BI 文件入库工具设计方案

(仅供内部使用）

For internal use only

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 拟制:  Prepared by | 张政勇 |  | 日期：  Date | 2014-06-27 |
| 审核:  Reviewed by |  |  | 日期：  Date | yyyy-mm-dd |
| 批准:  Granted by |  |  | 日期：  Date | yyyy-mm-dd |



华为技术有限公司

Huawei Technologies Co., Ltd.

版权所有 侵权必究

All rights reserved

修订记录Revision record·

|  |  |  |  |
| --- | --- | --- | --- |
| 日期  Date | 修订版本Revision version | 修改描述  change Description | 作者  Author |
| 2014-06-27 | V0.1 | 初稿 | 张政勇 |
| 2014-10-30 | V0.2 | 增强特性  \* 增加文件统计日志  \* 增加文件列数校验  \* 增加支持lzo,gz,bz压缩格式的文件  \* 增加超过等待时间检查文件标示  \* 增加非时间分区的支持  \* 增加是否入空文件标示 |  |
|  |  |  |  |

[1 概述 4](#_Toc391737797)

[1.1 背景 4](#_Toc391737798)

[1.2 系统上下文 4](#_Toc391737799)

[2 系统设计 5](#_Toc391737800)

[2.1 总体方案 5](#_Toc391737801)

[2.2 环境配置 5](#_Toc391737802)

[2.2.1 工具包结构定义 5](#_Toc391737803)

[2.2.2 环境配置信息 6](#_Toc391737804)

[2.2.3 命令行参数 8](#_Toc391737805)

[2.3 子任务设计 8](#_Toc391737806)

[2.3.1 文件入库到HDFS子任务 8](#_Toc391737807)

[2.3.2 HDFS数据导入MYSQL子任务 12](#_Toc391737808)

# 概述

## 背景

BI 报表服务主要由数据采集、后台的数据处理以及前台的报表展示三个部分组成，其中后台数据处理部分使用了两个入库工具：1. talend入库工具（文件入库到HDFS）、2.HIBI-EXPORT工具，但在在实际开发中遇到如下两个问题

* + - 1. talend open studio开发效率不高
      2. 各业务开发模版不统一，而且每个任务都需要维护一个工程；
      3. 入库配置修改复杂，每次修改都需要打开talend修改；
      4. 对于ODS入库任务需要在TCC单独配置入库参数，TCC逻辑复杂，成为”上帝类”；
      5. 入库脚本分散配置不集中，使用了1. talend入库工具（文件入库到HDFS）、2.HIBI-EXPORT工具 多个工具完成，TCC配置脚本不一致；
      6. 扩展性差，将来有可能其他方式的入库，还需要再开发工具。

为了更好的规范业务开发，提高业务开发效率，现将如上两个工具统一规划成BI文件入库工具，以能达到以下目的：

* + - 1. 工具归一化，入口统一、模版统一
      2. 仅需简单配置，业务便能快速完成入库脚本的开发
      3. 解决ODS入库TCC配置”上帝类”的问题
      4. 增加扩展性，对后续入库形式变化方便扩展(例如增加EXCEL到HIVE的入库)

## 系统上下文

BI文件入库工具是基于Talend开发的实现方案，在整个BI 报表服务中的位置如下图所示：



1. BI文件入库工具的位置

如图示， BI文件入库工具提供统一的入库业务开发环境，支持BI业务的开发。当推送工具将业务数据推到到MFS文件系统后，通过BI文件入库工具入库到HADOOP计算环境的HDFS文件系统，经过数据计算后，再次使用BI文件入库工具将计算后数据入库到MYSQL数据库，供报表展示使用。

# 总体方案

BI文件入库工具是基于Talend开发，将文件入库需求统一规划到该工具中。本期提供以下三个子任务的：

* 文件入库到HDFS子任务：
* 约束限制

目前对文件输入列表、文件数和结束标示的配置是在TCC中配置的，TCC对每个文件进行入库脚本的调度，中间的调度规则是TCC实现的，造成TCC除了单独任务调度作业以外，还维护了任务规则。

* 实现方案

为了满足组件功能单一职能需求，避免”上帝”组件，对TCC关于ODS入库文件逻辑处理移入入库工具中执行，来解放TCC的逻辑处理，并且将入库脚本的开发进行模板化，由talend open studio开发模版架构代码，业务仅需做简单的配置文件修改，即可完成入库脚本的开发。

