cloudera®

# CDS Powered by Apache Spark

**Important Notice**

© 2010-2018 Cloudera, Inc. All rights reserved.

Cloudera, the Cloudera logo, 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. If this documentation includes code, including but not limited to, code examples, Cloudera makes this available to you under the terms of the Apache License, Version 2.0, including any required notices. A copy of the Apache License Version 2.0, including any notices, is included herein. A copy of the Apache License Version 2.0 can also be found here: https://opensource.org/licenses/Apache-2.0

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.**
**395 Page Mill Road**
**Palo Alto, CA 94306**
**info@cloudera.com**
**US: 1-888-789-1488**
**Intl: 1-650-362-0488**
**www.cloudera.com**

**Release Information**

Version: CDS 2.1.x Powered by Apache Spark
Date: November 2, 2018

# Table of Contents

# CDS 2 Powered by Apache Spark Overview

> **Note:**
>
> This documentation refers to CDS 2.1 Release 3 powered by Apache Spark. This component is generally available and is supported on CDH 5.7 and higher.
>
> A Hive compatibility issue in CDS 2.0 Release 1 affects CDH 5.10.1 and higher, CDH 5.9.2 and higher, CDH 5.8.5 and higher, and CDH 5.7.6 and higher. If you are using one of these CDH versions, you must upgrade to CDS 2.0 Release 2 or higher to avoid Spark 2 job failures when using Hive functionality.

Apache Spark is a general framework for distributed computing that offers high performance for both batch and interactive processing. It exposes APIs for Java, Python, and Scala.

For detailed API information, see the Apache Spark project site.

> **Note:** Although this document makes some references to the external Spark site, not all the features, components, recommendations, and so on are applicable to Spark when used on CDH. Always cross-check the Cloudera documentation before building a reliance on some aspect of Spark that might not be supported or recommended by Cloudera. In particular, see Spark 2 Known Issues on page 9 for components and features to avoid.

CDS 2 Powered by Apache Spark consists of Spark core and several related projects:

**Spark SQL**

Module for working with structured data. Allows you to seamlessly mix SQL queries with Spark programs.

**Spark Streaming**

API that allows you to build scalable fault-tolerant streaming applications.

**MLlib**

API that implements common machine learning algorithms.

Cloudera distributes these versions of Apache Spark: 1.6, 2.0, and 2.1.

Spark 1.6 is distributed as part of Cloudera Enterprise 5.7.x and higher. The latest documentation is available at Cloudera Enterprise documentation.

This document describes the separately released CDS 2.1 Powered by Apache Spark. Spark 2 is shipped separately for ease of use and convenience of consumption. It enables customers to install and upgrade Spark 2 without going through a full upgrade of the CDH cluster.

A Spark 1.6 service can coexist with a Spark 2.1 service. The configurations of the two services do not conflict and both services use the same YARN service. The port of the Spark History Server is 18088 for Spark 1.6 and 18089 for Spark 2.

## Unsupported Features

Consult Spark 2 Known Issues on page 9 for a comprehensive list of features that are not supported with the CDS 2 Powered by Apache Spark.

## Related Information

- Cloudera Spark forum
- Apache Spark documentation

# CDS 2 Powered by Apache Spark Release Notes

The release notes provide information on requirements, new and changed features, known issues, and fixed issues, and version and packaging information for CDS 2 Powered by Apache Spark.

## CDS Powered by Apache Spark Requirements

The following sections describe software requirements for CDS Powered by Apache Spark.

### CDH Versions

> **Important:** CDS Powered by Apache Spark is available in parcel format only, and not packages. Because Cloudera Manager does not support using parcels and packages in the same cluster, you cannot use CDS if you are using a package-based installation of CDH.
>
> The CDS parcel version displayed in Cloudera Manager, which is also part of the parcel file name, is structured as follows:
>
> ***<CDS_version>*-1.*<cdh_build_version>*.p*<patch_version>*.*<build_number>***
>
> The *<cdh_build_version>* portion is the version of CDH upon which the release was built. It is *not* the minimum supported CDH version. For example, although CDS 2.1 Release 3 was built on CDH 5.13.3, it is still supported on CDH 5.7 and higher CDH 5 versions.

Supported versions of CDH are described below.

A Hive compatibility issue in CDS 2.0 Release 1 affects CDH 5.10.1 and higher, CDH 5.9.2 and higher, CDH 5.8.5 and higher, and CDH 5.7.6 and higher. If you are using one of these CDH versions, you must upgrade to CDS 2.0 Release 2 or higher to avoid Spark 2 job failures when using Hive functionality.

| CDS Powered by Apache Spark Version | CDH Version |
|---|---|
| 2.1 Release 3 | CDH 5.7, CDH 5.8, CDH 5.9, CDH 5.10, CDH 5.11, CDH 5.12, and any higher CDH 5.x versions |
| 2.1 Release 2 | CDH 5.7, CDH 5.8, CDH 5.9, CDH 5.10, CDH 5.11, CDH 5.12, and any higher CDH 5.x versions |
| 2.1 Release 1 | CDH 5.7, CDH 5.8, CDH 5.9, CDH 5.10, CDH 5.11, CDH 5.12 |
| 2.0 Release 2 | CDH 5.7, CDH 5.8, CDH 5.9, CDH 5.10, CDH 5.11 |
| 2.0 Release 1 | CDH 5.7 up to 5.7.5, CDH 5.8 up to 5.8.4, CDH 5.9 up to 5.9.1, CDH 5.10.0. Spark 2.0 Release 2 is required for any higher maintenance releases in any of these CDH versions. |

A Spark 1.6 service (included in CDH 5.7 and higher) can co-exist on the same cluster as Spark 2 (installed as a separate parcel). The two services are configured to not conflict, and both run on the same YARN service. Spark 2 uses the external shuffle service from the CDH installation if Spark 1 is already installed, or installs the shuffle service itself if necessary. Only the external shuffle service classes from the CDH installation can be used.

Although Spark 1 and Spark 2 can coexist in the same CDH cluster, you cannot use multiple Spark 2 versions simultaneously in the same Cloudera Manager instance. All CDH clusters managed by the same Cloudera Manager Server must use exactly the same version of CDS Powered by Apache Spark. For example, you cannot use the built-in CDH Spark service, a CDS 2.0 service, and a CDS 2.1 service. You must choose only one CDS 2 Powered by Apache Spark release. Make sure to install or upgrade the CDS 2 service descriptor and parcels across all machines of all clusters at the same time.

### Cloudera Manager Versions

**Important:**

Because CDS Powered by Apache Spark is only installable using the parcel mechanism, it can only be used on clusters managed by Cloudera Manager. Additionally, because Cloudera Manager does not support using parcels and packages in the same cluster, you cannot use CDS if you are using a package-based installation of CDH.

Applicable versions of Cloudera Manager for Spark 2 are described below.

| CDS Powered by Apache Spark Version | Cloudera Manager Version |
|---|---|
| 2.1 Release 3 | Cloudera Manager 5.8.3, 5.9 and higher |
| 2.1 Release 2 | Cloudera Manager 5.8.3, 5.9 and higher |
| 2.1 Release 1 | Cloudera Manager 5.8.3, 5.9 and higher |
| 2.0 Release 2 | Cloudera Manager 5.8.3, 5.9 and higher |
| 2.0 Release 1 | Cloudera Manager 5.8.3, 5.9 and higher |

### Scala 2.11 Requirement

Spark 2 does not work with Scala 2.10. Use Scala 2.11 only.

### Python Requirement

CDS Powered by Apache Spark requires one of the following Python versions:

- Python 2.7 or higher, when using Python 2.
- Python 3.4 or higher, when using Python 3. (CDS 2.0 only supports Python 3.4 and 3.5; CDS 2.1 includes support for Python 3.6 and higher.)

## Spark 2 New Features and Changes

The following sections describe what's new and changed in each CDS 2 Powered by Apache Spark release.

### What's New in CDS 2.1 Release 3

This is purely a maintenance release. See CDS Powered by Apache Spark Fixed Issues on page 12 for the list of fixed issues.

### What's New in CDS 2.1 Release 2

This is purely a maintenance release. See CDS Powered by Apache Spark Fixed Issues on page 12 for the list of fixed issues.

### What's New in CDS 2.1 Release 1

- New direct connector to Kafka that uses the new Kafka consumer API. See Spark 2 Kafka Integration on page 43 for details.

### What's New in CDS 2.0 Release 2

- A Hive compatibility issue in CDS 2.0 Release 1 affects CDH 5.10.1 and higher, CDH 5.9.2 and higher, CDH 5.8.5 and higher, and CDH 5.7.6 and higher. If you are using one of these CDH versions, you must upgrade to CDS 2.0 Release 2 or higher to avoid Spark 2 job failures when using Hive functionality.

### What's New in CDS 2.0 Release 1

- New `SparkSession` object replaces `HiveContext` and `SQLContext`.
  - Most of the Hive logic has been reimplemented in Spark.
  - Some Hive dependencies still exist:
    - SerDe support.
    - UDF support.
- Added support for the unified Dataset API.
- Faster Spark SQL achieved with whole stage code generation.
- More complete SQL syntax now supports subqueries.
- Adds the `spark-csv` library.
- Backport of SPARK-5847. The root for metrics is now the app name (`spark.app.name`) instead of the app ID. The app ID requires investigation to match to the app name, and changes when streaming jobs are stopped and restarted.

## Spark 2 Known Issues

The following sections describe the current known issues and limitations in CDS 2 Powered by Apache Spark. In some cases, a feature from the upstream Apache Spark project is currently not considered reliable enough to be supported by Cloudera. For a number of integration features in CDH that rely on Spark, the feature does not work with CDS 2 because CDH components are not introducing dependencies on Spark 2.

### Spark SQL does not respect Sentry ACLs when communicating with Hive metastore

Even if user is configured via Sentry to not have read permission to a Hive table, a Spark SQL job running as that user can still read the table's metadata directly from the Hive metastore.

**Cloudera Bug:** CDH-33658

### Empty result when reading Parquet table created by saveAsTable()

After a Parquet table is created by the `saveAsTable()` function, Spark SQL queries against the table return an empty result set. The issue is caused by the “path” property of the table not being written to the Hive metastore during the `saveAsTable()` call.

**Cloudera Bug:** CDH-60037

**Affects:** CDS 2.2 Release 1

**Severity:** High

**Workaround:** You can set the path manually before the call to `saveAsTable()`:

```
val options = Map("path" -> "/path/to/hdfs/directory/containing/table")
df.write.options(options).saveAsTable("db_name.table_name")
```

Or you can add the path to the metastore when the table already exists, for example:

```
spark.sql("alter table db_name.table_name set SERDEPROPERTIES
('path'='hdfs://host.example.com:8020/warehouse/path/db_name.db/table_name')")
spark.catalog.refreshTable("db_name.table_name")
```

**Resolution:** Upgrade to CDS 2.2 release 2, which contains the fix.

## Spark 2 Version Requirement for Clusters Managed by Cloudera Manager

> **Important:**
>
> Because CDS Powered by Apache Spark is only installable using the parcel mechanism, it can only be used on clusters managed by Cloudera Manager. Additionally, because Cloudera Manager does not support using parcels and packages in the same cluster, you cannot use CDS if you are using a package-based installation of CDH.

Although Spark 1 and Spark 2 can coexist in the same CDH cluster, you cannot use multiple Spark 2 versions simultaneously in the same Cloudera Manager instance. All CDH clusters managed by the same Cloudera Manager Server must use exactly the same version of CDS Powered by Apache Spark. For example, you cannot use the built-in CDH Spark service, a CDS 2.0 service, and a CDS 2.1 service. You must choose only one CDS 2 Powered by Apache Spark release. Make sure to install or upgrade the CDS 2 service descriptor and parcels across all machines of all clusters at the same time.

## Spark Standalone

Spark Standalone is not supported for Spark 2.

## HiveOnSpark is not Supported with Spark 2

The HiveOnSpark module is a CDH 5 component that has a dependency on Apache Spark 1.6. Because CDH 5 components do not have any dependencies on Spark 2, the HiveOnSpark module does not work with CDS Powered by Apache Spark. You can still use Spark 2 with Hive using other methods.

## SparkOnHBase is not Supported with Spark 2

The SparkOnHBase module is a CDH 5 component that has a dependency on Apache Spark 1.6. Because CDH 5 components do not have any dependencies on Spark 2, the SparkOnHBase module does not work with CDS Powered by Apache Spark. You can still use Spark 2 with HBase using other methods.

## Using the JDBC Datasource API to access Hive or Impala is not supported

## Dynamic allocation and Spark Streaming

If you are using Spark Streaming, Cloudera recommends that you disable dynamic allocation by setting `spark.dynamicAllocation.enabled` to `false` when running streaming applications.

## Structured Streaming is not supported

Cloudera does not support the Structured Streaming API.

## Oozie Spark2 Action is not Supported

The Oozie Spark action is a CDH component that has a dependency on Spark 1.6. Because CDH components do not have any dependencies on Spark 2, the Oozie Spark action does not work with Spark 2.

## SparkR is not Supported

SparkR is not supported for Spark 2. (SparkR is also not supported in CDH with Spark 1.6.)

## GraphX is not Supported

GraphX is not supported for Spark 2. (GraphX is also not supported in CDH with Spark 1.6.)

### Thrift Server

The Thrift JDBC/ODBC server is not supported for Spark 2. (The Thrift server is also not supported in CDH with Spark 1.6.)

### Spark SQL CLI is not Supported

The Spark SQL CLI is not supported for Spark 2. (The Spark SQL CLI is also not supported in CDH with Spark 1.6.)

### Rolling Upgrades are not Supported

Rolling upgrades are not possible from Spark 1.6 bundled with CDH, to CDS 2.

### Package Install is not Supported

CDS 2 is only installable as a parcel.

### Spark Avro is not Supported

The `spark-avro` library is not integrated into the CDS 2.x parcel.

### Accessing Multiple Clusters Simultaneously Not Supported

Spark does not support accessing multiple clusters in the same application.

### Hardware Acceleration for MLlib is not Supported

This feature, part of the GPL Extras package for CDH, is not supported with CDS 2 Powered By Apache Spark. This feature is supported for Spark 1.6.

### Spark Gateway Roles Needs to be Added to Every Host

If you have hosts with the NodeManager role but without the Spark Gateway role, you must add the Spark 2 Gateway role to all NodeManager hosts and redeploy the client configurations. If you do not use this workaround, Spark fails to locate the `topology.py` file, which can cause task localization issues and failures for very large jobs.

Fixed in Spark 2.1 release 2.

### Long-running apps on a secure cluster might fail if driver is restarted

If you submit a long-running app on a secure cluster using the `--principal` and `--keytab` options in cluster mode, and a failure causes the driver to restart after 7 days (the default maximum HDFS delegation token lifetime), the new driver fails with an error similar to the following:

```
Exception in thread "main"
org.apache.hadoop.ipc.RemoteException(org.apache.hadoop.security.token.SecretManager$InvalidToken):
 token <token_info> can't be found in cache
```

**Workaround:** None

**Affected Versions:** All CDS 2.0, 2.1, and 2.2 releases

**Fixed Versions:** CDS 2.3 Release 2

**Apache Issue:** SPARK-23361

**Cloudera Issue:** CDH-64865

## Spark 2 Incompatible Changes

The following sections describe changes in CDS 2 Powered by Apache Spark that might require special handling during upgrades, or code changes within existing applications.

