

数据湖计算 DLC







【版权声明】

©2013-2025 腾讯云版权所有

本文档(含所有文字、数据、图片等内容)完整的著作权归腾讯云计算(北京)有限责任公司单独所有,未经腾讯云 事先明确书面许可,任何主体不得以任何形式复制、修改、使用、抄袭、传播本文档全部或部分内容。前述行为构成 对腾讯云著作权的侵犯,腾讯云将依法采取措施追究法律责任。

【商标声明】

🔗 腾讯云

及其它腾讯云服务相关的商标均为腾讯云计算(北京)有限责任公司及其关联公司所有。本文档涉及的第三方主体的 商标,依法由权利人所有。未经腾讯云及有关权利人书面许可,任何主体不得以任何方式对前述商标进行使用、复 制、修改、传播、抄录等行为,否则将构成对腾讯云及有关权利人商标权的侵犯,腾讯云将依法采取措施追究法律责 任。

【服务声明】

本文档意在向您介绍腾讯云全部或部分产品、服务的当时的相关概况,部分产品、服务的内容可能不时有所调整。 您所购买的腾讯云产品、服务的种类、服务标准等应由您与腾讯云之间的商业合同约定,除非双方另有约定,否则, 腾讯云对本文档内容不做任何明示或默示的承诺或保证。

【联系我们】

我们致力于为您提供个性化的售前购买咨询服务,及相应的技术售后服务,任何问题请联系 4009100100或 95716。



文档目录

开发指南

使用 SDK 数据写入 SparkJar 作业开发指南 PySpark 作业开发指南 查询性能优化指南 UDF 函数开发指南 物化视图 系统约束 元数据信息

计算任务



开发指南 使用 SDK 数据写入

最近更新时间: 2024-11-28 10:44:42

概述

数据湖计算 DLC 提供用户通过 Java SDK 数据导入。

应用场景

通过 Java sdk 将源数据库的增量变更同步到 DLC 原生表,完成源数据入湖。 **适用场景:**

1. 需要实时处理数据流的业务场景。

2. 熟悉 Java 并需要自定义逻辑处理的开发人员。

前置条件

- 正确开通 DLC,已完成用户权限配置,开通托管存储。
- 正确创建 DLC 数据库。
- 正确配置 DLC 数据库数据优化,详细配置请参见 开启数据优化。

操作步骤

步骤1:下载依赖 Jar

请手动下载依赖 SDK jar 包 dlc-bridage.jar。

步骤2:编写 Java 代码

```
package org.example;
// 建表 create table ingest_stream(id int,name string);
import com.gotocompany.depot.dlc.models.SinkRecord;
import com.tencent.dlc.DlcClient;
import com.tencent.dlc.RealtimeStream;
import com.tencent.dlc.RowStream;
public class RealtimeStreamDemo {
    public static void main(String[] args) throws Exception {
        String endpoint = "dlc.tencentcloudapi.com";
```



```
.endpoint (endpoint)
        .secretKey(secretKey)
        .region(region)
        .operate(RowStream.RealTimeOperate.APPEND_ONLY)
// 只有调用 flush 之后数据才可见
((RealtimeStream) stream).flush();
// 必须调用 stream close接口, close 时会隐含执行 flush
client.close();
```

<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd"> <modelVersion>4.0.0</modelVersion>

<properties>

<project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>



</properties>

```
<artifactId>dlc-bridage</artifactId>
  <groupId>org.apache.maven.plugins</groupId>
 <artifactId>maven-compiler-plugin</artifactId>
```

步骤3: DLC 新建目标表

新建目标表详情可参见 DLC 原生表操作配置。

步骤4:执行 SDK 程序

登录 DLC 控制台,单击数据探索,查询目标表数据。



数据探索 🖏 广州 👻			SOL语法参考 II 数据探索使用
库表 査询 🗘・	+ x	ql 🔸 👻	1
数据目录 DataLakeCatalog	▼ ● 运行	C 刷新	
	33 select * from kafka_dl	c	
请输入表名称C	2		
🖩 kafka_dic	•		
🖩 kafka_dlc	X 查询结果 统计数据		运行历史 下载历史
数据结构 分区信息	Task ID SQL详情 导出结果 优化建议	(B	
字段 类型	查询耗时 14.63s 数据扫描量 1.4 MB	。 2.教促) 信制教授 局	Si
id int			
name string	Id	name	age
age int	2	Lisi	19
	10	Qi	27
	6	Hu	23
	3	W	20

SparkJar 作业开发指南

最近更新时间: 2024-07-03 14:15:21

应用场景

DLC 完全兼容开源 Apache Spark,支持用户编写业务程序在 DLC 平台上对数据进行读写和分析。本示例演示 通过编写 Java 代码在 COS 上读写数据和在 DLC 上建库表、读写表的详细操作,帮助用户在 DLC 上完成作业开 发。

环境准备

依赖: JDK1.8 Maven IntelliJ IDEA

开发流程

开发流程图

DLC Spark JAR 作业开发流程图如下:



创建资源

第一次在 DLC 上运行作业,需新建 Spark 作业计算资源,例如新建名称为 "dlc−demo" 的 Spark 作业资源。 1. 登录 数据湖计算 DLC 控制台,选择服务所在区域,在导航菜单中单击**数据引擎。**

- 2. 单击左上角创建资源,进入资源配置购买页面。
- 3. 在集群配置 > 计算引擎类型项选择 Spark 作业引擎。



基础配置											
计费模式	按量计费	包年包月	详细对比 🖸								
	按量计费适合周期性任务场易	8, 不使用时可以挂起资源;	<u>着</u> 免存在费用消耗造成浪费。	包年包月适合有稳定、持续	妙理的任务需求的场景,无	需等待资源拉起。					
地域				华东地区			地区	美国西部	亚太东南		
	7Ľ			上海	上海金融	成都	重庆	硅谷	新加坡	弗吉尼亚	
	BONHE.										
	法兰克福										
	法兰克福处于不同地域的云产品内网石	5互通,购买后不能更换,;	青您谨慎选择。 建议选择最素	5近您客户的地域,可降低访	问时延。						
集群配置	法兰克福处于不同地域的云产品内网络	下互通,购买后不能更换,计	请您谨慎选择, 建议选择最素	5년忽客户的地域,可降低访	问时过程。						
	法兰克福	「互通,购买后不能更换,计	<mark>请您谨慎选择,</mark> 建议选择最靠	850客户的地域,可降低的	10997 <u>95</u> .						
集群配置 基本配置 计算引等类型	法兰克福 处于不同地域的云产品内网/	Spark作业	<mark>请你准慎定保,建议进择最亲</mark>	起的客户的地域,可降低访	1000182.						
集群配置 基本配置 计算引擎类型	法兰男福 处于不同地域的云产品内闲/ SparkSQL 不可可愿于地不可数语分词	「互通」、购买店不能更换, i Spark/F业 R号。 Spark/SOLIS用于発売	<mark>集改建慎选择</mark> ,建议违择最素 Presto 率较的高俅SOL代码、Soari	1958春户的地域,可降低的 1959年一的地域,可降低的	问时轻。	1 8分析。					
集群配置 基本配置 计算引等关型	法兰克福 处于不同地域的云产品内风/ SparkSQL 不同引擎文持不同数据处理	「日通」、阿天后不能更換、 Spark/F业 5景、Spark/CL道用于色引	まで連慎正導、違い法導量素 Presto E電気的電磁SOLI任务、Spark	近悠喜产的地域,可降低访 1990年1993年 1991年1993年 1991年1993年 1991年1993年 1991年1993年 1991 1991 1995 1991	问时经。 过是,Presto选 于交互式重	E9967.					
集群配置 基本配置 计算引率关型	法兰克福 处于不同地域边云产品内保A SparkSQL 不同引擎支持不同数量处理4	5日連、购买店不能更換、計 Spark作业 5号、SparkのLIE用于特計	<mark>東切達慎選择、建议法得最素</mark> Presio 単成の1年後の1代名、Span	紀念廣中的地域,可降低功 1990年9月 1992年于中京、地市任務日	19时起。 地理,Presboll,于文互文重	eroff.					
集群配置 基本配置 计算引辱兴型	法兰英语 处于不同地域的云产品内保A SparkSQL 不同时 原文师不同的语义语 Spark 3.2	 「互通・购买店不能更換」。) Spark作业 Spark 生生 Spark 2.4 	<mark>東部建筑改祥、建议法将最素</mark> Presto 尾花が高峡SOL任務、Span	紀念廣中的地域。可降低的 地下型道用于所式,和式任務公	illentiž。 10년, Prestuži 子交互式重	18分析。					
集群配置 基本配置 计算引导关型 内核版本	法兰英福 处于不同地域的云产品内保A 不同引 原文持不同数据处理 Spark 3.2 取作业基于Spark版本能力。	 Spark代型 Spark代型 Spark(1) Spark 2.4 不同版本対点成時色不同。 	<mark>東部建筑改祥、建议选择最素</mark> Presto 足高功的高峡SOL任务、Spari 洋磁波教列表可参见: Spar i	紀念慶中的地域。可算低的 4作业道用于低力能力任务及 4环境路時 2	i向时起。 此思,Prestuži,于文互式重	EQ:9 FF.					
集群配置 基本配置 计算引攀类型 内核版本	法兰英福 处于不能地域的云产品内保/ SparkSQL 不同引 販支持不同数据改造部 Spark 3.2 取け上語子 Spark版本能力。 和売	Sparkf2业 Sparkf2业 Sparkf2业 Sparkf2.4 不限板本対应该動動不同。	<mark>書物遺情送祥、</mark> 建议送得最素 Presto E電気効高減SOL任务、Spark 详细成款列表可参见: Spark	2008年中的地域,可算低的 4月2016年于俄式地域任务的 45月16月18日	i同时延。 社通、Prestola 于文王式重	E0917.					
集群配置 基本配置 计算引导类型 内核版本	法兰英语 处于不能地的公产品内保/ 及中不能地的公产品内保/ 不同引 第支持不同数据处理和 不同引 第支持不同数据处理和 及作业基于Spark原本能力。 标在型	5月 , 約天后不能更迭 , 計 5月 , 5 9art /行业 5日 , 59art /日本 5 9art /1111111111111	高等連携結構、建议法保養素 Presto 国際放動業績SOL任务、Spari 洋畑成動列表可参切: Spari 西美いなエスス、 1980 年の1980		i的时起。 七道、Prestolal 于文互式重	5829 M.					

信息配置 > 资源名称 项填写 "dlc-demo"。新建资源详细介绍请参见 购买独享数据引擎。

§ 级配置 ^影 数配置	+ 燕加
集群网段	10.255.0.016 博文 该配置会影响服务之间的网络互通性。在非联邦面询场展下,推荐使用数以配置;(
信息配置	
资源名称	dic-demo
描述信息	不超过250个字符
	造填,长度限制为250
标签	智无标签 ≠
	标签用于从不同维度对资源分类管理,如需了解更多,请前住 标签产品文档 [2

- 4. 单击**立即开通,**确认资源配置信息。
- 5. 确认信息无误后,单击**提交,**完成资源配置。

上传数据到 COS



创建名称为 "dlc-demo"的存储桶,上传 people.json 文件,用作从 COS 读写数据的示例,people.json 文件的内容如下:

- {"name":"Michael"}
 {"name":"Andy", "age":30}
 {"name":"Justin", "age":3}
 {"name":"WangHua", "age":19}
 {"name":"ZhangSan", "age":10}
 {"name":"LiSi", "age":33}
 {"name":"ZhaoWu", "age":37}
 {"name":"MengXiao", "age":68}
 {"name":"KaiDa", "age":89}
- 1. 登录 对象存储 COS 控制台,在左侧菜单导航中单击 存储桶列表。
- 2. 创建存储桶:单击左上角创建存储桶,名称项填写 "dlc-dmo",单击下一步完成配置。
- 上传文件:单击**文件列表 > 上传文件**,选择本地 "people.json"文件上传到 "dlc-demo-1305424723" 桶里(-1305424723是建桶时平台生成的随机串),单击上传,完成文件上传。新建存储桶 详情可参见创建存储桶。

