车联网平台升级方案

# 1 背景

## 1.1 背景

车联网开发已经进入2.0阶段，从最初的项目产品，到作为应用平台，功能模块不断增多，系统复杂性也不断提高。业务管理方面，多个应用集成在一个系统内部，无法动态配置，模块划分不明确；系统部署方面，整个系统只能部署在单一服务器上，不能利用多服务器的优势，不仅效率无法保证，且存在单点故障风险；开发方面，所有模块集成在一个解决方案中，复杂冗余，团队分工不好明确，开发效率低。

为解决现有问题，考虑将整个系统进行微服务化，根据不同的业务将系统划分成多个子系统，并定义各子系统之间的交互逻辑，各子系统实现独立的业务逻辑，同时对外发布相关接口，做到多团队并行开发、业务模块化管理、系统分布式部署。在整个系统改进的过程中，能够重复利用现有的业务模块，如视图层、业务逻辑层、数据访问层相关实现。

## 1.2 目标

1. 扩大车联网接入服务的覆盖范围，形成物联网接入平台；

2. 通过将现有车联网平台进行模块化封装，形成标准的系列业务子系统，同时支持新增更多业务子系统；

3. 车联网平台化，通过研发基础应用平台，提供通用服务接口，可针对具体项目进行扩展开发；

4. 系统微服务化，业务模块化，研发独立化，版本可控、外包可控。

# 2 系统方案

## 2.1 业务管理

车联网平台按照业务属性划分为平台基础模块和业务应用模块。平台基础模块包括平台管理、基础信息管理，以及根据各行业应用提取出来的通用业务逻辑；业务应用模块则面向具体的业务应用，开发的专用业务逻辑，或者根据客户需要定制的业务逻辑。

系统提供发布模块管理界面，在系统发布前，配置需要发布的模块，并将生成的配置信息随系统文件一起发布到数据库中，发布后的系统只具有其中的功能模块。

### 2.1.1 平台基础模块

#### 2.1.1.1 平台管理

本模块进行车联网平台的系统管理，包括组织、机构、用户、角色、权限管理，以及系统多语言设置、操作日志记录。

#### 2.1.1.2 基础资料

基础资料指车联网平台运行不可或缺的要素，包括车辆、驾驶员、终端、SIM卡、证照等，包括相关的信息管理。具体包括：

1）车辆信息管理，需要对目前的车辆信息进行梳理，分为车辆基本信息、车辆扩展信息。车辆基本信息包括车辆本身的最基本属性，如车号、车牌、颜色、车架号等；扩展系统考虑不同业务应用的逻辑，保留系统扩展的可行性，如针对监控系统扩展车辆的状态信息，或者从相关业务系统同步的扩展资料。

与车辆相关的车辆类型、能源类型、车辆型号管理。

2）终端管理，终端作为独立的设备进行管理。

3）驾驶员管理，同样需要像车辆信息一样，将驾驶员本身的信息抽象出来，同时利用扩展信息保留扩展的可行性。

4）SIM卡管理，包括SIM卡的供应商管理、套餐管理、SIM卡信息管理、费用管理。

5）证照管理，包括驾驶员、车辆的相关证照信息管理，证照到期提醒管理。

#### 2.1.1.3 实时监控

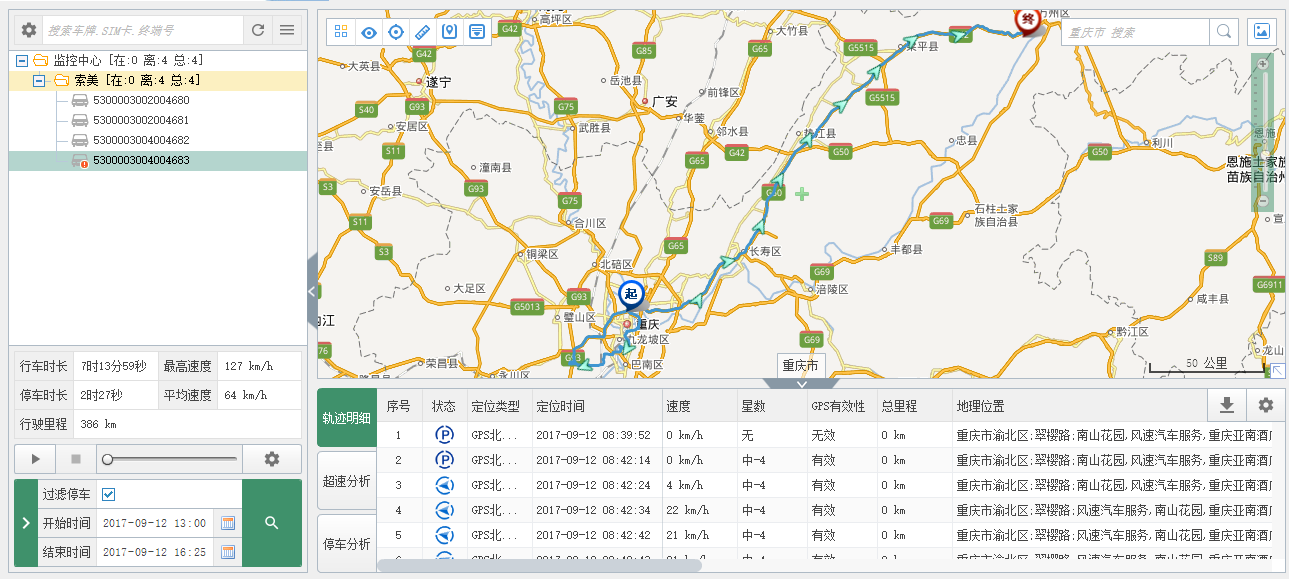
实时监控模块完成车辆地图显示、历史轨迹回放、视频查看、指令下发、车辆报警提醒、报警处理。具体包括：

1）地图监控

地图监控包括车辆实时位置显示，车辆运行状态展示，多分屏模式监控，车辆在线率、定位率统计；同时，地图监控模块还包括所有车辆的最新状态表，实时报警车辆查看，报警处置，以及按配置显示的直线图。

2）历史轨迹回放

优化现有逻辑，增加轨迹明细、停车分析，速度分析，报警查看，如下图所示。



3）视频监控

4）指令下发

指令下发包括调度短信、超速及预警值设置、终端参数（接入地址、心跳、上传间隔）、远程控制，可参考如下界面。



#### 2.1.1.4 安全管理

1）电子围栏

电子围栏管理、车辆关联，以及分段限速管理。

2）标识物

管理地图中需要显示的重要位置点。

3）安全设置

包括默认限速值、是否启用分段限速、是否启用超速延时、是否启用电子围栏、是否启用夜间限速。

4）报警明细

提供安全报警明细查询界面，以及查看轨迹、人工核警操作。

5）事故录入

提供安全事故录入界面，包括事故时间、地点、原因、相关车辆、人员，同时提供事故明细查询界面。

#### 2.1.1.5 机务管理

1）车辆保养

包括车辆保养计划设置、保养记录、保养提醒、明细查询。

2）故障登记

提供故障人工录入界面、明细查询界面，便于系统进行相关的统计分析。

可以将故障管理做成二级子系统，包括故障申报、故障分发、故障处理、专家库，形成故障的闭环管理。

3）油耗管理

提供加油录入界面，加油明细查询界面。

4）轮胎管理

轮胎基础资料管理，轮胎生命周期管理。

#### 2.1.1.6 统计分析

统计分析模块主要提供数据分析功能，形成图形化的分析报表。

上线/定位统计

油耗统计

报警统计

故障统计

CAN数据分析（国标标准项支持）

安全事故统计

### 2.1.2 业务应用模块

面向车厂、公交企业、运营商、政府监管部门、公务车、出租、社会车辆（两客一危、贷款车等）管理等的行业应用，包括第三方合作（如中移物联）的应用开发，由车联网平台提供基础应用模块，在此基础上进行二次业务开发。

目前，公交具有较为明显的个性化应用模块，货运、出租在提取出公共模块后，具体应用还需在项目实践中进行提炼。

#### 2.1.2.1 公交运营管理

该模块针对公交业务应用，包括：线路管理、站点管理、计划排班、调度发车、直线图监控、运营报表（日报及明细、统计）、票款管理。

#### 2.1.2.2 政府监管

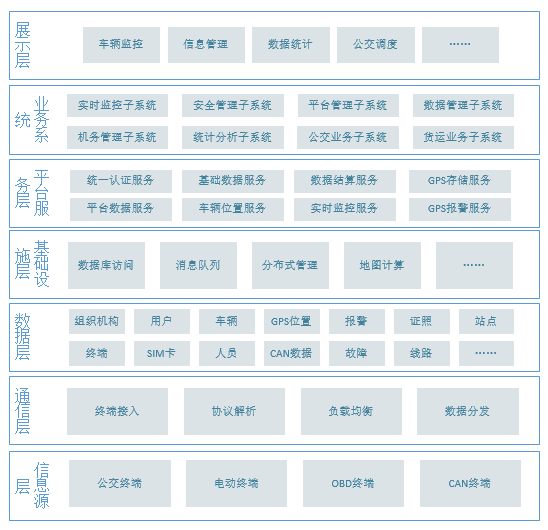
针对地方运管局、运管所或类似的行业监管部门，除一些车辆监控的基本功能外，还需要对车辆运营企业进行管理，因此本模块针对行业监管应用，包括：企业资料管理、企业车辆审核、企业数据统计、企业安全数据排名、企业点名管理。

