日志

目录

[Overview 6](#_Toc495908953)

[2017 7](#_Toc495908954)

[Week\_41 7](#_Toc495908955)

[2017-10-09 8](#_Toc495908956)

[2017-10-10 8](#_Toc495908957)

[2017-10-11 9](#_Toc495908958)

[2017-10-12 9](#_Toc495908959)

[2017-10-13 9](#_Toc495908960)

[2017-10-14 9](#_Toc495908961)

[Week\_39 10](#_Toc495908962)

[2017-09-25 10](#_Toc495908963)

[2017-09-26 10](#_Toc495908964)

[2017-09-27 10](#_Toc495908965)

[2017-09-28 10](#_Toc495908966)

[2017-09-29 11](#_Toc495908967)

[2017-09-30 11](#_Toc495908968)

[Week\_38 12](#_Toc495908969)

[Plan 12](#_Toc495908970)

[Require 12](#_Toc495908971)

[2017-09-18 12](#_Toc495908972)

[2017-09-19 12](#_Toc495908973)

[2017-09-20 12](#_Toc495908974)

[2017-09-21 12](#_Toc495908975)

[2017-09-22 13](#_Toc495908976)

[2017-09-23 13](#_Toc495908977)

[2017-09-24 13](#_Toc495908978)

[Week\_37 13](#_Toc495908979)

[Plan 13](#_Toc495908980)

[Require 13](#_Toc495908981)

[2017-09-11 14](#_Toc495908982)

[2017-09-12 14](#_Toc495908983)

[2017-09-13 14](#_Toc495908984)

[2017-09-14 15](#_Toc495908985)

[2017-09-15 15](#_Toc495908986)

[2017-09-16 15](#_Toc495908987)

[2017-09-17 15](#_Toc495908988)

[Week\_36 16](#_Toc495908989)

[Plan 16](#_Toc495908990)

[Require 16](#_Toc495908991)

[2017-09-04 16](#_Toc495908992)

[2017-09-05 16](#_Toc495908993)

[2017-09-06 16](#_Toc495908994)

[2017-09-07 17](#_Toc495908995)

[2017-09-08 18](#_Toc495908996)

[2017-09-09 18](#_Toc495908997)

[2017-09-10 18](#_Toc495908998)

[Week\_37 19](#_Toc495908999)

[引用 21](#_Toc495909000)

[附录 22](#_Toc495909001)

[0000-年模板 23](#_Toc495909002)

[Week\_00-周模板 23](#_Toc495909003)

[Plan 23](#_Toc495909004)

[Require 23](#_Toc495909005)

[0000-01-01 23](#_Toc495909006)

[0000-01-02 23](#_Toc495909007)

[0000-01-03 23](#_Toc495909008)

[0000-01-04 23](#_Toc495909009)

[0000-01-05 23](#_Toc495909010)

[0000-01-06 23](#_Toc495909011)

[0000-01-07 24](#_Toc495909012)

# Overview

# 2017

1. 确认，即使数据已限定为舟山市；百度转换的结果经纬度还是不准，并且偏移出舟山。

## 

## Week\_43

数据仓库使用方联调问题

多多询问思为工作中遇到的技术问题

### 2017/10/23

数据仓库概要设计文档 100%

1. 完成数据仓库概要设计的模块、流程图
2. 完成数据仓库概要设计书写

### 2017/10/24

1. 重构后的增量采集功能测试 50%
2. 向张思为讲解链路日志的路劲形成规则，即如何解析
   1. 介绍spark-sql数据如何读取（主要讲解etl-core对应的数据读取及保存部分代码）

### 2017/10/25

1. 重构后的增量采集功能测试 100%
   1. 实现增量采集与oozie结合实现定时采集，并测试通过

张思为6

10/25

1. 数据融合基于dataframe开发.

