

# MapReduce 服务

## 最佳实践

文档版本 01  
发布日期 2025-01-24



版权所有 © 华为云计算技术有限公司 2025。保留一切权利。

非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

## 商标声明



HUAWEI和其他华为商标均为华为技术有限公司的商标。

本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

## 注意

您购买的产品、服务或特性等应受华为云计算技术有限公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为云计算技术有限公司对本文档内容不做任何明示或暗示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

# 目录

<b>1 MRS 最佳实践汇总</b>	<b>1</b>
<b>2 数据分析</b>	<b>4</b>
2.1 使用 Spark2x 实现车联网车主驾驶行为分析	4
2.2 使用 Hive 加载 HDFS 数据并分析图书评分情况	12
2.3 使用 Hive 加载 OBS 数据并分析企业雇员信息	18
2.4 通过 Flink 作业处理 OBS 数据	27
2.5 通过 Spark Streaming 作业消费 Kafka 数据	32
2.6 通过 Flume 采集指定目录日志系统文件至 HDFS	39
2.7 基于 Kafka 的 Word Count 数据流统计案例	47
2.8 实时 OLAP 数据分析	52
2.8.1 实时 OLAP 场景介绍	52
2.8.2 离线数据加载：通过 CDM 将 Hive 外表数据迁移到 Doris	56
2.8.3 离线数据加载：通过 Doris Catalog 读取 Hive 外表数据并写入 Doris	68
2.8.4 离线数据加载：通过 Doris Catalog 读取 RDS-MySQL 数据并写入 Doris	77
2.8.5 离线数据加载：通过 Spark 视图读取 Hive 外表数据并写入 Doris	84
2.8.6 离线数据加载：通过 StreamLoad 将本地 CSV 文件导入 Doris	89
2.8.7 离线数据加载：通过 Broker Load 将 ORC 格式的 Hive 数据导入 Doris	94
2.8.8 实时数据加载：通过 FlinkSQL 将订单表 Kafka 数据实时同步到 Doris	100
2.8.9 实时 OLAP 数据查询：基于 Unique 表的单表检索查询和聚合查询	109
2.9 物联网时序数据分析	110
2.9.1 物联网时序数据分析场景介绍	110
2.9.2 手动将 CSV 离线数据导入至 IoTDB	112
2.9.3 使用 HetuEngine 查询 IoTDB 时序数据	125
2.9.4 使用 Grafana 对接 IoTDB 数据库	139
<b>3 数据迁移</b>	<b>149</b>
3.1 数据迁移方案介绍	149
3.1.1 准备工作	149
3.1.2 元数据导出	150
3.1.3 数据复制	151
3.1.4 数据恢复	152
3.2 数据迁移到 MRS 前信息收集	152
3.3 数据迁移到 MRS 前网络准备	157

3.4 使用 CDM 服务迁移 Hadoop 数据至 MRS 集群.....	160
3.5 使用 CDM 服务迁移 HBase 数据至 MRS 集群.....	163
3.6 使用 CDM 服务迁移 Hive 数据至 MRS 集群.....	169
3.7 使用 BulkLoad 工具向 HBase 中批量导入数据.....	171
3.8 使用 CDM 服务迁移 MySQL 数据至 MRS Hive.....	178
3.9 使用 CDM 服务迁移 MRS HDFS 数据至 OBS.....	187
<b>4 周边云服务对接.....</b>	<b>191</b>
4.1 MRS 对接 LakeFormation 云服务.....	191
4.1.1 LakeFormation 服务概述.....	191
4.1.2 配置 LakeFormation 数据连接前准备.....	193
4.1.3 创建集群时配置 LakeFormation 数据连接.....	200
4.1.4 通过 Ranger 为 MRS 集群内用户绑定 LakeFormation 角色.....	205
4.2 使用 MRS Spark SQL 访问 DWS 表.....	206
4.3 使用 MRS Hive 表对接 OBS 文件系统.....	210
4.4 MRS Hive 对接 CSS 服务配置说明.....	215
<b>5 生态组件对接.....</b>	<b>219</b>
5.1 使用 DBeaver 访问 Phoenix.....	219
5.2 使用 DBeaver 访问 MRS HetuEngine.....	227
5.3 使用 FineBI 访问 MRS HetuEngine.....	233
5.4 使用 Tableau 访问 MRS HetuEngine.....	239
5.5 使用永洪 BI 访问 MRS HetuEngine.....	241
5.6 Hive 对接外置自建关系型数据库.....	245
5.7 MRS Hive 对接外部 LDAP 配置说明.....	249
5.8 使用 Kafka Eagle 对接 MRS Kafka.....	253
5.9 使用 Jupyter Notebook 对接 MRS Spark.....	257
<b>6 MRS 集群管理.....</b>	<b>267</b>
6.1 MRS 集群阈值类告警配置说明.....	267
6.2 提交 Spark 任务到新增 Task 节点.....	287
6.3 配置 MRS 集群弹性伸缩.....	293

# 1 MRS 最佳实践汇总

本文汇总了MapReduce（MRS）服务的常见应用场景，并为每个场景提供详细的方案描述和操作指南，您可以根据本文查看适合您业务的实践教程文档。

## MRS 最佳实践

表 1-1 MRS 最佳实践

分类	相关文档
数据分析类	<a href="#">使用Spark2x实现车联网车主驾驶行为分析</a>
	<a href="#">使用Hive加载HDFS数据并分析图书评分情况</a>
	<a href="#">使用Hive加载OBS数据并分析企业雇员信息</a>
	<a href="#">通过Flink作业处理OBS数据</a>
	<a href="#">通过Spark Streaming作业消费Kafka数据</a>
	<a href="#">通过Flume采集指定目录日志系统文件至HDFS</a>
	<a href="#">基于Kafka的Word Count数据流统计案例</a>
数据迁移类	<a href="#">使用CDM服务迁移Hadoop数据至MRS集群</a>
	<a href="#">使用CDM服务迁移HBase数据至MRS集群</a>
	<a href="#">使用CDM服务迁移Hive数据至MRS集群</a>
	<a href="#">使用BulkLoad工具向HBase中批量导入数据</a>
	<a href="#">使用CDM服务迁移MySQL数据至MRS Hive</a>
	<a href="#">使用CDM服务迁移MRS HDFS数据至OBS</a>
周边云服务对接类	<a href="#">MRS对接LakeFormation云服务</a>
	<a href="#">使用MRS Spark SQL访问DWS表</a>
	<a href="#">使用MRS Hive表对接OBS文件系统</a>
	<a href="#">MRS Hive对接CSS服务配置说明</a>

分类	相关文档
生态组件对接类	<a href="#">使用DBeaver访问Phoenix</a>
	<a href="#">使用DBeaver访问MRS HetuEngine</a>
	<a href="#">使用FineBI访问MRS HetuEngine</a>
	<a href="#">使用Tableau访问MRS HetuEngine</a>
	<a href="#">使用永洪BI访问MRS HetuEngine</a>
	<a href="#">Hive对接外置自建关系型数据库</a>
	<a href="#">MRS Hive对接外部LDAP配置说明</a>
	<a href="#">使用Kafka Eagle对接MRS Kafka</a>
	<a href="#">使用Jupyter Notebook对接MRS Spark</a>
集群管理类	<a href="#">MRS集群阈值类告警配置说明</a>
	<a href="#">提交Spark任务到新增Task节点</a>
	<a href="#">配置MRS集群弹性伸缩</a>
组件应用开发规范类	<a href="#">ClickHouse应用开发规范</a>
	<a href="#">Doris应用开发规范</a>
	<a href="#">Flink应用开发规范</a>
	<a href="#">Hudi应用开发规范</a>

## 开发者社区精选最佳实践

### 须知

以下所有第三方教程均来自于[华为云社区](#)，由于云服务产品持续更新与迭代，相关社区教程中的内容可能存在时效性，不一定与产品最新版本能力完全保持一致，相关内容仅供学习和参考。

表 1-2 开发者社区精选最佳实践

分类	相关文档
热门组件介绍	<a href="#">MRS StarRocks，新一代极速全场景数据仓库</a>
	<a href="#">Hudi：新一代流式数据湖平台</a>
	<a href="#">高性能利器-MRS ClickHouse重磅推出</a>
	<a href="#">基于MRS-ClickHouse构建用户画像系统方案介绍</a>

分类	相关文档
	<a href="#">基于MRS-Hudi构建数据湖的典型应用场景介绍</a>
大数据分析实践	<a href="#">实时数据湖表存储设计方法（基于Hudi表）</a>
	<a href="#">如何基于MRS-Hudi实现拉链表</a>
	<a href="#">MRS HDFS细粒度锁优化实践</a>
	<a href="#">ClickHouse存算分离在华为云实践</a>
	<a href="#">MRS HBase MTTR优化实践</a>
二次开发类	<a href="#">通过API-Explorer优化MRS服务API开发流程</a>
	<a href="#">使用华为SDK开发MRS API应用初实践</a>
	<a href="#">Kerberos认证集群Flink读写HBase</a>
	<a href="#">HBase BulkLoad导入数据的代码实现</a>
	<a href="#">从Python编译到运行PySpark样例</a>
生态对接类	<a href="#">DBeaver对接MRS Hive</a>
	<a href="#">集群外Druid对接MRS Kerberos集群</a>
	<a href="#">分布式调度平台Airflow在华为云MRS中的实践</a>
	<a href="#">基于云服务MRS构建DolphinScheduler2调度系统</a>

# 2 数据分析

## 2.1 使用 Spark2x 实现车联网车主驾驶行为分析

### 应用场景

本实践基于华为云MapReduce服务所编写，用于指导您了解MRS的基本功能，利用MRS服务的Spark2x组件，对车主的驾驶行为进行分析统计，得到用户驾驶行为的分析结果。

原始数据为车主的驾驶行为信息，包括车主在日常的驾驶行为中，是否急加速、急减速、空挡滑行、超速、疲劳驾驶等信息，通过Spark2x组件的强大的分析能力，分析统计指定时间段内，车主急加速、急减速、空挡滑行、超速、疲劳驾驶等违法行为的次數。

#### 📖 说明

本实践相关样例数据及程序以MRS 3.1.0版本为例，建议按照指定版本创建集群。

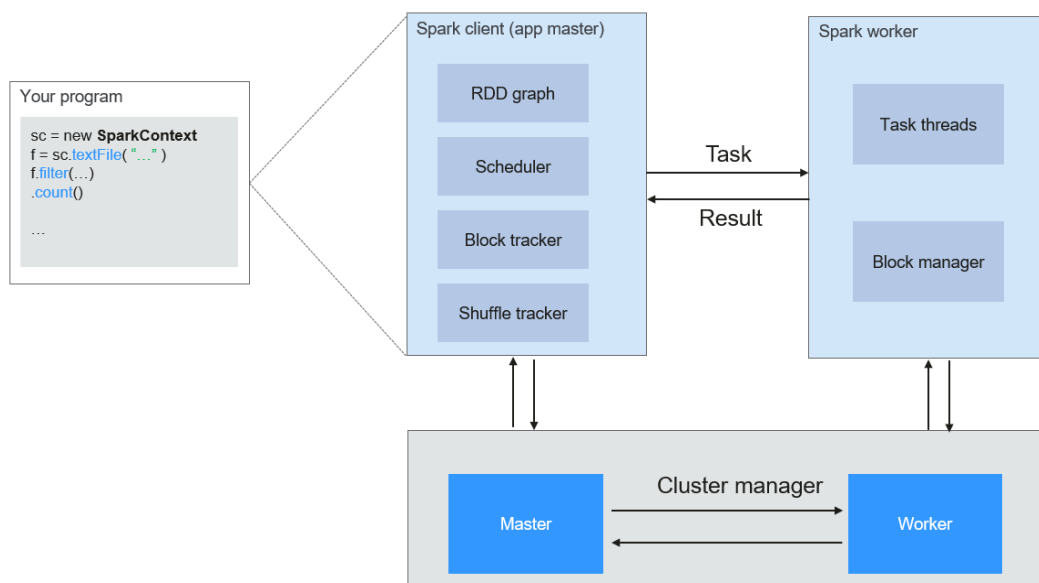
### 方案架构

Spark的应用运行架构如[图2-1](#)所示，运行流程如下所示：

1. 应用程序（Application）是作为一个进程的集合运行在集群上的，由Driver进行协调。
2. 在运行一个应用时，Driver会去连接集群管理器（Standalone、Mesos、YARN）申请运行Executor资源，并启动ExecutorBackend。然后由集群管理器在不同的应用之间调度资源。Driver同时会启动应用程序DAG调度、Stage划分、Task生成。
3. 然后Spark会把应用的代码（传递给SparkContext的JAR或者Python定义的代码）发送到Executor上。
4. 所有的Task执行完成后，用户的应用程序运行结束。



图 2-1 Spark 应用运行架构



## 操作流程

本实践基本操作流程如下所示：

1. **步骤1：创建MRS集群：** 创建一个分析集群，版本号为MRS 3.1.0，不启用Kerberos认证。
2. **步骤2：准备样例程序和数据：** 创建OBS并行文件系统，并将Spark2x样例程序和样例数据上传到OBS并行文件系统中。
3. **步骤3：创建作业：** 在MRS管理控制台创建SparkSubmit作业并运行。
4. **步骤4：查看作业执行结果：** 作业执行完成后在OBS路径中获取日志文件并查看执行结果。

### 步骤 1：创建 MRS 集群

**步骤1** 进入[购买MRS集群页面](#)。

**步骤2** 选择“自定义购买”。

参见[表2-1](#)配置集群信息。

表 2-1 基础配置和集群配置

参数名称	参数说明	取值样例
计费模式	选择待创建的MRS集群的计费模式。	按需计费
区域	选择区域。 不同区域的云服务产品之间内网互不相通。请就近选择靠近您业务的区域，可减少网络时延，提高访问速度。	选择“华北-北京四” <b>说明</b> 本指导以“华北-北京四”为例进行介绍，如果您需要选择其他区域进行操作，请确保所有操作均在同一区域进行。

参数名称	参数说明	取值样例
集群名称	待创建的MRS集群名称。	mrs_demo
集群类型	待创建的MRS集群类型。	选择“分析集群”，用来做离线数据分析。
版本类型	待创建的MRS集群版本类型。	选择“普通版”。
集群版本	待创建的MRS集群版本。	选择“MRS 3.1.0”。 <b>说明</b> 本实践仅适用于MRS 3.1.0版本。
组件选择	选择待创建的MRS集群配套的组件。	勾选所有组件。
元数据	是否使用外部数据源存储元数据。	选择“本地元数据”。

### 步骤3 配置网络配置和节点配置。

表 2-2 网络配置和节点配置

参数名称	参数含义	取值样例
可用区	选择集群工作区域下关联的可用区。	可用区2
虚拟私有云	选择需要创建集群的VPC，单击“查看虚拟私有云”进入VPC服务查看已创建的VPC名称和ID。如果没有VPC，需要创建一个新的VPC。	xxx
子网	选择需要创建集群的子网，可进入VPC服务查看VPC下已创建的子网名称和ID。如果VPC下未创建子网，请单击“创建子网”进行创建。	xxx
安全组	安全组是一组对弹性云服务器的访问规则的集合，为同一个VPC内具有相同安全保护需求并相互信任的弹性云服务器提供访问策略。	自动创建
弹性公网IP	通过将弹性公网IP与MRS集群绑定，实现使用弹性公网IP访问Manager的目的。	绑定一个弹性公网IP。
集群节点	配置集群节点信息。	保持默认值。

### 步骤4 参考表2-3配置以下登录凭证信息。

表 2-3 登录凭证

参数名称	参数说明	取值样例
Kerberos认证	登录Manager管理页面时是否启用Kerberos认证。	关闭
用户名	Manager管理员用户，目前默认为admin用户。	admin
密码	配置Manager管理员用户的密码。	xxx
确认密码	再次输入Manager管理员用户的密码。	xxx
登录方式	登录集群ECS节点的方式。	选择“密码”。
用户名	用于登录ECS的用户，目前默认为root用户。	root
密码	配置登录ECS的用户密码。	xxx
确认密码	再次输入登录ECS的用户密码。	xxx

**步骤5** 高级配置不开启，企业项目选择“default”。

**步骤6** 勾选通信安全授权后，单击“立即购买”，进入任务提交成功页面。

**步骤7** 单击“返回集群列表”，可以查看到集群创建的状态。

集群创建需要时间，所创集群的初始状态为“启动中”，创建成功后状态更新为“运行中”，请您耐心等待。

----结束

## 步骤 2: 准备样例程序和数据

**步骤1** 创建OBS并行文件系统，用于存放Spark样例程序、样例数据、作业执行结果和日志。

1. 登录华为云管理控制台。
2. 在“服务列表”中，选择“存储 > 对象存储服务”。
3. 单击“并行文件系统 > 创建并行文件系统”，创建一个名称为“obs-demo-analysis-hwt4”的文件系统。策略等参数保持默认值。

**步骤2** 单击文件系统名称。选择左侧导航栏“文件”，在“文件”页签下单击“新建文件夹”，分别新建“program”、“input”文件夹，如图2-2所示。

图 2-2 新建文件夹



**步骤3** 从 [https://mrs-obs-cn-north-4.obs.cn-north-4.myhuaweicloud.com/mrs-demon-samples/demon/driver\\_behavior.jar](https://mrs-obs-cn-north-4.obs.cn-north-4.myhuaweicloud.com/mrs-demon-samples/demon/driver_behavior.jar) 路径下载样例程序driver\_behavior.jar至本地。

**步骤4** 进入“program”文件夹，单击“上传文件”，选择本地存放的driver\_behavior.jar样例程序。

**步骤5** 单击“上传”，上传样例程序到OBS并行文件系统。

**步骤6** 从<https://mrs-obs-cn-north-4.obs.cn-north-4.myhuaweicloud.com/mrs-demon-samples/demon/detail-records.zip>获取Spark样例数据到本地。

**步骤7** 将下载的“detail-records.zip”解压，获取图2-3所示的样例数据。

图 2-3 样例数据

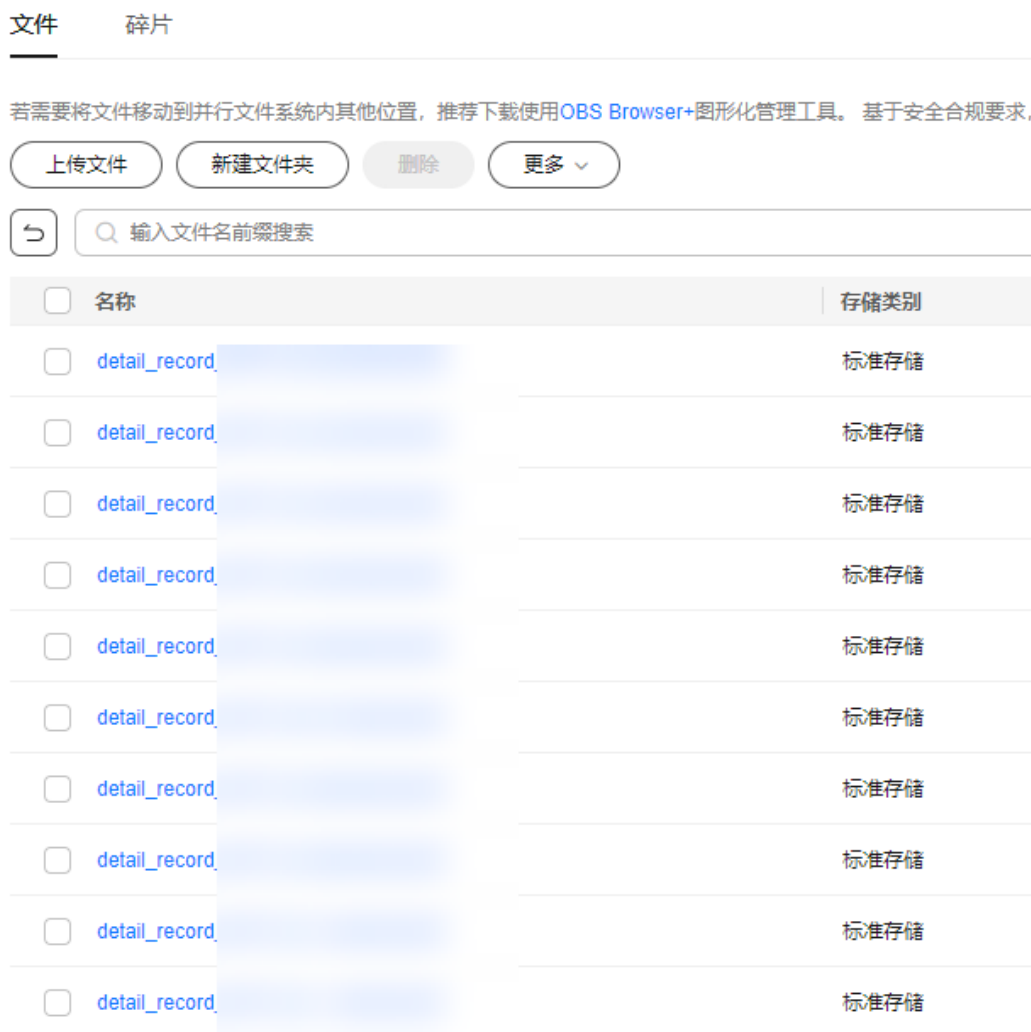
detail_record_2017_01_02_08_00_00	3,056 KB
detail_record_2017_01_03_08_00_00	2,955 KB
detail_record_2017_01_04_08_00_00	4,291 KB
detail_record_2017_01_05_08_00_00	2,324 KB
detail_record_2017_01_06_08_00_00	3,088 KB
detail_record_2017_01_07_08_00_00	2,739 KB
detail_record_2017_01_08_08_00_00	2,797 KB
detail_record_2017_01_09_08_00_00	3,383 KB
detail_record_2017_01_10_08_00_00	3,253 KB
detail_record_2017_01_11_08_00_00	3,497 KB

**步骤8** 进入“input”文件夹，单击“上传文件”，选择本地存放的Spark样例数据。

**步骤9** 单击“上传”，上传样例数据到OBS并行文件系统。

上传**步骤7**中解压后的数据至“input”文件夹下。

图 2-4 上传样例数据



---结束

### 步骤 3: 创建作业

**步骤1** 在MRS控制台左侧导航栏选择“现有集群”，单击名称为“mrs\_demo”的集群。

**步骤2** 在集群信息页面选择“作业管理”页签，单击“添加”，进入添加作业页面。

**步骤3** 完成作业参数配置。

表 2-4 配置作业信息

参数名称	参数说明	取值样例
作业类型	选择待创建的作业类型。	选择“SparkSubmit”。
作业名称	自定义作业名称。	输入 “driver_behavior_task”。

参数名称	参数说明	取值样例
执行程序路径	待执行程序包的存放路径。	单击“OBS”，选择 <b>步骤2：准备样例程序和数据</b> 中上传名称为driver_behavior.jar的jar包。
运行程序参数	为本次执行的作业配置相关优化参数，用于优化资源使用效率，提升作业的执行性能。	参数选择“--class”，值输入“com.huawei.bigdata.spark.examples.DriverBehavior”。
执行程序参数	<p>输入“<b>访问OBS的AK信息 访问OBS的SK信息 1 输入路径 输出路径</b>”。</p> <ul style="list-style-type: none"><li>AK/SK请参考<b>说明</b>方式获取。</li><li>1为固定输入，用于指定作业执行时调用的程序函数。</li><li>输入路径可通过单击“OBS”进行选择输入路径。</li><li>输出路径请手动输入一个不存在的目录，例如obs://obs-demo-analysis-hwt4/output/。</li></ul> <p><b>说明</b> 访问OBS的AK/SK信息可通过如下方式获取：</p> <ol style="list-style-type: none"><li>登录华为云管理控制台。</li><li>单击右上角的用户名，然后选择“我的凭证”。</li><li>系统跳转至“我的凭证”页面，单击“访问密钥”。</li><li>单击“新增访问密钥”申请新密钥，按照提示输入密码与验证码之后，浏览器自动下载一个“credentials.csv”文件，文件为csv格式，以英文逗号分隔，中间的为AK，最后一个为SK。</li></ol>	<b>AK信息 SK信息 1</b> obs://obs-demo-analysis-hwt4/input/ obs://obs-demo-analysis-hwt4/output/
服务配置参数	用于为本次执行的作业修改服务配置参数。	保持默认不配置。

图 2-5 添加作业

作业类型: SparkSubmit

作业名称: driver\_behavior\_task

执行程序路径: obs:// /program/driver\_behavior.jar

运行程序参数: --class com.huawei.bigdata.spark.examples.Driver

执行程序参数: AK SK obs://obs-demo-analysis-hwt4/input obs://obs-demo-analysis-hwt4/output |

服务配置参数: 参数 值

命令参考: spark-submit --class com.huawei.bigdata.spark.examples.DriverBehavior --master yarn-cluster obs:// /program/driver\_behavior.jar 1 obs://obs-demo-analysis-hwt4/input obs://obs-demo-analysis-hwt4/output

步骤4 单击“确定”，开始提交作业，执行程序。

----结束

### 步骤 4: 查看作业执行结果

步骤1 进入“作业管理”页面，查看作业执行状态。

步骤2 等待1~2分钟，登录OBS控制台，进入“obs-demo-analysis-hwt4”文件系统的“output”目录中，查看执行结果，在生成的csv文件所在行的“操作”列单击“下载”按钮将该文件下载到本地。

步骤3 在本地将下载后的csv文件使用Excel文本打开，按照样例程序中定义的字段为每列数据进行分类，得到如下图所示作业执行结果。

图 2-6 执行结果

driverID	车牌号	急加速次数	急刹车次数	空挡滑行次数	空挡滑行时间总和	超速次数	超速时间总和	疲劳驾驶次数	停车丢油门次数	漏油次数
panxian1000005	华AX542C	395	434	330	2930	3531	33946	4307	417	441
zouan1000007	华A58M8S	360	385	315	2997	3181	31248	3594	389	385
zengpeng1000000	华AZQ110	340	344	272	2894	2763	25479	3274	284	337
xie Xiao1000001	华AEE132	264	261	248	2525	2324	23434	2720	314	253
duxui1000009	华AT75H8	238	284	247	2632	2301	22338	2814	264	248
hanhui1000002	华AZI419	401	444	327	2844	3349	31813	3997	433	371
shenxian1000004	华ADJ750	374	356	297	2810	3126	31494	3767	383	366
likun1000003	华AVM93E	341	354	291	3043	3044	28728	3552	347	376
haowe1000008	华A709CE	321	314	255	2659	2639	25522	3204	312	318
xiezhi1000006	华A6CU11	255	310	254	2074	2535	23942	2931	312	279

----结束

## 2.2 使用 Hive 加载 HDFS 数据并分析图书评分情况

### 应用场景

MRS离线处理集群，可对海量数据进行分析 and 处理，形成结果数据，供下一步数据应用使用。

离线处理对处理时间要求不高，但是所处理数据量较大，占用计算存储资源较多，通常通过Hive/SparkSQL引擎或者MapReduce/Spark2x实现。

本实践基于华为云MapReduce服务，用于指导您创建MRS集群后，使用Hive对原始数据进行导入、分析等操作，展示了如何构建弹性、低成本的离线大数据分析。

### 方案架构

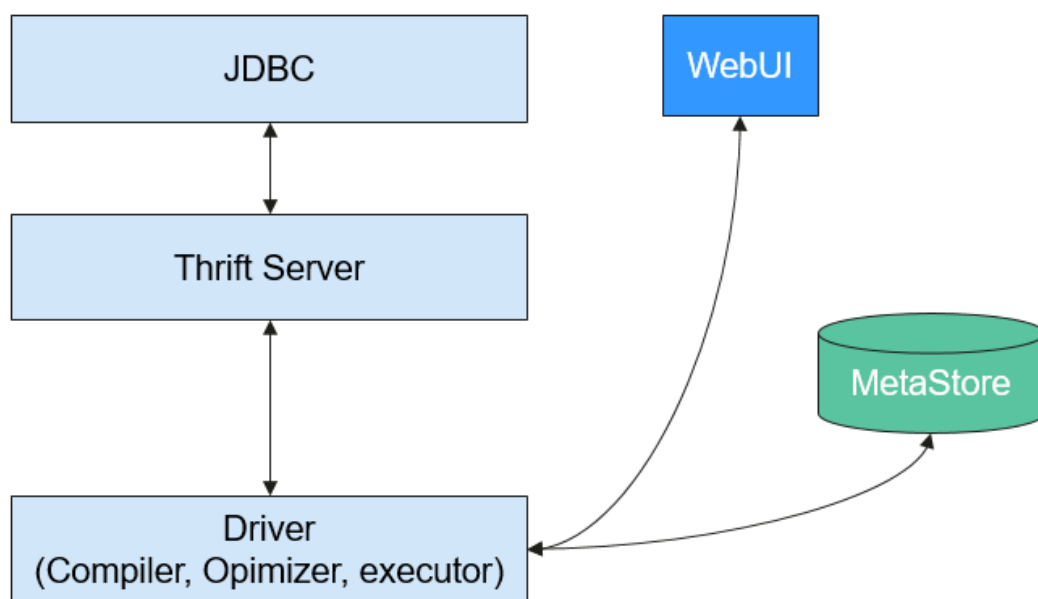
Hive是建立在Hadoop上的数据仓库框架，提供大数据平台批处理计算能力，能够对结构化/半结构化数据进行批量分析汇总完成数据计算。提供类似SQL的Hive Query Language语言操作结构化数据，其基本原理是将HQL语言自动转换成MapReduce任务，从而完成对Hadoop集群中存储的海量数据进行查询和分析。

Hive主要特点如下：

- 海量结构化数据分析汇总。
- 将复杂的MapReduce编写任务简化为SQL语句。
- 灵活的数据存储格式，支持JSON、CSV、TEXTFILE、RCFILE、SEQUENCEFILE、ORC等存储格式。

Hive作为一个基于HDFS和MapReduce架构的数据仓库，其主要能力是通过对HQL（Hive Query Language）编译和解析，生成并执行相应的MapReduce任务或者HDFS操作。

图 2-7 Hive 结构





- **Metastore**: 对表, 列和Partition等的元数据进行读写及更新操作, 其下层为关系型数据库。
- **Driver**: 管理HQL执行的生命周期并贯穿Hive任务整个执行期间。
- **Compiler**: 编译HQL并将其转化为一系列相互依赖的Map/Reduce任务。
- **Optimizer**: 优化器, 分为逻辑优化器和物理优化器, 分别对HQL生成的执行计划和MapReduce任务进行优化。
- **Executor**: 按照任务的依赖关系分别执行Map/Reduce任务。
- **ThriftServer**: 提供thrift接口, 作为JDBC的服务端, 并将Hive和其他应用程序集成起来。
- **Clients**: 包含WebUI和JDBC接口, 为用户访问提供接口。

## 操作流程

本实践以某图书网站后台用户的点评数据为原始数据, 导入Hive表后通过SQL命令筛选出最受欢迎的图书。

基本操作流程如下所示:

1. **步骤1: 创建MRS离线查询集群**
2. **步骤2: 将本地数据导入到HDFS中**
3. **步骤3: 创建Hive表**
4. **步骤4: 将原始数据导入Hive并进行分析**

### 步骤 1: 创建 MRS 离线查询集群

**步骤1** 进入[购买MRS集群页面](#)。

**步骤2** 选择“快速购买”, 填写软件配置参数, 以“按需计费”为例。

表 2-5 软件配置 (以下参数仅供参考, 可根据实际情况调整)

参数名称	参数说明	取值样例
计费模式	选择待创建的MRS集群的计费模式。	按需计费
区域	选择区域。 不同区域的云服务产品之间内网互不相通。请就近选择靠近您业务的区域, 可减少网络时延, 提高访问速度。	华北-北京四
集群名称	待创建的MRS集群名称。	MRS_demo
集群类型	待创建的MRS集群类型。	选择“自定义”
版本类型	待创建的MRS集群版本类型。	普通版
集群版本	待创建的MRS集群版本。	MRS 3.1.0
组件选择	选择待创建的MRS集群配套的组件。	Hadoop分析集群

参数名称	参数说明	取值样例
可用区	选择集群工作区域下关联的可用区。	可用区1
虚拟私有云	选择需要创建集群的VPC，单击“查看虚拟私有云”进入VPC服务查看已创建的VPC名称和ID。如果没有VPC，需要创建一个新的VPC。	vpc-01
子网	选择需要创建集群的子网，可进入VPC服务查看VPC下已创建的子网名称和ID。如果VPC下未创建子网，请单击“创建子网”进行创建。	subnet-01
集群节点	配置集群节点信息。	保持默认值。
Kerberos认证	登录Manager管理页面时是否启用Kerberos认证。	不开启
用户名	Manager管理员用户，目前默认为admin用户。	admin/root
密码	配置Manager管理员用户的密码。	设置密码登录集群管理页面及ECS节点用户的密码，例如：Test!@12345。
确认密码	再次输入Manager管理员用户的密码。	再次输入设置用户密码
企业项目	选择集群所属的企业项目。	default
通信安全授权	若不开启通信安全授权，MRS将无法创建集群。	勾选授权

**步骤3** 单击“立即购买”，等待MRS集群创建成功。

----结束

## 步骤 2：将本地数据导入到 HDFS 中

**步骤1** 在本地已获取某图书网站后台图书点评记录的原始数据文件“book\_score.txt”，例如内容如下。

字段信息依次为：用户ID、图书ID、图书评分、备注信息。

例如部分数据节选如下：

```
202001,242,3,Good!  
202002,302,3,Test.  
202003,377,1,Bad!  
220204,51,2,Bad!  
202005,346,1,aaa  
202006,474,4,None  
202007,265,2,Bad!  
202008,465,5,Good!  
202009,451,3,Bad!
```

```
202010,86,3,Bad!
202011,257,2,Bad!
202012,465,4,Good!
202013,465,4,Good!
202014,465,4,Good!
202015,302,5,Good!
202016,302,3,Good!
...
```

**步骤2** 登录对象存储服务OBS控制台，单击“并行文件系统 > 创建并行文件系统”，填写以下参数，单击“立即创建”。

表 2-6 并行文件系统参数

参数名称	参数说明	取值样例
区域	桶所属区域。	华北-北京四
数据冗余存储策略	<ul style="list-style-type: none"> <li>多AZ存储：数据冗余存储至多个可用区（AZ），可靠性更高。</li> <li>单AZ存储：数据仅存储在单个可用区（AZ），成本更低。</li> </ul>	单AZ存储
文件系统名称	文件系统的名称，需全局唯一，不能与已有的任何文件系统名称重复。	mrs-hive
策略	文件系统的读写权限控制。	私有
归档数据直读	通过归档数据直读，您可以直接下载存储类别为归档存储的对象，而无需提前恢复。	关闭
企业项目	将桶加入到企业项目中统一管理。	default
标签	可选。标签用于标识OBS中的桶，以此达到对OBS中的桶进行分类的目的。	-

等待文件系统创建好，单击文件系统名称，选择“文件 > 上传文件”，将数据文件上传至OBS并行文件系统内。

**步骤3** 切换回MRS控制台，单击创建好的MRS集群名称，进入“概览”，单击“IAM用户同步”所在行的“同步”，等待约5分钟同步完成。

图 2-8 同步 IAM 用户



**步骤4** 将数据文件上传HDFS。

1. 在“文件管理”页签，选择“HDFS文件列表”，进入数据存储目录，如“/tmp/test”。  
“/tmp/test”目录仅为示例，可以是界面上的任何目录，也可以通过“新建”创建新的文件夹。
2. 单击“导入数据”。
  - OBS路径：选择上面创建好的OBS并行文件系统名，找到“book\_score.txt”文件，勾选“我确认所选脚本安全，了解可能存在的风险，并接受对集群可能造成的异常或影响。”，单击“确定”。
  - HDFS路径：选择“/tmp/test”，单击“确定”。

图 2-9 从 OBS 导入数据到 HDFS



3. 单击“确定”，等待数据导入成功，此时数据文件已上传至MRS集群的HDFS文件系统内。

---结束

**步骤 3: 创建 Hive 表**

**步骤1** 下载并安装集群全量客户端，例如在主Master节点上安装，客户端安装目录为“/opt/client”，相关操作可参考[安装客户端](#)。

也可直接使用Master节点中自带的集群客户端，安装目录为“/opt/Bigdata/client”。

**步骤2** 为主Master节点绑定一个弹性IP并在安全组中放通22端口，具体请参考[配置集群安全组规则](#)，然后使用root用户登录主Master节点，进入客户端所在目录并加载变量。

```
cd /opt/client
```

```
source bigdata_env
```

**步骤3** 执行**beeline -n 'hdfs'**命令进入Hive Beeline命令行界面。

执行以下命令创建一个与原始数据字段匹配的Hive表：

```
create table bookscore (userid int,bookid int,score int,remarks string) row  
format delimited fields terminated by ','stored as textfile;
```

**步骤4** 查看表是否创建成功：

```
show tables;
```

```
+-----+  
| tab_name |  
+-----+  
| bookscore |  
+-----+
```

----结束

## 步骤 4：将原始数据导入 Hive 并进行分析

**步骤1** 继续在Hive Beeline命令行中执行以下命令，将已导入HDFS的原始数据导入Hive表中。

```
load data inpath '/tmp/test/book_score.txt' into table bookscore;
```

**步骤2** 数据导入完成后，执行如下命令，查看Hive表内容。

```
select * from bookscore;
```

```
+-----+-----+-----+-----+  
| bookscore.userid | bookscore.bookid | bookscore.score | bookscore.remarks |  
+-----+-----+-----+-----+  
| 202001           | 242              | 3               | Good!              |  
| 202002           | 302              | 3               | Test.              |  
| 202003           | 377              | 1               | Bad!               |  
| 220204           | 51               | 2               | Bad!               |  
| 202005           | 346              | 1               | aaa                |  
| 202006           | 474              | 4               | None               |  
| 202007           | 265              | 2               | Bad!               |  
| 202008           | 465              | 5               | Good!              |  
| 202009           | 451              | 3               | Bad!               |  
| 202010           | 86               | 3               | Bad!               |  
| 202011           | 257              | 2               | Bad!               |  
| 202012           | 465              | 4               | Good!              |  
| 202013           | 465              | 4               | Good!              |  
| 202014           | 465              | 4               | Good!              |  
| 202015           | 302              | 5               | Good!              |  
| 202016           | 302              | 3               | Good!              |  
| ...              | ...              | ...             | ...                |
```

执行以下命令统计表行数：

```
select count(*) from bookscore;
```

```
+-----+  
| _c0 |  
+-----+  
| 32 |  
+-----+
```

**步骤3** 执行以下命令，等待MapReduce任务完成后，筛选原始数据中累计评分最高的图书top3。

```
select bookid,sum(score) as summarize from bookscore group by bookid order  
by summarize desc limit 3;
```

例如最终显示内容如下：

```
...  
INFO : 2021-10-14 19:53:42,427 Stage-2 map = 0%, reduce = 0%  
INFO : 2021-10-14 19:53:49,572 Stage-2 map = 100%, reduce = 0%, Cumulative CPU 2.15 sec  
INFO : 2021-10-14 19:53:56,713 Stage-2 map = 100%, reduce = 100%, Cumulative CPU 4.19 sec  
INFO : MapReduce Total cumulative CPU time: 4 seconds 190 msec  
INFO : Ended Job = job_1634197207682_0025  
INFO : MapReduce Jobs Launched:
```

```
INFO : Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 4.24 sec HDFS Read: 7872 HDFS Write: 322
SUCCESS
INFO : Stage-Stage-2: Map: 1 Reduce: 1 Cumulative CPU: 4.19 sec HDFS Read: 5965 HDFS Write: 143
SUCCESS
INFO : Total MapReduce CPU Time Spent: 8 seconds 430 msec
INFO : Completed executing
command(queryId=omm_20211014195310_cf669633-5b58-4bd5-9837-73286ea83409); Time taken: 47.388
seconds
INFO : OK
INFO : Concurrency mode is disabled, not creating a lock manager
+-----+-----+
| bookid | summarize |
+-----+-----+
| 465    | 170      |
| 302    | 110      |
| 474    | 88       |
+-----+-----+
3 rows selected (47.469 seconds)
```

以上内容表示，ID为456、302、474的3本书籍，为累计评分最高的Top3图书。

----结束

## 2.3 使用 Hive 加载 OBS 数据并分析企业雇员信息

### 应用场景

MRS Hadoop分析集群，提供Hive、Spark离线大规模分布式数据存储和计算，进行海量数据分析与查询。

本实践基于华为云MapReduce服务，用于指导您创建MRS集群后，使用Hive对OBS中存储的原始数据进行导入、分析等操作，展示了如何构建弹性、低成本的存算分离大数据分析。

本实践中，雇员信息的原始数据包含以下两张表：

表 2-7 表 1 雇员信息数据

编号	姓名	支付薪水币种	薪水金额	纳税税种	工作地	入职时间
1	Wang	R	8000.01	personal income tax&0.05	China:Shenzhen	2014
3	Tom	D	12000.02	personal income tax&0.09	America:NewYork	2014
4	Jack	D	24000.03	personal income tax&0.09	America:Manhattan	2015
6	Linda	D	36000.04	personal income tax&0.09	America:NewYork	2014

编号	姓名	支付薪水币种	薪水金额	纳税税种	工作地	入职时间
8	Zhang	R	9000.05	personal income tax&0.05	China:Shanghai	2014

表 2-8 雇员联络信息数据

编号	电话	邮箱
1	135 XXXX XXXX	xxxx@example.com
3	159 XXXX XXXX	xxxxx@example.com.cn
4	186 XXXX XXXX	xxxx@example.org
6	189 XXXX XXXX	xxxx@example.cn
8	134 XXXX XXXX	xxxx@example.cn

通过数据应用，进行以下分析：

- 查看薪水支付币种为美元的雇员联系方式。
- 查询入职时间为2014年的雇员编号、姓名等字段，并将查询结果加载到新表中。
- 统计雇员信息共有多少条记录。
- 查询使用以“cn”结尾的邮箱的员工信息。

## 方案架构

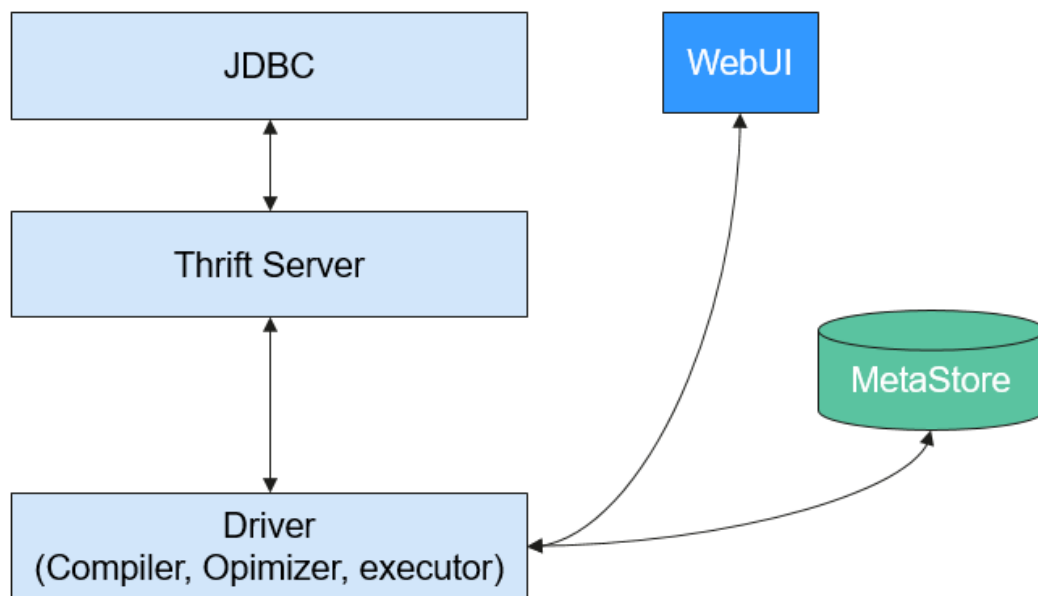
Hive是建立在Hadoop上的数据仓库框架，提供大数据平台批处理计算能力，能够对结构化/半结构化数据进行批量分析汇总完成数据计算。提供类似SQL的Hive Query Language语言操作结构化数据，其基本原理是将HQL语言自动转换成MapReduce任务，从而完成对Hadoop集群中存储的海量数据进行查询和分析。

Hive主要特点如下：

- 海量结构化数据分析汇总。
- 将复杂的MapReduce编写任务简化为SQL语句。
- 灵活的数据存储格式，支持JSON、CSV、TEXTFILE、RCFILE、SEQUENCEFILE、ORC等存储格式。

Hive作为一个基于HDFS和MapReduce架构的数据仓库，其主要能力是通过对HQL（Hive Query Language）编译和解析，生成并执行相应的MapReduce任务或者HDFS操作。

图 2-10 Hive 结构



- **Metastore:** 对表，列和Partition等的元数据进行读写及更新操作，其下层为关系型数据库。
- **Driver:** 管理HQL执行的生命周期并贯穿Hive任务整个执行期间。
- **Compiler:** 编译HQL并将其转化为一系列相互依赖的Map/Reduce任务。
- **Optimizer:** 优化器，分为逻辑优化器和物理优化器，分别对HQL生成的执行计划和MapReduce任务进行优化。
- **Executor:** 按照任务的依赖关系分别执行Map/Reduce任务。
- **ThriftServer:** 提供thrift接口，作为JDBC的服务端，并将Hive和其他应用程序集成起来。
- **Clients:** 包含WebUI和JDBC接口，为用户访问提供接口。

## 操作流程

本实践以用户开发一个Hive数据分析应用为例，通过客户端连接Hive后，执行HQL语句访问OBS中的Hive数据。进行企业雇员信息的管理、查询。如果需要基于MRS服务提供的样例代码工程开发构建应用，您可以参考[Hive应用开发简介](#)。

基本操作流程如下所示：

1. **步骤1: 创建MRS离线查询集群**
2. **步骤2: 创建OBS委托并绑定至MRS集群**
3. **步骤3: 创建Hive表并加载OBS中数据**
4. **步骤4: 基于HQL对数据进行分析**

### 步骤 1: 创建 MRS 离线查询集群

**步骤1** 进入[购买MRS集群页面](#)。

**步骤2** 选择“快速购买”，填写软件配置参数。



表 2-9 软件配置（以下参数仅供参考，可根据实际情况调整）

参数名称	参数说明	取值样例
计费模式	选择待创建的MRS集群的计费模式。	按需计费
区域	选择区域。 不同区域的云服务产品之间内网互不相通。请就近选择靠近您业务的区域，可减少网络时延，提高访问速度。	华北-北京四
集群名称	待创建的MRS集群名称。	MRS_demo
集群类型	待创建的MRS集群类型。	选择“自定义”
版本类型	待创建的MRS集群版本类型。	普通版
集群版本	待创建的MRS集群版本。	MRS 3.1.0
组件选择	选择待创建的MRS集群配套的组件。	Hadoop分析集群
可用区	选择集群工作区域下关联的可用区。	可用区1
虚拟私有云	选择需要创建集群的VPC，单击“查看虚拟私有云”进入VPC服务查看已创建的VPC名称和ID。如果没有VPC，需要创建一个新的VPC。	vpc-01
子网	选择需要创建集群的子网，可进入VPC服务查看VPC下已创建的子网名称和ID。如果VPC下未创建子网，请单击“创建子网”进行创建。	subnet-01
集群节点	配置集群节点信息。	保持默认值。
Kerberos认证	登录Manager管理页面时是否启用Kerberos认证。	不开启
用户名	Manager管理员用户，目前默认为admin用户。	admin/root
密码	配置Manager管理员用户的密码。	设置密码登录集群管理页面及ECS节点用户的密码，例如：Test!@12345。
确认密码	再次输入Manager管理员用户的密码。	再次输入设置用户密码
企业项目	选择集群所属的企业项目。	default
通信安全授权	若不开启通信安全授权，MRS将无法创建集群。	勾选授权

**步骤3** 单击“立即购买”，等待MRS集群创建成功。

---结束

## 步骤 2: 创建 OBS 委托并绑定至 MRS 集群

### 📖 说明

- MRS在IAM的委托列表中预置了MRS\_ECS\_DEFAULT\_AGENCY委托，可在创建自定义过程中可以直接选择该委托，该委托拥有对象存储服务的OBSOperateAccess权限和在集群所在区域拥有CESFullAccess（对开启细粒度策略的用户）、CES Administrator和KMS Administrator权限。
- 如需使用自定义委托，请参考如下步骤进行创建委托（创建或修改委托需要用户具有Security Administrator权限）。

**步骤1** 登录华为云管理控制台。

**步骤2** 在服务列表中选择“管理与监管 > 统一身份认证服务 IAM”。

**步骤3** 选择“委托 > 创建委托”。

**步骤4** 设置“委托名称”，“委托类型”选择“云服务”，在“云服务”中选择“弹性云服务器ECS 裸金属服务器BMS”，授权ECS或BMS调用OBS服务。

**步骤5** “持续时间”选择“永久”并单击“完成”。

图 2-11 创建委托

\* 委托名称

\* 委托类型  普通账号  
将账号内资源的操作权限委托给其他华为云账号。  
 云服务  
将账号内资源的操作权限委托给华为云服务。

\* 云服务

\* 持续时间

描述

0/255 ↕

**步骤6** 在创建的委托的“操作”列单击“授权”，搜索“OBS OperateAccess”策略，勾选“OBS OperateAccess”策略。

- 步骤7** 单击“下一步”，选择权限范围方案，默认选择“所有资源”，单击“展开其他方案”，选择“全局服务资源”。
- 步骤8** 单击“确定”，在弹出的提示框中单击“知道了”，开始授权。界面提示“授权成功。”，单击“完成”，委托成功创建。
- 步骤9** 返回MRS控制台，在集群列表中，单击已创建好的MRS集群名称，在集群的“概览”页面中，单击“选择委托”，选择创建好的OBS委托后单击“确定”。
- 结束

### 步骤 3: 创建 Hive 表并加载 OBS 中数据

- 步骤1** 在服务列表中选择“存储 > 对象存储服务 OBS”，登录OBS控制台，单击“并行文件系统 > 创建并行文件系统”，填写以下参数，单击“立即创建”。

表 2-10 并行文件系统参数

参数名称	参数说明	取值样例
区域	设置并行文件系统的区域。	华北-北京四
文件系统名称	设置并行文件系统的名称	hiveobs
数据冗余存储策略	<ul style="list-style-type: none"><li>多AZ存储：数据冗余存储至多个可用区（AZ），可靠性更高。</li><li>单AZ存储：数据仅存储在单个可用区（AZ），成本更低。</li></ul>	单AZ存储
策略	并行文件系统的读写策略。	私有
归档数据直读	通过归档数据直读，您可以直接下载存储类别为归档存储的文件，而无需提前恢复。	关闭
企业项目	将并行文件系统加入到企业项目中统一管理。	default
标签	可选。标签用于标识OBS中的并行文件系统，以此达到对并行文件系统进行分类的目的。	-

- 步骤2** 下载并安装MRS集群客户端，例如在主Master节点上安装，客户端安装目录为“/opt/client”，相关操作可参考[安装客户端](#)。

也可直接使用Master节点中自带的集群客户端，安装目录为“/opt/Bigdata/client”。

- 步骤3** 为主Master节点绑定一个弹性IP并在安全组中放通22端口，然后使用root用户登录主Master节点，进入客户端所在目录并加载变量。

```
cd /opt/client
source bigdata_env
```

**步骤4** 执行**beeline**命令进入Hive Beeline命令行界面。

执行以下命令创建一个与原始数据字段匹配的雇员信息数据表“employees\_info”：

```
create external table if not exists employees_info
(
  id INT,
  name STRING,
  usd_flag STRING,
  salary DOUBLE,
  deductions MAP<STRING, DOUBLE>,
  address STRING,
  entrytime STRING
)
row format delimited fields terminated by ',' map keys terminated by '&'
stored as textfile
location 'obs://hiveobs/employees_info';
```

执行以下命令创建一个与原始数据字段匹配的雇员联系信息数据表“employees\_contact”：

```
create external table if not exists employees_contact
(
  id INT,
  phone STRING,
  email STRING
)
row format delimited fields terminated by ','
stored as textfile
location 'obs://hiveobs/employees_contact';
```

**步骤5** 查看表是否创建成功：

```
show tables;
```

```
+-----+
| tab_name |
+-----+
| employees_contact |
| employees_info |
+-----+
```

**步骤6** 将数据导入OBS对应表目录下。

Hive内部表会默认在指定的存储空间中建立对应文件夹，只要把文件放入，表就可以读取到数据（需要和表结构匹配）。

登录OBS控制台，在已创建的文件系统的“文件”页面，将本地的原始数据分别上传至生成的“employees\_info”、“employees\_contact”文件夹下。

例如原始数据格式如下：

info.txt:

```
1,Wang,R,8000.01,personal income tax&0.05,China:Shenzhen,2014
3,Tom,D,12000.02,personal income tax&0.09,America:NewYork,2014
4,Jack,D,24000.03,personal income tax&0.09,America:Manhattan,2015
6,Linda,D,36000.04,personal income tax&0.09,America:NewYork,2014
8,Zhang,R,9000.05,personal income tax&0.05,China:Shanghai,2014
```

contact.txt:

```
1,135 XXXX XXXX,xxxx@xx.com
3,159 XXXX XXXX,xxxx@xx.com.cn
4,189 XXXX XXXX,xxxx@xx.org
6,189 XXXX XXXX,xxxx@xx.cn
8,134 XXXX XXXX,xxxx@xxx.cn
```

**步骤7** 在Hive Beeline客户端中，执行以下命令，查询源数据是否被正确加载。

**select \* from employees\_info;**

```
+-----+-----+-----+-----+
+-----+-----+-----+-----+
| employees_info.id | employees_info.name | employees_info.usd_flag | employees_info.salary |
employees_info.deductions | employees_info.address | employees_info.entrytime |
+-----+-----+-----+-----+
+-----+-----+-----+-----+
| 1 | Wang | R | 8000.01 | {"personal income tax":0.05} |
China:Shenzhen | 2014 | | |
| 3 | Tom | D | 12000.02 | {"personal income tax":0.09} |
America:NewYork | 2014 | | |
| 4 | Jack | D | 24000.03 | {"personal income tax":0.09} |
America:Manhattan | 2015 | | |
| 6 | Linda | D | 36000.04 | {"personal income tax":0.09} |
America:NewYork | 2014 | | |
| 8 | Zhang | R | 9000.05 | {"personal income tax":0.05} |
China:Shanghai | 2014 | | |
+-----+-----+-----+-----+
+-----+-----+-----+-----+
```

**select \* from employees\_contact;**

```
+-----+-----+-----+-----+
+-----+-----+-----+-----+
| employees_contact.id | employees_contact.phone | employees_contact.email |
+-----+-----+-----+-----+
| 1 | 135 XXXX XXXX | xxxx@xx.com |
| 3 | 159 XXXX XXXX | xxxx@xx.com.cn |
| 4 | 186 XXXX XXXX | xxxx@xx.org |
| 6 | 189 XXXX XXXX | xxxx@xx.cn |
| 8 | 134 XXXX XXXX | xxxx@xxx.cn |
+-----+-----+-----+-----+
```

----结束

## 步骤 4: 基于 HQL 对数据进行分析

在Hive Beeline客户端中，执行HQL语句，对原始数据进行分析。

**步骤1** 查看薪水支付币种为美元的雇员联系方式。

创建新数据表进行数据清洗。

```
create table employees_info_v2 as select id, name, regexp_replace(usr_flag, '\s+', '') as usr_flag, salary, deductions, address, entrytime from employees_info;
```

等待Map任务完成后，执行以下命令

```
select a.* from employees_info_v2 a inner join employees_contact b on a.id = b.id where a.usr_flag='D';
```

```
INFO : MapReduce Jobs Launched:
INFO : Stage-Stage-3: Map: 1 Cumulative CPU: 2.95 sec HDFS Read: 8483 HDFS Write: 317 SUCCESS
INFO : Total MapReduce CPU Time Spent: 2 seconds 950 msec
INFO : Completed executing command(queryId=omm_20211022162303_c26d4f1b-a577-4d6c-919c-6cb96095b24b); Time taken: 26.259 seconds
INFO : OK
INFO : Concurrency mode is disabled, not creating a lock manager
+-----+-----+-----+-----+-----+-----+
| a.id | a.name | a.usr_flag | a.salary | a.deductions | a.address | a.entrytime |
+-----+-----+-----+-----+-----+-----+
| 3 | Tom | D | 12000.02 | {"personal income tax":0.09} | America:NewYork | 2014 |
| 4 | Jack | D | 24000.03 | {"personal income tax":0.09} | America:Manhattan | 2015 |
| 6 | Linda | D | 36000.04 | {"personal income tax":0.09} | America:NewYork | 2014 |
+-----+-----+-----+-----+-----+-----+
3 rows selected (26.439 seconds)
```

**步骤2** 查询入职时间为2014年的雇员编号、姓名等字段，并将查询结果加载进表 employees\_info\_extended中的入职时间为2014的分区中。

创建一个表：

```
create table if not exists employees_info_extended (id int, name string, usr_flag string, salary double, deductions map<string, double>, address string) partitioned by (entrytime string) stored as textfile;
```

执行以下命令，在表中写入数据：

```
insert into employees_info_extended partition(entrytime='2014') select id,name,usr_flag,salary,deductions,address from employees_info_v2 where entrytime = '2014';
```

数据抽取成功后，查询表数据。

```
select * from employees_info_extended;
```

```
+-----+-----+-----+-----+-----+-----+
| employees_info_extended.id | employees_info_extended.name | employees_info_extended.usr_flag |
employees_info_extended.salary | employees_info_extended.deductions |
employees_info_extended.address | employees_info_extended.entrytime |
+-----+-----+-----+-----+-----+-----+
| 1 | Wang | R | 8000.01 | {"personal income tax":0.05} | China:Shenzhen | 2014 |
| 3 | Tom | D | 12000.02 | {"personal income tax":0.09} | America:NewYork | 2014 |
| 6 | Linda | D | 36000.04 | {"personal income tax":0.09} | America:NewYork | 2014 |
| 8 | Zhang | R | 9000.05 | {"personal income tax":0.05} | China:Shanghai | 2014 |
+-----+-----+-----+-----+-----+-----+
```

**步骤3** 统计雇员信息有多少条记录。

```
select count(1) from employees_info_v2;
```

```
+-----+
|_c0 |
+-----+
| 5 |
+-----+
```

**步骤4** 查询使用以“cn”结尾的邮箱的员工信息。

```
select a.*, b.email from employees_info_v2 a inner join employees_contact b
on a.id = b.id where b.email rlike '.*cn$';
```

```
+-----+-----+-----+-----+-----+-----+-----+-----+
| a.id | a.name | a.usd_flag | a.salary | a.deductions | a.address | a.entrytime |
b.email |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 3 | Tom | D | 12000.02 | {"personal income tax":0.09} | America:NewYork | 2014 |
xxxx@xx.com.cn |
| 6 | Linda | D | 36000.04 | {"personal income tax":0.09} | America:NewYork | 2014 |
xxxx@xx.cn |
| 8 | Zhang | R | 9000.05 | {"personal income tax":0.05} | China:Shanghai | 2014 |
xxxx@xxx.cn |
+-----+-----+-----+-----+-----+-----+-----+-----+
+-----+-----+-----+-----+-----+-----+-----+-----+
```

----结束

## 2.4 通过 Flink 作业处理 OBS 数据

### 应用场景

MRS支持在大数据存储容量大、计算资源需要弹性扩展的场景下，用户将数据存储在 OBS服务中，使用MRS集群仅做数据计算处理的存算分离模式。

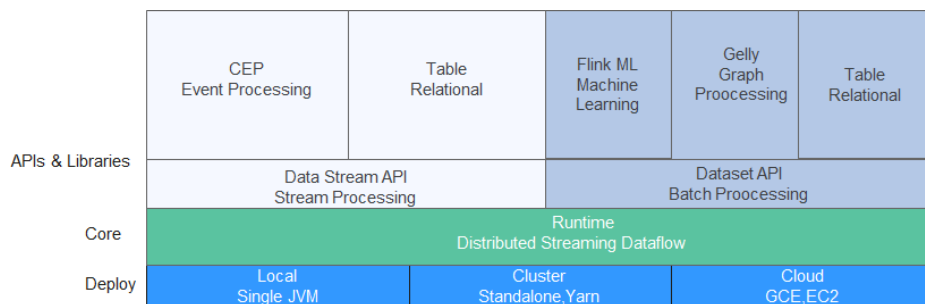
本文将向您介绍如何在MRS集群中运行Flink作业来处理OBS中存储的数据。

### 方案架构

Flink是一个批处理和流处理结合的统一计算框架，其核心是一个提供了数据分发以及并行化计算的流数据处理引擎。它的最大亮点是流处理，是业界最顶级的开源流处理引擎。

Flink最适合的应用场景是低时延的数据处理（Data Processing）场景：高并发 pipeline处理数据，时延毫秒级，且兼具可靠性。

图 2-12 Flink 架构



在本示例中，使用MRS集群内置的Flink WordCount作业程序，来分析OBS文件系统中保存的源数据，以统计源数据中的单词出现次数。

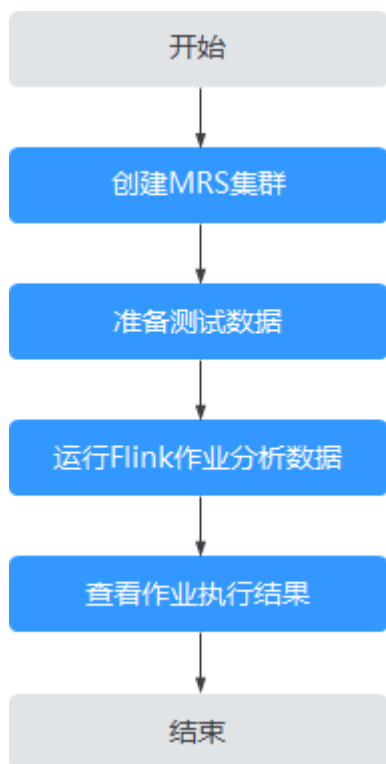
当然您也可以获取[MRS服务样例代码工程](#)，参考[Flink开发指南](#)开发其他Flink流作业程序。

## 操作流程

本实践操作流程如下所示：

1. **步骤1：创建MRS集群**：创建一个包含有Flink组件的MRS集群。
2. **步骤2：准备测试数据**：准备用于进行分析的源数据及程序，并上传至OBS。
3. **步骤3：创建并运行Flink作业**：在MRS集群中创建Flink作业并运行。
4. **步骤4：查看作业执行结果**：查看数据分析结果。

图 2-13 操作流程



### 步骤 1：创建 MRS 集群

创建并购买一个包含有Flink组件的MRS集群，详情请参见[购买自定义集群](#)。

#### 📖 说明

本文以购买的MRS 3.1.0版本的集群为例，集群未开启Kerberos认证。

在本示例中，由于要分析处理OBS文件系统中的数据，因此在集群的高级配置参数中要为MRS集群绑定IAM权限委托，使得集群内组件能够对接OBS并具有对应文件系统目录的操作权限。

您可以直接选择“MRS\_ECS\_DEFAULT\_AGENCY”，也可以自行创建其他具有OBS文件系统操作权限的自定义委托。



集群购买成功后，在MRS集群的任一节点内，使用omm用户安装集群客户端，具体操作可参考[安装并使用集群客户端](#)。

例如客户端安装目录为“/opt/client”。

## 步骤 2：准备测试数据

在创建Flink作业进行数据分析前，需要在提前准备待分析的测试数据，并将该数据上传至OBS文件系统中。

**步骤1** 本地创建一个“mrs\_flink\_test.txt”文件，例如文件内容如下：

```
This is a test demo for MRS Flink. Flink is a unified computing framework that supports both batch processing and stream processing. It provides a stream data processing engine that supports data distribution and parallel computing.
```

**步骤2** 在云服务列表中选择“存储 > 对象存储服务”，登录OBS管理控制台。

**步骤3** 单击“并行文件系统”，创建一个并行文件系统，并上传测试数据文件。

例如创建的文件系统名称为“mrs-demo-data”，单击系统名称，在“文件”页面中，新建一个文件夹“flink”，上传测试数据至该目录中。

则本示例的测试数据完整路径为“obs://mrs-demo-data/flink/mrs\_flink\_test.txt”。

**步骤4** （可选）上传数据分析应用程序。

使用管理台界面直接提交作业时，将已开发好的Flink应用程序jar文件也可以上传至OBS文件系统中，或者MRS集群内的HDFS文件系统中。

本示例中使用MRS集群内置的Flink WordCount样例程序，可从MRS集群的客户端安装目录中获取，即“/opt/client/Flink/flink/examples/batch/WordCount.jar”。

将“WordCount.jar”上传至“mrs-demo-data/program”目录下。

----结束

## 步骤 3：创建并运行 Flink 作业

**方式1：在控制台界面在线提交作业。**

**步骤1** 登录MRS管理控制台，单击MRS集群名称，进入集群详情页面。

**步骤2** 在集群详情页的“概览”页签，单击“IAM用户同步”右侧的“同步”进行IAM用户同步。

**步骤3** 单击“作业管理”，进入“作业管理”页签。

**步骤4** 单击“添加”，添加一个Flink作业。

- 作业类型：Flink
- 作业名称：自定义，例如flink\_obs\_test。
- 执行程序路径：本示例使用Flink客户端的WordCount程序为例。
- 运行程序参数：使用默认值。
- 执行程序参数：设置应用程序的输入参数，“input”为待分析的测试数据，“output”为结果输出文件。

例如本示例中，设置为“--input obs://mrs-demo-data/flink/mrs\_flink\_test.txt --output obs://mrs-demo-data/flink/output”。

- 服务配置参数：使用默认值即可，如需手动配置作业相关参数，可参考[运行Flink作业](#)。

**步骤5** 确认作业配置信息后，单击“确定”，完成作业的新增，并等待运行完成。

----结束

**方式2：通过集群客户端提交作业。**

**步骤1** 使用root用户登录集群客户端节点，进入客户端安装目录。

```
su - omm
```

```
cd /opt/client
```

```
source bigdata_env
```

**步骤2** 执行以下命令验证集群是否可以访问OBS。

```
hdfs dfs -ls obs://mrs-demo-data/flink
```

**步骤3** 提交Flink作业，指定源文件数据进行消费。

```
flink run -m yarn-cluster /opt/client/Flink/flink/examples/batch/  
WordCount.jar --input obs://mrs-demo-data/flink/mrs_flink_test.txt --output  
obs://mrs-demo/data/flink/output2
```

```
...  
Cluster started: Yarn cluster with application id application_1654672374562_0011  
Job has been submitted with JobID a89b561de5d0298cb2ba01fbc30338bc  
Program execution finished  
Job with JobID a89b561de5d0298cb2ba01fbc30338bc has finished.  
Job Runtime: 1200 ms
```

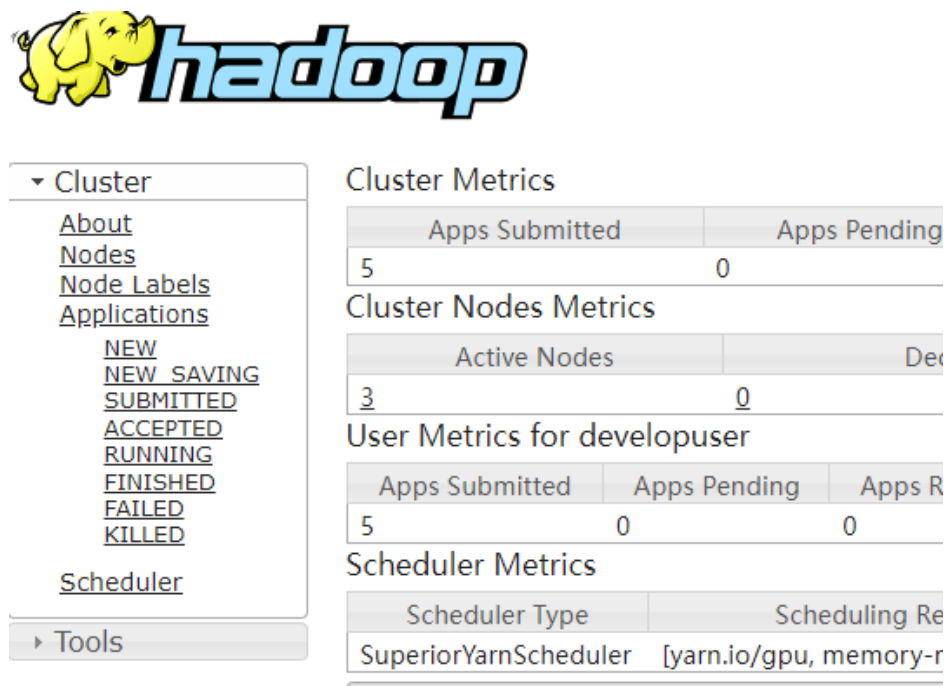
----结束

## 步骤 4：查看作业执行结果

**步骤1** 作业提交成功后，登录MRS集群的FusionInsight Manager界面，单击“集群 > 服务 > Yarn”。

**步骤2** 单击“ResourceManager WebUI”后的链接进入Yarn Web UI界面，在Applications页面查看当前Yarn作业的详细运行情况及运行日志。

图 2-14 Yarn WebUI 界面



**步骤3** 等待作业运行完成后，在OBS文件系统中指定的结果输出文件中可查看数据分析输出的结果。

下载“output”文件到本地并打开，可查看输出的分析结果。

```
a 3
and 2
batch 1
both 1
computing 2
data 2
demo 1
distribution 1
engine 1
flink 2
for 1
framework 1
is 2
it 1
mrs 1
parallel 1
processing 3
provides 1
stream 2
supports 2
test 1
that 2
this 1
unified 1
```

使用集群客户端命令行提交作业时，若不指定输出目录，在作业运行界面也可直接查看数据分析结果。

```
Job with JobID xxx has finished.
Job Runtime: xxx ms
Accumulator Results:
- e6209f96ffa423974f8c7043821814e9 (java.util.ArrayList) [31 elements]
(a,3)
```

```
(and,2)
(batch,1)
(both,1)
(computing,2)
(data,2)
(demo,1)
(distribution,1)
(engine,1)
(flink,2)
(for,1)
(framework,1)
(is,2)
(it,1)
(mrs,1)
(parallel,1)
(processing,3)
(provides,1)
(stream,2)
(supports,2)
(test,1)
(that,2)
(this,1)
(unified,1)
```

----结束

## 2.5 通过 Spark Streaming 作业消费 Kafka 数据

### 应用场景

本文介绍如何使用MRS集群运行Spark Streaming作业以消费Kafka数据。

假定某个业务Kafka每1秒就会收到1个单词记录。基于业务需要，开发的Spark应用程序实现实时累加计算每个单词的记录总数的功能。

Spark Streaming样例工程的数据存储在Kafka组件中，向Kafka组件发送数据。

MRS服务提供了Spark服务多种场景下的样例开发工程，本章节对应示例场景的开发思路：

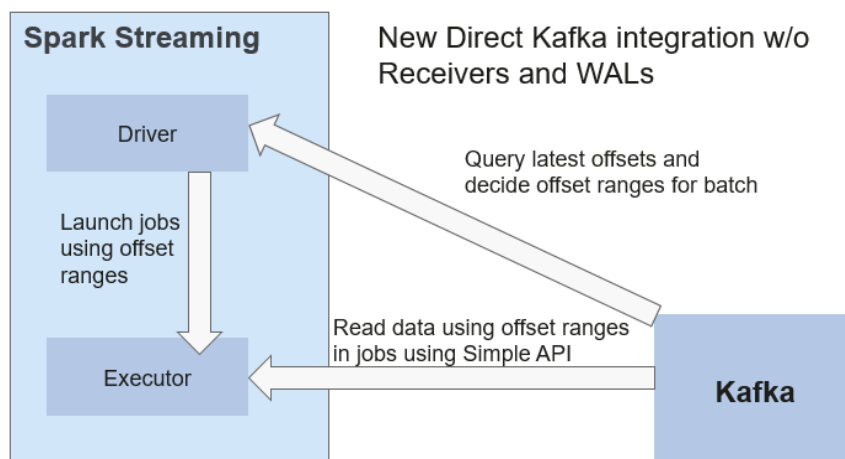
1. 接收Kafka中数据，生成相应DStream。
2. 对单词记录进行分类统计。
3. 计算结果，并进行打印。

### 方案架构

Spark Streaming是一种构建在Spark上的实时计算框架，扩展了Spark处理大规模流式数据的能力。当前Spark支持两种数据处理方式：Direct Streaming和Receiver。

Direct Streaming方式主要通过采用Direct API对数据进行处理。以Kafka Direct接口为例，与启动一个Receiver来连续不断地从Kafka中接收数据并写入到WAL中相比，Direct API简单地给出每个batch区间需要读取的偏移量位置。然后，每个batch的Job被运行，而对应偏移量的数据在Kafka中已准备好。这些偏移量信息也被可靠地存储在checkpoint文件中，应用失败重启时可以直接读取偏移量信息。

图 2-15 Direct Kafka 接口数据传输

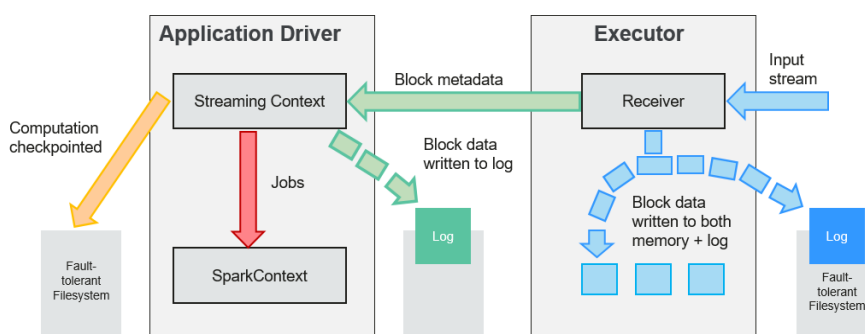


需要注意的是，Spark Streaming可以在失败后重新从Kafka中读取并处理数据段。然而，由于语义仅被处理一次，重新处理的结果和没有失败处理的结果是一致的。

因此，Direct API消除了需要使用WAL和Receivers的情况，且确保每个Kafka记录仅被接收一次，这种接收更加高效。使得Spark Streaming和Kafka可以很好地整合在一起。总体来说，这些特性使得流处理管道拥有高容错性、高效性及易用性，因此推荐使用Direct Streaming方式处理数据。

在一个Spark Streaming应用开始时（也就是Driver开始时），相关的StreamingContext（所有流功能的基础）使用SparkContext启动Receiver成为长驻运行任务。这些Receiver接收并保存流数据到Spark内存中以供处理。用户传送数据的生命周期如图2-16所示：

图 2-16 数据传输生命周期



1. 接收数据（蓝色箭头）

Receiver将数据流分成一系列小块，存储到Executor内存中。另外，在启用预写日志（Write-ahead Log，简称WAL）以后，数据同时还写入到容错文件系统的预写日志中。

2. 通知Driver（绿色箭头）

接收块中的元数据（Metadata）被发送到Driver的StreamingContext。这个元数据包括：

- 定位其在Executor内存中数据位置的块Reference ID。
- 若启用了WAL，还包括块数据在日志中的偏移信息。

3. 处理数据（红色箭头）  
对每个批次的数据，StreamingContext使用Block信息产生RDD及其Job。StreamingContext通过运行任务处理Executor内存中的Block来执行Job。
4. 周期性地设置检查点（橙色箭头）
5. 为了容错的需要，StreamingContext会周期性地设置检查点，并保存到外部文件系统中。

## 操作流程

本实践操作流程如下所示：

1. **步骤1：创建MRS集群**：创建一个包含有Spark2x、Kafka组件的MRS集群。
2. **步骤2：准备应用程序**：准备用于进行分析的源数据及程序。
3. **步骤3：上传jar包及源数据**：将相关程序和数据上传至MRS集群内。
4. **步骤4：运行作业并查看结果**：运行程序并查看数据分析结果。

### 步骤 1：创建 MRS 集群

**步骤1** 创建并购买一个包含有Spark2x、Kafka组件的MRS集群，详情请参见[购买自定义集群](#)。

#### 📖 说明

本文以购买的MRS 3.1.0版本的集群为例，集群未开启Kerberos认证。

**步骤2** 集群购买成功后，在MRS集群的任一节点内，安装集群客户端，具体操作可参考[安装并使用集群客户端](#)。

例如客户端安装目录为“/opt/client”。

----结束

### 步骤 2：准备应用程序

**步骤1** 通过开源镜像站获取样例工程。

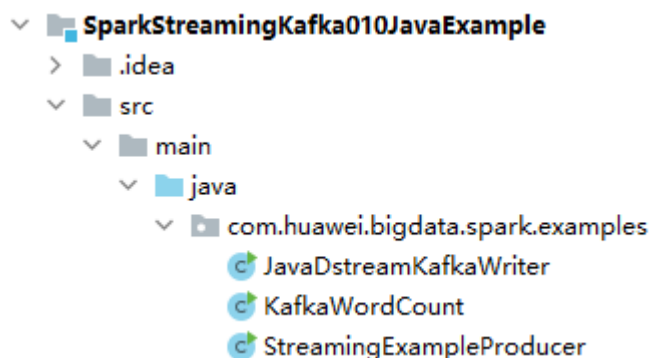
下载样例工程的Maven工程源码和配置文件，并在本地配置好相关开发工具，可参考[通过开源镜像站获取样例工程](#)。

根据集群版本选择对应的分支，下载并获取MRS相关样例工程。

例如本章节场景对应示例为“SparkStreamingKafka010JavaExample”样例，获取地址：<https://github.com/huaweicloud/huaweicloud-mrs-example/tree/mrs-3.1.0/src/spark-examples/sparknormal-examples/SparkStreamingKafka010JavaExample>。

**步骤2** 本地使用IDEA工具导入样例工程，等待Maven工程下载相关依赖包，具体操作可参考[配置并导入样例工程](#)。

图 2-17 导入 Spark 样例工程



在本示例工程中，通过使用Streaming调用Kafka接口来获取单词记录，然后把单词记录分类统计，得到每个单词记录数，关键代码片段如下：

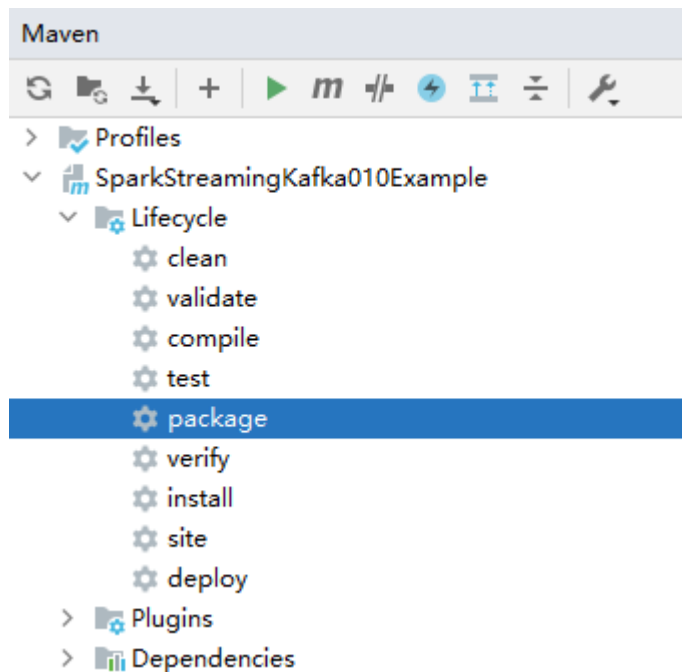
```
public class StreamingExampleProducer {
    public static void main(String[] args) throws IOException {
        if (args.length < 2) {
            printUsage();
        }
        String brokerList = args[0];
        String topic = args[1];
        String filePath = "/home/data/"; //源数据获取路径
        Properties props = new Properties();
        props.put(ProducerConfig.BOOTSTRAP_SERVERS_CONFIG, brokerList);
        props.put(ProducerConfig.CLIENT_ID_CONFIG, "DemoProducer");
        props.put(ProducerConfig.KEY_SERIALIZER_CLASS_CONFIG, StringSerializer.class.getName());
        props.put(ProducerConfig.VALUE_SERIALIZER_CLASS_CONFIG, StringSerializer.class.getName());
        Producer<String, String> producer = new KafkaProducer<String, String>(props);

        for (int m = 0; m < Integer.MAX_VALUE / 2; m++) {
            File dir = new File(filePath);
            File[] files = dir.listFiles();
            if (files != null) {
                for (File file : files) {
                    if (file.isDirectory()) {
                        System.out.println(file.getName() + "This is a directory!");
                    } else {
                        BufferedReader reader = null;
                        reader = new BufferedReader(new FileReader(filePath + file.getName()));
                        String tempString = null;
                        while ((tempString = reader.readLine()) != null) {
                            // Blank line judgment
                            if (!tempString.isEmpty()) {
                                producer.send(new ProducerRecord<String, String>(topic, tempString));
                            }
                        }
                        // make sure the streams are closed finally.
                        reader.close();
                    }
                }
            }
            try {
                Thread.sleep(3);
            } catch (InterruptedException e) {
                e.printStackTrace();
            }
        }

        private static void printUsage() {
            System.out.println("Usage: {brokerList} {topic}");
        }
    }
}
```

**步骤3** 本地配置好Maven及SDK相关参数后，样例工程会自动加载相关依赖包。加载完毕后，执行package打包，获取打包后的jar文件。

图 2-18 编译 Spark 应用程序



例如打包后的jar文件为“SparkStreamingKafka010JavaExample-1.0.jar”。

----结束

### 步骤 3: 上传 jar 包及源数据

**步骤1** 准备向Kafka发送的源数据，例如如下的“input\_data.txt”文件，将该文件上传到客户端节点的“/home/data”目录下。

```
ZhangSan  
LiSi  
WangwWU  
Tom  
Jemmy  
LinDa
```

**步骤2** 将编译后的jar包上传到客户端节点，例如上传到“/opt”目录。

#### 📖 说明

如果本地网络无法直接连接客户端节点上传文件，可先将jar文件或者源数据上传至OBS文件系统中，然后通过MRS管理控制台集群内的“文件管理”页面导入HDFS中，再通过HDFS客户端使用`hdfs dfs -get`命令下载到客户端节点本地。

----结束

### 步骤 4: 运行作业并查看结果

**步骤1** 使用root用户登录安装了集群客户端的节点。

```
cd /opt/client
```



**source bigdata\_env**

**步骤2** 创建用于接收数据的Kafka topic。

```
kafka-topics.sh --create --zookeeper quorumpeer实例IP地址:ZooKeeper客户端连接端口/kafka --replication-factor 2 --partitions 3 --topic topic名称
```

quorumpeer实例IP地址可登录集群的FusionInsight Manager界面，在“集群 > 服务 > ZooKeeper > 实例”界面中查询，多个地址可用“,”分隔。ZooKeeper客户端连接端口可通过ZooKeeper服务配置参数“clientPort”查询，默认为2181。

例如执行以下命令：

```
kafka-topics.sh --create --zookeeper 192.168.0.17:2181/kafka --replication-factor 2 --partitions 2 --topic sparkkafka
```

```
Created topic sparkkafka.
```

**步骤3** Topic创建成功后，运行程序向Kafka发送数据。

```
java -cp /opt/SparkStreamingKafka010JavaExample-1.0.jar:/opt/client/Spark2x/spark/jars/*:/opt/client/Spark2x/spark/jars/streamingClient010/* com.huawei.bigdata.spark.examples.StreamingExampleProducer Broker实例IP地址:Kafka连接端口 topic名称
```

Kafka Broker实例IP地址可登录集群的FusionInsight Manager界面，在“集群 > 服务 > Kafka > 实例”界面中查询，多个地址可用“,”分隔。Broker端口号可通过Kafka服务配置参数“port”查询，默认为9092。

例如执行以下命令：

```
java -cp /opt/SparkStreamingKafka010JavaExample-1.0.jar:/opt/client/Spark2x/spark/jars/*:/opt/client/Spark2x/spark/jars/streamingClient010/* com.huawei.bigdata.spark.examples.StreamingExampleProducer 192.168.0.131:9092 sparkkafka
```

```
...
transactional.id = null
value.serializer = class org.apache.kafka.common.serialization.StringSerializer

2022-06-08 15:43:42 INFO  AppInfoParser:117 - Kafka version: xxx
2022-06-08 15:43:42 INFO  AppInfoParser:118 - Kafka commitId: xxx
2022-06-08 15:43:42 INFO  AppInfoParser:119 - Kafka startTimeMs: xxx
2022-06-08 15:43:42 INFO  Metadata:259 - [Producer clientId=DemoProducer] Cluster ID: d54RYHthSUishVb6nTHP0A
```

**步骤4** 重新打开一个客户端连接窗口，执行以下命令，读取Kafka Topic中的数据。

```
cd /opt/client/Spark2x/spark
```

```
source bigdata_env
```

```
bin/spark-submit --master yarn --deploy-mode client --jars $(files=($SPARK_HOME/jars/streamingClient010/*.jar); IFS=,; echo "${files[*]}") --class com.huawei.bigdata.spark.examples.KafkaWordCount /opt/SparkStreamingKafka010JavaExample-1.0.jar <checkpointDir> <brokers> <topic> <batchTime>
```

- <checkPointDir>指应用程序结果备份到HDFS的路径，自行指定即可，例如“/tmp”。

- <brokers>指获取元数据的Kafka地址，格式为“*Broker实例IP地址:Kafka连接端口*”。
- <topic>指读取Kafka上的topic名称。
- <batchTime>指Streaming分批的处理间隔，例如设置为“5”。

例如执行以下命令：

```
cd /opt/client/Spark2x/spark
source bigdata_env
bin/spark-submit --master yarn --deploy-mode client --jars $(
files=$(SPARK_HOME/jars/streamingClient010/*.jar); IFS=,; echo "${files[*]}")
--class com.huawei.bigdata.spark.examples.KafkaWordCount /opt/
SparkStreamingKafka010JavaExample-1.0.jar /tmp 192.168.0.131:9092
sparkkafka 5
```

程序运行后，可查看到Kafka中数据的统计结果：

```
....
-----
Time: 1654674380000 ms
-----
(ZhangSan,6)
(Tom,6)
(LinDa,6)
(WangwWU,6)
(LiSi,6)
(Jemmmmy,6)
-----
Time: 1654674385000 ms
-----
(ZhangSan,717)
(Tom,717)
(LinDa,717)
(WangwWU,717)
(LiSi,717)
(Jemmmmy,717)
-----
Time: 1654674390000 ms
-----
(ZhangSan,2326)
(Tom,2326)
(LinDa,2326)
(WangwWU,2326)
(LiSi,2326)
(Jemmmmy,2326)
...

```

**步骤5** 登录FusionInsight Manager界面，单击“集群 > 服务 > Spark2x”。

**步骤6** 在服务概览页面单击Spark WebUI后的链接地址，可进入History Server页面。

单击待查看的App ID，您可以查看Spark Streaming作业的状态。

图 2-19 查看 Spark Streaming 作业状态

Spark Jobs (?)