## 2.2 技术方案

整个车联网平台按照业务不同划分成多个应用子系统，各个子系统可独立开发、独立部署、独立运行。同时，车联网平台提供多个公共基础服务接口，包括用户认证服务、组织机构服务、基础数据服务，供各业务子系统服务。

用户通过认证服务登陆授权后，进入车联网平台首页后，不同的业务模块链接到不同的子系统，子系统之间通过用户TOKEN参数辨别登陆用户以及权限验证。业务子系统可以部署在一个服务器上，也可以按照需要部署到多个不同服务器上。

具体架构如下图所示。



### 2.2.1 方案说明

1 业务系统独立

按照业务类型对车联网系统进行模块化划分，形成独立的解决方案，通过服务接口的形式进行数据交互，能够进行并行开发，互不影响。

2 前后端分离

将各种基础业务数据的访问通过服务封装起来，实现前后端分离，同时为其他业务系统提供数据接口。

3 分布式部署

各业务系统可以同时部署到一台服务器，也可以分开部署到不同的服务器上，同一业务系统或接口服务支持部署多份。需要统一的分布式会话管理机制。

### 2.2.2 数据库优化

当平台接入车辆不断增多时，车辆实时数据的量将会越来越多，目前单库、单表存储的方式已无法满足应用需求。另外，由于各种基础数据和GPS动态数据同库存储，GPS的大批量读写还会影响到其他业务数据的访问效率。因此，在本次升级中，从两个方面对数据进行分离：

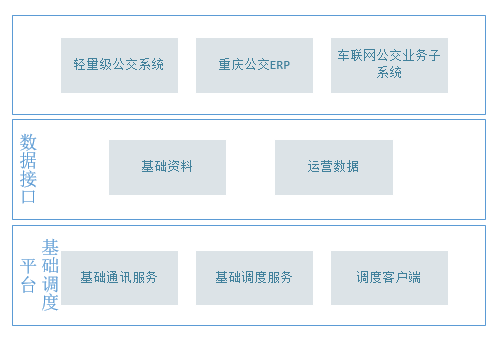
1）业务、动态数据分离。业务数据的数据量小、不常变化且访问频率较低，但对实时性要求高，影响客户体验，而GPS动态数据量大、写入频率高，容易对其他数据的访问造成影响。将业务数据和动态数据分离，通过不同服务器提高各自的读写效率，更便于系统维护。

2）GPS分表处理。实时数据中，量最大的是位置数据，且不断增长，根据以往经验，对于两万辆车10秒一个点的上传频率，一个月存储一张表，数据量和读写效率是可以接受的。在后台存储服务中，对不断实时上传的GPS位置数据、CAN数据，采用分表存储，一个月一张表，命名方式在现有的表名后增加日期后缀，如“…201709”、“…201710”等。前端业务如轨迹回放，在查询数据时需要进行同步修改。

在数据库部署方面，可以通过微软“Always On”技术，将多台服务器组成集群使用，保证数据不丢失、业务不停止的同时，实现读写分离，以提高系统整体性能。

## 2.3 轻量级调度系统

车联网本身提供的公交业务模块仅包含基本的调度业务，为了提供功能更丰富的调度管理，并保证车辆调度的实时性、便捷性，将车联网同轻量级公交调度基础平台结合起来，提供C/S版本的调度客户端子系统，完成逻辑更复杂的计划管理、调度发车、运营管理功能。车联网作为数据管理平台，提供基础数据，并完成报表统计分析功能。基础调度平台与轻量级调度系统、车联网等系统的关系如下图所示。



# 3 详细设计

本次项目改造的过程中，采用业务子系统独立开发、独立部署的方式进行。各子系统只进行本身业务逻辑开发，互相解耦，在展现形式上，只实现本业务的项目页面内容，只能通过主Web的菜单加TOKEN的形式进行访问，不能独立打开。各子系统为了方便调试，也可以实现类似于主Web的外部菜单框架。各业务子系统之间，禁止页面级的访问和跳转，只允许通过接口进行数据访问，各自实现自己的View内容。

主Web项目需要进行调整，在现有的基础上进行菜单和模块整理，菜单链接的地址由同一项目，修改为对应子系统的功能地址。

对于公共的名词定义、类型定义、枚举定义等，如数据有效性、线路上先行，由基础数据管理模块进行定义，通过发布DLL的方式由各业务子系统进行直接引用。

## 3.1 用户认证服务

服务名称：IOV.Authrize。（备注：服务名称同时是业务系统类型，也是项目解决方案名称和项目名称前缀。）

本服务基于Token的认证方式，提供用户信息验证、用户权限查询接口，也提供用户登陆界面，为所有应用子系统的公用服务。各业务子系统通过本服务判断用户授权信息，也可以通过本服务器重新登陆。认证流程如下图：



### 3.1.1 用户认证

用户认证流程如下：

1）客户端发送加密后的用户名、密码、客户端类型到身份认证服务；

2）身份认证服务判断客户端是否有效，同时验证用户名、密码是否准确，用户是否有访问客户端的权限；

3）验证失败，直接返回错误编码以及错误描述；

4）验证成功，验证接口生成Token，保存到Memcached或Redis中并设置过期时间（可配置），同时将Token和用户基本信息加密后返回到客户端；

5）客户端收到认证结果后，如果失败，给出用户验证失败提示，成功则进入客户端业务界面；

6）当需要进行跨业务子系统访问时，将Token加密后，作为参数传递到目标业务系统；

7）目标业务系统拿Token调用身份认证服务判断有效性，用户非法则报错，正确则刷新Token过期时间并返回用户基本信息给目标业务系统。

**备注：**整个车联网平台中的加、解密方式均采用DES加密，秘钥在数据库中保存，随时可以变更。所有业务系统跳转、服务接口调用，参数均需加密。

### 3.1.2 用户登陆

由于各业务子系统都涉及到用户验证，但整个平台只有一个登陆页面，各子系统通过用户TOKEN进行交互，为了使各个子系统能够独立运行，身份认证服务提供统一登录界面，具体流程如下：

1）业务子系统打开后，如果不是从其他业务系统通过TOKEN的方式跳转过来，而是直接打开，则跳转到身份认证服务的统一登录界面，将回调地址、业务系统类型加密后作为参数；

2）身份验证服务验证系统类型，有效则进入登录界面，否则直接报错；

3）登录界面验证用户名和密码，验证成功后，将用户Token和用户基本信息保存到Memcached或Redis中并设置过期时间（可配置），同时将Token加密后作为参数跳转到回调地址；

4）业务系统收到Token后进入业务界面。

## 3.2 服务网关

服务网关负责对微服务进行统一管理，包括动态发布新的微服务、服务接口配置、调用计数、并发数控制、计费收费、路由、微服务故障实时提醒等。

客户端不直接访问微服务，只需和服务网关打交道，这样可以简化客户端代码，所有请求通过服务网关再路由到对应的微服务。如下图：



## 3.3 平台管理子系统

服务名称：IOV.Platform。

### 3.3.1 模块简介

本业务子系统负责对车联网产品的各种基础数据进行管理，包括组织、用户、角色、权限、人员、多语言、操作日志、车辆、终端、SIM卡等先关数据。

### 3.3.2 平台基础

#### 3.3.2.1 多语言

多语言采用现有方式进行管理。

#### 3.3.2.2 操作日志

操作日志用于记录系统各种操作的记录，主要是对基础数据的编辑、设置、下发指令等操作的记录。

本模块提供日志写入接口，以及日志明细查询界面。需要记录增、删、改的数据类型包括：组织机构、人员、终端、SIM卡、车辆、用户、角色、权限、线路、计划时刻、人车安排、运行作业计划、电子围栏、安全设置、参数下发、调度短信。

操作日志记录由本模块提供接口，内部需要使用消息队列的方式进行异步写入，避免对业务流程产生影响。需要开发日志写入服务，从消息队列接收日志，并执行具体的数据库操作。

|  |  |  |
| --- | --- | --- |
| IOV\_OperationLog（操作日志） | | |
| 名称 | 类型 | 描述 |
| ID | long | 主键 |
| OpUser | long | 操作人ID |
| OpTime | datetime | 操作时间 |
| OpIpAddr | varchar(30) | 操作IP地址 |
| DataId | long | 操作数据的ID |
| OpType | int | 操作类型，数据类型，需定义枚举，包括组织、人员、终端、车辆等。 |
| Content | nvarchar(200) | 操作内容 |

### 3.3.2 组织人员

车联网基础平台对组织、人员的基本信息进行管理，各业务模块可根据实际情况进行扩展。

#### 3.3.2.1 组织

组织用于对系统数据进行归属划分，可以是实际的企业机构，也可以是虚拟的车辆分组。如下界面所示。



|  |  |  |
| --- | --- | --- |
| IOV\_Organization\_Type（组织类型） | | |
| 名称 | 类型 | 描述 |
| ID | long | 主键ID |
| Name | nvarchar(50) | 名称 |
| Code | nvarchar(50) | 类型编码 |
| Remarks | nvarchar(100) | 备注信息 |
| CreatedDate | datetime | 创建时间 |
| CreatedBy | long | 创建人ID |
| ModifiedDate | datetime | 修改时间 |
| ModifiedBy | long | 修改人ID |
| DataStatus | tinyint | 数据状态，需要定义枚举，如0：正常，1：禁用，2：注销，3：删除 |

|  |  |  |
| --- | --- | --- |
| IOV\_Organization（组织信息） | | |
| 名称 | 类型 | 描述 |
| ID | long | 主键ID |
| Name | nvarchar(50) | 名称 |
| ParentId | long | 上级组织ID，0表示顶级组织 |
| ContactName | nvarchar (30) | 联系人姓名 |
| ContactPhone | nvarchar(20) | 联系电话 |
| Address | nvarchar(50) | 所在地址 |
| Remarks | nvarchar(100) | 备注信息 |
| OrgTypeCode | nvarchar(50) | 机构类型编码 |
| CreatedDate | datetime | 创建时间 |
| CreatedBy | long | 创建人ID |
| ModifiedDate | datetime | 修改时间 |
| ModifiedBy | long | 修改人ID |
| DataStatus | tinyint | 数据状态，需要定义枚举，如0：正常，1：禁用，2：注销，3：删除 |