融合基于spark-sql dataframe开发 60%

### 2017/10/26

1. 解决oracle Sparta连不上问题
2. 解决转换清洗输出表，表名没带源表表名问题

增量采集的数据与转换清洗、融合整个处理链路进行集成及测试 30%

zhangsiwei6(张思为6) 10-26 18:24:02

10/26

1. 发现转换清洗输出路径问题, 讨论解决方案.

2. 数据融合基于dataframe开发.

数据融合优化 50%

融合基于spark-sql dataframe开发 70%

### 2017/10/27

1. 增量采集的数据与转换清洗整个处理链路进行集成及测试 100%
   1. 备注：采集+转换清洗功能已测试，融合数据已经读取转换清洗结果为dataframe；整个处理链路采集到融合输入均已按计划实现
2. 解决融合技术问题：dataframe重构问题
3. 实现融合基于dataframe实现的demo核心功能流程

zhangsiwei6(张思为6) 10-27 18:07:22

10/27

1. 数据融合基于dataframe技术难点讨论解决.

2. 数据融合基于dataframe开发.

3. 添加相应的Test.

数据融合优化 60%

融合基于spark-sql dataframe开发 80%

### 2017/10/28

### 2017/10/29

### 2017/10/30

## Week\_42

### Plan

|  |
| --- |
| 1. 专利书写 2. 车辆接口验证 |

### Require

|  |
| --- |
|  |

### Done

|  |
| --- |
| 增量数据在时间轴上更新图形化分析 |

### 2017/10/16

1. 参加周会
2. 专利资料准备

增量table数据采集功能与任务调度集成 20%

1. vulcanus规则机制理解.

2. udf通用规则开发, 相关文档整理.

3. spark-sql内置functions学习.

UDF开发: 80%

### 2017/10/17

1. 专利书写，完成相关流程图及数据截图

增量table数据采集功能与任务调度集成 30%

10/17

1. UDF通用规则开发.

2. UDF规则开发文档.

3. UDF规则测试.

4. 大数据算法比赛方案讨论.

UDF开发: 90%

### 2017/10/18

1. 专利书写阶段性完成
2. 参加易鲸捷数据库会议
3. 增量table数据采集功能与任务调度集成 40%

### 2017/10/19

1. 参加项目会议

2. 处理案件地图的30w数据

### 2017/10/20

1. 处理案件地图的30w数据按使用方要求转化地区、经纬度、类型
2. 初步处理前端3d功能模块所需数据

* 案件地图数据（李凯）
* 3d功能模块数据（李凯）
* 实体、关系、事件模拟数据（李金龙+张思为）
* 算法（王辉）代码优化（重构）【java、scala】（昨天通锋哥、及辉哥沟通；已取消）

### 2017/10/21

1. 处理案件地图的经纬度, 因为人工百度转换的数据有问题；采用poi库重新处理
2. create temporary function assign as 'com.hikvision.sparta.vulcanus.udf.impl.common.AssignUDF' using jar '/zsw/sparta-vulcanus-udf-assembly.jar';

### 2017/10/22

## Week\_41

* Plan
  + 5s webservice数据接入：重点车辆卡口过车记录数据接入
  + 专利文档书写：一种用于数据管理与认知的数据仓库方案
  + 绩效计划：Q3 绩效计划，Q2绩效计划总结
  + 原有计划开发任务
  + **验证集群化数据有无丢失**
    - **Hdfs 亿级别数据迁移**
    - **Hbase 亿级别数据迁移**