### Incompatible Changes in CDS 2.1

No new incompatible changes in this release.

### Incompatible Changes in CDS 2.0

- `HiveContext` and `SQLContext` have been removed.
- DataFrames have been removed from the Scala API. DataFrame is now a special case of Dataset.
- Spark 2.0 and higher do not use an assembly JAR for standalone applications.

## CDS Powered by Apache Spark Fixed Issues

The following sections describe the issues fixed in each CDS 2 Powered by Apache Spark release.

### Issues Fixed in CDS 2.1 Release 3

The following list includes issues fixed in CDS 2.1 Release 3. Test-only changes are omitted.

- [SPARK-23207][SPARK-22905][SPARK-24564][SPARK-25114][SQL][BACKPORT-2.1] Shuffle+Repartition on a DataFrame could lead to incorrect answers
- [SPARK-24950][SQL] DateTimeUtilsSuite daysToMillis and millisToDays fails w/java 8 181-b13
- [PYSPARK] Updates to Accumulators
- [SPARK-24809][SQL] Serializing LongToUnsafeRowMap in executor may result in data error
- [SPARK-20223][SQL] Fix typo in tpcds q77.sql
- [SPARK-24589][CORE] Correctly identify tasks in output commit coordinator [branch-2.1].
- [SPARK-22897][CORE] Expose stageAttemptId in TaskContext
- [SPARK-23732][DOCS] Fix source links in generated scaladoc.
- [WEBUI] Avoid possibility of script in query param keys
- Fix compilation caused by SPARK-24257
- [SPARK-24257][SQL] LongToUnsafeRowMap calculate the new size may be wrong
- [R][BACKPORT-2.2] backport lint fix
- [SPARKR] Match pyspark features in SparkR communication protocol.
- [PYSPARK] Update py4j to version 0.10.7.
- [SPARK-21278][PYSPARK] Upgrade to Py4J 0.10.6
- [SPARK-23697][CORE] LegacyAccumulatorWrapper should define isZero correctly
- [SPARK-23053][CORE][BRANCH-2.1] taskBinarySerialization and task partitions calculate in DagScheduler.submitMissingTasks should keep the same RDD checkpoint status
- [SPARK-22862] Docs on lazy elimination of columns missing from an encoder
- [SPARK-22688][SQL] Upgrade Janino version to 3.0.8
- [SPARK-22373][BUILD][FOLLOWUP][BRANCH-2.1] Updates other dependency lists too for Janino
- [SPARK-22373] Bump Janino dependency version to fix thread safety issue…
- [SPARK-22548][SQL] Incorrect nested AND expression pushed down to JDBC data source
- [SPARK-22377][BUILD] Use /usr/sbin/lsof if lsof does not exists in release-build.sh
- [SPARK-22327][SPARKR][TEST][BACKPORT-2.1] check for version warning
- [SPARK-22429][STREAMING] Streaming checkpointing code does not retry after failure
- [MINOR][DOC] automatic type inference supports also Date and Timestamp
- [SPARK-21991][LAUNCHER][FOLLOWUP] Fix java lint
- [SPARK-21991][LAUNCHER] Fix race condition in LauncherServer#acceptConnections
- [SPARK-22273][SQL] Fix key/value schema field names in HashMapGenerators.
- [SPARK-22206][SQL][SPARKR] gapply in R can't work on empty grouping columns
- [SPARK-20466][CORE] HadoopRDD#addLocalConfiguration throws NPE

- [SPARK-22167][R][BUILD] sparkr packaging issue allow zinc
- [SPARK-22129][SPARK-22138] Release script improvements
- [SPARK-18136] Fix SPARK_JARS_DIR for Python pip install on Windows
- [SPARK-22072][SPARK-22071][BUILD] Improve release build scripts
- [SPARK-19318][SPARK-22041][SPARK-16625][BACKPORT-2.1][SQL] Docker test case failure: `: General data types to be mapped to Oracle`
- [SPARK-22052] Incorrect Metric assigned in MetricsReporter.scala
- [SPARK-22043][PYTHON] Improves error message for show_profiles and dump_profiles
- [SPARK-21953] Show both memory and disk bytes spilled if either is present
- [SPARK-21985][PYSPARK] PairDeserializer is broken for double-zipped RDDs
- [SPARK-21976][DOC] Fix wrong documentation for Mean Absolute Error.
- [SPARK-21950][SQL][PYTHON][TEST] pyspark.sql.tests.SQLTests2 should stop SparkContext.
- [SPARK-21826][SQL][2.1][2.0] outer broadcast hash join should not throw NPE
- [MINOR] Correct validateAndTransformSchema in GaussianMixture and AFTSurvivalRegression
- [SPARK-18752][SQL] Follow-up: add scaladoc explaining isSrcLocal arg.
- [SPARK-18752][HIVE] isSrcLocal" value should be set from user query.
- [CDH-70445] Executor Plugin Api.
- [SPARK-23852][SQL] Add test that fails if PARQUET-1217 is not fixed.
- [CDH-68516] Check for null when writing decimal.
- [CDH-69165] Handle file names with spaces in classpath.
- [SPARK-23991][DSTREAMS] Fix data loss when WAL write fails in allocateBlocksToBatch
- [SPARK-24309][CORE] AsyncEventQueue should stop on interrupt.
- [SPARK-22850][CORE] Ensure queued events are delivered to all event queues.
- [CDH-68051] Try to fetch tokens for all KMS servers.
- [SPARK-23433][CORE] Late zombie task completions update all tasksets
- [SPARK-23660] Fix exception in yarn cluster mode when application ended fast
- [SPARK-23438][DSTREAMS] Fix DStreams data loss with WAL when driver crashes
- [SPARK-18971][CORE] Upgrade Netty to 4.0.43.Final
- [SPARK-21551][PYTHON] Increase timeout for PythonRDD.serveIterator

## Issues Fixed in CDS 2.1 Release 2

The following list includes issues fixed in CDS 2.1 Release 2. Test-only changes are omitted.

- [SPARK-19554][UI,YARN] Allow SHS URL to be used for tracking in YARN RM.
- [SPARK-22083][CORE] Release locks in MemoryStore.evictBlocksToFreeSpace
- [SPARK-18838][HOTFIX][YARN] Check internal context state before stopping it.
- [SPARK-18838][CORE] Add separate listener queues to LiveListenerBus.
- [SPARK-21928][CORE] Set classloader on SerializerManager's private kryo
- [SPARK-21254][WEBUI] History UI performance fixes
- [SPARK-21135][WEB UI] On history server page� duration of incompleted applications should be hidden instead of showing up as 0
- [SPARK-20942][WEB-UI] The title style about field is error in the history server web ui.
- [SPARK-20603][SS][TEST] Set default number of topic partitions to 1 to reduce the load
- [SPARK-18682][SS] Batch Source for Kafka
- [SPARK-19155][ML] MLlib GeneralizedLinearRegression family and link should case insensitive
- [SPARK-19542][HOTFIX][SS]Fix the missing import in DataStreamReaderWriterSuite
- [SPARK-20280][CORE] FileStatusCache Weigher integer overflow
- [SPARK-19646][BUILD][HOTFIX] Fix compile error from cherry-pick of SPARK-19646 into branch 2.1
- [SPARK-21834] Incorrect executor request in case of dynamic allocation
- [SPARK-21721][SQL][BACKPORT-2.1] Clear FileSystem deleteOnExit cache when paths are successfully removed
- [SPARK-21588][SQL] SQLContext.getConf(key, null) should return null

- [SPARK-21330][SQL] Bad partitioning does not allow to read a JDBC table with extreme values on the partition column
- [SPARK-12717][PYTHON][BRANCH-2.1] Adding thread-safe broadcast pickle registry
- [SPARK-21555][SQL] RuntimeReplaceable should be compared semantically by its canonicalized child
- [SPARK-21306][ML] For branch 2.1, OneVsRest should support setWeightCol
- [SPARK-21446][SQL] Fix setAutoCommit never executed
- [SPARK-21441][SQL] Incorrect Codegen in SortMergeJoinExec results failures in some cases
- [SPARK-21332][SQL] Incorrect result type inferred for some decimal expressions
- [SPARK-21344][SQL] BinaryType comparison does signed byte array comparison
- [SPARK-21083][SQL][BRANCH-2.1] Store zero size and row count when analyzing empty table
- [SPARK-21345][SQL][TEST][TEST-MAVEN][BRANCH-2.1] SparkSessionBuilderSuite should clean up stopped sessions.
- [SPARK-21312][SQL] correct offsetInBytes in UnsafeRow.writeToStream
- [SPARK-20256][SQL][BRANCH-2.1] SessionState should be created more lazily
- [SPARK-19104][BACKPORT-2.1][SQL] Lambda variables in ExternalMapToCatalyst should be global
- [SPARK-21203][SQL] Fix wrong results of insertion of Array of Struct
- [SPARK-20555][SQL] Fix mapping of Oracle DECIMAL types to Spark types in read path
- [SPARK-21181] Release byteBuffers to suppress netty error messages
- [SPARK-21167][SS] Decode the path generated by File sink to handle special characters
- [SPARK-21138][YARN] Cannot delete staging dir when the clusters of "spark.yarn.stagingDir" and "spark.hadoop.fs.defaultFS" are different
- [SPARK-19688][STREAMING] Not to read `spark.yarn.credentials.file` from checkpoint.
- [SPARK-21114][TEST][2.1] Fix test failure in Spark 2.1/2.0 due to name mismatch
- [SPARK-21072][SQL] TreeNode.mapChildren should only apply to the children node.
- [SPARK-16251][SPARK-20200][CORE][TEST] Flaky test: org.apache.spark.rdd.LocalCheckpointSuite.missing checkpoint block fails with informative message
- [SPARK-20211][SQL][BACKPORT-2.2] Fix the Precision and Scale of Decimal Values when the Input is BigDecimal between -1.0 and 1.0
- [SPARK-21064][CORE][TEST] Fix the default value bug in NettyBlockTransferServiceSuite
- [SPARK-20920][SQL] ForkJoinPool pools are leaked when writing hive tables with many partitions
- [SPARK-20940][CORE] Replace IllegalAccessError with IllegalStateException
- [SPARK-20275][UI] Do not display "Completed" column for in-progress applications
- [SPARK-20868][CORE] UnsafeShuffleWriter should verify the position after FileChannel.transferTo
- [SPARK-20250][CORE] Improper OOM error when a task been killed while spilling data
- [SPARK-20848][SQL] Shutdown the pool after reading parquet files
- [SPARK-20862][MLLIB][PYTHON] Avoid passing float to ndarray.reshape in LogisticRegressionModel
- [SPARK-18406][CORE][BACKPORT-2.1] Race between end-of-task and completion iterator read lock release
- [SPARK-20687][MLLIB] mllib.Matrices.fromBreeze may crash when converting from Breeze sparse matrix
- [SPARK-20798] GenerateUnsafeProjection should check if a value is null before calling the getter
- [SPARK-17424] Fix unsound substitution bug in ScalaReflection.
- [SPARK-20665][SQL] Bround" and "Round" function return NULL
- [SPARK-20685] Fix BatchPythonEvaluation bug in case of single UDF w/ repeated arg.
- [SPARK-20688][SQL] correctly check analysis for scalar sub-queries
- [SPARK-19933][SQL] Do not change output of a subquery
- [SPARK-20631][PYTHON][ML] LogisticRegression._checkThresholdConsistency should use values not Params
- [SPARK-20686][SQL] PropagateEmptyRelation incorrectly handles aggregate without grouping
- [SPARK-17685][SQL] Make SortMergeJoinExec's currentVars is null when calling createJoinKey
- [SPARK-20615][ML][TEST] SparseVector.argmax throws IndexOutOfBoundsException
- [SPARK-20616] RuleExecutor logDebug of batch results should show diff to start of batch
- [SPARK-20546][DEPLOY] spark-class gets syntax error in posix mode
- [SPARK-20558][CORE] clear InheritableThreadLocal variables in SparkContext when stopping it

- [SPARK-20540][CORE] Fix unstable executor requests.
- [SPARK-20517][UI] Fix broken history UI download link
- [SPARK-20404][CORE] Using Option(name) instead of Some(name)
- [SPARK-20451] Filter out nested mapType datatypes from sort order in randomSplit
- [SPARK-20450][SQL] Unexpected first-query schema inference cost with 2.1.1
- [SPARK-20439][SQL][BACKPORT-2.1] Fix Catalog API listTables and getTable when failed to fetch table metadata
- [SPARK-20407][TESTS][BACKPORT-2.1] ParquetQuerySuite 'Enabling/disabling ignoreCorruptFiles' flaky test
- [SPARK-20243][TESTS] DebugFilesystem.assertNoOpenStreams thread race
- [SPARK-20409][SQL] fail early if aggregate function in GROUP BY
- [SPARK-20359][SQL] Avoid unnecessary execution in EliminateOuterJoin optimization that can lead to NPE
- [SPARK-17647][SQL] Fix backslash escaping in 'LIKE' patterns.
- [SPARK-20335][SQL][BACKPORT-2.1] Children expressions of Hive UDF impacts the determinism of Hive UDF
- [SPARK-20131][CORE] Don't use `this` lock in StandaloneSchedulerBackend.stop
- [SPARK-20304][SQL] AssertNotNull should not include path in string representation
- [SPARK-20291][SQL] NaNvl(FloatType, NullType) should not be cast to NaNvl(DoubleType, DoubleType)
- [SPARK-17564][TESTS] Fix flaky RequestTimeoutIntegrationSuite.furtherRequestsDelay
- [SPARK-20270][SQL] na.fill should not change the values in long or integer when the default value is in double
- [SPARK-20264][SQL] asm should be non-test dependency in sql/core
- [SPARK-20260][MLLIB] String interpolation required for error message
- [SPARK-20262][SQL] AssertNotNull should throw NullPointerException
- [SPARK-20246][SQL] should not push predicate down through aggregate with non-deterministic expressions
- [SPARK-20214][ML] Make sure converted csc matrix has sorted indices
- [SPARK-20191][YARN] Crate wrapper for RackResolver so tests can override it.
- [SPARK-20190][APP-ID] applications//jobs' in rest api,status should be [running|s…
- [SPARK-20164][SQL] AnalysisException not tolerant of null query plan.
- [SPARK-20059][YARN] Use the correct classloader for HBaseCredentialProvider
- [SPARK-20134][SQL] SQLMetrics.postDriverMetricUpdates to simplify driver side metric updates
- [SPARK-20043][ML] DecisionTreeModel: ImpurityCalculator builder fails for uppercase impurity type Gini
- [SPARK-20125][SQL] Dataset of type option of map does not work
- [SPARK-19995][YARN] Register tokens to current UGI to avoid re-issuing of tokens in yarn client mode
- [SPARK-20086][SQL] CollapseWindow should not collapse dependent adjacent windows
- [SPARK-19959][SQL] Fix to throw NullPointerException in df[java.lang.Long].collect
- [SPARK-20017][SQL] change the nullability of function 'StringToMap' from 'false' to 'true'
- [SPARK-19912][SQL] String literals should be escaped for Hive metastore partition pruning
- [SPARK-17204][CORE] Fix replicated off heap storage
- [SPARK-19980][SQL][BACKPORT-2.1] Add NULL checks in Bean serializer
- [SPARK-19970][SQL][BRANCH-2.1] Table owner should be USER instead of PRINCIPAL in kerberized clusters
- [SPARK-19994][SQL] Wrong outputOrdering for right/full outer smj
- [SPARK-19946][TESTS][BACKPORT-2.1] DebugFilesystem.assertNoOpenStreams should report the open streams to help debugging
- [SPARK-19872] [PYTHON] Use the correct deserializer for RDD construction for coalesce/repartition
- [SPARK-19887][SQL] dynamic partition keys can be null or empty string
- [SPARK-19944][SQL] Move SQLConf from sql/core to sql/catalyst (branch-2.1)
- [SPARK-19924][SQL][BACKPORT-2.1] Handle InvocationTargetException for all Hive Shim
- [SPARK-19893][SQL] should not run DataFrame set oprations with map type
- [SPARK-19891][SS] Await Batch Lock notified on stream execution exit
- [SPARK-19861][SS] watermark should not be a negative time.
- [SPARK-19813] maxFilesPerTrigger combo latestFirst may miss old files in combination with maxFileAge in FileStreamSource
- [SPARK-18055][SQL] Use correct mirror in ExpresionEncoder