#### 3.3.2.2 人员

人员对应企业组织中的一个具体的人，如驾驶员、调度员、会计、经理等。车联网中主要对系统涉及的相关人员进行管理，这些人并不一定是系统用户，但是在业务中需要进行关联，如驾驶员、乘务员等。

人员信息只简单的包括基本信息，如姓名、机构、身份证号、电话、职务，均为通用信息。当需要对一类人进行深度管理时，如针对驾驶员，需要记录其行驶证等信息，则需要增加扩展表，来补充说明。

人员的职务类型需要进行单独管理，信息包括名称、编码。在人员信息中记录职务编码，职务ID可能会因自增或其他原因改变，但应保持编码不变。

人员管理界面类似于现有的用户管理界面，如下所示。



|  |  |  |
| --- | --- | --- |
| IOV\_Person\_Type（人员类型信息） | | |
| 名称 | 类型 | 描述 |
| ID | long | 主键ID |
| JobCode | nvarchar(20) | 类型编码 |
| JobName | nvarchar(50) | 类型名称，如驾驶员、调度员、乘务员、经理等 |
| Remarks | nvarchar(100) | 备注信息 |
| CreatedDate | datetime | 创建时间 |
| CreatedBy | long | 创建人ID |
| ModifiedDate | datetime | 修改时间 |
| ModifiedBy | long | 修改人ID |
| DataStatus | tinyint | 数据状态，需要定义枚举，如0：正常，1：禁用，2：注销，3：删除 |

|  |  |  |
| --- | --- | --- |
| IOV\_Person（人员信息） | | |
| 名称 | 类型 | 描述 |
| ID | long | 主键ID |
| RealName | nvarchar(50) | 姓名 |
| Gender | tinyint | 性别，需定义枚举，0：未知，1：男，2：女 |
| Birthday | datetime | 出生日期 |
| CertId | nvarchar(50) | 身份证号码 |
| PhoneNum | nvarchar(20) | 联系电话 |
| OrgId | long | 所在机构的ID |
| TypeCode | nvarchar(20) | 类型编码，也表示职务 |
| CreatedDate | datetime | 创建时间 |
| CreatedBy | long | 创建人ID |
| ModifiedDate | datetime | 修改时间 |
| ModifiedBy | long | 修改人ID |
| DataStatus | tinyint | 数据状态，需要定义枚举，如0：正常，1：禁用，2：注销，3：删除 |

|  |  |  |
| --- | --- | --- |
| IOV\_Person\_Details（人员详细信息，预留） | | |
| 名称 | 类型 | 描述 |
| PersonId | long | 主键ID，表示对应的人员基础信息 |
| Education | tinyint | 学历，需要定义枚举 |
| …… |  |  |
| CreatedDate | datetime | 创建时间 |
| CreatedBy | long | 创建人ID |
| ModifiedDate | datetime | 修改时间 |
| ModifiedBy | long | 修改人ID |
| DataStatus | tinyint | 数据状态，需要定义枚举，如0：正常，1：禁用，2：注销，3：删除 |

如果是人员类型是驾驶员，查看、编辑详细信息时，需要显示驾驶员相关信息。

|  |  |  |
| --- | --- | --- |
| IOV\_Drivers（驾驶员信息，预留） | | |
| 名称 | 类型 | 描述 |
| PersonId | long | 主键ID，表示对应的人员基础信息 |
| DriverLicense | nvarchar(50) | 驾驶证号码 |
| DriverLicenseDocNum | nvarchar(50) | 驾驶证档案号 |
| QuasiDrivingModel | tinyint | 准驾车型，需定义枚举 |
| DriverLicenseDate | datetime | 驾驶证发证日期 |
| DriverLicenseEndDate | datetime | 驾驶证有效日期 |
| QualificationCode | nvarchar(50) | 从业资格证号码 |
| QualificationEndDate | datetime | 从业资格证有效日期 |
| SignCardNum | nvarchar(50) | 驾驶员签到卡号 |
| CreatedDate | datetime | 创建时间 |
| CreatedBy | long | 创建人ID |
| ModifiedDate | datetime | 修改时间 |
| ModifiedBy | long | 修改人ID |
| DataStatus | tinyint | 数据状态，需要定义枚举，如0：正常，1：禁用，2：注销，3：删除 |

### 3.3.3 用户权限

#### 3.3.3.1 用户

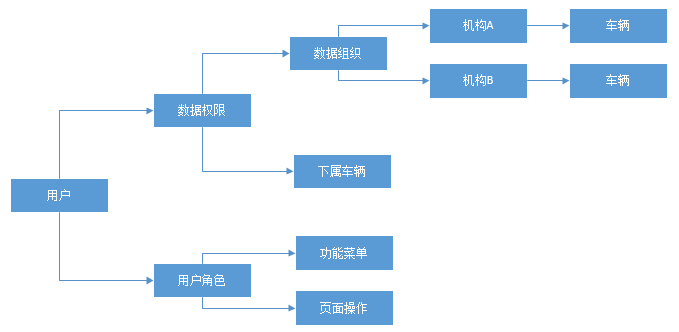
用户是系统的使用者，是一种角色，而不是具体的人员。在目前的车联网系统中，没有区分人员和用户的不同，造成用户必须归属到某个组织，不能独立授权，也不能同时管理多个组织。

用户单独进行管理，创建用户时，可以指定用户属于某个组织，以及是组织中的哪个人。例如，创建调度员用户时，需要指定其所属组织以及时哪个人员。在线路管理模块中，有线路、调度员的关系，调度员用户登陆时，通过所属组织和人员ID去查询线路。

|  |  |  |
| --- | --- | --- |
| IOV\_Users（用户信息） | | |
| 名称 | 类型 | 描述 |
| ID | long | 用户ID，主键 |
| LoginName | nvarchar(50) | 登录名称 |
| LoginPwd | nvarchar(50) | 登录密码 |
| LoginType | tinyint | 账户类型，需定义枚举，如平台用户，APP用户等 |
| LastLoginTime | datetime | 最近一次登录时间 |
| LoginCount | int | 登录次数合计 |
| PhoneNum | varchar(20) | 联系电话 |
| Email | nvarchar(100) | 邮箱地址 |
| OrgId | long | 所属机构ID，可为0，表示仅为系统用户。 |
| PersonId | long | 如果所属机构不为空，则此处对应相应人员的ID。 |
| UserType | int | 用户类型，定义枚举，1：系统管理员（不用绑定组织和数据，为超级管理员，具有所有数据权限），2：企业管理员（需指定所属的组织，具有该组织下的所有数据权限，无需数据绑定），3：普通用户 |
| CreatedDate | datetime | 创建时间 |
| CreatedBy | long | 创建人ID |
| ModifiedDate | datetime | 修改时间 |
| ModifiedBy | long | 修改人ID |
| DataStatus | tinyint | 数据状态，需要定义枚举，如0：正常，1：禁用，2：注销，3：删除 |

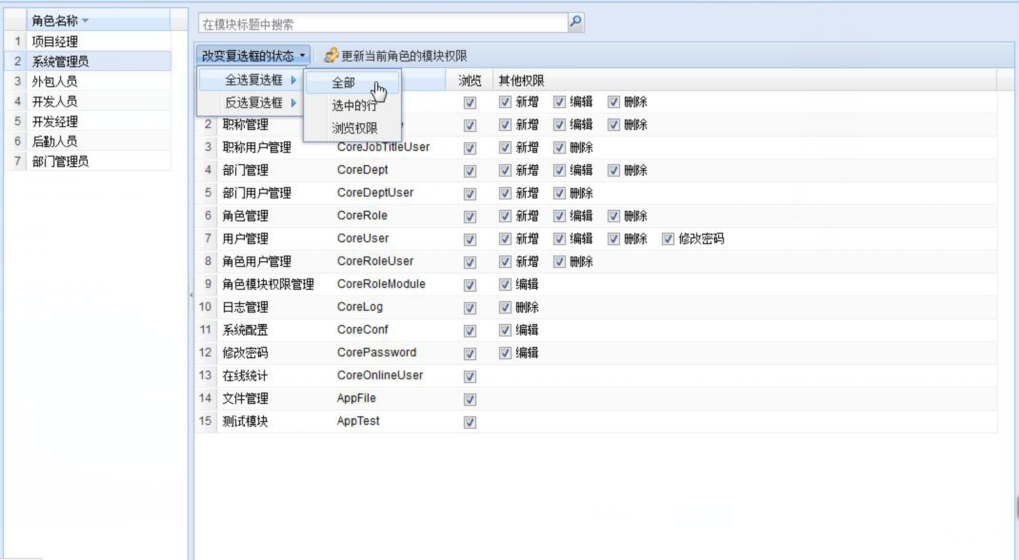
#### 3.3.3.2 权限

用户的权限包括两个部分，数据权限和角色权限，如下图所示。



**功能权限**

用户管理时，不能单独给用户分配权限，只能按角色进行分配，可分配多个角色。每个角色关联的权限包括功能权限和操作权限。如下图所示。



功能权限管理包括菜单管理和操作管理，菜单下对应的页面中，每个操作均进行编码。

|  |  |  |
| --- | --- | --- |
| IOV\_Roles（角色信息） | | |
| 名称 | 类型 | 描述 |
| ID | long | ID，主键 |
| Name | nvarchar(50) | 角色名称 |
| Remark | nvarchar(200) | 描述信息 |
| CreatedDate | datetime | 创建时间 |
| CreatedBy | long | 创建人ID |
| ModifiedDate | datetime | 修改时间 |
| ModifiedBy | long | 修改人ID |
| DataStatus | tinyint | 数据状态，需要定义枚举，如0：正常，1：禁用，2：注销，3：删除 |