**数据资产的工作：**

**1. 负责db模块，转换清洗模块，webservice模块的开发维护**

**2. 负责统一视图的更新维护；负责围绕在统一视图周围的功能的维护，包括视图解析，规则定义等。**

**3. 负责打包部署模块的维护，主要是编译部署脚本，命令脚本**

**4. 产出对应的markdown文档：vulcanus部署实施手册、vulcanus核心工作、vulcanus统一视图、vulcanus规则机制等**

**1. 数据生成模块开发测试.**

**2. 数据融合模块开发优化测试.**

**3. 服务器各项配置部署.**

**4. 单机版, 集群版CM平台安装配置部署及调试.**

**5. yum代理配置及服务器相关软件安装(oracle,mysql等)**

**6. 出差部署现场环境.**

**7. vulcanus项目熟悉及部署.**

**8. 大数据各个组件知识储备, spark, hdfs, hbase等.**

**9. 相关文档生成.**

**10. 根据算法组的需求对字典表分类整理.**

**11. 根据算法组要求插入数据到218张表,解决相关问题.**

**12. 案件地图经纬度转换.**

**集群**

**Vulcanus小组工作阶段性总结0.7**

* **配合测试同事进行0.6版本测试、专利书写[组内]**
  + **功能、相关知识讲解、缺陷修复**
* **数据资产池相关及识别客户数据更新频率[李凯]**
  + **已做模块基于spark-sql进行开发**
* **集群搭建及数据迁移[张思为]**
* **人脸、车辆本地部署[李金龙]**
* **根据重点id数据接入车辆、人脸数据[李凯]**

**0.6**

* **数据质量，数据资产池试用，大数据公共模块，案件地图地址转换工作[李凯]**

**0.5**

* **数据采集，任务调度，集群梳理[李凯]**
  + **Oracle及hdfs数据采集**

### 2017-10-09

1. 参加周会
2. 初步实现ETL（采集与转换清洗）链路日志重构（data\_version\_load、etl\_core）
   1. ETL链路日志重构（data\_version\_load、etl\_core）50%

<http://10.17.139.67:8080/face-web/ws/rest?_wadl>

./wadl2java -d ../src -p com.hikvision.face <http://10.17.139.67:8080/face-web/ws/rest?_wadl>

C:\ProgramData\Oracle\Java\javapath;%SystemRoot%\system32;%SystemRoot%;%SystemRoot%\System32\Wbem;%SYSTEMROOT%\System32\WindowsPowerShell\v1.0\;%JAVA\_HOME%\bin;%MVN\_HOME%\bin;%HADOOP\_HOME%;E:\developPlat\MinGW\msys\1.0\bin;C:\Program Files\Git\bin;C:\Program Files (x86)\scala\bin;C:\Program Files\nodejs\;C:\Program Files\PostgreSQL\9.4\bin;C:\Program Files\TortoiseSVN\bin;C:\Python33

Error: Registry key 'Software\JavaSoft\Java Runtime Environment'\CurrentVersion'

Unsupported major.minor version 52.0

### 2017-10-10

1. 初步实现ETL（采集与转换清洗）链路日志重构（data\_version\_load、etl\_core）
   1. ETL链路日志重构（data\_version\_load、etl\_core）80%
2. 处理5s webservice技术问题，通过java代码来实际测试人脸的抓拍库查询接口
   1. 解决java url编码问题

### 2017-10-11

1. 数据质量与大数据公共模块基于spark-sql dataframe开发 40%
2. 准备及实施vulcanus组阶段性工作总结讨论
3. 设计webservice数据接入技术实施方案

### 2017-10-12

1. 原有hdfs数据过渡到spark-sql方案设计与开发 60%

c) 实现hdfs文本数据结合mysql元数据写入到spark-sql数据仓库

d) 确定数据质量与大数据公共模块基于spark-SQL开发的具体方案

### 2017-10-13

1. 现场人脸的数据量：79602346
2. 现场的车辆数据接口情况
3. 现场数据的更新频率：通过备份mysql数据库回来
4. 绩效计划填写
5. 思为阅读转换清洗模块，以进行根据spark-sql开发

数据资产统筹规划、维护、开发

1. 原有hdfs数据过渡到spark-sql方案设计与开发 80%
2. ETL链路日志重构（data\_version\_load、etl\_core）80%
3. 数据质量与大数据公共模块基于spark-sql dataframe开发 70%