- [SPARK-19348][PYTHON] PySpark keyword_only decorator is not thread-safe
- [SPARK-19859][SS][FOLLOW-UP] The new watermark should override the old one.
- [SPARK-19859][SS] The new watermark should override the old one
- [SPARK-19561][SQL] add int case handling for TimestampType
- [SPARK-19774] StreamExecution should call stop() on sources when a stream fails
- [SPARK-19779][SS] Delete needless tmp file after restart structured streaming job
- [SPARK-19748][SQL] refresh function has a wrong order to do cache invalidate and regenerate the inmemory var for InMemoryFileIndex with FileStatusCache
- [SPARK-19594][STRUCTURED STREAMING] StreamingQueryListener fails to handle QueryTerminatedEvent if more then one listeners exists
- [SPARK-19707][CORE] Improve the invalid path check for sc.addJar
- [SPARK-19674][SQL] Ignore driver accumulator updates don't belong to …
- [SPARK-19691][SQL][BRANCH-2.1] Fix ClassCastException when calculating percentile of decimal column
- [SPARK-19646][CORE][STREAMING] binaryRecords replicates records in scala API
- [SPARK-19500] [SQL] Fix off-by-one bug in BytesToBytesMap
- [SPARK-19622][WEBUI] Fix a http error in a paged table when using a `Go` button to search.
- [SPARK-19603][SS] Fix StreamingQuery explain command
- [SPARK-19329][SQL][BRANCH-2.1] Reading from or writing to a datasource table with a non pre-existing location should succeed
- [SPARK-19501][YARN] Reduce the number of HDFS RPCs during YARN deployment
- [SPARK-17714][CORE][TEST-MAVEN][TEST-HADOOP2.6] Avoid using ExecutorClassLoader to load Netty generated classes
- [SPARK-19542][SS] Delete the temp checkpoint if a query is stopped without errors
- [SPARK-19543] from_json fails when the input row is empty
- [SPARK-19509][SQL] Grouping Sets do not respect nullable grouping columns
- [SPARK-18609][SPARK-18841][SQL][BACKPORT-2.1] Fix redundant Alias removal in the optimizer
- [SPARK-19407][SS] defaultFS is used FileSystem.get instead of getting it from uri scheme
- [SPARK-19472][SQL] Parser should not mistake CASE WHEN(...) for a function call
- [SPARK-19432][CORE] Fix an unexpected failure when connecting timeout
- [SPARK-19377][WEBUI][CORE] Killed tasks should have the status as KILLED
- [SPARK-19378][SS] Ensure continuity of stateOperator and eventTime metrics even if there is no new data in trigger
- [SPARK-19406][SQL] Fix function to_json to respect user-provided options
- [SPARK-19338][SQL] Add UDF names in explain
- [SPARK-18863][SQL] Output non-aggregate expressions without GROUP BY in a subquery does not yield an error
- [SPARK-19330][DSTREAMS] Also show tooltip for successful batches
- [SPARK-19017][SQL] NOT IN subquery with more than one column may return incorrect results
- [SPARK-16473][MLLIB] Fix BisectingKMeans Algorithm failing in edge case
- [SPARK-9435][SQL] Reuse function in Java UDF to correctly support expressions that require equality comparison between ScalaUDF
- [SPARK-19306][CORE] Fix inconsistent state in DiskBlockObject when expection occurred
- [SPARK-19155][ML] Make family case insensitive in GLM
- [SPARK-14536][SQL][BACKPORT-2.1] fix to handle null value in array type column for postgres.
- [SPARK-19267][SS] Fix a race condition when stopping StateStore
- [SPARK-19168][STRUCTURED STREAMING] StateStore should be aborted upon error
- [SPARK-19065][SQL] Don't inherit expression id in dropDuplicates
- [SPARK-18905][STREAMING] Fix the issue of removing a failed jobset from JobScheduler.jobSets
- [SPARK-17237][SQL] Remove backticks in a pivot result schema
- [SPARK-19140][SS] Allow update mode for non-aggregation streaming queries
- [SPARK-19137][SQL] Fix `withSQLConf` to reset `OptionalConfigEntry` correctly

- [SPARK-19765][SPARK-18549][SPARK-19093][SPARK-19736][BACKPORT-2.1][SQL] Backport Three Cache-related PRs to Spark 2.1
- [SPARK-18877][SQL][BACKPORT-2.1] CSVInferSchema.inferField` on DecimalType should find a common type with `typeSoFar`
- [SPARK-18837][WEBUI] Very long stage descriptions do not wrap in the UI
- [SPARK-18972][CORE] Fix the netty thread names for RPC
- [SPARK-18985][SS] Add missing @InterfaceStability.Evolving for Structured Streaming APIs
- [SPARK-18973][SQL] Remove SortPartitions and RedistributeData
- [SPARK-18947][SQL] SQLContext.tableNames should not call Catalog.listTables
- [SPARK-18927][SS] MemorySink for StructuredStreaming can't recover from checkpoint if location is provided in SessionConf
- [SPARK-18899][SPARK-18912][SPARK-18913][SQL] refactor the error checking when append data to an existing table
- [SPARK-18921][SQL] check database existence with Hive.databaseExists instead of getDatabase
- [SPARK-18108][SQL] Fix a schema inconsistent bug that makes a parquet reader fail to read data
- [SPARK-18892][SQL] Alias percentile_approx approx_percentile
- [SPARK-18555][MINOR][SQL] Fix the @since tag when backporting from 2.2 branch into 2.1 branch
- [SPARK-18555][SQL] DataFrameNaFunctions.fill miss up original values in long integers
- [SPARK-18535][SPARK-19720][CORE][BACKPORT-2.1] Redact sensitive information
- [SPARK-19506][ML][PYTHON] Import warnings in pyspark.ml.util
- [SPARK-13669][SPARK-20898][CORE] Improve the blacklist mechanism to handle external shuffle service unavailable situation
- [SPARK-13747][CORE] Add ThreadUtils.awaitReady and disallow Await.ready
- [SPARK-13747][CORE] Fix potential ThreadLocal leaks in RPC when using ForkJoinPool
- [SPARK-21522][CORE] Fix flakiness in LauncherServerSuite.
- [SPARK-20904][CORE] Don't report task failures to driver during shutdown.
- [SPARK-20393][WEBU UI] Strengthen Spark to prevent XSS vulnerabilities
- [SPARK-19146][CORE] Drop more elements when stageData.taskData.size > retainedTasks
- [SPARK-20084][CORE] Remove internal.metrics.updatedBlockStatuses from history files.
- [SPARK-18991][CORE] Change ContextCleaner.referenceBuffer to use ConcurrentHashMap to make it faster
- [SPARK-19185][DSTREAM] Make Kafka consumer cache configurable
- [SPARK-20756][YARN] yarn-shuffle jar references unshaded guava
- [SPARK-20922][CORE][HOTFIX] Don't use Java 8 lambdas in older branches.
- [SPARK-20922][CORE] Add whitelist of classes that can be deserialized by the launcher.
- Add health aggregation to CSD.
- Update Spark2 CSD to not provide "cdh-plugin" since no one in CDH (yarn, in particular) uses Spark2 bits.
- Don't localize topology.py location.
- Relax compatibility to all C5 versions.

## Issues Fixed in CDS 2.1 Release 1

The following list includes issues fixed in CDS 2.1 Release 1. Test-only changes are omitted.

- Preview of: [SPARK-19554][UI,YARN] Allow SHS URL to be used for tracking in YARN RM.
- [SPARK-16554][CORE] Automatically Kill Executors and Nodes when they are Blacklisted
- [SPARK-16654][CORE] Add UI coverage for Application Level Blacklisting
- [SPARK-8425][CORE] Application Level Blacklisting
- [SPARK-18117][CORE] Add test for TaskSetBlacklist
- [SPARK-18949][SQL][BACKPORT-2.1] Add recoverPartitions API to Catalog
- [SPARK-19459][SQL][BRANCH-2.1] Support for nested char/varchar fields in ORC
- [SPARK-19611][SQL] Introduce configurable table schema inference
- [SPARK-19082][SQL] Make ignoreCorruptFiles work for Parquet

- [SPARK-19766][SQL] Constant alias columns in INNER JOIN should not be folded by FoldablePropagation rule
- [SPARK-19677][SS] Committing a delta file atop an existing one should not fail on HDFS
- [SPARK-19038][YARN] Avoid overwriting keytab configuration in yarn-client
- [SPARK-19617][SS] Fix the race condition when starting and stopping a query quickly (branch-2.1)
- [SPARK-19599][SS] Clean up HDFSMetadataLog
- [SPARK-19529] TransportClientFactory.createClient() shouldn't call awaitUninterruptibly()
- [SPARK-18717][SQL] Make code generation for Scala Map work with immutable.Map also
- [SPARK-19512][BACKPORT-2.1][SQL] codegen for compare structs fails #16852
- [SPARK-19481] [REPL] [MAVEN] Avoid to leak SparkContext in Signaling.cancelOnInterrupt
- [SPARK-18750][YARN] Follow up: move test to correct directory in 2.1 branch.
- [SPARK-18750][YARN] Avoid using "mapValues" when allocating containers.
- [SPARK-19268][SS] Disallow adaptive query execution for streaming queries
- [SPARK-18850][SS] Make StreamExecution and progress classes serializable
- [SPARK-18589][SQL] Fix Python UDF accessing attributes from both side of join
- [SPARK-19314][SS][CATALYST] Do not allow sort before aggregation in Structured Streaming plan
- [SPARK-19129][SQL] SessionCatalog: Disallow empty part col values in partition spec
- [SPARK-19048][SQL] Delete Partition Location when Dropping Managed Partitioned Tables in InMemoryCatalog
- [SPARK-19019] [PYTHON] Fix hijacked `collections.namedtuple` and port cloudpickle changes for PySpark to work with Python 3.6.0
- [SPARK-19092][SQL][BACKPORT-2.1] Save() API of DataFrameWriter should not scan all the saved files #16481
- [SPARK-19120] Refresh Metadata Cache After Loading Hive Tables
- [SPARK-19180] [SQL] the offset of short should be 2 in OffHeapColumn
- [SPARK-19178][SQL] convert string of large numbers to int should return null
- [SPARK-18687][PYSPARK][SQL] Backward compatibility - creating a Dataframe on a new SQLContext object fails with a Derby error
- [SPARK-19055][SQL][PYSPARK] Fix SparkSession initialization when SparkContext is stopped
- [SPARK-16845][SQL] `GeneratedClass$SpecificOrdering` grows beyond 64 KB
- [SPARK-18952][BACKPORT] Regex strings not properly escaped in codegen for aggregations
- [SPARK-17807][CORE] split test-tags into test-JAR
- [SPARK-18908][SS] Creating StreamingQueryException should check if logicalPlan is created
- [SPARK-18528][SQL] Fix a bug to initialise an iterator of aggregation buffer
- [SPARK-18234][SS] Made update mode public
- [SPARK-18588][SS][KAFKA] Create a new KafkaConsumer when error happens to fix the flaky test
- [SPARK-18894][SS] Fix event time watermark delay threshold specified in months or years
- [SPARK-18281] [SQL] [PYSPARK] Remove timeout for reading data through socket for local iterator
- [SPARK-18761][CORE] Introduce "task reaper" to oversee task killing in executors
- [SPARK-18928] Check TaskContext.isInterrupted() in FileScanRDD, JDBCRDD & UnsafeSorter
- [SPARK-18700][SQL] Add StripedLock for each table's relation in cache
- [SPARK-18703][SPARK-18675][SQL][BACKPORT-2.1] CTAS for hive serde table should work for all hive versions AND Drop Staging Directories and Data Files
- [SPARK-18827][CORE] Fix cannot read broadcast on disk
- [SPARK-19520][STREAMING] Do not encrypt data written to the WAL.
- [SPARK-19857][YARN] Correctly calculate next credential update time.
- [SPARK-19626][YARN] Using the correct config to set credentials update time
- Preview of: [SPARK-4105] retry the fetch or stage if shuffle block is corrupt
- [SPARK-19307][PYSPARK] Make sure user conf is propagated to SparkContext.

