

MapReduce 服务

最佳实践

文档版本 01
发布日期 2024-06-06



版权所有 © 华为云计算技术有限公司 2024。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

1 数据分析	1
1.1 使用 Spark2x 实现车联网车主驾驶行为分析	1
1.2 使用 Hive 加载 HDFS 数据并分析图书评分情况	9
1.3 使用 Hive 加载 OBS 数据并分析企业雇员信息	15
1.4 通过 Flink 作业处理 OBS 数据	23
1.5 通过 Spark Streaming 作业消费 Kafka 数据	29
1.6 通过 Flume 采集指定目录日志系统文件至 HDFS	36
1.7 基于 Kafka 的 Word Count 数据流统计案例	43
2 数据迁移	48
2.1 数据迁移方案介绍	48
2.1.1 准备工作	48
2.1.2 元数据导出	49
2.1.3 数据拷贝	50
2.1.4 数据恢复	50
2.2 数据迁移到 MRS 前信息收集	51
2.3 数据迁移到 MRS 前网络准备	56
2.4 数据迁移常见端口要求	59
2.5 使用 CDM 服务迁移 Hadoop 数据至 MRS 集群	61
2.6 使用 CDM 服务迁移 HBase 数据至 MRS 集群	64
2.7 使用 CDM 服务迁移 Hive 数据至 MRS 集群	70
2.8 使用 BulkLoad 工具向 HBase 中批量导入数据	72
2.9 使用 CDM 服务迁移 MySQL 数据至 MRS Hive	78
2.10 使用 CDM 服务迁移 MRS HDFS 数据至 OBS	87
3 数据备份与恢复	91
3.1 HDFS 数据	91
3.2 Hive 元数据	92
3.3 Hive 数据	93
3.4 HBase 数据	93
3.5 Kafka 数据	98
4 周边云服务对接	101
4.1 使用 MRS Spark SQL 访问 DWS 表	101
4.2 Hive 对接 CSS 服务	104

5 生态组件对接.....	107
5.1 使用 DBeaver 访问 Phoenix.....	107
5.2 使用 DBeaver 访问 HetuEngine.....	114
5.3 使用 FineBI 访问 HetuEngine.....	119
5.4 使用 Tableau 访问 HetuEngine.....	125
5.5 使用永洪 BI 访问 HetuEngine.....	126
5.6 Hive 对接外置自建关系型数据库.....	129
5.7 Hive 对接外部 LDAP.....	132
5.8 使用 Kafka Eagle 对接 MRS Kafka.....	135
5.9 使用 Jupyter Notebook 对接 MRS Spark.....	139
6 MRS 集群管理.....	148
6.1 配置 MRS 集群阈值类告警.....	148
6.2 提交 Spark 任务到新增 Task 节点.....	167
6.3 配置 MRS 集群弹性伸缩.....	172

1 数据分析

1.1 使用 Spark2x 实现车联网车主驾驶行为分析

本手册基于华为云MapReduce服务实践所编写，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。

📖 说明

本实践仅适用于MRS 3.1.0版本，请按照指导创建集群。

本实践基本内容如下所示：

1. [场景描述](#)
2. [第一步：创建集群](#)
3. [第二步：准备Spark样例程序和样例数据](#)
4. [第三步：创建作业](#)
5. [第四步：查看作业执行结果](#)

场景描述

本次实践的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数。

创建集群

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1-1](#)配置集群软件信息。

表 1-1 软件配置

参数名称	配置方式
区域	选择“中国-香港” 说明 本指导以“中国-香港”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。
计费模式	按需计费
集群名称	mrs_demo
集群类型	选择“分析集群”，用来做离线数据分析
版本类型	选择“普通版”
集群版本	选择“MRS 3.1.0” 说明 本实践仅适用于MRS 3.1.0版本。
组件选择	勾选所有组件
元数据	选择“本地元数据”

图 1-1 自定义购买-软件配置

区域

不同区域的资源之间网不互通。请选择靠近您客户的区域，可以降低网络延时、提高访问速度。如何选择区域 [?](#)

计费模式 包年/包月 按需计费

集群名称

集群类型 自定义 分析集群

分析集群

- 用于离线数据分析场景，对海量数据进行分析处理，形成结果数据。
- 离线处理任务通常占用计算存储资源较多。
- 可根据需求选择Hadoop、Spark、HBase、Hive、Flink、Oozie、Tez等数据分析类组件。

版本类型 LTS版 普通版

集群版本

组件选择 必选组件默认勾选，被依赖的组件会被自动勾选。请根据业务需求合理选择需要的组件，部分类型集群创建后不支持添加服务。 [了解更多](#)

<input checked="" type="checkbox"/>	组件名	版本	描述
<input checked="" type="checkbox"/>	Hadoop	3.1.1	针对大数据集的分布式数据处理框架。
<input checked="" type="checkbox"/>	Spark2x	2.4.5	Spark2x是一个对大规模数据处理的快速和通用引擎，基于开源Spark2.x版本开发。
<input type="checkbox"/>	HBase	2.2.3	HBase是一个高可靠性、高性能、面向列、可伸缩的分布式存储系统。
<input checked="" type="checkbox"/>	Hive	3.1.0	方便查询、管理存储在分布式存储系统上的大数据集的数据仓库软件。
<input type="checkbox"/>	Hue	4.7.0	Apache Hadoop的UI界面。

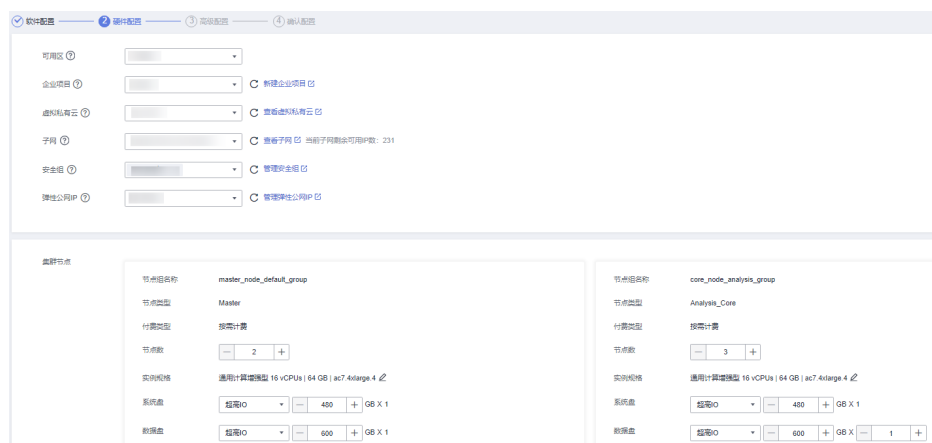
步骤3 单击“下一步”配置硬件信息。

参见表1-2配置集群硬件信息。

表 1-2 硬件配置

参数名称	配置方式
可用区	可用区2
企业项目	选择“default”。
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。
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。
安全组	选择“自动创建”。
弹性公网IP	选择“暂不绑定”。
集群节点	保持默认值。

图 1-2 自定义购买-硬件配置



步骤4 单击“下一步”，高级配置页签参考表1-3配置以下信息，其他选项保持默认值。


表 1-3 高级配置

参数名称	配置方式
Kerberos认证	关闭Kerberos认证。
用户名	Manager管理员用户，目前默认为admin用户。
密码	配置Manager管理员用户的密码。
确认密码	再次输入Manager管理员用户的密码。
登录方式	选择“密码”。
用户名	用于登录弹性云服务器的用户，目前默认为root用户。

参数名称	配置方式
密码	配置登录ECS的用户密码。
确认密码	再次输入登录ECS的用户密码。

图 1-3 自定义购买-高级配置

The screenshot shows a configuration interface with four tabs: 软件配置, 硬件配置, 高级配置 (selected), and 确认配置. Under the '高级配置' tab, there are two sections. The first section is for 'Kerberos认证' (Kerberos authentication), which is turned on. It includes fields for '用户名' (username) set to 'admin', '密码' (password) and '确认密码' (confirm password) (both masked with dots), and '登录方式' (login method) with '密码' (password) selected. The second section is for 'root' user, with '用户名' (username) set to 'root', '密码' (password) and '确认密码' (confirm password) (both masked with dots). Below these sections is a '主机名前缀' (hostname prefix) field (empty) with a note: '用作集群中ECS机器或BMS机器主机名的前缀'. At the bottom, there is a checkbox for '高级配置' (Advanced Configuration) which is currently unchecked, with the text '现在配置' (configure now).

步骤5 单击“下一步”，在“确认配置”页面检查配置集群信息，如需调整配置，可单击 ，跳转到对应页签后重新设置参数。

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

准备 Spark2x 样例程序和样例数据

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。

图 1-4 创建并行文件系统



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1-5所示。

图 1-5 新建文件夹



步骤3 从 https://mrs-obs-ap-southeast-1.obs.ap-southeast-1.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桶。

步骤6 从<https://mrs-obs-ap-southeast-1.obs.ap-southeast-1.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1-6所示的样例数据。

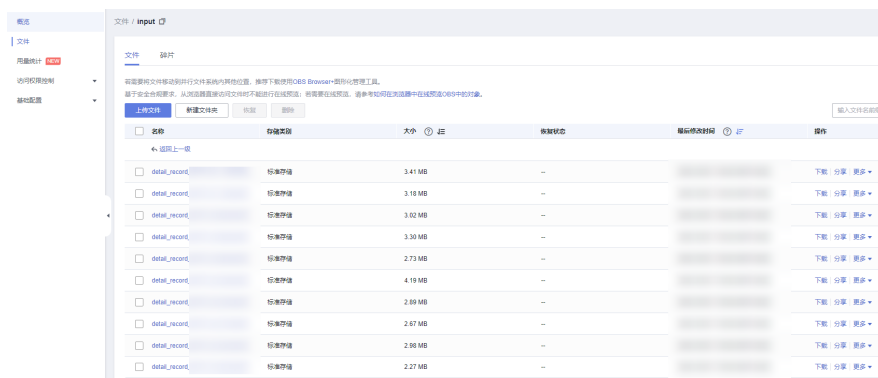
图 1-6 样例数据

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。单击“上传”，上传样例数据到OBS文件系统。

说明

上传**步骤7**解压后的数据至“input”文件夹下，上传后如下图所示：



----结束

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。

图 1-7 添加作业

作业是MRS提供的程序执行平台，帮助您便捷地处理和分析数据。 [了解更多](#)



步骤3 按**图1-8**完成作业参数配置。

表 1-4 配置作业信息

参数名称	配置方法
作业类型	选择“SparkSubmit”。
作业名称	输入“driver_behavior_task”。
执行程序路径	单击“OBS”，选择 准备Spark2x样例程序和样例数据 中上传的名称为driver_behavior.jar的jar包。
运行程序参数	参数选择“--class”，值输入“com.huawei.bigdata.spark.examples.DriverBehavior”。
执行程序参数	<p>输入“AK SK 1 输入路径 输出路径”。</p> <ul style="list-style-type: none"> AK/SK请参考说明方式获取。 1为固定输入，用于指定作业执行时调用的程序函数。 输入路径可通过单击“OBS”进行选择输入路径。 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/。 <p>说明 AK/SK，请通过如下方式获取。</p> <ol style="list-style-type: none"> 登录华为云管理控制台。 单击右上角的用户名，然后选择“我的凭证”。 系统跳转至“我的凭证”页面，单击“访问密钥”。 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。
服务配置参数	保持默认不配置。

图 1-8 添加作业

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 [HDFS] [OBS]

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 [HDFS] [OBS]

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

[确定] [取消]

步骤4 单击“确定”，开始提交作业，执行程序。

----结束

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。

图 1-9 作业执行状态

作业是MRS提供的程序执行平台，帮助您便捷地处理和数据分析。 [了解更多](#)

添加	删除	作业名称/ID	用户名称	作业类型	状态	执行结果	队列名称	作业提交时间	作业结束时间
<input type="checkbox"/>	<input type="checkbox"/>	driver_behavior_task b454b2c-6028-4832-b001-16c8ef6ca000		SparkSubmit	已完成	成功	default		2021/10/20 11:33:31 GMT+08:00

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。

图 1-10 下载作业执行结果

文件 碎片

请根据文件所在位置到文件系统中其他位置，推荐下载使用OBS Browser+图形化管理工具。
基于安全合规要求，从该位置直接访问文件时不能进行在线预览，请参照如何在浏览器中查看跨OBS年的对象。

名称	存储类型	大小	ET	校验状态	最后修改时间	操作
._SUCCESS	标准存储	0 byte	-	-		下载 分享 更多
1	标准存储	28.32 MB	-	-		下载 分享 更多

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 1-11 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A58M83	360	385	315	2997	3181	31248	3594	389	385
zengpeng1000000	华AZQ110	340	344	272	2894	2763	25479	3274	284	337
xie Xiao1000001	华AEB132	264	261	248	2525	2324	23434	2720	314	253
duxu1000009	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AVM936	341	354	291	3043	3044	28728	3552	347	376
haowei1000008	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

1.2 使用 Hive 加载 HDFS 数据并分析图书评分情况

MRS离线处理集群，可对海量数据进行分析和处理，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。

基本内容如下所示：

1. [创建MRS离线查询集群](#)
2. [将本地数据导入到HDFS中](#)
3. [创建Hive表](#)
4. [将原始数据导入Hive并进行分析](#)

场景描述

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 (Optimized Row Columnar) 这几种存储格式。

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畅销图书。

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”，填写软件配置参数。

表 1-5 软件配置（以下参数仅供参考，可根据实际情况调整）

参数项	取值
区域	中国-香港
计费模式	按需计费
集群名称	MRS_demo
版本类型	普通版
集群版本	MRS 3.1.0
组件选择	Hadoop分析集群
可用区	可用区1
虚拟私有云	vpc-01
子网	subnet-01
企业项目	default
Kerberos认证	不开启
用户名	admin/root
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	勾选“确认授权”

图 1-12 购买 Hadoop 分析集群



3. 单击“立即购买”，等待MRS集群创建成功。

图 1-13 集群购买成功

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

将本地数据导入到 HDFS 中

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!
...
```

2. 登录对象存储服务OBS控制台，单击“创建桶”，填写以下参数，单击“立即创建”。

表 1-6 桶参数

参数项	取值
区域	中国-香港
数据冗余存储策略	单AZ存储
桶名称	mrs-hive
默认存储类别	标准存储
桶策略	私有
归档数据直读	关闭
企业项目	default
标签	-

图 1-14 创建 OBS 桶

桶名称: mrs-hive
数据冗余存储策略: 多AZ存储
默认存储类别: 标准存储
桶策略: 私有
默认加密: 开启
归档数据直读: 开启

等待桶创建好，单击桶名称，选择“对象 > 上传对象”，将数据文件上传至OBS桶内。

图 1-15 上传对象

清空列表 添加文件

名称	大小	操作
book_score.txt	307 bytes	删除

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 1-16 同步 IAM 用户

基本信息	运维管理
集群名称: mrs	集群管理页面
集群状态: 运行中	IAM用户同步: 未同步 同步
集群版本: MRS 3.1.0	数据连接: 单击管理

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桶名，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 1-17 从 OBS 导入数据到 HDFS



从OBS导入数据至HDFS

OBS路径 浏览

HDFS路径 浏览

实际运行参考 `hadoop distcp -overwrite obs://`

确定 取消

- c. 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。

图 1-18 数据导入成功



创建 Hive 表

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”。
2. 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
```
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;
```
4. 查看表是否创建成功：

```
show tables;
```

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

将原始数据导入 Hive 并进行分析

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;
```

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 |  
+-----+
```

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。

1.3 使用 Hive 加载 OBS 数据并分析企业雇员信息

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。

基本内容如下所示：

1. [创建MRS离线查询集群](#)
2. [创建OBS委托并绑定至MRS集群](#)
3. [创建Hive表并加载OBS中数据](#)
4. [基于HQL对数据进行分析](#)

场景描述

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（Optimized Row Columnar）这几种存储格式。

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](#)。

本实践中，雇员信息的原始数据包含以下两张表：

表 1-7 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
8	Zhang	R	9000.05	personal income tax&0.05	China:Shanghai	2014

表 1-8 雇员联络信息数据

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

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”，填写软件配置参数。

表 1-9 软件配置（以下参数仅供参考，可根据实际情况调整）

参数项	取值
区域	中国-香港
计费模式	按需计费
集群名称	MRS_demo
版本类型	普通版
集群版本	MRS 3.1.0
组件选择	Hadoop分析集群
可用区	可用区1
虚拟私有云	vpc-01
子网	subnet-01
企业项目	default
Kerberos认证	不开启
用户名	admin/root
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	勾选“确认授权”

图 1-19 购买 Hadoop 分析集群

区域

不同区域的资源之间内网不互通。请选择靠近您客户的区域，可以降低网络延时，提高访问速度。 [如何选择区域](#)

计费模式 包年包月 按需计费 ①

集群名称 ①

版本类型 普通版 LTS版 ②

集群版本

组件选择

实时分析集群	ClickHouse集群	Hadoop分析集群	HBase查询集群
Hadoop 3.1.1, Flink 1.12.0, Kafka 2.11-2.4.0, ZooKeeper 3.5.6, Ranger 2.0.0 and ClickHouse 21.3.4.25	ZooKeeper 3.5.6 and ClickHouse 21.3.4.25	Hadoop 3.1.1, Hive 3.1.0, Spark2x 2.4.5, Tez 0.9.2, Flink 1.12.0, ZooKeeper 3.5.6, Ranger 2.0.0 and Presto 333	Hadoop 3.1.1, HBase 2.2.3, ZooKeeper 3.5.6 and Ranger 2.0.0
海量数据采集，数据实时分析与查询	用于联机分析(OLAP)的列式数据库管理系统(DBMS)	海量数据分析与查询	海量数据存储，实现毫秒级数据查询

3. 单击“立即购买”，等待MRS集群创建成功。

图 1-20 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

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）。
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 IAM”。
 3. 选择“委托 > 创建委托”。
 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。
 5. “持续时间”选择“永久”并单击“下一步”。

图 1-21 创建委托

* 委托名称

* 委托类型 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。

* 云服务

* 持续时间

描述

0/255

- 在弹出授权页面的搜索框内，搜索“OBS OperateAccess”策略，勾选“OBS OperateAccess”策略。

图 1-22 配置权限



- 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”。
- 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。
- 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“管理委托”，选择创建好的OBS委托后单击“确定”。

图 1-23 进入 MRS 集群的概览界面



图 1-24 为集群绑定委托



创建 Hive 表并加载 OBS 中数据

- 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 1-10 并行文件系统参数

参数项	取值
区域	中国-香港
文件系统名称	hiveobs

参数项	取值
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	-

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”。
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
```

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';

5. 查看表是否创建成功:

show tables;

