

**自定义工作流(CTSDB to TBDS)**

**示例说明**

**文档版本变更记录：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **版本号** | **\*变化状态** | **变更说明** | **变更人** | **日  期** |
| 1 | C | 文档新建 | 胡习远 | 2018/12/25 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**\*修改状态：C - 创建，A - 增加，M - 修改，D - 删除**

**目 录**

[1. 概述 5](#_Toc533529445)

[2. 环境搭建与代码配置说明 5](#_Toc533529446)

[2.1. 环境搭建 5](#_Toc533529447)

[2.1.1. 加入runner依赖的jar包 5](#_Toc533529448)

[2.1.2. 加入hadoop依赖的jar包 5](#_Toc533529449)

[2.1.3. 加入JSON相关依赖 6](#_Toc533529450)

[3. 自定义Runner代码设计 7](#_Toc533529451)

[3.1. 继承的基类 7](#_Toc533529452)

[3.2. 必须实现的抽象方法 7](#_Toc533529453)

[3.3. 启动任务流程 7](#_Toc533529454)

[3.4. 获取参数的方法 7](#_Toc533529455)

[3.5. 日志输出方法 8](#_Toc533529456)

[3.6. 提交任务实例执行状态 8](#_Toc533529457)

[3.7. 停止任务实例运行 9](#_Toc533529458)

[3.8. 运行示例 9](#_Toc533529459)

[4. 自定义工作流配置说明 9](#_Toc533529460)

[4.1. 基本信息配置 10](#_Toc533529461)

[4.2. 参数配置 10](#_Toc533529462)

[4.3. CTSDB TO HDFS 参数配置说明 11](#_Toc533529463)

[5. 示例使用方法 14](#_Toc533529464)

[5.1. 创建工作流 14](#_Toc533529465)

[5.2. 工作流参数配置 14](#_Toc533529466)

[5.3. 工作流任务调度 15](#_Toc533529467)

[5.3.1. 工作流任务运行 15](#_Toc533529468)

[5.3.2. 查看运行结果 15](#_Toc533529469)

1. 概述

工作流模块作为腾讯大数据套件（TBDS）最重要的功能之一，本文档提供了一种基于自定义工作流的实现，将CTSDB的表的数据导入TBDS的HDFS说明。CTSDB的相关API说明参考：https://cloud.tencent.com/document/product/652/13611。

1. 环境搭建与代码配置说明

此部分为基于java语言开发的自定义工作流的示例（CTSDB to HDFS），完整的代码参见：

https://github.com/TBDSUDC/TBDSDemo/tree/master/src/main/java/com/tencent/taskrunner。

* 1. 环境搭建

### 加入runner依赖的jar包

自定义runner任务依赖 common-core-0.0.1-SNAPSHOT.jar，pom.xml配置如下：

<dependency>

<groupId>com.tencent.teg.dc.runner</groupId>

<artifactId>runner-common</artifactId>

<version>0.0.1-SNAPSHOT</version>

<scope>system</scope>

<systemPath>${project.basedir}/lib/common-core-0.0.1-SNAPSHOT.jar</systemPath>

</dependency>

### 加入hadoop依赖的jar包

由于需要将CTSDB表的数据导入hdfs，需要加入hadoop的相关依赖及权限认证：

在pom.xml文件加入相关依赖，如下：

<dependency>

<groupId>org.apache.hadoop</groupId>

<artifactId>hadoop-client</artifactId>

<version>2.7.2-TBDS-4.0.3.3</version>

</dependency>

<dependency>

<groupId>org.apache.hadoop</groupId>

<artifactId>hadoop-common</artifactId>

<version>2.7.2-TBDS-4.0.3.3</version>

</dependency>

<dependency>

<groupId>org.apache.hadoop</groupId>

<artifactId>hadoop-hdfs</artifactId>

<version>2.7.2-TBDS-4.0.3.3</version>

</dependency>

在程序代码里面把四个参数添加到conf里面，如下：

conf.addResource(new Path("/etc/hadoop/conf/core-site.xml"));

conf.addResource(new Path("/etc/hadoop/conf/hdfs-site.xml"));

