processing
- [YARN-3079](#) - Scheduler should also update maximumAllocation when updateNodeResource.
- [YARN-2992](#) - ZKRMStateStore crashes due to session expiry
- [YARN-2675](#) - containersKilled metrics is not updated when the container is killed during localization
- [YARN-2715](#) - Proxy user is problem for RPC interface if yarn.resourcemanager.webapp.proxyuser is not set.
- [MAPREDUCE-6198](#) - NPE from JobTracker#resolveAndAddToTopology in MR1 cause initJob and heartbeat failure.
- [MAPREDUCE-6196](#) - Fix BigDecimal ArithmeticException in PiEstimator
- [HBASE-12540](#) - TestRegionServerMetrics#testMobMetrics test failure
- [HBASE-12533](#) - staging directories are not deleted after secure bulk load
- [HBASE-12077](#) - FilterLists create many ArrayList\$Itr objects per row.
- [HBASE-12386](#) - Replication gets stuck following a transient zookeeper error to remote peer cluster
- [HBASE-11979](#) - Compaction progress reporting is wrong
- [HBASE-12445](#) - hbase is removing all remaining cells immediately after the cell marked with marker = KeyValue.Type.DeleteColumn via PUT
- [HBASE-12837](#) - ReplicationAdmin leaks zk connections
- [HIVE-7647](#) - Beeline does not honor --headerInterval and --color when executing with "-e"
- [HIVE-7733](#) - Ambiguous column reference error on query
- [HIVE-9303](#) - Parquet files are written with incorrect definition levels
- [HIVE-8444](#) - update pom to junit 4.11
- [HIVE-9474](#) - truncate table changes permissions on the target
- [HIVE-9462](#) - HIVE-8577 - breaks type evolution
- [HIVE-9482](#) - Hive parquet timestamp compatibility
- [HIVE-6308](#) - COLUMNS_V2 Metastore table not populated for tables created without an explicit column list.
- [HIVE-9502](#) - Parquet cannot read Map types from files written with Hive 0.12 or earlier
- [HIVE-9445](#) - Revert HIVE-5700 - enforce single date format for partition column storage
- [HIVE-9393](#) - reduce noisy log level of ColumnarSerDe.java:116 from INFO to DEBUG
- [HIVE-7800](#) - Parquet Column Index Access Schema Size Checking
- [HIVE-9330](#) - DummyTxnManager will throw NPE if WriteEntity writeType has not been set
- [HIVE-9265](#) - Hive with encryption throws NPE to fs path without schema
- [HIVE-9199](#) - Excessive exclusive lock used in some DDLs with DummyTxnManager
- [HIVE-6978](#) - beeline always exits with 0 status, should exit with non-zero status on error

- [HUE-2556](#) - [core] Cannot update project tags of a document
- [HUE-2528](#) - Partitions limit gets capped to 1000 despite configuration
- [HUE-2548](#) - [metastore] Create table then load data does redirect to the table page
- [HUE-2525](#) - [core] Fix manual install of samples
- [HUE-2501](#) - [metastore] Creating a table with header files bigger than 64MB truncates it
- [HUE-2484](#) - [beeswax] Configure support for Hive Server2 LDAP authentication
- [HUE-2532](#) - [search] Fix share URL on Internet Explorer
- [HUE-2531](#) - [impala] Autogrow missing result list
- [HUE-2524](#) - [impala] Sort numerically recent queries tab
- [HUE-2495](#) - [oozie] Improve dashboards sorting mechanism
- [HUE-2511](#) - [impala] Infinite scroll keeps fetching results even if finished
- [HUE-2102](#) - [oozie] Workflow with credentials can't be used with Coordinator
- [HUE-2152](#) - [pig] Credentials support in editor
- [OOZIE-2131](#) - Add flag to sqoop action to skip hbase delegation token generation
- [OOZIE-2047](#) - Oozie does not support Hive tables that use datatypes introduced since Hive 0.8
- [OOZIE-2102](#) - Streaming actions are broken cause of incorrect method signature
- [PARQUET-173](#) - StatisticsFilter doesn't handle And properly
- [PARQUET-157](#) - Divide by zero in logging code
- [PARQUET-142](#) - parquet-tools doesn't filter _SUCCESS file
- [PARQUET-124](#) - parquet.hadoop.ParquetOutputCommitter.commitJob() throws parquet.io.ParquetEncodingException
- [PARQUET-136](#) - NPE thrown in StatisticsFilter when all values in a string/binary column trunk are null
- [PARQUET-168](#) - Wrong command line option description in parquet-tools
- [PARQUET-145](#) - InternalParquetRecordReader.close() should not throw an exception if initialization has failed
- [PARQUET-140](#) - Allow clients to control the GenericData object that is used to read Avro records
- [SOLR-7033](#) - [RecoveryStrategy should not publish any state when closed / cancelled.
- [SOLR-5961](#) - Solr gets crazy on /overseer/queue state change
- [SOLR-6640](#) - Replication can cause index corruption
- [SOLR-5875](#) - QueryComponent.mergelds() unmarshals all docs' sort field values once per doc instead of once per shard
- [SOLR-6919](#) - Log REST info before executing
- [SOLR-6969](#) - When opening an HDFSTransactionLog for append we must first attempt to recover it's lease to prevent data loss.
- [SOLR-5515](#) - NPE when getting stats on date field with empty result on solrcloud
- [SPARK-3778](#) - newAPIHadoopRDD doesn't properly pass credentials for secure hdfs on yarn
- [SPARK-4835](#) - Streaming saveAsHadoopFiles() methods may throw FileAlreadyExistsException during checkpoint recovery
- [SQOOP-2057](#) - Skip delegation token generation flag during hbase import
- [SQOOP-1779](#) - Add support for --hive-database when importing Parquet files into Hive
- [IMPALA-1622](#) - Fix overflow in StringParser::StringToFloatInternal()
- [IMPALA-1614](#) - Compute stats fails if table name starts with number
- [IMPALA-1623](#) - unix_timestamp() does not return correct time
- [IMPALA-1535](#) - Partition pruning with NULL
- [IMPALA-1606](#) - Impala does not always give short name to Llama
- [IMPALA-1120](#) - Fetch column statistics using Hive 0.13 bulk API

In addition, CDH 5.3.2 reverts [YARN-2713](#), which has caused problems since its inclusion in CDH 5.3.0.

Published Known Issues Fixed

As a result of the above fixes, the following issues, previously published as [Known Issues in CDH 5](#) on page 69, are also fixed.

—*Hive does not support Parquet schema evolution*

Adding a new column to a Parquet table causes queries on that table to fail with a `column not found` error.

Bug: [HIVE-7800](#)

Severity: Medium

Workaround: Use Impala instead; Impala handles Parquet schema evolution correctly.

Known Issues Fixed in CDH 5.3.1

Upstream Issues Fixed

The following upstream issues are fixed in CDH 5.3.1:

- [YARN-2975](#) - FSLeafQueue app lists are accessed without required locks
- [YARN-2010](#) - Handle app-recovery failures gracefully
- [YARN-3027](#) - Scheduler should use totalAvailable resource from node instead of availableResource for maxAllocation
- [HIVE-9445](#) - Revert HIVE-5700 - enforce single date format for partition column storage
- [IMPALA-1668](#) - TSaslServerTransport::Factory::getTransport() leaks transport map entries
- [IMPALA-1674](#) - IMPALA-1556 causes memory leak with secure connections

Published Known Issues Fixed

As a result of the above fixes, the following issues, previously published as [Known Issues in CDH 5](#) on page 69, are also fixed.

—*Upgrading a PostgreSQL Hive Metastore from Hive 0.12 to Hive 0.13 may result in a corrupt metastore*

[HIVE-5700](#) introduced a serious bug into the Hive Metastore upgrade scripts. This bug affects users who have a PostgreSQL Hive Metastore and have *at least one table* which is partitioned by date and the value is stored as a date type (not string).

Bug: [HIVE-5700](#)

Severity: High

Workaround: None. Do not upgrade your PostgreSQL metastore to version 0.13 if you satisfy the condition stated above.

Known Issues Fixed in CDH 5.3.0

The following topics describe known issues fixed in CDH 5.3.0.

Apache Hadoop HDFS

NameNode - KMS communication fails after long periods of inactivity

Encrypted files and encryption zones cannot be created if a long period of time (by default, 20 hours) has passed since the last time the KMS and NameNode communicated.

Bug: [HADOOP-11187](#)

Severity: Low

Workaround: There are two possible workarounds to this issue:

- You can increase the KMS authentication token validity period to a very high number. Since the default value is 10 hours, this bug will only be encountered after 20 hours of no communication between the NameNode and the KMS. Add the following property to the `kms-site.xml` Safety Valve:

```
<property>
<name>hadoop.kms.authentication.token.validity</name>
<value>SOME VERY HIGH NUMBER</value>
</property>
```

- You can switch the KMS signature secret provider to the string secret provider by adding the following property to the `kms-site.xml` Safety Valve:

```
<property>
<name>hadoop.kms.authentication.signature.secret</name>
<value>SOME VERY SECRET STRING</value>
</property>
```

DataNodes may become unresponsive to block creation requests

In releases earlier than CDH 5.2.3, DataNodes may become unresponsive to block creation requests from clients when the directory scanner is running.

Bug: [HDFS-7489](#)

Severity: High

Workaround: Upgrade to CDH 5.2.3 or later.

Apache Hive

— *UDF translate() does not accept arguments of type CHAR or VARCHAR*

Bug: [HIVE-6622](#)

Severity: Low

Workaround: Cast the argument to type String.

— *Hive's Timestamp type cannot be stored in Parquet*

Tables containing timestamp columns can't use Parquet as the storage engine.

Bug: [HIVE-6394](#)

Severity: Low

Workaround: Use a different file format.

Known Issues Fixed in CDH 5.2.5

Upstream Issues Fixed

The following upstream issues are fixed in CDH 5.2.5:

- [HADOOP-11350](#) - The size of header buffer of HttpServer is too small when HTTPS is enabled
- [HADOOP-11710](#) - Make CryptoOutputStream behave like DFSOutputStream wrt synchronization
- [HADOOP-11674](#) - oneByteBuf in CryptoInputStream and CryptoOutputStream should be non static
- [HDFS-6830](#) - BlockInfo.addStorage fails when DN changes the storage for a block replica
- [HDFS-7960](#) - The full block report should prune zombie storages even if they're not empty
- [HDFS-6425](#) - Large postponedMisreplicatedBlocks has impact on blockReport latency
- [HDFS-7575](#) - Upgrade should generate a unique storage ID for each volume
- [HDFS-7596](#) - NameNode should prune dead storages from storageMap
- [HDFS-7579](#) - Improve log reporting during block report rpc failure
- [HDFS-7208](#) - NN doesn't schedule replication when a DN storage fails
- [HDFS-6899](#) - Allow changing MiniDFScluster volumes per DN and capacity per volume
- [HDFS-6878](#) - Change MiniDFScluster to support StorageType configuration for individual directories
- [HDFS-6678](#) - MiniDFScluster may still be partially running after initialization fails.
- [HDFS-7575](#) - Upgrade should generate a unique storage ID for each volume
- [HDFS-7960](#) - The full block report should prune zombie storages even if they're not empty
- [YARN-570](#) - Time strings are formatted in different timezone
- [YARN-2251](#) - Avoid negative elapsed time in JHS/MRAM web UI and services
- [YARN-3242](#) - Asynchrony in ZK-close can lead to ZKRMStateStore watcher receiving events for old client

- [MAPREDUCE-5957](#) - AM throws ClassNotFoundException with job classloader enabled if custom output format/commiter is used
- [MAPREDUCE-6076](#) - Zero map split input length combine with none zero map split input length may cause MR1 job hung sometimes.
- [MAPREDUCE-6275](#) - Race condition in FileOutputCommitter v2 for user-specified task output subdirs
- [MAPREDUCE-4815](#) - Speed up FileOutputCommitter#commitJob for many output files
- [HIVE-2828](#) - make timestamp accessible in the hbase KeyValue
- [HIVE-2828](#) - make timestamp accessible in the hbase KeyValue
- [HIVE-7433](#) - ColumnMappers.ColumnMapping should expose public accessors for its fields
- [HIVE-6148](#) - Support arbitrary structs stored in HBase
- [HIVE-6147](#) - Support avro data stored in HBase columns
- [HIVE-6584](#) - Add HiveHBaseTableSnapshotInputFormat
- [HIVE-6411](#) - Support more generic way of using composite key for HBaseHandler
- [HIVE-6677](#) - HBaseSerDe needs to be refactored
- [HIVE-9934](#) - Vulnerability in LdapAuthenticationProviderImpl enables HiveServer2 client to degrade the authentication mechanism to "none", allowing authentication without password
- [HIVE-7737](#) - Hive logs full exception for table not found
- [HIVE-9716](#) - Map job fails when table's LOCATION does not have scheme
- [HIVE-8688](#) - serialized plan OutputStream is not being closed
- [HIVE-5857](#) - Reduce tasks do not work in uber mode in YARN
- [HUE-2446](#) - Migrating from CDH 4.7 to CDH 5.0.1+/Hue 3.5+ will fail
- [HUE-2371](#) - [sentry] Sentry URI should be created only with a ALL permission
- [HUE-1663](#) - [core] Option to either follow or not LDAP referrals for auth
- [SENTRY-654](#) - Calls to append_partition fail when Sentry is enabled
- [SOLR-7092](#) - Stop the HDFS lease recovery retries on HdfsTransactionLog on close and try to avoid lease recovery on closed files.
- [SOLR-7134](#) - Replication can still cause index corruption.
- [SOLR-7113](#) - Multiple calls to UpdateLog#init is not thread safe with respect to the HDFS FileSystem client object usage.
- [SOLR-7141](#) - RecoveryStrategy: Raise time that we wait for any updates from the leader before they saw the recovery state to have finished.
- [SQOOP-1764](#) - Numeric Overflow when getting extent map
- [IMPALA-1658](#): Add compatibility flag for Hive-Parquet-Timestamps
- [IMPALA-1794](#): Fix infinite loop opening/closing file w/ invalid metadata
- [IMPALA-1801](#): external-data-source-executor leaking global jni refs

Known Issues Fixed in CDH 5.2.4

Upstream Issues Fixed

The following upstream issues are fixed in CDH 5.2.4:

- [HDFS-7707](#) - Edit log corruption due to delayed block removal again
- [YARN-2846](#) - Incorrect persist exit code for running containers in reacquireContainer() that interrupted by NodeManager restart.
- [HIVE-7733](#) - Ambiguous column reference error on query
- [HIVE-8444](#) - update pom to junit 4.11
- [HIVE-9474](#) - truncate table changes permissions on the target
- [HIVE-6308](#) - COLUMNS_V2 Metastore table not populated for tables created without an explicit column list.
- [HIVE-9445](#) - Revert HIVE-5700 - enforce single date format for partition column storage
- [HIVE-7800](#) - Parquet Column Index Access Schema Size Checking Checking
- [HIVE-9393](#) - reduce noisy log level of ColumnarSerDe.java:116 from INFO to DEBUG

- [HUE-2501](#) - [metastore] Creating a table with header files bigger than 64MB truncates it
- [SOLR-7033](#) - [RecoveryStrategy should not publish any state when closed / cancelled.
- [SOLR-5961](#) - Solr gets crazy on /overseer/queue state change
- [SOLR-6640](#) - Replication can cause index corruption
- [SOLR-6920](#) - During replication use checksums to verify if files are the same
- [SOLR-5875](#) - QueryComponent.mergelds() unmarshals all docs' sort field values once per doc instead of once per shard
- [SOLR-6919](#) - Log REST info before executing
- [SOLR-6969](#) - When opening an HDFSTransactionLog for append we must first attempt to recover its lease to prevent data loss
- [IMPALA-1471](#): Bug in spilling of PHJ that was affecting left anti and outer joins.
- [IMPALA-1451](#): Empty Row in HBase triggers NPE in Planner
- [IMPALA-1535](#): Partition pruning with NULL
- [IMPALA-1483](#): Substitute TupleIsNullPredicates to refer to physical analytic output.
- [IMPALA-1674](#): Fix serious memory leak in TSaslTransport
- [IMPALA-1668](#): Fix leak of transport objects in TSaslServerTransport::Factory
- [IMPALA-1565](#): Python sasl client transport perf issue
- [IMPALA-1556](#): Kerberos fetches 3x slower
- [IMPALA-1120](#): Fetch column statistics using Hive 0.13 bulk API

Published Known Issues Fixed

As a result of the above fixes, the following issues, previously published as [Known Issues in CDH 5](#) on page 69, are also fixed:

—*Hive does not support Parquet schema evolution*

Adding a new column to a Parquet table causes queries on that table to fail with a `column not found` error.

Bug: [HIVE-7800](#)

Severity: Medium

Workaround: Use Impala instead; Impala handles Parquet schema evolution correctly.

Known Issues Fixed in CDH 5.2.3

Upstream Issues Fixed

The following upstream issues are fixed in CDH 5.2.3:

- [AVRO-1623](#) - GenericData#validate() of enum: IndexOutOfBoundsException
- [AVRO-1622](#) - Add missing license headers
- [AVRO-1604](#) - ReflectData.AllowNull fails to generate schemas when @Nullable is present.
- [AVRO-1407](#) - NettyTransceiver can cause a infinite loop when slow to connect
- [AVRO-834](#) - Data File corruption recovery tool
- [AVRO-1596](#) - Cannot read past corrupted block in Avro data file
- [CRUNCH-480](#) - AvroParquetFileSource doesn't properly configure user-supplied read schema
- [CRUNCH-479](#) - Writing to target with WriteMode.APPEND merges values into PCollection
- [CRUNCH-477](#) - Fix HFileTargetIT failures on hadoop1 under Java 1.7/1.8
- [CRUNCH-473](#) - Use specific class type for case class serialization
- [CRUNCH-473](#) - Use specific class type for case class serialization
- [CRUNCH-472](#) - Add Scrunch serialization support for Java Enums
- [HADOOP-11068](#) - Match hadoop.auth cookie format to jetty output
- [HADOOP-11343](#) - Overflow is not properly handled in calculating final iv for AES CTR
- [HADOOP-11301](#) - [optionally] update jmx cache to drop old metrics
- [HADOOP-11085](#) - Excessive logging by org.apache.hadoop.util.Progress when value is NaN
- [HADOOP-11247](#) - Fix a couple javac warnings in NFS

- [HADOOP-11195](#) - Move Id-Name mapping in NFS to the hadoop-common area for better maintenance
- [HADOOP-11130](#) - NFS updateMaps OS check is reversed
- [HADOOP-10990](#) - Add missed NFSv3 request and response classes
- [HADOOP-11323](#) - WritableComparator#compare keeps reference to byte array
- [HDFS-7560](#) - ACLs removed by removeDefaultAcl() will be back after NameNode restart/failover
- [HDFS-7367](#) - HDFS short-circuit read cannot negotiate shared memory slot and file descriptors when SASL is enabled on DataTransferProtocol.
- [HDFS-7489](#) - Incorrect locking in FsVolumeList#checkDirs can hang datanodes
- [HDFS-7158](#) - Reduce the memory usage of WebImageViewer
- [HDFS-7497](#) - Inconsistent report of decommissioning DataNodes between dfsadmin and NameNode webui
- [HDFS-7146](#) - NFS ID/Group lookup requires SSSD enumeration on the server
- [HDFS-7387](#) - NFS may only do partial commit due to a race between COMMIT and write
- [HDFS-7356](#) - Use DirectoryListing.hasMore() directly in nfs
- [HDFS-7180](#) - NFSv3 gateway frequently gets stuck due to GC
- [HDFS-7259](#) - Unresponsive NFS mount point due to deferred COMMIT response
- [HDFS-6894](#) - Add XDR parser method for each NFS response
- [HDFS-6850](#) - Move NFS out of order write unit tests into TestWrites class
- [HDFS-7385](#) - ThreadLocal used in FSEditLog class causes FSImage permission mess up
- [HDFS-7409](#) - Allow dead nodes to finish decommissioning if all files are fully replicated
- [HDFS-7373](#) - Clean up temporary files after fsimage transfer failures
- [HDFS-7225](#) - Remove stale block invalidation work when DN re-registers with different UUID
- [YARN-2721](#) - Race condition: ZKRMStateStore retry logic may throw NodeExist exception
- [YARN-2975](#) - FSLeafQueue app lists are accessed without required locks
- [YARN-2992](#) - ZKRMStateStore crashes due to session expiry
- [YARN-2910](#) - FSLeafQueue can throw ConcurrentModificationException
- [YARN-2816](#) - NM fail to start with NPE during container recovery
- [MAPREDUCE-6198](#) - NPE from JobTracker#resolveAndAddToTopology in MR1 cause initJob and heartbeat failure.
- [MAPREDUCE-6169](#) - MergeQueue should release reference to the current item from key and value at the end of the iteration to save memory.
- [HBASE-11794](#) - StripeStoreFlusher causes NullPointerException
- [HBASE-12077](#) - FilterLists create many ArrayList\$Itr objects per row.
- [HBASE-12386](#) - Replication gets stuck following a transient zookeeper error to remote peer cluster
- [HBASE-11979](#) - Compaction progress reporting is wrong
- [HBASE-12529](#) - Use ThreadLocalRandom for RandomQueueBalancer
- [HBASE-12445](#) - hbase is removing all remaining cells immediately after the cell marked with marker = KeyValue.Type.DeleteColumn via PUT
- [HBASE-12460](#) - Moving Chore to hbase-common module.
- [HBASE-12366](#) - Add login code to HBase Canary tool.
- [HBASE-12447](#) - Add support for setTimeRange for RowCounter and CellCounter
- [HIVE-9330](#) - DummyTxnManager will throw NPE if WriteEntity writeType has not been set
- [HIVE-9199](#) - Excessive exclusive lock used in some DDLs with DummyTxnManager
- [HIVE-6835](#) - Reading of partitioned Avro data fails if partition schema does not match table schema
- [HIVE-6978](#) - beeline always exits with 0 status, should exit with non-zero status on error
- [HIVE-8891](#) - Another possible cause to NucleusObjectNotFoundException from drops/rollback
- [HIVE-8874](#) - Error Accessing HBase from Hive via Oozie on Kerberos 5.0.1 cluster
- [HIVE-8916](#) - Handle user@domain username under LDAP authentication
- [HIVE-8889](#) - JDBC Driver ResultSet.getXXXXXX(String columnName) methods Broken
- [HIVE-9445](#) - Revert HIVE-5700 - enforce single date format for partition column storage
- [HIVE-5454](#) - HCatalog runs a partition listing with an empty filter

- [HIVE-8784](#) - Querying partition does not work with JDO enabled against PostgreSQL
- [HUE-2484](#) - [beeswax] Configure support for Hive Server2 LDAP authentication
- [HUE-2102](#) - [oozie] Workflow with credentials can't be used with Coordinator
- [HUE-2152](#) - [pig] Credentials support in editor
- [HUE-2472](#) - [impala] Stabilize result retrieval
- [HUE-2406](#) - [search] New dashboard page has a margin problem
- [HUE-2373](#) - [search] Heatmap can break
- [HUE-2395](#) - [search] Broken widget in Solr Apache logs example
- [HUE-2414](#) - [search] Timeline chart breaks when there's no extraSeries defined
- [HUE-2342](#) - [impala] SSL encryption
- [HUE-2426](#) - [pig] Dashboard gives a 500 error
- [HUE-2430](#) - [pig] Progress bars of running scripts not updated on Dashboard
- [HUE-2411](#) - [useradmin] Lazy load user and group list in permission sharing popup
- [HUE-2398](#) - [fb] Drag and Drop hover message should not appear when elements originating in DOM are dragged
- [HUE-2401](#) - [search] Visually report selected and excluded values for ranges too
- [HUE-2389](#) - [impala] Expand results table after the results are added to datatables
- [HUE-2360](#) - [sentry] Sometimes Groups are not loaded we see the input box instead
- [IMPALA-1453](#) - Fix many bugs with HS2 FETCH_FIRST
- [IMPALA-1623](#) - unix_timestamp() does not return correct time
- [IMPALA-1606](#) - Impala does not always give short name to Llama
- [IMPALA-1475](#) - accept unmangled native UDF symbols
- [OOZIE-2102](#) - Streaming actions are broken cause of incorrect method signature
- [PARQUET-145](#) - InternalParquetRecordReader.close() should not throw an exception if initialization has failed
- [PARQUET-140](#) - Allow clients to control the GenericData object that is used to read Avro records
- [PIG-4330](#) - Regression test for PIG-3584 - AvroStorage does not correctly translate arrays of strings
- [PIG-3584](#) - AvroStorage does not correctly translate arrays of strings
- [SOLR-5515](#) - NPE when getting stats on date field with empty result on solrcloud

Published Known Issues Fixed

As a result of the above fixes, the following issues, previously published as [Known Issues in CDH 5](#) on page 69, are also fixed.

— *Upgrading a PostgreSQL Hive Metastore from Hive 0.12 to Hive 0.13 may result in a corrupt metastore*

[HIVE-5700](#) introduced a serious bug into the Hive Metastore upgrade scripts. This bug affects users who have a PostgreSQL Hive Metastore and have *at least one table* which is partitioned by date and the value is stored as a date type (not string).

Bug: [HIVE-5700](#)

Severity: High

Workaround: None. Do not upgrade your PostgreSQL metastore to version 0.13 if you satisfy the condition stated above.

DataNodes may become unresponsive to block creation requests

DataNodes may become unresponsive to block creation requests from clients when the directory scanner is running.

Bug: [HDFS-7489](#)

Severity: Low

Workaround: Disable the directory scanner by setting `dfs.datanode.directoryscan.interval` to `-1`.

Known Issues Fixed in CDH 5.2.1

The following topics describe known issues fixed in CDH 5.2.1. See [What's New in CDH 5.2.1](#) on page 9 for a list of the most important upstream problems fixed in this release, and [Cloudera Impala Fixed Issues](#) on page 124 for fixed issues for Impala.

Apache Hadoop

— *Files inside encryption zones cannot be read in Hue*

Hue uses either WebHDFS or HttpFS to access files. Both are proxy user clients of KMS and the KMS client library does not currently handle proxy users correctly.

Bug: [HADOOP-11176](#)

Severity: High

Workaround: None

— *Both ResourceManagers can end up in Standby mode*

After a restart, if an application fails to recover, both Resource Managers could end up in Standby mode.

Bug: [YARN-2588](#), [YARN-2010](#)

Severity: High

Workarounds:

- Stop the RM. Format the state store using `yarn resourcemanager -format-state-store`. Applications that were running before the RM went down will not be recovered.
- You can limit the number of completed applications the RM state-store stores (`yarn.resourcemanager.state-store.max-completed-applications`) to reduce the chances of running into this problem.

Apache Oozie

— *Using cron-like syntax for Coordinator frequencies can result in duplicate actions*

Every `throttle` number of actions will be a duplicate. For example, if the throttle is set to 5, every fifth action will be a duplicate.

Bug: [OOZIE-2063](#)

Severity: High

Workaround: If possible, use the older syntax to specify an equivalent frequency.

Known Issues Fixed in CDH 5.2.0

The following topics describe known issues fixed in CDH 5.2.0.

Apache HBase

— *hbase.zookeeper.useMulti set to false by default*

The default value of `hbase.zookeeper.useMulti` was changed from `true` to `false` in CDH 5. In CDH 5.2, the default is changed back to `true`. This affects environments with HBase replication enabled and large replication queues.

Bug: None

Severity: Low

Workaround: Enable `hbase.zookeeper.useMulti` by setting the value to `true` in `hbase-site.xml`.

Apache Hadoop

— *Hadoop shell commands which reference the root directory ("/") do not work*

Bug: [HDFS-5888](#)

Severity: Low

Workaround: None

— *ResourceManager High Availability with manual failover does not work on secure clusters*

Bug: [YARN-1640](#)

Severity: High

Workaround: Enable automatic failover; this requires ZooKeeper.

Apache Hive

— *Select * fails on Parquet tables with the `map` data type*

Bug: [HIVE-6575](#)

Severity: Low

Workaround: Use the map's column name in the `SELECT` statement.

Cloudera Search

— *Malicious users could update information by circumventing Sentry checks*

A sophisticated malicious user could update restricted content by setting the `update.distrib` parameter to bypass Sentry's index-level checks.

With Search for CDH 5.2 and later, Sentry always checks for index-level access control settings, preventing unauthorized updates.

Bug: None.

Severity: Medium.

Workaround: None.

— *Kerberos name rules were not followed*

`SOLR_AUTHENTICATION_KERBEROS_NAME_RULES`, which is specified in `/etc/default/solr` would sometimes not be respected, even if those rules were also specified using the `hadoop.security.auth_to_local` property in `SOLR_HDFS_CONFIG/core-site.xml`.

With Search for CDH 5.2 and later, Kerberos name rules are followed.

Bug: None.

Severity: Medium.

Workaround: None.

Known Issues Fixed in CDH 5.1.5

This is a maintenance release that fixes the following issues:

- [HDFS-7960](#) - The full block report should prune zombie storages even if they're not empty
- [HDFS-7278](#) - Add a command that allows sysadmins to manually trigger full block reports from a DN
- [HDFS-6831](#) - Inconsistency between 'hdfs dfsadmin' and 'hdfs dfsadmin -help'
- [HDFS-7596](#) - NameNode should prune dead storages from storageMap
- [HDFS-7208](#) - NN doesn't schedule replication when a DN storage fails
- [HDFS-7575](#) - Upgrade should generate a unique storage ID for each volume
- [HDFS-6529](#) - Trace logging for RemoteBlockReader2 to identify remote datanode and file being read
- [YARN-570](#) - Time strings are formatted in different timezone
- [YARN-2251](#) - Avoid negative elapsed time in JHS/MRAM web UI and services
- [YARN-2588](#) - Standby RM does not transitionToActive if previous transitionToActive is failed with ZK exception.
- [HIVE-8693](#) - Separate out fair scheduler dependency from hadoop 0.23 shim
- [HIVE-8634](#) - HiveServer2 fair scheduler queue mapping doesn't handle the secondary groups rules correctly
- [HIVE-6403](#) - uncorrelated subquery is failing with `auto.convert.join=true`

- [HIVE-5945](#) - `ql.plan.ConditionalResolverCommonJoin.resolveMapJoinTask` also sums those tables which are not used in the child of this conditional task.
- [HIVE-8916](#) - Handle `user@domain` username under LDAP authentication
- [HIVE-8874](#) - Error Accessing HBase from Hive via Oozie on Kerberos 5.0.1 cluster
- [HIVE-9716](#) - Map job fails when table's LOCATION does not have scheme
- [HIVE-8784](#) - Querying partition does not work with JDO enabled against PostgreSQL
- [HUE-2484](#) - [beeswax] Configure support for Hive Server2 LDAP authentication
- [HUE-2446](#) - Migrating from CDH 4.7 to CDH 5.0.1+/Hue 3.5+ will fail
- [PARQUET-107](#) - Add option to disable summary metadata aggregation after MR jobs
- [SOLR-6268](#) - `HdfsUpdateLog` has a race condition that can expose a closed HDFS FileSystem instance and should close it's FileSystem instance if either inherited close method is called.
- [SOLR-6393](#) - Improve transaction log replay speed on HDFS.
- [SOLR-6403](#) - TransactionLog replay status logging.
- [IMPALA-1801](#) - external-data-source-executor leaking global jni refs
- [IMPALA-1794](#) - Fix infinite loop opening/closing file w/ invalid metadata
- [IMPALA-1674](#) - Fix serious memory leak in TSaslTransport
- [IMPALA-1668](#) - Fix leak of transport objects in TSaslServerTransport::Factory
- [IMPALA-1556](#) - TSaslTransport.read() should return available data before next frame
- [IMPALA-1565](#) - Python sasl client transport perf issue
- [IMPALA-1556](#) - Sasl transport should be wrapped with buffered transport
- [IMPALA-1442](#) - Better fix for non-buffered SASL transports The Thrift SASL implementation relies on the

Known Issues Fixed in CDH 5.1.4

The following topics describe known issues fixed in CDH 5.1.4. See [What's New in CDH 5.1.4](#) on page 16 for a list of the most important upstream problems fixed in this release.