### 2017-10-14

完成以下模块及通过功能测试

1. 原有hdfs数据过渡到spark-sql方案设计与开发 100%
2. ETL链路日志重构（data\_version\_load、etl\_core）100%
3. 数据质量与大数据公共模块基于spark-sql dataframe开发 100%

## Week\_39

### 2017-09-25

1. 在测试服务器上部署oozie并测试wordcount通过oozie成功
   1. Workflow执行成功
   2. Coordinator执行成功
2. 参加5s对接会议
3. 书写oozie实践文档到组内wiki上

### 2017-09-26

1. 解答测试同事关于oozie任务调度、数据资产池及数据质量的相关问题
2. 设计采集基于spark-sql数据仓库开发的具体方案
3. 进度：
   1. spark-sql数据仓库部署及测试 40%
   2. 采集、转换清洗基于spark-sql数据仓库开发 10%

### 2017-09-27

1. 给测试准备oozie增量数据采集的环境来测试任务调度模块
2. 采集基于spark-sql数据仓库开发 30%
   1. 实现功能逻辑方法基于dataframe处理
3. spark-sql数据仓库部署及测试 50%

### 2017-09-28

1. 给测试准备oozie使用文档,并讲解oozie相关知识
   1. 排查数据质量2张表未进行统计的问题，表属性带有关键字及特殊字符？
2. 采集基于spark-sql数据仓库开发 80%
   1. 实现处理结果封装保存到spark-sql相关的单元功能测试
   2. 实现采集spark-sql功能与原有功能基于输入参数配置可执行
3. spark-sql数据仓库部署及测试 70%

### 2017-09-29

1. 帮助葛挺组在测试服务器上搭建部署案件地图solr搜索引擎
   1. 并将POI数据及已转换百度的地址数据分别写入搜索引擎
2. 转换清洗基于spark-sql数据仓库开发50%
   1. 功能逻辑函数基于dataframe处理
   2. 解析采集load链路日志并实现转换清洗从spark-sql读取数据
3. spark-sql数据仓库部署及测试 80%

### 2017-09-30

1. 采集、转换清洗基于spark-sql数据仓库开发100%
   1. 转换清洗任务处理好的结果数据写入spark-sql数据仓库并将相应的链路日志写入mysql
   2. 并实现转换清洗简单的流程功能测试
2. spark-sql数据仓库部署及测试 100%

1. ssh 到master66

2. spark-sql

3. use vulcanus\_load;

3. select \* from vulcanus\_load.BADPERSON\_S\_FZXYRY;

## Week\_38

## Plan

|  |
| --- |
|  |

### Require

|  |
| --- |
|  |

### 2017-09-18

1. 同智能算法部同事沟通舟山案件地址识别和PIO连接工作
2. 给予测试0.6 数据质量 及 数据质量SQL接口(spark-sql数据仓库)相关文档
3. 实现转换清洗结果数据 数据质量统计

### 2017-09-19

1. 参加压力测试需求评审会议
2. 实现大数据公共模块---Spark-sql数据仓库数据访问接口

### 2017-09-20

1. 完成数据质量--识别源数据更新频率初步设计
   1. 结果mysql数据库表结构设计
   2. 代码处理逻辑流程初步设计
2. 完成测试环境(0.6版本测试)—spark-sql数据仓库部署及验证

### 2017-09-21

1. 协助测试定位解决0.6测试环境大数据环境
   1. 配置更改没重启服务,从而未生效;导致任务cluster模式运行不成功
2. 数据质量--识别客户方数据的条数更新（通过定时进行查询实现）80%
   1. 实现该功能的主要处理逻辑并测试
   2. 实现参数解析模块及调用调度逻辑
3. 大数据公共模块--Spark-sql数据仓库数据访问接口 100%