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

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。

图 1-25 上传数据



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
```

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 | xxx@xx.com |  
| 3          | 159 XXXX XXXX | xxx@xx.com.cn |  
| 4          | 186 XXXX XXXX | xxx@xx.org |  
| 6          | 189 XXXX XXXX | xxx@xx.cn |  
| 8          | 134 XXXX XXXX | xxx@xxx.cn |  
+-----+-----+-----+-----+-----+
```

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。

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)
```

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.usd_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 | |
+-----+-----+-----+-----+
+-----+-----+-----+-----+
```

3. 统计雇员信息有多少条记录。

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

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

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 |
xxx@xx.com.cn |
| 6 | Linda | D | 36000.04 | {"personal income tax":0.09} | America:NewYork | 2014 |
xxx@xx.cn |
| 8 | Zhang | R | 9000.05 | {"personal income tax":0.05} | China:Shanghai | 2014 |
xxx@xxx.cn |
+-----+-----+-----+-----+-----+-----+-----+-----+
+-----+-----+-----+-----+-----+-----+-----+-----+
```

1.4 通过 Flink 作业处理 OBS 数据

应用场景

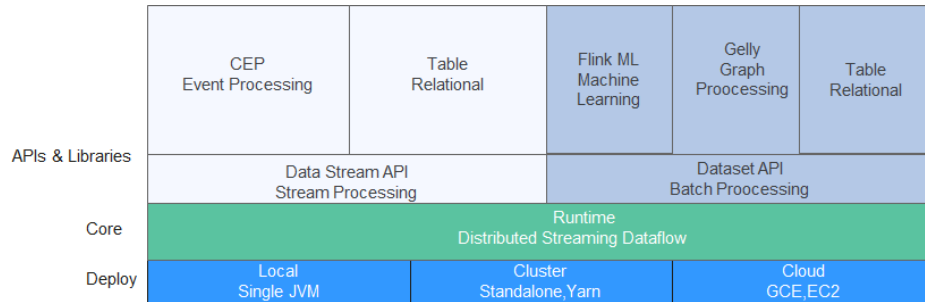
MRS支持在大数据存储容量大、计算资源需要弹性扩展的场景下，用户将数据存储在OBS服务中，使用MRS集群仅做数据计算处理的存算分离模式。

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。

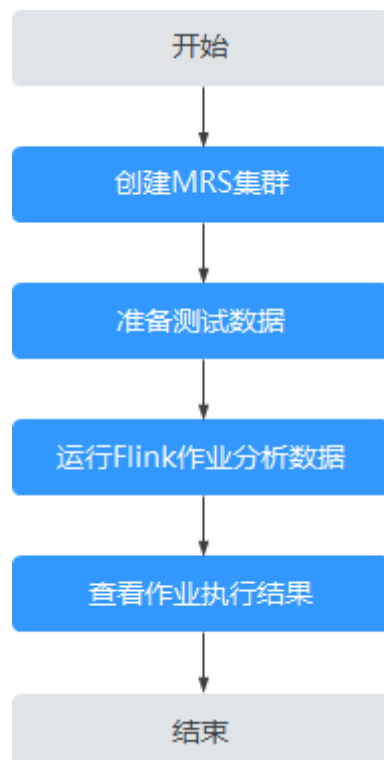


在本示例中，使用MRS集群内置的Flink WordCount作业程序，来分析OBS文件系统中保存的源数据，以统计源数据中的单词出现次数，相关程序代码信息介绍可参考 <https://github.com/apache/flink/tree/master/flink-examples/flink-examples-batch/src/main/java/org/apache/flink/examples/java/wordcount>。

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

本示例操作流程如下：



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。

您可以直接选择系统默认的“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”。

图 1-26 上传测试数据



步骤4 （可选）上传数据分析应用程序。

使用管理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“单击同步”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](#)。

添加作业

★ 作业类型	<input type="text" value="Flink"/>	
★ 作业名称	<input type="text" value="flink_obs_test"/>	
★ 执行程序路径	<input type="text" value="obs://mrs-demo-data/program/WordCount.jar"/>	<input type="button" value="HDFS"/> <input type="button" value="OBS"/>
运行程序参数 ②	<input type="text" value="参数"/>	<input type="text" value="值"/>
执行程序参数 ②	<input type="text" value="--input obs://mrs-demo-data/flink/mrs_flink_test.txt --output obs://mrs-demo-data/flink/output"/>	<input type="button" value="HDFS"/> <input type="button" value="OBS"/>
服务配置参数 ①	<input type="text" value="参数"/>	<input type="text" value="值"/>
命令参考	<pre>flink run -d -m yarn-cluster obs://mrs-demo-data/program/WordCount.jar --input obs://mrs-demo-data/flink/mrs_flink_test.txt --output obs://mrs-demo-data/flink/output</pre>	
<input type="button" value="确定"/> <input type="button" value="取消"/>		

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。



- ▼ Cluster
 - [About](#)
 - [Nodes](#)
 - [Node Labels](#)
 - [Applications](#)
 - NEW
 - NEW SAVING
 - SUBMITTED
 - ACCEPTED
 - RUNNING
 - FINISHED
 - FAILED
 - KILLED
 - [Scheduler](#)
- Tools

Cluster Metrics

Apps Submitted	Apps Pending
5	0

Cluster Nodes Metrics

Active Nodes	Decommissioned Nodes
3	0

User Metrics for developuser

Apps Submitted	Apps Pending	Apps Running
5	0	0

Scheduler Metrics

Scheduler Type	Scheduling Request
SuperiorYarnScheduler	[yarn.io/gpu, memory-r...

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。

<input type="checkbox"/> output	标准存储	203 bytes
<input type="checkbox"/> mrs_flink_test.txt	标准存储	232 bytes

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

1.5 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。

方案架构

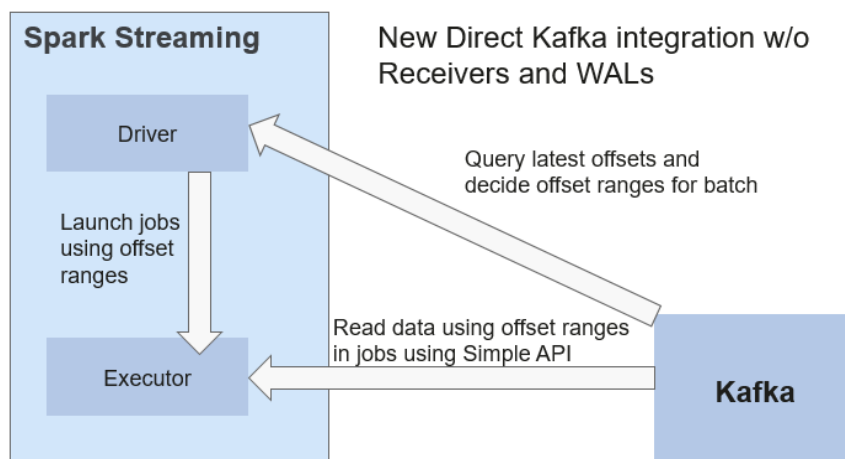
Spark是分布式批处理框架，提供分析挖掘与迭代式内存计算能力，支持多种语言（Scala/Java/Python）的应用开发。适用以下场景：

- 数据处理（Data Processing）：可以用来快速处理数据，兼具容错性和可扩展性。
- 迭代计算（Iterative Computation）：支持迭代计算，有效应对多步的数据处理逻辑。
- 数据挖掘（Data Mining）：在海量数据基础上进行复杂的挖掘分析，可支持各种数据挖掘和机器学习算法。
- 流式处理（Streaming Processing）：支持秒级延迟的流式处理，可支持多种外部数据源。
- 查询分析（Query Analysis）：支持标准SQL查询分析，同时提供DSL（DataFrame），并支持多种外部输入。

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方式。

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 1-27 Direct Kafka 接口数据传输

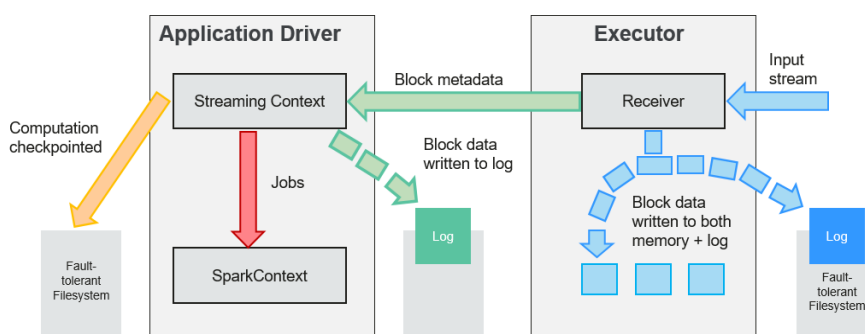


需要注意的是，Spark Streaming可以在失败后重新从Kafka中读取并处理数据段。然而，由于语义仅被处理一次，重新处理的结果和没有失败处理的结果是一致的。

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1-28所示：

图 1-28 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

华为云MapReduce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。

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

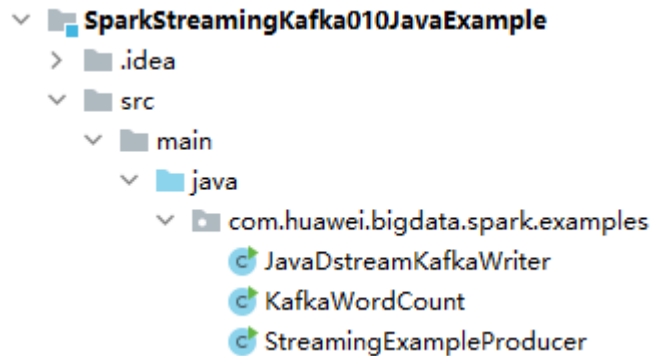
步骤1 通过开源镜像站获取样例工程。

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](#)。



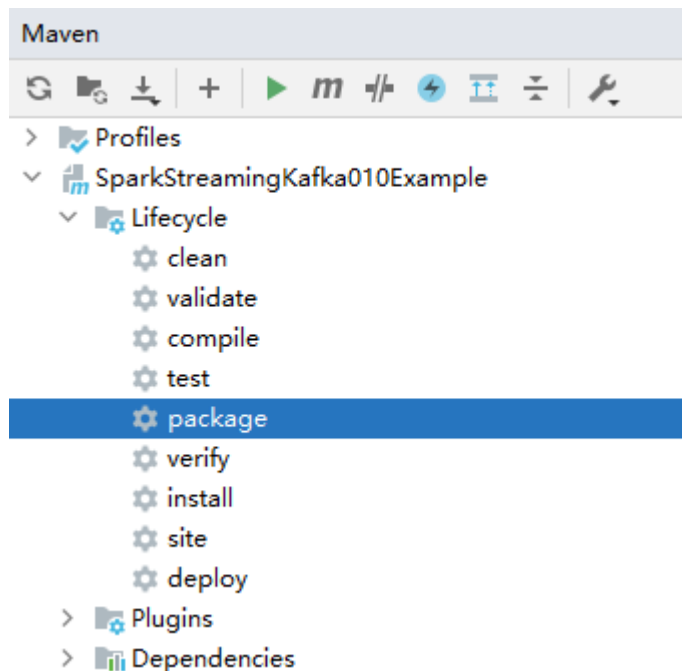
在本示例工程中，通过使用Streaming调用Kafka接口来获取单词记录，然后把单词记录分类统计，得到每个单词记录数，关键代码片段如下：

```
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。



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。

Spark Jobs (?)

User: root
Total Uptime: 7.4 min
Scheduling Mode: FIFO
Completed Jobs: 192
▶ Exec: 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 (8 skipped)
187	Streaming job from [output.operation.0, batch time 15:52:35] print at KafkaWordCount.java:112	2022/06/08 15:53:24	23 ms	2/2 (1 skipped)	4/8 (8 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

1.6 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由一个个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 1-29 Flume-NG 架构

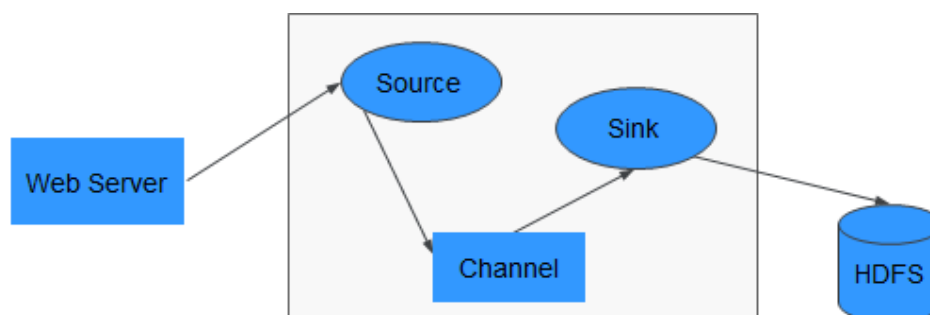
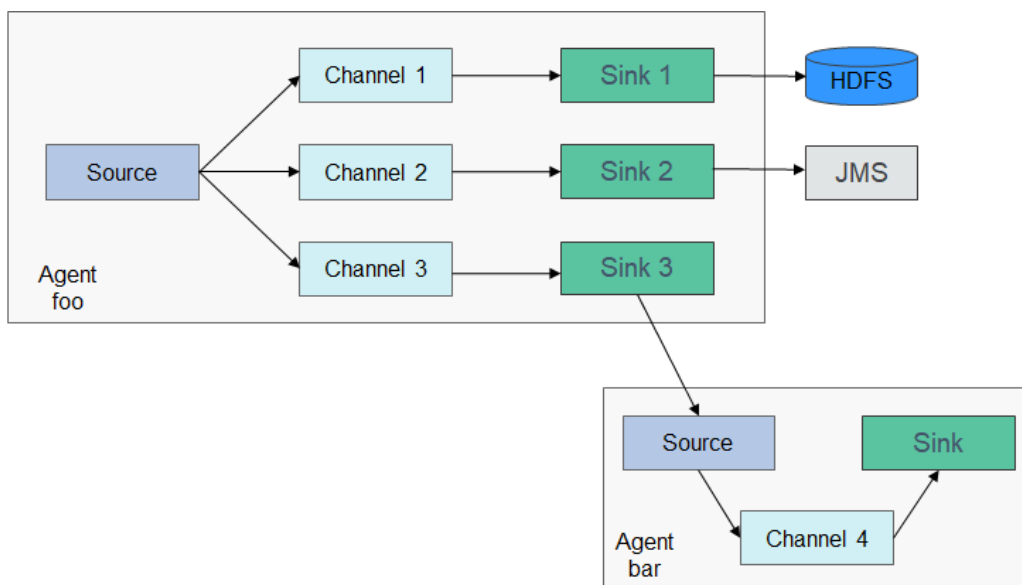


表 1-11 模块说明

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">• 和系统集成的Sources: Syslog、Netcat。• 自动生成事件的Sources: Exec、SEQ。• 用于Agent和Agent之间通信的IPC Sources: Avro。 <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">• Memory Channel: 非持久化。• File Channel: 基于WAL（预写式日志Write-Ahead Logging）的持久化实现。• JDBC Channel: 基于嵌入Database的持久化实现。 <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">• 存储数据到最终目的终端Sink，比如：HDFS、HBase。• 自动消耗的Sink，比如：Null Sink。• 用于Agent间通信的IPC sink: Avro。 <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1-30所示：

图 1-30 Flume 结构图



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 1-31 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
```

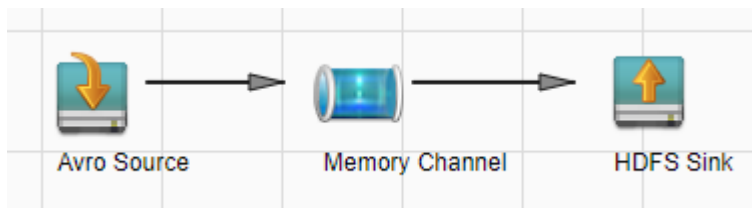
----结束

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。

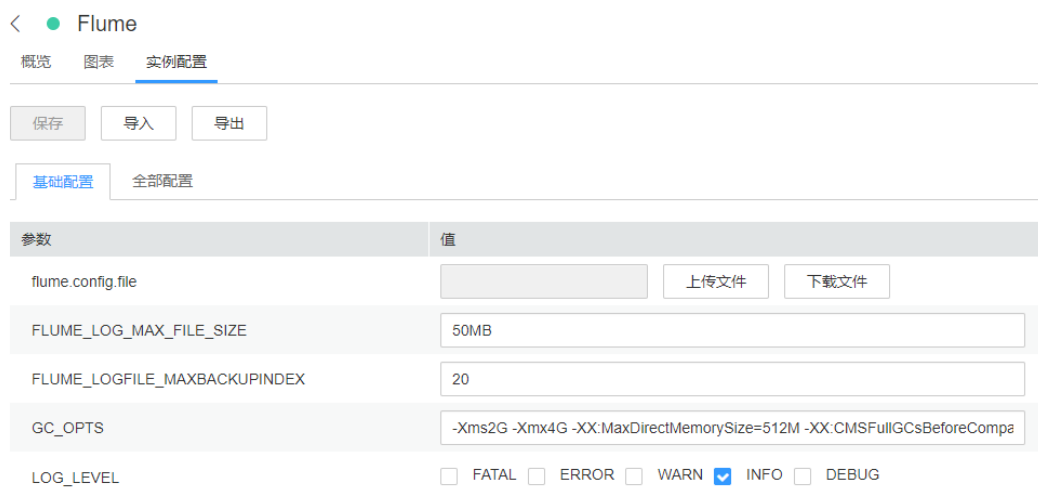


双击模块图标，配置对应参数，各模块配置参数如下（其他参数保持默认）：

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。



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。



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，例如客户端安装包以上传至“/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
```

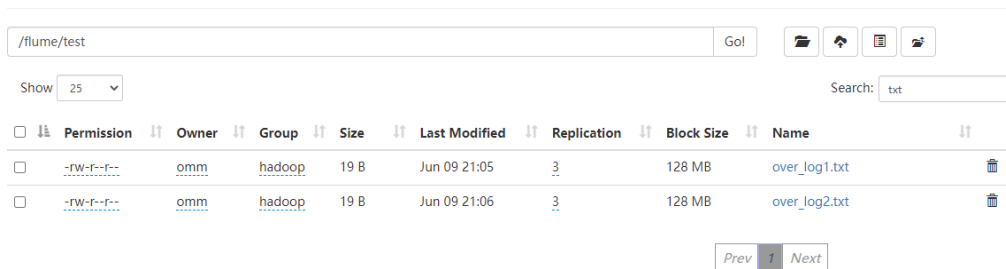
步骤3 登录FusionInsight Manager,选择“集群 > 服务 > HDFS”,单击“NameNode(主)”对应的链接,打开HDFS WebUI界面。

