

<!-- Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership. The ASF licenses this file to you 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. -->

Spark Load

Spark load 通过 Spark 实现对导入数据的预处理，提高 Doris 大数据量的导入性能并且节省 Doris 集群的计算资源。主要用于初次迁移，大数据量导入 Doris 的场景。

Spark load 是一种异步导入方式，用户需要通过 MySQL 协议创建 Spark 类型导入任务，并通过 `SHOW LOAD` 查看导入结果。

适用场景

- 源数据在 Spark 可以访问的存储系统中，如 HDFS。
- 数据量在几十 GB 到 TB 级别。

名词解释

1. Frontend (FE)：Doris 系统的元数据和调度节点。在导入流程中主要负责导入任务的调度工作。
2. Backend (BE)：Doris 系统的计算和存储节点。在导入流程中主要负责数据写入及存储。
3. Spark ETL：在导入流程中主要负责数据的 ETL 工作，包括全局字典构建、分区、排序、聚合等。

基本原理

基本流程

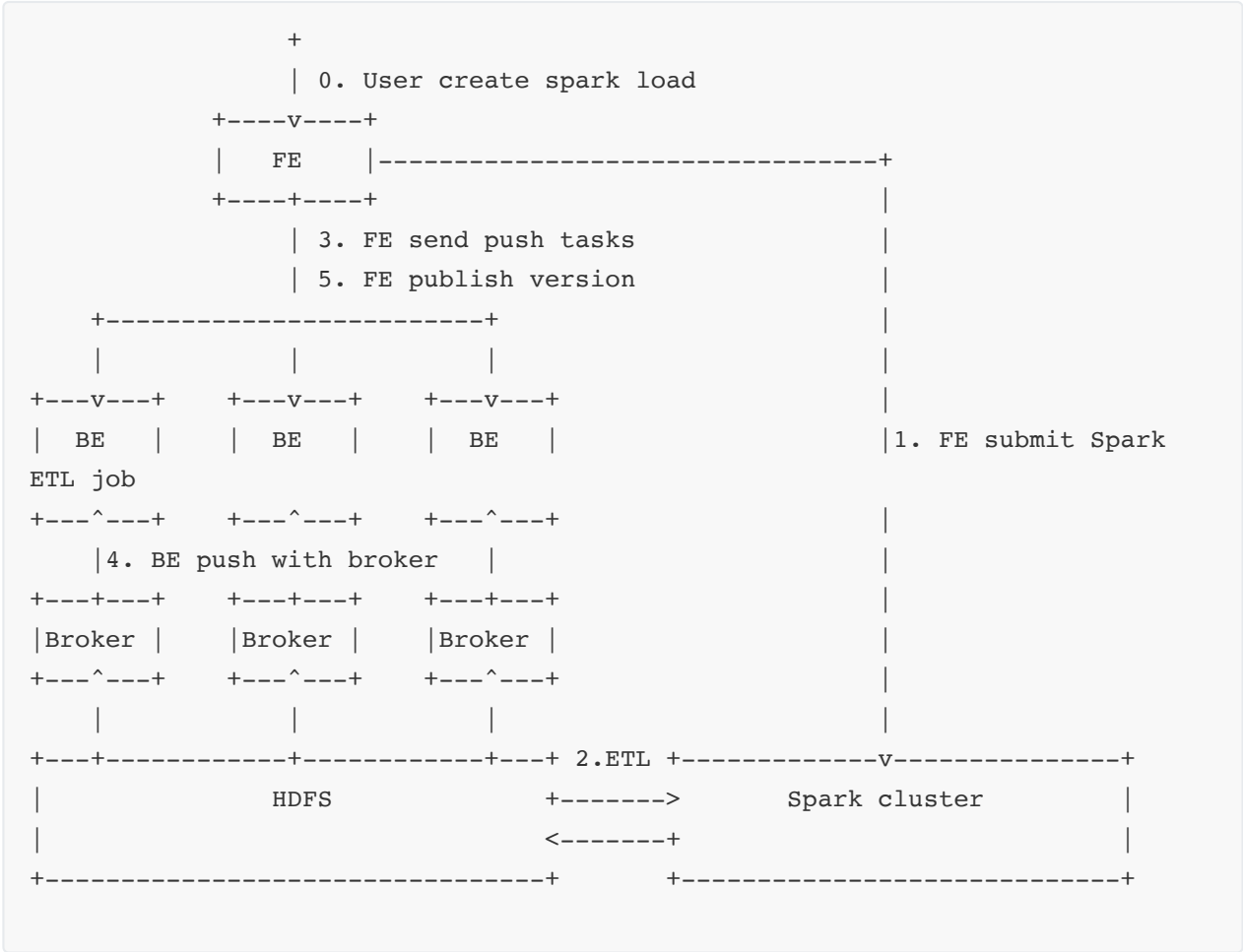
用户通过 MySQL 客户端提交 Spark 类型导入任务，FE 记录元数据并返回用户提交成功。