|  |  |  |
| --- | --- | --- |
| IOV\_User\_Role（用户角色） | | |
| 名称 | 类型 | 描述 |
| UserId | long | 用户ID |
| RoleId | long | 角色ID |
| CreatedDate | datetime | 创建时间 |
| CreatedBy | long | 创建人ID |
| ModifiedDate | datetime | 修改时间 |
| ModifiedBy | long | 修改人ID |
| DataStatus | tinyint | 数据状态，需要定义枚举，如0：正常，1：禁用，2：注销，3：删除 |

|  |  |  |
| --- | --- | --- |
| IOV\_Menu（菜单信息） | | |
| 名称 | 类型 | 描述 |
| Id | long | 主键ID |
| MenuName | nvarchar(50) | 名称 |
| MenuCode | nvarchar(20) | 菜单编码，一个菜单可以表示一个功能。 |
| ParentId | long | 上级菜单ID，0表示顶级菜单，不具有功能，仅用于分组展示 |
| Url | nvarchar(100) | URL地址，如果有下级菜单，可以为空 |
| SortNo | int | 菜单排序号 |
| Remarks | nvarchar(200) | 权限描述 |
| CreatedDate | datetime | 创建时间 |
| CreatedBy | long | 创建人ID |
| ModifiedDate | datetime | 修改时间 |
| ModifiedBy | long | 修改人ID |
| DataStatus | tinyint | 数据状态，需要定义枚举，如0：正常，1：禁用，2：注销，3：删除 |

|  |  |  |
| --- | --- | --- |
| IOV\_PermissionCode（操作权限编码信息） | | |
| 名称 | 类型 | 描述 |
| Id | long | 主键ID |
| MenuId | long | 所属的菜单ID |
| PermissionName | nvarchar(50) | 权限名称 |
| PermissionCode | nvarchar(20) | 权限编码 |
| Remarks | nvarchar(200) | 权限描述 |
| CreatedDate | datetime | 创建时间 |
| CreatedBy | long | 创建人ID |
| ModifiedDate | datetime | 修改时间 |
| ModifiedBy | long | 修改人ID |
| DataStatus | tinyint | 数据状态，需要定义枚举，如0：正常，1：禁用，2：注销，3：删除 |

|  |  |  |
| --- | --- | --- |
| IOV\_RoleMenu（角色菜单） | | |
| 名称 | 类型 | 描述 |
| RoleId | long | 角色ID |
| MenuId | long | 菜单ID |
| Permissions | varchar(500) | 该菜单下对应的权限编码，形式为“001,002,…” |
| CreatedDate | datetime | 创建时间 |
| CreatedBy | long | 创建人ID |
| ModifiedDate | datetime | 修改时间 |
| ModifiedBy | long | 修改人ID |

**数据权限**

数据权限指用户能够查看或操作的数据范围，包括三个方面：

1）管理员

包括超级管理员和企业管理员。

超级管理员具有所有数据和功能权限，全局唯一。

企业管理员，拥有所属组织的所有数据权限，但是功能权限还是需要单独设置。

2）隶属组织，将一个或多个组织划分给用户，使得用户拥有多个组织的数据权限。如下图所示。



3）隶属车辆

车辆为车联网的最小数据单位，用户管理时，可以单独为其分配可见的车辆信息。如下图所示。



权限管理、用户管理属于系统管理的内容，理论上只能由系统管理员进行操作。当给某个用户赋予权限管理和用户管理功能后，该用户只能查看由自己创建的用户信息，创建或编辑用户时，只能赋予下属用户自己已有的权限。

|  |  |  |
| --- | --- | --- |
| IOV\_UserData（用户数据） | | |
| 名称 | 类型 | 描述 |
| UserId | long | 用户ID |
| DataId | long | 数据ID，机构或车辆，根据类型而定 |
| DataType | tinyint | 数据类型，需定义枚举，如1：机构，2：车辆 |
| CreatedDate | datetime | 创建时间 |
| CreatedBy | long | 创建人ID |
| ModifiedDate | datetime | 修改时间 |
| ModifiedBy | long | 修改人ID |
| DataStatus | tinyint | 数据状态，需要定义枚举，如0：正常，1：禁用，2：注销，3：删除 |

### 3.3.4 基础资料

#### 3.3.4.1 终端

终端作为一类独立的设备进行管理，信息包括：

|  |  |  |
| --- | --- | --- |
| IOV\_GpsTerminal（终端信息） | | |
| 名称 | 类型 | 描述 |
| Id | long | 主键ID |
| Name | nvarchar(50) | 名称 |
| GpsNum | nvarchar(50) | 终端号 |
| Vip | nvarchar(20) | 通讯号 |
| SimId | long | SIM卡ID（可以不填，表示未启用SIM卡管理功能） |
| SimNum | nvarchar(20) | 安装的SIM卡号（如果是选择的SIM卡，则卡号和ICCID 均来自对应的SIM卡信息，无法编辑） |
| Iccid | nvarchar(20) | SIM卡的ICCID号 |
| Manufacturer | nvarchar(50) | 终端厂家 |
| VideoDeviceId | nvarchar(20) | 视频ID |
| TypeCode | nvarchar(20) | 终端类型编码 |
| CreatedDate | datetime | 创建时间 |
| CreatedBy | long | 创建人ID |
| ModifiedDate | datetime | 修改时间 |
| ModifiedBy | long | 修改人ID |
| DataStatus | tinyint | 数据状态，需要定义枚举，如0：正常，1：禁用，2：注销，3：删除 |

|  |  |  |
| --- | --- | --- |
| IOV\_GpsTerminal\_Type（终端类型） | | |
| 名称 | 类型 | 描述 |
| Id | long | 主键ID |
| Name | nvarchar(50) | 类型名称 |
| Code | nvarchar(20) | 类型编码 |
| Remarks | nvarchar(100) | 类型描述 |
| CreatedDate | datetime | 创建时间 |
| CreatedBy | long | 创建人ID |
| ModifiedDate | datetime | 修改时间 |
| ModifiedBy | long | 修改人ID |
| DataStatus | tinyint | 数据状态，需要定义枚举，如0：正常，1：禁用，2：注销，3：删除 |

#### 3.3.4.2 车辆

车辆信息仅保存车辆的基础信息，如果有发动机、能源类型、电池类型等相关详细信息，需要新增扩展属性表。

|  |  |  |
| --- | --- | --- |
| IOV\_Vehicle（车辆基本信息） | | |
| 名称 | 类型 | 描述 |
| Id | long | 主键ID |
| VehNum | nvarchar(30) | 车辆编号 |
| LicenseNum | nvarchar(30) | 车牌号 |
| LicenseColor | nvarchar(20) | 车牌颜色 |
| TerminalId | long | 绑定终端的ID |
| OrgId | long | 所属部门ID |
| Vin | nvarchar(50) | 车架号 |
| Brand | nvarchar(50) | 品牌 |
| Manufacturer | nvarchar(50) | 厂商 |
| VehColor | nvarchar(20) | 车身颜色 |
| CreatedDate | datetime | 创建时间 |
| CreatedBy | long | 创建人ID |
| ModifiedDate | datetime | 修改时间 |
| ModifiedBy | long | 修改人ID |
| DataStatus | tinyint | 数据状态，需要定义枚举，如0：正常，1：禁用，2：注销，3：删除 |

|  |  |  |
| --- | --- | --- |
| IOV\_Vehicle\_Details（车辆详情，预留） | | |
| 名称 | 类型 | 描述 |
| VehId | long | 主键ID，同车辆表一致 |
|  |  |  |
|  |  |  |
| CreatedDate | datetime | 创建时间 |
| CreatedBy | long | 创建人ID |
| ModifiedDate | datetime | 修改时间 |
| ModifiedBy | long | 修改人ID |
| DataStatus | tinyint | 数据状态，需要定义枚举，如0：正常，1：禁用，2：注销，3：删除 |

#### 3.3.4.3 SIM卡

SIM卡作为车联网系统中一个基础资料进行管理，提供数据管理、套餐管理、交费提醒、费用查询等功能。包括以下内容：

**SIM卡运营商**

普通SIM卡或专用物联网卡的网络通讯服务的通信服务商。

|  |  |  |
| --- | --- | --- |
| IOV\_SimSupplier（SIM卡运营商） | | |
| 名称 | 类型 | 描述 |
| Id | long | 主键ID |
| Name | nvarchar(100) | 运营商名称 |
| ContactName | nvarchar(50) | 联系人 |
| ContactPhone | nvarchar(30) | 联系电话 |
| Address | nvarchar(200) | 办公地址 |
| OrgId | long | 组织ID，为顶级组织 |
| CreatedDate | datetime | 创建时间 |
| CreatedBy | long | 创建人ID |
| ModifiedDate | datetime | 修改时间 |
| ModifiedBy | long | 修改人ID |
| DataStatus | tinyint | 数据状态，需要定义枚举，如0：正常，1：禁用，2：注销，3：删除 |