The screenshot shows the HDFS WebUI interface for a cluster named 'HDFS (mrs_Xm3f)'. The interface includes a navigation bar with tabs for '概览', '图表', '实例', '实例组', '配置', '管理NameService', '管理MountTable', and '资源'. The main content area displays '基本信息' (Basic Information) with the following details:

运行状态:	● 良好
配置状态:	⊕ 已同步
版本:	3.1.1
读速率:	0.00 MB/s
写速率:	0.00 MB/s
安全模式:	OFF
磁盘空间:	0.19% 3GB/1.55TB
丢失块数:	0
待复制副本的块数:	0
损坏的块数:	0
正常的DataNode总数:	3
NameNode WebUI:	NameNode(node-master2pxrD.mrs-u06p.com, 备) NameNode(node-master3VpZO.mrs-u06p.com, 主)
NameService个数:	1

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。

Browse Directory



Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
-rw-r--r--	omm	hadoop	19 B	Jun 09 21:05	3	128 MB	over_log1.txt
-rw-r--r--	omm	hadoop	19 B	Jun 09 21:06	3	128 MB	over_log2.txt

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

1.7 基于 Kafka 的 Word Count 数据流统计案例

应用场景

本文介绍如何使用MRS集群运行Kafka程序处理数据。

Kafka Streams是Apache Kafka提供的一个轻量级流式处理框架，其输入输出均存储在Kafka集群中。

接下来，以最常见的WordCount样例Demo进行说明。

方案架构

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

操作流程

华为云MapReduce服务提供了Kafka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. 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。

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。

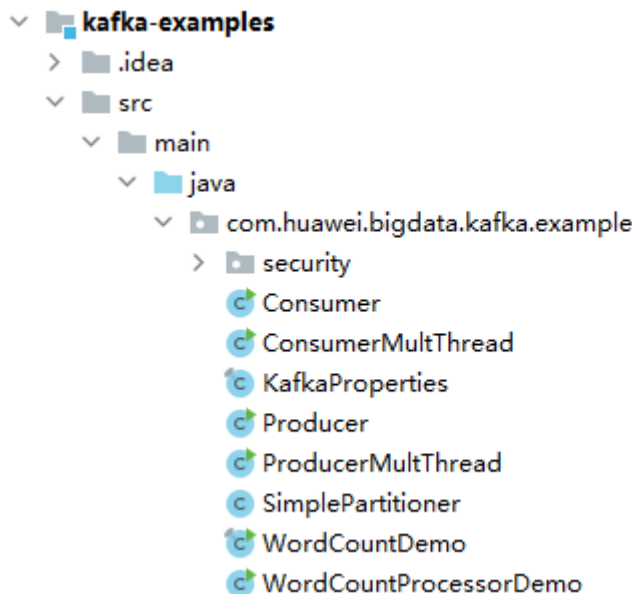
下载样例工程的Maven工程源码和配置文件，并在本地配置好相关开发工具，可参考[通过开源镜像站获取样例工程](#)。

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](#)。



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,
"hadooop.hadoop.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()));
}
...

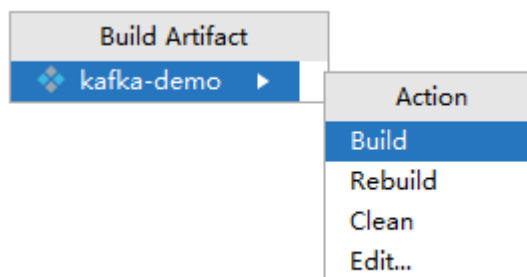
```

📖 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](#)。



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

步骤1 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。

----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”从输出Topic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
```

向输入Topic中写入消息：

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

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 数据迁移

2.1 数据迁移方案介绍

2.1.1 准备工作

本迁移指导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它任何业务。

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](#)。

2.1.2 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

2.1.3 数据拷贝

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拷贝场景。

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元数据备份文件拷贝至目的集群。

```
$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工具将源集群数据拷贝到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数据拷贝到目的集群的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拷贝，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进行拷贝。

2.1.4 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
```

2.2 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，以能够更好的进行迁移决策。

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 2-1 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，参见表2-2。 例如： <ul style="list-style-type: none">• 2台32U64G机器部署NameNode + ResourceManager• 2台32U64G机器部署HiveServer• 20台16U32G机器部署DataNode和NodeManager
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 2-2 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 2-3 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 2-4 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 2-5 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

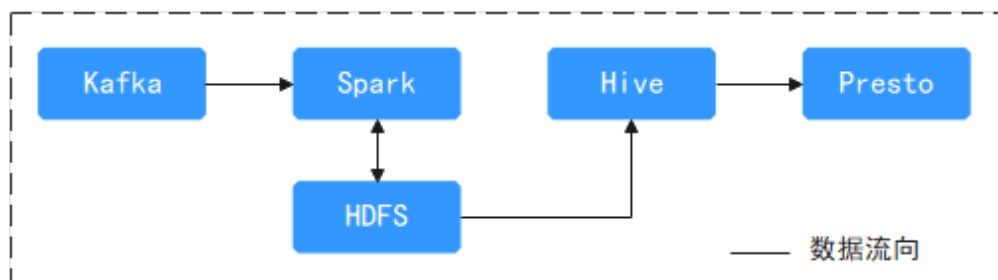
表 2-6 流式 Kafka 集群信息

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

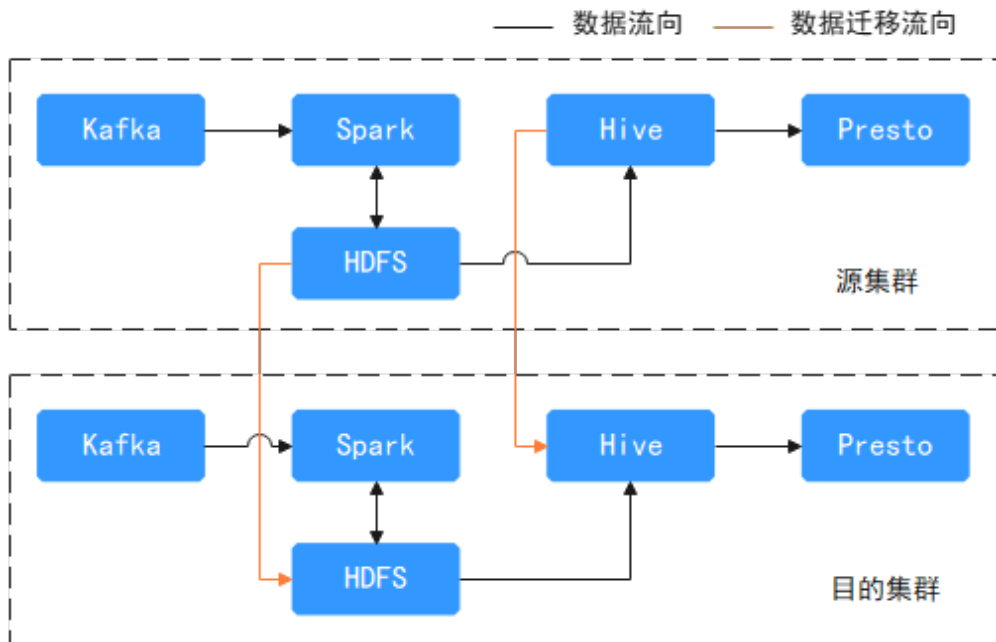
- 一个离线分析平台的客户业务系统，由Spark Streaming消费Kafka数据存入HDFS，HDFS上进行小文件合并后加载到Hive表中，运营人员可以通过Presto进行Hive数据查询。

图 2-1 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 2-2 迁移示意图



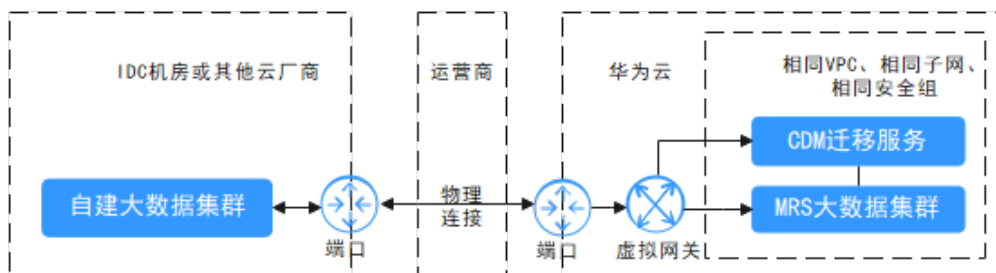
2.3 数据迁移到 MRS 前网络准备

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通的方式先打通两套集群之间网络连接。

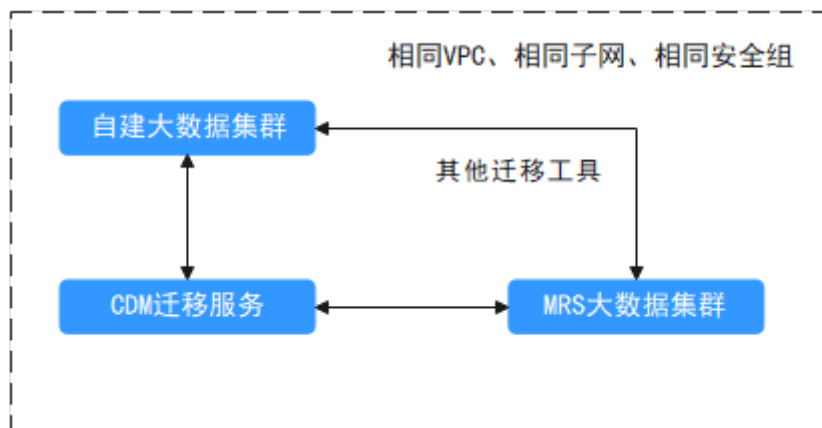
- 客户线下数据中心迁移数据到华为云 MRS 集群，通过云专线服务为用户搭建本地数据中心与云上 VPC 之间的专属连接通道。可以使用华为云的云专线服务或使用第三方的云专线服务来连通华为云网络。

图 2-3 线下数据中心迁移



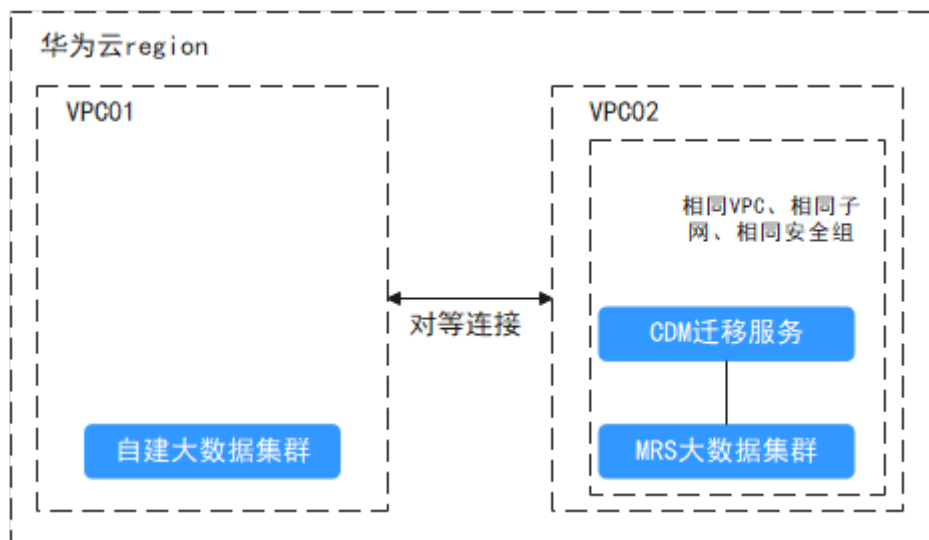
- 客户在华为云上自建大数据集群（或老版本的 MRS 集群）需要迁移到华为云 MRS 集群，且在同一个 Region 区域和 VPC 子网，可以使自建集群和 MRS 集群使用相同安全组、VPC、子网网络，从而保证网络连通。

图 2-4 线上同 Region 同 VPC 迁移



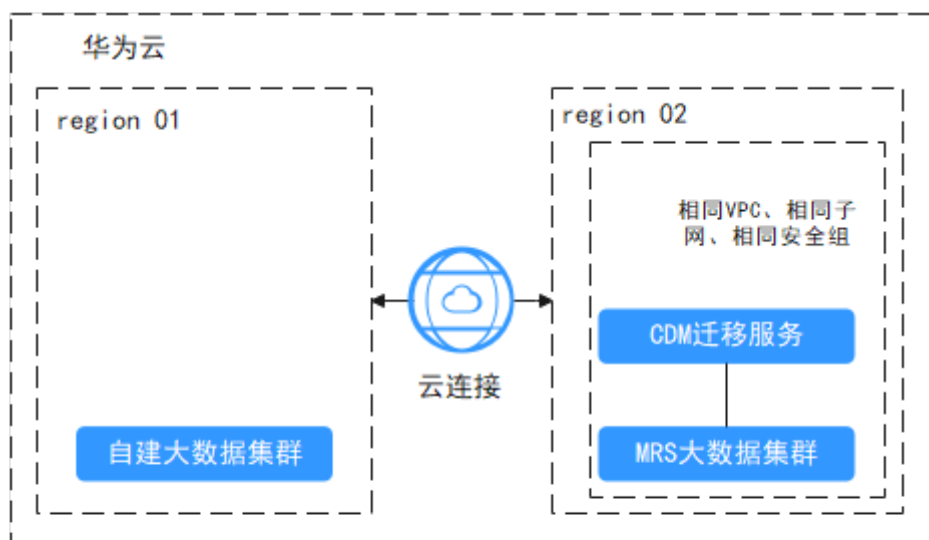
- 客户在华为云上自建大数据集群（或老版本的MRS集群）需要迁移到华为云MRS集群，且在同一个Region区域，但是使用不同VPC子网。需要使用VPC对等连接方式配置网络连通。

图 2-5 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 2-6 线上不同 Region 迁移



数据迁移常用端口

表 2-7 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。

2.4 数据迁移常见端口要求

HDFS 组件端口

表 2-8 HDFS 组件端口

配置参数	默认端口 (Hadoop 2.x和 Hadoop 3.x版本)	端口说明
dfs.namenode.rpc.port	9820	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 组件端口

表 2-9 ZooKeeper 组件端口

配置参数	默认端口	端口说明
clientPort	2181	ZooKeeper客户端连接ZooKeeper服务器。

Kerberos 组件端口

表 2-10 Kerberos 组件端口

配置参数	默认端口	端口说明
kdc_ports	21732	Kerberos服务认证，非Kerberos集群不涉及。

Hive 组件端口

表 2-11 Hive 组件端口

配置参数	默认端口 (Hive 2.x和Hive3.x版本)	端口说明
hive.metastore.port	9083	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 组件端口

表 2-12 HBase 组件端口

配置参数	默认端口 (HBase1.x和HBase 2.x版本)	端口说明
hbase.master.port	16000	HMaster RPC端口。该端口用于HBase客户端连接到HMaster。
hbase.regionserver.port	16020	RS (RegionServer) RPC端口。该端口用于HBase客户端连接到RegionServer。

Manager 组件端口

表 2-13 Manager 组件端口

配置参数	默认端口	端口说明
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。

2.5 使用 CDM 服务迁移 Hadoop 数据至 MRS 集群

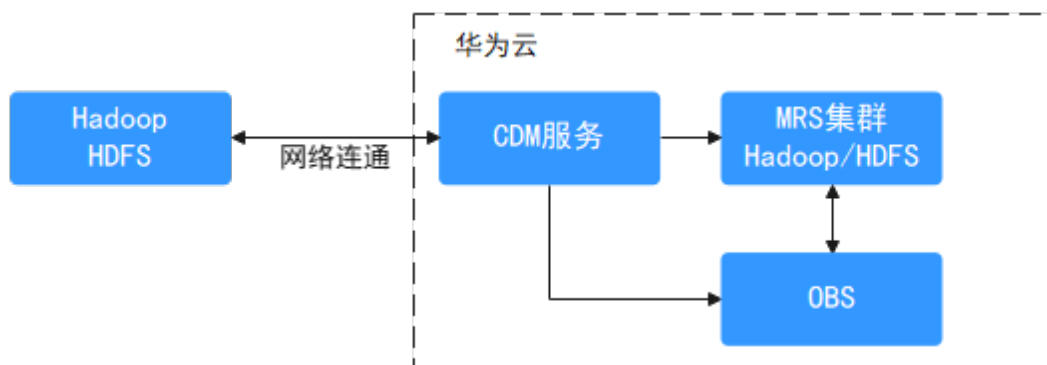
场景介绍

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。

图 2-7 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。

对系统影响

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。

操作步骤

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 2-8 HDFS 连接



步骤6 在“表/文件迁移”页签，单击“新建作业”。

步骤7 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8 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 2-9 配置作业参数

步骤9 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10 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14**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 2-10 任务配置

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HDFS文件数据迁移，并等待作业运行完成。

步骤12 登录迁移目的端集群主管理节点。

步骤13 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14 （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 2-11 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

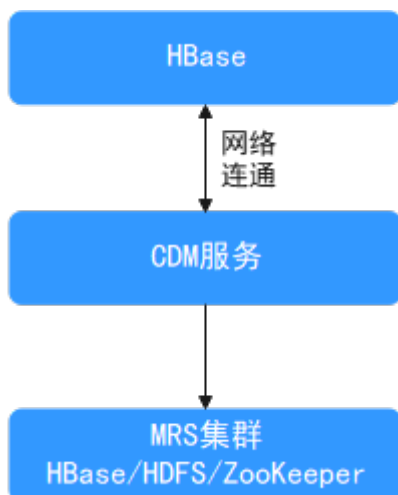
2.6 使用 CDM 服务迁移 HBase 数据至 MRS 集群

场景介绍

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。

图 2-12 HBase 数据迁移示意



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。

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。

对系统影响

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.user.name = 用户名（如omm用户）`”。

图 2-13 到迁移源端集群的连接

* 名称

* 连接器

* HBase类型

* ZK链接地址

* 认证类型

IP与主机名映射

* 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
<input type="text" value="hadoop.user.name"/>	<input type="text" value="omm"/>	删除

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 2-14 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	删除

[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 2-15 HBase 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

* 导入并清空数据 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 2-16 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 2-17 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 2-18 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