上次bug描述两个问题：

1. number没有实现，【已实现】

2. 日期类型没有 时分秒 这个，应该在数据库插入oracle数据库的时候，时间数据不对

### 2017-09-22

1. 数据质量--识别客户方数据的条数更新（通过定时进行查询实现）100%
   1. 添加定时调度逻辑，通过quartz实现
   2. 本地ide及master66服务器测试运行通过
   3. 书写部署运行文档
2. 组间联调
   1. 向测试同事通过实际操作讲解yarn相关知识点，及spark job on yarn相关配置

### 2017-09-23

1. 实现如果本次摸底数据与上次摸底数据一致不进行结果保存逻辑
2. 代码review

### 2017-09-24

## Week\_37

### Plan

|  |
| --- |
|  |

### Require

|  |
| --- |
| 1. 完成数据资产组0.7版本的需求和开发计划确定 2. 评估人脸和车辆系统接口可用性，形成技术方案 |

* V0.7需求及开发计划
  + 缺陷修复、代码优化
  + 基于spark-sql开发
    - Spark-sql数据仓库部署及测试
    - 采集、转换清洗、融合基于spark-sql dataframe开发
    - 与数据使用方老葛协商spark-sql使用
    - ETL链路日志重构（data\_version\_load、etl\_core）
    - 原有hdfs数据过渡到spark-sql方案设计与开发
    - 数据质量与大数据公共模块基于spark-sql dataframe开发
    - 上述模块设计及测试
  + 增量table数据采集重构及任务调度集成
    - 增量table数据采集功能重构设计
    - 增量table数据采集功能重构开发
    - 增量table数据采集功能与任务调度集成
    - 增量table数据采集功能与任务调度集成后功能测试
  + 5s人脸及车辆数据接入
* Face\traffic 技术方案
  + 评估人脸和车辆系统接口可用性

### 2017-09-11

1. 深入学习spark-sql技术
   1. 学习spark-sql function模块，了解spark-sql具体有哪些原生function
   2. 学习spark-sql2.2 Cost-Based Optimizer核心概要模型
      1. 通过计算table、column的条数、大小结合计算链路来计算cost

### 2017-09-12

1. 讨论数据资产组内v0.7的工作计划
   1. 具体需要做哪些功能
      1. 具体我和张思为先整理，然后还需要做的邱工补充
   2. 我与张思为分别做哪些功能
   3. 确定v0.7的开发计划任务

### 2017-09-13

1. Intellij 文件头注释模板统一确立
2. Scala 代码单元测试学习及简单实际应用
   * 1. 采用FunSuite风格

数据质量--识别源数据更新频率

data\_load\_pioneer 数据摸底表映射及操作实现

大数据公共模块：

Hdfs 增删封装--hdfs操作核心方法封装

### 2017-09-14

1. 和大家一起初步制定数据资产组v0.7版本的开发计划

a) 具体任务、完成时间及开发人员

2. 学习spark技术

a) 阅读spark 新增主推功能structure streaming官方文档

### 2017-09-15

1. 自己内部运行0.6版本代码,并且就主要功能进行简单功能测试
   1. 修复0.5版本测试提出的缺陷(增量采集未支持Number类型字段,重点测试)

评估人脸和车辆系统接口可用性，形成技术方案。

和锋哥沟通:交通那边没有服务器,给了我们安装文档,我们将他们那套安装在我们的服务器上,(自己造数据);关于数据,看测试那边可有方法解决

### 2017-09-16

### 2017-09-17

## Week\_36

### Plan

|  |
| --- |
|  |

### Require

|  |
| --- |
|  |

### 2017-09-04

### 2017-09-05

1. 部署solr集群
   1. 安装solr集群
   2. 更改solr相关配置并重启生效
   3. 上传solr配置文件并创建zsmap[百度转换的经纬度数据集]、mapzs[poi数据集] 的collection
   4. 将poi数据集写入mapzs collection
2. 数据资产组转换清洗及融合升级
   1. 备份元数据库、清除相关hdfs数据及元数据库表数据及更改相关配置
   2. 更新统一视图并执行转换清洗任务[第二天查看，运行成功(3.5h)]
3. 问题：
   1. 算法python导数据程序内存溢出异常
      1. 通过更改spark-submit参数予以解决