## Issues Fixed in CDS 2.0 Release 2

- [SPARK-4563][CORE] Allow driver to advertise a different network address.
- [SPARK-18993] Unable to build/compile Spark in IntelliJ due to missing Scala deps in spark-tags

- [SPARK-19314] Do not allow sort before aggregation in Structured Streaming plan
- [SPARK-18762] Web UI should be http:4040 instead of https:4040
- [SPARK-18745] java.lang.IndexOutOfBoundsException running query 68 Spark SQL on (100TB)
- [SPARK-18703] Insertion/CTAS against Hive Tables: Staging Directories and Data Files Not Dropped Until Normal Termination of JVM
- [SPARK-18091] Deep if expressions cause Generated SpecificUnsafeProjection code to exceed JVM code size limit

## Issues Fixed in CDS 2.0 Release 1

- [SPARK-4563][CORE] Allow driver to advertise a different network address.
- [SPARK-18685][TESTS] Fix URI and release resources after opening in tests at ExecutorClassLoaderSuite
- [SPARK-18677] Fix parsing ['key'] in JSON path expressions.
- [SPARK-18617][SPARK-18560][TESTS] Fix flaky test: StreamingContextSuite. Receiver data should be deserialized properly
- [SPARK-18617][SPARK-18560][TEST] Fix flaky test: StreamingContextSuite. Receiver data should be deserialized properly
- [SPARK-18274][ML][PYSPARK] Memory leak in PySpark JavaWrapper
- [SPARK-18674][SQL] improve the error message of using join
- [SPARK-18617][CORE][STREAMING] Close "kryo auto pick" feature for Spark Streaming
- [SPARK-17843][WEB UI] Indicate event logs pending for processing on h…
- [SPARK-17783][SQL][BACKPORT-2.0] Hide Credentials in CREATE and DESC FORMATTED/EXTENDED a PERSISTENT/TEMP Table for JDBC
- [SPARK-18640] Add synchronization to TaskScheduler.runningTasksByExecutors
- [SPARK-18553][CORE] Fix leak of TaskSetManager following executor loss
- [SPARK-18597][SQL] Do not push-down join conditions to the left side of a Left Anti join [BRANCH-2.0]
- [SPARK-18118][SQL] fix a compilation error due to nested JavaBeans
- [SPARK-17251][SQL] Improve `OuterReference` to be `NamedExpression`
- [SPARK-18436][SQL] isin causing SQL syntax error with JDBC
- [SPARK-18519][SQL][BRANCH-2.0] map type can not be used in EqualTo
- [SPARK-18053][SQL] compare unsafe and safe complex-type values correctly
- [SPARK-18504][SQL] Scalar subquery with extra group by columns returning incorrect result
- [SPARK-18477][SS] Enable interrupts for HDFS in HDFSMetadataLog
- [SPARK-18546][CORE] Fix merging shuffle spills when using encryption.
- [SPARK-18547][CORE] Propagate I/O encryption key when executors register.
- [SPARK-16625][SQL] General data types to be mapped to Oracle
- [SPARK-18462] Fix ClassCastException in SparkListenerDriverAccumUpdates event
- [SPARK-18459][SPARK-18460][STRUCTUREDSTREAMING] Rename triggerId to batchId and add triggerDetails to json in StreamingQueryStatus (for branch-2.0)
- [SPARK-18430][SQL][BACKPORT-2.0] Fixed Exception Messages when Hitting an Invocation Exception of Function Lookup
- [SPARK-18400][STREAMING] NPE when resharding Kinesis Stream
- [SPARK-18300][SQL] Do not apply foldable propagation with expand as a child [BRANCH-2.0]
- [SPARK-18337] Complete mode memory sinks should be able to recover from checkpoints
- [SPARK-16808][CORE] History Server main page does not honor APPLICATION_WEB_PROXY_BASE
- [SPARK-17348][SQL] Incorrect results from subquery transformation
- [SPARK-18416][STRUCTURED STREAMING] Fixed temp file leak in state store
- [SPARK-18432][DOC] Changed HDFS default block size from 64MB to 128MB
- [SPARK-18010][CORE] Reduce work performed for building up the application list for the History Server app list UI page
- [SPARK-18382][WEBUI] "run at null:-1" in UI when no file/line info in call site info
- [SPARK-18426][STRUCTURED STREAMING] Python Documentation Fix for Structured Streaming Programming Guide

- [SPARK-17982][SQL][BACKPORT-2.0] SQLBuilder should wrap the generated SQL with parenthesis for LIMIT
- [SPARK-18387][SQL] Add serialization to checkEvaluation.
- [SPARK-18368][SQL] Fix regexp replace when serialized
- [SPARK-18342] Make rename failures fatal in HDFSBackedStateStore
- [SPARK-18280][CORE] Fix potential deadlock in `StandaloneSchedulerBackend.dead`
- [SPARK-17703][SQL][BACKPORT-2.0] Add unnamed version of addReferenceObj for minor objects.
- [SPARK-18137][SQL] Fix RewriteDistinctAggregates UnresolvedException when a UDAF has a foldable TypeCheck
- [SPARK-18283][STRUCTURED STREAMING][KAFKA] Added test to check whether default starting offset in latest
- [SPARK-18125][SQL][BRANCH-2.0] Fix a compilation error in codegen due to splitExpression
- [SPARK-17849][SQL] Fix NPE problem when using grouping sets
- [SPARK-17693][SQL][BACKPORT-2.0] Fixed Insert Failure To Data Source Tables when the Schema has the Comment Field
- [SPARK-17981][SPARK-17957][SQL][BACKPORT-2.0] Fix Incorrect Nullability Setting to False in FilterExec
- [SPARK-18189][SQL][FOLLOWUP] Move test from ReplSuite to prevent java.lang.ClassCircularityError
- [SPARK-17337][SPARK-16804][SQL][BRANCH-2.0] Backport subquery related PRs
- [SPARK-18200][GRAPHX][FOLLOW-UP] Support zero as an initial capacity in OpenHashSet
- [SPARK-18200][GRAPHX] Support zero as an initial capacity in OpenHashSet
- [SPARK-18111][SQL] Wrong approximate quantile answer when multiple records have the minimum value(for branch 2.0)
- [SPARK-18160][CORE][YARN] spark.files & spark.jars should not be passed to driver in yarn mode
- [SPARK-16796][WEB UI] Mask spark.authenticate.secret on Spark environ...
- [SPARK-18133][BRANCH-2.0][EXAMPLES][ML] Python ML Pipeline Exampl...
- [SPARK-18144][SQL] logging StreamingQueryListener$QueryStartedEvent
- [SPARK-18114][HOTFIX] Fix line-too-long style error from backport of SPARK-18114
- [SPARK-18148][SQL] Misleading Error Message for Aggregation Without Window/GroupBy
- [SPARK-18189][SQL] Fix serialization issue in KeyValueGroupedDataset
- [SPARK-18114][MESOS] Fix mesos cluster scheduler generage command option error
- [SPARK-18030][TESTS] Fix flaky FileStreamSourceSuite by not deleting the files
- [SPARK-18143][SQL] Ignore Structured Streaming event logs to avoid breaking history server (branch 2.0)
- [SPARK-16312][FOLLOW-UP][STREAMING][KAFKA][DOC] Add java code snippet for Kafka 0.10 integration doc
- [SPARK-18164][SQL] ForeachSink should fail the Spark job if `process` throws exception
- [SPARK-16963][SQL] Fix test "StreamExecution metadata garbage collection"
- [SPARK-17813][SQL][KAFKA] Maximum data per trigger
- [SPARK-18132] Fix checkstyle
- [SPARK-18009][SQL] Fix ClassCastException while calling toLocalIterator() on dataframe produced by RunnableCommand
- [SPARK-16963][STREAMING][SQL] Changes to Source trait and related implementation classes
- [SPARK-13747][SQL] Fix concurrent executions in ForkJoinPool for SQL (branch 2.0)
- [SPARK-18063][SQL] Failed to infer constraints over multiple aliases
- [SPARK-16304] LinkageError should not crash Spark executor
- [SPARK-17733][SQL] InferFiltersFromConstraints rule never terminates for query
- [SPARK-18022][SQL] java.lang.NullPointerException instead of real exception when saving DF to MySQL
- [SPARK-16988][SPARK SHELL] spark history server log needs to be fixed to show https url when ssl is enabled
- [SPARK-18070][SQL] binary operator should not consider nullability when comparing input types
- [SPARK-17624][SQL][STREAMING][TEST] Fixed flaky StateStoreSuite.maintenance
- [SPARK-18044][STREAMING] FileStreamSource should not infer partitions in every batch
- [SPARK-17153][SQL] Should read partition data when reading new files in filestream without globbing
- [SPARK-18093][SQL] Fix default value test in SQLConfSuite to work rega...
- [SPARK-17810][SQL] Default spark.sql.warehouse.dir is relative to local FS but can resolve as HDFS path
- [SPARK-18058][SQL] [BRANCH-2.0]Comparing column types ignoring Nullability in Union and SetOperation

- [SPARK-17123][SQL][BRANCH-2.0] Use type-widened encoder for DataFrame for set operations
- [SPARK-17698][SQL] Join predicates should not contain filter clauses
- [SPARK-17986][ML] SQLTransformer should remove temporary tables
- [SPARK-16606][MINOR] Tiny follow-up to , to correct more instances of the same log message typo
- [SPARK-17853][STREAMING][KAFKA][DOC] make it clear that reusing group.id is bad
- [SPARK-16312][STREAMING][KAFKA][DOC] Doc for Kafka 0.10 integration
- [SPARK-17812][SQL][KAFKA] Assign and specific startingOffsets for structured stream
- [SPARK-17929][CORE] Fix deadlock when CoarseGrainedSchedulerBackend reset
- [SPARK-17926][SQL][STREAMING] Added json for statuses
- [SPARK-17811] SparkR cannot parallelize data.frame with NA or NULL in Date columns
- [SPARK-18034] Upgrade to MiMa 0.1.11 to fix flakiness
- [SPARK-17999][KAFKA][SQL] Add getPreferredLocations for KafkaSourceRDD
- [SPARK-18003][SPARK CORE] Fix bug of RDD zipWithIndex & zipWithUniqueId index value overflowing
- [SPARK-17989][SQL] Check ascendingOrder type in sort_array function rather than throwing ClassCastException
- [SPARK-17675][CORE] Expand Blacklist for TaskSets
- [SPARK-17623][CORE] Clarify type of TaskEndReason with a failed task.
- [SPARK-17304] Fix perf. issue caused by TaskSetManager.abortIfCompletelyBlacklisted
- [SPARK-15865][CORE] Blacklist should not result in job hanging with less than 4 executors
- [SPARK-15783][CORE] Fix Flakiness in BlacklistIntegrationSuite
- [SPARK-15783][CORE] still some flakiness in these blacklist tests so ignore for now
- [SPARK-15714][CORE] Fix flaky o.a.s.scheduler.BlacklistIntegrationSuite
- [SPARK-10372] [CORE] basic test framework for entire spark scheduler
- [SPARK-16106][CORE] TaskSchedulerImpl should properly track executors added to existing hosts
- [SPARK-18001][DOCUMENT] fix broke link to SparkDataFrame
- [SPARK-17711][TEST-HADOOP2.2] Fix hadoop2.2 compilation error
- [SPARK-17731][SQL][STREAMING][FOLLOWUP] Refactored StreamingQueryListener APIs for branch-2.0
- [SPARK-17841][STREAMING][KAFKA] drain commitQueue
- [SPARK-17711] Compress rolled executor log
- [SPARK-17751][SQL][BACKPORT-2.0] Remove spark.sql.eagerAnalysis and Output the Plan if Existed in AnalysisException
- [SPARK-17731][SQL][STREAMING] Metrics for structured streaming for branch-2.0
- [SPARK-17892][SQL][2.0] Do Not Optimize Query in CTAS More Than Once #15048
- [SPARK-17819][SQL][BRANCH-2.0] Support default database in connection URIs for Spark Thrift Server
- [SPARK-17953][DOCUMENTATION] Fix typo in SparkSession scaladoc
- [SPARK-17863][SQL] should not add column into Distinct
- [SPARK-17387][PYSPARK] Creating SparkContext() from python without spark-submit ignores user conf
- [SPARK-17834][SQL] Fetch the earliest offsets manually in KafkaSource instead of counting on KafkaConsumer
- [SPARK-17876] Write StructuredStreaming WAL to a stream instead of materializing all at once
- [SPARK-16827][BRANCH-2.0] Avoid reporting spill metrics as shuffle metrics
- [SPARK-17782][STREAMING][KAFKA] alternative eliminate race condition of poll twice
- [SPARK-17790][SPARKR] Support for parallelizing R data.frame larger than 2GB
- [SPARK-17884][SQL] To resolve Null pointer exception when casting from empty string to interval type.
- [SPARK-17808][PYSPARK] Upgraded version of Pyrolite to 4.13
- [SPARK-17880][DOC] The url linking to `AccumulatorV2` in the document is incorrect.
- [SPARK-17816][CORE][BRANCH-2.0] Fix ConcurrentModificationException issue in BlockStatusesAccumulator
- [SPARK-17346][SQL][TESTS] Fix the flaky topic deletion in KafkaSourceStressSuite
- [SPARK-17738][TEST] Fix flaky test in ColumnTypeSuite
- [SPARK-17417][CORE] Fix # of partitions for Reliable RDD checkpointing
- [SPARK-17832][SQL] TableIdentifier.quotedString creates un-parseable names when name contains a backtick
- [SPARK-17806] [SQL] fix bug in join key rewritten in HashJoin

- [SPARK-17782][STREAMING][BUILD] Add Kafka 0.10 project to build modules
- [SPARK-17346][SQL][TEST-MAVEN] Add Kafka source for Structured Streaming (branch 2.0)
- [SPARK-17707][WEBUI] Web UI prevents spark-submit application to be finished
- [SPARK-17805][PYSPARK] Fix in sqlContext.read.text when pass in list of paths
- [SPARK-17612][SQL][BRANCH-2.0] Support `DESCRIBE table PARTITION` SQL syntax
- [SPARK-17792][ML] L-BFGS solver for linear regression does not accept general numeric label column types
- [SPARK-17750][SQL][BACKPORT-2.0] Fix CREATE VIEW with INTERVAL arithmetic
- [SPARK-17803][TESTS] Upgrade docker-client dependency
- [SPARK-17780][SQL] Report Throwable to user in StreamExecution
- [SPARK-17798][SQL] Remove redundant Experimental annotations in sql.streaming
- [SPARK-17643] Remove comparable requirement from Offset (backport for branch-2.0)
- [SPARK-17758][SQL] Last returns wrong result in case of empty partition
- [SPARK-17778][TESTS] Mock SparkContext to reduce memory usage of BlockManagerSuite
- [SPARK-17773][BRANCH-2.0] Input/Output] Add VoidObjectInspector
- [SPARK-17549][SQL] Only collect table size stat in driver for cached relation.
- [SPARK-17559][MLLIB] persist edges if their storage level is non in PeriodicGraphCheckpointer
- [SPARK-17112][SQL] "select null" via JDBC triggers IllegalArgumentException in Thriftserver
- [SPARK-17753][SQL] Allow a complex expression as the input a value based case statement
- [SPARK-17587][PYTHON][MLLIB] SparseVector __getitem__ should follow __getitem__ contract
- [SPARK-17736][DOCUMENTATION][SPARKR] Update R README for rmarkdown,...
- [SPARK-17721][MLLIB][ML] Fix for multiplying transposed SparseMatrix with SparseVector
- [SPARK-17672] Spark 2.0 history server web Ui takes too long for a single application
- [SPARK-17712][SQL] Fix invalid pushdown of data-independent filters beneath aggregates
- [SPARK-16343][SQL] Improve the PushDownPredicate rule to pushdown predicates correctly in non-deterministic condition.
- [SPARK-17641][SQL] Collect_list/Collect_set should not collect null values.
- [SPARK-17673][SQL] Incorrect exchange reuse with RowDataSourceScan (backport)
- [SPARK-17644][CORE] Do not add failedStages when abortStage for fetch failure
- [SPARK-17666] Ensure that RecordReaders are closed by data source file scans (backport)
- [SPARK-17056][CORE] Fix a wrong assert regarding unroll memory in MemoryStore
- [SPARK-17618] Guard against invalid comparisons between UnsafeRow and other formats
- [SPARK-17652] Fix confusing exception message while reserving capacity
- [SPARK-17649][CORE] Log how many Spark events got dropped in LiveListenerBus
- [SPARK-17650] malformed url's throw exceptions before bricking Executors
- [SPARK-10835][ML] Word2Vec should accept non-null string array, in addition to existing null string array
- [SPARK-15703][SCHEDULER][CORE][WEBUI] Make ListenerBus event queue size configurable (branch 2.0)
- [SPARK-4563][CORE] Allow driver to advertise a different network address.
- [SPARK-17577][CORE][2.0 BACKPORT] Update SparkContext.addFile to make it work well on Windows
- [SPARK-17210][SPARKR] sparkr.zip is not distributed to executors when running sparkr in RStudio
- [SPARK-17640][SQL] Avoid using -1 as the default batchId for FileStreamSource.FileEntry
- [SPARK-16240][ML] ML persistence backward compatibility for LDA - 2.0 backport
- [SPARK-17502][17609][SQL][BACKPORT][2.0] Fix Multiple Bugs in DDL Statements on Temporary Views
- [SPARK-17599][SPARK-17569] Backport and to Spark 2.0 branch
- [SPARK-17616][SQL] Support a single distinct aggregate combined with a non-partial aggregate
- [SPARK-17638][STREAMING] Stop JVM StreamingContext when the Python process is dead
- [SPARK-17613] S3A base paths with no '/' at the end return empty DataFrames
- [SPARK-17421][DOCS] Documenting the current treatment of MAVEN_OPTS.
- [SPARK-17494][SQL] changePrecision() on compact decimal should respect rounding mode
- [SPARK-17627] Mark Streaming Providers Experimental
- [SPARK-17512][CORE] Avoid formatting to python path for yarn and mesos cluster mode

