

01_Doris消费kafka中数据（CSV+JSON）

时间：2023年1月11日20:04:43

官方文档：

- [订阅Kafka日志 – Apache Doris](#)
 - <https://doris.apache.org/zh-CN/docs/dev/sql-manual/sql-reference/Data-Manipulation-Statements/Load/CREATE-ROUTINE-LOAD/>
 - <https://doris.apache.org/zh-CN/docs/dev/sql-manual/sql-reference/Show-Statements/SHOW-ROUTINE-LOAD/>
 - <https://doris.apache.org/zh-CN/docs/dev/sql-manual/sql-reference/Show-Statements/SHOW-ROUTINE-LOAD-TASK/>
 - [JSON格式数据导入 – Apache Doris](#)
-

一、概述

用户可以通过提交例行导入作业，直接订阅Kafka中的消息数据，以近实时的方式进行数据同步。

Doris 自身能够保证不丢不重的订阅 Kafka 中的消息，即 Exactly-Once 消费语义。

订阅Kafka日志

用户可以通过提交例行导入作业，直接订阅Kafka中的消息数据，以近实时的方式进行数据同步。

Doris 自身能够保证不丢不重的订阅 Kafka 中的消息，即 `Exactly-Once` 消费语义。

订阅 Kafka 消息

订阅 Kafka 消息使用了 Doris 中的例行导入（Routine Load）功能。

用户首先需要创建一个例行导入作业。作业会通过例行调度，不断地发送一系列的任务，每个任务会消费一定数量 Kafka 中的消息。

请注意以下使用限制：

1. 支持无认证的 Kafka 访问，以及通过 SSL 方式认证的 Kafka 集群。
2. 支持的消息格式如下：
 - csv 文本格式。每一个 message 为一行，且行尾不包含换行符。
 - Json 格式，详见 [导入 Json 格式数据](#)。
3. 仅支持 Kafka 0.10.0.0(含) 以上版本。

访问 SSL 认证的 Kafka 集群

例行导入功能支持无认证的 Kafka 集群，以及通过 SSL 认证的 Kafka 集群。

访问 SSL 认证的 Kafka 集群需要用户提供用于认证 Kafka Broker 公钥的证书文件（ca.pem）。如果 Kafka 集群同时开启了客户端认证，则还需提供客户端的公钥（client.pem）、密钥文件（client.key），以及密钥密码。这里所需的文件需要先通过 `CREATE FILE` 命令上传到 Doris 中，并且 catalog 名称为 `kafka`。 `CREATE FILE` 命令的具体帮助可以参见 [CREATE FILE 命令手册](#)。这里给出示例：

- 上传文件

```
CREATE FILE "ca.pem" PROPERTIES("url" = "https://example_url/kafka-key/ca.pem", "catalog" = "kafka");
CREATE FILE "client.key" PROPERTIES("url" = "https://example_url/kafka-key/client.key", "catalog" = "kafka");
CREATE FILE "client.pem" PROPERTIES("url" = "https://example_url/kafka-key/client.pem", "catalog" = "kafka");
```

上传完成后，可以通过 `SHOW FILES` 命令查看已上传的文件。

创建例行导入作业

创建例行导入任务的具体命令，请参阅 [ROUTINE LOAD 命令手册](#)。这里给出示例：

1. 访问无认证的 Kafka 集群

```
CREATE ROUTINE LOAD demo.my_first_routine_load_job ON test_1
COLUMNS TERMINATED BY ","
PROPERTIES
(
  "max_batch_interval" = "20",
  "max_batch_rows" = "300000",
  "max_batch_size" = "209715200",
)
FROM KAFKA
(
  "kafka_broker_list" = "broker1:9092,broker2:9092,broker3:9092",
  "kafka_topic" = "my_topic",
  "property.group.id" = "xxx",
  "property.client.id" = "xxx",
  "property.kafka_default_offsets" = "OFFSET_BEGINNING"
);
```

- `max_batch_interval/max_batch_rows/max_batch_size` 用于控制一个子任务的运行周期。一个子任务的运行周期由最长运行

时间、最多消费行数和最大消费数据量共同决定。

2. 访问 SSL 认证的 Kafka 集群

```
CREATE ROUTINE LOAD demo.my_first_routine_load_job ON test_1
COLUMNS TERMINATED BY ",",
PROPERTIES
(
    "max_batch_interval" = "20",
    "max_batch_rows" = "300000",
    "max_batch_size" = "209715200",
)
FROM KAFKA
(
    "kafka_broker_list" = "broker1:9091,broker2:9091",
    "kafka_topic" = "my_topic",
    "property.security.protocol" = "ssl",
    "property.ssl.ca.location" = "FILE:ca.pem",
    "property.ssl.certificate.location" = "FILE:client.pem",
    "property.ssl.key.location" = "FILE:client.key",
    "property.ssl.key.password" = "abcdefg"
);
```

查看导入作业状态

查看作业状态的具体命令和示例请参阅 [SHOW ROUTINE LOAD](#) 命令文档。

查看某个作业的任务运行状态的具体命令和示例请参阅 [SHOW ROUTINE LOAD TASK](#) 命令文档。

只能查看当前正在运行中的任务，已结束和未开始的任务无法查看。

修改作业属性

用户可以修改已经创建的作业的部分属性。具体说明请参阅 [ALTER ROUTINE LOAD](#) 命令手册。

作业控制

用户可以通过 `STOP/PAUSE/RESUME` 三个命令来控制作业的停止，暂停和重启。

具体命令请参阅 [STOP ROUTINE LOAD](#)，[PAUSE ROUTINE LOAD](#)，[RESUME ROUTINE LOAD](#) 命令文档。

更多帮助

关于 ROUTINE LOAD 的更多详细语法和最佳实践，请参阅 [ROUTINE LOAD](#) 命令手册。

二、实际操作

```
CREATE TABLE user1(
    `id` varchar(32),
    `name` varchar(255),
    `age` smallint
)
UNIQUE KEY(`id`)
distributed by hash(`id`) buckets 1
```