### 2017-09-06

1. 经纬度数据写入solr
   1. 将百度转换的经纬度数据集写入zsmap collection
      1. 问题：程序报 not Serialization Exception
         1. 排查分析：本地运行正常，线上异常；推测为相关maven jar依赖未解决
         2. 解决：笔记本电脑联网解决maven依赖并重新打包
2. 执行融合任务
   1. 执行成功，运行3.5小时
3. 转换清洗、融合结果数据写入spark-sql数据仓库
   1. 删除vulcanus\_core、vulcanus\_merge相关表
   2. 写入转换清洗相关表数据
   3. 写入融合结果相关表数据
4. 执行李金龙vulcanus\_load(贴源层) sql统计语句
5. 帮助葛挺导出vulcanus\_load部分表数据并加密发给葛挺
6. Yarn上自带job，杀不死并且越杀越多
   1. 通过咨询组件组提供解决予以解决[不确定机器重启之后该问题是否重现]

**出差总结：**

* 1. 案件地图poi、百度转换经纬度数据集均按设计导入solr搜索引擎
  2. 数据资产组转换清洗、融合按设计进行升级
  3. 上述出差目标均已达到，并按计划完成

### 2017-09-07

1. 整理出差记录文档
2. 协助算法解决215个卡口7亿多过车数据提取困难
   1. 向邢金彪索要卡口id列表文件
   2. 根据实际需求生成抽数据spark-sql语句
   3. 使用说明：
      1. 
      2. columns 全部替换为实际需要的 列名并逗号分隔
      3. conditions 全部替换为实际需要的where 条件语句 and连接
3. 协助测试理解采集增量实现
4. 表conf\_dict ：为配置字典，
   1. 记录：
   2. 即表示自增规则已配置
5. 表 conf\_dict\_elem ：为字典规则应用
   1. `conf\_dict\_elem\_id` int(11) NOT NULL AUTO\_INCREMENT,
   2. `conf\_dict\_id` 外键 REFERENCES `conf\_dict` (`conf\_dict\_id`)
   3. `elem\_key` 表id
   4. `elem\_value` 属性id
6. 增量实现：在表conf\_dict\_elem中插入记录即可；
   1. Eg：
      1. 解释：即为id=1的源表的attr\_id=1的属性为增量字段
7. 测试问题记录，表属性类型为rowid的采集失败
   1. 已分析(通过走读源码)并测试为rowid底层解析为long，理应解析为string
   2. 解决：通过重写该代码予以解决
   3. 

### 2017-09-08

专利文档书写

1. 修复增量采集数据类型支持bug
   1. 丰富增量字段数据类型支持数学、string、日期等
   2. 测试增量字段数据类型（数学、string、日期）实现等功能[测试通过]
2. 组内工作：书写自己负责模块文档
   1. 学习github wiki使用
   2. 书写 数据质量模块文档

### 2017-09-09

自己负责模块文档书写

专利文档书写

1. 继续书写自己负责模块文档
   1. 书写数据仓库资料文档
      1. 数据仓库部署
      2. 数据仓库访问
      3. 使用数据仓库遇到的问题及解决方法
   2. 案件地图地址转经纬度文档
   3. 案件地图solr搜索引擎部署文档

### 2017-09-10

## Week\_37

# 引用

# 附录

# 0000-年模板

## Week\_00-周模板

### Plan

|  |
| --- |
|  |

### Require

|  |
| --- |
|  |

### Done

|  |
| --- |
|  |

### 0000-01-01

### 0000-01-02

### 0000-01-03

### 0000-01-04

### 0000-01-05

### 0000-01-06

### 0000-01-07