- [SPARK-17418] Prevent kinesis-asl-assembly artifacts from being published
- [SPARK-17617][SQL] Remainder(%) expression.eval returns incorrect result on double value
- [SPARK-15698][SQL][STREAMING] Add the ability to remove the old MetadataLog in FileStreamSource (branch-2.0)
- [SPARK-17051][SQL] we should use hadoopConf in InsertIntoHiveTable
- [SPARK-17513][SQL] Make StreamExecution garbage-collect its metadata
- [SPARK-17160] Properly escape field names in code-generated error messages
- [SPARK-17100] [SQL] fix Python udf in filter on top of outer join
- [SPARK-16439] [SQL] bring back the separator in SQL UI
- [SPARK-17611][yarn][test] Make shuffle service test really test auth.
- [SPARK-17433] YarnShuffleService doesn't handle moving credentials levelDb
- [SPARK-17438][WEBUI] Show Application.executorLimit in the application page
- [SPARK-17473][SQL] fixing docker integration tests error due to different versions of jars.
- [SPARK-17589][TEST][2.0] Fix test case `create external table` in MetastoreDataSourcesSuite
- [SPARK-17297][DOCS] Clarify window/slide duration as absolute time, not relative to a calendar
- [SPARK-17571][SQL] AssertOnQuery.condition should always return Boolean value
- [SPARK-16462][SPARK-16460][SPARK-15144][SQL] Make CSV cast null values properly
- [SPARK-17586][BUILD] Do not call static member via instance reference
- [SPARK-17546][DEPLOY] start-* scripts should use hostname -f
- [SPARK-17541][SQL] fix some DDL bugs about table management when same-name temp view exists
- [SPARK-17480][SQL][FOLLOWUP] Fix more instances which calls List.length/size which is O(n)
- [SPARK-17491] Close serialization stream to fix wrong answer bug in putIteratorAsBytes()
- [SPARK-17575][DOCS] Remove extra table tags in configuration document
- [SPARK-17548][MLLIB] Word2VecModel.findSynonyms no longer spuriously rejects the best match when invoked with a vector
- [SPARK-17561][DOCS] DataFrameWriter documentation formatting problems
- [SPARK-17567][DOCS] Use valid url to Spark RDD paper
- [SPARK-17558] Bump Hadoop 2.7 version from 2.7.2 to 2.7.3
- [SPARK-17484] Prevent invalid block locations from being reported after put() exceptions
- [SPARK-17364][SQL] Antlr lexer wrongly treats full qualified identifier as a decimal number token when parsing SQL string
- [SPARK-17483] Refactoring in BlockManager status reporting and block removal
- [SPARK-17114][SQL] Fix aggregates grouped by literals with empty input
- [SPARK-17547] Ensure temp shuffle data file is cleaned up after error
- [SPARK-17521] Error when I use sparkContext.makeRDD(Seq())
- [SPARK-17465][SPARK CORE] Inappropriate memory management in `org.apache.spark.storage.MemoryStore` may lead to memory leak
- [SPARK-17463][CORE] Make CollectionAccumulator and SetAccumulator's value can be read thread-safely
- [SPARK-17511] Yarn Dynamic Allocation: Avoid marking released container as Failed
- [SPARK-17514] df.take(1) and df.limit(1).collect() should perform the same in Python
- [SPARK-17445][DOCS] Reference an ASF page as the main place to find third-party packages
- [SPARK-16711] YarnShuffleService doesn't re-init properly on YARN rolling upgrade
- [SPARK-15074][SHUFFLE] Cache shuffle index file to speedup shuffle fetch
- [SPARK-17480][SQL] Improve performance by removing or caching List.length which is O(n)
- [SPARK-17525][PYTHON] Remove SparkContext.clearFiles() from the PySpark API as it was removed from the Scala API prior to Spark 2.0.0
- [SPARK-17531] Don't initialize Hive Listeners for the Execution Client
- [SPARK-17515] CollectLimit.execute() should perform per-partition limits
- [SPARK-17474] [SQL] fix python udf in TakeOrderedAndProjectExec
- [SPARK-17485] Prevent failed remote reads of cached blocks from failing entire job
- [SPARK-14818] Post-2.0 MiMa exclusion and build changes

- [SPARK-17503][CORE] Fix memory leak in Memory store when unable to cache the whole RDD in memory
- [SPARK-17486] Remove unused TaskMetricsUIData.updatedBlockStatuses field
- [SPARK-17336][PYSPARK] Fix appending multiple times to PYTHONPATH from spark-config.sh
- [SPARK-17439][SQL] Fixing compression issues with approximate quantiles and adding more tests
- [SPARK-17396][CORE] Share the task support between UnionRDD instances.
- [SPARK-17354] [SQL] Partitioning by dates/timestamps should work with Parquet vectorized reader
- [SPARK-17456][CORE] Utility for parsing Spark versions
- [SPARK-17339][CORE][BRANCH-2.0] Do not use path to get a filesystem in hadoopFile and newHadoopFile APIs
- [SPARK-16533][CORE] - backport driver deadlock fix to 2.0
- [SPARK-17370] Shuffle service files not invalidated when a slave is lost
- [SPARK-17296][SQL] Simplify parser join processing [BACKPORT 2.0]
- [SPARK-17372][SQL][STREAMING] Avoid serialization issues by using Arrays to save file names in FileStreamSource
- [SPARK-17279][SQL] better error message for exceptions during ScalaUDF execution
- [SPARK-17316][CORE] Fix the 'ask' type parameter in 'removeExecutor'
- [SPARK-17110] Fix StreamCorruptionException in BlockManager.getRemoteValues()
- [SPARK-17299] TRIM/LTRIM/RTRIM should not strips characters other than spaces
- [SPARK-16334] [BACKPORT] Reusing same dictionary column for decoding consecutive row groups shouldn't throw an error
- [SPARK-16922] [SPARK-17211] [SQL] make the address of values portable in LongToUnsafeRowMap
- [SPARK-17356][SQL] Fix out of memory issue when generating JSON for TreeNode
- [SPARK-17369][SQL][2.0] MetastoreRelation toJSON throws AssertException due to missing otherCopyArgs
- [SPARK-17358][SQL] Cached table(parquet/orc) should be shard between beelines
- [SPARK-17353][SPARK-16943][SPARK-16942][BACKPORT-2.0][SQL] Fix multiple bugs in CREATE TABLE LIKE command
- [SPARK-17391][TEST][2.0] Fix Two Test Failures After Backport
- [SPARK-17335][SQL] Fix ArrayType and MapType CatalogString.
- [SPARK-16663][SQL] desc table should be consistent between data source and hive serde tables
- [SPARK-16959][SQL] Rebuild Table Comment when Retrieving Metadata from Hive Metastore
- [SPARK-17347][SQL][EXAMPLES] Encoder in Dataset example has incorrect type
- [SPARK-17230] [SQL] Should not pass optimized query into QueryExecution in DataFrameWriter
- [SPARK-17261] [PYSPARK] Using HiveContext after re-creating SparkContext in Spark 2.0 throws "Java.lang.illegalStateException: Cannot call methods on a stopped sparkContext"
- [SPARK-16935][SQL] Verification of Function-related ExternalCatalog APIs
- [SPARK-17352][WEBUI] Executor computing time can be negative-number because of calculation error
- [SPARK-17342][WEBUI] Style of event timeline is broken
- [SPARK-17355] Workaround for HIVE-14684 / HiveResultSetMetaData.isSigned exception
- [SPARK-16926] [SQL] Remove partition columns from partition metadata.
- [SPARK-17271][SQL] Planner adds un-necessary Sort even if child orde...
- [SPARK-17318][TESTS] Fix ReplSuite replicating blocks of object with class defined in repl again
- [SPARK-17180][SPARK-17309][SPARK-17323][SQL][2.0] create AlterViewAsCommand to handle ALTER VIEW AS
- [SPARK-17316][TESTS] Fix MesosCoarseGrainedSchedulerBackendSuite
- [SPARK-17316][CORE] Make CoarseGrainedSchedulerBackend.removeExecutor non-blocking
- [SPARK-17243][WEB UI] Spark 2.0 History Server won't load with very large application history
- [SPARK-17318][TESTS] Fix ReplSuite replicating blocks of object with class defined in repl
- [SPARK-17264][SQL] DataStreamWriter should document that it only supports Parquet for now
- [SPARK-17301][SQL] Remove unused classTag field from AtomicType base class
- [SPARK-17063] [SQL] Improve performance of MSCK REPAIR TABLE with Hive metastore
- [SPARK-16216][SQL][FOLLOWUP][BRANCH-2.0] Bacoport enabling timestamp type tests for JSON and verify all unsupported types in CSV
- [SPARK-17216][UI] fix event timeline bars length
- [ML][MLLIB] The require condition and message doesn't match in SparseMatrix.

- [SPARK-15382][SQL] Fix a bug in sampling with replacement
- [SPARK-17274][SQL] Move join optimizer rules into a separate file
- [SPARK-17270][SQL] Move object optimization rules into its own file (branch-2.0)
- [SPARK-17269][SQL] Move finish analysis optimization stage into its own file
- [SPARK-17244] Catalyst should not pushdown non-deterministic join conditions
- [SPARK-17235][SQL] Support purging of old logs in MetadataLog
- [SPARK-17246][SQL] Add BigDecimal literal
- [SPARK-17165][SQL] FileStreamSource should not track the list of seen files indefinitely
- [SPARK-17242][DOCUMENT] Update links of external dstream projects
- [SPARK-17231][CORE] Avoid building debug or trace log messages unless the respective log level is enabled
- [SPARK-17205] Literal.sql should handle Infinity and NaN
- [SPARK-15083][WEB UI] History Server can OOM due to unlimited TaskUIData
- [SPARK-16700][PYSPARK][SQL] create DataFrame from dict/Row with schema
- [SPARK-17167][2.0][SQL] Issue Exceptions when Analyze Table on In-Memory Cataloged Tables
- [SPARK-16991][SPARK-17099][SPARK-17120][SQL] Fix Outer Join Elimination when Filter's isNotNull Constraints Unable to Filter Out All Null-supplying Rows
- [SPARK-17061][SPARK-17093][SQL][BACKPORT] MapObjects should make copies of unsafe-backed data
- [SPARK-17193][CORE] HadoopRDD NPE at DEBUG log level when getLocationInfo == null
- [SPARK-17228][SQL] Not infer/propagate non-deterministic constraints
- [SPARK-16216][SQL][BRANCH-2.0] Backport Read/write dateFormat/timestampFormat options for CSV and JSON
- [SPARK-16781][PYSPARK] java launched by PySpark as gateway may not be the same java used in the spark environment
- [SPARK-17086][ML] Fix InvalidArgumentException issue in QuantileDiscretizer when some quantiles are duplicated
- [SPARK-17186][SQL] remove catalog table type INDEX
- [SPARK-17194] Use single quotes when generating SQL for string literals
- [SPARK-13286] [SQL] add the next expression of SQLException as cause
- [SPARK-17182][SQL] Mark Collect as non-deterministic
- [SPARK-16550][SPARK-17042][CORE] Certain classes fail to deserialize in block manager replication
- [SPARK-17162] Range does not support SQL generation
- [SPARK-16320][DOC] Document G1 heap region's effect on spark 2.0 vs 1.6
- [SPARK-17085][STREAMING][DOCUMENTATION AND ACTUAL CODE DIFFERS - UNSUPPORTED OPERATIONS]
- [SPARK-17115][SQL] decrease the threshold when split expressions
- [SPARK-17098][SQL] Fix `NullPropagation` optimizer to handle `COUNT(NULL) OVER` correctly
- [SPARK-12666][CORE] SparkSubmit packages fix for when 'default' conf doesn't exist in dependent module
- [SPARK-17124][SQL] RelationalGroupedDataset.agg should preserve order and allow multiple aggregates per column
- [SPARK-17104][SQL] LogicalRelation.newInstance should follow the semantics of MultiInstanceRelation
- [SPARK-17150][SQL] Support SQL generation for inline tables
- [SPARK-17158][SQL] Change error message for out of range numeric literals
- [SPARK-17149][SQL] array.sql for testing array related functions
- [SPARK-17113] [SHUFFLE] Job failure due to Executor OOM in offheap mode
- [SPARK-16686][SQL] Remove PushProjectThroughSample since it is handled by ColumnPruning
- [SPARK-11227][CORE] UnknownHostException can be thrown when NameNode HA is enabled.
- [SPARK-16994][SQL] Whitelist operators for predicate pushdown
- [SPARK-16961][CORE] Fixed off-by-one error that biased randomizeInPlace
- [SPARK-16947][SQL] Support type coercion and foldable expression for inline tables
- [SPARK-17069] Expose spark.range() as table-valued function in SQL
- [SPARK-17117][SQL] 1 / NULL should not fail analysis
- [SPARK-16391][SQL] Support partial aggregation for reduceGroups