HTTPS does not work on the HTTPS configured port

If you enable HTTPS (SSL) for YARN services, these services (including ResourceManager, NodeManager, and Job History Server) will not continue to use non-secure HTTP, but HTTPS does not work on the HTTPS configured port.

Bug: [YARN-1553](#)

Severity: High

Workaround: None.

Known Issues Fixed in CDH 5.1.3

The following topics describe known issues fixed in CDH 5.1.3. See [New Features in CDH 5](#) on page 5 for a list of the most important upstream problems fixed in this release.

Apache Hadoop

— *The default setting of `dfs.client.block.write.replace-datanode-on-failure.policy` can cause an unrecoverable error in small clusters*

The default setting of `dfs.client.block.write.replace-datanode-on-failure.policy` (DEFAULT) can cause an unrecoverable error in a small cluster during HBase rolling restart.

Bug: [HDFS-4257](#)

Severity: Medium

Workaround: Set `dfs.client.block.write.replace-datanode-on-failure.policy` to NEVER for 1- 2- or 3-node clusters, and leave it as DEFAULT for all other clusters. Leave `dfs.client.block.write.replace-datanode-on-failure.enable` set to true.

Known Issues Fixed in CDH 5.1.2

The following topics describe known issues fixed in CDH 5.1.2. See [What's New in CDH 5.1.2](#) on page 18 for a list of the most important upstream problems fixed in this release.

Apache Hadoop

— *Jobs can hang on NodeManager decommission owing to a race condition when continuous scheduling is enabled.*

Bug: [YARN-2273](#)

Severity: Low

Workaround: Disable continuous scheduling by setting `yarn.scheduler.fair.continuous-scheduling-enabled` to `false`

Apache HBase

— *Sending a large amount of invalid data to the Thrift service can cause it to crash*

Bug: [HBASE-11052](#).

Severity: High

Workaround: None. This is a longstanding problem, not a new issue in CDH 5.1.

— *The metric `ageOfLastShippedOp` never decreases*

This can cause it to appear as though the cluster is in an inconsistent state even when there is no problem.

Bug: [HBASE-11143](#).

Severity: High

Workaround: None.

Known Issues Fixed in CDH 5.1.0

Apache Hadoop

HDFS

— The same DataNodes may appear in the NameNode web UI in both the live and dead node lists

Bug: [HDFS-6180](#)

Severity: Low

Workaround: None

MapReduce

YARN Fair Scheduler's Cluster Utilization Threshold check is broken

Bug: [YARN-1640](#)

Severity: Medium

Workaround: Set the `yarn.scheduler.fair.preemption.cluster-utilization-threshold` property in `yarn-site.xml` to `-1`.

ResourceManager High Availability with manual failover does not work on secure clusters

Bug: [YARN-2155](#)

Severity: Medium

Workaround: Enable automatic failover; this requires ZooKeeper.

Apache HBase

— *MapReduce over HBase Snapshot bypasses HBase-level security*

The MapReduce over HBase Snapshot bypasses HBase-level security completely since the files are read from the HDFS directly. The user who is running the scan/job has to have read permissions to the data and snapshot files.

Bug: [HBASE-8369](#)

Severity: Medium

Workaround: MapReduce users must be trusted to process/view all data in HBase.

— *HBase snapshots now saved to the /<hbase>/.hbase-snapshot directory*

HBase snapshots are now saved to the /<hbase>/.hbase-snapshot directory instead of the /.snapshot directory. This was a conflict introduced by the HDFS snapshot feature in Hadoop 2.2/CDH 5 HDFS.

Bug: [HBASE-8352](#)

Severity: High

Workaround: This should be handled in the upgrade process.

Hue

— *Oozie jobs don't support ResourceManager HA in YARN*

If the ResourceManager fails, the workflow will fail.

Bug: None

Severity: Medium

Workaround: None

Apache Oozie

— *Oozie HA does not work properly with HCatalog integration or SLA notifications*

This issue appears when you are using HCatalog as a data dependency in a coordinator; using HCatalog from an action (for example, Pig) works correctly.

Bug: [OOZIE-1492](#)

Severity: Medium

Workaround: None

Known Issues Fixed in CDH 5.0.6

Upstream Issues Fixed

The following upstream issues are fixed in CDH 5.0.6:

- [HDFS-7960](#) - The full block report should prune zombie storages even if they're not empty
- [HDFS-7278](#) - Add a command that allows sysadmins to manually trigger full block reports from a DN
- [HDFS-6831](#) - Inconsistency between `hdfs dfsadmin` and `hdfs dfsadmin -help`
- [HDFS-7596](#) - NameNode should prune dead storages from `storageMap`
- [HDFS-7208](#) - NN doesn't schedule replication when a DN storage fails
- [HDFS-7575](#) - Upgrade should generate a unique storage ID for each volume
- [YARN-570](#) - Time strings are formatted in different timezone
- [YARN-2251](#) - Avoid negative elapsed time in JHS/MRAM web UI and services
- [HIVE-8874](#) - Error Accessing HBase from Hive via Oozie on Kerberos 5.0.1 cluster
- [SOLR-6268](#) - `HdfsUpdateLog` has a race condition that can expose a closed HDFS FileSystem instance and should close its FileSystem instance if either inherited close method is called.
- [SOLR-6393](#) - Improve transaction log replay speed on HDFS.
- [SOLR-6403](#) - `TransactionLog` replay status logging.

Known Issues Fixed in CDH 5.0.5

See [What's New in CDH 5.0.5](#) on page 22 for the most important upstream problems fixed in this release.

Known Issues Fixed in CDH 5.0.4

See [What's New in CDH 5.0.4](#) on page 23 for a list of the most important upstream problems fixed in this release.

Known Issues Fixed in CDH 5.0.3

The following topics describe known issues fixed in CDH 5.0.3. See [What's New in CDH 5.0.3](#) on page 23 for a list of the most important upstream problems fixed in this release.

Apache Hadoop

MapReduce

YARN Fair Scheduler's Cluster Utilization Threshold check is broken

Bug: [YARN-2155](#)

Severity: Medium

Workaround: Set the `yarn.scheduler.fair.preemption.cluster-utilization-threshold` property in `yarn-site.xml` to `-1`.

Apache Oozie

— *When Oozie is configured to use MRv1 and SSL, YARN / MRv2 libraries are erroneously included in the classpath instead*

This problem causes much of the configured Oozie functionality to be unusable.

Bug: None

Severity: Medium

Workaround: Use a different configuration (non-SSL or YARN), if possible.

Known Issues Fixed in CDH 5.0.2

The following topics describe known issues fixed in CDH 5.0.2. See [What's New in CDH 5.0.2](#) on page 24 for a list of the most important upstream problems fixed in this release.

Apache Hadoop

CDH 5 clients running releases 5.0.1 and earlier cannot use WebHDFS to connect to a CDH 4 cluster

For example, a `hadoop fs -ls webhdfs` command run from the CDH 5 client to the CDH 4 cluster produces an error such as the following:

```
Found 21 items
ls: Invalid value for webhdfs parameter "op": No enum const class
org.apache.hadoop.hdfs.web.resources.GetOpParam.Op.GETACLSTATUS
```

Bug: [HDFS-6326](#)

Severity: Medium

Workaround: None; note that this is fixed as of CDH 5.0.2.

Apache HBase

— *Endless Compaction Loop*

If an empty HFile whose max timestamp is past its TTL (time-to-live) is selected for compaction, it is compacted into another empty HFile, which is selected for compaction, creating an endless compaction loop.

Bug: [HBASE-10371](#)

Severity: Medium

Workaround: None

Known Issues Fixed in CDH 5.0.1

Apache Hadoop HDFS

- NameNode LeaseManager may crash

Bug: [HDFS-6148/HDFS-6094](#)

Severity: High

Workaround: Restart the NameNode.

- Some group mapping providers can cause the NameNode to crash

In certain environments, some group mapping providers can cause the NameNode to segfault and crash.

Bug: [HADOOP-10442](#)

Severity: High

Workaround: Configure either `ShellBasedUnixGroupsMapping` in Hadoop or configure SSSD in the operating system on the NameNode.

Apache Hive

— *CREATE TABLE AS SELECT (CTAS) does not work with Parquet files*

Since CTAS does not work with Parquet files, the following example will return null values.

```
CREATE TABLE test_data(column1 string);
LOAD DATA LOCAL INPATH './data.txt' OVERWRITE INTO TABLE test_data;

CREATE TABLE parquet_test
ROW FORMAT SERDE 'parquet.hive.serde.ParquetHiveSerDe'
STORED AS
  INPUTFORMAT 'parquet.hive.DeprecatedParquetInputFormat'
  OUTPUTFORMAT 'parquet.hive.DeprecatedParquetOutputFormat'
AS
  SELECT column1 FROM test_data;

SELECT * FROM parquet_test;
SELECT column1 FROM parquet_test;
```

Bug: [HIVE-6375](#)

Severity: Medium

Workaround: A workaround for this is to follow up a CREATE TABLE query with an INSERT OVERWRITE TABLE SELECT * as in the example below.

```
CREATE TABLE parquet_test (column1 string)
ROW FORMAT SERDE 'parquet.hive.serde.ParquetHiveSerDe'
STORED AS
  INPUTFORMAT 'parquet.hive.DeprecatedParquetInputFormat'
  OUTPUTFORMAT 'parquet.hive.DeprecatedParquetOutputFormat';
INSERT OVERWRITE TABLE parquet_test SELECT * from test_data;
```

Apache Oozie

— *The oozie-workflow-0.4.5 schema has been removed*

Workflows using schema 0.4.5 will no longer be accepted by Oozie because this schema definition version has been removed.

Bug: [OOZIE-1768](#)

Severity: Low

Workaround: Use schema 0.5. It's backwards compatible with 0.4.5, so updating the workflow is as simple as changing the schema version number.

Known Issues Fixed in CDH 5.0.0

Apache Flume

— *AsyncHBaseSink does not work in CDH 5 Beta 1 and CDH 5 Beta 2*

Bug: None

Severity: High

Workaround: Use the HBASE sink (`org.apache.flume.sink.hbase.HBaseSink`) to write to HBase in CDH 5 Beta releases.

Apache Hadoop

HDFS

— DataNode can consume 100 percent of one CPU

A narrow race condition can cause one of the threads in the DataNode process to get stuck in a tight loop and consume 100 percent of one CPU.

Bug: [HDFS-5922](#)

Severity: Low

Workaround: Restart the DataNode process.

— HDFS NFS gateway does not work with Kerberos-enabled clusters

Bug: [HDFS-5898](#)

Severity: Medium

Workaround: None.

— Cannot browse filesystem via NameNode Web UI if any directory has the sticky bit set

When listing any directory which contains an entry that has the sticky bit permission set, for example `/tmp` is often set this way, nothing will appear where the list of files or directories should be.

Bug: [HDFS-5921](#)

Severity: Low

Workaround: Use the Hue File Browser.

— Appending to a file that has been snapshotted previously will append to the snapshotted file as well

If you append content to a file that exists in snapshot, the file in snapshot will have the same content appended to it, invalidating the original snapshot.

Bug: See also [HDFS-5343](#)

Severity: High

Workaround: None

MapReduce

— In MRv2 (YARN), the JobHistory Server has no information about a job if the ApplicationMasters fails while the job is running

Bug: None

Severity: Medium

Workaround: None.

Apache HBase

— *An empty rowkey is treated as the first row of a table*

An empty rowkey is allowed in HBase, but it was treated as the first row of the table, even if it was not in fact the first row. Also, multiple rows with empty rowkeys caused issues.

Bug: [HBASE-3170](#)

Severity: High

Workaround: Do not use empty rowkeys.

Apache Hive

— *Hive queries that combine multiple splits and query large tables fail on YARN*

Hive queries that scan large tables, or perform map side joins may fail with the following exception when the query is run using YARN:

```
java.io.IOException: Max block location exceeded for split:
InputFormatClass: org.apache.hadoop.mapred.TextInputFormat
splitSize: 21 maxSize: 10
at
org.apache.hadoop.mapreduce.split.JobSplitWriter.writeOldSplits(JobSplitWriter.java:162)
at
org.apache.hadoop.mapreduce.split.JobSplitWriter.createSplitFiles(JobSplitWriter.java:87)
at org.apache.hadoop.mapreduce.JobSubmitter.writeOldSplits(JobSubmitter.java:540)
at org.apache.hadoop.mapreduce.JobSubmitter.writeSplits(JobSubmitter.java:510)
at org.apache.hadoop.mapreduce.JobSubmitter.submitJobInternal(JobSubmitter.java:392)
at org.apache.hadoop.mapreduce.Job$10.run(Job.java:1268)
at org.apache.hadoop.mapreduce.Job$10.run(Job.java:1265)
at java.security.AccessController.doPrivileged(Native Method)
at javax.security.auth.Subject.doAs(Subject.java:415)
at org.apache.hadoop.security.UserGroupInformation.doAs(UserGroupInformation.java:1491)
at org.apache.hadoop.mapreduce.Job.submit(Job.java:1265)
at org.apache.hadoop.mapred.JobClient$1.run(JobClient.java:562)
at org.apache.hadoop.mapred.JobClient$1.run(JobClient.java:557)
at java.security.AccessController.doPrivileged(Native Method)
at javax.security.auth.Subject.doAs(Subject.java:415)
at org.apache.hadoop.security.UserGroupInformation.doAs(UserGroupInformation.java:1491)
at org.apache.hadoop.mapred.JobClient.submitJobInternal(JobClient.java:557)
at org.apache.hadoop.mapred.JobClient.submitJob(JobClient.java:548)
at org.apache.hadoop.hive.ql.exec.mr.ExecDriver.execute(ExecDriver.java:425)
at org.apache.hadoop.hive.ql.exec.mr.MapRedTask.execute(MapRedTask.java:136)
at org.apache.hadoop.hive.ql.exec.Task.executeTask(Task.java:151)
at org.apache.hadoop.hive.ql.exec.TaskRunner.runSequential(TaskRunner.java:65)
```

Bug: [MAPREDUCE-5186](#)

Severity: High

Workaround: Set `mapreduce.job.max.split.locations` to a high value such as 100.

— *Files in Avro tables no longer have .avro extension*

As of CDH 4.3.0 Hive no longer creates files in Avro tables with the `.avro` extension by default. This does not cause any problems in Hive, but could affect downstream components such as Pig, MapReduce, or Sqoop 1 that expect files with the `.avro` extension.

Bug: None

Severity: Low

Workaround: Manually set the extension to `.avro` before using a query that inserts data into your Avro table. Use the following `set` statement:

```
set hive.output.file.extension=".avro";
```

Apache Oozie

— *Oozie does not work seamlessly with ResourceManager HA*

Oozie workflows are not recovered on ResourceManager failover when ResourceManager HA is enabled. Further, users can not specify the `clusterId` for JobTracker to work against either ResourceManager.

Bug: None

Severity: Medium

Workaround: On non-secure clusters, users are required to specify either of the ResourceManagers' `host:port`. For secure clusters, users are required to specify the Active ResourceManager's `host:port`.

— *When using Oozie HA with security enabled, some znodes have world ACLs*

Oozie High Availability with security enabled will still work, but a malicious user or program can alter znodes used by Oozie for locking, possibly causing Oozie to be unable to finish processing certain jobs.

Bug: [OOZIE-1608](#)

Severity: Medium

Workaround: None

— *Oozie and Sqoop 2 may need additional configuration to work with YARN*

In CDH 5, MRv2 (YARN) MapReduce 2.0 is recommended over the Hadoop 0.20-based MRv1. The default configuration may not reflect this in Oozie and Sqoop 2 in CDH 5 Beta 2, however, unless you are using Cloudera Manager.

Bug: None

Severity: Low

Workaround: Check the value of `CATALINA_BASE` in `/etc/oozie/conf/oozie-env.sh` (if you are running an Oozie server) and `/etc/default/sqoop2-server` (if you are using a Sqoop 2 server). You should also ensure that `CATALINA_BASE` is correctly set in your environment if you are invoking `/usr/bin/sqoop2-server` directly instead of using the service `init` scripts. For Oozie, `CATALINA_BASE` should be set to `/usr/lib/oozie/oozie-server` for YARN, or `/usr/lib/oozie/oozie-server-0.20` for MRv1. For Sqoop 2, `CATALINA_BASE` should be set to `/usr/lib/sqoop2/sqoop-server` for YARN, or `/usr/lib/sqoop2/sqoop-server-0.20` on MRv1.

[Cloudera Search](#)

— *Creating cores using the web UI with default values causes the system to become unresponsive*

You can use the Solr Server web UI to create new cores. If you click **Create Core** without making any changes to the default attributes, the server may become unresponsive. Checking the log for the server shows a repeated error that begins:

```
ERROR org.apache.solr.cloud.Overseer: Exception in Overseer main queue loop
java.lang.IllegalArgumentException: Path must not end with / character
```

Bug: [Solr-5813](#)

Severity: Medium

Workaround: To avoid this issue, do not create cores without first updating values for the new core in the web UI. For example, you might enter a new name for the core to be created.

If you created a core with default settings and are seeing this error, you can address the problem by finding which node is having problems and removing that node. Find the problematic node by using a tool that can inspect ZooKeeper, such as the Solr Admin UI. Using such a tool, examine items in the ZooKeeper queue, reviewing the properties for the item. The problematic node will have an item in its queue with the property `collection=""`.

Remove the node with the item with the `collection=""` property using a ZooKeeper management tool. For example, you can remove nodes using the ZooKeeper command line tool or recent versions of HUE.

[Known Issues Fixed in CDH 5 Beta 2](#)

[Apache Hadoop](#)

[MapReduce](#)

— [ResourceManager High Availability does not work on secure clusters](#)

If JobTrackers in an High Availability configuration are shut down, migrated to new hosts, then restarted, no JobTracker becomes active. The logs show a `Mismatched address` exception.

Bug: None

Severity: High

Workaround: None.

— Default port conflicts

By default, the Shuffle Handler (which runs inside the YARN NodeManager), the [REST server](#), and many third-party applications, all use port 8080. This will result in conflicts if you deploy more than one of them without reconfiguring the default port.

Bug: None

Severity: Medium

Workaround: Make sure at most one service uses port 8080. To reconfigure the REST server, follow [these instructions](#). To change the default port for the Shuffle Handler, set the value of `mapreduce.shuffle.port` in `mapred-site.xml` to an unused port.

— JobTracker memory leak

The JobTracker has a memory leak caused by subtleties in the way `UserGroupInformation` interacts with the file-system cache. The number of cached file system objects can grow without bound.

Bug: [MAPREDUCE-5508](#)

Severity: Medium

Workaround: Set `keep.failed.task.files` to `true`, which will sidestep the memory leak but require job staging directories to be cleaned out manually.

[Hue](#)

— *Running a Hive Beeswax metastore on the same host as the Hue server will result in Simple Authentication and Security Layer (SASL) authentication failures on a Kerberos-enabled cluster*

Bug: None

Severity: Medium

Workaround: The simple workaround is to run the metastore server remotely on a different host and make sure all Hive and Hue configurations properly refer to it. A more complex workaround is to adjust network configurations to ensure that reverse DNS properly resolves the host's address to its fully qualified-domain name (FQDN) rather than `localhost`.

— *The Pig shell does not work when NameNode uses a wildcard address*

The Pig shell does not work from Hue if you use a wildcard for the NameNode's RPC or HTTP bind address. For example, `dfs.namenode.http-address` must be a real, routable address and port, not `0.0.0.0.<port>`.

Bug: [HUE-1060](#)

Severity: Medium

Workaround: Use a real, routable address and port, not `0.0.0.0.<port>`, for the NameNode; or use the Pig application directly, rather than from Hue.

[Apache Sqoop](#)

— *Oozie and Sqoop 2 may need additional configuration to work with YARN*

In CDH 5, MRv2 (YARN) MapReduce 2.0 is recommended over the Hadoop 0.20-based MRv1. The default configuration may not reflect this in Oozie and Sqoop 2 in CDH 5 Beta 2, however, unless you are using Cloudera Manager.

Bug: None

Severity: Low

Workaround: Check the value of `CATALINA_BASE` in `/etc/oozie/conf/oozie-env.sh` (if you are running an Oozie server) and `/etc/default/sqoop2-server` (if you are using a Sqoop 2 server). You should also ensure that `CATALINA_BASE` is correctly set in your environment if you are invoking `/usr/bin/sqoop2-server` directly instead of using the service `init` scripts. For Oozie, `CATALINA_BASE` should be set to `/usr/lib/oozie/oozie-server` for YARN, or `/usr/lib/oozie/oozie-server-0.20` for MRv1. For Sqoop 2, `CATALINA_BASE` should be set to `/usr/lib/sqoop2/sqoop-server` for YARN, or `/usr/lib/sqoop2/sqoop-server-0.20` on MRv1.

Apache Sentry (incubating)

Sentry allows unauthorized access to a directory whose name includes the scratch directory name as a prefix

As an example, if the scratch directory path is `/tmp/hive`, and you create a directory `/tmp/hive-data`, Sentry allows unauthorized read/write access to `/tmp/hive-data`.

Bug: None

Severity: Medium

Workaround: For external tables or data export location, do not use a pathname that includes the scratch directory name as a prefix. For example, if the scratch directory is `/tmp/hive`, do not locate external tables or exported data in `/tmp/hive-data` or any directory whose path uses `"/tmp/hive-"` as a prefix.

Apache Oozie

— Oozie Hive action against HiveServer2 fails on a secure cluster

Severity: Medium

Workaround: None

Cloudera Impala Fixed Issues

The following sections describe the major issues fixed in each Impala release.

For known issues that are currently unresolved, see [Cloudera Impala Known Issues](#) on page 86.

Issues Fixed in the 2.1.4 Release / CDH 5.3.4

This section lists the most significant Impala issues fixed in Impala 2.1.4 for CDH 5.3.4. Because CDH 5.3.5 does not include any code changes for Impala, Impala 2.1.4 is included with both CDH 5.3.4 and 5.3.5.

For the full list of Impala fixed issues in Impala 2.1.4 for CDH 5.3.4, see [Known Issues Fixed in CDH 5.3.4](#) on page 101.

- **Note:** This Impala maintenance release is only available as part of CDH 5, not under CDH 4.

Crash: `impala::TupleIsNullPredicate::Prepare`

When expressions that tested for `NULL` were used in combination with analytic functions, an error could occur. (The original crash issue was fixed by an earlier patch.)

Bug: [IMPALA-1519](#)

Severity: High

Expand parsing of decimals to include scientific notation

`DECIMAL` literals could include `e` scientific notation. For example, now `CAST(1e3 AS DECIMAL(5,3))` is a valid expression. Formerly it returned `NULL`. Some scientific expressions might have worked before in `DECIMAL` context, but only when the scale was 0.

Bug: [IMPALA-1952](#)

Severity: High

INSERT/CTAS evaluates and applies constant predicates.

An `INSERT OVERWRITE` statement would write new data, even if a constant clause such as `WHERE 1 = 0` should have prevented it from writing any rows.

Bug: [IMPALA-1860](#)

Severity: High

Assign predicates below analytic functions with a compatible partition by clause

If the `PARTITION BY` clause in an analytic function refers to partition key columns in a partitioned table, now Impala can perform partition pruning during the analytic query.

Bug: [IMPALA-1900](#)

Severity: High

FIRST_VALUE may produce incorrect results with preceding windows

A query using the `FIRST_VALUE` analytic function and a window defined with the `PRECEDING` keyword could produce wrong results.

Bug: [IMPALA-1888](#)

Severity: High

FIRST_VALUE rewrite fn type might not match slot type

A query referencing a `DECIMAL` column with the `FIRST_VALUE` analytic function could encounter an error.

Bug: [IMPALA-1559](#)

Severity: High

AnalyticEvalNode cannot handle partition/order by exprs with NaN

A query using an analytic function could encounter an error if the evaluation of an analytic `ORDER BY` or `PARTITION` expression resulted in a NaN value, for example if the `ORDER BY` or `PARTITION` contained a division operation where both operands were zero.

Bug: [IMPALA-1808](#)

Severity: High

AnalyticEvalNode not properly handling nullable tuples

An analytic function containing only an `OVER` clause could encounter an error if another part of the query (such as a join) produced all-NULL tuples.

Bug: [IMPALA-1562](#)

Severity: High

Issues Fixed in the 2.1.3 Release / CDH 5.3.3

This section lists the most significant issues fixed in Impala 2.1.3.

For the full list of fixed issues in Impala 2.1.3, see [Known Issues Fixed in CDH 5.3.3](#) on page 103.

- **Note:** Impala 2.1.3 is available as part of CDH 5.3.3, not under CDH 4.

Add compatibility flag for Hive-Parquet-Timestamps

When Hive writes `TIMESTAMP` values, it represents them in the local time zone of the server. Impala expects `TIMESTAMP` values to always be in the UTC time zone, possibly leading to inconsistent results depending on which component created the data files. This patch introduces a new startup flag,

`-convert_legacy_hive_parquet_utc_timestamps`, for the `impalad` daemon. Specify `-convert_legacy_hive_parquet_utc_timestamps=true` to make Impala recognize Parquet data files written

by Hive and automatically adjust `TIMESTAMP` values read from those files into the UTC time zone for compatibility with other Impala `TIMESTAMP` processing.

Bug: [IMPALA-1658](#)

Severity: High

Use `snprintf()` instead of `lexical_cast()` in float-to-string casts

Converting a floating-point value to a `STRING` could be slower than necessary.

Bug: [IMPALA-1738](#)

Severity: High

Fix partition spilling cleanup when new stream OOMs

Certain calls to aggregate functions with `STRING` arguments could encounter a serious error when the system ran low on memory and attempted to activate the spill-to-disk mechanism. The error message referenced the function `impala::AggregateFunctions::StringValGetValue`.

Bug: [IMPALA-1865](#)

Severity: High

Impala's ACLs check do not consider all group ACLs, only checked first one.

If the HDFS user ID associated with the `impalad` process had read or write access in HDFS based on group membership, Impala statements could still fail with HDFS permission errors if that group was not the first listed group for that user ID.

Bug: [IMPALA-1805](#)

Severity: High

Fix infinite loop opening or closing file with invalid metadata

Truncating a file in HDFS, after Impala had cached the file metadata, could produce a hang when Impala queried a table containing that file.

Bug: [IMPALA-1794](#)

Severity: High

external-data-source-executor leaking global jni refs

Successive calls to the data source API could result in excessive memory consumption, with memory allocated but never freed.

Bug: [IMPALA-1801](#)

Severity: High

Spurious stale block locality messages

Impala could issue messages stating the block locality metadata was stale, when the metadata was actually fine. The internal "remote bytes read" counter was not being reset properly. This issue did not cause an actual slowdown in query execution, but the spurious error could result in unnecessary debugging work and unnecessary use of the `INVALIDATE METADATA` statement.

Bug: [IMPALA-1712](#)

Severity: High

Issues Fixed in the 2.1.2 Release / CDH 5.3.2

This section lists the most significant issues fixed in Impala 2.1.2.

For the full list of fixed issues in Impala 2.1.2, see [this report in the JIRA system](#).

- **Note:** Impala 2.1.2 is available as part of CDH 5.3.2, not under CDH 4.

Impala incorrectly handles double numbers with more than 19 significant decimal digits

When a floating-point value was read from a text file and interpreted as a `FLOAT` or `DOUBLE` value, it could be incorrectly interpreted if it included more than 19 significant digits.

Bug: [IMPALA-1622](#)

Severity: High

unix_timestamp() does not return correct time

The `unix_timestamp()` function could return an incorrect value (a constant value of 1).

Bug: [IMPALA-1623](#)

Severity: High

Row Count Mismatch: Partition pruning with NULL

A query against a partitioned table could return incorrect results if the `WHERE` clause compared the partition key to `NULL` using operators such as `=` or `!=`.

Bug: [IMPALA-1535](#)

Severity: High

Fetch column stats in bulk using new (Hive .13) HMS APIs

The performance of the `COMPUTE STATS` statement and queries was improved, particularly for wide tables.

Bug: [IMPALA-1120](#)

Severity: High

Issues Fixed in the 2.1.1 Release / CDH 5.3.1

This section lists the most significant issues fixed in Impala 2.1.1.

For the full list of fixed issues in Impala 2.1.1, see [this report in the JIRA system](#).

IMPALA-1556 causes memory leak with secure connections

`impalad` daemons could experience a memory leak on clusters using Kerberos authentication, with memory usage growing as more data is transferred across the secure channel, either to the client program or between Impala nodes. The same issue affected LDAP-secured clusters to a lesser degree, because the LDAP security only covers data transferred back to client programs.

Bug: <https://issues.cloudera.org/browse/IMPALA-1674> IMPALA-1674

Severity: High

TSaslServerTransport::Factory::getTransport() leaks transport map entries

`impalad` daemons in clusters secured by Kerberos or LDAP could experience a slight memory leak on each connection. The accumulation of unreleased memory could cause problems on long-running clusters.

Bug: [IMPALA-1668](#)

Severity: High

Issues Fixed in the 2.1.0 Release / CDH 5.3.0

This section lists the most significant issues fixed in Impala 2.1.0.

For the full list of fixed issues in Impala 2.1.0, see [this report in the JIRA system](#).

Kerberos fetches 3x slower

Transferring large result sets back to the client application on Kerberos

Bug: [IMPALA-1455](#)

Severity: High

Compressed file needs to be hold on entirely in Memory

Queries on gzipped text files required holding the entire data file and its uncompressed representation in memory at the same time. `SELECT` and `COMPUTE STATS` statements could fail or perform inefficiently as a result. The fix enables streaming reads for gzipped text, so that the data is uncompressed as it is read.

Bug: [IMPALA-1556](#)

Severity: High

Cannot read hbase metadata with NullPointerException: null

Impala might not be able to access HBase tables, depending on the associated levels of Impala and HBase on the system.

Bug: [IMPALA-1611](#)

Severity: High

Serious errors / crashes

Improved code coverage in Impala testing uncovered a number of potentially serious errors that could occur with specific query syntax. These errors are resolved in Impala 2.1.

Bug: [IMPALA-1553](#) , [IMPALA-1528](#) , [IMPALA-1526](#) , [IMPALA-1524](#) , [IMPALA-1508](#) , [IMPALA-1493](#) , [IMPALA-1501](#) , [IMPALA-1483](#)

Severity: High

Issues Fixed in the 2.0.3 Release / CDH 5.2.4

This section lists the most significant issues fixed in Impala 2.0.3.

For the full list of fixed issues in Impala 2.0.3, see [this report in the JIRA system](#).

- **Note:** Impala 2.0.3 is available as part of CDH 5.2.4, not under CDH 4.

Anti join could produce incorrect results when spilling

An anti-join query (or a `NOT EXISTS` operation that was rewritten internally into an anti-join) could produce incorrect results if Impala reached its memory limit, causing the query to write temporary results to disk.