User: root  
Total Uptime: 7.4 min  
Scheduling Mode: FIFO  
Completed Jobs: 192  
Event Timeline  
- Completed Jobs (192)

Page: 1 2 > 2 Pages. Jump to: 1 Show: 100 Items in a page: Go

Job id	Description	Submitted	Duration	Stages: Succeeded/Total	Tasks (for all stages): Succeeded/Total
191	Streaming job from [output.operation 0, batch time 15:52:40] print at KafkaWordCount.java:112	2022/06/08 15:53:24	9 ms	1/1 (1 skipped)	1/1 (1 skipped)
190	Streaming job from [output.operation 0, batch time 15:52:40] print at KafkaWordCount.java:112	2022/06/08 15:53:24	19 ms	2/2	4/4
189	Streaming job from [output.operation 0, batch time 15:52:35] print at KafkaWordCount.java:112	2022/06/08 15:53:24	8 ms	1/1	1/1
188	Streaming job from [output.operation 0, batch time 15:52:30] print at KafkaWordCount.java:112	2022/06/08 15:53:24	67 ms	1/1 (2 skipped)	2/2 (2 skipped)
187	Streaming job from [output.operation 0, batch time 15:52:30] print at KafkaWordCount.java:112	2022/06/08 15:53:24	29 ms	2/2 (1 skipped)	4/4 (1 skipped)
186	Streaming job from [output.operation 0, batch time 15:52:30] print at KafkaWordCount.java:112	2022/06/08 15:52:30	15 ms	1/1 (1 skipped)	1/1 (1 skipped)

----结束

## 2.6 通过 Flume 采集指定目录日志系统文件至 HDFS

### 应用场景

Flume是一个分布式、可靠和高可用的海量日志聚合的系统。它能够将不同数据源的海量日志数据进行高效收集、聚合、移动，最后存储到一个中心化数据存储系统中。支持在系统中定制各类数据发送方，用于收集数据。同时，提供对数据进行简单处理，并写到各种数据接受方（可定制）的能力。

Flume分为客户端和服务端，两者都是FlumeAgent。服务端对应着FlumeServer实例，直接部署在集群内部。而客户端部署更灵活，可以部署在集群内部，也可以部署在集群外。它们之间没有必然联系，都可以独立工作，并且提供的功能是一样的。

Flume客户端需要单独安装，支持将数据直接导到集群中的HDFS和Kafka等组件上。

本案例中，通过MRS自定义集群中的Flume组件，自动采集指定节点日志目录下新产生的文件并存储到HDFS文件系统中。

### 方案架构

Flume-NG由多个Agent来组成，而每个Agent由Source、Channel、Sink三个模块组成，其中Source负责接收数据，Channel负责数据的传输，Sink则负责数据向下一端的发送。

图 2-20 Flume-NG 架构

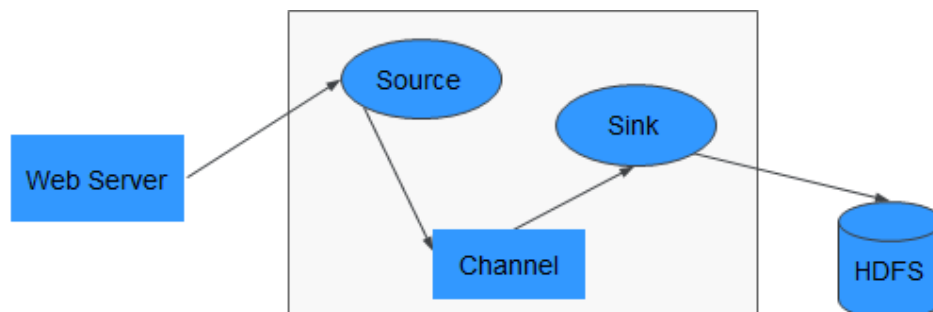
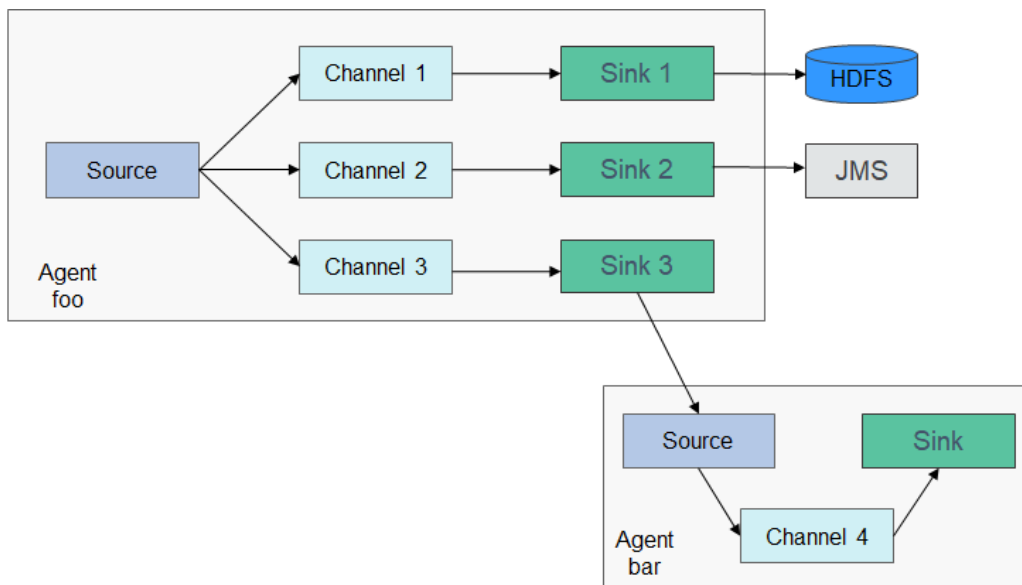


表 2-11 模块说明

名称	说明
Source	<p>Source负责接收数据或通过特殊机制产生数据，并将数据批量放到一个或多个Channel。Source的类型有数据驱动和轮询两种。</p> <p>典型的Source类型如下：</p> <ul style="list-style-type: none"><li>• 和系统集成的Sources: Syslog、Netcat。</li><li>• 自动生成事件的Sources: Exec、SEQ。</li><li>• 用于Agent和Agent之间通信的IPC Sources: Avro。</li></ul> <p>Source必须至少和一个Channel关联。</p>
Channel	<p>Channel位于Source和Sink之间，用于缓存来自Source的数据，当Sink成功将数据发送到下一跳的Channel或最终目的地时，数据从Channel移除。</p> <p>Channel提供的持久化水平与Channel的类型相关，有以下三类：</p> <ul style="list-style-type: none"><li>• Memory Channel: 非持久化。</li><li>• File Channel: 基于WAL（预写式日志Write-Ahead Logging）的持久化实现。</li><li>• JDBC Channel: 基于嵌入Database的持久化实现。</li></ul> <p>Channel支持事务，可提供较弱的顺序保证，可以和任何数量的Source和Sink工作。</p>
Sink	<p>Sink负责将数据传输到下一跳或最终目的，成功完成后将数据从Channel移除。</p> <p>典型的Sink类型如下：</p> <ul style="list-style-type: none"><li>• 存储数据到最终目的终端Sink，比如：HDFS、HBase。</li><li>• 自动消耗的Sink，比如：Null Sink。</li><li>• 用于Agent间通信的IPC sink: Avro。</li></ul> <p>Sink必须作用于一个确切的Channel。</p>

Flume也可以配置成多个Source、Channel、Sink，如图2-21所示：

图 2-21 Flume 结构图



## 操作流程

本实践操作流程如下所示：

1. **步骤1：创建MRS集群**：创建一个包含有Flume组件的MRS集群。
2. **步骤2：生成Flume配置文件**：根据业务场景配置Flume服务端和客户端侧配置文件。
3. **步骤3：安装Flume客户端**：在MRS集群中下载并安装Flume客户端。
4. **步骤4：查看日志采集结果**：查看日志采集结果。

### 步骤 1：创建 MRS 集群

**步骤1** 创建并购买一个包含有Flume、HDFS组件的MRS集群，详情请参见[购买自定义集群](#)。

#### 📖 说明

本文以购买的MRS 3.1.0版本的集群为例，集群未开启Kerberos认证。

**步骤2** 集群购买成功后，登录集群的FusionInsight Manager界面，下载集群客户端并解压。

由于Flume客户端需要单独安装，需要首先下载集群的客户端安装包到待安装Flume客户端的节点上并解压。

1. 在FusionInsight Manager “主页” 页签的集群名称后单击 **...**，单击“下载客户端” 下载集群客户端。
2. 在“下载集群客户端” 弹窗中填写集群客户端下载信息。

图 2-22 下载集群客户端

## 下载集群客户端

下载Cluster1的客户端，集群的客户端包括了所有服务

选择客户端类型： 完整客户端  仅配置文件

选择平台类型： x86\_64  aarch64

仅保存到如下路径：

- “选择客户端类型”中选择“完整客户端”。
  - “选择平台类型”必须与待安装节点的架构匹配，以“x86\_64”为例。
  - 勾选“仅保存到如下路径”，填写下载路径，本示例以“/tmp/FusionInsight-Client/”为例，需确保omm用户对该路径有操作权限。
3. 客户端软件包下载完成后，以root用户登录集群的主OMS节点，复制安装包到指定节点。

客户端软件包默认下载至集群的主OMS节点（可通过FusionInsight Manager的“主机”界面查看带有★标识的节点），如需要在集群内其他节点上安装，执行以下命令将软件包传输至其他节点，否则本步骤可忽略。

```
cd /tmp/FusionInsight-Client/
```

```
scp -p FusionInsight_Cluster_1_Services_Client.tar 待安装Flume客户端节点的IP地址:/tmp
```

4. 以root用户登录待安装Flume客户端的节点，进入客户端软件包所在目录后，执行以下命令解压软件包。

```
tar -xvf FusionInsight_Cluster_1_Services_Client.tar
```

```
tar -xvf FusionInsight_Cluster_1_Services_ClientConfig.tar
```

----结束

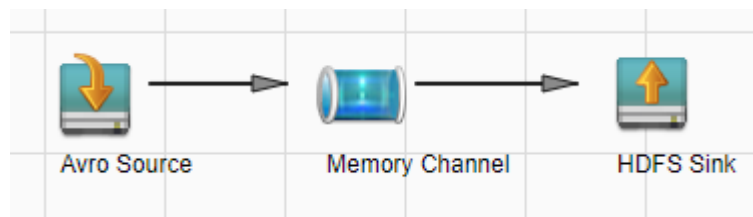
## 步骤 2：生成 Flume 配置文件

**步骤1** 登录FusionInsight Manager，选择“集群 > 服务 > Flume > 配置工具”。

**步骤2** 配置并导出“properties.properties”文件。

选择“Agent名”为“server”，分别选择“Avro Source”、“Memory Channel”和“HDFS Sink”模块，并连接。

图 2-23 生成 Flume 配置文件



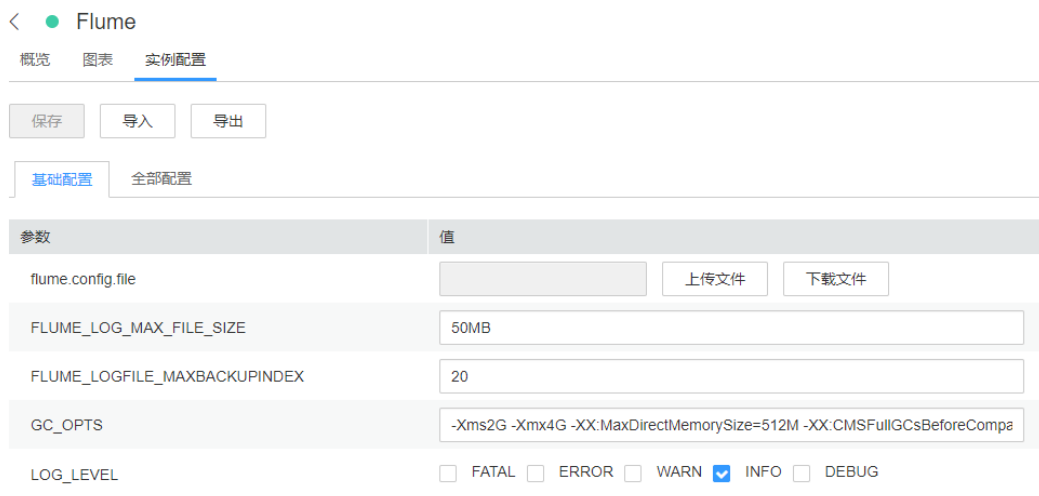
双击模块图标，配置对应参数，各模块配置参数如下（其他参数保持默认）：

类型	配置参数	描述	配置示例
Avro Source	名称	模块名称，可自定义。	test_source_1
	bind	待连接的Flume角色所在节点的IP地址，可在“集群 > 服务 > Flume > 实例”页面查看任一Flume角色实例的IP地址。	192.168.10.192
	port	连接端口，建议从21154开始配置。	21154
Memory Channel	名称	模块名称，可自定义。	test_channel_1
HDFS Sink	名称	模块名称，可自定义。	test_sink_1
	hdfs.path	日志文件写入HDFS的目录。	hdfs://hacluster/flume/test
	hdfs.filePrefix	写入HDFS后的文件名前缀。	over_% {basename}

**步骤3** 单击“导出”按钮，下载“properties.properties”文件到本地。

**步骤4** 在FusionInsight Manager界面，单击“集群 > 服务 > Flume > 实例”，单击准备上传配置文件的节点行的“Flume”角色，进入“实例配置”页面。

图 2-24 上传配置文件



**步骤5** 选择“上传文件”，上传“properties.properties”文件。

单击“保存”，单击“确定”后等待配置完成。

**步骤6** 选择“集群 > 服务 > Flume > 配置工具”

选择“Agent名”为“client”，分别选择“SpoolDir Source”、“Memory Channel”和“Avro Sink”模块，并连接。

图 2-25 配置 Flume



双击模块图标，配置对应参数，各模块配置参数如下（其他参数请保持默认）：

类型	配置参数	描述	配置示例
SpoolDir Source	名称	模块名称，可自定义。	test_source_1
	spoolDir	需要采集日志的目录，该目录需要对flume运行用户具有读写权限，并放入文件进行验证。	/var/log/Bigdata/audit/test
Memory Channel	名称	模块名称，可自定义。	test_channel_1
HDFS Sink	名称	模块名称，可自定义。	test_sink_1
	hostname	待连接的Flume角色所在节点的IP地址。	192.168.10.192
	port	连接端口，建议从21154开始配置。	21154



**步骤7** 单击“导出”按钮，下载“properties.properties”文件到本地。

**步骤8** 将“properties.properties”文件重命名为“client.properties.properties”，然后将该文件上传并覆盖到Flume客户端节点的“集群客户端安装包解压路径/Flume/FlumeClient/flume/conf”目录下。

----结束

### 步骤 3: 安装 Flume 客户端

**步骤1** 以root用户登录待安装Flume客户端所在节点。

**步骤2** 进入客户端安装包解压路径，例如客户端安装包上传至“/tmp”目录下并解压。

**步骤3** 执行以下命令安装Flume客户端，其中“/opt/FlumeClient”为自定义的Flume客户端安装路径。

```
cd /tmp/FusionInsight-Client/FusionInsight_Cluster_1_Services_ClientConfig/  
Flume/FlumeClient
```

```
./install.sh -d /opt/FlumeClient -c flume/conf/client.properties.properties
```

```
CST ... [flume-client install]: install flume client successfully.
```

----结束

### 步骤 4: 查看日志采集结果

**步骤1** Flume客户端安装成功后，向日志采集目录写入新的日志文件，验证日志是否传输成功。

例如在“/var/log/Bigdata/audit/test”目录下新建几个日志文件。

```
cd /var/log/Bigdata/audit/test
```

```
vi log1.txt
```

```
Test log file 1!!!
```

```
vi log2.txt
```

```
Test log file 2!!!
```

**步骤2** 写入日志文件后，执行ll命令查看文件列表，日志文件自动增加了后缀“.COMPLETED”，表示该文件已被采集。

```
-rw-----. 1 root root    75 Jun  9 19:59 log1.txt.COMPLETED  
-rw-----. 1 root root    75 Jun  9 19:59 log2.txt.COMPLETED
```

**步骤3** 登录FusionInsight Manager，选择“集群 > 服务 > HDFS”，单击“NameNode(主)”对应的链接，打开HDFS WebUI界面。

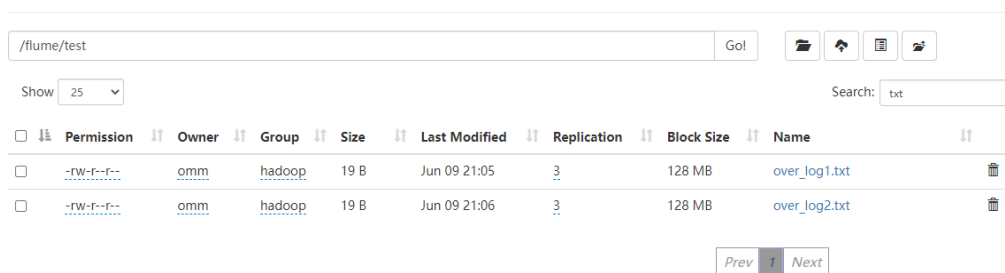
图 2-26 HDFS 服务概述



步骤4 选择“Utilities > Browse the file system”，观察HDFS上“/flume/test”目录下是否有产生数据。

图 2-27 查看 HDFS 文件

### Browse Directory



如上所示，文件目录下产生了收集的日志文件，文件名统一增加了前缀“over\_”。下载日志文件“over\_log1.txt”并查看内容，与写入的日志文件“log1.txt”内容一致。

Test log file 1!!!

----结束

## 2.7 基于 Kafka 的 Word Count 数据流统计案例

### 应用场景

Kafka是一个分布式的消息发布-订阅系统。它采用独特的设计提供了类似JMS的特性，主要用于处理活跃的流式数据。

Kafka有很多适用的场景：消息队列、行为跟踪、运维数据监控、日志收集、流处理、事件溯源、持久化日志等。

Kafka有如下几个特点：

- 高吞吐量
- 消息持久化到磁盘
- 分布式系统易扩展
- 容错性好

MRS服务提供了Kafka多种场景下的样例开发工程，本章节以最常见的WordCount样例Demo进行说明，对应示例场景的开发思路：

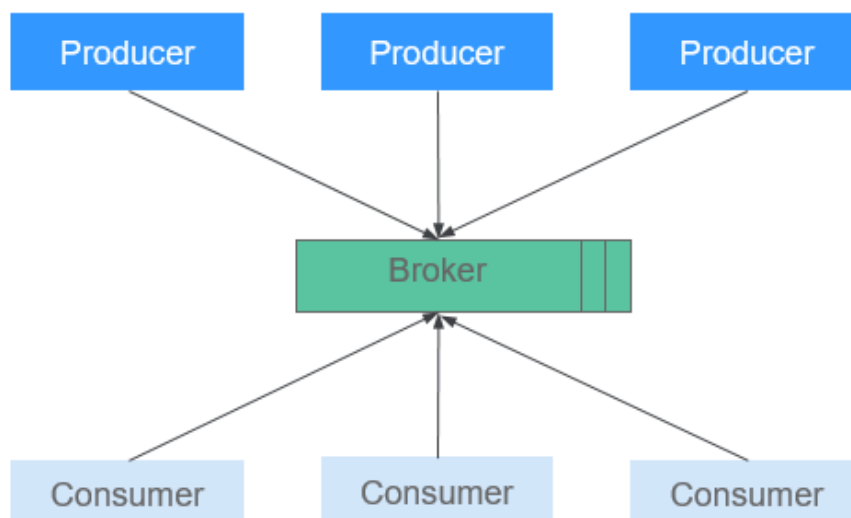
1. 使用Kafka客户端创建两个Topic，用于输入Topic和输出Topic。
2. 开发一个Kafka Streams完成单词统计功能，通过读取输入Topic中的消息，统计每条消息中的单词个数，从输出Topic消费数据，将统计结果以Key-Value的形式输出。

### 方案架构

生产者（Producer）将消息发布到Kafka主题（Topic）上，消费者（Consumer）订阅这些主题并消费这些消息。

在Kafka集群上一个服务器称为一个Broker。对于每一个主题，Kafka集群保留一个用于缩放、并行化和容错性的分区（Partition）。每个分区是一个有序、不可变的消息序列，并不断追加到提交日志文件。分区的消息每个也被赋值一个称为偏移顺序（Offset）的序列化编号。

图 2-28 Kafka 结构



## 操作流程

本实践操作流程如下所示：

1. **步骤1：创建MRS集群**：创建一个包含有Kafka组件的MRS集群。
2. **步骤2：准备应用程序**：准备用于进行分析的源数据及程序。
3. **步骤3：上传jar包及源数据**：将相关程序和数据上传至MRS集群内。
4. **步骤4：运行作业并查看结果**：运行程序并查看数据分析结果。

### 步骤 1：创建 MRS 集群

**步骤1** 创建并购买一个包含有Kafka组件的MRS集群，详情请参见[购买自定义集群](#)。

#### 📖 说明

本文以购买的MRS 3.1.0版本的集群为例，组件包含Hadoop、Kafka组件，集群未开启Kerberos认证。

**步骤2** 集群购买成功后，在MRS集群的任一节点内，安装集群客户端，具体操作可参考[安装并使用集群客户端](#)。

例如客户端安装在主管理节点中，安装目录为“/opt/client”。

**步骤3** 客户端安装完成后，在客户端内创建“lib”目录，用于放置相关jar包。

将安装客户端过程中解压的目录中Kafka相关jar包复制到“lib”目录。

例如客户端软件包的下载路径为主管理节点的“/tmp/FusionInsight-Client”目录，执行以下命令：

```
mkdir /opt/client/lib
```

```
cd /tmp/FusionInsight-Client/FusionInsight_Cluster_1_Services_ClientConfig
```

```
scp Kafka/install_files/kafka/libs/* /opt/client/lib
```

----结束

## 步骤 2: 准备应用程序

**步骤1** 通过开源镜像站获取样例工程。

下载样例工程的Maven工程源码和配置文件，并在本地配置好相关开发工具，可参考[通过开源镜像站获取样例工程](#)。

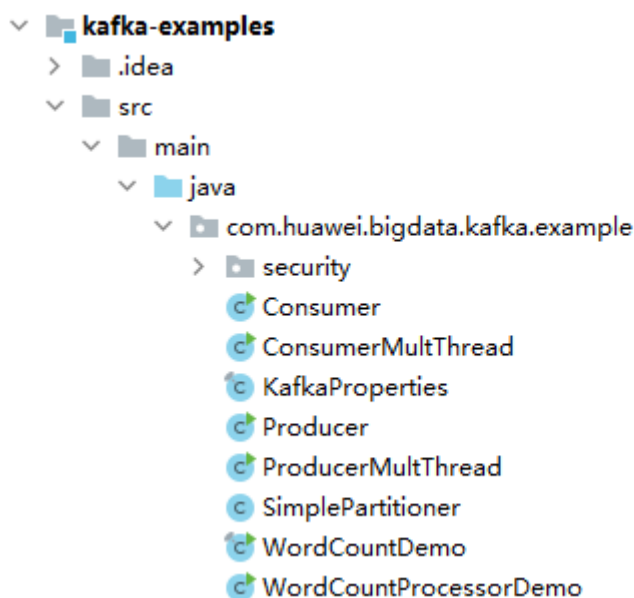
根据集群版本选择对应的分支，下载并获取MRS相关样例工程。

例如本章节场景对应示例为“WordCountDemo”样例，获取地址：<https://github.com/huaweicloud/huaweicloud-mrs-example/tree/mrs-3.1.0/src/kafka-examples>。

**步骤2** 本地使用IDEA工具导入样例工程，等待Maven工程下载相关依赖包。

本地配置好Maven及SDK相关参数后，样例工程会自动加载相关依赖包，具体操作可参考[配置并导入样例工程](#)。

图 2-29 导入 Kafka 样例程序



在示例程序“WordCountDemo”中，通过调用Kafka接口来获取单词记录，然后把单词记录分类统计，得到每个单词记录数，关键代码片段如下：

```
...
static Properties getStreamsConfig() {
    final Properties props = new Properties();
    KafkaProperties kafkaProc = KafkaProperties.getInstance();
    // Broker地址列表，根据集群实际情况配置
    props.put(BOOTSTRAP_SERVERS, kafkaProc.getValues(BOOTSTRAP_SERVERS, "node-  
group-1kLfk.mrs-rbmq.com:9092"));
    props.put(SASL_KERBEROS_SERVICE_NAME, "kafka");
    props.put(KERBEROS_DOMAIN_NAME, kafkaProc.getValues(KERBEROS_DOMAIN_NAME,
"hadop.hadop.com"));
    props.put(APPLICATION_ID, kafkaProc.getValues(APPLICATION_ID, "streams-wordcount"));
    // 协议类型：当前支持配置为SASL_PLAINTEXT或者PLAINTEXT
    props.put(SEcurity_PROTOCOL, kafkaProc.getValues(SEcurity_PROTOCOL, "PLAINTEXT"));
    props.put(CACHE_MAX_BYTES_BUFFERING, 0);
    props.put(DEFAULT_KEY_SERDE, Serdes.String().getClass().getName());
    props.put(DEFAULT_VALUE_SERDE, Serdes.String().getClass().getName());
    props.put(ConsumerConfig.AUTO_OFFSET_RESET_CONFIG, "earliest");
    return props;
}
```

```
}
static void createWordCountStream(final StreamsBuilder builder) {
    // 从 input-topic 接收输入记录
    final KStream<String, String> source = builder.stream(INPUT_TOPIC_NAME);
    // 聚合 key-value 键值对的计算结果
    final KTable<String, Long> counts = source
        .flatMapValues(value ->
Arrays.asList(value.toLowerCase(Locale.getDefault()).split(REGEX_STRING)))
        .groupBy((key, value) -> value)
        .count();
    // 将计算结果的 key-value 键值对从 output topic 输出
    counts.toStream().to(OUTPUT_TOPIC_NAME, Produced.with(Serdes.String(), Serdes.Long()));
}
...
}
```

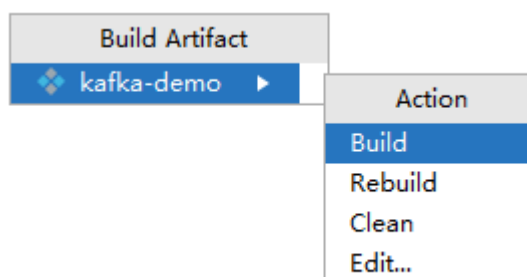
#### 📖 说明

- BOOTSTRAP\_SERVERS需根据集群实际情况，配置为Kafka Broker节点的主机名及端口，可通过集群FusionInsight Manager界面中单击“集群 > 服务 > Kafka > 实例”查看Broker在Linux中调测程序实例信息。
- SECURITY\_PROTOCOL为连接Kafka的协议类型，在本示例中，配置为“PLAINTEXT”。

**步骤3** 确认“WordCountDemo.java”内的参数无误后，将工程编译后进行打包，获取打包后的jar文件。

编译jar包详细操作可参考在Linux中调测程序。

图 2-30 编译 Kafka 程序



例如打包后的jar文件为“kafka-demo.jar”。

----结束

## 步骤 3：上传 jar 包及源数据

将编译后的jar包上传到客户端节点，例如上传到“/opt/client/lib”目录下。

#### 📖 说明

如果本地网络无法直接连接客户端节点上传文件，可先将jar文件或者源数据上传至OBS文件系统中，然后通过MRS管理控制台集群内的“文件管理”页面导入HDFS中，再通过HDFS客户端使用`hdfs dfs -get`命令下载到客户端节点本地。

## 步骤 4：运行作业并查看结果

**步骤1** 使用root用户登录安装了集群客户端的节点。

```
cd /opt/client
```

```
source bigdata_env
```

- 步骤2** 创建输入Topic和输出Topic，与样例代码中指定的Topic名称保持一致，输出Topic的清理策略设置为compact。

```
kafka-topics.sh --create --zookeeper quorumpeer实例IP地址:ZooKeeper客户端连接端口/kafka --replication-factor 1 --partitions 1 --topic Topic名称
```

quorumpeer实例IP地址可登录集群的FusionInsight Manager界面，在“集群 > 服务 > ZooKeeper > 实例”界面中查询，多个地址可用“,”分隔。ZooKeeper客户端连接端口可通过ZooKeeper服务配置参数“clientPort”查询，默认为2181。

例如执行以下命令：

```
kafka-topics.sh --create --zookeeper 192.168.0.17:2181/kafka --replication-factor 1 --partitions 1 --topic streams-wordcount-input
```

```
kafka-topics.sh --create --zookeeper 192.168.0.17:2181/kafka --replication-factor 1 --partitions 1 --topic streams-wordcount-output --config cleanup.policy=compact
```

- 步骤3** Topic创建成功后，执行以下命令运行程序。

```
java -cp ./opt/client/lib/*  
com.huawei.bigdata.kafka.example.WordCountDemo
```

- 步骤4** 重新打开一个客户端连接窗口，执行以下命令，使用“kafka-console-producer.sh”向输入Topic中写入消息：

```
cd /opt/client
```

```
source bigdata_env
```

```
kafka-console-producer.sh --broker-list Broker实例IP地址:Kafka连接端口（例如  
192.168.0.13:9092） --topic streams-wordcount-input --producer.config /opt/  
client/Kafka/kafka/config/producer.properties
```

- 步骤5** 重新打开一个客户端连接窗口，执行以下命令，使用“kafka-console-consumer.sh”命令消费数据，查看统计结果。

```
cd /opt/client
```

```
source bigdata_env
```

```
kafka-console-consumer.sh --topic streams-wordcount-output --bootstrap-server Broker实例IP地址:Kafka连接端口 --consumer.config /opt/client/Kafka/  
kafka/config/consumer.properties --from-beginning --property print.key=true  
--property print.value=true --property  
key.deserializer=org.apache.kafka.common.serialization.StringDeserializer --  
property  
value.deserializer=org.apache.kafka.common.serialization.LongDeserializer --  
formatter kafka.tools.DefaultMessageFormatter
```

向**步骤4**中打开的写入窗口中写入消息：

```
>This is Kafka Streams test  
>test starting  
>now Kafka Streams is running  
>test end
```

查看到消息输出：

```
this 1  
is 1
```

```
kafka 1
streams 1
test 1
test 2
starting 1
now 1
kafka 2
streams 2
is 2
running 1
test 3
end 1
```

----结束

## 2.8 实时 OLAP 数据分析

### 2.8.1 实时 OLAP 场景介绍

传统数据集市场景实时数据和离线数据分离，集市层指标通过预聚合进行定制化开发，时效性低，复杂性高，灵活性差。

基于Doris的实时OLAP场景离线和实时数据归一，无需离线预加工，基于海量明细数据直接进行多维秒级聚合/关联查询分析，具备实时、高效、灵活的特点。

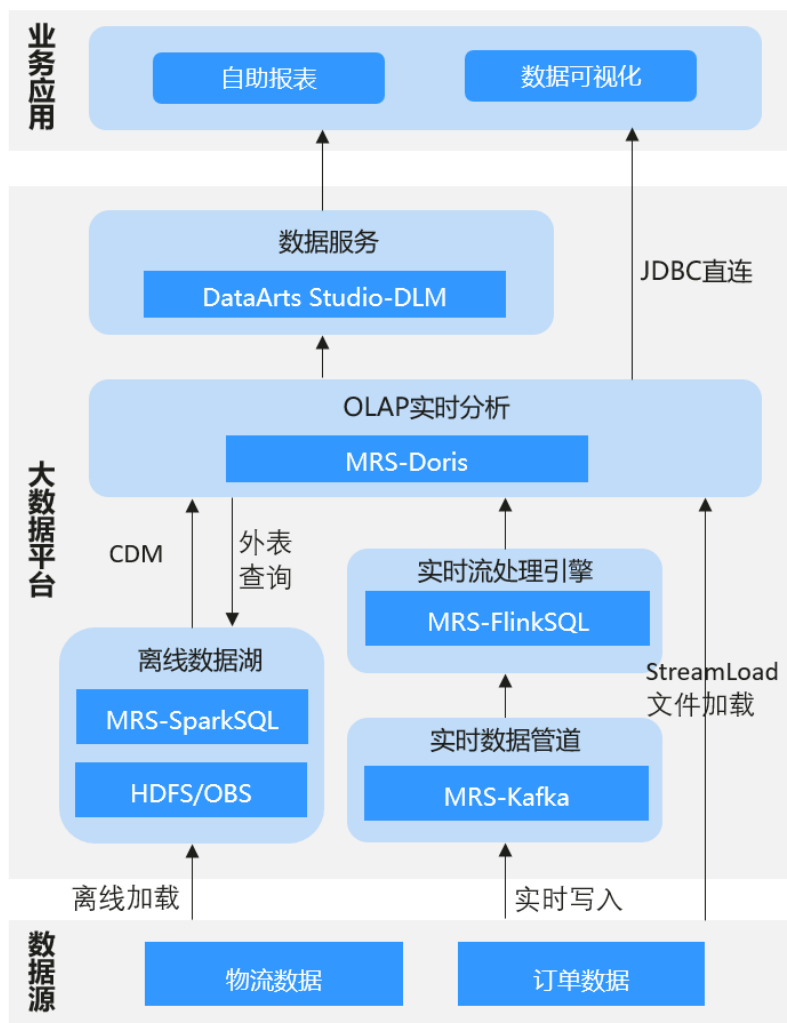
Doris全面综合的性能表现也能在轻量化大数据场景中独立承担数据从加工到查询的全链路数据处理需求。

### 实时 OLAP 场景化方案

实时OLAP场景化方案如[图2-31](#)所示。



图 2-31 实时 OLAP 场景化方案



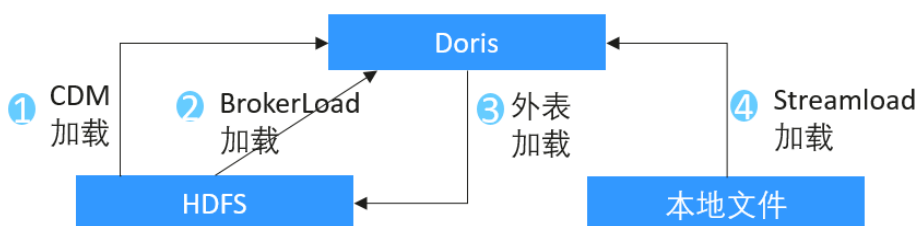
- 数据源：业务系统产生的数据，可以是离线的文件数据，关系型数据库数据，或者实时采集的流式数据等。
- 大数据平台：实时OLAP的核心数据处理平台，包含离线数据加工、实时数据加工、实时OLAP等模块。
  - 实时消息管道：用于接收实时流式数据，作为实时数据接入的统一管道，对应大数据组件MRS-Kafka。
  - 实时流处理引擎：从Kafka实时读取数据并进行复杂的实时计算，将结果送到下游，可以实现毫秒级的实时流式处理，对应大数据组件MRS-FlinkSQL。
  - 离线数据湖：针对业务系统的离线数据进行统一归集和处理，进行贴源数据加工。
  - OLAP实时分析：Doris是实时OLAP场景的核心组件，用于实时OLAP集市数据存储和查询，提供大数据量的高效关联/聚合分析能力。
  - 数据服务：提供数据服务API开发和开放能力，将数据集市的查询定义为数据服务接口，对上层应用提供接口化的数据服务能力。数据服务能力由DataArts Studio的DLM组件提供。
  - JDBC直连：Doris提供标准的JDBC接口，在BI工具对接等场景中，也可以直接使用JDBC接口连接Doris进行数据查询。

- 离线数据集成：离线数据集成可以从数据湖中集成离线数据，或者直接集成本地文件到Doris中。离线数据湖中的数据可以通过CDM或者加载外表的方式进行数据集成，本地文件则通过Doris自带的StreamLoad工具进行本地文件加载。

## 离线数据加载

离线数据可以从数据湖加载，也可以直接加载本地文件。从数据湖加载可以使用工具CDM，在没有CDM工具时，可以直接使用外表加载、BrokerLoad的方式。本地文件加载则使用Doris自带的StreamLoad工具。

图 2-32 离线数据加载

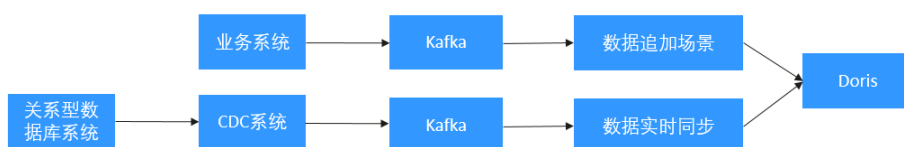


- **场景特点：**  
业务特点为离线数据，端到端时延无要求，仅对数据集市中的数据查询时延有要求，如离线报表场景、实时报表的数据初始化场景。
- **数据加载方式：**
  - 在存在DataArts Studio工具的场景下，数据加载统一使用CDM进行（DataArts Studio需要2.10及以上版本）。
  - 在不存在DataArts Studio工具的场景下，可以使用Spark或者外表方式加载。由数据加工方进行数据推送加载时，推荐使用BorkerLoad加载方式，方便数据加工方统一调度；如果有数据应用方负责数据加载，则使用外表方式进行数据加载较为方便应用层另外配置加载数据。
  - 在没有数据湖的场景下，轻量化数仓场景中，数据直接由源端文件加载到Doris中，可以使用Streamload方式加载本地文件。
- **注意事项：**  
Hive外表通过整体添加Hive的MetaStore的方式，将全量元数据在Doris中呈现。

## 实时数据加载

对实时性要求较高的场景，可直接将实时流式数据通过FlinkSQL传入Doris中，在Doris中进行实时查询。如果数据在加载到Doris之前需要处理，可直接在FlinkSQL中进行数据加工，将加工后的数据实时写入Doris中。

图 2-33 实时数据加载

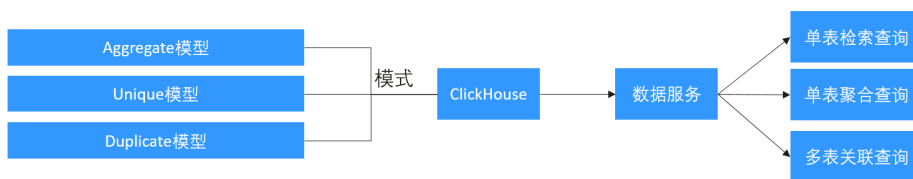


- **场景特点:**
  - 业务特点为实时流式数据，需要针对实时流式数据进行快速的统计查询。
  - 前端业务库系统实时同步到后端分析系统，在后端分析系统进行实时OLAP分析。
- **数据加载方式:**
  - 前端数据统一接入Kafka中，Kafka作为实时数据加工的唯一数据源。
  - 针对仅追加的数据，按正常的的数据插入逻辑，进行数据插入。
  - 针对实时数据同步方式，数据格式指定为Debezium方式，利用Flink的数据同步能力将数据同步到Flink。
- **注意事项:**
  - 实时流式数据写入Doris时，需要使用微批模式，一批数据写入一次，提高写入效率。
  - 使用DataArts DLF进行流任务调度开发时，需要使用2.10.1及以上的版本。

## 实时 OLAP 数据查询

Doris支持海量数据的亚秒级查询，支持单表数据的聚合查询和多表关联查询。Doris不同的表引擎，适合不同的业务场景，可以根据业务特点选择不同的表格式。

图 2-34 实时 OLAP 数据查询



- **表模型选择:**
  - Aggregate模型可以通过预聚合，极大地降低聚合查询时所需扫描的数据量和查询的计算量，适合有固定模式的报表类查询场景，但是该模型不适用于count(\*)查询。同时因为固定了Value列上的聚合方式，在进行其他类型的聚合查询时，需要考虑语意正确性。
  - Unique模型针对需要唯一主键约束的场景，可以保证主键唯一性约束。但是无法利用ROLLUP等预聚合带来的查询优势。适合数据有大量更新的场景。对于聚合查询有较高性能需求的用户，推荐使用写时合并实现。
  - Duplicate适合任意维度的Ad-hoc查询。虽然无法利用预聚合的特性，但是不受聚合模型的约束，可以发挥列存模型的优势。适合不要求数据更新的通用查询场景。
- **查询模式:**
  - 单表检索查询，按索引进行单表检索查询，可进行高并发，低延迟的数据查询访问。
  - 单表聚合查询，不同维度进行聚合查询，可提供低延迟的数据聚合查询能力。
  - 多表关联查询，通常用于事实表和维度表的多表关联查询场景。

## 2.8.2 离线数据加载：通过 CDM 将 Hive 外表数据迁移到 Doris

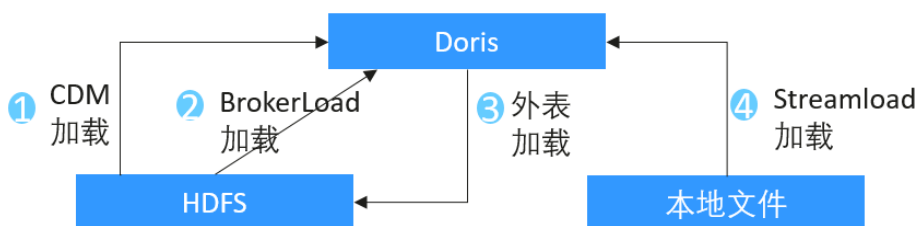
### 应用场景

离线数据，端到端时延无要求。在存在DataArts Studio工具的场景下，数据加载使用 CDM进行将离线的Hive外表数据迁移到MRS Doris。

### 方案架构

离线数据可以从数据湖加载，也可以直接加载本地文件。从数据湖加载可以使用工具 CDM，在没有CDM工具时，可以直接使用外表加载、BrokerLoad的方式。本地文件加载则使用Doris自带的StreamLoad工具。

图 2-35 离线数据加载



### 操作流程

- **步骤1：创建MRS集群并配置**
- **步骤2：在CDM创建Hive和Doris连接**
- **步骤3：Hive外表数据通过CDM迁移到Doris**

### 约束与限制

DataArts Studio实例、CDM集群、MRS集群需要具有相同的“虚拟私有云”和“安全组”。

### 步骤 1：创建 MRS 集群并配置

**步骤1** 已创建DataArts Studio实例，详细操作请参考[购买DataArts Studio实例](#)。

记录“虚拟私有云”和“安全组”信息。

**步骤2** 已创建CDM集群，详细操作请参考[创建CDM集群](#)。

其中“虚拟私有云”和“安全组”需要与已创建的DataArts Studio实例保持一致。

**步骤3** 创建具有访问OBS权限的ECS委托，详细操作请参考[配置MRS集群通过IAM委托对接OBS](#)中“创建具有访问OBS权限的ECS委托”部分。例如委托名称为“mrs\_ecs\_obs”。

**步骤4** 创建包含Hive、Spark等组件的MRS集群，并创建用户、安装客户端。

1. 创建MRS集群，详细操作请参考[自定义购买MRS集群](#)。其中：
  - 集群名称：自定义，例如“mrs\_test”
  - 版本类型：LTS版

- 集群版本：MRS 3.3.0-LTS
  - 集群组件包含：Hive、Spark等。
  - “虚拟私有云”和“安全组”需要与已创建的DataArts Studio实例保持一致。
  - Kerberos认证：开启
  - 其他参数根据实际需要进行配置。
2. 等待集群创建成功后，在MRS管理控制台“现有集群”中单击已创建的MRS集群名称（例如“mrs\_test”）。
- 在集群“概览”页签单击“委托”后的“选择委托”，选择已创建的委托名称（例如“mrs\_ecs\_obs”）。单击“确定”。

图 2-36 选择委托



3. 创建拥有Hive权限的用户。
- a. 登录集群Manager页面，详细操作请参考[访问MRS集群Manager](#)。
  - b. 在Manager界面选择“系统 > 权限 > 角色”创建一个Hive角色（例如“hivrole”），详细操作请参考[创建Hive角色](#)。
  - c. 选择“系统 > 权限 > 用户 > 添加用户”，根据界面提示创建一个人机用户（例如hiveuser），并绑定新创建的Hive角色（例如“hivrole”）。
  - d. 使用新创建的用户（例如hiveuser）登录Manager，并根据提示修改初始密码。
4. 集群购买完成后安装集群全量客户端，详细操作请参考[安装客户端（3.x版本）](#)。例如客户端安装路径为“/opt/newclient”。
5. 数据湖已在MRS Hive外表中存在：lineorder订单事实表，且有存量数据。

建表语句参考：

```
CREATE TABLE `lineorder`(  
  `lo_orderkey` bigint,  
  `lo_linenumber` int,  
  `lo_custkey` bigint,  
  `lo_partkey` bigint,  
  `lo_suppkey` bigint,  
  `lo_orderpriority` varchar(100),  
  `lo_shippriority` int,  
  `lo_quantity` int,  
  `lo_extendedprice` bigint,  
  `lo_ordtotalprice` bigint,  
  `lo_discount` int,
```

```
`lo_revenue` bigint,  
`lo_supplycost` int,  
`lo_tax` int,  
`lo_commitdate` date,  
`lo_shipmode` varchar(100))  
PARTITIONED BY (  
`lo_orderdate` date)  
LOCATION 'obs目录';
```

**步骤5** 创建MRS Doris集群，并在Doris集群安装MySQL客户端、创建拥有Doris权限的用户、创建Doris数据库。

1. 创建MRS Doris集群，详细操作请参考[自定义购买MRS集群](#)。其中：
  - 集群名称：自定义，例如“mrs\_doris”
  - 版本类型：LTS版
  - 集群版本：MRS 3.3.0-LTS
  - 组件选择：Doris集群。
  - “虚拟私有云”和“安全组”需要与已创建的DataArts Studio实例保持一致。
  - Kerberos认证：开启
  - 其他参数根据实际需要进行配置。
2. 等待集群创建成功后，在MRS管理控制台“现有集群”中单击已创建的MRS集群名称（例如“mrs\_doris”）。  
在集群“概览”页签单击“委托”后的“选择委托”，选择已创建的委托名称（例如“mrs\_ecs\_obs”）。单击“确定”。
3. 在Doris集群安装MySQL客户端，详细操作请参考[使用MySQL客户端连接Doris](#)。
4. 创建拥有Doris管理权限的人机用户（例如用户名为dorisuser），并修改初始密码。
  - a. 登录Doris集群Manager页面，详细操作请参考[访问MRS集群Manager](#)。
  - b. 选择“系统 > 权限 > 角色 > 添加角色”，填写角色名称，如“dorisrole”，在“配置资源权限”选择“待操作的集群 > Doris”，勾选“Doris管理员权限”，单击“确定”。
  - c. 选择“用户 > 添加用户”，填写用户名称，如“dorisuser”，“用户类型”选择“人机”，“密码策略”默认，输入用户密码并确认密码，关联“dorisrole”角色，单击“确定”。
  - d. 使用新建的dorisuser用户重新登录FusionInsight Manager，修改该用户初始密码。
5. 创建Doris数据库。
  - a. 使用dorisuser用户登录FusionInsight Manager页面。
  - b. 选择“集群 > 服务 > Doris”。
  - c. 在概览页面，单击“FE WebUI”右侧的超链接进入Doris WebUI登录页面。
  - d. 输入具有Doris管理权限的用户名和密码，单击“Login”。
  - e. 在“Playground”中输入以下命令创建数据库。

```
create database databasename;
```

----结束

## 步骤 2: 在 CDM 创建 Hive 和 Doris 连接

**步骤1** 创建CDM MRS Hive支持OBS的连接。

1. 在CDM集群管理界面，单击集群后的“作业管理”，选择“连接管理 > 新建连接”，进入连接器类型的选择界面。
2. 连接器类型选择“MRS Hive”，单击“下一步”，参考下表配置MRS Hive连接的参数，其它配置保持默认即可。

表 2-12 MRS Hive 连接参数

参数名	取值样例	说明
名称	hive_obs_link	连接的名称，自定义。
Manager IP	x.x.x.x	MRS Manager的浮动IP地址，可以单击输入框后的“选择”来选定已创建的MRS集群，CDM会自动填充下面的鉴权参数。 此处选择新创建的包含Hive组件的MRS集群。
认证类型	KERBEROS	访问MRS的认证类型
Hive版本	HIVE_3_X	Hive的版本。根据服务端Hive版本设置。
用户名	hiveuser	新创建的拥有Hive权限的用户。
密码	xxx	填写用户密码，此处填写为修改后的hiveuser用户密码。
开启LDAP认证	否	通过代理连接的时候，此项可配置。
OBS支持	是	开启后，在创建Hive表时，您可以指定将表存储在OBS中。

参数名	取值样例	说明
访问标识(AK)	xxxxx	<p>此处AK/SK对应的账号应具备OBS Buckets Viewer系统权限，否则会无法访问OBS并报“403 AccessDenied”错误。</p> <p>您需要先创建当前账号的访问密钥，并获得对应的AK和SK。</p> <ol style="list-style-type: none"> <li>1. 登录控制台，在用户名下拉列表中选择“我的凭证”。</li> <li>2. 进入“我的凭证”页面，选择“访问密钥 &gt; 新增访问密钥”。</li> <li>3. 单击“确定”，根据浏览器提示，保存密钥文件。密钥文件会直接保存到浏览器默认的下载文件夹中。打开名称为“credentials.csv”的文件，即可查看访问密钥AK（Access Key Id）和SK（Secret Access Key）。</li> </ol>
密钥(SK)	xxxxxx	
运行模式	EMBEDDED	<ul style="list-style-type: none"> <li>- EMBEDDED：连接实例与CDM运行在一起，该模式性能较好。</li> <li>- STANDALONE：连接实例运行在独立进程。</li> </ul>
检查Hive JDBC连通性	是	是否需要测试Hive JDBC连通。
是否使用集群配置	否	您可以通过使用集群配置，简化Hadoop连接参数配置。



图 2-37 创建 MRS Hive 连接

The screenshot shows a configuration form for creating a MRS Hive connection. The form contains the following fields and options:

- \* 名称**: hive\_obs\_link (with a [配置指南](#) link)
- \* 连接器**: Hive
- \* Hadoop类型**: MRS
- \* Manager IP**: [Redacted] (with a [选择](#) link)
- 认证类型**: KERBEROS
- \* Hive版本**: HIVE\_3\_X
- \* 用户名**: [Redacted]
- \* 密码**: [Redacted]
- \* 开启LDAP认证**: 是 / 否
- \* OBS支持**: 是 / 否
- \* 访问标识(AK)**: [Redacted]
- \* 密钥(SK)**: [Redacted]
- \* 运行模式**: EMBEDDED
- \* 检查Hive JDBC连通性**: 是 / 否
- 是否使用集群配置**: 是 / 否
- 隐藏高级属性**
- 属性配置**: + 添加
- Buttons: 取消, 测试, 保存

3. 单击“测试”，测试连接通过。
4. 单击“保存”。

## 步骤2 创建CDM MRS Doris数据连接。

1. 上传MySQL驱动。
  - a. 进入CDM主界面，单击左侧导航上的“集群管理”，选择CDM集群后的“作业管理 > 连接管理 > 驱动管理”，进入驱动管理页面。
  - b. 在“驱动管理”页面，单击MySQL驱动“建议版本”列中的资料链接，按照相应指导获取驱动文件。
  - c. 在“驱动管理”页面中，选择以下方式上传MySQL驱动。
    - 方式一：单击对应驱动名称右侧操作列的“上传”，选择本地已下载的驱动。
    - 方式二：单击对应驱动名称右侧操作列的“从sftp复制”，配置sftp连接器名称和驱动文件路径。
2. 在“集群管理”界面，单击集群后的“作业管理”，选择“连接管理 > 新建连接”，进入连接器类型的选择界面。

3. 连接器类型选择“MySQL”，单击“下一步”，参考下表配置MySQL连接的参数，其它配置保持默认即可。

表 2-13 MRS Doris 连接参数

参数名	取值样例	说明
名称	doris_link	连接的名称，自定义。
数据库服务器	x.x.x.x	Doris实例FE的IP。 可通过登录MRS Doris集群的Manager界面，选择“集群 > 服务 > Doris > 实例”，查看任一FE实例的业务IP地址获取。
端口	9030	Doris FE的查询连接端口，默认为9030。 可通过登录MRS Doris集群的Manager界面，选择“集群 > 服务 > Doris > 配置 > 全部配置”，搜索并查看“query_port”参数值获取。
数据库名称	<i>databasename</i>	已创建的数据库名称。
用户名	<b>dorisuser</b>	新创建的拥有Doris权限的用户。
密码	xxx	填写用户密码，此处填写为修改后的 <b>dorisuser</b> 用户密码。

图 2-38 创建 MRS Doris 连接

* 名称	<input type="text" value="doris_link"/>	<a href="#">配置指南</a>
* 连接器	<input type="text" value="关系数据库"/>	
数据库类型	<input type="text" value="MySQL"/>	
* 数据库服务器 ?	<input type="text" value="██████████"/>	
* 端口 ?	<input type="text" value="██████"/>	
* 数据库名称 ?	<input type="text" value="██████████"/>	
* 用户名 ?	<input type="text" value="██████████"/>	
* 密码 ?	<input type="password" value="██████████"/>	<input type="checkbox"/>
使用本地API ?	<input type="radio" value="是"/> <input checked="" type="radio" value="否"/>	
使用Agent ?	<input type="radio" value="是"/> <input checked="" type="radio" value="否"/>	
引用符号 ?	<input type="text" value="`"/>	
驱动版本	<a href="#">mysql-connector-java-5.1.48.jar</a> <a href="#">上传</a>   <a href="#">从sftp复制</a>	
<b>隐藏高级属性</b>		
单次请求行数 ?	<input type="text" value="1000"/>	
单次提交行数 ?	<input type="text" value="10000"/>	
连接属性 ?	<input type="button" value="+ 添加"/>	
单次写入行数 ?	<input type="text" value="100"/>	
<input type="button" value="× 取消"/> <input type="button" value="🔍 测试"/> <input checked="" type="button" value="💾 保存"/>		

4. 单击“测试”，测试连接通过。
5. 单击“保存”。

---结束

## 步骤 3: Hive 外表数据通过 CDM 迁移到 Doris

### 步骤1 创建Doris的Unique表。

1. 登录Doris集群中已安装好MySQL客户端的MRS Master节点，执行如下命令进入Doris数据库。

**export LIBMYSQL\_ENABLE\_CLEARTEXT\_PLUGIN=1**（若集群已启用Kerberos认证（安全模式）需执行该命令）

**mysql -u数据库登录用户 -p数据库登录用户密码 -P数据库连接端口 -hDoris FE实例IP地址**

#### 📖 说明

- 数据库连接端口为Doris FE的查询连接端口，可通过登录MRS Doris集群的Manager界面，选择“集群 > 服务 > Doris > 实例”，查看任一FE实例的业务IP地址获取。
  - Doris FE实例IP地址可通过登录MRS Doris集群的Manager界面，选择“集群 > 服务 > Doris > 配置 > 全部配置”，搜索并查看“query\_port”参数值获取。
2. 执行以下命令创建Doris的Unique表，写入数据不重复。

```
create database test;
```

```
use test;
```

```
CREATE TABLE `lineorder_u` (
```

```
  `LO_ORDERKEY` bigint NOT NULL COMMENT "订单id",
```

```
  `LO_LINENUMBER` int NOT NULL COMMENT "订单码",
```

```
  `LO_CUSTKEY` bigint NOT NULL COMMENT "客户id",
```

```
  `LO_PARTKEY` bigint NOT NULL COMMENT "商品id",
```

```
  `LO_SUPPKEY` bigint NOT NULL COMMENT "供应商id",
```

```
  `LO_ORDERDATE` date DEFAULT NULL COMMENT "订单日期",
```

```
  `LO_ORDERPRIORITY` varchar(100) DEFAULT NULL COMMENT "订单优先级",
```

```
  `LO_SHIPPRIORITY` int DEFAULT NULL COMMENT "物流优先级",
```

```
  `LO_QUANTITY` int DEFAULT NULL COMMENT "数量",
```

```
  `LO_EXTENDEDPRICE` bigint DEFAULT NULL COMMENT "扩展价格",
```

```
  `LO_ORDTOTALPRICE` bigint DEFAULT NULL COMMENT "订单总价",
```

```
  `LO_DISCOUNT` int DEFAULT NULL COMMENT "折扣",
```

```
  `LO_REVENUE` bigint DEFAULT NULL COMMENT "收入",
```

```
  `LO_SUPPLYCOST` int DEFAULT NULL COMMENT "成本",
```

```
  `LO_TAX` int DEFAULT NULL COMMENT "税款",
```

```
  `LO_COMMITDATE` date DEFAULT NULL COMMENT "承诺日期",
```

```
  `LO_SHIPMODE` varchar(100) NOT NULL COMMENT "运输方式"
```

```
)
```

```
UNIQUE KEY
```

```
(`LO_ORDERKEY`,`LO_LINENUMBER`,`LO_CUSTKEY`,`LO_PARTKEY`,`LO_SUPPKEY`,`LO_ORDERDATE`)
```

```
PARTITION BY RANGE(`LO_ORDERDATE`)
```

```
(
```

```
  PARTITION `p1992` VALUES LESS THAN ("1993-01-01"),
```

```

PARTITION `p1993` VALUES LESS THAN ("1994-01-01"),
PARTITION `p1994` VALUES LESS THAN ("1995-01-01"),
PARTITION `p1995` VALUES LESS THAN ("1996-01-01"),
PARTITION `p1996` VALUES LESS THAN ("1997-01-01"),
PARTITION `p1997` VALUES LESS THAN ("1998-01-01"),
PARTITION `p1998` VALUES LESS THAN ("1999-01-01"),
PARTITION `p1999` VALUES LESS THAN ("2000-01-01"),
PARTITION `p2000` VALUES LESS THAN ("2001-01-01"),
PARTITION `p2001` VALUES LESS THAN ("2002-01-01")
)
DISTRIBUTED BY HASH(`LO_ORDERDATE`) BUCKETS 1
PROPERTIES (
"replication_allocation" = "tag.location.default: 3"
);
    
```

**步骤2** 创建CDM作业，迁移Hive数据到Doris。

1. 进入CDM集群管理界面，单击待操作集群后的“作业管理”。

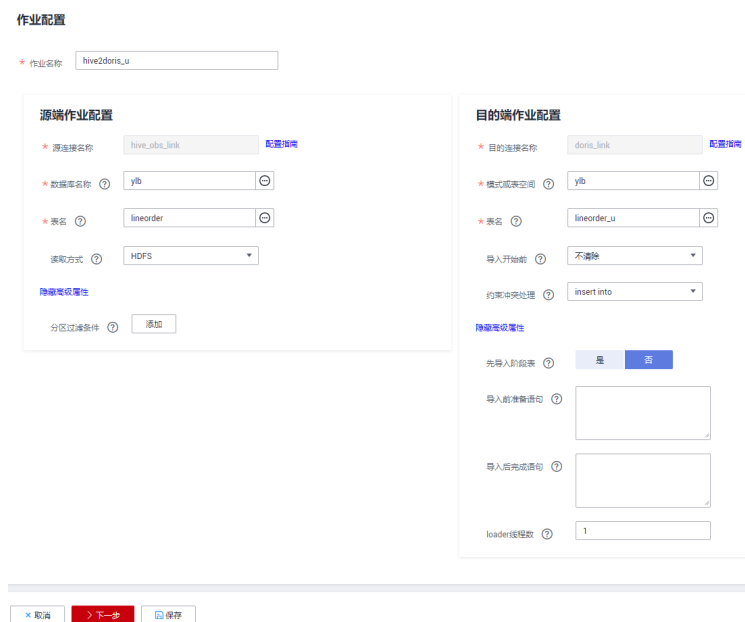


2. 选择“表/文件迁移 > 新建作业”，进入作业配置界面。参考下表配置相关参数，其它配置保持默认即可。

**表 2-14** 新建作业参数

参数名	取值样例	说明
作业名称	hive2doris_u	自定义。
源连接名称	hive_obs_link	选择已创建的Hive连接名称。
数据库名称	<i>databasename1</i>	根据实际情况配置。
表名	<i>tablename1</i>	根据实际情况配置。
目的连接名称	doris_link	选择已创建的Doris连接名称。
模式或表空间	<i>databasename2</i>	数据库名，根据实际情况配置。
表名	<i>tablename2</i>	根据实际情况配置。

图 2-39 新建作业



3. 单击“下一步”进入字段映射的操作页面。确认字段映射界面字段映射正确。
4. 单击“下一步”，确认配置界面信息。若迁移数据量较大，可适当调整“抽取并发数”参数。

图 2-40 任务配置



5. 单击“保存并运行”回到作业管理界面，并等待作业运行成功。

hive2doris\_u hive\_obs\_link... solution... 2024/01/27 15:20:38 GM... 11s 写入行数: 20 Succes... hive2doris 运行 | 历史记录 | 编辑 | 更多 ▾

**步骤3** 进入Doris客户端，查看数据迁移情况。

1. 登录Doris集群中已安装好MySQL客户端的MRS Master节点，执行如下命令进入Doris数据库。

**export LIBMYSQL\_ENABLE\_CLEARTEXT\_PLUGIN=1**（若集群已启用Kerberos认证（安全模式）需执行该命令）

**mysql -u数据库登录用户 -p数据库登录用户密码 -P数据库连接端口 -hDoris FE实例IP地址**

说明

- 数据库连接端口为Doris FE的查询连接端口，可通过登录MRS Doris集群的Manager界面，选择“集群 > 服务 > Doris > 实例”，查看任一FE实例的业务IP地址获取。
- Doris FE实例IP地址可通过登录MRS Doris集群的Manager界面，选择“集群 > 服务 > Doris > 配置 > 全部配置”，搜索并查看“query\_port”参数值获取。

2. 执行以下语句，查看数据迁移情况。

```
create database test;
use test;
select count(*) from lineorder_u;
select * from lineorder_u;
```

3. 查看数据迁移情况符合预期。

Doris的unique字段重复的数据只保留最后写入的数据。

hive表数据共20行，其中10行数据对应Doris的unique字段重复：

```
0: jdbc:hive2://7r-node-master3tebe.x0h.mrs.myhuaweicloud.com:22550/> select count(*) from lineorder;
+-----+
| count(1) |
+-----+
| 20       |
+-----+
```

```
0: jdbc:hive2://7r-node-master3tebe.x0h.mrs.myhuaweicloud.com:22550/> select * from lineorder;
+-----+
| lineorder_l1 | lineorder_l2 | lineorder_l3 | lineorder_l4 | lineorder_l5 | lineorder_l6 | lineorder_l7 | lineorder_l8 | lineorder_l9 | lineorder_l10 | lineorder_l11 | lineorder_l12 | lineorder_l13 | lineorder_l14 | lineorder_l15 | lineorder_l16 | lineorder_l17 | lineorder_l18 | lineorder_l19 | lineorder_l20 |
+-----+
| 1.2 | 1.2 | 1.2 | 1.2 | 01-01 | 1.0 | 1.2 | 1.200 | 13547402 | 1.2 | 211200 | 12280 | 1.2 | 1397-01-02 | 0280 | 1397-01-03 |
| 1.3 | 1.3 | 1.3 | 1.3 | 01-01 | 1.0 | 1.3 | 1.300 | 13547403 | 1.3 | 211200 | 12280 | 1.3 | 1397-01-03 | 0280 | 1397-01-04 |
| 1.4 | 1.4 | 1.4 | 1.4 | 01-01 | 1.0 | 1.4 | 1.400 | 13547404 | 1.4 | 211200 | 12280 | 1.4 | 1397-01-04 | 0280 | 1397-01-05 |
| 1.5 | 1.5 | 1.5 | 1.5 | 01-01 | 1.0 | 1.5 | 1.500 | 13547405 | 1.5 | 211200 | 12280 | 1.5 | 1397-01-05 | 0280 | 1397-01-06 |
| 1.6 | 1.6 | 1.6 | 1.6 | 01-01 | 1.0 | 1.6 | 1.600 | 13547406 | 1.6 | 211200 | 12280 | 1.6 | 1397-01-06 | 0280 | 1397-01-07 |
| 1.7 | 1.7 | 1.7 | 1.7 | 01-01 | 1.0 | 1.7 | 1.700 | 13547407 | 1.7 | 211200 | 12280 | 1.7 | 1397-01-07 | 0280 | 1397-01-08 |
| 1.8 | 1.8 | 1.8 | 1.8 | 01-01 | 1.0 | 1.8 | 1.800 | 13547408 | 1.8 | 211200 | 12280 | 1.8 | 1397-01-08 | 0280 | 1397-01-09 |
| 1.9 | 1.9 | 1.9 | 1.9 | 01-01 | 1.0 | 1.9 | 1.900 | 13547409 | 1.9 | 211200 | 12280 | 1.9 | 1397-01-09 | 0280 | 1397-01-10 |
| 1.1 | 1.1 | 1.1 | 1.1 | 01-01 | 1.0 | 1.1 | 1.100 | 13547410 | 1.1 | 211200 | 12280 | 1.1 | 1397-01-10 | 0280 | 1397-01-11 |
| 1.2 | 1.2 | 1.2 | 1.2 | 01-01 | 1.0 | 1.2 | 1.200 | 13547401 | 1.2 | 211200 | 12280 | 1.2 | 1397-01-01 | 0280 | 1397-01-02 |
| 1.3 | 1.3 | 1.3 | 1.3 | 01-01 | 1.0 | 1.3 | 1.300 | 13547402 | 1.3 | 211200 | 12280 | 1.3 | 1397-01-02 | 0280 | 1397-01-03 |
| 1.4 | 1.4 | 1.4 | 1.4 | 01-01 | 1.0 | 1.4 | 1.400 | 13547403 | 1.4 | 211200 | 12280 | 1.4 | 1397-01-03 | 0280 | 1397-01-04 |
| 1.5 | 1.5 | 1.5 | 1.5 | 01-01 | 1.0 | 1.5 | 1.500 | 13547404 | 1.5 | 211200 | 12280 | 1.5 | 1397-01-04 | 0280 | 1397-01-05 |
| 1.6 | 1.6 | 1.6 | 1.6 | 01-01 | 1.0 | 1.6 | 1.600 | 13547405 | 1.6 | 211200 | 12280 | 1.6 | 1397-01-05 | 0280 | 1397-01-06 |
| 1.7 | 1.7 | 1.7 | 1.7 | 01-01 | 1.0 | 1.7 | 1.700 | 13547406 | 1.7 | 211200 | 12280 | 1.7 | 1397-01-06 | 0280 | 1397-01-07 |
| 1.8 | 1.8 | 1.8 | 1.8 | 01-01 | 1.0 | 1.8 | 1.800 | 13547407 | 1.8 | 211200 | 12280 | 1.8 | 1397-01-07 | 0280 | 1397-01-08 |
| 1.9 | 1.9 | 1.9 | 1.9 | 01-01 | 1.0 | 1.9 | 1.900 | 13547408 | 1.9 | 211200 | 12280 | 1.9 | 1397-01-08 | 0280 | 1397-01-09 |
| 1.10 | 1.10 | 1.10 | 1.10 | 01-01 | 1.0 | 1.10 | 1.100 | 13547409 | 1.10 | 211200 | 12280 | 1.10 | 1397-01-09 | 0280 | 1397-01-10 |
| 1.10 | 1.10 | 1.10 | 1.10 | 01-01 | 1.0 | 1.10 | 1.100 | 13547401 | 1.10 | 211200 | 12280 | 1.10 | 1397-01-01 | 0280 | 1397-01-02 |
```

Doris的unique表数据：

```
mysql> select count(*) from lineorder_u;
+-----+
| count(*) |
+-----+
| 10       |
+-----+
```

```
mysql> select * from lineorder_u;
+-----+
| LO_ORDERKEY | LO_PARTNUMBER | LO_CUSTKEY | LO_PARTKEY | LO_SUPPKEY | LO_ORDERDATE | LO_ORDERPRIORITY | LO_SHIPPRIORITY | LO_QUANTITY | LO_EXTENDEDPRICE | LO_ORIGINALPRICE | LO_DISCOUNT | LO_REVENUE | LO_SUPP_VOUCHER | LO_TAX |
+-----+
| 1397-01-01 | 0001 | 1 | 1 | 1 | 1397-01-01 | 01-01 | 0 | 1 | 1200 | 13547401 | 1 | 211 | 210 | 1 | |
| 1397-01-01 | 0001 | 2 | 2 | 2 | 1397-01-01 | 01-01 | 0 | 1 | 1 | 1200 | 13547402 | 1 | 211 | 210 | 1 |
| 1397-01-01 | 0001 | 3 | 3 | 3 | 1397-01-01 | 01-01 | 0 | 1 | 1 | 1200 | 13547403 | 1 | 211 | 210 | 1 |
| 1397-01-01 | 0001 | 4 | 4 | 4 | 1397-01-01 | 01-01 | 0 | 1 | 1 | 1200 | 13547404 | 1 | 211 | 210 | 1 |
| 1397-01-01 | 0001 | 5 | 5 | 5 | 1397-01-01 | 01-01 | 0 | 1 | 1 | 1200 | 13547405 | 1 | 211 | 210 | 1 |
| 1397-01-01 | 0001 | 6 | 6 | 6 | 1397-01-01 | 01-01 | 0 | 1 | 1 | 1200 | 13547406 | 1 | 211 | 210 | 1 |
| 1397-01-01 | 0001 | 7 | 7 | 7 | 1397-01-01 | 01-01 | 0 | 1 | 1 | 1200 | 13547407 | 1 | 211 | 210 | 1 |
| 1397-01-01 | 0001 | 8 | 8 | 8 | 1397-01-01 | 01-01 | 0 | 1 | 1 | 1200 | 13547408 | 1 | 211 | 210 | 1 |
| 1397-01-01 | 0001 | 9 | 9 | 9 | 1397-01-01 | 01-01 | 0 | 1 | 1 | 1200 | 13547409 | 1 | 211 | 210 | 1 |
| 1397-01-01 | 0001 | 10 | 10 | 10 | 1397-01-01 | 01-01 | 0 | 1 | 1 | 1200 | 13547401 | 1 | 211200 | 12280 | 1 |
```

----结束

## 2.8.3 离线数据加载：通过 Doris Catalog 读取 Hive 外表数据并写入 Doris

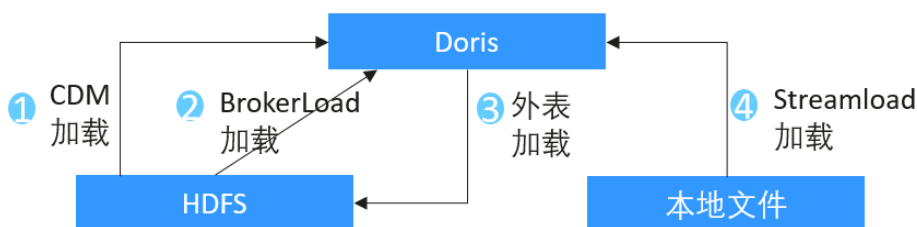
### 应用场景

通过Doris创建Catalog成功读取Hive外表数据并写入Doris，并按照Unique指定字段自动去重。

### 方案架构

离线数据可以从数据湖加载，也可以直接加载本地文件。从数据湖加载可以使用工具CDM，在没有CDM工具时，可以直接使用外表加载、BrokerLoad的方式。本地文件加载则使用Doris自带的StreamLoad工具。

图 2-41 离线数据加载



### 操作流程

- **步骤1：创建MRS Hive集群并配置**
- **步骤2：创建MRS Doris集群并配置**
- **步骤3：创建Doris Catalog并将Hive外表数据写入Doris**

#### 步骤 1：创建 MRS Hive 集群并配置

**步骤1** 创建MRS集群，详细操作请参考[快速购买MRS集群](#)。其中：

- 集群名称：自定义，例如“mrs\_test”
- 版本类型：LTS版
- 集群版本：MRS 3.3.0-LTS
- 集群组件包含：Hive等。
- Kerberos认证：开启
- 其他参数根据实际需要进行配置。

**步骤2** 集群购买完成后安装集群全量客户端，详细操作请参考[安装客户端（3.x版本）](#)。

例如客户端安装路径为“/opt/newclient”。

**步骤3** 配置Hive集群存算分离，详细操作请参考[配置MRS集群通过IAM委托对接OBS](#)。

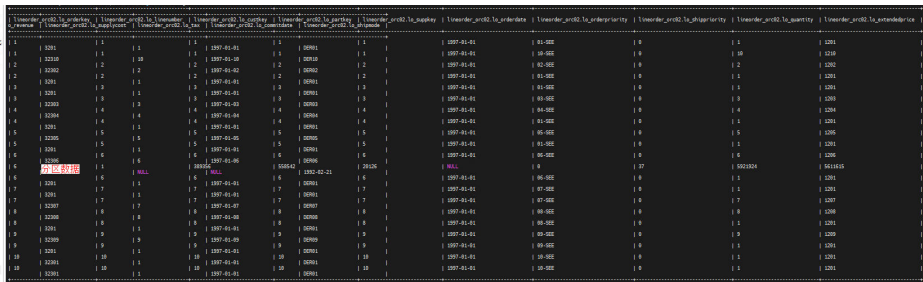
**步骤4** 已存在Hive外表（数据格式仅限orc格式），如wxk.lineorder\_orc02。且存在一定量数据。

建表语句参考：



```
CREATE TABLE wxk.`lineorder_orc02`(  
  `lo_orderkey` bigint,  
  `lo_linenumber` int,  
  `lo_custkey` bigint,  
  `lo_partkey` bigint,  
  `lo_suppkey` bigint,  
  `lo_orderdate` date,  
  `lo_orderpriority` varchar(300),  
  `lo_shippriority` int,  
  `lo_quantity` int,  
  `lo_extendedprice` bigint,  
  `lo_ordtotalprice` bigint,  
  `lo_discount` int,  
  `lo_revenue` bigint,  
  `lo_supplycost` int,  
  `lo_tax` int,  
  `lo_commitdate` date,  
  `lo_shipmode` varchar(300))  
ROW FORMAT SERDE  
  'org.apache.hadoop.hive ql.io.orc.OrcSerde'  
STORED AS INPUTFORMAT  
  'org.apache.hadoop.hive ql.io.orc.OrcInputFormat'  
OUTPUTFORMAT  
  'org.apache.hadoop.hive ql.io.orc.OrcOutputFormat'  
LOCATION  
  'obs://test-wxk/doristest'  
TBLPROPERTIES (  
  'bucketing_version'='2',  
  'orc.compress'='ZLIB',  
  'transient_lastDdlTime'='1705923234') ;
```

准备数据如下:



The screenshot shows a terminal window displaying a large table of data. The table has multiple columns, including what appears to be a primary key, several numerical columns, and date columns. The data is organized in a grid-like format, typical of a data export or a table view in a database interface. The terminal background is dark, and the text is light-colored, making it easy to read.

---结束

## 步骤 2: 创建 MRS Doris 集群并配置

**步骤1** 创建MRS Doris集群，详细操作请参考[快速购买MRS集群](#)。其中：

- 集群名称：自定义，例如“mrs\_doris”
- 版本类型：LTS版
- 集群版本：MRS 3.3.0-LTS
- 组件选择：Doris集群。
- Kerberos认证：开启
- 其他参数根据实际需要进行配置。

**步骤2** 集群购买完成后安装集群客户端，详细操作请参考[安装客户端（3.x版本）](#)。

例如客户端安装路径为“/opt/dorisclient”。

**步骤3** 在Doris集群安装MySQL客户端，详细操作请参考[使用MySQL客户端连接Doris](#)。

**步骤4** 创建拥有Doris管理权限的人机用户（例如用户名为dorisuser），并修改初始密码。

1. 登录Doris集群Manager页面，详细操作请参考[访问MRS集群Manager](#)。
2. 选择“系统 > 权限 > 角色 > 添加角色”，填写角色名称，如“dorisrole”，在“配置资源权限”选择“待操作的集群 > Doris”，勾选“Doris管理员权限”，单击“确定”。
3. 选择“用户 > 添加用户”，填写用户名称，如“dorisuser”，“用户类型”选择“人机”，“密码策略”默认，输入用户密码并确认密码，关联“dorisrole”角色，单击“确定”。
4. 使用新建的dorisuser用户重新登录FusionInsight Manager，修改该用户初始密码。

**步骤5** 配置MRS Hive集群与Doris集群互信。

1. 配置MRS Hive集群与Doris集群互信，详细操作请参考[集群互信管理](#)。
2. 配置后验证。
  - a. 以root用户登录Doris集群客户端，执行以下命令加载环境变量。  
**source 客户端安装路径/bigdata\_env**
  - b. 执行以下命令认证Hive集群用户。  
**kinit Hive集群业务用户@Hive集群系统域名**  
输入Hive集群用户相应密码，认证成功即为互信成功。

 说明

- Hive集群用户：可以自行创建也可使用默认的系统用户，例如hive，用户默认密码请参考[MRS集群用户账号一览表](#)。
- Hive集群系统域名：可以登录Hive集群Manager页面，选择“系统 > 权限 > 域和互信”，查看“本端域”参数获取。

----结束

### 步骤 3: 创建 Doris Catalog 并将 Hive 外表数据写入 Doris

**步骤1** 登录Doris集群中已安装好MySQL客户端的MRS Master节点，执行如下命令进入Doris数据库。

`export LIBMYSQL_ENABLE_CLEARTEXT_PLUGIN=1`（若集群已启用Kerberos认证（安全模式）需执行该命令）

`mysql -u数据库登录用户 -p数据库登录用户密码 -P数据库连接端口 -hDoris FE实例IP地址`

 说明

- 数据库连接端口为Doris FE的查询连接端口，可通过登录MRS Doris集群的Manager界面，选择“集群 > 服务 > Doris > 实例”，查看任一FE实例的业务IP地址获取。
- Doris FE实例IP地址可通过登录MRS Doris集群的Manager界面，选择“集群 > 服务 > Doris > 配置 > 全部配置”，搜索并查看“query\_port”参数值获取。

**步骤2** 创建Catalog。

例如执行以下命令，相关参数说明见[表2-15](#)，参数取值请以时间为准。

`CREATE CATALOG hive_catalog_obs11 PROPERTIES (`

`'type'='hms',`

`'hive.metastore.uris' = 'thrift://192.168.xx.xx:21088',`

`'hive.metastore.sasl.enabled' = 'true',`

`'hive.server2.thrift.sasl.qop' = 'auth-conf',`

`'hive.server2.authentication' = 'KERBEROS',`

`'dfs.nameservices'='hacluster',`

`'dfs.ha.namenodes.hacluster'='24,25',`

`'dfs.namenode.rpc-address.hacluster.24'='主NameNodeIP地址:RPC通信端口',`

`'dfs.namenode.rpc-address.hacluster.25'='主NameNodeIP地址:RPC通信端口',`

`'dfs.client.failover.proxy.provider.hacluster'='org.apache.hadoop.hdfs.server.namenode.ha.ConfiguredFailoverProxyProvider',`

`'hive.version' = '3.1.0',`

`'yarn.resourcemanager.address' = '192.168.xx.xx:26004',`

`'yarn.resourcemanager.principal' = 'mapred/hadoop.hadoop.com@HADOOP.COM',`

```

'hive.metastore.kerberos.principal' = 'hive/
hadoop.hadoop.com@HADOOP.COM',

'hadoop.security.authentication' = 'kerberos',

'hadoop.kerberos.keytab' = '/opt/Bigdata/FusionInsight_Doris_8.3.0/install/
FusionInsight-Doris-1.2.3/doris-fe/bin/doris.keytab',

'hadoop.kerberos.principal' = 'doris/hadoop.hadoop.com@hadoop.com',

'java.security.krb5.conf' = '/opt/Bigdata/
FusionInsight_BASE_8.3.0/1_2_KerberosClient/etc/krb5.conf',

'AWS_ACCESS_KEY' = 'AK信息',

'AWS_SECRET_KEY' = 'SK信息',

'AWS_ENDPOINT' = '集群所在地区的终端节点',

'AWS_REGION' = '集群所在region',

'hadoop.rpc.protection' = 'privacy'

);

```

表 2-15 参数说明

参数	取值样例	参数描述
hive.metastore.uris	thrift:// 192.168.xx.xx:210 88	Hive MetaStore的URL，格式为“thrift://<Hive MetaStore的IP地址>:<端口号>”，支持多个值，以逗号分隔。
hive.metastore.sasl.enabled	true	MetaStore的管理权限开关。值为“true”。
hive.server2.thrift.sasl.qop	auth-conf	HiveServer2和客户端交互是否加密传输，值为“auth-conf”。
hive.server2.authentication	KERBEROS	访问HiveServer的安全认证方式，值为“KERBEROS”。
dfs.nameservices	hacluster	集群NameService名称。 可在NameNode所在节点的“\${BIGDATA_HOME}/FusionInsight_HD_*/1_*_NameNode/etc”目录下的“hdfs-site.xml”中查找该配置项的值。
dfs.ha.namenodes.hacluster	24,25	集群NameService前缀，包含两个值。 可在NameNode所在节点的“\${BIGDATA_HOME}/FusionInsight_HD_*/1_*_NameNode/etc”目录下的“hdfs-site.xml”中查找该配置项的值。

参数	取值样例	参数描述
dfs.namenode.rpc-address.hacluster.xx1	-	主NameNode的RPC通信地址。 可在NameNode所在节点的“\${BIGDATA_HOME}/FusionInsight_HD_*/1_*_NameNode/etc”目录下的“hdfs-site.xml”中查找该配置项的值。xx为“dfs.ha.namenodes.hacluster”参数的值。
dfs.namenode.rpc-address.hacluster.xx2	-	备NameNode的RPC通信地址。 可在NameNode所在节点的“\${BIGDATA_HOME}/FusionInsight_HD_*/1_*_NameNode/etc”目录下的“hdfs-site.xml”中查找该配置项的值。xx为“dfs.ha.namenodes.hacluster”参数的值。
hive.version	3.1.0	Hive版本。 可在Manager界面，选择“集群 > 服务 > Hive”，在概览页面查看“版本”获取。
yarn.resourcemanager.address	192.168.xx.xx	主ResourceManager实例的IP地址。 可在Manager界面，选择“集群 > 服务 > Yarn > 实例”，查看主ResourceManager实例的业务IP地址。
yarn.resourcemanager.principal	mapred/hadoop.hadoop.com@HADOOP.COM	访问Yarn集群的Principal。 可登录Manager页面，选择“系统 > 权限 > 域和互信”，查看“本端域”参数获取。
hive.metastore.kerberos.principal	hive/hadoop.hadoop.com@HADOOP.COM	访问Hive集群的Principal，值为“hive/hadoop.hadoop.com@HADOOP.COM”。
hadoop.security.authentication	kerberos	访问Hadoop的安全认证方式，值为“kerberos”。
hadoop.kerberos.keytab	/opt/Bigdata/FusionInsight_Doris_8.3.0/install/FusionInsight-Doris-1.2.3/doris-fe/bin/doris.keytab	访问Hadoop集群的keytab文件所在的具体路径，值为“\${BIGDATA_HOME}/FusionInsight_Doris_*/install/FusionInsight-Doris-*/doris-be/bin/doris.keytab”。

参数	取值样例	参数描述
hadoop.kerberos.principal	doris/ hadoop.hadoop.com@HADOOP.COM	访问Hadoop集群的Principal，值为“doris/hadoop.hadoop.com@HADOOP.COM”。 <b>说明</b> 该参数值的系统域名要严格区分大小写。如系统域名查询为11B1A87B_48C8_xxx_CC4039.COM时，对应配置应为： 'hadoop.kerberos.principal' = 'doris/hadoop.11b1a87b_48c8_451f_a283_dd0be9cc4039.com@11B1A87B_48C8_xxx_CC4039.COM',
java.security.krb5.conf	/opt/Bigdata/ FusionInsight_BASE_8.3.0/1_2_KerberosClient/etc/ krb5.conf	krb5文件所在的具体路径，值为“\${BIGDATA_HOME}/FusionInsight_BASE_*/1_*_KerberosClient/etc/krb5.conf”。
AWS_ACCESS_KEY	xxx	AK、SK信息。可以通过以下操作以下获取： 1. 登录控制台，在用户名下拉列表中选择“我的凭证”。 2. 进入“我的凭证”页面，选择“访问密钥 > 新增访问密钥”。 3. 单击“确定”，根据浏览器提示，保存密钥文件。密钥文件会直接保存到浏览器默认的下载文件夹中。打开名称为“credentials.csv”的文件，即可查看访问密钥AK（Access Key Id）和SK（Secret Access Key）。
AWS_SECRET_KEY	xxx	
AWS_ENDPOINT	xxx	登录“对象存储服务 OBS”管理控制台，单击“并行文件系统”，单击Hive表所在的OBS并行文件系统名称，在概览界面查看“Endpoint”参数值，该值为创建Catalog时设置AWS_ENDPOINT参数的值。
AWS_REGION	xxx	可在 <a href="#">地区</a> 和 <a href="#">终端节点</a> 获取。
hadoop.rpc.protection	AWS_SECRET_KEY	设置Hadoop中各模块的RPC通道是否加密，默认为“privacy”。 可在Manager界面，选择“集群 > 服务 > HDFS > 配置”，搜索“hadoop.rpc.protection”获取。

**步骤3** 执行以下命令查询Hive表。

- 执行以下命令查询Catalogs：

**show catalogs;**

- 执行以下命令查询Catalog下面的数据库。  
**show databases from hive\_catalog;**
- 执行以下命令切换到Catalog下，再进入到数据库中。  
**switch hive\_catalog;**  
**use default;**
- 查询Catalog中某个库的所有表。  
**show tables from `hive\_catalog`.`default`;**
- 查询指定表。  
**select \* from `hive\_catalog`.`default`.`test\_table`;**
- 执行以下命令查看表的Schema。  
**DESC test\_table;**
- 新建或操作Hive表后，需要在Doris中执行刷新。  
**refresh catalog hive\_catalog;**

例如执行后结果如下，表示查询成功：

```
mysql> switch hive_catalog_obs11;
Query OK, 0 rows affected (0.01 sec)

mysql> use wxk;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

shoDatabase changed
mysql> show tables;
+-----+
| Tables_in_wxk |
+-----+
| test_hive     |
+-----+
1 row in set (0.00 sec)

mysql> refresh catalog hive_catalog_obs11;
Query OK, 0 rows affected (0.01 sec)

mysql> show tables;
+-----+
| Tables_in_wxk |
+-----+
| lineorder_orc02 |
| test_hive     |
+-----+
2 rows in set (1.65 sec)

mysql> refresh catalog hive_catalog_obs11;
Query OK, 0 rows affected (0.00 sec)

mysql> show tables;
+-----+
| Tables_in_wxk |
+-----+
| lineorder_orc02 |
```

**步骤4** 执行以下命令创建Doris表（使用引擎为 UNIQUE）

```
CREATE TABLE wxk.`lineorder_u` (  
  `LO_ORDERKEY` bigint NOT NULL COMMENT "订单id",
```

```
`LO_LINENUMBER` int NOT NULL COMMENT "订单码",
`LO_CUSTKEY` bigint NOT NULL COMMENT "客户id",
`LO_PARTKEY` bigint NOT NULL COMMENT "商品id",
`LO_SUPPKEY` bigint NOT NULL COMMENT "供应商id",
`LO_ORDERDATE` date DEFAULT NULL COMMENT "订单日期",
`LO_ORDERPRIORITY` varchar(100) DEFAULT NULL COMMENT "订单优先级",
`LO_SHIPPRIORITY` int DEFAULT NULL COMMENT "物流优先级",
`LO_QUANTITY` int DEFAULT NULL COMMENT "数量",
`LO_EXTENDEDPRICE` bigint DEFAULT NULL COMMENT "扩展价格",
`LO_ORDTOTALPRICE` bigint DEFAULT NULL COMMENT "订单总价",
`LO_DISCOUNT` int DEFAULT NULL COMMENT "折扣",
`LO_REVENUE` bigint DEFAULT NULL COMMENT "收入",
`LO_SUPPLYCOST` int DEFAULT NULL COMMENT "成本",
`LO_TAX` int DEFAULT NULL COMMENT "税款",
`LO_COMMITDATE` date DEFAULT NULL COMMENT "承诺日期",
`LO_SHIPMODE` varchar(100) NOT NULL COMMENT "运输方式"
)
UNIQUE KEY
(`LO_ORDERKEY`,`LO_LINENUMBER`,`LO_CUSTKEY`,`LO_PARTKEY`,`LO_SUPPKEY`,`LO_ORDERDATE`)
PARTITION BY RANGE(`LO_ORDERDATE`)
(
PARTITION `p1992` VALUES LESS THAN ("1993-01-01"),
PARTITION `p1993` VALUES LESS THAN ("1994-01-01"),
PARTITION `p1994` VALUES LESS THAN ("1995-01-01"),
PARTITION `p1995` VALUES LESS THAN ("1996-01-01"),
PARTITION `p1996` VALUES LESS THAN ("1997-01-01"),
PARTITION `p1997` VALUES LESS THAN ("1998-01-01"),
PARTITION `p1998` VALUES LESS THAN ("1999-01-01"),
PARTITION `p1999` VALUES LESS THAN ("2000-01-01"),
PARTITION `p2000` VALUES LESS THAN ("2001-01-01"),
PARTITION `p2001` VALUES LESS THAN ("2002-01-01")
)
DISTRIBUTED BY HASH(`LO_ORDERDATE`) BUCKETS 1
```



```
PROPERTIES (  
"replication_allocation" = "tag.location.default: 3"  
);
```

例如执行后结果如下，表示创建Doris表成功：

```
mysql> use wxk;  
Reading table information for completion of table and column names  
You can turn off this feature to get a quicker startup with -A  
  
Database changed  
mysql> show tables;  
+-----+  
| Tables_in_wxk |  
+-----+  
| lineorder_u   |  
+-----+  
1 row in set (0.00 sec)  
  
mysql>
```

**步骤5** 执行以下命令读取Hive外表数据写入Doris。

```
insert into internal.wxk.lineorder_u select * from hive_catalog.wxk.lineorder02
```

例如执行后结果如下，数据写入Doris成功，且按照UNIQUE KEY指定字段自动去除重复数据。

```
mysql> select * from lineorder_u  
+-----+  
| LD_ORDERKEY | LD_LINEANUMBER | LD_CUSTKEY | LD_PARTKEY | LD_SUPPKEY | LD_ORDERDATE | LD_ORDERPRIORITY | LD_SHIPPRIORITY | LD_QUANTITY | LD_EXTENDEDPRICE | LD_ORIGINALPRICE | LD_DISCOUNT | LD_REVENUE | LD_SUPPLYCOST | LD_TAX | LD_COMMENT |  
+-----+  
| 0 | 1 | 399956 | 558542 | 20120 | NULL | 0 | 37 | 5921924 | 5611655 | 8 | 5448370 | 96031 | 3 | NULL | NULL |  
+-----+  
| 1 | 1992-02-21 | 1 | 1 | 1 | 1 | 1997-01-01 | 01-SEE | 0 | 1 | 1201 | 18547401 | 1 | 231 | 3201 | 1 | 1997-01-01 |  
+-----+  
| 2 | 1992 | 2 | 2 | 2 | 2 | 1997-01-01 | 01-SEE | 0 | 1 | 1201 | 18547401 | 1 | 231 | 3201 | 1 | 1997-01-01 |  
+-----+  
| 3 | 1992 | 3 | 3 | 3 | 3 | 1997-01-01 | 01-SEE | 0 | 1 | 1201 | 18547401 | 1 | 231 | 3201 | 1 | 1997-01-01 |  
+-----+  
| 4 | 1992 | 4 | 4 | 4 | 4 | 1997-01-01 | 01-SEE | 0 | 1 | 1201 | 18547401 | 1 | 231 | 3201 | 1 | 1997-01-01 |  
+-----+  
| 5 | 1992 | 5 | 5 | 5 | 5 | 1997-01-01 | 01-SEE | 0 | 1 | 1201 | 18547401 | 1 | 231 | 3201 | 1 | 1997-01-01 |  
+-----+  
| 6 | 1992 | 6 | 6 | 6 | 6 | 1997-01-01 | 06-SEE | 0 | 1 | 1201 | 18547401 | 1 | 231 | 3201 | 1 | 1997-01-01 |  
+-----+  
| 7 | 1992 | 7 | 7 | 7 | 7 | 1997-01-01 | 07-SEE | 0 | 1 | 1201 | 18547401 | 1 | 231 | 3201 | 1 | 1997-01-01 |  
+-----+  
| 8 | 1992 | 8 | 8 | 8 | 8 | 1997-01-01 | 08-SEE | 0 | 1 | 1201 | 18547401 | 1 | 231 | 3201 | 1 | 1997-01-01 |  
+-----+  
| 9 | 1992 | 9 | 9 | 9 | 9 | 1997-01-01 | 09-SEE | 0 | 1 | 1201 | 18547401 | 1 | 231 | 3201 | 1 | 1997-01-01 |  
+-----+  
| 10 | 1992 | 10 | 10 | 10 | 10 | 1997-01-01 | 10-SEE | 0 | 1 | 1201 | 18547401 | 1 | 2313301 | 32301 | 1 | 1997-01-01 |  
+-----+  
| 11 | 1992 | 11 | 11 | 11 | 11 | 1997-01-01 | 11-SEE | 0 | 1 | 1201 | 18547401 | 1 | 2313301 | 32301 | 1 | 1997-01-01 |
```

----结束

## 2.8.4 离线数据加载：通过 Doris Catalog 读取 RDS-MySQL 数据并写入 Doris

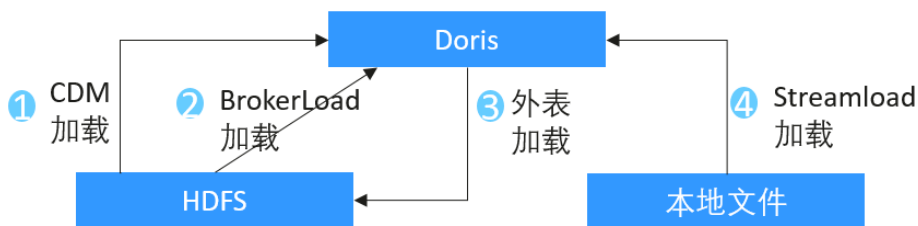
### 应用场景

通过Doris创建Catalog成功读取RDS-MySQL数据并写入Doris，并按照Unique指定字段自动去重。

### 方案架构

离线数据可以从数据湖加载，也可以直接加载本地文件。从数据湖加载可以使用工具CDM，在没有CDM工具时，可以直接使用外表加载、BrokerLoad的方式。本地文件加载则使用Doris自带的StreamLoad工具。

图 2-42 离线数据加载



## 操作流程

- **步骤1: 创建RDS for MySQL实例、创建数据表**
- **步骤2: 创建MRS Doris集群并配置**
- **步骤3: 创建Doris Catalog**

### 步骤 1: 创建 RDS for MySQL 实例、创建数据表

**步骤1** 创建RDS for MySQL实例，详细操作请参考[购买RDS for MySQL实例](#)。

**步骤2** 已创建相应数据表，如lineorder\_noid。且存在一定量数据。

建表语句参考：

```
CREATE TABLE `lineorder_noid`(  
  `lo_orderkey` bigint,  
  `lo_linenumber` int,  
  `lo_custkey` bigint,  
  `lo_partkey` bigint,  
  `lo_suppkey` bigint,  
  `lo_orderdate` date,  
  `lo_orderpriority` varchar(100),  
  `lo_shippriority` int,  
  `lo_quantity` int,  
  `lo_extendedprice` bigint,  
  `lo_ordtotalprice` bigint,  
  `lo_discount` int,  
  `lo_revenue` bigint,  
  `lo_supplycost` int,  
  `lo_tax` int,  
  `lo_commitdate` date,  
  `lo_shipmode` varchar(100));
```

