# 数据测试功能(数据沙箱)

数据测试功能,可生效在离线开发和自助分析两个子产品中,主要解决的是数据开发过程中的线上数据和开发数据 隔离问题。

# 背景

当前线上调度任务中运行着如下的代码,代码中的表intern\_new.dim\_user\_info\_p被下游所使用,此时需要对该表 进行逻辑修改或增加字段等操作。代码示例如下:

```
insert overwrite table intern_new.dim_user_info_p
partition (ds='2021-07-25')
select id as user_id,
name as user_name,
province,
age
from intern_new.ods_user_info
where ds='2021-07-25';
```

通常来说会有两种方式进行线上表的修改:

- 第一种,开发人员选择新建一个临时表,并复制出现有代码,先进行测试验证。当验证通过时,再将该表更新,并更新代码,然后将任务提交上线。
- 第二种,开发人员选择直接修改该表,并修改原有代码,修改完成后,将任务提交上线。

对于上述两种方法,第一种方法更可靠,能保证所有操作不影响线上数据,也不影响下游,但是会多一些额外的工作;第二种方法可以节约额外的创建临时表,以及代码反复拷贝修改的时间,但是特别容易出纰漏,影响线上数据。

为了更好解决上述问题,数开平台支持**数据测试**功能。数据测试功能引入了一种新的代码规则,可以实现同一份代码根据运行环境不同,自动进行部分参数替换,实现操作离线表或测试表的功能,以及操作默认HDFS文件或测试HDFS集群文件的功能。当前可生效在离线开发和自助分析两个子产品中,解决了数据开发过程中的线上数据和开发数据隔离问题。

# 功能介绍

### 针对Hive库

现将上述案例引入数据测试功能后,对库名intern\_new进行修改为\${intern\_new},代码如下:

insert overwrite table \${intern\_new}.dim\_user\_info\_p --修改点是intern\_new 变成了 \${intern\_new}
partition (ds='2021-07-25')
select id as user\_id,
name as user\_name,
province,
age
from intern\_new.ods\_user\_info
where ds='2021-07-25';

在离线开发的**开发模式**下以及自助分析的**测试模式**下,运行上述sql,系统会自动将上述代码进行转换,并且使用测试Yarn集群的资源。自动转换后的代码如下:

```
insert overwrite table intern_new_dev.dim_user_info_p
partition (ds='2021-07-25')
select id as user_id,
name as user_name,
province,
age
from intern_new.ods_user_info
where ds='2021-07-25';
```

此处,将**\${intern\_new}**作为参数去匹配当前项目集群下intern\_new库所对应的测试库intern\_new\_dev,从而实现 操作测试库下的表。

如果上述代码,提交上线并设置定时调度,在线上调度、重跑、补数据运行时,系统会自动将上述代码进行转换, 并且使用默认Yarn集群的资源(即项目的默认Yarn队列)。自动转换后的代码如下:

```
insert overwrite table intern_new.dim_user_info_p
partition (ds='2021-07-25')
select id as user_id,
name as user_name,
province,
age
from intern_new.ods_user_info
where ds='2021-07-25';
```

这里将\${intern\_new}作为参数去匹配当前项目集群下的普通库,即intern\_new。

**注意:** 要使用上述自动库替换功能,需要先确保当前项目集群下要操作的普通库已经存在对应的测试库,并且 有待测试的表结构的同名表。对应测试库,可由项目的管理员进行申请,并进行授权。

#### 针对HDFS

离线开发中,支持使用Script、MR等节点,用户可在节点中自行指定hdfs相对路径。

当开启数据测试功能,数据的HDFS存储路径也会根据运行环境自动转换:

- 当在离线开发的开发模式运行会自动在存储路径前补充测试HDFS集群的前缀。
- 当在离线开发的线上模式进行任务调度,或对实例进行重跑、补数据等操作时,会自动在存储路径前补充默认
   HDFS集群的前缀。

上述例子中,表的存储路径如下:

- 表intern\_new\_dev.dim\_user\_info\_p的HDFS存储路径为: hdfs://cluster1/user/intern/hive\_db/intern\_new.
   db/dim\_user\_info\_p。
- 表intern\_new.dim\_user\_info\_p的HDFS存储路径为: hdfs://dev4/user/intern/hive\_db/intern\_new.db/dim\_ user\_info\_p。