Bug: [IMPALA-1471](#)

Severity: High

Row Count Mismatch: Partition pruning with NULL

A query against a partitioned table could return incorrect results if the `WHERE` clause compared the partition key to `NULL` using operators such as `=` or `!=`.

Bug: [IMPALA-1535](#)

Severity: High

Fetch column stats in bulk using new (Hive .13) HMS APIs

The performance of the `COMPUTE STATS` statement and queries was improved, particularly for wide tables.

Bug: [IMPALA-1120](#)

Severity: High

Issues Fixed in the 2.0.2 Release / CDH 5.2.3

This section lists the most significant issues fixed in Impala 2.0.2.

For the full list of fixed issues in Impala 2.0.2, see [this report in the JIRA system](#).

- **Note:** Impala 2.0.2 is available as part of CDH 5.2.3, not under CDH 4.

GROUP BY on STRING column produces inconsistent results

Some operations in queries submitted through Hue or other HiveServer2 clients could produce inconsistent results.

Bug: [IMPALA-1453](#)

Severity: High

Fix leaked file descriptor and excessive file descriptor use

Impala could encounter an error from running out of file descriptors. The fix reduces the amount of time file descriptors are kept open, and avoids leaking file descriptors when read operations encounter errors.

Severity: High

unix_timestamp() does not return correct time

The `unix_timestamp()` function could return a constant value 1 instead of a representation of the time.

Bug: [IMPALA-1623](#)

Severity: High

Impala should randomly select cached replica

To avoid putting too heavy a load on any one node, Impala now randomizes which scan node processes each HDFS data block rather than choosing the first cached block replica.

Bug: [IMPALA-1586](#)

Severity: High

Impala does not always give short name to Llama.

In clusters secured by Kerberos or LDAP, a discrepancy in internal transmission of user names could cause a communication error with Llama.

Bug: [IMPALA-1606](#)

Severity: High

accept unmangled native UDF symbols

The `CREATE FUNCTION` statement could report that it could not find a function entry point within the `.so` file for a UDF written in C++, even if the corresponding function was present.

Bug: [IMPALA-1475](#)

Severity: High

Issues Fixed in the 2.0.1 Release / CDH 5.2.1

This section lists the most significant issues fixed in Impala 2.0.1.

For the full list of fixed issues in Impala 2.0.1, see [this report in the JIRA system](#).

Queries fail with metastore exception after upgrade and compute stats

After running the `COMPUTE STATS` statement on an Impala table, subsequent queries on that table could fail with the exception message `Failed to load metadata for table: default.stats_test`.

Bug: <https://issues.cloudera.org/browse/IMPALA-1416> IMPALA-1416

Severity: High

Workaround: Upgrading to CDH 5.2.1, or another level of CDH that includes the fix for HIVE-8627, prevents the problem from affecting future `COMPUTE STATS` statements. On affected levels of CDH, or for Impala tables that have become inaccessible, the workaround is to disable the `hive.metastore.try.direct.sql` setting in the Hive metastore `hive-site.xml` file and issue the `INVALIDATE METADATA` statement for the affected table. You do not need to rerun the `COMPUTE STATS` statement for the table.

Issues Fixed in the 2.0.0 Release / CDH 5.2.0

This section lists the most significant issues fixed in Impala 2.0.0.

For the full list of fixed issues in Impala 2.0.0, see [this report in the JIRA system](#).

Join Hint is dropped when used inside a view

Hints specified within a view query did not take effect when the view was queried, leading to slow performance. As part of this fix, Impala now supports hints embedded within comments.

Bug: [IMPALA-995](#)

Severity: High

WHERE condition ignored in simple query with RIGHT JOIN

Potential wrong results for some types of queries.

Bug: [IMPALA-1101](#)

Severity: High

Query with self joined table may produce incorrect results

Potential wrong results for some types of queries.

Bug: [IMPALA-1102](#)

Severity: High

Incorrect plan after reordering predicates (inner join following outer join)

Potential wrong results for some types of queries.

Bug: [IMPALA-1118](#)

Severity: High

Combining fragments with compatible data partitions can lead to incorrect results due to type incompatibilities (missing casts).

Potential wrong results for some types of queries.

Bug: [IMPALA-1123](#)

Severity: High

Predicate dropped: Inline view + DISTINCT aggregate in outer query

Potential wrong results for some types of queries.

Bug: [IMPALA-1165](#)

Severity: High

Reuse of a column in JOIN predicate may lead to incorrect results

Potential wrong results for some types of queries.

Bug: [IMPALA-1353](#)

Severity: High

Usage of TRUNC with string timestamp reliably crashes node

Serious error for certain combinations of function calls and data types.

Bug: [IMPALA-1105"](#)

Severity: High

Timestamp Cast Returns invalid TIMESTAMP

Serious error for certain combinations of function calls and data types.

Bug: [IMPALA-1109"](#)

Severity: High

IllegalStateException upon JOIN of DECIMAL columns with different precision

DECIMAL columns with different precision could not be compared in join predicates.

Bug: [IMPALA-1121"](#)

Severity: High

Allow creating Avro tables without column definitions. Allow COMPUTE STATS to always work on Impala-created Avro tables.

Hive-created Avro tables with columns specified by a JSON file or literal could produce errors when queried in Impala, and could not be used with the `COMPUTE STATS` statement. Now you can create such tables in Impala to avoid such errors.

Bug: [IMPALA-1104"](#)

Severity: High

Ensure all webserver output is escaped

The Impala debug web UI did not properly encode all output.

Bug: [IMPALA-1133"](#)

Severity: High

Queries with union in inline view have empty resource requests

Certain queries could run without obeying the limits imposed by resource management.

Bug: [IMPALA-1236"](#)

Severity: High

Impala does not employ ACLs when checking path permissions for LOAD and INSERT

Certain `INSERT` and `LOAD DATA` statements could fail unnecessarily, if the target directories in HDFS had restrictive HDFS permissions, but those permissions were overridden by HDFS extended ACLs.

Bug: [IMPALA-1279"](#)

Severity: High

Impala does not map principals to lowercase, affecting Sentry authorisation

In a Kerberos environment, the principal name was not mapped to lowercase, causing issues when a user logged in with an uppercase principal name and Sentry authorization was enabled.

Bug: [IMPALA-1334"](#)

Severity: High

Issues Fixed in the 1.4.4 Release / CDH 5.1.5

For the list of fixed issues, see [Known Issues Fixed in CDH 5.1.5](#) in the *CDH 5 Release Notes*.

- **Note:** Impala 1.4.4 is available as part of CDH 5.1.5, not under CDH 4.

Issues Fixed in the 1.4.3 Release / CDH 5.1.4

Impala 1.4.3 includes fixes to address what is known as the POODLE vulnerability in SSLv3. SSLv3 access is disabled in the Impala debug web UI.

- **Note:** Impala 1.4.3 is available as part of CDH 5.1.4, and under CDH 4.

Issues Fixed in the 1.4.2 Release / CDH 5.1.3

This section lists the most significant issues fixed in Impala 1.4.2.

For the full list of fixed issues in Impala 1.4.2, see [this report in the JIRA system](#).

- **Note:** Impala 1.4.3 is available as part of CDH 5.1.4, and under CDH 4.

Issues Fixed in the 1.4.1 Release / CDH 5.1.2

This section lists the most significant issues fixed in Impala 1.4.1.

For the full list of fixed issues in Impala 1.4.1, see [this report in the JIRA system](#).

- **Note:** Impala 1.4.1 is only available as part of CDH 5.1.2, not under CDH 4.

impalad terminating with Boost exception

Occasionally, a non-trivial query run through Llama could encounter a serious error. The detailed error in the log was:

```
boost::exception_detail::clone_impl  
<boost::exception_detail::error_info_injector<boost::lock_error> >
```

Severity: High

Impalad uses wrong string format when writing logs

Impala log files could contain internal error messages due to a problem formatting certain strings. The messages consisted of a Java call stack starting with:

```
[jni-util.cc:177] java.util.MissingFormatArgumentException: Format specifier 's'
```

Severity: High

Update HS2 client API.

A downlevel version of the HiveServer2 API could cause difficulty retrieving the precision and scale of a DECIMAL value.

Bug: [IMPALA-1107](#)

Severity: High

Impalad catalog updates can fail with error: "IllegalArgumentException: fromKey out of range" at com.cloudera.impala.catalog.CatalogDeltaLog

The error in the title could occur following a DDL statement. This issue was discovered during internal testing and has not been reported in customer environments.

Bug: [IMPALA-1093](#)

Severity: High

"Total" time counter does not capture all the network transmit time

The time for some network operations was not counted in the report of total time for a query, making it difficult to diagnose network-related performance issues.

Bug: [IMPALA-1131](#)**Severity:** High*Impala will crash when reading certain Avro files containing bytes data*

Certain Avro fields for byte data could cause Impala to be unable to read an Avro data file, even if the field was not part of the Impala table definition. With this fix, Impala can now read these Avro data files, although Impala queries cannot refer to the “bytes” fields.

Bug: [IMPALA-1149](#)**Severity:** High*Support specifying a custom AuthorizationProvider in Impala*

The `--authorization_policy_provider_class` option for `impalad` was added back. This option specifies a custom `AuthorizationProvider` class rather than the default `HadoopGroupAuthorizationProvider`. It had been used for internal testing, then removed in Impala 1.4.0, but it was considered useful by some customers.

Bug: [IMPALA-1142](#)**Severity:** High*Issues Fixed in the 1.4.0 Release / CDH 5.1.0*

This section lists the most significant issues fixed in Impala 1.4.0.

For the full list of fixed issues in Impala 1.4.0, see [this report in the JIRA system](#).

Failed DCHECK in disk-io-mgr-reader-context.cc:174

The serious error in the title could occur, with the supplemental message:

```
num_used_buffers_ < 0: #used=-1 during cancellation HDFS cached data
```

The issue was due to the use of HDFS caching with data files accessed by Impala. Support for HDFS caching in Impala was introduced in Impala 1.4.0 for CDH 5.1.0. The fix for this issue was backported to Impala 1.3.x, and is the only change in Impala 1.3.2 for CDH 5.0.4.

Bug: [IMPALA-1019](#)**Severity:** High

Workaround: On CDH 5.0.x, upgrade to CDH 5.0.4 with Impala 1.3.2, where this issue is fixed. In Impala 1.3.0 or 1.3.1 on CDH 5.0.x, do not use HDFS caching for Impala data files in Impala internal or external tables. If some of these data files are cached (for example because they are used by other components that take advantage of HDFS caching), set the query option `DISABLE_CACHED_READS=true`. To set that option for all Impala queries across all sessions, start `impalad` with the `-default_query_options` option and include this setting in the option argument, or on a cluster managed by Cloudera Manager, fill in this option setting on the **Impala Daemon** options page.

Resolution: This issue is fixed in Impala 1.3.2 for CDH 5.0.4. The addition of HDFS caching support in Impala 1.4 means that this issue does not apply to any new level of Impala on CDH 5.

impala-shell only works with ASCII characters

The `impala-shell` interpreter could encounter errors processing SQL statements containing non-ASCII characters.

Bug: [IMPALA-489](#)**Severity:** High*The extended view definition SQL text in Views created by Impala should always have fully-qualified table names*

When a view was accessed while inside a different database, references to tables were not resolved unless the names were fully qualified when the view was created.

Bug: [IMPALA-962](#)

Severity: High

Impala forgets about partitions with non-existent locations

If an `ALTER TABLE` specified a non-existent HDFS location for a partition, afterwards Impala would not be able to access the partition at all.

Bug: [IMPALA-741](#)

Severity: High

CREATE TABLE LIKE fails if source is a view

The `CREATE TABLE LIKE` clause was enhanced to be able to create a table with the same column definitions as a view. The resulting table is a text table unless the `STORED AS` clause is specified, because a view does not have an associated file format to inherit.

Bug: [IMPALA-834](#)

Severity: High

Improve partition pruning time

Operations on tables with many partitions could be slow due to the time to evaluate which partitions were affected. The partition pruning code was speeded up substantially.

Bug: [IMPALA-887](#)

Severity: High

Improve compute stats performance

The performance of the `COMPUTE STATS` statement was improved substantially. The efficiency of its internal operations was improved, and some statistics are no longer gathered because they are not currently used for planning Impala queries.

Bug: [IMPALA-1003](#)

Severity: High

When I run CREATE TABLE new_table LIKE avro_table, the schema does not get mapped properly from an avro schema to a hive schema

After a `CREATE TABLE LIKE` statement using an Avro table as the source, the new table could have incorrect metadata and be inaccessible, depending on how the original Avro table was created.

Bug: [IMPALA-185](#)

Severity: High

Race condition in IoMgr. Blocked ranges enqueued after cancel.

Impala could encounter a serious error after a query was cancelled.

Bug: [IMPALA-1046](#)

Severity: High

Deadlock in scan node

A deadlock condition could make all `impalad` daemons hang, making the cluster unresponsive for Impala queries.

Bug: [IMPALA-1083](#)

Severity: High

Issues Fixed in the 1.3.3 Release / CDH 5.0.5

Impala 1.3.3 includes fixes to address what is known as the POODLE vulnerability in SSLv3. SSLv3 access is disabled in the Impala debug web UI.

- **Note:** Impala 1.3.3 is only available as part of CDH 5.0.5, not under CDH 4.

Issues Fixed in the 1.3.2 Release / CDH 5.0.4

This backported bug fix is the only change between Impala 1.3.1 and Impala 1.3.2.

- **Note:** Impala 1.3.3 is only available as part of CDH 5.0.5, not under CDH 4.

Failed DCHECK in disk-io-mgr-reader-context.cc:174

The serious error in the title could occur, with the supplemental message:

```
num_used_buffers_ < 0: #used=-1 during cancellation HDFS cached data
```

The issue was due to the use of HDFS caching with data files accessed by Impala. Support for HDFS caching in Impala was introduced in Impala 1.4.0 for CDH 5.1.0. The fix for this issue was backported to Impala 1.3.x, and is the only change in Impala 1.3.2 for CDH 5.0.4.

Bug: [IMPALA-1019](#)

Severity: High

Workaround: On CDH 5.0.x, upgrade to CDH 5.0.4 with Impala 1.3.2, where this issue is fixed. In Impala 1.3.0 or 1.3.1 on CDH 5.0.x, do not use HDFS caching for Impala data files in Impala internal or external tables. If some of these data files are cached (for example because they are used by other components that take advantage of HDFS caching), set the query option `DISABLE_CACHED_READS=true`. To set that option for all Impala queries across all sessions, start `impalad` with the `-default_query_options` option and include this setting in the option argument, or on a cluster managed by Cloudera Manager, fill in this option setting on the **Impala Daemon** options page.

Resolution: This issue is fixed in Impala 1.3.2 for CDH 5.0.4. The addition of HDFS caching support in Impala 1.4 means that this issue does not apply to any new level of Impala on CDH 5.

Issues Fixed in the 1.3.1 Release / CDH 5.0.3

This section lists the most significant issues fixed in Impala 1.3.1.

For the full list of fixed issues in Impala 1.3.1, see [this report in the JIRA system](#). Because 1.3.1 is the first 1.3.x release for CDH 4, if you are on CDH 4, also consult [Issues Fixed in the 1.3.0 Release / CDH 5.0.0](#) on page 137.

Impalad crashes when left joining inline view that has aggregate using distinct

Impala could encounter a severe error in a query combining a left outer join with an inline view containing a `COUNT(DISTINCT)` operation.

Bug: [IMPALA-904](#)

Severity: High

Incorrect result with group by query with null value in group by data

If the result of a `GROUP BY` operation is `NULL`, the resulting row might be omitted from the result set. This issue depends on the data values and data types in the table.

Bug: [IMPALA-901](#)

Severity: High

Drop Function does not clear local library cache

When a UDF is dropped through the `DROP FUNCTION` statement, and then the UDF is re-created with a new `.so` library or JAR file, the original version of the UDF is still used when the UDF is called from queries.

Bug: [IMPALA-786](#)

Severity: High

Workaround: Restart the `impalad` daemon on all nodes.

Compute stats doesn't propagate underlying error correctly

If a `COMPUTE STATS` statement encountered an error, the error message is “Query aborted” with no further detail. Common reasons why a `COMPUTE STATS` statement might fail include network errors causing the coordinator node to lose contact with other `impalad` instances, and column names that match Impala [reserved words](#). (Currently, if a column name is an Impala reserved word, `COMPUTE STATS` always returns an error.)

Bug: [IMPALA-762](#)

Severity: High

Inserts should respect changes in partition location

After an `ALTER TABLE` statement that changes the `LOCATION` property of a partition, a subsequent `INSERT` statement would always use a path derived from the base data directory for the table.

Bug: [IMPALA-624](#)

Severity: High

Text data with carriage returns generates wrong results for count()*

A `COUNT(*)` operation could return the wrong result for text tables using nul characters (ASCII value 0) as delimiters.

Bug: [IMPALA-13](#)

Severity: High

Workaround: Impala adds support for ASCII 0 characters as delimiters through the clause `FIELDS TERMINATED BY '\0'`.

IO Mgr should take instance memory limit into account when creating io buffers

Impala could allocate more memory than necessary during certain operations.

Bug: [IMPALA-488](#)

Severity: High

Workaround: Before issuing a `COMPUTE STATS` statement for a Parquet table, reduce the number of threads used in that operation by issuing `SET NUM_SCANNER_THREADS=2` in `impala-shell`. Then issue `UNSET NUM_SCANNER_THREADS` before continuing with queries.

Impala should provide an option for new sub directories to automatically inherit the permissions of the parent directory

When new subdirectories are created underneath a partitioned table by an `INSERT` statement, previously the new subdirectories always used the default HDFS permissions for the `impala` user, which might not be suitable for directories intended to be read and written by other components also.

Bug: [IMPALA-827](#)

Severity: High

Resolution: In Impala 1.3.1 and higher, you can specify the `--insert_inherit_permissions` configuration when starting the `impalad` daemon.

Illegal state exception (or crash) in query with UNION in inline view

Impala could encounter a severe error in a query where the `FROM` list contains an inline view that includes a `UNION`. The exact type of the error varies.

Bug: [IMPALA-888](#)

Severity: High

INSERT column reordering doesn't work with SELECT clause

The ability to specify a subset of columns in an `INSERT` statement, with order different than in the target table, was not working as intended.

Bug: [IMPALA-945](#)

Severity: High

Issues Fixed in the 1.3.0 Release / CDH 5.0.0

This section lists the most significant issues fixed in Impala 1.3.0, primarily issues that could cause wrong results, or cause problems running the `COMPUTE STATS` statement, which is very important for performance and scalability.

For the full list of fixed issues, see [this report in the JIRA system](#).

Inner join after right join may produce wrong results

The automatic join reordering optimization could incorrectly reorder queries with an outer join or semi join followed by an inner join, producing incorrect results.

Bug: [IMPALA-860](#)

Severity: High

Workaround: Including the `STRAIGHT_JOIN` keyword in the query prevented the issue from occurring.

Incorrect results with codegen on multi-column group by with NULLs.

A query with a `GROUP BY` clause referencing multiple columns could introduce incorrect `NULL` values in some columns of the result set. The incorrect `NULL` values could appear in rows where a different `GROUP BY` column actually did return `NULL`.

Bug: [IMPALA-850](#)

Severity: High

Using distinct inside aggregate function may cause incorrect result when using having clause

A query could return incorrect results if it combined an aggregate function call, a `DISTINCT` operator, and a `HAVING` clause, without a `GROUP BY` clause.

Bug: [IMPALA-845](#)

Severity: High

Aggregation on union inside (inline) view not distributed properly.

An aggregation query or a query with `ORDER BY` and `LIMIT` could be executed on a single node in some cases, rather than distributed across the cluster. This issue affected queries whose `FROM` clause referenced an inline view containing a `UNION`.

Bug: [IMPALA-831](#)

Severity: High

Wrong expression may be used in aggregate query if there are multiple similar expressions

If a `GROUP BY` query referenced the same columns multiple times using different operators, result rows could contain multiple copies of the same expression.

Bug: [IMPALA-817](#)

Severity: High

Incorrect results when changing the order of aggregates in the select list with codegen enabled

Referencing the same columns in both a `COUNT()` and a `SUM()` call in the same query, or some other combinations of aggregate function calls, could incorrectly return a result of 0 from one of the aggregate functions. This issue affected references to `TINYINT` and `SMALLINT` columns, but not `INT` or `BIGINT` columns.

Bug: [IMPALA-765](#)

Severity: High

Workaround: Setting the query option `DISABLE_CODEGEN=TRUE` prevented the incorrect results. Switching the order of the function calls could also prevent the issue from occurring.

Union queries give Wrong result in a UNION followed by SIGSEGV in another union

A `UNION` query could produce a wrong result, followed by a serious error for a subsequent `UNION` query.

Bug: [IMPALA-723](#)

Severity: High

String data in MR-produced parquet files may be read incorrectly

Impala could return incorrect string results when reading uncompressed Parquet data files containing multiple row groups. This issue only affected Parquet data files produced by MapReduce jobs.

Bug: [IMPALA-729](#)

Severity: High

Compute stats need to use quotes with identifiers that are Impala keywords

Using a column or table name that conflicted with Impala keywords could prevent running the `COMPUTE STATS` statement for the table.

Bug: [IMPALA-777](#)

Severity: High

COMPUTE STATS child queries do not inherit parent query options.

The `COMPUTE STATS` statement did not use the setting of the `MEM_LIMIT` query option in `impala-shell`, potentially causing problems gathering statistics for wide Parquet tables.

Bug: [IMPALA-903](#)

Severity: High

COMPUTE STATS should update partitions in batches

The `COMPUTE STATS` statement could be slow or encounter a timeout while analyzing a table with many partitions.

Bug: [IMPALA-880](#)

Severity: High

Fail early (in analysis) when COMPUTE STATS is run against Avro table with no columns

If the columns for an Avro table were all defined in the `TBLPROPERTIES` or `SERDEPROPERTIES` clauses, the `COMPUTE STATS` statement would fail after completely analyzing the table, potentially causing a long delay. Although the `COMPUTE STATS` statement still does not work for such tables, now the problem is detected and reported immediately.

Bug: [IMPALA-867](#)

Severity: High

Workaround: Re-create the Avro table with columns defined in SQL style, using the output of `SHOW CREATE TABLE`. (See the JIRA page for detailed steps.)

Issues Fixed in the 1.2.4 Release

This section lists the most significant issues fixed in Impala 1.2.4. For the full list of fixed issues, see [this report in the JIRA system](#).

The Catalog Server exits with an OOM error after a certain number of CREATE statements

A large number of concurrent `CREATE TABLE` statements can cause the `catalogd` process to consume excessive memory, and potentially be killed due to an out-of-memory condition.

Bug: [IMPALA-818](#)

Severity: High

Workaround: Restart the `catalogd` service and re-try the DDL operations that failed.

Catalog Server consumes excessive cpu cycle

A large number of tables and partitions could result in unnecessary CPU overhead during Impala idle time and background operations.

Bug: [IMPALA-821](#)

Severity: High

Resolution: Catalog server processing was optimized in several ways.

Query against Avro table crashes Impala with codegen enabled

A query against a `TIMESTAMP` column in an Avro table could encounter a serious issue.

Bug: [IMPALA-828](#)

Severity: High

Workaround: Set the query option `DISABLE_CODEGEN=TRUE`

Statestore seems to send concurrent heartbeats to the same subscriber leading to repeated "Subscriber 'hostname' is registering with statestore, ignoring update" messages

Impala nodes could produce repeated error messages after recovering from a communication error with the statestore service.

Bug: [IMPALA-809](#)

Severity: High

Join predicate incorrectly ignored

A join query could produce wrong results if multiple equality comparisons between the same tables referred to the same column.

Bug: [IMPALA-805](#)

Severity: High

Query result differing between Impala and Hive

Certain outer join queries could return wrong results. If one of the tables involved in the join was an inline view, some tests from the `WHERE` clauses could be applied to the wrong phase of the query.

Severity: High

ArrayIndexOutOfBoundsException / Invalid query handle when reading large HBase cell

An HBase cell could contain a value larger than 32 KB, leading to a serious error when Impala queries that table. The error could occur even if the applicable row is not part of the result set.

Bug: [IMPALA-715](#)

Severity: High

Workaround: Use smaller values in the HBase table, or exclude the column containing the large value from the result set.

select with distinct and full outer join, impalad coredump

A query involving a `DISTINCT` operator combined with a `FULL OUTER JOIN` could encounter a serious error.

Bug: [IMPALA-735](#)

Severity: High

Workaround: Set the query option `DISABLE_CODEGEN=TRUE`

Impala cannot load tables with more than Short.MAX_VALUE number of partitions

If a table had more than 32,767 partitions, Impala would not recognize the partitions above the 32K limit and query results could be incomplete.

Bug: [IMPALA-749](#)

Severity: High

Various issues with HBase row key specification

Queries against HBase tables could fail with an error if the row key was compared to a function return value rather than a string constant. Also, queries against HBase tables could fail if the `WHERE` clause contained combinations of comparisons that could not possibly match any row key.

Severity: High

Resolution: Queries now return appropriate results when function calls are used in the row key comparison. For queries involving non-existent row keys, such as `WHERE row_key IS NULL` or where the lower bound is greater than the upper bound, the query succeeds and returns an empty result set.

Issues Fixed in the 1.2.3 Release

This release is a fix release that supercedes Impala 1.2.2, with the same features and fixes as 1.2.2 plus one additional fix for compatibility with Parquet files generated outside of Impala by components such as Hive, Pig, or MapReduce.

Impala cannot read Parquet files with multiple row groups

The `parquet-mr` library included with CDH4.5 writes files that are not readable by Impala, due to the presence of multiple row groups. Queries involving these data files might result in a crash or a failure with an error such as "Column chunk should not contain two dictionary pages".

This issue does not occur for Parquet files produced by Impala `INSERT` statements, because Impala only produces files with a single row group.

Bug: [IMPALA-720](#)

Severity: High

Issues Fixed in the 1.2.2 Release

This section lists the most significant issues fixed in Impala 1.2.2. For the full list of fixed issues, see [this report in the JIRA system](#).

Order of table references in FROM clause is critical for optimal performance

Impala does not currently optimize the join order of queries; instead, it joins tables in the order in which they are listed in the `FROM` clause. Queries that contain one or more large tables on the right hand side of joins (either an explicit join expressed as a `JOIN` statement or a join implicit in the list of table references in the `FROM` clause) may run slowly or crash Impala due to out-of-memory errors. For example:

```
SELECT ... FROM small_table JOIN large_table
```

Severity: Medium

Anticipated Resolution: Fixed in Impala 1.2.2.

Workaround: In Impala 1.2.2 and higher, use the `COMPUTE STATS` statement to gather statistics for each table involved in the join query, after data is loaded. Prior to Impala 1.2.2, modify the query, if possible, to join the largest table first. For example:

```
SELECT ... FROM small_table JOIN large_table
```

should be modified to:

```
SELECT ... FROM large_table JOIN small_table
```

Parquet in CDH4.5 writes data files that are sometimes unreadable by Impala

Some Parquet files could be generated by other components that Impala could not read.

Bug: [IMPALA-694](#)

Severity: High

Resolution: The underlying issue is being addressed by a fix in the CDH Parquet libraries. Impala 1.2.2 works around the problem and reads the existing data files.

Deadlock in statestore when unregistering a subscriber and building a topic update

The statestore service could experience an internal error leading to a hang.

Bug: [IMPALA-699](#)

Severity: High

IllegalStateException when doing a union involving a group by

A `UNION` query where one side involved a `GROUP BY` operation could cause a serious error.

Bug: [IMPALA-687](#)

Severity: High

Impala Parquet Writer hit DCHECK in RleEncoder

A serious error could occur when doing an `INSERT` into a Parquet table.

Bug: [IMPALA-689](#)

Severity: High

Hive UDF jars cannot be loaded by the FE

If the JAR file for a Java-based Hive UDF was not in the `CLASSPATH`, the UDF could not be called during a query.

Bug: [IMPALA-695](#)

Severity: High

Issues Fixed in the 1.2.1 Release

This section lists the most significant issues fixed in Impala 1.2.1. For the full list of fixed issues, see [this report in the JIRA system](#).

Scanners use too much memory when reading past scan range

While querying a table with long column values, Impala could over-allocate memory leading to an out-of-memory error. This problem was observed most frequently with tables using uncompressed RCFile or text data files.

Bug: [IMPALA-525](#)

Severity: High

Resolution: Fixed in 1.2.1

Join node consumes memory way beyond mem-limit

A join query could allocate a temporary work area that was larger than needed, leading to an out-of-memory error. The fix makes Impala return unused memory to the system when the memory limit is reached, avoiding unnecessary memory errors.

Bug: [IMPALA-657](#)

Severity: High

Resolution: Fixed in 1.2.1

Excessive memory consumption when query tables with 1k columns (Parquet file)

Impala could encounter an out-of-memory condition setting up work areas for Parquet tables with many columns. The fix reduces the size of the allocated memory when not actually needed to hold table data.

Bug: [IMPALA-652](#)

Severity: High

Resolution: Fixed in 1.2.1

Issues Fixed in the 1.2.0 Beta Release

This section lists the most significant issues fixed in Impala 1.2 (beta). For the full list of fixed issues, see [this report in the JIRA system](#).

Issues Fixed in the 1.1.1 Release

This section lists the most significant issues fixed in Impala 1.1.1. For the full list of fixed issues, see [this report in the JIRA system](#).

Unexpected LLVM Crash When Querying Doubles on CentOS 5.x

Certain queries involving `DOUBLE` columns could fail with a serious error. The fix improves the generation of native machine instructions for certain chipsets.

Bug: [IMPALA-477](#)

Severity: High

"block size is too big" error with Snappy-compressed RCFile containing null

Queries could fail with a "block size is too big" error, due to `NULL` values in RCFile tables using Snappy compression.

Bug: [IMPALA-482](#)

Severity: High

Cannot query RC file for table that has more columns than the data file

Queries could fail if an Impala RCFile table was defined with more columns than in the corresponding RCFile data files.

Bug: [IMPALA-510](#)

Severity: High

Views Sometimes Not Utilizing Partition Pruning

Certain combinations of clauses in a view definition for a partitioned table could result in inefficient performance and incorrect results.

Bug: [IMPALA-495](#)

Severity: High

Update the serde name we write into the metastore for Parquet tables

The SerDes class string written into Parquet data files created by Impala was updated for compatibility with Parquet support in Hive. See [Incompatible Changes Introduced in Cloudera Impala 1.1.1](#) on page 64 for the steps to update older Parquet data files for Hive compatibility.

Bug: [IMPALA-485](#)

Severity: High

Selective queries over large tables produce unnecessary memory consumption

A query returning a small result sets from a large table could tie up memory unnecessarily for the duration of the query.

Bug: [IMPALA-534](#)

Severity: High

Impala stopped to query AVRO tables

Queries against Avro tables could fail depending on whether the Avro schema URL was specified in the `TBLPROPERTIES` or `SERDEPROPERTIES` field. The fix causes Impala to check both fields for the schema URL.

Bug: [IMPALA-538](#)

Severity: High

Impala continues to allocate more memory even though it has exceed its mem-limit

Queries could allocate substantially more memory than specified in the `impalad -mem_limit` startup option. The fix causes more frequent checking of the limit during query execution.

Bug: [IMPALA-520](#)

Severity: High

Issues Fixed in the 1.1.0 Release

This section lists the most significant issues fixed in Impala 1.1. For the full list of fixed issues, see [this report in the JIRA system](#).