数据准备如下：

```
mysql> SELECT * FROM ORDERPRTS ORDER BY LO_ORDERKEY ASC;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| LO_ORDERKEY | LO_LINENUMBER | LO_CUSTKEY | LO_PARTKEY | LO_SUPPKEY | LO_ORDERDATE | LO_ORDERPRIORITY | LO_SHIPPRIORITY | LO_QUANTITY | LO_EXTENDEDPRICE | LO_ORDTOTALPRICE | LO_DISCOUNT | LO_REVENUE |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 3201 | 1 | 1997-01-01 | DER01 | 1 | 1997-01-01 | 01-SEE | 0 | 1 | 1201 | 18547401 | 1 | 231 |
| 32310 | 10 | 1997-01-10 | DER10 | 1 | 1997-01-01 | 10-SEE | 0 | 10 | 1210 | 18547410 | 10 | 2311310 |
| 32302 | 2 | 1997-01-02 | DER02 | 2 | 1997-01-01 | 02-SEE | 0 | 2 | 1202 | 18547402 | 2 | 2311302 |
| 3201 | 1 | 1997-01-01 | DER01 | 3 | 1997-01-01 | 03-SEE | 0 | 3 | 1201 | 18547401 | 3 | 231 |
| 32303 | 3 | 1997-01-03 | DER03 | 3 | 1997-01-01 | 03-SEE | 0 | 3 | 1203 | 18547403 | 3 | 2311303 |
| 3201 | 1 | 1997-01-01 | DER01 | 4 | 1997-01-01 | 04-SEE | 0 | 4 | 1201 | 18547401 | 4 | 231 |
| 32304 | 4 | 1997-01-04 | DER04 | 4 | 1997-01-01 | 04-SEE | 0 | 4 | 1204 | 18547404 | 4 | 2311304 |
| 3201 | 1 | 1997-01-01 | DER01 | 5 | 1997-01-01 | 05-SEE | 0 | 5 | 1201 | 18547401 | 5 | 231 |
| 32305 | 5 | 1997-01-05 | DER05 | 5 | 1997-01-01 | 05-SEE | 0 | 5 | 1205 | 18547405 | 5 | 2311305 |
| 3201 | 1 | 1997-01-01 | DER01 | 6 | 1997-01-01 | 06-SEE | 0 | 6 | 1201 | 18547401 | 6 | 231 |
| 32306 | 6 | 1997-01-06 | DER06 | 6 | 1997-01-01 | 06-SEE | 0 | 6 | 1206 | 18547406 | 6 | 2311306 |
| 3201 | 1 | 1997-01-01 | DER01 | 7 | 1997-01-01 | 07-SEE | 0 | 7 | 1201 | 18547401 | 7 | 231 |
| 32307 | 7 | 1997-01-07 | DER07 | 7 | 1997-01-01 | 07-SEE | 0 | 7 | 1207 | 18547407 | 7 | 2311307 |
| 3201 | 1 | 1997-01-01 | DER01 | 8 | 1997-01-01 | 08-SEE | 0 | 8 | 1201 | 18547401 | 8 | 231 |
| 32308 | 8 | 1997-01-08 | DER08 | 8 | 1997-01-01 | 08-SEE | 0 | 8 | 1208 | 18547408 | 8 | 2311308 |
| 3201 | 1 | 1997-01-01 | DER01 | 9 | 1997-01-01 | 09-SEE | 0 | 9 | 1201 | 18547401 | 9 | 231 |
| 32309 | 9 | 1997-01-09 | DER09 | 9 | 1997-01-01 | 09-SEE | 0 | 9 | 1209 | 18547409 | 9 | 2311309 |
| 3201 | 1 | 1997-01-01 | DER01 | 10 | 1997-01-01 | 10-SEE | 0 | 10 | 1201 | 18547401 | 10 | 231 |
| 32310 | 10 | 1997-01-10 | DER10 | 10 | 1997-01-01 | 10-SEE | 0 | 10 | 1201 | 18547401 | 10 | 2311310 |
| 32301 | 1 | 1997-01-01 | DER01 | 10 | 1997-01-01 | 10-SEE | 0 | 10 | 1201 | 18547401 | 10 | 2311301 |
```

----结束

## 步骤 2：创建 MRS Doris 集群并配置

**步骤1** 创建MRS Doris集群，详细操作请参考[快速购买MRS集群](#)。其中：

- 集群名称：自定义，例如 “mrs\_doris”
- 版本类型：LTS版
- 集群版本：MRS 3.3.0-LTS
- 组件选择：Doris集群。
- Kerberos认证：开启
- 其他参数根据实际需要进行配置。

**步骤2** 集群购买完成后安装集群客户端，详细操作请参考[安装客户端（3.x版本）](#)。

例如客户端安装路径为 “/opt/dorisclient”。

**步骤3** 在Doris集群安装MySQL客户端，详细操作请参考[使用MySQL客户端连接Doris](#)。

**步骤4** 创建拥有Doris管理权限的人机用户（例如用户名为dorisuser），并修改初始密码。

1. 登录Doris集群Manager页面，详细操作请参考[访问MRS集群Manager](#)。
2. 选择“系统 > 权限 > 角色 > 添加角色”，填写角色名称，如“dorisrole”，在“配置资源权限”选择“待操作的集群 > Doris”，勾选“Doris管理员权限”，单击“确定”。
3. 选择“用户 > 添加用户”，填写用户名称，如“dorisuser”，“用户类型”选择“人机”，“密码策略”默认，输入用户密码并确认密码，关联“dorisrole”角色，单击“确定”。
4. 使用新建的dorisuser用户重新登录FusionInsight Manager，修改该用户初始密码。

----结束

## 步骤 3：创建 Doris Catalog

**步骤1** 登录Doris集群中已安装好MySQL客户端的MRS Master节点，执行如下命令进入Doris数据库。

**export LIBMYSQL\_ENABLE\_CLEARTEXT\_PLUGIN=1**（若集群已启用Kerberos认证（安全模式）需执行该命令）

**mysql -u数据库登录用户 -p数据库登录用户密码 -P数据库连接端口 -hDoris FE实例IP地址**

#### 📖 说明

- 数据库连接端口为Doris FE的查询连接端口，可通过登录MRS Doris集群的Manager界面，选择“集群 > 服务 > Doris > 实例”，查看任一FE实例的业务IP地址获取。
- Doris FE实例IP地址可通过登录MRS Doris集群的Manager界面，选择“集群 > 服务 > Doris > 配置 > 全部配置”，搜索并查看“query\_port”参数值获取。

#### 步骤2 创建Catalog。

例如执行以下命令，相关参数说明见[表2-16](#)，参数取值请以时间为准。

```
CREATE CATALOG jdbc_mysql PROPERTIES (  
'type'='jdbc',  
'user'='用户名',  
'password'='用户密码',  
'jdbc_url'='jdbc:mysql://MySQL实例IP:3306/数据库名',  
'driver_url'='file://预置的MySQL驱动包绝对路径',  
'driver_class'='com.mysql.cj.jdbc.Driver'  
);
```

表 2-16 参数说明

参数	取值样例	参数描述
user	user	MySQL数据库用户名。
password	password	MySQL数据库用户密码。
jdbc_url	jdbc:mysql:// MySQL实例 IP:3306/数据库名	jdbc:mysql://MySQL实例IP:3306/数据库名

参数	取值样例	参数描述
driver_url	<b>file://预置的MySQL驱动包绝对路径</b>	driver_url可以通过以下三种方式指定： <ul style="list-style-type: none"> <li>• 文件名。如“mysql-connector-java-*.jar”。需将Jar包预先存放在FE和BE部署目录的“jdbc_drivers/”目录下。系统会自动在这个该目录下寻找。该目录的位置，也可以由“fe.conf”和“be.conf”中的“jdbc_drivers_dir”配置修改。</li> <li>• 本地绝对路径。如“file:///path/to/mysql-connector-java-*.jar”。需将Jar包预先存放在所有FE/BE节点指定的路径下并修改所属用户/组（<b>chown omm:wheel mysql-connector-java-*.jar</b>）。</li> <li>• HTTP地址。如“https://doris-community-test-1308700295.cos.ap-hongkong.myqcloud.com/jdbc_driver/mysql-connector-java-8.0.25.jar”。系统会从这个HTTP地址下载Driver文件。仅支持无认证的HTTP服务。</li> </ul>
driver_class	<b>com.mysql.cj.jdbc.Driver</b>	-

**步骤3** 执行以下命令查询MySQL表。

- 执行以下命令查询Catalogs：  
**show catalogs;**
- 执行以下命令查询catalog下面的库：  
**show databases from jdbc\_mysql;**
- 执行以下命令切换到Catalog下，再进入到数据库中：  
**switch jdbc\_mysql;**  
**use default;**
- 查询Catalog中某个库的所有表：  
**show tables from `jdbc\_mysql`.`default`;**
- 查询指定表：  
**select \* from `jdbc\_mysql`.`default`.`test\_table`;**
- 执行以下命令查看表的Schema：  
**DESC test\_table;**
- 新建或操作MySQL表后，需要在Doris中执行刷新：  
**refresh catalog jdbc\_mysql;**

例如执行后结果如下，表示查询成功：

```
mysql> switch jdbc_mysql;
Query OK, 0 rows affected (0.00 sec)

mysql> show databases;
ERROR 1105 (HY000): errCode = 2, detailMessage = Syntax error in line 1:
show databases
^
Encountered: IDENTIFIER
Expected

mysql> show databases;
+-----+
| Database |
+-----+
| bigdata  |
| cdc      |
| cdl      |
| cdl-test |
| cdl_yu   |
| cdl_zy   |
| db_wzqtest |
| db_zhctest |
| information_schema |
| loadertest |
| mysql    |
| performance_schema |
| sb       |
| sys      |
| xiaohongshu |
| xingneng |
| xntest   |
| zytest   |
+-----+
18 rows in set (0.06 sec)

mysql> use cdl_yu;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> show tables;
+-----+
| Tables_in_cdl_yu |
+-----+
| customer |
| lin       |
| lineorder |
| lineorder_noid |
| part      |
| part1     |
| partlin   |
| supplier  |
+-----+
8 rows in set (0.00 sec)
```

**步骤4** 执行以下命令创建Doris表（使用引擎为 UNIQUE）

```
CREATE TABLE wxk.`lineorder_mysql_u` (
  `LO_ORDERKEY` bigint NOT NULL COMMENT "订单id",
  `LO_LINENUMBER` int NOT NULL COMMENT "订单码",
  `LO_CUSTKEY` bigint NOT NULL COMMENT "客户id",
  `LO_PARTKEY` bigint NOT NULL COMMENT "商品id",
  `LO_SUPPKEY` bigint NOT NULL COMMENT "供应商id",
  `LO_ORDERDATE` date DEFAULT NULL COMMENT "订单日期",
  `LO_ORDERPRIORITY` varchar(100) DEFAULT NULL COMMENT "订单优先级",
  `LO_SHIPPRIORITY` int DEFAULT NULL COMMENT "物流优先级",
  `LO_QUANTITY` int DEFAULT NULL COMMENT "数量",
  `LO_EXTENDEDPRICE` bigint DEFAULT NULL COMMENT "扩展价格",
```

```
`LO_ORDTOTALPRICE` bigint DEFAULT NULL COMMENT "订单总价",
`LO_DISCOUNT` int DEFAULT NULL COMMENT "折扣",
`LO_REVENUE` bigint DEFAULT NULL COMMENT "收入",
`LO_SUPPLYCOST` int DEFAULT NULL COMMENT "成本",
`LO_TAX` int DEFAULT NULL COMMENT "税款",
`LO_COMMITDATE` date DEFAULT NULL COMMENT "承诺日期",
`LO_SHIPMODE` varchar(100) NOT NULL COMMENT "运输方式"
)
UNIQUE KEY
(`LO_ORDERKEY`,`LO_LINENUMBER`,`LO_CUSTKEY`,`LO_PARTKEY`,`LO_SUPPKY`,`LO_ORDERDATE`)
PARTITION BY RANGE(`LO_ORDERDATE`)
(
PARTITION `p1992` VALUES LESS THAN ("1993-01-01"),
PARTITION `p1993` VALUES LESS THAN ("1994-01-01"),
PARTITION `p1994` VALUES LESS THAN ("1995-01-01"),
PARTITION `p1995` VALUES LESS THAN ("1996-01-01"),
PARTITION `p1996` VALUES LESS THAN ("1997-01-01"),
PARTITION `p1997` VALUES LESS THAN ("1998-01-01"),
PARTITION `p1998` VALUES LESS THAN ("1999-01-01"),
PARTITION `p1999` VALUES LESS THAN ("2000-01-01"),
PARTITION `p2000` VALUES LESS THAN ("2001-01-01"),
PARTITION `p2001` VALUES LESS THAN ("2002-01-01")
)
DISTRIBUTED BY HASH(`LO_ORDERDATE`) BUCKETS 1
PROPERTIES (
"replication_allocation" = "tag.location.default: 3"
);
```

例如执行show tables;后结果如下，表示创建Doris表成功：

```
mysql> use wxk;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> show tables;
+-----+
| Tables_in_wxk |
+-----+
| lineorder_u   |
+-----+
1 row in set (0.00 sec)

mysql>
```

**步骤5** 执行以下命令读取MySQL数据写入Doris。

```
insert into internal.wxk.lineorder_mysql_u select * from
jdbc_mysql.cdl_dy.lineorder_noid;
```

例如执行后结果如下，MySQL数据写入Doris成功，且按照UNIQUE KEY指定字段自动去除重复数据。

```
mysql> select * from internal.wxk.lineorder_mysql_u
```

LO_ORDERKEY	LO_CUSTKEY	LO_PARTKEY	LO_SUPPKEY	LO_ORDERDATE	LO_ORDERPRIORITY	LO_SHIPPRIORITY	LO_QUANTITY	LO_EXTENDEDPRICE	LO_PROFITPRICE	LO_DISCOUNT	LO_REVENUE	LO_SUPPLYCOST	LO_TAX	LO_COMP
1	1	1	1	1997-01-01	01-SEE	1	0	1	1201	18547401	1	231	3301	1
1	2	2	2	1997-01-01	01-SEE	1	0	1	1201	18547401	1	231	3301	1
1	3	3	3	1997-01-01	01-SEE	1	0	1	1201	18547401	1	231	3301	1
1	4	4	4	1997-01-01	01-SEE	1	0	1	1201	18547401	1	231	3301	1
1	5	5	5	1997-01-01	01-SEE	1	0	1	1201	18547401	1	231	3301	1
1	6	6	6	1997-01-01	06-SEE	1	0	1	1201	18547401	1	231	3301	1
1	7	7	7	1997-01-01	07-SEE	1	0	1	1201	18547401	1	231	3301	1
1	8	8	8	1997-01-01	08-SEE	1	0	1	1201	18547401	1	231	3301	1
1	9	9	9	1997-01-01	09-SEE	1	0	1	1201	18547401	1	231	3301	1
1	10	10	10	1997-01-01	10-SEE	1	0	1	1201	18547401	1	2311301	32301	1

----结束

## 2.8.5 离线数据加载：通过 Spark 视图读取 Hive 外表数据并写入 Doris

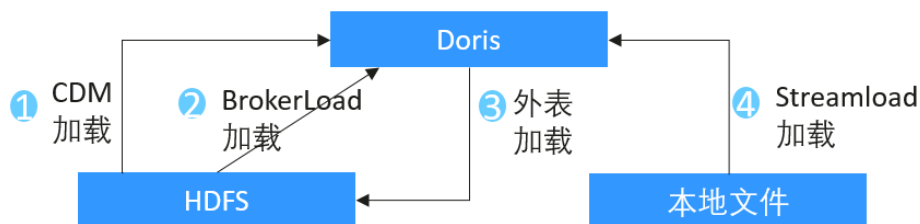
### 应用场景

通过Spark视图读取Hive外表数据写入到Doris，并按照Unique指定字段自动去重。

### 方案架构

离线数据可以从数据湖加载，也可以直接加载本地文件。从数据湖加载可以使用工具CDM，在没有CDM工具时，可以直接使用外表加载、BrokerLoad的方式。本地文件加载则使用Doris自带的StreamLoad工具。

图 2-43 离线数据加载



### 操作流程

- **步骤1：创建MRS Hive集群并配置**
- **步骤2：创建MRS Doris集群并配置**
- **步骤3：创建Doris Catalog并将Hive外表数据写入Doris**

### 步骤 1：创建 MRS Hive 集群并配置

**步骤1** 创建MRS集群，详细操作请参考[快速购买MRS集群](#)。其中：

- 集群名称：自定义，例如 “mrs\_test”



- 版本类型：LTS版
- 集群版本：MRS 3.3.0-LTS
- 集群组件包含：Hive、Spark等。
- Kerberos认证：开启
- 其他参数根据实际需要进行配置。

**步骤2** 集群购买完成后安装集群全量客户端，详细操作请参考[安装客户端（3.x版本）](#)。

例如客户端安装路径为“/opt/newclient”。

**步骤3** 已存在Hive外表（数据格式仅限orc格式），如wxk.lineorder\_orc02。且存在一定量数据。

建表语句参考：

```
CREATE TABLE wxk.`lineorder_orc02`(  
  `lo_orderkey` bigint,  
  `lo_linenumber` int,  
  `lo_custkey` bigint,  
  `lo_partkey` bigint,  
  `lo_suppkey` bigint,  
  `lo_orderdate` date,  
  `lo_orderpriority` varchar(300),  
  `lo_shippriority` int,  
  `lo_quantity` int,  
  `lo_extendedprice` bigint,  
  `lo_ordtotalprice` bigint,  
  `lo_discount` int,  
  `lo_revenue` bigint,  
  `lo_supplycost` int,  
  `lo_tax` int,  
  `lo_commitdate` date,  
  `lo_shipmode` varchar(300))  
ROW FORMAT SERDE  
  'org.apache.hadoop.hive.ql.io.orc.OrcSerde'  
STORED AS INPUTFORMAT  
  'org.apache.hadoop.hive.ql.io.orc.OrcInputFormat'  
OUTPUTFORMAT  
  'org.apache.hadoop.hive.ql.io.orc.OrcOutputFormat'
```

## LOCATION

'obs://test-wxk/doristest'

## TBLPROPERTIES (

'bucketing\_version'='2',

'orc.compress'='ZLIB',

'transient\_lastDdlTime'='1705923234') ;

准备数据如下：

---结束

## 步骤 2：创建 MRS Doris 集群并配置

**步骤1** 创建MRS Doris集群，详细操作请参考[快速购买MRS集群](#)。其中：

- 集群名称：自定义，例如“mrs\_doris”
- 版本类型：LTS版
- 集群版本：MRS 3.3.0-LTS
- 组件选择：Doris集群。
- Kerberos认证：开启
- 其他参数根据实际需要进行配置。

**步骤2** 集群购买完成后安装集群客户端，详细操作请参考[安装客户端（3.x版本）](#)。

例如客户端安装路径为“/opt/dorisclient”。

**步骤3** 在Doris集群安装MySQL客户端，详细操作请参考[使用MySQL客户端连接Doris](#)。

**步骤4** 创建拥有Doris管理权限的人机用户（例如用户名为dorisuser），并修改初始密码。

1. 登录Doris集群Manager页面，详细操作请参考[访问MRS集群Manager](#)。
2. 选择“系统 > 权限 > 角色 > 添加角色”，填写角色名称，如“dorisrole”，在“配置资源权限”选择“待操作的集群 > Doris”，勾选“Doris管理员权限”，单击“确定”。
3. 选择“用户 > 添加用户”，填写用户名称，如“dorisuser”，“用户类型”选择“人机”，“密码策略”默认，输入用户密码并确认密码，关联“dorisrole”角色，单击“确定”。
4. 使用新建的dorisuser用户重新登录FusionInsight Manager，修改该用户初始密码。

---结束

### 步骤 3: 创建 Doris Catalog 并将 Hive 外表数据写入 Doris

**步骤1** 登录Doris集群中已安装好MySQL客户端的MRS Master节点, 执行如下命令进入Doris数据库。

```
export LIBMYSQL_ENABLE_CLEARTEXT_PLUGIN=1 (若集群已启用Kerberos认证  
(安全模式)需执行该命令)
```

```
mysql -u数据库登录用户 -p数据库登录用户密码 -P数据库连接端口 -hDoris FE实例IP  
地址
```

#### 📖 说明

- 数据库连接端口为Doris FE的查询连接端口, 可通过登录MRS Doris集群的Manager界面, 选择“集群 > 服务 > Doris > 实例”, 查看任一FE实例的业务IP地址获取。
- Doris FE实例IP地址可通过登录MRS Doris集群的Manager界面, 选择“集群 > 服务 > Doris > 配置 > 全部配置”, 搜索并查看“query\_port”参数值获取。

**步骤2** 执行以下命令创建数据库(例如名称为“sparkconnector”)并切换。

```
create database if not exists sparkconnector;
```

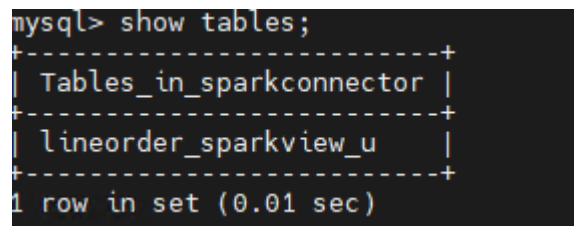
```
use sparkconnector;
```

**步骤3** 执行以下命令创建表(注意表结构要和hive表结构一致)。

```
CREATE TABLE lineorder_view_u(  
  `LO_ORDERKEY` bigint(20) NOT NULL COMMENT '订单id',  
  `LO_LINENUMBER` int(11) NOT NULL COMMENT '订单码',  
  `LO_CUSTKEY` bigint(20) NOT NULL COMMENT '客户id',  
  `LO_PARTKEY` bigint(20) NOT NULL COMMENT '商品id',  
  `LO_SUPPKEY` bigint(20) NOT NULL COMMENT '供应商id',  
  `LO_ORDERDATE` date NULL COMMENT '订单日期',  
  `LO_ORDERPRIORITY` varchar(100) NULL COMMENT '订单优先级',  
  `LO_SHIPPRIORITY` int(11) NULL COMMENT '物流优先级',  
  `LO_QUANTITY` int(11) NULL COMMENT '数量',  
  `LO_EXTENDEDPRICE` bigint(20) NULL COMMENT '扩展价格',  
  `LO_ORDTOTALPRICE` bigint(20) NULL COMMENT '订单总价',  
  `LO_DISCOUNT` int(11) NULL COMMENT '折扣',  
  `LO_REVENUE` bigint(20) NULL COMMENT '收入',  
  `LO_SUPPLYCOST` int(11) NULL COMMENT '成本',  
  `LO_TAX` int(11) NULL COMMENT '税款',  
  `LO_COMMITDATE` date NULL COMMENT '承诺日期',  
  `LO_SHIPMODE` varchar(100) NOT NULL COMMENT '运输方式'  
) ENGINE=OLAP
```

```
UNIQUE KEY(`LO_ORDERKEY`, `LO_LINENUMBER`)  
COMMENT 'OLAP'  
DISTRIBUTED BY HASH(`LO_ORDERKEY`) BUCKETS 1  
PROPERTIES (  
  "replication_allocation" = "tag.location.default: 3",  
  "in_memory" = "false",  
  "storage_format" = "V2",  
  "disable_auto_compaction" = "false"  
);
```

例如执行show tables;后结果如下，表示建表成功：



```
mysql> show tables;  
+-----+  
| Tables_in_sparkconnector |  
+-----+  
| lineorder_sparkview_u   |  
+-----+  
1 row in set (0.01 sec)
```

**步骤4** 以root用户登录MRS Hive集群主节点，执行以下命令登录Spark客户端。

```
cd /opt/newclient  
source bigdata_env  
kinit 组件业务用户  
spark-beeline
```

**步骤5** 执行以下命令创建临时视图。

```
CREATE TEMPORARY VIEW spark_doris_decimal  
USING doris  
OPTIONS(  
  "table.identifier"="doris数据库名.doris数据表名",  
  "fenodes"="FE实例IP地址:FE配置https_port",  
  "user"="用户名",  
  "password"="用户密码",  
  'doris.enable.https' = 'true',  
  'doris.ignore.https.ca' = 'true'  
);
```

**步骤6** 执行以下命令读取Hive外表数据写入Doris。

```
insert into spark_doris_decimal select * from wxk.lineorder_orc02;
```

例如执行后结果如下，数据写入Doris成功，且数据符合预期结果。

```
mysql> select * from l_unorder_sparkview_u;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| LO_ORDERKEY | LO_ORDERPUSER | LO_COSTKEY | LO_PARTKEY | LO_SUPPKEY | LO_ORDERDATE | LO_ORDERPRIORITY | LO_SHIPPRIORITY | LO_QUANTITY | LO_EXTENDEDPRICE | LO_ORDTOTALPRICE | LO_DISCOUNT | LO_REVENUE | LO_SUPPLYCOST | LO_TAX | LO_COMMITDATE | LO_SHIPMODE |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 3201 | 1 | 1997-01-01 | DERG1 | 1 | 1 | 1997-01-01 | 01-SEE | 1 | 18547401 | 18547401 | 1 | 231 | 18547401 | 1 | 1997-01-01 | 1 |
| 3201 | 2 | 1997-01-01 | DERG1 | 2 | 2 | 1997-01-01 | 01-SEE | 1 | 18547401 | 18547401 | 1 | 231 | 18547401 | 1 | 1997-01-01 | 1 |
| 3201 | 3 | 1997-01-01 | DERG1 | 3 | 3 | 1997-01-01 | 01-SEE | 1 | 18547401 | 18547401 | 1 | 231 | 18547401 | 1 | 1997-01-01 | 1 |
| 3201 | 4 | 1997-01-01 | DERG1 | 4 | 4 | 1997-01-01 | 01-SEE | 1 | 18547401 | 18547401 | 1 | 231 | 18547401 | 1 | 1997-01-01 | 1 |
| 3201 | 5 | 1997-01-01 | DERG1 | 5 | 5 | 1997-01-01 | 01-SEE | 1 | 18547401 | 18547401 | 1 | 231 | 18547401 | 1 | 1997-01-01 | 1 |
| 3201 | 6 | 1997-01-01 | DERG1 | 6 | 6 | 1997-01-01 | 06-SEE | 1 | 18547401 | 18547401 | 1 | 231 | 18547401 | 1 | 1997-01-01 | 1 |
| 3201 | 7 | 1997-01-01 | DERG1 | 7 | 7 | 1997-01-01 | 07-SEE | 1 | 18547401 | 18547401 | 1 | 231 | 18547401 | 1 | 1997-01-01 | 1 |
| 3201 | 8 | 1997-01-01 | DERG1 | 8 | 8 | 1997-01-01 | 08-SEE | 1 | 18547401 | 18547401 | 1 | 231 | 18547401 | 1 | 1997-01-01 | 1 |
| 3201 | 9 | 1997-01-01 | DERG1 | 9 | 9 | 1997-01-01 | 09-SEE | 1 | 18547401 | 18547401 | 1 | 231 | 18547401 | 1 | 1997-01-01 | 1 |
| 3201 | 10 | 1997-01-01 | DERG1 | 10 | 10 | 1997-01-01 | 10-SEE | 1 | 2311301 | 18547401 | 1 | 2311301 | 18547401 | 1 | 1997-01-01 | 1 |
| 32001 | 6 | 1997-02-21 | 389358 | 558542 | 20126 | NULL | 0 | 37 | 5921924 | 5611615 | 8 | 5448170 | 96031 | 1992-02-21 | 1 |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
11 rows in set (0.14 sec)
```

----结束

## 2.8.6 离线数据加载：通过 StreamLoad 将本地 CSV 文件导入 Doris

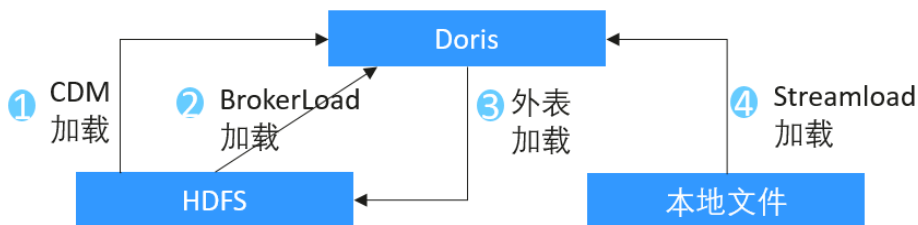
### 应用场景

本地CSV文件数据通过StreamLoad导入到Doris，需要写入Doris的数据不重复，通过引入 Unique 数据模型保证 Key 的唯一性。

### 方案架构

离线数据可以从数据湖加载，也可以直接加载本地文件。从数据湖加载可以使用工具 CDM，在没有CDM工具时，可以直接使用外表加载、BrokerLoad的方式。本地文件加载则使用Doris自带的StreamLoad工具。

图 2-44 离线数据加载



### 操作流程

- 步骤1：创建MRS Doris集群并配置
- 步骤2：创建Doris数据表及CSV文件
- 步骤3：使用Stream Load导入CSV文件至Doris

### 步骤 1：创建 MRS Doris 集群并配置

步骤1 创建MRS Doris集群，详细操作请参考[快速购买MRS集群](#)。其中：

- 集群名称：自定义，例如 “mrs\_doris”
- 版本类型： LTS版
- 集群版本： MRS 3.3.0-LTS
- 组件选择： Doris集群。

- 其他参数根据实际需要进行配置。

**步骤2** 集群购买完成后安装集群客户端，详细操作请参考[安装客户端（3.x版本）](#)。

例如客户端安装路径为“/opt/dorisclient”。

**步骤3** 在Doris集群安装MySQL客户端，详细操作请参考[使用MySQL客户端连接Doris](#)。

**步骤4** 创建拥有Doris管理权限的人机用户（例如用户名为dorisuser），并修改初始密码。

1. 登录Doris集群Manager页面，详细操作请参考[访问MRS集群Manager](#)。
2. 选择“系统 > 权限 > 角色 > 添加角色”，填写角色名称，如“dorisrole”，在“配置资源权限”选择“待操作的集群 > Doris”，勾选“Doris管理员权限”，单击“确定”。
3. 选择“用户 > 添加用户”，填写用户名称，如“dorisuser”，“用户类型”选择“人机”，“密码策略”默认，输入用户密码并确认密码，关联“dorisrole”角色，单击“确定”。
4. 使用新建的dorisuser用户重新登录FusionInsight Manager，修改该用户初始密码。

**步骤5** 如果要导入的文件大小超过10G，则需要修改超时时间及最大导入文件大小参数。

1. 登录FusionInsight Manager，选择“集群 > 服务 > Doris > 配置 > 全部配置”。
2. 选择“FE（角色） > 自定义”，在自定义参数“fe.conf.customized.configs”中新增参数“stream\_load\_default\_timeout\_second”，参数描述及取值请参见[表 2-17](#)。
3. 选择“BE（角色） > 自定义”，在自定义参数“be.conf.customized.configs”中新增参数“streaming\_load\_max\_mb”，参数描述及取值请参见[表 2-17](#)。

表 2-17 自定义参数描述

参数名	参数描述
stream_load_default_timeout_second	表示导入任务的超时时间，默认超时时间为600秒，单位为秒。 若导入任务在设定的时间内未完成则会被系统取消，状态变为“CANCELLED”。默认超时时间为600秒，如果导入的源文件无法在规定时间内完成导入，可以在Stream Load请求中设置单独的超时时间，或调整“stream_load_default_timeout_second”参数值设置全局的默认超时时间。
streaming_load_max_mb	表示Stream Load的最大导入文件大小，默认值为10G，单位为MB。 如果原始文件超过该值，则需要适当调整该参数值。

## 📖 说明

由于Stream Load是由BE进行导入并分发数据，建议导入数据量在1G到10G之间。由于默认最大Stream Load的导入数据量为10G，所以如果要导入超过10G的文件需要修改BE的配置参数“streaming\_load\_max\_mb”。

----结束

## 步骤 2: 创建 Doris 数据表及 CSV 文件

**步骤1** 登录Doris集群中已安装好MySQL客户端的MRS Master节点, 执行如下命令进入Doris数据库。

```
export LIBMYSQL_ENABLE_CLEARTEXT_PLUGIN=1 (若集群已启用Kerberos认证  
(安全模式)需执行该命令)
```

```
mysql -u数据库登录用户 -p数据库登录用户密码 -P数据库连接端口 -hDoris FE实例IP  
地址
```

### 📖 说明

- 数据库连接端口为Doris FE的查询连接端口, 可通过登录MRS Doris集群的Manager界面, 选择“集群 > 服务 > Doris > 实例”, 查看任一FE实例的业务IP地址获取。
- Doris FE实例IP地址可通过登录MRS Doris集群的Manager界面, 选择“集群 > 服务 > Doris > 配置 > 全部配置”, 搜索并查看“query\_port”参数值获取。

**步骤2** 执行以下命令创建数据库(例如名称为“example\_db”)并切换。

```
create database if not exists example_db;
```

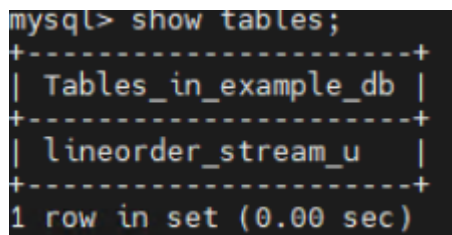
```
use example_db;
```

**步骤3** 执行以下命令创建表。

```
CREATE TABLE `lineorder_stream_u` (  
  `LO_ORDERKEY` bigint NOT NULL COMMENT "订单id",  
  `LO_LINENUMBER` int NOT NULL COMMENT "订单码",  
  `LO_CUSTKEY` bigint NOT NULL COMMENT "客户id",  
  `LO_PARTKEY` bigint NOT NULL COMMENT "商品id",  
  `LO_SUPPKEY` bigint NOT NULL COMMENT "供应商id",  
  `LO_ORDERDATE` date DEFAULT NULL COMMENT "订单日期",  
  `LO_ORDERPRIORITY` varchar(100) DEFAULT NULL COMMENT "订单优先级",  
  `LO_SHIPPRIORITY` int DEFAULT NULL COMMENT "物流优先级",  
  `LO_QUANTITY` int DEFAULT NULL COMMENT "数量",  
  `LO_EXTENDEDPRICE` bigint DEFAULT NULL COMMENT "扩展价格",  
  `LO_ORDTOTALPRICE` bigint DEFAULT NULL COMMENT "订单总价",  
  `LO_DISCOUNT` int DEFAULT NULL COMMENT "折扣",  
  `LO_REVENUE` bigint DEFAULT NULL COMMENT "收入",  
  `LO_SUPPLYCOST` int DEFAULT NULL COMMENT "成本",  
  `LO_TAX` int DEFAULT NULL COMMENT "税款",  
  `LO_COMMITDATE` date DEFAULT NULL COMMENT "承诺日期",  
  `LO_SHIPMODE` varchar(100) NOT NULL COMMENT "运输方式"  
)
```

```
UNIQUE KEY
(`LO_ORDERKEY`,`LO_LINENUMBER`,`LO_CUSTKEY`,`LO_PARTKEY`,`LO_SUPPKEY`,`LO_ORDERDATE`)
PARTITION BY RANGE(`LO_ORDERDATE`)
(
PARTITION `p1992` VALUES LESS THAN ("1993-01-01"),
PARTITION `p1993` VALUES LESS THAN ("1994-01-01"),
PARTITION `p1994` VALUES LESS THAN ("1995-01-01"),
PARTITION `p1995` VALUES LESS THAN ("1996-01-01"),
PARTITION `p1996` VALUES LESS THAN ("1997-01-01"),
PARTITION `p1997` VALUES LESS THAN ("1998-01-01"),
PARTITION `p1998` VALUES LESS THAN ("1999-01-01"),
PARTITION `p1999` VALUES LESS THAN ("2000-01-01"),
PARTITION `p2000` VALUES LESS THAN ("2001-01-01"),
PARTITION `p2001` VALUES LESS THAN ("2002-01-01")
)
DISTRIBUTED BY HASH(`LO_ORDERDATE`) BUCKETS 1
PROPERTIES (
"replication_allocation" = "tag.location.default: 3"
);
```

例如执行 `show tables;` 后结果如下，表示建表成功：



```
mysql> show tables;
+-----+
| Tables_in_example_db |
+-----+
| lineorder_stream_u   |
+-----+
1 row in set (0.00 sec)
```

**步骤4** 退出Doris数据库至客户端，创建数据文件“data.csv”，内容如下。

```
1,1,1,1,1,1997-01-01,01-SEE,0,1,1201,18547401,1,231,3201,1,1997-01-01,DER01
1,1,1,1,1,1997-01-01,10-SEE,0,10,1210,18547410,10,2311310,32310,10,1997-01-10,DER10
2,2,2,2,2,1997-01-01,02-SEE,0,2,1202,18547402,2,2311302,32302,2,1997-01-02,DER02
2,2,2,2,2,1997-01-01,01-SEE,0,1,1201,18547401,1,231,3201,1,1997-01-01,DER01
3,3,3,3,3,1997-01-01,03-SEE,0,3,1203,18547403,3,2311303,32303,3,1997-01-03,DER03
3,3,3,3,3,1997-01-01,01-SEE,0,1,1201,18547401,1,231,3201,1,1997-01-01,DER01
4,4,4,4,4,1997-01-01,04-SEE,0,4,1204,18547404,4,2311304,32304,4,1997-01-04,DER04
4,4,4,4,4,1997-01-01,01-SEE,0,1,1201,18547401,1,231,3201,1,1997-01-01,DER01
5,5,5,5,5,1997-01-01,05-SEE,0,5,1205,18547405,5,2311305,32305,5,1997-01-05,DER05
5,5,5,5,5,1997-01-01,01-SEE,0,1,1201,18547401,1,231,3201,1,1997-01-01,DER01
6,6,6,6,6,1997-01-01,06-SEE,0,6,1206,18547406,6,2311306,32306,6,1997-01-06,DER06
6,6,6,6,6,1997-01-01,06-SEE,0,1,1201,18547401,1,231,3201,1,1997-01-01,DER01
7,7,7,7,7,1997-01-01,07-SEE,0,7,1207,18547407,7,2311307,32307,7,1997-01-07,DER07
7,7,7,7,7,1997-01-01,07-SEE,0,1,1201,18547401,1,231,3201,1,1997-01-01,DER01
8,8,8,8,8,1997-01-01,08-SEE,0,8,1208,18547408,8,2311308,32308,8,1997-01-08,DER08
```



```
8,8,8,8,1997-01-01,08-SEE,0,1,1201,18547401,1,231,3201,1,1997-01-01,DER01
9,9,9,9,1997-01-01,09-SEE,0,9,1209,18547409,9,2311309,32309,9,1997-01-09,DER09
9,9,9,9,1997-01-01,09-SEE,0,1,1201,18547401,1,231,3201,1,1997-01-01,DER01
10,10,10,10,1997-01-01,10-SEE,0,1,1201,18547401,1,2311301,32301,1,1997-01-01,DER01
10,10,10,10,1997-01-01,10-SEE,0,1,1201,18547401,1,2311301,32301,1,1997-01-01,DER01
```

----结束

### 步骤 3: 使用 Stream Load 导入 CSV 文件至 Doris

**步骤1** 以root用户登录 “data.csv” 文件所在节点。

**步骤2** 执行以下命令使用Stream Load导入 “data.csv” 中的数据到Doris数据表中。

- 集群已启用Kerberos认证（安全模式）

```
curl --location-trusted -u doris用户名:用户密码 -H "label:table1_20230217" -H "column_separator;" -T data.csv https://Doris FE实例IP地址:HTTPS端口/api/{数据库名}/{表名}_stream_load
```

- 集群未启用Kerberos认证（普通模式）

```
curl -k --location-trusted -u doris用户名:用户密码 -H "label:table1_20230217" -H "column_separator;" -T data.csv http://Doris FE实例IP地址:HTTP端口/api/{数据库名}/{表名}_stream_load
```

#### 📖 说明

- -H "label:table1\_20230217"表示，指定导入数据标签为table1\_20230217，不指定时系统会自动生成。
- -H "column\_separator;"表示，CSV文件中多字段值的分隔符为逗号，可根据实际数据格式调整分隔符。
- Doris FE实例IP地址：可在Manager界面，选择“集群 > 服务 > Doris > 实例”查看FE实例业务IP地址。
- HTTPS端口号：可在Manager界面，选择“集群 > 服务 > Doris > 配置”，搜索“https\_port”查看。
- HTTP端口号：可在Manager界面，选择“集群 > 服务 > Doris > 配置”，搜索“http\_port”查看。

**步骤3** 执行以下命令查看表数据。

```
select * from example_db.lineorder_stream_u;
```

例如查询结果如下，查询数据成功，且符合预期，表示导入成功。

```
mysql> select * from lineorder_stream_u;
```

LO_ORDERKEY	LO_LINENUMBER	LO_CUSTKEY	LO_PARTKEY	LO_SUPPKEY	LO_ORDERDATE	LO_ORDERPRIORITY	LO_SHIPPRIORITY	LO_QUANTITY	LO_EXTENDEDPRICE	LO_ORDTOTALPRICE	LO_DISCOUNT	LO_REVENUE
32310	1	10	1	1	1997-01-01	10-SEE	0	10	1210	18547410	10	2311310
3201	2	1	2	2	1997-01-01	01-SEE	0	1	1201	18547401	1	231
3201	3	1	3	3	1997-01-01	01-SEE	0	1	1201	18547401	1	231
3201	4	1	4	4	1997-01-01	01-SEE	0	1	1201	18547401	1	231
3201	5	1	5	5	1997-01-01	01-SEE	0	1	1201	18547401	1	231
3201	6	1	6	6	1997-01-01	06-SEE	0	1	1201	18547401	1	231
3201	7	1	7	7	1997-01-01	07-SEE	0	1	1201	18547401	1	231
3201	8	1	8	8	1997-01-01	08-SEE	0	1	1201	18547401	1	231
3201	9	1	9	9	1997-01-01	09-SEE	0	1	1201	18547401	1	231
3201	10	1	10	10	1997-01-01	10-SEE	0	1	1201	18547401	1	2311301
32301	1	1	1	1	1997-01-01	10-SEE	0	1	1201	18547401	1	2311301

----结束

## 2.8.7 离线数据加载：通过 Broker Load 将 ORC 格式的 Hive 数据导入 Doris

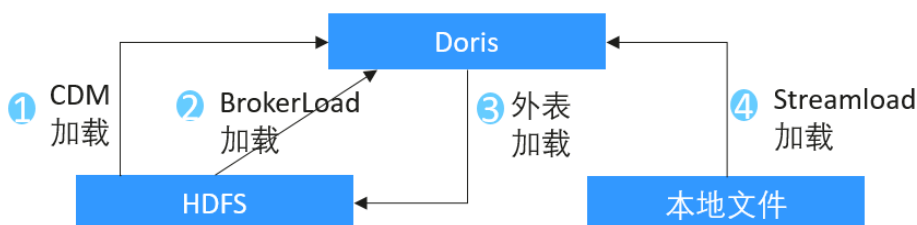
### 应用场景

通过 Broker Load 将 ORC 格式的 Hive 表数据导入到 Doris。

### 方案架构

离线数据可以从数据湖加载，也可以直接加载本地文件。从数据湖加载可以使用工具 CDM，在没有 CDM 工具时，可以直接使用外表加载、BrokerLoad 的方式。本地文件加载则使用 Doris 自带的 StreamLoad 工具。

图 2-45 离线数据加载



### 操作流程

- **步骤1：创建MRS Doris集群并配置**
- **步骤2：创建MRS Hive集群并配置**
- **步骤3：Doris与Hive跨集群配置**
- **步骤4：创建Doris数据表**
- **步骤5：通过Broker Load导入数据至Doris**

### 步骤 1：创建 MRS Doris 集群并配置

**步骤1** 创建MRS Doris集群，详细操作请参考[快速购买MRS集群](#)。其中：

- 集群名称：自定义，例如“mrs\_doris”
- 版本类型：LTS版
- 集群版本：MRS 3.3.0-LTS
- 组件选择：Doris集群。
- 其他参数根据实际需要进行配置。

**步骤2** 集群购买完成后安装集群客户端，详细操作请参考[安装客户端（3.x版本）](#)。

例如客户端安装路径为“/opt/dorisclient”。

**步骤3** 在Doris集群安装MySQL客户端，详细操作请参考[使用MySQL客户端连接Doris](#)。

**步骤4** 创建拥有Doris管理权限的人机用户（例如用户名为dorisuser），并修改初始密码。

1. 登录Doris集群Manager页面，详细操作请参考[访问MRS集群Manager](#)。

2. 选择“系统 > 权限 > 角色 > 添加角色”，填写角色名称，如“dorisrole”，在“配置资源权限”选择“待操作的集群 > Doris”，勾选“Doris管理员权限”，单击“确定”。
3. 选择“用户 > 添加用户”，填写用户名称，如“dorisuser”，“用户类型”选择“人机”，“密码策略”默认，输入用户密码并确认密码，关联“dorisrole”角色，单击“确定”。
4. 使用新建的dorisuser用户重新登录FusionInsight Manager，修改该用户初始密码。

----结束

## 步骤 2: 创建 MRS Hive 集群并配置

**步骤1** 创建MRS集群，详细操作请参考[快速购买MRS集群](#)。其中：

- 集群名称：自定义，例如“mrs\_test”
- 版本类型：LTS版
- 集群版本：MRS 3.3.0-LTS
- 集群组件包含：Hive等。
- 其他参数根据实际需要进行配置。

**步骤2** 集群购买完成后安装集群全量客户端，详细操作请参考[安装客户端（3.x版本）](#)。

例如客户端安装路径为“/opt/newclient”。

**步骤3** 已存在Hive外表（数据格式仅限orc格式），如wxk.lineorder\_broker。且存在一定量数据。

建表语句参考：

```
CREATE TABLE wxk.`lineorder_broker`(  
  `lo_orderkey` bigint,  
  `lo_linenumber` int,  
  `lo_custkey` bigint,  
  `lo_partkey` bigint,  
  `lo_suppkey` bigint,  
  `lo_orderpriority` varchar(300),  
  `lo_shippriority` int,  
  `lo_quantity` int,  
  `lo_orderdate` date,  
  `lo_extendedprice` bigint,  
  `lo_ordtotalprice` bigint,  
  `lo_discount` int,  
  `lo_revenue` bigint,  
  `lo_supplycost` int,
```

```
`lo_commitdate` date,  
`lo_shipmode` varchar(300))  
PARTITIONED BY (`lo_tax` int)  
  
row format delimited fields terminated by ',' lines terminated by '\n' stored as  
orc;  
----结束
```

### 步骤 3: Doris 与 Hive 跨集群配置

如果Hive、Doris服务在两个集群部署时需要执行该操作，否则请跳过。

1. 配置MRS Hive集群与Doris集群互信，详细操作请参考[集群互信管理](#)。
2. 配置后验证。
  - a. 以root用户登录Doris集群客户端，执行以下命令加载环境变量。  
**source 客户端安装路径/bigdata\_env**
  - b. 执行以下命令认证Hive集群用户。  
**kinit Hive集群业务用户@Hive集群系统域名**  
输入Hive集群用户相应密码，认证成功即为互信成功。

#### 📖 说明

- Hive集群用户：可以自行创建也可使用默认的系统用户，例如hive，用户默认密码请参考[MRS集群用户账号一览表](#)。
- Hive集群系统域名：可以登录Hive集群Manager页面，选择“系统 > 权限 > 域和互信”，查看“本端域”参数获取。

### 步骤 4: 创建 Doris 数据表

**步骤1** 登录Doris集群中已安装好MySQL客户端的MRS Master节点，执行如下命令进入Doris数据库。

```
export LIBMYSQL_ENABLE_CLEARTEXT_PLUGIN=1 (若集群已启用Kerberos认证  
(安全模式)需执行该命令)
```

```
mysql -u数据库登录用户 -p数据库登录用户密码 -P数据库连接端口 -hDoris FE实例IP  
地址
```

#### 📖 说明

- 数据库连接端口为Doris FE的查询连接端口，可通过登录MRS Doris集群的Manager界面，选择“集群 > 服务 > Doris > 实例”，查看任一FE实例的业务IP地址获取。
- Doris FE实例IP地址可通过登录MRS Doris集群的Manager界面，选择“集群 > 服务 > Doris > 配置 > 全部配置”，搜索并查看“query\_port”参数值获取。

**步骤2** 执行以下命令创建数据库（例如名称为“example\_db”）并切换。

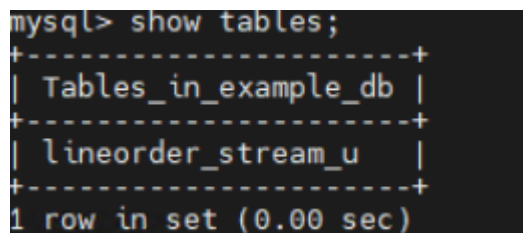
```
create database if not exists example_db;  
  
use example_db;
```

**步骤3** 执行以下命令创建表。

```
CREATE TABLE `lineorder_brokerload_u` (  
  `LO_ORDERKEY` bigint NOT NULL COMMENT "订单id",  
  `LO_LINENUMBER` int NOT NULL COMMENT "订单码",  
  `LO_CUSTKEY` bigint NOT NULL COMMENT "客户id",  
  `LO_PARTKEY` bigint NOT NULL COMMENT "商品id",  
  `LO_SUPPKEY` bigint NOT NULL COMMENT "供应商id",  
  `LO_ORDERDATE` date DEFAULT NULL COMMENT "订单日期",  
  `LO_ORDERPRIORITY` varchar(100) DEFAULT NULL COMMENT "订单优先级",  
  `LO_SHIPPRIORITY` int DEFAULT NULL COMMENT "物流优先级",  
  `LO_QUANTITY` int DEFAULT NULL COMMENT "数量",  
  `LO_EXTENDEDPRICE` bigint DEFAULT NULL COMMENT "扩展价格",  
  `LO_ORDTOTALPRICE` bigint DEFAULT NULL COMMENT "订单总价",  
  `LO_DISCOUNT` int DEFAULT NULL COMMENT "折扣",  
  `LO_REVENUE` bigint DEFAULT NULL COMMENT "收入",  
  `LO_SUPPLYCOST` int DEFAULT NULL COMMENT "成本",  
  `LO_TAX` int DEFAULT NULL COMMENT "税款",  
  `LO_COMMITDATE` date DEFAULT NULL COMMENT "承诺日期",  
  `LO_SHIPMODE` varchar(100) NOT NULL COMMENT "运输方式"  
)  
  
UNIQUE KEY  
(`LO_ORDERKEY`,`LO_LINENUMBER`,`LO_CUSTKEY`,`LO_PARTKEY`,`LO_SUPPKEY`,`LO_ORDERDATE`)  
  
PARTITION BY RANGE(`LO_ORDERDATE`)  
(  
  PARTITION `p1992` VALUES LESS THAN ("1993-01-01"),  
  PARTITION `p1993` VALUES LESS THAN ("1994-01-01"),  
  PARTITION `p1994` VALUES LESS THAN ("1995-01-01"),  
  PARTITION `p1995` VALUES LESS THAN ("1996-01-01"),  
  PARTITION `p1996` VALUES LESS THAN ("1997-01-01"),  
  PARTITION `p1997` VALUES LESS THAN ("1998-01-01"),  
  PARTITION `p1998` VALUES LESS THAN ("1999-01-01"),  
  PARTITION `p1999` VALUES LESS THAN ("2000-01-01"),  
  PARTITION `p2000` VALUES LESS THAN ("2001-01-01"),  
  PARTITION `p2001` VALUES LESS THAN ("2002-01-01")  
)
```

```
)  
DISTRIBUTED BY HASH(`LO_ORDERDATE`) BUCKETS 1  
PROPERTIES (  
  "replication_allocation" = "tag.location.default: 3",  
  "dynamic_partition.enable" = "true",  
  "dynamic_partition.time_unit" = "MONTH",  
  "dynamic_partition.start" = "-2147483648",  
  "dynamic_partition.end" = "2",  
  "dynamic_partition.prefix" = "P_",  
  "dynamic_partition.buckets" = "1",  
  "in_memory" = "false",  
  "storage_format" = "V2"  
);
```

例如执行show tables;后结果如下，表示建表成功：



----结束

## 步骤 5：通过 Broker Load 导入数据至 Doris

**步骤1** 在Doris客户端执行以下命令，使用Broker Load导入数据。

```
LOAD LABEL broker_load_u(  
  DATA INFILE("hdfs://主NameNode实例IP地址:RPC端口号/user/hive/warehouse/  
wzk.db/lineorder_broker/*/*")  
  INTO TABLE lineorder_brokerload_u  
  FORMAT AS "orc"  
  (LO_ORDERKEY,LO_LINENUMBER,LO_CUSTKEY,LO_PARTKEY,LO_SUPPKEY,LO_O  
RDERDATE,LO_ORDERPRIORITY,LO_SHIPPRIORITY,LO_QUANTITY,LO_EXTENDE  
DPRICE,LO_ORDTOTALPRICE,LO_DISCOUNT,LO_REVENUE,LO_SUPPLYCOST,LO_  
COMMITDATE,LO_SHIPMODE,LO_TAX)  
)  
WITH BROKER "broker_192_168_0_221"  
(
```

```
"hadoop.security.authentication"="kerberos",  
"kerberos_principal"="doris/hadoop.hadoop.com@HADOOP.COM",  
"kerberos_keytab"="${BIGDATA_HOME}/FusionInsight_Doris_*/install/  
FusionInsight-Doris-*/doris-fe/bin/doris.keytab"  
)  
PROPERTIES  
(  
"timeout"="1200",  
"max_filter_ratio"="0.1"  
);
```

### 须知

- broker\_load\_u: 表示任务名称, 自定义。
- 主NameNode实例IP地址: 可在Manager界面, 选择“集群 > 服务 > HDFS > 实例”查看。(此配置中要注意集群重启会导致NameNode实例主备倒换)
- RPC端口号: 可在Manager界面, 选择“集群 > 服务 > HDFS > 配置”, 搜索“dfs.namenode.rpc.port”查看。
- broker\_192\_168\_0\_221: Doris数据库的Broker名称, 连接Doris数据库后执行 **show broker**; 获取。
- kerberos\_principal: 访问Hadoop集群的keytab文件对应的principal, 格式为“**doris/hadoop. \${系统域名转换为小写}@ \${系统域名}**”, 系统域名可登录Manager页面, 选择“系统 > 权限 > 域和互信”, 查看“本端域”参数获取。Hive与Doris非同集群部署时要使用Hive用户@Hive集群的系统域名, 如hive/hadoop.hadoop.com@HADOOP.COM。
- kerberos\_keytab:  
访问Hadoop集群的keytab文件, 该keytab位于FE节点的“**\${BIGDATA\_HOME}/FusionInsight\_Doris\_\*/install/FusionInsight-Doris-\*/doris-be/bin/doris.keytab**”路径中, 需要拷贝该keytab文件到所有Broker节点上, 例如“/home/omm/doris\_keytab”目录下, 并执行**chown omm:wheel /home/omm/doris\_keytab -R**命令设置“doris.keytab”文件属组。  
Hive与Doris非同集群部署时, 要从Hive集群拷贝hive.keytab文件到DBroker实例对应目录下。

**步骤2** 执行以下命令查看任务执行情况。

```
show load order by createtime desc limit 1\G;
```

例如执行结果如下, 表示数据导入成功。

```
mysql> show load order by createtime desc limit 1\G;
***** 1. row *****
      JobId: 824401
      Label: broker_load_u1
      State: FINISHED
      Progress: ETL:100%; LOAD:100%
      Type: BROKER
      EtlInfo: unselected.rows=0; dpp.abnorm.ALL=0; dpp.norm.ALL=20
      TaskInfo: cluster:N/A; timeout(s):1200; max_filter_ratio:0.1
      ErrorMsg: NULL
      CreateTime: 2024-02-27 10:52:30
      EtlStartTime: 2024-02-27 10:52:31
      EtlFinishTime: 2024-02-27 10:52:31
      LoadStartTime: 2024-02-27 10:52:31
      LoadFinishTime: 2024-02-27 10:52:31
      URL: NULL
      JobDetails: {"Unfinished backends":{"d9516cac5d304362-901ee3ec17a34a7d":[]},"ScannedRows":20,"TaskNumber":1,"LoadB
      "FileSize":13095}
      TransactionId: 1060046
      ErrorTablets: {}
      1 row in set (0.00 sec)

ERROR:
No query specified
```

步骤3 执行以下命令查看目标表数据情况。

```
select * from lineorder_brokerload_u
```

可以查看到MySQL数据导入Doris成功，且根据UNIQUE KEY指定字段自动去重

```
mysql> select * from lineorder_brokerload_u
```

LO_ORDERKEY   LO_LINEORDER   LO_CUSTKEY   LO_PARTKEY   LO_SUPPKEY   LO_ORDERDATE   LO_ORDERPRIORITY   LO_SHIPPRIORITY   LO_QUANTITY   LO_EXTENDEDPRICE   LO_ORDTOTALPRICE   LO_DISCOUNT   LO_REVENUE   LO...
1   10   1997-01-10   DER10   1   1997-01-01   10-SEE   0   10   1210   18547410   10   2311316   ...
2   2   1997-01-02   DER02   2   1997-01-01   02-SEE   0   2   1202   18547402   2   2311302   ...
3   3   1997-01-03   DER03   3   1997-01-01   03-SEE   0   3   1203   18547403   3   2311303   ...
4   4   1997-01-04   DER04   4   1997-01-01   04-SEE   0   4   1204   18547404   4   2311304   ...
5   5   1997-01-05   DER05   5   1997-01-01   05-SEE   0   5   1205   18547405   5   2311305   ...
6   6   1997-01-06   DER06   6   1997-01-01   06-SEE   0   6   1206   18547406   6   2311306   ...
7   7   1997-01-07   DER07   7   1997-01-01   07-SEE   0   7   1207   18547407   7   2311307   ...
8   8   1997-01-08   DER08   8   1997-01-01   08-SEE   0   8   1208   18547408   8   2311308   ...
9   9   1997-01-09   DER09   9   1997-01-01   09-SEE   0   9   1209   18547409   9   2311309   ...
10   10   1997-01-10   DER10   10   1997-01-01   10-SEE   0   10   1210   18547410   10   2311310   ...

----结束

## 2.8.8 实时数据加载：通过 FlinkSQL 将订单表 Kafka 数据实时同步到 Doris

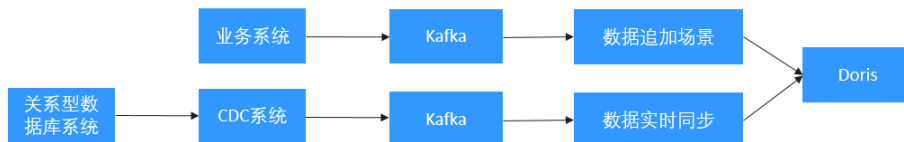
### 应用场景

创建MRS FlinkServer作业，将订单表Kafka数据通过FlinkSQL实时同步到Doris，数据新增。

### 方案架构

对实时性要求较高的场景，可直接将实时流式数据通过FlinkSQL传入Doris中，在Doris中进行实时查询。如果数据在加载到Doris之前需要处理，可直接在FlinkSQL中进行数据处理，将加工后的数据实时写入Doris中。

图 2-46 实时数据加载



### 操作流程

- 步骤1: 创建MRS Flink集群并配置



- [步骤2: 创建MRS Doris集群并配置](#)
- [步骤3: 准备数据](#)
- [步骤4: 创建Flink SQL作业](#)
- [步骤5: Kafka数据通过FlinkSQL实时同步到Doris](#)

## 步骤 1: 创建 MRS Flink 集群并配置

**步骤1** 创建MRS集群，详细操作请参考[快速购买MRS集群](#)。其中：

- 集群名称：自定义，例如“mrs\_test”
- 版本类型：LTS版
- 集群版本：MRS 3.3.0-LTS
- 集群组件包含：Kafka、Flink等。
- Kerberos认证：开启
- 其他参数根据实际需要进行配置。

**步骤2** 为集群添加拥有OBS权限的委托。

1. 创建具有访问OBS权限的ECS委托，详细操作请参考[配置MRS集群通过IAM委托对接OBS](#)中“创建具有访问OBS权限的ECS委托”部分。例如委托名称为“mrs\_ecs\_obs”。（如果已存在可用的OBS权限委托，则跳过该步骤）
2. 在MRS管理控制台“现有集群”中单击已创建完成的MRS集群名称。在集群“概览”页签单击“委托”后的“选择委托”，选择已创建的委托名称（例如“mrs\_ecs\_obs”）。单击“确定”。



**步骤3** 创建具有FlinkServer管理权限的用户。

1. 登录集群Manager页面，详细操作请参考[访问MRS集群Manager](#)。
2. 选择“系统 > 权限 > 角色 > 添加角色”。  
填写角色名称，如“flinkrole”，在“配置资源权限”的表格中选择“待操作集群的名称 > Flink”，勾选“FlinkServer管理操作权限”。单击“确定”。
3. 选择“用户 > 添加用户”，填写用户名称，如“flinkuser”，“用户类型”选择“人机”，“密码策略”默认，输入用户密码并确认密码，关联“flinkrole”、“System\_administrator”角色，单击“确定”。
4. 使用新建的flinkuser用户重新登录FusionInsight Manager，修改该用户初始密码。

**步骤4** 集群购买完成后安装集群全量客户端，详细操作请参考[安装客户端（3.x版本）](#)。

例如客户端安装路径为“/opt/newclient”。

----结束

## 步骤 2: 创建 MRS Doris 集群并配置

**步骤1** 创建MRS Doris集群，详细操作请参考[快速购买MRS集群](#)。其中：

- 集群名称：自定义，例如“mrs\_doris”
- 版本类型：LTS版
- 集群版本：MRS 3.3.0-LTS
- 组件选择：Doris集群。
- Kerberos认证：开启
- 其他参数根据实际需要进行配置。

**步骤2** 为集群添加拥有OBS权限的委托。

1. 创建具有访问OBS权限的ECS委托，详细操作请参考[配置MRS集群通过IAM委托对接OBS](#)中“创建具有访问OBS权限的ECS委托”部分。例如委托名称为“mrs\_ecs\_obs”。（如果已存在可用的OBS权限委托，则跳过该步骤）
2. 在MRS管理控制台“现有集群”中单击已创建完成的MRS集群名称。在集群“概览”页签单击“委托”后的“选择委托”，选择已创建的委托名称（例如“mrs\_ecs\_obs”）。单击“确定”。



**步骤3** 创建拥有Doris管理权限的人机用户（例如用户名为dorisuser），并修改初始密码。

1. 登录Doris集群Manager页面，详细操作请参考[访问MRS集群Manager](#)。
2. 选择“系统 > 权限 > 角色 > 添加角色”，填写角色名称，如“dorisrole”，在“配置资源权限”选择“待操作的集群 > Doris”，勾选“Doris管理员权限”，单击“确定”。
3. 选择“用户 > 添加用户”，填写用户名称，如“dorisuser”，“用户类型”选择“人机”，“密码策略”默认，输入用户密码并确认密码，关联“dorisrole”角色，单击“确定”。
4. 使用新建的dorisuser用户重新登录FusionInsight Manager，修改该用户初始密码。

**步骤4** 集群购买完成后安装集群客户端，详细操作请参考[安装客户端（3.x版本）](#)。

例如客户端安装路径为“/opt/dorisclient”。

**步骤5** 在Doris集群安装MySQL客户端，详细操作请参考[使用MySQL客户端连接Doris](#)。

----结束

## 步骤 3: 准备数据

### 步骤1 创建Doris的Unique表。

1. 登录Doris集群中已安装好MySQL客户端的MRS Master节点，执行如下命令进入Doris数据库。

**export LIBMYSQL\_ENABLE\_CLEARTEXT\_PLUGIN=1**（若集群已启用Kerberos认证（安全模式）需执行该命令）

**mysql -u数据库登录用户 -p数据库登录用户密码 -P数据库连接端口 -hDoris FE实例IP地址**

#### 📖 说明

- 数据库连接端口为Doris FE的查询连接端口，可通过登录MRS Doris集群的Manager界面，选择“集群 > 服务 > Doris > 实例”，查看任一FE实例的业务IP地址获取。
  - Doris FE实例IP地址可通过登录MRS Doris集群的Manager界面，选择“集群 > 服务 > Doris > 配置 > 全部配置”，搜索并查看“query\_port”参数值获取。
2. 执行以下命令创建Doris的Unique表。

```
create database test;
```

```
use test;
```

```
CREATE TABLE `lineorder_u` (  
  `LO_ORDERKEY` bigint NOT NULL COMMENT "订单id",  
  `LO_LINENUMBER` int NOT NULL COMMENT "订单码",  
  `LO_CUSTKEY` bigint NOT NULL COMMENT "客户id",  
  `LO_PARTKEY` bigint NOT NULL COMMENT "商品id",  
  `LO_SUPPKEY` bigint NOT NULL COMMENT "供应商id",  
  `LO_ORDERDATE` date DEFAULT NULL COMMENT "订单日期",  
  `LO_ORDERPRIORITY` varchar(100) DEFAULT NULL COMMENT "订单优先级",  
  `LO_SHIPPRIORITY` int DEFAULT NULL COMMENT "物流优先级",  
  `LO_QUANTITY` int DEFAULT NULL COMMENT "数量",  
  `LO_EXTENDEDPRICE` bigint DEFAULT NULL COMMENT "扩展价格",  
  `LO_ORDTOTALPRICE` bigint DEFAULT NULL COMMENT "订单总价",  
  `LO_DISCOUNT` int DEFAULT NULL COMMENT "折扣",  
  `LO_REVENUE` bigint DEFAULT NULL COMMENT "收入",  
  `LO_SUPPLYCOST` int DEFAULT NULL COMMENT "成本",  
  `LO_TAX` int DEFAULT NULL COMMENT "税款",  
  `LO_COMMITDATE` date DEFAULT NULL COMMENT "承诺日期",  
  `LO_SHIPMODE` varchar(100) NOT NULL COMMENT "运输方式"  
)  
UNIQUE KEY  
(`LO_ORDERKEY`,`LO_LINENUMBER`,`LO_CUSTKEY`,`LO_PARTKEY`,`LO_SUPPKEY`,`LO_ORDERDATE`)  
PARTITION BY RANGE(`LO_ORDERDATE`)  
(  
PARTITION `p1992` VALUES LESS THAN ("1993-01-01"),
```

```
PARTITION `p1993` VALUES LESS THAN ("1994-01-01"),
PARTITION `p1994` VALUES LESS THAN ("1995-01-01"),
PARTITION `p1995` VALUES LESS THAN ("1996-01-01"),
PARTITION `p1996` VALUES LESS THAN ("1997-01-01"),
PARTITION `p1997` VALUES LESS THAN ("1998-01-01"),
PARTITION `p1998` VALUES LESS THAN ("1999-01-01"),
PARTITION `p1999` VALUES LESS THAN ("2000-01-01"),
PARTITION `p2000` VALUES LESS THAN ("2001-01-01"),
PARTITION `p2001` VALUES LESS THAN ("2002-01-01")
)
DISTRIBUTED BY HASH(`LO_ORDERDATE`) BUCKETS 1
PROPERTIES (
"replication_allocation" = "tag.location.default: 3"
);
```

**步骤2** 创建Kafka多分区的Topic。

1. 以客户端安装用户，登录MRS Flink集群中安装了客户端的节点。
2. 执行以下命令登录客户端。“`/opt/newclient`”为客户端安装目录，具体以实际为准。

```
source /opt/newclient/bigdata_env
```

```
kinit 业务用户
```

3. 执行以下命令创建Kafka多分区的Topic

```
cd /opt/newclient/Kafka/kafka/bin
```

```
sh kafka-topics.sh --create --topic 主题名称 --partitions 主题占用的分区数 --
replication-factor 主题的备份个数 --zookeeper ZooKeeper角色实例所在节点
IP地址:clientPort/kafka
```

#### 📖 说明

- ZooKeeper角色实例所在节点IP地址：登录FusionInsight Manager，选择“集群 > 服务 > ZooKeeper > 实例”，查看并记录ZooKeeper角色实例业务IP地址获取。
- clientPort：可在ZooKeeper的全部配置参数中搜索“`clientPort`”查看。

例如创建3分区的Topic，则执行以下命令：

```
sh kafka-topics.sh --create --topic topic_lineorder --partitions 3 --
replication-factor 3 --zookeeper 10.10.10.x:2181/kafka
```

----结束

## 步骤 4：创建 Flink SQL 作业

**步骤1** 使用**步骤3**创建的具有FlinkServer管理员权限的用户（例如**flinkuser**），登录MRS Flink集群Manager页面。

**步骤2** 选择“集群 > 服务 > Flink”，在“Flink WebUI”右侧，单击链接，访问Flink的WebUI。

**步骤3** 选择“作业管理 > 新建作业”，在新建作业页面配置相关信息后，单击“确定”。

- 类型: Flink SQL
- 名称: 自定义
- 作业类型: 流作业
- 描述: 自定义

## 新建作业

\* 类型  Flink SQL  Flink Jar

\* 名称

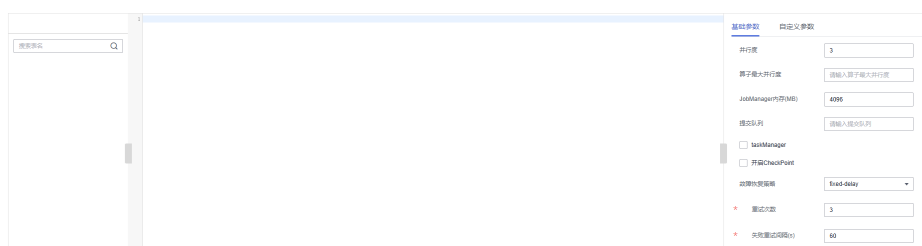
\* 作业类型  流作业  批作业

描述

**步骤4** 配置Flink SQL作业信息。

图 2-47 配置 Flink SQL 作业

作业管理 > flinkjob



1. 在作业开发界面配置作业信息，例如，配置如下：

```
CREATE TABLE IF NOT EXISTS flink_kafka_source (  
  `LO_ORDERKEY` BIGINT,  
  `LO_LINENUMBER` INT,  
  `LO_CUSTKEY` BIGINT,  
  `LO_PARTKEY` BIGINT,  
  `LO_SUPPKEY` BIGINT,  
  `LO_ORDERDATE` DATE,  
  `LO_ORDERPRIORITY` STRING,  
  `LO_SHIPPRIORITY` INT,
```

```
`LO_QUANTITY` INT,  
`LO_EXTENDEDPRICE` BIGINT,  
`LO_ORDTOTALPRICE` BIGINT,  
`LO_DISCOUNT` INT,  
`LO_REVENUE` BIGINT,  
`LO_SUPPLYCOST` INT,  
`LO_TAX` INT,  
`LO_COMMITDATE` DATE,  
`LO_SHIPMODE` STRING  
) WITH (  
'connector' = 'kafka',  
'topic' = 'topic名称',  
'properties.group.id' = 'group1',  
'scan.startup.mode' = 'earliest-offset',  
'properties.bootstrap.servers' = 'Kafka的Broker实例业务IP1:Kafka端口号,Kafka的Broker实例业务  
IP2:Kafka端口号',  
'format' = 'json',  
'properties.sasl.kerberos.service.name' = 'kafka',  
'properties.security.protocol' = 'SASL_PLAINTEXT',  
'properties.kerberos.domain.name' = 'hadoop.系统域名'  
);  
CREATE TABLE IF NOT EXISTS flink_doris_sink (  
`LO_ORDERKEY` BIGINT,  
`LO_LINENUMBER` INT,  
`LO_CUSTKEY` BIGINT,  
`LO_PARTKEY` BIGINT,  
`LO_SUPPKEY` BIGINT,  
`LO_ORDERDATE` DATE,  
`LO_ORDERPRIORITY` STRING,  
`LO_SHIPPRIORITY` INT,  
`LO_QUANTITY` INT,  
`LO_EXTENDEDPRICE` BIGINT,  
`LO_ORDTOTALPRICE` BIGINT,  
`LO_DISCOUNT` INT,  
`LO_REVENUE` BIGINT,  
`LO_SUPPLYCOST` INT,  
`LO_TAX` INT,  
`LO_COMMITDATE` DATE,  
`LO_SHIPMODE` STRING  
) WITH (  
'connector' = 'doris',  
'fenodes' = 'Doris的FE实例IP地址1:Doris端口号, Doris的FE实例IP地址2:Doris端口号, Doris的FE实例IP地  
址3:Doris端口号',  
'table.identifier' = 'database_name.table_name',  
'username' = 'user',  
'password' = 'password',  
'sink.label-prefix' = 'doris_label_xxx',  
'doris.enable.https' = 'true',  
'doris.ignore.https.ca' = 'true'  
);  
  
insert into  
flink_doris_sink  
select  
*  
from  
flink_kafka_source;
```

## 📖 说明

kafka:

- 'topic' = 'topic名称': 已创建kafka多分区的Topic。
- 'properties.group.id' = 'group1': Kafka的group
- Kafka的Broker实例业务IP: 登录Manager, 选择“集群 > 服务 > Kafka > 实例”, 查看Broker实例业务IP。
- Kafka端口号: 登录Manager, 选择“集群 > 服务 > Kafka > 配置”, 搜索Kafka服务的“sasl.port”参数获取。
- hadoop.系统域名: 登录Manager, 选择“系统 > 权限 > 域和互信”, 查看“本端域”参数, 即为当前系统域名。

Doris:

- Doris的FE实例IP地址: 登录Manager, 选择“集群 > 服务 > Doris > 实例”, 查询FE实例的业务IP地址获取。
- Doris端口号: 登录Manager, 选择“集群 > 服务 > Doris > 配置”, 查询Doris服务的“https\_port”参数获取。
- database\_name.table\_name: 数据库名, 表名。
- 'username' = 'user': 为在两集群上分别创建的具有FlinkServer、Doris管理员权限的同名用户。
- 'password' = 'password': 创建的具有FlinkServer、Doris管理员权限的用户密码。
- doris\_label\_xxx: sink.label-prefix的值, 必须是唯一的, 每次启任务必须保持其值的唯一性。

### 2. 在界面左侧配置作业“基础参数”。

- 并行度: 3
- JobManager内存(MB): 4096
- 提交队列: 自定义, 默认default
- 勾选“taskManager”
  - slot数量: 1
  - 内存(MB): 4096
- 勾选“开启CheckPoint”
  - 时间间隔(ms): 20000 (默认最小时间间隔20s)
  - 模式: EXACTLY\_ONCE, 其他默认参数即可
- 故障恢复策略: fixed-delay
  - 重试次数: 3
  - 失败重试间隔(s): 60
- 其他配置保持默认即可

### 3. 配置作业完成后, 可以单击上方“语义校验”对输入内容校验, 单击“SQL格式化”对SQL语句进行格式化。确认无问题后, 单击“保存”, 单击“提交”提交作业。

### 4. 在“作业管理”页面单击Flink SQL作业“操作”列的“启动”, 运行作业。

----结束

## 步骤 5: Kafka 数据通过 FlinkSQL 实时同步到 Doris

**步骤1** 以客户端安装用户，登录MRS Flink集群中安装了客户端的节点。

**步骤2** 执行以下命令登录客户端。“/opt/newclient”为客户端安装目录，具体以实际为准。

```
source /opt/newclient/bigdata_env
```

```
kinit 业务用户
```

**步骤3** 执行以下命令向Kafka写入数据。

```
cd /opt/newclient/Kafka/kafka/bin
```

```
sh kafka-console-producer.sh --broker-list Kafka角色实例所在节点的IP地址:Kafka  
端口号 --topic topic名称 --producer.config /opt/newclient/Kafka/kafka/config/  
producer.properties
```

### 📖 说明

- Kafka角色实例所在节点的IP地址：登录Manager后，选择“集群 > 服务 > Kafka > 实例”，记录Broker角色实例其中任意一个的业务IP地址。
- Kafka端口号：登录Manager后，选择“集群 > 服务 > Kafka > 配置 > 全部配置”，集群已启用Kerberos认证（安全模式）搜索并获取“sasl.port”参数的值，集群未启用Kerberos认证（普通模式）搜索并获取“port”的值。

例如，写入数据内容如下：

```
{"LO_ORDERKEY":"1","LO_LINENUMBER":"1","LO_CUSTKEY":"1","LO_PARTKEY":"1","LO_SUPPKEY":"1","L  
O_ORDERDATE":"1996-01-02","LO_ORDERPRIORITY":"5-  
LOW","LO_SHIPPRIORITY":"0","LO_QUANTITY":"17","LO_EXTENDEDPRICE":"6511","LO_ORDTOTALPRICE":  
"9991","LO_DISCOUNT":"4","LO_REVENUE":"99999","LO_SUPPLYCOST":"95999","LO_TAX":"1001","LO_C  
OMMITDATE":"1996-01-09","LO_SHIPMODE":"AIR"}  
{"LO_ORDERKEY":"2","LO_LINENUMBER":"1","LO_CUSTKEY":"1","LO_PARTKEY":"1","LO_SUPPKEY":"1","L  
O_ORDERDATE":"1996-01-02","LO_ORDERPRIORITY":"5-  
LOW","LO_SHIPPRIORITY":"0","LO_QUANTITY":"17","LO_EXTENDEDPRICE":"6511","LO_ORDTOTALPRICE":  
"9991","LO_DISCOUNT":"4","LO_REVENUE":"99999","LO_SUPPLYCOST":"95999","LO_TAX":"1001","LO_C  
OMMITDATE":"1996-01-09","LO_SHIPMODE":"AIR"}  
{"LO_ORDERKEY":"3","LO_LINENUMBER":"1","LO_CUSTKEY":"1","LO_PARTKEY":"1","LO_SUPPKEY":"1","L  
O_ORDERDATE":"1996-01-02","LO_ORDERPRIORITY":"5-  
LOW","LO_SHIPPRIORITY":"0","LO_QUANTITY":"17","LO_EXTENDEDPRICE":"6511","LO_ORDTOTALPRICE":  
"9991","LO_DISCOUNT":"4","LO_REVENUE":"99999","LO_SUPPLYCOST":"95999","LO_TAX":"1001","LO_C  
OMMITDATE":"1996-01-09","LO_SHIPMODE":"AIR"}
```

**步骤4** 登录Doris集群中已安装好MySQL客户端的MRS Master节点，执行如下命令进入Doris数据库。

```
export LIBMYSQL_ENABLE_CLEARTEXT_PLUGIN=1 （若集群已启用Kerberos认证  
（安全模式）需执行该命令）
```

```
mysql -u数据库登录用户 -p数据库登录用户密码 -P数据库连接端口 -hDoris FE实例IP  
地址
```

### 📖 说明

- 数据库连接端口为Doris FE的查询连接端口，可通过登录MRS Doris集群的Manager界面，选择“集群 > 服务 > Doris > 实例”，查看任一FE实例的业务IP地址获取。
- Doris FE实例IP地址可通过登录MRS Doris集群的Manager界面，选择“集群 > 服务 > Doris > 配置 > 全部配置”，搜索并查看“query\_port”参数值获取。

**步骤5** 执行以下命令查询新增的数据。

```
use 数据库名称;
```



```
select * from lineorder_u;
```

例如查看到如下信息，表示查询新增的数据成功。

```
mysql> select * from lineorder_u;
```

LO_ORDERKEY	LO_LINENUMBER	LO_CUSTKEY	LO_PARTKEY	LO_SUPPKEY	LO_ORDERDATE	LO_ORDERPRIORITY	LO_SHIPPRIORITY	LO_QUANTITY	LO_EXTENDEDPRICE	LO_ORIGINALPRICE	LO_DISCOUNT	LO_REVENUE	LO_SUPPLYCOST	LO_TAX
1996-01-09	1	1	1	1	1996-01-02	5-LOW	0	17	6511	9991	4	99999	59999	1001
1996-01-09	2	1	1	1	1996-01-02	5-LOW	0	17	6511	9991	4	99999	59999	1001
1996-01-09	3	1	1	1	1996-01-02	5-LOW	0	17	6511	9991	4	99999	59999	1001

----结束

## 2.8.9 实时 OLAP 数据查询：基于 Unique 表的单表检索查询和聚合查询

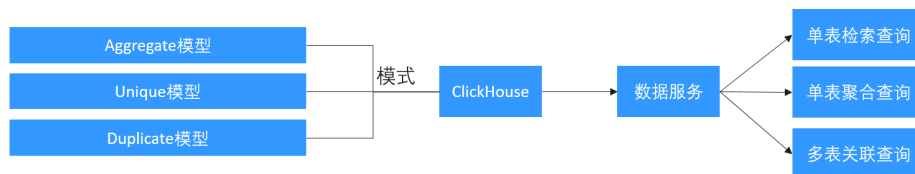
### 应用场景

本场景通过基于Unique模型表查询符合条件的数据。基于Unique模型表聚合查询，支持MIN，MAX，SUM，REPLACE四种聚合算法。

### 方案架构

Doris支持海量数据的亚秒级查询，支持单表数据的聚合查询和多表关联查询。Doris不同的表引擎，适合不同的业务场景，可以根据业务特点选择不同的表格式。

图 2-48 实时 OLAP 数据查询



### 操作流程

- 步骤1：通过StreamLoad将本地CSV文件导入Doris
- 步骤2：基于Unique表的单表查询

#### 步骤 1：通过 StreamLoad 将本地 CSV 文件导入 Doris

参考[离线数据加载：通过StreamLoad将本地CSV文件导入Doris](#)章节，将数据实时同步到Doris的unique模型表。

#### 步骤 2：基于 Unique 表的单表查询

步骤1 登录Doris集群中已安装好MySQL客户端的MRS Master节点，执行如下命令进入Doris数据库。

```
export LIBMYSQL_ENABLE_CLEARTEXT_PLUGIN=1 （若集群已启用Kerberos认证（安全模式）需执行该命令）
```

```
mysql -u数据库登录用户 -p数据库登录用户密码 -P数据库连接端口 -hDoris FE实例IP地址
```

### 📖 说明

- 数据库连接端口为Doris FE的查询连接端口，可通过登录MRS Doris集群的Manager界面，选择“集群 > 服务 > Doris > 实例”，查看任一FE实例的业务IP地址获取。
- Doris FE实例IP地址可通过登录MRS Doris集群的Manager界面，选择“集群 > 服务 > Doris > 配置 > 全部配置”，搜索并查看“query\_port”参数值获取。

**步骤2** 执行以下命令对Doris数据表（Unique模型）进行查询。

**use** {数据库名}

- 单表检索查询：

```
select LO_REVENUE,LO_SUPPLYCOST,LO_TAX from lineorder_stream_u
where LO_ORDERKEY <= 10 and LO_ORDERKEY >= 5 and LO_DISCOUNT
BETWEEN 0 AND 9 AND LO_QUANTITY < 10;
```

例如，执行结果如下，表示查询成功：

```
mysql> select LO_REVENUE,LO_SUPPLYCOST,LO_TAX from lineorder_stream_u where LO_ORDERKEY <= 10 and LO_ORDERKEY >= 5 and LO_DISCOUNT BETWEEN 0 AND 9 AND LO_QUANTITY < 10;
+-----+-----+-----+
| LO_REVENUE | LO_SUPPLYCOST | LO_TAX |
+-----+-----+-----+
| 231       | 3201         | 1       |
| 231       | 3201         | 1       |
| 231       | 3201         | 1       |
| 231       | 3201         | 1       |
| 231       | 3201         | 1       |
| 231       | 3201         | 1       |
+-----+-----+-----+
```

- 单表聚合查询：

```
SELECT LO_CUSTKEY,sum(LO_REVENUE) as
LO_REVENUE,max(LO_SUPPLYCOST) as LO_SUPPLYCOST,min(LO_TAX) as
LO_TAX FROM lineorder_stream_u GROUP BY LO_CUSTKEY having
LO_CUSTKEY<= 10 and LO_CUSTKEY>= 5;
```

例如，执行结果如下，表示查询成功：

```
mysql> SELECT LO_CUSTKEY,sum(LO_REVENUE) as LO_REVENUE,max(LO_SUPPLYCOST) as LO_SUPPLYCOST,min(LO_TAX) as LO_TAX FROM lineorder_stream_u GROUP BY LO_CUSTKEY having LO_CUSTKEY <= 10 and LO_CUSTKEY >= 5;
+-----+-----+-----+
| LO_CUSTKEY | LO_REVENUE | LO_SUPPLYCOST | LO_TAX |
+-----+-----+-----+
| 5         | 231       | 3201         | 1       |
| 6         | 231       | 3201         | 1       |
| 7         | 231       | 3201         | 1       |
| 8         | 231       | 3201         | 1       |
| 9         | 231       | 3201         | 1       |
| 10        | 2313001  | 32301        | 1       |
+-----+-----+-----+
```

----结束

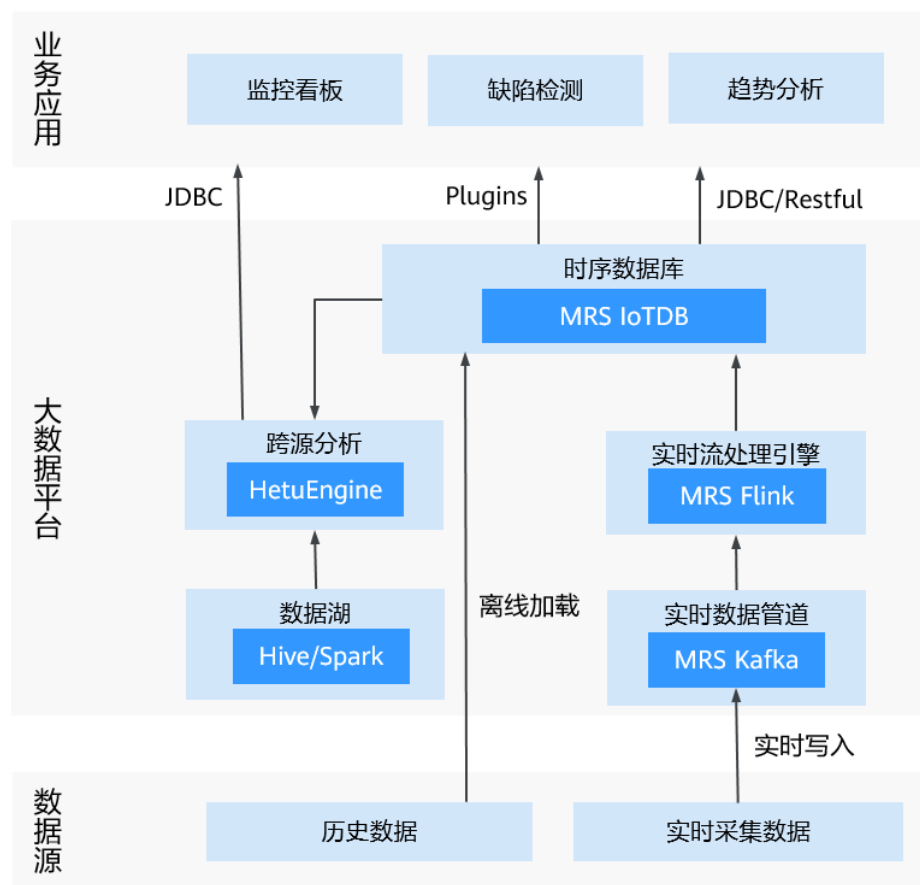
## 2.9 物联网时序数据分析

### 2.9.1 物联网时序数据分析场景介绍

在物联网场景中，海量设备采集的数据需要进行低延迟的高效接入、存储和分析，数据具有高度的时间属性相关特点，传统数据库基于二维表的数据模型，无法满足物联网场景下高效的时序数据存储和分析需求。时序数据库则是专门针对物联网时序数据场景设计的存算引擎，数据按设备管理架构和时间属性进行组织存储和访问，具备低延迟、高并发、高性价比的特点，并提供丰富的时间序列语义，可以更好的满足物联网场景下时序数据分析的需求。

物联网时序数据分析架构如图2-49所示。

图 2-49 物联网时序数据分析



- 数据源：物联网时序数据，包括离线的CSV文件数据、实时采集器采集的数据、或支持MQTT协议的物联网终端设备。
- 大数据平台：物联网时序数据分析的核心平台，包括实时数据管道、实时数据处理引擎、时序数据库等模块。
  - 实时消息管道：用于接收前端采集器推送的实时时序数据，作为实时时序数据接入的统一管道，对应MRS Kafka组件。
  - 实时流处理引擎：从Kafka实时读取时序数据，并进行一些逻辑运算，将实时时序数据写入时序数据库中。
  - 数据湖：其他IT系统上的数据，时序数据库可以通过跨源分析能力与数据湖中的全量数据进行关联，与IT数据进行进一步的复杂分析。
  - HetuEngine：跨源分析引擎，可以连接多个大数据组件进行数据查询，并将多个大数据组件中的数据在HetuEngine中进行统一分析。
  - IoTDB时序数据库：时序数据分析的核心组件，提供海量时序数据的存储和计算能力。
  - 离线加载：IoTDB提供的数据导入工具，可以将本地csv文件批量导入到IoTDB中。
  - 数据访问：提供JDBC/Restful的访问接口供应用程序访问，也可以直接提供IoTDB的生态插件与Grafana对接。
- 业务应用：查询并使用时序数据库中的数据的业务应用。

## 2.9.2 手动将 CSV 离线数据导入至 IoTDB

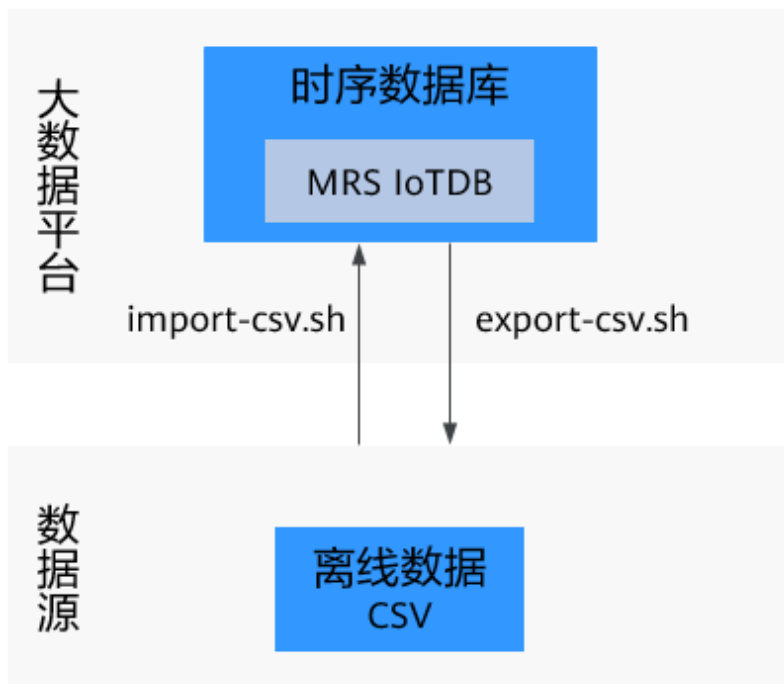
### 应用场景

IoTDB（物联网数据库）是一体化收集、存储、管理与分析物联网时序数据的软件系统，采用轻量式架构，具有高性能和丰富的功能。

IoTDB支持以下方式批量导入或导出数据：

- 批量导入数据：通过import-csv.sh离线方式进行数据的导入。
- 批量导出数据：通过export-csv.sh离线方式进行数据的导出。

图 2-50 导入导出 IoTDB 数据

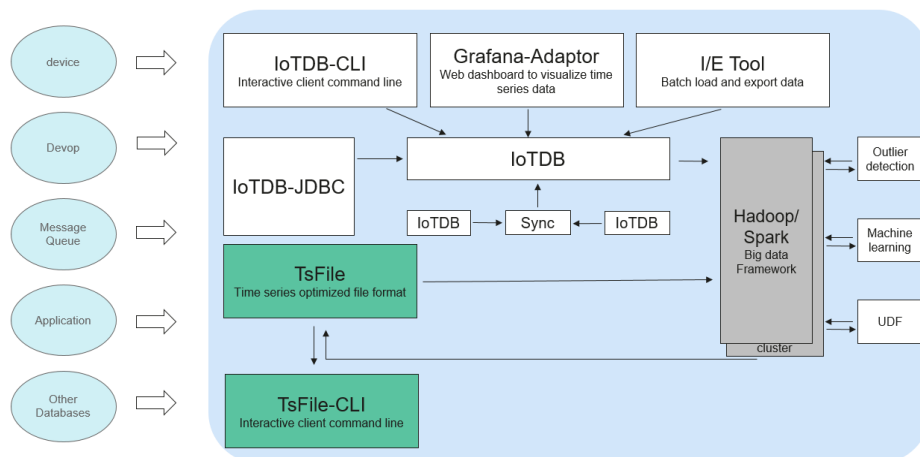


### 方案架构

IoTDB套件由若干个组件构成，共同形成数据收集、数据写入、数据存储、数据查询、数据可视化、数据分析等一系列功能。

图2-51展示了使用IoTDB套件的全部组件形成的整体应用架构，IoTDB特指其中的时间序列数据库组件。

图 2-51 IoTDB 结构



- 用户可以通过 JDBC/Session 将来自设备传感器上采集的时序数据和服务器负载、CPU 内存等系统状态数据、消息队列中的时序数据、应用程序的时序数据或者其他数据库中的时序数据导入到本地或者远程的 IoTDB 中。用户还可以将上述数据直接写成本地（或位于 HDFS 上）的 TsFile 文件。
- 用户可以将 TsFile 文件写入到 HDFS 上，进而满足 Hadoop、Flink 等数据处理任务的访问。
- 对于写入到 HDFS 或者本地的 TsFile 文件，可以利用 TsFile-Hadoop 或 TsFile-Flink 连接器，允许 Hadoop 或 Flink 进行数据处理。
- 对于分析的结果，可以写回成 TsFile 文件。
- IoTDB 和 TsFile 还提供了相应的客户端工具，满足用户以 SQL 形式、脚本形式和图形形式写入和查看数据的各种需求。