- [SPARK-16995][SQL] TreeNodeException when flat mapping RelationalGroupedDataset created from DataFrame containing a column created with lit/expr
- [SPARK-17096][SQL][STREAMING] Improve exception string reported through the StreamingQueryListener
- [SPARK-17102][SQL] bypass UserDefinedGenerator for json format check
- [SPARK-15285][SQL] Generated SpecificSafeProjection.apply method grows beyond 64 KB
- [SPARK-17084][SQL] Rename ParserUtils.assert to validate
- [SPARK-17089][DOCS] Remove api doc link for mapReduceTriplets operator
- [SPARK-16964][SQL] Remove private[sql] and private[spark] from sql.execution package [Backport]
- [SPARK-17065][SQL] Improve the error message when encountering an incompatible DataSourceRegister
- [SPARK-16508][SPARKR] Split docs for arrange and orderBy methods
- [SPARK-17027][ML] Avoid integer overflow in PolynomialExpansion.getPolySize
- [SPARK-16966][SQL][CORE] App Name is a randomUUID even when "spark.app.name" exists
- [SPARK-17013][SQL] Parse negative numeric literals
- [SPARK-16975][SQL] Column-partition path starting '_' should be handled correctly
- [SPARK-17022][YARN] Handle potential deadlock in driver handling messages
- [SPARK-17018][SQL] literals.sql for testing literal parsing
- [SPARK-17015][SQL] group-by/order-by ordinal and arithmetic tests
- [SPARK-15899][SQL] Fix the construction of the file path with hadoop Path for Spark 2.0
- [SPARK-17011][SQL] Support testing exceptions in SQLQueryTestSuite
- [SPARK-17007][SQL] Move test data files into a test-data folder
- [SPARK-17008][SPARK-17009][SQL] Normalization and isolation in SQLQueryTestSuite.
- [SPARK-16866][SQL] Infrastructure for file-based SQL end-to-end tests
- [SPARK-17010][MINOR][DOC] Wrong description in memory management document
- [SPARK-15639] [SPARK-16321] [SQL] Push down filter at RowGroups level for parquet reader
- [SPARK-16324][SQL] regexp_extract should doc that it returns empty string when match fails
- [SPARK-16522][MESOS] Spark application throws exception on exit.
- [SPARK-16905] SQL DDL: MSCK REPAIR TABLE
- [SPARK-16956] Make ApplicationState.MAX_NUM_RETRY configurable
- [SPARK-16950] [PYSPARK] fromOffsets parameter support in KafkaUtils.createDirectStream for python3
- [SPARK-16610][SQL] Add `orc.compress` as an alias for `compression` option.
- [SPARK-16563][SQL] fix spark sql thrift server FetchResults bug
- [SPARK-16953] Make requestTotalExecutors public Developer API to be consistent with requestExecutors/killExecutors
- [SPARK-16586][CORE] Handle JVM errors printed to stdout.
- [SPARK-16936][SQL] Case Sensitivity Support for Refresh Temp Table
- [SPARK-16457][SQL] Fix Wrong Messages when CTAS with a Partition By Clause
- [SPARK-16939][SQL] Fix build error by using `Tuple1` explicitly in StringFunctionsSuite
- [SPARK-16409][SQL] regexp_extract with optional groups causes NPE
- [SPARK-16911] Fix the links in the programming guide
- [SPARK-16870][DOCS] Summary:add "spark.sql.broadcastTimeout" into docs/sql-programming-gu…
- [SPARK-16932][DOCS] Changed programming guide to not reference old accumulator API in Scala
- [SPARK-16925] Master should call schedule() after all executor exit events, not only failures
- [SPARK-16772][PYTHON][DOCS] Fix API doc references to UDFRegistration + Update "important classes"
- [SPARK-16750][FOLLOW-UP][ML] Add transformSchema for StringIndexer/VectorAssembler and fix failed tests.
- [SPARK-16907][SQL] Fix performance regression for parquet table when vectorized parquet record reader is not being used
- [SPARK-16863][ML] ProbabilisticClassifier.fit check threshoulds' length
- [SPARK-16877][BUILD] Add rules for preventing to use Java annotations (Deprecated and Override)
- [SPARK-16880][ML][MLLIB] make ann training data persisted if needed
- [SPARK-16875][SQL] Add args checking for DataSet randomSplit and sample

- [SPARK-16802] [SQL] fix overflow in LongToUnsafeRowMap
- [SPARK-16873][CORE] Fix SpillReader NPE when spillFile has no data
- [SPARK-14204][SQL] register driverClass rather than user-specified class
- [SPARK-16714][SPARK-16735][SPARK-16646] array, map, greatest, least's type coercion should handle decimal type
- [SPARK-16796][WEB UI] Visible passwords on Spark environment page
- [SPARK-16831][PYTHON] Fixed bug in CrossValidator.avgMetrics
- [SPARK-16787] SparkContext.addFile() should not throw if called twice with the same file
- [SPARK-16850][SQL] Improve type checking error message for greatest/least
- [SPARK-16836][SQL] Add support for CURRENT_DATE/CURRENT_TIMESTAMP literals
- [SPARK-16062] [SPARK-15989] [SQL] Fix two bugs of Python-only UDTs
- [SPARK-16837][SQL] TimeWindow incorrectly drops slideDuration in constructors
- [SPARK-15541] Casting ConcurrentHashMap to ConcurrentMap (master branch)
- [SPARK-16558][EXAMPLES][MLLIB] examples/mllib/LDAExample should use MLVector instead of MLlib Vector
- [SPARK-16734][EXAMPLES][SQL] Revise examples of all language bindings
- [SPARK-16818] Exchange reuse incorrectly reuses scans over different sets of partitions
- [SPARK-15869][STREAMING] Fix a potential NPE in StreamingJobProgressListener.getBatchUIData
- [SPARK-16774][SQL] Fix use of deprecated timestamp constructor & improve timezone handling
- [SPARK-16791][SQL] cast struct with timestamp field fails
- [SPARK-16778][SQL][TRIVIAL] Fix deprecation warning with SQLContext
- [SPARK-16805][SQL] Log timezone when query result does not match
- [SPARK-16813][SQL] Remove private[sql] and private[spark] from catalyst package
- [SPARK-16812] Open up SparkILoop.getAddedJars
- [SPARK-16800][EXAMPLES][ML] Fix Java examples that fail to run due to exception
- [SPARK-16748][SQL] SparkExceptions during planning should not wrapped in TreeNodeException
- [SPARK-16761][DOC][ML] Fix doc link in docs/ml-guide.md
- [SPARK-16750][ML] Fix GaussianMixture training failed due to feature column type mistake
- [SPARK-16664][SQL] Fix persist call on Data frames with more than 200...
- [SPARK-16772] Correct API doc references to PySpark classes + formatting fixes
- [SPARK-16764][SQL] Recommend disabling vectorized parquet reader on OutOfMemoryError
- [SPARK-16740][SQL] Fix Long overflow in LongToUnsafeRowMap
- [SPARK-16639][SQL] The query with having condition that contains grouping by column should work
- [SPARK-15232][SQL] Add subquery SQL building tests to LogicalPlanToSQLSuite
- [SPARK-16730][SQL] Implement function aliases for type casts
- [SPARK-16729][SQL] Throw analysis exception for invalid date casts
- [SPARK-16621][SQL] Generate stable SQLs in SQLBuilder
- [SPARK-16633][SPARK-16642][SPARK-16721][SQL] Fixes three issues related to lead and lag functions
- [SPARK-16724] Expose DefinedByConstructorParams
- [SPARK-16672][SQL] SQLBuilder should not raise exceptions on EXISTS queries
- [SPARK-16722][TESTS] Fix a StreamingContext leak in StreamingContextSuite when eventually fails
- [SPARK-14131][STREAMING] SQL Improved fix for avoiding potential deadlocks in HDFSMetadataLog
- [SPARK-16715][TESTS] Fix a potential ExprId conflict for SubexpressionEliminationSuite."Semantic equals and hash"
- [SPARK-16485][DOC][ML] Fixed several inline formatting in ml features doc
- [SPARK-16703][SQL] Remove extra whitespace in SQL generation for window functions
- [SPARK-16698][SQL] Field names having dots should be allowed for datasources based on FileFormat
- [SPARK-16648][SQL] Make ignoreNullsExpr a child expression of First and Last
- [SPARK-16699][SQL] Fix performance bug in hash aggregate on long string keys
- [SPARK-16515][SQL][FOLLOW-UP] Fix test `script` on OS X/Windows...
- [SPARK-16690][TEST] rename SQLTestUtils.withTempTable to withTempView

- [SPARK-16380][EXAMPLES] Update SQL examples and programming guide for Python language binding
- [SPARK-16651][PYSPARK][DOC] Make `withColumnRenamed/drop` description more consistent with Scala API
- [SPARK-16650] Improve documentation of spark.task.maxFailures
- [SPARK-16287][HOTFIX][BUILD][SQL] Fix annotation argument needs to be a constant
- [SPARK-16287][SQL] Implement str_to_map SQL function
- [SPARK-16334] Maintain single dictionary per row-batch in vectorized parquet reader
- [SPARK-16656][SQL] Try to make CreateTableAsSelectSuite more stable
- [SPARK-16644][SQL] Aggregate should not propagate constraints containing aggregate expressions
- [SPARK-16440][MLLIB] Destroy broadcasted variables even on driver
- [SPARK-5682][CORE] Add encrypted shuffle in spark
- [SPARK-16901] Hive settings in hive-site.xml may be overridden by Hive's default values
- [SPARK-16272][CORE] Allow config values to reference conf, env, system props.
- [SPARK-16632][SQL] Use Spark requested schema to guide vectorized Parquet reader initialization
- [SPARK-16634][SQL] Workaround JVM bug by moving some code out of ctor.
- [SPARK-16505][YARN] Optionally propagate error during shuffle service startup.
- [SPARK-14963][MINOR][YARN] Fix typo in YarnShuffleService recovery file name
- [SPARK-14963][YARN] Using recoveryPath if NM recovery is enabled
- [SPARK-16349][SQL] Fall back to isolated class loader when classes not found.
- [SPARK-16119][sql] Support PURGE option to drop table / partition.

## CDS Powered by Apache Spark Version, Packaging, and Download Information

The following sections provide links to the parcel and service descriptor files for the different CDS versions, as well as information about using CDS with Maven.

### CDS Versions Available for Download

**Note:** The parcel version displayed in Cloudera Manager, which is also part of the parcel file name, is structured as follows:

***<CDS_version>*-1.*<cdh_build_version>*.p*<patch_version>*.*<build_number>***

For example:

```
2.1.0.cloudera3-1.cdh5.13.3.p0.569822
```

The *<cdh_build_version>* portion is the version of CDH upon which the release was built. It is *not* the minimum supported CDH version. For example, although CDS 2.1 Release 3 was built on CDH 5.13.3, it is still supported on CDH 5.7 and higher CDH 5 versions.

To view the supported CDH versions and other requirements, see CDS Powered by Apache Spark Requirements on page 7.

**Table 1: Available CDS Versions**

| Version | Custom Service Descriptor | Parcel |
|---|---|---|
| 2.1 Release 3 | SPARK2_ON_YARN-2.1.0.cloudera3.jar | The exact parcel name is dependent on the OS. You can find all the parcels at http://archive.cloudera.com/spark2/parcels/2.1.0.cloudera3/. |
| 2.1 Release 2 | SPARK2_ON_YARN-2.1.0.cloudera2.jar | The exact parcel name is dependent on the OS. You can find all the parcels at http:// |

| Version | Custom Service Descriptor | Parcel |
|---|---|---|
| | | archive.cloudera.com/spark2/parcels/2.1.0.cloudera2/. |
| 2.1 Release 1 | SPARK2_ON_YARN-2.1.0.cloudera1.jar | The exact parcel name is dependent on the OS. You can find all the parcels at http://archive.cloudera.com/spark2/parcels/2.1.0.cloudera1/. |
| 2.0 Release 2 | SPARK2_ON_YARN-2.0.0.cloudera2.jar | The exact parcel name is dependent on the OS. You can find all the parcels at http://archive.cloudera.com/spark2/parcels/2.0.0.cloudera2/. |
| 2.0 Release 1 | SPARK2_ON_YARN-2.0.0.cloudera1.jar | The exact parcel name is dependent on the OS. You can find all the parcels at http://archive.cloudera.com/spark2/parcels/2.0.0.cloudera1/. |

### CDS Maven Artifacts

For information about using CDS Maven artifacts, see Using the CDS Powered by Apache Spark Maven Repository on page 29.

## Using the CDS Powered by Apache Spark Maven Repository

**Important:** CDS 2 no longer includes an assembly JAR. When you build an application JAR, *do not* include CDH or CDS JARs, because they are already provided. If you do, upgrading CDH or CDS can break your application. To avoid this situation, set the Maven dependency `scope` to `provided`. If you have already built applications which include the CDH or CDS JARs, update the dependency to set `scope` to `provided` and recompile.

For information on how to use CDH artifacts, see Using the CDH 5 Maven Repository.

If you want to build applications or tools for use with CDS Powered by Apache Spark, and you are using Maven or Ivy for dependency management, you can pull the CDS artifacts from the Cloudera Maven repository. The repository is available at https://repository.cloudera.com/artifactory/cloudera-repos/.

The following is a sample POM (`pom.xml`) file:

```
<project xmlns="http://maven.apache.org/POM/4.0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
http://maven.apache.org/maven-v4_0_0.xsd">
  <repositories>
    <repository>
      <id>cloudera</id>
      <url>https://repository.cloudera.com/artifactory/cloudera-repos/</url>
    </repository>
  </repositories>
</project>
```

For more information about the Maven artifacts for each CDS release, see the following topics:

### CDS 2.1 Powered by Apache Spark Maven Artifacts

The following tables lists the `groupId`, `artifactId`, and `version` required to access the artifacts for each CDS 2.1 Powered by Apache Spark release:

### CDS 2.1 Release 3 Maven Artifacts

The following pom fragment shows how to access a CDS 2.1 Release 3 artifact from a Maven POM.

```
<dependency>
      <groupId>org.apache.spark</groupId>
      <artifactId>spark-core_2.11</artifactId>
      <version>2.1.0.cloudera3</version>
      <scope>provided</scope>
</dependency>
```

The complete artifact list for this release follows.

| Project | groupId | artifactId | version |
|---|---|---|---|
| Apache Spark | org.apache.spark | spark-catalyst_2.11 | 2.1.0.cloudera3 |
| | org.apache.spark | spark-core_2.11 | 2.1.0.cloudera3 |
| | org.apache.spark | spark-graphx_2.11 | 2.1.0.cloudera3 |
| | org.apache.spark | spark-hive-exec_2.11 | 2.1.0.cloudera3 |
| | org.apache.spark | spark-hive_2.11 | 2.1.0.cloudera3 |
| | org.apache.spark | spark-launcher_2.11 | 2.1.0.cloudera3 |
| | org.apache.spark | spark-mllib-local_2.11 | 2.1.0.cloudera3 |
| | org.apache.spark | spark-mllib_2.11 | 2.1.0.cloudera3 |
| | org.apache.spark | spark-network-common_2.11 | 2.1.0.cloudera3 |
| | org.apache.spark | spark-network-shuffle_2.11 | 2.1.0.cloudera3 |
| | org.apache.spark | spark-network-yarn_2.11 | 2.1.0.cloudera3 |
| | org.apache.spark | spark-repl_2.11 | 2.1.0.cloudera3 |
| | org.apache.spark | spark-sketch_2.11 | 2.1.0.cloudera3 |
| | org.apache.spark | spark-sql-kafka-0-10_2.11 | 2.1.0.cloudera3 |
| | org.apache.spark | spark-sql_2.11 | 2.1.0.cloudera3 |
| | org.apache.spark | spark-streaming-flume-sink_2.11 | 2.1.0.cloudera3 |
| | org.apache.spark | spark-streaming-flume_2.11 | 2.1.0.cloudera3 |

| Project | groupId | artifactId | version |
|---|---|---|---|
| | org.apache.spark | spark-streaming-kafka-0-10-assembly_2.11 | 2.1.0.cloudera3 |
| | org.apache.spark | spark-streaming-kafka-0-10_2.11 | 2.1.0.cloudera3 |
| | org.apache.spark | spark-streaming-kafka-0-8-assembly_2.11 | 2.1.0.cloudera3 |
| | org.apache.spark | spark-streaming-kafka-0-8_2.11 | 2.1.0.cloudera3 |
| | org.apache.spark | spark-streaming_2.11 | 2.1.0.cloudera3 |
| | org.apache.spark | spark-tags_2.11 | 2.1.0.cloudera3 |
| | org.apache.spark | spark-unsafe_2.11 | 2.1.0.cloudera3 |
| | org.apache.spark | spark-yarn_2.11 | 2.1.0.cloudera3 |