10-20% perf regression for most queries across all table formats

This issue is due to a performance tradeoff between systems running many queries concurrently, and systems running a single query. Systems running only a single query could experience lower performance than in early beta releases. Systems running many queries simultaneously should experience higher performance than in the beta releases.

Severity: High

planner fails with "Join requires at least one equality predicate between the two tables" when "from" table order does not match "where" join order

A query could fail if it involved 3 or more tables and the last join table was specified as a subquery.

Bug: [IMPALA-85](#)

Severity: High

Parquet writer uses excessive memory with partitions

`INSERT` statements against partitioned tables using the Parquet format could use excessive amounts of memory as the number of partitions grew large.

Bug: [IMPALA-257](#)

Severity: High

Comments in impala-shell in interactive mode are not handled properly causing syntax errors or wrong results

The `impala-shell` interpreter did not accept comment entered at the command line, making it problematic to copy and paste from scripts or other code examples.

Bug: [IMPALA-192](#)

Severity: Low

Cancelled queries sometimes aren't removed from the inflight query list

The Impala web UI would sometimes display a query as if it were still running, after the query was cancelled.

Bug: [IMPALA-364](#)

Severity: High

Impala's 1.0.1 Shell Broke Python 2.4 Compatibility (AttributeError: 'module' object has no attribute 'field_size_limit')

The `impala-shell` command in Impala 1.0.1 does not work with Python 2.4, which is the default on Red Hat 5.

For the `impala-shell` command in Impala 1.0, the `-o` option (pipe output to a file) does not work with Python 2.4.

Bug: [IMPALA-396](#)

Severity: High

Issues Fixed in the 1.0.1 Release

This section lists the most significant issues fixed in Impala 1.0.1. For the full list of fixed issues, see [this report in the JIRA system](#).

Impala parquet scanner can not read all data files generated by other frameworks

Impala might issue an erroneous error message when processing a Parquet data file produced by a non-Impala Hadoop component.

Bug: [IMPALA-333](#)

Severity: High

Resolution: Fixed

Impala is unable to query RCFile tables which describe fewer columns than the file's header.

If an RCFile table definition had fewer columns than the fields actually in the data files, queries would fail.

Bug: [IMPALA-293](#)

Severity: High

Resolution: Fixed

Impala does not correctly substitute _HOST with hostname in --principal

The `_HOST` placeholder in the `--principal` startup option was not substituted with the correct hostname, potentially leading to a startup error in setups using Kerberos authentication.

Bug: [IMPALA-351](#)

Severity: High

Resolution: Fixed

HBase query missed the last region

A query for an HBase table could omit data from the last region.

Bug: [IMPALA-356](#)

Severity: High

Resolution: Fixed

Hbase region changes are not handled correctly

After a region in an HBase table was split or moved, an Impala query might return incomplete or out-of-date results.

Bug: [IMPALA-300](#)

Severity: High

Resolution: Fixed

Query state for successful create table is EXCEPTION

After a successful CREATE TABLE statement, the corresponding query state would be incorrectly reported as EXCEPTION.

Bug: [IMPALA-349](#)

Severity: High

Resolution: Fixed

Double check release of JNI-allocated byte-strings

Operations involving calls to the Java JNI subsystem (for example, queries on HBase tables) could allocate memory but not release it.

Bug: [IMPALA-358](#)

Severity: High

Resolution: Fixed

Impala returns 0 for bad time values in UNIX_TIMESTAMP, Hive returns NULL

Impala returns 0 for bad time values in UNIX_TIMESTAMP, Hive returns NULL.

Impala:

```
impala> select UNIX_TIMESTAMP('10:02:01') ;
impala> 0
```

Hive:

```
hive> select UNIX_TIMESTAMP('10:02:01') FROM tmp;
hive> NULL
```

Bug: [IMPALA-16](#)

Severity: Low

Anticipated Resolution: Fixed

INSERT INTO TABLE SELECT <constant> does not work.

Insert INTO TABLE SELECT <constant> will not insert any data and may return an error.

Severity: Low

Anticipated Resolution: Fixed

Issues Fixed in the 1.0 GA Release

Here are the major user-visible issues fixed in Impala 1.0. For a full list of fixed issues, see [this report in the public issue tracker](#).

Undeterministically receive "ERROR: unknown row batch destination..." and "ERROR: Invalid query handle" from impala shell when running union query

A query containing both `UNION` and `LIMIT` clauses could intermittently cause the `impalad` process to halt with a segmentation fault.

Bug: [IMPALA-183](#)

Severity: High

Resolution: Fixed

Insert with NULL partition keys results in SIGSEGV.

An `INSERT` statement specifying a `NULL` value for one of the partitioning columns could cause the `impalad` process to halt with a segmentation fault.

Bug: [IMPALA-190](#)

Severity: High

Resolution: Fixed

INSERT queries don't show completed profiles on the debug webpage

In the Impala web user interface, the profile page for an `INSERT` statement showed obsolete information for the statement once it was complete.

Bug: [IMPALA-217](#)

Severity: High

Resolution: Fixed

Impala HBase scan is very slow

Queries involving an HBase table could be slower than expected, due to excessive memory usage on the Impala nodes.

Bug: [IMPALA-231](#)

Severity: High

Resolution: Fixed

Add some library version validation logic to impalad when loading impala-lzo shared library

No validation was done to check that the `impala-lzo` shared library was compatible with the version of Impala, possibly leading to a crash when using LZO-compressed text files.

Bug: [IMPALA-234](#)

Severity: High

Resolution: Fixed

Workaround: Always upgrade the `impala-lzo` library at the same time as you upgrade Impala itself.

Problems inserting into tables with TIMESTAMP partition columns leading table metadata loading failures and failed dchecks

`INSERT` statements for tables partitioned on columns involving datetime types could appear to succeed, but cause errors for subsequent queries on those tables. The problem was especially serious if an improperly formatted timestamp value was specified for the partition key.

Bug: [IMPALA-238](#)

Severity: Critical

Resolution: Fixed

Ctrl-C sometimes interrupts shell in system call, rather than cancelling query

Pressing Ctrl-C in the `impala-shell` interpreter could sometimes display an error and return control to the shell, making it impossible to cancel the query.

Bug: [IMPALA-243](#)

Severity: Critical

Resolution: Fixed

Empty string partition value causes metastore update failure

Specifying an empty string or `NULL` for a partition key in an `INSERT` statement would fail.

Bug: [IMPALA-252](#)

Severity: High

Resolution: Fixed. The behavior for empty partition keys was made more compatible with the corresponding Hive behavior.

Round() does not output the right precision

The `round()` function did not always return the correct number of significant digits.

Bug: [IMPALA-266](#)

Severity: High

Resolution: Fixed

Cannot cast string literal to string

Casting from a string literal back to the same type would cause an “invalid type cast” error rather than leaving the original value unchanged.

Bug: [IMPALA-267](#)

Severity: High

Resolution: Fixed

Excessive mem usage for certain queries which are very selective

Some queries that returned very few rows experienced unnecessary memory usage.

Bug: [IMPALA-288](#)

Severity: High

Resolution: Fixed

HdfsScanNode crashes in UpdateCounters

A serious error could occur for relatively small and inexpensive queries.

Bug: [IMPALA-289](#)

Severity: High

Resolution: Fixed

Parquet performance issues on large dataset

Certain aggregation queries against Parquet tables were inefficient due to lower than required thread utilization.

Bug: [IMPALA-292](#)

Severity: High

Resolution: Fixed

impala not populating hive metadata correctly for create table

The Impala `CREATE TABLE` command did not fill in the `owner` and `tbl_type` columns in the Hive metastore database.

Bug: [IMPALA-295](#)

Severity: High

Resolution: Fixed. The metadata was made more Hive-compatible.

impala daemons die if statestore goes down

The `impalad` instances in a cluster could halt when the `statestored` process became unavailable.

Bug: [IMPALA-312](#)

Severity: High

Resolution: Fixed

Constant SELECT clauses do not work in subqueries

A subquery would fail if the `SELECT` statement inside it returned a constant value rather than querying a table.

Bug: [IMPALA-67](#)

Severity: High

Resolution: Fixed

Right outer Join includes NULLs as well and hence wrong result count

The result set from a right outer join query could include erroneous rows containing `NULL` values.

Bug: [IMPALA-90](#)

Severity: High

Resolution: Fixed

Parquet scanner hangs for some queries

The Parquet scanner non-deterministically hangs when executing some queries.

Bug: [IMPALA-204](#)

Severity: Medium

Resolution: Fixed

Issues Fixed in Version 0.7 of the Beta Release

Impala does not gracefully handle unsupported Hive table types (INDEX and VIEW tables)

When attempting to load metadata from an unsupported Hive table type (`INDEX` and `VIEW` tables), Impala fails with an unclear error message.

Bug: [IMPALA-167](#)

Severity: Low

Resolution: Fixed in 0.7

DDL statements (CREATE/ALTER/DROP TABLE) are not supported in the Impala Beta Release

Severity: Medium

Resolution: Fixed in 0.7

Avro is not supported in the Impala Beta Release

Severity: Medium

Resolution: Fixed in 0.7

Workaround: None

Impala does not currently allow limiting the memory consumption of a single query

It is currently not possible to limit the memory consumption of a single query. All tables on the right hand side of JOIN statements need to be able to fit in memory. If they do not, Impala may crash due to out of memory errors.

Severity: High

Resolution: Fixed in 0.7

Aggregate of a subquery result set returns wrong results if the subquery contains a 'limit' and data is distributed across multiple nodes

Aggregate of a subquery result set returns wrong results if the subquery contains a 'limit' clause and data is distributed across multiple nodes. From the query plan, it looks like we are just summing the results from each slave.

Bug: [IMPALA-20](#)

Severity: Low

Resolution: Fixed in 0.7

Partition pruning for arbitrary predicates that are fully bound by a particular partition column

We currently can't utilize a predicate like "country_code in ('DE','FR','US')" to do partitioning pruning, because that requires an equality predicate or a binary comparison.

We should create a superclass of `planner.ValueRange`, `ValueSet`, that can be constructed with an arbitrary predicate, and whose `isInRange(analyzer, valueExpr)` constructs a literal predicate by substitution of the `valueExpr` into the predicate.

Bug: [IMPALA-144](#)

Severity: Medium

Resolution: Fixed in 0.7

Issues Fixed in Version 0.6 of the Beta Release

Impala reads the NameNode address and port as command line parameters

Impala reads the NameNode address and port as command line parameters rather than reading them from `core-site.xml`. Updating the NameNode address in the `core-site.xml` file does not propagate to Impala.

Severity: Low

Resolution: Fixed in 0.6 - Impala reads the namenode location and port from the Hadoop configuration files, though setting `-nn` and `-nn_port` overrides this. Users are advised not to set `-nn` or `-nn_port`.

Queries may fail on secure environment due to impalad Kerberos ticket expiration

Queries may fail on secure environment due to `impalad` Kerberos tickets expiring. This can happen if the Impala `-kerberos_reinit_interval` flag is set to a value ten minutes or less. This may lead to an `impalad` requesting a ticket with a lifetime that is less than the time to the next ticket renewal.

Bug: [IMPALA-64](#)

Severity: Medium

Resolution: Fixed in 0.6

Concurrent queries may fail when Impala uses Thrift to communicate with the Hive Metastore

Concurrent queries may fail when Impala is using Thrift to communicate with part of the Hive Metastore such as the Hive Metastore Service. In such a case, the error `get_fields failed: out of sequence response` may occur because Impala shared a single Hive Metastore Client connection across threads. With Impala 0.6, a separate connection is used for each metadata request.

Bug: [IMPALA-48](#)

Severity: Low

Resolution: Fixed in 0.6

impalad fails to start if unable to connect to the Hive Metastore

Impala fails to start if it is unable to establish a connection with the Hive Metastore. This behavior was fixed, allowing Impala to start, even when no Metastore is available.

Bug: [IMPALA-58](#)

Severity: Low

Resolution: Fixed in 0.6

Impala treats database names as case-sensitive in some contexts

In some queries (including "USE database" statements), database names are treated as case-sensitive. This may lead queries to fail with an `IllegalStateException`.

Bug: [IMPALA-44](#)

Severity: Medium

Resolution: Fixed in 0.6

Impala does not ignore hidden HDFS files

Impala does not ignore hidden HDFS files, meaning those files prefixed with a period '.' or underscore '_'. This diverges from Hive/MapReduce, which skips these files.

Bug: [IMPALA-18](#)

Severity: Low

Resolution: Fixed in 0.6

Issues Fixed in Version 0.5 of the Beta Release

Impala may have reduced performance on tables that contain a large number of partitions

Impala may have reduced performance on tables that contain a large number of partitions. This is due to extra overhead reading/parsing the partition metadata.

Severity: High

Resolution: Fixed in 0.5

Backend client connections not getting cached causes an observable latency in secure clusters

Backend impalads do not cache connections to the coordinator. On a secure cluster, this introduces a latency proportional to the number of backend clients involved in query execution, as the cost of establishing a secure connection is much higher than in the non-secure case.

Bug: [IMPALA-38](#)

Severity: Medium

Resolution: Fixed in 0.5

Concurrent queries may fail with error: "Table object has not been initialised : `PARTITIONS`"

Concurrent queries may fail with error: "Table object has not been initialised : `PARTITIONS`". This was due to a lack of locking in the Impala table/database metadata cache.

Bug: [IMPALA-30](#)

Severity: Medium

Resolution: Fixed in 0.5

UNIX_TIMESTAMP format behaviour deviates from Hive when format matches a prefix of the time value

The Impala UNIX_TIMESTAMP(val, format) operation compares the length of format and val and returns NULL if they do not match. Hive instead effectively truncates val to the length of the format parameter.

Bug: [IMPALA-15](#)

Severity: Medium

Resolution: Fixed in 0.5

*Issues Fixed in Version 0.4 of the Beta Release**Impala fails to refresh the Hive metastore if a Hive temporary configuration file is removed*

Impala is impacted by Hive bug [HIVE-3596](#) which may cause metastore refreshes to fail if a Hive temporary configuration file is deleted (normally located at /tmp/hive-<user>-<tmp_number>.xml). Additionally, the impala-shell will incorrectly report that the failed metadata refresh completed successfully.

Severity: Medium

Anticipated Resolution: To be fixed in a future release

Workaround: Restart the `impalad` service. Use the `impalad` log to check for metadata refresh errors.

lpad/rpad builtin functions is not correct.

The lpad/rpad builtin functions generate the wrong results.

Severity: Mild

Resolution: Fixed in 0.4

Files with .gz extension reported as 'not supported'

Compressed files with extensions incorrectly generate an exception.

Bug: [IMPALA-14](#)

Severity: High

Resolution: Fixed in 0.4

Queries with large limits would hang.

Some queries with large limits were hanging.

Severity: High

Resolution: Fixed in 0.4

Order by on a string column produces incorrect results if there are empty strings

Severity: Low

Resolution: Fixed in 0.4

*Issues Fixed in Version 0.3 of the Beta Release**All table loading errors show as unknown table*

If Impala is unable to load the metadata for a table for any reason, a subsequent query referring to that table will return an `unknown table` error message, even if the table is known.

Severity: Mild

Resolution: Fixed in 0.3

A table that cannot be loaded will disappear from SHOW TABLES

After failing to load metadata for a table, Impala removes that table from the list of known tables returned in `SHOW TABLES`. Subsequent attempts to query the table returns 'unknown table', even if the metadata for that table is fixed.

Severity: Mild

Resolution: Fixed in 0.3

Impala cannot read from HBase tables that are not created as external tables in the hive metastore.

Attempting to select from these tables fails.

Severity: Medium

Resolution: Fixed in 0.3

Certain queries that contain OUTER JOINS may return incorrect results

Queries that contain OUTER JOINS may not return the correct results if there are predicates referencing any of the joined tables in the WHERE clause.

Severity: Medium

Resolution: Fixed in 0.3.

Issues Fixed in Version 0.2 of the Beta Release

Subqueries which contain aggregates cannot be joined with other tables or Impala may crash

Subqueries that contain an aggregate cannot be joined with another table or Impala may crash. For example:

```
SELECT * FROM (SELECT sum(col1) FROM some_table GROUP BY col1) t1 JOIN other_table ON (...);
```

Severity: Medium

Resolution: Fixed in 0.2

An insert with a limit that runs as more than one query fragment inserts more rows than the limit.

For example:

```
INSERT OVERWRITE TABLE test SELECT * FROM test2 LIMIT 1;
```

Severity: Medium

Resolution: Fixed in 0.2

Query with limit clause might fail.

For example:

```
SELECT * FROM test2 LIMIT 1;
```

Severity: Medium

Resolution: Fixed in 0.2

Files in unsupported compression formats are read as plain text.

Attempting to read such files does not generate a diagnostic.

Severity: Medium

Resolution: Fixed in 0.2

Impala server raises a null pointer exception when running an HBase query.

When querying an HBase table whose row-key is string type, the Impala server may raise a null pointer exception.

Severity: Medium

Resolution: Fixed in 0.2

Cloudera Manager 5 Release Notes

These Release Notes provide information on the new features and known issues and limitations for Cloudera Manager 5. These Release Notes also include fixed issues for releases starting from Cloudera Manager 5.0.0 beta 1.

To view the Release Notes (or other documentation) for a specific Cloudera Manager release, go to [Cloudera Documentation](#), click a major version link, and use the drop-down menu to select the release.

For information about supported operating systems, and other requirements for using Cloudera Manager, see [Cloudera Manager 5 Requirements and Supported Versions](#).

New Features and Changes in Cloudera Manager 5

The following sections describe what's new and changed in each Cloudera Manager 5 release.

What's New in Cloudera Manager 5

- **Note:** Although there is a CDH 5.3.5 release, there is no synchronous Cloudera Manager 5.3.5 release.

What's New In Cloudera Manager 5.3.4

A number of issues have been fixed. See [Issues Fixed in Cloudera Manager 5.3.4](#) on page 173.

What's New In Cloudera Manager 5.3.3

A number of issues have been fixed. See [Issues Fixed in Cloudera Manager 5.3.3](#) on page 174.

What's New in Cloudera Manager 5.3.2

A number of issues have been fixed. See [Issues Fixed in Cloudera Manager 5.3.2](#) on page 175 for details.

What's New in Cloudera Manager 5.3.1

A number of issues have been fixed. See [Issues Fixed in Cloudera Manager 5.3.1](#) on page 176 for details.

What's New in Cloudera Manager 5.3.0

- **JDK 1.8** - Cloudera Manager adds support for Oracle JDK 1.8.
- **Single user mode** - The Cloudera Manager Agent and all service processes can now be run as a single configured user in environments where running as root is not permitted.
- **CDH upgrade wizard enhanced** - The CDH upgrade wizard now supports minor and maintenance version upgrade as well as major version upgrade.
- **Oozie Sharelib** - The Oozie Sharelib can be updated without restarting the Oozie service.
- **Read-only users prevented from viewing process logs or environment** - Read-only users can no longer view the environment or logs of a process. This is to prevent read-only users from seeing potentially sensitive information.
- New icons for the KMS and Key Trustee services.
- **Data-at-rest encryption**

- **Important:** Cloudera provides two solutions:
 - **Navigator Encrypt** is production ready and available to Cloudera customers licensed for Cloudera Navigator. Navigator Encrypt operates at the Linux volume level, so it can encrypt cluster data inside and outside HDFS. Consult your Cloudera account team for more information.
 - **HDFS Encryption** is production ready and operates at the HDFS directory level, enabling encryption to be applied only to HDFS folders where needed.

HDFS encryption implements transparent, end-to-end encryption of data read from and written to HDFS by creating encryption zones. An encryption zone is a directory in HDFS with every file and subdirectory in it encrypted. Use *one* of the following services to store, manage, and access encryption zone keys:

- **KMS (File)** - The Hadoop Key Management Server with a file-based Java keystore; maintains a single copy of keys, using simple password-based protection.
 - **KMS (Navigator Key Trustee)** - An enterprise-grade key management service that replaces the file-based Java keystore and leverages the advanced key-management capabilities of Cloudera Navigator Key Trustee. Navigator Key Trustee is designed for secure, authenticated administration and cryptographically strong storage of keys on multiple redundant servers that can be located outside the cluster.
- The Cloudera Manager Server now reports the correct number of physical cores and hyper-threading cores if hyper-threading is enabled.
 - **Client configurations** - Client configurations are now managed so that they are redeployed when a machine is re-imaged.

- **Important:** The changes to client configurations affect some API calls, as follows:
 - When a host ceases to have a client configuration assigned to it, Cloudera Manager will remove it, rather than leaving it behind. If a host has a client configuration assigned and the client configuration is missing, Cloudera Manager will recreate it.
 - If you currently use the API command `deployClientConfig` to deploy the client configurations for a particular service, and you pass a specific set of role names to this call to narrow the set of hosts that receive the new client configuration, then you should be aware that:
 - The API command will continue to generate and deploy the client configuration only to the hosts that correspond to the specified role names.
 - Any other hosts that previously had deployed client configurations, but do not have gateway roles assigned to them, will have those client configurations removed from them. This is the new behavior.
 - The behavior of the cluster level `deployClientConfig` command, and calling the service level command with no arguments, is unchanged. The command still deploys a new client configuration to all hosts with roles corresponding to the specified service or cluster.
 - As this change is due to internal functional changes inside CM, it is not restricted to any new API level. The `deployClientConfig` command in all API levels is affected.

- **Configuration**
 - **NameNode configuration** - The decommissioning parameters `dfs.namenode.replication.max-streams` and `dfs.namenode.replication.max-streams-hard-limit` are now available.
 - **Hue debug options** - Two service-level configuration parameters have been added to the Hue service to enable Django debug mode and debugging of internal server error responses.

What's New in Cloudera Manager 5.2.5

A number of issues have been fixed, see [Issues Fixed in Cloudera Manager 5.2.5](#) on page 178.

What's New in Cloudera Manager 5.2.4

There are no changes for Cloudera Manager 5.2.4. It was released to provide the Cloudera Navigator fix in [What's New in Cloudera Navigator 2.1.4](#) on page 192.

- **Note:** Although there is a CDH 5.2.3 release, there is no synchronous Cloudera Manager 5.2.3 release.

What's New in Cloudera Manager 5.2.2

- **HDFS Decommissioning** - The following decommissioning properties have been exposed in Cloudera Manager 5.2.2.

- **Maximum number of replication threads on a Datanode** (`dfs.namenode.replication.max-streams`)
- **Hard limit on the number of replication threads on a Datanode** (`dfs.namenode.replication.max-streams-hard-limit`)

- New icons for the KMS and Key Trustee services.

A number of issues have been fixed. See [Issues Fixed in Cloudera Manager 5.2.2](#) on page 178.

What's New in Cloudera Manager 5.2.1

This release fixes the “POODLE” vulnerability and a number of other issues. See [Issues Fixed in Cloudera Manager 5.2.1](#) on page 178.

- The YARN `yarn.nodemanager.recovery.dir` property can be configured.
- A health check indicates whether the HDFS metadata upgrade has not been finalized.

What's New in Cloudera Manager 5.2.0

- **OS and database support** - Adds support for Ubuntu Trusty (version 14.04) and PostgreSQL 9.3.
- **Services** - the following new services have been added:
 - **Isilon** - supports the EMC Isilon distributed filesystem
 - **KMS** - the Java keystore-based key management server
 - **Key Trustee** - the enterprise-grade key management server using Cloudera Navigator Key Trustee
 - **Spark** - running Spark applications on YARN. The existing Spark service has been renamed Spark (Standalone).
- **Accumulo** - Kerberos authentication is now supported. If you have been using advanced configuration snippets (safety valves) to configure Kerberos with Accumulo, you may now remove those settings and have Cloudera Manager generate the principal and keytab file for you.
- **HDFS Data at Rest Encryption** -

- **Note:** Cloudera provides the following two solutions for data at rest encryption:
 - **Navigator Encrypt** - is production ready and available for Cloudera customers licensed for Cloudera Navigator. Navigator Encrypt operates at the Linux volume level, so it can encrypt cluster data inside and outside HDFS. Talk to your Cloudera account team for more information about this capability.
 - **HDFS Encryption** - included in CDH 5.2.0 operates at the HDFS folder level, enabling encryption to be applied only to HDFS folders where needed. This feature has several known limitations. Therefore, Cloudera does not currently support this feature in CDH 5.2 and it is *not* recommended for production use. If you're interested in trying the feature out, upgrade to the latest version of CDH 5.

HDFS now implements transparent, end-to-end encryption of data read from and written to HDFS by creating encryption zones. An encryption zone is a directory in HDFS with all of its contents, that is, every file and subdirectory in it, encrypted. You can use either the **KMS** or the **Key Trustee** service to store, manage, and access encryption zone keys.

- **HBase** - Support for configuring hedged reads has been added for HBase. The default configuration is to turn hedged reads off. Cloudera Manager will emit two properties, `dfs.client.hedged.read.threadpool.size` (default: 0) and `dfs.client.hedged.read.threshold.millis` (default: 500ms) to `hbase-site.xml`.
- **ZooKeeper** - the RMI port can be configured. The port is configured using the JDK7 flag `-Dcom.sun.management.jmxremote.rmi.port`. The default value is set to be same as the JMX Agent port. Also, a special value of 0 or -1 disables the setting and a random port is used. The configuration has no effect on versions lower than Oracle JDK 7u4.
- **Cloudera Manager Agent configuration**

- The supervisord port can now be configured in the Agent configuration `supervisord_port`. The change takes effect the next time supervisord is restarted (not simply when the Agent is restarted).
- Added an Agent configuration `local_filesystem_whitelist` that allows configuring the list of local filesystems that should always be monitored.
- **Proxy user configuration**
 - All services' proxy user configuration properties have been moved to the HDFS service. Other services running on the cluster inherit the configuration values provided in HDFS. If you have previously configured a service to have values different from those configured in HDFS, then the proxy user configuration properties will be moved to that service's Advanced Configuration Snippet (Safety Valve) for `core-site.xml` to retain existing behavior.

Oozie and Solr are exceptions to this. Oozie proxy user configuration properties have been moved to **Oozie Server Advanced Configuration Snippet (Safety Valve) for `oozie-site.xml`** if they differ from HDFS. Solr proxy user configuration properties have been moved to **Solr Service Environment Advanced Configuration Snippet (Safety Valve)** if they differ from HDFS.
- **Resource management** - YARN and Llama integrated resource management and Llama high availability wizard.
- **New and changed user roles** - BDR Administrator, Cluster Administrator, Navigator Administrator, and User Administrator. The Administrator role has been renamed Full Administrator.
- **Configuration UI**
 - Cluster-wide configuration - you can view all modified settings and configure log directories, disk space thresholds, and port settings.
 - New configuration layout - the new layout provides an alternate way to view configuration pages. In the **classic** layout, pages are organized by role group and categories within the role groups. The **new** layout allows you to filter on configuration status, category, and scope. On each configuration page you can easily switch between the classic and new layout.

- **Important:** The classic layout is the default. All the configuration procedures described in the Cloudera Manager documentation assume the classic layout.

What's New in Cloudera Manager 5.1.5

A number of issue have been fixed. See [Fixed Issues in Cloudera Manager 5.1.5](#) on page 180.

What's New in Cloudera Manager 5.1.4

A number of issues have been fixed. See [Fixed Issues in Cloudera Manager 5.1.4](#) on page 180.

What's New in Cloudera Manager 5.1.3

A number of issues have been fixed. See [Fixed Issues in Cloudera Manager 5.1.3](#) for details.

- **JDK Installation**
 - Users who are adding or upgrading hosts can now choose not to install the JDK that ships with Cloudera Manager.

What's New in Cloudera Manager 5.1.2

A number of issues have been fixed. See [Fixed Issues in Cloudera Manager 5.1.2](#) for details.

- **New SAML configuration option**
 - You can now specify the binding protocol to be used for AuthNResponses sent from the IDP to Cloudera Manager. Previously, Cloudera Manager would only use HTTP-Artifact, but it is now possible to choose HTTP-Post. HTTP-Artifact remains the default binding.

What's New in Cloudera Manager 5.1.1

An issue has been fixed. See [Issues Fixed in Cloudera Manager 5.1.1](#) on page 182 for details.

What's New in Cloudera Manager 5.1.0

- **Important:** Cloudera Manager 5.1.0 is no longer available for download from the Cloudera website or from archive.cloudera.com due to the JCE policy file issue described in the [Fixed Issues in Cloudera 5.1.1](#) section of the Release Notes. The download URL at archive.cloudera.com for Cloudera Manager 5.1.0 now forwards to Cloudera Manager 5.1.1 for the RPM-based distributions for Linux RHEL and SLES.

■ SSL Encryption

- Supports several new SSL-related configuration parameters for HDFS, MapReduce, YARN and HBase, which allow you to configure and enable encrypted shuffle and encrypted web UIs for these services.
- Cloudera Manager now also supports the monitoring of HDFS, MapReduce, YARN, and HBase when SSL is enabled for these services. New configuration parameters allow you to specify the location and password of the truststore used to verify certificates in HTTPS communication with CDH services and the Cloudera Manager Server.

■ Sentry Service

- A new Sentry service that stores the authorization metadata in an underlying relational database and allows you to use Grant/Revoke statements to modify privileges.
- You can also configure the Sentry service to allow Pig, MapReduce, and WebHCat queries access to Sentry-secured data stored in Hive.

■ Kerberos Authentication

- Now supports a Kerberos cluster using an Active Directory KDC.
- New wizard to enable Kerberos on an existing cluster. The wizard works with both MIT KDC and Active Directory KDC.
- Ability to configure and deploy Kerberos client configuration (`krb5.conf`) on a cluster.

■ Spark Service - added the History Server role

■ Impala - added support for Llama ApplicationMaster High Availability

■ User Roles - there are two new roles: Operator and Configurator that support fine-grained access to Cloudera Manager features.

■ Monitoring

- Updates to Oozie monitoring
- New Hive Metastore Canary

■ UI - The UI has been updated to improve scalability. The Home page Status tab can be configured to display clusters in a full or summary format. There is a new Cluster page for each cluster. The Hosts and Instances pages have added faceted filters.

What's New in Cloudera Manager 5.0.6

A number of issues have been fixed. See [Fixed Issues in Cloudera Manager 5.0.6](#) on page 184.

What's New in Cloudera Manager 5.0.5

A number of issues have been fixed. See [Fixed Issues in Cloudera Manager 5.0.5](#) on page 185.

What's New in Cloudera Manager 5.0.2

A number of issues have been fixed. See [Issues Fixed in Cloudera Manager 5.0.2](#) on page 185 for details.

What's New in Cloudera Manager 5.0.1

A number of issues have been fixed. See [Issues Fixed in Cloudera Manager 5.0.1](#) on page 185 for details.

- **Monitoring**

- The Java Garbage Collection Duration health test for the Service Monitor, Host Monitor, and Activity Monitor has been replaced with the new Java Pause Duration health test.