**SIM卡套餐**

SIM卡套餐用于描述SIM卡的分类、资费、流量等信息。

|  |  |  |
| --- | --- | --- |
| IOV\_SimTariffPackage（SIM卡资费套餐） | | |
| 名称 | 类型 | 描述 |
| Id | long | 主键ID |
| Name | nvarchar(100) | 名称 |
| SupplierId | long | 所属运营商名称 |
| Price | decimal(10,2) | 月资费 |
| Flow | decimal(10,2) | 月流量大小(M) |
| PaymentMethod | tinyint | 交费方式，需定义枚举，1：按月交费，2：半年交费，3：无固定周期 |
| PaymentDay | tinyint | 交费日，如每月1号 |
| OrgId | long | 组织ID，为顶级组织 |
| CreatedDate | datetime | 创建时间 |
| CreatedBy | long | 创建人ID |
| ModifiedDate | datetime | 修改时间 |
| ModifiedBy | long | 修改人ID |
| DataStatus | tinyint | 数据状态，需要定义枚举，如0：正常，1：禁用，2：注销，3：删除 |

**SIM卡信息**

提供SIM信息的管理，以及SIM交费记录查询。

|  |  |  |
| --- | --- | --- |
| IOV\_SimInfo（SIM卡信息） | | |
| 名称 | 类型 | 描述 |
| Id | long | 主键ID |
| SimNum | nvarchar(20) | 卡号 |
| Iccid | nvarchar(20) | 串号 |
| UseState | tinyint | 使用状态，需定义枚举，0：启用，1：禁用 |
| OpenDate | datetime | 开通日期 |
| PackageId | long | 套餐ID |
| PaymentDate | datetime | 最后一次交费日期 |
| OrgId | long | 组织ID，为顶级组织 |
| CreatedDate | datetime | 创建时间 |
| CreatedBy | long | 创建人ID |
| ModifiedDate | datetime | 修改时间 |
| ModifiedBy | long | 修改人ID |
| DataStatus | tinyint | 数据状态，需要定义枚举，如0：正常，1：禁用，2：注销，3：删除 |

SIM卡管理界面如下图所示：



可按照运营商、套餐、卡号、ICCID、状态、开通日期（默认所有）进行查询，查询结果包括：卡号、ICCID、开通日期、状态、运营商、套餐、资费、交费方式、交费日、交费日期（无日期为未交费）。

操作列包括3项内容：编辑、删除、交费、交费记录，最好使用图标加提示的形式，避免按钮文字较长导致界面不友好。

交费记录界面按时间顺序显示对应卡的最新交费信息。

SIM卡信息可通过Excel模板导入，也可以导出到Excel。

**交费记录**

|  |  |  |
| --- | --- | --- |
| IOV\_SimPayment（SIM卡交费记录） | | |
| 名称 | 类型 | 描述 |
| Id | long | 主键ID |
| SimId | long | SIM卡ID |
| PayTime | datetime | 交费日期 |
| PayAmount | decimal(10,2) | 交费金额 |
| CreatedDate | datetime | 创建时间 |
| CreatedBy | long | 创建人ID |
| ModifiedDate | datetime | 修改时间 |
| ModifiedBy | long | 修改人ID |
| DataStatus | tinyint | 数据状态，需要定义枚举，如0：正常，1：禁用，2：注销，3：删除 |

交费记录提供SIM卡的交费记录明细查询，查询条件包括：运营商、套餐、卡号、开始及结束时间。查询结果包括：卡号、交费日期、交费金额、录入人、录入日期。结果可导出Excel。

**费用统计**

查询条件：运营商、套餐、SIM卡号、开始及结束日期。

查询结果以资费套餐费分组，显示费用统计情况，包括：运营商、套餐、资费、卡数量、启用数（启用状态）、使用数（绑定了终端）、产生费用（合计交费金额）。如下图所示。



## 3.4 实时监控子系统

服务名称：IOV.RealTime。

实时监控系统，是整个车联网平台的核心模块，用于对车辆状态进行实时查看，包括位置、状态、报警等。核心数据为车辆状态，其流程如下图所示。



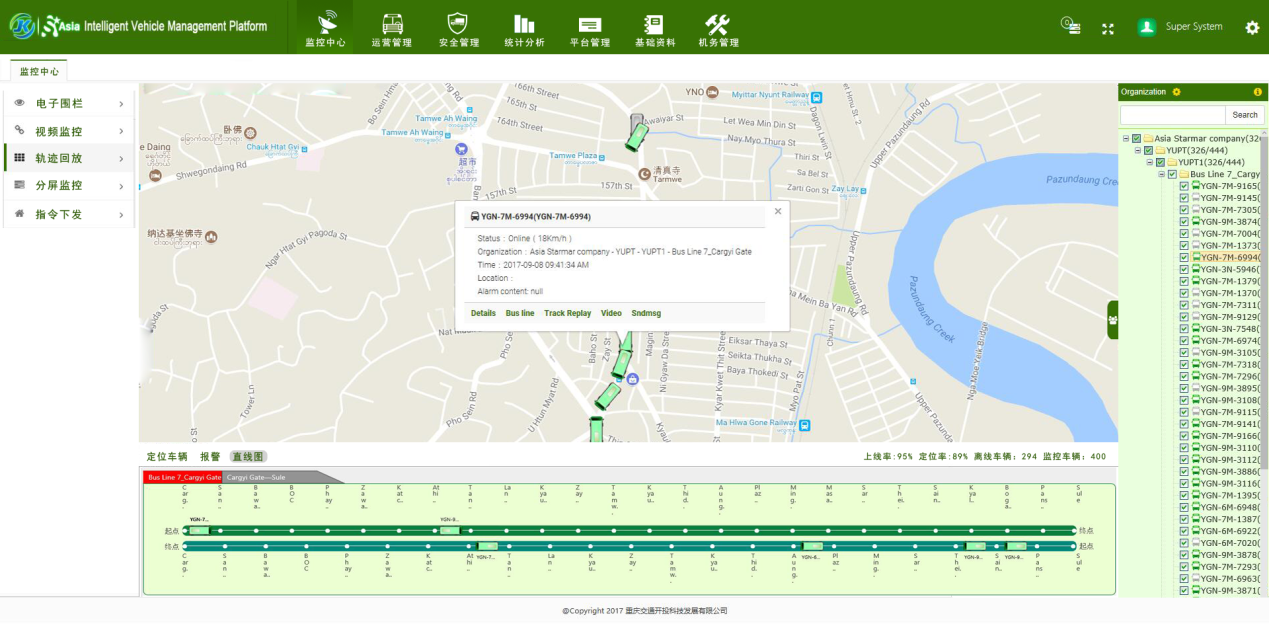
实时监控使用到的相关数据，可以从集中缓存服务器中读取；历史数据需要从数据库中查询。

### 3.4.1 监控中心

本业务系统复用现有内容，进行模块化封装，同时进行界面优化，实现车辆监控，报警处理，地图显示，车辆查看等功能。

#### 3.4.1.1 地图监控

地图界面为系统的核心部分，用于展示车辆的各种实时状态，包括位置、速度、报警等。登陆进入系统后，默认打开的就是地图监控界面，如下图所示。



地图界面包含三个部分：车辆列表、地图、状态栏。

**车辆列表**

以组织架构树的形式展示登陆账号下的所有车辆，以及车辆所属的机构、组织。组织只显示名称，车辆显示编号和车牌，形式为“编号-车牌”，没有车号的只显示车牌。树形上方有搜索框，支持按编号、车牌进行模糊查找，如果只有唯一搜索结果，直接展开树形结构，定位并选择该车辆，同时在地图上显示该车辆并弹出对应信息框；如果有多个搜索结果，分别展开树形结构到对应的车辆节点，选中所有车辆，并默认定位、选择最后一个车辆。

**地图**

主要用来显示车辆位置，并使用不同颜色的图标来区别车辆的状态，至少包括行驶、停车、离线、异常、报警五种状态。行驶表示车辆状态正常，且速度大于0；停车表示车辆状态正常，速度等于0；离线表示车辆最新时间超过最大范围；异常表示车辆不定位或GPS经纬度为0；报警标识车辆当前正处于报警或故障状态。上述五种状态，以离线为优先，即车辆离线时，不显示其他状态。车辆状态每10秒更新一次。

进入地图页面后，默认显示最新上传时间的前100辆车，少于100辆是全部显示，使地图看起来不空泛。默认显示时，车辆树不做对应选中。车辆树的状态修改后清空地图，只显示用户勾选的车辆。

单击地图上的车辆或车辆树上对应的车辆，需要弹出信息框，包括基本信息、详细信息、操作链接。基本信息包括：车号、车牌、组织（最后一级）或线路（如果有）、最新时间、速度、限速值、定位状态（是、否）、经度、纬度、报警、地理位置。详细信息包括：所在组织（形式为“A公司-B分公司-C部门”）、车辆类型（平台管理）、颜色、通讯号、SIM卡号。操作链接包括：点名、调度短信、轨迹回放、视频监控、单车跟踪。