2.7 使用 CDM 服务迁移 Hive 数据至 MRS 集群

场景介绍

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。

Hive数据迁移分两部分内容：

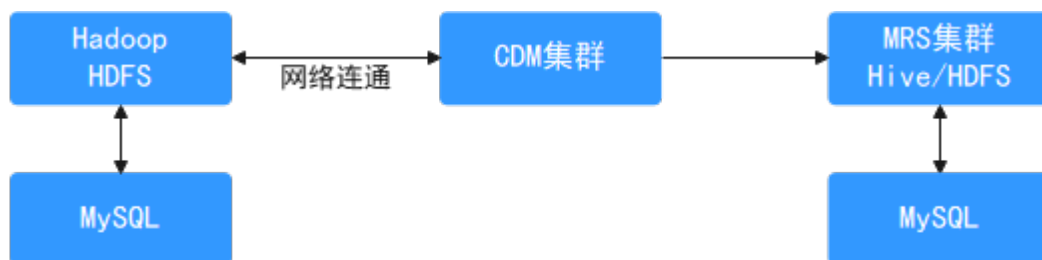
- Hive的元数据信息，存储在MySQL等数据库中。MRS Hive集群的元数据会默认存储到MRS DBService组件，也可以选择RDS（MySQL）作为外置元数据库。
- Hive的业务数据，存储在HDFS文件系统或OBS对象存储中。

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。

图 2-19 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。

对系统影响

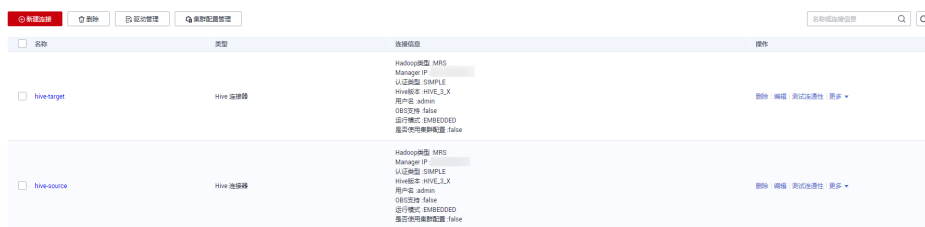
搬迁数据量较大时，对网络通信要求较高，执行搬迁任务时，可能会影响其他业务，建议在业务空闲期进行数据迁移任务。

操作步骤

- 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 2-20 创建 Hive 连接

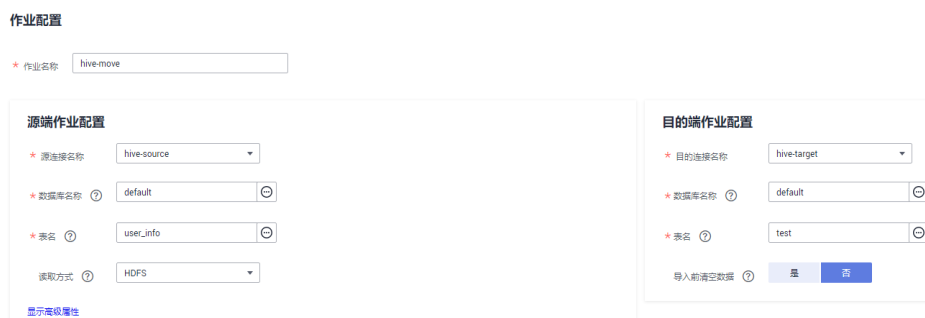


步骤6 在迁移目的端集群中创建数据迁移后的存储数据库。

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 2-21 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;
```


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。

基本内容如下所示：

1. [创建MRS离线查询集群](#)。
2. [将本地数据导入到HDFS中](#)。
3. [创建HBase表](#)。
4. [生成HFile文件并导入HBase](#)。

场景描述

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。

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 2-14 表 1 软件配置

参数项	取值
区域	中国-香港
计费模式	按需计费
集群名称	MRS_hbase
版本类型	普通版
集群版本	MRS 3.1.0
组件选择	HBase查询集群
可用区	可用区1
企业项目	default
虚拟私有云	vpc-01
子网	subnet-01
Kerberos认证	开启

参数项	取值
用户名	root/admin
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	勾选“确认授权”

图 2-24 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

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“创建桶”，填写以下参数，单击“立即创建”。

表 2-15 桶参数

参数项	取值
区域	中国-香港
桶名称	mrs-hbase
数据冗余存储策略	单AZ存储
默认存储类别	标准存储
桶策略	私有
默认加密	关闭
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“对象 > 上传对象”，将数据文件上传至OBS桶内。

图 2-25 上传数据文件



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桶名，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 2-26 导入数据



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hbasetest，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
hbasetest	人机		2021/05/14 11:04:31 GMT+08:00

用户名: hbasetest
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 hbasetest
```

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. 查看表是否创建成功，然后退出HBase Shell命令行界面。

list

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，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。

2.9 使用 CDM 服务迁移 MySQL 数据至 MRS Hive

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。

操作场景

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2-27](#)所示。

图 2-27 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中的导入到MRS Hive分区表，流程如下：

1. [在MRS Hive上创建Hive分区表](#)
2. [创建CDM集群并绑定EIP](#)
3. [创建MySQL连接](#)
4. [创建Hive连接](#)
5. [创建迁移作业](#)

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。

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。

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 2-28 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
[Cluster Name]	不可用	[IP]	[IP]	CDM	default	作业管理 绑定弹性IP 更多
[Cluster Name]	运行中	[IP]	-	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

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 2-29 选择连接器类型

数据仓库	数据仓库服务 (DWS)	数据湖探索 (DLI)		
Hadoop	MRS HDFS	MRS HBase	MRS Hive	Apache HDFS
	Apache HBase	Apache Hive		
对象存储	对象存储服务 (OBS)			
文件系统	FTP	SFTP	HTTP	
关系型数据库	云数据库 MySQL	云数据库 PostgreSQL	云数据库 SQL Server	MySQL
	PostgreSQL	Microsoft SQL Server	Oracle	
NoSQL	Redis	MongoDB		
消息系统	数据接入服务 (DIS)	MRS Kafka	Apache Kafka	
搜索	Elasticsearch			
公测中				

步骤2 选择“MySQL”后单击“下一步”，配置MySQL连接的参数。

图 2-30 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="checkbox"/> 是 <input checked="" type="checkbox"/> 否
使用Agent ?	<input checked="" type="checkbox"/> 是 <input type="checkbox"/> 否
Agent ?	<input type="text" value="agent1"/> ↗

[显示高级属性](#)

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2-16所示。

表 2-16 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

参数名	说明	取值样例
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。	-

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

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2-31所示。

图 2-31 创建 Hive 连接

* 名称

* 连接器

* Hadoop类型

* Manager IP [选择](#)

认证类型

* Hive版本

* 用户名

* 密码

* OBS支持

* 运行模式

[显示高级属性](#)

各参数说明如表2-17所示，需要您根据实际情况配置。

表 2-17 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"> • SIMPLE：非安全模式选择Simple鉴权。 • KERBEROS：安全模式选择Kerberos鉴权。 	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。 说明 <ul style="list-style-type: none"> • 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。 • 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。 • 仅具备Manager_tenant或Manager_auditor权限，无法创建连接。 	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"> • EMBEDDED：连接实例与CDM运行在一起，该模式性能较好。 • 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。 说明：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。 	EMBEDDED
是否使用集群配置	您可以通过使用集群配置，简化Hadoop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 管理集群配置 。	hive_01

步骤3 单击“保存”回到连接管理界面。

----结束

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2-32](#)所示。

图 2-32 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

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2-33](#)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 2-33 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

2.10 使用 CDM 服务迁移 MRS HDFS 数据至 OBS

操作场景

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。

流程如下：

1. [创建CDM集群并绑定EIP](#)
2. [创建MRS HDFS连接](#)
3. [创建OBS连接](#)
4. [创建迁移作业](#)

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。

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。

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

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

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2-34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桶类型：保持默认。
- 访问标识（AK）、密钥（SK）：登录OBS的AK、SK信息。

图 2-34 创建 OBS 连接

The screenshot shows a configuration form for creating an OBS connection. The fields are as follows:

- * 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) (?): ...

At the bottom, there are four buttons: "取消" (Cancel), "上一步" (Previous Step), "测试" (Test), and "保存" (Save).

步骤2 单击“保存”回到连接管理界面。

----结束

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 2-35 创建 MRS HDFS 到 OBS 的迁移任务

The screenshot shows the 'Task Configuration' tab of the CDM interface. It includes the following fields and options:

- 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
- 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; 重复文件处理方式: 替换重复文件

Buttons at the bottom: 取消 (Cancel), 下一步 (Next Step)

- 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[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[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

3 数据备份与恢复

3.1 HDFS 数据

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，需为两个集群配置互信。

HDFS 数据备份

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备份场景：

- 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元数据备份文件拷贝至目的集群。

```
$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工具将源集群数据拷贝到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数据拷贝到目的集群的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拷贝，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进行拷贝。

HDFS 元数据备份

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，且以超级管理员身份运行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。

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>
```

3.2 Hive 元数据

Hive 元数据备份

Hive表数据存储存储在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。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。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

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。

3.3 Hive 数据

当前没有单独的Hive数据备份，请参考[HDFS数据](#)进行操作。

3.4 HBase 数据

当前HBase上可以使用的数据备份方式主要有以下几种方式，本指导为您介绍通过以下几种方式进行HBase数据导出、HBase数据导入的操作步骤。

- Snapshots
- Replication

- Export
- CopyTable
- HTable API
- Offline backup of HDFS data

表3-1分别从操作对集群的性能影响、数据空间消耗、业务中断影响、增量备份、易用性、可恢复性几个维度进行对比。

表 3-1 HBase 上数据备份对比

备份方式	性能影响	数据空间消耗	业务中断影响	增量备份	易用性	可恢复性
Snapshots	低	小	短暂中断（仅恢复时）	不支持	易	秒
Replication	低	大	无	固有	中	秒
Export	高	大	无	支持	易	高
CopyTable	高	大	无	支持	易	高
HTable API	中	大	无	支持	难	取决于用户
Offline backup of HDFS data	-	大	长时间中断	不支持	中	高

Snapshots

对表执行snapshot操作生成快照，既可以作为原表的备份，当原表出现问题的时候可以回滚恢复，也可以作为跨集群的数据备份工具。执行快照会在当前HBase上的根目录（默认为/hbase）生成“.hbase-snapshot”目录，里面有每个快照的详细信息。当执行ExportSnapshot导出快照时，会在本地提交MR任务，将快照信息以及表的HFile分别拷贝到备集群的/hbase/.hbase-snapshot和/hbase/archive中。详情请参考<http://hbase.apache.org/2.2/book.html#ops.snapshots>。

- 该方式数据备份的优点：
单表备份效率高，在线数据本地/远程备份，不中断主集群和备集群业务，可以灵活配置map的个数和限制流量，MapReduce的执行节点可不在主备集群（不占资源）。
- 该方式数据备份的缺点和限制：
只能单表操作，备份的表名在snapshot中已经指定无法更改，且无法增量备份，运行MR需要占用本地集群资源。

在主集群执行如下操作：

步骤1 对表创建快照。例如对表member创建快照member_snapshot。

```
snapshot 'member','member_snapshot'
```

步骤2 将快照拷贝到备集群上。

```
hbase org.apache.hadoop.hbase.snapshot.ExportSnapshot -snapshot  
member_snapshot -copy-to hdfs://备集群HDFS服务主NameNode节点IP:端口号/  
hbase -mappers 3
```

- 备集群的数据目录必须为HBASE根目录（/hbase）
- mappers表示MR任务需要提交的map个数

----结束

在备集群执行如下操作：

使用restore命令在备集群自动新建表，以及与archive里的HFile建立link。

```
restore_snapshot 'member_snapshot'
```

📖 说明

如果只是备份表数据的话，建议使用此种方式备份，SnapshotExport会在本地提交MR任务，将Snapshot和HFile拷贝到备集群，之后可以在备集群直接加载数据，效率比其他方式高很多。

Replication

Replication备份是在HBase上建立主备集群的容灾关系，当数据写入主集群，主集群通过WAL来主动push数据到备集群上，从而达到主备集群的实时同步。详情请参考http://hbase.apache.org/2.2/book.html#_cluster_replication。

- 该方式数据备份的优点：
 - 使用replication有别于其他几种数据备份导入方式，当配置了集群间的主备关系后，数据可以实时同步（无需人为操作）。
 - 相对而言，“备份”的动作占用集群的资源较少，对集群的性能影响小。
 - 数据同步可靠性较高，如果备集群停止一段时间后再恢复，这中间主机群的数据依然会同步到备集群。
- 该方式数据备份的缺点和限制：
 - 如果客户端写入的数据设置不写WAL，则数据无法备份到备集群。
 - 由于占用的资源少，后台是通过异步的方式同步数据，实际数据没有实时同步。
 - 对于开启表replication同步之前，主集群就已经存在的数据无法同步，需要借助其他方式导入的备集群。
 - bulkload方式写入到主集群的数据无法同步（MRS上的HBase对replication做了增强，支持bulkload on replication）。

具体的使用和配置方法请参考[配置HBase备份](#)和[使用ReplicationSyncUp工具](#)来进行备份数据。

Export/Import

Export/Import主要是启动MapReduce任务对数据的表进行扫描（scan），往远端HDFS写入SequenceFile，之后Import再把SequenceFile读出来写入HBase（put）中。

- 该方式数据备份的优点：
在线拷贝不中断业务，由于是scan->put的方式写入新表，所以比CopyTable更加灵活，可灵活配置需要获取的数据，数据可增量写入。
- 该方式数据备份的缺点和限制：
由于Export是通过MapReduce任务往远端HDFS写入SequenceFile，之后Import再把SequenceFile读出来写入HBase，需要执行两次MapReduce任务，实际效率不高。

在主集群执行如下操作：

执行Export命令导出表。

```
hbase org.apache.hadoop.hbase.mapreduce.Export <tablename> <outputdir>
```

例如：`hbase org.apache.hadoop.hbase.mapreduce.Export member hdfs://备集群HDFS服务主NameNode节点IP:端口号/user/table/member`

其中，member为待导出表的名称。

在备集群执行如下操作：

步骤1 主集群执行完之后可以在备集群上查看生成的目录数据如图3-1。

图 3-1 目录数据

```
Cvtl:- # hdfs dfs -ls -R /user/table/member
-rw-r--r--  3 root hadoop          0 2018-06-28 14:18 /user/table/member/_SUCCESS
-rw-r--r--  3 root hadoop      2937 2018-06-28 14:18 /user/table/member/part-m-00000
```

步骤2 执行create命令在备集群上新建与主集群相同结构的表，例如member_import。

步骤3 执行Import命令生成HFile数据在HDFS上。

```
hbase org.apache.hadoop.hbase.mapreduce.Import <tablename> <inputdir>
```

例如：`hbase org.apache.hadoop.hbase.mapreduce.Import member_import /user/table/member -Dimport.bulk.output=/tmp/member`

- member_import为备集群上与主集群相同表结构的表
- Dimport.bulk.output为输出的HFile数据目录
- /user/table/member为从主集群上导出的数据目录

步骤4 执行Load操作将HFile数据写入HBase。

```
hbase org.apache.hadoop.hbase.mapreduce.LoadIncrementalHFiles /tmp/member member
```

- /tmp/member为步骤3中HFile数据的输出目录
- member为备集群上要导入数据的表名

----结束

CopyTable

拷贝表功能与导出功能类似，拷贝表也使用HBase API创建了一个MapReduce任务，以便从源表读取数据。不同的地方是拷贝表的输出是hbase中的另一张表，这张表可以在本地集群，也可以在远程集群。详情请参考<http://hbase.apache.org/2.2/book.html#copy.table>。

- 该方式数据备份的优点：
操作简单，在线拷贝不中断业务，可以指定备份数据的startrow/endrow/timestamp。
- 该方式数据备份的缺点和限制：
只能单表操作，远程拷贝数据量大时效率较低，MapReduce需要占用本地资源，MapReduce的map个数以表region的个数划分。

在备集群执行如下操作：

执行create命令在备集群上新建与主集群相同结构的表，例如member_copy。

在主集群执行如下操作：

执行CopyTable的命令拷贝表。

```
hbase org.apache.hadoop.hbase.mapreduce.CopyTable [--starttime=xxxxxx]
[--endtime=xxxxxx] --new.name=member_copy --
peer.adr=server1,server2,server3:2181:/hbase [--
families=myOldCf:myNewCf,cf2,cf3] TestTable
```

- starttime/endtime为待拷贝数据的时间戳。
- new.name为备集群中目的表的表名，默认为和原来表名相同。
- peer.adr为备集群zookeeper节点的信息，格式为quorum:port:/hbase。
- families为待拷贝的表的family列。

📖 说明

如果是拷贝数据到远端集群，此种方式导入数据会在主机群上提交MapReduce任务，读取原始表的全量/部分数据之后采用put的方式写入远端集群，所以如果表的数据量很大（远程拷贝不支持bulkload），则效率会比较低。

HTable API

API的方式主要也是在代码中对原始HBase表的数据导入导出，你可以使用公用的API写自己定制的客户端应用程序来直接查询表格，或通过MapReduce任务的批量处理优势自己设计其他方法。该方式需要对Hadoop开发以及因此对生产集群带来的影响有深入的理解。应用程序开发详情请参考[MRS开发指南](#)。

Offline backup of HDFS data

离线备份HDFS数据，即关闭HBase服务并手工在HDFS上拷贝数据。

- 该方式数据备份的优点：
 - 可以把主集群上所有数据（包含元数据）整个复制到备集群。
 - 由于是通过Distcp直接拷贝的，所以数据备份的效率相对较高。
 - 实际操作时可以根据具体的需求灵活拷贝，可以只拷贝其中一个表的数据，也可以拷贝region中的其中一个HFile等。
- 该方式数据备份的缺点和限制：
 - 此操作会覆盖备集群上的HDFS的数据目录。
 - 如果主备集群间的HBase版本不同，HDFS目录直接拷贝可能会出现问題，例如MRS上的hbase1.3版本新增了系统表index，如果使用老版本的HDFS目录直接覆盖，会找不到该数据表。所以此种方案在执行前需要慎重考虑。

- 此操作对用户使用HBase的能力有一定的要求，如出现异常情况需要根据实际情况执行恢复。

在主集群执行如下操作：

步骤1 执行如下命令将当前集群内存中的数据持久化到HDFS中。

```
flush 'tableName'
```

步骤2 停止HBase服务。

步骤3 使用distcp命令拷贝当前集群HDFS上的数据到备集群上。

```
hadoop distcp -i /hbase/data hdfs://备集群HDFS服务主NameNode节点IP:端口号/hbase
```

```
hadoop distcp -update -append -delete /hbase/ hdfs://备集群HDFS服务主NameNode节点IP:端口号/hbase/
```

第二条命令为增量拷贝除了data目录以外的文件，例如archive里面的数据可能还有被数据目录所引用。

----结束

在备集群执行如下操作：

步骤1 重启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。

步骤2 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
```

步骤3 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
```

说明

当用户使用了HBase协处理器，自定义jar包放在主集群的regionserver/hmaster上时，在备集群重启HBase之前，需要把这些自定义jar包也拷贝过来。