What's New in Cloudera Manager 5.0.0

- **Service and Configuration Management**

- **HDFS** - cache management

- **Resource Management** - Impala admission control

- **Monitoring**

- Host disks overview
- Impala best practices
- HBase table statistics
- HDFS cache statistics

What's New in Cloudera Manager 5.0.0 Beta 2

- **Service and Configuration Management**

- **HDFS**

- HDFS NFS Gateway role
- Supports restoration of HDFS data from a snapshot

- **YARN**

- YARN Resource Manager High Availability
- Resource pool scheduler

- Support for Spark service
- Support for Accumulo service
- Support for service extensibility
- Support to set up Oozie server High Availability
- Granular configuration staleness UI
- Support for setting maximum file descriptors

- **Monitoring**

- Support for monitoring the Cloudera Search/Solr service
- New "failed" and "killed" badges displayed for unsuccessful YARN applications
- More attributes available for filtering displays of YARN applications and Impala queries
- New operational reports added for HBase tables and namespaces, Impala queries, and YARN applications
- Support for creating user-defined triggers for metrics accessible via charts/tsquery

- **Important:** Because triggers are a new and evolving feature, backward compatibility between releases is not guaranteed at this time.

- Charting improvements
 - New table chart type
 - New options for displaying data and metadata from charts
 - Support for exporting data from charts to CSV or JSON files

- **Administrative Settings**

- Added a new role type with limited administrator capabilities
- Cloudera Manager Server and all JVMs will create a heap dump if they run out of memory

- Configure the location of the parcel directory and specify whether and when to remove old parcels from cluster hosts

What's New in Cloudera Manager 5.0.0 Beta 1

■ CDH Version

- Supports both CDH 4 and CDH 5
- CDH 4 to CDH 5 upgrade wizard
- Support for YARN as a production execution environment
 - MapReduce (MRv1) to YARN (MRv2) configuration import
 - YARN-based resource management for Impala 1.2

■ JDK Version - Cloudera Manager 5 supports and installs both JDK 6 and JDK 7.

■ Resource Management

- Static and dynamic partitioning of resources: provides a wizard for configuring static partitioning of resources (cgroups) across core services (HBase, HDFS, MapReduce, Solr, YARN) and dynamic allocation of resources for YARN and Impala.
- Pool, resource group, and queue administration for YARN and Impala.
- Usage monitoring and trending

■ Monitoring

- YARN service monitoring
- YARN (MRv2) job monitoring
- Configurable histograms of Impala query and YARN job attributes that can be used to quickly filter query and application lists
- Scalable back-end database for monitoring metrics
- Charting improvements
 - New chart types: histogram and heatmap
 - New scale types: logarithmic and power
 - Updates to tsquery language: new attribute values to support YARN and new functions to support new chart types

■ Extensibility

- Ability to manage both ISV applications and non-CDH services (for example, Accumulo, Spark, and so on)
- Working with select ISVs as part of Beta 1

■ Single Sign-On - Support for SAML to enable single sign-on

■ Parcels

- Dependency enforcement to ensure incompatible parcels are not used together
- Option to not cache downloaded parcels, to save disk space
- Improved error reporting for management operations

■ Backup and Disaster Recovery (BDR)

- HBase and HDFS snapshots: Supports scheduling snapshots on a recurring basis.
- Support for YARN (MRv2): Replication jobs can now run using YARN (MRv2) instead of MRv1.
- Global replication page: All scheduled snapshots (HDFS and HBase) and replication jobs for either HDFS or Hive are shown on a single Replications page

■ Other

- Global Search box
- Several usability improvements

- Comprehensive detection of configuration changes that require service restarts, refresh and redeployment of client configurations.

Incompatible Changes in Cloudera Manager 5

The following sections describe incompatible changes in each Cloudera Manager 5 release.

Incompatible Changes Introduced in Cloudera Manager 5.3.0

- Oozie metrics - The Oozie metrics framework is now controlled by the **Enable The Metrics Instrumentation Service** flag, which is enabled by default. When enabled, the old 'instrumentation' REST end-point is disabled and metrics are available on the new 'metrics' REST end-point (*hostname:port/v2/admin/metrics*).

Incompatible Changes Introduced in Cloudera Manager 5.2.0

- Due to various internal changes to configuration generation, all service and client configurations will be stale after upgrade. To propagate the updates, restart the cluster and redeploy client configurations.

Incompatible Changes Introduced in Cloudera Manager 5.1.0

- The Limited Administrator role has been renamed Limited Operator. The Limited Operator role is no longer available in Cloudera Manager Express. If you upgrade a Cloudera Manager Express installation, users in the Limited Operator role will not be able to log in. A user in the Administrator role must assign the Read-Only or Administrator role to those users.

Incompatible Changes Introduced in Cloudera Manager 5.0.0

- **Cloudera Manager API**
 - New [upgradeCdh](#) command, which upgrades CDH cluster versions. Use this command to upgrade clusters from CDH 4 to CDH 5. The `upgradeServices` command previously used to upgrade CDH cluster versions is no longer supported.
 - The `hostId` field now contains a unique UUID and no longer matches the `hostName` field. When referring to a host, both `hostId` and `hostName` are accepted. However, any API clients that were previously cross-referencing host records with external information by `hostName`, but were using the `hostId` field in the API, must be updated to use the `hostName` field. Clients updated in this manner will function correctly with older versions of Cloudera Manager because the `hostName` field has always been present.
 - The `clusterName` field displayed when viewing service and role references is now an internal name and may not match the external `displayName` field of the cluster.
- CDH 5 Hue will only work with the default Python version of the operating system on which it is being installed. For example, on RHEL/CentOS 6 you will need Python 2.6 to start Hue.
- Cloudera Manager 5.0 includes a change to the value of the `snmpTrapOID`. Earlier releases set the value of `snmpTrapOID` (OID: `.1.3.6.1.6.3.1.1.4.1.0`) wrongly to `clouderaManagerMIBNotifications` (OID `.1.3.6.1.4.1.38374.1.1.1`). This is fixed in Cloudera Manager 5.0 with the correct value, which is `clouderaManagerAlert` (OID `.1.3.6.1.4.1.38374.1.1.1`). This change will break SNMP server setups that are configured to expect `clouderaManagerMIBNotifications`. Cloudera Manager administrators should configure their SNMP receivers to accept the corrected OID.
- The default values for the following configurations have changed to include the JVM option `-Djava.net.preferIPv4Stack=true`, which sets the preferred protocol stack to IPv4 on dual-stack machines. Any values set to the old defaults will automatically be changed to the new default when upgrading to Cloudera Manager 5.
 - MapReduce client configuration:
 - `hadoop-env.sh`: added to `HADOOP_CLIENT_OPTS`
 - `mapred-site.xml`: added to `mapred.child.java.opts`
 - YARN client configuration:
 - `hadoop-env.sh`: added to `YARN_OPTS`

- `mapred-site.xml`: added to `yarn.app.mapreduce.am.command-opts`, `mapreduce.map.java.opts`, and `mapreduce.reduce.java.opts`
- HDFS client configuration: `hadoop-env.sh`: added to `HADOOP_CLIENT_OPTS`
- Hive client configuration: `hive-env.sh`: added to `HADOOP_CLIENT_OPTS`
- MapReduce health tests have been removed:
 - Job failure
 - Map backlog
 - Reduce backlog
 - Map locality

If needed, the test can be replaced with a trigger. For example:

- Looks at all the jobs that completed in the last hour and if there are more than 10% of failed jobs, change the health of the service to concerning:

```
IF (select (jobs_failed_rate * 3600) as jobs_failed, ((jobs_failed_rate +
jobs_completed_rate + jobs_killed_rate) * 3600) as all_jobs where
roleType=JOBTRACKER AND serviceName=$SERVICENAME and last(jobs_failed_rate /
(jobs_failed_rate + jobs_completed_rate +
jobs_killed_rate)) >= 10 ending at $END_TIME duration "PT3600S") DO
health:concerning
```

- If there are more than 50% maps waiting than total slots available, health goes concerning.

```
IF (select waiting_maps / map_slots where roleType=JOBTRACKER and
serviceName=$SERVICENAME and last(waiting_maps / map_slots) > 50) DO
health:concerning
```

- If there are more than 50% reduce waiting than total slots available, health goes concerning.

```
IF (select waiting_reduces / reduce_slots where roleType=JOBTRACKER and
serviceName=$SERVICENAME and last(waiting_reduces / reduce_slots) > 50) DO
health:concerning
```

- HDFS checkpointing metrics have been removed:

- `end_checkpoint_num_ops`
- `end_checkpoint_avg_time`
- `start_checkpoint_num_ops`
- `start_checkpoint_avg_time`

Incompatible Changes Introduced in Cloudera Manager 5.0.0 Beta 2

- Impala releases earlier than 1.2.1 are no longer supported.
- Some of the constants identifying health tests have changed. The following existed in Cloudera Manager 4:
 - `FAILOVERCONTROLLER_FILE_DESCRIPTOR`
 - `FAILOVERCONTROLLER_HOST_HEALTH`
 - `FAILOVERCONTROLLER_LOG_DIRECTORY_FREE_SPACE`
 - `FAILOVERCONTROLLER_SCM_HEALTH`
 - `FAILOVERCONTROLLER_UNEXPECTED_EXITS`

They are now:

- `MAPREDUCE_FAILOVERCONTROLLER_FILE_DESCRIPTOR`
- `MAPREDUCE_FAILOVERCONTROLLER_HOST_HEALTH`
- `MAPREDUCE_FAILOVERCONTROLLER_LOG_DIRECTORY_FREE_SPACE`
- `MAPREDUCE_FAILOVERCONTROLLER_SCM_HEALTH`
- `MAPREDUCE_FAILOVERCONTROLLER_UNEXPECTED_EXITS`

and

- HDFS_FAILOVERCONTROLLER_FILE_DESCRIPTOR
- HDFS_FAILOVERCONTROLLER_HOST_HEALTH
- HDFS_FAILOVERCONTROLLER_LOG_DIRECTORY_FREE_SPACE
- HDFS_FAILOVERCONTROLLER_SCM_HEALTH
- HDFS_FAILOVERCONTROLLER_UNEXPECTED_EXITS

The reason for the change is to better distinguish between MapReduce and HDFS failover controller monitoring in the health system.

Incompatible Changes Introduced in Cloudera Manager 5.0.0 Beta 1

■ Services

- **Impala** - With Cloudera Manager 4.8 (released in late November 2013), only Impala 1.2.1 is supported, due to the introduction of the Impala Catalog Server. However, CDH 5.0.0 Beta 1 was released with Impala 1.2.0 (Beta). Therefore, if you upgrade from Cloudera Manager 4.8 (with Impala 1.2.1) to Cloudera Manager 5.0.0 Beta 1, and then upgrade your CDH to CDH 5.0.0 Beta 1, your version of Impala will be downgraded to Impala 1.2.0 from 1.2.1. This will result in some loss of functionality. See [New Features in Impala](#) for a list of the new features in Impala 1.2.1 that are not in Impala 1.2.0 (Beta).
- **Hive** - HiveServer2 is a mandatory role for Hive in CDH 5.
- **Hue** - In CDH 5, Hue no longer has a Beeswax Server role. Hue now submits queries to HiveServer2.
- **HDFS** - Cloudera Manager 5 does not support NFS-mounted shared edits directories for HDFS High Availability. It only supports the Quorum Journal method for shared edits. If you upgrade from Cloudera Manager 4 with a working CDH 4 High Availability configuration that uses NFS-mounted directories, your installation will continue to work until you disable High Availability. You will not be able to re-enable High Availability with NFS-mounted directories. Furthermore, you will not be able to upgrade to CDH 5 unless you disable High Availability, and you will need to use Quorum-based storage in order to re-enable High Availability after the upgrade.
- **YARN**
 - The YARN (MRv2) configuration `mapreduce.job.userlog.retain.hours` has been replaced by `yarn.log-aggregation.retain-seconds`. Any existing value in `mapreduce.job.userlog.retain.hours` will be lost. However, this configuration never had any effect, so no functionality is affected.
 - The following configuration parameters were removed from YARN. These never had any effect, so no functionality is affected.
 - `mapreduce.jobtracker.maxtasks.perjob`
 - `mapreduce.jobtracker.handler.count` (non-functional duplicate of `yarn.resourcemanager.resource-tracker.client.thread-count`)
 - `mapreduce.jobtracker.persist.jobstatus.active`
 - `mapreduce.jobtracker.persist.jobstatus.hours`
 - `mapreduce.job.jvm.numtasks`
 - The following YARN configuration parameters were replaced. Only the YARN parameters were replaced. Old configurations will be lost, but they never had any effect so this does not affect functionality.
 - `mapreduce.jobtracker.restart.recover` replaced by `yarn.resourcemanager.recovery.enabled` (changed from Gateway to ResourceManager)
 - `mapreduce.tasktracker.http.threads` replaced by `mapreduce.shuffle.max.connections`
 - `mapreduce.jobtracker.staging.root.dir` replaced by `yarn.app.mapreduce.am.staging-dir`
 - Cloudera Manager 5 sets the default YARN Resource Scheduler to FairScheduler. If a cluster was previously running YARN with the FIFO scheduler, it will be changed to FairScheduler the next time YARN restarts. The FairScheduler is only supported with CDH 4.2.1 and later, and older clusters may hit failures and need to manually change the scheduler to FIFO or CapacityScheduler. See the Known

Issues section of this Release Note for information on how to change the scheduler back to FIFO or CapacityScheduler.

Changed Features and Behaviors in Cloudera Manager 5

The following sections describe what's changed in each Cloudera Manager 5 release.

- **Note:** Rolling upgrade is not supported between CDH 4 and CDH 5. Rolling upgrade will also *not* be supported from CDH 5.0.0 Beta 2 to any later releases, and may not be supported between any future beta versions of CDH 5 and the General Availability release of CDH 5.

What's Changed in Cloudera Manager 5.3.4

- In CDH 5.3 and later releases, refreshable DataNode properties reside in a separate configuration file called `hdfs-site-refreshable.xml`. This file is included in the standard `hdfs-site.xml` file using the XInclude mechanism. The only property currently stored in `hdfs-site-refreshable.xml` is `dfs.datanode.data.dir`, which specifies the DataNode data directories. Note that `hdfs-site-refreshable.xml` is only present in the DataNode daemon configuration. Client configuration is not affected.

What's Changed in Cloudera Manager 5.3.3

- The property `hive.log.explain.output` is known to create instability of Cloudera Manager Agents in some specific circumstances, specially when the hive queries generate extremely large EXPLAIN outputs. Therefore, the property has been hidden from the Cloudera Manager configuration UI. The property can still be configured through the use of advanced configuration snippets.

What's Changed in Cloudera Manager 5.3.2

- Turning on the internal HBase canary (not to be confused with Cloudera Manager monitoring canary) is optional. On new clusters, it will not be enabled by default. Existing clusters will continue to run the canary until it is disabled from the HBase configuration page.

What's Changed in Cloudera Manager 5.3.0

- **Cloudera Manager upgrade** - If you have any active commands running before upgrade, the server *will fail to start* after upgrade. This includes commands a user might have run and also for commands Cloudera Manager automatically triggers, either in response to a state change, or something that's on a schedule.

What's Changed in Cloudera Manager 5.2.1

- The default value of the YARN `yarn.nodemanager.recovery.dir` property has changed from `{hadoop.tmp.dir}/yarn-nm-recovery` to `/var/lib/hadoop-yarn/yarn-nm-recovery`.

What's Changed in Cloudera Manager 5.2.0

- **Rolling upgrade** - As a result of a recent change in the way DataNodes handle block deletions during a rolling upgrade ([HDFS-5907](#)), the Trash directory may grow unexpectedly while the upgrade is in progress. Deleted blocks are kept during upgrade in case you want to roll back. The blocks are cleaned up after you finalize the upgrade.
- **Agent** -
 - The `hard_stop`, `hard_restart`, and `clean_restart` commands now show a warning message about the impact of using these commands instead of performing the actions. To actually perform the actions, you use the `hard_stop_confirmed`, `hard_restart_confirmed`, and `clean_restart_confirmed` commands.
 - The default supervisor port is changed from 9001 to 19001
- YARN application attributes renamed: `slot_millis` to `slots_millis` and `fallow_slot_millis` to `fallow_slots_millis`

What's Changed in Cloudera Manager 5.1.0



- **UI refresh for scalability**

- Revised authorization privilege model in Sentry.

What's Changed in Cloudera Manager 5.0.0

- MapReduce now inherits topology from HDFS NameNode. Topology configuration for MapReduce JobTracker was removed. The configuration was redundant and the two parameters should always have been set to the same value.
- **UI**
 - The Clusters tab no longer has Activities, Other, and Manage Resources sections.

What's Changed in Cloudera Manager 5.0.0 Beta 2

- **Product**
 - Cloudera Backup and Disaster Recovery (BDR) is now included with Cloudera Enterprise.
 - Cloudera Standard has been renamed to Cloudera Express.
- **OS and packaging**
 - The name of the Cloudera Manager embedded database package has changed from `cloudera-manager-server-db` to `cloudera-manager-server-db-2`. For details, read the upgrade and install topics for your OS.
 - Support for Ubuntu 10.04 and Debian 6.0 is deprecated.
- **HDFS** - enabling High Availability automatically enables auto-failover, unlike in Cloudera Manager 4 where enable auto-failover was a separate command.
- **HBase**
 - In CDH 5 there is no HBase canary because HBase is now monitored by a watchdog process. In CDH 4, the HBase canary is still used.
 - The RegionServer default heap size has been increased to 4GB.
- **Monitoring**
 - Chart "Views" and actions related to views have been renamed to "Dashboard".
 - Changes to how attribute filters are displayed in the Impala queries and YARN applications screens
 - The outdated configuration indicator on the Home, service, and role pages has a new graphic  and now has a tooltip that displays whether a cluster refresh or restart is required. There is a new indicator  for changes that require redeploying client configurations. You can click an indicator to go to the new Stale Configurations page to view and resolve the conditions that gave rise to the indicator.
 - To match the naming convention of tsquery metrics, multiword Impala query and YARN application attribute names have changed from camel case to using an underscore separator. For example `queryType` has changed to `query_type`. For backward compatibility, camel case names are still supported.
- **UI**
 - The main navigation bar in Cloudera Manager Admin Console has been reorganized. The Services tab has been replaced by a Clusters tab that contains links to individual services, which were previously under the Services tab, Activities and Reports sections, which were removed from the main bar, and a new Manage Resources section, which contains links to the new resource pools and service pools features. The All Services page has been removed.
 - The "Safety Valve" properties have been renamed "Advanced Configuration Snippet".
 - The screen for specifying assignment of roles to hosts has been redesigned for improved scalability and usability
- **Misc**
 - The `io.compression.codecs` property has moved from MapReduce to HDFS

What's Changed in Cloudera Manager 5.0.0 Beta 1

- When CDH 5 is installed, YARN is installed by default, rather than MapReduce, and is the default execution environment. MapReduce is deprecated in CDH 5 but is fully supported for backward compatibility through CDH 5. In CDH 4, MapReduce is still the default.
- The setting for `yarn.scheduler.maximum-allocation-mb` has been increased to a default of 64GB.
- The minimum heap size for the Solr service has been increased to 200MB (from 50MB previously) to enable it to better handle collection creation.

Known Issues and Workarounds in Cloudera Manager 5

The following sections describe the current known issues in Cloudera Manager 5.

NameNode Incorrectly Reports Missing Blocks During Rolling Upgrade

During a rolling upgrade to any of the CDH releases listed below, the NameNode may report missing blocks after rolling multiple DataNodes. This is caused by a race condition with block reporting between the DataNode and the NameNode. No permanent data loss occurs, but data can be unavailable for up to six hours before the problem corrects itself.

Releases affected: CDH 5.0.6, 5.1.5, 5.2.5, 5.3.3

Releases containing the fix: 5.3.4

Workaround:

- **To avoid the problem** - Cloudera advises skipping the affected releases and installing a release containing the fix. For example, do not upgrade to CDH 5.3.3; upgrade to CDH 5.3.4 instead.
- **If you have already completed an upgrade to an affected release, or are installing a new cluster** - You can continue to run the release, or upgrade to a release that is not affected, as you choose.

Using ext3 for server dirs easily hit inode limit

Using the ext3 filesystem for the Cloudera Manager command storage directory may exceed the maximum subdirectory size of 32000.

Workaround: Either decrease the value of the **Command Eviction Age** property so that the directories are more aggressively cleaned up, or migrate to the ext4 filesystem.

Monitoring Fails on Impala Catalog Server with SSL Enabled

When enabling SSL for Impala web servers (`webserver_certificate_file`), Cloudera Manager doesn't emit `use_ssl` in the `cloudera-monitor.properties` file for the Catalog Server. Other services (Impala Daemon and StateStore) are configured correctly. This causes monitoring to fail for the Catalog Server even though it is working as expected.

New Sentry synchronization path prefixes added in NameNode configuration are not enforced correctly

Any new path prefixes added in the NameNode configuration are not correctly enforced by Sentry. The ALCs are initially set correctly, however they would be reset to old default after some time interval.

Workaround: Set following property in **Sentry Service Advanced Configuration Snippet (Safety Valve)** and **Hive Metastore Server Advanced Configuration Snippet (Safety Valve)** for `hive-site.xml`:

```
<property>
<name>sentry.hdfs.integration.path.prefixes</name>
<value>/user/hive/warehouse, ADDITIONAL_DATA_PATHS</value>
</property>
```

where `ADDITIONAL_DATA_PATHS` is a comma separated list of HDFS paths where Hive data will be stored. The value should be the same value as `sentry.authorization-provider.hdfs-path-prefixes` set in the `hdfs-site.xml` on the NameNode.

Hive log file causes agent failure with very long messages

When the Hive `hive.log.explain.output` property is set to its default value of `true`, very large messages with EXPLAIN outputs can cause the Cloudera Manager Agent to hang or become unstable.

This issue applies only to the following Cloudera Manager versions: 5.3.0, 5.3.1, and 5.3.2.

Workaround:

Set `hive.log.explain.output` to `false`.

Spurious validation warnings and missing validations when multiple Hue Server roles are present

When multiple Hue Server roles are created for a single Hue Service, Cloudera Manager displays a spurious validation warning for Hue with the label "Failed parameter validation." The Cloudera Manager server log also logs an exception message of the form:

```
2015-03-30 17:15:45,077 WARN
ActionablesProvider-0:com.cloudera.cmf.service.ServiceModelValidatorImpl:
Parameter validation failed java.lang.IllegalArgumentException:
There is more than one role with roletype: HUE_SERVER
```

Workaround:

These messages do not correspond to actual validation warnings, and you can ignore them. However, some validations that are normally performed are skipped when this spurious warning is generated. You should then make these validations manually. Specifically, if the Hue back end authentication mechanism is set to LDAP, you should validate the following configurations:

- The Hue LDAP URL property must be properly set
- For CDH 4.4 and earlier, set one (but not both) of the following two Hue properties: **NT Domain** and **LDAP Username Pattern**.
- For CDH 4.5 and later, if the Hue property **Use Search Bind Authentication** is selected, then you must set one of these two Hue properties: **NT Domain** or **LDAP Username Pattern** as described above.

After running the YARN Import MapReduce Configuration wizard, Hive client configuration may still be stale

After running the Import MapReduce Configuration wizard, Hive client configuration may still be stale.

Workaround: Deploy cluster client configuration to update the stale client configuration.

Cloudera Manager 5.3.1 upgrade fails if Spark standalone and Kerberos are configured

CDH upgrade fails if Kerberos is enabled and Spark standalone is installed. Spark standalone doesn't work in a kerberized cluster.

Workaround: To upgrade, remove the Spark standalone service first and then proceed with upgrade.

KMS and Key Trustee ACLs do not work in Cloudera Manager 5.3

ACLs configured for the KMS (File) and KMS (Navigator Key Trustee) services do not work since these services do not receive the values for `hadoop.security.group.mapping` and related group mapping configuration properties.

Workaround:

KMS (File): Add all configuration properties starting with `hadoop.security.group.mapping` from the NameNode `core-site.xml` to the KMS (File) property, **Key Management Server Advanced Configuration Snippet (Safety Valve) for core-site.xml**

KMS (Navigator Key Trustee): Add all configuration properties starting with `hadoop.security.group.mapping` from the NameNode `core-site.xml` to the KMS (Navigator Key Trustee) property, **Key Management Server Proxy Advanced Configuration Snippet (Safety Valve) for core-site.xml**.

Exporting and importing Hue database sometimes times out after 90 seconds

Executing 'dump database' or 'load database' of Hue from Cloudera Manager returns "command aborted because of exception: Command timed-out after 90 seconds". The Hue database can be exported to JSON from within Cloudera Manager. Unfortunately, sometimes the Hue database is quite large and the export times out after 90 seconds.

Workaround: Ignore the timeout. The command should eventually succeed even though Cloudera Manager reports that it timed out.

Changing hostname of key trustee server requires editing the `keytrustee.conf` file

If you change the hostname of your primary or backup server, you will need to edit your `keytrustee.conf` file. This issue typically arises if you replace a primary or backup server with a server having a different hostname. If the same hostname is used on the new server, there will be no issues.

Workaround: Use the same hostname on the replacement server.

Hosts with Impala Llama roles must also have at least one YARN role

When integrated resource management is enabled for Impala, host(s) where the Impala Llama role(s) are running must have at least one YARN role. This is because Llama requires the `topology.py` script from the YARN configuration. If this requirement is not met, you may see errors such as:

```
"Exception running /etc/hadoop/conf.cloudera.yarn/topology.py
java.io.IOException: Cannot run program "/etc/hadoop/conf.cloudera.yarn/topology.py"
```

in the Llama role logs, and Impala queries may fail.

Workaround: Add a YARN gateway role to each Llama host that does not already have at least one YARN role (of any type).

The high availability wizard does not verify that there is a running ZooKeeper service

If one of the following is true:

- 1. ZooKeeper present and not running and the HDFS dependency on ZooKeeper dependency is not set
- 2. ZooKeeper absent

the enable high-availability wizard fails.

Workaround: Before enabling high availability, do the following:

1. Create and start a ZooKeeper service if one doesn't exist.
2. Go to the HDFS service.
3. Click the **Configuration** tab.
4. In the **Service-Wide** category, set the **ZooKeeper Service** property to the ZooKeeper service.
5. Click **Save Changes** to commit the changes.

Cloudera Manager Installation Path A fails on RHEL 5.7 due to PostgreSQL conflict

On RHEL 5.7, `cloudera-manager-installer.bin` fails due to a PostgreSQL conflict if PostgreSQL 8.1 is already installed on your host.

Workaround: Remove PostgreSQL from host and rerun `cloudera-manager-installer.bin`.

Bug in openssl-1.0.1e-15 disrupts SSL communication between Cloudera Manager Agents and CDH services

This issue was observed in SSL-enabled clusters running CentOS 6.4 and 6.5, where the Cloudera Manager Agent failed when trying to communicate with CDH services. You can see the bug report [here](#).

Workaround: Upgrade to openssl-1.0.1e-16.el6_5.7.x86_64.

Cloudera Management Service roles fail to start after upgrade to Cloudera Manager

If you have enabled TLS security for the Cloudera Manager Admin Console before upgrading to Cloudera Manager, after the upgrade, the Cloudera Management Service roles will try to communicate with Cloudera Manager using TLS and will fail to start unless the following SSL properties have been configured.

Hence, if you have the following property enabled in Cloudera Manager, use the workaround below to allow the Cloudera Management Service roles to communicate with Cloudera Manager.

Property	Description
Use TLS Encryption for Admin Console	Select this option to enable TLS encryption between the Server and user's web browser.

Workaround:

1. Open the Cloudera Manager Admin Console and navigate to the **Cloudera Management Service**.
2. Click **Configuration**.
3. In the Search field, type **SSL** to show the SSL properties (found under the **Service-Wide > Security** category).
4. Edit the following SSL properties according to your cluster configuration.

Table 1: Cloudera Management Service SSL Properties

Property	Description
SSL Client Truststore File Location	Path to the client truststore file used in HTTPS communication. The contents of this truststore can be modified without restarting the Cloudera Management Service roles. By default, changes to its contents are picked up within ten seconds.
SSL Client Truststore File Password	Password for the client truststore file.

5. Click **Save Changes**.
6. Restart the Cloudera Management Service. For more information, see [HTTPS Communication in Cloudera Manager](#).

Spurious warning on Accumulo 1.6 gateway hosts

When using the Accumulo shell on a host with only an Accumulo 1.6 Service gateway role, users will receive a warning about failing to create the directory `/var/log/accumulo`. The shell works normally otherwise.

Workaround: The warning is safe to ignore.

Accumulo 1.6 service log aggregation and search does not work

Cloudera Manager log aggregation and search features are incompatible with the log formatting needed by the Accumulo Monitor. Attempting to use either the "Log Search" diagnostics feature or the log file link off of an individual service role's summary page will result in empty search results.

Severity: High

Workaround: Operators can use the Accumulo Monitor to see recent severe log messages. They can see recent log messages below the WARNING level via a given role's process page and can inspect full logs on individual hosts by looking in `/var/log/accumulo`.

Cloudera Manager incorrectly sizes Accumulo Tablet Server max heap size after 1.4.4-cdh4.5.0 to 1.6.0-cdh4.6.0 upgrade

Because the upgrade path from Accumulo 1.4.4-cdh4.5.0 to 1.6.0-cdh4.6.0 involves having both services installed simultaneously, Cloudera Manager will be under the impression that worker hosts in the cluster are oversubscribed on memory and attempt to downsize the max heap size allowed for 1.6.0-cdh4.6.0 Tablet Servers.

Severity: High

Workaround: Manually verify that the Accumulo 1.6.0-cdh4.6.0 Tablet Server max heap size is large enough for your needs. Cloudera recommends you set this value to the sum of 1.4.4-cdh4.5.0 Tablet Server and Logger heap sizes.

Cluster CDH version not configured correctly for package installs

If you have installed CDH as a package, after the install make sure that the cluster CDH version matches the package CDH version. If the cluster CDH version does not match the package CDH version, Cloudera Manager will incorrectly enable and disable service features based on the cluster's configured CDH version.

Workaround:

Manually configure the cluster CDH version to match the package CDH version. Click **ClusterName** > **Actions** > **Configure CDH Version**. In the dialog, Cloudera Manager displays the installed CDH version, and asks for confirmation to configure itself with the new version.

Accumulo installations using LZO do not indicate dependence on the GPL Extras parcel

Accumulo 1.6 installations that use LZO compression functionality do not indicate that LZO depends on the GPL Extras parcel. When Accumulo is configured to use LZO, Cloudera Manager has no way to track that the Accumulo service now relies on the GPL Extras parcel. This prevents Cloudera Manager from warning administrators before they remove the parcel while Accumulo still requires it for proper operation.

Workaround: Check your Accumulo 1.6 service for the configuration changes mentioned in the Cloudera documentation for using Accumulo with CDH prior to removing the GPL Extras parcel. If the parcel is mistakenly removed, reinstall it and restart the Accumulo 1.6 service.

Created pools are not preserved when Dynamic Resource Pools page is used to configure YARN or Impala

Pools created on demand are not preserved when changes are made using the Dynamic Resource Pools page. If the Dynamic Resource Pools page is used to configure YARN and/or Impala services in a cluster, it is possible to specify pool placement rules that create a pool if one does not already exist. If changes are made to the configuration using this page, pools created as a result of such rules are not preserved across the configuration change.

Workaround: Submit the YARN application or Impala query as before, and the pool will be created on demand once again.

User should be prompted to add the AMON role when adding MapReduce to a CDH 5 cluster

When the MapReduce service is added to a CDH 5 cluster, the user is not asked to add the AMON role. Then, an error displays when the user tries to view MapReduce activities.

Workaround: Manually add the AMON role after adding the MapReduce service.

Enterprise license expiration alert not displayed until Cloudera Manager Server is restarted

When an enterprise license expires, the expiration notification banner is not displayed until the Cloudera Manager Server has been restarted (although the expired enterprise features stop working immediately upon license expiration, as expected).