单车跟踪需要弹出新页面，除地图跟踪定位以外，还需要展示和车辆相关的所有实时数据，包括仪表数据、车身状态、CAN数据等，以及点名、下发短信两种常用操作。这个界面需要根据实际情况进行定制，并按照车辆类型进行不同数据的展示，如电动车、燃油车的监控数据不同。

地图顶部显示常用的地图操作工具，如下图所示。



包括以下操作：

刷新：刷新地图，重新初始化地图并显示勾选车辆。

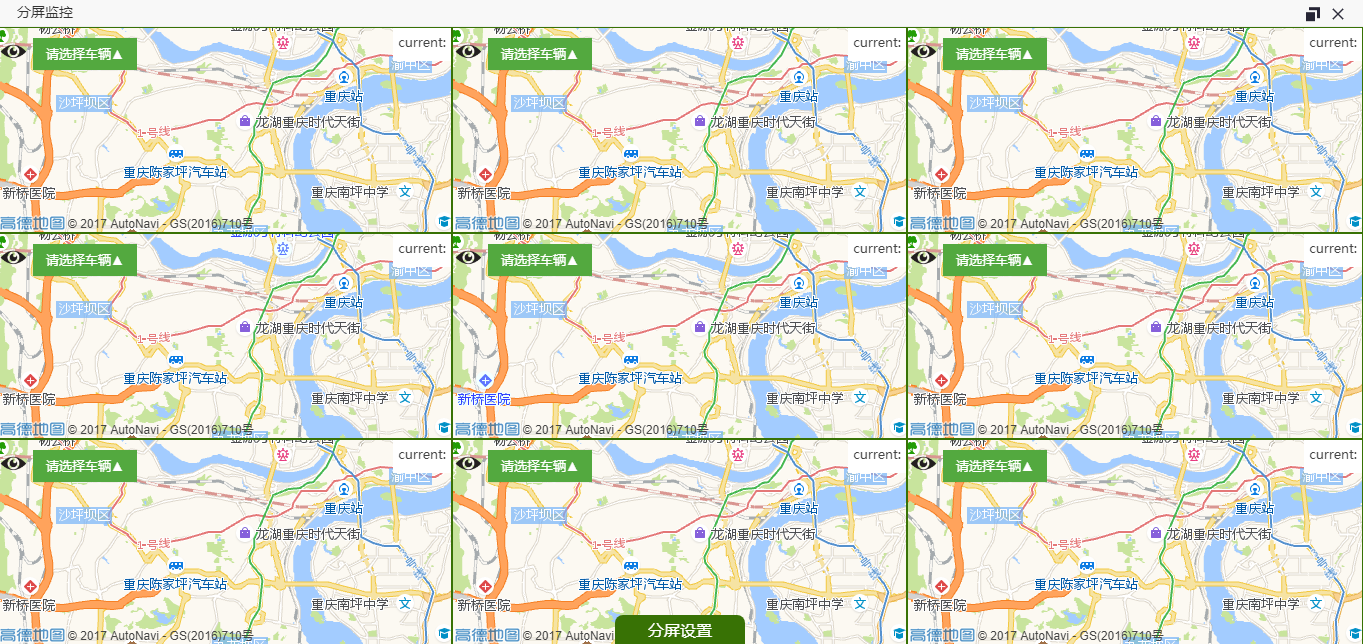
地图切换：用于多地图切换展示。

测距：地图测距功能。

区域查车：打开后，用户可以在地图界面上画电子围栏，画完后自动查询出所有在此范围内的实时车辆，并选中这些车辆，在地图上进行显示。可以清除电子围栏。

设置：用于设置地图上显示哪些内容，包括：显示标记点、显示天气、显示线路（公交产品）、站点（公交产品）。

分屏：弹出分屏监控界面，提供1\*2、2\*2、2\*3、3\*3四中显示模式，每个分屏均是一个地图界面，可显示一个或多个车辆。每个分屏可独立选择要显示的车辆。如下图所示。



**状态栏**

状态栏主要展示更多的状态及提醒信息，如报警、异常等。状态以可收缩的方式显示在地图底部，如下图所示。



状态栏包括3个部分：离线车辆、报警车辆、异常车辆、在线统计。

离线车辆：根据指定的时间范围（如20分钟，可提供设置按钮进行自定义），以列表的形式显示所有超过时间没有上线的车辆，信息包括：车号或车牌、公司、上线时间、定位时间、下线时长、报警状态、最后位置（地理位置，在线服务获取）。双击某个车辆，可进行地图定位。每一列可进行排序。

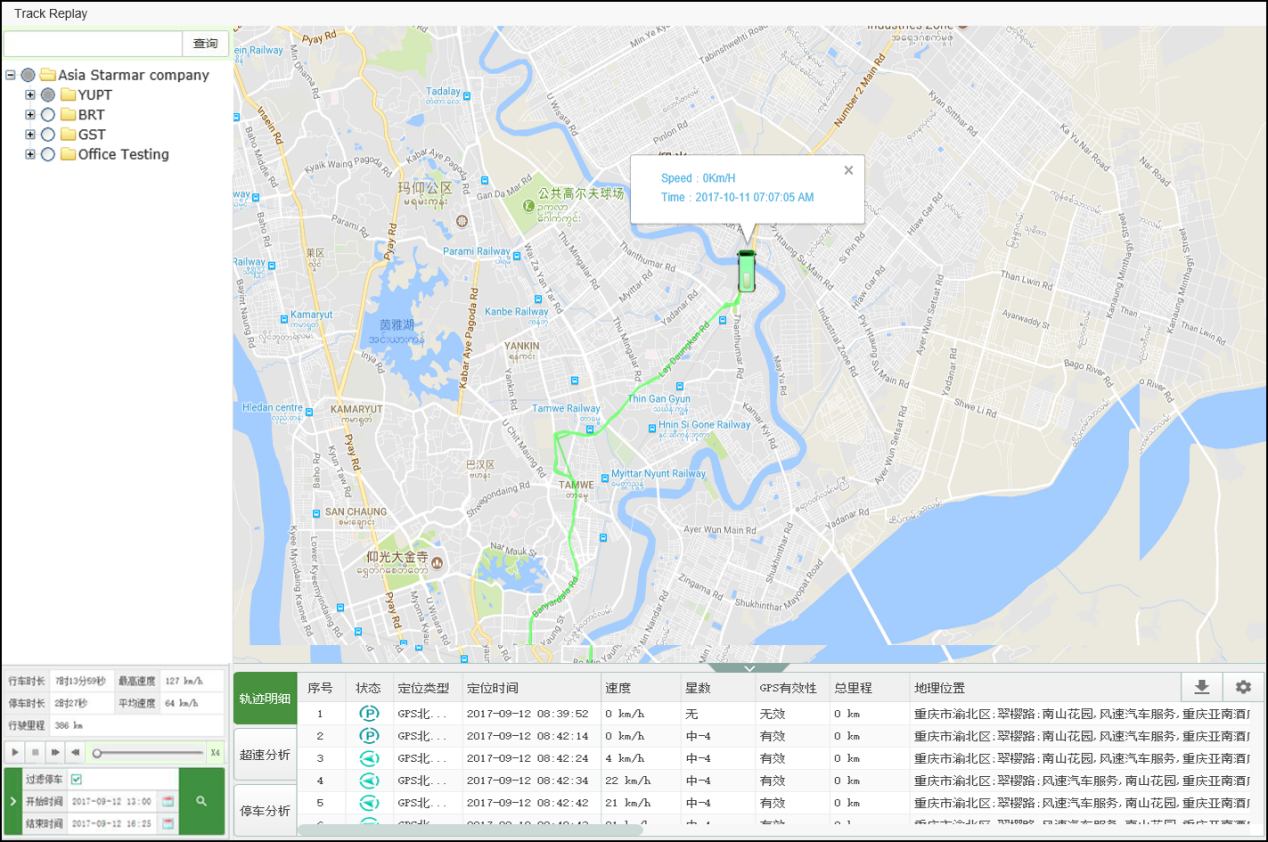
报警车辆：以列表的形式显示正在报警且未经查看的车辆，内容包括：车号或车牌、公司、报警类型、报警时间、速度、限速值、报警地点。每条记录提供“处警”操作。双击某个记录，可以在地图中定位到车辆。每一列可进行排序。处警表示监控人员已查看此报警，无需再进行显示。

异常车辆：以列表的形式显示状态异常的车辆，内容包括：车号或车牌、公司、线路、速度、最新时间、最后定位时间、异常类型。异常类型包括：无数据异常（经纬度为0）、不定位异常（定位标识为否）、终端故障。

在线统计：实时显示所有车辆的在线情况，包括在线率、定位率。

#### 3.4.1.2 轨迹回放

轨迹回放功能用于展示车辆的历史运行状态，包括行驶轨迹、行驶速度、报警状态、行为分析等。进入轨迹回放页面，展示界面如下图所示。



该界面包含4个部分：车辆列表、地图、操作面板、结果信息。

**车辆列表**：以组织架构树的形式展示登陆账号下的所有车辆，以及车辆所属的机构、组织。组织只显示名称，车辆显示编号和车牌，形式为“编号-车牌”，没有车号的只显示车牌。树形上方有搜索框，支持按编号、车牌进行模糊查找，如果只有唯一结果，直接展开树形结构，定位并选择该车辆；如果有多个结果，分别展开树形结构到对应的车辆节点，并默认定位、选择最后一个节点。

**地图**：显示车辆的历史轨迹信息，并与结果信息中的相关数据进行互动，以及按照操作进行轨迹点依次播放。查询出的轨迹点全部显示在地图上，报警点和连续报警点之间的连线用红色标识。点击某个轨迹点，弹出信息框，内容包括：时间、GPS速度、仪表速度、限速值、定位状态、报警状态、地理位置。弹出框如下图所示。