← 返回桶列表	dlc-demo-1305424723 /							文档指引 12
接索菜単名称 Q		件夹 文件碎片 清空存储桶	更多操作 ▼				在线编辑器 团	
報題 文件列表 1	前缀搜索 ▼ 只支持担	<u>素当前虚拟目录下的对象</u>	❷ Q 刷新 共0个文件		-	每页 100 个对象	K < 1 → Ⅲ Ξ	
基础配置	文件名 \$	上传文件			× ^{RBI}	间 \$	操作	
安全管理		🕢 选择上传对象 👌 🚺	设置对象属性					
权限管理		3			- 1			
域名与传输管理 🗸 🗸		选择文件 上传至 dlc-demo-13	305424723/					
容措容灾管理		若上传路径中存在同名文件,上传将覆盖	皇原有文件。					
日志管理		上传播作将产生请求次数和上行流量,其	§中请求 ☑ 按万次收费,上行流量 ☑ 免费。详情潇	查看计赛项 🖸 及产品定价 🗹				
内容审核 1807 >		文件/文件夹	大小	操作				
数据处理		people.json	241.00B	删除				
<u>教</u> 建监控								
ELWIT异								
0100 1100 000								
			参数配置 <u>上传</u>					9

新建 Maven 项目

- 1. 通过 IntelliJ IDEA 新建一个名称为 "demo" 的 Maven 项目。
- 2. 添加依赖:在 pom.xml 中添加如下依赖:



<groupid>org.apache.spark</groupid>
<artifactid>spark-core_2.12</artifactid>
<version>3.2.1</version>
<scope>provided</scope>
<groupid>org.apache.spark</groupid>
<artifactid>spark-sql_2.12</artifactid>
<version>3.2.1</version>
<scope>provided</scope>

编写代码

编写代码功能为从 COS 上读写数据和在 DLC 上建库、建表、查询数据和写入数据。

1. 从 COS 上读写数据代码示例:

```
package com.tencent.dlc;
import org.apache.spark.sql.Dataset;
import org.apache.spark.sql.Row;
import org.apache.spark.sql.SaveMode;
import org.apache.spark.sql.SparkSession;
public class CosService {
    public static void main( String[] args )
    {
        //1.dl建SparkSession
        SparkSession spark = SparkSession
        .builder()
        .appName("Operate data on cos")
        .config("spark.some.config.option", "some-value")
        .getOrCreate();
        //2.i硬Ros_Lbijson文件生成数据集,支持多种类型的文件, 如
        json,csv,parquet,orc,text
        String readPath = "cosn://dlc-demo-1305424723/people.json";
        Dataset<Row> readData = spark.read().json(readPath);
        //3.对数据集做业务计算操作生成结果数据,计算支持API和sol形式,这里生成临
时表用sqli读数据
```



<pre>readData.createOrReplaceTempView("people");</pre>
Dataset <row> result = spark.sql("SELECT * FROM people where</row>
age > 3");
//4. 结果数据保存到 cos
String writePath = "cosn://dlc-demo-
1305424723/people_output";
// 写入支持多种类型的文件,如 json,csv,parquet,orc,text
<pre>result.write().mode(SaveMode.Append).json(writePath);</pre>
<pre>spark.read().json(writePath).show();</pre>
//5. 关闭 session
<pre>spark.stop();</pre>

2. DLC 上建库、建表、查询数据和写入数据:

```
public static void main(String[] args) {
    SparkSession spark = SparkSession
    //2.建数据库
    String dbSql = "CREATE DATABASE IF NOT EXISTS" + dbName + "
    spark.sql(dbSql);
    //3.建内表
    String tableSql = "CREATE TABLE IF NOT EXISTS " + dbName + "."
    spark.sql(tableSql);
    //4.写数据
    spark.sql("INSERT INTO " + dbName + "." + tableName + "VALUES
```



//5. 查询数据
<pre>spark.sql(" SELECT * FROM " + dbName + "." +</pre>
<pre>tableName).show();</pre>
//6. 建外表
<pre>String extTableName = "`ext_test`";</pre>
spark.sql(
"CREATE EXTERNAL TABLE IF NOT EXISTS " + dbName + "."
+ extTableName + ""
+ " (`id` int, `name` string, `age` int) "
+ "ROW FORMAT SERDE
'org.apache.hive.hcatalog.data.JsonSerDe' "
+ "STORED AS TEXTFILE LOCATION 'cosn://dlc-
demo-1305424723/ext_test '");
//7. 写外表数据
<pre>spark.sql("INSERT INTO " + dbName + "." + extTableName +</pre>
"VALUES (1, 'LiLy', 12), (2, 'Lucy', 3) ");
//8. 查询外表数据
<pre>spark.sql(" SELECT * FROM " + dbName + "." +</pre>
<pre>extTableName).show();</pre>
//9. 关闭 Session
<pre>spark.stop();</pre>

建外表时,需按照 上传数据到COS的步骤 先在桶里建对应表名文件夹保存表文件

dlc-demo-1305424723 /				文档指引 [
1 上传文件 创建文件夹 文件碎片	清空存储桶 更多操作 ▼			在线编辑器 团
前缀搜索 ▼ 只支持搜索当前虚拟目录下的对象	3 Q 刷新	共1个文件	每页 100 个对象	₩ ◀ 1 ▶ 📰 📃
文件名 \$	大小 \$	存储类型 👅	修改时间 🛊	操作
ext_test/ 🖍		-		设置权限 统计 更多 ▼
				c



调试、编译代码并打成 JAR 包

通过 IntelliJ IDEA 对 demo 项目编译打包,在项目 target 文件夹下生成JAR包 demo-1.0-SNAPSHOT.jar。

上传 JAR 包到 COS

登录 COS 控制台,参考 上传数据到 COS 的步骤将 demo-1.0-SNAPSHOT.jar 上传到 COS。

新建 Spark Jar 数据作业

创建数据作业前,您需先完成数据访问策略配置,保证数据作业能安全地访问到数据。配置数据访问策略详情请参见 配置数据访问策略。如已配置数据策略名称为:qcs::cam::uin/100018379117:roleName/dlc-demo。

- 1. 登录 数据湖计算 DLC 控制台,选择服务所在区域,在导航菜单中单击数据作业。
- 2. 单击左上角创建作业,进入创建页面。
- 3. 在作业配置页面,配置作业运行参数,具体说明如下:

配置参数	说明
作业名称	自定义 Spark Jar 作业名称,例如:cosn-demo
作业类型	选择 批处理类型
数据引擎	选择 创建资源 步骤创建的 dlc-demo 计算引擎
程序包	选择 COS,在 上传 JAR 包到 COS 步骤上传的 JAR 包 demo−1.0− SNAPSHOT.jar
主类(Main Class)	根据程序代码填写,如: • 从 COS 上读写数据填: com.tencent.dlc.CosService • 在 DLC 上建库、建表等填: com.tencent.dlc.DbService
数据访问策略	选择该步骤前创建的策略 qcs::cam::uin/100018379117:roleName/dlc- demo

其他参数值保持默认。

🔗 腾讯云

编辑作业		
基本信息 ▲		
作业名称 *	cosn_demo	
	支持中文、英文、数字与"_",最多100个字符	
作业类型 *	批处理 流处理 SQL作业	
数据引擎 *	▼ (按量计费)	
	└ 计费以所选数据引擎计费模式为准,可至数据引擎 ☑ 查看管理。数据引擎的网	┘ 络配置信息可至 <mark>网络配置 </mark>
程序包 *	○ 对象存储COS ○ 本地上传	
程序包 *	 ● 对象存储COS 本地上传 cosn:// -1305424723/demo-1.0-SNAPSHOT.jar 	选择COS位置
程序包 *	 ● 对象存储COS 本地上传 cosn:// -1305424723/demo-1.0-SNAPSHOT.jar 需具备COS相关权限,可选择jar/py文件 	选择COS位置
程序包 * 主类(Main Class) *	 ▶ 对象存储COS 本地上传 cosn:// -1305424723/demo-1.0-SNAPSHOT.jar 需具备COS相关权限,可选择jar/py文件 com.tencent.dlc.CosService 	选择COS位置
程序包 * 主类(Main Class) * 程序入口参数	 对象存储COS 本地上传 cosn://1305424723/demo-1.0-SNAPSHOT.jar 需具备COS相关权限,可选择jar/py文件 com.tencent.dlc.CosService 请输入程序入口参数,多个参数空格分隔。不超过65536个字符 	选择COS位置
程序包 * 主类(Main Class) * 程序入口参数	 ▶ 对象存储COS ▲地上传 cosn:// -1305424723/demo-1.0-SNAPSHOT.jar 需具备COS相关权限,可选择jar/py文件 com.tencent.dlc.CosService 请输入程序入口参数,多个参数空格分隔。不超过65536个字符 	选择COS位置
程序包 * 主类(Main Class) * 程序入口参数	 ▶ 对象存储COS ▲地上传 cosn:// -1305424723/demo-1.0-SNAPSHOT.jar 需具备COS相关权限,可选择jar/py文件 com.tencent.dlc.CosService 请输入程序入口参数,多个参数空格分隔。不超过65536个字符 	选择COS位置
程序包 * 主类(Main Class) * 程序入口参数	 ▶ 对象存储COS ▲地上传 cosn:// -1305424723/demo-1.0-SNAPSHOT.jar 需具备COS相关权限,可选择jar/py文件 com.tencent.dlc.CosService 请输入程序入口参数,多个参数空格分隔。不超过65536个字符 	选择COS位置
程序包 * 主类(Main Class) * 程序入口参数 作业参数 (-config)	 ▶ 对象存储COS ▲地上传 cosn:// -1305424723/demo-1.0-SNAPSHOT.jar 需具备COS相关权限,可选择jar/py文件 com.tencent.dlc.CosService 请输入程序入口参数,多个参数空格分隔。不超过65536个字符 示例: spark.network.timeout=120s 	选择COS位置

4. 单击保存,在Spark 作业页面可以看到创建的作业。

运行并查看作业结果

1. 运行作业:在Spark 作业页面,找到新建的作业,单击运行,即可运行作业。

2. 查看作业运行结果: 可查看作业运行日志和运行结果。

查看作业运行日志

1. 单击**作业名称 > 历史任务**,查看任务运行状态。



作业。	→「卅				Spark作业详情													
Spark作业	作业配置 Session	管理			作业信息	历史任务	2 监控告警											
创建作业		Q	全部	▼ 请选择创建人	请选择执行状态	J 	近7天	近30天	2023-07-0	7 ~ 2023-07-13	i.							ļ
作业名称	作业ID	作业类型	作业文件	当前任务数	任务ID 3	执行状态	任务提交时间	\$ 计	算耗时 排	影作								
6	6	-		. 0	ed56fcd2-5		2023-07-12 1	6:04:48 2m	iin36s 1	重看详情 Spark UI								
6	6			. 0			1		đ	查看详情 Spark UI								
1 -									2	昏着详情 Spark UI								
	6			0		-19-19 -	-		2	查看详情 Spark UI								
cosn_demo Г	batch_366ae4	批处理	cosn: 1305424	1	共 4 条					10 ▼ 条/页	K 4	1	1	1 /1	1 /1页	1 /1页	1 /1页	1 /1页 ▶

2. 单击任务ID > 运行日志,查看作业运行日志:

				_				
作业名称 協制公司名属	est 向 作	ē业ID:		ū				
全利百取多展	示取近1000余	16尽						
近7天	近30天	2023-08-01 17:36:02	~ 2023-08-07 17:36:02		按时间降序 🔻	Φ	创建下载	任务
日志名称:				日志级别	: All		•	
23/08/07 1	7:32:24 INFO K	KubernetesClusterSched	ulerBackend\$Kubernetes	DriverEndp	pint: Asking eacl	h executor	to shut down	
23/08/07 17	7:32:24 INFO K	KubernetesClusterSched	ulerBackend: Shutting do	wn all execu	utors			
00/00/07 1		Dearly II. Otoppod Operiy	wah III at http://apark. Efc	44000 of E E o	Cha driver eve d	afault avau	4040	
23/08/07 13	7:32:24 INFO S	SparkUI: Stopped Spark v	web UI at http://spark-5f6	41289cf55a	6bc-driver-svc.d	efault.svc:4	4040	
23/08/07 1	7:32:24 INFO S -+	SparkUI: Stopped Spark v	web UI at http://spark-5f6	41289cf55a	6bc-driver-svc.d	efault.svc:4	4040	
23/08/07 13 +++	7:32:24 INFO S -+	SparkUI: Stopped Spark v	web UI at http://spark-5f6	41289cf55a	6bc-driver-svc.d	efault.svc:4	4040	
23/08/07 1 ++ 2 Lucy 3	7:32:24 INFO S -+ 3	SparkUI: Stopped Spark \	web UI at http://spark-5f6	41289cf55a	6bc-driver-svc.d	efault.svc:4	4040	
23/08/07 1 +++ 2 Lucy 3 1 LiLy 12	7:32:24 INFO S -+ 3 2	SparkUI: Stopped Spark v	web UI at http://spark-5f6	41289cf55a	6bc-driver-svc.d	efault.svc:4	4040	
23/08/07 1 +++ 2 Lucy 3 1 LiLy 12	7:32:24 INFO S -+ 3 2	SparkUI: Stopped Spark v	web UI at http://spark-5f6	41289cf55a	6bc-driver-svc.d	efault.svc:4	4040	
23/08/07 1 ++ 2 Lucy 3 1 LiLy 12 +++	7:32:24 INFO S -+ 3 2 -+	SparkUI: Stopped Spark v	web UI at http://spark-5f6	41289cf55a	6bc-driver-svc.d	efault.svc:4	4040	
23/08/07 1 ++ 2 Lucy 3 1 LiLy 12 +++ id name a	7:32:24 INFO S -+ 3 2 -+ ge	SparkUI: Stopped Spark v	web UI at http://spark-5f6	41289cf55a	6bc-driver-svc.d	efault.svc:4	4040	
23/08/07 1 +++ 2 Lucy 3 1 LiLy 12 +++ id name a	7:32:24 INFO S -+ 3 2 -+ ge -+	SparkUI: Stopped Spark v	web UI at http://spark-5f6	41289cf55a	6bc-driver-svc.d	efault.svc:4	4040	
23/08/07 1 +++ 2 Lucy 3 1 LiLy 12 +++ id name a +++	7:32:24 INFO S -+ 3 2 -+ ge -+	SparkUI: Stopped Spark v	web UI at http://spark-5f6	41289cf55a	6bc-driver-svc.d	efault.svc:4	4040	
23/08/07 1 ++ 2 Lucy 3 1 LiLy 12 ++ id name a +++ 23/08/07 17	7:32:24 INFO S -+ 3 2 -+ ge -+ 7:32:24 INFO D	SparkUI: Stopped Spark v	web UI at http://spark-5f6 shed: show at DbService	.java:37, too	6bc-driver-svc.d	efault.svc:4	4040	
23/08/07 1 ++ 2 Lucy 3 1 LiLy 12 ++ id name a ++ 23/08/07 17 23/08/07 17	7:32:24 INFO S -+ 3 2 -+ 7:32:24 INFO D 7:32:24 INFO T	SparkUI: Stopped Spark v DAGScheduler: Job 5 fini FaskSchedulerImpl: Killing	web UI at http://spark-5f6 shed: show at DbService g all running tasks in stag	,java:37, too e 5: Stage f	6bc-driver-svc.d ok 0.161365 s inished	efault.svc:4	4040	
23/08/07 1 ++ 2 Lucy 3 1 LiLy 12 ++ id name a ++ 23/08/07 17 23/08/07 17	7:32:24 INFO S -+ 3 2 -+ 7:32:24 INFO D 7:32:24 INFO T 7:32:24 INFO D	SparkUI: Stopped Spark v DAGScheduler: Job 5 fini askSchedulerImpI: Killing	web UI at http://spark-5f6 shed: show at DbService g all running tasks in stag inished. Cancelling poter	,java:37, too e 5: Stage f tial specula	6bc-driver-svc.d ok 0.161365 s inished tive or zombie ta	efault.svc:4	4040 s job	
23/08/07 1 ++ 2 Lucy 3 1 LiLy 12 +++ id name a +++ 23/08/07 17 23/08/07 17	7:32:24 INFO S -+ 3 2 -+ 7:32:24 INFO D 7:32:24 INFO D 7:32:24 INFO D	SparkUI: Stopped Spark v DAGScheduler: Job 5 fini ^r askSchedulerImpI: Killing DAGScheduler: Job 5 is f	web UI at http://spark-5f6 shed: show at DbService g all running tasks in stag inished. Cancelling poter	41289cf55a .java:37, too e 5: Stage f tial specula	6bc-driver-svc.d ok 0.161365 s inished tive or zombie ta	efault.svc:4	4040 s job	

查看作业运行结果

分 腾讯云

1. 运行从 COS 读写数据示例,则到 COS 控制台 查看数据写入结果。



🔶 返回桶列表		-1305424723 / people_output				文档指引 🗹
搜索菜单名称	Q	上传文件 创建文件夹 更多操作 ▼				在线编辑器 💿
196,945		前缀搜索 ▼ 只支持搜索当前虚拟目录下的对象	8 Q 🕅	新 共 2 个文件	每页 100 个对象	₭ < 1 > Ξ
文件列表		+10-57 A				+= <i>Lt</i> =
基础配置	~	又忤名 ♀	太小 ≄	仔陌尖至「	1◎戊的间 辛	探作
安全管理	~	_SUCCESS 🖋 🗖	0.00B	标准存储	2023-07-12 16:51:49	详情 预览 下载 更多 ▼
权限管理	Ý	part-00000-4695aae9-f286-4e97-8d97-bcf190f16910-c000.csv	11.00B	标准存储	2023-07-12 16:51:48	详情 预览 下载 更多 ▼
域名与传输管理	Ý					

2. 运行在 DLC 上建表、建库,则到 DLC 数据探索页面查看建库、建表。

数据探索		SOL语法参考 🖸 数据探
库表 查询 ♀ +	调度sst_sql ● + ▼	
数据目录 DataLakeCatalog v	部分运行 回 保存 〇、刷新 二 筒 格式化 いい	● 请选择默认数据库 ▼ E farley-test(SuperSQL-S
dlc_db_test	<pre>33 select * from `DataLakeCatalog`.`dlc_db_test`.`ext_test` 34 35 select * from `DataLakeCatalog`.`dlc_db_test`.`ext_test`</pre>	
▼		
▶ 88 视图 ▶ 团 函数	查询结果 统计数据 Task ID SOL详情 导出结果 优化建议 I2 查询耗时 2.75s 数据扫描量 21 B 共2条数据(控制台最多可展示1000条数据) 复制数据 โ	运行历史 下4
	id name	age
	1 Andy 2 Justin	3



PySpark 作业开发指南

最近更新时间: 2025-01-20 12:17:52

应用场景

DLC 支持 Python 语言编写的程序运行作业 。本示例演示通过编写 Python 代码在 对象存储(COS)上读写数 据和在 DLC 上建库表、读写表的详细操作,帮助用户在 DLC 上完成作业开发。

环境准备

依赖: PyCharm 或其他 Python 编程开发工具。

开发流程

开发流程图

DLC Spark JAR 作业开发流程图如下:



创建资源

第一次在 DLC 上运行作业,需新建 Spark 作业计算资源,例如新建名称为 "dlc−demo" 的 Spark 作业资源。 1. 登录 数据湖计算 DLC 控制台,选择服务所在区域,在导航菜单中单击**数据引擎。**

- 2. 单击左上角创建资源,进入资源配置购买页面。
- 3. 在集群配置 > 计算引擎类型选项选择 Spark 作业引擎。



甘山和雪											
基础能直											
计费模式	按量计费	包年包月	详细对比 🖸								
	按量计费适合周期性任务场	最,不使用时可以挂起资源	避免存在费用消耗造成浪费。	包年包月适合有稳定、持续	敕处理的任务需求的场景,无罪	青等待资源拉起。					
地域				华东地区		西南	地区	美国西部	亚太东南		
	1 L			上海	上海金融	成都	重庆	硅谷	新加坡	弗吉尼亚	
	法兰克福										
	区洲地。 法兰克福 处于不同地域的云产品内网2	下互通,购买后不能更换,	请您谨慎选择。 建议选择最素	眨您客户的地域,可降低访	前问时延。						
集群配置	时洲地。 法兰克福 处于不同地域的云产品内网	下互通,购买后不能更换,	请您谨慎选择, 建议选择最亲	起您客户的地域,可降低访	SiPetje,						
	时消地。 法兰克福 处于不同地域的云产品内网2	<u>与遭,购买后不能更换</u> ,	<mark>清您達質选择。</mark> 建议选择最靠	到后您看户的地域,可降低访	500052						
集群配置 基本配置 计第引等类型	DTMHE。 法兰克福 处于不同地域的云产品内网/	K互通,购买后不能更换, Snatk/Falk	<mark>请您逻慎选择,</mark> 建议选择最最 Drestn	起怒客户的地域,可降低的	519at16.						
集群配置 基本配置 计算引察类型	Epittes。 法兰克福 处于不同地域的云产品内保 SparkSQL 不同日間市场不可能学问题。	「互通、駒沢后不能更換。 Spark行业 - Spark201 活用干燥子	<mark>请你谨慎选择,</mark> 建议选择最素 Presto 平平sto	2508年內約地域,可降低方)问时廷。 NH课 Prestore 日本女日本帝	igo.tf.					
集群配置 基本配置 计算引察类型	EXHES。 法兰野福 处于不同地域的云产品内风 SparkSQL 不同引擎文件不同就能让提	<mark>、 Spark作业</mark> SparkのLi近用子相応	<mark>请参谋慎选择,</mark> 建议选择最差 Presto IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII	1528客户的地域,可降低访 1999年,1999年,1999年,1999年 1999年,1999年,1999年,1999年,1999年,1999年,1999年,1999年,1999年,1999年,1999年,1999年,1999年,1999年,199	3月时廷。 过速,Presto选 #子交互式童	能分析。					
集群配置 基本配置 计算引导类型	B)개명。 法兰男福 处于不同地域的云产品内网。 SparkSQL 不同时 原文時不同就面於我 不同时 原文時不同就面於我	「日直、院天岳不能更強。 Spark行业 日本、Sparkのし上海子特定 Spark 0 4	<mark>東空波慎選弾、</mark> 建议送得着素 Presto E電気的高校SOL任务、Spart	1200章中的地域,可译成的 1410章中的地域,可译成的 1410章带于波动地的大学和	5月时经。 世界,Prestone (于文王文重	R9 H .					
■ 集群配置 基本配置 计算引厚类型 内核版本	BYHES, 法兰劳福 处于不同地域的云产品内风 SparkSQL 不同引擎式终不喝款需处理 Spark 32 地向山英王-Const 任于地方	「日通、時天后不能要決。 Sparkfや业 登録、SparkSOLE用子をお Spark 2.4 ズロビックロール・ズロ	第92世前送祥、建议送得着素 Presto 医式的電話のしビジ、Snam	2000年中的地域、可降低の 2011年1月日本地域の任務の 2011年1日日本地域の任務の	5月时经。 处理、PresbJE 于文五式重	能分析。					
■ 集群配置 基本配置 计算列學类型 内核版本	BYNES. 法兰男福 处于不同地域的云产品内保 SparkSQL 不同引進支持不同執筆改變 SparkSQL 不同引進支持不同執筆改變 SparkSQL 不同引進支持不同執筆改變 Bynex 3.2 批作业器子Spark所有主题力。	5日道、 院交后不能要強、 Spark作3业 雪麗、 Spart SOL近明子地で Spark 2.4 不同版本対应成効性不同、		260日本的地域、可降低的 本作业造用于1月5日和式に多 な研究時間2	的时程。 处理、Predože ·デ文互式重	跑分析.					
集群配置 基本配置 计算引擎类型 内碳版本 集群类型	BYNES. 法兰男福 处于不同地域的云产品内闲 SparkSQL 不同引電支持不同前端公理 取作业量子Gpark版本能力. 取作业量子Gpark版本能力. 功准型	Spark行业 Spark名は、1000000000000000000000000000000000000		私どで変 中的地域、 可降低的 MT-W道用于 所式: 株式 任務 MT-W道用于 所式: 株式 任務	ippha. 此语,Prestolal 行文互式重	现分析.					