Workaround: None.

HDFS replication does not work from CDH 5 to CDH 4 with different realms

HDFS replication does not work from CDH 5 to CDH 4 with different realms. This is because authentication fails for services in a non-default realm via the WebHdfs API due to a JDK bug. This has been fixed in JDK6-u34 (b03)) and in JDK7.

Workaround: Use JDK 7 or upgrade JDK6 to at least version u34.

Configurations for decommissioned roles not migrated from MapReduce to YARN

When the **Import MapReduce Configuration** wizard is used to import MapReduce configurations to YARN, decommissioned roles in the MapReduce service do not cause the corresponding imported roles to be marked as decommissioned in YARN.

Workaround: Delete or decommission the roles in YARN after running the import.

The HDFS command Roll Edits does not work in the UI when HDFS is federated

The HDFS command Roll Edits does not work in the Cloudera Manager UI when HDFS is federated because the command doesn't know which nameservice to use.

Workaround: Use the API, not the Cloudera Manager UI, to execute the Roll Edits command.

Cloudera Manager reports a confusing version number if you have oozie-client, but not oozie installed on a CDH 4.4 node

In CDH versions before 4.4, the metadata identifying Oozie was placed in the client, rather than the server package. Consequently, if the client package is not installed, but the server is, Cloudera Manager will report Oozie has been present but as coming from CDH 3 instead of CDH 4.

Workaround: Either install the oozie-client package, or upgrade to at least CDH 4.4. Parcel based installations are unaffected.

Cloudera Manager doesn't work with CDH 5.0.0 Beta 1

When you upgrade from Cloudera Manager 5.0.0 Beta 1 with CDH 5.0.0 Beta 1 to Cloudera Manager 5.0.0 Beta 2, Cloudera Manager won't work with CDH 5.0.0 Beta 1 and there's no notification of that fact.

Workaround: None. Do a new installation of CDH 5.0.0 Beta 2.

On CDH 4.1 secure clusters managed by Cloudera Manager 4.8.1 and higher, the Impala Catalog server needs advanced configuration snippet update

Impala queries fail on CDH 4.1 when Hive "Bypass Hive Metastore Server" option is selected.

Workaround: Add the following to Impala catalog server advanced configuration snippet for `hive-site.xml`, replacing `Hive_Metastore_Server_Host` with the host name of your Hive Metastore Server:

```
<property>
<name>hive.metastore.local</name>
<value>false</value>
</property>
<property>
<name>hive.metastore.uris</name>
<value>thrift://Hive_Metastore_Server_Host:9083</value>
</property>
```

Rolling Upgrade to CDH 5 is not supported.

Rolling upgrade between CDH 4 and CDH 5 is not supported. Incompatibilities between major versions means rolling restarts are not possible. In addition, rolling upgrade will *not* be supported from CDH 5.0.0 Beta 1 to any later releases, and may not be supported between *any future beta versions* of CDH 5 and the General Availability release of CDH 5.

Workaround: None.

Error reading .zip file created with the Collect Diagnostic Data command.

After collecting Diagnostic Data and using the Download Diagnostic Data button to download the created zip file to the local system, the zip file cannot be opened using the FireFox browser on a Macintosh. This is because the zip file is created as a Zip64 file, and the unzip utility included with Macs does not support Zip64. The zip utility must be version 6.0 or later. You can determine the zip version with `unzip -v`.

Workaround: Update the unzip utility to a version that supports Zip64.

Enabling wildcarding in a secure environment causes NameNode to fail to start.

In a secure cluster, you cannot use a wildcard for the NameNode's RPC or HTTP bind address, or the NameNode will fail to start. For example, `dfs.namenode.http-address` must be a real, routable address and port, not `0.0.0.0.port`. In Cloudera Manager, the "Bind NameNode to Wildcard Address" property must not be enabled. This should affect you only if you are running a secure cluster and your NameNode needs to bind to multiple local addresses.

Bug: [HDFS-4448](#)

Severity: Medium

Workaround: Disable the "Bind NameNode to Wildcard Address" property found under the Configuration tab for the NameNode role group.

After JobTracker failover, complete jobs from the previous active JobTracker are not visible.

When a JobTracker failover occurs and a new JobTracker becomes active, the new JobTracker UI does not show the completed jobs from the previously active JobTracker (that is now the standby JobTracker). For these jobs the "Job Details" link does not work.

Severity: Med

Workaround: None.

After JobTracker failover, information about rerun jobs is not updated in Activity Monitor.

When a JobTracker failover occurs while there are running jobs, jobs are restarted by the new active JobTracker by default. For the restarted jobs the Activity Monitor will not update the following: 1) The start time of the restarted job will remain the start time of the original job. 2) Any Map or Reduce task that had finished before the failure happened will not be updated with information about the corresponding task that was rerun by the new active JobTracker.

Severity: Med

Workaround: None.

Installing on AWS, you must use private EC2 hostnames.

When installing on an AWS instance, and adding hosts using their public names, the installation will fail when the hosts fail to heartbeat.

Severity: Med

Workaround:

Use the Back button in the wizard to return to the original screen, where it prompts for a license.

Rerun the wizard, but choose "Use existing hosts" instead of searching for hosts. Now those hosts show up with their internal EC2 names.

Continue through the wizard and the installation should succeed.

If HDFS uses Quorum-based Storage without HA enabled, the SecondaryNameNode cannot checkpoint.

If HDFS is set up in non-HA mode, but with Quorum-based storage configured, the `dfs.namenode.edits.dir` is automatically configured to the Quorum-based Storage URI. However, the SecondaryNameNode cannot currently read the edits from a Quorum-based Storage URI, and will be unable to do a checkpoint.

Severity: Medium

Workaround: Add to the NameNode's advanced configuration snippet the `dfs.namenode.edits.dir` property with both the value of the Quorum-based Storage URI as well as a local directory, and restart the NameNode. For example,

```
<property> <name>dfs.namenode.edits.dir</name>
<value>qjournal://jn1HostName:8485;jn2HostName:8485;jn3HostName:8485/journalhdfs1,file:///dfs/edits</value>
</property>
```

Changing the rack configuration may temporarily cause mis-replicated blocks to be reported.

A rack re-configuration will cause HDFS to report mis-replicated blocks until HDFS rebalances the system, which may take some time. This is a normal side-effect of changing the configuration.

Severity: Low

Workaround: None

Cannot use '/' as a mount point with a Federated HDFS Nameservice.

A Federated HDFS Service doesn't support nested mount points, so it is impossible to mount anything at '/'. Because of this issue, the root directory will always be read-only, and any client application that requires a writeable root directory will fail.

Severity: Low

Workaround:

1. In the CDH 4 HDFS Service > Configuration tab of the Cloudera Manager Admin Console, search for "nameservice".
2. In the Mountpoints field, change the mount point from "/" to a list of mount points that are in the namespace that the Nameservice will manage. (You can enter this as a comma-separated list - for example, "/hbase, /tmp, /user" or by clicking the plus icon to add each mount point in its own field.) You can determine the list of mount points by running the command `hadoop fs -ls /` from the CLI on the NameNode host.

Historical disk usage reports do not work with federated HDFS.

Severity: Low

Workaround: None.

(CDH 4 only) Activity monitoring does not work on YARN activities.

Activity monitoring is not supported for YARN in CDH 4.

Severity: Low

Workaround: None

HDFS monitoring configuration applies to all Nameservices

The monitoring configurations at the HDFS level apply to all Nameservices. So, if there are two federated Nameservices, it's not possible to disable a check on one but not the other. Likewise, it's not possible to have different thresholds for the two Nameservices.

Severity: Low

Workaround: None

Supported and Unsupported Replication Scenarios and Limitations

See [Data Replication](#).

Restoring snapshot of a file to an empty directory does not overwrite the directory

Restoring the snapshot of an HDFS file to an HDFS path that is an empty HDFS directory (using the Restore As action) will result in the restored file present inside the HDFS directory instead of overwriting the empty HDFS directory.

Workaround: None.

HDFS Snapshot appears to fail if policy specifies duplicate directories.

In an HDFS snapshot policy, if a directory is specified more than once, the snapshot appears to fail with an error message on the Snapshot page. However, in the HDFS Browser, the snapshot is shown as having been created successfully.

Severity: Low

Workaround: Remove the duplicate directory specification from the policy.

Hive replication fails if "Force Overwrite" is not set.

The Force Overwrite option, if checked, forces overwriting data in the target metastore if there are incompatible changes detected. For example, if the target metastore was modified and a new partition was added to a table, this option would force deletion of that partition, overwriting the table with the version found on the source. If the Force Overwrite option is not set, recurring replications may fail.

Severity: Med

Workaround: Set the Force Overwrite option.

Oozie health bad when Oozie is HA, cluster is kerberized, and Cloudera Manager and CDH are upgraded

Oozie health will go bad if high availability is enabled in a kerberized cluster with Cloudera Manager 5.0 and CDH 5.0 and Cloudera Manager and CDH are then upgraded to 5.1 or higher.

Workaround: Disable Oozie HA and then re-enable HA again.

Issues Fixed in Cloudera Manager 5

The following sections describe issues fixed in each Cloudera Manager 5 release.

Issues Fixed in Cloudera Manager 5.3.4

Slowness observed when accessing replication page in Cloudera Manager

When you access the replication page in Cloudera Manager, the page responds slowly due to a large number of replication history records. The number of displayed historical records has been changed from 100 to 20.

Cloudera Manager overwrites the `krb5.conf` file after disabling management by Cloudera Manager

When a cluster has been configured to enable Kerberos by clicking the **Manage `krb5.conf` through Cloudera Manager** button, Cloudera Manager writes out a `krb5.conf` file. If a user subsequently disables this feature by disabling the **Manage `krb5.conf` through Cloudera Manager** option and then restarting Cloudera Manager and the Agent, Cloudera Manager overwrites the existing `krb5.conf` file.

Header injection vulnerability in internal logging call

An internal call to a servlet with a malformed logger name created information that could be used in a cross-site scripting attack.

Graph for "Total Containers Running Across NodeManagers" displays high values

The graph **Total Containers Running Across NodeManagers**, which displays on the YARN status page, displayed incorrect high values.

Clicking the "Revert to default" icon stores the default value as a user-defined value in the new configuration pages

Saving an empty configuration value causes the value to be replaced by the default value. The empty value is now saved instead of the default value.

Some Operational Reports do not return results

The following reports do not return results:

- Overpopulated Directories
- Large Directories
- Custom Reports where the **Replication** parameter is set to 0.

Xalan has been upgraded to version 2.7.2

To address a vulnerability identified by [CVE-2014-0107](#), Xalan has been updated to version 2.7.2.

Setting threshold values of -1 or -2 not accepted in new configuration layout pages

When you set threshold values in various properties, you could not select **Specify** and then enter -1 to indicate "Any" or -2 to indicate "Never". You can now enter -1 or -2.

Rolling upgrade arguments are reversed

The following arguments for rolling upgrade are reversed:

- Sleep seconds
- Failure threshold

Issues Fixed in Cloudera Manager 5.3.3

`hive.metastore.client.socket.timeout` default value changed to 60

The default value of the **`hive.metastore.client.socket.timeout`** property has changed to 60 seconds.

SSL Enablement property name changes

The property **`hadoop.ssl.enabled`** is deprecated. Cloudera Manager has been updated to use either **`dfs.http.policy`** or **`yarn.http.policy`** properties instead.

Changing the Service Monitor Client Config Overrides property requires restart

Cloudera Manager no longer requires you to restart your cluster after changing the **Service Monitor Client Config Overrides** property for a service.

Cluster name changed from specified name to "cluster" after upgrade

After updating to a new release, Cloudera Manager replaces the specified cluster name with `cluster`. Cloudera Manager now uses the correct cluster name.

Configuration without `host_id` in upgrade DDL causes upgrade problems

A client configuration row in the database DDL did not set `host_id`, causing upgrade problems. Cloudera Manager now catches this condition before upgrading.

`hive.log.explain.output` property is hidden

The property `hive.log.explain.output` is known to create instability of Cloudera Manager Agents in some specific circumstances, especially when the hive queries generate extremely large EXPLAIN outputs. Therefore, the property has been hidden from the Cloudera Manager configuration screens. You can still configure the property through the use of advanced configuration snippets.

Slow staleness calculation can lead to ZooKeeper data loss when new servers are added

In Cloudera Manager 5.x, starting new ZooKeeper Servers shortly after adding them can cause ZooKeeper data loss when the number of new servers exceeds the number of old servers.

Issues Fixed in Cloudera Manager 5.3.2

The Review Changes page sometimes hangs

The **Review Changes** page hangs due to the inability to handle the "File missing" scenario.

High volume of TGT events against AD server with "bad token" messages

A fix has been made to how Kerberos credential caching is handled by management services, resulting in a reduction in the number of Kerberos Ticket Granting Ticket (TGT) requests from the cluster to a KDC. This would have been noticed as "Bad Token" messages being seen in high volume in KDC logging and unnecessarily causing re-authentication by management services.

Accumulo missing `kinit` when running with Kerberos

Cloudera Manager is unable to run Accumulo when `hostname` command doesn't return FQDN of hosts.

HiveServer2 leaks threads when using impersonation

For CDH 5.3 and higher, Cloudera Manager will configure HiveServer2 to use the HDFS cache even when impersonation is on. For earlier CDH, there were bugs with the cache when impersonation was in use, so it is still disabled.

Deploying client configurations fails if there are dead hosts present in the cluster

If there are hosts in the cluster where the Cloudera Manager agent heartbeat is not working, then deploying client configurations doesn't work. Starting with Cloudera Manager 5.3.2, such hosts are ignored while deploying client configurations. When the issues with the host are fixed, Cloudera Manager will show those hosts as having stale client configurations, at which point you can redeploy them.

Health test monitors free space available on the wrong filesystem

The Cloudera Manager Health Test to monitor free space available for the Cloudera Manager Agent's process directory monitors space on the wrong filesystem. It should monitor the `tmpfs` that the Cloudera Manager Agent creates, but instead monitors the Cloudera Manager Agent working directory.

Starting ZooKeeper Servers from Service or Instance page fails

Stopped ZooKeeper servers cannot be started from the Service or Instance page, but only from the Role page of the server using the `start` action for the role.

Flume Metrics page doesn't render agent metrics

Starting in Cloudera Manager 5.3, some or all Flume component data was missing from the Flume Metrics Details page.

Broken link to help pages on Chart Builder page

The help icon (question mark) on the Chart Builder page returns a 404 error.

Import MapReduce configurations to YARN now handles NodeManager vcores and memory

Running the wizard to import MapReduce configurations to YARN will now populate `yarn.nodemanager.resource.cpu-vcores` and `yarn.nodemanager.resource.memory-mb` correctly based on equivalent MapReduce configuration.

Issues Fixed in Cloudera Manager 5.3.1

Deploy client configuration no longer fails after 60 seconds

When configuring a gateway role on a host that already contains a role of the same type—for example, an HDFS gateway on a DataNode—the deploy client configuration command no longer fails after 60 seconds.

service cloudera-scm-server force_start now works

After deleting services, the Cloudera Manager Server log no longer contains foreign key constraint failure exceptions

When using Isilon, Cloudera Manager now sets mapred_submit_replication correctly

When EMC Isilon storage is used, there is no DataNode, so you cannot set `mapred_submit_replication` to a number smaller than or equal to the number of DataNodes in the network. Cloudera Manager now does the following when setting `mapred_submit_replication`:

- If using HDFS, sets to a minimum of 1 and issues a warning when greater than the number of DataNodes
- If using Isilon, sets to 1 and does not check against the number of DataNodes

The Cloudera Manager Agent now sets the file descriptor ulimit correctly on Ubuntu

During upgrade, bootstrapping the standby NameNode step no longer fails with standby NameNode connection refused when connecting to active NameNode

Deploy krb5.conf now also deploys it on hosts with Cloudera Management Service roles

Cloudera Manager allows upgrades to unknown CDH maintenance releases

Cloudera Manager 5.3.0 supports any CDH release less than or equal to 5.3, even if the release did not exist when Cloudera Manager 5.3.0 was released. For packages, you cannot currently use the upgrade wizard to upgrade to such a release. This release adds a custom CDH field for the package case, where you can type in a version that did not exist at the time of the Cloudera Manager release.

impalad memory limit units error in EnableLlamaRMCommand

The `EnableLlamaRMCommand` sets the value of the `impalad` memory limit to equal the NM container memory value. But the latter is in MB, and the former is in bytes. Previously, the command did not perform the conversion; this has been fixed.

Running MapReduce v2 jobs are now visible using the Application Master view

In the Application view, selecting **Application Master** for a MRv2 job previously resulted in no action.

Deleting services no longer results in foreign key constraint exceptions

The Cloudera Manager Server log previously showed several foreign key constraint exceptions that were associated with deleted services. This has been fixed.

HiveServer2 keystore and LDAP group mapping passwords are no longer exposed in client configuration files

The HiveServer2 keystore password and LDAP group mapping passwords were emitted into the client configuration files. This exposed the passwords in plain text in a world-readable file. This has been fixed.

A cross-site scripting vulnerability in Cloudera Management Service web UIs fixed

The high availability wizard now sets the HDFS dependency on ZooKeeper

Workaround: Before enabling high availability, do the following:

1. Create and start a ZooKeeper service if one does not exist.
2. Go to the HDFS service.
3. Click the **Configuration** tab.
4. In the **Service-Wide** category, set the **ZooKeeper Service** property to the ZooKeeper service.

5. Click **Save Changes** to commit the changes.

BDR no longer assumes superuser is common if clusters have the same realm

If source and destination clusters are in the same Kerberos realm, Cloudera Manager assumed that superuser of the destination is also the superuser on the source cluster. However, HDFS can be configured so that this is not the case.

Issues Fixed in Cloudera Manager 5.3.0

Deploy client configuration across cluster after upgrade from Cloudera Manager 4.x to 5.3

Following a 4.x -> 5.3 upgrade, you must deploy client configuration across the entire cluster before deleting any gateway roles, any services, or any hosts. Otherwise the existing 4.x client configurations may be left registered and orphaned on the hosts where they were deployed, requiring you to manually intervene to delete them.

Spark and Spark (standalone) services fail to start if you upgrade to CDH 5.2.x parcels from an older CDH package

Spark and Spark standalone services fail to start if you upgrade to CDH 5.2.x parcels from an older CDH package.

Workaround: After upgrading rest of the services, uninstall the old CDH packages, and then start the Spark service.

Cluster installation with CDH 4.1 and Impala fails

In Cloudera Manager 5.0, installing a new cluster through the wizard with CDH 4.1 and Impala fails with the following error message, "dfs.client.use.legacy.blockreader.local is not enabled."

Workaround: Perform one of the following:

1. Use CDH 4.2 or higher, or
2. Install all desired services except Impala in your initial cluster setup. From the home page, use the drop-down menu near the cluster name and select **Configure CDH Version**. Confirm the version, then add Impala.

Fixed MapReduce Usage by User reports when using an Oracle database backend

Setting the default umask in HDFS fails in new configuration layout

Setting the default umask in the HDFS configuration section to 002 in the new configuration layout displays an error: "Could not parse: Default Umask : Could not parse parameter 'dfs_umaskmode'. Was expecting an octal value with a leading 0. Input: 2", preventing the change from being submitted.

Workaround: Submit the change using the classic configuration layout.

Enabling Integrated Resource Management for Impala sets Impala Daemon Memory Limit Incorrectly

The Enable Integrated Resource Management command for Impala (available from the **Actions** pull-down menu on the Impala service page) sets the Impala Daemon Memory Limit to an unusably small value. This can cause Impala queries to fail.

Workaround 1: Upgrade to Cloudera Manager 5.3.

Workaround 2:

1. Run the Enable Integrated Resource Management wizard up to the **Restart Cluster** step. Do not click **Restart Now**.
2. Click on the **leave this wizard** link to exit the wizard without restarting the cluster.
3. Go to the YARN service page. Click **Configuration**, expand the category **NodeManager Default Group**, and click **Resource Management**.
4. Note the value of the Container Memory property.
5. Go to the Impala service page and click **Configuration**. Type `impala daemon memory limit` into the search box.
6. Set the value of the Impala Daemon Memory Limit property to the value noted in step 4 above.
7. Restart the cluster.

Rolling restart and upgrade of Oozie fails if there is a single Oozie server

Rolling restart and upgrade of Oozie fails if there is only a single Oozie server. Cloudera Manager will show the error message "There is already a pending command on this role."

Workaround: If you have a single Oozie server, do a normal restart.

Allow "Started but crashed" processes to be restarted by a Start command

In Cloudera Manager 5.3, it is now possible to restart a crashed process with the **Start** command and not just the **Restart** command.

Add dependency from Agent to Daemons package to yum

In Cloudera Manager 5.3, an explicit dependency has been added from the Agent package to the Daemons package so that upgrading Cloudera Manager 5.2.0 or later to Cloudera Manager 5.3 causes the agent to be upgraded as well. Previously, the Cloudera Manager installer always installed both packages, but this is now enforced at the package dependency level as well.

Setting the default umask in HDFS fails in new configuration layout

Setting the default umask in the HDFS configuration section to 002 in the new configuration layout displays an error: "Could not parse: Default Umask : Could not parse parameter 'dfs_umaskmode'. Was expecting an octal value with a leading 0. Input: 2", preventing the change from being submitted.

Workaround: Submit the change using the classic configuration layout.

Issues Fixed in Cloudera Manager 5.2.5

Slow staleness calculation can lead to ZooKeeper data loss when new servers are added

In Cloudera Manager 5, starting new ZooKeeper Servers shortly after adding them can cause ZooKeeper data loss when the number of new servers exceeds the number of old servers.

Permissions set incorrectly on YARN Keytab files

Permissions on YARN Keytab files for NodeManager were set incorrectly to allow read access to any user.

Issues Fixed in Cloudera Manager 5.2.2

Impalad memory limit units error in EnableLlamaRMCommand has been fixed

The EnableLlamaRMCommand sets the value of the impalad memory limit to equal the NM container memory value. But the latter is in MB, and the former is in bytes. Previously, the command did not perform the conversion; this has been fixed.

Fixed MapReduce Usage by User reports when using an Oracle database backend

HiveServer2 keystore and LDAP group mapping passwords are no longer exposed in client configuration files

The HiveServer2 keystore password and LDAP group mapping passwords were emitted into the client configuration files. This exposed the passwords in plain text in a world-readable file. This has been fixed.

Running MapReduce v2 jobs are now visible using the Application Master view

In the Application view, selecting **Application Master** for a MRv2 job previously resulted in no action.

Deleting services no longer results in foreign key constraint exceptions

The Cloudera Manager Server log previously showed several foreign key constraint exceptions that were associated with deleted services. This has been fixed.

Issues Fixed in Cloudera Manager 5.2.1

"POODLE" vulnerability on SSL/TLS enabled ports

The POODLE (Padding Oracle On Downgraded Legacy Encryption) attack takes advantage of a cryptographic flaw in the obsolete SSLv3 protocol, after first forcing the use of that protocol. The only solution is to disable SSLv3 entirely. This requires changes across a wide variety of components of CDH and Cloudera Manager in 5.2.0 and all earlier versions. Cloudera Manager 5.2.1 provides these changes for Cloudera Manager 5.2.0

deployments. All Cloudera Manager 5.2.0 users should upgrade to 5.2.1 as soon as possible. For more information, see the [Cloudera Security Bulletin](#).

[Can use the log4j advanced configuration snippet to override the default audit logging configuration even if not using Navigator](#)

In Cloudera Manager 5.2.0 only, it was not possible to use the log4j advanced configuration snippet to override the default audit logging configuration when Navigator was not being used.

[Cloudera Manager now collects metrics for CDH 5.0 DataNodes and NameNodes](#)

A number of NameNode and DataNode charts show no data and a number of NameNode and DataNode health checks show unknown results. Metric collection for CDH 5.1 roles is unaffected.

Workaround: None.

[The Reports Manager and Event Server Thrift servers no longer crash on HTTP requests](#)

HTTP queries against the Reports Manager and Event Server Thrift server would earlier cause it to crash with out-of-memory exception.

[Replication commands now use the correct JAVA_HOME if an override has been provided for it](#)

[ZooKeeper connection leaks from HBase clients in Service Monitor have been fixed](#)

[When a parcel is activated, user home directories are now created with umask 022 instead of using the "useradd" default 077](#)

Issues Fixed in Cloudera Manager 5.2.0

[Alternatives database points to client configurations of deleted service](#)

In the past, if you created a service, deployed its client configurations, and then deleted that service, the client configurations lived in the alternative database, with a possibly high priority, until cleaned up manually. Now, for a given "alternatives path" (for example `/etc/hadoop/conf`) if there exist both "live" client configurations (ones that would be pushed out with deploy client configurations for active services) and ones that have been "orphaned" client configurations (the service they correspond to has been deleted), the orphaned ones will be removed from the alternatives database. In other words, to trigger cleanup of client configurations associated with a deleted service you must create a service to replace it.

[The YARN property ApplicationMaster Max Retries has no effect in CDH 5](#)

The issue arises because `yarn.resourcemanager.am.max-retries` was replaced with `yarn.resourcemanager.am.max-attempts`.

Workaround:

1. Add the following to **ResourceManager Advanced Configuration Snippet for yarn-site.xml**, replacing `MAX_ATTEMPTS` with the desired maximum number of attempts:

```
<property>
<name>yarn.resourcemanager.am.max-attempts</name><value>MAX_ATTEMPTS</value>
</property>
```

2. Restart the ResourceManager(s) to pick up the change.

[The Spark History Server does not start when Kerberos authentication is enabled.](#)

The Spark History Server does not start when managed by a Cloudera Manager 5.1 instance when Kerberos authentication is enabled.

Workaround:

1. Go to the Spark service.
2. Expand the **Service-Wide > Advanced** category.

3. Add the following configuration to the **History Server Environment Advanced Configuration Snippet** property:

```
SPARK_HISTORY_OPTS=-Dspark.history.kerberos.enabled=true \  
-Dspark.history.kerberos.principal=principal \  
-Dspark.history.kerberos.keytab=keytab
```

where *principal* is the name of the Kerberos principal to use for the History Server, and *keytab* is the path to the principal's keytab file on the local filesystem of the host running the History Server.

Hive replication issue with TLS enabled

Hive replication will fail when the source Cloudera Manager instance has TLS enabled, even though the required certificates have been added to the target Cloudera Manager's trust store.

Workaround: Add the required Certificate Authority or self-signed certificates to the default Java trust store, which is typically a copy of the cacerts file named `jssecacerts` in the `$JAVA_HOME/jre/lib/security/` path of your installed JDK. Use `keytool` to import your private CA certificates into the `jssecacert` file.

The Spark Upload Jar command fails in a secure cluster

The Spark **Upload Jar** command fails in a secure cluster.

Workaround: To run Spark on YARN, manually upload the Spark assembly jar to HDFS `/user/spark/share/lib`. The Spark assembly jar is located on the local filesystem, typically in `/usr/lib/spark/assembly/lib` or `/opt/cloudera/parcels/CDH/lib/spark/assembly/lib`.

Clients of the JobHistory Server Admin Interface Require Advanced Configuration Snippet

Clients of the JobHistory server administrative interface, such as the `mapred hsadmin` tool, may fail to connect to the server when run on hosts other than the one where the JobHistory server is running.

Workaround: Add the following to both the **MapReduce Client Advanced Configuration Snippet for mapred-site.xml** and the **Cluster-wide Advanced Configuration Snippet for core-site.xml**, replacing `JOBHISTORY_SERVER_HOST` with the hostname of your JobHistory server:

```
<property>  
<name>mapreduce.history.admin.address</name>  
<value>JOBHISTORY_SERVER_HOST:10033</value>  
</property>
```

Fixed Issues in Cloudera Manager 5.1.5

Slow staleness calculation can lead to ZooKeeper data loss when new servers are added

In Cloudera Manager 5, starting new ZooKeeper Servers shortly after adding them can cause ZooKeeper data loss when the number of new servers exceeds the number of old servers.

Permissions set incorrectly on YARN Keytab files

Permissions on YARN Keytab files for NodeManager were set incorrectly to allow read access to any user.

Fixed Issues in Cloudera Manager 5.1.4

"POODLE" vulnerability on SSL/TLS enabled ports

The POODLE (Padding Oracle On Downgraded Legacy Encryption) attack takes advantage of a cryptographic flaw in the obsolete SSLv3 protocol, after first forcing the use of that protocol. The only solution is to disable SSLv3 entirely. This requires changes across a wide variety of components of CDH and Cloudera Manager. Cloudera Manager 5.1.4 provides these changes for Cloudera Manager 5.1.x deployments. All Cloudera Manager 5.1.x users should upgrade to 5.1.4 as soon as possible. For more information, see the [Cloudera Security Bulletin](#).

Issues Fixed in Cloudera Manager 5.1.3

Improved speed and heap usage when deleting hosts on cluster with long history

Speed and heap usage have been improved when deleting hosts on clusters that have been running for a long time.

When there are multiple clusters, each cluster's topology files and validation for legal topology is limited to hosts in that cluster

When there are multiple clusters, each cluster's topology files and validation for legal topology is limited to hosts in that cluster. Most commands will now fail up front if the cluster's topology is invalid.

The size of the statement cache has been reduced for Oracle databases

For users of Oracle databases, the size of the statement cache has been reduced to help with memory consumption.

Improvements to memory usage of "cluster diagnostics collection" for large clusters.

Memory usage of "cluster diagnostics collection" has been improved for large clusters.

Issues Fixed in Cloudera Manager 5.1.2

If a NodeManager that is used as ApplicationMaster is decommissioned, YARN jobs will hang

Jobs can hang on NodeManager decommission due to a race condition when continuous scheduling is enabled.

Workaround:

1. Go to the YARN service.
2. Expand the **ResourceManager Default Group > Resource Management** category.
3. Uncheck the **Enable Fair Scheduler Continuous Scheduling** checkbox.
4. Click **Save Changes** to commit the changes.
5. Restart the YARN service.

Could not find a healthy host with CDH 5 on it to create HiveServer2 error during upgrade

When upgrading from CDH 4 to CDH 5, if no parcel is active then the error message "Could not find a healthy host with CDH5 on it to create HiveServer2" displays. This can happen when transitioning from packages to parcels, or if you explicitly deactivate the CDH 4 parcel (which is not necessary) before upgrade.

Workaround: Wait 30 seconds and retry the upgrade.

AWS installation wizard requires Java 7u45 to be installed on Cloudera Manager Server host

Cloudera Manager 5.1 installs Java 7u55 by default. However, the AWS installation wizard does not work with Java 7u55 due to a bug in the jClouds version packaged with Cloudera Manager.

Workaround:

1. Stop the Cloudera Manager Server.

```
$ sudo service cloudera-scm-server stop
```

2. Uninstall Java 7u55 from the Cloudera Manager Server host.
3. Install Java 7u45 (which you can download from <http://www.oracle.com/technetwork/java/javase/downloads/java-archive-downloads-javase7-521261.html#jdk-7u45-oth-JPR>) on the Cloudera Manager Server host.
4. Start the Cloudera Manager Server.

```
$ sudo service cloudera-scm-server start
```

5. Run the AWS installation wizard.

- **Note:** Due to a bug in Java 7u45 (http://bugs.java.com/bugdatabase/view_bug.do?bug_id=8014618), SSL connections between the Cloudera Manager Server and Cloudera Manager Agents and between the Cloudera Management Service and CDH processes break intermittently. If you do not have SSL enabled on your cluster, there is no impact.