其中配置文件包含老组件的两部分：1.TCC关于入库文件的配置部分 2.原入库组件每次业务修改部分。

* HDFS数据导入MYSQL子任务：
* 约束限制

HIBI-EXPORT工具属于自研工具，主要过程是将HIVE中的表批量导出到MYSQL中数据库中。

* 实现方案

为了实现入库脚本的统一入库，将HIBI-EXPORT使用talend open studio重新开发模版化，配置文件融合入工具的统一配置文件中完成归一化。

组件支持两种数据导入方式：

* 单表全表导出
* 单表按分区导出
* 单表按条件导出
* 导出列可选
* HDFS数据导入文件子任务
* 约束限制

HIBI-EXPORT工具属于自研工具，主要过程是将HIVE中的表批量导出到文件中，以供推送工具做数据源推送。

* 实现方案

为了实现入库脚本的统一入库，将HIBI-EXPORT使用talend open studio重新开发模版化，配置文件融合入工具的统一配置文件中完成归一化。

组件支持两种数据导入方式：

* 单表全表导出
* 单表按分区导出
* 单表按条件导出
* 导出列可选
* 导出文件分隔符可配置

# 环境配置

## 工具包结构定义

BI文件入库工具包结构如下所示：

DataLoad

|-----------appconf

|--------------vmall

|----------VMallProd\_Conf.xml

|-----------DataLoad\_0.1

|-----------logs

|-----------work

|-----------DataLoad.sh

|-----------EnvConf.xml

|-----------log4j.properties

目录说明如下：

* appconf: 任务中应用配置信息，每个业务一个配置文件，配置格式说明参照子任务设计中的描述。
* DataLoad\_0.1: 存放对应的jar包、执行壳等程序，由talend studio直接生成。
* log : 存放任务执行过程中的日志信息
* work: 存放工作过程中的数据，如任务执行过程中生成的各种临时文件等。
* DataLoad.sh 任务启动脚本
* EnvConf.xml: 任务的环境配置信息，内容参考下节描述。
* log4j.properties 日志配置文件

## 环境配置信息

BI文件入库工具通常部署在数据通道服务器，一台机器所有用户共用一套环境，通常需要把环境配置文件把公共信息提取出来。

BI文件入库工具环境相关的信息包括以下内容：

* 数据库连接信息：一个任务可能涉及到多个数据库连接。对每个连接，一般包括
  + 数据库类型
  + IP地址信息
  + 数据库名称
  + 数据库用户名
  + 数据库用户密码（AES加密后信息）
  + 数据库连接其它附加信息
* 文件配置信息：每个业务的文件环境配置信息
  + hive/hdfs存储目录
* TCC 服务器信息，包括
  + TCC服务器所在的IP地址
  + 接收告警信息的端口
* 应用配置路径，包括
  + 业务配置文件相对路径