## 操作流程

- **步骤1：创建 MRS 集群：** 创建一个包含有 IoTDB 组件的 MRS 集群。
- **步骤2：创建 IoTDB 用户：** 在 Manager 页面创建具有操作 IoTDB 表的权限的用户。
- **步骤3：导入 CSV 文件数据到 IoTDB 表中：** 使用 import-csv.sh 脚本导入 CSV 文件数据到 IoTDB 表中。
- **步骤4：导出 IoTDB 数据到 CSV 文件中：** 使用 export-csv.sh 脚本导出 IoTDB 表数据到 CSV 文件中。

### 步骤 1：创建 MRS 集群

**步骤1** 创建一个包含有 IoTDB 组件的 MRS 集群，详情请参见[购买自定义集群](#)。

#### 📖 说明

本文以自定义购买的 MRS 3.2.0-LTS.1 版本的集群为例，集群已开启 Kerberos 认证。

**步骤2** 集群购买成功后，在 MRS 集群的任一节点内，安装集群客户端，具体操作可参考[安装并使用集群客户端](#)。

例如客户端安装目录为“/opt/client”。

**步骤3** 关闭 IoTDB 组件的 SSL 配置。

1. 登录FusionInsight Manager，选择“集群 > 服务 > IoTDB > 配置”，搜索“SSL\_ENABLE”，如果该参数值为“true”，需修改为“false”并单击“保存”，关闭SSL配置。
2. 修改IoTDB客户端对应的参数关闭SSL配置：  
`cd 客户端安装目录/IoTDB/iotdb/conf`  
`vi iotdb-client.env`  
修改“iotdb\_ssl\_enable”参数值为“false”，保存并退出。
3. 在FusionInsight Manager界面，选择“集群 > 服务 > IoTDB”，选择页面右上角的“更多 > 重启服务”，输入当前用户密码并单击“确定”使配置生效。

----结束

## 步骤 2: 创建 IoTDB 用户

对于开启Kerberos认证的MRS集群，需提前准备具有相关组件操作权限的用户用于安全认证。

**步骤1** 登录FusionInsight Manager，选择“系统 > 权限 > 角色 > 添加角色”，配置以下参数并单击“确定”：

- 角色名称：填写角色的名称，例如**testrole**。
- 配置资源权限：在“配置资源权限”的表格中选择“待操作集群的名称 > IoTDB > 普通用户权限”，勾选**root**根目录的“设置数据库”权限；再单击“root”，选择对应的存储组，勾选“创建”、“修改”、“写”、“读”权限。

**步骤2** 选择“用户 > 添加用户”，在新增用户界面，创建一个人机用户，例如**testuser**。

- “用户组”需加入“iotdbgroup”用户组。
- “角色”加入**步骤1**新增的角色。

**步骤3** 使用新创建的**testuser**用户重新登录FusionInsight Manager，修改该用户的初始密码。

----结束

## 步骤 3: 导入 CSV 文件数据到 IoTDB 表中

- 按单设备时间有序地导入单文件多设备多测点数据，数据中存在空值
  - a. 在本地准备CSV文件，例如文件名为“y1.csv”，文件中只包含一个设备的所有数据，且数据中存在空值，内容如下：

```
Time,root.y1.r1.s1.d1.value,root.y1.r1.s1.d1.temperature,root.y1.r1.s1.d1.humidity,root.y1.r1.s1.d1.s
tatus,root.y1.r1.s1.d2.value,root.y1.r1.s1.d2.temperature,root.y1.r1.s1.d2.humidity,root.y1.r1.s1.d2.s
tatus,root.y1.r1.s1.d3.value,root.y1.r1.s1.d3.temperature,root.y1.r1.s1.d3.humidity,root.y1.r1.s1.d3.s
tatus,root.y1.r1.s1.d4.value,root.y1.r1.s1.d4.temperature,root.y1.r1.s1.d4.humidity,root.y1.r1.s1.d4.s
tatus
2023-12-01 22:22:00,9426,43,30,False,5797,79,33,True,8740,29,53,True,9378,,,False
2023-12-02 10:09:19,2969,73,84,False,6017,80,42,False,6624,18,19,False,6903,,,True
2023-12-03 08:57:37,7553,74,41,True,1857,19,25,False,2131,93,36,True,4421,26,34,False
2023-12-04 23:49:51,4417,67,16,False,7098,18,5,True,9887,15,90,True,9670,87,68,False
2023-12-05 13:08:15,7708,69,55,False,8304,47,13,False,5877,100,87,True,8671,31,29,False
2023-12-06 12:05:09,3573,70,75,True,5674,80,38,False,3408,21,24,False,4557,47,37,False
2023-12-07 11:41:04,4506,67,81,True,4445,62,37,False,6847,20,39,True,1018,43,16,False
2023-12-08 00:15:47,48,84,94,False,1055,70,1,True,7625,76,69,True,8430,80,91,True
2023-12-09 15:55:22,2187,87,30,True,9367,84,2,False,1565,14,3,False,4465,62,51,True
2023-12-10 17:42:02,4099,0,59,False,5532,29,81,False,4153,29,29,True,4418,17,4,False
```
  - b. 将CSV文件上传到安装客户端的节点中，例如“客户端安装目录/IoTDB/iotdb/tools”目录下。

- c. 不创建元数据，直接执行脚本导入CSV文件到IoTDB数据库中：

```
cd 客户端安装目录
```

```
source bigdata_env
```

```
kinit 组件业务用户
```

```
cd IoTDB/iotdb/tools
```

```
./import-csv.sh -h IoTDBServer的业务IP地址 -p IoTDBServer RPC端口 -f  
文件名
```

命令执行后根据提示交互式输入业务用户名和对应密码。

#### 说明

- IoTDBServer实例节点的业务IP地址可在Manager界面，选择“集群 > 服务 > IoTDB > 实例”查看。
  - IoTDBServer RPC端口可在Manager界面，选择“集群 > 服务 > IoTDB > 配置”，在参数“IOTDB\_SERVER\_RPC\_PORT”中自行配置。默认端口如下：
    - 开源端口默认值为：6667
    - 定制端口默认值为：22260
- 端口定制/开源区分：创建LTS版本类型集群时，可以选择“组件端口”为“开源”或是“定制”，选择“开源”使用开源端口，选择“定制”使用定制端口。
- f: 用于指定想要导入的数据，可以指定文件或者文件夹。如果指定的是文件夹，将会把文件夹中所有后缀为txt与csv的文件进行批量导入。例如：**-f filename.csv**。
  - fd: 用于指定一个目录来存放导入失败的文件，如果没有指定该参数，失败的文件将会被保存到源数据的目录中，文件名为源文件名加上.failed后缀。例如：**-fd ./failed/**。
  - aligned: 是否使用aligned接口，默认参数值为“false”。例如：**-aligned true**。
  - tp: 用于指定时间精度，可选值包括ms（毫秒）、ns（纳秒）、us（微秒），默认值为ms。
  - linesPerFailedFile <int>: 用于指定每个导入失败的文件写入数据的行数，默认值为10000，例如：**-linesPerFailedFile 1**。

执行结果中包含以下以下信息则表示数据导入成功：

```
Import completely!
```

- d. 执行以下命令登录IoTDB客户端：

```
cd 客户端安装目录/IoTDB/iotdb/sbin
```

```
./start-cli.sh -h IoTDBServer实例节点的业务IP地址 -p IoTDBServer RPC端  
口
```

运行该命令后，根据实际需求指定业务用户名：

- 指定业务用户名，则输入“yes”，并根据提示输入业务用户名和对应的业务用户密码：



```
IoTDB      sbin# ./start-cli.sh -h 192.168.34.21 -p 22260
do you want to specify your own user(yes/no):yes
Please Enter username:
Please Enter password:*****
15:39:28.493 [main] INFO org.apache.iotdb.jdbc.IoTDBConnection - Attempt to connect 192.168.34.21:22260
15:39:28.498 [main] WARN com.aliyun.iotdb.rpc.ssl.SSLContextFactory - Could not load property 'iotdb.security.ssl.config' from system.
15:39:28.498 [main] INFO com.aliyun.iotdb.rpc.ssl.SSLContextFactory - iotdb_ssl_enable is false in config, disabling SSL
Starting IoTDB cli
.....
IoTDB version
IoTDB@192.168.34.21:22260> login successfully
IoTDB@192.168.34.21:22260>
```





```
2023-11-05 01:54:33,7942,70,14,False
2023-11-06 20:36:14,5047,30,3,True
2023-11-07 14:20:10,1434,4,87,True
2023-11-08 03:56:57,8651,39,61,True
2023-11-09 11:58:08,7109,88,64,True
2023-11-10 10:58:13,2333,41,40,False
```

- **y3d1.csv:**

```
Time,root.y3.r1.s1.d1.value,root.y3.r1.s1.d1.temperature,root.y3.r1.s1.d1.humidity,root.y3.r1.s1.d1.status
2023-11-01 06:34:19,6621,25,83,True
2023-11-02 06:28:18,4066,98,46,False
2023-11-03 07:44:50,7909,58,37,True
2023-11-04 15:27:01,837,42,27,False
2023-11-05 01:54:33,7942,70,14,False
2023-11-06 20:36:14,5047,30,3,True
2023-11-07 14:20:10,1434,4,87,True
2023-11-08 03:56:57,8651,39,61,True
2023-11-09 11:58:08,7109,88,64,True
2023-11-10 10:58:13,2333,41,40,False
```

- **y3d2.csv:**

```
Time,root.y3.r1.s1.d2.value,root.y3.r1.s1.d2.temperature,root.y3.r1.s1.d2.humidity,root.y3.r1.s1.d2.status
2023-11-01 06:34:19,6621,25,83,True
2023-11-02 06:28:18,4066,98,46,False
2023-11-03 07:44:50,7909,58,37,True
2023-11-04 15:27:01,837,42,27,False
2023-11-05 01:54:33,7942,70,14,False
2023-11-06 20:36:14,5047,30,3,True
2023-11-07 14:20:10,1434,4,87,True
2023-11-08 03:56:57,8651,39,61,True
2023-11-09 11:58:08,7109,88,64,True
2023-11-10 10:58:13,2333,41,40,False
```

- 在“客户端安装目录/loTDB/iotdb/tools”目录下创建“d1”目录，并将准备好的4个CSV文件上传到该目录中。
- 不创建元数据，直接执行脚本导入CSV文件到loTDB数据库中：

```
cd 客户端安装目录
```

```
source bigdata_env
```

```
kinit 组件业务用户
```

```
cd loTDB/iotdb/tools
```

```
./import-csv.sh -h loTDBServer的业务IP地址 -p loTDBServer RPC端口 -f d1
```

命令执行后根据提示交互式输入业务用户名和对应密码。

## 说明

- IoTDBServer实例节点的业务IP地址可在Manager界面，选择“集群 > 服务 > IoTDB > 实例”查看。
  - IoTDBServer RPC端口可在参数“IOTDB\_SERVER\_RPC\_PORT”中自行配置。默认端口如下：
    - 开源端口默认值为：6667
    - 定制端口默认值为：22260
- 端口定制/开源区分：创建LTS版本类型集群时，可以选择“组件端口”为“开源”或是“定制”，选择“开源”使用开源端口，选择“定制”使用定制端口。
- -f: 用于指定想要导入的数据，可以指定文件或者文件夹。如果指定的是文件夹，将会把文件夹中所有后缀为txt与csv的文件进行批量导入。例如: **-f d1**。
  - -fd: 用于指定一个目录来存放导入失败的文件，如果没有指定该参数，失败的文件将会被保存到源数据的目录中，文件名为源文件名加上.failed后缀。例如: **-fd ./failed/**。
  - -aligned: 是否使用aligned接口，默认参数值为“false”。例如: **-aligned true**。
  - -tp: 用于指定时间精度，可选值包括ms（毫秒）、ns（纳秒）、us（微秒），默认值为ms。
  - -linesPerFailedFile <int>: 用于指定每个导入失败的文件写入数据的行数，默认值为10000，例如: **-linesPerFailedFile 1**。

执行结果中包含以下以下信息则表示数据导入成功：

```
Import completely!
```

- d. 执行以下命令登录IoTDB客户端：

```
cd 客户端安装目录/IoTDB/iotdb/sbin
```

```
./start-cli.sh -h IoTDBServer实例节点的业务IP地址 -p IoTDBServer RPC端口
```

运行该命令后，根据实际需求指定业务用户名：

- 指定业务用户名，则输入“yes”，并根据提示输入业务用户名和对应的业务用户密码：

```
root@ ~ # sbin# ./start-cli.sh -h 192.168.34.21 -p 22260
do you want to specify your own user(yes/no):yes
Please Enter username:
Please Enter password:*****
15:39:28.403 [main] INFO org.apache.iotdb.jdbc.IoTDBConnection - Attempt to connect 192.168.34.21:22260
15:39:28.408 [main] WARN com... .iotdb.rpc.ssl.SSLContextFactory - Could not load property 'iotdb.security.ssl.config' from system.
15:39:28.408 [main] INFO com... .iotdb.rpc.ssl.SSLContextFactory - iotdb_ssl_enable is false in config, disabling SSL
Starting IoTDB Cli
.....
IoTDB version
IoTDB@ :22260> login successfully
IoTDB@ :22260>
```

- 不指定业务用户名，则输入“no”；此时，则使用进行安全认证的用户执行后续操作：

```
root@ ~ # sbin# ./start-cli.sh -h 192.168.34.21 -p 22260
do you want to specify your own user(yes/no):no
15:31:06.569 [main] INFO org.apache.iotdb.jdbc.IoTDBConnection - Attempt to connect 192.168.34.21:22260
15:31:06.574 [main] WARN com... .iotdb.rpc.ssl.SSLContextFactory - Could not load property 'iotdb.security.ssl.config' from system.
15:31:06.575 [main] INFO com... .iotdb.rpc.ssl.SSLContextFactory - iotdb_ssl_enable is false in config, disabling SSL
Starting IoTDB Cli
.....
IoTDB version
IoTDB@ :22260> login successfully
```

- 输入其它，则退出登录：

```
[root@host-192.168.0.141 ~]# sbin]# ./start-cli.sh -h -p 22260
do you want to specify your own user(yes/no):asda
Exit.
```

e. 执行以下命令查看导入的数据:

```
select * from root.y2.r1.s1.d1;
select * from root.y2.r1.s1.d2;
select * from root.y3.r1.s1.d1;
select * from root.y3.r1.s1.d2;
查询导入数据正常。
```

图 2-53 数据导入成功

```
IoTDB@192.168.0.141:6667> select * from root.y2.r1.s1.d1;
+-----+-----+-----+-----+-----+
| Time|root.y2.r1.s1.d1.temperature|root.y2.r1.s1.d1.humidity|root.y2.r1.s1.d1.value|root.y2.r1.s1.d1.status|
+-----+-----+-----+-----+-----+
| 2023-11-01T06:34:19.000+08:00| 25.0| 83.0| 6621.0| true|
| 2023-11-02T06:28:18.000+08:00| 98.0| 46.0| 4066.0| false|
| 2023-11-03T07:44:50.000+08:00| 58.0| 37.0| 7909.0| true|
| 2023-11-04T15:27:01.000+08:00| 42.0| 27.0| 837.0| false|
| 2023-11-05T01:54:33.000+08:00| 70.0| 14.0| 7942.0| false|
| 2023-11-06T20:36:14.000+08:00| 30.0| 3.0| 5047.0| true|
| 2023-11-07T14:20:10.000+08:00| 4.0| 14.0| 1434.0| true|
| 2023-11-08T03:56:57.000+08:00| 39.0| 61.0| 8651.0| true|
| 2023-11-09T11:58:08.000+08:00| 88.0| 64.0| 7109.0| true|
| 2023-11-10T10:58:13.000+08:00| 41.0| 40.0| 2333.0| false|
+-----+-----+-----+-----+-----+
Total line number = 10
It costs 0.046s
IoTDB@192.168.0.141:6667> select * from root.y2.r1.s1.d2;
+-----+-----+-----+-----+-----+
| Time|root.y2.r1.s1.d2.temperature|root.y2.r1.s1.d2.humidity|root.y2.r1.s1.d2.value|root.y2.r1.s1.d2.status|
+-----+-----+-----+-----+-----+
| 2023-11-01T06:34:19.000+08:00| 25.0| 83.0| 6621.0| true|
| 2023-11-02T06:28:18.000+08:00| 98.0| 46.0| 4066.0| false|
| 2023-11-03T07:44:50.000+08:00| 58.0| 37.0| 7909.0| true|
| 2023-11-04T15:27:01.000+08:00| 42.0| 27.0| 837.0| false|
| 2023-11-05T01:54:33.000+08:00| 70.0| 14.0| 7942.0| false|
| 2023-11-06T20:36:14.000+08:00| 30.0| 3.0| 5047.0| true|
| 2023-11-07T14:20:10.000+08:00| 4.0| 14.0| 1434.0| true|
| 2023-11-08T03:56:57.000+08:00| 39.0| 61.0| 8651.0| true|
| 2023-11-09T11:58:08.000+08:00| 88.0| 64.0| 7109.0| true|
| 2023-11-10T10:58:13.000+08:00| 41.0| 40.0| 2333.0| false|
+-----+-----+-----+-----+-----+
Total line number = 10
It costs 0.034s
IoTDB@192.168.0.141:6667> select * from root.y3.r1.s1.d1;
+-----+-----+-----+-----+-----+
| Time|root.y3.r1.s1.d1.temperature|root.y3.r1.s1.d1.humidity|root.y3.r1.s1.d1.value|root.y3.r1.s1.d1.status|
+-----+-----+-----+-----+-----+
| 2023-11-01T06:34:19.000+08:00| 25.0| 83.0| 6621.0| true|
| 2023-11-02T06:28:18.000+08:00| 98.0| 46.0| 4066.0| false|
| 2023-11-03T07:44:50.000+08:00| 58.0| 37.0| 7909.0| true|
| 2023-11-04T15:27:01.000+08:00| 42.0| 27.0| 837.0| false|
| 2023-11-05T01:54:33.000+08:00| 70.0| 14.0| 7942.0| false|
| 2023-11-06T20:36:14.000+08:00| 30.0| 3.0| 5047.0| true|
| 2023-11-07T14:20:10.000+08:00| 4.0| 14.0| 1434.0| true|
| 2023-11-08T03:56:57.000+08:00| 39.0| 61.0| 8651.0| true|
| 2023-11-09T11:58:08.000+08:00| 88.0| 64.0| 7109.0| true|
| 2023-11-10T10:58:13.000+08:00| 41.0| 40.0| 2333.0| false|
+-----+-----+-----+-----+-----+
Total line number = 10
It costs 0.031s
IoTDB@192.168.0.141:6667> select * from root.y3.r1.s1.d2;
+-----+-----+-----+-----+-----+
| Time|root.y3.r1.s1.d2.temperature|root.y3.r1.s1.d2.humidity|root.y3.r1.s1.d2.value|root.y3.r1.s1.d2.status|
+-----+-----+-----+-----+-----+
| 2023-11-01T06:34:19.000+08:00| 25.0| 83.0| 6621.0| true|
| 2023-11-02T06:28:18.000+08:00| 98.0| 46.0| 4066.0| false|
| 2023-11-03T07:44:50.000+08:00| 58.0| 37.0| 7909.0| true|
| 2023-11-04T15:27:01.000+08:00| 42.0| 27.0| 837.0| false|
| 2023-11-05T01:54:33.000+08:00| 70.0| 14.0| 7942.0| false|
| 2023-11-06T20:36:14.000+08:00| 30.0| 3.0| 5047.0| true|
| 2023-11-07T14:20:10.000+08:00| 4.0| 14.0| 1434.0| true|
| 2023-11-08T03:56:57.000+08:00| 39.0| 61.0| 8651.0| true|
| 2023-11-09T11:58:08.000+08:00| 88.0| 64.0| 7109.0| true|
| 2023-11-10T10:58:13.000+08:00| 41.0| 40.0| 2333.0| false|
+-----+-----+-----+-----+-----+
```

● 按时间有序地导入单文件内单设备多测点数据，导入数据前创建元数据

a. 在本地准备CSV文件，文件名为“y4.csv”，文件中只包含一个设备的所有数据，内容如下:

```
Time,root.y4.r1.s1.d1.value,root.y4.r1.s1.d1.temperature,root.y4.r1.s1.d1.humidity,root.y4.r1.s1.d1.s
tatus
2023-11-01 06:34:19,6621,25,83,True
2023-11-02 06:28:18,4066,98,46,False
2023-11-03 07:44:50,7909,58,37,True
2023-11-04 15:27:01,837,42,27,False
2023-11-05 01:54:33,7942,70,14,False
2023-11-06 20:36:14,5047,30,3,True
2023-11-07 14:20:10,1434,4,87,True
2023-11-08 03:56:57,8651,39,61,True
2023-11-09 11:58:08,7109,88,64,True
2023-11-10 10:58:13,2333,41,40,False
```

b. 将CSV文件上传至客户端所在节点，例如“客户端安装目录/iotdb/tools”目录下。

c. 执行以下命令登录IoTDB客户端:

```
cd 客户端安装目录
```

```
source bigdata_env
```

```
kinit 组件业务用户
```

```
cd IoTDB/iotdb/sbin
```

```
./start-cli.sh -h IoTDBServer实例节点的业务IP地址 -p IoTDBServer RPC端口
```

运行该命令后，根据实际需求指定业务用户名：

- 指定业务用户名，则输入“yes”，并根据提示输入业务用户名和对应的业务用户密码：

```
[root@ ~]# sbin# ./start-cli.sh -h 192.168.34.21 -p 22260
do you want to specify your own user(yes/no):yes
Please Enter username:
Please Enter password:*****
15:39:28.403 [main] INFO org.apache.iotdb.jdbc.IoTDBConnection - Attempt to connect 192.168.34.21:22260
15:39:28.408 [main] WARN com.iotdb.rpc.ssl.SSLContextFactory - Could not load property 'iotdb.security.ssl.config' from system.
15:39:28.408 [main] INFO com.iotdb.rpc.ssl.SSLContextFactory - iotdb_ssl_enable is false in config, disabling SSL
Starting IoTDB Cli
.....
IoTDB version
IoTDB@ :22260> login successfully
IoTDB@ :22260>
```

- 不指定业务用户名，则输入“no”；此时，则使用进行安全认证的用户执行后续操作：

```
[root@host-1 ~]# sbin# ./start-cli.sh -h 192.168.34.21 -p 22260
do you want to specify your own user(yes/no):no
15:31:06.569 [main] INFO org.apache.iotdb.jdbc.IoTDBConnection - Attempt to connect 192.168.34.21:22260
15:31:06.574 [main] WARN com.iotdb.rpc.ssl.SSLContextFactory - Could not load property 'iotdb.security.ssl.config' from system.
15:31:06.575 [main] INFO com.iotdb.rpc.ssl.SSLContextFactory - iotdb_ssl_enable is false in config, disabling SSL
Starting IoTDB Cli
.....
IoTDB version
IoTDB@ :22260> login successfully
```

- 输入其它，则退出登录：

```
[root@host-1 ~]# sbin# ./start-cli.sh -h 192.168.34.21 -p 22260
do you want to specify your own user(yes/no):asda
Exit.
```

- d. 执行以下命令在导入数据前创建元数据：

```
SET STORAGE GROUP TO root.y4.r1.s1.d1;
```

```
CREATE TIMESERIES root.y4.r1.s1.d1.value WITH
DATATYPE=DOUBLE,ENCODING=RLE;
```

```
CREATE TIMESERIES root.y4.r1.s1.d1.temperature WITH
DATATYPE=DOUBLE,ENCODING=PLAIN;
```

```
CREATE TIMESERIES root.y4.r1.s1.d1.humidity WITH
DATATYPE=DOUBLE,ENCODING=RLE;
```

```
CREATE TIMESERIES root.y4.r1.s1.d1.status WITH
DATATYPE=BOOLEAN,ENCODING=RLE;
```

- e. 执行以下命令，退出客户端。

```
quit;
```

- f. 执行脚本导入CSV文件到IoTDB数据库中：

```
cd 客户端安装目录/IoTDB/iotdb/tools
```

```
./import-csv.sh -h IoTDBServer实例节点的业务IP地址 -p IoTDBServer RPC端口 -f 文件名
```

命令执行后根据提示交互式输入业务用户名和对应密码。

### 说明

- IoTDBServer实例节点的业务IP地址可在Manager界面，选择“集群 > 服务 > IoTDB > 实例”查看。
- IoTDBServer RPC端口可在Manager界面，选择“集群 > 服务 > IoTDB > 配置”，在参数“IOTDB\_SERVER\_RPC\_PORT”中自行配置。默认端口如下：
  - 开源端口默认值为：6667
  - 定制端口默认值为：22260

端口定制/开源区分：创建LTS版本类型集群时，可以选择“组件端口”为“开源”或是“定制”，选择“开源”使用开源端口，选择“定制”使用定制端口。

- -f: 用于指定想要导入的数据，可以指定文件或者文件夹。如果指定的是文件夹，将会把文件夹中所有后缀为txt与csv的文件进行批量导入。例如: `-f filename.csv`。
- -fd: 用于指定一个目录来存放导入失败的文件，如果没有指定该参数，失败的文件将会被保存到源数据的目录中，文件名为源文件名加上.failed后缀。例如: `-fd ./failed/`。
- -aligned: 是否使用aligned接口，默认参数值为“false”。例如: `-aligned true`。
- -tp: 用于指定时间精度，可选值包括ms（毫秒）、ns（纳秒）、us（微秒），默认值为ms。
- -linesPerFailedFile <int>: 用于指定每个导入失败的文件写入数据的行数，默认值为10000，例如: `-linesPerFailedFile 1`。

执行结果中包含以下以下信息则表示数据导入成功：

```
Import completely!
```

- 重新登录IoTDB客户端，执行下命令查看导入的数据：

```
select * from root.y4.r1.s1.d1;
```

查询导入数据正常。

图 2-54 数据导入成功

```
IoTDB@192.168.0.141:6667> select * from root.y4.r1.s1.d1;
+-----+-----+-----+-----+-----+
| Time|root.y4.r1.s1.d1.temperature|root.y4.r1.s1.d1.humidity|root.y4.r1.s1.d1.value|root.y4.r1.s1.d1.status|
+-----+-----+-----+-----+-----+
|2023-11-01T06:34:19.000+08:00|25.0|83.0|6621.0|true|
|2023-11-02T06:28:18.000+08:00|98.0|46.0|4866.0|false|
|2023-11-03T07:44:50.000+08:00|58.0|37.0|7909.0|true|
|2023-11-04T15:27:01.000+08:00|42.0|27.0|837.0|false|
|2023-11-05T01:54:33.000+08:00|76.0|14.0|7842.0|false|
|2023-11-06T20:36:14.000+08:00|39.0|3.0|5047.0|true|
|2023-11-07T14:20:10.000+08:00|4.0|87.0|1434.0|true|
|2023-11-08T03:56:57.000+08:00|39.0|61.0|8051.0|true|
|2023-11-09T11:58:00.000+08:00|98.0|64.0|7106.0|true|
|2023-11-10T10:58:13.000+08:00|41.0|40.0|2333.0|false|
+-----+-----+-----+-----+-----+
```

- 按时间有序地导入指定数据库下单设备单测点的数据
  - 在本地准备CSV文件，文件名为“y5.csv”，文件中只包含一个设备的所有数据，内容如下：
 

```
Time,root.y5.r1.s1.d1.temperature
2023-12-01 10:22:58,39
2023-12-02 03:26:55,51
2023-12-04 08:21:48,38
2023-12-05 04:08:12,54
2023-12-06 08:00:56,90
2023-12-07 19:27:50,48
2023-12-08 22:52:46,64
2023-12-09 14:52:12,78
2023-12-10 12:52:40,33
2023-12-11 13:47:16,58
```
  - 将CSV文件上传到安装了客户端的节点中，例如“客户端安装目录/IoTDB/iotdb/tools”目录下。
  - 执行脚本导入CSV文件到IoTDB数据库中：
 

```
cd 客户端安装目录
```

```
source bigdata_env
```

```
kinit 组件业务用户
```

```
cd IoTDB/iotdb/tools
```

```
./import-csv.sh -h IoTDBServer实例节点的业务IP地址 -p IoTDBServer RPC  
端口 -f 文件名
```

命令执行后根据提示交互式输入业务用户名和对应密码。

### 📖 说明

- IoTDBServer实例节点的业务IP地址可在Manager界面，选择“集群 > 服务 > IoTDB > 实例”查看。
- IoTDBServer RPC端口可在Manager界面，选择“集群 > 服务 > IoTDB > 配置”，在参数“IOTDB\_SERVER\_RPC\_PORT”中自行配置。默认端口如下：
  - 开源端口默认值为：6667
  - 定制端口默认值为：22260

端口定制/开源区分：创建LTS版本类型集群时，可以选择“组件端口”为“开源”或是“定制”，选择“开源”使用开源端口，选择“定制”使用定制端口。

- -f: 用于指定想要导入的数据，可以指定文件或者文件夹。如果指定的是文件夹，将会把文件夹中所有后缀为txt与csv的文件进行批量导入。例如: -f *filename.csv*。
- -fd: 用于指定一个目录来存放导入失败的文件，如果没有指定该参数，失败的文件将会被保存到源数据的目录中，文件名为源文件名加上.failed后缀。例如: -fd *./failed/*。
- -aligned: 是否使用aligned接口，默认参数值为“false”。例如: -aligned **true**。
- -tp: 用于指定时间精度，可选值包括ms（毫秒）、ns（纳秒）、us（微秒），默认值为ms。
- -linesPerFailedFile <int>: 用于指定每个导入失败的文件写入数据的行数，默认值为10000，例如: -linesPerFailedFile **1**。

执行结果中包含以下以下信息则表示数据导入成功：

```
Import completely!
```

d. 执行以下命令登录IoTDB客户端：

```
cd 客户端安装目录/IoTDB/iotdb/sbin
```

```
./start-cli.sh -h IoTDBServer实例节点的业务IP地址 -p IoTDBServer RPC端  
口
```

运行该命令后，根据实际需求指定业务用户名：

- 指定业务用户名，则输入“yes”，并根据提示输入业务用户名和对应的业务用户密码：

```
IoTDB      sbin# ./start-cli.sh -h          -p 22260
do you want to specify your own user(yes/no):yes
Please Enter username:
Please Enter password:*****
15:39:28.493 [main] INFO org.apache.iotdb.jdbc.IoTDBConnection - Attempt to connect 192.168.34.21:22260
15:39:28.498 [main] WARN com... .iotdb.rpc.ssl.SSLContextFactory - Could not load property 'iotdb.security.ssl.config' from system.
15:39:28.498 [main] INFO com... .iotdb.rpc.ssl.SSLContextFactory - iotdb_ssl_enable is false in config, disabling SSL
Starting IoTDB CLI
.....
IoTDB
.....
version
IoTDB@:22260> login successfully
IoTDB@:22260>
```

- 不指定业务用户名，则输入“no”；此时，则使用进行安全认证的用户执行后续操作：

```
[root@host-1 ~]# ./start-cli.sh -h -p 22260
do you want to specify your own user(yes/no):no
15:31:06.569 [main] INFO org.apache.iotdb.jdbc.IoTDBConnection - Attempt to connect "jdbc:iotdb://192.168.0.141:22260"
15:31:06.574 [main] WARN com.aliyun.iotdb.rpc.ssl.SSLContextFactory - Could not load property 'iotdb.security.ssl.config' from system
15:31:06.575 [main] INFO com.aliyun.iotdb.rpc.ssl.SSLContextFactory - iotdb_ssl_enable is false in config, disabling SSL
Starting IoTDB Cli
IoTDB version 3.14.0
IoTDB@192.168.0.141:22260 login successfully
```

- 输入其它，则退出登录：

```
[root@host-1 ~]# ./start-cli.sh -h -p 22260
do you want to specify your own user(yes/no):asda
Exit.
```

- e. 执行以下命令查看导入的数据：  
**select \* from root.y5.r1.s1.d1;**  
查询导入数据正常。

图 2-55 数据导入成功

```
IoTDB@192.168.0.141:6667> select * from root.y5.r1.s1.d1;
+-----+-----+
|                Time|root.y5.r1.s1.d1.temperature|
+-----+-----+
|2023-12-01T10:22:58.000+08:00|          39.0|
|2023-12-02T03:26:55.000+08:00|          51.0|
|2023-12-04T08:21:48.000+08:00|          38.0|
|2023-12-05T04:08:12.000+08:00|          54.0|
|2023-12-06T08:00:56.000+08:00|          90.0|
|2023-12-07T19:27:50.000+08:00|          48.0|
|2023-12-08T22:52:46.000+08:00|          64.0|
|2023-12-09T14:52:12.000+08:00|          78.0|
|2023-12-10T12:52:40.000+08:00|          33.0|
|2023-12-11T13:47:16.000+08:00|          58.0|
+-----+-----+
```

## 步骤 4：导出 IoTDB 数据到 CSV 文件中

**步骤1** 以客户端安装用户登录安装了客户端的节点，执行以下命令配置环境变量并认证用户：

```
cd 客户端安装目录
```

```
source bigdata_env
```

```
kinit 组件业务用户
```

**步骤2** 执行以下命令创建存放导出数据的目录“e1”：

```
mkdir ./e1
```

**步骤3** 切换到数据导出脚本所在目录：

```
cd IoTDB/iotdb/tools
```

**步骤4** （可选）新建一个SQL文件，例如文件名为“s1.sql”，文件中包含一条或多条SQL语句。例如，导出指定数据库下设备为d1、物理量为temperature的所有数据的SQL语句为：

```
select temperature from root.y1.r1.s1.d1;
```

**步骤5** 执行以下数据导出脚本导出IoTDB数据：

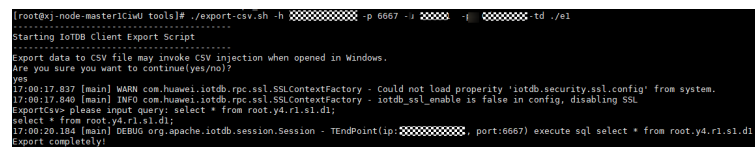
```
./export-csv.sh -h IoTDBServer实例节点的业务IP地址 -p IoTDBServer RPC端口 -  
td ./e1 -s s1.sql
```

命令运行后会出现CSV注入风险提示，输入“yes”继续执行命令，输入其他，则取消数据导出操作。

如果数据导出命令中没有设置-s参数，则输入yes后，还需输入要导出数据的查询语句，不同的查询结果会被保存到不同的CSV文件中。例如，导出指定数据库下设备为d1的所有数据，则命令为：

```
select * from root.y4.r1.s1.d1;
```

图 2-56 导出 IoTDB 数据至 CSV 文件中



```
root@ky-node-master1cid:~# ./export-csv.sh -h 10.10.10.10 -p 6667 -td ./e1 -s s1.sql
Starting IoTDB Client Export Script
Export data to CSV file may invoke CSV injection when opened in Windows.
Are you sure you want to continue(yes/no)?
yes
17:00:17.837 [main] WARN com.huawei.iotdb.rpc.ssl.SSLContextFactory - Could not load property 'iotdb.security.ssl.config' from system.
17:00:17.840 [main] INFO com.huawei.iotdb.rpc.ssl.SSLContextFactory - iotdb_ssl_enable is false in config, disabling SSL
Export csv please input query: select * from root.y4.r1.s1.d1;
select * from root.y4.r1.s1.d1;
17:00:20.184 [main] DEBUG org.apache.iotdb.session.Session - TEndPoint(ip:10.10.10.10, port:6667) execute sql select * from root.y4.r1.s1.d1
Export completely.
```

其他相关的数据导出参数如下：

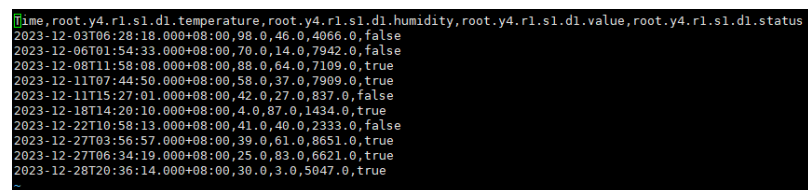
- -datatype:
  - 默认为“true”，表示在CSV文件的header中打印出时间序列对应的数据类型。例如：  
Time, root.sg1.d1.s1(INT32), root.sg1.d1.s2(INT64)
  - 值为“false”时，表示只在CSV的header中打印出时间序列名称，例如：  
Time, root.sg1.d1.s1, root.sg1.d1.s2
- -s <sql file>: 用于指定一个SQL文件，里面包含一条或多条SQL语句。如果一个SQL文件中包含多条SQL语句，SQL语句之间应该用换行符进行分隔。每一条SQL语句对应一个输出的CSV文件。
- -td <directory>: 用于为导出的CSV文件指定输出路径。
- -tf <time-format>: 用于指定时间格式，时间格式必须遵守ISO 8601标准。如果想要以时间戳方式来保存时间，需设置为-tf timestamp。例如，-tf yyyy-MM-dd\ HH:mm:ss或-tf timestamp。
- -linesPerFile <int>: 用于指定导出的dump文件的最大行数，默认值为“10000”。例如，-linesPerFile 1。
- -t <timeout>: 用于指定session查询时的超时时间，单位为ms。

步骤6 切换到数据存放目录，查看导出的CSV文件的数据情况：

```
vim dump0_0.csv
```

例如，查看导出指定数据库下设备为d1的所有数据为：

图 2-57 查看导出的数据



```
Time,root.y4.r1.s1.d1.temperature,root.y4.r1.s1.d1.humidity,root.y4.r1.s1.d1.value,root.y4.r1.s1.d1.status
2023-12-03T06:28:18.000+08:00,98.0,46.0,4066.0,false
2023-12-06T01:54:33.000+08:00,70.0,14.0,7942.0,false
2023-12-08T11:58:08.000+08:00,88.0,64.0,7109.0,true
2023-12-11T07:44:50.000+08:00,58.0,37.0,7909.0,true
2023-12-11T15:27:01.000+08:00,42.0,27.0,837.0,false
2023-12-18T14:20:10.000+08:00,4.0,87.0,1434.0,true
2023-12-22T10:58:13.000+08:00,41.0,40.0,2333.0,false
2023-12-27T03:56:57.000+08:00,39.0,61.0,8651.0,true
2023-12-27T06:34:19.000+08:00,25.0,83.0,6621.0,true
2023-12-28T20:36:14.000+08:00,30.0,3.0,5047.0,true
```

----结束



## 2.9.3 使用 HetuEngine 查询 IoTDB 时序数据

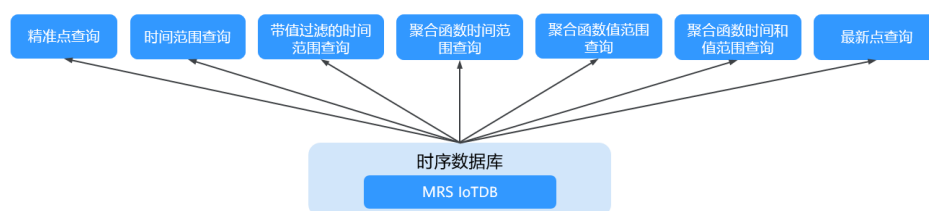
### 应用场景

IoTDB（物联网数据库）是一体化收集、存储、管理与分析物联网时序数据的软件系统。Apache IoTDB采用轻量式架构，具有高性能和丰富的功能。IoTDB数据查询及分析包括以下两种场景：

- IoTDB时序数据查询

IoTDB以时间序列为基础，提供丰富的时间序列查询语义，包含精确点查询，时间范围查询，分组查询，最新点查询等。

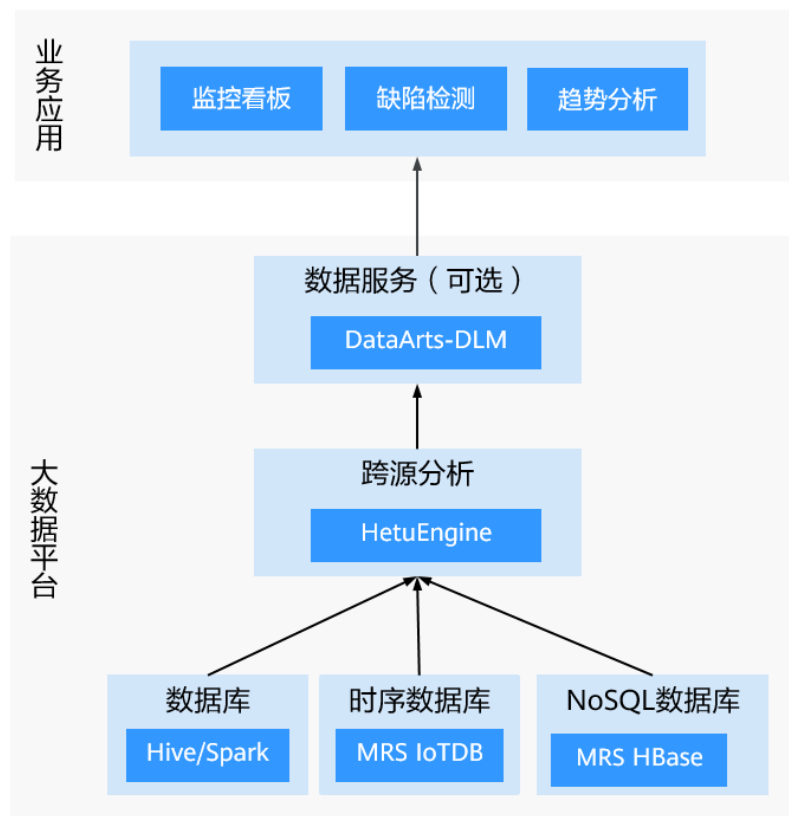
图 2-58 IoTDB 时序数据查询



- 跨源IT/OT数据统一分析

IoTDB可以借助HetuEngine提供的数据跨源分析能力，使用HetuEngine作为统一数据查询入口，也可以将IT数据和OT数据进行联合分析，提供复杂的IT和OT数据统一分析能力。

图 2-59 跨源 IT/OT 数据统一分析



使用场景如下：

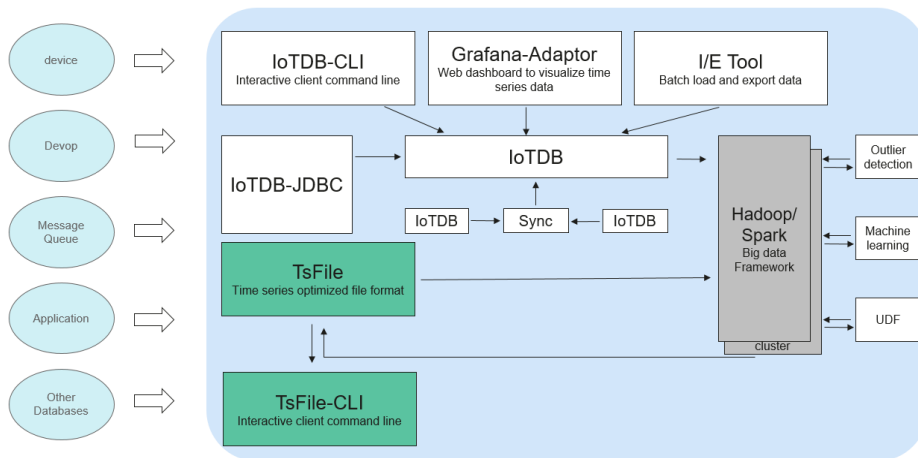
- 统一SQL入口  
为简化应用系统对接多系统的复杂性，可在HetuEngine添加IoTDB作为数据源，通过HetuEngine的SQL语法查询IoTDB中的数据。
- 与数据湖中的数据进行关联查询  
IT数据需要与OT数据进行关联分析查询。可在HetuEngine添加IoTDB、Hive作为数据源，通过HetuEngine的SQL语法同时查询Hive、IoTDB中的数据，并将两个子查询的数据进行关联。
- 与NoSQL中的数据进行关联查询  
NoSQL中的数据需要与OT数据进行关联分析查询，可在HetuEngine添加IoTDB、HBase作为数据源，通过HetuEngine的SQL语法同时查询HBase、IoTDB中的数据，并将两个子查询的数据进行关联。

## 方案架构

IoTDB套件由若干个组件构成，共同形成数据收集、数据写入、数据存储、数据查询、数据可视化、数据分析等一系列功能。

图2-60展示了使用IoTDB套件的全部组件形成的整体应用架构，IoTDB特指其中的时间序列数据库组件。

图 2-60 IoTDB 结构



- 用户可以通过JDBC/Session将来自设备传感器上采集的时序数据和服务器负载、CPU内存等系统状态数据、消息队列中的时序数据、应用程序的时序数据或者其他数据库中的时序数据导入到本地或者远程的IoTDB中。用户还可以将上述数据直接写成本地（或位于HDFS上）的TsFile文件。
- 用户可以将TsFile文件写入到HDFS上，进而满足Hadoop、Flink等数据处理任务的访问。
- 对于写入到HDFS或者本地的TsFile文件，可以利用TsFile-Hadoop或TsFile-Flink连接器，允许Hadoop或Flink进行数据处理。
- 对于分析的结果，可以写回成TsFile文件。
- IoTDB和TsFile还提供了相应的客户端工具，满足用户以SQL形式、脚本形式和图形形式写入和查看数据的各种需求。

## 操作流程

- **步骤1：创建MRS集群**：创建一个包含有HBase、HetuEngine、Hive、IoTDB组件的MRS集群。
- **步骤2：创建集群用户**：创建具有HBase、HetuEngine、Hive、IoTDB管理员权限的用户。
- **步骤3：添加HetuEngine数据源**：在HSConsole WebUI页面添加HBase和IoTDB数据源。
- **步骤4：准备数据**：创建用于查询的HBase、Hive和IoTDB表并插入数据。
- **步骤6：在HetuEngine客户端跨源查询表数据**：在HetuEngine客户端查询表数据。

## 步骤 1：创建 MRS 集群

**步骤1** 创建一个包含有HBase、HetuEngine、Hive、IoTDB组件的MRS集群，详情请参见[购买自定义集群](#)。

### 📖 说明

本文以自定义购买的MRS 3.2.0-LTS.1版本的集群为例，集群已开启Kerberos认证。

**步骤2** 集群购买成功后，在MRS集群的任一节点内，安装集群客户端，具体操作可参考[安装并使用集群客户端](#)。

例如客户端安装目录为“/opt/client”。

**步骤3** 关闭IoTDB组件的SSL配置。

1. 登录FusionInsight Manager，选择“集群 > 服务 > IoTDB > 配置”，搜索“SSL\_ENABLE”，如果该参数值为“true”，需修改为“false”并单击“保存”，关闭SSL配置。
2. 修改IoTDB客户端对应的参数关闭SSL配置：  

```
cd 客户端安装目录/IoTDB/iotdb/conf
vi iotdb-client.env
```

修改“iotdb\_ssl\_enable”参数值为“false”，保存并退出。
3. 在FusionInsight Manager界面，选择“集群 > 服务 > IoTDB”，选择页面右上角的“更多 > 重启服务”，输入当前用户密码并单击“确定”使配置生效。

----结束

## 步骤 2：创建集群用户

对于开启Kerberos认证的MRS集群，需提前准备具有相关组件操作权限的用户用于安全认证和创建HetuEngine数据源，本示例以创建对应组件的管理员用户为例进行演示，请根据实际需求创建相关权限的用户。

**步骤1** 登录FusionInsight Manager，选择“系统 > 权限 > 角色 > 添加角色”，配置以下参数并单击“确定”创建HBase管理员角色：

- 角色名称：输入角色名称，例如“hbaserole”。
- 配置资源权限：在配置资源权限列表中选择“待操作的集群名称 > HBase”，勾选“HBase管理员权限”。

**步骤2** 选择“用户 > 添加用户”，配置以下参数并单击“确定”创建用户：

- 用户名：填写“用户名”，例如“test”。
- 用户类型：选择“人机”。
- 填写“密码”和“确认新密码”。
- 用户组：单击“添加”，为该用户添加“hive”、“hetuadmin”、“iotdbgroup”、“hadoop”、“hetuuser”、“yarnviewgroup”用户组。
- 主组：在下拉列表中选择“hive”作为主组。
- 角色：单击“添加”，为该用户绑定“default”、“System\_administrator”以及新创建的HBase管理员角色。

**步骤3** 使用新创建的test用户重新登录FusionInsight Manager，修改该用户的初始密码。

----结束

### 步骤 3：添加 HetuEngine 数据源

本章节以使用HetuEngine数据源实现关联查询同一集群的HBase、Hive、IoTDB数据为例进行演示，Hive数据源已默认对接。

**添加IoTDB数据源。**

**步骤1** 使用新创建的具有HetuEngine管理员权限的用户登录FusionInsight Manager，选择“集群 > 服务 > HetuEngine”，进入HetuEngine服务页面。

**步骤2** 在概览页签下的“基本信息”区域单击“HSConsole WebUI”后的链接，进入HSConsole界面。

**步骤3** 选择“数据源”，单击“添加数据源”，在“添加数据源”页面填写参数。

1. 配置“基本配置”，填写数据源名称，例如“iotdb”，选择数据源类型“JDBC > IoTDB”，“驱动”默认为“iotdb”。
2. 配置“IoTDB配置”的以下参数：
  - JDBC URL：连接IoTDB的JDBC URL地址，格式为“jdbc:iotdb://IoTDBServer业务IP1,IoTDBServer业务IP2.端口号”。
  - IoTDBServer业务IP地址可在Manager界面，选择“集群 > 服务 > IoTDB > 实例”，查看IoTDBServer的业务IP。端口号可单击“配置”，搜索并查看“IOTDB\_SERVER\_RPC\_PORT”的值，默认为“22260”。
  - 用户名：连接IoTDB数据源的IoTDB用户名。当IoTDB所在集群为非安全模式时，需使用IoTDB默认用户“root”。
  - 密码：连接IoTDB数据源的IoTDB用户密码。当IoTDB所在集群为非安全模式时，需使用IoTDB默认用户“root”的密码。
  - 开启ssl：选择“否”。
3. （可选）根据需求可添加自定义配置。
4. 单击“确定”创建IoTDB数据。

**添加HBase数据源。**

**步骤4** 获取HBase数据源的“hbase-site.xml”、“hdfs-site.xml”和“core-site.xml”配置文件。

1. 登录FusionInsight Manager页面，在“主页”右上方单击“下载客户端”，根据界面提示下载“完整客户端”文件。

2. 将下载的客户端文件压缩包解压，获取“FusionInsight\_Cluster\_1\_Services\_ClientConfig/HBase/config”路径下的“hbase-site.xml”、“core-site.xml”和“hdfs-site.xml”文件。

**步骤5** 获取HBase数据源的代理用户的“user.keytab”和“krb5.conf”文件。

1. 在FusionInsight Manager页面，选择“系统 > 权限 > 用户”。
2. 选择对应的HBase数据源用户，在“操作”列中选择“更多 > 下载认证凭据”。
3. 从下载的文件中解压获取“user.keytab”和“krb5.conf”文件。

#### 说明

数据源的代理用户需要具有对HBase的相关操作权限。

**步骤6** 使用HetuEngine管理员用户登录FusionInsight Manager，选择“集群 > 服务 > HetuEngine”，进入HetuEngine服务页面。

**步骤7** 在概览页签下的“基本信息”区域，单击“HSConsole WebUI”后的链接，进入HSConsole界面。

**步骤8** 选择“数据源”，单击“添加数据源”。在“添加数据源”页面填写参数。

1. 配置“基本配置”，填写数据源名称，例如“hbase”，选择数据源类型“HBase”，驱动默认为“hbase-connector”。
2. 配置“HBase配置”的以下参数：
  - ZooKeeper Quorum地址：该数据源ZooKeeper服务所有quorumpeer实例业务IP。可登录FusionInsight Manager，选择“集群 > 服务 > ZooKeeper > 实例”，查看所有quorumpeer实例所在主机业务IP地址。
  - ZooKeeper客户端端口号：表示ZooKeeper客户端端口号。可登录FusionInsight Manager，选择“集群 > 服务 > ZooKeeper”，在“配置”页签查看“clientPort”的值。
  - HBase RPC通信保护：根据**步骤4**获取的“hbase-site.xml”里配置项“hbase.rpc.protection”的值进行选择：
    - 为“authentication”时选择“否”。
    - 为“privacy”时选择“是”。
  - Principal：值为**步骤5**中获取的“user.keytab”文件所属的用户。
  - keytab文件：安全认证的密钥，在本地选择**步骤5**获取的“user.keytab”文件。
  - krb5文件：Kerberos认证用到的配置文件，在本地选择**步骤5**获取的“krb5.conf”文件。
  - hbase-site文件：连接HDFS时，需要的配置文件。在本地选择**步骤4**获取的“hbase-site.xml”文件。
  - core-site文件：连接HDFS时需要用到的配置，在本地选择**步骤4**获取的“core-site.xml”文件。
  - hdfs-site文件：连接HDFS时需要用到的配置，在本地选择**步骤4**获取的“hdfs-site.xml”文件。

图 2-61 添加 HBase 数据源

添加数据源

地址: [?]

HBase配置

驱动: hbase-connector

\* ZooKeeper Quorum地址: [?]

\* ZooKeeper客户端端口号: 2181 [?]

\* HBase RPC通信保护:  否 [?]

安全认证机制: KERBEROS

\* Principal: test01@[?].COM [?]

\* keytab文件: user.keytab (196 B) [X] [?]

\* krb5文件: krb5.conf (1.42 KB) [X] [?]

\* hbase-site文件: hbase-site.xml (19.10 KB) [X] [?]

\* core-site文件: core-site.xml (14.86 KB) [X] [?]

\* hdfs-site文件: hdfs-site.xml (4.18 KB) [X] [?]

自定义配置

[增加]

3. 单击“确定”创建HBase数据源。

**验证数据源连接是否成功。**

**步骤9** 登录集群客户端所在节点，执行以下命令，切换到客户端安装目录并认证用户。

```
cd /opt/client
```

```
source bigdata_env
```

```
kinit HetuEngine组件操作用户
```

**步骤10** 执行以下命令，登录数据源的catalog。

```
hetu-cli --catalog 数据源名称 --schema 数据库名
```

例如执行以下命令连接HBase数据源：

```
hetu-cli --catalog hbase --schema default
```

**步骤11** 执行以下命令，可正常查看数据库表信息或不报错即表示连接成功。

```
show tables;
```

```
----结束
```

## 步骤 4：准备数据

**准备IoTDB数据。**

**步骤1** 以客户端安装用户登录安装客户端的节点，执行以下命令配置环境变量并认证用户：

**cd** 客户端安装目录

**source** bigdata\_env

**kinit** 组件业务用户

**步骤2** 执行以下命令登录IoTDB客户端：

**cd** IoTDB/iotdb/sbin

**./start-cli.sh -h** IoTDBserver实例业务IP地址 **-p** IoTDBserver RPC端口

运行该命令后，根据实际需求指定业务用户名：

- 指定业务用户名，则输入“yes”，并根据提示输入业务用户名和对应的业务用户密码：

```
[root@host-1 ~]# cd iotdb/sbin && ./start-cli.sh -h 192.168.34.21 -p 22260
do you want to specify your own user(yes/no):yes
Please Enter username:
Please Enter password:*****
15:39:28.483 [main] INFO org.apache.iotdb.jdbc.IoTDBConnection - Attempt to connect 192.168.34.21:22260
15:39:28.488 [main] WARN com.aliyun.iotdb.rpc.ssl.SSLContextFactory - Could not load property 'iotdb.security.ssl.config' from system.
15:39:28.488 [main] INFO com.aliyun.iotdb.rpc.ssl.SSLContextFactory - iotdb_ssl_enable is false in config, disabling SSL
-----
Starting IoTDB Cli
-----
IoTDB
-----
version
-----
IoTDB@192.168.34.21:22260> login successfully
IoTDB@192.168.34.21:22260>
```

- 不指定业务用户名，则输入“no”；此时，则使用进行安全认证的用户执行后续操作：

```
[root@host-1 ~]# cd iotdb/sbin && ./start-cli.sh -h 192.168.34.21 -p 22260
do you want to specify your own user(yes/no):no
15:31:06.569 [main] INFO org.apache.iotdb.jdbc.IoTDBConnection - Attempt to connect 192.168.34.21:22260
15:31:06.574 [main] WARN com.aliyun.iotdb.rpc.ssl.SSLContextFactory - Could not load property 'iotdb.security.ssl.config' from system.
15:31:06.575 [main] INFO com.aliyun.iotdb.rpc.ssl.SSLContextFactory - iotdb_ssl_enable is false in config, disabling SSL
-----
Starting IoTDB Cli
-----
IoTDB
-----
version
-----
IoTDB@192.168.34.21:22260> login successfully
```

- 输入其它，则退出登录：

```
[root@host-1 ~]# cd iotdb/sbin && ./start-cli.sh -h 192.168.34.21 -p 22260
do you want to specify your own user(yes/no):asda
Exit.
```

## 📖 说明

- IoTDBServer实例节点IP地址可在Manager界面，选择“集群 > 服务 > IoTDB > 实例”查看。
- IoTDBServer RPC端口可在参数“IOTDB\_SERVER\_RPC\_PORT”中自行配置。默认端口如下：
  - 开源端口默认值为：6667
  - 定制端口默认值为：22260

端口定制/开源区分：创建LTS版本类型集群时，可以选择“组件端口”为“开源”或是“定制”，选择“开源”使用开源端口，选择“定制”使用定制端口。

**步骤3** 创建IoTDB数据库并插入数据，例如：

1. 创建数据库：

**create database root.c1.r1.s1.d1;**

- create database root.c1.r1.s1.d2;**
2. 创建时间序列:

**create timeseries root.c1.r1.s1.d1.value WITH DATATYPE=FLOAT, ENCODING=PLAIN;**

**create timeseries root.c1.r1.s1.d1.status WITH DATATYPE=BOOLEAN, ENCODING=PLAIN;**

**create timeseries root.c1.r1.s1.d1.temperature WITH DATATYPE=FLOAT, ENCODING=PLAIN;**

**create timeseries root.c1.r1.s1.d1.humidity WITH DATATYPE=FLOAT, ENCODING=PLAIN;**
3. 创建带有标签的时间序列（后续可以通过标签与其他数据源进行关联查询）:

**create timeseries root.c1.r1.s1.d1.t3 WITH DATATYPE=TEXT, ENCODING=PLAIN tags(tag1=d00000001);**
4. 插入数据至时间序列中:

**insert into root.c1.r1.s1.d1(value,status,temperature,humidity) values(20,false,15,20);**

**insert into root.c1.r1.s1.d1(value,status,temperature,humidity) values(20,true,17,21);**

**insert into root.c1.r1.s1.d1(value,status,temperature,humidity) values(20,false,19,22);**

**insert into root.c1.r1.s1.d1(value,status,temperature,humidity) values(20,true,20,23);**

**insert into root.c1.r1.s1.d1(value,status,temperature,humidity) values(25,false,20,24);**

**insert into root.c1.r1.s1.d1(value,status,temperature,humidity) values(25,true,19,25);**

**insert into root.c1.r1.s1.d1(value,status,temperature,humidity) values(25,false,18,26);**

**insert into root.c1.r1.s1.d1(value,status,temperature,humidity) values(25,true,16,27);**

**insert into root.c1.r1.s1.d1(value,status,temperature,humidity) values(26,true,16,27);**
5. 执行以下命令查看插入的数据:

**select \* from root.c1.r1.s1.\*;**

图 2-62 查看表数据

Time	root.c1.r1.s1.d1.temperature	root.c1.r1.s1.d1.humidity	root.c1.r1.s1.d1.t3	root.c1.r1.s1.d1.value	root.c1.r1.s1.d1.status
2023-11-28T14:35:47.181+08:00	15.0	20.0		20.0	false
2023-11-28T14:37:38.611+08:00	17.0	21.0		20.0	true
2023-11-28T14:37:55.107+08:00	19.0	22.0		20.0	false
2023-11-28T14:38:02.692+08:00	20.0	23.0		20.0	true
2023-11-28T14:38:28.379+08:00	20.0	24.0		25.0	false
2023-11-28T14:38:35.813+08:00	19.0	25.0		25.0	true
2023-11-28T14:38:45.442+08:00	18.0	26.0		25.0	false
2023-11-28T14:38:45.442+08:00	16.0	27.0		26.0	true

6. 数据插入成功后，可直接在IoTDB客户端执行查询操作，也可参考[步骤6：在HetuEngine客户端跨源查询表数据](#)在HetuEngine客户端查询IoTDB表数据。

  - 根据时间精确查询单个设备的温度：



**select temperature from root.c1.r1.s1.d1 where time=2023-11-28T14:38:53.318+08:00;**

图 2-63 查询单个设备的温度

```
IoTDB@192.168.0.141:6667> select temperature from root.c1.r1.s1.d1 where time=2023-11-28T14:38:53.318+08:00;
+-----+-----+
| Time | root.c1.r1.s1.d1.temperature |
+-----+-----+
| 2023-11-28T14:38:53.318+08:00 | 16.0 |
+-----+-----+
```

- 根据时间精确查询单个设备所有测量值：  
**select \* from root.c1.r1.s1.d1 where time=2023-11-28T14:38:53.318+08:00;**

图 2-64 查询单个设备的所有测量值

```
IoTDB@192.168.0.141:6667> select * from root.c1.r1.s1.d1 where time=2023-11-28T14:38:53.318+08:00;
+-----+-----+-----+-----+-----+-----+-----+-----+
| Time | root.c1.r1.s1.d1.temperature | root.c1.r1.s1.d1.humidity | root.c1.r1.s1.d1.value | root.c1.r1.s1.d1.t3 | root.c1.r1.s1.d1.status |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 2023-11-28T14:38:53.318+08:00 | 16.0 | 27.0 | null | 25.0 | null | null | true |
+-----+-----+-----+-----+-----+-----+-----+-----+
```

- 根据时间精确查询所有设备的所有测量值：  
**select \* from root.c1.r1.s1.\* where time=2023-11-28T14:38:53.318+08:00;**

图 2-65 查询所有设备的所有测量值

```
IoTDB@192.168.0.141:6667> select * from root.c1.r1.s1.* where time=2023-11-28T14:38:53.318+08:00;
+-----+-----+-----+-----+-----+-----+-----+-----+
| Time | root.c1.r1.s1.d1.temperature | root.c1.r1.s1.d1.humidity | root.c1.r1.s1.d1.value | root.c1.r1.s1.d1.t3 | root.c1.r1.s1.d1.status |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 2023-11-28T14:38:53.318+08:00 | 16.0 | 27.0 | null | 25.0 | null | null | true |
+-----+-----+-----+-----+-----+-----+-----+-----+
```

- 根据设备健康状态查询该设备的所有测量值：  
**select \* from root.c1.r1.s1.d1 where status=false;**

图 2-66 根据设备健康状态查询所有测量值

```
IoTDB@192.168.0.141:6667> select * from root.c1.r1.s1.d2 where status=false;
+-----+-----+-----+-----+-----+
| Time | root.c1.r1.s1.d2.temperature | root.c1.r1.s1.d2.humidity | root.c1.r1.s1.d2.value | root.c1.r1.s1.d2.status |
+-----+-----+-----+-----+-----+
| 2023-11-28T14:42:02.545+08:00 | 15.0 | 30.0 | 30.0 | false |
| 2023-11-28T14:42:02.838+08:00 | 19.0 | 22.0 | 30.0 | false |
| 2023-11-28T14:42:02.844+08:00 | 30.0 | 24.0 | 25.0 | false |
| 2023-11-28T14:42:02.848+08:00 | 18.0 | 26.0 | 25.0 | false |
+-----+-----+-----+-----+-----+
Total line number = 4
```

- 根据时间范围查询多个设备的所有测量值：  
**select \* from root.c1.r1.s1.\* where time>2023-11-28T14:35:47.181+08:00 and time<2023-11-28T14:38:53.318+08:00;**

图 2-67 根据时间范围查询所有测量值

```
IoTDB@192.168.0.141:6667> select * from root.c1.r1.s1.* where time>2023-11-28T14:35:47.181+08:00 and time<2023-11-28T14:38:53.318+08:00;
+-----+-----+-----+-----+-----+-----+-----+-----+
| Time | root.c1.r1.s1.d1.temperature | root.c1.r1.s1.d1.humidity | root.c1.r1.s1.d1.value | root.c1.r1.s1.d1.t3 | root.c1.r1.s1.d1.status |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 2023-11-28T14:37:36.613+08:00 | null | null | null | null | null | |
| 2023-11-28T14:37:55.107+08:00 | 17.0 | 21.0 | 20.0 | true | null | null |
| 2023-11-28T14:37:55.107+08:00 | 18.0 | 22.0 | 20.0 | false | null | null |
| 2023-11-28T14:38:02.692+08:00 | 20.0 | 23.0 | 20.0 | true | null | null |
| 2023-11-28T14:38:29.379+08:00 | 20.0 | 24.0 | 25.0 | false | null | null |
| 2023-11-28T14:38:35.813+08:00 | 19.0 | 25.0 | 25.0 | true | null | null |
| 2023-11-28T14:38:45.442+08:00 | 18.0 | 26.0 | 25.0 | false | null | null |
+-----+-----+-----+-----+-----+-----+-----+-----+
```

- 根据时间范围查询某设备的最大湿度及测量值上传次数（聚合查询）：  
**select count(humidity),max\_value(humidity) from root.c1.r1.s1.d1 where time>2023-11-28T14:35:47.181+08:00 and time<2023-11-28T14:38:53.318+08:00;**

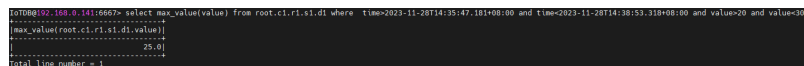
图 2-68 根据时间范围进行聚合查询

```
IoTDB@192.168.0.141:6667> select count(humidity), max_value(humidity) from root.c1.r1.s1.d1 where time>2023-11-28T14:35:47.181+08:00 and time<2023-11-28T14:38:53.318+08:00;
+-----+-----+
| count(root.c1.r1.s1.d1.humidity) | max_value(root.c1.r1.s1.d1.humidity) |
+-----+-----+
| 6 | 26.0 |
+-----+-----+
```

- 查询某设备在指定时间内且电量使用值在20%~30%之间，设备电量使用的最大值：

```
select max_value(value) from root.c1.r1.s1.d1 where
time>2023-11-28T14:35:47.181+08:00 and
time<2023-11-28T14:38:53.318+08:00 and value>20 and value<30;
```

图 2-69 根据时间和数值范围进行聚合查询



7. 执行以下命令，退出IoTDB客户端。

```
quit;
```

### 准备Hive数据。

- 步骤4 执行以下命令登录Hive客户端：

```
beeline
```

- 步骤5 执行以下命令创建Hive表并插入数据：

1. 创建Hive表，Hive表结构如表2-18所示：

```
create table device_hive(D_ID int,D_NAME string,D_DATA
data,D_PRODUCT string,D_LIFECYCLE int,D_STATUS string,D_REGION
string,D_TYPE string);
```

表 2-18 Hive 表结构

字段名称	字段类型	字段含义
D_ID	int	设备ID。
D_NAME	string	设备名称。
D_DATA	data	生产日期。
D_PRODUCT	string	生产厂家。
D_LIFECYCLE	int	设备保质期。
D_STATUS	string	设备当前状态。
D_REGION	string	设备归属区域。
D_TYPE	string	设备类型。

2. 向Hive表中插入数据：

```
insert into table
device_hive(D_ID,D_NAME,D_DATA,D_PRODUCT,D_LIFECYCLE,D_STATUS,D_
REGION,D_TYPE) values("ID1","device1","2024-10-25", "a", "在保质期", "
工作中","region1","传感器");
```

3. 执行以下命令，退出Hive客户端。

```
!q
```

----结束

**准备HBase数据。**

**步骤1** 执行以下命令登录HBase客户端：

```
hbase shell
```

**步骤2** 执行以下命令创建HBase表并插入数据：

1. 创建HBase表，HBase表结构如表2-19所示：

```
create 'device_hbase','f1'
```

表 2-19 HBase 表结构（开户信息）

字段名称	字段类型	字段含义
A_ID	int	账户ID。
D_ID	int	设备ID。
D_NAME	int	开户名。
D_ADDRESS	string	开户地址。
D_CITY	string	开户城市。
D_NATION	string	开户区域。
D_Date	data	开户时间。
D_PHONE	string	联系电话。

2. 向HBase表中插入数据：

```
put 'device_hbase', '001', 'f1:A_ID', '账户1'  
put 'device_hbase', '001', 'f1:D_ID', 'd00000001'  
put 'device_hbase', '001', 'f1:A_NAME', '开户名1'  
put 'device_hbase', '001', 'f1:A_ADDRESS', '开户地址1'  
put 'device_hbase', '001', 'f1:A_CITY', '开户城市1'  
put 'device_hbase', '001', 'f1:A_NATION', '开户区域1'  
put 'device_hbase', '001', 'f1:A_Date', '开户时间1'  
put 'device_hbase', '001', 'f1:A_PHONE', '联系电话1'  
put 'device_hbase', '002', 'A_ID', '账户2'  
put 'device_hbase', '002', 'D_ID', 'd00000002'  
put 'device_hbase', '002', 'A_NAME', '开户名2'  
put 'device_hbase', '002', 'A_ADDRESS', '开户地址2'  
put 'device_hbase', '002', 'A_CITY', '开户城市2'  
put 'device_hbase', '002', 'A_NATION', '开户区域2'  
put 'device_hbase', '002', 'A_Date', '开户时间2'  
put 'device_hbase', '002', 'A_PHONE', '联系电话2'
```

3. 执行以下命令，退出HBase客户端。

```
quit;
```

**步骤3** 执行以下命令在HetuEngine客户端为已创建的HBase表创建映射表：

1. 执行以下命令，登录数据源的catalog：

```
hetu-cli --catalog 数据源名称 --schema 数据库名
```

例如执行以下命令：

```
hetu-cli --catalog hbase --schema default
```

2. 创建结构化映射表，字段类型要大于HBase的字段类型，例如：

```
CREATE TABLE default.device_hbase (  
  rowId VARCHAR,  
  aid VARCHAR,  
  did VARCHAR,  
  aname VARCHAR,  
  aaddress VARCHAR,  
  acity VARCHAR,  
  anation VARCHAR,  
  adate VARCHAR,  
  aphone VARCHAR  
)  
WITH (  
  column_mapping =  
  'aid:f1:A_ID,did:f1:D_ID,aname:f2:A_NAME,aaddress:f2:A_ADDRESS,acity:f2:  
  A_CITY,anation:f3:A_NATION,adate:f3:A_Date,aphone:f3:A_PHONE',  
  row_id = 'rowId',  
  hbase_table_name = 'default.device_hbase',  
  external = true  
);
```

---结束

## 步骤 6：在 HetuEngine 客户端跨源查询表数据

**步骤1** 为执行数据查询的用户赋予Ranger相关权限。

1. 使用Ranger管理员用户rangeradmin登录Ranger管理页面。
2. 在首页中单击“TRINO”区域的“HetuEngine”。
3. 在“Access”页签单击“all - catalog, sessionproperty”所在行的编辑按钮进入编辑页面，在“Allow Conditions”区域新增策略允许条件，“Select User”列选择已创建的用户，例如“test”，“Permissions”列选择“Alter”权限，单击“Save”保存。
4. 单击“all - systemproperty”所在行的编辑按钮进入编辑页面，在“Allow Conditions”区域新增策略允许条件，“Select User”列选择已创建的用户，例如“test”，“Permissions”列选择“Alter”权限，单击“Save”保存。

**步骤2** 登录集群客户端所在节点，执行以下命令，切换到客户端安装目录并认证用户。

```
cd /opt/client  
source bigdata_env
```

### kinit HetuEngine组件操作用户

**步骤3** 执行以下命令，登录IoTDB数据源的catalog。

**hetu-cli --catalog 数据源名称**

例如执行以下命令：

**hetu-cli --catalog iotdb**

**步骤4** 执行以下命令，修改session配置：

**set session implicit\_conversion=true;**

**步骤5** 数据查询。

- 跨源关联查询IoTDB、HBase、Hive数据。

- 关联查询IoTDB和Hive表数据：

```
select * from iotdb."root.c1.r1.s1.*".value t1 join
hive.default.device_hive d2 on t1.tag1=d2.d_id where t1.value>25;
```

例如，关联查询的数据如下：

图 2-70 关联查询 IoTDB 和 Hive 表数据

```
hetucli> select * from iotdb."root.c1.r1.s1.*".value t1 join
hive.default.device_hive d2 on t1.tag1=d2.d_id where t1.value>25;
```

name	time	value	tag1	d_id	d_name	d_date	d_product	d_lifecycle	d_status	d_region	d_type
root.c1.r1.s1.d2.value	2023-11-28 06:42:02.545	30.0	d0000002	d0000002		2002-10-30			5		
root.c1.r1.s1.d2.value	2023-11-28 06:42:02.834	30.0	d0000002	d0000002		2002-10-30			5		
root.c1.r1.s1.d2.value	2023-11-28 06:42:02.838	30.0	d0000002	d0000002		2002-10-30			5		
root.c1.r1.s1.d2.value	2023-11-28 06:42:02.841	30.0	d0000002	d0000002		2002-10-30			5		
root.c1.r1.s1.d1.value	2023-11-29 02:24:12.200	26.0	d0000001	d0000001		2022-10-30			5		
root.c1.r1.s1.d1.value	2023-11-29 11:58:49.602	26.0	d0000001	d0000001		2022-10-30			5		
root.c1.r1.s1.d1.value	2023-11-29 11:59:02.304	26.0	d0000001	d0000001		2022-10-30			5		
root.c1.r1.s1.d1.value	2023-11-29 11:59:03.677	26.0	d0000001	d0000001		2022-10-30			5		
root.c1.r1.s1.d1.value	2023-11-29 11:59:04.501	26.0	d0000001	d0000001		2022-10-30			5		
root.c1.r1.s1.d1.value	2023-11-29 12:01:38.966	42.0	d0000001	d0000001		2022-10-30			5		
root.c1.r1.s1.d1.value	2023-11-29 12:01:41.681	42.0	d0000001	d0000001		2022-10-30			5		
root.c1.r1.s1.d1.value	2023-11-29 12:02:54.311	42.0	d0000001	d0000001		2022-10-30			5		
root.c1.r1.s1.d1.value	2023-11-29 12:02:55.729	42.0	d0000001	d0000001		2022-10-30			5		
root.c1.r1.s1.d1.value	2023-11-29 12:02:57.199	42.0	d0000001	d0000001		2022-10-30			5		

- 关联查询IoTDB和HBase表数据：

```
select * from iotdb."root.c1.r1.s1.*".value t1 join
hbase.default.device_hbase d2 on t1.tag1=d2.did where t1.value>25;
```

例如，关联查询的数据如下：

图 2-71 关联查询 IoTDB 和 HBase 表数据

```
hetucli> select * from iotdb."root.c1.r1.s1.*".value t1 join
hbase.default.device_hbase d2 on t1.tag1=d2.did where t1.value>25;
```

name	time	value	tag1	rowid	atd	did	aname	address	actty	anaton	adate	aphone
root.c1.r1.s1.d2.value	2023-11-28 06:42:02.545	30.0	d0000002	002		d0000002						
root.c1.r1.s1.d2.value	2023-11-28 06:42:02.834	30.0	d0000002	002		d0000002						
root.c1.r1.s1.d2.value	2023-11-28 06:42:02.838	30.0	d0000002	002		d0000002						
root.c1.r1.s1.d2.value	2023-11-28 06:42:02.841	30.0	d0000002	002		d0000002						
root.c1.r1.s1.d1.value	2023-11-29 02:24:12.200	26.0	d0000001	001		d0000001						
root.c1.r1.s1.d1.value	2023-11-29 11:58:49.602	26.0	d0000001	001		d0000001						
root.c1.r1.s1.d1.value	2023-11-29 11:59:02.304	26.0	d0000001	001		d0000001						
root.c1.r1.s1.d1.value	2023-11-29 11:59:03.677	26.0	d0000001	001		d0000001						
root.c1.r1.s1.d1.value	2023-11-29 11:59:04.501	26.0	d0000001	001		d0000001						
root.c1.r1.s1.d1.value	2023-11-29 12:01:38.966	42.0	d0000001	001		d0000001						
root.c1.r1.s1.d1.value	2023-11-29 12:01:41.681	42.0	d0000001	001		d0000001						
root.c1.r1.s1.d1.value	2023-11-29 12:02:54.311	42.0	d0000001	001		d0000001						
root.c1.r1.s1.d1.value	2023-11-29 12:02:55.729	42.0	d0000001	001		d0000001						
root.c1.r1.s1.d1.value	2023-11-29 12:02:57.199	42.0	d0000001	001		d0000001						
root.c1.r1.s1.d1.value	2023-11-29 12:02:58.120	42.0	d0000001	001		d0000001						
root.c1.r1.s1.d1.value	2023-11-29 12:02:59.329	42.0	d0000001	001		d0000001						
root.c1.r1.s1.d1.value	2023-11-29 12:03:00.216	42.0	d0000001	001		d0000001						
root.c1.r1.s1.d1.value	2023-11-29 12:04:36.279	42.0	d0000001	001		d0000001						
root.c1.r1.s1.d1.value	2023-11-29 12:04:37.253	42.0	d0000001	001		d0000001						
root.c1.r1.s1.d1.value	2023-11-29 12:04:38.404	42.0	d0000001	001		d0000001						

- 关联查询IoTDB和Hive+HBase表数据：

```
select * from iotdb."root.c1.r1.s1.*".status t1 join
hive.default.device_hive d2 on t1.tag1=d2.d_id join
hbase.default.device_hbase d3 on t1.tag1=d3.did where t1.value=false;
```

例如，关联查询的数据如下：

图 2-72 关联查询 IoTDB 和 Hive+HBase 表数据

```

hetuengine> select * from iotdb.`root.c1.r1.s1`.`status` t1 join hive.`default`.`device` hive_d2 on t1.tag1=d2.did join hive.`default`.`device` hive_d3 on t1.tag1=d3.d_id where t1.value=false;
-----
d_name | d_date | d_product | d_lifecycle | d_status | d_region | d_type | d_value | total | row_id | did | dname | address | acty | anation | adre | aphone | d_id
-----
root.c1.r1.s1.d1.status | 2023-11-28 06:35:47.181 | false | d00000001 | 001 | 1 | d00000001 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | d00000001 | 2022-10-30
root.c1.r1.s1.d1.status | 2023-11-28 06:37:55.107 | false | d00000001 | 001 | 1 | d00000001 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | d00000001 | 2022-10-30
root.c1.r1.s1.d1.status | 2023-11-28 06:38:28.379 | false | d00000001 | 001 | 1 | d00000001 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | d00000001 | 2022-10-30
root.c1.r1.s1.d1.status | 2023-11-28 06:38:45.442 | false | d00000001 | 001 | 1 | d00000001 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | d00000001 | 2022-10-30
root.c1.r1.s1.d2.status | 2023-11-28 06:42:02.545 | false | d00000002 | 002 | 2 | d00000002 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | d00000002 | 2002-10-30
root.c1.r1.s1.d2.status | 2023-11-28 06:42:02.838 | false | d00000002 | 002 | 2 | d00000002 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | d00000002 | 2002-10-30
root.c1.r1.s1.d2.status | 2023-11-28 06:42:02.844 | false | d00000002 | 002 | 2 | d00000002 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | d00000002 | 2002-10-30
root.c1.r1.s1.d2.status | 2023-11-28 06:42:02.848 | false | d00000002 | 002 | 2 | d00000002 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | d00000002 | 2002-10-30
(8 rows)

Query 20231206_070914_00100_1.yjvcv@default@HetuEngine, FINISHED, 2 nodes
Splits: 164 total, 164 done (100.00%)
0.56 [0.00 row, 5.19] [465 row/s, 9706/s]
    
```

**说明**

本示例中的关联查询的关键key是“D\_ID”，在IoTDB中是通过tag实现的。例如，d1设备的所有测量值设置的标签为tag1=d00000001，d2设备的所有测量值设置的标签为tag1=d00000002，依次类推。

- 在HetuEngine客户端查询IoTDB表数据。  
执行以下命令查看IoTDB中的Schema信息：

**show databases;**

执行以下命令切换至需要查询数据的Schema中：

**use "root.c1.r1.s1.d1";**

查询示例如下：

- 根据时间精确查询单个设备的温度：

```
select value from "root.c1.r1.s1.d1".temperature where time='2023-11-28 06:38:28.379';
```

图 2-73 查询单个设备的温度

```

hetuengine:root.c1.r1.s1.d1> select value from "root.c1.r1.s1.d1".temperature where time=2023-11-28 06:38:28.379;
-----
value
-----
20.0
(1 row)

Query 20231205_110836_00037_3zb6u@default@HetuEngine, FINISHED, 1 node
Splits: 1 total, 1 done (100.00%)
    
```

- 根据时间范围查询单个设备的所有测量值：

```
select * from "root.c1.r1.s1.d1"."*" where time>'2023-11-28 06:35:47.181' and time<'2023-11-29 11:58:49.602';
```

图 2-74 查询单个设备的所有测量值

```

hetuengine:root.c1.r1.s1.d1> select * from "root.c1.r1.s1.d1"."*" where time>2023-11-28 06:35:47.181 and time<2023-11-29 11:58:49.602
-----
name | time | value | tag1
-----
root.c1.r1.s1.d1.temperature | 2023-11-28 06:37:36.611 | 17.0 | 13022
root.c1.r1.s1.d1.humidity | 2023-11-28 06:37:36.611 | 21.0 | NULL
root.c1.r1.s1.d1.value | 2023-11-28 06:37:36.611 | 20.0 | NULL
root.c1.r1.s1.d1.status | 2023-11-28 06:37:36.611 | true | NULL
root.c1.r1.s1.d1.temperature | 2023-11-28 06:37:55.107 | 19.0 | 13022
root.c1.r1.s1.d1.humidity | 2023-11-28 06:37:55.107 | 22.0 | NULL
root.c1.r1.s1.d1.value | 2023-11-28 06:37:55.107 | 20.0 | NULL
root.c1.r1.s1.d1.status | 2023-11-28 06:37:55.107 | false | NULL
root.c1.r1.s1.d1.temperature | 2023-11-28 06:38:02.692 | 20.0 | 13022
root.c1.r1.s1.d1.humidity | 2023-11-28 06:38:02.692 | 23.0 | NULL
root.c1.r1.s1.d1.value | 2023-11-28 06:38:02.692 | 20.0 | NULL
root.c1.r1.s1.d1.status | 2023-11-28 06:38:02.692 | true | NULL
root.c1.r1.s1.d1.temperature | 2023-11-28 06:38:28.379 | 20.0 | 13022
root.c1.r1.s1.d1.humidity | 2023-11-28 06:38:28.379 | 24.0 | NULL
root.c1.r1.s1.d1.value | 2023-11-28 06:38:28.379 | 25.0 | NULL
root.c1.r1.s1.d1.status | 2023-11-28 06:38:28.379 | false | NULL
root.c1.r1.s1.d1.temperature | 2023-11-28 06:38:35.813 | 19.0 | 13022
root.c1.r1.s1.d1.humidity | 2023-11-28 06:38:35.813 | 25.0 | NULL
root.c1.r1.s1.d1.value | 2023-11-28 06:38:35.813 | 25.0 | NULL
root.c1.r1.s1.d1.status | 2023-11-28 06:38:35.813 | true | NULL
root.c1.r1.s1.d1.temperature | 2023-11-28 06:38:45.442 | 18.0 | 13022
root.c1.r1.s1.d1.humidity | 2023-11-28 06:38:45.442 | 26.0 | NULL
root.c1.r1.s1.d1.value | 2023-11-28 06:38:45.442 | 25.0 | NULL
root.c1.r1.s1.d1.status | 2023-11-28 06:38:45.442 | false | NULL
root.c1.r1.s1.d1.temperature | 2023-11-28 06:38:53.318 | 16.0 | 13022
root.c1.r1.s1.d1.humidity | 2023-11-28 06:38:53.318 | 27.0 | NULL
root.c1.r1.s1.d1.value | 2023-11-28 06:38:53.318 | 25.0 | NULL
root.c1.r1.s1.d1.status | 2023-11-28 06:38:53.318 | true | NULL
root.c1.r1.s1.d1.temperature | 2023-11-28 08:24:12.200 | 16.0 | 13022
root.c1.r1.s1.d1.humidity | 2023-11-28 08:24:12.200 | 27.0 | NULL
root.c1.r1.s1.d1.value | 2023-11-28 08:24:12.200 | 26.0 | NULL
root.c1.r1.s1.d1.status | 2023-11-28 08:24:12.200 | true | NULL
(32 rows)
    
```

- 根据时间查询所有设备所有测量值：

```
select * from "root.c1.r1.s1.*"."*" where time='2023-11-28 06:38:28.379';
```

图 2-75 查询所有设备的所有测量值

```
hetuengine:root.c1.r1.s1.d1> select * from "root.c1.r1.s1.*"."*" where time='2023-11-28 06:38:28.379';
-----
name                | time                | value | tag1
-----
root.c1.r1.s1.d1.temperature | 2023-11-28 06:38:28.379 | 20.0  | 13022
root.c1.r1.s1.d1.humidity   | 2023-11-28 06:38:28.379 | 24.0  | NULL
root.c1.r1.s1.d1.value      | 2023-11-28 06:38:28.379 | 25.0  | NULL
root.c1.r1.s1.d1.status     | 2023-11-28 06:38:28.379 | false | NULL
root.c1.r1.s1.d2.temperature | 2023-11-28 06:38:28.379 | NULL  | NULL
root.c1.r1.s1.d2.humidity   | 2023-11-28 06:38:28.379 | NULL  | NULL
root.c1.r1.s1.d2.value      | 2023-11-28 06:38:28.379 | NULL  | NULL
root.c1.r1.s1.d2.status     | 2023-11-28 06:38:28.379 | NULL  | NULL
(8 rows)

Query 20231205_113618_00043_3zb6u@default@HetuEngine, FINISHED, 1 node
Splits: 1 total, 1 done (100.00%)
0.53 [1.02K rows, 0B] [1.92K rows/s, 0B/s]
```

- 查询某个设备处于不健康状态的时间点:

```
select * from "root.c1.r1.s1.d1".status where status=false;
```

图 2-76 查询设备处于不健康状态的时间点

```
hetuengine:root.c1.r1.s1.d1> select * from "root.c1.r1.s1.d1".status where value='false';
-----
name                | time                | value
-----
root.c1.r1.s1.d1.status | 2023-11-28 06:35:47.181 | false
root.c1.r1.s1.d1.status | 2023-11-28 06:37:55.107 | false
root.c1.r1.s1.d1.status | 2023-11-28 06:38:28.379 | false
root.c1.r1.s1.d1.status | 2023-11-28 06:38:45.442 | false
(4 rows)

Query 20231205_115256_00045_3zb6u@default@HetuEngine, FINISHED, 1 node
Splits: 1 total, 1 done (100.00%)
0.43 [120 rows, 0B] [281 rows/s, 0B/s]
```

- 聚合查询。例如，查询某设备在某个温度范围内的最大温度及上传次数:

```
select count(value),max_value(value) from
"root.c1.r1.s1.d1".temperature where value>15 and value<20;
```

图 2-77 聚合查询

```
hetuengine:root.c1.r1.s1.d1> select count(value),max(value) from "root.c1.r1.s1.d1".temperature where value>15 and value<20;
-----
_col0 | _col1
-----
75 | 19.0
(1 row)

Query 20231206_015856_00038_iyucv@default@HetuEngine, FINISHED, 2 nodes
```

----结束

## 2.9.4 使用 Grafana 对接 IoTDB 数据库

### 应用场景

IoTDB（物联网数据库）是一体化收集、存储、管理与分析物联网时序数据的软件系统。

IoTDB支持对接Grafana，对接成功后，即可通过Grafana丰富的图形报表展示能力，对IoTDB中的时序数据进行图形化展示，以分析时序数据的变化趋势等信息。

图 2-78 Grafana 对接 IoTDB

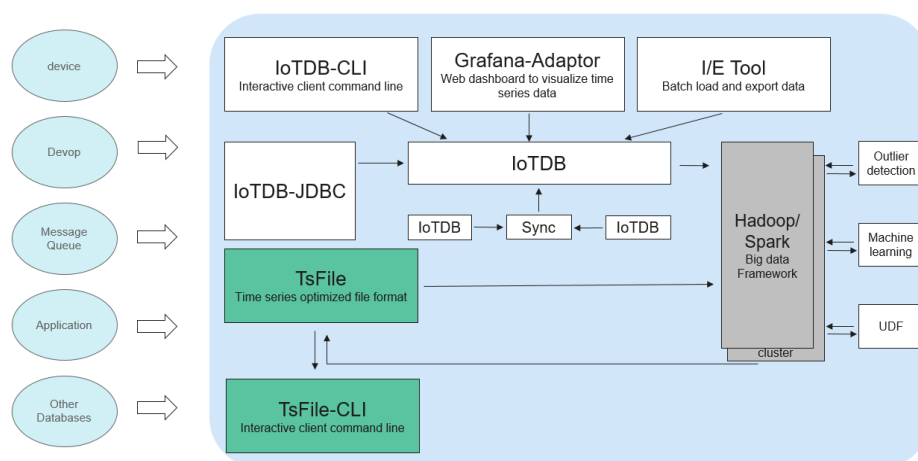


## 方案架构

IoTDB套件由若干个组件构成，共同形成数据收集、数据写入、数据存储、数据查询、数据可视化、数据分析等一系列功能。

图2-79展示了使用IoTDB套件的全部组件形成的整体应用架构，IoTDB特指其中的时间序列数据库组件。

图 2-79 IoTDB 结构



- 用户可以通过JDBC/Session将来自设备传感器上采集的时序数据和服务器负载、CPU内存等系统状态数据、消息队列中的时序数据、应用程序的时序数据或者其他数据库中的时序数据导入到本地或者远程的IoTDB中。用户还可以将上述数据直接写成本地（或位于HDFS上）的TsFile文件。
- 用户可以将TsFile文件写入到HDFS上，进而满足Hadoop、Flink等数据处理任务的访问。
- 对于写入到HDFS或者本地的TsFile文件，可以利用TsFile-Hadoop或TsFile-Flink连接器，允许Hadoop或Flink进行数据处理。
- 对于分析的结果，可以写回成TsFile文件。
- IoTDB和TsFile还提供了相应的客户端工具，满足用户以SQL形式、脚本形式和图形形式写入和查看数据的各种需求。