The YARN property ApplicationMaster Max Retries has no effect in CDH 5

The issue arises because `yarn.resourcemanager.am.max-retries` was replaced with `yarn.resourcemanager.am.max-attempts`.

Workaround:

1. Add the following to **ResourceManager Advanced Configuration Snippet for yarn-site.xml**, replacing `MAX_ATTEMPTS` with the desired maximum number of attempts:

```
<property>
<name>yarn.resourcemanager.am.max-attempts</name><value>MAX_ATTEMPTS</value>
</property>
```

2. Restart the ResourceManager(s) to pick up the change.

(BDR) Replications can be affected by other replications or commands running at the same time

Replications can be affected by other replications or commands running at the same time, causing replications to fail unexpectedly or even be silently skipped sometimes. When this occurs, a `StaleObjectException` is logged to the Cloudera Manager logs. This is known to occur even with as few as four replications starting at the same time.

Issues Fixed in Cloudera Manager 5.1.1

Checking "Install Java Unlimited Strength Encryption Policy Files" During Add Cluster or Add/Upgrade Host Wizard on RPM based distributions if JDK 7 or above is pre-installed will cause Cloudera Manager and CDH to fail

If you have manually installed Oracle's official JDK 7 or 8 rpm on a host (or hosts), and check the **Install Java Unlimited Strength Encryption Policy Files** checkbox in the Add Cluster or Add Host wizard when installing Cloudera Manager on that host (or hosts), or when upgrading Cloudera Manager to 5.1, Cloudera Manager installs JDK 6 policy files, which will prevent any Java programs from running against that JDK. Additionally, if this situation does apply, Cloudera Manager/CDH will also choose that particular Java as the default to run against, meaning that Cloudera Manager/CDH fail to start, throwing the following message in logs: `Caused by: java.lang.SecurityException: The jurisdiction policy files are not signed by a trusted signer!`.

Workaround: Do not select the **Install Java Unlimited Strength Encryption Policy Files** checkbox during the aforementioned wizards. Instead download and install them manually, following the instructions on Oracle's website.

- JDK 7 Instructions:
<http://www.oracle.com/technetwork/java/javase/downloads/jce-7-download-432124.html>
- JDK 8 Instructions:
<http://www.oracle.com/technetwork/java/javase/downloads/jce8-download-2133166.html>

■ **Note:** To return to the default limited strength files, reinstall the original Oracle rpm:

- `yum -y yum reinstall jdk`
- `zypper - zypper in -f jdk`
- `rpm -rpm -iv --replacepkgs filename`, where *filename* is `jdk-7u65-linux-x64.rpm` or `jdk-8u11-linux-x64.rpm`

Issues Fixed in Cloudera Manager 5.1.0

- **Important:** Cloudera Manager 5.1.0 is no longer available for download from the Cloudera website or from archive.cloudera.com due to the JCE policy file issue described in the [Issues Fixed in Cloudera Manager 5.1.1](#) on page 182 section of the Release Notes. The download URL at archive.cloudera.com for Cloudera Manager 5.1.0 now forwards to Cloudera Manager 5.1.1 for the RPM-based distributions for Linux RHEL and SLES.

Changes to property for `yarn.nodemanager.remote-app-log-dir` are not included in the JobHistory Server `yarn-site.xml` and Gateway `yarn-site.xml`

When "Remote App Log Directory" is changed in YARN configuration, the property `yarn.nodemanager.remote-app-log-dir` are not included in the JobHistory Server `yarn-site.xml` and Gateway `yarn-site.xml`.

Workaround: Set **JobHistory Server Advanced Configuration Snippet (Safety Valve) for `yarn-site.xml`** and **YARN Client Advanced Configuration Snippet (Safety Valve) for `yarn-site.xml`** to:

```
<property>
<name>yarn.nodemanager.remote-app-log-dir</name>
<value>/path/to/logs</value>
</property>
```

Secure CDH 4.1 clusters can't have Hue and Impala share the same Hive

In a secure CDH 4.1 cluster, Hue and Impala cannot share the same Hive instance. If "Bypass Hive Metastore Server" is disabled on the Hive service, then Hue will not be able to talk to Hive. Conversely, if "Bypass Hive Metastore" enabled on the Hive service, then Impala will have a validation error.

Severity: High

Workaround: Upgrade to CDH 4.2.

The command history has an option to select the number of commands, but doesn't always return the number you request

Workaround: None.

Hue doesn't support YARN ResourceManager High Availability

Workaround: Configure the Hue Server to point to the active ResourceManager:

1. Go to the Hue service.
2. Click the **Configuration** tab.
3. Click **Hue Server Default Group > Advanced**.
4. In the **Hue Server Advanced Configuration Snippet for `hue_safety_valve_server.ini`** field, add the following:

```
[hadoop]
[[ yarn_clusters ]]
[[[default]]]
resourcemanager_host=<hostname of active ResourceManager>
resourcemanager_api_url=http://<hostname of active resource manager>:<web port of active resource manager>
proxy_api_url=http://<hostname of active resource manager>:<web port of active resource manager>
```

The default web port of Resource Manager is 8088.

5. Click **Save Changes** to have these configurations take effect.
6. Restart the Hue service.

Cloudera Manager does not support encrypted shuffle.

Encrypted shuffle has been introduced in CDH 4.1, but it is not currently possible to enable it through Cloudera Manager.

Severity: Medium

Workaround: None.

[Hive CLI does not work in CDH 4 when "Bypass Hive Metastore Server" is enabled](#)

Hive CLI does not work in CDH 4 when "Bypass Hive Metastore Server" is enabled.

Workaround: Configure Hive and disable the "Bypass Hive Metastore Server" option.

Alternatively, an approach can be taken that will cause the "Hive Auxiliary JARs Directory" to not work, but will enable basic Hive commands to work. Add the following to "Gateway Client Environment Advanced Configuration Snippet for hive-env.sh," then re-deploy the Hive client configuration:

```
HIVE_AUX_JARS_PATH=""
AUX_CLASSPATH=/usr/share/java/mysql-connector-java.jar:/usr/share/java/oracle-connector-java.jar:$(find /usr/share/cmf/lib/postgresql-jdbc.jar 2> /dev/null | tail -n 1)
```

[Incorrect Absolute Path to topology.py in Downloaded YARN Client Configuration](#)

The downloaded client configuration for YARN includes the `topology.py` script. The location of this script is given by the `net.topology.script.file.name` property in `core-site.xml`. But the `core-site.xml` file downloaded with the client configuration has an incorrect absolute path to `/etc/hadoop/...` for `topology.py`. This can cause clients that run against this configuration to fail (including Spark clients run in `yarn-client` mode, as well as YARN clients).

Workaround: Edit `core-site.xml` to change the value of the `net.topology.script.file.name` property to the path where the downloaded copy of `topology.py` is located. This property must be set to an absolute path.

[search_bind_authentication for Hue is not included in .ini file](#)

When `search_bind_authentication` is set to `false`, CM does not include it in `hue.ini`.

Workaround: Add the following to the Hue Service Advanced Configuration Snippet (Safety Valve) for `hue_safety_valve.ini`:

```
[desktop]
[[ldap]]
search_bind_authentication=false
```

[Erroneous warning displayed on the HBase configuration page on CDH 4.1 in Cloudera Manager 5.0.0](#)

An erroneous "Failed parameter validation" warning is displayed on the HBase configuration page on CDH 4.1 in Cloudera Manager 5.0.0

Severity: Low

Workaround: Use CDH4.2 or higher, or ignore the warning.

[Host recommissioning and decommissioning should occur independently](#)

In large clusters, when problems appear with a host or role, administrators may choose to decommission the host or role to fix it and then recommission the host or role to put it back in production. Decommissioning, especially host decommissioning, is slow, hence the importance of parallelization, so that host recommissioning can be initiated before decommissioning is done.

[Fixed Issues in Cloudera Manager 5.0.6](#)

[Slow staleness calculation can lead to ZooKeeper data loss when new servers are added](#)

In Cloudera Manager 5, starting new ZooKeeper Servers shortly after adding them can cause ZooKeeper data loss when the number of new servers exceeds the number of old servers.

Fixed Issues in Cloudera Manager 5.0.5

"POODLE" vulnerability on SSL/TLS enabled ports

The POODLE (Padding Oracle On Downgraded Legacy Encryption) attack takes advantage of a cryptographic flaw in the obsolete SSLv3 protocol, after first forcing the use of that protocol. The only solution is to disable SSLv3 entirely. This requires changes across a wide variety of components of CDH and Cloudera Manager. Cloudera Manager 5.0.5 provides these changes for Cloudera Manager 5.0.x deployments. All Cloudera Manager 5.0.x users should upgrade to 5.0.5 as soon as possible. For more information, see the [Cloudera Security Bulletin](#).

Issues Fixed in Cloudera Manager 5.0.2

Cloudera Manager Impala Query Monitoring does not work with Impala 1.3.1

Impala 1.3.1 contains changes to the runtime profile format that break the Cloudera Manager Query Monitoring feature. This leads to exceptions in the Cloudera Manager Service Monitor logs, and Impala queries no longer appear in the Cloudera Manager UI or API. The issue affects Cloudera Manager 5.0 and 4.6 - 4.8.2.

Workaround: None. The issue will be fixed in Cloudera Manager 4.8.3 and Cloudera Manager 5.0.1. To avoid the Service Monitor exceptions, turn off the Cloudera Manager Query Monitoring feature by going to **Impala Daemon > Monitoring** and setting the Query Monitoring Period to 0 seconds. Note that the Impala Daemons must be restarted when changing this setting, and the setting must be restored once the fix is deployed to turn the query monitoring feature back on. Impala queries will then appear again in Cloudera Manager's Impala query monitoring feature.

Issues Fixed in Cloudera Manager 5.0.1

Failure of HDFS Replication between clusters with YARN

HDFS replication between clusters in different Kerberos realms fails when using YARN if the target cluster is CDH 5.

Workaround: Use MapReduce (MRv1) instead of YARN.

If installing CDH 4 packages, the Impala 1.3.0 option does not work because Impala 1.3 is not yet released for CDH 4.

If installing CDH 4 packages, the Impala 1.3.0 option listed in the install wizard does not work because Impala 1.3.0 is not yet released for CDH 4.

Workaround: Install using parcels (where the unreleased version of Impala does not appear), or select a different version of Impala when installing with packages.

When updating dynamic resource pools, Cloudera Manager updates roles but may fail to update role information displayed in the UI

When updating dynamic resource pools, Cloudera Manager automatically refreshes the affected roles, but they sometimes get marked incorrectly as running with outdated configurations and requiring a refresh.

Workaround: Invoke the **Refresh Cluster** command from the cluster actions drop-down menu.

Upgrade of secure cluster requires installation of JCE policy files

When upgrading a secure cluster via Cloudera Manager, the upgrade initially fails due to the JDK not having Java Cryptography Extension (JCE) unlimited strength policy files. This is because Cloudera Manager installs a copy of the Java 7 JDK during the upgrade, which does not include the unlimited strength policy files. To ensure that unlimited strength functionality continues to work, install the unlimited strength JCE policy files immediately after completing the Cloudera Manager Upgrade Wizard and before taking any other actions in Cloudera Manager.

Workaround: Install the unlimited strength JCE policy files immediately after completing the Cloudera Manager Upgrade Wizard and before taking any other action in Cloudera Manager.

The Details page for MapReduce jobs displays the wrong id for YARN-based replications

The **Details** link for MapReduce jobs is wrong for YARN-based replications.

Workaround: Find the job id in the link and then go to the **YARN Applications** page and look for the job there.

Reset non-default HDFS File Block Storage Location Timeout value after upgrade from CDH 4 to CDH 5

During an upgrade from CDH 4 to CDH 5, if the HDFS File Block Storage Locations Timeout was previously set to a custom value, it will now be set to 10 seconds or the custom value, whichever is higher. This is required for Impala to start in CDH 5, and any value under 10 seconds is now a validation error. This configuration is only emitted for Impala and no services should be adversely impacted.

Workaround: None.

HDFS NFS gateway works only on RHEL and similar systems

Because of a bug in native versions of `portmap/rpcbind`, the HDFS NFS gateway does not work out of the box on SLES, Ubuntu, or Debian systems if you install CDH from the command-line, using packages. It does work on [supported versions](#) of RHEL-compatible systems on which `rpcbind-0.2.0-10.el6` or later is installed, and it does work if you use Cloudera Manager to install CDH, or if you start the gateway as root.

Bug: [731542](#) (Red Hat), [823364](#) (SLES), [594880](#) (Debian)

Severity: High

Workarounds and caveats:

- On Red Hat and similar systems, make sure `rpcbind-0.2.0-10.el6` or later is installed.
- On SLES, Debian, and Ubuntu systems, do one of the following:
 - Install CDH using Cloudera Manager; *or*
 - As of CDH 5.1, start the NFS gateway as root; *or*
 - [Start the NFS gateway without using packages](#); *or*
 - You can use the gateway by running `rpcbind` in insecure mode, using the `-i` option, but keep in mind that this allows anyone from a remote host to bind to the portmap.

Sensitive configuration values exposed in Cloudera Manager

Certain configuration values that are stored in Cloudera Manager are considered sensitive, such as database passwords. These configuration values should be inaccessible to non-administrator users, and this is enforced in the Cloudera Manager Administration Console. However, these configuration values are not redacted when they are read through the API, possibly making them accessible to users who should not have such access.

Gateway role configurations not respected when deploying client configurations

Gateway configurations set for gateway role groups other than the default one or at the role level were not being respected.

Documentation reflects requirement to enable at least Level 1 encryption before enabling Kerberos authentication

[Cloudera Security](#) now indicates that before enabling Kerberos authentication you should first enable at least Level 1 encryption.

HDFS NFS gateway does not work on all Cloudera-supported platforms

The NFS gateway cannot be started on some Cloudera-supported platforms.

Workaround: None. Fixed in Cloudera Manager 5.0.1.

Replace YARN_HOME with HADOOP_YARN_HOME during upgrade

If `yarn.application.classpath` was set to a non-default value on a CDH 4 cluster, and that cluster is upgraded to CDH 5, the classpath is not updated to reflect that `$YARN_HOME` was replaced with `$HADOOP_YARN_HOME`. This will cause YARN jobs to fail.

Workaround: Reset `yarn.application.classpath` to the default, then re-apply your classpath customizations if needed.

Insufficient password hashing in Cloudera Manager

In versions of Cloudera Manager earlier than 4.8.3 and earlier than 5.0.1, user passwords are only hashed once. Passwords should be hashed multiple times to increase the cost of dictionary based attacks, where an attacker

tries many candidate passwords to find a match. The issue only affects user accounts that are stored in the Cloudera Manager database. User accounts that are managed externally (for example, with LDAP or Active Directory) are not affected.

In addition, because of this issue, Cloudera Manager 4.8.3 cannot be upgraded to Cloudera Manager 5.0.0. Cloudera Manager 4.8.3 must be upgraded to 5.0.1 or later.

Workaround: Upgrade to Cloudera Manager 5.0.1.

Upgrade to Cloudera Manager 5.0.0 from SLES older than Service Pack 3 with PostgreSQL older than 8.4 fails

Upgrading to Cloudera Manager 5.0.0 from SUSE Linux Enterprise Server (SLES) older than Service Pack 3 will fail if the embedded PostgreSQL database is in use and the installed version of PostgreSQL is less than 8.4.

Workaround: Either migrate away from the embedded PostgreSQL database (use MySQL or Oracle) or upgrade PostgreSQL to 8.4 or greater.

MR1 to MR2 import fails on a secure cluster

When running the MR1 to MR2 import on a secure cluster, YARN jobs will fail to find `container-executor.cfg`.

Workaround: Restart YARN after the import.

After upgrade from CDH 4 to CDH 5, Oozie is missing workflow extension schemas

After an upgrade from CDH 4 to CDH 5, Oozie does not pick up the new workflow extension schemas automatically. User will need to update `oozie.service.SchemaService.wf.ext.schemas` manually and add the schemas added in CDH 5: `shell-action-0.3.xsd`, `sqoop-action-0.4.xsd`, `distcp-action-0.2.xsd`, `oozie-sla-0.1.xsd`, `oozie-sla-0.2.xsd`. Note: None of the existing jobs will be affected by this bug, only new workflows that require new schemas.

Workaround: Add the new workflow extension schemas to Oozie manually by editing `oozie.service.SchemaService.wf.ext.schemas`.

Issues Fixed in Cloudera Manager 5.0.0

The Sqoop Upgrade command in Cloudera Manager may report success even when the upgrade fails

Workaround: Do one of the following:

- 1. Click the Sqoop service and then the **Instances** tab.
- 2. Click the Sqoop server role then the **Commands** tab.
- 3. Click the **stdout** link and scan for the Sqoop Upgrade command.
- In the All Recent Commands page, select the **stdout** link for latest Sqoop Upgrade command.

Verify that the upgrade did not fail.

Cannot restore a snapshot of a deleted HBase table

If you take a snapshot of an HBase table, and then delete that table in HBase, you will not be able to restore the snapshot.

Severity: Med

Workaround: Use the "Restore As" command to recreate the table in HBase.

Stop dependent HBase services before enabling HDFS Automatic Failover.

When enabling HDFS Automatic Failover, you need to first stop any dependent HBase services. The Automatic Failover configuration workflow restarts both NameNodes, which could cause HBase to become unavailable.

Severity: Medium

New schema extensions have been introduced for Oozie in CDH 4.1

In CDH 4.1, Oozie introduced new versions for Hive, Sqoop and workflow schema. To use them, you must add the new schema extensions to the Oozie SchemaService Workflow Extension Schemas configuration property in Cloudera Manager.

Severity: Low

Workaround: In Cloudera Manager, do the following:

1. Go to the CDH 4 **Oozie** service page.
2. Go to the **Configuration** tab, **View and Edit**.
3. Search for "Oozie Schema". This should show the **Oozie SchemaService Workflow Extension Schemas** property.
4. Add the following to the **Oozie SchemaService Workflow Extension Schemas** property:

```
shell-action-0.2.xsd  
hive-action-0.3.xsd  
sqoop-action-0.3.xsd
```

5. Save these changes.

[YARN Resource Scheduler user FairScheduler rather than FIFO.](#)

Cloudera Manager 5.0.0 sets the default YARN Resource Scheduler to FairScheduler. If a cluster was previously running YARN with the FIFO scheduler, it will be changed to FairScheduler next time YARN restarts. The FairScheduler is only supported with CDH4.2.1 and later, and older clusters may hit failures and need to manually change the scheduler to FIFO or CapacityScheduler.

Severity: Medium

Workaround: For clusters running CDH 4 prior to CDH 4.2.1:

1. Go the YARN service Configuration page
2. Search for "scheduler.class"
3. Click in the Value field and select the scheduler you want to use.
4. Save your changes and restart YARN to update your configurations.

[Resource Pools Summary is incorrect if time range is too large.](#)

The Resource Pools Summary does not show correct information if the Time Range selector is set to show 6 hours or more.

Severity: Medium

Workaround: None.

[When running the MR1 to MR2 import on a secure cluster, YARN jobs will fail to find container-executor.cfg](#)

Workaround: Restart YARN after the import steps finish. This causes the file to be created under the YARN configuration path, and the jobs now work.

[When upgrading to Cloudera Manager 5.0.0, the "Dynamic Resource Pools" page is not accessible](#)

When upgrading to Cloudera Manager 5.0.0, users will not be able to directly access the "Dynamic Resource Pools" page. Instead, they will be presented with a dialog saying that the Fair Scheduler XML Advanced Configuration Snippet is set.

Workaround:

1. Go to the YARN service.
2. Click the **Configuration** tab.
3. Copy the value of the **Fair Scheduler XML Advanced Configuration Snippet** into a file.
4. Clear the value of **Fair Scheduler XML Advanced Configuration Snippet**.
5. Recreate the desired Fair Scheduler allocations in the **Dynamic Resource Pools** page, using the saved file for reference.

[New Cloudera Enterprise licensing is not reflected in the wizard and license page](#)

Workaround: None.

The AWS Cloud wizard fails to install Spark due to missing roles

Workaround: Do one of the following:

- Use the Installation wizard.
- Open a new window, click the Spark service, click on the **Instances** tab, click **Add**, add all required roles to Spark. Once the roles are successfully added, click the **Retry** button in the Installation wizard.

Spark on YARN requires manual configuration

Spark on YARN requires the following manual configuration to work correctly: modify the YARN Application Classpath by adding `/etc/hadoop/conf`, making it the very first entry.

Workaround: Add `/etc/hadoop/conf` as the first entry in the YARN Application classpath.

Monitoring works with Solr and Sentry only after configuration updates

Cloudera Manager monitoring does not work out of the box with Solr and Sentry on Cloudera Manager 5. The Solr service is in Bad health, and all Solr Servers have a failing "Solr Server API Liveness" health check.

Severity: Medium

Workaround: Complete the configuration steps below:

1. Create "HTTP" user and group on all machines in the cluster (with `useradd 'HTTP'` on RHEL-type systems).
2. The instructions that follow this step assume there is no existing Solr Sentry policy file in use. In that case, first create the policy file on `/tmp` and then copy it over to the appropriate location in HDFS that Solr Servers check. If there is already a Solr Sentry policy in use, it must be modified to add the following `[group]` / `[role]` entries for 'HTTP'. Create a file (for example, `/tmp/cm-authz-solr-sentry-policy.ini`) with the following contents:

```
[groups]
HTTP = HTTP
[roles]
HTTP = collection = admin->action=query
```

3. Copy this file to the location for the "Sentry Global Policy File" for Solr. The associated config name for this location is `sentry.solr.provider.resource`, and you can see the current value by navigating to the **Sentry** sub-category in the **Service Wide** configuration editing workflow in the Cloudera Manager UI. The default value for this entry is `/user/solr/sentry/sentry-provider.ini`. This refers to a path in HDFS.
4. Check if you have entries in HDFS for the parent(s) directory:

```
sudo -u hdfs hadoop fs -ls /user
```

5. You may need to create the appropriate parent directories if they are not present. For example:

```
sudo -u hdfs hadoop fs -mkdir /user/solr/sentry
```

6. After ensuring the parent directory is present, copy the file created in step 2 to this location, as follows:

```
sudo -u hdfs hadoop fs -put /tmp/cm-authz-solr-sentry-policy.ini
/user/solr/sentry/sentry-provider.ini
```

7. Ensure that this file is owned/readable by the solr user (this is what the Solr Server runs as):

```
sudo -u hdfs hadoop fs -chown solr /user/solr/sentry/sentry-provider.ini
```

8. Restart the Solr service. If both Kerberos and Sentry are being enabled for Solr, the MGMT services also need to be restarted. The Solr Server liveness health checks should clear up once SMON has had a chance to contact the servers and retrieve metrics.

Out-of-memory errors may occur when using the Reports Manager

Out-of-memory errors may occur when using the Cloudera Manager Reports Manager.

Workaround: Set the value of the "Java Heap Size of Reports Manager" property to at least the size of the HDFS filesystem image (`fsimage`) and restart the Reports Manager.

Applying license key using Internet Explorer 9 and Safari fails

Cloudera Manager is designed to work with IE 9 and above and Safari. However the file upload widget used to upload a license currently doesn't work with IE 9 or Safari. Therefore, installing an enterprise license doesn't work.

Workaround: Use another supported browser.

Issues Fixed in Cloudera Manager 5.0.0 Beta 2

The Sqoop Upgrade command in Cloudera Manager may report success even when the upgrade fails

Workaround: Do one of the following:

- 1. Click the Sqoop service and then the **Instances** tab.
- 2. Click the Sqoop server role then the **Commands** tab.
- 3. Click the **stdout** link and scan for the Sqoop Upgrade command.
- In the All Recent Commands page, select the **stdout** link for latest Sqoop Upgrade command.

Verify that the upgrade did not fail.

The HDFS Canary Test is disabled for secured CDH 5 services.

Due to a bug in Hadoop's handling of multiple RPC clients with distinct configurations within a single process with Kerberos security enabled, Cloudera Manager will disable the HDFS canary test when security is enabled so as to prevent interference with Cloudera Manager's MapReduce monitoring functionality.

Severity: Medium

Workaround: None

Not all monitoring configurations are migrated from MR1 to MR2.

When MapReduce v1 configurations are imported for use by YARN (MR2), not all of the monitoring configuration values are currently migrated. Users may need to reconfigure custom values for properties such as thresholds.

Severity: Medium

Workaround: Manually reconfigure any missing property values.

"Access Denied" may appear for some features after adding a license or starting a trial.

After starting a 60-day trial or installing a license for Enterprise Edition, you may see an "access denied" message when attempting to access certain Enterprise Edition-only features such as the Reports Manager. You need to log out of the Admin Console and log back in to access these features.

Severity: Low

Workaround: Log out of the Admin Console and log in again.

Hue must set impersonation on when using Impala with impersonation.

When using Impala with impersonation, the `impersonation_enabled` flag must be present and configured in the `hue.ini` file. If impersonation is enabled in Impala (i.e. Impala is using Sentry) then this flag must be set **true**. If Impala is not using impersonation, it should be set **false** (the default).

Workaround: Set advanced configuration snippet value for `hue.ini` as follows:

1. Go to the **Hue Service Configuration Advanced Configuration Snippet for hue_safety_valve.ini** under the Hue service Configuration settings, **Service-Wide > Advanced** category.

2. Add the following, then uncomment the setting and set the value True or False as appropriate:

```
#####
# Settings to configure Impala
#####

[impala]
....
# Turn on/off impersonation mechanism when talking to Impala
## impersonation_enabled=False
```

Cloudera Manager Server may fail to start when upgrading using a PostgreSQL database.

If you're upgrading to Cloudera Manager 5.0.0 beta 1 and you're using a PostgreSQL database, the Cloudera Manager Server may fail to start with a message similar to the following:

```
ERROR [main:dbutil.JavaRunner@57] Exception while executing
com.cloudera.cmf.model.migration.MigrateConfigRevisions
java.lang.RuntimeException: java.sql.SQLException: Batch entry <xxx> insert into
REVISIONS
(REVISION_ID, OPTIMISTIC_LOCK_VERSION, USER_ID, TIMESTAMP, MESSAGE) values (...)
was aborted. Call getNextException to see the cause.
```

Workaround: Use `psql` to connect directly to the server's database and issue the following SQL command:

```
alter table REVISIONS alter column MESSAGE type varchar(1048576);
```

After that, your Cloudera Manager server should start up normally.

Issues Fixed in Cloudera Manager 5.0.0 Beta 1

After an upgrade from Cloudera Manager 4.6.3 to 4.7, Impala does not start.

After an upgrade from Cloudera Manager 4.6.3 to 4.7 when Navigator is used, Impala will fail to start because the Audit Log Directory property has not been set by the upgrade procedure.

Severity: Low.

Workaround: Manually set the property to `/var/log/impalad/audit`. See [Service Auditing Properties](#) for more information.

Cloudera Navigator 2 Release Notes

These release notes provide information on the new and changed features, known issues, and fixed issues for Cloudera Navigator.

New Features and Changes in Cloudera Navigator 2

The following sections describe what's new and changed in each Cloudera Navigator 2 release.

New Features in Cloudera Navigator 2

The following sections describe what's new in each Cloudera Navigator 2 release.

- **Note:** Although there is a CDH 5.3.5 release, there is no synchronous Cloudera Navigator 2.2.5 release.

What's New in Cloudera Navigator 2.2.4

An issue has been fixed. See [Issues Fixed in Cloudera Navigator 2.2.4](#) on page 195.

What's New in Cloudera Navigator 2.2.3

A number of issues have been fixed. See [Issues Fixed in Cloudera Navigator 2.2.3](#) on page 195.

What's New in Cloudera Navigator 2.2.2

An issue has been fixed. See [Issues Fixed in Cloudera Navigator 2.2.2](#) on page 196.

What's New in Cloudera Navigator 2.2.1

A number of issues have been fixed. See [Issues Fixed in Cloudera Navigator 2.2.1](#) on page 196.

What's New in Cloudera Navigator 2.2.0

- **Policies** are generally available and are always enabled. Policy properties now support Java expressions.
- **Search** - Search functionality now includes autocomplete.

What's New in Cloudera Navigator 2.1.5

A number of issues have been fixed. See [Issues Fixed in Cloudera Navigator 2.1.5](#) on page 196.

What's New in Cloudera Navigator 2.1.4

An issue has been fixed. See [Issues Fixed in Cloudera Navigator 2.1.4](#) on page 196.

- **Note:** There was no Cloudera Navigator 2.1.3 release.

What's New in Cloudera Navigator 2.1.2

A number of issues have been fixed. See [Issues Fixed in Cloudera Navigator 2.1.2](#) on page 197.

What's New in Cloudera Navigator 2.1.1

A number of issues have been fixed. See [Issues Fixed in Cloudera Navigator 2.1.1](#) on page 198.

What's New in Cloudera Navigator 2.1.0

- **Auditing Component**
 - New auditing UI featuring saved audit reports. The Navigator auditing UI is no longer available from Cloudera Manager. Instead, auditing is integrated with lineage, discovery and the policy engine. The UI is provided by the Metadata Server, which is now required for the auditing component.
 - Sentry auditing now includes Sentry commands issued from Impala.
- **Metadata Component**
 - Search results contain a type appropriate link to a Hue browser.
 - HDFS directories and files - File Browser
 - Hive database and tables - Metastore Manager
 - MapReduce, YARN, Pig - Job Browser
 - **Policies** - support rules for modifying metadata and sending notifications when entities are extracted.

- **Note:** Policies is a beta feature that is disabled by default.

- **Security**
 - **Role-Based Access Control** - support assigning groups to roles that constrain access to Navigator features
 - **Authentication** - LDAP and Active Directory authentication of Navigator users
 - **SSL** - enable SSL for encrypted communication
- **API**
 - Version changed to v3
 - Supports auditing and policies

What's New in Cloudera Navigator 2.0.5

An issue was fixed. See [Issues Fixed in Cloudera Navigator 2.0.5](#) on page 198.

- **Note:** There is no Cloudera Navigator 2.0.4 release.

What's New in Cloudera Navigator 2.0.3

A number of issues have been fixed. See [Issues Fixed in Cloudera Navigator 2.0.3](#) on page 198.

What's New in Cloudera Navigator 2.0.2

An issue was fixed. See [Issues Fixed in Cloudera Navigator 2.0.2](#) on page 198.

What's New in Cloudera Navigator 2.0.1

- Masking of personally identifiable information (PII) in query strings that appear in audit events and lineage. Enabled by default.
- REST API support for registering business metadata for entities before they appear in Navigator.

A number of issues have been fixed. See [Issues Fixed in Cloudera Navigator 2.0.1](#) on page 198.

What's New in Cloudera Navigator 2.0.0

- **Auditing Component**
 - Added support for auditing the Sentry service
 - Added support for publishing audit logs to syslog
- **Metadata Component**
 - Newly designed Query Builder with faceted filtering
 - Simplified Pig lineage
 - Added support for Sqoop and Oozie lineage
 - Many performance and stability improvements
- **Security** - includes Navigator Encrypt and Navigator Key Trustee, formerly known as Gazzang zNcrypt and Gazzang zTrustee. These features provide enterprise-grade encryption and key management. For information on these features, see the [Cloudera Security Datasheet](#) and contact your account team.

Changed Features in Cloudera Navigator 2

The following sections describe what's changed in each Cloudera Navigator 2 release.