### CDS 2.1 Release 2 Maven Artifacts

The following pom fragment shows how to access a CDS 2.1 Release 2 artifact from a Maven POM.

```
<dependency>
      <groupId>org.apache.spark</groupId>
      <artifactId>spark-core_2.11</artifactId>
      <version>2.1.0.cloudera2</version>
      <scope>provided</scope>
</dependency>
```

The complete artifact list for this release follows.

| Project | groupId | artifactId | version |
|---|---|---|---|
| Apache Spark | org.apache.spark | spark-catalyst_2.11 | 2.1.0.cloudera2 |
| | org.apache.spark | spark-core_2.11 | 2.1.0.cloudera2 |
| | org.apache.spark | spark-graphx_2.11 | 2.1.0.cloudera2 |
| | org.apache.spark | spark-hive-exec_2.11 | 2.1.0.cloudera2 |
| | org.apache.spark | spark-hive_2.11 | 2.1.0.cloudera2 |
| | org.apache.spark | spark-launcher_2.11 | 2.1.0.cloudera2 |
| | org.apache.spark | spark-mllib-local_2.11 | 2.1.0.cloudera2 |
| | org.apache.spark | spark-mllib_2.11 | 2.1.0.cloudera2 |

| Project | groupId | artifactId | version |
|---|---|---|---|
| | org.apache.spark | spark-network-common_2.11 | 2.1.0.cloudera2 |
| | org.apache.spark | spark-network-shuffle_2.11 | 2.1.0.cloudera2 |
| | org.apache.spark | spark-network-yarn_2.11 | 2.1.0.cloudera2 |
| | org.apache.spark | spark-repl_2.11 | 2.1.0.cloudera2 |
| | org.apache.spark | spark-sketch_2.11 | 2.1.0.cloudera2 |
| | org.apache.spark | spark-sql-kafka-0-10_2.11 | 2.1.0.cloudera2 |
| | org.apache.spark | spark-sql_2.11 | 2.1.0.cloudera2 |
| | org.apache.spark | spark-streaming-flume-sink_2.11 | 2.1.0.cloudera2 |
| | org.apache.spark | spark-streaming-flume_2.11 | 2.1.0.cloudera2 |
| | org.apache.spark | spark-streaming-kafka-0-10-assembly_2.11 | 2.1.0.cloudera2 |
| | org.apache.spark | spark-streaming-kafka-0-10_2.11 | 2.1.0.cloudera2 |
| | org.apache.spark | spark-streaming-kafka-0-8-assembly_2.11 | 2.1.0.cloudera2 |
| | org.apache.spark | spark-streaming-kafka-0-8_2.11 | 2.1.0.cloudera2 |
| | org.apache.spark | spark-streaming_2.11 | 2.1.0.cloudera2 |
| | org.apache.spark | spark-tags_2.11 | 2.1.0.cloudera2 |
| | org.apache.spark | spark-unsafe_2.11 | 2.1.0.cloudera2 |
| | org.apache.spark | spark-yarn_2.11 | 2.1.0.cloudera2 |

### CDS 2.1 Release 1 Maven Artifacts

The following pom fragment shows how to access a CDS 2.1 Release 1 artifact from a Maven POM.

```
<dependency>
      <groupId>org.apache.spark</groupId>
      <artifactId>spark-core_2.11</artifactId>
      <version>2.1.0.cloudera1</version>
      <scope>provided</scope>
</dependency>
```

The complete artifact list for this release follows.

| Project | groupId | artifactId | version |
|---|---|---|---|
| Apache Spark | org.apache.spark | spark-catalyst_2.11 | 2.1.0.cloudera1 |
| | org.apache.spark | spark-core_2.11 | 2.1.0.cloudera1 |
| | org.apache.spark | spark-graphx_2.11 | 2.1.0.cloudera1 |
| | org.apache.spark | spark-hive-exec_2.11 | 2.1.0.cloudera1 |
| | org.apache.spark | spark-hive_2.11 | 2.1.0.cloudera1 |
| | org.apache.spark | spark-launcher_2.11 | 2.1.0.cloudera1 |
| | org.apache.spark | spark-mllib-local_2.11 | 2.1.0.cloudera1 |
| | org.apache.spark | spark-mllib_2.11 | 2.1.0.cloudera1 |
| | org.apache.spark | spark-network-common_2.11 | 2.1.0.cloudera1 |
| | org.apache.spark | spark-network-shuffle_2.11 | 2.1.0.cloudera1 |
| | org.apache.spark | spark-network-yarn_2.11 | 2.1.0.cloudera1 |
| | org.apache.spark | spark-repl_2.11 | 2.1.0.cloudera1 |
| | org.apache.spark | spark-sketch_2.11 | 2.1.0.cloudera1 |
| | org.apache.spark | spark-sql-kafka-0-10_2.11 | 2.1.0.cloudera1 |
| | org.apache.spark | spark-sql_2.11 | 2.1.0.cloudera1 |
| | org.apache.spark | spark-streaming-flume-sink_2.11 | 2.1.0.cloudera1 |
| | org.apache.spark | spark-streaming-flume_2.11 | 2.1.0.cloudera1 |
| | org.apache.spark | spark-streaming-kafka-0-10-assembly_2.11 | 2.1.0.cloudera1 |
| | org.apache.spark | spark-streaming-kafka-0-10_2.11 | 2.1.0.cloudera1 |
| | org.apache.spark | spark-streaming-kafka-0-8-assembly_2.11 | 2.1.0.cloudera1 |
| | org.apache.spark | spark-streaming-kafka-0-8_2.11 | 2.1.0.cloudera1 |

| Project | groupId | artifactId | version |
|---|---|---|---|
| | org.apache.spark | spark-streaming_2.11 | 2.1.0.cloudera1 |
| | org.apache.spark | spark-tags_2.11 | 2.1.0.cloudera1 |
| | org.apache.spark | spark-unsafe_2.11 | 2.1.0.cloudera1 |
| | org.apache.spark | spark-yarn_2.11 | 2.1.0.cloudera1 |

## CDS 2.0 Powered by Apache Spark Maven Artifacts

The following tables lists the `groupId`, `artifactId`, and `version` required to access the artifacts for each CDS 2.0 Powered by Apache Spark release:

### CDS 2.0 Release 2 Maven Artifacts

The following pom fragment shows how to access a CDS 2.0 Release 2 artifact from a Maven POM.

```
<dependency>
      <groupId>org.apache.spark</groupId>
      <artifactId>spark-core_2.11</artifactId>
      <version>2.0.0.cloudera2</version>
      <scope>provided</scope>
</dependency>
```

The complete artifact list for this release follows.

| Project | groupId | artifactId | version |
|---|---|---|---|
| Apache Spark | org.apache.spark | spark-catalyst_2.11 | 2.0.0.cloudera2 |
| | org.apache.spark | spark-core_2.11 | 2.0.0.cloudera2 |
| | org.apache.spark | spark-graphx_2.11 | 2.0.0.cloudera2 |
| | org.apache.spark | spark-hive-exec_2.11 | 2.0.0.cloudera2 |
| | org.apache.spark | spark-hive_2.11 | 2.0.0.cloudera2 |
| | org.apache.spark | spark-launcher_2.11 | 2.0.0.cloudera2 |
| | org.apache.spark | spark-mllib-local_2.11 | 2.0.0.cloudera2 |
| | org.apache.spark | spark-mllib_2.11 | 2.0.0.cloudera2 |
| | org.apache.spark | spark-network-common_2.11 | 2.0.0.cloudera2 |
| | org.apache.spark | spark-network-shuffle_2.11 | 2.0.0.cloudera2 |

| Project | groupId | artifactId | version |
|---|---|---|---|
| | org.apache.spark | spark-network-yarn_2.11 | 2.0.0.cloudera2 |
| | org.apache.spark | spark-repl_2.11 | 2.0.0.cloudera2 |
| | org.apache.spark | spark-sketch_2.11 | 2.0.0.cloudera2 |
| | org.apache.spark | spark-sql_2.11 | 2.0.0.cloudera2 |
| | org.apache.spark | spark-streaming-flume-sink_2.11 | 2.0.0.cloudera2 |
| | org.apache.spark | spark-streaming-flume_2.11 | 2.0.0.cloudera2 |
| | org.apache.spark | spark-streaming-kafka-0-8_2.11 | 2.0.0.cloudera2 |
| | org.apache.spark | spark-streaming_2.11 | 2.0.0.cloudera2 |
| | org.apache.spark | spark-tags_2.11 | 2.0.0.cloudera2 |
| | org.apache.spark | spark-unsafe_2.11 | 2.0.0.cloudera2 |
| | org.apache.spark | spark-yarn_2.11 | 2.0.0.cloudera2 |

### CDS 2.0 Release 1 Maven Artifacts

The following pom fragment shows how to access a CDS 2.0 Release 1 artifact from a Maven POM.

```
<dependency>
      <groupId>org.apache.spark</groupId>
      <artifactId>spark-core_2.11</artifactId>
      <version>2.0.0.cloudera1</version>
      <scope>provided</scope>
</dependency>
```

The complete artifact list for this release follows.

| Project | groupId | artifactId | version |
|---|---|---|---|
| Apache Spark | org.apache.spark | spark-catalyst_2.11 | 2.0.0.cloudera1 |
| | org.apache.spark | spark-core_2.11 | 2.0.0.cloudera1 |
| | org.apache.spark | spark-graphx_2.11 | 2.0.0.cloudera1 |
| | org.apache.spark | spark-hive_2.11 | 2.0.0.cloudera1 |
| | org.apache.spark | spark-launcher_2.11 | 2.0.0.cloudera1 |

| Project | groupId | artifactId | version |
|---|---|---|---|
| | org.apache. spark | spark-mllib-local_2.11 | 2.0.0. cloudera1 |
| | org.apache. spark | spark-mllib_2.11 | 2.0.0. cloudera1 |
| | org.apache. spark | spark-network-common_2.11 | 2.0.0. cloudera1 |
| | org.apache. spark | spark-network-shuffle_2.11 | 2.0.0. cloudera1 |
| | org.apache. spark | spark-network-yarn_2.11 | 2.0.0. cloudera1 |
| | org.apache. spark | spark-repl_2.11 | 2.0.0. cloudera1 |
| | org.apache. spark | spark-sketch_2.11 | 2.0.0. cloudera1 |
| | org.apache. spark | spark-sql_2.11 | 2.0.0. cloudera1 |
| | org.apache. spark | spark-streaming-flume-sink_2.11 | 2.0.0. cloudera1 |
| | org.apache. spark | spark-streaming-flume_2.11 | 2.0.0. cloudera1 |
| | org.apache. spark | spark-streaming-kafka-0-8_2.11 | 2.0.0. cloudera1 |
| | org.apache. spark | spark-streaming_2.11 | 2.0.0. cloudera1 |
| | org.apache. spark | spark-tags_2.11 | 2.0.0. cloudera1 |
| | org.apache. spark | spark-unsafe_2.11 | 2.0.0. cloudera1 |
| | org.apache. spark | spark-yarn_2.11 | 2.0.0. cloudera1 |

# Installing or Upgrading CDS Powered by Apache Spark

**Minimum Required Role: Cluster Administrator** (also provided by **Full Administrator**)

CDS Powered by Apache Spark is distributed as two files: a custom service descriptor file and a parcel, both of which must be installed on the cluster.

> **Note:** Due to the potential for confusion between *CDS Powered by Apache Spark* and the initialism *CSD*, references to the custom service descriptor (CSD) file in this documentation use the term *service descriptor*.

## Install CDS Powered by Apache Spark

> **Note:**
>
> Although Spark 1 and Spark 2 can coexist in the same CDH cluster, you cannot use multiple Spark 2 versions simultaneously in the same Cloudera Manager instance. All CDH clusters managed by the same Cloudera Manager Server must use exactly the same version of CDS Powered by Apache Spark. For example, you cannot use the built-in CDH Spark service, a CDS 2.0 service, and a CDS 2.1 service. You must choose only one CDS 2 Powered by Apache Spark release. Make sure to install or upgrade the CDS 2 service descriptor and parcels across all machines of all clusters at the same time.

Follow these steps to install CDS Powered by Apache Spark:

1. Check that all the software prerequisites are satisfied. If not, you might need to upgrade or install other software components first. See CDS Powered by Apache Spark Requirements on page 7 for details.
2. Install the CDS Powered by Apache Spark service descriptor into Cloudera Manager.

   > **Important:**
   >
   > Because CDS Powered by Apache Spark is only installable using the parcel mechanism, it can only be used on clusters managed by Cloudera Manager. Additionally, because Cloudera Manager does not support using parcels and packages in the same cluster, you cannot use CDS if you are using a package-based installation of CDH.

   a. To download the CDS Powered by Apache Spark service descriptor, in the Version Information table in CDS Versions Available for Download on page 28, click the service descriptor link for the version you want to install.
   b. Log on to the Cloudera Manager Server host, and copy the CDS Powered by Apache Spark service descriptor in the location configured for service descriptor files.
   c. Set the file ownership of the service descriptor to `cloudera-scm:cloudera-scm` with permission 644.
   d. Restart the Cloudera Manager Server with the following command:

```
service cloudera-scm-server restart
```

3. In the Cloudera Manager Admin Console, add the CDS Powered by Apache Spark parcel repository to the Remote Parcel Repository URLs in Parcel Settings as described in Parcel Configuration Settings.

> **Note:** If your Cloudera Manager Server does not have Internet access, you can use the CDS Powered by Apache Spark parcel files, put them into a new parcel repository, and then configure the Cloudera Manager Server to target this newly created repository.

4. Download the CDS Powered by Apache Spark parcel, distribute the parcel to the hosts in your cluster, and activate the parcel. See Managing Parcels.
5. Add the Spark 2 service to your cluster.
   1. In step #1, select a dependency option:
      - HDFS, YARN, ZooKeeper: Choose this option if you do not need access to a Hive service.
      - HDFS, Hive, YARN, ZooKeeper: Hive is an optional dependency for the Spark service. If you have a Hive service and want to access Hive tables from your Spark applications, choose this option to include Hive as a dependency and have the Hive client configurations always available to Spark applications.
   2. In step #2, when customizing the role assignments for CDS Powered by Apache Spark, add a gateway role to every host.
   3. Note that the History Server port is 18089 instead of the usual 18088.
   4. Complete the steps to add the Spark 2 service.
6. Return to the Home page by clicking the Cloudera Manager logo.
7. Click ⏻ to restart the cluster.

## Upgrading to CDS 2.1 Powered By Apache Spark

> **Note:**
>
> Although Spark 1 and Spark 2 can coexist in the same CDH cluster, you cannot use multiple Spark 2 versions simultaneously in the same Cloudera Manager instance. All CDH clusters managed by the same Cloudera Manager Server must use exactly the same version of CDS Powered by Apache Spark. For example, you cannot use the built-in CDH Spark service, a CDS 2.0 service, and a CDS 2.1 service. You must choose only one CDS 2 Powered by Apache Spark release. Make sure to install or upgrade the CDS 2 service descriptor and parcels across all machines of all clusters at the same time.