----结束

3.5 Kafka 数据

当需要在两个Kafka集群间作数据同步，或者将原有Kafka集群上的数据搬迁到新的Kafka集群上时就需要用到Kafka数据同步的利器——MirrorMaker。MirrorMaker是Kafka内嵌的一个工具，其内部实际上是集成了Kafka的Consumer和Producer，它可以从一个Kafka集群消费数据然后写入另一个Kafka集群，从而实现Kafka集群间的数据同步。

本章节介绍利用MRS服务提供的MirrorMaker工具实现Kafka集群数据同步、迁移的方法，请先参考[打通数据传输通道](#)完成两个集群的网络互通后再参考本章节操作Kafka数据迁移。

操作步骤

3.x之前版本：

步骤1 配置集群Kerberos互信。

步骤2 若计划在源集群使用MirrorMaker工具，请登录目的集群的集群详情页面，选择“组件管理”。若计划在目的集群使用MirrorMaker工具，请登录源集群的集群详情页面，选择“组件管理”。

步骤3 选择“Kafka > 服务配置”，将页面右侧的“基础配置”切换为“全部配置”。

步骤4 选择“Broker > 自定义”配置项增加如下规则。

```
sasl.kerberos.principal.to.local.rules = RULE:[1:$1@$0]
(*@XXXYYYYZZZ.COM)s/@.*//,RULE:[2:$1@$0](.*@
XXXYYYYZZZ.COM)s/@.*//,DEFAULT
```

其中：XXXYYYYZZZ.COM为数据发送端集群（源集群）的域名(字母需大写)。

图 3-2 增加规则



步骤5 单击“保存配置”，按照提示选择“重新启动受影响的服务或实例。”并单击“确定”，重启Kafka服务。

说明

开启Kerberos认证的安全集群需要执行步骤**步骤1-步骤5**，未开启Kerberos认证的普通集群请跳过步骤**步骤1-步骤5**，直接从**步骤6**开始执行。

步骤6 在计划使用MirrorMaker工具的集群，登录集群详情页面，选择“组件管理”。

步骤7 选择“Kafka > 服务配置”，切换“基础配置”为“全部配置”，“角色”为“MirrorMaker”。

图 3-3 配置 Kafka 服务的 MirrorMaker 参数



配置参数说明：

- source和dest标签中的bootstrap.servers参数分别是源Kafka集群和目的Kafka集群的broker节点列表及端口信息
- source和dest标签中的security.protocol参数请根据源Kafka集群和目的Kafka集群的服务端的实际配置情况自行选择
- 如果源Kafka集群（source）或者目的Kafka集群（dest）为安全集群则需要配置source和dest标签中的kerberos.domain.name（如果是本机可不配置，非本机根据实际值进行配置）和sas.l.kerberos.service.name（默认：kafka）参数
- 在mirror标签中设置whitelist，即需要同步的topic名称。

步骤8 单击“保存配置”，按照提示选择“重新启动受影响的服务或实例。”并单击“是”，重启MirrorMaker实例。

待MirrorMaker重启完成后，数据迁移任务即已启动，可通过[KafkaManager](#)或者命令行工具监控MirrorMaker数据同步进度。

----结束

4 周边云服务对接

4.1 使用 MRS Spark SQL 访问 DWS 表

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。

📖 说明

本章节仅适用于MRS 3.x及之后版本。

前提条件

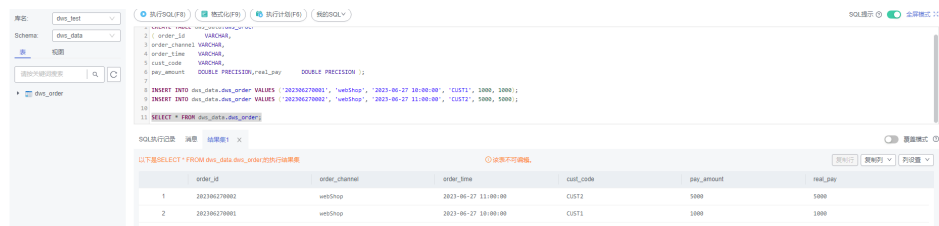
- [创建MRS集群](#)，包含Spark组件。
- 如果MRS集群开启了Kerberos认证，登录FusionInsight Manager页面，选择“系统 > 权限 > 用户”，添加一个人机用户sparkuser，用户组（hadoop、hive），主组（hadoop）。并参考[添加Spark2x的Ranger访问权限策略](#)章节，添加“ADD JAR操作”权限。如果MRS集群未开启Kerberos认证，则无需创建用户。
- 安装MRS集群客户端，具体请参考[安装客户端](#)。
- [创建GaussDB\(DWS\)集群](#)，为确保网络连通，GaussDB (DWS)集群需与MRS集群“可用区”、“虚拟私有云”、“安全组”配置相同。
- 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。

操作步骤

步骤1 准备数据，在GaussDB(DWS)集群中创建数据库和表：

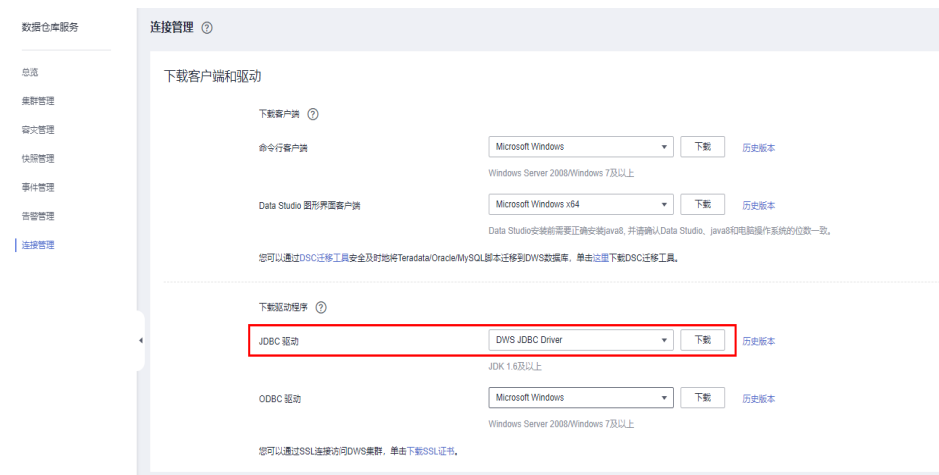
1. 登录GaussDB(DWS)管理控制台，单击DWS集群“操作”列的“登录”。
2. 登录现有GaussDB(DWS)集群的默认数据库gaussdb，执行以下命令，创建数据库“dws_test”。

- 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，单击左侧的“连接管理”，下载JDBC驱动，如下图所示：



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 tmp1# 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/im
pl/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/Stati
cLoggerBinder.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 tmp1#
```

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.car
bondata.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 s
ystem. Use the default value HDFSLOCK instead. | org.apache.carbondata.core.util.CarbonProperties.validateAndConfigureLockType(C
arbonProperties.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

4.2 Hive 对接 CSS 服务

操作场景

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。

📖 说明

该章节适用于MRS 3.x及之后版本。

前提条件

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。

操作步骤

- 步骤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"
}
```

- 步骤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。

- 步骤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4 将**步骤3**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
```

步骤5 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。

步骤6 在**Maven中心仓**下载“commons-httpclient-3.1.jar”，并将该Jar包及**步骤3**的Jar上传至集群中安装了HDFS和Hive客户端的任一节点上。

步骤7 以客户端安装用户，登录**步骤6**上传了Jar包的节点。

步骤8 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）
```

步骤9 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
```

步骤10 执行以下命令将**步骤6**的Jar上传至HDFS中。

```
hdfs dfs -put Jar包存放路径 Jar包存放在hdfs中的路径
```

步骤11 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）
```

步骤12 执行以下命令创建Elasticsearch外部表。

```
CREATE EXTERNAL TABLE `ddj_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'='ddj_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中创建的具有步骤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：步骤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>。

步骤13 执行以下命令查看步骤12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：

```
jdbc:hive2://192.168.0.129:10000/> select * from ddj_study_card_ratio_v12_test002;  
INFO : State: Compiling.  
INFO : Compiling command(queryId=ommm_20220727154319_1c7f3fdf-5c8c-4c3d-80a2-f95888bcc5a4): select * from ddj_study_card_ratio_v12_test002; Current sessionId=a9f6dd55-f30  
INFO : hive.compile.auto.avoid.cbostTrue  
INFO : Concurrency mode is disabled, not creating a lock manager  
INFO : Current sql is not contains smart syntax, not need record dest table flag  
INFO : Semantic Analysis Completed (retrial = false)  
INFO : Returning hive schema: Schema(FieldSchemas:[FieldSchema(name:ddj_study_card_ratio_v12_test002_uniq_id, type:string, comment:null)], properties:null)  
INFO : Completed compiling command(queryId=ommm_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=ommm_20220727154319_1c7f3fdf-5c8c-4c3d-80a2-f95888bcc5a4): select * from ddj_study_card_ratio_v12_test002; Current sessionId=a9f6dd55-f30  
INFO : Completed executing command(queryId=ommm_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_uniq_id |  
+-----+  
NULL |  
+-----+  
row selected [0.0] seconds)
```

---结束

5 生态组件对接

5.1 使用 DBeaver 访问 Phoenix

本章节以DBeaver 6.3.5版本为例，讲解如何访问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。

前提条件

- 已安装DBeaver 6.3.5，DBeaver软件下载链接为：https://dbeaver.io/files/6.3.5/dbeaver-ce-6.3.5-x86_64-setup.exe。
- 已创建包含HBase组件的MRS 3.1.0未开启Kerberos认证的集群。
- 已安装HBase客户端。
- 已安装JDK 1.8.0_x。

操作步骤

步骤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 5-1 新增 JDK 的 bin 目录

```
-vm  
C:\Program Files\Java\jdk1.8.0_121\bin
```

步骤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”。

步骤3 从客户端安装节点上的“客户端安装目录/HBase/hbase/conf”目录下下载“hbase-site.xml”文件。使用压缩软件打开**步骤2**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。

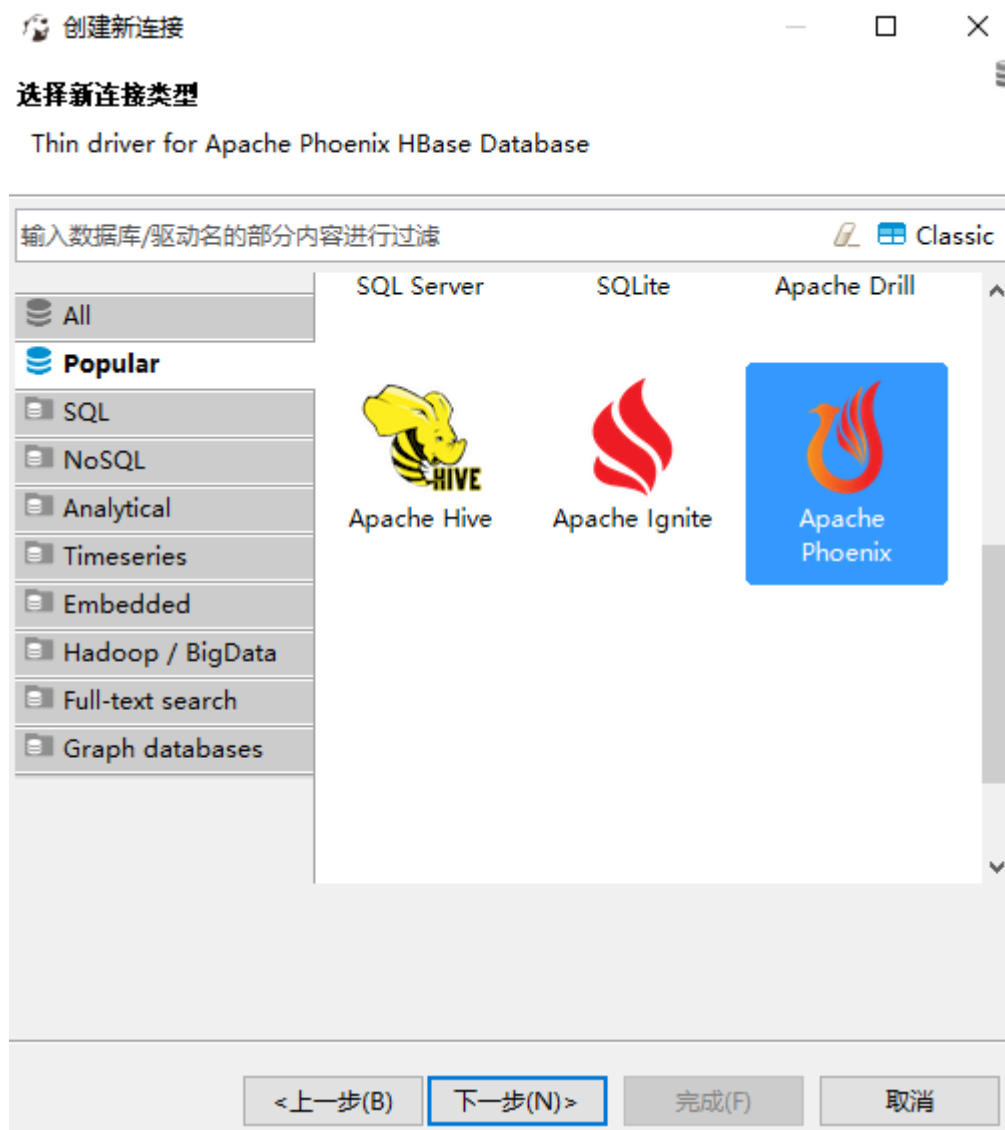
步骤4 打开DBeaver，选择“文件 > 新建 > DBeaver > 数据库连接”。

图 5-2 创建数据库连接



步骤5 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 5-3 选择数据连接类型



步骤6 单击“编辑驱动设置”。

图 5-4 编辑驱动设置



步骤7 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 5-5 删除原有的驱动包

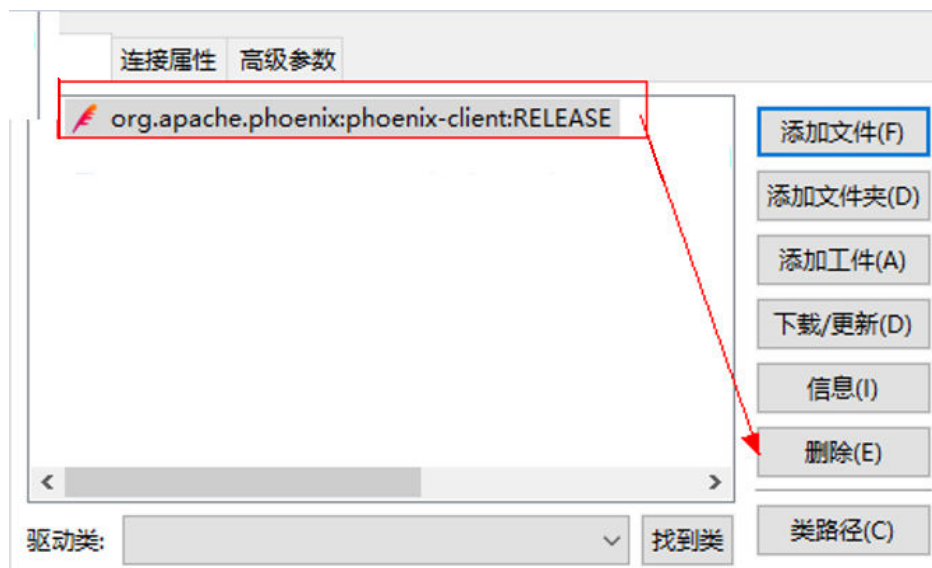
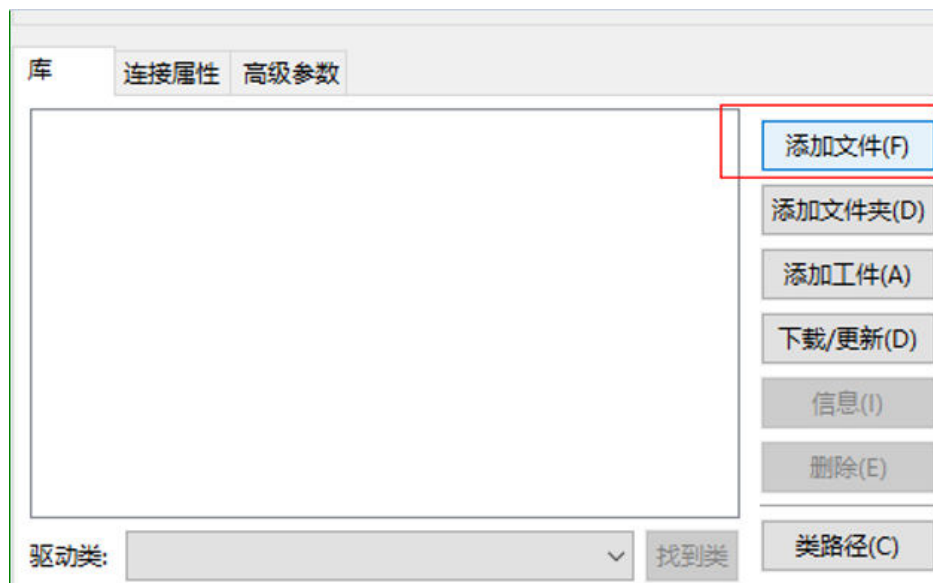
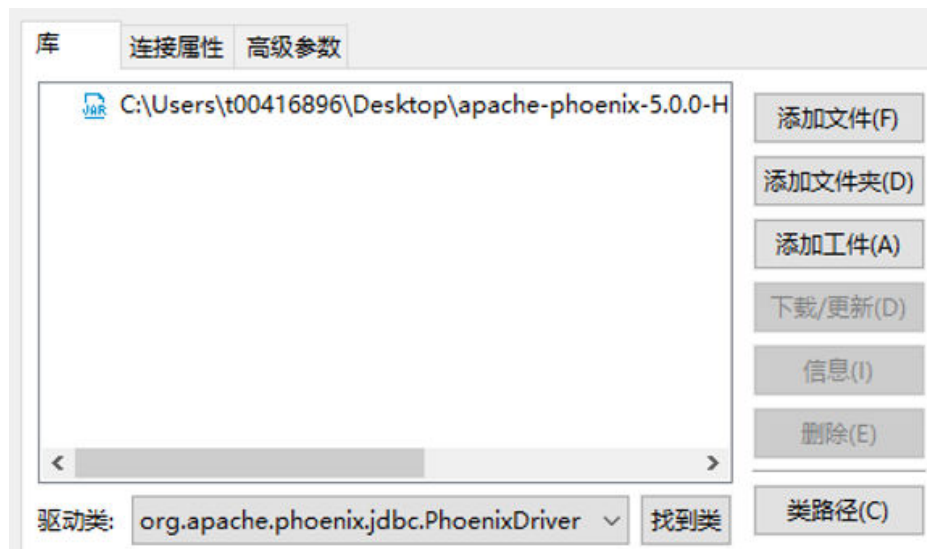


图 5-6 添加 Phoenix Jar 包



步骤8 单击“找到类”，加载完成后在左侧框中选择“org.apache.phoenix.jdbc.PhoenixDriver”。

图 5-7 加载驱动类



步骤9 添加Zookeeper Base Path。

1. 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 5-8 查看 Zookeeper Base Path

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	15ad5f94eaf31a9cb0fbff55bd79
ZooKeeper Client Version	?, revision=12-c9b3def3b445dca9f3ad21427ec3846b81a92453
ZooKeeper Client Compiled	04/28/2021 10:20 GMT
ZooKeeper Quorum	node-master1jfmD:2181 node-master2uIQz:2181 node-master3xcpw:2181
ZooKeeper Base Path	/hbase

2. 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 5-9 配置 URL 模板

设置

驱动名称*: 驱动类型:

类名:

URL 模板:

默认端口: 嵌入 无认证 Allow Empty Password

描述

目录: ID:

描述:

网址: <http://phoenix.apache.org/>

- 步骤10** 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.
```