## 操作流程

- **步骤1：创建MRS集群：**创建一个包含有IoTDB组件的MRS集群。
- **步骤2：下载并安装Grafana：**下载并安装Grafana。



- **步骤3: 使用Grafana查询IoTDB:** 使用Grafana对接IoTDB并查询表数据。

## 约束与限制

安装Grafana的Linux节点需要与MRS集群具有相同的“可用区”、“虚拟私有云”和“安全组”。

## 步骤 1: 创建 MRS 集群

**步骤1** 创建一个包含有IoTDB组件的MRS集群，详情请参见[购买自定义集群](#)。

### 📖 说明

本文以自定义购买的MRS 3.3.0-LTS.1版本的集群为例，集群已开启Kerberos认证。

**步骤2** 集群购买成功后，在MRS集群的任一节点内，安装集群客户端，具体操作可参考[安装并使用集群客户端](#)。

例如客户端安装目录为“/opt/client”。

**步骤3** 修改IoTDB配置。

1. 登录FusionInsight Manager，选择“集群 > 服务 > IoTDB > 配置”，搜索“ssl”，配置以下参数并单击“保存”。
  - 将“IoTDB->ConfigNode”的“SSL\_ENABLE”参数值修改为“false”。
  - 将“IoTDB->IoTDBServer”的“enable\_https”和“SSL\_ENABLE”的参数值都修改为“false”。
2. 使用root用户分别登录所有的IoTDBserver节点，修改“\${BIGDATA\_HOME}/FusionInsight\_IoTDB\_8.3.0/1\_1\_IoTDBServer/etc”目录下的“iotdb-common.properties”文件中的“enable\_rest\_service”参数值为“true”。
3. 在FusionInsight Manager界面，选择“集群 > 服务 > IoTDB”，选择页面右上角的“更多 > 重启服务”，输入当前用户密码并单击“确定”使配置生效。

**步骤4** 创建访问IoTDB的用户。

在FusionInsight Manager界面，选择“系统 > 权限 > 用户 > 添加用户”，配置以下参数并单击“确定”创建IoTDB用户。

- 用户名：填写用户名称，例如“iotdbuser”。
- 用户类型：选择“人机”。
- 输入“密码”和“确认新密码”。
- 用户组：单击“添加”，勾选“iotdbgroup”用户组，单击“确定”。
- 主组：主组选择“iotdbgroup”。

**步骤5** 使用新创建的iotdbuser用户重新登录FusionInsight Manager，修改该用户的初始密码。

----结束

## 步骤 2: 下载并安装 Grafana

**步骤1** 登录MRS云服务管理控制台，在“现有集群”列表中，单击[步骤1: 创建MRS集群](#)新创建的集群名称。

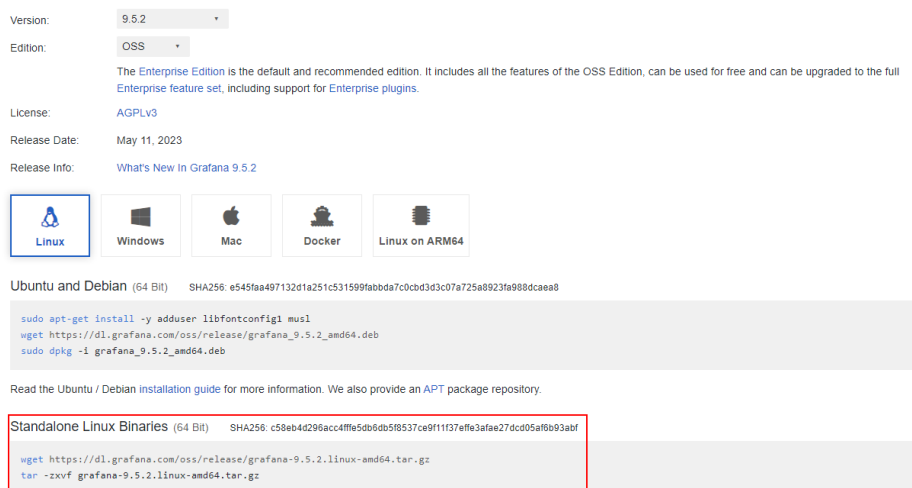
记录集群的“可用区”、“虚拟私有云”、“安全组”。

**步骤2** 在管理控制台首页服务列表中选择“弹性云服务器”，进入ECS列表，单击“购买弹性云服务器”，配置以下参数并单击“立即购买”创建一个Linux弹性云服务器：

- 计费模式：选择“按需计费”。
- 区域：与MRS集群的区域保持一致。
- 可用区：与MRS集群的可用区保持一致。
- 实例：根据实际需求选择对应的规格类型。
- 操作系统：根据实际需求选择操作系统，例如选择公共镜像“Huawei Cloud EulerOS”的“Huawei Cloud EulerOS 2.0 标准版 64位(10GiB)”。
- 网络：“虚拟私有云”与MRS集群保持一致。
- 安全组：与MRS集群的安全组保持一致。
- 公网访问：设置“弹性公网IP”为“现在购买”，并设置“线路”、“公网带宽”和“带宽大小”，勾选“随实例释放”。

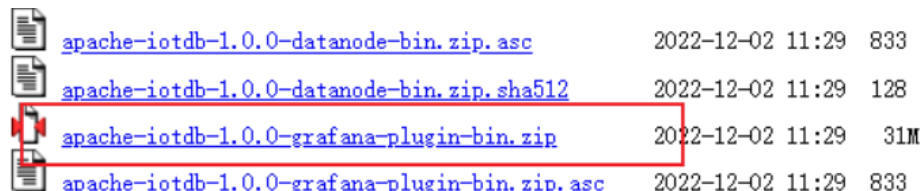
**步骤3** 在<https://grafana.com/grafana/download/9.5.2?edition=oss&platform=linux>下载Grafana安装包到本地。例如，下载“Version”为“9.5.2”，“Edition”为“OSS”的Linux的Grafana安装包：

图 2-80 下载 Grafana 安装包



**步骤4** 在<https://archive.apache.org/dist/iotdb/1.0.0/>下载IoTDB对应的插件“apache-iotdb-1.0.0-grafana-plugin-bin.zip”到本地。

图 2-81 下载 IoTDB 插件



**步骤5** 登录ECS管理控制台，在ECS列表中，单击新创建的ECS所在行的“远程登录”，输入用户名和密码登录ECS。

1. 执行以下命令创建用于放置Grafana安装包和IoTDB插件的目录：  
`cd /opt`

### mkdir grafana

2. 将下载到本地的Grafana安装包和iotdb-grafana插件包上传到“/opt/grafana”目录下。

图 2-82 上传软件包和 IoTDB 插件

```
[root@192-168-34-191 grafana]# pwd
/opt/grafana
[root@192-168-34-191 grafana]# ll
total 108492
drwxr-xr-x 4 root root 4096 Dec 2 2022 apache-iotdb-1.0.0-grafana-plugin-bin
-rw-r--r-- 1 root root 32465784 Jun 27 17:30 apache-iotdb-1.0.0-grafana-plugin-bin.zip
drwxr-xr-x 7 root root 4096 Jun 27 17:32 grafana-9.5.2
-rw-r--r-- 1 root root 78499085 Jun 27 17:25 grafana-9.5.2.linux-amd64.tar.gz
[root@192-168-34-191 grafana]#
```

步骤6 执行以下命令解压Grafana安装包和iotdb-grafana插件:

```
tar -zxvf grafana-9.5.2.linux-amd64.tar.gz
unzip apache-iotdb-1.0.0-grafana-plugin-bin.zip
```

步骤7 执行以下命令创建插件目录:

```
cd /opt/grafana/grafana-9.5.2
mkdir -p data/plugins
```

步骤8 将“/opt/grafana/apache-iotdb-1.0.0-grafana-plugin-bin/iotdb-grafana-plugin”目录及目录下的文件拷贝到“/opt/grafana/grafana-9.5.2/data/plugins”目录下。

图 2-83 拷贝插件文件

```
[root@192-168-34-191 plugins]# pwd
/opt/grafana/grafana-9.5.2/data/plugins
[root@192-168-34-191 plugins]#
[root@192-168-34-191 plugins]#
[root@192-168-34-191 plugins]# ll
total 4
drwxr-xr-x 3 root root 4096 Jun 27 17:33 iotdb-grafana-plugin
[root@192-168-34-191 plugins]#
```

步骤9 配置“/opt/grafana/grafana-9.5.2/conf”目录下的“defaults.ini”文件，修改以下内容:

```
allow_loading_unsigned_plugins = apache-iotdb-datasource
plugins = data/plugins //Grafana的插件目录，即步骤7创建的目录。
```

步骤10 切换至Grafana的安装目录，执行以下命令启动Grafana服务:

```
cd /opt/grafana/
nohup ./bin/grafana-server web > ./grafana.log 2>&1 &
```

Grafana启动和运行日志可在“grafana.log”中查看。

----结束

## 步骤 3: 使用 Grafana 查询 IoTDB

准备需要查询的IoTDB数据。

步骤1 以客户端安装用户登录安装客户端的节点，执行以下命令配置环境变量并认证用户:

```
cd 客户端安装目录
```

source bigdata\_env

kinit 组件业务用户

步骤2 执行以下命令登录IoTDB客户端：

cd IoTDB/iotdb/sbin

./start-cli.sh -h IoTDBserver实例业务IP地址 -p IoTDBserver RPC端口

运行该命令后，根据实际需求指定业务用户名：

- 指定业务用户名，则输入“yes”，并根据提示输入业务用户名和对应的业务用户密码：

```
[root@... sbin]# ./start-cli.sh -h ... -p 22260
do you want to specify your own user(yes/no):yes
Please Enter username:
Please Enter password:*****
15:39:28.483 [main] INFO org.apache.iotdb.jdbc.IoTDBConnection - Attempt to connect 192.168.34.21:22260
15:39:28.488 [main] WARN com... .iotdb.rpc.ssl.SSLContextFactory - Could not load property 'iotdb.security.ssl.config' from system.
15:39:28.488 [main] INFO com... .iotdb.rpc.ssl.SSLContextFactory - iotdb_ssl_enable is false in config, disabling SSL
-----
Starting IoTDB Cli
-----
IoTDB
version
IoTDB@ :22260> login successfully
IoTDB@ :22260>
```

- 不指定业务用户名，则输入“no”；此时，则使用进行安全认证的用户执行后续操作：

```
[root@host-... sbin]# ./start-cli.sh -h ... -p 22260
do you want to specify your own user(yes/no):no
15:31:06.569 [main] INFO org.apache.iotdb.jdbc.IoTDBConnection - Attempt to connect ...:22260
15:31:06.574 [main] WARN com... .iotdb.rpc.ssl.SSLContextFactory - Could not load property 'iotdb.security.ssl.config' from system.
15:31:06.575 [main] INFO com... .iotdb.rpc.ssl.SSLContextFactory - iotdb_ssl_enable is false in config, disabling SSL
-----
Starting IoTDB Cli
-----
IoTDB
version
IoTDB@ :22260> login successfully
```

- 输入其它，则退出登录：

```
[root@host-... sbin]# ./start-cli.sh -h ... -p 22260
do you want to specify your own user(yes/no):asda
Exit.
```

## 说明

- IoTDBServer实例节点IP地址可在Manager界面，选择“集群 > 服务 > IoTDB > 实例”查看。
- IoTDBServer RPC端口可在参数“IOTDB\_SERVER\_RPC\_PORT”中自行配置。默认端口如下：
  - 开源端口默认值为：6667
  - 定制端口默认值为：22260

端口定制/开源区分：创建LTS版本类型集群时，可以选择“组件端口”为“开源”或是“定制”，选择“开源”使用开源端口，选择“定制”使用定制端口。

步骤3 创建IoTDB数据库并插入数据，例如：

1. 创建数据库：

```
create database root.c1.r1.s1.d1;
```

2. 创建时间序列:  

```
create timeseries root.c1.r1.s1.d1.value WITH DATATYPE=FLOAT,  
ENCODING=PLAIN;  
  
create timeseries root.c1.r1.s1.d1.status WITH DATATYPE=BOOLEAN,  
ENCODING=PLAIN;  
  
create timeseries root.c1.r1.s1.d1.temperature WITH DATATYPE=FLOAT,  
ENCODING=PLAIN;  
  
create timeseries root.c1.r1.s1.d1.humidity WITH DATATYPE=FLOAT,  
ENCODING=PLAIN;
```
3. 插入数据至时间序列中:  

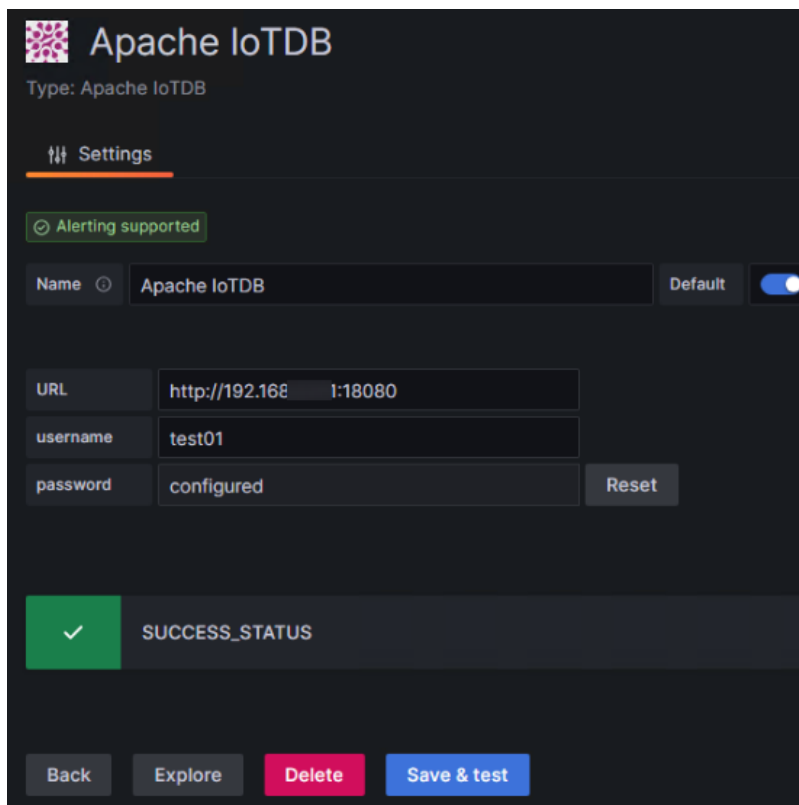
```
insert into root.c1.r1.s1.d1 (value,status,temperature,humidity)  
values(20,false,15,20);  
  
insert into root.c1.r1.s1.d1 (value,status,temperature,humidity)  
values(20,true,17,21);  
  
insert into root.c1.r1.s1.d1 (value,status,temperature,humidity)  
values(20,false,19,22);  
  
insert into root.c1.r1.s1.d1 (value,status,temperature,humidity)  
values(20,true,20,23);  
  
insert into root.c1.r1.s1.d1 (value,status,temperature,humidity)  
values(25,false,20,24);  
  
insert into root.c1.r1.s1.d1 (value,status,temperature,humidity)  
values(25,true,19,25);  
  
insert into root.c1.r1.s1.d1 (value,status,temperature,humidity)  
values(25,false,18,26);  
  
insert into root.c1.r1.s1.d1 (value,status,temperature,humidity)  
values(25,true,16,27);  
  
insert into root.c1.r1.s1.d1 (value,status,temperature,humidity)  
values(26,true,16,27);
```

#### 使用Grafana对接IoTDB。

##### 步骤4 访问Grafana，创建IoTDB数据源。

1. 在浏览器中访问Grafana，地址为“<http://Grafana所在主机的IP地址:Port>”，默认的登录用户名和密码都是**admin**。  
其中，Port为Grafana的运行端口，默认为3000。
2. 在左侧导航栏中选择“Home > Data sources > Add data source > Apache IoTDB”，在“Settings”配置以下参数并单击“Save & test”：
  - URL: 访问IoTDB的地址，格式为“<http://IoTDBServer节点IP地址:18080>”。IoTDBServer的IP地址可登录FusionInsight Manager，选择“集群 > 服务 > IoTDB > 实例”查看。
  - username: 在Manger界面创建的用于访问IoTDB的用户名，即**步骤4**创建的用户。
  - password: 在Manger界面创建的用于访问IoTDB的用户密码，即**步骤4**创建的用户密码。

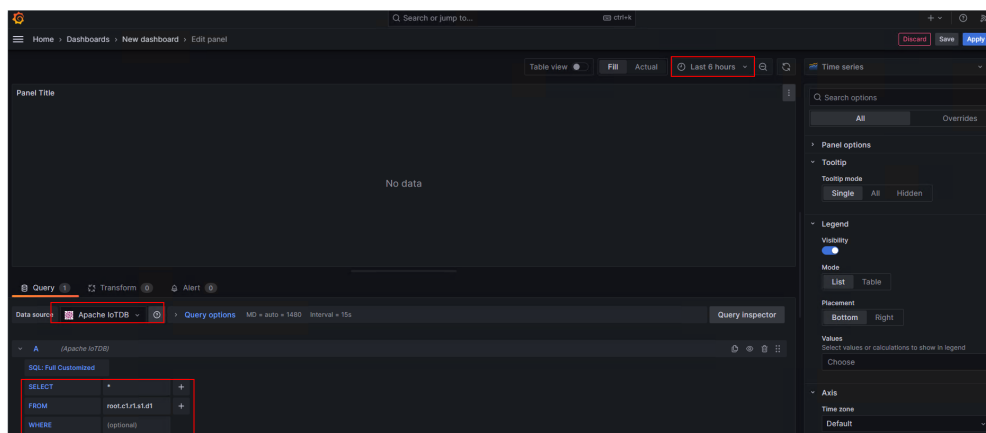
图 2-84 创建 IoTDB 数据源



**步骤5** 在Grafana页面，添加数据监控。

1. 在左侧导航栏选择“Home > Dashboards”，在搜索框右侧选择“New > New Dashboard”，在新建页面单击“Add visualization”进入“Edit panel”页面，并配置以下参数：
  - Data source: 选择已创建好的Apache IoTDB数据源。
  - SQL: 选择要查询的SQL类型，例如“Full Customized”，并填写需要查询的时间序列数据。
  - 在页面右上角选择要查询的数据的时间范围，例如“Last 6 hours”。

图 2-85 添加监控



2. 单击页面右下角的“Query inspector”，再单击“Refresh”，检查监控面板显示的数据。数据无误后单击“Apply”，单击“Save”保存该监控面板。

图 2-86 监控面板数据



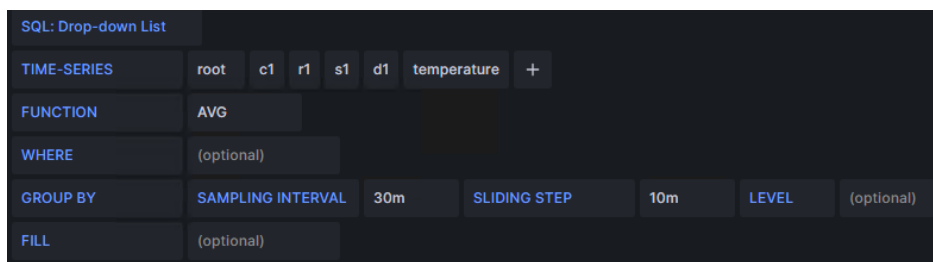
在Grafana查询IoTDB数据。

**步骤6** 在5.1创建的Dashboards中，配置对应的SQL以实现不同数据的查询功能。

例如，查询过去3个小时内，每隔10分钟某一设备的温度湿度平均值指标（滑动窗口）。

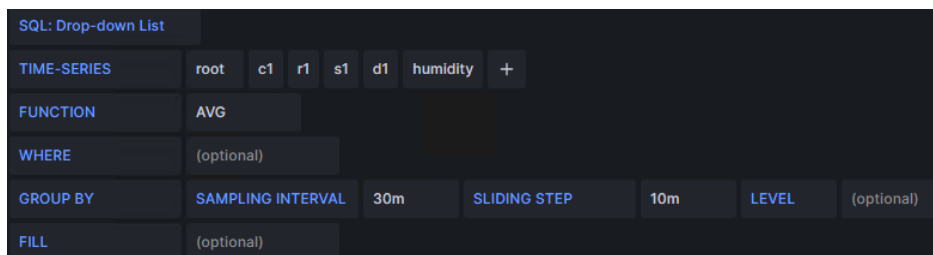
1. 选择SQL类型为“Drop-down List”，并配置查询温度的命令：
  - TIME-SERIES: 设置为“root”，并单击“+”选择指定的时间序列。
  - FUNCTION: 选择聚合函数“AVG”。
  - GROUP BY: “SAMPLING INTERVAL”设置为“30m”，“SLIDING STEP”设置为“10m”。表示采样间隔时间是30分钟，滑动步长是10分钟。
  - 其他配置保持默认即可。

图 2-87 配置温度的查询命令



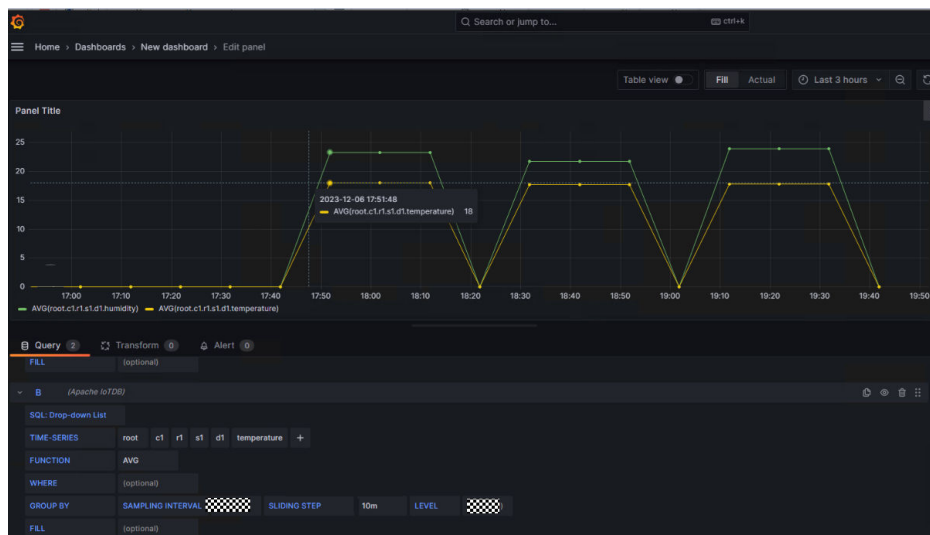
2. 单击“+ Query”，参考6.1配置查询湿度的命令。

图 2-88 配置湿度的查询命令



3. 在页面右上角配置时间范围，设置为“Last 3 hours”；配置显示样式为“Time series”。
4. 单击页面右下角的“Query inspector”，再单击“Refresh”，即可图形化展示查询的IoTDB数据：

图 2-89 图形化查询到的 IoTDB 数据



---结束



# 3 数据迁移

## 3.1 数据迁移方案介绍

### 3.1.1 准备工作

本迁移将指导适用于多种不同场景下的HDFS、HBase、Hive数据向MRS集群的迁移工作。由于数据迁移过程中可能存在数据覆盖、丢失、损坏等风险，因此本指导只作为参考，具体的数据迁移方案的制定及实施需要华为云支持人员协同完成。

数据迁移前源集群的准备工作，目的是防止在数据迁移过程中源集群产生新数据，导致源集群与迁移后的目标集群数据不一致。在数据迁移完成之前，目标集群应处于初始状态，期间不能运行除数据迁移作业外的其他任何业务。

云数据迁移（Cloud Data Migration，简称CDM），是一种高效、易用的数据集成服务。CDM围绕大数据迁移上云和智能数据湖解决方案，提供了简单易用的迁移能力和多种数据源到数据湖的集成能力，降低了客户数据源迁移和集成的复杂性，有效的提高数据迁移和集成的效率，可参考[Hadoop数据迁移到华为云MRS服务](#)、[HBase数据迁移到华为云MRS服务](#)相关内容。

#### 停止集群业务及相关服务

- 如果您的集群涉及到Kafka业务，请先停止所有向Kafka中生产数据的作业，等待Kafka的消费作业消费完Kafka中的存量数据后，再执行下一步操作。
- 停止所有与HDFS、HBase、Hive相关的业务和作业，然后停止HBase、Hive服务。

#### 打通数据传输通道

- 当源集群与目标集群部署在同一区域的不同VPC时，请创建两个VPC之间的网络连接，打通网络层面的数据传输通道。请参见[VPC对等连接](#)。
- 当源集群与目标集群部署在同一VPC但属于不同安全组时，在VPC管理控制台，为每个安全组分别添加安全组规则。规则的“协议”为“ANY”，“方向”为“入方向”，“源地址”为“安全组”且是对端集群的安全组。
  - 为源集群的安全组添加入方向规则，源地址选择目标集群的安全组。
  - 为目标集群的安全组添加入方向规则，源地址选择源集群的安全组。

- 当源集群与目标集群部署在同一VPC同一安全组且两个集群都开启了Kerberos认证，需要为两个集群配置互信，具体请参考[配置跨Manager集群互信](#)。

### 3.1.2 元数据导出

为了保持迁移后数据的属性及权限等信息在目标集群上与源集群一致，需要将源集群的元数据信息导出，以便在完成数据迁移后进行必要的元数据恢复。

需要导出的元数据包括HDFS文件属主/组及权限信息、Hive表描述信息。

#### HDFS 元数据导出

HDFS数据需要导出的元数据信息包括文件及文件夹的权限和属主/组信息，可通过如下HDFS客户端命令导出。

```
$HADOOP_HOME/bin/hdfs dfs -ls -R <migrating_path> > /tmp/hdfs_meta.txt
```

其中，各参数的含义如下：

- *\$HADOOP\_HOME*: 源集群Hadoop客户端安装目录。
- <migrating\_path>: HDFS上待迁移的数据目录。
- /tmp/hdfs\_meta.txt: 导出的元数据信息保存在本地的路径。

#### 说明

如果源集群与目标集群网络互通，且以管理员身份运行hadoop distcp命令进行数据拷贝，可以添加参数“-p”让distcp在拷贝数据的同时在目标集群上分别恢复相应文件的元数据信息，因此在这种场景下可直接跳过本步骤。

#### Hive 元数据导出

Hive表数据存储在HDFS上，表数据及表数据的元数据由HDFS统一按数据目录进行迁移。而Hive表的元数据根据集群的不同配置，可以存储在不同类型的关系型数据库中（如MySQL、PostgreSQL、Oracle等）。

本指导中导出的Hive表元数据即存储在关系型数据库中的Hive表的描述信息。

业界主流大数据发行版均支持Sqoop的安装，如果是自建的社区版大数据集群，可下载社区版Sqoop进行安装。借助Sqoop来解耦导出的元数据与关系型数据库的强依赖，将Hive元数据导出到HDFS上，与表数据一同迁移后进行恢复。

参考步骤如下：

**步骤1** 在源集群上下载并安装Sqoop工具。

请参见<http://sqoop.apache.org/>。

**步骤2** 下载相应关系型数据库的JDBC驱动放置到“*/\${Sqoop\_Home}/lib*”目录。

**步骤3** 执行如下命令导出所有Hive元数据表。

例如所有导出数据保存在HDFS上的“*/user/<user\_name>/<table\_name>*”目录。

```
$Sqoop_Home/bin/sqoop import --connect jdbc:<driver_type>://<ip>:<port>/<database> --table <table_name> --username <user> -password <passwd> -m 1
```

其中，各参数的含义如下：

- *\$Sqoop\_Home*: Sqoop的安装目录。

- `<driver_type>`: 数据库驱动类型。
- `<ip>`: 源集群数据库的IP地址。
- `<port>`: 源集群数据库的端口号。
- `<table_name>`: 待导出的表名称。
- `<user>`: 用户名。
- `<passwd>`: 用户密码。

#### 📖 说明

命令中如果携带认证密码信息可能存在安全风险，在执行命令前建议关闭系统的history命令记录功能，避免信息泄露。

---结束

### 3.1.3 数据复制

根据源集群与目标集群分别所处的区域及网络连通性，可分为以下几种数据复制场景。

#### 同 Region

当源集群与目标集群处于同一Region时，根据[打通数据传输通道](#)进行网络配置，打通网络传输通道。使用Distcp工具执行如下命令将源集群的HDFS、HBase、Hive数据文件以及Hive元数据备份文件复制至目的集群。

```
$HADOOP_HOME/bin/hadoop distcp <src> <dist> -p
```

其中，各参数的含义如下：

- `$HADOOP_HOME`: 目的集群Hadoop客户端安装目录。
- `<src>`: 源集群HDFS目录。
- `<dist>`: 目的集群HDFS目录。

#### 不同 Region

当源集群与目标集群处于不同Region时，用Distcp工具将源集群数据复制到OBS，借助OBS跨区域复制功能（请参见[跨区域复制](#)）将数据复制到对应目的集群所在Region的OBS，然后通过Distcp工具将OBS数据复制到目的集群的HDFS上。由于执行Distcp无法为OBS上的文件设置权限、属主/组等信息，因此当前场景在进行数据导出时也需要将HDFS的元数据信息进行导出并复制，以防HDFS文件属性信息丢失。

#### 线下集群向云迁移

线下集群可以通过如下两种方式将数据迁移至云：

- 云专线（DC）  
为源集群与目标集群之间建立[云专线](#)，打通线下集群出口网关与线上VPC之间的网络，然后参考[同Region](#)执行Distcp进行复制。
- 数据快递服务（DES）  
对于TB或PB级数据上云的场景，华为云提供[数据快递服务 DES](#)。将线下集群数据及已导出的元数据复制到DES盒子，快递服务将数据递送到华为云机房，然后通过[云数据迁移 CDM](#)将DES盒子数据复制到HDFS。

## 3.1.4 数据恢复

### HDFS 文件属性恢复

根据导出的权限信息在目的集群的后台使用HDFS命令对文件的权限及属主/组信息进行恢复。

```
$HADOOP_HOME/bin/hdfs dfs -chmod <MODE> <path>  
$HADOOP_HOME/bin/hdfs dfs -chown <OWNER> <path>
```

### Hive 元数据恢复

在目的集群中安装并使用Sqoop命令将导出的Hive元数据导入MRS集群DBService。

```
$Sqoop_Home/bin/sqoop export --connect jdbc:postgresql://<ip>:20051/hivemeta --table <table_name> --  
username hive -password <passwd> --export-dir <export_from>
```

其中，各参数的含义如下：

- `$Sqoop_Home`: 目的集群上Sqoop的安装目录。
- `<ip>`: 目的集群上数据库的IP地址。
- `<table_name>`: 待恢复的表名称。
- `<passwd>`: hive用户的密码。
- `<export_from>`: 元数据在目的集群的HDFS地址。

#### 📖 说明

命令中如果携带认证密码信息可能存在安全风险，在执行命令前建议关闭系统的**history**命令记录功能，避免信息泄露。

### HBase 表重建

重启目的集群的HBase服务，使数据迁移生效。在启动过程中，HBase会加载当前HDFS上的数据并重新生成元数据。启动完成后，在Master节点客户端执行如下命令加载HBase表数据。

```
$HBase_Home/bin/hbase hbck -fixMeta -fixAssignments
```

命令执行完成后，重复执行如下命令查看HBase集群健康状态直至正常。

```
hbase hbck
```

## 3.2 数据迁移到 MRS 前信息收集

由于离线大数据搬迁有一定的灵活性，迁移前需要掌握现有集群的详细信息，以能够更好地进行迁移决策。

### 业务信息调研

1. 大数据平台及业务的架构图。
2. 大数据平台和业务的数据流图（包括峰值和均值流量等）。

识别平台数据接入源、大数据平台数据流入方式（实时数据上报、批量数据抽取）、分析平台数据流向。

数据在平台内各个组件间的流向，比如使用什么组件采集数据，采集完数据后数据如何流向下一层组件，使用什么组件存储数据，数据处理过程中的工作流等。

3. 业务作业类型Hive SQL、Spark SQL、Spark Python等，是否需要使用MRS的第三方包，参考[MRS应用开发样例](#)。
4. 调度系统，需要考虑调度系统对接MRS集群。
5. 迁移后，业务割接允许中断时长，识别平台业务优先级。  
识别在迁移过程中不能中断的业务、可短时中断的业务、整体业务迁移可接受的迁移时长，梳理业务迁移顺序。
6. 客户端部署要求。
7. 业务执行时间段和高峰时间段。
8. 大数据集群的数量和大数据集群功能划分，分析平台业务模型。  
各个集群或各个组件分别负责什么业务，处理什么类型的数据。比如实时/离线数据分别使用什么组件处理、数据格式类型、压缩算法等。

## 集群基本信息收集

表 3-1 集群基本信息

参数	说明
集群名称	-
集群版本	MRS、CDM等集群的版本信息。
节点数及规格	调研现有集群节点数和节点规格。 如果集群硬件异构，请收集多种规格和对应节点数，参见 <a href="#">表3-2</a> 。 例如： <ul style="list-style-type: none"><li>• 2台32U64G机器部署NameNode + ResourceManager</li><li>• 2台32U64G机器部署HiveServer</li><li>• 20台16U32G机器部署DataNode和NodeManager</li></ul>
是否开启Kerberos认证	是或否
权限控制及说明	调研各个开启ACL权限控制的组件和配置，通常涉及Yarn、Hive、Impala、HBase等组件。 使用Ranger、Sentry或组件开源的权限能力进行权限控制。
所在Region/AZ	云上资源填写项
虚拟私有云	云上资源填写项
子网	云上资源填写项
安全组	云上资源填写项

表 3-2 硬件信息调研表

节点组	CPU和内存信息	磁盘和网络（按节点组统计）		HDFS			Yarn	
				Name Node	DataNode	JournalNode	Node Manager	ResourceManager
-	-	磁盘信息（数据盘大小、磁盘IO、当前磁盘使用率和IO情况）	网络（网卡带宽大小、网络读写速度和峰值）					
master 1	(16U6 4G)	-	-	1	-	1	-	1
master 2	(16U6 4G)	-	-	1	-	1	-	1
master 3	(16U6 4G)	-	-	-	-	1	-	-
Core-group 1	(32U1 28G)* 数量	-	-	-	1	-	1	-
Core-group 1	(32U1 29G)	-	-	-	-	-	-	-
Core-group 1	(32U1 30G)	-	-	-	-	-	-	-

## 大数据组件信息

使用的大数据组件信息和规划的新版本大数据集群版本信息比较，主要识别版本差异可能对迁移过程的影响，以及对迁移后业务兼容性的影响。

表 3-3 大数据组件信息

大数据组件	源端集群版本	目的端集群版本（以MRS集群版本为准）	说明
HDFS/OBS（或其他文件存储系统）	<i>Hadoop 2.8.3</i>	<i>Hadoop 3.3.1</i>	-
Hive	<i>1.2.1</i>	<i>2.3.3</i>	存储元数据的数据库：MySQL

大数据组件	源端集群版本	目的端集群版本 (以MRS集群版本 为准)	说明
HBase	1.3.1	1.3.1	-
Spark	2.2.2	3.1.1	-
Kafka	1.1.0	2.11-2.4.0	-
Oozie	2.x	5.1.0	-
MySQL	5.7.1	RDS	-
Flink	1.7	1.15	-
...	...	...	-

## 待迁移的存量数据及数据量统计

如果使用HDFS作为文件存储系统，可以通过客户端执行`hadoop fs -du -h HDFS文件目录`命令统计路径下的文件大小。

表 3-4 现有数据量统计

大数据组件	待迁移数据的路径	数据量大小	文件个数或表个数
HDFS/OBS (或其他文件存储系统)	<code>/user/helloworld</code>	XXX	总共: XXX个文件 小于2 MB的文件数量: XXX个
Hive	<code>/user/hive/warehouse/</code>	XXX	表个数: XXX
HBase	<code>/hbase</code>	XXX	表个数: XXX Region个数: XXX

## 每天新增数据量统计

每天新增数据量主要评估数据增长速度(可以按天/小时等周期维度)。在第一次全量迁移数据后，后续可以定期搬迁老集群新增数据，直到业务完成最终割接。

表 3-5 新增数据量统计

大数据组件	待迁移的数据路径	新增数据量大小
HDFS/OBS (或其他文件存储系统)	<code>/user/helloworld</code>	XXX
Hive	<code>/user/hive/warehouse/</code>	XXX

大数据组件	待迁移的数据路径	新增数据量大小
HBase	/hbase	XXX

## 网络出口带宽能力

- 迁移数据可以使用的最大网络带宽和专线带宽（是否可调）。
- 迁移数据作业每天可以运行的时间段。

## 流式 Kafka 集群信息收集

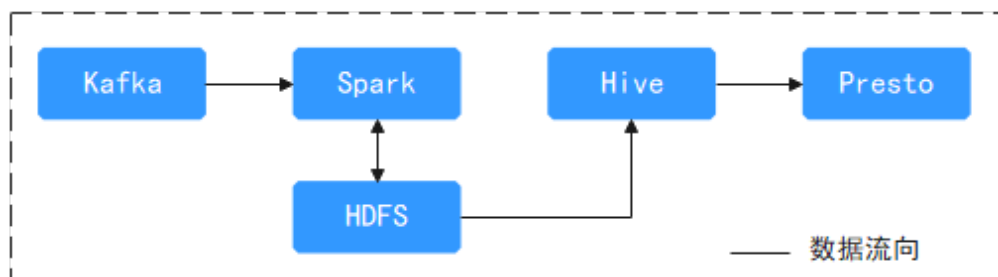
表 3-6 流式 Kafka 集群信息

收集信息项	描述
Kafka的Topic数量和名称	-
Kafka的本地数据暂存时间	如果每个Topic配置不一样，按Topic粒度收集。
每个Topic的副本数和Partition数量	默认为2，副本数越多数据越可靠，也会消耗磁盘空间。 如果每个Topic配置不一样，按Topic粒度收集。
Kafka生产和消费的流量大小	细化到Topic级别。
Kafka客户端ACK配置Acks	-

## 数据迁移模型样例

- 一个离线分析平台的客户业务系统，由Spark Streaming消费Kafka数据存入HDFS，HDFS上进行小文件合并后加载到Hive表中，运营人员可以通过Presto进行Hive数据查询。

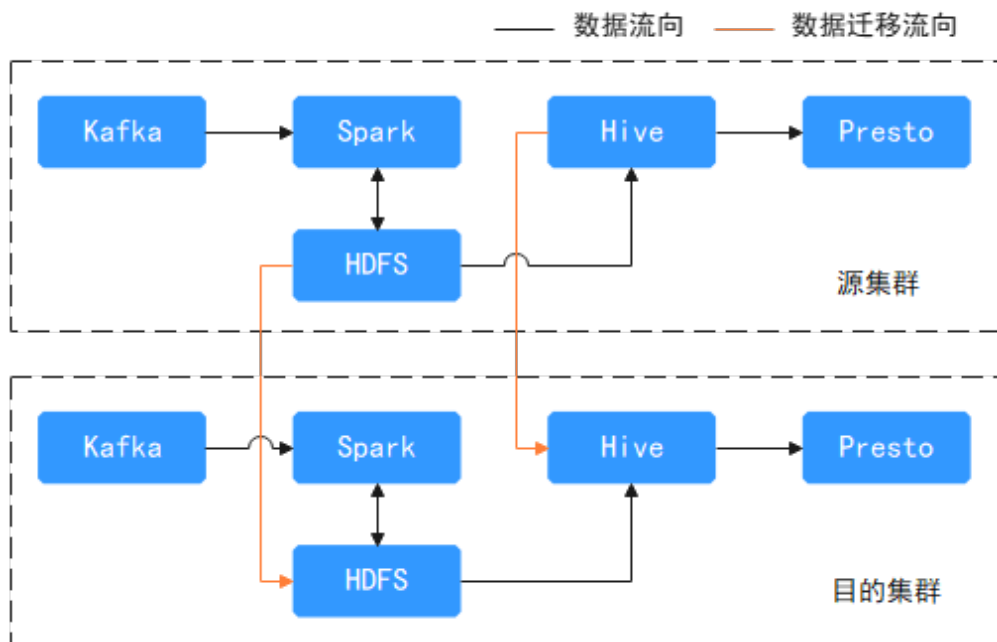
图 3-1 源集群业务图



- 针对大数据离线平台包括HDFS和Hive数据需要迁移，Kafka、Spark Streaming、HDFS、Hive、Presto的业务程序要在目的端集群上部署。



图 3-2 迁移示意图



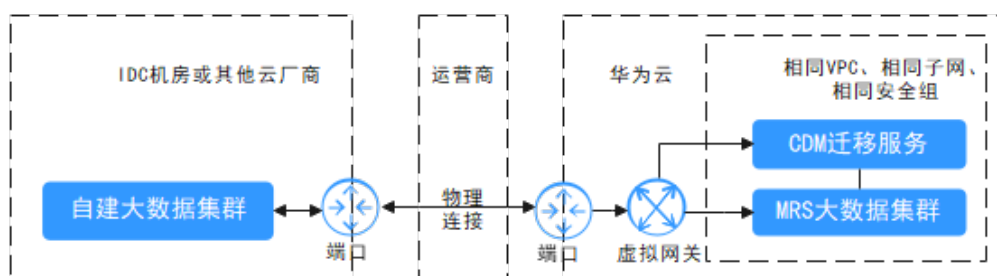
### 3.3 数据迁移到 MRS 前网络准备

#### 数据迁移网络方案说明

进行大数据迁移时，需要保证源端集群和目的端集群之间的网络互通，例如使用 `hadoop distcp` 命令跨集群复制数据时需要所有 DataNode 节点网络互通。根据不同的迁移场景需要使用不同的方式先打通两套集群之间网络连接。

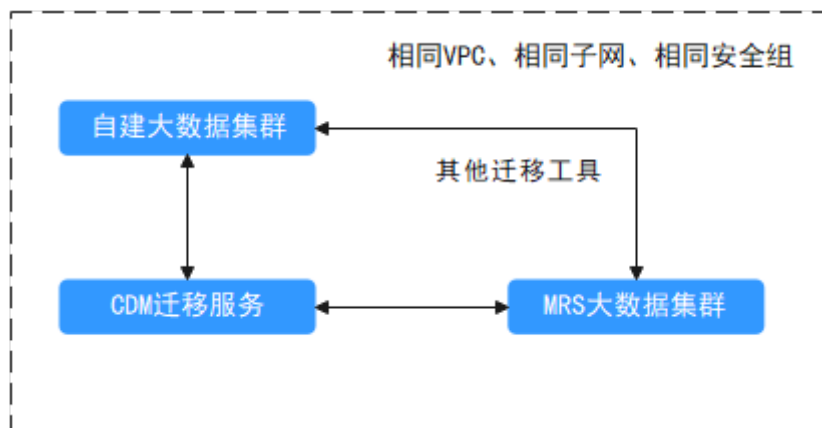
- 客户线下数据中心迁移数据到华为云 MRS 集群，通过云专线服务为用户搭建本地数据中心与云上 VPC 之间的专属连接通道。可以使用华为云的云专线服务或使用第三方的云专线服务来连通华为云网络。

图 3-3 线下数据中心迁移



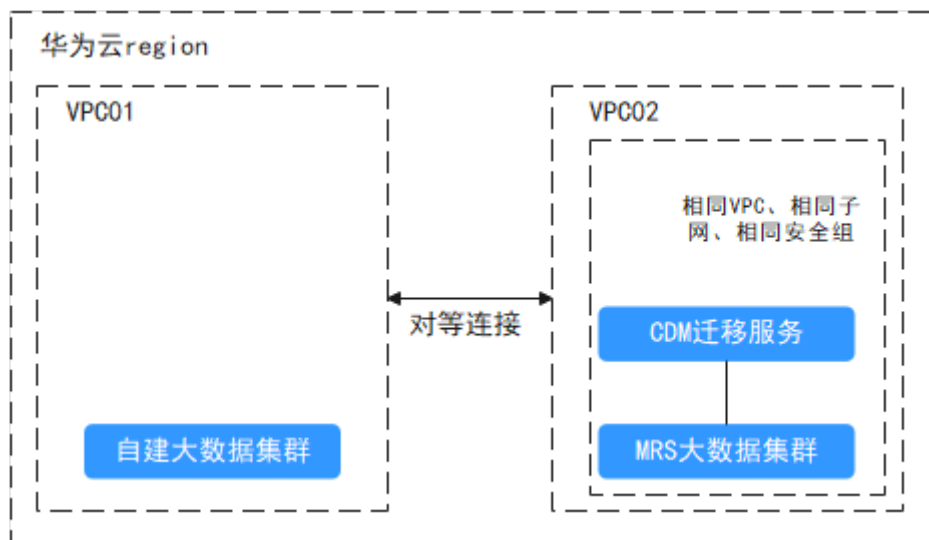
- 客户在华为云上自建大数据集群（或老版本的 MRS 集群）需要迁移到华为云 MRS 集群，且在同一个 Region 区域和 VPC 子网，可以使自建集群和 MRS 集群使用相同安全组、VPC、子网网络，从而保证网络连通。

图 3-4 线上同 Region 同 VPC 迁移



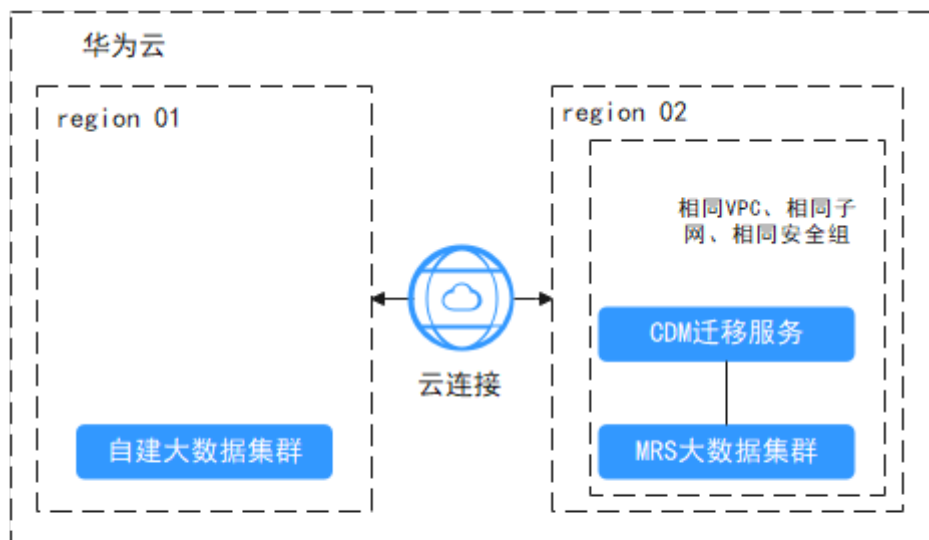
- 客户在华为云上自建大数据集群（或老版本的MRS集群）需要迁移到华为云MRS集群，且在同一个Region区域，但是使用不同VPC子网。需要使用VPC对等连接方式配置网络连通。

图 3-5 线上同 Region 不同 VPC 迁移



- 客户在华为云上自建大数据集群（或老版本的MRS集群）需要迁移到华为云MRS集群，但在不同Region区域，可以通过使用云连接构建跨区域VPC的网络连接。

图 3-6 线上不同 Region 迁移



## 数据迁移常用端口

表 3-7 MRS 数据迁移常用组件

组件	配置参数	默认端口	端口说明
HDFS	dfs.namenode.rpc.port	9820	Hadoop 2.x和Hadoop 3.x版本。迁移过程中，需要访问NameNode获取文件列表。
	dfs.datanode.port	25009	迁移过程中，需要访问DataNode读取具体文件数据。
ZooKeeper	clientPort	2181	ZooKeeper客户端连接ZooKeeper服务器。
Kerberos	kdc_ports	21732	Kerberos服务认证，非Kerberos集群不涉及。
Hive	hive.metastore.port	9083	Hive 2.x和Hive3.x版本。MetaStore提供Thrift服务的端口。迁移过程中，需要访问该端口查询表元数据信息。
HBase	hbase.master.port	16000	HBase1.x和HBase 2.x版本。HMaster RPC端口。该端口用于HBase客户端连接到HMaster。

组件	配置参数	默认端口	端口说明
	hbase.regions erver.port	16020	RS (RegionServer) RPC端口。该端口用于HBase客户端连接到RegionServer。
Manager	N/A	28443	FusionInsight/MRS Manager页面端口。 CDM迁移时候访问该地址获取集群配置。
	N/A	20009	FusionInsight/MRS Manager CAS协议端口，用于登录认证。

## 3.4 使用 CDM 服务迁移 Hadoop 数据至 MRS 集群

### 应用场景

本章节适用于将线下IDC机房或者公有云Hadoop集群中的数据（支持数据量在几十TB级别或以下的数据量级）迁移到华为云MRS服务。

本章节以通过[华为云CDM服务](#) 2.9.1.200版本进行数据迁移为例介绍。不同版本操作可能有差异，具体操作详情以实际版本对应的操作指导为准。

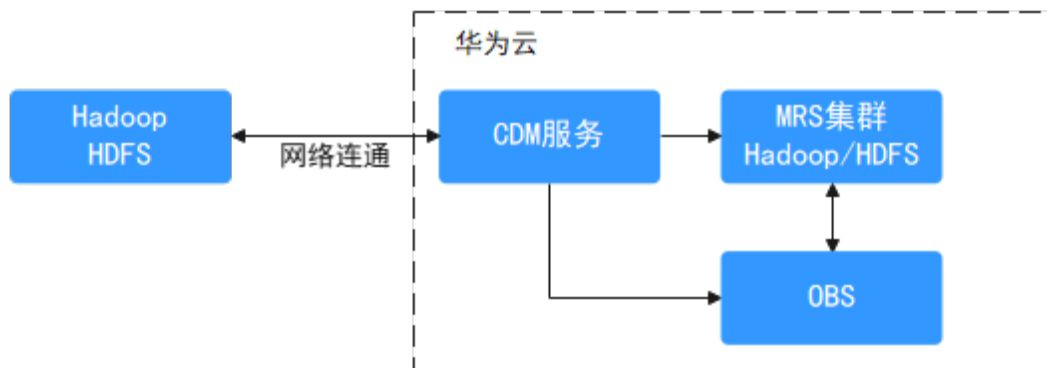
CDM服务支持迁移的数据源可参考[支持的数据源](#)，数据源为Apache HDFS时，建议使用的版本为2.8.X、3.1.X，请执行搬迁前务必确认是否支持搬迁。

### 方案架构

CDM围绕大数据迁移上云和智能数据湖解决方案，提供了简单易用的迁移能力和多种数据源到数据湖的集成能力，降低了客户数据源迁移和集成的复杂性，有效的提高您数据迁移和集成的效率。

CDM服务迁移Hadoop数据至MRS集群方案如[图3-7](#)所示。

图 3-7 Hadoop 数据迁移示意



## 方案优势

- 简单易用：免编程，向导式任务开发界面，通过简单配置几分钟即可完成迁移任务开发。
- 迁移效率高：基于分布式计算框架进行数据任务执行和数据传输优化，并针对特定数据源写入做了专项优化，迁移效率高。
- 实时监控：迁移过程中可以执行自动实时监控、告警和通知操作。

## 约束与限制

搬迁数据量较大时，对网络通信要求较高，执行搬迁任务时，可能会影响其他业务，建议在业务空闲期进行数据迁移任务。

## 步骤 1：新建数据连接

**步骤1** 登录CDM管理控制台。

**步骤2** **创建CDM集群**，该CDM集群的安全组、虚拟私有云、子网需要和迁移目的端集群保持一致，保证CDM集群和MRS集群之间网络互通。

**步骤3** 在“集群管理”页面单击待操作集群对应“操作”列的“作业管理”。

**步骤4** 在“连接管理”页签，单击“新建连接”。

**步骤5** 参考**CDM服务的新建连接**页面，分别添加到迁移源端集群和迁移目的端集群的两个HDFS连接。

连接类型根据实际集群来选择，如果是MRS集群，连接器类型可以选择“MRS HDFS”，如果是自建集群可以选择“Apache HDFS”。

图 3-8 HDFS 连接



----结束

## 步骤 2：新建迁移作业

**步骤1** 在“表/文件迁移”页签，单击“新建作业”。

**步骤2** 选择源连接、目的连接：

- 作业名称：用户自定义任务名称，名称由英文字母、下划线或者数字组成，长度必须在1到256个字符之间。
- 源连接名称：选择迁移源端集群的HDFS连接，作业运行时将从此端复制导出数据。
- 目的连接名称：选择迁移目的端集群的HDFS连接，作业运行时会将数据导入此端。

**步骤3** 请参见**配置HDFS源端参数**配置源端连接的作业参数，需要迁移的文件夹可通过“目录过滤器”和“文件过滤器”参数设置符合规则的目录和文件进行迁移。

例如迁移匹配“/user/test\*”文件夹下文件，该场景下“文件格式”固定为“二进制格式”。

图 3-9 配置作业参数

The screenshot shows a configuration page for a job named 'move-hdfs'. It is divided into two main sections: 'Source Job Configuration' (源端作业配置) and 'Destination Job Configuration' (目的端作业配置).  
 In the 'Source Job Configuration' section:  
 - 'Source Connection Name' (源连接名称) is 'hdfs-source'.  
 - 'Source Path or File' (源目录或文件) is '/user'.  
 - 'List Files' (列举文件) is set to 'Yes' (是).  
 - 'File Format' (文件格式) is 'Binary' (二进制格式), highlighted with a red box.  
 - 'Automatic Job Identification' (自动作业标识文件) is 'Yes' (是).  
 - 'Filter Type' (过滤类型) is 'Wildcard' (通配符).  
 - 'Destination Filter' (目标过滤器) is 'test\*', highlighted with a red box.  
 - 'File Filter' (文件过滤器) is empty.  
 In the 'Destination Job Configuration' section:  
 - 'Destination Connection Name' (目的连接名称) is 'target'.  
 - 'Destination Path' (写入目录) is '/user'.  
 - 'File Format' (文件格式) is 'Binary' (二进制格式), highlighted with a red box.  
 - 'Overwrite File Handling' (重复文件处理方式) is 'Replace Duplicate Files' (替换重复文件).  
 - 'Show Advanced Properties' (显示高级属性) is a link.

步骤4 请参见配置HDFS目的端参数配置目的端连接的作业参数。

步骤5 单击“下一步”进入任务配置页面。

- 如需定期将新增数据迁移至目的端集群，可在该页面进行配置，也可在任务执行后再参考步骤3配置定时任务。
- 如无新增数据需要后续定期迁移，则跳过该页面配置直接单击“保存”回到作业管理界面。

图 3-10 任务配置

The screenshot shows the 'Task Configuration' (任务配置) page. It includes the following settings:  
 - 'Job Failure Retries' (作业失败重试): 'No Retries' (不重试).  
 - 'Job Group' (作业分组): 'DEFAULT'.  
 - 'Scheduled Execution' (是否定时执行): 'Yes' (是).  
 - 'Repeat Cycle' (重复周期): '1' day, with a note '隔\*\*天执行一次'.  
 - 'Validity Period' (有效期):  
 - 'Start Time' (开始时间): '2022/12/12 14:45'.  
 - 'End Time' (结束时间): '请选择日期时间'.  
 - 'Show Advanced Properties' (显示高级属性) is a link.

步骤6 选择“作业管理”的“表/文件迁移”页签，在待运行作业的“操作”列单击“运行”，即可开始HDFS文件数据迁移，并等待作业运行完成。

----结束

### 步骤 3: 检查迁移后的文件信息

步骤1 登录迁移目的端集群主管理节点。

步骤2 在集群客户端内执行 `hdfs dfs -ls -h /user/` 命令查看迁移目的端集群中已迁移的文件。

步骤3 (可选) 如果源端集群中有新增数据需要定期将新增数据迁移至目的端集群，则配置定期任务增量迁移数据，直到所有业务迁移至目的端集群。

1. 在CDM集群中选择“作业管理”的“表/文件迁移”页签。
2. 在迁移作业的“操作”列选择“更多 > 配置定时任务”。
3. 开启定时执行功能，根据具体业务需求设置重复周期，并设置有效期的结束时间为所有业务割接到新集群之后的时间。

图 3-11 配置定时任务

**配置定时任务**

是否定时执行  是  否 [了解如何配置定时任务参数规则](#)

分 小时 **天** 周 月

重复周期 (天)  隔\*\*天执行一次

有效期

开始时间

结束时间

----结束

## 3.5 使用 CDM 服务迁移 HBase 数据至 MRS 集群

### 应用场景

本章节适用于将线下IDC机房或者公有云HBase集群中的数据（支持数据量在几十TB级别或以下的数据量级）迁移到华为云MRS服务。

本章节以通过[华为云CDM服务](#) 2.9.1.200版本进行数据迁移为例介绍。不同版本操作可能有差异，具体操作详情以实际版本对应的操作指导为准。

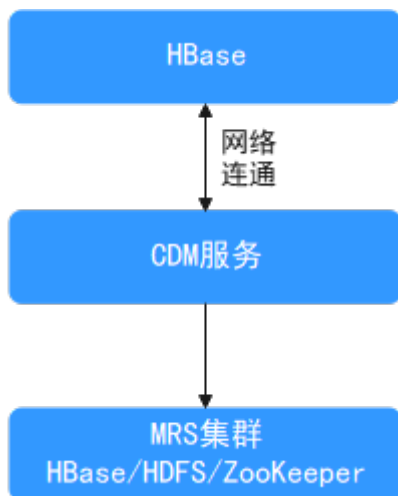
CDM服务支持迁移的数据源可参考[支持的数据源](#)，数据源为Apache HBase时，建议使用的版本为2.1.X、1.3.X，请执行搬迁前务必确认是否支持搬迁。

### 方案架构

HBase会把数据存储到HDFS上，主要包括HFile文件和WAL文件，由配置项“hbase.rootdir”指定在HDFS上的路径，华为云MRS集群的默认存储位置是“/hbase”文件夹下。

HBase自带的一些机制和工具命令也可以实现数据搬迁，例如通过导出Snapshots快照、Export/Import、CopyTable方式等，可以参考Apache官网相关内容。

图 3-12 HBase 数据迁移示意



## 方案优势

场景化迁移通过迁移快照数据然后再恢复表数据的方法，能大大提升迁移效率。

## 约束与限制

搬迁数据量较大时，对网络通信要求较高，执行搬迁任务时，可能会影响其他业务，建议在业务空闲期进行数据迁移任务。

## 全量数据迁移

- 步骤1** 登录CDM管理控制台。
- 步骤2** **创建CDM集群**，该CDM集群的安全组、虚拟私有云、子网需要和迁移目的端集群保持一致，保证CDM集群和待迁移的MRS集群之间网络互通。
- 步骤3** 在“集群管理”页面单击待操作集群对应“操作”列的“作业管理”。
- 步骤4** 在“连接管理”页签，单击“新建连接”。
- 步骤5** 参考**CDM服务的新建连接**页面，添加到迁移源端集群的连接，连接器类型根据实际集群来选择，例如连接器类型选择“Apache HBase”。

### 📖 说明

(可选) HBase迁移建议使用高权限用户，单击“显示高级属性”，新增迁移所需用户“hadoop.username = 用户名(如omm用户)”。



图 3-13 到迁移源端集群的连接

\* 名称: hbase-source

\* 连接器: HBase

\* HBase类型: Apache HBase

\* ZK链接地址: .2181

\* 认证类型: SIMPLE

IP与主机名映射

\* HBase版本: HBASE\_2\_X

\* 运行模式: STANDALONE

是否使用集群配置:  是  否

隐藏高级属性

属性名称	值	操作
hadoop.user.name	omm	删除

属性配置

**步骤6** 在“连接管理”页签，单击“新建连接”。

**步骤7** 参考[CDM服务的新建连接](#)页面，添加到迁移目的端集群的连接，连接器类型根据实际集群来选择，例如连接器类型选择“MRS HBase”。

#### 说明

(可选) HBase迁移建议使用高权限用户，单击“显示高级属性”，新增迁移所需用户“hadoop.user.name = 用户名(如omm用户)”。

图 3-14 到迁移目的端集群的连接

\* 名称

\* 连接器

\* HBase类型

\* Manager IP  [选择](#)

\* 用户名

\* 密码

\* 认证类型

\* HBase版本

\* 运行模式

是否使用集群配置  是  否

[隐藏高级属性](#)

属性名称	值	操作
hadoop.user.name	omm	<a href="#">删除</a>

[属性配置](#)

**步骤8** 选择“作业管理”的“表/文件迁移”页签，单击“新建作业”。

**步骤9** 进入作业参数配置界面，配置作业名称、源端作业和目的端作业参数，并选择要迁移的数据表，单击“下一步”。

图 3-15 HBase 作业配置

作业配置

\* 作业名称

**源端作业配置**

\* 源连接器名称

\* 表名

整表迁移  是  否

列族

[显示高级属性](#)

**目的端作业配置**

\* 目的连接器名称

\* 表名

\* 导入前清空数据  是  否

自动创表

[显示高级属性](#)

**步骤10** 配置源字段和目的字段的映射关系，并单击“下一步”。

**步骤11** 进入任务配置页面，不做修改，直接单击“保存”。

**步骤12** 选择“作业管理”的“表/文件迁移”页签，在待运行作业的“操作”列单击“运行”，即可开始HBase数据迁移。

**步骤13** 迁移完成后，可以在目的端集群和源端集群的HBase Shell命令行中，通过同样的查询语句，对比查询结果进行验证。

例如：

- 在目的端集群和源端集群上通过查询BTable表的记录数来确认数据条数是否一致，可添加“--endtime”参数排除迁移期间源端集群上有数据更新的影响。

```
hbase org.apache.hadoop.hbase.mapreduce.RowCounter BTable --  
endtime=1587973835000
```

图 3-16 查询 BTable 表的记录数

```
2020-04-27 16:15:09,500 INFO [main] mapreduce.Job: map 56% reduce 0%  
2020-04-27 16:15:17,528 INFO [main] mapreduce.Job: map 67% reduce 0%  
2020-04-27 16:15:25,566 INFO [main] mapreduce.Job: map 89% reduce 0%  
2020-04-27 16:15:30,584 INFO [main] mapreduce.Job: map 100% reduce 0%  
2020-04-27 16:15:30,592 INFO [main] mapreduce.Job: Job job_1587471561730_0063 completed successfully  
2020-04-27 16:15:30,653 INFO [main] mapreduce.Job: Counters: 46  
File System Counters  
  FILE: Number of bytes read=0  
  FILE: Number of bytes written=2163033  
  FILE: Number of read operations=0  
  FILE: Number of large read operations=0  
  FILE: Number of write operations=0  
  HDFS: Number of bytes read=2474  
  HDFS: Number of bytes written=0  
  HDFS: Number of read operations=9  
  HDFS: Number of large read operations=0  
  HDFS: Number of write operations=0  
Job Counters  
  Killed map tasks=3  
  Launched map tasks=12  
  Data-local map tasks=10  
  Rack-local map tasks=2  
  Total time spent by all maps in occupied slots (ms)=1442868  
  Total time spent by all reduces in occupied slots (ms)=0  
  Total time spent by all map tasks (ms)=360717  
  Total vcore-milliseconds taken by all map tasks=360717  
  Total megabyte-milliseconds taken by all map tasks=738748416  
Map-Reduce Framework  
  Map input records=20646264  
  Map output records=0  
  Input split bytes=2474  
  Spilled Records=0  
  Failed Shuffles=0  
  Merged Map outputs=0  
  GC time elapsed (ms)=1674  
  CPU time spent (ms)=86120  
  Physical memory (bytes) snapshot=3175682048  
  Virtual memory (bytes) snapshot=33582198784  
  Total committed heap usage (bytes)=1950875648  
HBase Counters  
  BYTES_IN_REMOTE_RESULTS=0  
  BYTES_IN_RESULTS=3280268799  
  MILLIS_BETWEEN_NEXTS=280535  
  NOT_SERVING_REGION_EXCEPTION=0  
  NUM_SCANNER_RESTARTS=0  
  NUM_SCAN_RESULTS_STALE=0  
  REGIONS_SCANNED=9  
  REMOTE_RPC_CALLS=0  
  REMOTE_RPC_RETRIES=0  
  ROWS_FILTERED=0  
  ROWS_SCANNED=20646264  
  RPC_CALLS=206485  
  RPC_RETRIES=0  
  org.apache.hadoop.hbase.mapreduce.RowCounter$RowCounterMapper$Counters  
  ROWS=20646264  
File Input Format Counters  
  Bytes Read=0  
File Output Format Counters  
  Bytes Written=0  
root@node-master1a0b:~#
```

- 可执行以下命令查询指定时间段内的数据进行对比。

```
scan 'BTable', {TIMERANGE=>[1587973235000, 1587973835000]}
```

----结束

## 增量数据迁移

在业务割接前，如果源端集群上有新增数据，需要定期将新增数据搬迁到目的端集群。一般每天更新的数据量在GB级别可以使用CDM的“整库迁移”指定时间段的方式进行HBase新增数据迁移。

当前使用CDM的“整库迁移”功能时的限制：如果源HBase集群中被删除操作的数据无法同步到目的端集群上。

场景迁移的HBase连接器不能与“整库迁移”共用，因此需要单独配置“HBase”连接器。

**步骤1** 参考**全量数据迁移的步骤1~步骤7**步骤新增两个“HBase”连接器，连接器类型根据实际集群来选择。

例如选择连接器类型时分别为源端集群和目的端集群选择“MRS HBase”和“Apache HBase”。

图 3-17 HBase 增量迁移连接

名称	类型	连接信息
<input type="checkbox"/>	HBase 连接器	HBase类型 :MRS Manager IP : 用户名 : 认证类型 :SIMPLE HBase版本 :HBASE_2_X 运行模式 :EMBEDDED 是否使用集群配置 :false
<input type="checkbox"/>	HBase 连接器	HBase类型 :MRS Manager IP : 用户名 : 认证类型 :SIMPLE HBase版本 :HBASE_2_X 运行模式 :EMBEDDED 是否使用集群配置 :false

**步骤2** 选择“作业管理”的“整库迁移”页签，单击“新建作业”。

**步骤3** 进入作业参数配置界面，作业相关信息配置完成后单击“下一步”。

- 作业名称：用户自定义作业名称，例如hbase-increase。
- 源端作业配置：源连接名称请选择新创建的到源端集群的连接名称，并展开高级属性配置迁移数据的时间段。
- 目的端作业配置：目的连接名称请选择新创建的到目的端集群的连接名称，其他不填写。

图 3-18 HBase 增量迁移作业配置

作业配置

\* 作业名称

源端作业配置

\* 源连接名称

隐藏高级属性

起始时间

终止时间

目的端作业配置

\* 目的连接名称

\* 导入前清空数据

自动创表

显示高级属性

**步骤4** 选择要迁移的数据表，单击“下一步”，单击“保存”。

**步骤5** 选择“作业管理”的“整库迁移”页签，在待运行作业的“操作”列单击“运行”，即可开始HBase数据增量迁移。

----结束

## 3.6 使用 CDM 服务迁移 Hive 数据至 MRS 集群

### 应用场景

本章节适用于将线下IDC机房或者公有云Hive集群中的数据（支持数据量在几十TB级别或以下的数据量级）迁移到华为云MRS服务。

使用华为云CDM服务“场景迁移功能”可以一键式便捷地完成Hive数据的迁移。

本章节以通过华为云CDM服务 2.9.1.200版本进行数据迁移为例介绍。不同版本操作可能有差异，具体操作详情以实际版本对应的操作指导为准。

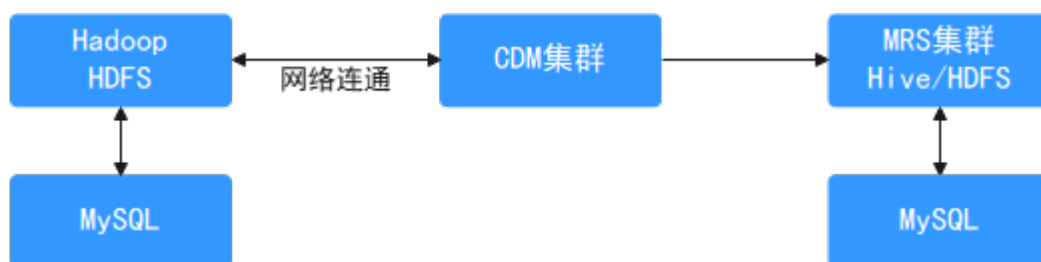
CDM服务支持迁移的数据源可参考[支持的数据源](#)，数据源为Apache Hive时，不支持2.x版本，建议使用的版本为1.2.X、3.1.X，请执行搬迁前务必确认是否支持搬迁。

### 方案架构

Hive数据迁移分两部分内容：

- Hive的元数据信息，存储在MySQL等数据库中。MRS Hive集群的元数据会默认存储到MRS DBService组件，也可以选择RDS（MySQL）作为外置元数据库。
- Hive的业务数据，存储在HDFS文件系统或OBS对象存储中。

图 3-19 Hive 数据迁移示意



### 方案优势

场景化迁移通过迁移快照数据然后再恢复表数据的方法，能大大提升迁移效率。

### 约束与限制

搬迁数据量较大时，对网络通信要求较高，执行搬迁任务时，可能会影响其他业务，建议在业务空闲期进行数据迁移任务。

### 迁移 Hive 数据

- 步骤1** 登录CDM管理控制台。
- 步骤2** [创建CDM集群](#)，该CDM集群的安全组、虚拟私有云、子网需要和迁移目的端集群保持一致，保证CDM集群和MRS集群之间网络互通。
- 步骤3** 在“集群管理”页面单击待操作集群对应“操作”列的“作业管理”。
- 步骤4** 在“连接管理”页签，单击“新建连接”。

**步骤5** 参考**CDM服务的新建连接**页面，分别添加到迁移源端集群和迁移目的端集群的连接。

连接类型根据实际集群来选择，如果是MRS集群，连接器类型可以选择“MRS Hive”，如果是自建集群可以选择“Apache Hive”。

图 3-20 创建 Hive 连接

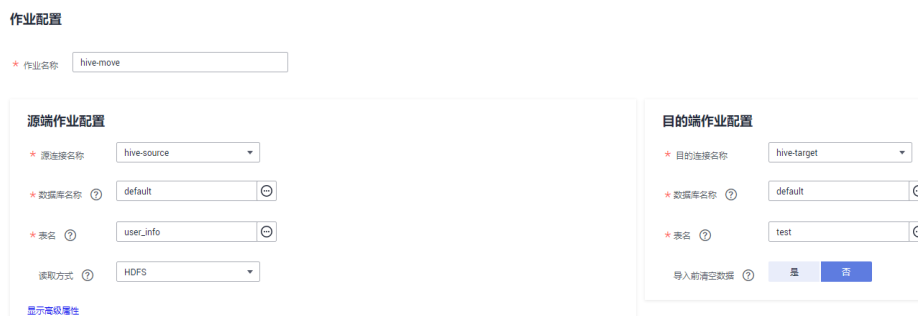


**步骤6** 在迁移目的端集群中创建数据迁移后的存储数据库。

**步骤7** 选择“作业管理”的“表/文件迁移”页签，单击“新建作业”。

**步骤8** 进入作业参数配置界面，配置作业名称，并分别为源连接和目的连接选择**步骤5**中创建的对应数据连接并选择要迁移的数据库和表名，单击“下一步”。

图 3-21 Hive 作业配置



**步骤9** 配置源字段和目的字段的映射关系，并单击“下一步”。

**步骤10** 进入任务配置页面，不做修改，直接单击“保存”。

**步骤11** 选择“作业管理”的“表/文件迁移”页签，在待运行作业的“操作”列单击“运行”，即可开始Hive数据迁移。

**步骤12** 迁移完成后，可以在目的端集群和源端集群的Hive Beeline命令行中，通过同样的查询语句，对比查询结果进行验证。

例如在目的端集群和源端集群上通过查询catalog\_sales表的记录数来确认数据条数是否一致。

```
select count(*) from catalog_sales;
```

图 3-22 源端集群数据记录

```
1 row selected (0.098 seconds)
0: jdbc:hive2://192.168.0.216:2181,192.168.0.> select count(*) from catalog_sales;
INFO : Compiling command(queryId=omm_20200424173337_aaf0d972-b100-4f4a-87c4-959fdb6a2c4f): select count(*) from catalog_sales
INFO : Concurrency mode is disabled, not creating a lock manager
INFO : Semantic Analysis Completed (retrial = false)
INFO : Returning Hive schema: Schema(fieldSchemas:[FieldSchema(name: c0, type:bigint, comment:null)], properties:null)
INFO : EXPLAIN output for queryid omm_20200424173337_aaf0d972-b100-4f4a-87c4-959fdb6a2c4f : STAGE DEPENDENCIES:
      Stage-0 is a root stage [FETCH]

STAGE PLANS:
  Stage: Stage-0
    Fetch Operator
      limit: 1
      Processor Tree:
        ListSink

INFO : Completed compiling command(queryId=omm_20200424173337_aaf0d972-b100-4f4a-87c4-959fdb6a2c4f); Time taken: 0.263 seconds
INFO : Concurrency mode is disabled, not creating a lock manager
INFO : Executing command(queryId=omm_20200424173337_aaf0d972-b100-4f4a-87c4-959fdb6a2c4f): select count(*) from catalog_sales
INFO : Completed executing command(queryId=omm_20200424173337_aaf0d972-b100-4f4a-87c4-959fdb6a2c4f); Time taken: 0.001 seconds
INFO : OK
INFO : Concurrency mode is disabled, not creating a lock manager
+-----+
|   _c0   |
+-----+
| 43204059 |
+-----+
1 row selected (0.275 seconds)
0: jdbc:hive2://192.168.0.216:2181,192.168.0.>
```

图 3-23 目的端集群数据记录

```
INFO : Completed compiling command(queryId=omm_20200424173329_53ad05b4-e097-44c4-9a8f-9f77e7087888); Time taken: 0.845 seconds
INFO : Concurrency mode is disabled, not creating a lock manager
INFO : Executing command(queryId=omm_20200424173329_53ad05b4-e097-44c4-9a8f-9f77e7087888): select count(*) from catalog_sales
INFO : Query ID = omm_20200424173329_53ad05b4-e097-44c4-9a8f-9f77e7087888
INFO : Total jobs = 1
INFO : Launching Job 1 out of 1
INFO : Starting task [Stage-1:MAPRED] in serial mode
INFO : Subscribed to counters: [] for queryId: omm_20200424173329_53ad05b4-e097-44c4-9a8f-9f77e7087888
INFO : Session is already open
INFO : Dag name: select count(*) from catalog_sales (Stage-1)
INFO : Tez session was closed. Reopening...
INFO : Session re-established.
INFO : Session re-established.
INFO : Status: Running (Executing on YARN cluster with App id application_1587628367568_0006)

INFO : Completed executing command(queryId=omm_20200424173329_53ad05b4-e097-44c4-9a8f-9f77e7087888); Time taken: 22.54 seconds
INFO : OK
INFO : Concurrency mode is disabled, not creating a lock manager
+-----+
|   _c0   |
+-----+
| 43204059 |
+-----+
1 row selected (29.898 seconds)
0: jdbc:hive2://192.168.0.186:2181,192.168.0.>
```

**步骤13** (可选) 如果源端集群中有新增数据需要定期将新增数据迁移至目的端集群, 则根据数据新增方式进行不同方式的迁移。配置定期任务增量迁移数据, 直到所有业务迁移至目的端集群。

- Hive表数据修改、未新增删除表、未修改已有表的数据结构: 此时Hive表已经创建好, 仅需迁移Hive存储在HDFS或OBS上的文件即可, 请参考[使用CDM服务迁移Hadoop数据至MRS集群](#)页面新增数据迁移方式进行数据迁移。
- Hive表有新增: 请选择“作业管理”的“表/文件迁移”页签, 在Hive迁移作业的“操作”列单击“编辑”, 选择新增的数据表进行数据迁移。
- Hive表有删除或已有表的数据结构有修改: 请在目的端集群中手动删除对应表或手动更新变更的表结构。

----结束

## 3.7 使用 BulkLoad 工具向 HBase 中批量导入数据

### 应用场景

经常面临向HBase中导入大量数据的情景, 向HBase中批量加载数据的方式有很多种, 最直接方式是调用HBase的API使用put方法插入数据; 另外一种是用MapReduce的方式从HDFS上加载数据。但是这两种方式效率都不是很高, 因为HBase频繁进行flush、

compact、split操作需要消耗较大的CPU和网络资源，并且RegionServer压力也比较大。

本实践基于华为云MapReduce服务，用于指导您创建MRS集群后，使用BulkLoad方式向HBase中批量导入本地数据，在首次数据加载时，能极大的提高写入效率，并降低对Region Server节点的写入压力。

## 方案架构

Bulkload是HBase提供的一个数据导入工具，它可以将数据批量地导入到HBase表中，直接写入底层数据文件和WAL日志，从而显著提升了数据加载的速度和效率。

BulkLoad方式调用MapReduce的job直接将数据输出成HBase table内部的存储格式的文件HFile，然后将生成的StoreFiles加载到集群的相应节点。这种方式无需进行flush、compact、split等过程，不占用Region资源，不会产生巨量的写入I/O，所以需要较少的CPU和网络资源。

BulkLoad适合的场景：

- 大量数据一次性加载到HBase。
- 对数据加载到HBase可靠性要求不高，不需要生成WAL文件。
- 使用put加载大量数据到HBase速度变慢，且查询速度变慢时。
- 加载到HBase新生成的单个HFile文件大小接近HDFS block大小。

## 步骤 1：创建 MRS 离线查询集群

1. 进入[购买MRS集群页面](#)。
2. 选择“快速购买”，填写配置参数。

表 3-8 软件配置

参数项	参数说明	取值
区域	不同区域的云服务产品之间内网互不相通。请就近选择靠近您业务的区域，可减少网络时延，提高访问速度。	华北-北京四
计费模式	MRS提供两种计费模式： <ul style="list-style-type: none"><li>● 包年/包月</li><li>● 按需计费</li></ul> 选择“按需计费”时，可能需要冻结一定的保证金。详细内容，请参见 <a href="#">计费说明</a> 。	按需计费
集群名称	集群名称不允许重复。只能由字母、数字、中划线和下划线组成，并且长度为1~64个字符。	MRS_hbase



参数项	参数说明	取值
集群类型	提供几种集群类型： <ul style="list-style-type: none"><li>• 分析集群</li><li>• 流式集群</li><li>• 混合集群</li><li>• 自定义</li></ul>	自定义
版本类型	提供以下版本类型： <ul style="list-style-type: none"><li>• 普通版</li><li>• LTS版</li></ul>	普通版
集群版本	MRS目前支持的版本。	MRS 3.1.0
组件选择	MRS集群配套的组件，MRS不同版本集群支持的组件版本请参见 <a href="#">MRS组件版本一览表</a> 。	HBase查询集群
可用区	可用区是使用独立电源和网络资源的物理区域。通过内部网络互联，再以物理方式进行隔离，提高了应用程序的可用性。建议您在不同的可用区下创建集群。	可用区1
企业项目	企业项目所在的企业资源管理控制台以面向企业资源管理为出发点，帮助企业以公司、部门、项目等分级管理方式实现企业云上的人员、资源、权限、财务的管理。	default
虚拟私有云	VPC即虚拟私有云，是通过逻辑方式进行网络隔离，提供安全、隔离的网络环境。	vpc-01
子网	通过子网提供与其他网络隔离的、可以独享的网络资源，以提高网络安全。	subnet-01
Kerberos认证	当集群开启Kerberos认证时，需要确认是否需要开启Kerberos认证，若确认开启请单击“继续”，若无需开启Kerberos认证请单击“返回”关闭Kerberos认证后再创建集群。购买集群后，不支持修改。	开启

参数项	参数说明	取值
用户名	默认为“root/admin”，root用于远程登录ECS机器，admin用于登录集群管理页面。	root/admin
密码	设置root用户和admin用户密码。	设置密码登录集群管理页面及ECS节点用户的密码，例如：Test!@12345。
确认密码	-	再次输入设置用户密码
通信安全授权	MRS集群通过管理控制台为用户发放、管理和使用大数据组件，大数据组件部署在用户的VPC内部，MRS管理控制台需要直接访问部署在用户VPC内的大数据组件时需要开通相应的安全组规则，而开通相应的安全组规则需要获取用户授权，此授权过程称为通信安全授权。  若不开启通信安全授权，MRS将无法创建集群。	勾选“确认授权”

图 3-24 创建 HBase 查询集群

3. 单击“立即购买”，等待MRS集群创建成功。

名称/ID	集群版本	集群类型	节点数	状态
MRS_demo 42184f5e-ab21-4377-a258-e1d0f58a0b54	MRS 3.1.0	分析集群	5	运行中

## 步骤 2：将本地数据导入到 HDFS 中

1. 在本地准备一个学生信息文件“info.txt”，例如内容如下：  
字段信息依次为：学号、姓名、生日、性别、住址  
20200101245,张xx,20150324,男,City1  
20200101246,李xx,20150202,男,City2  
20200101247,杨xx,20151101,女,City3  
20200101248,陈xx,20150218,男,City4  
20200101249,李xx,20150801,女,City5  
20200101250,王xx,20150315,男,City6  
20200101251,李xx,20151201,男,City7  
20200101252,孙xx,20150916,女,City8  
20200101253,林xx,20150303,男,City9
2. 登录对象存储服务OBS控制台，单击“并行文件系统 > 创建并行文件系统”，填写以下参数，单击“立即创建”。

表 3-9 并行文件系统参数

参数项	取值
区域	华北-北京四
文件系统名称	mrs-hbase
数据冗余存储策略	单AZ存储
策略	私有
归档数据直读	关闭
企业项目	default
标签	-

等待桶创建好，单击桶名称，选择“文件 > 上传文件”，将数据文件上传至OBS并行文件系统内。

3. 切换回MRS控制台，单击创建好的MRS集群名称，进入“概览”，单击“IAM用户同步”所在行的“单击同步”，等待约5分钟同步完成。
4. 将数据文件上传HDFS。
  - a. 在“文件管理”页签，选择“HDFS文件列表”，进入数据存储目录，如“/tmp/test”。  
“/tmp/test”目录仅为示例，可以是界面上的任何目录，也可以通过“新建”创建新的文件夹。
  - b. 单击“导入数据”。
    - OBS路径：选择上面创建好的OBS并行文件系统名，找到info.txt文件，单击“是”。
    - HDFS路径：选择HDFS路径，例如“/tmp/test”，单击“是”。
  - c. 单击“确定”，等待导入成功，此时数据文件已上传至HDFS。

图 3-25 导入数据



### 步骤 3: 创建 HBase 表

1. 登录集群的FusionInsight Manager页面（如果没有弹性IP，需提前购买弹性IP），新建一个用户hbasestest，绑定用户组supergroup，绑定角色System\_administrator。

用户名	用户类型	描述	创建时间
hbasestest	人机		2021/05/14 11:04:31 GMT+08:00

用户名: hbasestest  
用户类型: 人机  
主组: compcommon  
创建时间: 2021/05/14 11:04:31 GMT+08:00

用户组: supergroup  
角色: System\_administrator  
描述:

2. 下载并安装集群全量客户端，例如在主Master节点上安装，客户端安装目录为“/opt/client”，相关操作可参考[安装客户端](#)。  
也可直接使用Master节点中自带的集群客户端，安装目录为“/opt/Bigdata/client”。
3. 为主Master节点绑定一个弹性IP，然后使用root用户登录主Master节点，并进入客户端所在目录并认证用户。

```
cd /opt/client
source bigdata_env
kinit hbasestest
```

4. 执行**hbase shell**进入HBase Shell命令行界面。  
需要根据导入数据，规划HBase数据表的表名、rowkey、列族、列，考虑好rowkey分配在创建表时进行预分割。

执行以下命令创建表“student\_info”。

```
create 'student_info', {NAME => 'base', COMPRESSION => 'SNAPPY',
DATA_BLOCK_ENCODING => 'FAST_DIFF'}, SPLITS =>
['1','2','3','4','5','6','7','8']
```

- NAME => 'base': HBase表列族名称。
- COMPRESSION: 压缩方式
- DATA\_BLOCK\_ENCODING: 编码算法
- SPLITS: 预分region

5. 执行如下命令，查看表是否创建成功，然后退出HBase Shell命令行界面。

## list

## 步骤 4：生成 HFile 文件并导入 HBase

1. 创建自定义导入的模板文件，例如模板文件为 “/opt/configuration\_index.xml”（模板文件样例可从 “客户端安装目录/HBase/hbase/conf/index\_import.xml.template” 获取）。

**vi /opt/configuration\_index.xml**

例如本案例中，模板文件如下：

```
<?xml version="1.0" encoding="UTF-8"?>
<configuration>
<!--column_num要和数据文件中的列的数量对应：5列 -->
<import column_num="5" id="first">
  <columns>
    <column type="string" index="1">P_ID</column>
    <column type="string" index="2">P_NAME</column>
    <column type="string" index="3">P_BIRTH</column>
    <column type="string" index="4">P_GENDER</column>
    <column type="string" index="5">P_DISTRICT</column>
  </columns>
  <!--reverse(P_BIRTH)：反转出生年月避免热点 -->
  <!--substring(P_NAME,0,1)：截取姓 -->
  <!--substring(P_ID,0,6)：截身学号前六位 -->
  <rowkey>
    reverse(P_BIRTH)+'_'+substring(P_NAME,0,1)+'_'+substring(P_ID,0,6)
  </rowkey>
  <qualifiers>
  <!--family的指定要和表的列族名称对应。 -->
  <normal family="base">
    <qualifier column="P_ID">H_ID</qualifier>
    <qualifier column="P_NAME">H_NAME</qualifier>
    <qualifier column="P_BIRTH">H_BIRTH</qualifier>
    <qualifier column="P_GENDER">H_GENDER</qualifier>
    <qualifier column="P_DISTRICT">H_DISTRICT</qualifier>
  </normal>
  </qualifiers>
</import>
</configuration>
```

2. 执行如下命令，生成HFile文件。

```
hbase com.huawei.hadoop.hbase.tools.bulkload.ImportData -  
Dimport.separator=',' -Dimport.hfile.output=/tmp/test/hfile /opt/  
configuration_index.xml student_info /tmp/test/info.txt
```

- Dimport.separator：分隔符。
  - Dimport.hfile.output：执行结果输出路径。
  - /opt/configuration\_index.xml：指向自定义的模板文件。
  - student\_info：要操作的HBase表名。
  - /tmp/test/info.txt：指的是要批量上传的HDFS数据目录。
  - com.huawei.hadoop.hbase.tools.bulkload.IndexImportData：导入时创建二级索引使用IndexImportData；如果不创建二级索引，使用ImportData
- 等待MapReduce任务执行成功，执行如下命令，输出路径下生成HFile文件。

```
hdfs dfs -ls /tmp/test/hfile
```

```
Found 2 items
-rw-r--r-- 3 hbasetest hadoop 0 2021-05-14 11:39 /tmp/test/hfile/_SUCCESS
drwxr-xr-x - hbasetest hadoop 0 2021-05-14 11:39 /tmp/test/hfile/base
```

3. 执行如下命令将HFile导入HBase表。

```
hbase org.apache.hadoop.hbase.mapreduce.LoadIncrementalHFiles /tmp/  
test/hfile student_info
```

4. 进入HBase Shell命令行界面，查看表内容。

```
hbase shell
```

```
scan 'student_info', {FORMATTER => 'toString'}
```

ROW	COLUMN+CELL
10115102_杨_202001 value=20151101	column=base:H_BIRTH, timestamp=2021-05-14T15:28:56.755,
10115102_杨_202001 value=City3	column=base:H_DISTRICT, timestamp=2021-05-14T15:28:56.755,
10115102_杨_202001 value=女	column=base:H_GENDER, timestamp=2021-05-14T15:28:56.755,
10115102_杨_202001 value=20200101247	column=base:H_ID, timestamp=2021-05-14T15:28:56.755,
10115102_杨_202001 value=杨xx	column=base:H_NAME, timestamp=2021-05-14T15:28:56.755,
10215102_李_202001 value=20151201	column=base:H_BIRTH, timestamp=2021-05-14T15:28:56.755,
10215102_李_202001 value=City7	column=base:H_DISTRICT, timestamp=2021-05-14T15:28:56.755,
...	