信息配置 > 资源名称填写 "dlc-demo"。新建资源详细介绍请参见购买独享数据引擎。

高级配置 参数配置	+添加			
集群网段	10.255.0.016 得政 该武国金委明经务之间的网络互通性、在非察药面间场展下,推荐使用款以武置:使用能	关邦查询时,引擎网会请勿与数据源网段重复。		
信息配置				
资源名称	dic-demo			
描述信息	不超过250个字符			
	造镇,长度限制为250			
1	戦于病体 ノ			
标金				

- 4. 单击立即开通,确认资源配置信息。
- 5. 确认信息无误后,单击**提交**,完成资源配置。

上传数据到 COS



🕥 腾讯云

创建名称为 "dlc-demo"的存储桶,上传people.json文件,供 从COS 读写数据示例用,people.json 文件 的内容如下:

- {"name":"Michael"}
 {"name":"Andy", "age":30}
 {"name":"Justin", "age":3}
 {"name":"WangHua", "age":19}
 {"name":"ZhangSan", "age":10}
 {"name":"LiSi", "age":33}
 {"name":"ZhaoWu", "age":37}
 {"name":"MengXiao", "age":68}
 {"name":"KaiDa", "age":89}
- 1. 登录 对象存储 COS 控制台,在左侧菜单导航中单击 存储桶列表。
- 2. 创建存储桶:

单击左上角 创建存储桶,名称项填写 "dlc-dmo",单击下一步完成配置。

3. 上传文件:

单击**文件列表 > 上传文件**,选择本地"people.json"文件上传到"dlc-demo-1305424723"桶里 (-1305424723是建桶时平台生成的随机串),单击**上传**,完成文件上传。新建存储桶详情请参见 创建存储 桶 。

← 返回桶列表		dic-demo-1305424723 /							文档指引 12
搜索菜单名称	Q		夹 文件碎片 清空存储桶	更多操作				在线编辑器 团	
概范			当前 唐 拟目 录下的 对象	3 Q 刷新 共0个文件			每页 100 个对象	K K 1 > ::: =	
文件列表	1	文件名 ±					a ±	操作	
基础配置			上传文件			×			
安全管理			送择上传对象 之 2 1	设置对象属性					
权限管理			3			- 1			
域名与传输管理			选择文件 上传至 dlc-demo-130542	24723/					
容错容灾管理			若上传路径中存在同名文件,上传将覆盖原有	文件。					
日志管理			上传操作将产生请求次数和上行流量,其中请	求 暨按万次收费,上行流量 暨免费。洋情清重	管着计费项 🖸 及产品定价 🖸				
内容审核 HOT			文件/文件夹	大小	操作				
数据处理			people.ison	241.00B	删除				
任务与工作流 🛄	~ т		h - h - h - h - h						
数据监控									
函数计算									
CVM 挂载 COS									
				Λ					
				参数配置 上传 4					9

新建 Python 项目

通过 PyCharm 新建一个名称为"demo"的 项目。



编写代码

1. 新建 cos.py 文件,编写代码,功能为从 COS 上读写数据和在 DLC 上建库、建表、查询数据和写入数据。

```
from pyspark.sql import SparkSession
from pyspark.sql import Row
    spark = SparkSession\
       .builder\
        .appName("Operate data on cos") \
        .enableHiveSupport() \
        .getOrCreate()
   # 1.读cos上的数据 支持多种类型的文件 如 json,csv,parquet,orc,text
    read_path = "cosn://dlc-demo-1305424723/people.json"
   peopleDF = spark.read.json(read_path)
   # 2.对数据做操作
   peopleDF.createOrReplaceTempView("people")
   data_src = spark.sql("SELECT * FROM people WHERE age BETWEEN 13
   data_src.show()
   # 3. 写数据
   write_path = "cosn://dlc-demo-1305424723/people_output"
   data_src.write.csv(path=write_path, header=True, sep=",",
    spark.stop()
```

2. 新建 db.py 文件,编写代码,功能为 DLC 上建库、建表、查询数据和写入数据。





```
.builder \
    .enableHiveSupport() \
    .getOrCreate()
# 1.建数据库
# 2.建内表
spark.sql("CREATE TABLE IF NOT EXISTS
# 3. 写内数据
spark.sql("INSERT INTO `DataLakeCatalog`.`dlc_db_test_py`.`test`
# 4.查内数据
spark.sql("SELECT * FROM `DataLakeCatalog`.`dlc_db_test_py`.`test`
# 5.建外表
spark.sql("CREATE EXTERNAL TABLE IF NOT EXISTS
# 6.写外数据
spark.sql("INSERT INTO
# 7.查外数据
spark.sql("SELECT * FROM
```

建外表时,可按照 上传数据到 COS 的步骤 先在桶里建对应表名文件夹保存表文件。



demo-1305424723 /				文档
1 上传文件 创建文件夹 文件碎片	清空存储桶 更多操作 •	7		在线编辑器[
前缀搜索 ▼ 只支持搜索当前虚拟目录下的对象		析 共1个文件	每页 100 个对象	हे 🖌 ◀ 1 ▶ 📰
文件名 💲	* 小大	存储类型 🔻	修改时间 🕈	操作
🖻 ext_test/ 🖍			-	设置权限 统计 更多 ▼
🗎 ext_test/ 🖍	-			设置权限 统计 更

调式

PyCharm 调式无语法错误。

上传 py 文件到 COS

登录 COS 控制台,参考上文上传数据到 COS 的步骤将 cos.py、db.py上传到 COS。

新建 Spark Jar 数据作业

创建数据作业前,您需先完成数据访问策略配置,保证数据作业能安全地访问到数据。配置数据访问策略详情请参见 配置数据访问策略。如已配置数据策略名称为:qcs::cam::uin/100018379117:roleName/dlc-demo。

- 1. 登录 数据湖计算 DLC 控制台,选择服务所在区域,在导航菜单中单击**数据作业。**
- 2. 单击左上角创建作业按钮,进入创建页面。
- 3. 在作业配置页面,配置作业运行参数,具体说明如下:

配置参数	说明
作业名称	自定义 Spark 作业名称,例如:cosn_py
作业类型	选择 批处理类型
数据引擎	选择 创建资源 步骤创建的 dlc-demo 计算引擎

程序包	选择 COS,在 上传 py 文件到 COS 步骤的上传 py 文件: • 从 COS 上读写数据就选择: cosn://dlc-demo-1305424723/cos.py • 在 DLC 上建库、建表等选择: cosn://dlc-demo-1305424723/db.py
数据访问策略	选择该步骤前创建的策略 qcs::cam::uin/100018379117:roleName/dlc-demo

其他参数值保持默认。

编辑作业		
基本信息 ▲		
作业名称*	db_py	
	支持中文、英文、数字与"_", 最多100个字符	
作业类型 *	批处理 流处理 SQL作业	
数据引擎*	(按量计费)	
		一 」络配置信息可至网络配置〔
程序包*	○ 对象存储COS ○ 本地上传	
	cosn://	选择COS位置
	需具备COS相关权限,可选择jar/py文件	
程序入口参数	请输入程序入口参数,多个参数空格分隔。不超过65536个字符	
作业参数 (config)	示例: spark.network.timeout=120s	
保存	取消	

4. 单击保存,在 Spark 作业页面可以看到创建的作业。

运行并查看作业结果

- 1. 运行作业:在Spark 作业页面,找到新建的作业,单击运行,即可运行作业。
- 2. 查看作业运行结果:可查看作业运行日志和运行结果。

查看作业运行日志

1. 单击作业名称 > 历史任务 查看任务运行状态:



数据作业(S,┌₩ →				Spark作业详情	i				
Spark作业	作业配置 Session	管理			作业信息	2 历史任务	监控告警			
创建作业		Q	全部	请选择创建人	请选择执行状	·····································	近7天	近30天	2023-07-08 ~ 2023-07-14	刷新
作业名称	作业ID	作业类型	作业文件	当前任务数	任务ID 3	执行状态	任务提交时间	\$ ř		
db_py 🖻	batch_3ae998	批处理	cosn://e1305424	0	02f21ed4-1	成功	2023-07-12 17	:54:24 1	min47s 查看详情 Spark UI	
cons_py	batch_df8ee8 T	批处理	cosn:/ -1305424	0	565ecb8a-2	成功	2023-07-12 17	:48:19 1	min46s 查看详情 Spark UI	
					b28c614d-4	失败	2023-07-12 17	:45:42 4	7.2s 查看详情 Spark UI	

2. 单击任务ID > 运行日志,查看作业运行日志。

A constraint of part of par	There a							
作业名称 きt f 作业D: 「 定到台最多展示最近1000条信息: 近7天 近30天 2023-08-01 17:36:02 ~ 2023-08-07 17:36:02 「 按时间降序 ▼ の 創建下載任务 ご近7天 近30天 2023-08-01 17:36:02 ~ 2023-08-07 17:36:02 「 按时间降序 ▼ 日志级别: All 23/08/07 17:32:24 INFO KubernetesClusterSchedulerBackend\$KubernetesDriverEndpoint: Asking each executor to shut down 23/08/07 17:32:24 INFO KubernetesClusterSchedulerBackend\$KubernetesDriverEndpoint: Asking each executor to shut down 23/08/07 17:32:24 INFO SparkUI: Stopped Spark web UI at http://spark-5f641289cf55a6bc-driver-svc.default.svc:4040 ++++ 2[Lucy] 3] 1[LLy] 12] ++++ 23/08/07 17:32:24 INFO DAGScheduler: Job 5 finished: show at DbService java:37, took 0.161365 s 23/08/07 17:32:24 INFO DAGScheduler: Job 5 finished: show at DbService java:37, took 0.161365 s 23/08/07 17:32:24 INFO DAGScheduler: Job 5 is finished. Cancelling potential speculative or zombie tasks for this job 23/08/07 17:32:24 INFO DAGScheduler: ResultStage 5 (show at DbService java:37) finished in 0.157 s 23/08/07 17:32:24 INFO DAGScheduler: ResultStage 5 (show at DbService java:37) finished in 0.157 s 23/08/07 17:32:24 INFO TaskSchedulerImpI: Removed TaskSet 5.0, whose tasks have all completed, from pool	基本信息	运行日志						
控制台最多展示最近1000条信息 近7天 近30天 2023-08-01 17:36:02 ~ 2023-08-07 17:36:02 授助问降序 》	作业名称	est 🖬 作	业ID:		6			
近7天 近30天 2023-08-01 17:36:02 ~ 2023-08-07 17:36:02 計 按时间降序 () () () () () () () () () () () () ()	控制台最多展	示最近1000条(言息					
日志 公部: 日志 级别: 日志 级别: All 、 10000000000000000000000000000000000	近7天	近30天	2023-08-01 17:36:02	~ 2023-08-07 17:36:02	i i	按时间降序 🔻	φ	创建下载任务
23/08/07 17:32:24 INFO KubernetesClusterSchedulerBackend\$KubernetesDriverEndpoint: Asking each executor to shut down 23/08/07 17:32:24 INFO KubernetesClusterSchedulerBackend: Shutting down all executors 23/08/07 17:32:24 INFO SparkUI: Stopped Spark web UI at http://spark-5f641289cf55a6bc-driver-svc.default.svc:4040 +++-+ 2[Lucy] 3] 1[LiLy] 12] +++-+ id[name]age] +++ 23/08/07 17:32:24 INFO DAGScheduler: Job 5 finished: show at DbService.java:37, took 0.161365 s 23/08/07 17:32:24 INFO TaskSchedulerImpI: Killing all running tasks in stage 5: Stage finished 23/08/07 17:32:24 INFO DAGScheduler: Job 5 is finished. Cancelling potential speculative or zombie tasks for this job 23/08/07 17:32:24 INFO DAGScheduler: ResultStage 5 (show at DbService.java:37) finished in 0.157 s 23/08/07 17:32:24 INFO TaskSchedulerImpI: Removed TaskSet 5.0, whose tasks have all completed, from pool	日志名称:				日志级别:	All		•
+++ id name age +++ 23/08/07 17:32:24 INFO DAGScheduler: Job 5 finished: show at DbService.java:37, took 0.161365 s 23/08/07 17:32:24 INFO TaskSchedulerImpl: Killing all running tasks in stage 5: Stage finished 23/08/07 17:32:24 INFO DAGScheduler: Job 5 is finished. Cancelling potential speculative or zombie tasks for this job 23/08/07 17:32:24 INFO DAGScheduler: ResultStage 5 (show at DbService.java:37) finished in 0.157 s 23/08/07 17:32:24 INFO TaskSchedulerImpl: Removed TaskSet 5.0, whose tasks have all completed, from pool	2 Lucy 3 1 LiLy 12	- 						
 Id[name]age] +++ 23/08/07 17:32:24 INFO DAGScheduler: Job 5 finished: show at DbService.java:37, took 0.161365 s 23/08/07 17:32:24 INFO TaskSchedulerImpl: Killing all running tasks in stage 5: Stage finished 23/08/07 17:32:24 INFO DAGScheduler: Job 5 is finished. Cancelling potential speculative or zombie tasks for this job 23/08/07 17:32:24 INFO DAGScheduler: ResultStage 5 (show at DbService.java:37) finished in 0.157 s 23/08/07 17:32:24 INFO TaskSchedulerImpl: Removed TaskSet 5.0, whose tasks have all completed, from pool 	++	+						
23/08/07 17:32:24 INFO DAGScheduler: Job 5 finished: show at DbService.java:37, took 0.161365 s 23/08/07 17:32:24 INFO TaskSchedulerImpl: Killing all running tasks in stage 5: Stage finished 23/08/07 17:32:24 INFO DAGScheduler: Job 5 is finished. Cancelling potential speculative or zombie tasks for this job 23/08/07 17:32:24 INFO DAGScheduler: ResultStage 5 (show at DbService.java:37) finished in 0.157 s 23/08/07 17:32:24 INFO TaskSchedulerImpl: Removed TaskSet 5.0, whose tasks have all completed, from pool	Id name ao	je +						
23/08/07 17:32:24 INFO TaskSchedulerImpl: Killing all running tasks in stage 5: Stage finished 23/08/07 17:32:24 INFO DAGScheduler: Job 5 is finished. Cancelling potential speculative or zombie tasks for this job 23/08/07 17:32:24 INFO DAGScheduler: ResultStage 5 (show at DbService.java:37) finished in 0.157 s 23/08/07 17:32:24 INFO TaskSchedulerImpl: Removed TaskSet 5.0, whose tasks have all completed, from pool	23/08/07 17	:32:24 INFO D	AGScheduler: Job 5 fini	shed: show at DbService.	ava:37, too	ok 0.161365 s		
23/08/07 17:32:24 INFO DAGScheduler: Job 5 is finished. Cancelling potential speculative or zombie tasks for this job 23/08/07 17:32:24 INFO DAGScheduler: ResultStage 5 (show at DbService.java:37) finished in 0.157 s 23/08/07 17:32:24 INFO TaskSchedulerImpl: Removed TaskSet 5.0, whose tasks have all completed, from pool	23/08/07 17	:32:24 INFO Ta	askSchedulerImpl: Killing	g all running tasks in stage	e 5: Stage fi	inished		
23/08/07 17:32:24 INFO DAGScheduler: ResultStage 5 (show at DbService.java:37) finished in 0.157 s 23/08/07 17:32:24 INFO TaskSchedulerImpl: Removed TaskSet 5.0, whose tasks have all completed, from pool	23/08/07 17	:32:24 INFO D	AGScheduler: Job 5 is f	inished. Cancelling potent	ial speculat	tive or zombie ta	sks for this	job
23/08/07 17:32:24 INFO TaskSchedulerImpl: Removed TaskSet 5.0, whose tasks have all completed, from pool	23/08/07 17	:32:24 INFO D	AGScheduler: ResultSta	age 5 (show at DbService	java:37) fin	ished in 0.157 s		
	23/08/07 17	:32:24 INFO Ta	askSchedulerImpl: Remo	oved TaskSet 5.0, whose	asks have	all completed, fro	om pool	