**说明:**对于测试表,默认location的测试HDFS集群是**cluster1**,这个是测试HDFS集群。对于离线表,默认loc ation的默认HDFS集群是**dev4**,这个是默认HDFS集群。而两个路径的后半部分地址相同,都是**user/intern/hi ve\_db/intern\_new.db/dim\_user\_info\_p**。

# 功能实现

使用数据测试功能需要当前平台具备生产集群和测试集群,并安装了相关服务,具体要求可联系平台的运维人员。

# 数据测试功能开启

目前,有3种方式来开启数据测试功能。

- 方式一:当申请新建项目时,可同时申请开通测试功能。此时,进行新建项目配置过程中还需要进行测试功能
   区块的配置。
  - ~ 测试功能

测试功能					
测试存储主路径	/user/*				
测试存储配额*	1 T				
测试Hive库 *	测试库名 ?			路径	
	请先填写项目名称				
测试队列所在集群	测试Yarn集群				
测试队列名称	请先填写队列名称				
测试资源配置*	默认策略				
	方案	CPU配额	内存配额	最大并行任务数	
	○ 方案1	100 核	300 G	20 个	
	○ 方案2	300 核	900 G	20 个	
	○ 方案3	500 核	1500 G	20 个	
	○ 自定义	10	核 10	G 20	个

#### 重要参数说明:

参数 名称	说明
测试 存储 配额	是后续测试库表可存放测试数据的空间大小,取决于当前平台的测试HDFS集群的空间,具体可联 系平台运维人员确定。
测试 Hive 库	一个新建的项目默认会按照项目名称创建一个同名的Hive离线库。当开通测试功能后,系统也会 自动创建一个对应的测试库。比如项目名叫 <b>my_dwd</b> ,则测试Hive库就叫 <b>my_dwd_dev</b> 。

测试	
盗源	一个新建的项目需要指定一个Yam队列的名称,当开通了测试功能后,则会往测试集群上自动创
	建一个同名的Yarn队列。比如默认Yarn队列名叫 <b>my_queue</b> ,则测试Yarn队列也叫 <b>my_queue</b> 。
距直	

- 方式二:对于已有的项目,申请在新集群开通时,可同时申请开通测试功能。需要配置的内容基本和方式一相同。
- 方式三: 支持在项目中, 对已存在的集群申请开通测试功能。

资源管理页面,在具备开通测试功能的集群上,单击**申请开通测试功能**按钮,在如下界面进行配置:

申请开通新集群					
dev4 查看详情	注册Yam队列	注册Impala队列 注册Hive库 申请开通测;	武功能		
Yam	队列  个	CPU配额 化 核	内存配额    G		
Impala	队列 🦳 个	CPU配额 🧧 核	内存配额 G	存储配数 TB	Hive库 个

#### 重要参数说明:

参数 名称	说明
测试 存储 配额	是后续测试库表可存放测试数据的空间大小,取决于当前平台的测试HDFS集群的空间,具体可联系 平台运维人员确定。
Hive 库选 择	在申请开通时,有需要选择一个已有离线库作为当前项目-集群的主库。系统会在申请通过后,自动创 建一个名为 <b>{主库}_dev</b> 的测试库。
测试 资源 配置	对于已有项目,会存在一些Yarn队列。在申请开通测试功能后,系统会自动在测试Yarn集群上新建相 同数量的同名Yarn队列。后续管理员可在 <b>项目中心</b> 的 <b>资源管理</b> 中,申请修改测试队列的资源。

# 典型场景

案例背景:当前生产环境中存在一个ods表和dim维表,dim维表由ods表加工得到。现在需要给ods表增加一个字段,因此dim维表也需要进行同步更新。在整个过程中,使用了两次数据测试功能,并使用了离线开发的**离线表新** 增字段和表克隆功能。

### 场景1: 基于ods表加工dim维表

#### 1) ods表的DDL语句

假设ods表为一张用户信息表:

```
CREATE TABLE `intern_new_dev`.`ods_user_info`(
  `id` string COMMENT 'id',
  `name` string COMMENT '姓名',
  `province` string COMMENT '省份',
  `age` string COMMENT '年龄'
  )
COMMENT '用户信息表'
PARTITIONED BY (
  `ds` string COMMENT '时间分区')
ROW FORMAT SERDE
  'org.apache.hadoop.hive.ql.io.parquet.serde.ParquetHiveSerDe'
STORED AS INPUTFORMAT
  'org.apache.hadoop.hive.ql.io.parquet.MapredParquetInputFormat'
OUTPUTFORMAT
  'org.apache.hadoop.hive.ql.io.parquet.MapredParquetOutputFormat';
```