**操作面板**：

1）查询条件，包括开始时间、结束时间、是否过滤停车、查询按钮、播放、快进、暂停、停止按钮。过滤停车，表示在地图上显示轨迹时，是否加载所有速度为0的点，如果选中，则地图上不显示速度为0的点。（注意：只过滤连续的速度为0的点，假设有速度为1，“1 0 1 1”这段轨迹的0点需要显示，“1 0 0 0 1 1”这段轨迹的连续3个0点，只显示第一个0点，后续连续0点过滤。）查询出结果后，在地图上显示所有点及连续线段，每个点可以点击并查看详细信息。点击播放后，清空地图，依次显示每个点及连线。

2）结果概要：包括总的行驶时长、停车时长、最高速度、平均速度、行驶里程。GPS点速度为0，表示停车状态，连续两个点速度为0表示停车状态。行驶时长为排除停车时长后的车辆行驶时间。最高速度、平均速度计算均不考虑停车状态。行驶里程计算使用连续GPS坐标点之间的直线距离进行累加，供参考使用。（需要排除经纬度为0的点，同时需要考虑两点之间的距离，连续两段的平均时速超过120公里/小时，从第三段开始不计入里程，直到速度低于120公里/小时后重新累加。）

**结果数据**：

1）轨迹明细

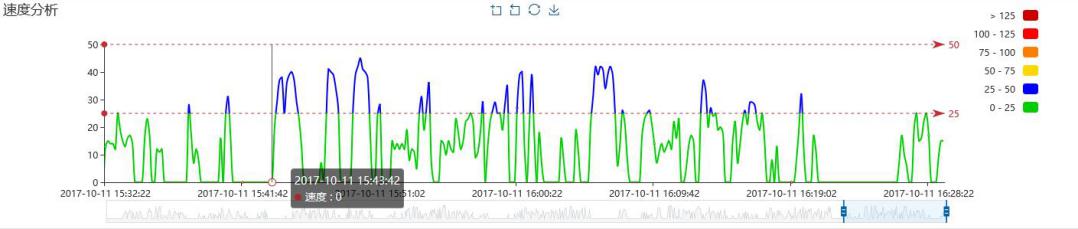
以列表形式显示查询时间内的所有轨迹点信息，按时间（GPS时间）升序排列，内容包括：时间、速度、限速值、报警（没有为空）、定位状态（是、否）、补传（是、否）、经度、纬度、角度、信号强度、卫星颗数、系统时间。在非播放状态，点击每一条记录，需要定位到对应的地图轨迹点，并弹出信息框。轨迹播放时，从选中的那个点开始播放，默认选中第一个点。轨迹明细提供导出功能，导出数据格式为Excel文件。

2）报警记录

以列表形式显示查询时间段内的报警记录，内容包括：报警类型、开始时间、结束时间、持续时间（秒）、报警等级、速度、限速值、报警地点。这个数据需要与地图进行互动，在非播放状态，点击某个记录后，地图需要定位到报警开始点。

3）速度分析

速度分析以曲线图的方式，展示查询时间段内，车辆的整个运行态势。横坐标为时间，纵坐标为速度值，包含两条线：曲线为车辆的运行速度，直线为每个点对应的限速值，超过同一时间段的曲线段用红色标识。



4）停车分析

GPS点速度为0，表示停车状态，连续两个点速度为0表示停车状态，将连续的停车点进行聚合，形成一条条的停车记录，以列表的形式显示，内容包括：开始时间、结束时间、停车时长（分钟）、停车地点（根据GPS位置通过在线服务获取）。这个数据需要与地图进行互动，在非播放状态，点击某个记录后，地图需要定位到对应的停车点。停车记录提供导出功能，导出数据格式为Excel文件。

#### 3.4.1.3 终端参数设置及查询

参数设置及查询，用于与车载终端进行交互，设置及查看终端内的参数情况。本次开发实现：调度短信、超速报警及预警设置、疲劳驾驶设置、接入参数设置。除调度短信外，其他参数均包括设置和查询两个功能。

**调度短信**

调度短信保留原有的功能，但是前端按照如下方式进行修改。



调度短信利用现有的短信下发服务器接口。

新增定时短信功能，可参考重庆公交的方式。WEB端只提供定时短信的管理功能，定时短信信息包括：短信主题、短信内容、下发时间、是否重复发送、重复间隔、结束时间，以及车辆与定时短信的关系。

**超速报警**

超速报警设置页面如下图所示。设置内容包括：超速值、超速持续时间、超速提醒次数、超速提醒间隔、超速报警语音、报警类型、是否启用。这些设置内容同时作为车辆的超速设置信息，保存数据库，每次进入此页面时，初始数据为上一次设置的信息。

查询参数为下发指令到选中的车辆，返回的数据依次填入车辆对应的行和列。参数设置时弹出参数填写窗口。

报警类型指平台报警还是终端报警。如果是平台报警，则只将参数信息保存到数据库，如果是终端报警，还需要将数据进行组合，发送到终端，并查询返回状态。参数下发、查询接口，由现有的通讯服务实现。



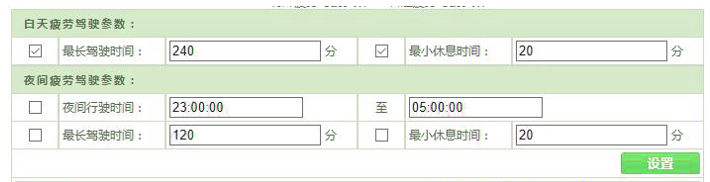


**超速预警**

超速预警和超速报警的设置内容相同，界面类似。

**疲劳驾驶**

疲劳驾驶的设置界面和超速报警类似，都是以列表的形式展示每个车辆的 设置信息，内容包括：白天最长驾驶时长（分钟）、白天最小休息时间（分钟）、夜间时段、夜间最长驾驶时长（分钟）、夜间最小休息时间（分钟）、设置类型、是否启用。界面如下。



**接入参数**

接入参数包括：主服务器地址及端口、备服务器地址及端口、心跳间隔、数据上传频率。

接入参数的界面和短信下发的界面类似。但是包含参数查询按钮，在状态列表中现实查询结果。

### 3.4.2 技术方案

#### 3.4.2.1 通用业务方案

采用循序渐进的方式在现有系统上进行升级改造，将现有的单一系统，改成服务化的方式，如下图所示。



Web Server：前端对实时监控、历史轨迹两个模块进行改造，按照上述详细设计进行开发；后端尽量复用现有的逻辑，只修改数据获取的业务模块，由直接数据库访问，改成从服务访问。Web后端不需要关注数据来源和处理方式，只需要按需获取数据，并进行前端展示。Web Server包含现有的Web和信息下发服务。

Business Service：车辆监控相关的通用业务逻辑，以Web API的方式提供数据。当前只按照前端需要，定义业务模型，不能输出原始数据，同时提供相关接口，以需求为准，格式为JSON。接口内部实现处理相关逻辑，如位置、报警、轨迹的获取以及处理，包括停车分析、速度分析等。本服务层属于内部通用业务服务，只对内部相关子系统提供服务，不对外；如果需要提供第三方服务，需要在Web Server同级增加封装接口，处理权限、有效性等验证。

Data Layer：数据访问接口，在本次开发中，暂时不封装成独立的Web服务接口，但是也需要进行独立的封装和抽象，以Business Service的需求为准，自主判断数据的来源是哪个数据库或哪个缓存服务，屏蔽数据访问细节。

本次开发需要对数据进行整理，初步分层基础数据和动态数据两个数据库，后台存储服务要进行相应的改造。动态数据包括GPS轨迹、报警、CAN相关数据等，由车辆实时上传，由于数据量大，进行单独分库处理，并按月进行分表。除动态数据外，其他数据暂时用一个库进行存储。

本次开发增加部署集中式缓存服务器，采用Redis服务。缓存内容包括：车辆基础资料、组织架构树、车辆最新状态。

**注意**：以上内容，Web Server在原车联网项目的基础上，独立解决方案进行开发，尽量复用Web前端后端逻辑；同时对本系统的相关业务，采用服务方式调用，但是涉及到其他子系统的内容时，暂时不修改，待其他模块服务化后同步改进。其他内容，均独立到另外项目中开发，但是部分业务逻辑可以服务现有的内容。

#### 3.4.2.2 后台服务设计

由于需要对各种数据进行存储、监测、处理，需要多个后台服务进行协调运行，所有实时服务均连接到数据分发中心。

为了面向应用进行服务开发，屏蔽底层细节，需要定义统一的业务数据模型，这里主要是设备位置和状态信息，并开发统一的数据转换工具类。各实时服务接收到数据后，即调用统一的转换工具，将设备数据转换为业务模型。服务根据业务模型进行处理，而不是面向不同的设备协议，简化服务处理的里程。



本次开发设计以下服务：

**数据存储服务**

数据存储服务需要从两个方面改进：

1）实时数据现在是单表存储，需要修改为分库、分表存储，同时为了兼容现有数据，只针对2018年的数据进行分库按月存储，其他不变，相应的数据查询逻辑也相应修改。

2）数据存储服务现在还负责终端报警的判断和存储，本次对该部分进行分离，让存储服务仅做数据存储、数据回复、实时位置更新缓存操作，使其职责更单一、明确。

**报警判断服务**

报警判断包括终端报警、平台报警。报警服务除生成报警记录外，还负责维护每个车辆的最新报警信息，每个车辆一个数据，数据存储到数据库和缓存服务器。

终端报警的判断目前是在存储服务中，本次开发将其提取并封装到报警判断服务中。之前也是按照终端上传的报警类型进行处理，本次开发按照统一定义的业务模型进行处理，报警类型、报警描述等统一规划。