5. 数据导入集群后，就可以继续基于大数据平台上层应用对数据进行分析处理了。

## 3.8 使用 CDM 服务迁移 MySQL 数据至 MRS Hive

### 应用场景

MapReduce服务（MapReduce Service，简称MRS）提供企业级大数据集群云服务，里面包含HDFS、Hive、Spark等组件，适用于企业海量数据分析。

其中Hive提供类SQL查询语言，帮助用户对大规模的数据进行提取、转换和加载，即通常所称的ETL（Extraction, Transformation, and Loading）操作。对庞大的数据集查询需要耗费大量的时间去处理，在许多场景下，可以通过建立Hive分区方法减少每一次扫描的总数据量，这种做法可以显著地改善性能。

Hive的分区使用HDFS的子目录功能实现，每一个子目录包含了分区对应的列名和每一列的值。当分区很多时，会有很多HDFS子目录，如果不依赖工具，将外部数据加载到Hive表各分区不是一件容易的事情。

云数据迁移服务（CDM）可以轻松将外部数据源（关系数据库、对象存储服务、文件系统服务等）加载到Hive分区表。

本实践为您演示使用CDM云服务将MySQL数据导入到MRS集群内的Hive分区表中。

假设MySQL数据库中有一张表“trip\_data”，保存了自行车骑行记录，里面有起始时间、结束时间，起始站点、结束站点、骑手ID等信息。

“trip\_data”表字段定义如[图3-26](#)所示。

图 3-26 MySQL 表字段

Column Name	#	Data Type
TripID	1	int(11)
Duration	2	int(11)
StartDate	3	timestamp
StartStation	4	varchar(64)
StartTerminal	5	int(11)
EndDate	6	timestamp
EndStation	7	varchar(64)
EndTerminal	8	int(11)
Bike	9	int(11)
SubscriberType	10	varchar(32)
ZipCodev	11	varchar(10)

使用CDM将MySQL中的数据导入到MRS Hive分区表，流程如下：

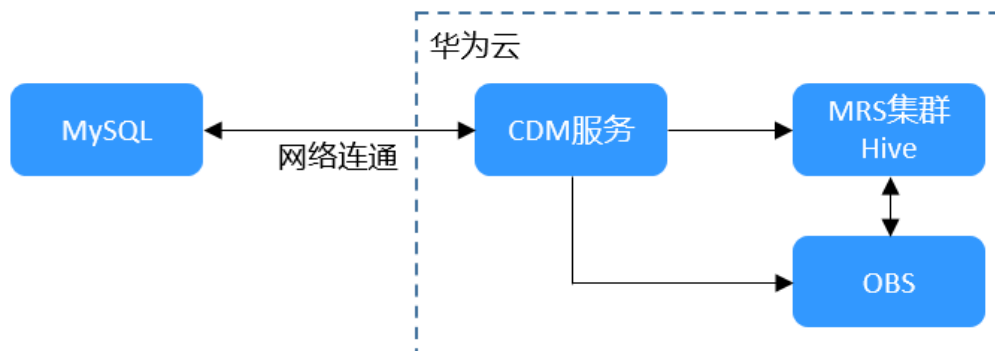
1. **步骤1：在MRS Hive上创建Hive分区表**
2. **步骤2：创建CDM集群并绑定EIP**
3. **步骤3：创建MySQL连接**
4. **步骤4：创建Hive连接**
5. **步骤5：创建迁移作业**

## 方案架构

CDM围绕大数据迁移上云和智能数据湖解决方案，提供了简单易用的迁移能力和多种数据源到数据湖的集成能力，降低了客户数据源迁移和集成的复杂性，有效的提高您数据迁移和集成的效率。

CDM服务迁移MySQL数据至MRS集群方案如图3-27所示。

图 3-27 MySQL 数据迁移示意



## 前提条件

- 已经购买包含有Hive服务的MRS集群。

- 已获取连接MySQL数据库的IP地址、端口、数据库名称、用户名、密码，且该用户拥有MySQL数据库的读写权限。
- 已参考[管理驱动](#)，上传了MySQL数据库驱动。

## 步骤 1：在 MRS Hive 上创建 Hive 分区表

在MRS的Hive客户端中，执行以下SQL语句创建一张Hive分区表，表名与MySQL上的表trip\_data一致，且Hive表比MySQL表多建三个字段y、ym、ymd，作为Hive的分区字段。

SQL语句如下：

```
create table trip_data(TripID int,Duration int,StartDate timestamp,StartStation varchar(64),StartTerminal int,EndDate timestamp,EndStation varchar(64),EndTerminal int,Bike int,SubscriberType varchar(32),ZipCodev varchar(10))partitioned by (y int,ym int,ymd int);
```

### 说明

Hive表trip\_data有三个分区字段：骑行起始时间的年、骑行起始时间的年月、骑行起始时间的年月日。

例如一条骑行记录的起始时间为2018/5/11 9:40，那么这条记录会保存在分区trip\_data/2018/201805/20180511下面。

对trip\_data进行按时间维度统计汇总时，只需要对局部数据扫描，大大提升性能。

## 步骤 2：创建 CDM 集群并绑定 EIP

**步骤1** 如果是独立CDM服务，参考[创建集群](#)创建CDM集群；如果是作为DataArts Studio服务CDM组件使用，参考[创建集群](#)创建CDM集群。

关键配置如下：

- CDM集群的规格，按待迁移的数据量选择，一般选择cdm.medium即可，满足大部分迁移场景。
- CDM集群所在VPC、子网、安全组，选择与MRS集群所在的网络一致。

**步骤2** CDM集群创建完成后，选择集群操作列的“绑定弹性IP”，CDM通过EIP访问MySQL。

图 3-28 集群列表



集群名称	集群状态	内网地址	公网地址	创建来源	企业项目	操作
	不可用			CDM	default	作业管理   绑定弹性IP   更多
	运行中			CDM	default	作业管理   绑定弹性IP   更多

### 说明

如果用户对本地数据源的访问通道做了SSL加密，则CDM无法通过弹性IP连接数据源。

----结束

## 步骤 3：创建 MySQL 连接

**步骤1** 在CDM集群管理界面，单击集群后的“作业管理”，选择“连接管理 > 新建连接”，进入连接器类型的选择界面。



图 3-29 选择连接器类型



**步骤2** 选择“MySQL”后单击“下一步”，配置MySQL连接的参数。

单击“显示高级属性”可查看更多可选参数，具体请参见[配置关系数据库连接](#)。此处保持默认，必填参数如[表3-10](#)所示。

表 3-10 MySQL 连接参数

参数名	说明	取值样例
名称	输入便于记忆和区分的连接名称。	mysqllink
数据库服务器	MySQL数据库的IP地址或域名。	192.168.1.110
端口	MySQL数据库的端口。	3306
数据库名称	MySQL数据库的名称。	sqoop
用户名	拥有MySQL数据库的读、写和删除权限的用户。	admin
密码	用户的密码。	-
使用Agent	是否选择通过Agent从源端提取数据。	是
Agent	单击“选择”，选择已创建的Agent。	-

图 3-30 创建 MySQL 连接

* 名称	<input type="text" value="mysqllink"/>
* 连接器	关系数据库
数据库类型	MySQL
* 数据库服务器 ?	<input type="text" value="192.168.1.110"/>
* 端口 ?	<input type="text" value="3306"/>
* 数据库名称 ?	<input type="text" value="sqoop"/>
* 用户名 ?	<input type="text" value="admin"/>
* 密码 ?	<input type="password" value="....."/>
使用本地API ?	<input type="radio"/> 是 <input checked="" type="radio"/> 否
使用Agent ?	<input checked="" type="radio"/> 是 <input type="radio"/> 否
Agent ?	<input type="text" value="agent1"/> <a href="#">送</a>

[显示高级属性](#)

**步骤3** 单击“保存”回到连接管理界面。

#### 📖 说明

如果保存时出错，一般是由于MySQL数据库的安全设置问题，需要设置允许CDM集群的EIP访问MySQL数据库。

---结束

## 步骤 4: 创建 Hive 连接

**步骤1** 在连接管理界面，单击“新建连接”，连接器类型选择“MRS Hive”。

步骤2 单击“下一步”配置Hive连接参数，如图3-31所示。

图 3-31 创建 Hive 连接

The screenshot shows a configuration form for creating a Hive connection. The fields and their values are as follows:

- \* 名称**: (Empty text input)
- \* 连接器**: Hive (Dropdown menu)
- \* Hadoop类型**: MRS (Dropdown menu)
- \* Manager IP**: (Empty text input) with a blue "选择" button to the right.
- 认证类型**: SIMPLE (Dropdown menu)
- \* Hive版本**: HIVE\_3\_X (Dropdown menu)
- \* 用户名**: (Empty text input)
- \* 密码**: (Masked password input)
- \* OBS支持**: 否 (Radio buttons, "否" is selected)
- \* 运行模式**: EMBEDDED (Dropdown menu)

Below the form is a link "显示高级属性". At the bottom, there are four buttons: "取消" (Cancel), "上一步" (Previous Step), "测试" (Test), and "保存" (Save).

各参数说明如表3-11所示，需要您根据实际情况配置。

表 3-11 MRS Hive 连接参数

参数名	说明	取值样例
名称	连接的名称，根据连接的数据源类型，用户可自定义便于记忆、区分的连接名。	hivelink
Manager IP	MRS Manager的浮动IP地址，可以单击输入框后的“选择”来选定已创建的MRS集群，CDM会自动填充下面的鉴权参数。	127.0.0.1

参数名	说明	取值样例
认证类型	访问MRS的认证类型： <ul style="list-style-type: none"> <li>• SIMPLE：非安全模式选择Simple鉴权。</li> <li>• KERBEROS：安全模式选择Kerberos鉴权。</li> </ul>	SIMPLE
Hive版本	Hive的版本。根据服务端Hive版本设置。	HIVE_3_X
用户名	选择KERBEROS鉴权时，需要配置MRS Manager的用户名和密码。从HDFS导出目录时，如果需要创建快照，这里配置的用户需要HDFS系统的管理员权限。 如果要创建MRS安全集群的数据连接，不能使用admin用户。因为admin用户是默认的管理页面用户，这个用户无法作为安全集群的认证用户来使用。您可以创建一个新的MRS用户，然后在创建MRS数据连接时，“用户名”和“密码”填写为新建的MRS用户及其密码。 <b>说明</b> <ul style="list-style-type: none"> <li>• 如果CDM集群为2.9.0版本及之后版本，且MRS集群为3.1.0及之后版本，则所创建的用户至少需具备Manager_viewer的角色权限才能在CDM创建连接；如果需要对应组件的进行库、表、数据的操作，还需要添加对应组件的用户组权限。</li> <li>• 如果CDM集群为2.9.0之前的版本，或MRS集群为3.1.0之前的版本，则所创建的用户需要具备Manager_administrator或System_administrator权限，才能在CDM创建连接。</li> <li>• 仅具备Manager_tenant或Manager_auditor权限，无法创建连接。</li> </ul>	cdm
密码	访问MRS Manager的用户密码。	-
OBS支持	需服务端支持OBS存储。在创建Hive表时，您可以指定将表存储在OBS中。	否
运行模式	“HIVE_3_X”版本支持该参数。支持以下模式： <ul style="list-style-type: none"> <li>• EMBEDDED：连接实例与CDM运行在一起，该模式性能较好。</li> <li>• STANDALONE：连接实例运行在独立进程。如果CDM需要对接多个Hadoop数据源（MRS、Hadoop或CloudTable），并且既有KERBEROS认证模式又有SIMPLE认证模式，只能使用STANDALONE模式或者配置不同的Agent。  <b>说明：</b>STANDALONE模式主要是用来解决版本冲突问题的运行模式。当同一种数据连接的源端或者目的端连接器的版本不一致时，存在jar包冲突的情况，这时需要将源端或目的端放在STANDALONE进程里，防止冲突导致迁移失败。</li> </ul>	EMBEDDED
是否使用集群配置	您可以通过使用集群配置，简化Hive连接参数配置。	否

参数名	说明	取值样例
集群配置名	仅当“是否使用集群配置”为“是”时，此参数有效。此参数用于选择用户已经创建好的集群配置。集群配置的创建方法请参见 <a href="#">管理集群配置</a> 。	hive_01

**步骤3** 单击“保存”回到连接管理界面。

----结束

## 步骤 5: 创建迁移作业

**步骤1** 在CDM集群管理界面，单击集群后的“作业管理”，选择“表/文件迁移 > 新建作业”，开始创建数据迁移任务，如[图3-32](#)所示。

**图 3-32** 创建 MySQL 到 Hive 的迁移任务

**作业配置**

\* 作业名称

---

**源端作业配置**

\* 源连接名称  新建连接

\* 模式或表空间  ...

\* 表名  ...

[显示高级属性](#)

**目的端作业配置**

\* 目的连接名称  新建连接

\* 数据库名称  ...

\* 表名  ...

\* 自动创表

导入前清空数据  是  否

取消
下一步

### 说明


“导入前清空数据”选“是”，这样每次导入前，会将之前已经导入到Hive表的数据清空。

**步骤2** 作业参数配置完成后，单击“下一步”，进入字段映射界面，如[图3-33](#)所示。

映射MySQL表和Hive表字段，Hive表比MySQL表多三个字段y、ym、ymd，即是Hive的分区字段。由于没有源表字段直接对应，需要配置表达式从源表的StartDate字段抽取。

图 3-33 Hive 字段映射



**步骤3** 单击  进入转换器列表界面，再选择“新建转换器 > 表达式转换”。

y、ym、ymd字段的表达式分别配置如下：

```
DateUtils.format(DateUtils.parseDate(row[2],"yyyy-MM-dd HH:mm:ss.SSS"),"yyyy")
```

```
DateUtils.format(DateUtils.parseDate(row[2],"yyyy-MM-dd HH:mm:ss.SSS"),"yyyyMM")
```

```
DateUtils.format(DateUtils.parseDate(row[2],"yyyy-MM-dd HH:mm:ss.SSS"),"yyyyMMdd")
```

#### 📖 说明

CDM的表达式已经预置常用字符串、日期、数值等类型的字段内容转换，详细请参见[字段转换](#)。

**步骤4** 单击“下一步”配置任务参数，一般情况下全部保持默认即可。

该步骤用户可以配置如下可选功能：

- 作业失败重试：如果作业执行失败，可选择是否自动重试，这里保持默认值“不重试”。
- 作业分组：选择作业所属的分组，默认分组为“DEFAULT”。在CDM“作业管理”界面，支持作业分组显示、按组批量启动作业、按分组导出作业等操作。
- 是否定时执行：如果需要配置作业定时自动执行，请参见[配置定时任务](#)。这里保持默认值“否”。
- 抽取并发数：设置同时执行的抽取任务数。这里保持默认值“1”。
- 是否写入脏数据：如果需要将作业执行过程中处理失败的数据、或者被清洗过滤掉的数据写入OBS中，以便后面查看，可通过该参数配置，写入脏数据前需要先配置好OBS连接。这里保持默认值“否”即可，不记录脏数据。
- 作业运行完是否删除：这里保持默认值“不删除”。

**步骤5** 单击“保存并运行”，回到作业管理界面，在作业管理界面可查看作业执行进度和结果。

**步骤6** 作业执行成功后，单击作业操作列的“历史记录”，可查看该作业的历史执行记录、读取和写入的统计数据。

在历史记录界面单击“日志”，可查看作业的日志信息。

----结束

## 3.9 使用 CDM 服务迁移 MRS HDFS 数据至 OBS

### 应用场景

MRS支持在大数据存储容量大、计算资源需要弹性扩展的场景下，用户将数据存储到OBS服务中，使用MRS集群仅作数据计算处理的存算分离模式，从而实现按需灵活扩展资源、低成本的海量数据分析方案。

CDM支持文件到文件类数据的迁移，本章节以MRS HDFS至OBS为例，介绍如何通过CDM将文件类数据迁移到OBS文件系统中。

使用CDM服务迁移MRS HDFS数据至OBS流程如下：

1. [步骤1：创建CDM集群并绑定EIP](#)
2. [步骤2：创建MRS HDFS连接](#)
3. [步骤3：创建OBS连接](#)
4. [步骤4：创建迁移作业](#)

### 方案架构

CDM围绕大数据迁移上云和智能数据湖解决方案，提供了简单易用的迁移能力和多种数据源到数据湖的集成能力，降低了客户数据源迁移和集成的复杂性，有效的提高您数据迁移和集成的效率。

CDM服务迁移HDFS数据至MRS集群方案如[图3-34](#)所示。

图 3-34 HDFS 数据迁移示意



### 前提条件

- 已获取OBS的访问域名、端口，以及AK、SK信息。
- 已经创建包含有Hadoop服务的MRS集群。
- 拥有EIP配额并创建EIP。

### 步骤 1：创建 CDM 集群并绑定 EIP

**步骤1** 如果是独立CDM服务，参考[创建集群](#)创建CDM集群；如果是作为DataArts Studio服务CDM组件使用，参考[创建集群](#)创建CDM集群。

关键配置如下：

- CDM集群的规格，按待迁移的数据量选择，一般选择“cdm.medium”即可，满足大部分迁移场景。
- CDM集群所在VPC、子网、安全组，选择与MRS集群所在的网络一致。

**步骤2** CDM集群创建完成后，选择集群操作列的“绑定弹性IP”，CDM通过EIP访问MRS HDFS。

图 3-35 集群列表

集群名称	集群状态	内网地址	公网地址	创建来源	企业项目	操作
...	不可用	...	...	CDM	default	作业管理 绑定弹性IP 更多
...	运行中	...	-	CDM	default	作业管理 绑定弹性IP 更多

### 说明

如果用户对本地数据源的访问通道做了SSL加密，则CDM无法通过弹性IP连接数据源。

----结束

## 步骤 2: 创建 MRS HDFS 连接

**步骤1** 在CDM集群管理界面，单击集群后的“作业管理”，选择“连接管理 > 新建连接”，连接器类型选择“MRS HDFS”后，单击“下一步”，配置MRS HDFS链接参数。

- 名称：用户自定义连接名称，例如“mrs\_hdfs\_link”。
- Manage IP：MRS Manager的IP地址，可以单击输入框后的“选择”来选定已创建的MRS集群，CDM会自动填充下面的鉴权参数。
- 用户名：选择KERBEROS鉴权时，需要配置MRS Manager中创建的用户。  
从HDFS导出目录时，如果需要创建快照，这里配置的用户需要具有HDFS系统的管理员权限。
- 密码：访问MRS Manager的用户密码。
- 认证类型：访问MRS的认证类型。
- 运行模式：选择HDFS连接的运行模式。

----结束

## 步骤 3: 创建 OBS 连接

**步骤1** 在CDM集群管理界面，单击集群后的“作业管理”，选择“连接管理 > 新建连接”，连接器类型选择“对象存储服务（OBS）”后，单击“下一步”配置OBS连接参数，如图3-36所示。

- 名称：用户自定义连接名称，例如“obslink”。
- OBS终端节点、端口：配置为OBS实际的地址信息。
- OBS类型：保持默认。
- 访问标识（AK）、密钥（SK）：登录OBS的AK、SK信息。



图 3-36 创建 OBS 连接

\* 名称: obslink

\* 连接器: OBS

对象存储类型: 对象存储OBS

\* OBS终端节点 (?): [redacted]

\* 端口 (?): 443

\* OBS桶类型 (?): 对象存储

\* 访问标识(AK) (?): [redacted]

\* 密钥(SK) (?): [redacted]

取消 | < 上一步 | 测试 | 保存

步骤2 单击“保存”回到连接管理界面。

----结束

### 步骤 4: 创建迁移作业

步骤1 在CDM集群管理界面，单击集群后的“作业管理”，选择“表/文件迁移 > 新建作业”，开始创建从MRS HDFS导出数据到OBS的任务。

图 3-37 创建 MRS HDFS 到 OBS 的迁移任务

1 基本信息配置 | 2 字段映射 | 3 任务配置

作业配置

\* 作业名称: hdfs2obs\_004more

源端作业配置

\* 源连接名称: hdfs\_link

\* 源目录或文件: /Interface/hdfsfrom/more1

\* 文件格式: CSV格式

显示高级属性

目的端作业配置

\* 目的连接名称: obs\_link

\* 桶名: cdm-autotest

\* 写入目录: /Interface/obsto/

\* 文件格式: CSV格式

覆盖文件处理方式: 替换覆盖文件

显示高级属性

取消 | > 下一步

- 作业名称：用户自定义便于记忆、区分的任务名称。
- 源端作业配置
  - 源连接名称：选择[步骤2：创建MRS HDFS连接](#)中的“hdfs\_link”。
  - 源目录或文件：待迁移数据的目录或单个文件路径。
  - 文件格式：传输数据时所用的文件格式，这里选择“二进制格式”。不解析文件内容直接传输，不要求文件格式必须为二进制。适用于文件到文件的原样复制。
  - 其他可选参数一般情况下保持默认即可，详细说明请参见[配置HDFS源端参数](#)。
- 目的端作业配置
  - 目的连接名称：选择[步骤3：创建OBS连接](#)中的“obs\_link”。
  - 桶名：待迁移数据的桶。
  - 写入目录：写入数据到OBS服务器的目录。
  - 文件格式：迁移文件类数据到文件时，文件格式选择“二进制格式”。
  - 高级属性里的可选参数一般情况下保持默认既可，详细说明请参见[配置OBS目的端参数](#)。

**步骤2** 单击“下一步”进入字段映射界面，CDM会自动匹配源和目的字段。

- 如果字段映射顺序不匹配，可通过拖拽字段调整。
- CDM的表达式已经预置常用字符串、日期、数值等类型的字段内容转换，详细请参见[字段转换](#)。

**步骤3** 单击“下一步”配置任务参数，一般情况下全部保持默认即可。

该步骤用户可以配置如下可选功能：

- 作业失败重试：如果作业执行失败，可选择是否自动重试，这里保持默认值“不重试”。
- 作业分组：选择作业所属的分组，默认分组为“DEFAULT”。在CDM“作业管理”界面，支持作业分组显示、按组批量启动作业、按分组导出作业等操作。
- 是否定时执行：如果需要配置作业定时自动执行，请参见[配置定时任务](#)。这里保持默认值“否”。
- 抽取并发数：设置同时执行的抽取任务数。CDM支持多个文件的并发抽取，调大参数有利于提高迁移效率
- 是否写入脏数据：否，文件到文件属于二进制迁移，不存在脏数据。
- 作业运行完是否删除：这里保持默认值“不删除”。根据使用场景，也可配置为“删除”，防止迁移作业堆积。

**步骤4** 单击“保存并运行”，回到作业管理界面，在作业管理界面可查看作业执行进度和结果。

**步骤5** 作业执行成功后，单击作业操作列的“历史记录”，可查看该作业的历史执行记录、读取和写入的统计数据。

在历史记录界面单击“日志”，可查看作业的日志信息。

----结束

# 4 周边云服务对接

## 4.1 MRS 对接 LakeFormation 云服务

### 4.1.1 LakeFormation 服务概述

该LakeFormation入门教程介绍了如何创建一个LakeFormation实例并与MRS集群对接，实现统一的数据湖元数据及权限管理。

#### 使用流程简介

MRS与LakeFormation对接的使用流程如下图所示：

图 4-1 LakeFormation 使用流程



## 约束说明

- **MRS对接LakeFormation前，需要注意以下约束限制：**
  - MRS集群和LakeFormation实例必须同在一个云账户下且属于同一个Region。
  - LakeFormation侧创建的接入客户端所在虚拟私有云，必须与MRS集群在同一虚拟私有云下。
  - MRS集群仅支持对接LakeFormation实例中名称为hive的Catalog。
  - MRS存量集群需要先完成元数据库和权限策略向LakeFormation实例上迁移，再配置对接。
  - 如果需要迁移多个MRS集群中的元数据到同一个LakeFormation实例，MRS集群之间的Database名称不能重复。
- **MRS对接LakeFormation后，MRS组件功能约束限制：**
  - Hive暂不支持临时表功能。
  - Hive暂不支持跨集群的列加密表功能。
  - Hive WebHCat暂不支持对接LakeFormation。
  - Hive创建内表时如果表目录不为空，则禁止创建表。
  - Hudi表创建前，需要先在LakeFormation上添加Hudi表目录的路径授权，赋予OBS读写权限。
  - Hudi表不支持在LakeFormation管理面编辑表的字段，只能通过Hudi客户端增删改表的字段。
  - Flink读写Hudi场景下同步Hive表，仅支持使用hive\_sync.mode=jdbc，不支持hms方式。
  - Spark使用小权限用户登录客户端创建数据库时，如果用户没有default库的OBS路径权限，将提示缺少权限，实际创建数据库成功。
- **MRS对接LakeFormation后，权限策略约束限制：**
  - 通过LakeFormation授权仅支持将LakeFormation角色作为授权主体，不支持IAM用户或IAM用户组作为授权主体。
  - PolicySync进程不会修改集群内RangerAdmin Hive模块的默认策略，默认策略仍然生效。
  - PolicySync进程启动后，会与LakeFormation实例的权限进行比对，删除LakeFormation上不存在的非默认策略，请先完成权限策略迁移到LakeFormation实例上。
  - RangerAdmin WebUI界面的Hive模块，禁止执行添加、删除权限非默认策略的操作，统一在LakeFormation实例的数据权限界面进行授权操作。
  - RangerAdmin WebUI界面的OBS模块，禁止执行添加、删除权限非默认策略的操作，统一在LakeFormation实例的数据权限界面进行OBS路径授权操作。
  - MRS集群取消对接LakeFormation后，RangerAdmin的非默认策略不会清理，需要人工进行清理。
  - Hive暂不支持Grant授权的SQL语句，需统一在LakeFormation实例的数据权限界面进行授权操作。
  - MRS暂不支持LakeFormation行过滤权限能力。

## 4.1.2 配置 LakeFormation 数据连接前准备

### 配置 LakeFormation 实例

- 步骤1** 登录华为云管理控制台，在左上角单击“☰”，选择“大数据 > 湖仓构建 LakeFormation”进入LakeFormation控制台。
- 步骤2** 单击页面右上角“购买实例”，参考[创建LakeFormation实例](#)创建LakeFormation实例。
- 步骤3** 创建名称为“hive”的Catalog、名称为“default”的数据库，如果实例中已存在则请跳过该步骤。详细操作可参考[管理元数据](#)。

MRS对接LakeFormation仅支持对接LakeFormation实例的数据目录名称为“hive”的Catalog。

1. 确认左上角实例是新创建的LakeFormation实例名称后，进入“元数据 > Catalog”页面。（如果当前实例已包含名称为“hive”的Catalog，则跳过Catalog的创建操作。）
2. 单击“创建Catalog”，配置以下参数后，单击“提交”。
  - Catalog名称：hive（**固定名称，不可自定义**）
  - 选择位置：单击“+”选择Catalog对应的OBS存储路径，例如选择“obs://lakeformation-test/hive”（需提前创建），单击“确定”。
  - 其他参数根据实际需要进行配置。

< 创建Catalog

基本信息

\* Catalog名称  选择位置  +

Catalog类型  描述

数据库存储位置

序号	数据库存储位置
----	---------

3. 在左侧导航栏选择“元数据 > 数据库”，单击“创建数据库”，配置以下信息并单击“提交”。（如果当前已包含名称为“default”的数据库，则跳过数据库的创建操作。）
  - 库名称：default（**固定名称，不可自定义**）
  - 所属Catalog：hive
  - 选择位置：单击“+”选择hive Catalog存储路径下的位置，例如“obs://lakeformation-test/hive/default”（需提前创建），单击“确定”。
  - 其他参数根据实际需要进行配置。

< | 创建数据库

基本信息

\* 库名称  \* 所属Catalog

\* 选择位置  + 描述

数据表存储位置

序号	数据表存储位置
----	---------

函数存储位置

序号	函数存储位置
----	--------

**步骤4** 在“数据权限 > 数据授权”页面，可根据业务需求对hive Catalog进行基于用户、用户组的授权。详细操作请参考[新增授权](#)章节。

**步骤5** 选择“接入管理 > 创建客户端”，创建LakeFormation实例接入管理客户端。其中“虚拟私有云”和“所属子网”需要与待对接的MRS集群保持一致。详细操作请参考[管理接入客户端](#)。

创建客户端

如果没有合适的虚拟私有云或子网，可以前往VPC创建。

\* 客户端名称

\* 虚拟私有云

\* 所属子网

**⚠** 创建前请确保当前账号有足够的VPC、DNS内网域名等资源配额，如果创建失败，客户端将自动回滚删除，并回收相关资源。

MRS集群的VPC子网信息可通过MRS管理控制台中，MRS集群的概览页面中获取。客户端创建完成后，在客户端详情信息中获取对应客户端的“接入IP”信息并记录。

----结束

## 创建对接 LakeFormation 权限的委托

**步骤1** 登录华为云管理控制台，选择“统一身份认证服务 IAM”。

**步骤2** 在左侧导航栏选择“委托”，单击右上角的“创建委托”，配置相关参数，单击“下一步”。

参数配置如下：

- 委托名称：例如“visit\_lakeformation\_agency”
- 委托类型：选择“普通账号”
- 委托的账号：输入被委托的华为云账号名称
- 持续时间：根据实际情况自定义

图 4-2 创建委托

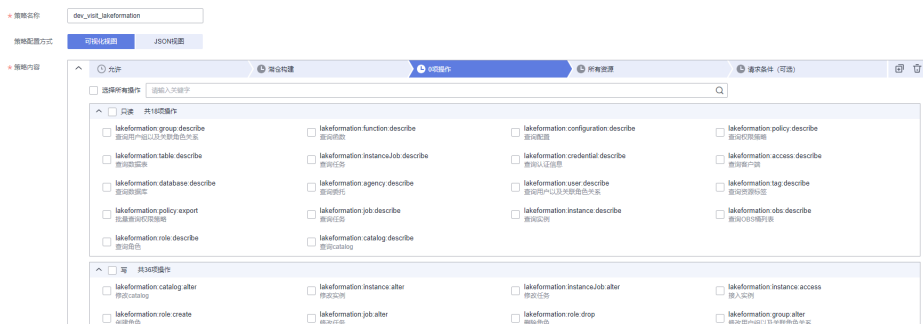
The screenshot shows the 'Create Delegation' (创建委托) form. It contains the following fields and options:

- \* 委托名称**: A text input field.
- \* 委托类型**: Two radio button options:
  - 普通帐号: 将帐号内资源的操作权限委托给其他华为云帐号。
  - 云服务: 将帐号内资源的操作权限委托给华为云服务。
- \* 委托的帐号**: A text input field.
- \* 持续时间**: A dropdown menu currently showing '永久' (Permanent).
- 描述**: A text area with the placeholder '请输入委托信息。' (Please enter delegation information.) and a character count '0/255'.

At the bottom, there are two buttons: a red '下一步' (Next Step) button and a grey '取消' (Cancel) button.

**步骤3** 在选择策略界面右上角单击“新建策略”，配置如下信息，单击“下一步”。

- 策略名称：例如“dev\_visit\_lakeformation”
- 策略配置方式：“可视化视图”或“JSON视图”
- 策略内容：
  - 策略中必须包含“lakeformation:policy:export”和“lakeformation:role:describe”。其他参数按照实际需求进行配置。
  - 可视化视图：“云服务”选择“湖仓构建”；“操作”中选择所需操作权限。其他参数按照实际需求进行配置。



- JSON视图，例如配置策略内容如下：

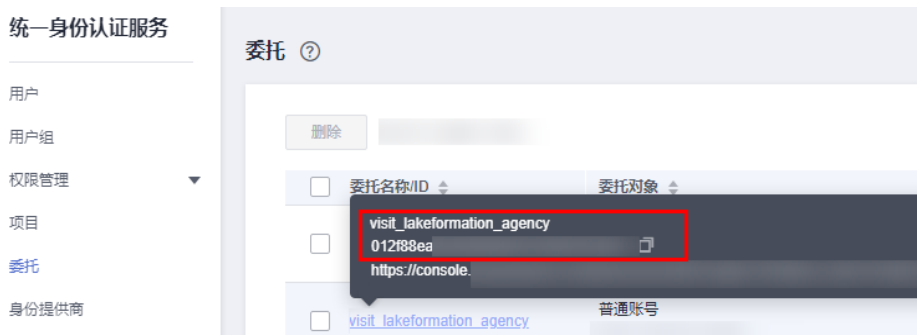
```

{
  "Version": "1.1",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "lakeformation:table:create",
        "lakeformation:database:alter",
        "lakeformation:table:alter",
        "lakeformation:database:drop",
        "lakeformation:database:create",
        "lakeformation:role:describe",
        "lakeformation:policy:create",
        "lakeformation:policy:export",
        "lakeformation:function:alter",
        "lakeformation:function:describe",
        "lakeformation:table:drop",
        "lakeformation:catalog:describe",
        "lakeformation:table:describe",
        "lakeformation:function:drop",
        "lakeformation:database:describe",
        "lakeformation:function:create",
        "lakeformation:transaction:operate"
      ]
    }
  ]
}
    
```

**步骤4** 勾选新建的策略名称例如“dev\_visit\_lakeformation”，单击“下一步”。

**步骤5** “设置最小授权范围”根据实际情况选择授权的资源范围，单击“确定”，创建委托。

**步骤6** 在“委托”页面，将鼠标移动到新创建的委托名称上，获取具备访问LakeFormation权限的委托ID。



----结束



## 创建对接 OBS 权限的委托

**步骤1** 登录华为云管理控制台，选择“统一身份认证服务”。

**步骤2** 在左侧导航栏选择“委托”，单击右上角的“创建委托”，选择相关参数，单击“下一步”。

参数选择如下：

- 委托名称：例如“visit\_obs\_agency”
- 委托类型：选择“普通账号”
- 委托的账号：输入被委托的华为云账号名称
- 持续时间：根据实际情况自定义

**步骤3** 在选择策略界面右上角单击“新建策略”，配置如下信息，单击“下一步”。

- 策略名称：例如“dev\_visit\_obs”
- 策略配置方式：JSON视图
- 策略内容：填入如下信息。

```
{
  "Version": "1.1",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "obs:bucket:GetBucketLocation",
        "obs:bucket:ListBucketMultipartUploads",
        "obs:object:GetObject",
        "obs:object:ModifyObjectMetaData",
        "obs:object:DeleteObject",
        "obs:object:ListMultipartUploadParts",
        "obs:bucket:HeadBucket",
        "obs:object:AbortMultipartUpload",
        "obs:bucket:ListBucket",
        "obs:object:PutObject"
      ],
      "Resource": [
        "OBS:*:*:bucket:*",
        "OBS:*:*:object:*"
      ]
    }
  ]
}
```

### 说明

Resource参数中“bucket”的参数值表示OBS桶名称，“object”的参数值表示OBS对象名称，可根据需要指定名称。配置为“\*”表示对所有OBS桶或OBS对象适用此策略。

- 其他参数按照实际需求进行配置。

**步骤4** 勾选新建的策略名称例如“dev\_visit\_obs”，单击“下一步”。

**步骤5** “设置最小授权范围”根据实际情况选择授权的资源范围，单击“确定”，创建委托。

**步骤6** 在“委托”页面，将鼠标移动到新创建的委托名称上，获取具备访问OBS权限的委托ID。

----结束

## 创建对接 ECS/BMS 云服务委托

**步骤1** 登录华为云管理控制台，选择“统一身份认证服务”。

**步骤2** 在左侧导航栏选择“委托”，单击右上角的“创建委托”，设置相关参数，单击“下一步”。

参数选择如下：

- 委托名称：例如“lakeformation\_test”
- 委托类型：选择“云服务”
- 云服务：选择“ECS BMS”
- 持续时间：根据实际情况自定义

**步骤3** 在选择策略界面右上角单击“新建策略”，配置如下信息，单击“下一步”。

- 策略名称：自定义
- 策略配置方式：选择JSON视图
- 策略内容：配置如下信息

```
{
  "Version": "1.1",
  "Statement": [
    {
      "Action": [
        "iam:agencies:assume"
      ],
      "Resource": {
        "uri": [
          "/iam/agencies/授予给自身账号具备访问LakeFormation权限的委托ID",
          "/iam/agencies/授予给自身账号具备访问OBS权限的委托ID"
        ]
      },
      "Effect": "Allow"
    }
  ]
}
```

### 📖 说明

- 授予给自身账号具备访问LakeFormation权限的委托ID：可参考[步骤6](#)获取。
- 授予给自身账号具备访问OBS权限的委托ID：可参考[步骤6](#)获取。

**步骤4** 选择新创建的自定义委托名称，单击“下一步”。

**步骤5** “设置最小授权范围”根据实际情况选择授权的资源范围，单击“确定”，创建委托完成。

----结束

## 创建 LakeFormation 数据连接

### 📖 说明

创建LakeFormation数据连接的功能，需要联系技术支持申请开通白名单。

**步骤1** 登录MRS控制台，在导航栏选择“数据连接”。

**步骤2** 单击“新建数据连接”。

**步骤3** 参考[表4-1](#)配置相关参数，单击“确定”完成创建。

表 4-1 配置 LakeFormation 数据连接

参数	说明
类型	选择“LakeFormation”，当前仅MRS 3.3.0-LTS及之后版本支持连接该类型。
名称	数据连接的名称。
LakeFormation实例	选择LakeFormation实例名称。 该实例需要先在LakeFormation实例创建后在此处引用，具体请参考 <a href="#">创建LakeFormation实例</a> 。单击“查看LakeFormation实例”查看已创建的实例。
虚拟私有云	需要与待对接的MRS集群在同一虚拟私有云。
子网	选择子网名称。
VPC终端节点	选择VPC终端节点，或单击“创建对应LakeFormation实例的VPC终端节点”进行创建。 选择VPC终端节点后，产生的费用将由VPCEP服务收取。
LakeFormation委托	选择“现有委托”，并选择 <a href="#">创建对接LakeFormation权限的委托</a> 创建的委托，例如“visit_lakeformation_agency”。

图 4-3 新建 LakeFormation 数据连接

新建数据连接
×

类型  ↻

名称

LakeFormation实例  ↻

[查看LakeFormation实例](#) [LakeFormation对接说明](#)

虚拟私有云  ↻ ? [查看虚拟私有云](#)

子网  ↻ ? [查看子网](#)

VPC终端节点  ↻ ? [查看VPC终端节点](#)

[创建对应LakeFormation实例的VPC终端节点](#)

LakeFormation委托 MRS\_LAKEFORMA... 现有委托 ?

↻

MRS的数据连接是用来管理集群中组件使用的外部源连接。 [了解更多](#)

确定
取消

**步骤4** 创建完成后，在“数据连接”页面记录已创建数据连接的ID。

----结束

## 获取账号 ID 信息

**步骤1** 使用待配置MRS与LakeFormation对接的用户，登录管理控制台。

**步骤2** 单击用户名，在下拉列表中单击“我的凭证”。

**步骤3** 在“API凭证”页面获取“账号ID”、项目列表中查看项目ID。



**步骤4** 为当前用户授权使用LakeFormation的权限。

1. 在左上角单击“☰”，选择“大数据 > 湖仓构建 LakeFormation”。
2. 查看是否弹出服务授权页面，或进入“服务授权”页面查看是否已授权。
  - 是，勾选“同意LakeFormation服务声明”，并单击“同意授权”，为当前服务授权。
  - 否，当前用户已有操作LakeFormation的权限。

----结束

## 4.1.3 创建集群时配置 LakeFormation 数据连接

该章节指导用户在创建MRS 3.3.0-LTS集群时配置LakeFormation数据连接，并在创建完成后配置MRS集群相关参数完成与LakeFormation的对接。

### 创建集群时配置 LakeFormation 数据连接

**步骤1** 进入[购买MRS集群页面](#)。



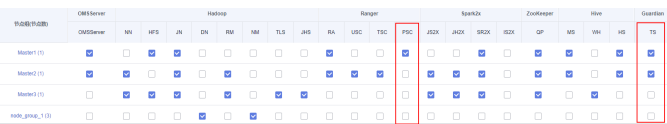
**步骤2** 单击“购买集群”，进入“购买集群”页面。


**步骤3** 在购买集群页面，选择“自定义购买”。

**步骤4** 参考[购买自定义拓扑集群](#)进行配置并创建集群，且集群需满足[表4-2](#)中要求。

表 4-2 LakeFormation 数据连接参数说明

参数	参数说明
版本类型	LTS版
集群版本	选择配置对接的MRS集群版本。 当前仅MRS 3.3.0-LTS及之后版本支持在创建集群时配置LakeFormation数据连接。

参数	参数说明
组件选择	<p>必须包含Hadoop、Ranger、Hive、Guardian、Spark（可选）、Flink（可选）等组件。</p> <p>例如，配置如下图所示，不同版本集群可能存在差异，具体界面显示以实际为准。</p> 
元数据	<p>选择“外置数据连接”，并配置以下参数：</p> <ol style="list-style-type: none"> <li>LakeFormation元数据：单击按钮开启。</li> <li>LakeFormation连接实例：选择<b>创建LakeFormation数据连接</b>已创建的LakeFormation数据连接名称。</li> <li>数据连接类型：保持默认。</li> </ol> <p>例如，配置如下图所示，不同版本集群可能存在差异，具体界面显示以实际为准。</p> 
虚拟私有云	与LakeFormation数据连接所在的虚拟私有云保持一致。
子网	选择子网名称。
拓扑调整	<p>选择“开启”，并确认Ranger组件至少添加1个PolicySync（PSC）实例（该实例部署节点需要同时包含RangerAdmin实例）、Guardian组件至少添加2个TokenSever（TS）实例。</p> <p>例如，配置如下图所示，不同版本集群可能存在差异，具体界面显示以实际为准。</p> 
Kerberos认证	开启

参数	参数说明
委托	<p>勾选“高级配置”后的“现在配置”，“委托”选择“现有委托”，并选择<a href="#">创建对接ECS/BMS云服务委托</a>创建的委托。</p> <p>例如，配置如下图所示，不同版本集群可能存在差异，具体界面显示以实际为准。</p> 

**步骤5** 等待集群创建完成后，在“现有集群”页面单击已创建的MRS集群名称，在“概览”页签单击“IAM用户同步”后的“同步”，根据界面提示同步当前用户。



**步骤6** 参考[配置MRS 3.3.0-LTS版本集群](#)配置组件存算分离、下载客户端等操作。

----结束

## 配置 MRS 3.3.0-LTS 版本集群

**步骤1** 登录MRS集群的FusionInsight Manager页面，具体操作请参考[访问FusionInsight Manager \(MRS 3.x及之后版本\)](#)。

**步骤2** 配置Guardian。

1. 在FusionInsight Manager界面，选择“集群 > 服务 > Guardian > 配置 > 全部配置”，搜索并修改以下参数后，单击“保存”。

表 4-3 配置 Guardian 参数

参数	含义	取值
token.server.access.iam.domain.id	访问IAM的用户对应的账号ID。 参考 <a href="#">获取账号ID信息</a> 获取账号ID信息。	xxx

参数	含义	取值
token.server.access.iam.project.id	访问IAM的用户对应的项目ID。 参考 <a href="#">获取账号ID信息</a> 获取项目ID信息。	xxx
token.server.access.label.agency.name	指定IAM委托的名字，需要具有访问OBS的权限。 即 <a href="#">创建对接OBS权限的委托</a> 创建的委托名称。	visit_obs_agency
fs.obs.delegation.token.providers	delegation.token的产生类名，默认为空。 此处同时勾选以下参数值： - com.huawei.mrs.dt.MRSDelegationTokenProvider - com.huawei.mrs.dt.GuardianDTPProvider	com.huawei.mrs.dt.MRSDelegationTokenProvider,com.huawei.mrs.dt.GuardianDTPProvider
fs.obs.guardian.accesslabel.enabled	是否开启使用Guardian对接OBS的access label。	true
fs.obs.guardian.enabled	是否开启使用Guardian。	true

2. 进入Guardian服务“概览”页面，选择“更多 > 重启服务”。

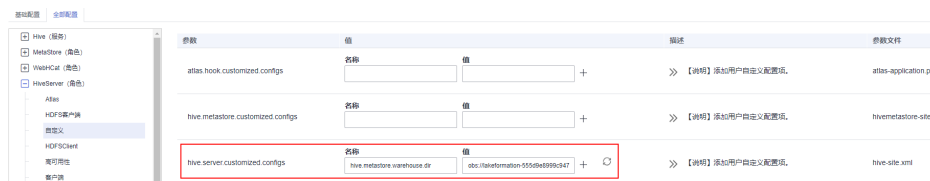
### 步骤3 配置Hive对接OBS文件系统。

1. 在FusionInsight Manager界面，选择“集群 > 服务 > Hive > 配置 > 全部配置”。
2. 在左侧的导航列表中选择“HiveServer > 自定义”。在自定义配置项中添加如下参数。

表 4-4 HiveServer 自定义参数配置说明

参数	描述	取值样例
hive.server.customized.configs	<ul style="list-style-type: none"> <li>- 添加参数“hive.metastore.warehouse.dir”。</li> <li>- 设置值为<a href="#">配置LakeFormation实例</a>章节获取的hive Catalog在OBS中的存储路径。</li> </ul>	<ul style="list-style-type: none"> <li>- 名称：hive.metastore.warehouse.dir</li> <li>- 值：obs://lakeformation-test/hive</li> </ul>

图 4-4 hive.metastore.warehouse.dir 配置



3. 单击“保存”，保存配置。

**步骤4** 配置Spark对接OBS文件系统。如果集群不存在Spark组件请跳过该步骤。

1. 在FusionInsight Manager界面，选择“集群 > 服务 > Spark > 配置 > 全部配置”。
2. 在左侧的导航列表中选择“JDBCServer > 自定义”，参考下表增加自定义参数及值。

表 4-5 Spark 参数配置

自定义参数	参数值
custom	- 参数: spark.sql.warehouse.location.first - 值: true
spark.hive-site.customized.configs	- 参数: hive.metastore.warehouse.dir - 值: 设置为配置LakeFormation实例章节获取的hive Catalog在OBS中的存储路径，例如“obs://lakeformation-test/hive”。

3. 在左侧的导航列表中选择“SparkResource > 自定义”，参考表4-5配置参数。
4. 单击“保存”，保存配置。

**步骤5** 在MRS集群“组件管理”页签，查看是否存在“配置超期”的组件，如果存在请单击“操作”列的“重启”，重启相关组件。

**步骤6** 重新下载并安装MRS集群完整客户端。具体操作请参考[安装客户端](#)。

**步骤7** 如果需要在管理控制台执行作业提交操作，需要更新集群内置客户端配置文件。

在MRS集群概览页面，获取弹性IP，使用该IP登录Master节点，执行如下命令刷新集群内置客户端。

```
su - omm
```

```
sh /opt/executor/bin/refresh-client-config.sh
```

**步骤8** 登录客户端安装节点，通过Hive客户端查看数据库，确认对接成功。

```
source 客户端安装路径/bigdata_env
```

```
kinit 组件业务用户
```

```
beeline
```

```
show databases;desc database default;
```

```
!q
```



**步骤9** 通过Spark客户端，查看数据库，确认对接成功。如果集群不存在Spark组件请跳过该步骤。

```
source 客户端安装路径/Spark/component_env
spark-sql
show databases;desc database default;
----结束
```

#### 4.1.4 通过 Ranger 为 MRS 集群内用户绑定 LakeFormation 角色

LakeFormation与MRS集群对接后，可以在Ranger WebUI界面为MRS集群内的用户或用户组绑定LakeFormation角色，则绑定的用户或用户组拥有该角色的权限。

##### 前提条件

- 已创建角色，并为该角色添加权限，具体操作请参考[角色授权](#)。
- 已完成MRS与LakeFormation实例的对接。

##### 操作步骤

**步骤1** 登录MRS管理控制台，选择“现有集群”，单击待操作的集群名称。

**步骤2** 单击“IAM用户同步”后的“同步”，等待同步成功。

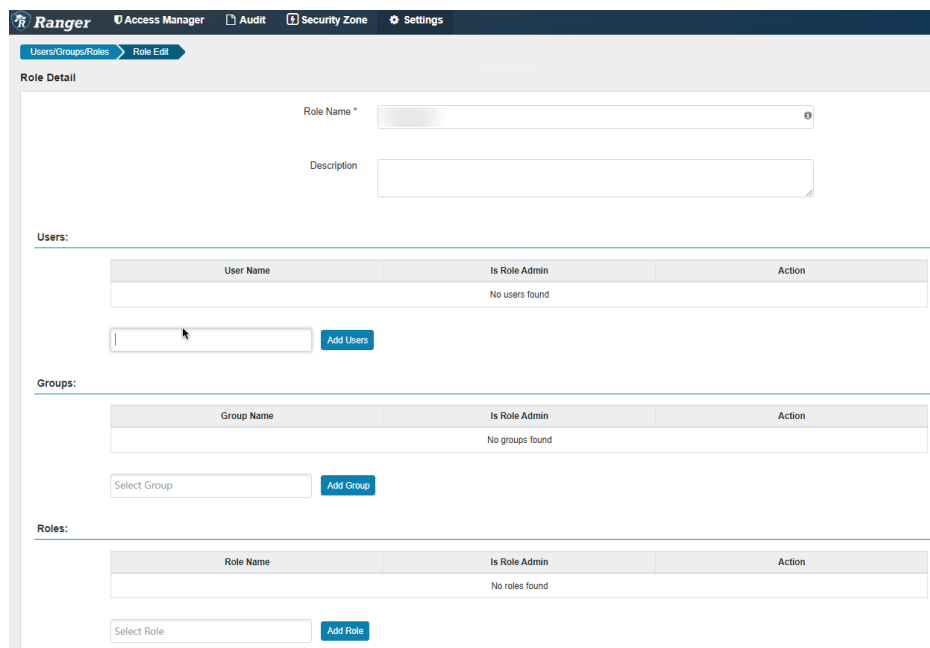
**步骤3** 以admin用户登录MRS集群的FusionInsight Manager页面，具体操作请参考[访问 FusionInsight Manager \(MRS 3.x及之后版本\)](#)。

**步骤4** 以rangeradmin用户登录Ranger WebUI界面。

1. 选择“集群 > 服务 > Ranger”，进入Ranger服务概览页面。
2. 单击“基本信息”区域中的“RangerAdmin”，进入Ranger WebUI界面。
3. 在Ranger WebUI界面，单击右上角用户名，选择“Log Out”，退出当前用户。
4. 使用rangeradmin用户或者其他具有Ranger管理员权限用户重新登录。  
rangeradmin用户默认密码请参考[用户账号一览表](#)获取。

**步骤5** 在Ranger WebUI界面选择“Settings > Roles”。

**步骤6** 单击已创建的LakeFormation角色名称，进入“Role Edit”界面。



**步骤7** 分别在“Users”区域单击“Select User”、“Groups”区域单击“Select Group”配置拥有该角色权限的用户或者用户组。

- Users: MRS集群系统中已创建的用户。
- Groups: MRS集群系统中已创建的用户组。

**步骤8** 单击“Save”保存配置。

配置完成后，所选择的用户和用户组即拥有该角色的权限。

----结束

## 4.2 使用 MRS Spark SQL 访问 DWS 表

### 应用场景

华为云提供MapReduce服务（MRS），可在云上快速构建和运营全栈云原生大数据平台。它包含HDFS、Hive、HBase、Spark等大数据组件，专为分析海量企业数据而量身定制。

Spark提供了类似SQL的Spark SQL语言，用于对结构化数据进行操作。使用Spark SQL，可以访问不同的数据库，用户可以从这些数据库中提取数据，处理并加载到不同的数据存储中。

本实践演示如何使用MRS Spark SQL访问GaussDB(DWS)数据。

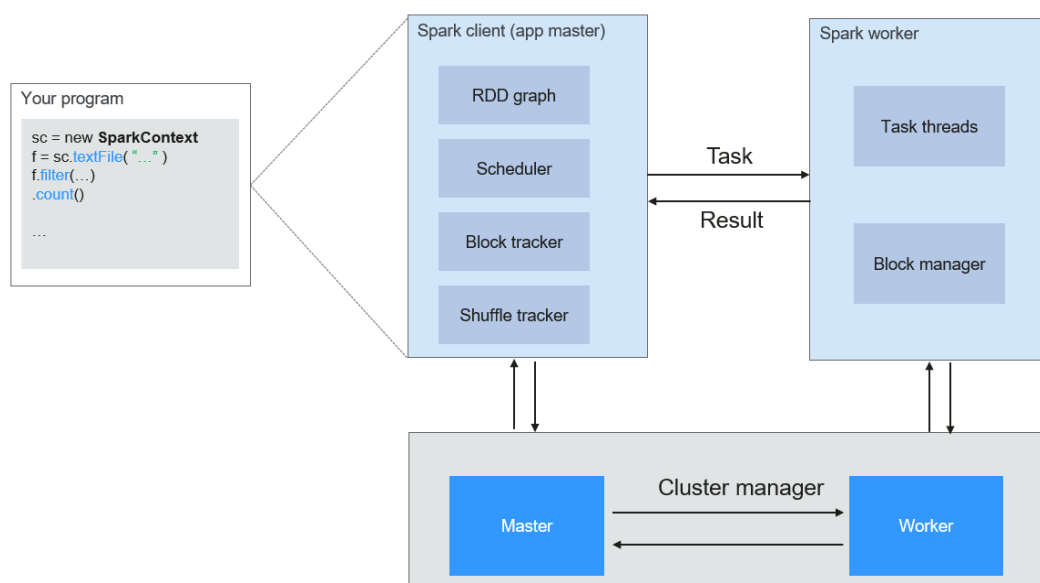
### 方案架构

Spark的应用运行架构如图4-5所示，运行流程如下所示：

1. 应用程序（Application）是作为一个进程的集合运行在集群上的，由Driver进行协调。

2. 在运行一个应用时，Driver会去连接集群管理器（Standalone、Mesos、YARN）申请运行Executor资源，并启动ExecutorBackend。然后由集群管理器在不同的应用之间调度资源。Driver同时会启动应用程序DAG调度、Stage划分、Task生成。
3. 然后Spark会把应用的代码（传递给SparkContext的JAR或者Python定义的代码）发送到Executor上。
4. 所有的Task执行完成后，用户的应用程序运行结束。

图 4-5 Spark 应用运行架构



## 约束与限制

- 本章节仅适用于MRS 3.x及之后版本。
- 为确保网络连通，GaussDB(DWS)集群需与MRS集群“可用区”、“虚拟私有云”、“安全组”配置相同。

## 前提条件

- 已创建DWS集群，请参考[创建GaussDB\(DWS\)集群](#)。
- 已获取连接GaussDB(DWS)数据库的IP地址、端口、数据库名称、用户名和密码。此外，操作用户必须具有GaussDB(DWS)表的读写权限。

## 步骤一：创建 MRS 集群

1. 创建MRS集群。  
创建并购买一个包含Spark组件的MRS集群，详情可参考[创建MRS集群](#)。
2. 如果MRS集群开启了Kerberos认证，登录FusionInsight Manager页面，选择“系统 > 权限 > 用户”，添加一个人机用户sparkuser，用户组（hadoop、hive），主组（hadoop）。  
参考[添加Spark2x的Ranger访问权限策略](#)章节，添加“ADD JAR操作”权限。  
如果MRS集群未开启Kerberos认证，则无需创建用户。
3. 安装MRS集群客户端。  
安装MRS集群客户端，具体请参考[安装客户端](#)。

## 步骤二：配置 MRS Spark SQL 访问 DWS 表

**步骤1** 准备数据，在GaussDB(DWS)集群中创建数据库和表：

1. 登录GaussDB(DWS)管理控制台，单击DWS集群“操作”列的“登录”。
2. 登录现有GaussDB(DWS)集群的默认数据库gaussdb，执行以下命令，创建数据库“dws\_test”。

```
CREATE DATABASE dws_test;
```

3. 连接到创建的新数据库，执行以下命令，创建表“dws\_order”。

```
CREATE SCHEMA dws_data;  
CREATE TABLE dws_data.dws_order
```

```
( order_id VARCHAR,  
  order_channel VARCHAR,  
  order_time VARCHAR,  
  cust_code VARCHAR,  
  pay_amount DOUBLE PRECISION,  
  real_pay DOUBLE PRECISION );
```

4. 执行以下命令插入数据到表“dws\_order”中。

```
INSERT INTO dws_data.dws_order VALUES ('202306270001', 'webShop',  
                                         '2023-06-27 10:00:00', 'CUST1', 1000, 1000);
```

```
INSERT INTO dws_data.dws_order VALUES ('202306270002', 'webShop',  
                                         '2023-06-27 11:00:00', 'CUST2', 5000, 5000);
```

5. 查询表数据，验证数据是否插入。

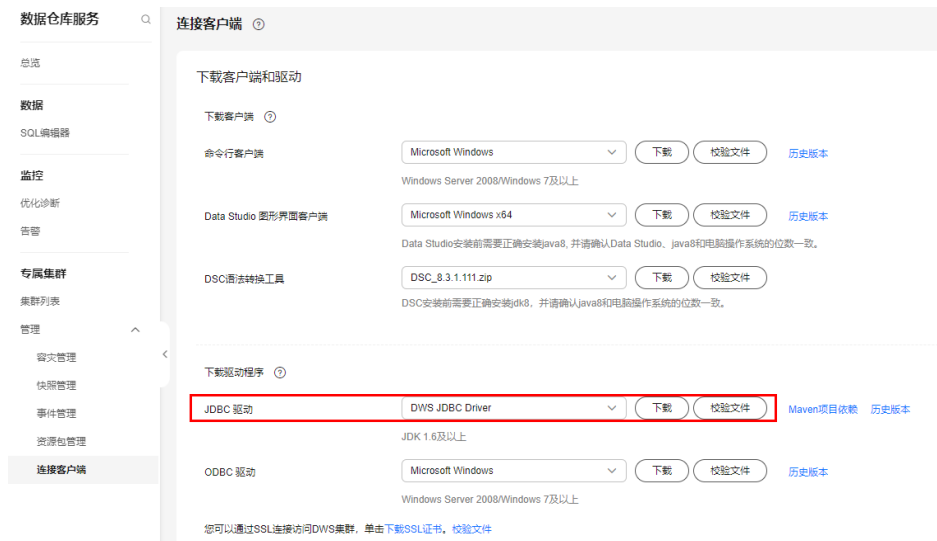
```
SELECT * FROM dws_data.dws_order;
```



**步骤2** 下载GaussDB(DWS)数据库JDBC驱动并上传到MRS集群。

1. 登录GaussDB(DWS)管理控制台，选择左侧的“管理 > 连接客户端”，下载JDBC驱动，如下图所示：

图 4-6 下载 JDBC 驱动



2. 解压，获取“gsjdbc200.jar”文件，并上传到MRS集群主Master节点，例如上传到“/tmp”目录下。
3. 使用root用户登录MRS集群主Master节点，执行如下命令：

```
cd {客户端安装目录}
```

```
source bigdata_env
```

```
kinit sparkuser （首次认证需要修改密码，未开启Kerberos认证，则无需执行kinit命令。）
```

```
hdfs dfs -put /tmp/gsjdbc200.jar /tmp
```

```

[root@node-master21snt tmp]# hdfs dfs -put ./gsjdbc200.jar /tmp
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/opt/Bigdata/client/HDFS/hadoop/share/hadoop/common/lib/slf4j-log4j12-1.7.30.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/opt/client/HDFS/hadoop/share/hadoop/common/lib/slf4j-log4j12-1.7.30.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
SLF4J: Actual binding is of type [org.slf4j.impl.Log4jLoggerFactory]
put: /tmp/gsjdbc200.jar: File exists
[root@node-master21snt tmp]#
    
```

**步骤3** 在MRS Spark中创建数据源表，并访问DWS表：

1. 登录Spark客户端节点，执行如下命令：

```
cd 客户端安装目录
```

```
source ./bigdata_env
```

```
kinit sparkuser
```

```
spark-sql --master yarn
```

2. 执行如下命令，添加驱动程序jar：

```
add jar hdfs://hacluster/tmp/gsjdbc200.jar;
```

```

spark-sql> add jar hdfs://hacluster/tmp/gsjdbc200.jar;
2023-06-28 01:36:39,554 | WARN | main | The enable mv value "null" is invalid. Using the default value "false" | org.apache.carbondata.core.util.CarbonProperties.validateEnableMV(CarbonProperties.java:512)
2023-06-28 01:36:39,568 | WARN | main | The value "LOCALLOCK" configured for key carbon.lock.type is invalid for current file system. Use the default value HDFSLOCK instead. | org.apache.carbondata.core.util.CarbonProperties.validateAndConfigureLockType(CarbonProperties.java:441)
add JAR hdfs://hacluster/tmp/gsjdbc200.jar
Added [/opt/Bigdata/client/Spark2x/tmp/b52347ce-d7c4-44d4-8868-cefac46b2d0e_resources/gsjdbc200.jar] to class path
Added resources: [hdfs://hacluster/tmp/gsjdbc200.jar]
add JAR hdfs://hacluster/tmp/gsjdbc200.jar
Added [/opt/Bigdata/client/Spark2x/tmp/b52347ce-d7c4-44d4-8868-cefac46b2d0e_resources/gsjdbc200.jar] to class path
Added resources: [hdfs://hacluster/tmp/gsjdbc200.jar]
Time taken: 1.967 seconds
    
```

3. 执行如下命令，在Spark中创建数据源表，访问DWS数据：

```
CREATE TABLE IF NOT EXISTS spk_dws_order
```

```
USING JDBC OPTIONS (  
'url'='jdbc:gaussdb://192.168.0.228:8000/dws_test',  
'driver'='com.huawei.gauss200.jdbc.Driver',  
'dbtable'='dws_data.dws_order',  
'user'='dbadmin',  
'password'='xxx');
```

4. 查询Spark表，验证显示的数据是否与DWS数据相同：

```
SELECT * FROM spk_dws_order;
```

```
spark-sql> SELECT * FROM spk_dws_order;  
202306270001    webShop 2023-06-27 10:00:00    CUST1    1000.0    1000.0  
202306270002    webShop 2023-06-27 11:00:00    CUST2    5000.0    5000.0  
Time taken: 3.416 seconds, Fetched 2 row(s)  
spark-sql>
```

可以验证返回的数据与[步骤1](#)中所示的数据相同。

----结束

## 4.3 使用 MRS Hive 表对接 OBS 文件系统

### 应用场景

MRS支持用户将数据存储于OBS服务中，使用MRS集群仅作数据计算处理的存算分离场景。

用户通过IAM服务的“委托”机制进行简单配置，即可实现OBS的访问。

### 方案架构

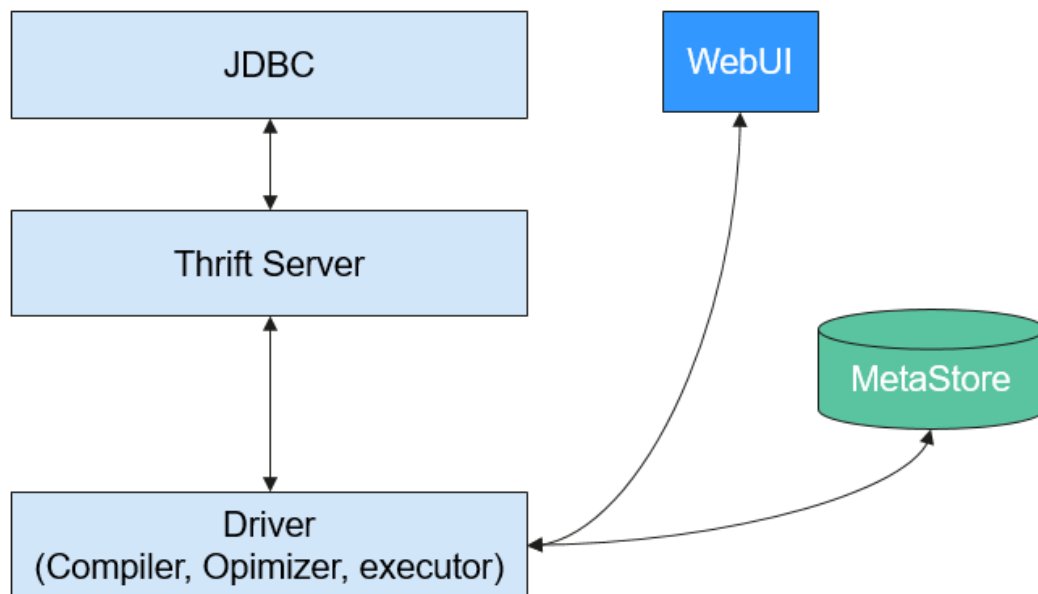
Hive是建立在Hadoop上的数据仓库框架，提供大数据平台批处理计算能力，能够对结构化/半结构化数据进行批量分析汇总完成数据计算。提供类似SQL的Hive Query Language语言操作结构化数据，其基本原理是将HQL语言自动转换成MapReduce任务，从而完成对Hadoop集群中存储的海量数据进行查询和分析。

Hive主要特点如下：

- 海量结构化数据分析汇总。
- 将复杂的MapReduce编写任务简化为SQL语句。
- 灵活的数据存储格式，支持JSON、CSV、TEXTFILE、RCFILE、SEQUENCEFILE、ORC等存储格式。

Hive作为一个基于HDFS和MapReduce架构的数据仓库，其主要能力是通过对HQL（Hive Query Language）编译和解析，生成并执行相应的MapReduce任务或者HDFS操作。

图 4-7 Hive 结构



- **Metastore**: 对表、列和 Partition 等的元数据进行读写及更新操作，其下层为关系型数据库。
- **Driver**: 管理 HQL 执行的生命周期并贯穿 Hive 任务整个执行期间。
- **Compiler**: 编译 HQL 并将其转化为一系列相互依赖的 Map/Reduce 任务。
- **Optimizer**: 优化器，分为逻辑优化器和物理优化器，分别对 HQL 生成的执行计划和 MapReduce 任务进行优化。
- **Executor**: 按照任务的依赖关系分别执行 Map/Reduce 任务。
- **ThriftServer**: 提供 thrift 接口，作为 JDBC 的服务端，并将 Hive 和其他应用程序集成起来。
- **Clients**: 包含 WebUI 和 JDBC 接口，为用户访问提供接口。

## 方案优势

MRS 支持在大数据存储容量大、计算资源需要弹性扩展的场景下，用户将数据存储在大数 OBS 服务中，使用 MRS 集群仅作数据计算处理的存算分离模式，从而实现按需灵活扩展资源、低成本的海量数据分析方案。

## 约束与限制

大数据存算分离场景，请务必使用 OBS 并行文件系统，使用普通对象桶会对集群性能产生较大影响。

## 操作流程

本章节指导用户创建 Hive 表存放数据到 OBS，基本流程如下所示：

1. **步骤一：创建 ECS 委托**
2. **步骤二：为 MRS 集群配置委托**
3. **步骤三：创建 OBS 文件系统**

#### 4. 步骤四：Hive访问OBS文件系统

### 步骤一：创建 ECS 委托

1. 登录华为云管理控制台。
2. 在服务列表中选择“管理与监管 > 统一身份认证服务”。
3. 选择“委托 > 创建委托”。
4. 设置“委托名称”。例如：mrs\_ecs\_obs。
5. “委托类型”选择“云服务”，在“云服务”中选择“弹性云服务器 ECS 裸金属服务器 BMS”，授权ECS或BMS调用OBS服务，如图4-8所示。
6. “持续时间”选择“永久”并单击“完成”。

图 4-8 创建委托

\* 委托名称

\* 委托类型  普通账号  
将账号内资源的操作权限委托给其他华为云账号。  
 云服务  
将账号内资源的操作权限委托给华为云服务。

\* 云服务

\* 持续时间

描述  0/255

7. 在委托的“操作”列单击“授权”，搜索“OBS OperateAccess”策略，勾选“OBS OperateAccess”策略。
8. 单击“下一步”，选择权限范围方案，默认选择“所有资源”，单击“展开其他方案”，选择“全局服务资源”，单击“确定”。
9. 在弹出的提示框中单击“知道了”，开始授权。界面提示“授权成功。”，单击“完成”，委托成功创建。

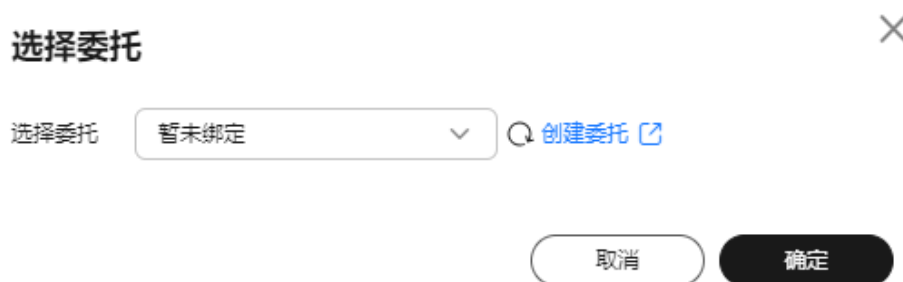
### 步骤二：为 MRS 集群配置委托

配置存算分离支持在新建集群中配置委托实现，也可以通过为已有集群绑定委托实现。本示例以为已有集群配置委托为例介绍。



1. 登录MRS控制台，在导航栏选择“现有集群”。
2. 单击集群名称，进入集群详情页面。
3. 在集群详情页的“概览”页签，单击“IAM用户同步”右侧的“同步”进行IAM用户同步。
4. 在集群详情页的“概览”页签，单击“委托”右侧的“选择委托”选择**步骤一：创建ECS委托**的委托并单击“确定”进行绑定，或单击“创建委托”进入IAM控制台进行创建后再在此处进行绑定。

图 4-9 绑定委托



### 步骤三：创建 OBS 文件系统

1. 登录OBS控制台。
2. 选择“并行文件系统 > 创建并行文件系统”。
3. 填写文件系统名称，例如“mrs-demo01”。其他参数请根据需要填写。
4. 单击“立即创建”。
5. 在OBS控制台并行文件系统列表中，单击文件系统名称进入详情页面。
6. 在左侧导航栏选择“文件”，新建program、input文件夹。
  - program：请上传程序包到该文件夹。
  - input：请上传输入数据到该文件夹。

### 步骤四：Hive 访问 OBS 文件系统

1. 用root用户登录集群Master节点，具体请参见[登录集群节点](#)。
2. 验证Hive访问OBS。
  - a. 用root用户登录集群Master节点，执行如下命令：

```
cd /opt/Bigdata/client
source bigdata_env
source Hive/component_env
```
  - b. 查看文件系统mrs-demo01下面的文件列表。

```
hadoop fs -ls obs://mrs-demo01/
```
  - c. 返回文件列表即表示访问OBS成功。

图 4-10 查看 mrs-demo01 下的文件列表

```
Found 2 items
drwxrwxrwx - hive hive          0 2021-10-22 10:08 obs://mrs-demo01/input
drwxrwxrwx - hive hive          0 2021-10-22 10:08 obs://mrs-demo01/program
```

- d. 执行以下命令进行用户认证（普通模式即未开启Kerberos认证无需执行此步骤）。

**kinit hive**

输入用户hive密码，默认密码为Hive@123，第一次使用需要修改密码。

- e. 执行Hive组件的客户端命令。

**beeline**

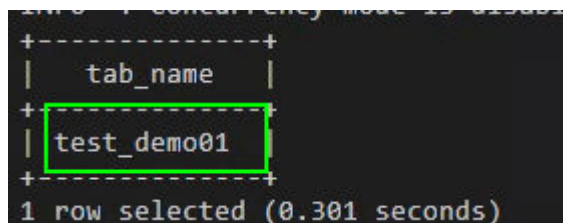
- f. 在beeline中直接使用OBS的目录进行访问。例如，执行如下命令创建Hive表并指定数据存储在不mrs-demo01文件系统的test\_demo01目录中。

```
create table test_demo01(name string) location "obs://mrs-demo01/test_demo01";
```

- g. 执行如下命令查询所有表，返回结果中存在表test\_demo01，即表示访问OBS成功。

```
show tables;
```

图 4-11 查看是否存在表 test\_demo01



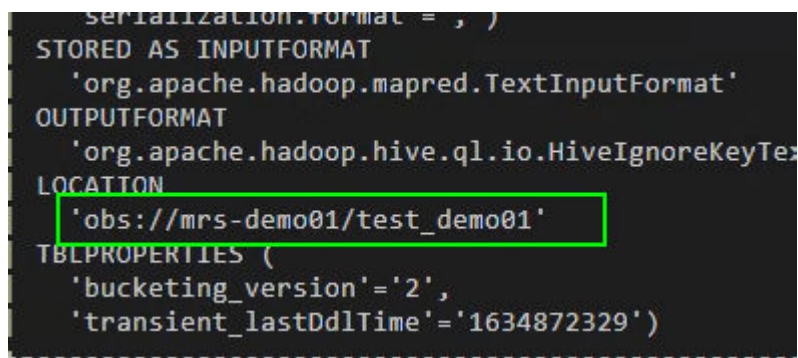
```
+-----+
| tab_name |
+-----+
| test_demo01 |
+-----+
1 row selected (0.301 seconds)
```

- h. 查看表的Location。

```
show create table test_demo01;
```

查看表的Location是否为“obs://OBS桶名/”开头。

图 4-12 查看表 test\_demo01 的 Location



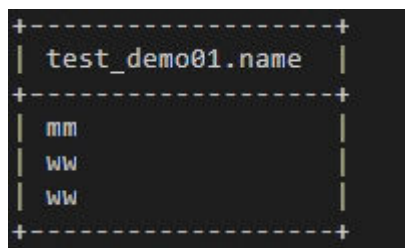
```
serialization.format = ,
STORED AS INPUTFORMAT
  'org.apache.hadoop.mapred.TextInputFormat'
OUTPUTFORMAT
  'org.apache.hadoop.hive.ql.io.HiveIgnoreKeyTextOutputFormat'
LOCATION
  'obs://mrs-demo01/test_demo01'
TBLPROPERTIES (
  'bucketing_version'='2',
  'transient_lastDdlTime'='1634872329')
```

- i. 写入数据。

```
insert into test_demo01 values('mm'),('ww'),('ww');
```

执行select \* from test\_demo01;查询是否写入成功。

图 4-13 查看表 test\_demo01 中的数据



- j. 执行命令!q退出beeline客户端。
- k. 重新登录OBS控制台。
- l. 单击“并行文件系统”，选择创建的文件系统名称。
- m. 单击“文件”，查看是否存在创建的数据。

## 4.4 MRS Hive 对接 CSS 服务配置说明

### 应用场景

利用Elasticsearch-Hadoop插件，完成Hive和CSS服务的Elasticsearch直接的数据交互，通过Hive外部表的方式，可以快速将Elasticsearch索引数据映射到Hive表中。

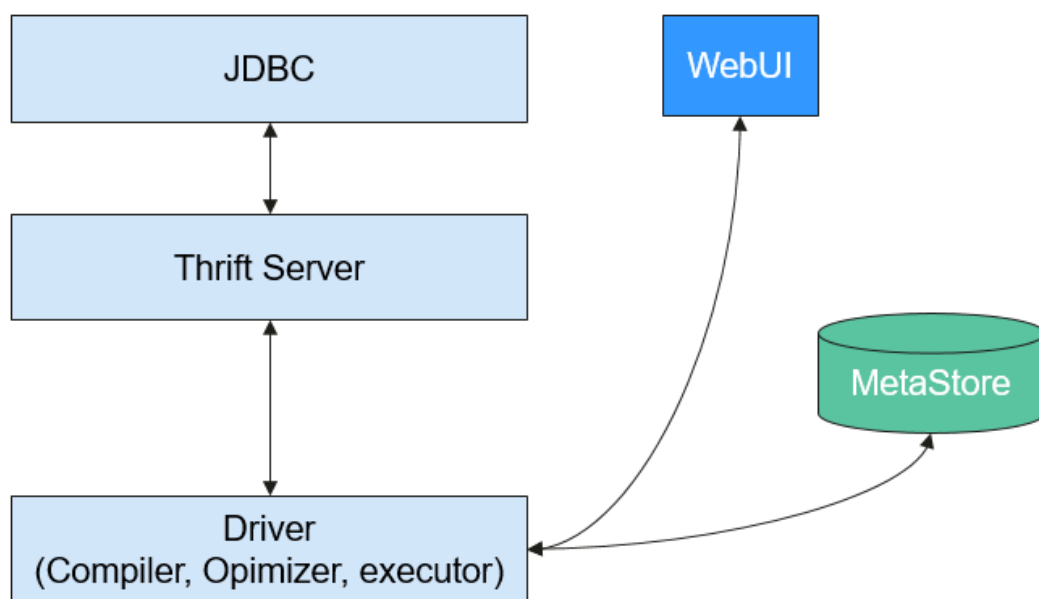
Elasticsearch-Hadoop (ES-Hadoop) 连接器将Hadoop海量的数据存储和深度加工能力与Elasticsearch实时搜索和分析功能结合在一起。

它能够让您快速深入了解大数据，并让您在Hadoop生态系统中更好地开展工作。

### 方案架构

Hive是建立在Hadoop上的数据仓库框架，提供大数据平台批处理计算能力，能够对结构化/半结构化数据进行批量分析汇总完成数据计算。提供类似SQL的Hive Query Language语言操作结构化数据，其基本原理是将HQL语言自动转换成MapReduce任务，从而完成对Hadoop集群中存储的海量数据进行查询和分析。

图 4-14 Hive 结构



云搜索服务（Cloud Search Service，简称CSS）是一个基于Elasticsearch、OpenSearch且完全托管的在线分布式搜索服务，为用户提供结构化、非结构化文本、以及基于AI向量的多条件检索、统计、报表。云搜索服务是华为云ELK生态的一系列软件集合，为您全方位提供托管的ELK生态云服务，兼容Elasticsearch、Logstash、Kibana、Cerebro等软件。

## 约束与限制

- 已安装MRS的Hive服务和CSS的Elasticsearch服务，并且两个集群之间网络互通。
- 适用于MRS 3.x及之后版本。

## 步骤一：配置云搜索服务索引并下载 jar 包

1. 在云搜索服务的“集群管理”页面上，单击集群“操作”列的“Kibana”访问集群，在Kibana的左侧导航中选择“Dev Tools”，进入Console界面，输入以下命令创建索引“ddj\_study\_card\_ratio\_v12”：

```
PUT ddj_study_card_ratio_v12
{
  "mappings": {
    "properties": {
      "uniq_id": {
        "type": "text",
        "fields": {
          "keyword": {
            "type": "keyword",
            "ignore_above": 256
          }
        }
      }
    }
  }
}
```

返回如下结果表示索引创建成功：

```
{
  "acknowledged": true,
  "shards_acknowledged": true,
  "index": "ddj_study_card_ratio_v12"
}
```

2. 执行以下命令向“ddj\_study\_card\_ratio\_v12”索引中插入数据：

```
POST /ddj_study_card_ratio_v12/_doc/_bulk
{"index":{}}
{"id":"1","uniq_id":"23323"}
```

当返回结果信息中“errors”字段的值为“false”时，表示导入数据成功。

3. 根据当前CSS服务中的Elasticsearch版本在[Past Releases](#)下载对应的Jar包。  
例如：Elasticsearch 7.6.2对应的Jar包为“elasticsearch-hadoop-7.6.2.jar”。

### 📖 说明

- Jar包版本需与CSS服务的Elasticsearch版本保持一致，本章节以开启安全模式的“Elasticsearch 7.6.2”集群为例进行相关操作，具体请以实际情况为准。
- 若有额外的自定义模块，也单独打包成一份Jar包。

## 步骤二：在 MRS 设置对接云搜索服务

1. 将在Past Releases下载的对应的Jar包分别上传到所有的HiveServer所在节点的“/opt/Bigdata/third\_lib/Hive”目录下，并执行以下命令修改权限。

```
chown omm:wheel -R /opt/Bigdata/third_lib/Hive
```

2. 登录FusionInsight Manager，选择“集群 > 服务 > Hive > 实例”，勾选所有的HiveServer实例，选择“更多 > 重启实例”，重启HiveServer实例。
3. 在Maven中心仓下载“commons-httpclient-3.1.jar”，并将该Jar包及在Past Releases下载的Jar包上传至集群中安装了HDFS和Hive客户端的任一节点上。
4. 以客户端安装用户，登录上一步上传了Jar包的节点。
5. 执行以下命令认证用户。  
**cd 客户端安装目录**  
**source bigdata\_env**  
**kinit 组件业务用户**（未开启Kerberos认证的集群跳过此操作）
6. 执行以下命令在HDFS上创建Jar包存放目录。  
**hdfs dfs -mkdir Jar包存放在hdfs中的路径**
7. 执行以下命令将在Maven中心仓下载的Jar包和在Past Releases下载的Jar包上传至HDFS中。  
**hdfs dfs -put Jar包存放路径 Jar包存放在hdfs中的路径**
8. 执行以下命令让Hive在执行命令行任务时加载指定Jar包。  
**beeline**  
**add jar Jar包存放在hdfs中的路径**（每个Jar包分别执行一次该命令）
9. 执行以下命令创建Elasticsearch外表。

```
CREATE EXTERNAL TABLE `dj_study_card_ratio_v12_test` (  
  `uniq_id` string)  
ROW FORMAT SERDE  
  'org.elasticsearch.hadoop.hive.EsSerDe'  
STORED BY  
  'org.elasticsearch.hadoop.hive.EsStorageHandler'  
WITH SERDEPROPERTIES (  
  'field.delim'=',',  
  'serialization.format'='')  
TBLPROPERTIES (  
  'bucketing_version'='2',  
  'es.index.auto.create'='false',  
  'es.mapping.date.rich'='false',  
  'es.net.http.auth.pass'='Pzh6537projectX',  
  'es.net.http.auth.user'='elastic',  
  'es.nodes'='vpcep-e0b33065-75b7-4193-8395-dbd00d10bc39.cn-east-3.huaweicloud.com',  
  'es.nodes.wan.only'='true',  
  'es.port'='9200',  
  'es.read.metadata'='true',  
  'es.resource'='dj_study_card_ratio_v12',  
  'es.set.netty.runtime.available.processors'='false',  
  'es.write.operation'='index',  
  'last_modified_by'='root',  
  'last_modified_time'='1655264909',  
  'transient_lastDdlTime'='1655264909');
```

### 📖 说明

关键参数说明：

- es.net.http.auth.pass、es.net.http.auth.user：在Kibana中创建的具有1创建的索引的操作权限的用户密码及用户名，详细内容请参见[使用Kibana创建用户并授权](#)。
- es.nodes：需要连接的IP，可登录CSS管理控制台，在集群列表的“内网访问地址”列即可查看对应集群的IP地址。
- es.port：外部访问Elasticsearch集群端口，默认为9200。
- es.resource：1创建的索引名称。

更多参数配置可参考开源社区文档<https://www.elastic.co/guide/en/elasticsearch/hadoop/6.1/hive.html>。

10. 执行以下命令查看创建的Elasticsearch外部表:

```
select * from ddj_study_card_ratio_v12_test;
```

当返回结果信息中无报错信息，并且查询成功时，表示Hive成功对接CSS服务。查询结果如下所示:

```
jd@bcshive2://192.168.0.129:10000/> select * from ddj_study_card_ratio_v12_test002;
INFO : State: Compiling.
INFO : Compiling command(queryId=oom_20220727154319_1c7f3fdf-5c8c-4c3d-80a2-f95888bcc5a4): select * from ddj_study_card_ratio_v12_test002; Current sessionId=a9f6dd55-f30
INFO : hive.compile.auto.avoid.rboptions
INFO : Concurrency mode is disabled, not creating a lock manager
INFO : Current sql is not contains insert syntax, not need record dest table flag
INFO : Semantic Analysis Completed (retiral = false)
INFO : Returning Hive schema: Schema(fieldsSchemas:[FieldSchema(name:ddj_study_card_ratio_v12_test002.uni_id, type:string, comment:null)], properties:null)
INFO : Completed compiling command(queryId=oom_20220727154319_1c7f3fdf-5c8c-4c3d-80a2-f95888bcc5a4); Time taken: 0.086 seconds
INFO : Concurrency mode is disabled, not creating a lock manager
INFO : State: Executing.
INFO : Executing command(queryId=oom_20220727154319_1c7f3fdf-5c8c-4c3d-80a2-f95888bcc5a4): select * from ddj_study_card_ratio_v12_test002; Current sessionId=a9f6dd55-f30
INFO : Completed executing command(queryId=oom_20220727154319_1c7f3fdf-5c8c-4c3d-80a2-f95888bcc5a4); Time taken: 0.0 seconds
INFO : OK
INFO : Concurrency mode is disabled, not creating a lock manager
+-----+
| ddj_study_card_ratio_v12_test002.uni_id |
+-----+
| NULL |
+-----+
Total selected: [0.31 sec:0096]
```

# 5 生态组件对接

## 5.1 使用 DBeaver 访问 Phoenix

### 应用场景

DBeaver是一个SQL客户端和数据库管理工具。对于关系数据库，使用JDBC API通过JDBC驱动程序与数据库交互。对于其他数据库（NoSQL），使用专有数据库驱动程序。

本章节以DBeaver 6.3.5版本为例，讲解如何使用DBeaver访问MRS 3.1.0未开启Kerberos认证的集群，且该集群的HBase服务未开启Ranger鉴权。

### 方案架构

HBase以表的形式存储数据，数据模型如[图 HBase数据模型](#)所示。表中的数据划分为多个Region，并由Master分配给对应的RegionServer进行管理。

每个Region包含了表中一段RowKey区间范围内的数据，HBase的一张数据表开始只包含一个Region，随着表中数据的增多，当一个Region的大小达到容量上限后会分裂成两个Region。可以在创建表时定义Region的RowKey区间，或者在配置文件中定义Region的大小。

图 5-1 HBase 数据模型

Row Key	Timestamp	Column Family 1		Column Family N	
		URI	Content	Column 1	Column 2
row1	t2	www. .com	"<html>..."	...	...
	t1	www. .com	"<html>..."	...	...
...	...	...	...	...	...
rowM	...	...	...	...	...
rowM+1	t1	...	...	...	...
rowM+2	t3	...	...	...	...
	t2	...	...	...	...
...	...	...	...	...	...
rowN	t1	...	...	...	...
...	...	...	...	...	...