配置文件示例如下：

<Conf>

<!--

数据目标数据库，支持myql

可多个，以id唯一标示，业务根据需要自行配置

-->

<!-- 数据源数据库 Hive -->

<DBSource id="Hive">

<Type>Hive</Type>

<!-- 如果为空会取去本地IP -->

<Host>10.36.65.242</Host>

<Port>10000</Port>

<Database>default</Database>

<Username>hadoop</Username>

<!-- AES加密 -->

<Password>7BE216A80761D3875576BFDD422A3D65</Password>

<AdditionalInfo/>

</DBSource>

<!-- 数据目标数据库 Myql -->

<DBSource id="Mysql\_FR">

<Type>mysql</Type>

<Host>192.168.8.246</Host>

<Port>3306</Port>

<Database>test</Database>

<Username>root</Username>

<!-- AES加密 -->

<Password>6EC0F2176B23266F3F998D7C46BB5BE4</Password>

<AdditionalInfo/>

</DBSource>

<!--

文件配置信息

可多个，以id唯一标示，业务根据需要自行配置

-->

<Fileconf id="VMallProd">

<!-- hive/hdfs存储目录 -->

<Hadoopdir>/AppData/VMallProd/VMall/data/</Hadoopdir>

</Fileconf>

<!--

TCC监听上报告警事件用，预留，功能未实现

-->

<TCC>

<Host>10.120.5.45</Host>

<Port>8888</Port>

</TCC>

<AppConf>

<!-- 业务应用配置信息路径 -->

<include file="appconf/vmall/VMallProd\_Conf.xml"/>

</AppConf>

</Conf>

## 命令行参数

通用的命令行命令Dataload.sh

通用参数：

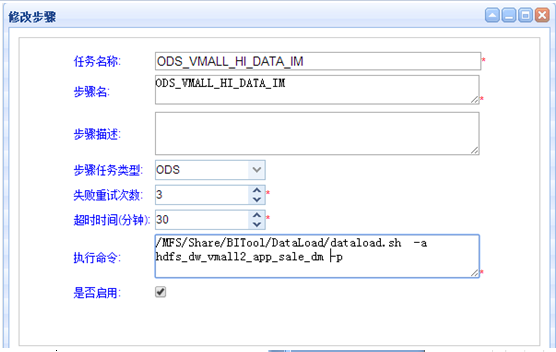
-a 任务标识 对应下章节降到的应用标示action字段

-p 任务周期信息 周期开始日期+周期结束日期+周期类型+周期长度，该参数如果是tcc调度，自动传入

例子：Dataload.sh -a hdfs\_dw\_vmall2\_app\_sale\_dm -p ‘20140628-0000 20140628-0000 D 1’

## TCC调度方式

对比以前的TCC配置，只需要修改执行命令处：



# 子任务设计

## 共通组件

直接基于talend open studio开发子任务时，由于各子任务间有共同的某些操作，为了规范开发，实现子任务间代码复用，现将共同使用的某些操作封装成模版以便复用，这些模块包含一下：

* 环境和应用配置信息读取组件
* TCC通信组件
* 业务告警组件
* 数据库数据抽取组件

下几小节将会对各个模块详细介绍。

### 环境和应用配置信息读取组件

功能：该组件完成环境应用配置信息读取，将EnvConf.xml和各业务的AppConf.xml中的配置信息读取到java结构中，供各子任务使用。

输入内容：

* 配置文件名称 - 配置文件名称及目录位置固定，业务无需更改该参数

输出内容：

* java结构定义的环境配置信息



### HIVE数据库数据抽取组件

功能：完成HIVE中的表数据（全表数据、分区数据、按指定条件的数据）的提取操作，供后续导入其他数据库或者文件使用。

输入内容：

* + 数据库类型
  + IP地址信息
  + 数据库名称
  + 数据库用户名
  + 数据库用户密码（AES加密后信息）
  + 数据库连接其它附加信息
  + 表名
  + 分区信息
  + 抽取SQL语句

输出内容：

抽取的数据集ResultSet

实现说明：

首先会根据表名、分区进行抽取条件的拼装，分区和条件的映射规则

pt\_d：pt\_d >= ${startDate} and pt\_d <= ${endDate}

pt\_w：pt\_w = ${start-endDate}

pt\_m：pt\_m = ${startMonth}

如果表名、分区为空，则直接使用业务定义的抽取SQL语句，并进行环境变量的替换构造出最后抽取条件，然后执行jdbc的连接进行数据抽取操作。

### TCC通信组件

功能：支持将业务执行过程的状态和告警信息发送给TCC，支持用户在TCC上实时监控任务状态。 向TCC发送消息时，在组件内部必须自动填写任务标识信息，方便TCC根据任务标识将信息与具体的任务（包括周期）关联起来。

（未实现）

输入：消息类型 – 状态通知、告警信息、错误信息。

消息体 - 由业务填写的消息体内容。

任务与TCC 之间通过UDP通信，所能支持的报文大小受UDP报文大小限制。报文格式如下：



1. TCC通信报文定义

## 文件入库到HDFS子任务

* 功能：将DATAPUSH推送到MFS文件系统的文件入库到HDFS文件系统。
* 应用配置信息的信息包括以下内容：
  + 任务唯一标识
  + 文件配置信息 对应环境配置信息中的文件配置信息id
  + 输入文件列表
  + 输入文件个数
  + 空文件是否入库
  + 等待输入时间（单位：分钟）：

1. 任务会在等待时间内不断处理输入文件，一直到输入个数达到设置的个数，如果超过等待时间仍未到达指定个数，则进行告警；

2. TCC任务的配置超时时间要大于这里的配置，否则TCC会认为任务超时。

* 超过等待时间是否检查文件个数
* 文件列数
* 文件字段的分割符
  + 入库表名
  + 表存储位置
  + 分区信息 none,pt\_y,pt\_m,pt\_d,pt\_h,pt\_min

None:会每次都全量覆盖，不保留以前的数据

多分区情况请用,隔开 例如：pt\_y,pt\_m 会自动按年和月分区 pt\_y=2014/pt\_m=08/

配置文件示例如下：

<!-- 文件入库到HDFS: action唯一标示一个Job ，全局唯一-->

<FileToHDFS action="filetohdfs\_tmp\_zzy\_test1\_7">

<!-- 文件配置信息 -->

<Fileconf>VMallProd</Fileconf>

<!-- 输入文件配置-->

<MultiInputFileConf>

<!-- 输入文件列表-->

<InputFileList>/data1/z00215119/temp/testfiletohdfs/data/tmp\_zzy\_test1\_7/DATA\_IM\_\*.log</InputFileList>