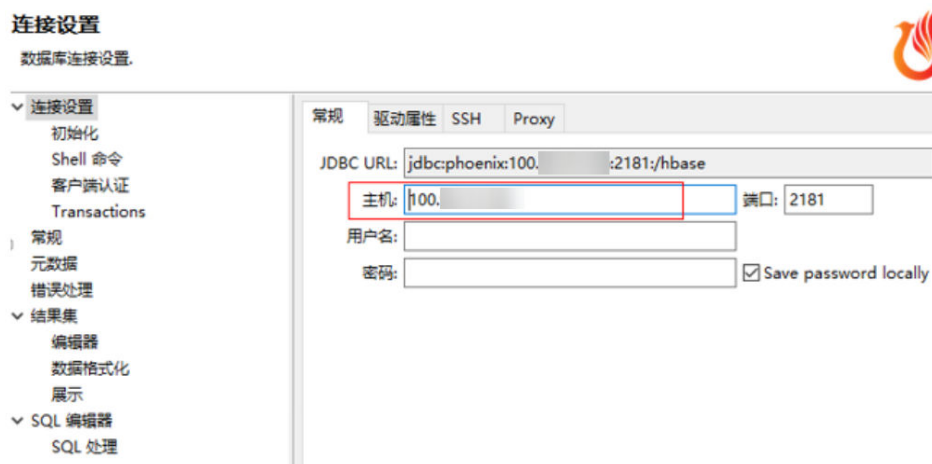
说明

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。

步骤11 登录FusionInsight Manager，选择“集群 > 服务 > ZooKeeper > 实例”。

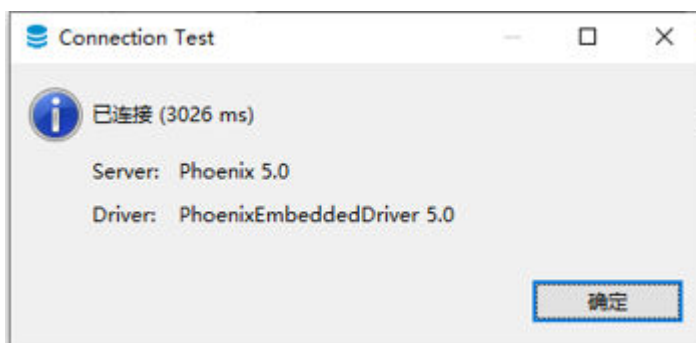
选择任一节点，将该节点对应的EIP填入主机框中（如果使用的是Windows云服务器，并且与集群的网络是通的，直接填写节点的业务IP地址即可）。

图 5-10 配置主机



步骤12 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 5-11 测试连接



步骤13 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14 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：

1. 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');

2. 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');
```

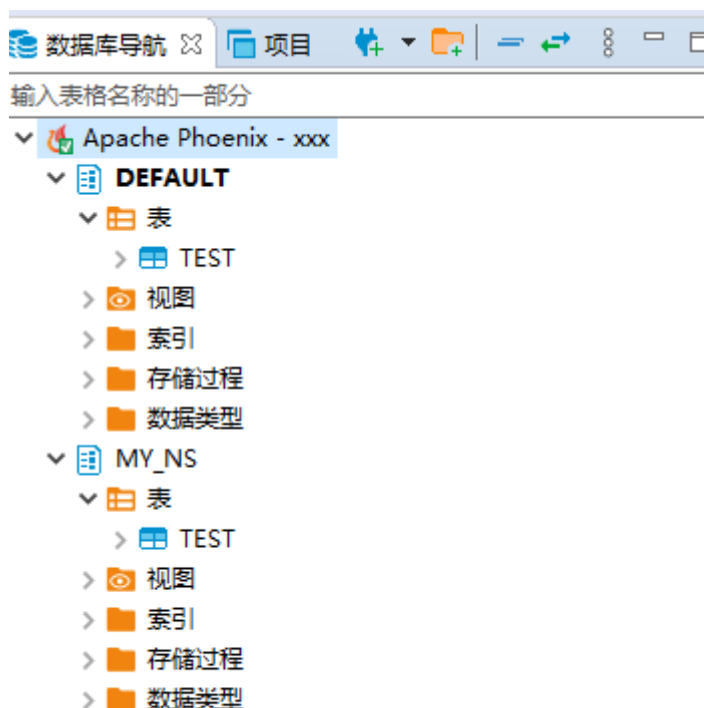
```
UPSERT INTO MY_NS.TEST VALUES(2,'Tom');
```