#### 2) 在测试库设计表

离线库为intern\_new,测试库为intern\_new\_dev,设计的dim维表DDL如下:

```
CREATE TABLE `intern_new_dev`.`dim_user_info_p`(
  `user_id` string COMMENT 'id',
  `user_name` string COMMENT '姓名',
  `province` string COMMENT '省份',
  `age` string COMMENT '年龄'
  )
COMMENT '用户全量维表'
PARTITIONED BY (
  `ds` string COMMENT '时间分区')
ROW FORMAT SERDE
  'org.apache.hadoop.hive.ql.io.parquet.serde.ParquetHiveSerDe'
STORED AS INPUTFORMAT
  'org.apache.hadoop.hive.ql.io.parquet.MapredParquetInputFormat'
OUTPUTFORMAT
  'org.apache.hadoop.hive.ql.io.parquet.MapredParquetOutputFormat';
```

#### 3) 在离线开发新建任务,并拖入SQL节点

SQL节点的代码如下(其中\${azkaban.flow.1.days.ago}是azkaban调度系统的系统参数,表示昨天):

```
insert overwrite table ${intern_new}.dim_user_info_p
partition (ds='${azkaban.flow.1.days.ago}')
select id as user_id,
name as user_name,
province,
age
from intern_new.ods_user_info
where ds='${azkaban.flow.1.days.ago}';
```

在**开发模式**下,运行SQL节点。在**运行设置**弹框中,保持**测试模式**开关打开。运行后,数据实际插入到的是intern\_ new\_dev.dim\_user\_info\_p中。确认测试库的表的数据没有问题后,可进行下一步。

#### 4) 通过表克隆功能将测试库表克隆到离线库

通过单击<mark>辅助功能区</mark>的图标 —,选择**表克隆**。在**表克隆**的弹框中,设置待克隆表为intern\_new\_dev.dim\_user\_inf o\_p,目标库为intern\_new。

表结构克隆						×
表选择 > 克隆	操作设置					
表选择	【 待克降	隆表		┃目标₪	Ę	
	库*	intern_new_dev	•	库*	intern_new	<b>▼</b> 徒半
	表*	dim_user_info_p	•			
	克隆检验	۵				
	1、目标库)	不存在待克隆同名表;				
	2、有目标 结论:可克	库建表权限; 3隆				

#### 克隆检验通过后,则可进入下一步,系统会根据源表内容自动生成建表语句:

表结构克隆						×
表选择 > <b>克隆</b>	操作设置					
建表语句	1 CI 2	REATE TABLE `intern_new.dim_user_in	fo_p` (			
	3	user_id string COMMEN	τ 'id',			
	4	user_name string COMM	ENT '姓名',			
	5	province string COMME	NT '省份',			
	6	age string COMMENT '全	F齿☆ '			
	7	) PARTITIONED BY (ds st io.parquet.serde.Parquet MapredParquetInputFormat	ring COMMENT ' HiveSerDe' STO ' OUTPUTFORMAT	时间分区') ROW FORMAT RED AS INPUTFORMAT 'o 'org.apache.hadoop.h	SERDE 'org.apache. rg.apache.hadoop.hi ive.ql.io.parquet	nadoop.hive.ql ve.ql.io.parquet
	.1	MapredParquetOutputForma	t'; 🦟			
					修华	行 取消

#### 5) 将任务提交上线并编辑调度

最后,将任务提交上线,编辑调度。在线上调度的任务,实际执行时,就会执行如下代码,将数据插入到"intern\_ new.dim user info p"表中:

```
insert overwrite table intern_new.dim_user_info_p
partition (ds='${azkaban.flow.1.days.ago}')
select id as user_id,
name as user_name,
province,
age
from intern_new.ods_user_info
where ds='${azkaban.flow.1.days.ago}';
```