查看作业运行结果

分 腾讯云

1. 运行从 COS 读写数据示例,则到 COS 控制台查看数据写入结果。



🗲 返回桶列表		-1305424723 / people_output				文档指引 🗹
搜索菜单名称 Q		上传文件 创建文件夹 更多操作 ▼				在线编辑器 😡
概览						
文件列表		前缀搜索 ▼ 只支持搜索当前虚拟目录下的对象	☑ Q 刷新	共 2 个文件	每页 100 个对象	₩ < 1 > Ξ
基础配置	~	文件名 \$	大小 🛊	存储类型 🔻	修改时间 🕈	操作
安全管理	Ŷ	_SUCCESS 🖍 🖬	0.00B	标准存储	2023-07-12 16:51:49	详情 预览 下载 更多 ▼
权限管理 域名与传输管理	×	part-00000-4695aae9-1286-4e97-8d97-bcf190116910-c000.csv	11.00B	标准存储	2023-07-12 16:51:48	详情 预览 下载 更多 ▼

2. 运行在 DLC 上建表、建库,则到 DLC 数据探索页面查看建库、建表。

数据探	家 ◎ 广州 ▼				
库表	查询	¢ +	调度-cry_test_sql ●	+ •	
数据目录 dlc_db_t 请输入表 ▼	DataLakeCatalog est_py 名称 _db_test_py 表 ei ext_test	V V Q	部分运行 32 33 select.*.fro 34 35 36 37	〕保存 🕃 刷新 🛆 格式化 ΩΩ m `DataLakeCatalog`.`dlc_db_test_py`.`test`	€ 请选择默认数据库 ▼
÷ 6	视图 函数		查询结果 统计数据 Task ID SQL详情 导出; 查询耗时 10.43s 数据指 共 2 条数据 (控制台最多) id 1 2	结果 优化建议 ☑ 描量 21 B 可展示1000条数据)复制数据□ name Andy Justin	age 12 3

查询性能优化指南

最近更新时间: 2025-04-21 17:21:22

前言

为了提升任务执行效率,您可以首先对现有任务或引擎进行洞察分析(仅 Spark)查看是否有可调优空间,其次 DLC 在计算过程中有许多优化措施,例如数据治理、lceberg 索引、缓存等,您可以结合这些优化措施对任务做更 全面的优化。正确使用不仅可以减少不必要的扫描费用,甚至可以提升几倍甚至几十倍的效率。 下面提供一些不同层面的优化思路。

Spark

任务洞察

基于 Spark 引擎采集的各种 Metrics 信息, 任务洞察模块 实现了算法自动分析 Metrics 指标,为您提供任务的 洞察情况,建议先基于推荐的参数和调优方案实施。

可以解决常见的 SQL 问题,包括:

- 资源抢占
- 数据倾斜
- ■磁盘不足或内存 OOM
- 慢 Task
- 小文件过多
- Shuffle 不合理

执行计划分析

当任务洞察仍然无法满足性能调优时,根据Spark UI提供的当前任务详细的执行计划,可以进一步分析 SQL 的瓶 颈。



☰ ❷腾讯云 ⋔┆	空制台						Q 支持通过实例	ID、IP、名称等扬	建索资源	快捷罐 /	集团账号	备案	工具 箸	服支持	费用	中文	٢	¢ ₿	2475491	^{5@q} 子账号 🛛
数据湖计算	●热门新品 1	V1指导,拖拉式完成	戊多端建站,像做Pl	PT一样做网站 查看	详情〉															x
概览	历史任务实例	©r₩ ·	Y													5	回到旧版	ā 历史任务家	例使用指南	💈 <mark>/ i</mark> AI助手
数据开发与探索 ②数据探索		模块中提交运行的任	务运行信息,包括SC	L任务等。管理员可以	(查询近45天所)	有的任务,普通月	用户可以查询与自己	已相关的近45天的	任务。了 解更多 亿											x
■ 数据作业	批量终止	导出全部记录			全部	近7天	近24小时	选择时间	选择时间	ö	默认搜索	任务ID,支	持下拉选择搜索	發任务名称	、任务内容	子渠道,言	多个过滤机	i签用回车键分I	100 o	¢φ
资源管理 「SuperSQL引擎	全部(208)	▶ 执行中(0)	▲ 排队中(0)	○ 启动中(0)																
⑦ 标准引擎	任务名称/ID			计算资源	▼ 资源	胚组名称	消耗CU	*时 (i) \$	计算耗时 🛈 🛊	数据扫描	量 (1) \$	创新	建人/执行人	T	子渠道		3 1	肇执行时间 🛈	\$ 操作	
② 网络连接配置 ○ 存储管理	cu: mizer 7325-25. 🎽		06c11	≝ ■~~	PAR def	ault-rg-ç ız7c	uoj8 0.5275	00	21s			100	000d :9 d 17	Ì			20	25-04-18 18:1	1:2 查看详情	Spark UI
🗟 元数据管理	customiz _2 69862a	2504 £ 31/ 5 _9 %	6 d7d0 cd5	▶ 近春生	SPAR def	ault-rg-ç ız7c ii	iu ja 0.0725	00	4s			100	000(:9 21:	i			20	25-04-18 18:1	D:! 查看详情	Spark UI
运维管理																				
■ 历史任务实例	custon d 61d07 /l 3.	5 1 ,8 8 04 9 ,71 - 54 1 7,	7baf6 "3d		SPAR def	ault-rg-g iz q6i	iuo, 0.05111	1	3s			100	000(9) 1	i			20	25-04-18 18:1	0:4 查看详情	Spark UI
□ 会话管理	cus Jmiz . 2	0/ 8 10 714	4f b 4	est 项标 ii	SPAR def	ault-rg-g 1 7q6i	u) 0.11444	14	8s			100	0000 8 7	Ð			20	25-04-18 18:1	0:: 查看详情	Spark UI

Spark UI 可以后从 SQL DataFrame 页面查看执行计划。



执行计划的查看和分析可以结合 Stages 页面的 Stage 耗时,按照以下步骤及优化思路。优化前,建议提前对输入、Join、输出的数据量级有初步的判断。



第一步:分析 Scan 算子

Scan 算子是负责读取所有库表的操作,Stages 页面包含有 Input 数据的 Stage 一般对应 Scan 算子,当这种 Stage 耗时长时需要重点查看 Scan 算子的指标。

Details for Query 5

Submitted Time: 2025/04/15 15:17:41 Duration: 3 s Succeeded Jobs: 3

 \Box Show the Stage ID and Task ID that corresponds to the max metr



Scan算子示例

以下 Metrics 信息在 Hive/Iceberg 表上名称有所不同,但是指标内容及含义基本相同。

- number of files read: 如果读取的文件数太多,而 size of files read/number of output rows 太小,可以认为小文件太多,需要合并小文件,或者使用 数据优化来配置自动合并。
- scan time total (min, med, max):分布式读取时的时间分布,如果存在部分任务读取太慢,可能是因为存储的 COS 桶限流或限频,建议查看 COS 桶监控,可以提工单给 COS 调整带宽。
- size of files read: 辅助判断读取的数据量是否合理。如果发现读取的文件太大,需要判断是否存储的文件格式 不对,或有脏数据存在。
- number of output rows: 辅助判断读取的数据量是否合理。如果发现读取的数据量太大,需要结合 Scan 的 Filter 条件判断是否过滤条件有误导致扫描了全表。Spark引擎可以自动下推过滤条件到 Scan 算子,但如果某 些特殊情况没有自动下推过滤条件,可以提前将 SQL 中的 where 下推。

第二步:寻找计算瓶颈

建议先从 Stages 页面查看耗时最长的 StageId,再根据 StageID 到执行计划(页面上需要勾选 Show the Stage ID and Task ID)页面分析。

在任务洞察中可以识别常见的性能瓶颈点,而在执行计划中分析,往往需要结合瓶颈算子的上下文来分析。

Project 算子

Porject 算子往往需要关注是否有大量重复的 Expression 定义,这种建议 SQL 中改写来避免大量的重复计算。

• Filter 算子

过滤算子的过滤条件一般重点检查是否正确过滤数据,重点关注 Filter 前后数据量是否有正常的减少。

Join 算子

Join 是较为常见出现性能瓶颈的算子。常见的优化思路如下。



SortMergeJoin 出现 skewJoin,尽管 Spark 会默认对倾斜的 Join 做优化,但是当数据膨胀系数不大

(spark.sql.adaptive.skewJoin.skewedPartitionFactor=5.0,超过平均值5倍认为是倾斜),或者当 倾斜并不是特别突出

(spark.sql.adaptive.skewJoin.skewedPartitionThresholdInBytes=256MB,大于256MB才会 认为倾斜)时一旦导致 spill 性能会急剧下降,仍然有一定的调优空间。



Skew Join示例

SortMergeJoin 数据量太大,可以结合 SQL 和执行计划分析:

- 1. 是否出现了笛卡尔积,join on 的条件字段存在重复,导致 left join 或 inner join 数据膨胀。
- 是否有些过滤条件没有下推,例如 Aggregate 操作可能会隔绝过滤条件的下推,设置 spark.sql.optimizer.AggregatePushdownThroughJoins.enabled=true 可以查看是否可以下推 Aggregate 后的算子。

SortMergeJoin 出现大表 Join 小表,是否可以考虑替换为 BroadcastHashJoin。通过 Spark hints 或调大 spark.sql.autoBroadcastJoinThreshold=10MB 来让小的表可以广播。

Aggregate 算子

Aggregate 比较常见的是倾斜的问题,当 aggregate key 明显过多时,会导致慢 task 的产生。如果调整分 区数无法解决倾斜,一般只能通过重新调整 key 来打散过多的 Key,来重新做业务 SQL 设计。此外,如果 Aggregate 的函数存在重复计算,也可以考虑从 SQL 层面做精简。