```
UPSERT INTO MY_NS.TEST VALUES(3,'Manson');
```

```
UPSERT INTO MY_NS.TEST VALUES(4,'Aurora');
```

- 步骤15** 右键单击连接名称，单击“刷新”，再单击连接名称左侧的 \vee ，即可查看在 `default` 和 `MY_NS` 中新建的表：

图 5-12 查看表



----结束

5.2 使用 DBeaver 访问 HetuEngine

本章节以 DBeaver 7.2.0 版本为例，讲解如何访问 HetuEngine。

前提条件

- DBeaver 已正常安装。DBeaver 软件下载链接：<https://dbeaver.io/files/7.2.0/>。
- 已在集群中创建“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](#)。

- 已创建计算实例并运行正常，可参考[创建HetuEngine计算实例](#)。

操作步骤

步骤1 下载HetuEngine客户端获取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3 打开DBeaver，选择“数据库 > 新建连接”，在“ALL”中搜索“PrestoSQL”并打开PrestoSQL。

步骤4 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

参数名	参数值
URL模板	<ul style="list-style-type: none"> 通过HSFabric方式访问HetuEngine <code>jdbc:presto:// <HSFabricIP1:port1>,<HSFabricIP2:port2>,<HSFabricIP3:port3>/hive/ default?serviceDiscoveryMode=hsfabric</code> 示例: <code>jdbc:presto:// 192.168.42.90:29902,192.168.42.91:29902,192.168.42.92:29902/hive/ default?serviceDiscoveryMode=hsfabric</code> 通过HSBroker方式访问HetuEngine <code>jdbc:presto:// <HSBrokerIP1:port1>,<HSBrokerIP2:port2>,<HSBrokerIP3:port3>/ hive/default?serviceDiscoveryMode=hsbroker</code> 示例: <code>jdbc:presto:// 192.168.42.90:29860,192.168.42.91:29860,192.168.42.92:29860/hive/ default?serviceDiscoveryMode=hsbroker</code>

📖 说明

- 获取HSFabric、HSBroker节点IP及端口号：
 - 登录FusionInsight Manager。
 - 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。
 - 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。

步骤5 单击“添加文件”，上传**步骤1**中获取的JDBC驱动包。

步骤6 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 5-13 安全模式驱动设置

设置

驱动名称: PrestoSQL 驱动类型: Generic

类名: io.prestosql.jdbc.PrestoDriver

URL 模板: presto://192.168.42.90:29902/hive/default?serviceDiscoveryMode=hsfabric

默认端口: 8080 Default Database:

Default User:

嵌入 无认证 Allow Empty Password Use legacy JDBC instantiation

描述

目录: ID: prestoql_jdbc

描述: Community Presto (prestoql) JDBC driver

网址: <https://prestosql.io/>

库 连接属性 高级参数

D:\test\hetu-jdbc- -SNAPSHOT.jar

添加文件(F) 添加文件夹(D) 添加工件(A) 下载/更新(D) 信息(I) 删除(E) 类路径(C)

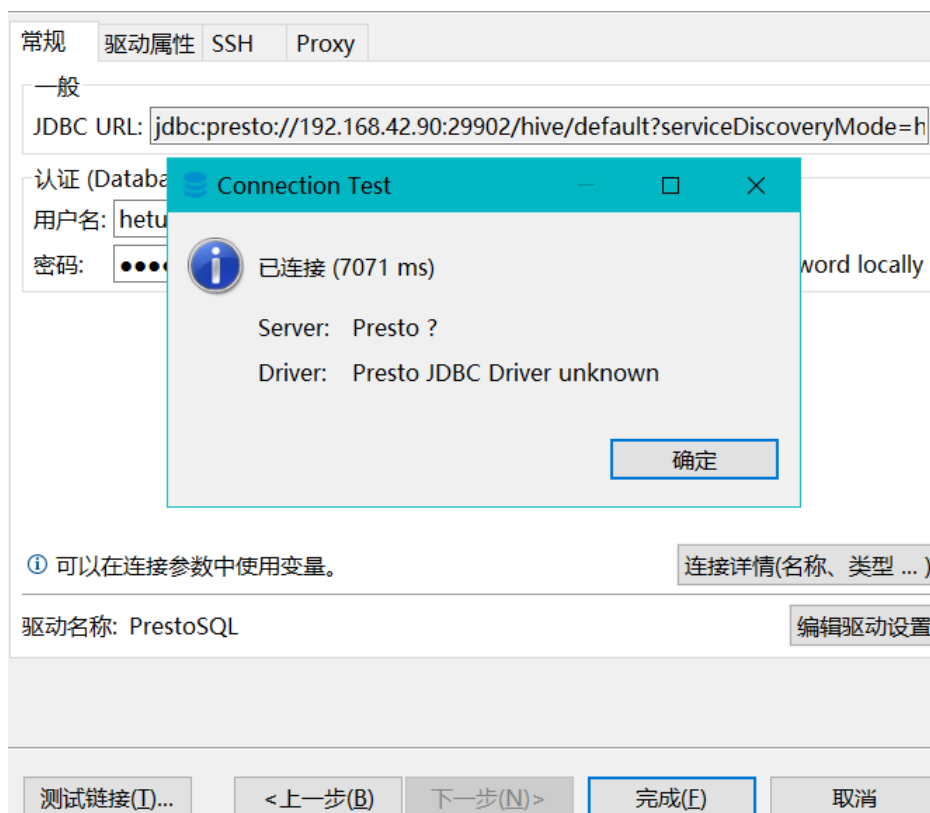
驱动类: io.prestosql.jdbc.PrestoDriver 找到类

重置为默认状态 确定 取消

步骤7 连接设置。

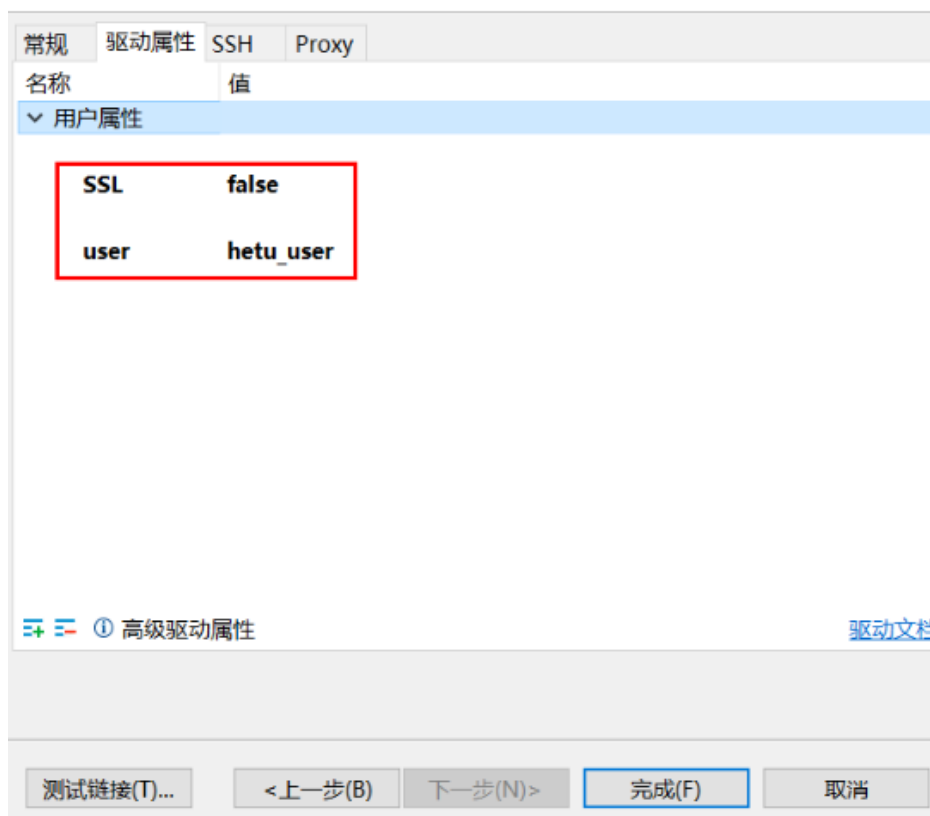
- 安全模式（开启Kerberos认证的集群）：
在创建新连接页面的“常规”页签，输入前提条件中创建的用户名和密码，单击“测试链接”，连接成功后，单击“确定”，再单击“完成”。可单击“连接详情（名称、类型...）”修改连接名称。

图 5-14 安全模式“常规”参数设置



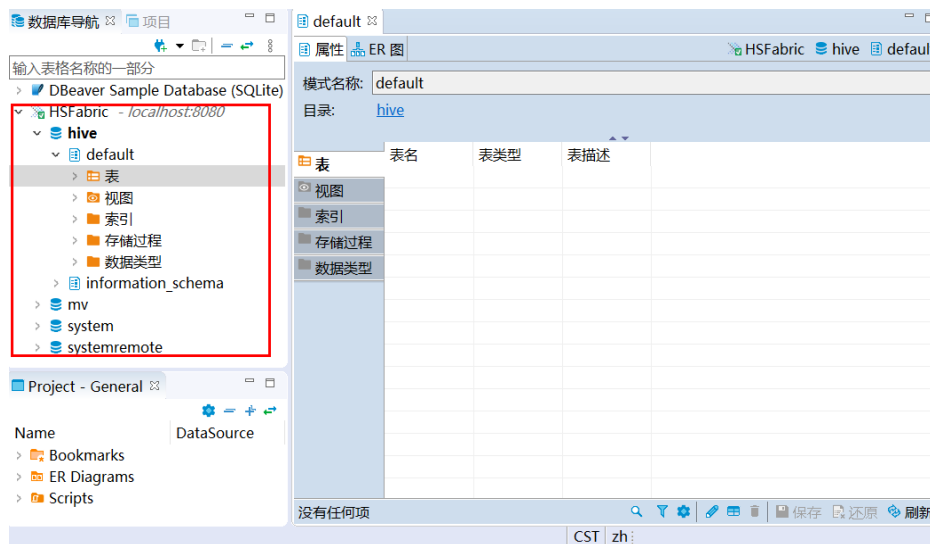
- 普通模式（未开启Kerberos认证的集群）：
在创建新连接页面的“驱动属性”配置如下参数，“user”为前提条件中创建的用户。单击“测试链接”，连接成功后，单击“确定”，再单击“完成”。可单击“连接详情（名称、类型...）”修改连接名称。

图 5-15 普通模式“驱动属性”参数设置



步骤8 连接成功后进入如图5-16所示页面。

图 5-16 连接成功



----结束

5.3 使用 FineBI 访问 HetuEngine

本章节以FineBI 5.1.9版本为例，讲解如何访问安全模式集群的HetuEngine。

前提条件

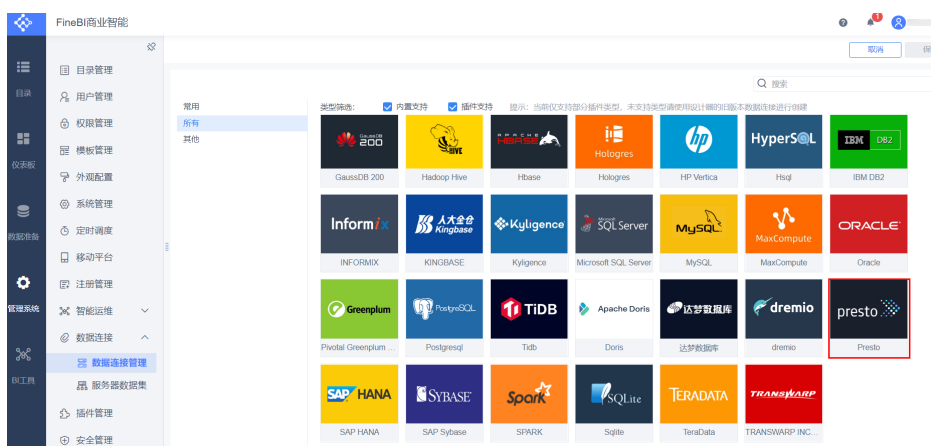
- 已安装FineBI。
- 已获取JDBC jar包文件，获取方法请参考[步骤1](#)。
- 已在集群中创建“人机”用户，如`hetu_user`，可参考[创建HetuEngine用户](#)。启用Ranger鉴权的集群需根据业务需求为该`hetu_user`添加Ranger权限，可参考[添加HetuEngine的Ranger访问权限策略](#)。
- 已创建计算实例并运行正常，可参考[创建HetuEngine计算实例](#)。

操作步骤

步骤1 将获取的jar包放在目录“%FineBI%\webapps\webroot\WEB-INF\lib”，重启FineBI。

步骤2 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5-17](#)新建连接。

图 5-17 新建连接



步骤3 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

参数名	参数值
密码	已创建的“人机”用户的用户密码 说明 <ul style="list-style-type: none"> 使用用户名密码方式登录时需要配置该参数。 未启用Kerberos认证（普通模式）的集群不填写该参数。
编码	自动
数据库连接URL	<ul style="list-style-type: none"> 启用Kerberos认证（安全模式）的集群 jdbc:presto:// <HSFabricIP1:port1>,<HSFabricIP2:port2>,<HSFabricIP3:port3>/hive/default?serviceDiscoveryMode=hsfabric，详情请参考表5-1。 未启用Kerberos认证（普通模式）的集群 jdbc:presto:// <HSFabricIP1:port1>,<HSFabricIP2:port2>,<HSFabricIP3:port3>/hive/default? serviceDiscoveryMode=hsfabric&SSL=false。

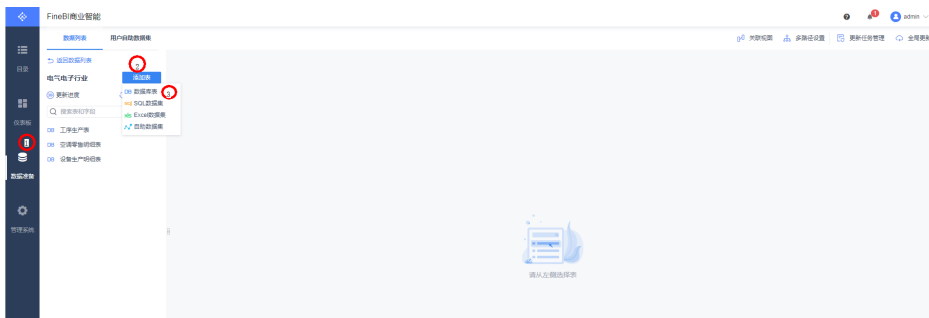
表 5-3 HSbroker 连接参数

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 说明 <ul style="list-style-type: none"> 使用用户名密码方式登录时需要配置该参数。 未启用Kerberos认证（普通模式）的集群不填写该参数。
编码	自动

参数名	参数值
数据库连接URL	<ul style="list-style-type: none"> • 启用Kerberos认证（安全模式）的集群 jdbc:presto:// <HSBrokerIP1:port1>,<HSBrokerIP2:port2>,<HSBrokerIP3:port3>/hive/default?serviceDiscoveryMode=hsbroker, 详情请参考表5-1。 • 未启用Kerberos认证（普通模式）的集群 jdbc:presto:// <HSBrokerIP1:port1>,<HSBrokerIP2:port2>,<HSBrokerIP3:port3>/hive/default? serviceDiscoveryMode=hsbroker&SSL=false

步骤4 参考图5-18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 5-18 配置数据



步骤5 设置需要用于做分析的表，如图5-19~图5-21所示。

图 5-19 单击“数据连接”

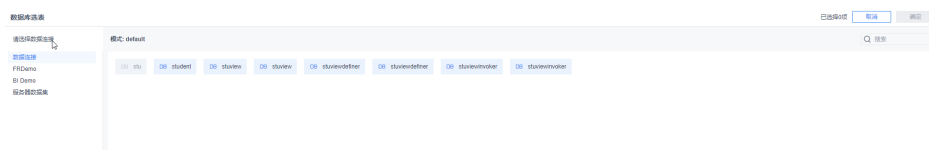


图 5-20 选择数据库

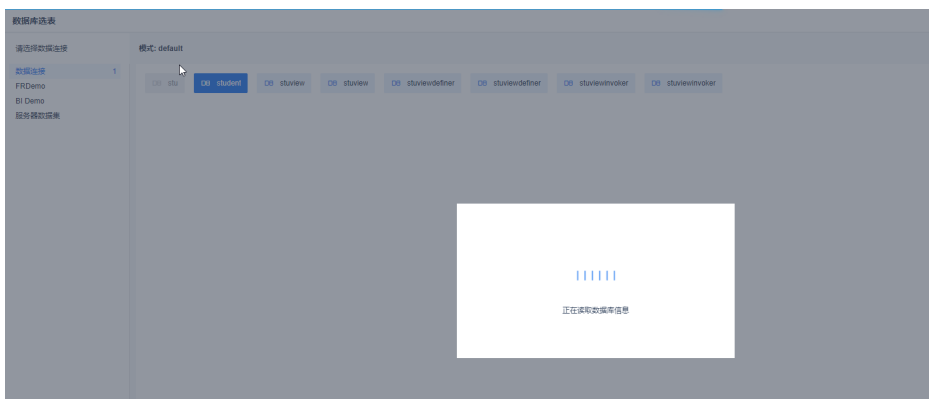
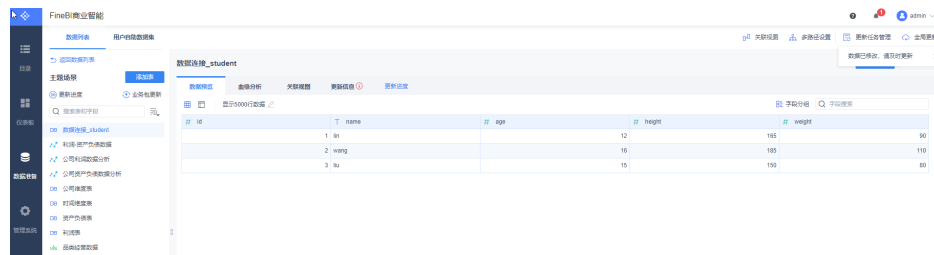


图 5-21 数据预览



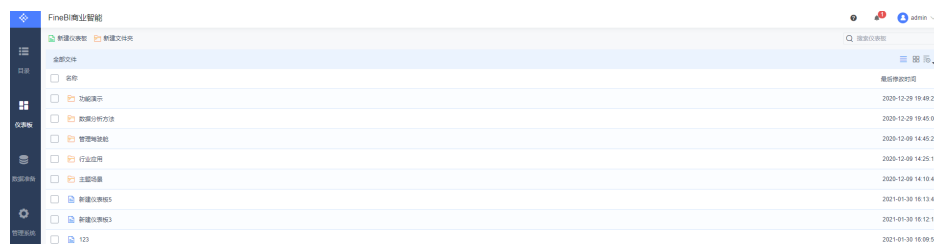
步骤6 单击“更新信息”中的“单表更新”，进行数据同步。

图 5-22 数据同步



步骤7 单击“仪表板”，单击“新建仪表板”，输入相关名称单击“确定”。

图 5-23 新建仪表板



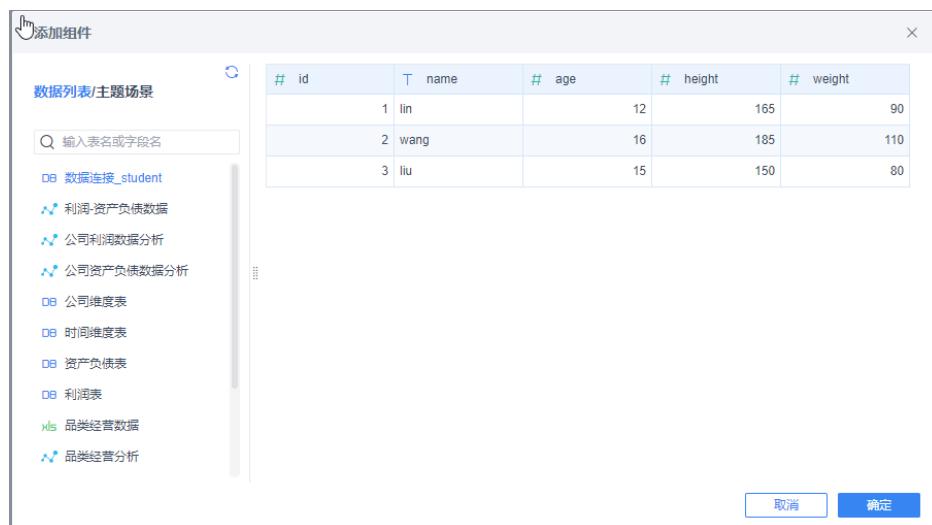
步骤8 单击“添加组件”。

图 5-24 添加组件



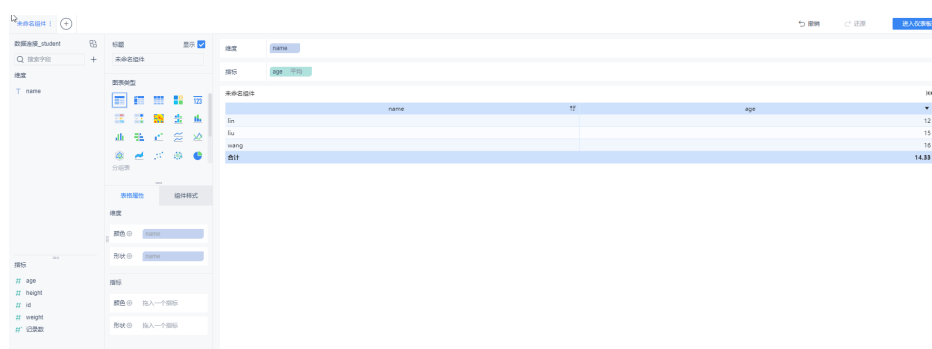
步骤9 添加步骤5配置的需要用于分析的数据表。

图 5-25 添加数据表



步骤10 将“name”拖入“维度”，将“age”拖入“指标”，即可分析年龄的平均值。如图 5-26所示。

图 5-26 分析表



步骤11 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 5-27 选择图表类型



----结束

5.4 使用 Tableau 访问 HetuEngine

本章节以Tableau Desktop 2022.2版本为例，讲解如何访问安全模式集群的HetuEngine。

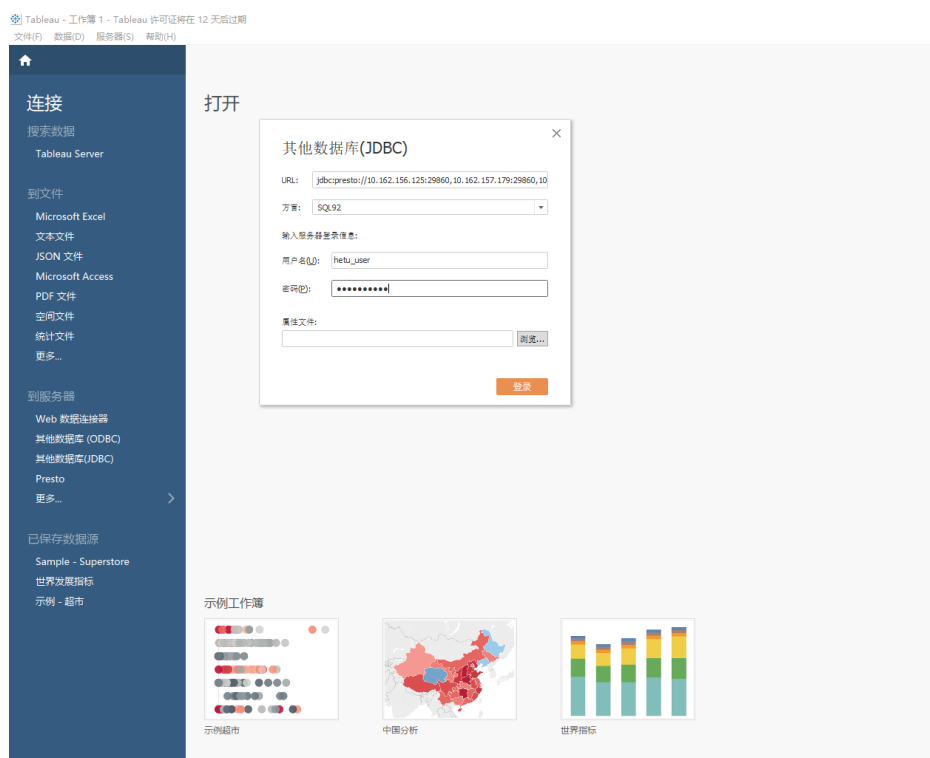
前提条件

- 已安装Tableau Desktop。
- 已获取JDBC jar包文件，获取方法请参考[步骤1](#)。
- 已在集群中创建“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](#)。
- 已创建计算实例并运行正常，可参考[创建HetuEngine计算实例](#)。

操作步骤

步骤1 将获取的Jar包放在Tableau安装目录，如“C:\Program Files\Tableau\Drivers”。

步骤2 打开Tableau，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”。支持HSFabric方式和HSBroker方式连接，URL格式详情可参考[表5-1](#)。



步骤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。

----结束

5.5 使用永洪 BI 访问 HetuEngine

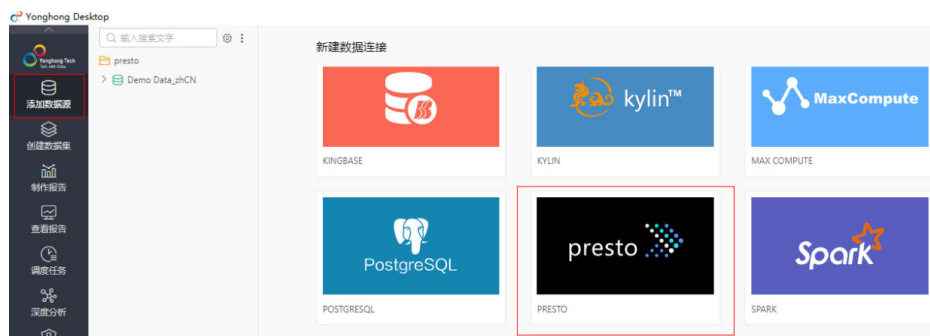
本章节以Yonghong Desktop 9.1版本为例，讲解如何访问安全模式集群的HetuEngine。

前提条件

- 已安装Yonghong Desktop。
- 已获取JDBC jar包文件，获取方法请参考[步骤1](#)。
- 已在集群中创建“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](#)。
- 已创建计算实例并运行正常，可参考[创建HetuEngine计算实例](#)。

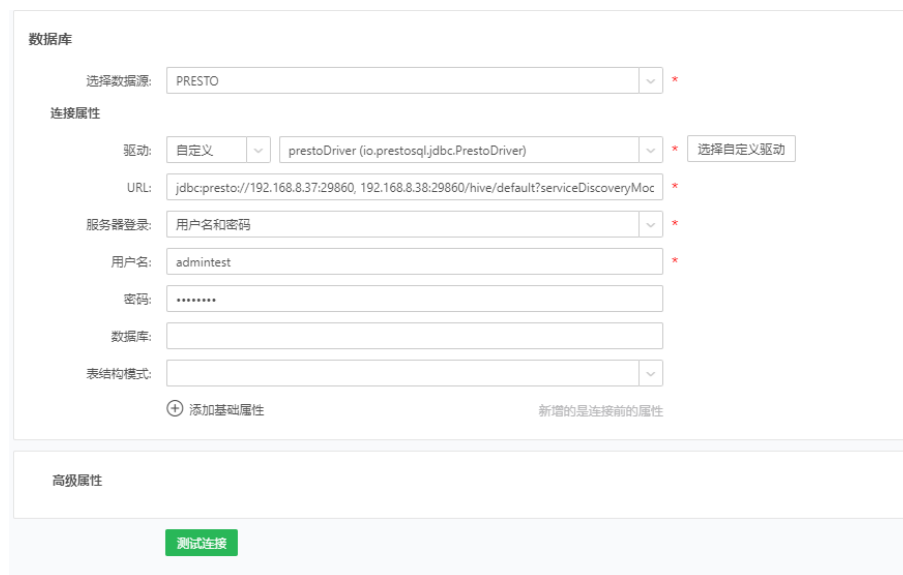
操作步骤


步骤1 打开Yonghong Desktop，选择“添加数据源 > presto”。

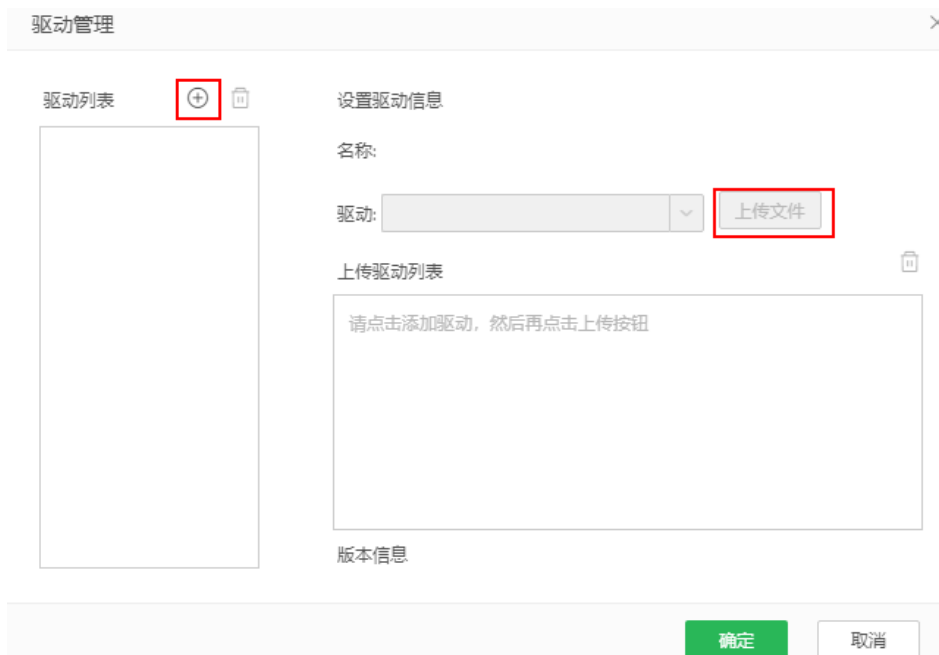


步骤2 在数据源配置页面参考[图5-28](#)完成参数配置，“用户名”和“密码”为已创建的“人机”用户的用户名和用户密码。配置完成后可以单击“测试连接”测试。

图 5-28 数据源配置

The screenshot shows the '数据库' (Database) configuration dialog. Under '选择数据源' (Select Data Source), 'PRESTO' is selected. Under '连接属性' (Connection Properties), '驱动' (Driver) is set to '自定义' (Custom) with the value 'prestoDriver (io.prestosql.jdbc.PrestoDriver)'. The 'URL' is 'jdbc:presto://192.168.8.37:29860, 192.168.8.38:29860/hive/default?serviceDiscoveryMoc'. '服务器登录' (Server Login) is set to '用户名和密码' (Username and Password). '用户名' (Username) is 'admintest' and '密码' (Password) is masked with dots. There are '添加基础属性' (Add Basic Properties) and '新增的是连接前的属性' (New properties are added before connection) options. At the bottom, there is a '高级属性' (Advanced Properties) section and a green '测试连接' (Test Connection) button.

- 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”。



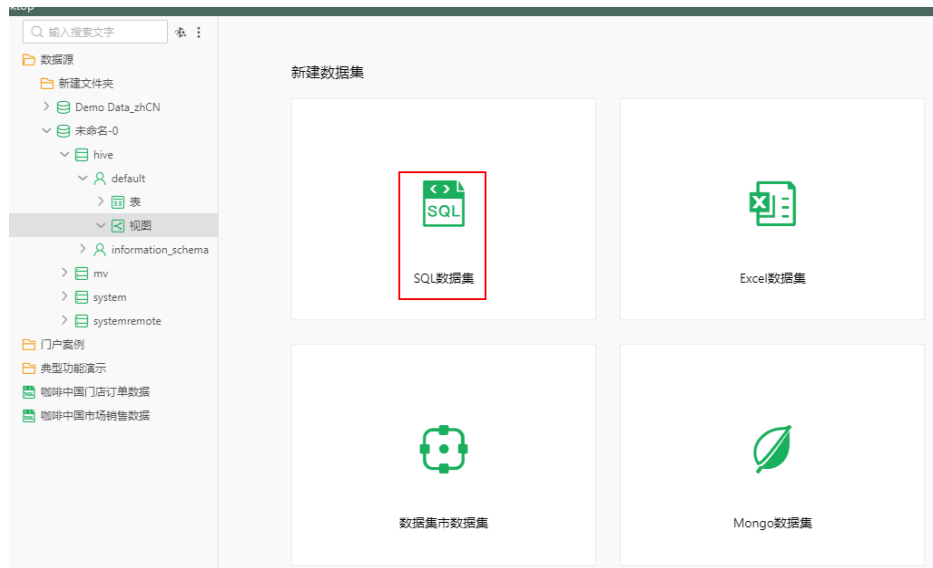
- URL：支持HSFabric方式和HSBroker方式，详情请参考[表5-1](#)。
- 服务器登录：选择“用户名和密码”并填写相应的用户名及密码。

步骤3 单击“新建数据集”，在弹出的页面参考[图5-29](#)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。

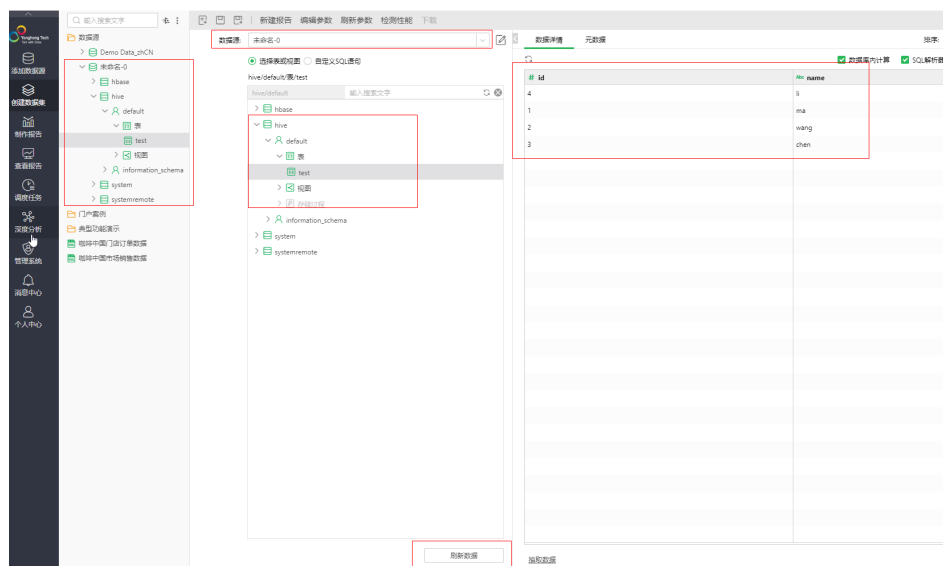
图 5-29 修改路径及名称



步骤4 在数据源选择**步骤3**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”。



步骤5 在“数据源”处选择**步骤3**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。



----结束

5.6 Hive 对接外置自建关系型数据库

📖 说明

- 本章节适用于MRS 3.x及后续版本。
- 自建关系型数据库：本章节以对接开源MySQL和Postgres数据库进行说明。
- 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。
- 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。

Hive支持开源MySQL和Postgres元数据库。

步骤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。

步骤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3 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;
```