//参数 tbds\_username，tbds\_username，tbds\_securekey通过工作流自定义参数传入

conf.set("hadoop.security.authentication", "tbds");

conf.set("hadoop\_security\_authentication\_tbds\_username",tbds\_username);

conf.set("hadoop\_security\_authentication\_tbds\_secureid",tbds\_secureid);

conf.set("hadoop\_security\_authentication\_tbds\_securekey",tbds\_securekey);

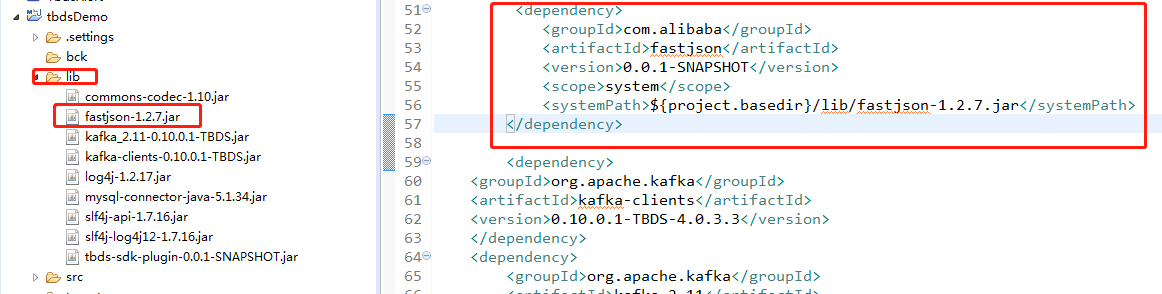
UserGroupInformation.setConfiguration( conf );

UserGroupInformation.loginUserFromSubject(null);

### 加入JSON相关依赖

加入解析时序数据库CTSDB查询数据的JSON依赖包，fastjson-1.2.7.jar，

pom.xml如下图：



1. 自定义Runner代码设计
   1. 继承的基类
2. 与hive相关的实现可以继承AbstractTDWDDCTaskRunner这个基类，继承其中对数据库表的debug。
3. HDFS与传统数据库导入的runner可以继承HDFSToDBRunner或者DBToHdfsRunner，调用DBUtil类来调用JDBC执行DAO
4. 普通的runner直接继承jar包中的AbstractTaskRunner基类就可以了。
   1. 必须实现的抽象方法

重载AbstractTaskRunner的execute()方法和kill（）方法。

* 1. 启动任务流程

1. 新建main 函数 public static void main(String[] args) {} 并在main 函数中实例化runner对象,将args[0] 作为实例化对象的传入参数。
2. 在main函数中调用对象startwork()方法，启动自定义runner。
   1. 获取参数的方法
3. 通过this.getTask()方法获取TaskRuntimeInfo对象。
4. 通过TaskRuntimeInfo对象，获取获取用户填写参数

2.1 获取实例和任务信息

  task.getId() 获取实例id  
  task.getType() 获取任务类型  
  task.getCurRunDate() 获取实例数据时间  
  task.getNextRunDate() 获取实例下一个数据时间

2.2 获取用户在ui上填写的填写任务参数

  this.getProperties().get(key) //获取任务参数。

2.3 获取用户在ui上选择的源服务器和目标服务器信息，获取方式如下

List<ServerRuntime> sourceServers = taskRuntime.getSourceServers();

List<ServerRuntime> targetServers = taskRuntime.getTargetServers();

虽然返回值是一个list队列，但目前只支持一个源服务器和一个目标服务器，服务器对象 ServerRuntime 结构如下,通过get\*() 方法获取对应的值：

private String tag; // 服务器标示符

private String type;// 服务器类型 支持hive,hdfs,ftp,postgreSQL,mysql,sql server,oracle

private String host;// 服务器地址(对应主机地址)

private int port;// 服务器连接端口(对应端口)

private String userName;// 连接用户名

private String password;// 连接密码

private String service;// 当前不用

private String userGroup;// 当前不用

private String version;// 当前不用

* 1. 日志输出方法

this.writeLocalLog(Level.INFO, "\*\*");

* 1. 提交任务实例执行状态

this.commitTask(state, runtimeId, desc);