Phoenix是构建在HBase之上的一个SQL中间层，提供一个客户端可嵌入的JDBC驱动，Phoenix查询引擎将SQL输入转换为一个或多个HBase scan，编译并执行扫描任务以产生一个标准的JDBC结果集。

## 约束与限制

- 已安装DBeaver 6.3.5版本，DBeaver软件下载链接为：[https://dbeaver.io/files/6.3.5/dbeaver-ce-6.3.5-x86\\_64-setup.exe](https://dbeaver.io/files/6.3.5/dbeaver-ce-6.3.5-x86_64-setup.exe)。
- 已安装JDK 1.8.0\_x。

## 步骤一：配置 DBeaver 安装环境

1. 向DBeaver安装目录下的“dbeaver.ini”文件中增加JDK 1.8.0\_x的bin目录，例如：C:\Program Files\Java\jdk1.8.0\_121\bin，则新增如下内容：

图 5-2 新增 JDK 的 bin 目录

```
-vm  
C:\Program Files\Java\jdk1.8.0_121\bin
```

2. 在<https://archive.apache.org/dist/phoenix/apache-phoenix-5.0.0-HBase-2.0/bin/apache-phoenix-5.0.0-HBase-2.0-bin.tar.gz>中下载Phoenix软件包并解压，获取“phoenix-5.0.0-HBase-2.0-client.jar”。
3. 创建并购买一个包含HBase组件的MRS 3.1.0未开启Kerberos认证的集群，详情可参考[创建MRS集群](#)。
4. 安装包含HBase的客户端，并从客户端安装节点上的“客户端安装目录/HBase/hbase/conf”目录下载“hbase-site.xml”文件。使用压缩软件打开获取到的“phoenix-5.0.0-HBase-2.0-client.jar”，将“hbase-site.xml”拖拽到该Jar文件中。



## 步骤二：在 DBeaver 配置访问 Phoenix 参数

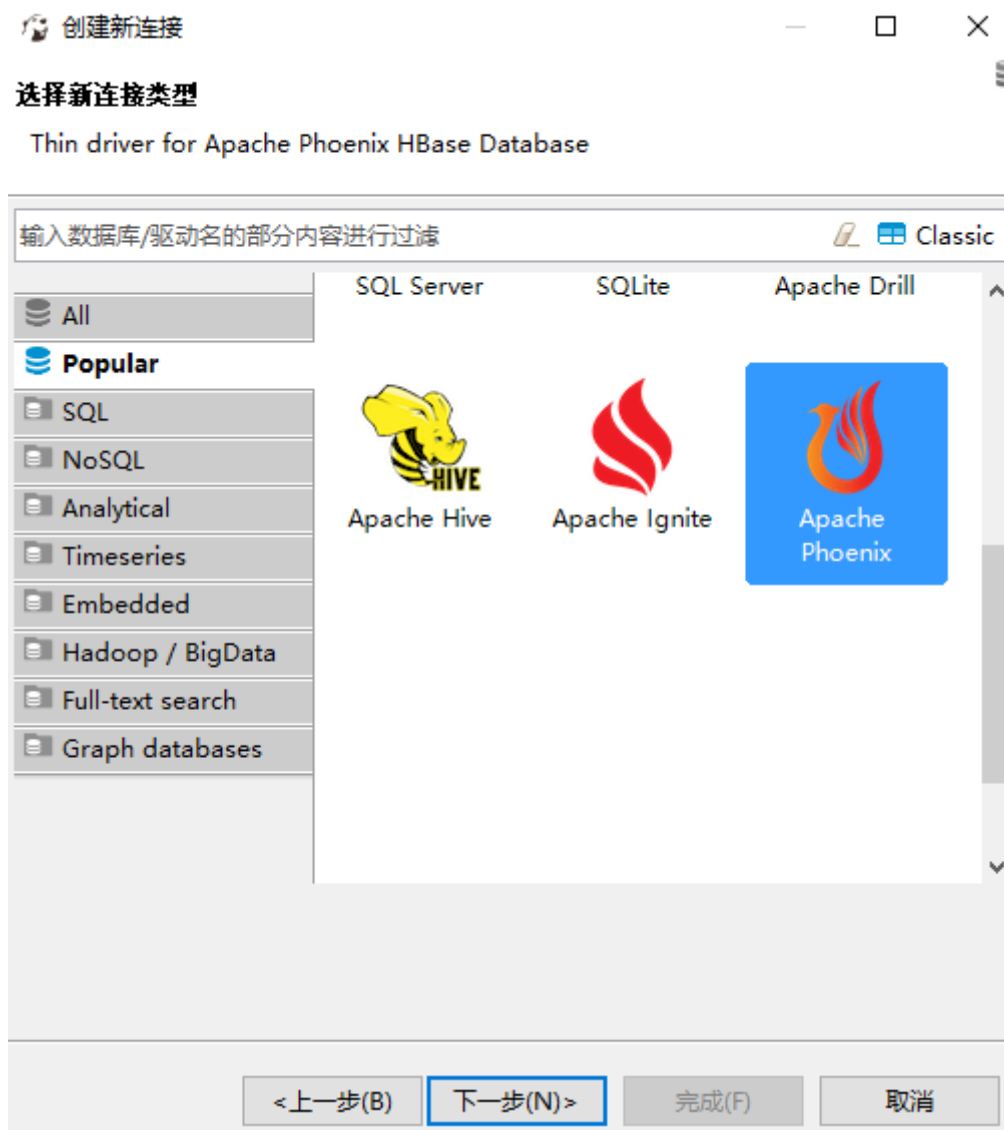
1. 打开DBeaver，选择“文件 > 新建 > DBeaver > 数据库连接”。

图 5-3 创建数据库连接



2. 单击“下一步”，在选择新连接类型界面选中“Apache Phoenix”并单击“下一步”。

图 5-4 选择数据连接类型



3. 单击“编辑驱动设置”。

图 5-5 编辑驱动设置



- 单击“添加文件”，选择准备好的“phoenix-5.0.0-HBase-2.0-client.jar”，如果有多个驱动包，需先删除，只保留手动添加的“phoenix-5.0.0-HBase-2.0-client.jar”。

图 5-6 删除原有的驱动包

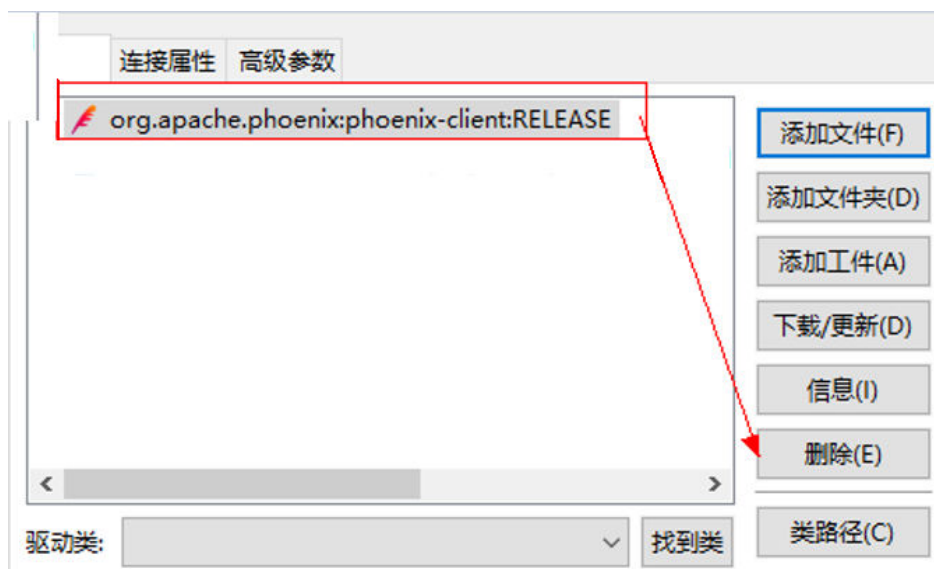
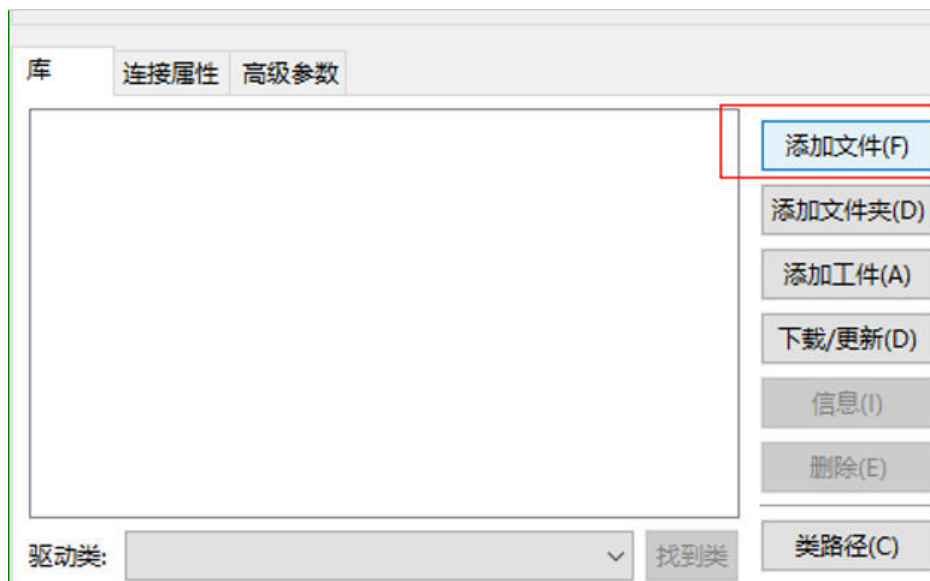
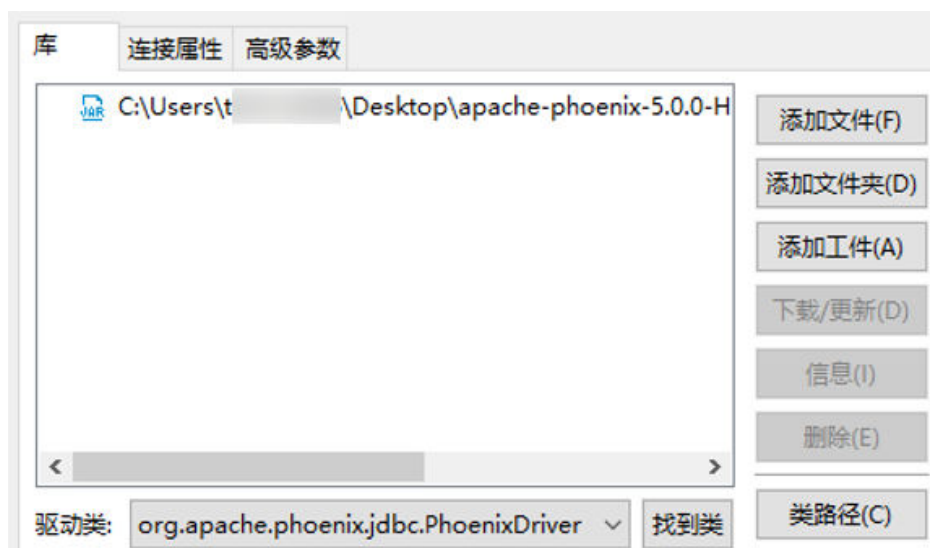


图 5-7 添加 Phoenix Jar 包



5. 单击“找到类”，加载完成后在左侧框中选择“org.apache.phoenix.jdbc.PhoenixDriver”。

图 5-8 加载驱动类



6. 添加Zookeeper Base Path。
  - a. 登录FusionInsight Manager，选择“集群 > 服务 > HBase”，单击“HMaster WebUI”右侧的超链接，登录HBase WebUI界面，搜索“Zookeeper Base Path”，并获取该值，如下图所示“Zookeeper Base Path”的值为“/hbase”：

图 5-9 查看 Zookeeper Base Path

### Software Attributes

Attribute Name	Value
JVM Version	
HBase Version	?, revision=9c59dbc63eb2daf08b29c51f4bce7c77f642ed12
HBase Compiled	Wed Apr 28 18:49:13 CST 2021, root
HBase Source Checksum	6cfcc863c31df1d8127824d2b08d604d
Hadoop Version	?, revision=3f6d58324da792aaa3a5592c59561de6387cbe93
Hadoop Compiled	2021-04-28T10:26Z, root
Hadoop Source Checksum	15ad5f9e94eaf31a9cb0fbff55bd79
ZooKeeper Client Version	?, revision=12-c9b3def3b445dca9f3ad21427ec3846b81a92453
ZooKeeper Client Compiled	04/28/2021 10:20 GMT
ZooKeeper Quorum	node-master1jfm:2181 node-master2uiqz:2181 node-master3xcpw:2181
ZooKeeper Base Path	/hbase

- b. 将“ZooKeeper Base Path”值配置到URL模板中，即在原有的URL地址后面增加“:/hbase”即可，并单击确定：

图 5-10 配置 URL 模板

**设置**

驱动名称\*: Apache Phoenix 驱动类型: Generic

类名: org.apache.phoenix.jdbc.PhoenixDriver

URL 模板: jdbc:phoenix:{host}[:{port}]/hbase

默认端口: 2181  嵌入  无认证  Allow Empty Password

**描述**

目录: Hadoop ID: phoenix\_hbase

描述: Thin driver for Apache Phoenix HBase Database

网址: <http://phoenix.apache.org/>

7. 配置EIP。如果本地Windows与集群之间网络不通，需要为每个HBase节点以及ZooKeeper节点配置EIP，并且在本地Windows的hosts文件添加所有节点的公网IP对应主机域名的映射关系，例如：

```
100.100.100.100 node-master3xCPw node-master3xCPw.
100.100.100.100 node-group-1ZqBd0001 node-group-1ZqBd0001.
100.100.100.100 node-master2uIQz node-master2uIQz.
100.100.100.100 node-group-1ZqBd0002 node-group-1ZqBd0002.
```

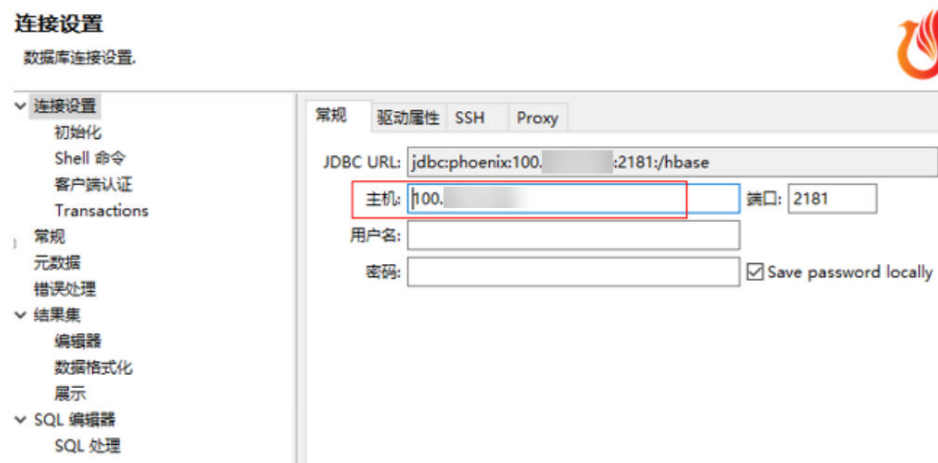
#### 说明

如果使用的是Windows云服务器，并且与集群的网络是通的，则无需配置EIP。

8. 登录FusionInsight Manager，选择“集群 > 服务 > ZooKeeper > 实例”。

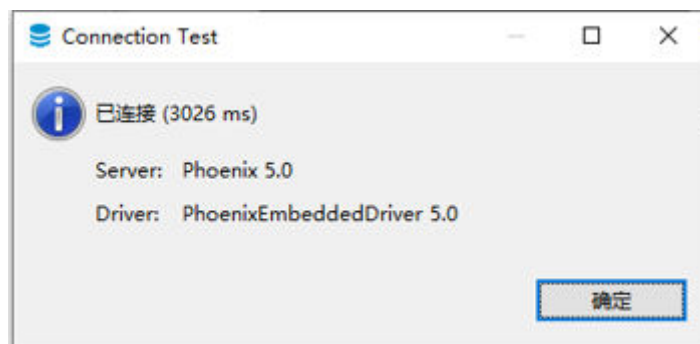
选择任一节点，将该节点对应的EIP填入主机框中（如果使用的是Windows云服务器，并且与集群的网络是通的，直接填写节点的业务IP地址即可）。

图 5-11 配置主机



- 单击“测试连接”，如[图10 测试连接](#)所示表示对接成功，单击“确定”。

图 5-12 测试连接



- 以客户端安装用户登录安装HBase客户端的节点，执行以下命令创建新的命名空间MY\_NS:

```
cd 客户端安装目录
source bigdata_env
hbase shell
create_namespace "MY_NS"
```

### 步骤三：在 DBeaver 验证连接结果

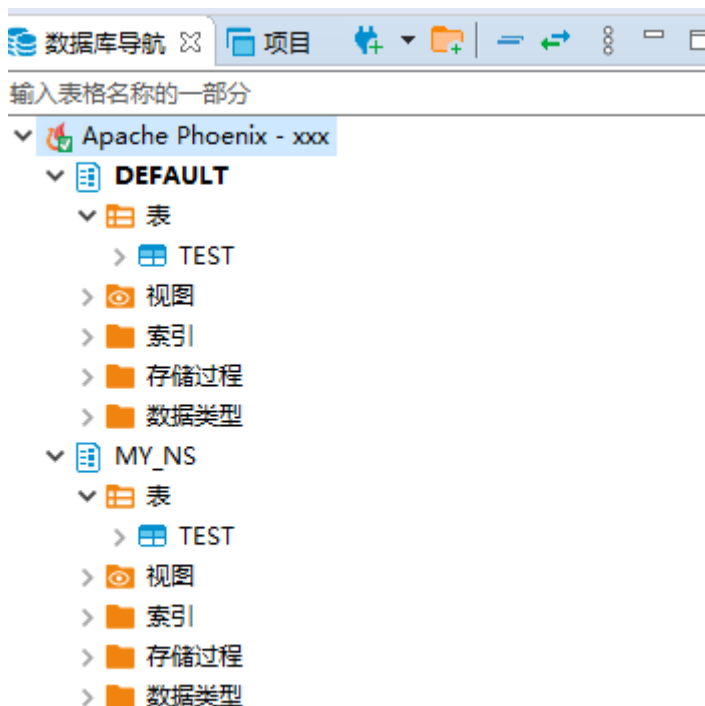
- 打开DBeaver，选择“SQL编辑器 > 新建SQL编辑器”，即可执行相关SQL语句，例如：
  - 在编辑器中输入以下命令并选择“SQL 编辑器 > 执行 SQL 语句”，即可在default中创建表TEST：

```
CREATE TABLE IF NOT EXISTS TEST (id VARCHAR PRIMARY KEY, name VARCHAR);
UPSERT INTO TEST(id,name) VALUES ('1','jamee');
```
  - 在编辑器中输入以下命令并选择“SQL 编辑器 > 执行 SQL 语句”，即可在MY\_NS中创建表TEST并插入数据：

```
CREATE TABLE IF NOT EXISTS MY_NS.TEST (id integer not null
primary key, name varchar);
UPSERT INTO MY_NS.TEST VALUES(1,'John');
UPSERT INTO MY_NS.TEST VALUES(2,'Tom');
UPSERT INTO MY_NS.TEST VALUES(3,'Manson');
UPSERT INTO MY_NS.TEST VALUES(4,'Aurora');
```

2. 右键单击连接名称，单击“刷新”，再单击连接名称左侧的 ▾，即可查看在 default 和 MY\_NS 中新建的表：

图 5-13 查看表



## 5.2 使用 DBeaver 访问 MRS HetuEngine

### 应用场景

DBeaver 是一个 SQL 客户端和数据库管理工具。对于关系数据库，使用 JDBC API 通过 JDBC 驱动程序与数据库交互。对于其他数据库（NoSQL），使用专有数据库驱动程序。

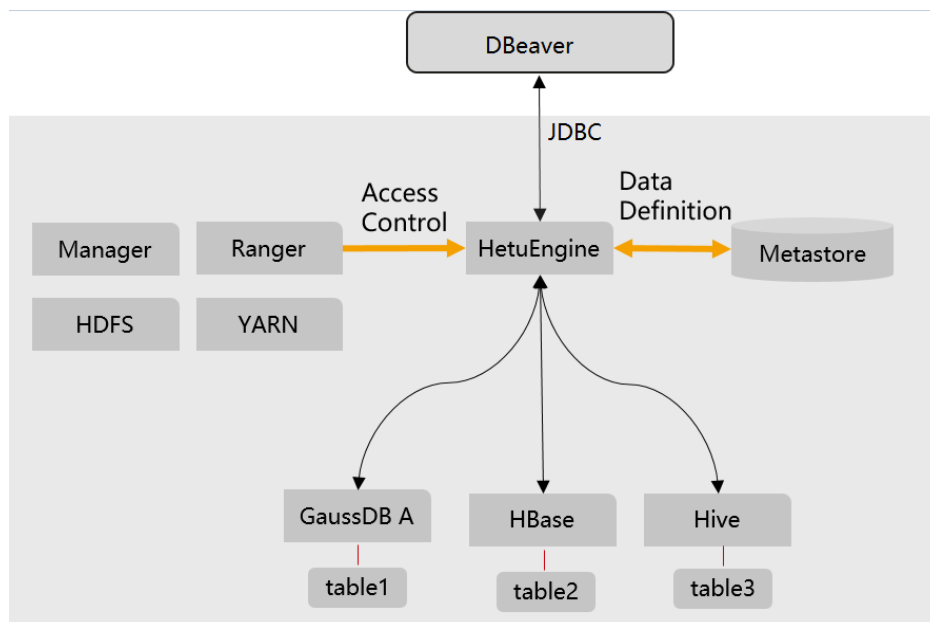
本章节以 DBeaver 7.2.0 版本为例，讲解如何使用 DBeaver 访问 MRS HetuEngine。

### 方案架构

出于管理和信息收集的需要，企业内部会存储海量数据，包括数目众多的各种数据库、数据仓库等，此时会面临数据源种类繁多、数据集结构化混合、相关数据存放分散等困境，导致跨源查询开发成本高，跨源复杂查询耗时长。

HetuEngine提供了统一标准SQL实现跨源协同分析，简化跨源分析操作。

图 5-14 DBeaver 访问 MRS HetuEngine



## 约束与限制

- 已安装DBeaver 7.2.0版本。DBeaver软件下载链接：<https://dbeaver.io/files/7.2.0/>。
- 适用于LTS版的MRS 3.1.2及以后版本集群。

## 步骤一：创建 MRS 集群和计算实例

1. 创建MRS集群。  
创建并购买一个包含HetuEngine组件的MRS集群，详情可参考[创建MRS集群](#)。
2. 创建hetu\_user用户。  
在集群中创建“人机”用户，如hetu\_user，可参考[创建HetuEngine用户](#)。启用Ranger鉴权的集群需根据业务需求为该hetu\_user添加Ranger权限，可参考[添加HetuEngine的Ranger访问权限策略](#)。
3. 创建HetuEngine计算实例。  
创建计算实例并确保运行正常，可参考[创建HetuEngine计算实例](#)。

## 步骤二：获取 JDBC jar 包并配置主机映射

1. 下载HetuEngine客户端获取JDBC jar包。
  - a. 登录FusionInsight Manager。
  - b. 选择“集群 > 服务 > HetuEngine > 概览”。
  - c. 在页面右上角，选择“更多 > 下载客户端”，根据界面提示下载“完整客户端”文件到本地。
  - d. 解压HetuEngine客户端压缩包文件“FusionInsight\_Cluster\_集群ID\_HetuEngine\_Client.tar”获取jdbc文件，并存放在本地，例如“D:\test”。



**说明**

jdbc文件获取方法:

在“FusionInsight\_Cluster\_集群ID\_HetuEngine\_ClientConfig\HetuEngine\xxx\”路径下解压获取“hetu-jdbc-\*.jar”文件。

备注: xxx为“arm”或“x86”。

2. 在本地hosts文件添加主机映射。

根据使用HSFabric方式或HSBroker方式添加对应实例所在主机映射, 格式为: 主机IP 主机名

例如: 192.168.42.90 server-2110081635-0001

**说明**

Windows本地hosts文件存放路径举例: “C:\Windows\System32\drivers\etc”。

### 步骤三: 在 DBeaver 配置访问 HetuEngine 参数

1. 打开DBeaver, 选择“数据库 > 新建连接”, 在“ALL”中搜索“PrestoSQL”并打开PrestoSQL。
2. 单击“编辑驱动设置”, 参考下表信息设置相关参数。

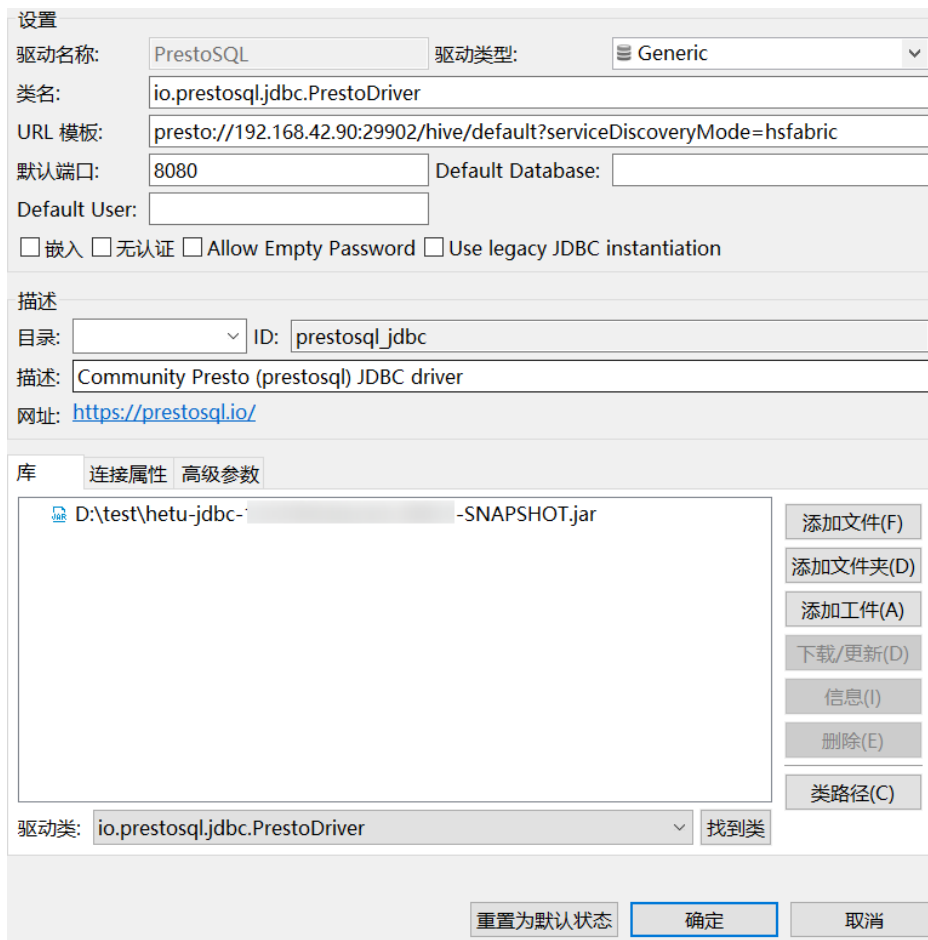
表 5-1 驱动设置信息

参数名	参数值
类名	io.prestosql.jdbc.PrestoDriver
URL模板	<ul style="list-style-type: none"> <li>通过HSFabric方式访问HetuEngine jdbc:presto:// &lt;HSFabricIP1:port1&gt;,&lt;HSFabricIP2:port2&gt;,&lt;HSFabricIP3:port3&gt;/hive/default?serviceDiscoveryMode=hsfabric 示例: jdbc:presto:// 192.168.42.90:29902,192.168.42.91:29902,192.168.42.92:29902/hive/default?serviceDiscoveryMode=hsfabric</li> <li>通过HSBroker方式访问HetuEngine jdbc:presto:// &lt;HSBrokerIP1:port1&gt;,&lt;HSBrokerIP2:port2&gt;,&lt;HSBrokerIP3:port3&gt;/hive/default?serviceDiscoveryMode=hsbroker 示例: jdbc:presto:// 192.168.42.90:29860,192.168.42.91:29860,192.168.42.92:29860/hive/default?serviceDiscoveryMode=hsbroker</li> </ul>

**说明**

- 获取HSFabric、HSBroker节点IP及端口号：
    1. 登录FusionInsight Manager。
    2. 选择“集群 > 服务 > HetuEngine > 实例”，获取HSFabric或HSBroker所有实例的业务IP，可选择一个或多个正常状态的进行连接。
    3. 获取端口号，选择“集群 > 服务 > HetuEngine > 配置 > 全部配置”：
      - 搜索“gateway.port”，获取HSFabric的端口号，安全模式默认为29902，普通模式默认29903；
      - 搜索“server.port”，获取HSBroker的端口号，安全模式默认为29860，普通模式默认29861；
  - 如果连接不成功，请关闭代理重试。
3. 单击“添加文件”，上传**步骤二：获取JDBC jar包并配置主机映射**中获取的JDBC驱动包。
  4. 单击“找到类”，自动获取驱动类，单击“确定”完成驱动设置，如下图所示。如果“库”中存在“io.prestosql:presto-jdbc:RELEASE”，单击“找到类”前需将其删掉。

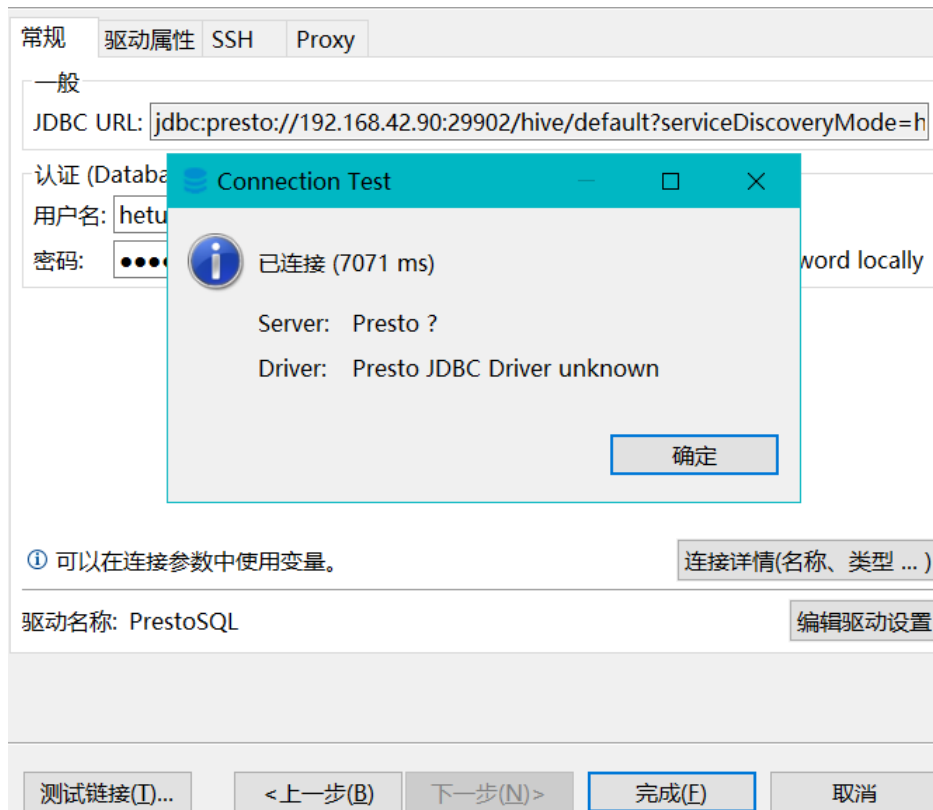
**图 5-15 安全模式驱动设置**



5. 连接设置。
  - 安全模式（开启Kerberos认证的集群）：

在创建新连接页面的“常规”页签，输入2中创建的用户名和密码，单击“测试链接”，连接成功后，单击“确定”，再单击“完成”。可单击“连接详情（名称、类型...）”修改连接名称。

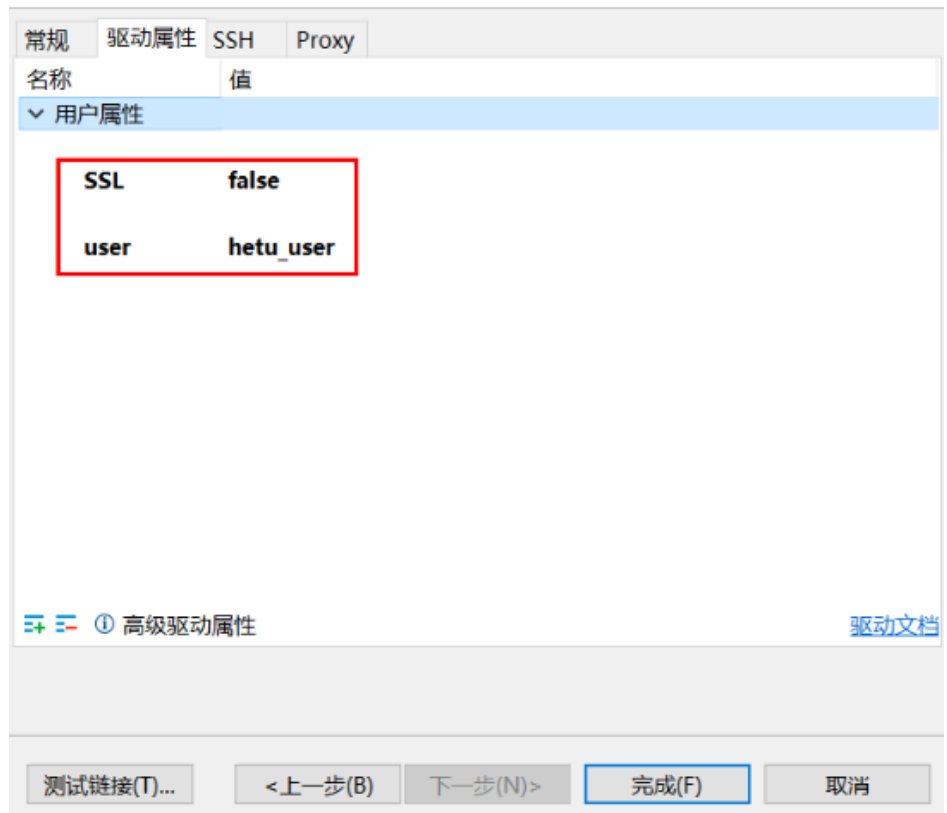
图 5-16 安全模式“常规”参数设置



- 普通模式（未开启Kerberos认证的集群）：

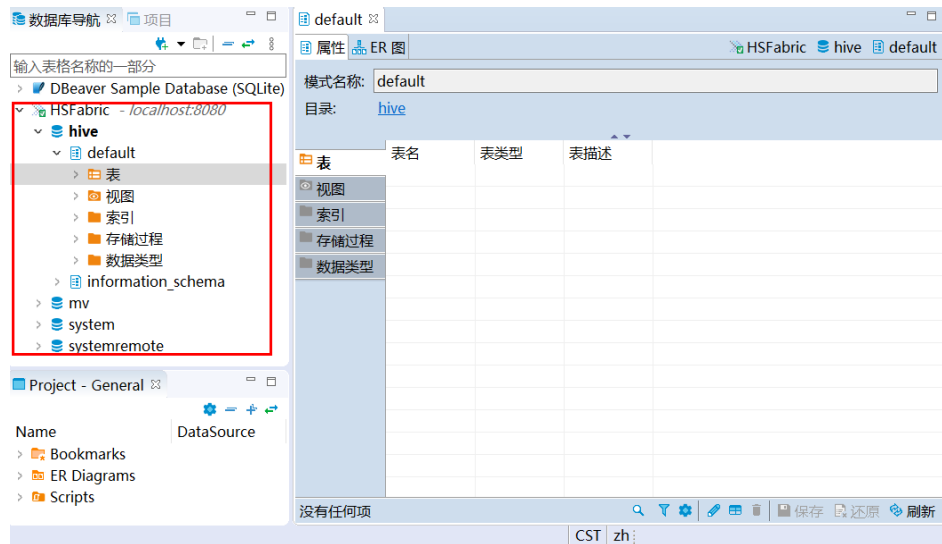
在创建新连接页面的“驱动属性”配置如下参数，“user”为2中创建的用户。单击“测试链接”，连接成功后，单击“确定”，再单击“完成”。可单击“连接详情（名称、类型...）”修改连接名称。

图 5-17 普通模式“驱动属性”参数设置



6. 连接成功后进入下图所示页面。

图 5-18 连接成功



## 5.3 使用 FineBI 访问 MRS HetuEngine

### 应用场景

FineBI是一款商业智能产品，针对企业信息化遇到的困难，为企业提供专业的商业智能解决方案。

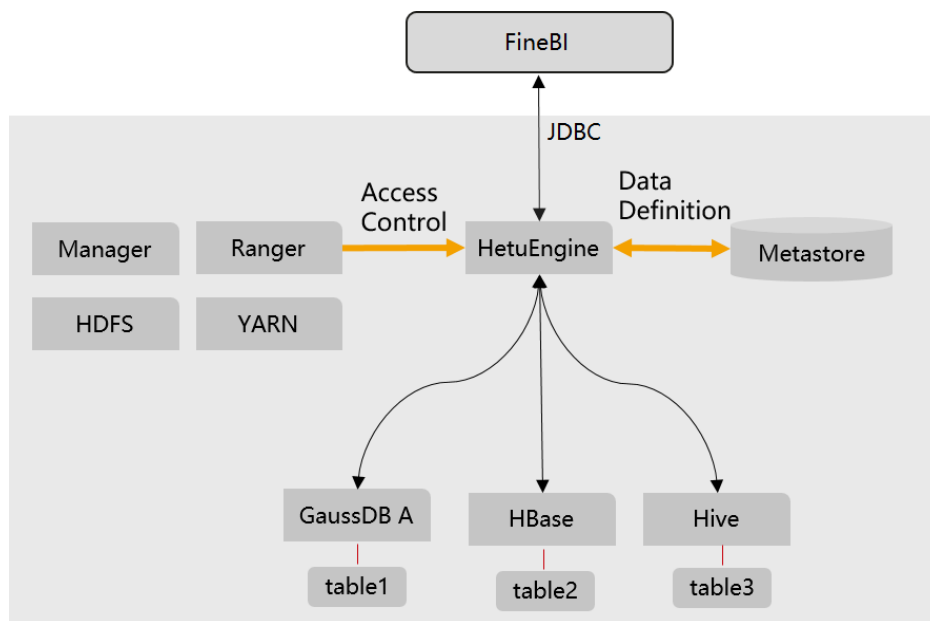
本章节以FineBI 5.1.9版本为例，讲解如何使用FineBI访问安全模式MRS集群的HetuEngine。

### 方案架构

出于管理和信息收集的需要，企业内部会存储海量数据，包括数目众多的各种数据库、数据仓库等，此时会面临数据源种类繁多、数据集结构化混合、相关数据存放分散等困境，导致跨源查询开发成本高，跨源复杂查询耗时长。

HetuEngine提供了统一标准SQL实现跨源协同分析，简化跨源分析操作。

图 5-19 FineBI 访问 MRS HetuEngine



### 约束与限制

- 已安装FineBI 5.1.9版本。
- 适用于LTS版的MRS 3.1.2及以后版本集群。

### 步骤一：创建 MRS 集群和计算实例

1. 创建MRS集群。  
创建并购买一个包含HetuEngine组件的MRS集群，详情可参考[创建MRS集群](#)。
2. 创建hetu\_user用户。

在集群中创建“人机”用户，如`hetu_user`，可参考[创建HetuEngine用户](#)。启用Ranger鉴权的集群需根据业务需求为该`hetu_user`添加Ranger权限，可参考[添加HetuEngine的Ranger访问权限策略](#)。

### 3. 创建HetuEngine计算实例。

创建计算实例并确保运行正常，可参考[创建HetuEngine计算实例](#)。

## 步骤二：获取 JDBC jar 包

1. 登录FusionInsight Manager。
2. 选择“集群 > 服务 > HetuEngine > 概览”。
3. 在页面右上角，选择“更多 > 下载客户端”，根据界面提示下载“完整客户端”文件到本地。
4. 解压HetuEngine客户端压缩包文件“FusionInsight\_Cluster\_集群ID\_HetuEngine\_Client.tar”获取jdbc文件，并存放在本地，例如“D:\test”。

### 📖 说明

jdbc文件获取方法：

在“FusionInsight\_Cluster\_集群ID\_HetuEngine\_ClientConfig\HetuEngine\xxx\”路径下解压获取“hetu-jdbc-\*.jar”文件。

备注：xxx为“arm”或“x86”。

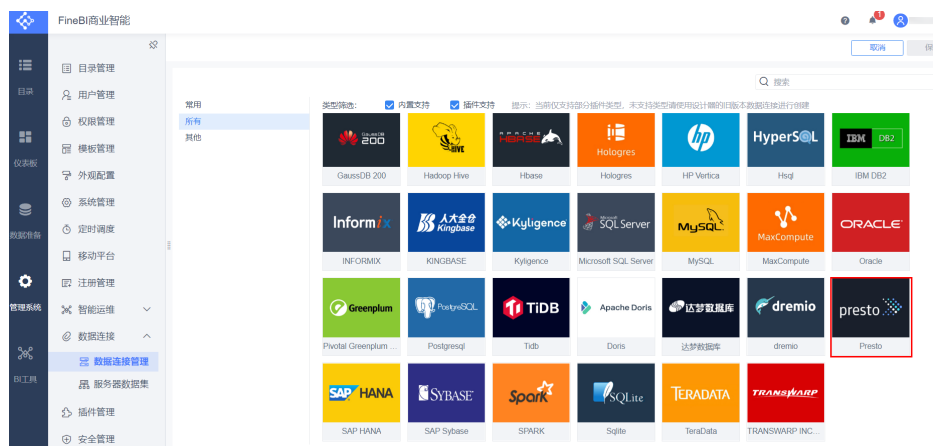
## 步骤三：在 FineBI 配置访问 HetuEngine 参数

**步骤1** 请自行准备并安装FineBI至本地。

**步骤2** 将[步骤二：获取JDBC jar包](#)获取的jar包放在FineBI目录“%FineBI%\webapps\webroot\WEB-INF\lib”，重启FineBI。

**步骤3** 打开FineBI，选择“管理系统 > 数据连接 > 数据连接管理 > 新建数据连接 > 所有”，选择“Presto”并打开。参考[图5-20](#)新建连接。

图 5-20 新建连接



**步骤4** 参考下表配置连接参数。配置完成后单击“测试连接”，或在“模式”单击“点击连接数据库”测试数据连接，测试成功后单击“保存”。

表 5-2 HSFabric 连接参数

参数名	参数值
数据连接名称	自定义
驱动	io.prestosql.jdbc.PrestoDriver
数据库名称	hive/default?serviceDiscoveryMode=hsfabric
主机	hsfabric实例所在节点IP
端口	HSFabric服务gateway.port端口
用户名	已创建的“人机”用户的用户名，如：admintest
密码	已创建的“人机”用户的用户密码 <b>说明</b> <ul style="list-style-type: none"><li>使用用户名密码方式登录时需要配置该参数。</li><li>未启用Kerberos认证（普通模式）的集群不填写该参数。</li></ul>
编码	自动
数据库连接URL	<ul style="list-style-type: none"><li>启用Kerberos认证（安全模式）的集群 jdbc:presto:// &lt;HSFabricIP1:port1&gt;,&lt;HSFabricIP2:port2&gt;,&lt;HSFabricIP3:port3&gt;/hive/default?serviceDiscoveryMode=hsfabric，详情请参考表5-1。</li><li>未启用Kerberos认证（普通模式）的集群 jdbc:presto:// &lt;HSFabricIP1:port1&gt;,&lt;HSFabricIP2:port2&gt;,&lt;HSFabricIP3:port3&gt;/hive/default? serviceDiscoveryMode=hsfabric&amp;SSL=false。</li></ul>

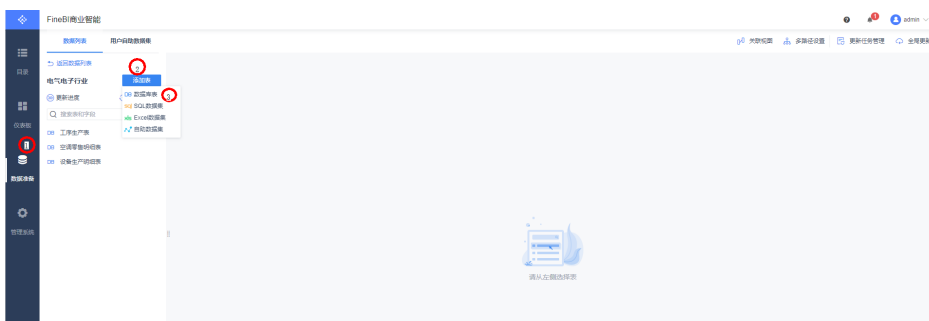
表 5-3 HSbroker 连接参数

参数名	参数值
数据连接名称	自定义
驱动	io.prestosql.jdbc.PrestoDriver
数据库名称	hive/default?serviceDiscoveryMode=hsbroker
主机	hsbroker实例所在节点ip
端口	Hsbroker服务端口
用户名	已创建的“人机”用户的用户名，如：admintest
密码	已创建的“人机”用户的用户密码 <b>说明</b> <ul style="list-style-type: none"><li>使用用户名密码方式登录时需要配置该参数。</li><li>未启用Kerberos认证（普通模式）的集群不填写该参数。</li></ul>

参数名	参数值
编码	自动
数据库连接URL	<ul style="list-style-type: none"> <li>启用Kerberos认证（安全模式）的集群 jdbc:presto:// &lt;HSBrokerIP1:port1&gt;,&lt;HSBrokerIP2:port2&gt;,&lt;HSBrokerIP3:port3&gt;/hive/default?serviceDiscoveryMode=hsbroker, 详情请参考表5-1。</li> <li>未启用Kerberos认证（普通模式）的集群 jdbc:presto:// &lt;HSBrokerIP1:port1&gt;,&lt;HSBrokerIP2:port2&gt;,&lt;HSBrokerIP3:port3&gt;/hive/default? serviceDiscoveryMode=hsbroker&amp;SSL=false</li> </ul>

**步骤5** 参考图5-21所示配置数据库表，选择“数据准备 > 数据列表”，单击“添加分组”，选择“添加表 > 数据库表”。

图 5-21 配置数据



**步骤6** 设置需要用于做分析的表，如图5-22~图5-24所示。

图 5-22 单击“数据连接”

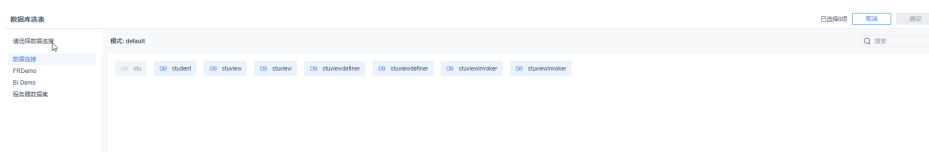


图 5-23 选择数据库

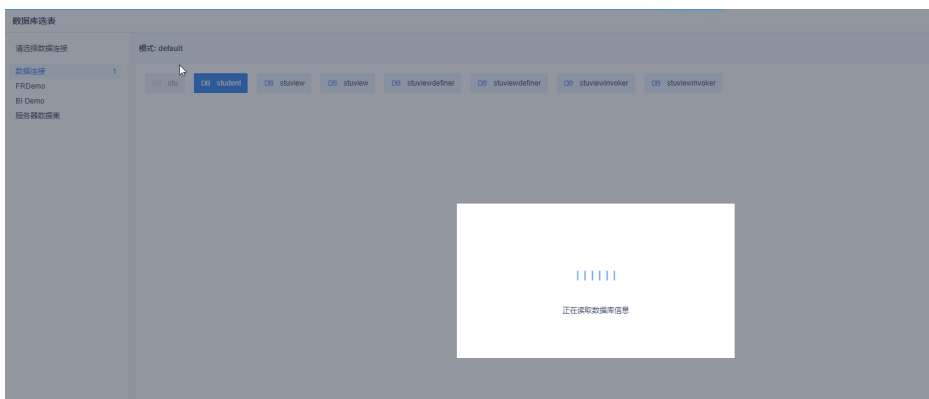
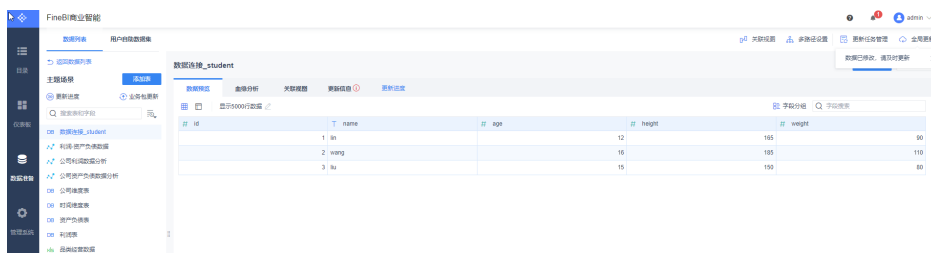




图 5-24 数据预览



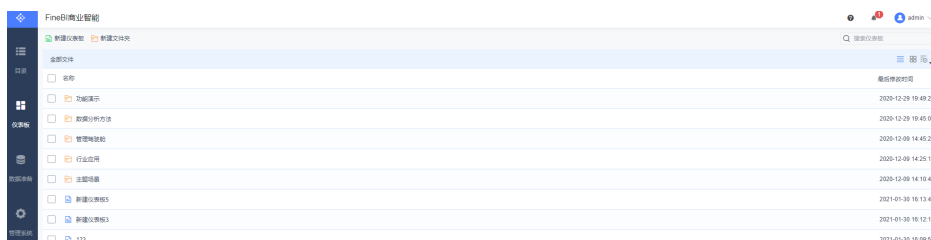
步骤7 单击“更新信息”中的“单表更新”，进行数据同步。

图 5-25 数据同步



步骤8 单击“仪表盘”，单击“新建仪表盘”，输入相关名称单击“确定”。

图 5-26 新建仪表盘



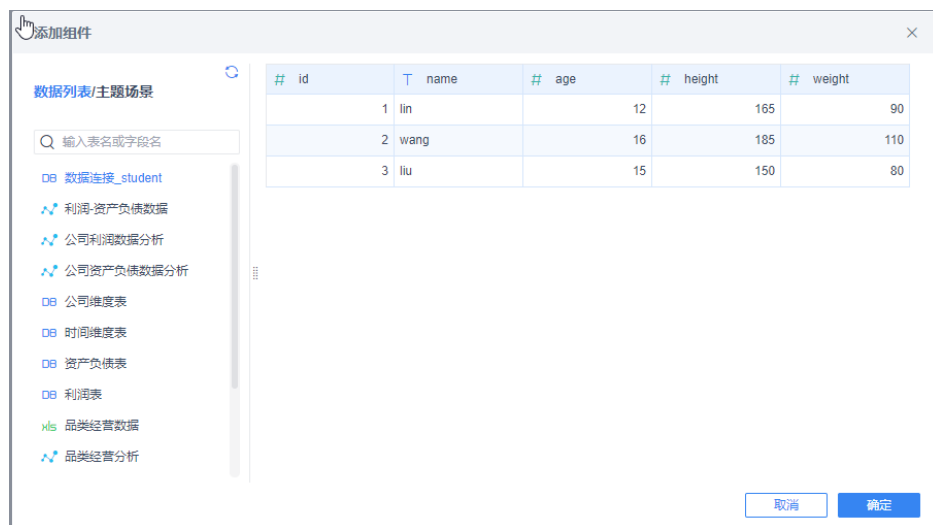
步骤9 单击“添加组件”。

图 5-27 添加组件



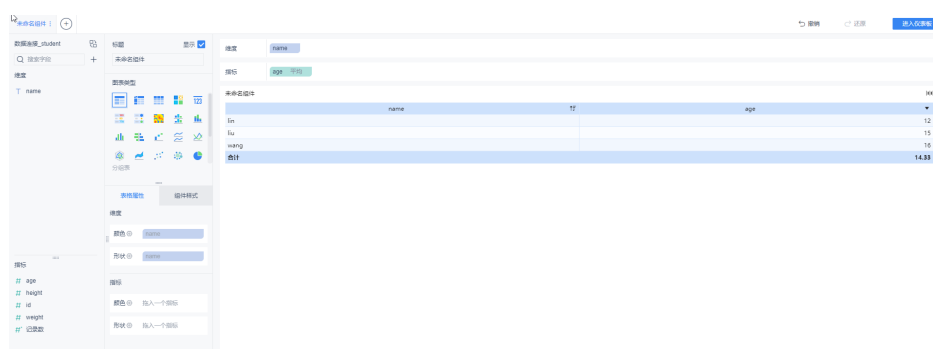
步骤10 添加步骤6配置的需要用于分析的数据表。

图 5-28 添加数据表



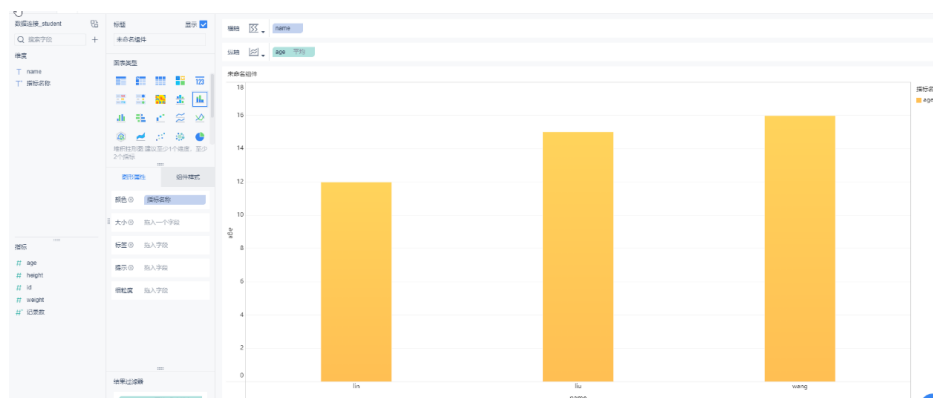
步骤11 将“name”拖入“维度”，将“age”拖入“指标”，即可分析年龄的平均值。如图 5-29 所示。

图 5-29 分析表



步骤12 如果需要用图显示，则可在“图表类型”中选择相对应的图。样例中是选择“柱状图”。

图 5-30 选择图表类型



----结束

## 5.4 使用 Tableau 访问 MRS HetuEngine

### 应用场景

Tableau是一款商业智能工具软件，将可信的数据转化为可行的见解。借助直观的人工智能分析平台，帮助人们查看和理解数据并根据数据采取行动，让每一次决策都更加明智。

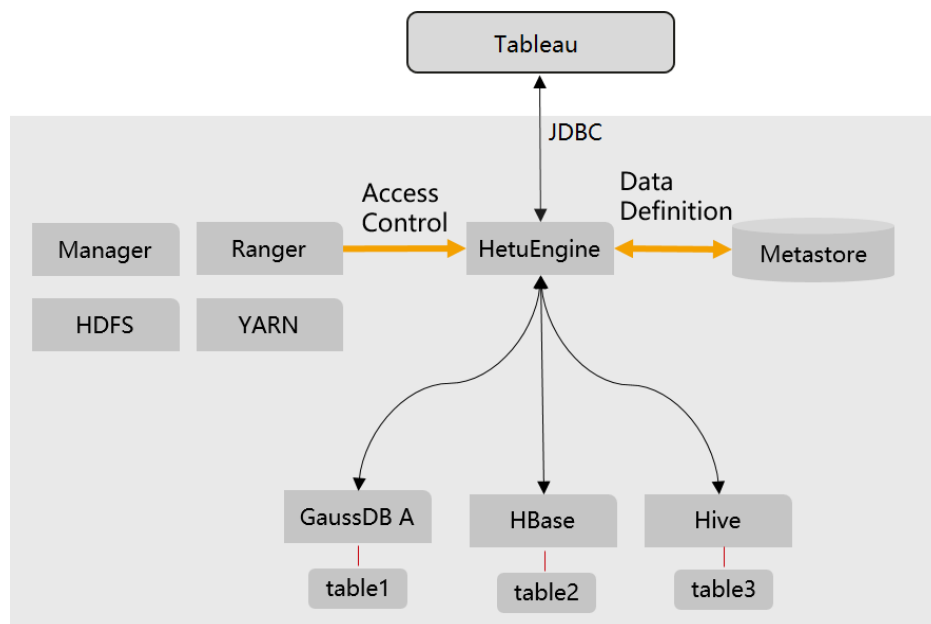
本章节以Tableau Desktop 2022.2版本为例，讲解如何使用Tableau访问安全模式集群的HetuEngine。

### 方案架构

出于管理和信息收集的需要，企业内部会存储海量数据，包括数目众多的各种数据库、数据仓库等，此时会面临数据源种类繁多、数据集结构化混合、相关数据存放分散等困境，导致跨源查询开发成本高，跨源复杂查询耗时长。

HetuEngine提供了统一标准SQL实现跨源协同分析，简化跨源分析操作。

图 5-31 Tableau 访问 MRS HetuEngine



### 约束与限制

- 已安装Tableau Desktop 2022.2版本。
- 适用于LTS版的MRS 3.1.2及以后版本集群。

### 步骤一：创建 MRS 集群和计算实例

1. 创建MRS集群。  
创建并购买一个包含HetuEngine组件的MRS集群，详情可参考[创建MRS集群](#)。
2. 创建hetu\_user用户。

在集群中创建“人机”用户，如`hetu_user`，可参考[创建HetuEngine用户](#)。启用Ranger鉴权的集群需根据业务需求为该`hetu_user`添加Ranger权限，可参考[添加HetuEngine的Ranger访问权限策略](#)。

3. 创建HetuEngine计算实例。  
创建计算实例并确保运行正常，可参考[创建HetuEngine计算实例](#)。

## 步骤二：获取 JDBC jar 包

1. 下载HetuEngine客户端获取JDBC jar包。
  - a. 登录FusionInsight Manager。
  - b. 选择“集群 > 服务 > HetuEngine > 概览”。
  - c. 在页面右上角，选择“更多 > 下载客户端”，根据界面提示下载“完整客户端”文件到本地。
  - d. 解压HetuEngine客户端压缩包文件“FusionInsight\_Cluster\_集群ID\_HetuEngine\_Client.tar”获取jdbc文件，并存放在本地，例如“D:\test”。

### 说明

jdbc文件获取方法：

在“FusionInsight\_Cluster\_集群ID\_HetuEngine\_ClientConfig\HetuEngine\xxx\”路径下解压获取“hetu-jdbc-\*.jar”文件。

备注：xxx为“arm”或“x86”。

2. 将获取的Jar包放在Tableau安装目录，如“C:\Program Files\Tableau\Drivers”。

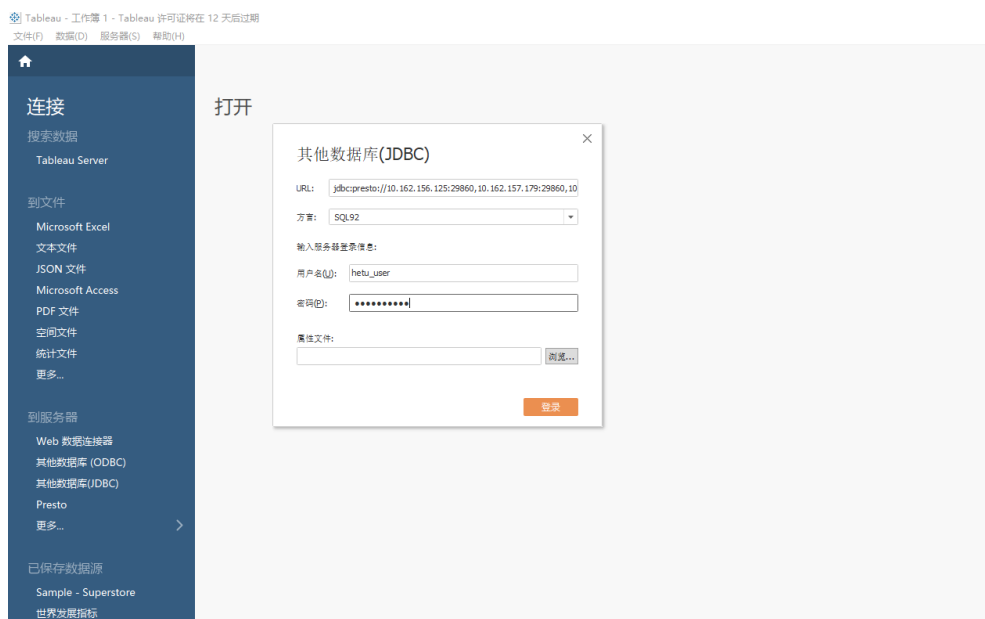
## 步骤三：在 Tableau 配置访问 HetuEngine 参数

1. 打开Tableau。
2. 选择“到服务器 > 其他数据库(JDBC)”，输入URL和已创建的“人机”用户的用户名及密码，单击“登录”。

### 说明

支持HSFabric方式和HSBroker方式连接，URL格式详情可参考[表5-1](#)。

图 5-32 使用 Tableau 访问 HetuEngine



3. 登录成功后，将要操作的数据表拖到右边操作窗口，刷新数据。

## 5.5 使用永洪 BI 访问 MRS HetuEngine

### 应用场景

永洪BI是一款一站式大数据BI平台，全面覆盖数据分析过程中的各个环节，轻松完成全流程数据分析任务，包括数据采集、清洗、整合、存储、计算、建模、训练、展现、协作等，极大降低了实施、集成、培训的成本。

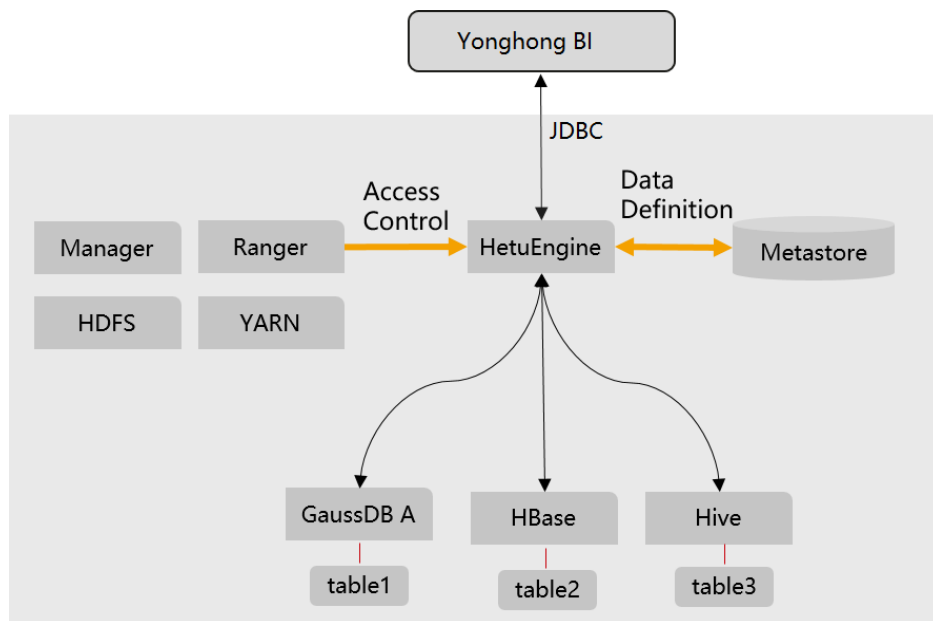
本章节以Yonghong Desktop 9.1版本为例，讲解如何使用永洪BI访问安全模式集群的HetuEngine。

### 方案架构

出于管理和信息收集的需要，企业内部会存储海量数据，包括数目众多的各种数据库、数据仓库等，此时会面临数据源种类繁多、数据集结构化混合、相关数据存放分散等困境，导致跨源查询开发成本高，跨源复杂查询耗时长。

HetuEngine提供了统一标准SQL实现跨源协同分析，简化跨源分析操作。

图 5-33 永洪 BI 访问 MRS HetuEngine



### 约束与限制

- 已安装Yonghong Desktop 9.1版本。
- 适用于LTS版的MRS 3.1.2及以后版本集群。

### 步骤一：创建 MRS 集群和计算实例

1. 创建MRS集群。  
创建并购买一个包含HetuEngine组件的MRS集群，详情可参考[创建MRS集群](#)。

2. 创建hetu\_user用户。  
在集群中创建“人机”用户，如hetu\_user，可参考[创建HetuEngine用户](#)。启用Ranger鉴权的集群需根据业务需求为该hetu\_user添加Ranger权限，可参考[添加HetuEngine的Ranger访问权限策略](#)。
3. 创建HetuEngine计算实例。  
创建计算实例并确保运行正常，可参考[创建HetuEngine计算实例](#)。

## 步骤二：获取 JDBC jar 包

1. 登录FusionInsight Manager。
2. 选择“集群 > 服务 > HetuEngine > 概览”。
3. 在页面右上角，选择“更多 > 下载客户端”，根据界面提示下载“完整客户端”文件到本地。
4. 解压HetuEngine客户端压缩包文件“FusionInsight\_Cluster\_集群ID\_HetuEngine\_Client.tar”获取jdbc文件，并存放在本地，例如“D:\test”。

### 说明

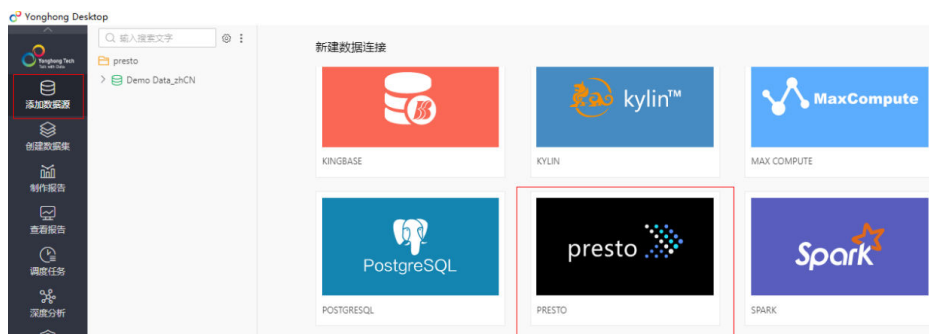
jdbc文件获取方法：

在“FusionInsight\_Cluster\_集群ID\_HetuEngine\_ClientConfig\HetuEngine\xxx\”路径下解压获取“hetu-jdbc-\*.jar”文件。

备注：xxx为“arm”或“x86”。

## 步骤三：在 Yonghong 配置访问 HetuEngine 参数

1. 打开Yonghong Desktop，选择“添加数据源 > presto”。



2. 在数据源配置页面参考下图完成参数配置，“用户名”和“密码”为已创建的“人机”用户的用户名和用户密码。配置完成后可以单击“测试连接”测试。

图 5-34 数据源配置

数据库

选择数据源: PRESTO \*

连接属性

驱动: 自定义 | prestoDriver (io.prestosql.jdbc.PrestoDriver) \* 选择自定义驱动

URL: jdbc:presto://192.168.8.37:29860, 192.168.8.38:29860/hive/default?serviceDiscoveryMoc \*

服务器登录: 用户名和密码 \*

用户名: admintest \*

密码: .....


数据库:

表结构模式:



+ 添加基础属性 新增的是连接前的属性

高级属性

测试连接


- 驱动: 选择“自定义 > 选择自定义驱动”，单击 ，编辑驱动名称，单击“上传文件”上传已获取的JDBC jar包，单击“确定”。


驱动管理

驱动列表  

设置驱动信息

名称:

驱动: 自定义 | 

上传驱动列表 

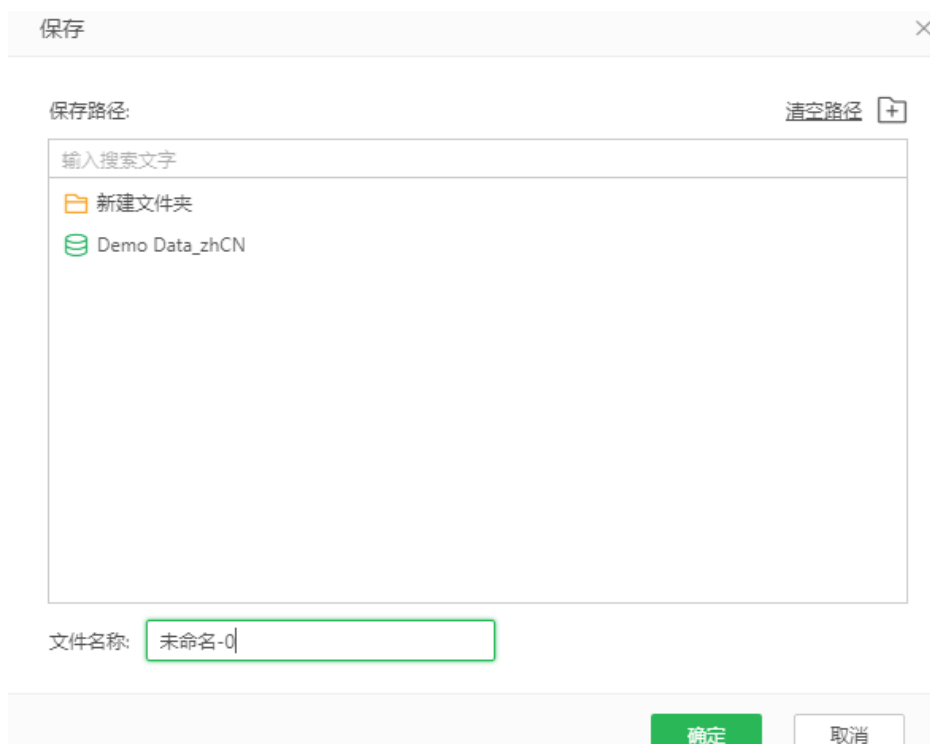
请点击添加驱动，然后再点击上传按钮

版本信息

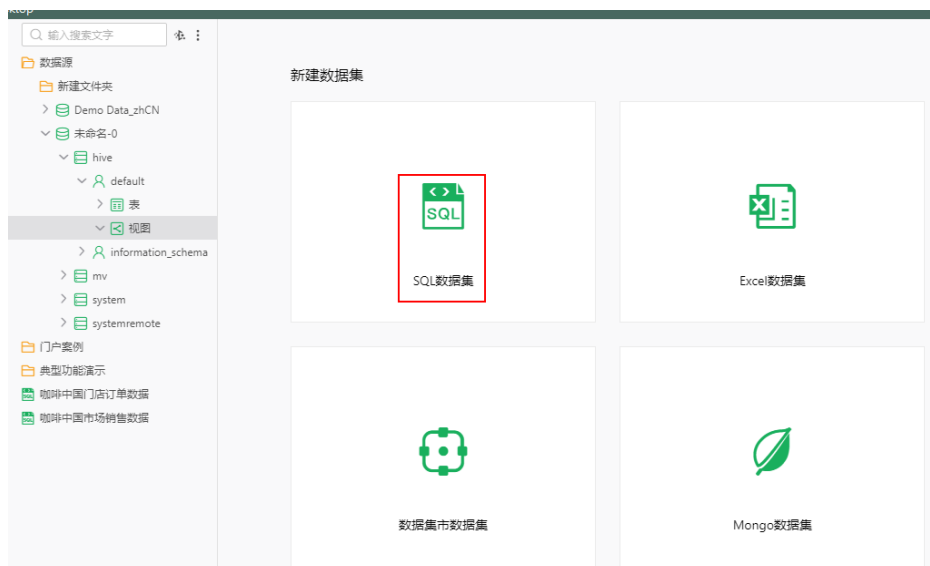
确定 取消

- URL: 支持HSFabric方式和HSBroker方式，详情请参考表5-1。
  - 服务器登录: 选择“用户名和密码”并填写相应的用户名及密码。
3. 单击“新建数据集”，在弹出的页面参考下图修改保存路径及文件名称，单击“确定”保存修改路径及文件名称。

图 5-35 修改路径及名称

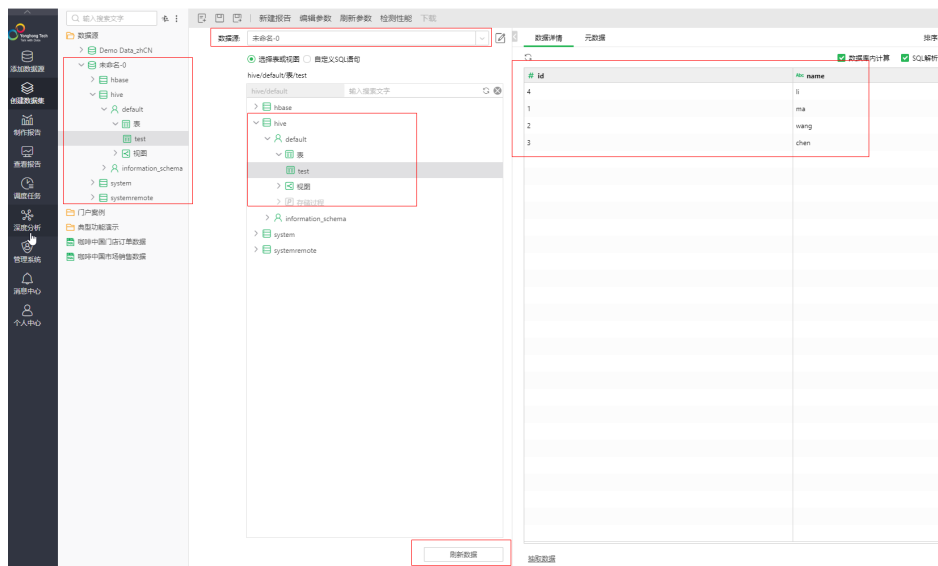


4. 在“数据源”选择新建的数据集的文件名称，此处以默认文件名称“未命名-0”为例，选择“未命名-0 > hive > default > 视图”，在右侧“新建数据集”选择“SQL数据集”。



5. 在“数据源”处选择新建的数据集，显示所有表信息，选中其中一个表，如“test”表，单击“刷新数据”，可在右侧“数据详情”中显示表的所有信息。





## 5.6 Hive 对接外置自建关系型数据库

### 应用场景

在已有Hive数据的集群上外置元数据库后，之前的元数据表不会自动同步。因此在安装Hive之初就要确认好元数据是外置数据库还是内置到DBService，如果是外置自建数据库，则需在安装Hive时或者暂无Hive数据时将元数据外置，安装后不允许修改，否则将会造成原有元数据丢失。

Hive支持开源MySQL和Postgres元数据库，本章节以对接开源MySQL和Postgres数据库进行说明。

### 约束与限制

- 当外置元数据到MySQL后，Hive仅表名、字段名、表描述支持中文，其余暂不支持。
- 适用于MRS 3.x及以后版本。

### 步骤一：安装数据库并上传驱动包

1. 安装开源MySQL或Postgres数据库。

#### 📖 说明

数据库安装节点需与集群处于同一网段，能互相访问。

2. 上传驱动包。

#### – Postgres:

使用开源驱动包替换集群已有的驱动包。将Postgres驱动包“postgresql-42.2.5.jar”上传至所有MetaStore实例节点“\${BIGDATA\_HOME}/third\_lib/Hive”目录下（开源驱动包下载地址：<https://repo1.maven.org/maven2/org/postgresql/postgresql/42.2.5/>）。

在上传驱动包的所有MetaStore实例节点上执行以下命令修改驱动包权限：

```
cd ${BIGDATA_HOME}/third_lib/Hive
```

```
chown omm:wheel postgresql-42.2.5.jar
```

```
chmod 600 postgresql-42.2.5.jar
```

- MySQL:

进入MySQL官网 ( <https://www.mysql.com/> )，选择“DOWNLOADS > MySQL Community(GPL) DownLoads > Connector/J”下载对应版本的驱动包。

- MRS 8.2.0之前版本，将MySQL对应版本的驱动包上传至所有Metastore实例节点“/opt/Bigdata/FusionInsight\_HD\_\*/install/FusionInsight-Hive-\*/hive-\*/lib/”目录下。

- MRS 8.2.0及之后版本，将MySQL对应版本的驱动包上传至所有Metastore实例节点“\${BIGDATA\_HOME}/third\_lib/Hive”目录下。

在上传驱动包的所有MetaStore实例节点上执行以下命令修改驱动包权限:

```
cd /opt/Bigdata/FusionInsight_HD_*/install/FusionInsight-Hive-*/hive-*/lib/
```

```
chown omm:wheel mysql-connector-java-*.jar
```

```
chmod 600 mysql-connector-java-*.jar
```

## 步骤二：在自建数据库中建表及赋权

1. 在自建数据库中创建用户、元数据库，并为用户赋予该库的所有权限。例如：

- 以数据库管理员用户在Postgres中执行以下命令创建数据库“test”和用户“testuser”，并授予“test”数据库的所有权限给“testuser”用户。

```
create user testuser with password 'password';
```

```
create database test owner testuser;
```

```
grant all privileges on database test to testuser;
```

- 以数据库管理员用户在MySQL中执行以下命令创建数据库“test”和用户“testuser”，并授予“test”数据库的所有权限给“testuser”用户。

```
create database test;
```

```
create user 'testuser'@'%' identified by 'password';
```

```
grant all privileges on test.* to 'testuser';
```

```
flush privileges;
```

2. 导入元数据建表SQL。

- Postgres的SQL文件路径：\${BIGDATA\_HOME}/FusionInsight\_HD\_\*/install/FusionInsight-Hive-\*/hive-\*/scripts/metastore/upgrade/postgres/hive-schema-3.1.0.postgres.sql

Postgres导入sql文件的命令：

```
./bin/psql -U username -d databasename -f hive-schema-3.1.0.postgres.sql
```

其中：

./bin/psql：在Postgres安装目录下。

username：登录Postgres的用户名。

databasename：数据库库名。

- MySQL的SQL文件路径: `${BIGDATA_HOME}/FusionInsight_HD_*/install/FusionInsight-Hive-*/hive-*/scripts/metastore/upgrade/mysql/hive-schema-3.1.0.mysql.sql`

MySQL导入sql文件的命令:

```
./bin/mysql -u username -p -D databasename<hive-schema-3.1.0.mysql.sql
```

其中:

`./bin/mysql`: 在MySQL安装目录下。

`username`: 登录MySQL的用户名。

`databasename`: 数据库库名。

### 步骤三：在 MRS 集群配置参数对接数据库

1. 登录FusionInsight Manager，选择“集群 > 服务 > Hive > 配置 > 全部配置 > Hive（服务） > MetaDB”，修改以下参数并保存，使Hive的配置对接到开源数据库。

表 5-4 参数说明

参数名	默认值	描述
<code>javax.jdo.option.ConnectionDriverName</code>	<code>org.postgresql.Driver</code>	Metastore上连接元数据的驱动类。 <ul style="list-style-type: none"> <li>• 外置MySQL，则值为：<code>com.mysql.jdbc.Driver</code></li> <li>• 外置Postgres，则值为：<code>org.postgresql.Driver</code></li> </ul>
<code>javax.jdo.option.ConnectionURL</code>	<code>jdbc:postgresql://%{DBSERVICE_FLOAT_IP}%{DBServer}:%{DBSERVICE_CPORT}/hivemeta?socketTimeout=60</code>	Metastore元数据JDBC链接的URL。 <ul style="list-style-type: none"> <li>• 外置MySQL，则值为：<code>jdbc:mysql://MySQL的IP.MySQL的端口/test?characterEncoding=utf-8</code></li> <li>• 外置Postgres，则值为：<code>jdbc:postgresql://Postgres的IP.Postgres的端口号/test</code></li> </ul> <p><b>说明</b> “test”为1中在MySQL或PgSQL中创建的数据库名称。</p>

参数名	默认值	描述
javax.jdo.option.ConnectionUserName	hive\${SERVICE_INDEX}\${SERVICE_INDEX}	Metastore上连接外置元数据的数据库用户名。

- 在MetaStore中修改Postgres数据库密码，选择“集群 > 服务 > Hive > 配置 > 全部配置 > MetaStore（角色） > MetaDB”，修改以下参数并保存。

表 5-5 参数说明

参数名	默认值	描述
javax.jdo.option.extend.ConnectionPassword	*****	Metastore上连接外置元数据的数据库用户密码。密码后台会加密。

- 登录所有MetaStore服务的后台节点，检查本地目录“/opt/Bigdata/tmp”是否存在。
  - 存在，直接执行4。
  - 不存在，则先执行以下命令，创建目录。  
**mkdir -p /opt/Bigdata/tmp**  
**chmod 755 /opt/Bigdata/tmp**
- 保存配置，选择“概览 > 更多 > 重启服务”，输入密码开始重启Hive服务。
- Hive重启完成后，登录MySQL或Postgres数据库，可以查看到1创建的元数据库中元数据表生成：

```
-----+
Tables_in_hivemeta
-----+
aux_table
bucketing_cols
cds
columns_v2
compaction_queue
completed_compactions
completed_txn_components
ctlgs
database_params
db_privs
dbs
delegation_tokens
```

#### 步骤四：验证 Hive 元数据库是否外置成功

- 以客户端安装用户登录安装Hive客户端的节点：  
**cd 客户端安装目录**  
**source bigdata\_env**  
**kinit 组件业务用户**（未开启Kerberos认证的集群请跳过该操作）
- 执行以下命令登录Hive客户端命令行：

**beeline**

3. 执行以下命令创建表**test**:

```
create table test(id int,str1 string,str2 string);
```

4. 在MySQL或Postgres的**test**数据库中执行以下命令查看是否有表**test**相关信息:

```
select * from TBLS;
```

能查看到表**test**相关信息则说明外置Hive数据库成功，例如:

- 在MySQL中查看的结果为:

```
mysql> mysql> select * from TBLS;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| TBL_ID | CREATE_TIME | DB_ID | LAST_ACCESS_TIME | OWNER | OWNER_TYPE | RETENTION | SD_ID | TBL_NAME | TBL_TYPE | VIEW_EXPANDED_TEXT | VIEW_ORIGINAL_TEXT | IS_REWRITE_ENABLED |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 6 | 1673413291 | 1 | 0 | root | USER | 0 | 6 | test1 | MANAGED_TABLE | NULL | NULL | 0 |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

- 在Postgres中查看的结果为:

```
hive> select * from "TBLS";
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| TBL_ID | CREATE_TIME | DB_ID | LAST_ACCESS_TIME | OWNER | OWNER_TYPE | RETENTION | SD_ID | TBL_NAME | TBL_TYPE | VIEW_EXPANDED_TEXT | VIEW_ORIGINAL_TEXT | IS_REWRITE_ENABLED |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 2 | 1673425195 | 1 | 0 | root | USER | 0 | 2 | test1 | MANAGED_TABLE |  |  | 0 |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
(1 row)
```

## 5.7 MRS Hive 对接外部 LDAP 配置说明

### 应用场景

本章节介绍Hive如何对接外部LDAP并访问HiveServer。

为了管理集群中数据与资源的访问控制权限，在安全模式下，客户端应用程序在访问集群中的任意资源之前均需要通过身份认证，建立安全会话链接。

MRS通过KrbServer为所有组件提供Kerberos认证功能，实现了可靠的认证机制。

LdapServer支持轻量目录访问协议（Lightweight Directory Access Protocol，简称为LDAP），为Kerberos认证提供用户和用户组数据保存能力。

### 方案架构

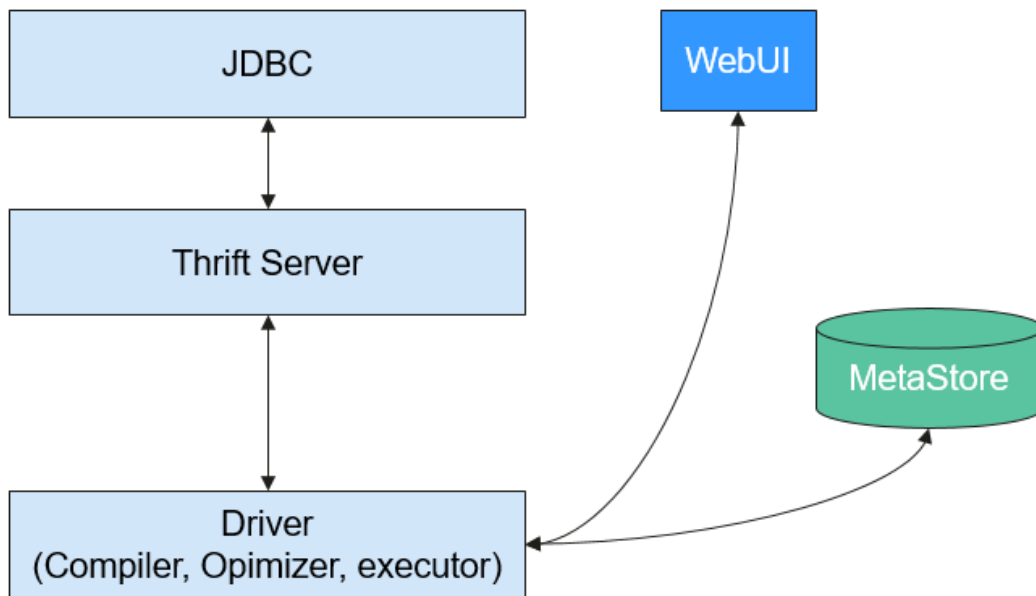
Hive是建立在Hadoop上的数据仓库框架，提供大数据平台批处理计算能力，能够对结构化/半结构化数据进行批量分析汇总完成数据计算。提供类似SQL的Hive Query Language语言操作结构化数据，其基本原理是将HQL语言自动转换成MapReduce任务，从而完成对Hadoop集群中存储的海量数据进行查询和分析。

Hive主要特点如下:

- 海量结构化数据分析汇总。
- 将复杂的MapReduce编写任务简化为SQL语句。
- 灵活的数据存储格式，支持JSON、CSV、TEXTFILE、RCFILE、SEQUENCEFILE、ORC等存储格式。

Hive作为一个基于HDFS和MapReduce架构的数据仓库，其主要能力是通过对HQL（Hive Query Language）编译和解析，生成并执行相应的MapReduce任务或者HDFS操作。

图 5-36 Hive 结构



### 约束与限制

- 需要在Manager的“系统 > 权限 > 用户”界面上创建一个LDAP中已存在的同名用户，并添加“hive”和“hadoop”用户组，该用户用于HiveServer的健康检查。
- 本章节配置说明适用于MRS 3.1.0及之后版本。

### 步骤一：配置 LDAP 认证

1. 登录Manager界面，选择“集群 > 服务 > Hive > 配置 > 全部配置 > HiveServer（角色） > 安全”，配置以下参数：

表 5-6 配置 LDAP 认证参数

参数名称	参数描述	参数取值示例
hive.server2.authentication	用于指定 HiveServer 的认证方式，取值范围为“KERBEROS”或“LDAP”，需设置为“LDAP”。	LDAP
hive.server2.authentication.ldap.baseDN	LDAP服务用户所在的Base DN。	-

参数名称	参数描述	参数取值示例
hive.server2.authentication.ldap.password	<b>约束与限制</b> 中创建的与LDAP中同名的用户密码，即 HiveServer健康检查所使用的用户对应的密码。	-
hive.server2.authentication.ldap.url.ip	仅适用于MRS 3.5.0之前版本，用于指定LDAP IP地址。	-
hive.server2.authentication.ldap.url.port	仅适用于MRS 3.5.0之前版本，用于指定LDAP 端口，默认值为 389。	389
hive.server2.authentication.ldap.url	<p>仅适用于MRS 3.5.0及之后版本，用于指定连接LDAP服务的 URL，格式为： <b>ldap://\$ {ldap_host}:\$ {ldap_port}。</b></p> <p><b>说明</b></p> <ul style="list-style-type: none"> <li>• <code>{ldap_host}</code> 为LDAP服务的主机名或IP地址，<code>{ldap_port}</code> 为LDAP服务的的端口号。</li> <li>• 必须确保配置的LDAP服务主机名或者IP与HiveServer节点互通。如果URL中配置的LDAP主机名，则需要在HiveServer节点的“/etc/hosts”文件中配置相应LDAP主机名和IP映射关系。</li> </ul>	<b>ldap://xxx:xxx</b>

参数名称	参数描述	参数取值示例
hive.server2.authentication.ldap.userDNPattern	在此目录中查找用户DN时使用的模式列表，如果该参数值有多个，使用“:”分隔。	cn=%s,ou=People1,dc=huawei,dc=com: cn=%s,ou=People2,dc=huawei,dc=com
hive.server2.authentication.ldap.username	<b>约束与限制</b> 中创建的与LDAP中同名的用户密码，即 HiveServer健康检查所使用的用户对应的密码。	-

- 修改完成后，单击左上方“保存”，在弹出的对话框中单击“确定”保存配置。
- 单击“实例”，勾选配置状态为“配置过期”的实例，选择“更多 > 重启实例”重启受影响的Hive实例。



## 步骤二：访问 HiveServer

开启LDAP认证后，访问HiveServer需要提供LDAP的用户名和密码。即可在Hive客户端节点执行以下命令连接HiveServer：

```
beeline -u "jdbc:hive2://${hs_ip}:${hs_port}" -n ${user} -p ${password}
```

相关参数说明如下：

- `${user}`：访问HiveServer的LDAP的用户名。
- `${password}`：访问HiveServer的LDAP用户密码。
- `${hs_ip}`：MRS集群HiveServer实例部署节点的业务IP，可以在Manager界面，选择“集群 > 服务 > Hive > 实例”中查看。
- `${hs_port}`：MRS集群HiveServer的端口号，可以在任一HiveServer实例所在节点执行以下命令获取：

```
grep -A 1 'hive.server2.thrift.port' ${BIGDATA_HOME}/  
FusionInsight_HD_*/*_HiveServer/etc/hive-site.xml | grep 'value' | awk -  
F'[<>]' '{print $3}'
```



## 5.8 使用 Kafka Eagle 对接 MRS Kafka

### 应用场景

Kafka Eagle是一款分布式、高可用的Kafka监控软件，提供丰富的Kafka监控指标，例如：Kafka集群的Broker数、Topic数、Consumer数、Topic LogSize Top10、Topic Capacity Top10、Lag挤压、CPU/Memory监控等。

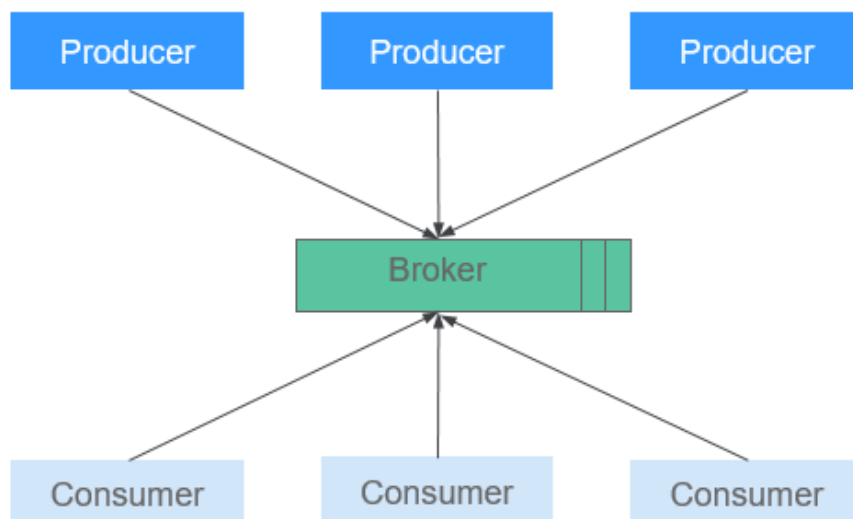
Eagle新版本中改名为EFAK。

### 方案架构

Kafka是一个分布式的、分区的、多副本的消息发布-订阅系统，它提供了类似于JMS的特性，但在设计上完全不同，它具有消息持久化、高吞吐、分布式、多客户端支持、实时等特性，适用于离线和在线的消息消费，如常规的消息收集、网站活性跟踪、聚合统计系统运营数据（监控数据）、日志收集等大量数据的互联网服务的数据收集场景。

生产者（Producer）将消息发布到Kafka主题（Topic）上，消费者（Consumer）订阅这些主题并消费这些消息。在Kafka集群上一个服务器称为一个Broker。对于每一个主题，Kafka集群保留一个用于缩放、并行化和容错性的分区（Partition）。每个分区是一个有序、不可变的消息序列，并不断追加到提交日志文件。分区的消息每个也被赋值一个称为偏移顺序（Offset）的序列化编号。

图 5-37 Kafka 结构

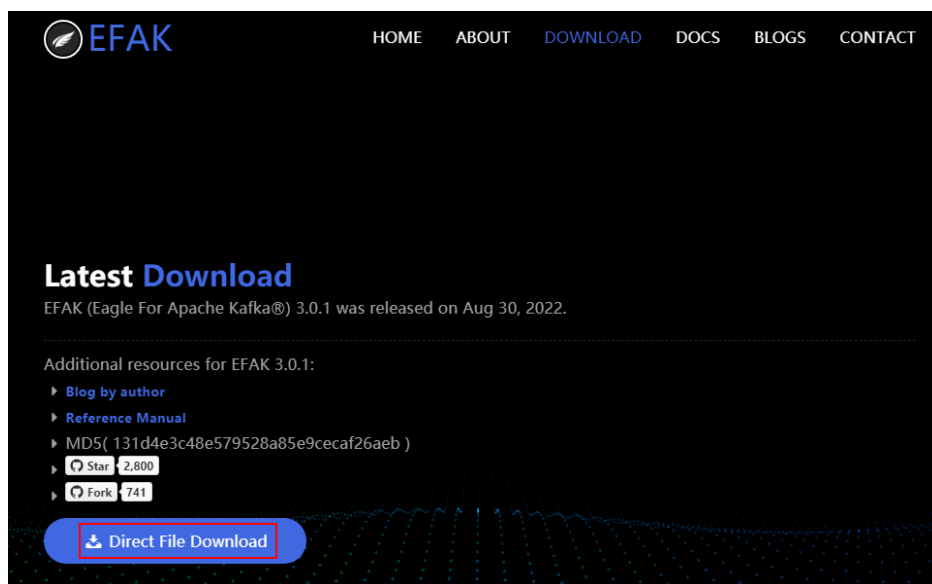


### 约束与限制

- 创建并购买一个包含Kafka组件的MRS 3.1.0版本集群，**集群未开启Kerberos认证**，详情可参考[创建MRS集群](#)。
- 安装MRS集群客户端，具体请参考[安装客户端](#)。

## 步骤一：配置 Kafka Eagle 对接 MRS 参数

1. 下载 **Kafka Eagle**，此处以 EFAK3.0.1 版本为例，具体以实际为准。  
例如获取到 kafka-eagle-bin-3.0.1.tar.gz 软件包。



2. 登录 FusionInsight Manager，选择“集群 > 服务 > Kafka > 配置 > 全部配置”搜索并修改参数“KAFKA\_JMX\_IP”的值为“\${BROKER\_IP}”。

图 5-38 修改 Kafka 参数



3. 修改完成后，单击左上方“保存”，在弹出的对话框中单击“确定”保存配置。
4. 单击“概览”页签，选择右上方“更多 > 重启服务”重启 Kafka 服务。
5. 以 root 用户登录集群主节点，将获取到的 EFAK 安装包 kafka-eagle-bin-3.0.1.tar.gz 放到集群目录下，例如“/opt”，执行以下命令解压。

```
cd /opt
```

```
tar -xvf kafka-eagle-bin-3.0.1.tar.gz
```

```
cd kafka-eagle-bin-3.0.1
```

```
tar -xvf efak-web-3.0.1-bin.tar.gz
```

6. 在“opt”目录下新建目录，例如“efak”，并将“efak-web-3.0.1”复制到“/opt/efak”目录下。

```
mkdir /opt/efak
```

```
cp -r /opt/kafka-eagle-bin-3.0.1/efak-web-3.0.1 /opt/efak/
```

## 7. 添加环境变量。

**vi /etc/profile**

新增 “export KE\_HOME” 参数，参数值为efak-web-3.0.1文件所在路径（例如 “/opt/efak/efak-web-3.0.1”）；在 “export PATH” 参数值后添加 “\$KE\_HOME/bin”。例如：

```
export KE_HOME=/opt/efak/efak-web-3.0.1
export PATH=$PATH:$KE_HOME/bin
```

## 8. 修改 “system-config.properties” 配置文件。

**cd /opt/efak/efak-web-3.0.1/conf/****vi system-config.properties**

```
#设置集群
eagle.zk.cluster.alias=cluster1
cluster1.zk.list=10.20.90.24:2181
#cluster2.zk.list=xdn10:2181,xdn11:2181,xdn12:2181
#修改kafka jmx uri的配置
cluster1.efak.jmx.uri=service:jmx:rmi:///jndi/rmi://%s/kafka
#修改kafka mysql jdbc driver address数据库相关的配置
efak.driver=com.mysql.cj.jdbc.Driver
efak.url=jdbc:mysql://IP:Port/ke?
useUnicode=true&characterEncoding=UTF-8&zeroDateTimeBehavior=convertToNull
efak.username=root
efak.password=XXX
```

**📖 说明**

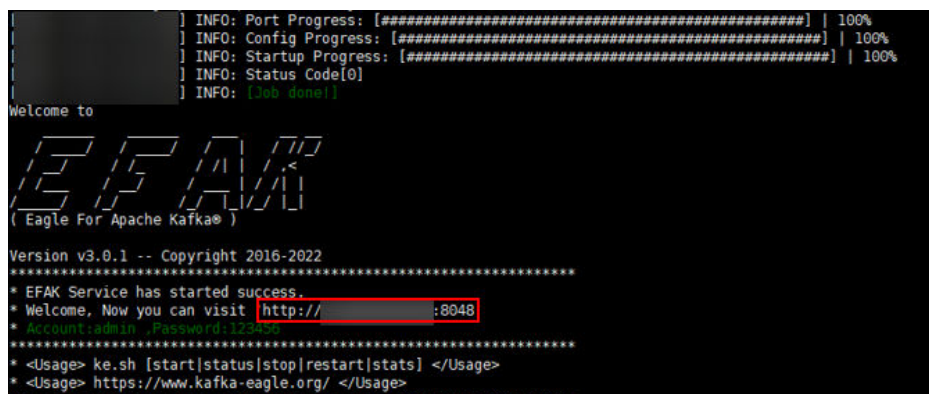
- “cluster1.zk.list” 的值是Kafka组件参数 “metrics.reporter.zookeeper.url” 的值，具体可通过登录FusionInsight Manager，选择 “集群 > 服务 > Kafka > 配置 > 全部配置” 搜索参数 “metrics.reporter.zookeeper.url” 查看。
- “efak.url” 的值为MySQL JDBC连接字符串，具体以实际为准。
- “efak.username” 的值为连接数据库使用的用户名称。
- “efak.password” 的值为连接数据库使用的用户名称所对应的密码。

**步骤二：启动 EFAK 服务验证对接结果**

## 1. 启动EFAK服务。

**sh /opt/efak/efak-web-3.0.1/bin/ke.sh start**

启动成功显示如下，获取EFAK WebUI登录地址。



```
] INFO: Port Progress: [#####] | 100%
] INFO: Config Progress: [#####] | 100%
] INFO: Startup Progress: [#####] | 100%
] INFO: Status Code[0]
] INFO: [Job done!]
```

Welcome to  
**EFAK**  
(Eagle For Apache Kafka®)

Version v3.0.1 -- Copyright 2016-2022