What's Changed in Cloudera Navigator 2.2.0

- **Metadata Component** - Policies created with Cloudera Navigator 2.1 (containing the Beta version policy engine) are not retained when upgrading to Cloudera Navigator 2.2.

What's Changed in Cloudera Navigator 2.1.0

- **Auditing Component**
 - Events generated by the `impala` user are discarded by default.

What's Changed in Cloudera Navigator 2.0.1

- **Metadata Component**
 - The REST API version changed to v2.

What's Changed in Cloudera Navigator 2.0.0

- **Auditing Component**
 - Events generated by the `solr` user are discarded by default.

Known Issues and Workarounds in Cloudera Navigator 2

The following sections describe the current known issues in Cloudera Navigator 2.

Spurious errors about missing database connectors are reported in the Metadata Server log file

Workaround: Ignore the errors.

Audit CSV has extra columns and is missing some data

When you export audits to CSV, Sentry data is not visible in the generated CSV file. Also, some of the column names show up twice (Operation Text, Database Name, Object Type, etc.), but the data only shows up in one of the columns.

Workaround: Export audits to JSON format to see Sentry data.

Metadata component in Cloudera Navigator 1.2 (included with Cloudera Manager 5.0) cannot be upgraded to 2.0

Workaround:

Cloudera does not provide an upgrade path from the Navigator Metadata component which was a beta release in Cloudera Navigator 1.2 to the Cloudera Navigator 2 release. If you are upgrading from Cloudera Navigator 1.2 (included with Cloudera Manager 5.0), you must perform a clean install of Cloudera Navigator 2. Therefore, if you have Cloudera Navigator roles from a 1.2 release:

1. Delete the Navigator Metadata Server role.
2. Remove the contents of the [Navigator Metadata Server storage directory](#).
3. Add the Navigator Metadata Server role according to the process described in [Adding the Navigator Metadata Server Role](#).
4. Clear the cache of any browser that had used the 1.2 release of the Navigator Metadata component. Otherwise, you may observe errors in the Navigator Metadata UI.

See [Upgrading Cloudera Navigator](#).

Variables in Sqoop queries are not instantiated

If you have a query that includes a variable, such as A (suppose `$A = "EMPID = 123456"`), the Sqoop command line will list `$A` but it will not instantiate the variable.

Workaround: None.

The Hive extractor fails if entities don't exist.

If a Hive query is executed and the tables/views/columns etc. used in the query are deleted before the query is captured by Navigator; then Navigator will not be able to parse the query and will not capture any information for that query.

Workaround: None.

The Hive extractor does not handle all Hive statements

The Hive extractor does not handle the following cases:

- Table generating functions
- Lateral views

- Transform clauses
- Regular expression in select clause

If a query involves any of the above, lineage will not be complete for that Hive query.

Workaround: None.

[Impala auditing does not support filtering during event capture.](#)

Impala auditing does not support filtering during event capture.

Severity: Low

Workaround: None.

[The IP address in a Hue service audit log shows as "unknown"](#)

The IP address in a Hue service audit log shows as "unknown".

Severity: Low

Workaround: None.

[Hive service configuration impact on Hue service auditing](#)

If the audit configuration for a Hive service is changed, Beeswax must be restarted to pick up the change in the Hue service audit log.

Severity: Low

Workaround: None.

[Hive service configuration in auditing component](#)

For Hive services, the auditing component doesn't support the "Shutdown" option for the "Queue Policy" property.

Severity: Low

Workaround: None.

Issues Fixed in Cloudera Navigator 2

The following sections describe the issues fixed in each Cloudera Navigator 2 release.

Issues Fixed in Cloudera Navigator 2.2.4

["POODLE" vulnerability on SSL/TLS enabled ports](#)

The POODLE (Padding Oracle On Downgraded Legacy Encryption) attack takes advantage of a cryptographic flaw in the obsolete SSLv3 protocol, after first forcing the use of that protocol. The only solution is to disable SSLv3 entirely. This requires changes across a wide variety of components of Cloudera Navigator in 2.2.x and all earlier versions. Cloudera Navigator 2.2.4 provides these changes for Cloudera Navigator 2.2.x deployments. All Cloudera Navigator users should upgrade to 2.2.4 as soon as possible. For more information, see the [Cloudera Security Bulletin](#).

Issues Fixed in Cloudera Navigator 2.2.3

[Navigator Audit Server reports invalid null characters in HBase audit events when using the PostgreSQL database](#)

Navigator Audit Server reports invalid null characters in HBase audit events when using the PostgreSQL database. HBase allows null characters in qualifiers, so now Navigator escapes them.

Oozie extractor throws too many Boolean clauses exception

Issues Fixed in Cloudera Navigator 2.2.2

The audit reports UI now returns results when there are a large number of audit records

The audit reports UI was not returning results when there were a large number of audit records matching a particular time period, especially when the period included multiple days. The UI is now also much more responsive.

Issues Fixed in Cloudera Navigator 2.2.1

Browser autocomplete no longer enabled before authentication

Form fields before authentication in the application have auto-complete enabled. Any user using the same computer would be able to see information entered by a previous user.

Navigator Web UI no longer exposes paths to directory listing/forceful browsing

The web server is configured to display the list of files contained in this directory. This is not recommended because the directory may contain files that are not normally exposed through links on the web site.

Navigator Audit Server no longer throws OOM for very long Impala queries

Issues Fixed in Cloudera Navigator 2.2.0

If Hue is added after Navigator, search results do not have links to Hue

In the Metadata UI, search results contain links to an appropriate application in Hue. However, if you add a Hue service after Navigator roles, there will be no links to Hue.

Workaround:

1. Set the cluster's display name and name properties to be the same:
 - a. Get cluster's name and display name using following API: `http://hostname:7180/api/v6/clusters`.
 - b. In the Cloudera Manager Admin Console, at the right of the cluster name, click the down arrow and select **Rename Cluster**. Set the cluster display name to match its name.
2. Restart the Navigator Metadata server.

Issues Fixed in Cloudera Navigator 2.1.5

Navigator Audit Server reports invalid null characters in HBase audit events when using the PostgreSQL database

Navigator Audit Server reports invalid null characters in HBase audit events when using the PostgreSQL database. HBase allows null characters in qualifiers, so now Navigator escapes them.

Oozie extractor throws too many Boolean clauses exception

Issues Fixed in Cloudera Navigator 2.1.4

The audit reports UI now returns results when there are a large number of audit records

The audit reports UI was not returning results when there were a large number of audit records matching a particular time period, especially when the period included multiple days. The UI is now also much more responsive.

Issues Fixed in Cloudera Navigator 2.1.2

Search results in Navigator now have links to Hue

In the Metadata UI, search results contain links to an appropriate application in Hue. The links may be missing for either of the following reasons:

- Hue was added after Navigator Metadata server was started.
- The cluster name and display name are different.

Workaround:

1. Set the cluster's display name and name properties to be the same:
 - a. Get cluster's name and display name using following API: `http://hostname:7180/api/v6/clusters`.
 - b. In the Cloudera Manager Admin Console, at the right of the cluster name, click the down arrow and select **Rename Cluster**. Set the cluster display name to match its name.
2. Restart the Navigator Metadata server.

Browser autocomplete no longer enabled before authentication

Form fields before authentication in the application have auto-complete enabled. Any user using the same computer would be able to see information entered by a previous user.

Navigator Web UI no longer exposes paths to directory listing/forceful browsing

The web server is configured to display the list of files contained in this directory. This is not recommended because the directory may contain files that are not normally exposed through links on the web site.

Navigator Audit Server no longer throws OOM for very long Impala queries

Issues Fixed in Cloudera Navigator 2.1.1

LDAP lookups in Active Directory to resolve group membership are now working

Dropping a Hive table and creating a view with same name or vice versa no longer raises an error

HDFS extraction now works after upgrading CDH from 5.1 to 5.2

Setting a property in the Hue advanced configuration snippet no longer throws a "too many Boolean clauses" error in Navigator Metadata

Issues Fixed in Cloudera Navigator 2.0.5

Memory leak in Navigator Audit Server due to error during batch operations

Issues Fixed in Cloudera Navigator 2.0.3

Dropping a Hive table and creating a view with same name or vice versa no longer raises an error

Setting a property in the Hue advanced configuration snippet no longer throws a "too many Boolean clauses" error in Navigator Metadata

Issues Fixed in Cloudera Navigator 2.0.2

HBase auditing initialization failure can prevent region opening indefinitely

Issues Fixed in Cloudera Navigator 2.0.1

Hive View to Table lineage is missing

The lineage of the underlying tables does not appear in the lineage view.

Workaround: Launch lineage on the underlying tables directly.

The "allowed" query selector is missing from the audit REST API

Queries such as `http://hostname:7180/api/v7/audits?maxResults=10&query=allowed==false` are now supported.

Workaround: None.

Metadata in metadata files is not processed

Workaround: None.

Lineage does not work when launched on a field

When you launch lineage on a field (Hive column or Pig field, or a Sqoop sub-operation), the UI displays the initial graph properly. However if you expand the parent item (Hive table in case of a Hive column), then things start disappearing from the lineage diagram.

Workaround: None.

When you specify an end date for the `created` property, no results are returned.

Workaround: Clear the end date control or specify an end date of `TO+*%5D%22%7D`.

Navigating back to the parent entity in a Pig lineage diagram sometimes displays the error: Cannot read property 'x' of undefined.

Workaround: None.

Issues Fixed in Cloudera Navigator 2.0.0

The last accessed time for Hive table is incorrect.

Workaround: None.

Pig job that has relations with self is unreadable in lineage view.

The Metadata UI currently does not handle situation where there is a data-flow relation between elements that are also related via parent-child relation.

Workaround: None.

If auditing is enabled, during an upgrade from Cloudera Manager 4.6.3 to 4.7, the Impala service won't start.

Impala auditing requires the Audit Log Directory property to be set, but the upgrade process fails to set the property.

Workaround: Do one of the following:

- Stop the Cloudera Navigator Audit server.
- Ensure that you have a Cloudera Navigator license and manually set the property to `/var/log/impalad/audit`.

Empty box appears over the list of results after adding a tag to a file

When tags are added to an entity, in some cases a white box remains after pressing **Enter**.

Workaround: Refresh the page to remove the artifact.

Issues Fixed in Cloudera Navigator 1.2.0

Certain complex multi-level lineages, such as directory/file and database/table, may not be fully represented visually.

Workaround: None.

Version and Download Information

Version and download information for Cloudera Manager, CDH, Impala, and Search can be found in the HTML documentation on the website at [Cloudera Documentation](#). Select the release version number and go to the HTML version of the Release Guide.

Product Compatibility Matrix

In an enterprise data hub, Cloudera Manager and CDH will interact with several projects such as Apache Accumulo, Cloudera Impala, Hue, Cloudera Search, Cloudera Navigator and so on. This guide provides information about which major and minor release version of a product is supported with which release version of Cloudera Manager, CDH and if applicable, Cloudera Search and Cloudera Impala.

Compatibility across different release versions of Cloudera Manager and CDH must be taken into account especially before carrying out install/upgrade procedures.

JDK compatibility also varies across different Cloudera Manager and CDH versions. Certain versions of CDH 4 are compatible with both JDK 6 and JDK 7. In such cases, ensure all your services are deployed on the same major version. For example, you should not run Hadoop on JDK 6 while running Sqoop on JDK 7. Additionally, since Cloudera does not support mixed environments, all nodes in your cluster must be running the same major JDK version.

Each product matrix contains at least a subset of the following fields:

- **Feature:** This column lists notable new features that have been included in a particular release. For products/releases that do not have this column, refer the respective Release Notes for detailed information.
- **Lowest supported Cloudera Manager version:** Specifies the earliest version of Cloudera Manager that supports a product release version.
- **Lowest supported CDH version:** Specifies the earliest version of CDH that supports a product release version. The Cloudera Search and Impala matrices are an exception to this since each release is only compatible with a specific CDH release.
- **Lowest supported Impala version:** This column may not apply to all products, for example, Cloudera Search.
- **Lowest supported Search version:** This column may not apply to all products, for example, Cloudera Impala.
- **Integrated into CDH:** This column specifies whether a particular release is shipped with CDH or available independently. This field may not apply to all products, for example, Cloudera Navigator.

JDK Compatibility

This topic contains compatibility information across versions of JDK and CDH/Cloudera Manager.

Cloudera Manager - JDK Compatibility

Cloudera Manager must be running on a JDK version that is supported across all Hadoop clusters that it manages. This means Cloudera Manager 5 on JDK 6 can only manage CDH 4 clusters, whereas you must use Cloudera Manager 5 on JDK 7 to be able to manage CDH 5 clusters.

Cloudera Manager Version	Oracle JDK 1.6	Oracle JDK 1.7	Oracle JDK 1.8
Cloudera Manager 5.3.x	1.6.0_31	1.7.0_67	1.8.0_11
Cloudera Manager 5.2.x	1.6.0_31	1.7.0_67	Not Supported
Cloudera Manager 5.1.x	1.6.0_31	1.7.0_55	Not Supported
Cloudera Manager 5.0.x	1.6.0_31	1.7.0_45	Not Supported
Cloudera Manager 4.8.x	1.6.0_31	1.7.0_45 1.7.0_55 has been certified against Cloudera Manager 4.8.3	Not Supported
Cloudera Manager 4.7.x	1.6.0_31	1.7.0_45	Not Supported

Product Compatibility Matrix

Cloudera Manager Version	Oracle JDK 1.6	Oracle JDK 1.7	Oracle JDK 1.8
Starting CDH 4.4, all cluster nodes and services must be running the same JDK version (that is, <i>all</i> deployed on JDK 6 or <i>all</i> deployed on a supported JDK 7 version).			
Cloudera Manager 4.6.x	1.6.0_31	1.7.0_45	Not Supported
Cloudera Manager 4.5.x Cloudera Manager 4.5.1 supports CDH 4.2 running with JDK 7, with restrictions .	1.6.0_31	1.7.0_15	Not Supported
Cloudera Manager 4.1.x	1.6.0_31	Not Supported	Not Supported
Cloudera Manager 4.0.x	1.6.0_31	Not Supported	Not Supported

CDH - JDK Compatibility

- For JDK 1.6, CDH 4 is certified with 1.6.0_31, but any later maintenance (_xx) release should be acceptable for production, following Oracle's release notes and restrictions. The minimum supported version is 1.6.0_8.
- For JDK 1.7, the table below lists the JDK version that is certified with each CDH release, but any later maintenance (_xx) release should be acceptable for production, following Oracle's release notes and restrictions.

CDH Version	Oracle JDK 1.6	Oracle JDK 1.7	Oracle JDK 1.8
CDH 5.3.x	Not Supported	1.7.0_67	1.8.0_11
CDH 5.2.x	Not Supported	1.7.0_67	Not Supported
CDH 5.1.x	Not Supported	1.7.0_55	Not Supported
CDH 5.0.x	Not Supported	1.7.0_55	Not Supported
CDH 4.7.x	1.6.0_31	1.7.0_55	Not Supported
CDH 4.6.x	1.6.0_31	1.7.0_55	Not Supported
CDH 4.5.x	1.6.0_31	1.7.0_55	Not Supported
CDH 4.4.x	1.6.0_31	1.7.0_15	Not Supported
CDH 4.3.x	1.6.0_31	1.7.0_15	Not Supported
CDH 4.2.x Cloudera Manager 4.5.1 supports CDH 4.2 running with JDK 7, with restrictions .	1.6.0_31	1.7.0_15	Not Supported
CDH 4.1.x	1.6.0_31	Not Supported	Not Supported
CDH 4.0.x	1.6.0_31	Not Supported	Not Supported

CDH and Cloudera Manager

This matrix contains compatibility information across release versions of CDH and Cloudera Manager. For detailed documentation, see [Cloudera Documentation](#).

■ **Note:**

The Cloudera Manager minor version must always be *equal to or greater than* the CDH minor version because older versions of Cloudera Manager may not support features in newer versions of CDH. For example, if you want to upgrade to CDH 5.1.2 you must first upgrade to Cloudera Manager 5.1 or higher.

Cloudera Manager Version	Supported CDH Versions
Cloudera Manager 5.3.x	CDH 4.0.0 - 4.x.x, CDH 5.0.0 - 5.3.x
Cloudera Manager 5.2.x	CDH 4.0.0 - 4.x.x, CDH 5.0.0 - 5.2.x
Cloudera Manager 5.1.x	CDH 4.0.0 - 4.x.x, CDH 5.0.0 - 5.1.x
Cloudera Manager 5.0.x	CDH 4.0.0 - 4.x.x, CDH 5.0.0 - 5.0.x
Cloudera Manager 5.0.0 Beta 2	CDH 4.0.0 - 4.x.x, CDH 5.0.0 Beta 2
Cloudera Manager 5.0.0 Beta 1	CDH 4.0.0 - 4.x.x, CDH 5.0.0 Beta 1
Cloudera Manager 4.8.x	CDH 3 Update 1-6, CDH 4.0.0 - 4.8.x
Cloudera Manager 4.7.x	CDH 3 Update 1-6, CDH 4.0.0 - 4.7.x
Cloudera Manager 4.6.x	CDH 3 Update 1-6, CDH 4.0.0 - 4.6.x
Cloudera Manager 4.5.x	CDH 3 Update 1-6, CDH 4.0.0 - 4.5.x
Cloudera Manager 4.1.x	CDH 3 Update 1-6, CDH 4.0.0 - 4.1.x
Cloudera Manager 4.0.x	CDH 3 Update 1-6, CDH 4.0.0 (not including beta)

Apache Accumulo

This matrix contains compatibility information across versions of Apache Accumulo and CDH/Cloudera Manager. For detailed information on each release, see [Apache Accumulo documentation](#).

Product	Lowest supported Cloudera Manager version	Lowest supported CDH version	Lowest supported Impala version	Lowest supported Search version	Integrated into CDH
Accumulo 1.6.0	Cloudera Manager 5.0.0	CDH 4.6.0 - 4.x.x, CDH 5.1.0	Not Supported	Not Supported	No
Accumulo 1.4.4	Cloudera Manager 5.0.0	CDH 4.5.0 (Not for use with CDH 5)	Not Supported	Not Supported	No
Accumulo 1.4.3	Cloudera Manager 5.0.0	CDH 4.3.0 (Not for use with CDH 5)	Not Supported	Not Supported	No

Backup and Disaster Recovery

This matrix contains compatibility information across features of Cloudera Manager Backup and Disaster Recovery and CDH/Cloudera Manager. Refer the [Cloudera](#) documentation for more details.

Product	Feature	Lowest supported Cloudera Manager version	Lowest supported CDH version	Lowest supported Impala version	Lowest supported Search version	Integrated into CDH
Backup & Disaster Recovery	Replication	Cloudera Manager 4.5.0	CDH 4.0.1	Impala 1.0.x	Not Supported	No
	Snapshots	Cloudera Manager 5.0.0	CDH 5.0.0	Not Supported	Not Supported	No

Cloudera Impala

This matrix contains compatibility information across versions of Cloudera Impala and CDH/Cloudera Manager. For detailed information on each release, see [Cloudera Impala documentation](#).

Product	Supported Cloudera Manager versions	Supported CDH versions	Integrated into CDH
Cloudera Impala 2.1.4	Cloudera Manager 5.0.0 - 5.x.x	CDH 5.3.4 and CDH 5.3.5	CDH 5.3.4 and CDH 5.3.5
Cloudera Impala 2.1.3	Cloudera Manager 5.0.0 - 5.x.x	CDH 5.3.3	CDH 5.3.3
Cloudera Impala 2.1.2	Cloudera Manager 5.0.0 - 5.x.x	CDH 5.3.2	CDH 5.3.2
Cloudera Impala 2.1.1	Cloudera Manager 5.0.0 - 5.x.x	CDH 5.3.1	CDH 5.3.1
Cloudera Impala 2.1.0	Cloudera Manager 5.0.0 - 5.x.x	CDH 5.3.0	CDH 5.3.0
Cloudera Impala 2.1.x for CDH 4	Cloudera Manager 4.8.0 - 4.x.x, Cloudera Manager 5.0.0 - 5.x.x	CDH 4.1.0 and later 4.x.x	No
Cloudera Impala 2.0.4	Cloudera Manager 5.0.0 - 5.x.x	CDH 5.2.5	CDH 5.2.5
Cloudera Impala 2.0.3	Cloudera Manager 5.0.0 - 5.x.x	CDH 5.2.4	CDH 5.2.4
Cloudera Impala 2.0.2	Cloudera Manager 5.0.0 - 5.x.x	CDH 5.2.3	CDH 5.2.3
Cloudera Impala 2.0.1	Cloudera Manager 5.0.0 - 5.x.x	CDH 5.2.1	CDH 5.2.1
Cloudera Impala 2.0.0	Cloudera Manager 5.0.0 - 5.x.x	CDH 5.2.0	CDH 5.2.0

Product	Supported Cloudera Manager versions	Supported CDH versions	Integrated into CDH
Cloudera Impala 2.0.x for CDH 4	Cloudera Manager 4.8.0 - 4.x.x, Cloudera Manager 5.0.0 - 5.x.x	CDH 4.1.0 and later 4.x.x	No
Cloudera Impala 1.4.4	Cloudera Manager 5.0.0 - 5.x.x	CDH 5.1.5	CDH 5.1.5
Cloudera Impala 1.4.3	Cloudera Manager 5.0.0 - 5.x.x	CDH 5.1.4	CDH 5.1.4
Cloudera Impala 1.4.2	Cloudera Manager 5.0.0 - 5.x.x	CDH 5.1.3	CDH 5.1.3
Cloudera Impala 1.4.1	Cloudera Manager 5.0.0 - 5.x.x	CDH 5.1.2	CDH 5.1.2
Cloudera Impala 1.4.0	Cloudera Manager 5.0.0 - 5.x.x; Recommended: Cloudera Manager 5.1.0	CDH 5.1.0	CDH 5.1.0
Cloudera Impala 1.4.x for CDH 4	Cloudera Manager 4.8.0 - 4.x.x, Cloudera Manager 5.0.0 - 5.x.x	CDH 4.1.0 - 4.x.x	No
Cloudera Impala 1.3.3	Cloudera Manager 5.0.0 - 5.x.x	CDH 5.0.5	CDH 5.0.5
Cloudera Impala 1.3.2	Cloudera Manager 5.0.0 - 5.x.x	CDH 5.0.4	CDH 5.0.4
Cloudera Impala 1.3.1	Cloudera Manager 4.8.0 - 4.x.x, Cloudera Manager 5.0.0 - 5.x.x	CDH 4.1.0 and later 4.x.x, CDH 5.0.1 - 5.0.x	CDH 5.0.1
Cloudera Impala 1.3.0	Cloudera Manager 5.0.0 - 5.x.x	CDH 5.0.0	CDH 5.0.0
Cloudera Impala 1.3.x for CDH 4	Cloudera Manager 4.8.0 - 4.x.x, Cloudera Manager 5.0.0 - 5.x.x	CDH 4.1.0 and later 4.x.x	No
Cloudera Impala 1.2.4	Cloudera Manager 4.8.0 - 4.x.x, Cloudera Manager 5.0.0 - 5.x.x	CDH 4.1.0 - 4.x.x	No
Cloudera Impala 1.2.3	Cloudera Manager 4.8.0 - 4.x.x, Cloudera Manager 5.0.0 - 5.x.x	CDH 4.1.0 - 4.x.x, CDH 5 Beta 2	CDH 5.0.0 Beta 2
Cloudera Impala 1.2.2	Cloudera Manager 4.8.0 - 4.x.x, Cloudera Manager 5.0.0 - 5.x.x	CDH 4.1.0 - 4.x.x	No
Cloudera Impala 1.2.1	Cloudera Manager 4.8.0 - 4.x.x, Cloudera Manager 5.0.0 - 5.x.x	CDH 4.1.0 - 4.x.x	No
Cloudera Impala 1.2.0 Beta	Cloudera Manager 5.0.0 Beta 1	CDH 5.0.0 Beta 1	CDH 5.0.0 Beta 1

Product	Supported Cloudera Manager versions	Supported CDH versions	Integrated into CDH
Cloudera Impala 1.1.x	Cloudera Manager 4.6.0 - 4.x.x (Does not work with Cloudera Manager 4.8.0)	CDH 4.1.0 - 4.x.x	No
Cloudera Impala 1.0.x	Cloudera Manager 4.5.2 - 4.x.x	CDH 4.1.0 - 4.x.x	No

Cloudera Navigator

This matrix contains compatibility information across versions of Cloudera Navigator, Cloudera Manager, and CDH. For detailed information on each release, see [Cloudera Navigator documentation](#).

Product	Feature	Lowest supported Cloudera Manager version	Lowest supported CDH version	Lowest supported Impala version	Lowest supported Search version	Integrated into CDH
Cloudera Navigator 2.2.x	Audit, Metadata and Security Components	5.3.0	<ul style="list-style-type: none"> Audit Component <ul style="list-style-type: none"> Hue - 4.2.0 Hive - 4.2.0, 4.4.0 for operations denied due to lack of privileges. HDFS, HBase - 4.0.0 Sentry - 5.1.0 Metadata Component <ul style="list-style-type: none"> HDFS, Hive, Oozie, MapReduce, Sqoop 1, YARN - 4.4.0 Pig - 4.6.0 	Impala 1.2.1 with CDH 4.4.0	Not Supported	No
Cloudera Navigator 2.1.x	Audit, Metadata and Security Components	5.2.0	<ul style="list-style-type: none"> Audit Component <ul style="list-style-type: none"> Hue - 4.2.0 	Impala 1.2.1 with CDH 4.4.0	Not Supported	No

Product	Feature	Lowest supported Cloudera Manager version	Lowest supported CDH version	Lowest supported Impala version	Lowest supported Search version	Integrated into CDH
			<ul style="list-style-type: none"> – Hive - 4.2.0, 4.4.0 for operations denied due to lack of privileges. – HDFS, HBase - 4.0.0 – Sentry - 5.1.0 ▪ Metadata Component <ul style="list-style-type: none"> – HDFS, Hive, Oozie, MapReduce, Sqoop 1, YARN - 4.4.0 – Pig - 4.6.0 			
Cloudera Navigator 2.0.1	Audit, Metadata, and Security Components	5.1.2	<ul style="list-style-type: none"> ▪ Audit Component <ul style="list-style-type: none"> – Hue - 4.2.0 – Hive - 4.2.0, 4.4.0 for operations denied due to lack of privileges. – HDFS, HBase - 4.0.0 – Sentry - 5.1.0 ▪ Metadata Component <ul style="list-style-type: none"> – HDFS, Hive, Oozie, MapReduce, 	Impala 1.2.1 with CDH 4.4.0	Not Supported	No

Product Compatibility Matrix

Product	Feature	Lowest supported Cloudera Manager version	Lowest supported CDH version	Lowest supported Impala version	Lowest supported Search version	Integrated into CDH
			Sqoop 1, YARN - 4.4.0 – Pig - 4.6.0			
Cloudera Navigator 2.0.0	Audit, Metadata, and Security Components	5.1.0	<ul style="list-style-type: none"> Audit Component <ul style="list-style-type: none"> Hue - 4.2.0 Hive - 4.2.0, 4.4.0 for operations denied due to lack of privileges. HDFS, HBase - 4.0.0 Sentry - 5.1.0 Metadata Component <ul style="list-style-type: none"> HDFS, Hive, Oozie, MapReduce, Sqoop 1, YARN - 4.4.0 Pig - 4.6.0 	Impala 1.2.1 with CDH 4.4.0	Not Supported	No
Cloudera Navigator 1.2.x	Audit Component	5.0.0	Hue - 4.2.0 Hive - 4.2.0, 4.4.0 for operations denied due to lack of privileges. HDFS, HBase - 4.0.0	Impala 1.2.1 with CDH 4.4.0	Not Supported	No
	Metadata Component (2.0 beta 2)	5.0.0	4.4.0 and higher for all components except Pig. For Pig, 4.6.0 and higher.	Not Supported	Not Supported	No

Product	Feature	Lowest supported Cloudera Manager version	Lowest supported CDH version	Lowest supported Impala version	Lowest supported Search version	Integrated into CDH
Cloudera Navigator 1.1.x	Audit Component	4.8.0 and 4.7.0	Hive, Hue - 4.2.0 HDFS, HBase - 4.0.0	Impala 1.1.1 with CDH 4.4.0	Not Supported	No
Cloudera Navigator 1.0.x	Audit Component	4.6.0 and 4.5.0	Hive, Hue - 4.2.0 HDFS, HBase - 4.0.0	Not Supported	Not Supported	No

Cloudera Search

This topic contains compatibility information across versions of Cloudera Search and CDH/Cloudera Manager. For detailed documentation, see [Cloudera Search with CDH 4](#) and [Cloudera Search with CDH 5](#). Compatibility information for Cloudera Search depends on the version of CDH you are using.

Cloudera Search for CDH 4

Product	Lowest supported Cloudera Manager version	Supported CDH version	Lowest supported Impala version	Integrated into CDH
Cloudera Search 1.3.0	Cloudera Manager 4.8.0	CDH 4.7.0	NA	No
Cloudera Search 1.2.0	Cloudera Manager 4.8.0	CDH 4.6.0	NA	No
Cloudera Search 1.1.x	Cloudera Manager 4.8.0	CDH 4.5.0	NA	No
Cloudera Search 1.0.x	Cloudera Manager 4.6.0	CDH 4.3.0 & CDH 4.4.0	NA	No

Cloudera Search for CDH 5

Since Cloudera Search has been integrated into the CDH 5 package, its compatibility with Cloudera Manager depends on the CDH 5.x.x release it is shipped with. For more information, see [CDH and Cloudera Manager Compatibility](#).

Apache Sentry (incubating)

Sentry enables role-based, fine-grained authorization for HiveServer2 and provides classic database-style authorization for Hive, Cloudera Impala and Cloudera Search. You can use either the Sentry service (introduced in Cloudera Manager 5.1.0 and CDH 5.1.0) or the policy file approach to secure your data. For more information, refer the [Cloudera](#) documentation.

- **Note:** It's possible for a single cluster to use both, the Sentry service (for Hive and Impala) and Sentry policy files (for Solr).

Product	Feature	Lowest supported Cloudera Manager version	Lowest supported CDH version	Lowest supported Impala version	Lowest supported Search version	Integrated into CDH
Apache Sentry (incubating)	Sentry Service	Cloudera Manager 5.1.0	CDH 5.1.0	Impala 1.4.0 for CDH 5	Sentry service not supported. Use policy files instead.	Yes
	Policy File	Cloudera Manager 4.7.0	CDH 4.3.0	Impala 1.2.1	Search 1.1.0	Yes; Starting CDH 4.4.0

Apache Spark

Spark is a fast, general engine for large-scale data processing. For installation and configuration instructions, see [Spark Installation](#). To see new features introduced with each release, refer the [CDH 5 Release Notes](#) on page 5.

Product	Supported Cloudera Manager version	Supported CDH version	Lowest supported Impala version	Lowest supported Search version	Integrated into CDH
Apache Spark 1.2.x	Cloudera Manager 5.3.x	CDH 5.3.x	Not Supported	Not Supported	CDH 5.3.0
Apache Spark 1.1.x	Cloudera Manager 5.2.x	CDH 5.2.x	Not Supported	Not Supported	CDH 5.2.0
Apache Spark 1.0.x	Cloudera Manager 5.1.x	CDH 5.1.x	Not Supported	Not Supported	CDH 5.1.0
Apache Spark 0.9.x	Cloudera Manager 5.0.x	CDH 5.0.x	Not Supported	Not Supported	CDH 5.0.0