Ps:常用运行状态有：RUNNING（正在运行），KILLED（已经停止），SUCCESSFUL（成功），FAILED（失败）：

* 1. 停止任务实例运行

1. 停止实例运行前，清理runner资源，并提交停止状态。
2. 我们实现了runner调度资源的清理方，建议开发者，使用统一的停止方法，如下：
3. public void kill() throws IOException {
4. this.writeLocalLog(Level.INFO, " hello word had been kill ");
5. boolean killResult = false;
6. try {
7. killResult = CommonUtils.killProcess(this.taskRuntime, this);
8. if (killResult) {
9. this.writeLocalLog(Level.SEVERE, "kill job succeed!");
10. this.commitTask(LState.KILLED, "", "kill job succeed!");
11. } else {
12. this.writeLocalLog(Level.SEVERE, "kill job failed!");
13. this.commitTask(LState.HANGED, "", "kill job failed!");
14. }
15. } catch (Exception e) {
16. this.writeLocalLog(Level.SEVERE,
17. "kill job failed:" + CommonUtils.stackTraceToString(e));
18. this.commitTask(LState.HANGED, "", "kill job failed!");
19. }
20. }
    1. 运行示例

https://github.com/TBDSUDC/TBDSDemo/tree/master/src/main/java/com/tencent/taskrunner

1. 自定义工作流配置说明

配置路径：工作流模块->任务管理，如下图：



点击“新建”按钮。

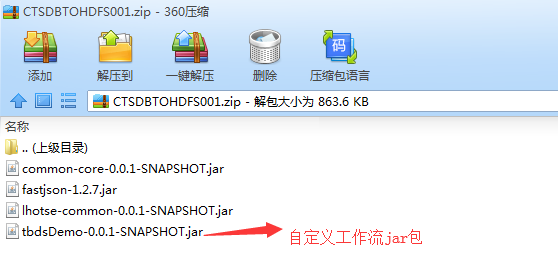
* 1. 基本信息配置

如下图：



参数说明：（1）任务类型名：任务类型的名称标识；（2）上传zip包：上传需要执行的zip文件，所有用到文件全部打包为zip文件（可以将相关依赖包打进去），不要打包目录，打包成文件；（3）任务执行命令：执行自定义工作流jar包的启动脚本；（4）任务类型标签：标签说明，如：CTSDB、HDFS等，程序不做校验；（5）任务类型描述：相关描述信息。

（2）zip包说明：



* 1. 参数配置

如下图：（可以根据自己需求，添加多个属性）



参数说明：（1）属性名：英文名称，后台获取相应参数名称；（2）属性别名：添加工作流前台展示属性名称（后台不校验）；（3）属性类型：选择“基本属性”，“高级属性”点击没变化，目前未使用；（4）设置属性正则验证：未使用，可随意填写；（5）正则验证失败提示：未使用，可随意填写。