Spark load 任务的执行主要分为以下5个阶段。

1. FE 调度提交 ETL 任务到 Spark 集群执行。
2. Spark 集群执行 ETL 完成对导入数据的预处理。包括全局字典构建（BITMAP 类型）、分区、排

序、聚合等。

- 3. ETL 任务完成后，FE 获取预处理过的每个 tablet 的数据路径，并调度相关的 BE 执行 Push 任务。
- 4. BE 通过 Broker 读取数据，转化为 Doris 底层存储格式。
- 5. FE 调度生效版本，完成导入任务。



全局字典

待补

数据预处理（DPP）

待补

基本操作

配置 ETL 集群

提交 Spark 导入任务之前，需要配置执行 ETL 任务的 Spark 集群。

语法：

```
// 添加 ETL 集群
ALTER SYSTEM ADD LOAD CLUSTER cluster_name
PROPERTIES("key1" = "value1", ...)

// 删除 ETL 集群
ALTER SYSTEM DROP LOAD CLUSTER cluster_name

// 查看 ETL 集群
SHOW LOAD CLUSTERS
```

`cluster_name` 为 Doris 中配置的 Spark 集群的名字。

PROPERTIES是 ETL 集群相关参数，如下：

- `type`：集群类型，必填，目前仅支持 spark。
- Spark ETL 集群相关参数如下：
 - `master`：必填，目前支持 yarn，spark://host:port。
 - `deploy_mode`：可选，默认为 cluster。支持 cluster，client 两种。
 - `hdfs_etl_path`：ETL 使用的 HDFS 目录。必填。例如：hdfs://host:port/tmp/doris。
 - `broker`：broker 名字。必填。需要使用 ALTER SYTEM ADD BROKER 命令提前完成配置。
 - `yarn_configs`：HDFS YARN 参数，master 为 yarn 时必填。需要指定 yarn.resourcemanager.address 和 fs.defaultFS。不同 configs 之间使用 ; 拼接。
 - `spark_args`：Spark 任务提交时指定的参数，可选。具体可参考 spark-submit 命令，每个 arg 必须以 -- 开头，不同 args 之间使用 ; 拼接。例如--files=/file1,/file2;--jars=/a.jar,/b.jar。
 - `spark_configs`：Spark 参数，可选。具体参数可参考<http://spark.apache.org/docs/latest/configuration.html>。不同 configs 之间使用 ; 拼接。

示例：

```
ALTER SYSTEM ADD LOAD CLUSTER "cluster0"
PROPERTIES
(
  "type" = "spark",
  "master" = "yarn",
  "hdfs_etl_path" = "hdfs://1.1.0.1:801/tmp/doris",
  "broker" = "broker0",
  "yarn_configs" =
    "yarn.resourcemanager.address=1.1.1.1:800;fs.defaultFS=hdfs://1.1.1.1:801",
  "spark_args" = "--files=/file1,/file2;--jars=/a.jar,/b.jar",
  "spark_configs" = "spark.driver.memory=1g;spark.executor.memory=1g"
);

// 如果 broker 需要用户名密码等额外参数
ALTER SYSTEM ADD LOAD CLUSTER "cluster1"
PROPERTIES
(
  "type" = "spark",
```

```
"master" = "spark://1.1.1.1:802",
"deploy_mode" = "client",
"hdfs_etl_path" = "hdfs://1.1.1.1:801/tmp/doris",
"broker" = "broker1",
"broker.username" = "user1",
"broker.password" = "password1"
);
```

创建导入

语法：

```
LOAD LABEL load_label
    (data_desc, ...)
    WITH CLUSTER cluster_name cluster_properties
    [PROPERTIES (key1=value1, ... )]

* load_label:
    db_name.label_name

* data_desc:
    DATA INFILE ( 'file_path', ...)
    [NEGATIVE]
    INTO TABLE tbl_name
    [PARTITION (p1, p2)]
    [COLUMNS TERMINATED BY separator ]
    [(col1, ...)]
    [SET (k1=f1(xx), k2=f2(xx))]
    [WHERE predicate]

* cluster_properties:
    (key2=value2, ...)
```

示例：

```
LOAD LABEL db1.label1
(
    DATA INFILE("hdfs://abc.com:8888/user/palo/test/ml/file1")
    INTO TABLE tbl1
    COLUMNS TERMINATED BY ","
    (tmp_c1,tmp_c2)
    SET
    (
        id=tmp_c2,
        name=tmp_c1
    ),

```

```

DATA INFILE("hdfs://abc.com:8888/user/palo/test/ml/file2")
INTO TABLE tbl2
COLUMNS TERMINATED BY ","
(col1, col2)
where col1 > 1
)
WITH CLUSTER 'cluster0'
(
    "broker.username"="user",
    "broker.password"="pass"
)
PROPERTIES
(
    "timeout" = "3600"
);