• Exchange + AQEShuffleRead 算子

AQE 是 Spark 提供的在 shuffle 时自动合并过小的分区、打散过大的分区的特性。上面的 Join/Aggregate 往往也会需要查看 Exchange+AQEShuffleRead 的算子。AQE 是根据 spark.sql.adaptive.advisoryPartitionSizeInBytes=64MB 来判断 ShuffleRead 时的分区大小的。所 以当业务的出现数据膨胀非常多,例如小数据量 Join 产生非常大的数据量,可以考虑调小 advisory的 partition size;或者在 Shuffle 经过处理数据量缩小得非常多,可以考虑调大 advisory 的 partition size 来避免产生过多的小文件。

第三步:查看 Insert/Write 算子

Stages 页面包含 Output 的 Stage, 一般可以对应到写入的算子。

一般任务结尾都为写入的算子,例如 Insertxxx/Overwritexxx/AppendData 等,常常需要关注是否有写入过 多的小文件,以及 commit 的时间。



InsertIntoHadoopFsRelationCommand示例

常见的指标含义:

- number of output row:写入的数据量是否符合预期,尤其在性能变动大时需要关注。
- task/job commit time:数据写入后,需要 commit 确认整张表的数据写入正确,如果 commit 时间过长, 一般是普通桶 rename 的时间过长,可以考虑换元数据加速桶,可以工单咨询 COS 桶。
- number of dynmaic part: 默认动态分区写入,如果只写入了一个分区,需要查看写入算子前是否有 Exchange 算子只对某个只有一个值的字段做了分区,这会导致分布式写边成单线程写。如果是 Hive 表写入 可以通过 Spark hints 调整写入时的分区策略;如果是 Iceberg 表则需要调整 `spark.sql.iceberg.distribution-mode`=none。

Presto

优化 SQL 语句

场景: SQL 语句本身不合理,导致执行效率不高。

优化 JOIN 语句

当查询涉及 JOIN 多个表时, Presto 引擎会优先完成查询右侧的表的 JOIN 操作,通常来说,先完成小表的 JOIN,再用结果集和大表进行 JOIN,执行效率会更高,因此 JOIN 的顺序会直接影响查询的性能,DLC presto 会自动收集内表的统计数据, 利用 CBO 对查询中的表进行重排序。

对于外表,通常用户可以通过 analyze 语句完成统计数据的收集,或者手动指定 JOIN 的顺序。如需手动指定请按 表的大小顺序,将小表放在右侧,大表放在左侧, 如表A > B > C, 例如:select * from A Join B Join C。需 要注意的是,这不能保证所有场景下都能提升效率,实际上这取决于 JOIN 后的数据量大小。

优化 GROUP BY 语句

合理安排 GROUP BY 语句中字段顺序对性能有一定提升,请根据聚合字段的基数从高到低进行排序,例如:

//高效的写法
SELECT id,gender,COUNT(*) FROM table_name GROUP BY id, gender;
//低效的写法
SELECT id,gender,COUNT(*) FROM table_name GROUP BY gender, id;

另一种优化方式是,尽可能地使用数字代替具体分组字段。这些数字是 SELECT 关键字后的列名的位置,例如上面 的 SQL 可以用以下方式代替:

SELECT id,gender,COUNT(*) FROM table_name GROUP BY 1, 2;

使用近似聚合函数

对于允许有少量误差的查询场景,使用这一些近似聚合函数对查询性能有大幅提升。 例如,Presto 可以使用 APPROX_DISTINCT()函数代替 COUNT(distinct x),Spark 中对应函数为 APPROX COUNT DISTINCT。该方案缺点是近似聚合函数有大概2.3%的误差。

使用 REGEXP_LIKE 代替多个 LIKE

当 SQL 中有多个 LIKE 语句时,通常可以使用正则表达式来代替多个 LIKE,这样可以大幅提升执行效率。例如:

SELECT COUNT(*) FROM table_name WHERE field_name LIKE '%guangzhou%' OR LIKE '%beijing%' OR LIKE '%chengdu%' OR LIKE '%shanghai%'

可以优化成:

```
SELECT COUNT(*) FROM table_name WHERE regexp_like(field_name,
'guangzhou|beijing|chengdu|shanghai')
```

数据治理

数据治理适用场景

场景:实时写入。Flink CDC 实时写入通常采用 upsert 的方式写入,该流程在写入过程中会产生大量的小文件, 当小文件堆积到一定程度后会导致数据查询变慢,甚至超时无法查询。 可以通过以下方式查看表文件数量和快照信息。

SELECT COUNT(*) FROM [catalog_name.][db_name.]table_name\$files; SELECT COUNT(*) FROM [catalog_name.][db_name.]table_name\$snapshots;

例如:

SELECT COUNT(*) FROM `DataLakeCatalog`.`db1`.`tb1\$files`; SELECT COUNT(*) FROM `DataLakeCatalog`.`db1`.`tb1\$snapshots`;

表文件、快照数量过多时,可以参考文档 开启数据治理 启用数据治理功能。

数据治理效果

开启数据治理后,查询效率得到显著提升,例如下表对比了合并文件前后的查询耗时,该实验采用16CU presto, 数据量为14M,文件数量2921,平均每个文件0.6KB。

执行语句	是否合并文件	文件数量	记录条数	查询耗时	效果
SELECT count(*) FROM tb	否	2921个	7895条	32s	油在400 %
SELECT count(*) FROM tb	是	1个	7895条	2s	(1)2 (1)2 (1)2 (1)2 (1)2 (1)2 (1)2 (1)2

分区

分区能够根据时间、地域等具有不同特征的列值将相关数据分类存储,这有助于大幅减少扫描量,提升查询效率。关 于 DLC 外表分区更多详情信息,请参考一分钟入门分区表。下表展示了在数据量为66.6GB,数据记录为14亿 条,数据格式为 orc 的单表中,分区和不分区时查询耗时和扫描量的效果对比。其中`dt`是含有1837个分区的分区 字段。

	未分	X	分区		非可	
查询语句	耗 时	扫描量	耗 时	扫描 量	对比	扫描量对比



SELECT count(*) FROM tb WHERE dt='2001-01-08'	2. 6 s	235.9 MB	48 0m s	16. 5 KB	快 81%	少99.9%
SELECT count(*) FROM tb WHERE dt<'2022-01-08' AND dt>'2001-07-08'	3. 8 s	401.6 MB	2.2 s	2.8 MB	快 42%	少99.3%

从上表中可以看出,分区可以有效地降低查询延时和扫描量,但过度分区可能适得其反。如下表所示。

本沟语句	未分区		分区		キモロナマナトレ	ᆟᄲᄅᆎᅛ	
旦间后り	耗时	扫描量	耗时	扫描量	ҟѢҏ҄ӱӼӱӷҍ	扫 捆重刈16	
SELECT count(*) FROM tb	4s	24MB	15s	34.5M B	慢73%	多30%	

建议您在 SQL 语句中通过 WHERE 关键字来过滤分区。

缓存

在如今分布式计算和存算分离的趋势下,通过网络访问元数据以及海量数据将会受到网络 IO 的限制。DLC 默认开 启以下缓存技术大幅降低响应延时,无需您介入管理。

- Alluxio: 是一种数据编排技术。它提供缓存,将数据从存储层移动到距离数据驱动型应用更近的位置从而能够 更容易被访问。Alluxio 内存至上的层次化架构使得数据的访问速度能比现有方案快几个数量级。
- RaptorX:是Presto的一个连接器。它像 Presto 一样运行在存储之上,提供亚秒级延迟。目标是为 OLAP 和交互式用例提供统一、廉价、快速且可扩展的解决方案。
- 结果缓存: Result Cache,对于重复的同一查询进行缓存,极大提高速度和效率。

DLC Presto 引擎默认支持 RaptorX 和 Alluxio 分级缓存,在短时间内相同任务场景中可以有效地降低延时。 Spark、Presto引擎均支持结果缓存。

下表是在总数据量为1TB的 Parquet 文件中的 TPCH 测试数据,本次测试选用16CU Presto。因为测试的是缓存功能,所以主要从 TPCH 中选择 IO 占用比较大的 SQL ,涉及的表主要有 lineitem、orders、customer 等表,涉及的 SQL 为 Q1、Q4、Q6、Q12、Q14、Q15、Q17、Q19 以及 Q20。其中横坐标表示SQL语句,纵坐标表示运行时间(单位秒)。





需要注意的是,DLC Presto 引擎会根据数据访问频率动态加载缓存,所以引擎启动后首次执行任务无法命中缓 存,这导致首次执行仍受网络 IO 限制,但随着**执行次数增加**,该限制明显得到缓解。如下表展示了 presto 16cu 集群三次查询的性能比较。

查询语句	查询	耗时	数据扫描量
SELECT *	第一次查询	3.2s	40.66MB
FROM table_namewhere udid='xxx';	第二次查询	2.5s	40.66MB
	第三次查询	1.6s	40.66MB

您可以在DLC控制台 数据探索 功能中查看执行的SQL任务的缓存命中情况。

← 查询结果 统计数据			运行历史 下载历史
数据扫描量 798 _Β	自适应shuffle ① shuffle子节数 溢出到COS	張存加達 ① ResultCache 緩存命中 否 FragmenCache命中事 Altudo 缓存命中	物化视觉加速① 自动改变到 DataLakeCatalog.mv_test4.mv_student 利用数据 4468
■ 预处理:2.86s		■ 执行: 8.51s	■ 获取结果:192.00ms

索引

内表+索引的建表方式相对于外表,在时间和扫描量上均会大幅减小,关于创建表的更多详细信息,请参考数据表管理。

创建表后根据业务使用频率在 insert 前建立索引,WRITE ORDERED BY 后的索引字段。

alter table `DataLakeCatalog`.`dbname`.`tablename` WRITE ORDERED BY
udid;



下表展示了 presto 16cu 集群在外表和内表(加索引)上查询性能比较

表类型	查询	耗时	数据扫描量
	第一次查询	16.5s	2.42GB
外表	第二次查询	15.3s	2.42GB
	第三次查询	14.3s	2.42GB
	第一次查询	3.2s	40.66MB
内表(索引)	第二次查询	2.5s	40.66MB
	第三次查询	1.6s	40.66MB

从表中可以看出,内表+索引的建表方式相对于外表,在时间和扫描量上均会大幅减小,并且由于缓存加速,**执行时 间也会随着执行次数的增加而减少**。

同步查询和异步查询

DLC 针对于 BI 场景进行了特别的优化,可以通过配置引擎参数dlc.query.execution.mode来开启同步模式或 者异步模式(只支持 presto 引擎)。取值介绍如下。

- async(默认):该模式任务会完成全量查询计算,并将结果保存到COS,再返回给用户,允许用户在查询完成后下载查询结果。
- sync: 该模式下,查询不一定会执行全量计算,部分结果可用后,会直接由引擎返回给用户,不再保存到
 COS。因此用户可获得更低查询延迟和耗时,但结果只在系统中保存30s。推荐不需要从 COS 下载完整查询
 结果,但期望更低查询延迟和耗时时使用该模式,例如查询探索阶段、BI 结果展示。

配置方式:选择数据引擎后,支持对数据引擎进行参数配置,选择数据引擎后,在高级设置单击添加即可进行配置。



	📄 请选择默认数据库 🔻 담 public	-engine 💮 presto 🔻
🗄 数据引擎		
public-engine		•
⑦ 引擎 先选择数据引擎	⁸ 才可选择引擎	
presto		▼
① 创建引擎		
宫级设置 ▲		配置说明 🖸

资源瓶颈

评估资源是否达到瓶颈,DLC 提供引擎的 CPU、内存、云盘、网络等资源监控。您可以根据业务规模调整资源规格,变配请参考调整配置费用说明。查看引擎资源使用情况步骤如下:

- 1. 打开左侧数据引擎标签页。
- 2. 单击相应引擎的右侧监控按钮。
- 3. 跳转到腾讯云可观测平台,可以查看到所有监控指标,如下图所示。详细操作以及监控指标请参考数据引擎监控。同时您也可以针对每个指标进行告警配置,详细介绍请参考监控告警配置。





☆ 返回列表 ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ●	○ 关照 ▼ ··· 2 显示磁例	
⇒ сри		
518/54/6/wr 17 Acpu/58.42/819(1) () () () () () () () () () () () () ()	1932 27.74 1932 27.74 1933 29.74 1933 1934 1952 2000 2008 20:16 1938 19:44 19:52 2000 20:08 20:16 配大性: 27.74 配小性: 1:16 平均性: 8:00	SERRENANT SAGANAT HARRENN () I) <
51994(executor 9.6c) 12 12 13 13 13 13 13 13 13 13 13 13	は用料用的+pup-11分割用(h) ① 1934 9.15 1934 9.15 1938 19,44 19,52 2000 2008 2016 ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■	Startingshows/2018(N) () 2014 State 35 2024 State 21 2014 State 14 1928 State 1936 State 2000 State
	3185630mr380588.x888(n) () () (2019 91.30 100	