<!-- 输入文件个数 -->

<InputFileMinCount>1</InputFileMinCount>

<!-- 空文件是否入库 默认true -->

<DumpFileLoad>true</DumpFileLoad>

<!-- 等待输入时间（分钟） -->

<WaitInputMinutes>15</WaitInputMinutes>

<!-- 超过等待时间是否检查文件个数，默认true -->

<TimeOutCheckFile>true</TimeOutCheckFile>

<!-- 文件列数 <=0表示不校验 -->

<FileColumns>2</FileColumns>

<!-- 文件字段的分割符，默认使用\001 -->

<Separator>,</Separator>

</MultiInputFileConf>

<!-- 表名 -->

<Tablename>tmp\_zzy\_test1\_7</Tablename>

<!-- 表存储位置 -->

<Tablelocation>warehouse</Tablelocation>

<!-- 分区 none pt\_y pt\_m pt\_d pt\_h pt\_min -->

<Partition>pt\_d</Partition>

</FileToHDFS>

* 输出
* 返回码 成功或失败错误码

实现说明：

将执行单文件入库子模块化，所以流程图分成两部分描述

* + - 1. 文件入库到HDFS总览
      2. 单一文件入库模块



1. 文件入库到HDFS子任务流程图



1. 单一文件入库流程图

## HDFS数据导入MYSQL子任务

* 功能：将HIVE中的表数据（全表数据、分区数据、按指定条件的数据）入库到MYSQL的报表数据库中，并能在入库出错记录数达到最大门限时终止加载并告警。
* 输入：
  + 任务唯一标识
  + 源数据库，对应环境配置信息中的数据库id
  + 目标数据库，对应环境配置信息中的数据库id
  + 表名
  + 分区信息，未配置分区则全表导入，分区和条件的映射规则

pt\_d：pt\_d >= ${startDate} and pt\_d <= ${endDate}

pt\_w：pt\_w = ${start-endDate}

pt\_m：pt\_m = ${startMonth}

* 抽取SQL语句 为了增加抽取数据的灵活性，该参数和表名分区互斥

内置7个时间变量，入库工具会将其替换为实际值

${startDate}: 周期开始年月日，格式yyyyMMdd

${endDate} ： 周期结束年月日，格式yyyyMMdd

${startWeek} ： 周期开始年月日的第一个星期，格式yyyyMMdd-yyyyMMdd

${startMonth} ： 周期开始年月，格式yyyyMM

${startHour}: 周期开始小时，格式hh

${startMinute}: 周期开始分钟，格式mm

${start-endDate} ：周期开始结束年月日，格式yyyyMMdd-yyyyMMdd

* 表别名 ： mysql中对应的表名，如果hive和mysql对应的表名相同就不用配置

注意：ExtractSql不为空时，AliasTablename也不能为空

* 最大允许出错记录数： 当出错记录数到达该门槛时，终止加载并告警
* 最大出错记录百分比： 当出错记录百分比到达该门槛时，终止加载并告警

配置文件示例如下：

<!-- HDFS导出到MYSQL: action唯一标示一个Job ，全局唯一-->

<HDFSToDB action="hdfstodb\_tmp\_zzy\_test1\_7">

<!-- 源数据库标识，环境配置文件中获取 -->

<SourceDB>Hive</SourceDB>

<!-- 目标数据标识，环境配置文件中获取 -->

<DestDB>Mysql\_FR</DestDB>

<!-- 抽取表名 表名和ExtractSql 必须有一个不是空 -->

<Tablename>tmp\_zzy\_test1\_7</Tablename>

<!-- 抽取分区，默认规则如下

pt\_d：pt\_d >= ${startDate} and pt\_d <= ${endDate}

pt\_w：pt\_w = ${start-endDate}

pt\_m：pt\_m = ${startMonth}

-->

<Partition>pt\_d</Partition>

<!-- 提取表的HIVE 数据提取SQL语句，此配置项目和表名、分区互斥，如果均配置以表名、分区优先

内置7个时间变量，入库工具会将其替换为实际值

${startDate}: 周期开始年月日，格式yyyyMMdd

${endDate} ： 周期结束年月日，格式yyyyMMdd

${startWeek} ： 周期开始年月日的第一个星期，格式yyyyMMdd-yyyyMMdd

${startMonth} ： 周期开始年月，格式yyyyMM

${startHour}: 周期开始小时，格式hh

${startMinute}: 周期开始分钟，格式mm

${start-endDate} ：周期开始结束年月日，格式yyyyMMdd-yyyyMMdd

-->

<ExtractSql>

<![CDATA[

SELECT dev\_id FROM tmp\_zzy\_test1\_7 WHERE pt\_d=${startDate} or pt\_d=${endDate}