步骤4 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5 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
javax.jdo.option.ConnectionDriverName	org.postgresql.Driver	Metastore上连接元数据的驱动类。 <ul style="list-style-type: none"> 外置MySQL，则值为：<code>com.mysql.jdbc.Driver</code> 外置Postgres，则值为：<code>org.postgresql.Driver</code>

参数名	默认值	描述
javax.jdo.option.ConnectionURL	jdbc:postgresql://% {DBSERVICE_FLOAT_IP}% {DBServer}:% {DBSERVICE_CPORT}/ hivemeta? socketTimeout=60	Metastore元数据JDBC连接的URL。 <ul style="list-style-type: none"> 外置MySQL，则值为： jdbc:mysql://MySQL的IP.MySQL的端口/test? characterEncoding=utf-8 外置Postgres，则值为： jdbc:postgresql://Postgres的IP.Postgres的端口号/test <p>说明 “test”为步骤3中在MySQL或PgSQL中创建的数据库名称。</p>
javax.jdo.option.ConnectionUserName	hive\${SERVICE_INDEX}\${SERVICE_INDEX}	Metastore上连接外置元数据数据库的用户名。

步骤6 在MetaStore中修改Postgres数据库密码，选择“集群 > 待操作集群名 > 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上连接外置元数据数据库的用户密码。密码后台会加密。

步骤7 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，直接执行步骤8。
- 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
```

步骤8 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。

步骤9 Hive重启完成后，登录MySQL或Postgres数据库，可以查看到步骤3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10 验证Hive元数据库是否外置成功：

1. 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）
2. 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 | f |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
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 | | | f |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
(1 row)
```

----结束

5.7 Hive 对接外部 LDAP

本操作适用于MRS 3.1.0及之后版本。

步骤1 登录Manager。

步骤2 在Manager界面，选择“集群 > 待操作集群的名称 > 服务 > Hive > 配置 > 全部配置 > HiveServer（角色） > 安全”。

基础配置

全部配置

HiveServer (角色)

Atlas

HDFS客户端

自定义

DLCatalog

HDFSClient

高可用性

客户端

Hook

JVM

日志

物化视图

MetaDB

MetaStore客户端

性能

Ranger

安全

步骤3 配置如下参数的值。

表 5-6 参数配置

参数名称	参数描述	备注
hive.server2.authentication	hive server认证方式	【取值范围】KERBEROS或LDAP 【默认值】KERBEROS
hive.server2.authentication.ldap.baseDN	LDAP basedn	-
hive.server2.authentication.ldap.password	LDAP密码	健康检查所使用的LDAP密码
hive.server2.authentication.ldap.url.ip	LDAP IP	-
hive.server2.authentication.ldap.url.port	LDAP端口	【默认值】389
hive.server2.authentication.ldap.userDNPattern	LDAP userDNPattern	若该参数值有多个，使用“.”分隔。例如：cn=%s,ou=People1,dc=huawei,dc=com: cn=%s,ou=People2,dc=huawei,dc=com
hive.server2.authentication.ldap.username	LDAP用户名	健康检查所使用的LDAP用户名

步骤4 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。

步骤5 选择“集群 > 待操作集群的名称 > 服务 > Hive > 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。



---结束

5.8 使用 Kafka Eagle 对接 MRS Kafka

Kafka Eagle 简介

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。

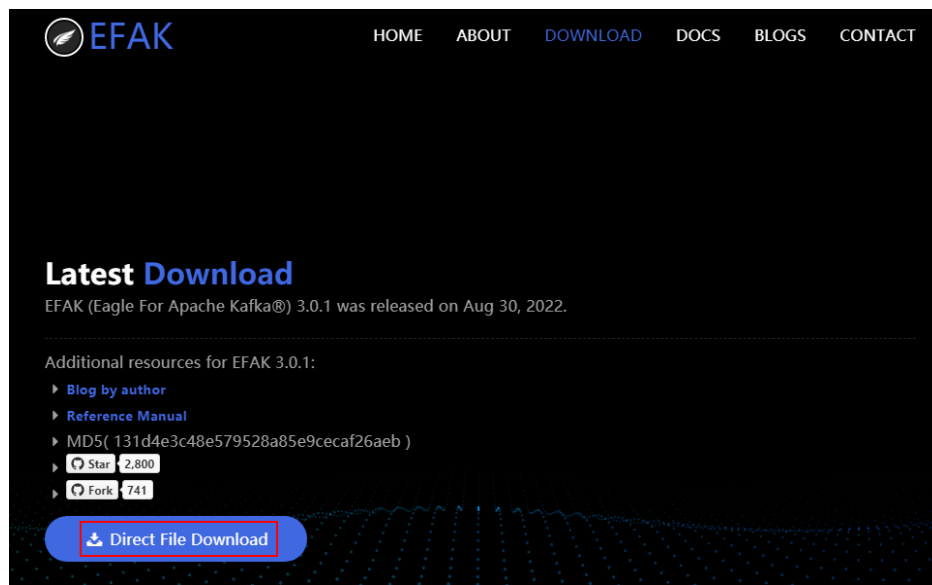
Eagle新版本中改名为EFAK。

前提条件

- 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](#)。

Kafka Eagle 安装步骤

步骤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。

步骤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 5-30 修改 Kafka 参数



步骤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。

步骤4 单击“概览”页签，选择右上方“更多 > 重启服务”重启Kafka服务。

步骤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
```

步骤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/
```

步骤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
```

步骤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
kafka.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://%/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9 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>
*****
    
```

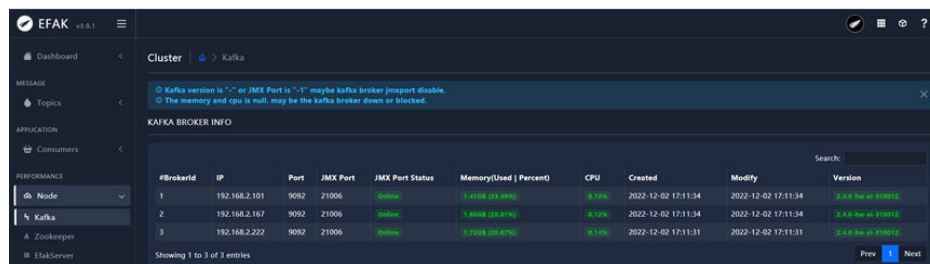
步骤10 使用获取到的登录地址，访问EFAK WebUI界面。

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 5-31 Kafka 集群监控



5.9 使用 Jupyter Notebook 对接 MRS Spark

Jupyter 对接方案概述

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。

具体流程如下：

1. [步骤1：在MRS集群外节点安装客户端](#)
2. [步骤2：安装Python3](#)
3. [安装Jupyter Notebook](#)
4. [验证Jupyter Notebook访问MRS](#)

说明

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。

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 5-34 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/f0f69f122150e0e98b5d95987a7ef5add3f8a348c6eb70d5871f855ca84e/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 information.
>>> 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/e2/09cacecafbab071c787019f00ad84ca3185952f6bb9bc9559ed83870d4d/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/dec1cc18c51b8df33c7c4d0a321b084cf38e1733b98f9d1581888fb4970/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/b2/6c7f545bb7a38754d63048c7969804a0d947328125d81bf12beaa692c3ae3/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/d5/0163eb0cfa0b673aa4fe1cd3ea9d8a81ea0f32e50807b0c295871e4aab2e/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

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 widgetsnbextension-3.5.2 zipp-3.6.0
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

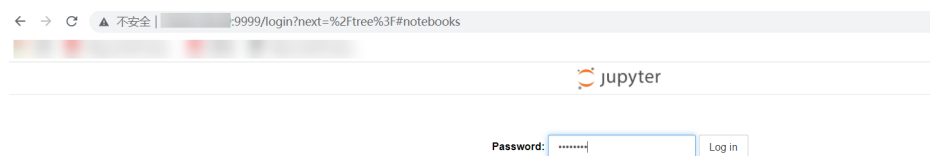
验证 Jupyter Notebook 访问 MRS

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。

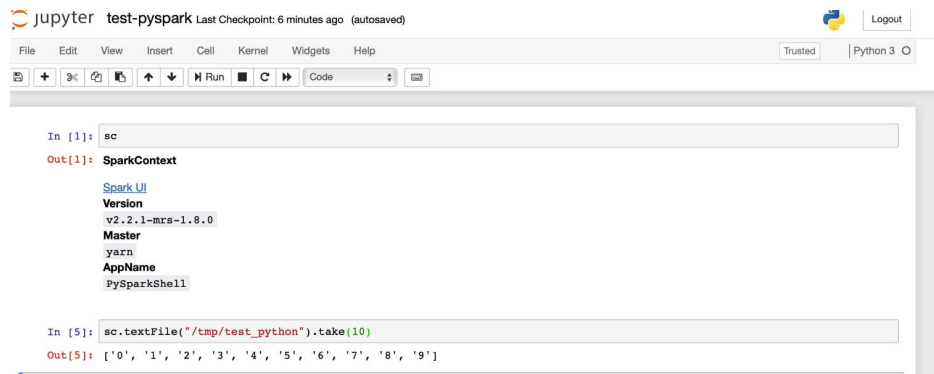
图 5-35 登录 Jupyter WebUI



步骤3 创建代码。

创建一个新的python3任务，使用Spark读取文件。

图 5-36 创建 Python 任务



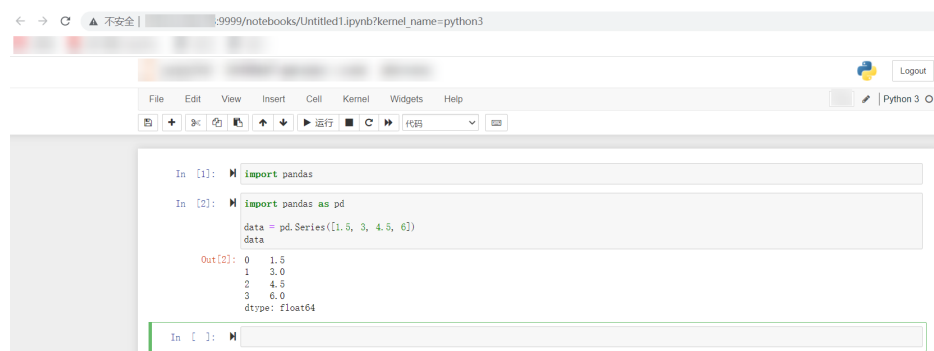
登录到集群Manager界面，在Yarn的WebUI页面上查看提交的pyspark应用。

图 5-37 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 5-38 验证 pandas



----结束

对接 Jupyter 常见问题

pandas本地import使用时，报错如下：

```
>>> import pandas
/usr/local/python3/lib/python3.7/site-packages/pandas/compat/_init_.py:85: UserWarning: Could not import the lzma module. Your installed Python is incomplete. Attempting to use lzma compression wi
ll result in a RuntimeError.
warnings.warn(msg)
/usr/local/python3/lib/python3.7/site-packages/pandas/compat/_init_.py:85: UserWarning: Could not import the lzma module. Your installed Python is incomplete. Attempting to use lzma compression wi
ll result in a RuntimeError.
warnings.warn(msg)
>>>
```

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：

```
[root@master ~]# python -m pip install backports.lzma
Looking in indexes: http://mirrors.aliyun.com/pypi/simple/
Requirement already satisfied: backports.lzma in /usr/local/python3/lib/python3.7/site-packages (0.0.14)
You are using pip version 10.0.1, however version 19.3.1 is available.
You should consider upgrading via the 'pip install --upgrade pip' command.
```

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 配置 MRS 集群阈值类告警

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](#)。

须知

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 6-1 查看阈值告警信息



其中：

- 开关：开启表示将触发告警。
- 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
 - 单击“添加规则”，参考表6-1新增指标的监控行为。
 - 单击“确定”保存规则。
 - 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。
 - 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
 - 单击待修改规则“操作”列的“修改”。
 - 参考表6-1修改相关规则参数。
 - 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： <ul style="list-style-type: none"> ● 紧急 ● 重要 ● 次要 ● 提示 	重要

参数名	参数解释	取值样例
阈值类型	选择某指标的最大值或最小值。 <ul style="list-style-type: none"> 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。 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。 	最大值
日期	设置规则生效的日期，即哪一天运行规则。 <ul style="list-style-type: none"> 每天 每周 其他 	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 6-2 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。

表 6-2 节点信息监控指标转告警列表

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%

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%
	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

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%

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%
	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，可能会导致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，可能会导致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

MRS自定义类型集群可以通过增加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。

说明

当前仅MRS自定义类型集群支持增加Task节点。

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。基本内容如下所示:

1. [添加Task节点](#)
2. [添加资源池](#)
3. [添加租户](#)
4. [队列配置](#)
5. [配置资源分布策略](#)
6. [创建用户](#)
7. [使用spark-submit提交任务](#)
8. [删除Task节点](#)

添加 Task 节点

1. 在MRS自定义类型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”。

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[添加Task节点](#)添加的节点。

步骤5 单击“确定”保存。

----结束

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。
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	计算资源选择“Yarn”时，“配置模式”可选“基础”或“高级”。 <ul style="list-style-type: none">• 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。• 高级：需配置如下参数。<ul style="list-style-type: none">- 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。- 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。- 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。- 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。
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

参数名	描述
存储空间配额	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。 说明 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 $500/2=250$ MB大小的文件。
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。
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


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**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”。

----结束

配置资源分布策略

步骤1 在集群详情页，单击“租户管理”。

步骤2 单击“资源分布策略”页签，资源池选择**添加资源池**创建的资源池。

步骤3 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4 单击“确定”。

----结束

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

使用 spark-submit 提交任务

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）。

```
cd Spark2x/spark/bin
```

```
sh spark-submit --queue tenant_spark --class  
org.apache.spark.examples.SparkPi --master yarn-client ../examples/jars/  
spark-examples_*.jar
```

删除 Task 节点

1. 在集群详情页面，单击“节点管理”页签。
2. 在Task节点组的“操作”列，选择“缩容”。
3. 缩容类型选择“指定节点缩容”，勾选需要缩容的节点。

📖 说明

缩容指定的节点需要该节点处理关机状态。

4. 勾选“我已经阅读上述信息，并了解具体影响。”，并单击“确定”。

6.3 配置 MRS 集群弹性伸缩

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。

操作场景

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。

视频介绍

配置MRS集群弹性伸缩操作案例可参考[配置MRS集群弹性伸缩操作视频](#)，该视频以MRS 3.1.0版本集群为例，介绍如何在购买集群时配置弹性伸缩、如何为已有集群新增弹性伸缩策略。

说明

因不同版本操作界面可能存在差异，相关视频供参考，具体以实际环境为准。

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 6-3 添加 Task 节点组

角色	部署倾向	数量限制	角色类型	共部署角色	多实例最大值	操作限制
TagSync	所有节点组都可部署	0-1	控制角色	-	--	-
Hadoop		Flink		Ranger		
DN	NM	FR	FS	TSC		
<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		

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。

配置Task节点

×

Task节点是处理数据的实例，不存储集群数据（如HDFS数据）。

节点类型

节点规格

节点数量

系统盘

添加数据盘

数据盘

数据盘数量

内存优化型

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”按钮，并配置“节点数量范围”为“2 - 4”。

图 6-4 配置弹性伸缩



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 6-5 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 6-6 添加规则

添加规则

规则名称

如果

持续 个五分钟

添加 个节点

冷却时间 分钟

“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。