1. 创建Routine_Load（例行导入+CSV格式消息+JSON格式消息）

- kafka消费者相关配置（Kafka官方）
 - <https://docs.confluent.io/platform/current/installation/configuration/consumer-configs.html>

语法：

```
CREATE ROUTINE LOAD [db.]job_name ON tbl_name
[merge_type]
[load_properties]
[job_properties]
FROM data_source [data_source_properties]
```

指定kafka partition的默认起始offset

如果没有指定 `kafka_partitions/kafka_offsets`，默认消费所有分区。

此时可以指定 `kafka_default_offsets` 指定起始 offset。默认为 `OFFSET_END`，即从末尾开始订阅。

示例：

```
"property.kafka_default_offsets" = "OFFSET_BEGINNING"
```

(CSV数据)

```
CREATE ROUTINE LOAD demo.my_first_routine_load_job ON test_1
COLUMNS TERMINATED BY ","
PROPERTIES
(
  "max_batch_interval" = "20",
  "max_batch_rows" = "300000",
  "max_batch_size" = "209715200",
)
FROM KAFKA
(
  "kafka_broker_list" = "kafka-server-1:9092,kafka-server-2:9092,kafka-server-3:9092",
  "kafka_topic" = "topic_xx_doris_dbname_tablename",
  "property.group.id" = "group_xx_doris_load_data_from_kafka",
  "property.client.id" = "client_xx_doris_load_data_from_kafka",
  "property.kafka_default_offsets" = "OFFSET_END"
);
```

(JSON数据，每条消息为一个JSONObject字符串)

```
CREATE ROUTINE LOAD dev_test.user1_json_load_1 ON user1
COLUMNS(id,name,age)
PROPERTIES
(
  "desired_concurrent_number"="3",
  "max_batch_interval" = "20",
  "max_batch_rows" = "300000",
  "max_batch_size" = "209715200",
  "strict_mode" = "false",
  "format" = "json"
)
FROM KAFKA
(
  "kafka_broker_list" = "kafka-server-1:9092,kafka-server-2:9092,kafka-server-3:9092",
  "kafka_topic" = "topic_xx_doris_dev_test_user1",
```



```
@RequiredArgsConstructor
public class KafkaProducer {

    private final KafkaTemplate<String, String> kafkaTemplate;

    private static String TOPIC_NAME = "topic_xx_doris_dev_test_user1";

    public void produce(String msg) {
        // auto-create topic
        kafkaTemplate.send(TOPIC_NAME, msg);
    }
}
```

```
for (int i = 0; i < 10_0000; i++) {
    int age = ThreadLocalRandom.current().nextInt(1, 130);
    User user = new User(UUID.randomUUID().toString().replaceAll("-", ""), "AAA-" + age, age);
    kafkaProducer.produce(JSONObject.toJSONString(user));
    System.out.println("第" + (i + 1) + "条消息发送成功！");
}
```

2. 查看状态

SHOW-ROUTINE-LOAD

SHOW-ROUTINE-LOAD

Name

SHOW ROUTINE LOAD

Description

该语句用于展示 Routine Load 作业运行状态

语法：

```
SHOW [ALL] ROUTINE LOAD [FOR jobName];
```

结果说明：

```
Id: 作业ID
Name: 作业名称
CreateTime: 作业创建时间
PauseTime: 最近一次作业暂停时间
EndTime: 作业结束时间
DbName: 对应数据库名称
TableName: 对应表名称
State: 作业运行状态
DataSourceType: 数据源类型: KAFKA
CurrentTaskNum: 当前子任务数量
JobProperties: 作业配置详情
DataSourceProperties: 数据源配置详情
CustomProperties: 自定义配置
Statistic: 作业运行状态统计信息
Progress: 作业运行进度
Lag: 作业延迟状态
ReasonOfStateChanged: 作业状态变更的原因
ErrorLogUrls: 被过滤的质量不合格的数据的查看地址
OtherMsg: 其他错误信息
```

SHOW-ROUTINE-LOAD-TASK

SHOW-ROUTINE-LOAD-TASK

Name

SHOW ROUTINE LOAD TASK

Description

查看一个指定的 Routine Load 作业的当前正在运行的子任务情况。

```
SHOW ROUTINE LOAD TASK
WHERE JobName = "job_name";
```

返回结果如下：

```
TaskId: d67ce537f1be4b86-abf47530b79ab8e6
TxnId: 4
TxnStatus: UNKNOWN
JobId: 10280
CreateTime: 2020-12-12 20:29:48
ExecuteStartTime: 2020-12-12 20:29:48
Timeout: 20
BeId: 10002
DataSourceProperties: {"0":19}
```

- `TaskId`：子任务的唯一 ID。
- `TxnId`：子任务对应的导入事务 ID。
- `TxnStatus`：子任务对应的导入事务状态。通常为 UNKNOWN。并无实际意思。
- `JobId`：子任务对应的作业 ID。
- `CreateTime`：子任务的创建时间。
- `ExecuteStartTime`：子任务被调度执行的时间，通常晚于创建时间。
- `Timeout`：子任务超时时间，通常是作业设置的 `MaxIntervalS` 的两倍。
- `BeId`：执行这个子任务的 BE 节点 ID。
- `DataSourceProperties`：子任务准备消费的 Kafka Partition 的起始 offset。是一个 Json 格式字符串。Key 为 Partition Id。Value 为消费的起始 offset。