```

创建导入的详细语法执行 `HELP SPARK LOAD` 查看语法帮助。这里主要介绍 Spark load 的创建导入语法中参数意义和注意事项。

Label

导入任务的标识。每个导入任务，都有一个在单 database 内部唯一的 Label。具体规则与 `Broker Load` 一致。

数据描述类参数

支持的数据源取决于Spark支持的数据源。其他规则与 `Broker Load` 一致。

导入作业参数

导入作业参数主要指的是 Spark load 创建导入语句中的属于 `opt_properties` 部分的参数。导入作业参数是作用于整个导入作业的。规则与 `Broker Load` 一致。（timeout待测）

Cluster 参数

ETL cluster需要提前配置到 Doris系统中才能使用 Spark load。

当用户有临时性的需求，比如增加任务使用的资源而修改 Spark configs，可以在这里设置，设置仅对本次任务生效，并不影响 Doris 集群中已有的配置。

另外如果需要指定额外的 Broker 参数，则需要指定"broker.key" = "value"。具体参数请参阅 [Broker文档](#)。例如需要指定用户名密码，如下：

```

WITH CLUSTER 'cluster0'
(
    "broker.username"="user1",
    "broker.password"="password1"
)

```

查看导入

Spark load 导入方式同 Broker load 一样都是异步的，所以用户必须将创建导入的 Label 记录，并且在查看导入命令中使用 **Label** 来查看导入结果。查看导入命令在所有导入方式中是通用的，具体语法可执行 `HELP SHOW LOAD` 查看。

示例：

```
mysql> show load order by createtime desc limit 1\G
***** 1. row *****
      JobId: 76391
      Label: label1
      State: FINISHED
      Progress: ETL:100%; LOAD:100%
      Type: SPARK
      EtlInfo: unselected.rows=4; dpp.abnorm.ALL=15; dpp.norm.ALL=28133376
      TaskInfo: cluster:cluster0; timeout(s):10800; max_filter_ratio:5.0E-5
      ErrorMsg: N/A
      CreateTime: 2019-07-27 11:46:42
      EtlStartTime: 2019-07-27 11:46:44
      EtlFinishTime: 2019-07-27 11:49:44
      LoadStartTime: 2019-07-27 11:49:44
      LoadFinishTime: 2019-07-27 11:50:16
      URL: http://1.1.1.1:8089/proxy/application_1586619723848_0035/
      JobDetails:
      {"ScannedRows":28133395,"TaskNumber":1,"FileNumber":1,"FileSize":200000}
```

返回结果集中参数意义可以参考 Broker load。不同点如下：

- State

导入任务当前所处的阶段。任务提交之后状态为 PENDING，提交 Spark ETL 之后状态变为 ETL，ETL 完成之后 FE 调度 BE 执行 push 操作状态为 LOADING，push 完成并且版本生效后状态变为 FINISHED。

导入任务的最终阶段有两个：CANCELLED 和 FINISHED，当 Load job 处于这两个阶段时导入完成。其中 CANCELLED 为导入失败，FINISHED 为导入成功。

- Progress

导入任务的进度描述。分为两种进度：ETL 和 LOAD，对应了导入流程的两个阶段 ETL 和 LOADING。

LOAD 的进度范围为：0~100%。

`LOAD 进度 = 当前已完成所有replica导入的tablet个数 / 本次导入任务的总tablet个数 * 100%`

如果所有导入表均完成导入，此时 **LOAD** 的进度为 **99%** 导入进入到最后生效阶段，整个导入完成后，LOAD 的进度才会改为 100%。

导入进度并不是线性的。所以如果一段时间内进度没有变化，并不代表导入没有在执行。

- Type

导入任务的类型。Spark load 为 SPARK。

- CreateTime/EtlStartTime/EtlFinishTime/LoadStartTime/LoadFinishTime

这几个值分别代表导入创建的时间，ETL 阶段开始的时间，ETL 阶段完成的时间，LOADING 阶段开始的时间和整个导入任务完成的时间。

- JobDetails

显示一些作业的详细运行状态，ETL 结束的时候更新。包括导入文件的个数、总大小（字节）、子任务个数、已处理的原始行数等。

```
{"ScannedRows":139264,"TaskNumber":1,"FileNumber":1,"FileSize":940754064}
```

取消导入

当 Spark load 作业状态不为 CANCELLED 或 FINISHED 时，可以被用户手动取消。取消时需要指定待取消导入任务的 Label。取消导入命令语法可执行 `HELP CANCEL LOAD` 查看。

相关系统配置

FE 配置

下面配置属于 Spark load 的系统级别配置，也就是作用于所有 Spark load 导入任务的配置。主要通过修改 `fe.conf` 来调整配置值。

- spark_load_default_timeout_second
任务默认超时时间为259200秒（3天）。

最佳实践

应用场景

使用 Spark load 最适合的场景就是原始数据在文件系统（HDFS）中，数据量在几十 GB 到 TB 级别。小数据量还是建议使用 Stream load 或者 Broker load。

常见问题

- 待补充