其他因素

自适应 shuffle

DLC 默认关闭自适应 shuffle。当您任务出现 No space left 等磁盘空间不足的问题时,可以开启自适应 shuffle,这是一套即能支持有限本地磁盘的常规 shuffle,又能保证在大 shuffle 和数据倾斜等场景下的稳定性。 自适应 shuffle 带来的优势:

1. 降低存储成本:集群节点的磁盘挂载量进一步降低,一般规模集群每节点只要50G、大规模集群也不超200G。

2. 稳定性:对于 shuffle 数据量剧增或数据倾斜场景任务执行的稳定性不会再因本地磁盘限制而失败。

尽管自适应 shuffle 带来存储成本的降低和稳定性提升,但在某些场景下,如资源不足时,会带来约15%的延时。 【 开启方式 】

集群配置中新增以下配置开启:

spark.shuffle.manager=org.apache.spark.shuffle.enhance.EnhancedShuffleManager 可选配置:

如果您希望提前写入数据到远端,确保磁盘有足够空间,避免影响其他任务,可以考虑以下配置:

spark.shuffle.enhanced.usage.waterLevel1=磁盘比例,shuffle 结果写到远端时磁盘已使用的比例,默认0.7。

spark.shuffle.enhanced.usage.waterLevel2=磁盘比例,shuffle spill 数据写到远端时磁盘已使用的比 例,默认0.9。

集群冷启动

DLC 支持自动或者手动挂起集群,挂起后不再产生费用,所以在集群启动后,首次执行任务可能存在"正在排 队"的提示,这是因为集群冷启动中正在拉起资源。如果您频繁提交任务,建议购买包年包月集群,该类型集群不 存在冷启动,能在任何时间快速执行任务。



UDF 函数开发指南

最近更新时间: 2025-04-29 15:18:12

UDF 说明

用户可通过编写 UDF 函数,打包为 JAR 文件后,在数据湖计算定义为函数在查询分析中使用。目前数据湖计算 DLC 的 UDF 为 HIVE 格式,继承 org.apache.hadoop.hive.ql.exec.UDF,实现 evaluate 方法。 示例:简单数组 UDF 函数。

```
public class MyDiff extends UDF {
    public ArrayList<Integer> evaluate(ArrayList<Integer> input) {
        ArrayList<Integer> result = new ArrayList<Integer>();
        result.add(0, 0);
        for (int i = 1; i < input.size(); i++) {
            result.add(i, input.get(i) - input.get(i - 1));
            }
            return result;
        }
    }
}</pre>
```

pom 文件参考:

<dependencies></dependencies>
<dependency></dependency>
<groupid>org.slf4j</groupid>
<artifactid>slf4j-log4j12</artifactid>
<version>1.7.16</version>
<scope>test</scope>
<dependency></dependency>
<groupid>org.apache.hive</groupid>
<artifactid>hive-exec</artifactid>
<pre><version>1.2.1</version></pre>

创建函数

<u>小 注意:</u>



理。

如您创建的是 udaf/udtf 函数,需要在函数名相应加上 _udaf/_udtf 后缀。

若您了解 SQL 语法,可通过**数据探索**执行 CREATE FUNCTION 语法完成函数创建,或通过可视化界面创建, 流程如下:

- 1. 登录 数据湖计算控制台 ,选择服务地域。
- 2. 通过左侧导航菜单进入**数据管理**,选择需要创建的函数的数据库,如果需要创建新的数据库,可参见 数据库管

数据管理	◎ 广州 ▼					使用指引 🗊 📑	数据管理使用指南 🖸
数据目录	数据库 存储桶列表						任务历史
 管理数据 ビ 	目录下的数据库,可以点击数据库名称	,管理数据库下的表、视图等数据对象,	手动导入数据到数据表。 了解更多 	数据操作需要相应的数据权限,	更多权限操作指导	川,请参考 用户权限管理	指南 X
创建数据库	DataLakeCatalog •				请输	入名称搜索	Q Ø
数据库名称 💲	创建时间	✿ 描述信息	ſ	创建人	操作		
_test I	2023-03-2	1 10:05:48			编辑 删除		
ifig I	2023-02-2	0 11:11:15		• •	编辑 删除		

- 3. 单击函数进入函数管理页面。
- 4. 单击创建函数进行创建。

හි	腾讯云 介 怠敗 │ 云产品 →	授素产品、文档 Q ② 小程序 🖸 集团账号 × 备案 工具 × 客服支持 × 费用 ×
٢	← 数据库 /	创建函数
	数据表 视图 函数	函数名称• raw_dff
© ≣	③ 数据库下的Spark自定义函数,支持将JAR包上传到DLC,或者挂在用户的COS存储上,建议使用上传到DLC方式,	描述信息 送填
33	创建函数 更新时间 全部 近7天 近30天 选择日期 🛗	
Ē	函数名称 创建时间 \$ 更新时间 描述信息	存插方式 上传到系统保存 / 指定COS位置挂载
\$		函数包的存储方式,可以选择将函数包上传到系统并保存(推荐),或者直接指定函数包保存的对象存储COS路径。
50 101		程序包来源 本地上传 O 数据存储COS
-	共 2 条	文件路径* cosn:/ 1305424723/demo-1.0-SNAPSHOT.jar 选择COS位置 程序包只支持JAR类型,文件最大支持100M
		函数类名 * com.tencent.dlc.MyDiff

UDF 的程序包支持本地上传或选择 COS 路径(需具备 COS 相关权限),示例为选择 COS 路径创建。 函数类名包含"包信息"及"函数的执行类名"。

函数使用



- 1. 登录 数据湖计算控制台,选择服务地域。
- 2. 通过左侧导航菜单进入数据探索,选择计算引擎后即可使用 SQL 调用函数(注意:请选择独享引擎,共享引擎 暂不支持自定义 udf 函数使用)。

D jž	新 協 保存 の 刷新 協 格式化 いいのう しんしょう ほうしょう しょうしょう ひょうしょう ひょうしょう ひょうしょう しょう ひょう しょう しょう しょう しょう しょう しょう しょう しょう しょう し	🛢 t
35 Se	elect raw_diff(Array[1,2,3])	
37		
查询结果		
Task ID : 查询耗时	SQL计值 导出结果 优化建议 L2 5.21s	
共1条数	居(控制台最多可展示1000条数据)复制数据 🗗	
test_ca	che.raw_diff(array(1, 2, 3))	
[0,1,1]		

Python UDF 开发

如需使用 Python UDF 功能,请将集群升级到最新版本,您可以 提交工单 咨询我们当前集群是否支持。 1. 购买集群,请选择 SuperSQL-S 1.0 版本(暂支持该版本)集群。



☆ 購讯云 進	购其他云产品 →								Q &	───
数据湖计	算 DLC _题								产品文档 🗹	计费说明 记 产品控制台 已
引擎版本	SuperSQL引擎	Beta 标准引擎								
计费模式	按量计费 按CU量计费。没有任务时可将	包年包月 記集業、結記时不产生行	详细对比 ☑ 何费用。话合有一定任务量	但任务周期不规律的数据计	算话号					
地域		区	——华南地区——— 广州	南 京		地区	上海自动驾驶云	西南地	1区	美国西部 硅谷
		美国东部	欧洲地区	港澳台地区	2.17	مر <u>ا</u> مد ر - ر حد		799 MP		
	处于不同地域的云产品内网不	互通,购买后不能更换,	请您谨慎选择。建议选择最高	都近您客户的地域,可降低订	前问时延。					
集群配置										
基本配置 计算引擎类型	SparkSQL	Spark作业	Presto							Ж
内核版本	支持实时提交sql任务执行,词 Beta SunarSOL-S 3.5	師于在数据量较大的情况	下使用,如:ETL数据清洗,	, 引擎特点: 稳定性较高。						系 销 售
	SuperSQL-S 1.0是基于Spark	3.2自研的引擎内核,适	用于离线SQL任务。不同版本	*兼容的语法略有不同,更多	5.版本信息参见: 内核版本说 I	A 12				4

- 2. 开启集群 Python UDF 功能
- 默认情况下, Python UDF 功能未开,请在 SuperSQL 引擎页面,选择需要开启的引擎,单击参数配置,加 入配置。输入 spark.sql.pyudf.enabled = true 后,单击确认,进行保存。
- SparkSQL 集群待集群重启后生效。
- SparkBatch 集群下一个任务生效。



配置变更	×
① 修改引擎参数配置将需要重启集群.	
数据加密 ③ ①	
参数配置 1 spark.sql.pyudf.enabled true - +添加	
确认 取消	
3. 注册 Python UDF 函数	
• DLC 支持完全社区兼容的 ODF 函数。 示例,请编辑如下文件 demo.py :	
<pre>def name_sensitivity(name):</pre>	
<pre>if len(name) > 2: return name[0] + '*' * (len(name) - 2) + name[-1]</pre>	
elif len(name) == 2:	
return '*' + name[-1] else:	
return name	

- 目前支持单 Python 文件。
- 只支持 import 系统自带模块。
- 请注意函数输入、输出类型。
- 请注意处理函数异常。

📃 💩 腾讯云		L Q 搜索资源、产品、API、文档	快捷戰/ 集团账号 备案 工具 客服支持 费用• ② 〔1999〕 🗔
数据湖计算	数据管理		创建函数 ×
器 概 览	数据目录 数据库 存储桶列表 函	<u>8</u>	函数名称 • namesensitivity
 ○ 数据探索 Ⅲ 数据调度 	创建函数方式:支持通过python文件、JAR包上传到DL 函数生效范围:Python文件创建函数支持datalakecata	C或挂在用户的COS存储上。为更好的方便程序包的集中管理,建议通过上 log目录下全局有效,JAR包等其他方式创建函数针对库级别生效。	描述信息 送頃
	创建函数 更新时间 全部 近7天	近30天 选择日期 选择日期 首 数据库 请注	
 致娠TF亚 历史任务 	函数名称 函数类型	生效范围(库级别) 创建时间 \$ 更新时间	存能方式 ○上传到系统保存 ○指定COS位置挂载 函数包的存储方式,可以选择将函数包上传到系统并保存(推荐),或者直接指定函数包保存的对象存储COS器径。
引擎管理	aaab 🗗	ey_tpcds_10_parquet 2024-04-18 16:11:43 2024-04	程序包来源 〇本地上传 〇 数据存储COS
5 SuperSQL引擎	carry_type I	idanli_test 2023-11-09 11:56:09 2023-11-	文件路径・ demo.py 重新上传 割除 程序向支持JAR_python学型。JAR文件書大支持5M_Python文件書大支持2M。
⑦ 标准引擎 (BETA)	carry_type_2	rdanli_test 2023-11-09 12:07:28 2023-11-	生效范围• 全局生效 🔻
」「「「「「「」」」」「「」」」「「」」」「「」」」「「」」「「」」」「「」」」「」」「」」」」	mapagg_udaf 🍯	rensli_test43 2023-11-03 16:22:04 2023-11-	函数包名 • demo.name_sensit
o 权限管理	test_fun T	_ory 2023-08-22 18:08:01 2023-09	
🖬 存储配置	fy_encrypt T	ay_test 2023-08-16 20:00:13 2024-05	
 前 审计日志 二 监控告警 C 	my_test2 F	rie_test 2023-08-01 14:42:42 2023-08	
	my_test I	ie_test 2023-08-01 14:41:55 2023-08	
	mytest	rie_test 2023-08-01 14:38:26 2024-05	
	test4 T	2023-04-21 10:53:19 2024-05	
三 给产品打个分 🕥	共 26 条		ane Roll

4. 在数据探索界面,使用开启了支持 Python UDF 的集群,并设置 Session 参数 eos.sql.processType=DIRECT 如果您不能编辑该参数,请 提交工单 联系我们。



数排	降 MySQL Q 搜索资源、产品、API、文档…	快捷键 / 集团账号 备案 工具	客服支持 费用 • 🕜 🗘 🤒 99 •	₽
			SQL语法参考	⑦ 数据探索使用指南 2
	查询-2023-0 ^{草稿} x + ▼			✿ 存储配置
•			edemo2 🔻 🔚 pyspark-udf 🔻	⊱ SuperSQL语法 ▼ │ •••
·	1 SELECT namesensitivity(name) FROM demo2.udf_demo_table	世 数据引擎	刷新	Bar maning wer With a d (a) (b)
		pyspark-udf	SuperSQL-Spark 🔻	
Ĭ		该引擎支持SuperSQL语法的查询, 查看语法说	明区。	
		分 引擎 (内核版本) 不同内核兼容不同SQL语	去,详情可参见内核版本说明	
		spark (SuperSQL-S 1.0)	v	
		① 创建引擎		
		高级设置 ▲	配置说明 记	
		1 eos.sql.processType	▼ DIRECT -	
		+选择配置 更多 ▼		
	查询结果		运行历史	下载历史
		=		
		请运行一个SOL任务		
		LA THE LAND		