]]>

</ExtractSql>

<!-- mysql中对应的表名，如果hive和mysql对应的表名相同就不用配置

ExtractSql不为空时，AliasTablename也不能为空 -->

<AliasTablename>tmp\_zzy\_test1\_7</AliasTablename>

<!-- 当出错记录数到达该门槛时，终止加载并告警，当为-1时表示不限制出错记录数 -->

<MaxErrorNumber>4</MaxErrorNumber>

<!-- 当出错记录百分比到达该门槛时，终止加载并告警，当为-1时表示不限制出错百分比-->

<MaxPercentNumber>33</MaxPercentNumber>

</HDFSToDB>

* 输出
* 返回码 成功或失败错误码

实现说明：



1. HDFS数据导入MYSQL子任务流程

## HDFS数据导入文件子任务

* 功能：将HIVE中的表数据（全表数据、分区数据、按指定条件的数据）导出到文件中。
* 输入：
  + 任务唯一标识
  + 源数据库，对应环境配置信息中的数据库id
  + 目标文件目录
  + 导出文件字段的分割符，默认使用\001
  + 表名
  + 导出的文件名
  + 分区信息，未配置分区则全表导入，分区和条件的映射规则

pt\_d：pt\_d >= ${startDate} and pt\_d <= ${endDate}

pt\_w：pt\_w = ${start-endDate}

pt\_m：pt\_m = ${startMonth}

* 抽取SQL语句 为了增加抽取数据的灵活性，该参数和表名分区互斥

内置7个时间变量，入库工具会将其替换为实际值

${startDate}: 周期开始年月日，格式yyyyMMdd

${endDate} ： 周期结束年月日，格式yyyyMMdd

${startWeek} ： 周期开始年月日的第一个星期，格式yyyyMMdd-yyyyMMdd

${startMonth} ： 周期开始年月，格式yyyyMM

${startHour}: 周期开始小时，格式hh

${startMinute}: 周期开始分钟，格式mm

${start-endDate} ：周期开始结束年月日，格式yyyyMMdd-yyyyMMdd

配置文件示例如下：

<!-- HDFS导出到文件: action唯一标示一个Job ，全局唯一-->

<HDFSToFile action="hdfstofile\_tmp\_zzy\_test1\_7">

<!-- 源数据库标识，环境配置文件中获取 -->

<SourceDB>Hive</SourceDB>

<!-- 目标文件目录 -->

<DestFilePath>/data1/z00215119/temp/testfiletohdfs/exportdata</DestFilePath>

<!-- 导出文件字段的分割符，默认使用\001 -->

<Separator>,</Separator>

<!-- 抽取表名 -->

<Tablename>tmp\_zzy\_test1\_7</Tablename>

<!-- 抽取分区，默认规则如下

pt\_d：pt\_d >= ${startDate} and pt\_d <= ${endDate}

pt\_w：pt\_w = ${start-endDate}

pt\_m：pt\_m = ${startMonth}

-->

<Partition>pt\_d</Partition>

<!-- 提取表的HIVE 数据提取SQL语句，此配置项目和表名、分区互斥，如果均配置以表名、分区优先

内置7个时间变量，入库工具会将其替换为实际值

${startDate}: 周期开始时间，格式yyyyMMdd

${endDate} ： 周期结束时间，格式yyyyMMdd

${startWeek} ： 周期开始年月日的第一个星期，格式yyyyMMdd-yyyyMMdd

${startMonth} ： 周期结束时间，格式yyyyMM

${startHour}: 周期开始时间，格式hh

${startMinute}: 周期开始时间，格式mm

${start-endDate} ：周期结束时间，格式yyyyMMdd-yyyyMMdd

-->

<ExtractSql>

<![CDATA[

SELECT dev\_id , pt\_d FROM tmp\_zzy\_test1\_7 WHERE pt\_d=${startWeek}

]]>

</ExtractSql>

<!-- 导出的文件名，如果导出文件名与和hive对应的表名相同就不用配置 -->

<AliasFilename>tmp\_zzy\_test1\_7\_2</AliasFilename>

</HDFSToFile>

* 输出
* 返回码 成功或失败错误码

实现说明：