其它字段根据自己的需求选择填写。

* 1. CTSDB TO HDFS 参数配置说明

##tbds权限认证参数配置

tbds\_username=usertest

tbds\_secureid=xxxx

tbds\_securekey=xxxx

##CTSDB数据库访问用户名

ctsdb.user=root

##CTSDB数据库访问密码

ctsdb.password=tbds1234xx

##CTSDB数据库访问地址

ctsdb.url=http://10.0.\*.\*:9200

##CTSDB数据导入HDFS存储数据的临时目录，提前创建好，修改创建目录权限为777

ctsdb.dataInput=/tmp/kafkatest/ctsdb/

##CTSDB数据导入HDFS存储的目录，需要提前创建好（自己所属项目的路径下创建）

ctsdb.hdfsOutput= /project/tbds\_test/ctsdb/output/

##CTSDB需要导入的数据表名称（导入HDFS）

ctsdb.tableName=ctsdb\_tbds

##CTSDB需要导入数据表的列名称

ctsdb.colNames=0001,0002

#n\*60000 一个批次查询60000条，设置多少个批次生成一个文件

#设置成5，表示5\*600000条数据生成一个文件

ctsdb.fileDataSum=5

#test 测试用，运行多少个批次查询，示例采集26\*60000条数据

ctsdb.runBatchSum=26

#10m 数据10分钟后会过期，过期后 scroll\_id将变得无效，后续查询会失败

ctsdb.validDataTime=10m

配置完成后，如下图：



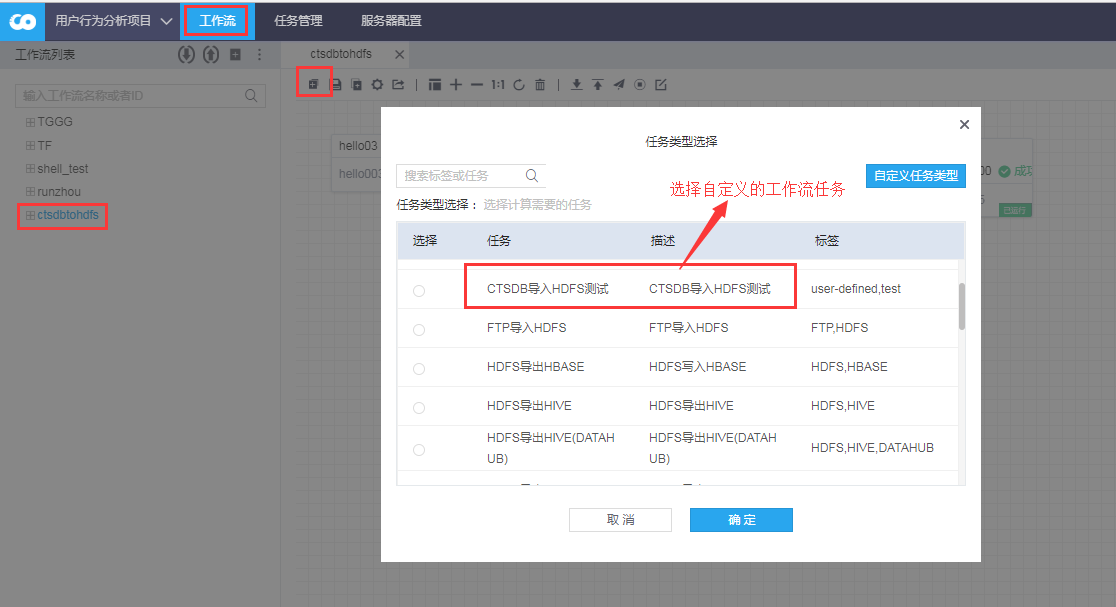






1. 示例使用方法
   1. 创建工作流

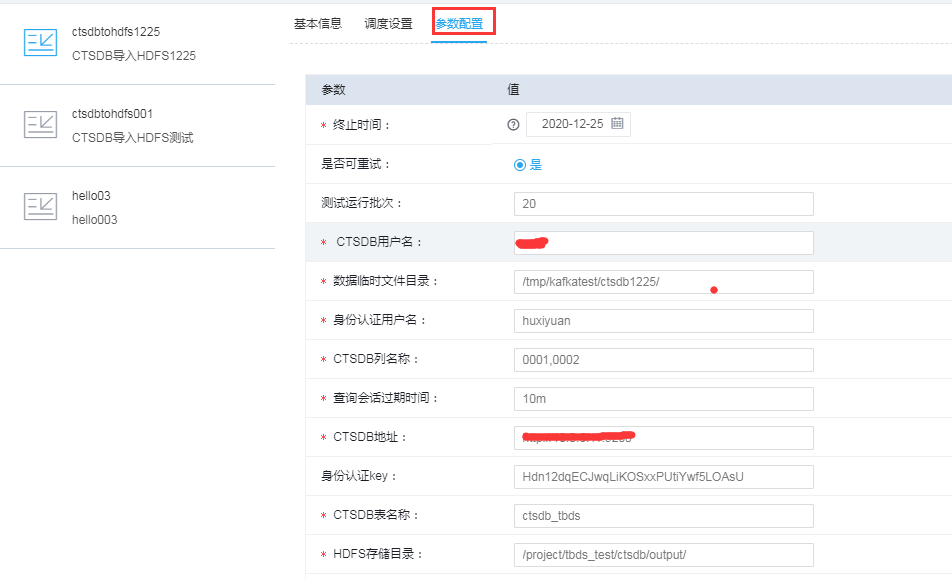
选择自定义的工作流任务，创建工作流，如下图：



* 1. 工作流参数配置

具体配置参数说明，见文档4.3。

配置完成后如下图：



* 1. 工作流任务调度

### 工作流任务运行

配置工作流完成后，提交审核，审核通过后，向TBDS的工作流调度系统提交任务进行调度执行。

### 查看运行结果

直接在运维中心-文件管理-基础文件管理，找到目标目录查看文件如下：