If you are already using CDS 2.0, here are the steps to upgrade to CDS 2.1, while keeping any non-default configurations for Spark 2 that have already been applied:

- Remove the service descriptor JAR for the older version of CDS Powered by Apache Spark from `/opt/cloudera/csd`. Refer to CDS Powered by Apache Spark Version, Packaging, and Download Information on page 28 for the names of the JAR files corresponding to each version.
- Add the service descriptor JAR for CDS 2.1 to `/opt/cloudera/csd`. Set correct permissions and ownership.
- Restart the `cloudera-scm-server` service.
- In Cloudera Manager, deactivate the parcel corresponding to the older version of CDS Powered by Apache Spark.
- In Cloudera Manager, activate the parcel corresponding to CDS 2.1.
- Restart services and deploy the client configurations.

# Administering CDS 2 Powered by Apache Spark

Most administration tasks are the same whether you are using Spark 1 or Spark 2. To configure and manage Spark, follow the procedures in the Cloudera Enterprise Spark Guide.

In addition, follow these procedures that are specific to Spark 2:

## Configuring Spark 2 Tools as the Default

**Important:** If you configure Spark 2 as the default, modules such as SparkOnHBase and HiveOnSpark no longer work due to dependencies on Spark 1.6 in CDH. For more information, see Spark 2 Known Issues on page 9.

When you start trying out Spark 2, you can do most of your testing by running the standard Spark 1 commands such as `pyspark` and `spark-shell` alongside their Spark 2 equivalents such as `pyspark2` and `spark2-shell`. All of these commands are represented as symbolic links in `/usr/bin`.

If you are testing a workflow that has the original command names hardcoded in other scripts, you might configure the system so that issuing the `pyspark` command really runs the `pyspark2` script, and so on for other Spark-related binaries. This change is done using the Linux “alternatives” mechanism, which keeps track of the appropriate target for each of the `/usr/bin` symlinks.

To use Spark 2 tools as the default, run the following script *on all hosts in the cluster*:

```
for binary in pyspark spark-shell spark-submit; do
  # Generate the name of the new binary e.g. pyspark2, spark2-shell, etc.
  new_binary=$(echo $binary | sed -e 's/spark/spark2/')
  # Update the old alternative to the client binary to the new client binary
  # Use priority 11 because the default priority with which these alternatives are
created is 10
  update-alternatives --install /usr/bin/${binary} ${binary} /usr/bin/${new_binary} 11
done
# For configuration, we need to have a separate command
# because the destination is under /etc/ instead of /usr/bin like for binaries.
# The priority is different - 52 because Cloudera Manager sets up configuration symlinks
# with priority 51.
update-alternatives --install /etc/spark/conf spark-conf /etc/spark2/conf 52
```

To remove this setting and return to using the Spark contained in CDH, run the following script *on all hosts in the cluster*. It removes the Spark 2 targets of the symlinks and points those symlinks back to the original Spark-related scripts:

```
for binary in pyspark spark-shell spark-submit; do
  new_binary=$(echo $binary | sed -e 's/spark/spark2/')
  update-alternatives --remove ${binary} /usr/bin/${new_binary}
done
update-alternatives --remove spark-conf /etc/spark2/conf
```

# Security Considerations for CDS 2 Powered by Apache Spark

The following sections cover any special considerations for security in CDS 2 Powered by Apache Spark, or differences in security settings between Spark 1.6 and Spark 2.

## Configuration Settings for Encryption

To enable encryption for the shuffle service with CDS 2, use the following configuration setting:

```
spark.io.encryption.enabled=true
```

This is a change from Spark 1.6, where the corresponding option is `spark.shuffle.encryption.enabled=true`.

# Running Applications with CDS Powered by Apache Spark

With CDS Powered by Apache Spark, you can run Apache Spark 2 applications locally or distributed across a cluster, either by using an interactive shell or by submitting an application. Running Spark applications interactively is commonly performed during the data-exploration phase and for ad hoc analysis.

## The Spark 2 Job Commands

With Spark 2, you use slightly different command names than in Spark 1, so that you can run both versions of Spark side-by-side without conflicts:

- `spark2-submit` instead of `spark-submit`.
- `spark2-shell` instead of `spark-shell`.
- `pyspark2` instead of `pyspark`.

For development and test purposes, you can also configure each host so that invoking the Spark 1 command name runs the corresponding Spark 2 executable. See Configuring Spark 2 Tools as the Default on page 39 for details.

## Canary Test for pyspark2 Command

The following example shows a simple `pyspark2` session that refers to the SparkContext, calls the `collect()` function which runs a Spark 2 job, and writes data to HDFS. This sequence of operations helps to check if there are obvious configuration issues that prevent Spark 2 jobs from working at all. For the HDFS path for the output directory, substitute a path that exists on your own system.

```
$ hdfs dfs -mkdir /user/jdoe/spark
$ pyspark2
...
SparkSession available as 'spark'.
>>> strings = ["one","two","three"]
>>> s2 = sc.parallelize(strings)
>>> s3 = s2.map(lambda word: word.upper())
>>> s3.collect()
['ONE', 'TWO', 'THREE']
>>> s3.saveAsTextFile('hdfs:///user/jdoe/spark/canary_test')
>>> quit()
$ hdfs dfs -ls /user/jdoe/spark
Found 1 items
drwxr-xr-x   - jdoe spark-users  0 2016-08-26 14:41 /user/jdoe/spark/canary_test
$ hdfs dfs -ls /user/jdoe/spark/canary_test
Found 3 items
-rw-r--r--   3 jdoe spark-users  0 2016-08-26 14:41 /user/jdoe/spark/canary_test/_SUCCESS
-rw-r--r--   3 jdoe spark-users  4 2016-08-26 14:41
/user/jdoe/spark/canary_test/part-00000
-rw-r--r--   3 jdoe spark-users 10 2016-08-26 14:41
/user/jdoe/spark/canary_test/part-00001
$ hdfs dfs -cat /user/jdoe/spark/canary_test/part-00000
ONE
$ hdfs dfs -cat /user/jdoe/spark/canary_test/part-00001
TWO
THREE
```

## Fetching Spark 2 Maven Dependencies

The Maven coordinates are a combination of groupId, artifactId and version. The groupId and artifactId are the same as for the upstream Apache Spark project. For example, for `spark-core`, groupId is `org.apache.spark`, and artifactId is spark-core_2.11, both the same as the upstream project. The version is different for the Cloudera packaging: see Using the CDS Powered by Apache Spark Maven Repository on page 29 for the exact name depending on which release you are using.

## Adapting the Spark WordCount App for Spark 2

The following pom fragment shows how to access a CDS Powered by Apache Spark artifact from a Maven POM.

```
<dependency>
      <groupId>org.apache.spark</groupId>
      <artifactId>spark-core_2.11</artifactId>
      <version>2.1.0.cloudera3</version>
      <scope>provided</scope>
</dependency>
```

Use this dependency definition to update pom.xml for the example described in Developing and Running a Spark WordCount Application. If you are using a different CDS version, see Using the CDS Powered by Apache Spark Maven Repository on page 29.

To account for changes in the Spark 2 API, before building the example, make the following updates to com.cloudera.sparkwordcount.JavaWordCount:

- Add import java.util.Iterator;
- Replace all instances of Iterable with Iterator.
- Perform the following replacements:
  - return Arrays.asList(s.split(" ")); to return Arrays.asList(s.split(" ")).iterator();
  - return chars; to return chars.iterator();

## Accessing the Spark 2 History Server

The Spark 2 history server is available on port 18089, rather than port 18088 as with the Spark 1 history server.

# Spark 2 Kafka Integration

**Minimum Required Role:** **Cluster Administrator** (also provided by **Full Administrator**)

Cloudera Spark 2.1 release 1 and later include a Kafka integration feature that uses the new Kafka consumer API. This new Kafka consumer API supports reading data from secure Kafka clusters. In this context, secure clusters are those that are authenticated by Kerberos, and optionally using TLS/SSL for wire encryption.

## Requirements

To read data securely from Kafka, or to use the new Spark-Kafka integration that uses the new Kafka consumer API, requires the following software versions:

- Cloudera Spark 2.1 release 1 or later.
- CDK 2.1 or later. (Although the Kafka consumer API is available starting in CDK 2.0, the Spark integration requires Kafka 2.1.)

## Running Spark Jobs that Integrate with Kafka

To run jobs that use the new Kafka integration, you can use one of the following two techniques.

> **Note:**
>
> If you do not intend to use the new Kafka consumer integration and are using the existing Kafka integration (either the Kafka receiver or the direct connector), you do not need to do anything regardless of what Kafka version you are using. If you want to use the new Kafka consumer API integration, use one of the following two techniques to make sure you are using the newer Kafka jars in the Spark classpath.

**Technique #1: Set SPARK_KAFKA_VERSION environment variable**

When running jobs that require the new Kafka integration, set `SPARK_KAFKA_VERSION=0.10` in the shell before launching spark-submit. Use the appropriate environment variable syntax for your shell, such as:

```
# Permanently set the environment variable
export SPARK_KAFKA_VERSION=0.10
spark-submit arguments

# Or:

# Set the environment variable only for the duration of 1 command
SPARK_KAFKA_VERSION=0.10 spark-submit arguments
```

**Technique #2: Set spark_kafka_version setting through Cloudera Manager**

Set `spark_kafka_version` configuration in Cloudera Manager’s Spark 2 service to be 0.10 and redeploy the client configuration. No need to source any environment variables when launching `spark-submit`.

Technique #2 is preferable if you have upgraded your Kafka brokers to CDK 2.1 or higher, and do not intend to have your Spark jobs communicate with brokers running a version of Kafka prior to CDK 2.1.

If you modify the default Kafka version to 0.10 by using technique #2, you can connect to old Kafka brokers (for example, one based on CDK 2.0) by setting `SPARK_KAFKA_VERSION=0.9` when running your application.

**Note:** To keep compatibility and not break any existing users, the default is to use older (Cloudera Kafka 2.0-based) client jars in the Spark classpath. Those jars let users communicate with older (Cloudera Kafka 2.0) brokers but also newer brokers (Cloudera Kafka 2.1) brokers as long as the old Kafka consumer APIs are being used. However, to use the new kafka consumer API integration, one of the two steps are required. This is to ensure that the new Kafka client jars (from Cloudera Kafka 2.1) are in the Spark classpath. It is not practical to put both Kafka 2.0 client jars and Kafka 2.1 clients jars on the Spark classpath due to incompatibilities; the engineering best practice is to only use one version of any jar file in the classpath.

## Building Applications

To build applications against the new Kafka integration, you can add the dependency by using the following Maven coordinates:

```
<dependency>
  <groupId>org.apache.spark</groupId>
  <artifactId>spark-streaming-kafka-0-10_2.11</artifactId>
  <version>2.1.0.cloudera1</version>
</dependency>
```

## Reading from Authorized Kafka

To read from a Kafka cluster authorized by Sentry, give privileges to your consumer group as described in Configuring Kafka Security. You must also grant privileges to another consumer group `spark-executor-`*`your_consumer_group`* in the same way. This is because the driver uses the consumer group specified in your app, but the executors use a different consumer group, which is hardcoded to `spark-executor-`*`your_consumer_group`*.

# Troubleshooting for Spark 2

Troubleshooting for Spark mainly involves checking configuration settings and application code to diagnose performance and scalability issues.

### Commercial support for GA version

Cloudera customers with commercial support can now use normal support channels for the CDS 2.0 Powered by Apache Spark component.

### Error instantiating Hive metastore class

A Hive compatibility issue in CDS 2.0 Release 1 affects CDH 5.10.1 and higher, CDH 5.9.2 and higher, CDH 5.8.5 and higher, and CDH 5.7.6 and higher. If you are using one of these CDH versions, you must upgrade to CDS 2.0 Release 2 or higher to avoid Spark 2 job failures when using Hive functionality.

When you encounter a problem due to the Hive compatibility issue, the error stack starts like this:

```
java.lang.RuntimeException: Unable to instantiate
  org.apache.hadoop.hive.ql.metadata.SessionHiveMetaStoreClient
    at org.apache.hadoop.hive.metastore.MetaStoreUtils.newInstance
      (MetaStoreUtils.java:1545)
        at org.apache.hadoop.hive.metastore
          .RetryingMetaStoreClient.<init>(RetryingMetaStoreClient.
```

The solution is to upgrade to  2.0 Release 2 or higher.

### Wrong version of Python

When you use Spark 2 with Python 2.x, you must use Python 2.7 or higher. You might need to install a new version of Python on all hosts in the cluster, because some Linux distributions come with Python 2.6 by default. If the right level of Python is not picked up by default, set the PYSPARK_PYTHON and PYSPARK_DRIVER_PYTHON environment variables to point to the correct Python executable before running the `pyspark2` command.

### API changes that are not backward-compatible

Between Spark 1.6 and Spark 2.0, some APIs have changed in ways that are not backward compatible. Recompile all applications to take advantage of Spark 2 capabilities. For any compilation errors, check if the corresponding function has changed in Spark 2, and if so, change your code to use the latest function name, parameters, and return type.

### A Spark component does not work or is unstable

Certain components from the Spark ecosystem are explicitly not supported with CDS 2 Powered by Apache Spark. Check against the compatibility matrix for Spark to make sure the components you are using are all intended to work with CDS 2 Powered by Apache Spark and CDH.

# Frequently Asked Questions about CDS 2 Powered by Apache Spark

**Note:**

This documentation refers to CDS 2.1 Release 3 powered by Apache Spark. This component is generally available and is supported on CDH 5.7 and higher.

A Hive compatibility issue in CDS 2.0 Release 1 affects CDH 5.10.1 and higher, CDH 5.9.2 and higher, CDH 5.8.5 and higher, and CDH 5.7.6 and higher. If you are using one of these CDH versions, you must upgrade to CDS 2.0 Release 2 or higher to avoid Spark 2 job failures when using Hive functionality.

This Frequently Asked Questions (FAQ) page covers general information about CDS 2 Powered by Apache Spark, coexistence with Spark 1, and other questions that are relevant for early adopters of the latest Spark 2 features.

## Running Spark 1 and Spark 2 Side-by-Side

The Spark 2 service does not conflict with Spark 1 if it is installed. The history server uses a different port. Spark 2 shares the Spark 1 shuffle service if already available, or installs the shuffle service if not.

## Why doesn't feature or library XYZ work?

A number of features, components, libraries, and integration points from Spark 1.6 are not supported with the CDS 2 Powered by Apache Spark. See Spark 2 Known Issues on page 9 for details.

# Appendix: Apache License, Version 2.0

**SPDX short identifier: Apache-2.0**

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License.

Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License.

Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution.

You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

1. You must give any other recipients of the Work or Derivative Works a copy of this License; and
2. You must cause any modified files to carry prominent notices stating that You changed the files; and
3. You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
4. If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions.

Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks.

This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty.

Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability.

In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability.

While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

### APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

```
Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

  http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```