## 场景2: dim维表增加字段

1) ods源头表增加了性别字段

ods表新的ddl如下:

```
CREATE TABLE `intern_new_dev`.`ods_user_info`(
  `id` string COMMENT 'id',
  `name` string COMMENT '姓名',
  `province` string COMMENT '省份',
  `age` string COMMENT '年龄',
  `sex` string COMMENT '性别' --新增的字段
  )
COMMENT '用户信息表'
PARTITIONED BY (
  `ds` string COMMENT '时间分区')
ROW FORMAT SERDE
  'org.apache.hadoop.hive.ql.io.parquet.serde.ParquetHiveSerDe'
STORED AS INPUTFORMAT
  'org.apache.hadoop.hive.ql.io.parquet.MapredParquetInputFormat'
OUTPUTFORMAT
  'org.apache.hadoop.hive.ql.io.parquet.MapredParquetOutputFormat';
```

#### 2) 对测试dim维表表新增字段

使用离线开发的离线表新增字段功能,给测试维表intern\_new\_dev.dim\_user\_info\_p增加一个性别字段。

通过单击辅助功能区的图标 — ,选择离线表新增字段。

如下图,为新增性别字段:

ाञ्चेय	0-1-E	== 3	211	ඩර ප	<u>-</u>	≤л
困	纥	衣法	ொ	<u> </u>	f i	汉

数据库*	intern_new_dev		• 徐华		
表名称*	dim_user_info_p		•		
级联更新	• ?				
表描述	用户全量维表				
表负责人					
现有表结构	字段名	字段类型		字段描述	
	user_id	string		id	
	user_name	string		姓名	
	province	string		省份	
	age	string		年齢	
	字段名	字段类型	字段描	說述	
	sex	STRING	▼ 性類	9	删除 上移 下移
	新增字段				
	> 分区字段				

#### 3) 修改原SQL节点代码

新的SQL节点代码如下:

```
insert overwrite table ${intern_new}.dim_user_info_p
partition (ds='${azkaban.flow.1.days.ago}')
select id as user_id,
name as user_name,
province,
age,
sex --为新增加的代码
from intern_new.ods_user_info
where ds='${azkaban.flow.1.days.ago}';
```

修改后,在**开发模式**下运行SQL节点。在运行设置弹框中,保持测试模式开关打开。运行后,数据实际插入到的是intern\_new\_dev.dim\_user\_info\_p中。确认测试库的表的数据没有问题后,可进行下一步。

#### 4) 通过表克隆功能将测试库表更新到离线库

х

在**表克隆**的弹框中,设置**待克隆表**为intern\_new\_dev.dim\_user\_info\_p,目标库为intern\_new。经过**克隆检验**, 发现待克隆表比目标库的表多一个字段,可由待克隆表更新表结构至目标表。

表结构克隆									×
表选择 > 克隆操	作设置								
表选择	┃待克隆表				日日	标库			
	库* in 表* di	tern_new_dev m_user_info_p	徐华	•	<i>→</i>	≣* intern_	new		徐华
1 1 1 1	<mark>克隆检验</mark> 1、目标库存在征 2、有目标表修行 结论:可由待克	待克隆同名表,巨 攻权限; 2隆表更新表结构3	可由待克隆表 全目标表	更新;					
表结构差异	待克隆表:	intern_new_d	ev.dim_use	r_info_p	目标表: inf	tern_new.d	im_user_in ⇒m#∺*	fo_p	
	1	user_id	string	id	user_id	string	id		徐华
	2	user_name	string	姓名	user_name	string	姓名		
	3	province	string	省份	province	string	省份		
	4 5	age sex	string	牛般	age	-	<b>年</b> 紀 後4年	<ul> <li>✓</li> <li>✓</li> </ul>	
	#	分区字段	字段类型	字段描述	分区字段	字段类型	字段描述	比对结果	1?
	1	ds	string	时间分区	ds	string	时间分区	1	
							徐华	下一步	取消

在表结构克隆界面查看建表语句,如下图所示,图中建表语句实际是对intern\_new.dim\_user\_info\_p表执行加字段 操作。

表结构克隆					×
表选择 > 克隆	操作设计	置			
建表语句	1	ALTER TABLE			
	2	`intern_new.dim_use	er_info_p`		
	3	ADD			
	4	COLUMNS (sex string	; COMMENT '性别');		
	5				
				执	行取消

#### 5) 将任务提交上线

最后,将任务提交上线。在线上调度的任务,实际执行时,就会执行如下代码,将数据插入到intern\_new.dim\_use r\_info\_p表中:

```
insert overwrite table intern_new.dim_user_info_p
partition (ds='${azkaban.flow.1.days.ago}')
select id as user_id,
name as user_name,
province,
age,
sex
from intern_new.ods_user_info
where ds='${azkaban.flow.1.days.ago}';
```