```
*****
* EFAK Service has started success.
* Welcome, Now you can visit http://[redacted]:8048
* Account: admin, Password: 123456
*****
* <Usage> ke.sh [start|status|stop|restart|stats] </Usage>
* <Usage> https://www.kafka-eagle.org/ </Usage>
*****
```

## 2. 使用获取到的登录地址，访问EFAK WebUI界面。

**📖 说明**

访问EFAK WebUI界面默认初始账号密码admin/123456

登录后可以查看Kafka集群监控页面、Topic监控页面、Consumer监控页面，例如：

图 5-39 Kafka 集群监控

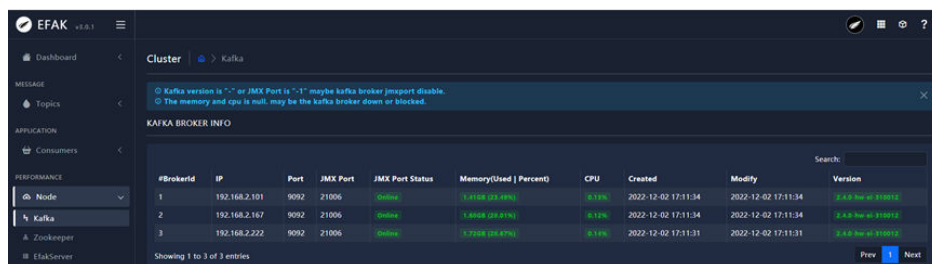


图 5-40 Topic 监控

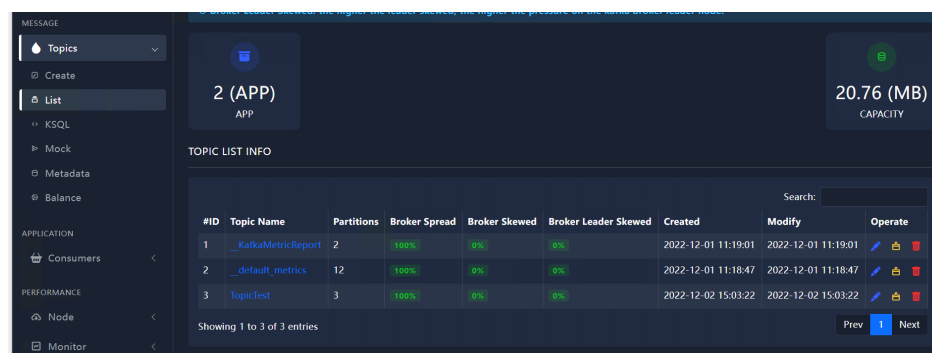
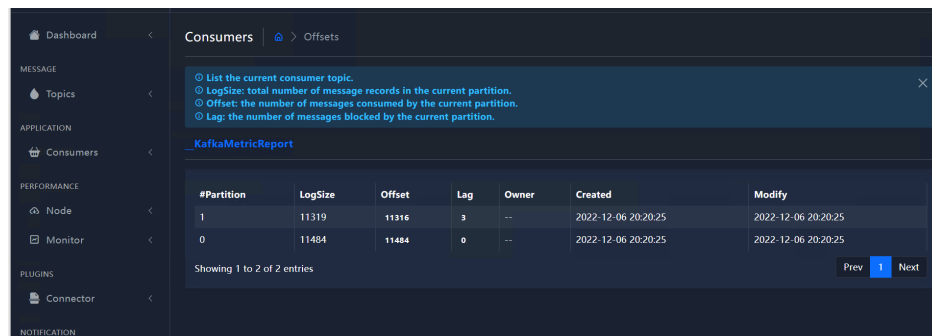


图 5-41 Consumer 监控



## Kafka Eagle 对接常见问题

### 问题现象：

无法获取Kafka CPU与内存监控信息日志提示。

java.io.IOException cannot be cast to javax.management.remote.JMXConnector

```
at org.smartloli.kafka.eagle.web.quartz.shard.task.sub.TopicThroughputByteOutTask.throughput(TopicThroughputByteOutTask.java:110)
# org.smartloli.kafka.eagle.web.quartz.shard.task.sub.TopicThroughputByteOutTask.run(TopicThroughputByteOutTask.java:67)
[2022-12-02 15:31:00] Mx4jServiceImpl.Thread-478 - ERROR - Get topic size from jmx has error, msg is java.io.IOException cannot be cast to javax.management.remote.JMXConnector
[2022-12-02 15:31:00] Mx4jServiceImpl.Thread-478 - ERROR - Get topic size from jmx has error, msg is java.io.IOException cannot be cast to javax.management.remote.JMXConnector
[2022-12-02 15:31:00] Mx4jServiceImpl.Thread-478 - ERROR - Get topic size from jmx has error, msg is java.io.IOException cannot be cast to javax.management.remote.JMXConnector
[2022-12-02 15:31:00] Mx4jServiceImpl.Thread-478 - ERROR - Get topic size from jmx has error, msg is java.io.IOException cannot be cast to javax.management.remote.JMXConnector
[2022-12-02 15:31:00] Mx4jServiceImpl.Thread-478 - ERROR - Get topic size from jmx has error, msg is java.io.IOException cannot be cast to javax.management.remote.JMXConnector
[2022-12-02 15:31:00] Mx4jServiceImpl.Thread-478 - ERROR - Get topic size from jmx has error, msg is java.io.IOException cannot be cast to javax.management.remote.JMXConnector
[2022-12-02 15:31:00] Mx4jServiceImpl.Thread-478 - ERROR - Get topic size from jmx has error, msg is java.io.IOException cannot be cast to javax.management.remote.JMXConnector
[2022-12-02 15:31:00] Mx4jServiceImpl.Thread-478 - ERROR - Get topic size from jmx has error, msg is java.io.IOException cannot be cast to javax.management.remote.JMXConnector
[2022-12-02 15:31:00] Mx4jServiceImpl.Thread-478 - ERROR - Get topic size from jmx has error, msg is java.io.IOException cannot be cast to javax.management.remote.JMXConnector
[2022-12-02 15:31:00] Mx4jServiceImpl.Thread-478 - ERROR - Get topic size from jmx has error, msg is java.io.IOException cannot be cast to javax.management.remote.JMXConnector
[2022-12-02 15:31:00] Mx4jServiceImpl.Thread-478 - ERROR - Get topic size from jmx has error, msg is java.io.IOException cannot be cast to javax.management.remote.JMXConnector
[2022-12-02 15:31:00] Mx4jServiceImpl.Thread-478 - ERROR - Get topic size from jmx has error, msg is java.io.IOException cannot be cast to javax.management.remote.JMXConnector
[2022-12-02 15:31:00] Mx4jServiceImpl.Thread-478 - ERROR - Get topic size from jmx has error, msg is java.io.IOException cannot be cast to javax.management.remote.JMXConnector
[2022-12-02 15:31:00] Mx4jServiceImpl.Thread-478 - ERROR - Get topic size from jmx has error, msg is java.io.IOException cannot be cast to javax.management.remote.JMXConnector
```

### 问题原因：

jmx地址配置错误。默认jmx地址为：  
cluster1.efak.jmx.uri=service:jmx:rmi:///jndi/rmi://%/jmxrmi

**解决办法：**

MRS内Kafka jmx名称为kafka，需要设置为：  
cluster1.efak.jmx.uri=service:jmx:rmi:///jndi/rmi://%/kafka

## 5.9 使用 Jupyter Notebook 对接 MRS Spark

### 应用场景

在MRS服务中可以配合Jupyter Notebook使用PySpark，能够提高机器学习、数据探索和ETL应用开发效率。

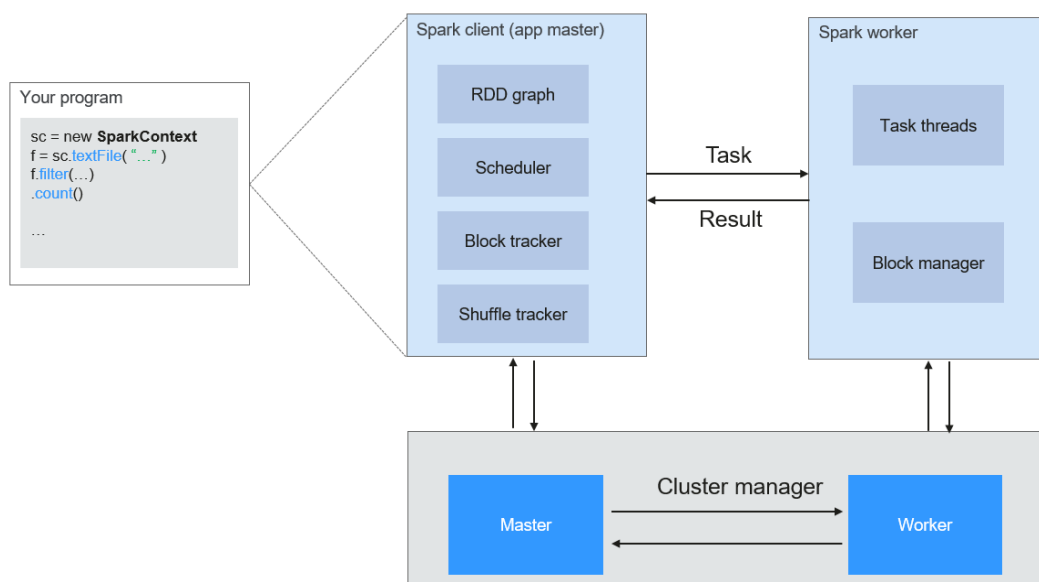
本实践指导用户如何在MRS集群中配置Jupyter Notebook来使用Pyspark。

### 方案架构

Spark的应用运行架构如图5-42所示，运行流程如下所示：

1. 应用程序（Application）是作为一个进程的集合运行在集群上的，由Driver进行协调。
2. 在运行一个应用时，Driver会去连接集群管理器（Standalone、Mesos、YARN）申请运行Executor资源，并启动ExecutorBackend。然后由集群管理器在不同的应用之间调度资源。Driver同时会启动应用程序DAG调度、Stage划分、Task生成。
3. 然后Spark会把应用的代码（传递给SparkContext的JAR或者Python定义的代码）发送到Executor上。
4. 所有的Task执行完成后，用户的应用程序运行结束。

图 5-42 Spark 应用运行架构



### 约束与限制

本实践仅适用于MRS 3.x及之后版本，且在集群外客户端节点中安装Python3。

## 操作流程

本实践基本操作流程如下所示：

1. **步骤1：在MRS集群外节点安装客户端**
2. **步骤2：安装Python3**
3. **步骤3：安装Jupyter Notebook**
4. **步骤4：验证Jupyter Notebook访问MRS**

### 步骤 1：在 MRS 集群外节点安装客户端

**步骤1** 准备一台不属于MRS集群的Linux弹性云服务器，绑定一个弹性IP，参考[集群外节点安装客户端](#)章节安装集群客户端，例如安装目录为“/opt/client”。

**步骤2** 确认MRS集群是否开启了Kerberos认证。

- 是，执行**步骤3**。
- 否，执行**步骤2：安装Python3**。

**步骤3** 登录集群的Manager界面。

**步骤4** 单击“系统 > 权限 > 用户”，创建一个业务用户。

用户类型为人机用户，用户组选择“hadoop”，主组选择“hadoop”，角色选择“Manager\_operator”。

例如创建的用户为“mrs-test”。

图 5-43 创建 MRS 业务用户

\* 用户名:

\* 用户类型:  人机  机机

\* 密码策略:

\* 密码:

\* 确认密码:

用户组: [添加](#) [清除全部](#) [创建新用户组](#)

hadoop ×

主组:

角色: [添加](#) [清除全部](#) [创建新角色](#)

Manager\_operator ×

**步骤5** 使用root用户，登录到集群客户端节点，执行如下命令配置环境变量并进行认证，首次进行用户认证需要修改用户密码。

```
source /opt/client/bigdata_env
```

```
kinit mrs-test
```

----结束

## 步骤 2: 安装 Python3

**步骤1** 使用root用户，登录集群外客户端节点，执行如下命令，检查是否安装了Python3。

```
python3 --version
```

```
[root@ecs-notebook FusionInsight_Cluster_1_Services_ClientConfig]# python3 --version  
-bash: python3: command not found
```

- 是，执行[步骤8](#)。
- 否，执行[步骤2](#)。

## 📖 说明

本案例仅适用于**集群外客户端**节点安装Python3。

**步骤2** 安装Python，此处以Python 3.6.6为例。

1. 执行如下命令，安装相关依赖：

```
yum install zlib zlib-devel zip -y
```

```
yum install gcc-c++
```

```
yum install openssl-devel
```

```
yum install sqlite-devel -y
```

如果pandas库需要额外安装如下依赖：

```
yum install -y xz-devel
```

```
yum install bzip2-devel
```

2. 下载对应Python版本源码。

```
wget https://www.python.org/ftp/python/3.6.6/Python-3.6.6.tgz
```

3. 执行如下命令，解压python源码压缩包，例如下载在“opt”目录下。

```
cd /opt
```

```
tar -xvf Python-3.6.6.tgz
```

4. 创建Python的安装目录，此处以“/opt/python36”为例。

```
mkdir /opt/python36
```

5. 编译Python。

```
cd /opt/python-3.6.6
```

```
./configure --prefix=/opt/python36
```

执行成功，显示结果如下：

```
configure: creating ./config.status
config.status: creating Makefile.pre
config.status: creating Modules/Setup.config
config.status: creating Misc/python.pc
config.status: creating Misc/python-config.sh
config.status: creating Modules/ld_so_aix
config.status: creating pyconfig.h
creating Modules/Setup
creating Modules/Setup.local
creating Makefile

If you want a release build with all stable optimizations active (PGO, etc),
please run ./configure --enable-optimizations
```

执行make -j8命令，执行成功，显示结果如下：

```
creating build/scripts-3.6
copying and adjusting /tmp/python366/Python-3.6.6/Tools/scripts/pydoc3 -> build/scripts-3.6
copying and adjusting /tmp/python366/Python-3.6.6/Tools/scripts/idle3 -> build/scripts-3.6
copying and adjusting /tmp/python366/Python-3.6.6/Tools/scripts/2to3 -> build/scripts-3.6
copying and adjusting /tmp/python366/Python-3.6.6/Tools/scripts/pyvenv -> build/scripts-3.6
changing mode of build/scripts-3.6/pydoc3 from 644 to 755
changing mode of build/scripts-3.6/idle3 from 644 to 755
changing mode of build/scripts-3.6/2to3 from 644 to 755
changing mode of build/scripts-3.6/pyvenv from 644 to 755
renaming build/scripts-3.6/pydoc3 to build/scripts-3.6/pydoc3.6
renaming build/scripts-3.6/idle3 to build/scripts-3.6/idle3.6
renaming build/scripts-3.6/2to3 to build/scripts-3.6/2to3-3.6
renaming build/scripts-3.6/pyvenv to build/scripts-3.6/pyvenv-3.6
```

执行make install命令，执行成功，显示结果如下：



```
rm -f /opt/python36/share/man/man1/python3.1
(cd /opt/python36/share/man/man1; ln -s python3.6.1 python3.1)
if test "xupgrade" != "xno" ; then \
  case upgrade in \
    upgrade) ensurepip="--upgrade" ;; \
    install|*) ensurepip="" ;; \
    esac; \
  ./python -E -m ensurepip \
  $ensurepip --root=/ ; \
fi
Looking in links: /tmp/tmp6ldv525m
Collecting setuptools
Collecting pip
Installing collected packages: setuptools, pip
Successfully installed pip-10.0.1 setuptools-39.0.1
```

6. 执行如下命令，配置Python环境变量。

```
export PYTHON_HOME=/opt/python36
```

```
export PATH=$PYTHON_HOME/bin:$PATH
```

7. 执行python3 --version命令，显示结果如下，表示Python已经安装完成。

```
Python 3.6.6
```

### 步骤3 验证Python3。

```
pip3 install helloworld
```

```
python3
```

```
import helloworld
```

```
helloworld.say_hello("test")
```

```
[root@ecs-notebook Python-3.6.6]# pip3 install helloworld
Collecting helloworld
  Downloading https://files.pythonhosted.org/packages/1b/bf/f0f69f122150e0e98b5d95987a7ef5add3f8a348c6b70d5871f855ca84e/helloworld-0.0.1-py3-none-any.whl
Installing collected packages: helloworld
Successfully installed helloworld-0.0.1
You are using pip version 10.0.1, however version 21.3.1 is available.
You should consider upgrading via the 'pip install --upgrade pip' command.
[root@ecs-notebook Python-3.6.6]# python3
Python 3.6.6 (default, Dec 15 2021, 06:12:48)
[(GCC 4.8.5 20150623 (Red Hat 4.8.5-44)) on linux
Type "help", "copyright", "credits" or "license()" for more
>>> import helloworld
helloworld.say_hello("test")Hello, Sara!
>>>
'Hello, test!'
>>>
```

### 步骤4 测试安装第三方Python库（如pandas、sklearn）。

```
pip3 install pandas
```

```
[root@ecs-mrs-test Python-3.6.6]# pip3 install pandas
Collecting pandas
  Downloading https://files.pythonhosted.org/packages/c3/e2/09cacecafbab071c787019f00ad84ca3185952f6bb9bca9550ed83870d4d/pandas-1.1.5-cp36-cp36m-manylinux_2_17_x86_64.whl (9.5MB)
100% |#####| 9.5MB 6.5MB/s
Collecting pytz>=2017.2 (from pandas)
  Downloading https://files.pythonhosted.org/packages/60/2e/dec1cc18c51b8df33c7c4d0a321b084cf38e1733b98f9d15818880fb4970/pytz-2022.1-py2.py3-none-any.whl (247kB)
100% |#####| 512kB 47.2MB/s
Collecting python-dateutil>=2.7.3 (from pandas)
  Downloading https://files.pythonhosted.org/packages/36/7a/87837f39d0296e723bb9b62bb257d0355c7f6128853c78955f57342a56d/python_dateutil-2.8.2-py2.py3-none-any.whl (247kB)
100% |#####| 256kB 54.5MB/s
Collecting numpy>=1.15.4 (from pandas)
  Downloading https://files.pythonhosted.org/packages/45/b2/6c7f545bb7a38754d63048c7966804a0d947328125d81bf12beaa692c3ae3/numpy-1.19.5-cp36-cp36m-manylinux_2_17_x86_64.whl (13.4MB)
100% |#####| 13.4MB 4.2MB/s
Collecting six>=1.5 (from python-dateutil)
  Downloading https://files.pythonhosted.org/packages/d9/5a/e7c31adbe875f2abb91bd84cf2dc52d792b5a81586781dbc f25c91daf11/six-1.16.0-py2.py3-none-any.whl (10kB)
Installing collected packages: pytz, six, python-dateutil, numpy, pandas
Successfully installed numpy-1.19.5 pandas-1.1.5 python-dateutil-2.8.2 pytz-2022.1 six-1.16.0
You are using pip version 10.0.1, however version 21.3.1 is available.
You should consider upgrading via the 'pip install --upgrade pip' command.
```

```
pip3 install backports.lzma
```

```
[root@ecs-mrs-test Python-3.6.6]# pip3 install backports.lzma
Collecting backports.lzma
  Using cached https://files.pythonhosted.org/packages/21/0f/1a9990233076d48aa2084108ba289ca162975e73a688f3a56c0ee2bb441a/backports.lzma-0.0.14.tar.gz
Installing collected packages: backports.lzma
Running setup.py install for backports.lzma ... done
Successfully installed backports.lzma-0.0.14
You are using pip version 10.0.1, however version 21.3.1 is available.
You should consider upgrading via the 'pip install --upgrade pip' command.
```

```
pip3 install sklearn
```

```

root@ecs-mrs-test Python-3.6.6]# pip3 install sklearn
Collecting sklearn
  Downloading https://files.pythonhosted.org/packages/1e/7a/dbb3be0ce9bd5c8b7e3d87328e79663f8b263b2b1bfa4774cb1147bfcdf/sklearn-0.0.tar.gz
Collecting scikit-learn (from sklearn)
  Downloading https://files.pythonhosted.org/packages/f5/ef/bcd79e8d59250d6e8478eb1290dc6e05be42b3be8a86e3954146adbc171a/scikit_learn-0.24.2-cp36-cp36m-linux_x86_64.whl (20.6MB)
100% |#####| 20.0MB 3.4MB/s
Collecting joblib>=0.11 (from scikit-learn->sklearn)
  Downloading https://files.pythonhosted.org/packages/3e/d5/0163eb0cfa0b673aa4f1cd3ea9d8a81ea0f32e50807b0c295871e4aab2e/joblib-1.1.0-py2.py3-none-any.whl (306kB)
100% |#####| 307kB 46.5MB/s
Requirement already satisfied: scipy>=0.19.1 in /root/.local/lib/python3.6/site-packages (from scikit-learn->sklearn) (1.5.4)
Collecting threadpoolctl>=2.0.0 (from scikit-learn->sklearn)
  Downloading https://files.pythonhosted.org/packages/61/cf/6e354304bc9c6413c4e02a747b60061c21d30ba51e7e544ac7bc66aacc/threadpoolctl-3.1.0-py3-none-any.whl
Requirement already satisfied: numpy>=1.13.3 in /opt/python36/lib/python3.6/site-packages (from scikit-learn->sklearn) (1.19.5)
Installing collected packages: joblib, threadpoolctl, scikit-learn, sklearn
Running setup.py install for sklearn ... done
Successfully installed joblib-1.1.0 scikit-learn-0.24.2 sklearn-0.0 threadpoolctl-3.1.0
You are using pip version 10.0.1, however version 21.3.1 is available.
You should consider upgrading via the 'pip install --upgrade pip' command.
    
```

步骤5 执行命令python3 -m pip list，查看安装结果。

```

[root@ecs-mrs-test Python-3.6.6]# python3 -m pip list
Package            Version
-----
cycler              0.11.0
joblib              1.1.0
kiwisolver          1.3.1
numpy               1.19.5
pandas              1.1.5
pip                 10.0.1
pyparsing           3.0.7
python-dateutil     2.8.2
pytz                2022.1
scikit-learn        0.24.2
scipy               1.5.4
setuptools          39.0.1
six                 1.16.0
sklearn             0.0
threadpoolctl       3.1.0
    
```

步骤6 打包Python.zip

```

cd /opt/python36/
zip -r python36.zip ./
    
```

步骤7 上传到HDFS指定目录。

```

hdfs dfs -mkdir /user/python
hdfs dfs -put python36.zip /user/python
    
```

步骤8 配置MRS客户端。

进入Spark客户端安装目录“/opt/client/Spark2x/spark/conf”，在“spark-defaults.conf”配置文件如下参数。

```

spark.pyspark.driver.python=/usr/bin/python3
spark.yarn.dist.archives=hdfs://hacluster/user/python/python36.zip#Python
    
```

----结束

### 步骤 3: 安装 Jupyter Notebook

步骤1 使用root用户登录客户端节点，执行如下命令安装Jupyter Notebook。

```

pip3 install jupyter notebook
    
```

显示结果如下，表示安装成功：

```

Successfully installed MarkupSafe-2.0.1 Send2Trash-1.8.0 argon2-cffi-21.3.0 argon2-cffi-bindings-21.2.0 async-generator-1.10 attrs-21.2.0 backcall-0.2.0 bleach-4.1.0 cffi-1.15.0 dataclasses-0.8 decorator-5.1.0 defusedxml-0.7.1 entrypoints-0.3 importlib-metadata-4.8.2 ipykernel-5.5.6 ipython-7.16.2 ipython-genutils-0.2.0 ipywidgets-7.6.5 jedi-0.17.2 Jinja2-3.0.3 jsonschema-4.0.0 jupyter-1.0.0 jupyter-client-7.1.0 jupyter-console-6.4.0 jupyter-core-4.9.1 jupyterlab-pygments-0.1.2 jupyterlab-widgets-1.0.2 mistune-0.8.4 nbclient-0.5.9 nbconvert-6.0.7 nbformat-5.1.3 nest-asyncio-1.5.4 notebook-6.4.0 packaging-21.3 pandocfilters-1.5.0 parso-0.7.1 pexpect-4.8.0 pickleshare-0.7.5 prometheus-client-0.12.0 prompt-toolkit-3.0.24 ptyprocess-0.7.0 pycparser-2.21 pygments-2.10.0 pyparsing-3.0.6 pysistent-0.10.0 python-dateutil-2.8.2 pyzmq-22.3.0 rfc3339-1.5.2 six-1.16.0 terminado-0.12.1 testpath-0.5.0 tornado-6.1 traitlets-4.3.3 typing-extensions-4.0.1 wcwidth-0.2.5 webencodings-0.5.1 widgets-0.5.1
You are using pip version 10.0.1, however version 21.3.1 is available.
You should consider upgrading via the 'pip install --upgrade pip' command.
    
```

**步骤2** 为保障系统安全，需要生成一个密文密码用于登录Jupyter，放到Jupyter Notebook的配置文件中。

执行如下命令，需要输入两次密码：（进行到Out[3]退出）

**ipython**

```
[root@ecs-notebook python36]# ipython
Python 3.6.6 (default, Dec 20 2021, 09:32:25)
Type 'copyright', 'credits' or 'license' for more information
IPython 7.16.2 -- An enhanced Interactive Python. Type '?' for help.
In [1]: from notebook.auth import passwd
In [2]: passwd()
Enter password:
Verify password:
Out[2]: 'argon2:$argon2id$v=19$m=10240,t=10,p=8$g14BqLddl927n/unsyPLlQ
$YmoKJzbUfNG7LcxyUzm90bgbKWUliHy6ZV+ObTzdcA'
```

**步骤3** 执行如下命令生成Jupyter配置文件。

**jupyter notebook --generate-config**

**步骤4** 修改配置文件。

**vi ~/.jupyter/jupyter\_notebook\_config.py**

添加如下配置：

```
# -*- coding: utf-8 -*-
c.NotebookApp.ip='*' #此处填写ecs对应的内网IP
c.NotebookApp.password = u'argon2:$argon2id$v=19$m=10240,t=10,p=8$NmoAVwd8F6vFP2rX5ZbV7w
$SyueJoC0a5TbCuHYzqfSx1vQcFvOTTryR+0uk2MNNZA' # 填写步骤2， Out[2]密码生成的密文
c.NotebookApp.open_browser = False # 禁止自动打开浏览器
c.NotebookApp.port = 9999 # 指定端口号
c.NotebookApp.allow_remote_access = True
```

----结束

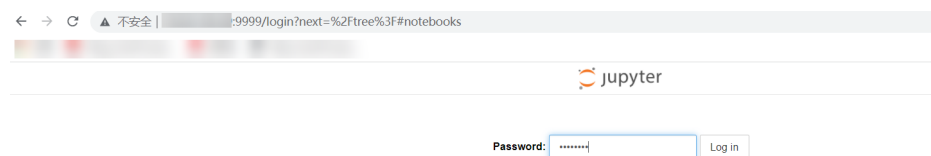
## 步骤 4：验证 Jupyter Notebook 访问 MRS

**步骤1** 在客户端节点执行如下命令，启动Jupyter Notebook。

```
PYSPARK_PYTHON=./Python/bin/python3 PYSPARK_DRIVER_PYTHON=jupyter-
notebook PYSPARK_DRIVER_PYTHON_OPTS="--allow-root" pyspark --master
yarn --executor-memory 2G --driver-memory 1G
```

**步骤2** 在浏览器中输入“弹性IP地址:9999”地址，登录到Jupyter WebUI（保证ECS的安全组对外放通本地公网IP和9999端口），登录密码为**步骤2**设置的密码。

图 5-44 登录 Jupyter WebUI



**步骤3** 创建代码。

创建一个新的python3任务，使用Spark读取文件。

图 5-45 创建 Python 任务



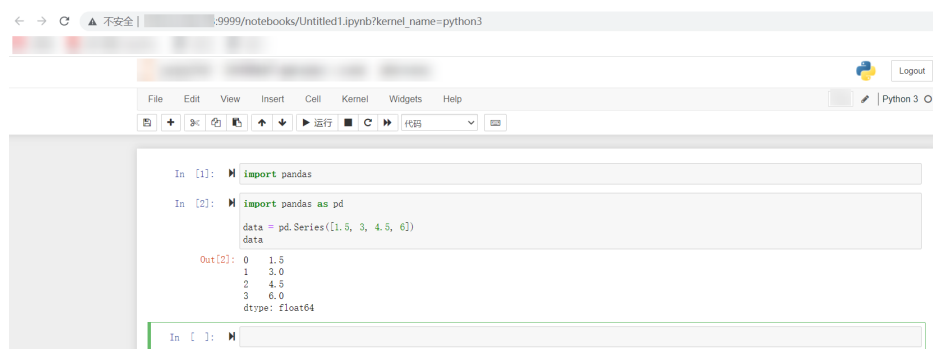
登录到集群Manager界面，在Yarn的WebUI页面上查看提交的pyspark应用。

图 5-46 查看任务运行情况

ID	User	Name	Application Type	Queue	Application Priority	StartTime	FinishTime	State	FinalStatus	Containers	CPU VCoers	Memory MB	Queue
application_1544588847237_0011		PySparkShell	SPARK	default	0	Wed Dec 12 21:51:17 +0800	N/A	RUNNING	UNDEFINED	3	3	6144	375.1

步骤4 验证pandas库调用。

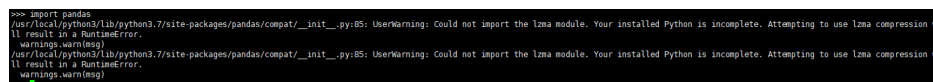
图 5-47 验证 pandas



----结束

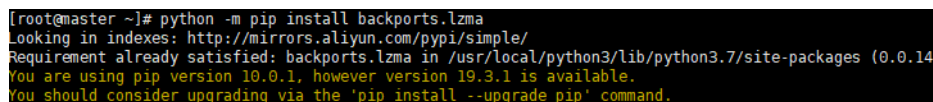
对接 Jupyter 常见问题

pandas本地import使用时，报错如下：



参考以下步骤进行处理：

步骤1 执行命令python -m pip install backports.lzma安装lzma模块，如下图所示：



**步骤2** 进入 “/usr/local/python3/lib/python3.6” 目录（机器不同，目录也有所不同，可以通过which命令来查找当前运行python是使用的那个目录的），然后编辑lzma.py文件。

将：

```
from _lzma import *  
from _lzma import _encode_filter_properties, _decode_filter_properties
```

更改为：

```
try:  
    from _lzma import *  
    from _lzma import _encode_filter_properties, _decode_filter_properties  
except ImportError:  
    from backports.lzma import *  
    from backports.lzma import _encode_filter_properties, _decode_filter_properties
```

修改前：

```
1 """Interface to the liblzma compression library.  
2  
3 This module provides a class for reading and writing compressed files,  
4 classes for incremental (de)compression, and convenience functions for  
5 one-shot (de)compression.  
6  
7 These classes and functions support both the XZ and legacy LZMA  
8 container formats, as well as raw compressed data streams.  
9 """  
10  
11 __all__ = [  
12     "CHECK_NONE", "CHECK_CRC32", "CHECK_CRC64", "CHECK_SHA256",  
13     "CHECK_ID_MAX", "CHECK_UNKNOWN",  
14     "FILTER_LZMA1", "FILTER_LZMA2", "FILTER_DELTA", "FILTER_X86", "FILTER_IA64",  
15     "FILTER_ARM", "FILTER_ARMTHUMB", "FILTER_POWERPC", "FILTER_SPARC",  
16     "FORMAT_AUTO", "FORMAT_XZ", "FORMAT_ALONE", "FORMAT_RAW",  
17     "MF_HC3", "MF_HC4", "MF_BT2", "MF_BT3", "MF_BT4",  
18     "MODE_FAST", "MODE_NORMAL", "PRESET_DEFAULT", "PRESET_EXTREME",  
19  
20     "LZMACompressor", "LZMADecompressor", "LZMAFile", "LZMAError",  
21     "open", "compress", "decompress", "is_check_supported",  
22 ]  
23  
24 import builtins  
25 import io  
26 import os  
27 from _lzma import *  
28 from _lzma import _encode_filter_properties, _decode_filter_properties  
29 import compression
```

修改后：

```
These classes and functions support both the XZ and legacy LZMA
container formats, as well as raw compressed data streams.
.....

__all__ = [
    "CHECK_NONE", "CHECK_CRC32", "CHECK_CRC64", "CHECK_SHA256",
    "CHECK_ID_MAX", "CHECK_UNKNOWN",
    "FILTER_LZMA1", "FILTER_LZMA2", "FILTER_DELTA", "FILTER_X86", "FILTER_IA64",
    "FILTER_ARM", "FILTER_ARMTHUMB", "FILTER_POWERPC", "FILTER_SPARC",
    "FORMAT_AUTO", "FORMAT_XZ", "FORMAT_ALONE", "FORMAT_RAW",
    "MF_HC3", "MF_HC4", "MF_BT2", "MF_BT3", "MF_BT4",
    "MODE_FAST", "MODE_NORMAL", "PRESET_DEFAULT", "PRESET_EXTREME",

    "LZMACompressor", "LZMADecompressor", "LZMAFile", "LZMAError",
    "open", "compress", "decompress", "is_check_supported",
]

import builtins
import io
import os
#from _lzma import *
#from _lzma import _encode_filter_properties, _decode_filter_properties
try:
    from _lzma import *
    from _lzma import _encode_filter_properties, _decode_filter_properties
except ImportError:
    from backports.lzma import *
    from backports.lzma import _encode_filter_properties, _decode_filter_properties
import compression
```

步骤3 保存退出，然后再次执行import。

```
[root@master python3.7]# python
Python 3.7.0 (default, Oct 26 2019, 01:19:22)
[GCC 4.8.5 20150623 (Red Hat 4.8.5-36)] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import pandas
>>>
```

----结束

# 6 MRS 集群管理

## 6.1 MRS 集群阈值类告警配置说明

### 应用场景

MRS集群提供可视化、便捷的监控告警功能。用户可以快速获取集群关键性能指标，并评测集群健康状态。

MRS支持配置监控指标阈值用于关注各指标的健康情况，如果出现异常的数据并满足预设条件后，系统将会触发告警信息，并在告警页面中出现此告警信息。

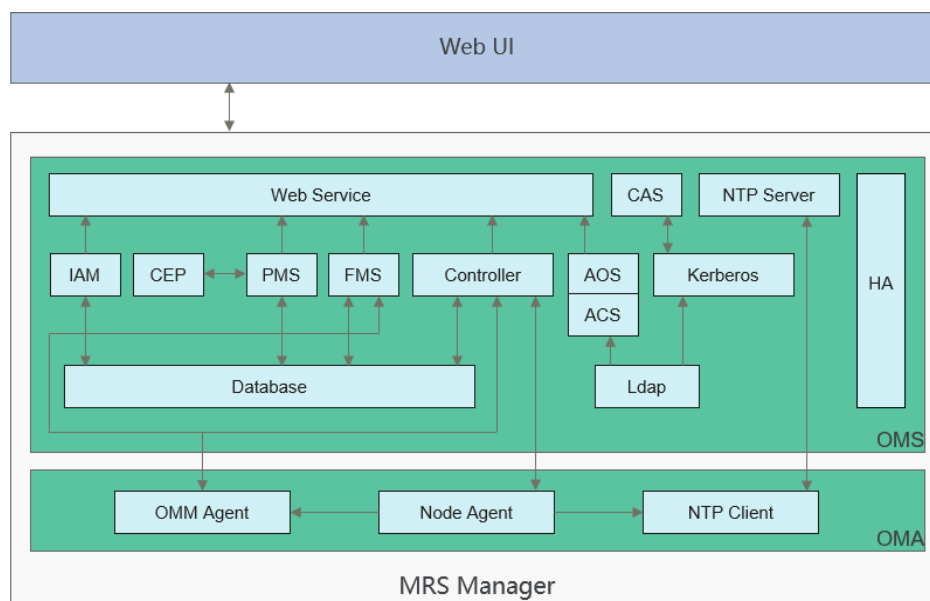
如果部分阈值类监控告警**经评估后**对业务影响可忽略、或告警阈值可进行调整，用户也可以根据需要自定义集群监控指标，或屏蔽对应告警，使告警不再上报。

MRS集群阈值转告警监控指标可分为节点信息指标与集群服务指标，相关指标及其对系统的影响、默认阈值等信息请参考[监控指标参考](#)。

### 方案架构

Manager的整体逻辑架构如[图6-1](#)所示。

图 6-1 Manager 逻辑架构



Manager由OMS和OMA组成：

- OMS：操作维护系统的管理节点，OMS一般有两个，互为主备。
- OMA：操作维护系统中的被管理节点，一般有多个。

FMS为Manager中的告警模块，负责收集每一个OMA上的告警并提供查询。

## 约束与限制

阈值类告警通常会对集群功能的正常使用、或作业的运行等有一定影响，如需屏蔽或修改告警规则等，请提前评估操作风险。

## 修改阈值类告警触发规则

- 步骤1** 参考[访问FusionInsight Manager \(MRS 3.x及之后版本\)](#) 登录MRS集群的 FusionInsight Manager 界面。
- 步骤2** 选择“运维 > 告警 > 阈值设置”。
- 步骤3** 在监控分类中选择集群内指定主机或服务的监控指标。例如选择“主机内存使用率”。

图 6-2 查看阈值告警信息





其中：

- 开关：开启表示将触发告警。
- 分级告警开关：开启时启用分级告警，系统会根据实时监控指标值和分级阈值，动态上报相应级别的告警。仅MRS 3.3.0及之后版本支持配置该参数。
- 平滑次数：Manager会检查监控指标数值是否满足阈值条件，若连续检查且不满足的次数等于“平滑次数”设置的值则发送告警，支持自定义。对于同一告警上报较频繁的场景，用户可通过单击编辑按钮适当调高平滑次数，降低告警频率。
- 检查周期（秒）：表示Manager检查监控指标的时间间隔。
- 规则列表中的条目为触发告警的规则。

#### 步骤4 修改告警规则。

- 添加新规则
  - a. 单击“添加规则”，参考表6-1新增指标的监控行为。
  - b. 单击“确定”保存规则。
  - c. 在当前已应用规则的所在行，单击“操作”中的“取消应用”。如果没有已应用的规则，则请跳过该步骤。
  - d. 在新添加规则的所在行，单击“操作”中的“应用”，此时规则的“生效状态”变成“生效”。
- 修改已有规则
  - a. 单击待修改规则“操作”列的“修改”。
  - b. 参考表6-1修改相关规则参数。
  - c. 单击“确定”保存。

以下样例以修改“主机内存使用率”为例进行说明。

表 6-1 监控指标规则参数

参数名	参数解释	取值样例
规则名称	规则名称	mrs_test
告警级别	告警级别： MRS 3.3.0及之后版本集群开启“分级告警开关”后，可以在“阈值设置”参数中配置告警级别。 <ul style="list-style-type: none"> <li>• 紧急</li> <li>• 重要</li> <li>• 次要</li> <li>• 提示</li> </ul>	重要
阈值类型	选择某指标的最大值或最小值。 <ul style="list-style-type: none"> <li>• 最大值：表示指标的实际值大于设置的阈值时系统将产生告警。</li> <li>• 最小值：表示指标的实际值小于设置的阈值时系统将产生告警。</li> </ul>	最大值

参数名	参数解释	取值样例
日期	设置规则生效的日期，即哪一天运行规则。开启“分级告警开关”后，仅支持“每天”。 <ul style="list-style-type: none"> <li>• 每天</li> <li>• 每周</li> <li>• 其他</li> </ul>	每天
添加日期	仅在“日期”模式为“其他”时可见，设置规则运行的自定义日期，支持多选。	-
阈值设置	起止时间：设置规则运行的具体时间范围。	00:00 - 23:59
	阈值：设置规则监控指标的阈值。	85

----结束

## 屏蔽指定告警上报

- 步骤1** 参考[访问FusionInsight Manager \(MRS 3.x及之后版本\)](#) 登录MRS集群的FusionInsight Manager界面。
- 步骤2** 选择“运维 > 告警 > 屏蔽设置”。
- 步骤3** 在“屏蔽设置”区域，选择指定的服务或模块。
- 步骤4** 单击待操作告警对应“操作”列的“屏蔽”，在弹出的对话框中单击“确定”，修改告警的屏蔽状态为“屏蔽”。

图 6-3 屏蔽告警



### 说明

- 可以在屏蔽列表上方筛选指定的告警。
- 如果需要取消屏蔽，可以单击指定告警后的“取消屏蔽”，在弹出的对话框中单击“确定”，修改告警的屏蔽状态为“显示”。
- 如果需要一次操作多个告警，可以勾选多个待操作的告警后，单击列表上方的“屏蔽”或“取消屏蔽”。

----结束

## 常见问题

- **如何查看当前集群未清除告警？**
  - a. 登录MRS管理控制台。
  - b. 单击待操作的集群名称，选择“告警管理”页签。

- c. 单击“高级搜索”，将“告警状态”设置为“未清除”，单击“搜索”。
  - d. 界面将显示当前集群未清除的告警。
- **集群发生告警后如何清除？**  
 集群发生告警后，可以查看对应告警的帮助文档进行处理。帮助文档查看入口如下：
    - 管理控制台：登录MRS管理控制台，单击待操作的集群名称，选择“告警管理”页签，在告警列表中单击对应操作列的“查看帮助”。然后参考对应告警帮助文档处理步骤进行处理。
    - Manager页面：登录Manager页面，选择“运维 > 告警 > 告警”，单击对应操作列的“查看帮助”。然后参考对应告警帮助文档处理步骤进行处理。

## 监控指标参考

FusionInsight Manager转告警监控指标可分为节点信息指标与集群服务指标。[表6-2](#)表示节点中可配置阈值的指标、[表6-3](#)表示组件可配置阈值的指标。

下表中的“默认阈值”参数值以未开启分级告警开关为例，具体以实际界面显示为准。

**表 6-2** 节点信息监控指标转告警列表（未开启分级告警开关）

监控指标组名称	监控指标名称	告警ID	告警名称	告警对系统的影响	默认阈值
CPU	主机CPU使用率	12016	CPU使用率超过阈值	业务进程响应缓慢或不可用。	90.0%
磁盘	磁盘使用率	12017	磁盘容量不足	业务进程不可用。	90.0%
	磁盘inode使用率	12051	磁盘Inode使用率超过阈值	文件系统无法正常写入。	80.0%
内存	主机内存使用率	12018	内存使用率超过阈值	业务进程响应缓慢或不可用。	90.0%
主机状态	主机文件句柄使用率	12053	主机文件句柄使用率超过阈值	系统应用无法打开文件、网络等IO操作，程序异常。	80.0%
	主机PID使用率	12027	主机PID使用率超过阈值	无法分配PID给新的业务进程，业务进程不可用。	90%
网络状态	TCP临时端口使用率	12052	TCP临时端口使用率超过阈值	主机上业务无法发起对外建立连接，业务中断。	80.0%
网络读信息	读包错误率	12047	网络读包错误率超过阈值	通信闪断，业务超时。	0.5%
	读包丢包率	12045	网络读包丢包率超过阈值	业务性能下降或者个别业务出现超时问题。	0.5%

监控指标组名称	监控指标名称	告警ID	告警名称	告警对系统的影响	默认阈值
	读吞吐率	12049	网络读吞吐率超过阈值	业务系统运行不正常或不可用。	80%
网络写信息	写包错误率	12048	网络写包错误率超过阈值	通信闪断，业务超时。	0.5%
	写包丢包率	12046	网络写包丢包率超过阈值	业务性能下降或者个别业务出现超时问题。	0.5%
	写吞吐率	12050	网络写吞吐率超过阈值	业务系统运行不正常或不可用。	80%
进程	D状态和Z状态进程总数	12028	主机D状态和Z状态进程数超过阈值	占用系统资源，业务进程响应变慢。	0
	omm进程使用率	12061	进程使用率超过阈值	无法切换到omm用户。无法创建新的omm线程。	90

表 6-3 集群监控指标转告警列表（未开启分级告警开关）

服务	监控指标名称	告警ID	告警名称	告警对系统的影响	默认阈值
DBService	数据库连接数使用率	27005	数据库连接数使用率超过阈值	可能导致上层服务无法连接DBService的数据库，影响正常业务。	90%
	数据目录磁盘空间使用率	27006	数据目录磁盘空间使用率超过阈值	业务进程不可用。当数据目录磁盘空间使用率超过90%时，数据库进入只读模式并发送告警“数据库进入只读模式”，业务数据丢失。	80%
Flume	Flume堆内存使用率	24006	Flume Server堆内存使用率超过阈值	堆内存溢出可能导致服务崩溃。	95.0%
	Flume直接内存使用率	24007	Flume Server直接内存使用率超过阈值	直接内存溢出可能导致服务崩溃。	80.0%

服务	监控指标名称	告警 ID	告警名称	告警对系统的影响	默认阈值
	Flume非堆内存使用率	24008	Flume Server 非堆内存使用率超过阈值	非堆内存溢出可能导致服务崩溃。	80.0%
	Flume垃圾回收 (GC) 总时间	24009	Flume Server 垃圾回收(GC) 时间超过阈值	导致Flume数据传输效率低下。	12000 ms
HBase	GC中回收old区所花时长	19007	HBase GC时间超出阈值	老年代GC时间超出阈值, 会影响到HBase数据的读写。	5000ms
	RegionServer直接内存使用率统计	19009	HBase服务进程直接内存使用率超出阈值	HBase可用的直接内存不足, 可能会造成内存溢出导致服务崩溃。	90%
	RegionServer堆内存使用率统计	19008	HBase服务进程堆内存使用率超出阈值	HBase可用内存不足, 可能会造成内存溢出导致服务崩溃。	90%
	HMaster直接内存使用率统计	19009	HBase服务进程直接内存使用率超出阈值	HBase可用的直接内存不足, 可能会造成内存溢出导致服务崩溃。	90%
	HMaster堆内存使用率统计	19008	HBase服务进程堆内存使用率超出阈值	HBase可用内存不足, 可能会造成内存溢出导致服务崩溃。	90%
	单个RegionServer的region数目	19011	RegionServer的Region数量超出阈值	RegionServer的Region数超出阈值, 会影响HBase的数据读写性能。	2000
	处在RIT状态达到阈值时长的region数	19013	region处在RIT状态的时长超过阈值。	表的部分数据丢失或不可用。	1

服务	监控指标名称	告警 ID	告警名称	告警对系统的影响	默认阈值
	RegionServer 的 handler 使用	19021	RegionServer 活跃 handler 数超过阈值	RegionServer 的 handler 使用率超出阈值，会影响 RegionServer 对外提供服务的能力，如果集群的大部分 RegionServer 的 handler 使用率超过阈值，可导致 HBase 无法对外提供服务。	90%
	容灾同步失败次数	19006	HBase 容灾同步失败	无法同步集群中 HBase 的数据到备集群，导致主备集群数据不一致。	1
	主集群等待同步的日志文件数量	19020	HBase 容灾等待同步的 wal 文件数量超过阈值	RegionServer 等待同步的 wal 文件数量超出阈值，会影响 HBase 使用的 znode 超出阈值，影响 HBase 服务状态。	128
	主集群等待同步的 HFile 文件数量	19019	HBase 容灾等待同步的 HFile 文件数量超过阈值	RegionServer 等待同步的 HFile 文件数量超出阈值，会影响 HBase 使用的 znode 超出阈值，影响 HBase 服务状态。	128
	Compaction 操作队列大小	19018	HBase 合并队列超出阈值	产生该告警表示 HBase 服务的 compaction 队列长度已经超过规定的阈值，如果不及处理，可能会导致集群性能下降，影响数据读写。	100
HDFS	HDFS 缺失的块数量	14003	丢失的 HDFS 块数量超过阈值	HDFS 存储数据丢失，HDFS 可能会进入安全模式，无法提供写服务。丢失的块数据无法恢复。	0

服务	监控指标名称	告警 ID	告警名称	告警对系统的影响	默认阈值
	需要复制副本的块总数	14028	待补齐的块数超过阈值	HDFS存储数据丢失，HDFS可能会进入安全模式，无法提供写服务。丢失的块数据无法恢复。	1000
	主NameNode RPC处理平均时间	14021	NameNode RPC处理平均时间超过阈值	NameNode无法及时处理来自HDFS客户端、依赖于HDFS的上层服务、DataNode等的RPC请求，表现为访问HDFS服务的业务运行缓慢，严重时会导致HDFS服务不可用。	100ms
	主NameNode RPC队列平均时间	14022	NameNode RPC队列平均时间超过阈值	NameNode无法及时处理来自HDFS客户端、依赖于HDFS的上层服务、DataNode等的RPC请求，表现为访问HDFS服务的业务运行缓慢，严重时会导致HDFS服务不可用。	200ms
	HDFS磁盘空间使用率	14001	HDFS磁盘空间使用率超过阈值	HDFS集群磁盘容量不足，会影响到HDFS的数据写入。	80%
	DataNode磁盘空间使用率	14002	DataNode磁盘空间使用率超过阈值	DataNode容量不足，会影响到HDFS的数据写入。	80%
	总副本预留磁盘空间所占比率	14023	总副本预留磁盘空间所占比率超过阈值	HDFS集群磁盘容量不足，会影响到HDFS的数据写入。如果DataNode的剩余空间都已经给副本预留，则写入HDFS数据失败。	90%

服务	监控指标名称	告警 ID	告警名称	告警对系统的影响	默认阈值
	故障的 DataNode 总数	14009	Dead DataNode 数量超过阈值	故障状态的 DataNode 节点无法提供 HDFS 服务。	3
	NameNode 非堆内存使用百分比统计	14018	NameNode 非堆内存使用率超过阈值	HDFS NameNode 非堆内存使用率过高，会影响 HDFS 的数据读写性能。	90%
	NameNode 直接内存使用百分比统计	14017	NameNode 直接内存使用率超过阈值	NameNode 可用直接内存不足，可能会造成内存溢出导致服务崩溃。	90%
	NameNode 堆内存使用百分比统计	14007	NameNode 堆内存使用率超过阈值	HDFS NameNode 堆内存使用率过高，会影响 HDFS 的数据读写性能。	95%
	DataNode 直接内存使用百分比统计	14016	DataNode 直接内存使用率超过阈值	DataNode 可用直接内存不足，可能会造成内存溢出导致服务崩溃。	90%
	DataNode 堆内存使用百分比统计	14008	DataNode 堆内存使用率超过阈值	HDFS DataNode 堆内存使用率过高，会影响到 HDFS 的数据读写性能。	95%
	DataNode 非堆内存使用百分比统计	14019	DataNode 非堆内存使用率超过阈值	HDFS DataNode 非堆内存使用率过高，会影响 HDFS 的数据读写性能。	90%
	垃圾回收时间统计 (GC) (NameNode)	14014	NameNode 进程垃圾回收 (GC) 时间超过阈值	NameNode 进程的垃圾回收时间过长，可能影响该 NameNode 进程正常提供服务。	12000 ms
	垃圾回收时间统计 (GC) (DataNode)	14015	DataNode 进程垃圾回收 (GC) 时间超过阈值	DataNode 进程的垃圾回收时间过长，可能影响该 DataNode 进程正常提供服务。	12000 ms
Hive	Hive 执行成功的 HQL 百分比	16002	Hive SQL 执行成功率低于阈值	系统执行业务能力过低，无法正常响应客户请求。	90.0%



服务	监控指标名称	告警 ID	告警名称	告警对系统的影响	默认阈值
	Background线程使用率	16003	Background线程使用率超过阈值	后台Background线程数过多，导致新提交的任务无法及时运行。	90%
	MetaStore的总GC时间	16007	Hive GC时间超出阈值	GC时间超出阈值，会影响到Hive数据的读写。	12000 ms
	HiveServer的总GC时间	16007	Hive GC时间超出阈值	GC时间超出阈值，会影响到Hive数据的读写。	12000 ms
	Hive已经使用的HDFS空间占可使用空间的百分比	16001	Hive数据仓库空间使用率超过阈值	系统可能无法正常写入数据，导致部分数据丢失。	85.0%
	MetaStore直接内存使用率统计	16006	Hive服务进程直接内存使用超出阈值	Hive直接内存使用率过高，会影响Hive任务运行的性能，甚至造成内存溢出导致Hive服务不可用。	95%
	MetaStore非堆内存使用率统计	16008	Hive服务进程非堆内存使用超出阈值	Hive非堆内存使用率过高，会影响Hive任务运行的性能，甚至造成内存溢出导致Hive服务不可用。	95%
	MetaStore堆内存使用率统计	16005	Hive服务进程堆内存使用超出阈值	Hive堆内存使用率过高，会影响Hive任务运行的性能，甚至造成内存溢出导致Hive服务不可用。	95%
	HiveServer直接内存使用率统计	16006	Hive服务进程直接内存使用超出阈值	Hive直接内存使用率过高，会影响Hive任务运行的性能，甚至造成内存溢出导致Hive服务不可用。	95%
	HiveServer非堆内存使用率统计	16008	Hive服务进程非堆内存使用超出阈值	Hive非堆内存使用率过高，会影响Hive任务运行的性能，甚至造成内存溢出导致Hive服务不可用。	95%

服务	监控指标名称	告警 ID	告警名称	告警对系统的影响	默认阈值
	HiveServer堆内存使用率统计	16005	Hive服务进程堆内存使用超出阈值	Hive堆内存使用率过高，会影响Hive任务运行的性能，甚至造成内存溢出导致Hive服务不可用。	95%
	连接到HiveServer的session数占最大允许session数的百分比	16000	连接到HiveServer的session数占最大允许数的百分比超过阈值	发生连接数告警时，表示连接到HiveServer的session数过多，将会导致无法建立新的连接。	90.0%
Kafka	未完全同步的Partition百分比	38006	Kafka未完全同步的Partition百分比超过阈值	Kafka服务未完全同步的Partition数过多，会影响服务的可靠性，一旦发生leader切换，可能会导致丢数据。	50%
	broker上用户连接数使用率	38011	Broker上用户连接数使用率超过设定阈值	当同一个用户连接数太多时，产生告警的用户将无法与Broker建立新的连接。	80%
	Broker磁盘使用率	38001	Kafka磁盘容量不足	磁盘容量不足会导致Kafka写入数据失败。	80.0%
	Broker磁盘IO使用率	38009	Broker磁盘IO繁忙	Partition所在的磁盘分区IO过于繁忙，产生告警的Kafka Topic上可能无法写入数据。	80%
	Broker每分钟的垃圾回收时间统计（GC）	38005	Broker进程垃圾回收（GC）时间超过阈值	Broker进程的垃圾回收时间过长，可能影响该Broker进程正常提供服务。	12000 ms
	Kafka堆内存使用率	38002	Kafka堆内存使用率超过阈值	Kafka可用内存不足，可能会造成内存溢出导致服务崩溃。	95%
	Kafka直接内存使用率	38004	Kafka直接内存使用率超过阈值	Kafka可用直接内存不足，可能会造成内存溢出导致服务崩溃。	95%

服务	监控指标名称	告警 ID	告警名称	告警对系统的影响	默认阈值
Loader	Loader堆内存使用率	23004	Loader堆内存使用率超过阈值	堆内存溢出可能导致服务崩溃。	95%
	Loader直接内存使用率统计	23006	Loader直接内存使用率超过阈值	直接内存溢出可能导致服务崩溃。	80.0%
	Loader非堆内存使用率	23005	Loader非堆内存使用率超过阈值	非堆内存溢出可能导致服务崩溃。	80%
	Loader的总GC时间	23007	Loader进程垃圾回收（GC）时间超过阈值	导致Loader服务响应缓慢。	12000 ms
Mapreduce	垃圾回收时间统计（GC）	18012	JobHistoryServer进程垃圾回收（GC）时间超过阈值	JobHistoryServer进程的垃圾回收时间过长，可能影响该JobHistoryServer进程正常提供服务。	12000 ms
	JobHistoryServer直接内存使用百分比统计	18015	JobHistoryServer直接内存使用率超过阈值	MapReduce可用直接内存不足，可能会造成内存溢出导致服务崩溃。	90%
	JobHistoryServer非堆内存使用百分比统计	18019	JobHistoryServer非堆内存使用率超过阈值	MapReduce JobHistoryServer非堆内存使用率过高，会影响MapReduce任务提交和运行的性能，甚至造成内存溢出导致MapReduce服务不可用。	90%
	JobHistoryServer堆内存使用百分比统计	18009	JobHistoryServer堆内存使用率超过阈值	Mapreduce JobHistoryServer堆内存使用率过高，会影响Mapreduce服务日志归档的性能，甚至造成内存溢出导致Mapreduce服务不可用。	95%
Oozie	Oozie堆内存使用率	17004	Oozie堆内存使用率超过阈值	堆内存溢出可能导致服务崩溃。	95.0%

服务	监控指标名称	告警 ID	告警名称	告警对系统的影响	默认阈值
	Oozie直接内存使用率	17006	Oozie直接内存使用率超过阈值	直接内存溢出可能导致服务崩溃。	80.0%
	Oozie非堆内存使用率	17005	Oozie非堆内存使用率超过阈值	非堆内存溢出可能导致服务崩溃。	80%
	Oozie垃圾回收 (GC) 总时间	17007	Oozie垃圾回收(GC)时间超过阈值	导致Oozie提交任务响应变慢。	12000 ms
Spark2x	JDBCServer2x堆内存使用率统计	43010	JDBCServer2x进程堆内存使用超出阈值	JDBCServer2x进程堆内存使用率过高, 会影响JDBCServer2x进程运行的性能, 甚至造成内存溢出导致JDBCServer2x进程不可用。	95%
	JDBCServer2x直接内存使用率统计	43012	JDBCServer2x进程直接内存使用超出阈值	JDBCServer2x进程直接内存使用率过高, 会影响JDBCServer2x进程运行的性能, 甚至造成内存溢出导致JDBCServer2x进程不可用。	95%
	JDBCServer2x非堆内存使用率统计	43011	JDBCServer2x进程非堆内存使用超出阈值	JDBCServer2x进程非堆内存使用率过高, 会影响JDBCServer2x进程运行的性能, 甚至造成内存溢出导致JDBCServer2x进程不可用。	95%
	JobHistory2x直接内存使用率统计	43008	JobHistory2x进程直接内存使用超出阈值	JobHistory2x进程直接内存使用率过高, 会影响JobHistory2x进程运行的性能, 甚至造成内存溢出导致JobHistory2x进程不可用。	95%

服务	监控指标名称	告警 ID	告警名称	告警对系统的影响	默认阈值
	JobHistory2x 非堆内存使用率统计	43007	JobHistory2x 进程非堆内存使用超出阈值	JobHistory2x进程非堆内存使用率过高，会影响JobHistory2x进程运行的性能，甚至造成内存溢出导致JobHistory2x进程不可用。	95%
	JobHistory2x 堆内存使用率统计	43006	JobHistory2x 进程堆内存使用超出阈值	JobHistory2x进程堆内存使用率过高，会影响JobHistory2x进程运行的性能，甚至造成内存溢出导致JobHistory2x进程不可用。	95%
	IndexServer2x 直接内存使用率统计	43021	IndexServer2x 进程直接内存使用超出阈值	IndexServer2x进程直接内存使用率过高，会影响IndexServer2x进程运行的性能，甚至造成内存溢出导致IndexServer2x进程不可用。	95%
	IndexServer2x 堆内存使用率统计	43019	IndexServer2x 进程堆内存使用超出阈值	IndexServer2x进程堆内存使用率过高，会影响IndexServer2x进程运行的性能，甚至造成内存溢出导致IndexServer2x进程不可用。	95%
	IndexServer2x 非堆内存使用率统计	43020	IndexServer2x 进程非堆内存使用超出阈值	IndexServer2x进程非堆内存使用率过高，会影响IndexServer2x进程运行的性能，甚至造成内存溢出导致IndexServer2x进程不可用。	95%
	JDBCServer2x 的Full GC次数	43017	JDBCServer2x 进程Full GC次数超出阈值	GC次数超出阈值，会影响JDBCServer2x进程运行的性能，甚至造成JDBCServer2x进程不可用。	12

服务	监控指标名称	告警 ID	告警名称	告警对系统的影响	默认阈值
	JobHistory2x 的 Full GC 次数	43018	JobHistory2x 进程 Full GC 次数超出阈值	GC 次数超出阈值，会影响 JobHistory2x 进程运行的性能，甚至造成 JobHistory2x 进程不可用。	12
	IndexServer2x 的 Full GC 次数	43023	IndexServer2x 进程 Full GC 次数超出阈值	GC 次数超出阈值，会影响 IndexServer2x 进程运行的性能，甚至造成 IndexServer2x 进程不可用。	12
	JDBCServer2x 的总 GC 时间	43013	JDBCServer2x 进程 GC 时间超出阈值	GC 时间超出阈值，会影响 JDBCServer2x 进程运行的性能，甚至造成 JDBCServer2x 进程不可用。	12000 ms
	JobHistory2x 的总 GC 时间	43009	JobHistory2x 进程 GC 时间超出阈值	GC 时间超出阈值，会影响 JobHistory2x 进程运行的性能，甚至造成 JobHistory2x 进程不可用。	12000 ms
	IndexServer2x 的总 GC 时间	43022	IndexServer2x 进程 GC 时间超出阈值	GC 时间超出阈值，会影响 IndexServer2x 进程运行的性能，甚至造成 IndexServer2x 进程不可用。	12000 ms
Storm	Supervisor 数	26052	Storm 服务可用 Supervisor 数量小于阈值	集群已经存在的任务无法运行；集群可接收新的 Storm 任务，但是无法运行。	1
	已用 Slot 比率	26053	Storm Slot 使用率超过阈值	用户无法执行新的 Storm 任务。	80.0%
	Nimbus 堆内存使用率	26054	Nimbus 堆内存使用率超过阈值	Storm Nimbus 堆内存使用率过高时可能造成频繁 GC，甚至造成内存溢出，进而影响 Storm 任务提交。	80%

服务	监控指标名称	告警 ID	告警名称	告警对系统的影响	默认阈值
Yarn	NodeManager直接内存使用百分比统计	18014	NodeManager直接内存使用率超过阈值	NodeManager可用直接内存不足，可能会造成内存溢出导致服务崩溃。	90%
	NodeManager堆内存使用百分比统计	18018	NodeManager堆内存使用率超过阈值	NodeManager堆内存使用率过高，会影响Yarn任务提交和运行的性能，甚至可能会造成内存溢出导致Yarn服务崩溃。	95%
	NodeManager非堆内存使用百分比统计	18017	NodeManager非堆内存使用率超过阈值	Yarn NodeManager非堆内存使用率过高，会影响Yarn任务提交和运行的性能，甚至造成内存溢出导致Yarn服务不可用。	90%
	ResourceManager直接内存使用百分比统计	18013	ResourceManager直接内存使用率超过阈值	ResourceManager可用直接内存不足，可能会造成内存溢出导致服务崩溃。	90%
	ResourceManager堆内存使用百分比统计	18008	ResourceManager堆内存使用率超过阈值	Yarn ResourceManager堆内存使用率过高，会影响Yarn任务提交和运行的性能，甚至造成内存溢出导致Yarn服务不可用。	95%
	ResourceManager非堆内存使用百分比统计	18016	ResourceManager非堆内存使用率超过阈值	Yarn ResourceManager非堆内存使用率过高，会影响Yarn任务提交和运行的性能，甚至造成内存溢出导致Yarn服务不可用。	90%
	垃圾回收时间统计（GC）（NodeManager）	18011	NodeManager进程垃圾回收（GC）时间超过阈值	NodeManager进程的垃圾回收时间过长，可能影响该NodeManager进程正常提供服务。	12000 ms

服务	监控指标名称	告警 ID	告警名称	告警对系统的影响	默认阈值
	垃圾回收时间统计 (GC) (ResourceManager)	18010	ResourceManager进程垃圾回收 (GC) 时间超过阈值	ResourceManager进程的垃圾回收时间过长, 可能影响该ResourceManager进程正常提供服务。	12000ms
	root队列下失败的任务数	18026	Yarn上运行失败的任务数超过阈值	大量应用任务运行失败。 运行失败的任务需要重新提交。	50
	root队列下被杀死的任务数	18025	Yarn被终止的任务数超过阈值	大量应用任务被强制终止。	50
	挂起的内存量	18024	Yarn任务挂起内存超过阈值	应用任务结束时间变长。 新应用提交后长时间无法运行。	83886080MB
	正在挂起的任务	18023	Yarn任务挂起数超过阈值	应用任务结束时间变长。 新应用提交后长时间无法运行。	60
ZooKeeper	ZooKeeper连接数使用率	13001	ZooKeeper可用连接数不足	ZooKeeper可用连接数不足, 当连接率超过100%时无法处理外部连接。	80%
	ZooKeeper堆内存使用率	13004	ZooKeeper堆内存使用率超过阈值	ZooKeeper可用内存不足, 可能会造成内存溢出导致服务崩溃。	95%
	ZooKeeper直接内存使用率	13002	ZooKeeper直接内存使用率超过阈值	ZooKeeper可用内存不足, 可能会造成内存溢出导致服务崩溃。	80%
	ZooKeeper每分钟的垃圾回收时间统计 (GC)	13003	ZooKeeper进程垃圾回收 (GC) 时间超过阈值	ZooKeeper进程的垃圾回收时间过长, 可能影响该ZooKeeper进程正常提供服务。	12000ms
Ranger	UserSync垃圾回收 (GC) 时间	45284	UserSync垃圾回收(GC)时间超过阈值	导致UserSync响应缓慢。	12000ms



服务	监控指标名称	告警 ID	告警名称	告警对系统的影响	默认阈值
	PolicySync垃圾回收 ( GC ) 时间	45292	PolicySync垃圾回收(GC)时间超过阈值	导致PolicySync响应缓慢。	12000 ms
	RangerAdmin垃圾回收 ( GC ) 时间	45280	RangerAdmin垃圾回收(GC)时间超过阈值	导致RangerAdmin响应缓慢。	12000 ms
	TagSync垃圾回收 ( GC ) 时间	45288	TagSync垃圾回收(GC)时间超过阈值	导致TagSync响应缓慢。	12000 ms
	UserSync非堆内存使用率	45283	UserSync非堆内存使用率超过阈值	非堆内存溢出可能导致服务崩溃。	80.0%
	UserSync直接内存使用率	45282	UserSync直接内存使用率超过阈值	直接内存溢出可能导致服务崩溃。	80.0%
	UserSync堆内存使用率	45281	UserSync堆内存使用率超过阈值	堆内存溢出可能导致服务崩溃。	95.0%
	PolicySync直接内存使用率	45290	PolicySync直接内存使用率超过阈值	直接内存溢出可能导致服务崩溃。	80.0%
	PolicySync堆内存使用率	45289	PolicySync堆内存使用率超过阈值	堆内存溢出可能导致服务崩溃。	95.0%
	PolicySync非堆内存使用率	45291	PolicySync非堆内存使用率超过阈值	非堆内存溢出可能导致服务崩溃。	80.0%
	RangerAdmin非堆内存使用率	45279	RangerAdmin非堆内存使用率超过阈值	非堆内存溢出可能导致服务崩溃。	80.0%
	RangerAdmin堆内存使用率	45277	RangerAdmin堆内存使用率超过阈值	堆内存溢出可能导致服务崩溃。	95.0%
	RangerAdmin直接内存使用率	45278	RangerAdmin直接内存使用率超过阈值	直接内存溢出可能导致服务崩溃。	80.0%
	TagSync直接内存使用率	45286	TagSync直接内存使用率超过阈值	直接内存溢出可能导致服务崩溃。	80.0%

服务	监控指标名称	告警 ID	告警名称	告警对系统的影响	默认阈值
	TagSync非堆内存使用率	45287	TagSync非堆内存使用率超过阈值	非堆内存溢出可能导致服务崩溃。	80.0%
	TagSync堆内存使用率	45285	TagSync堆内存使用率超过阈值	堆内存溢出可能导致服务崩溃。	95.0%
ClickHouse	Clickhouse服务在 ZooKeeper 的数量配额使用率	45426	ClickHouse服务在 ZooKeeper 的数量配额使用率超过阈值	ClickHouse在 ZooKeeper 的数量配额超过阈值后，无法通过 FusionInsight Manager 对 ClickHouse 进行集群操作，无法使用 ClickHouse 服务功能。	90%
	Clickhouse服务在 ZooKeeper 的容量配额使用率	45427	ClickHouse服务在 ZooKeeper 的容量配额使用率超过阈值	ClickHouse在 ZooKeeper 的容量配额超过阈值后，无法通过 FusionInsight Manager 对 ClickHouse 进行集群操作，无法使用 ClickHouse 服务功能。	90%
IoTDB	合并任务 (空间内合并) 的最大时延	45594	IoTDBServer 空间内合并执行时长超过阈值	空间内合并任务超时，会阻塞写数据，影响写操作的性能。	300000ms
	合并任务 (Flush) 的最大时延	45593	IoTDBServer Flush 执行时长超过阈值	Flush 任务超时，会阻塞写数据，影响写操作的性能。	300000ms
	合并任务 (跨空间合并) 的最大时延	45595	IoTDBServer 跨空间合并执行时长超过阈值	跨空间合并任务超时，会阻塞写数据，影响写操作的性能。	300000ms
	RPC (executeStatement) 的最大时延	45592	IoTDBServer RPC 执行时长超过阈值	IoTDBServer 进程 RPC 过高，会影响 IoTDBServer 进程运行的性能。	10000s

服务	监控指标名称	告警 ID	告警名称	告警对系统的影响	默认阈值
	IoTDBServer 垃圾回收 (GC) 总时间	45587	IoTDBServer 垃圾回收(GC) 时间超过阈值	IoTDBServer进程的垃圾回收 (GC) 时间过长, 可能影响该IoTDBServer 进程正常提供服务。	12000 ms
	ConfigNode垃圾回收 (GC) 总时间	45590	ConfigNode垃圾回收(GC)时间超过阈值	ConfigNode进程的垃圾回收 (GC) 时间过长, 可能影响该ConfigNode进程正常提供服务。	12000 ms
	IoTDBServer 堆内存使用率	45586	IoTDBServer 堆内存使用率超过阈值	IoTDBServer进程堆内存使用率过高, 会影响 IoTDBServer进程运行的性能, 甚至造成内存溢出导致 IoTDBServer进程不可用。	90%
	IoTDBServer 直接内存使用率	45588	IoTDBServer 直接内存使用率超过阈值	直接内存溢出可能导致服务崩溃。	90%
	ConfigNode堆内存使用率	45589	ConfigNode堆内存使用率超过阈值	ConfigNode进程堆内存使用率过高, 会影响ConfigNode进程运行的性能, 甚至造成内存溢出导致ConfigNode进程不可用。	90%
	ConfigNode直接内存使用率	45591	ConfigNode直接内存使用率超过阈值	直接内存溢出可能导致IoTDB实例不可用。	90%

## 6.2 提交 Spark 任务到新增 Task 节点

### 应用场景

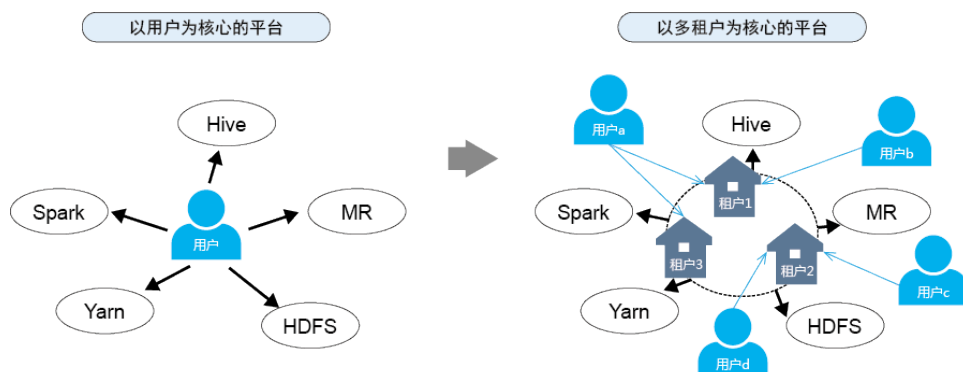
MRS集群可以通过增加Task节点, 提升计算能力, 集群Task节点主要用于处理数据, 不存放持久数据。

本章节指导用户通过租户资源绑定新增的Task节点, 并提交Spark任务到新增的Task节点。

## 方案架构

租户是MRS大数据平台的核心概念，使传统的以用户为核心的大数据平台向以多租户为核心的大数据平台转变，更好的适应现代企业多租户应用环境，如图6-4所示。

图 6-4 以用户为核心的平台和以多租户为核心的平台



对于以用户为核心的大数据平台，用户直接访问并使用全部的资源和服务。

- 用户的应用可能只用到集群的部分资源，资源利用效率低。
- 不同用户的数据可能存放在一起，难以保证数据安全。

对于以租户为核心的大数据平台，用户通过访问租户来使用需要的资源和服务。

- 按照应用需求分配和调度出需要的资源，以租户来统一使用，资源利用效率高。
- 用户通过分配不同的角色获得使用不同租户资源的权限，以保障访问安全。
- 不同的租户之间数据隔离，以保证数据安全。

## 约束与限制

创建并购买一个包含Spark组件的MRS集群，详情可参考[创建MRS集群](#)。

## 操作流程

本实践操作流程如下所示：

1. **步骤一：添加Task节点**：为已有的MRS集群新增Task节点组。
2. **步骤二：添加资源池**：通过管理控制台将新增的节点配置为租户资源池。
3. **步骤三：添加租户**：新创建一个租户。
4. **步骤四：队列配置**：修改租户对应的资源队列配置及资源分布策略，将其与新资源池绑定。
5. **步骤五：创建用户**：新创建一个MRS集群用户并绑定租户。
6. **步骤六：使用spark-submit提交任务**：使用指定的队列运行作业任务。
7. **步骤七：删除Task节点**：释放资源。

### 步骤一：添加 Task 节点

1. 在MRS集群的集群详情页面，选择“节点管理”页签，单击“新增节点组”，进入“新增节点组”页面。

2. 根据需求配置参数。

表 6-4 新增节点组参数说明

参数名称	描述
节点规格	选择节点组内主机的规格类型。
节点数量	设置新增节点组内的节点数量。
系统盘	设置新增节点的系统盘的规格与容量。
数据盘/数据盘数量	设置新增节点的数据盘的规格与容量及数量。
部署角色	添加“NodeManager”角色。

3. 单击“确定”。

## 步骤二：添加资源池

**步骤1** 在集群详情页，单击“租户管理”。

**步骤2** 单击“资源池”页签。

**步骤3** 单击“添加资源池”。

**步骤4** 在“添加资源池”设置资源池的属性。

- “名称”：填写资源池的名称，例如“test1”。
- “资源标签”：填写资源池的标签。例如“1”。
- “可用主机”：选择**步骤一：添加Task节点**添加的节点。

**步骤5** 单击“确定”保存。

----结束

## 步骤三：添加租户

**步骤1** 在集群详情页，单击“租户管理”。

**步骤2** 单击“添加租户”，打开添加租户的配置页面，参见以下表格内容为租户配置属性（以MRS 3.x版本集群为例）。

表 6-5 租户参数一览表

参数名	描述
名称	例如：tenant_spark
租户类型	选择“叶子租户”。当选中“叶子租户”时表示当前租户为叶子租户，无法再添加子租户。当选中“非叶子租户”时表示当前租户可以再添加子租户。

参数名	描述
计算资源	选择“Yarn”，系统将自动在Yarn中以租户名称创建任务队列。计算资源不选择“Yarn”时，系统不会自动创建任务队列。
配置模式	<p>计算资源选择“Yarn”时，“配置模式”可选“基础”或“高级”。</p> <ul style="list-style-type: none"> <li>基础：需配置“默认资源池容量(%)”，配置当前租户在“default”资源池中使用的计算资源百分比。</li> <li>高级：需配置如下参数。 <ul style="list-style-type: none"> <li>权重：资源分配权重，取值范围从0到100。租户资源占比=租户权重/同级别租户总权重之和。</li> <li>最小资源：保证租户能获得的资源（有抢占支持）。取值可以是父租户资源的百分比或绝对值。当租户作业量比较少时，资源会自动借给其他租户，当租户能使用的资源不满足最小资源时，可以通过抢占来要回之前借出的资源。</li> <li>最大资源：租户最多能使用的资源，租户不能得到比最大资源设定更多的资源。取值可以是父租户资源的百分比或绝对值。</li> <li>预留资源：租户最多能使用的资源，租户不能得到比最大资源设定更多的资源。取值可以是父租户资源的百分比或绝对值。</li> </ul> </li> </ul>
默认资源池容量 (%)	配置当前租户在“default”资源池中使用的计算资源百分比，例如“20%”。
储存资源	选择“HDFS”，第一次创建租户时，系统自动在HDFS根目录创建“/tenant”目录。存储资源不选择“HDFS”时，系统不会在HDFS中创建存储目录。
文件/目录数上限	例如：100000000000
存储空间配额	<p>配置当前租户使用的HDFS存储空间配额。最小值为“1”，最大值为父租户的全部存储配额。单位为MB或GB。例如：50000，单位为MB。此参数值表示租户可使用的HDFS存储空间上限，不代表一定使用了这么多空间。如果参数值大于HDFS物理磁盘大小，实际最多使用全部的HDFS物理磁盘空间。</p> <p><b>说明</b></p> <p>为了保证数据的可靠性，HDFS中每保存一个文件则自动生成1个备份文件，即默认共2个副本。HDFS存储空间表示所有副本文件在HDFS中占用的磁盘空间大小总和。例如“存储空间配额”设置为“500MB”，则实际只能保存约500/2=250MB大小的文件。</p>
存储路径	例如：“tenant/spark_test”，系统默认将自动在“/tenant”目录中以租户名称创建文件夹。例如租户“spark_test”，默认HDFS存储目录为“tenant/spark_test”。第一次创建租户时，系统自动在HDFS根目录创建“/tenant”目录。支持自定义存储路径。

参数名	描述
服务	配置当前租户关联使用的其他服务资源，支持HBase。单击“关联服务”，在“服务”选择“HBase”。在“关联类型”选择“独占”表示独占服务资源，选择“共享”表示共享服务资源。
描述	配置当前租户的描述信息。

**步骤3** 单击“确定”保存，完成租户添加。

保存配置需要等待一段时间，界面右上角弹出提示“租户创建成功。”，租户成功添加。

#### 说明

- 创建租户时将自动创建租户对应的角色、计算资源和存储资源。
- 新角色包含计算资源和存储资源的权限。此角色及其权限由系统自动控制，不支持通过“角色管理”进行手动管理。
- 使用此租户时，请创建一个系统用户，并分配Manager\_tenant角色以及租户对应的角色。

----结束


## 步骤四：队列配置

**步骤1** 在集群详情页，单击“租户管理”。

**步骤2** 单击“队列配置”页签。

**步骤3** 在租户队列表格，指定租户队列的“操作”列，单击“修改”。

#### 说明

- 在“租户管理”页签左侧租户列表，单击目标的租户，切换到“资源”页签，单击 也能打开修改队列配置页面（仅适用于MRS 3.x之前版本）。
- 一个队列只能绑定一个非default资源池。

默认资源标签选择**步骤二：添加资源池**时填写的标签，其他参数请根据实际情况填写。

**步骤4** 单击“确定”。

**步骤5** 在集群详情页，单击“租户管理”。

**步骤6** 单击“资源分布策略”页签，资源池选择**步骤二：添加资源池**创建的资源池。

**步骤7** 在租户tenant\_spark“操作”列，单击“修改”。

- 权重：20
- 最小资源：20
- 最大资源：80
- 预留资源：10

**步骤8** 单击“确定”。

----结束

## 步骤五：创建用户

**步骤1** 登录FusionInsight Manager管理界面，具体请参考[访问FusionInsight Manager](#)。

**步骤2** 选择“系统 > 权限 > 用户”，单击“添加用户”。

- 用户名：spark\_test
- 用户类型：人机
- 用户组：hadoop、hive
- 主组：hadoop
- 角色：tenant\_spark

**步骤3** 单击“确定”，完成用户添加。

----结束

## 步骤六：使用 spark-submit 提交任务

**步骤1** 使用root用户登录客户端节点，执行如下命令：

```
cd 客户端安装目录  
source bigdata_env  
source Spark2x/component_env
```

安全集群（开启kerberos认证）执行命令**kinit spark\_test**，普通模式（关闭kerberos认证）无需执行。

输入密码，完成认证（第一次登录需要修改密码）。

**步骤2** 使用认证后的用户提交Spark作业任务。

```
cd Spark2x/spark/bin  
sh spark-submit --queue tenant_spark --class  
org.apache.spark.examples.SparkPi --master yarn-client ../examples/jars/  
spark-examples_*.jar
```

----结束

## 步骤七：删除 Task 节点

**步骤1** 在集群详情页面，单击“节点管理”页签。

**步骤2** 在Task节点组的“操作”列，选择“扩容”。

**步骤3** 扩容类型选择“指定节点扩容”，勾选需要扩容的节点。

### 说明

当节点处于关机、失联、未知、已隔离、故障状态时，可以指定节点扩容。

**步骤4** 勾选“我已经阅读上述信息，并了解具体影响。”，并单击“确定”。

----结束



## 6.3 配置 MRS 集群弹性伸缩

### 应用场景

本入门指导以如下业务场景为例，介绍弹性伸缩规则与资源计划均叠加使用的操作：

某项实时处理业务数据量在周一、周二和周六7:00~13:00出现规律性变化，但是数据量变化并非非常平稳。假设在周一、周二和周六7:00~13:00期间，需要Task节点的数量范围是5~8个，其他时间需要Task节点数量是2~4个。

因此可以在资源计划的基础上，设置基于负载的弹性伸缩规则，以实现当数据量超出预期后，Task节点数量可以在资源计划规定的范围内根据负载情况进行浮动，但不会超出该规定范围。资源计划触发时，会以变化最小的方式使节点数量满足计划规定范围，即如果需要扩容则扩容到计划节点数量范围的下限，如果需要缩容则缩容到计划节点数量范围的上限。

### 约束与限制

在大数据应用，尤其是实时分析处理数据的场景中，常常需要根据数据量的变化动态调整集群节点数量以增减资源。MRS的弹性伸缩规则功能支持根据集群负载对集群进行弹性伸缩。

- 弹性伸缩规则：根据集群实时负载对Task节点数量进行调整，数据量变化后触发扩缩容，有一定的延后性。
- 资源计划（按时间段设置Task节点数量范围）：若数据量变化存在周期性规律，则可通过资源计划在数据量变化前提前完成集群的扩缩容，避免出现增加或减少资源的延后。

弹性伸缩规则与资源计划均可触发弹性伸缩，两者至少配置其中一种，也可以叠加使用。

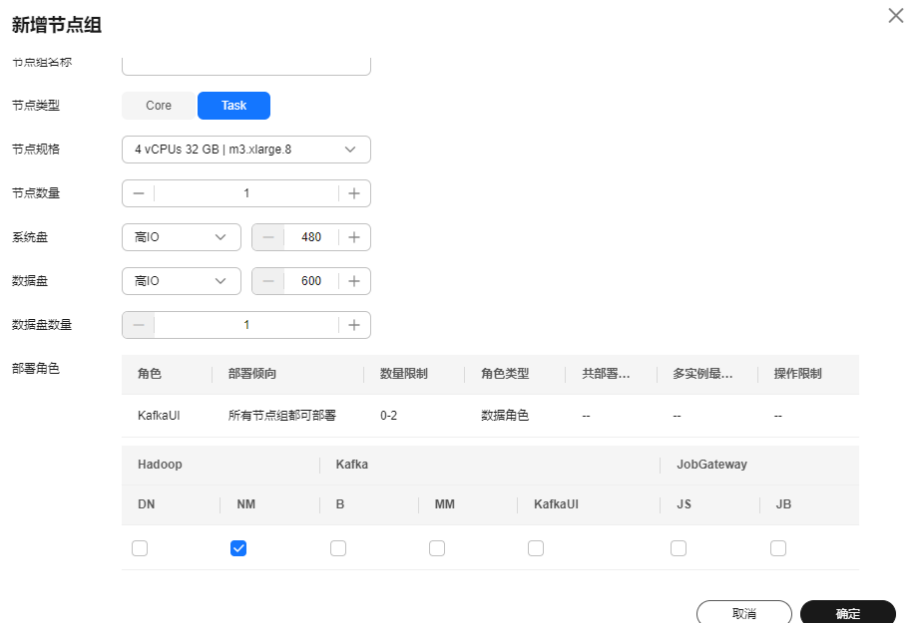
### 添加 Task 节点

MRS集群创建成功之后，如果需要规划Task类型的节点组进行弹性伸缩，可通过手动添加节点组的方式进行配置。

**“自定义”类型集群添加Task节点操作步骤：**

1. 在集群详情页面，选择“节点管理”页签，单击“新增节点组”，进入“新增节点组”页面。
2. 节点类型选择“Task”，“部署角色”参数默认选择“NM”部署NodeManager角色，则新增节点组为Task节点组，其他参数根据需要配置。

图 6-5 添加 Task 节点组



**非“自定义”类型集群添加Task节点操作步骤：**

1. 在集群详情页面，选择“节点管理”页签，单击“配置Task节点”，进入“配置Task节点”页面。
2. 配置“节点类型”、“节点规格”、“节点数量”、“系统盘”，如开启“添加数据盘”后，还需要配置数据盘的存储类型、大小和数量。
3. 单击“确定”。

**弹性伸缩规则与资源计划叠加使用**

**步骤1** 登录MRS管理控制台。

**步骤2** 选择“现有集群”，单击待操作的集群名称，进入集群详情页面。

**步骤3** 选择“弹性伸缩”页签，进入配置弹性伸缩界面。

**步骤4** 单击“新增弹性伸缩策略”按钮，并配置“节点数量范围”为“2 - 4”。

**步骤5** 配置资源计划。

1. 单击默认范围下方的“配置指定时间段的节点数量范围”。
2. 配置“时间范围”和“节点数量范围”。  
“时间范围”：“07:00-13:00”  
“节点数量范围”：“5-8”

图 6-6 弹性伸缩



**步骤6** 配置弹性伸缩规则。

1. 勾选“扩容”。
2. 单击右侧“添加规则”，进入“添加规则”页面。

**图 6-7** 添加规则

The screenshot shows a '添加规则' (Add Rule) dialog box. It contains the following fields and values:

- 规则名称 (Rule Name): default-expand-2
- 如果 (If): YARNAppRunning (selected from a dropdown), 大于 (greater than), 75
- 持续 (Duration): 1 (selected from a dropdown), 个五分钟 (5 minutes)
- 添加 (Add): 1 (selected from a dropdown), 个节点 (nodes)
- 冷却时间 (Cooling Time): 20 (selected from a dropdown), 分钟 (minutes)

At the bottom right, there are two buttons: '取消' (Cancel) and '确定' (Confirm).

“规则名称”：default-expand-2

“如果”：在下拉框中选择规则对象及约束要求，例如YARNAppRunning大于75

“持续”：1个五分钟

“添加”：1个节点

“冷却时间”：20分钟

3. 单击“确定”。

**步骤7** 勾选“我同意授权MRS服务根据以上策略自动进行节点扩容/缩容操作。”。

**步骤8** 单击“确定”，等待弹性伸缩集群设置完成。

----结束

## 参考信息

在添加规则时，可以参考[表6-6](#)配置相应的指标。

### 📖 说明

- 混合集群的支持分析集群和流式集群的所有指标。
- [表6-6](#)该表中不同数值类型取值精度如下：
  - 整型：整数。
  - 百分比：0.01。
  - 比率：0.01。

表 6-6 弹性伸缩指标列表

集群类型	指标名称	数值类型	说明
流式集群	StormSlotAvailable	整型	Storm组件的可用slot数。 取值范围为[0 ~ 2147483646]。
	StormSlotAvailablePercentage	百分比	Storm组件可用slot百分比。是可用slot数与总slot数的比值。 取值范围为[0 ~ 100]。
	StormSlotUsed	整型	Storm组件的已用slot数。 取值范围为[0 ~ 2147483646]。
	StormSlotUsedPercentage	百分比	Storm组件已用slot百分比。是已用slot数与总slot数的比值。 取值范围为[0 ~ 100]。
	StormSupervisorMemAverageUsage	整型	Storm组件Supervisor的内存平均使用量。 取值范围为[0 ~ 2147483646]。
	StormSupervisorMemAverageUsagePercentage	百分比	Storm组件Supervisor进程使用的内存占系统总内存的平均百分比。 取值范围[0 ~ 100]。
	StormSupervisorCPUAverageUsagePercentage	百分比	Storm组件Supervisor进程使用的CPU占系统总CPU的平均百分比。 取值范围[0 ~ 6000]。
分析集群	YARNAppPending	整型	YARN组件挂起的任务数。 取值范围为[0 ~ 2147483646]。
	YARNAppPendingRatio	比率	YARN组件挂起的任务数比例。是YARN挂起的任务数与YARN运行中的任务数比值。 取值范围为[0 ~ 2147483646]。
	YARNAppRunning	整型	YARN组件运行中的任务数。 取值范围为[0 ~ 2147483646]。
	YARNContainerAllocated	整型	YARN组件中已分配的container个数。 取值范围为[0 ~ 2147483646]。
	YARNContainerPending	整型	YARN组件挂起的container个数。 取值范围为[0 ~ 2147483646]。
	YARNContainerPendingRatio	比率	YARN组件挂起的container比率。是挂起的container数与运行中的container数的比值。 取值范围为[0 ~ 2147483646]。

集群类型	指标名称	数值类型	说明
	YARNCPUAllocated	整型	YARN组件已分配的虚拟CPU核心数。 取值范围为[0 ~ 2147483646]。
	YARNCPUAvailable	整型	YARN组件可用的虚拟CPU核心数。 取值范围为[0 ~ 2147483646]。
	YARNCPUAvailablePercentage	百分比	YARN组件可用虚拟CPU核心数百分比。是可用虚拟CPU核心数与总虚拟CPU核心数比值。 取值范围为[0 ~ 100]。
	YARNCPUPending	整型	YARN组件挂起的虚拟CPU核心数。 取值范围为[0 ~ 2147483646]。
	YARNMemoryAllocated	整型	YARN组件已分配内存大小。单位为MB。 取值范围为[0 ~ 2147483646]。
	YARNMemoryAvailable	整型	YARN组件可用内存大小。单位为MB。 取值范围为[0 ~ 2147483646]。
	YARNMemoryAvailablePercentage	百分比	YARN组件可用内存百分比。是YARN组件可用内存大小与YARN组件总内存大小的比值。 取值范围为[0 ~ 100]。
	YARNMemoryPending	整型	YARN组件挂起的内存大小。 取值范围为[0 ~ 2147483646]。

在添加资源计划时，可以参考[表6-7](#)配置相应的参数。

**表 6-7** 资源计划配置项说明

配置项	说明
生效日期	资源计划的生效日期。默认是每日生效，也可以选择周一至周日任意一天或几天生效。
时间范围	资源计划的起始时间和结束时间，精确到分钟，取值范围[00:00, 23:59]。例如资源计划开始于早上8:00，结束于10:00，则配置为8:00-10:00。结束时间必须晚于开始时间至少30分钟。不同资源计划配置的时间段不可交叉。

配置项	说明
节点数量范围	资源计划内的节点数量上下限，取值范围[0,500]，在资源计划时间内，集群Task节点数量小于最小节点数时，弹性伸缩会将集群Task节点一次性扩容到最小节点数。在资源计划时间内，集群Task节点数量大于最大节点数时，弹性伸缩会将集群Task节点一次性缩容到最大节点数。最小节点数必须小于或等于最大节点数。