- 如果您需要编辑变更 Python UDF 函数, SparkSQL 集群加载机制约有30s延迟, SparkBatch 集群下一个 任务立即生效。
- 如果您需要删除当前 Python UDF 函数,SparkSQL 集群需要重启集群,SparkBatch 集群下一个任务立即生效。

Python 函数编辑权限配置

说明:
 在数据湖 DLC 控制台 > 元数据管理 > 函数,点击函数名称进入函数详情页,仅支持查看,不支持修改。

当通过 Python 创建函数时,允许创建者及管理员对函数进行编辑权限配置。流程如下:

入口一: 创建函数时进行编辑权限配置



- 1. 登录数据湖 DLC 控制台,选择服务地域。
- 2. 左侧菜单栏进入元数据管理,选择函数页面,点击函数栏左上方蓝色按钮"创建函数",进入页面。
- 点击 "函数编辑权限" 右侧 "展开",创建者可选择用户或工作组(可组合选择)进行对函数 "编辑"和 "删除"的授权,创建者和管理员默认拥有全部权限。
- 4. 设置完成后,点击"确定",即可完成函数编辑权限配置。

入口二: 对已有函数进行编辑权限配置

- 1. 登录数据湖 DLC 控制台,选择服务地域。
- 左侧菜单栏进入 元数据管理,选择函数页面,选择需要编辑的函数右侧操作栏的编辑按钮,即可进入编辑页面。
 进入编辑函数页面后,点击"函数编辑权限"右侧"展开",可以进行权限的修改与删除。



物化视图

最近更新时间: 2023-06-20 14:58:41

△ 注意:

目前数据湖计算 DLC 物化视图只支持 SparkSQL 引擎和 Presto 引擎。

物化视图(Materialized View)是数据库中的一种特殊对象,它是一个预先计算和存储的查询结果集。物化视图 在处理大量数据和复杂查询时可以提供快速的查询性能。

物化视图提高查询性能的同时也引入了存储成本和计算成本。我们建议您在以下场景使用物化视图:

- 源表变更不频繁
- 相比于源表,物化视图表的字段和结果数量有明显的减少

DLC 支持普通物化视图和映射物化视图,以下是介绍和完整的使用示例,支持的语法列表可以参考物化视图语法。

普通物化视图

普通物化视图的基本使用流程包括创建、刷新、使用。 以下基于 Presto 引擎操作举例完整流程。

准备数据

执行 SQ L创建库表,并插入数据。以下语句创建了一个名为 student 的表。

```
CREATE DATABASE IF NOT EXISTS mv_test3;
create table student(id int, name string, score int);
insert into student values (1,'zhangsan', 90);
insert into student values (2,'lisi', 100);
insert into student values (3,'wangwu', 80);
insert into student values (4,'zhaoliu', 30);
select * from student order by id;
```

创建普通物化视图

使用 CREATE MATERIALIZED VIEW 语句来创建物化视图。指定物化视图的名称和查询语句,可以选择性地指定 查询的来源表和条件。

以下例子,使用了一个简单的 SELECT 语句从表 student 中选择所有分数,并对它们进行求和操作。然后将这 个求和结果作为物化视图 mv_student_sum 的内容。

```
CREATE MATERIALIZED VIEW mv_student_sum AS (
select sum(score) from student
```



);

查看物化视图详情

使用 DESCRIBE MATERIALIZED VIEW 语句来查看物化视图的详细信息,包括名称、查询语句和刷新状态等。

DESCRIBE MATERIALIZED VIEW mv_student_sum;

	(
查询结果		运行历史 下	载历史	~ ~	
expr_0	bigint				
Materializ	edView Detail:				
state	NORMAL				
mvType	SINGLE				
viewOrigin	alText SELECT SUM(`score`)				
FROM `stude	ent`				
autoRewrite	e ENABLE				
dataLatest	FRESH			Ξ	
freshType	RUNTIME			-	
updateType	FULL				
createTime	Sat May 06 19:57:52 CST 2023				
modifiedTi	ne Sat May 06 19:58:00 CST 2023				

手动刷新物化视图

使用 REFRESH MATERIALIZED VIEW 语句来手动刷新物化视图的数据。

此处仅作演示,大部分情况下,您并不需要手动刷新物化视图,只要 SQL 命中了源表有变更的物化视图就会自动刷 新。

REFRESH MATERIALIZED VIEW mv_student_sum;

查看物化视图的执行任务列表

使用 SHOW MATERIALIZED VIEW JOBS 语句来查看物化视图的执行任务列表,可以了解到物化视图的刷新历史和 状态。

SHOW MATERIALIZED VIEW JOBS IN mv_student_sum;



查询结果						运行历史下载历史 🕴 🔺 🗸
Task ID SQL详情 导出结果 查询耗时 490 ms 数据扫描 共2条数据(控制台最多可展 1000000000000000000000000000000000000	优化建议 🖸 🖬 0 B 示1000条数据)复制数据 🔂					
xid	taskld	state	buildType	execEngine	createTime	updateTime
19b61c931fa646159ea	95284167ec0511ed9ae	FINISHED	MANUAL_REFRESH	PRESTO	2023-05-06 20:00:27	2023-05-06 20:00:37
6bb622691e68471b909	328b5149ec0511ed9ae	FINISHED	CREATE	PRESTO	2023-05-06 19:57:47	2023-05-06 19:58:00

SQL 改写执行

使用 SELECT 语句查询数据,期望自动改写并命中物化视图。可以通过查询结果里的统计数据,查看是否自动改写 到了物化视图上。

select sum(score) from student;

← 查询结果	统计数据			运行历史 下载历史	Q
数据扫描量		自适应shuffle (i)	缓存加速 ①	物化视图加速 ①	2
798 [₿]		shuffle字节数 溢出到COS	ResultCache 缓存命中 合 FragmentCache命中率 Alluxio 缓存命中	目动改写到 DataLakeCatalog.mv_test4.mv_student 利用数据 446B	
■ 预处理: 2.86s			■ 执行: 8.51s	■ 获取结果: 192.00ms	-

删除物化视图

DROP MATERIALIZED VIEW mv_student_sum;

映射物化视图

映射物化视图是一种特殊类型的物化视图,它与现有的表进行映射关联。通过映射物化视图,可以将物化视图的查询 结果与现有表的数据进行关联,从而实现对现有表的查询性能优化。

限制

物化视图相对于普通物化视图有以下限制:

- 映射物化视图不支持刷新操作,即无法通过REFRESH MATERIALIZED VIEW语句来刷新物化视图的数据。
 因此,物化视图的数据只能与映射表的数据保持一致,无法自动更新。
- 映射物化视图不进行自动SQL改写,即查询语句不会自动转换为使用物化视图。需要手动指定使用物化视图的查询语句。
- 删除映射物化视图时,只会删除与映射表的关联关系,而不会删除映射表本身。映射表仍然存在,可以继续使用。

推荐场景

腾讯云

推荐您在以下场景使用映射物化视图:

- 当已经存在一个数据量较大的表,并且该表的查询性能较低时,可以通过映射物化视图来优化查询性能。
- 当需要保持物化视图的数据与现有表的数据保持一致,并且不需要自动刷新物化视图时,可以使用映射物化视图。

Iceberg 类型的源表

Iceberg 表为源表时,完整示例如下:

基于 CTAS 创建映射物化视图

映射物化视图需要与待映射的表保持名称一致。以下例子先基于CTAS创建表,用于映射MV的创建。数据的准备可 以参考普通物化视图中完整示例中的数据准备一节。

```
CREATE TABLE link_mv_student AS (
   select sum(score) from student
);
--创建映射物化视图: 使用CREATE MATERIALIZED VIEW语句创建映射物化视图。
--在创建物化视图时,使用WITH META LINK子句,并指定映射表的名称作为关联。
CREATE MATERIALIZED VIEW link_mv_student WITH META LINK AS (
   select sum(score) from student
);
```

查看映射物化视图

使用 DESCRIBE MATERIALIZED VIEW 语句可以查看映射物化视图的详细信息,包括名称、查询语句和刷新 状态等。

```
DESCRIBE MATERIALIZED VIEW link_mv_student;
SHOW MATERIALIZED VIEW JOBS IN link_mv_student;
```

映射物化视图不支持刷新操作



映射物化视图不支持 REFRESH 操作,即无法通过 REFRESH MATERIALIZED VIEW 语句来刷新物化视图 的数据。因此,物化视图的数据只能与映射表的数据保持一致,无法自动更新。

SQL 改写

映射物化视图不会自动对查询语句进行 SQL 改写。 如执行 select sum(score) from student; 不会命中映射物化视图。 可以通过使用 Hint 或 TaskConf 参数来指定允许基于映射物化视图进行 SQL 改写。

--**手动指定需要改写**SQL select /*+ OPTIONS('eos.sql.materializedView.enableRewrite'='true') */ sum(score) from student;

删除映射物化视图

使用 DROP MATERIALIZED VIEW 语句来删除映射物化视图。删除映射物化视图后,仅会删除与映射表的关联 关系,映射表本身仍然存在。

```
DROP MATERIALIZED VIEW link_mv_student;
DESCRIBE link_mv_student; --可查看源表还存在
```

Hive 类型的源表

Hive 表为源表时,完整示例如下:

准备初始化数据

首先,需要准备初始化数据并创建Hive基表。使用 CREATE EXTERNAL TABLE 语句创建 Hive 基表,并通 过 INSERT 语句手动插入数据。

```
CREATE EXTERNAL TABLE student_2(id int, name string, score int)
LOCATION 'cosn://guangzhou-test-1305424723/mv_test4/student_2';
insert into student_2 values (1,'zhangsan', 90);
insert into student_2 values (2,'lisi', 100);
insert into student_2 values (3,'wangwu', 80);
insert into student_2 values (4,'zhaoliu', 30);
select * from student 2;
```

创建被映射的 Hive 外表

使用 CREATE EXTERNAL TABLE 语句创建一个被映射的 Hive 外表。

CREATE EXTERNAL TABLE link_mv_student_hive (





--**向映射表插入数据** INSERT OVERWRITE link_mv_student_hive select sum(score) from student;

基于 Hive 外表创建映射物化视图

使用 CREATE MATERIALIZED VIEW 语句创建映射物化视图。在创建物化视图时,使用 WITH META LINK 子句,并指定上述 Hive 外表的名称作为关联。

```
CREATE MATERIALIZED VIEW link_mv_student_hive WITH META LINK AS (
    select sum(score) from student_2
);
```

系统约束 元数据信息

最近更新时间: 2023-12-27 20:28:21

数据库、数据表、属性列、分区的个数。

项目	最大数量
每个账户的数据库数量	1,000
每个账户的数据表数量	10,000
每个数据库的表数	4,096
每个数据表的列数	4,096
每个表的分区数	100,000
每个主账户的分区数	1,000,000
每张表字段数量上限	4096
每个账户的自定义函数数量	100
可创建 catalog 数量	20

数据库

- 名称: 最多不超过127个字符,同一个数据链接下,不允许有相同的数据库名称。
- 描述: 最多不超过2048 个字符。
- 外表的数据地址(COS 地址): 888个字符(COS 路径长度限制)。
- 参数: Map<string:string> 的形式,每个参数的长度限制为127个字符,总长度限制为3000个字符。

数据表/视图

- 名称:最多不超过127个字符,同一个数据库下,不允许有相同的数据表名称。
- 描述: 最多不超过1000个字符。
- 外表的数据地址(COS 地址): 最多不超过888个字符(COS 路径长度限制)。
- 参数: Map<string:string> 的形式,每个参数的长度限制为127个字符,总长度限制为512000个字符。

属性列

• 名称: 最多不超过127个字符,同一个数据表下,不允许有相同的属性列名称。

🔗 腾讯云

- 描述: 最多不超过256个字符。
- 类型:最多不超过131072个字符。若超过将无法创建。

分区

• 分区字段名称:最多不超过127个字符。



计算任务

最近更新时间: 2022-05-23 17:15:01

单条 SQL 语句大小限制为2MB。