平台报警本次实现：平台超速、疲劳驾驶报警、禁行报警、进/出电子围栏报警。

平台超速根据平台设置的报警值，并在平台超速启用时，对车辆的实时速度进行判断，生成超速报警，同时设置车辆的状态。平台超速还需要进行预警、报警信息下发。

疲劳驾驶判断需要平台进行启用，并设置最长连续行驶时间和最小休息时间。平台对车辆的连续行驶进行时间统计，并判断疲劳驾驶报警。需要进行预警、报警信息下发。

禁行报警，由平台设置禁行时间段，当车辆在指定时间段内启动、行驶时，生成报警记录。

进/出电子围栏，结合平台设置的车辆、电子围栏绑定关系，实时判断车辆的相对位置，并生成报警。

注意：当同一个报警，终端报警和平台报警同时出现时，以平台报警为准。如超速判断时，如果终端上传了超速报警，但是设置了平台超速，忽略终端上传的超速标识。

**分段限速服务**

分段限速服务包括两个内容：区域限速、路段提醒。

区域限速：采用电子围栏的方式对车辆进行区域限速，平台通过两个设置进行控制：车辆的设置信息（是否启用分段限速、默认限速值）、区域管理（电子围栏及与车辆的绑定关系、速度设置）。对于启用分段限速的车浪，进行实时位置判断，当车辆在区域内且速度与区域限速不同时，将区域限速作为参数设置值进行下发。如果车辆不在区域内，且速度与默认速度不同，将默认速度作为参数设置进行下发。

路段提醒：平台可以规划危险路段，危险路段以电子围栏的形式体现。当关联车辆进入某个电子围栏时，通过调度短信的方式，下发短信提醒。

**数据处理服务**

本服务根据系统业务情况，进行数据处理，如公交业务相关服务。本次开发中保留现有的直线图位置判断服务，但该服务需要进行优化，一是要以统一的业务模型为输入，二是输出数据需要更新到缓存服务器。

定时短信发送服务。

### 3.4.3 对外接口

## 3.5 安全管理子系统

服务名称：IOV.SafeMgr。

本业务系统复用现有内容，进行模块化封装，同时优化电子围栏管理、报警设置、报警明细查询相关界面，增加分段限速、安全设置模块。

### 3.5.1 报警明细

提供安全报警明细查询界面，以及查看轨迹、人工核警操作。可参考重庆公交的界面。

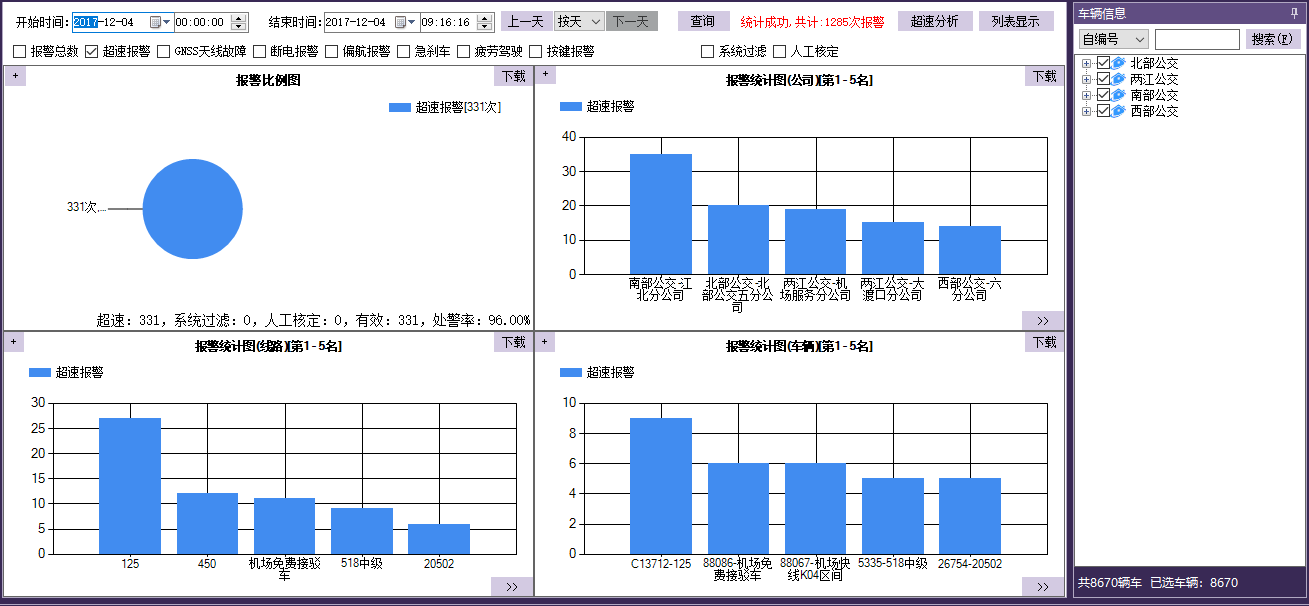


### 3.5.2 报警统计

报警统计可参考重庆公交的流程，形成如下数据报表：

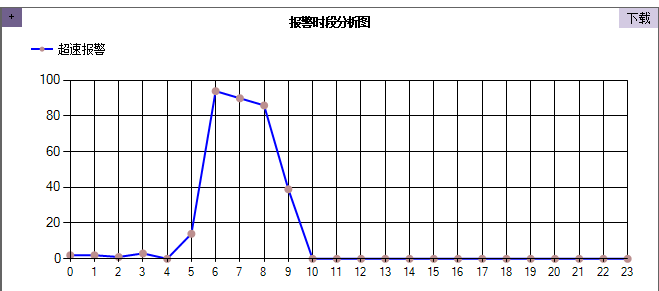
1）图形展示

类型统计：不同报警类型的数量和占比。如下图所示。



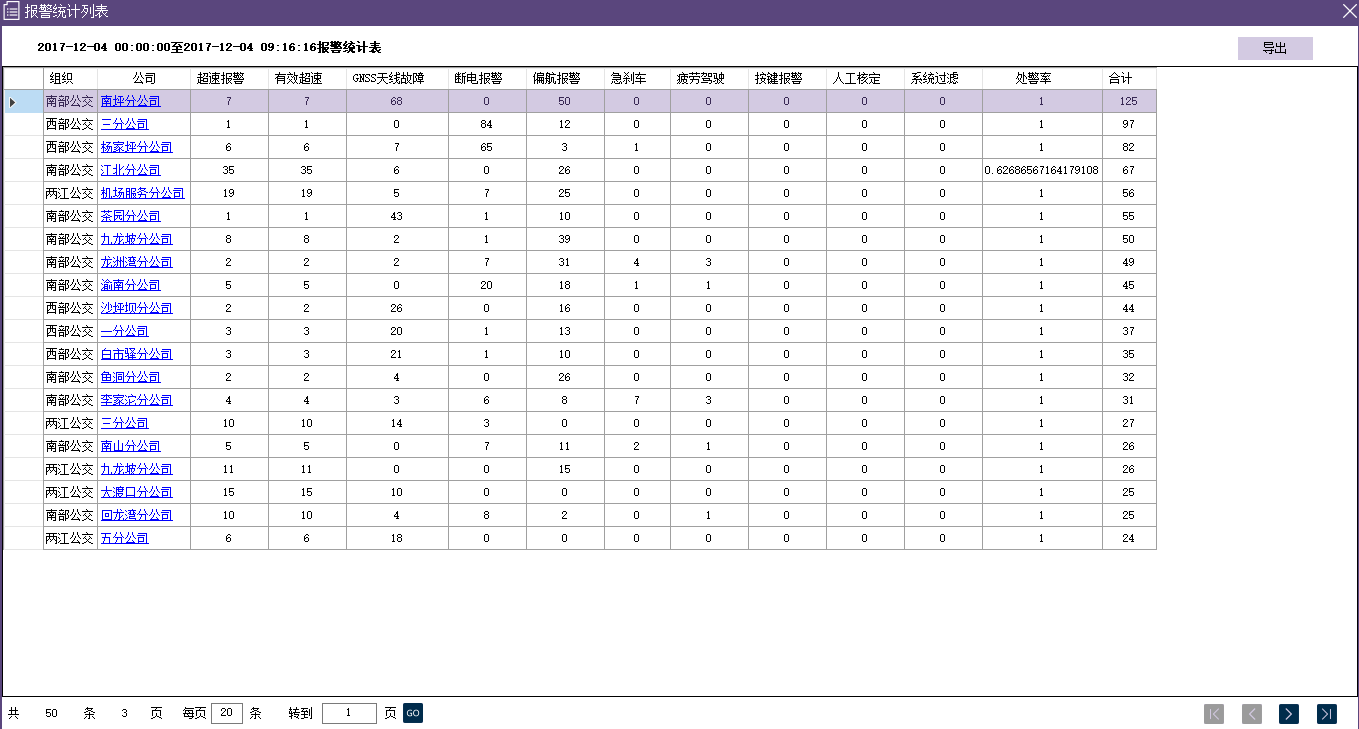
报警排名：包括组织排名、车辆排名。

趋势分析：包括查询时间段的趋势，以及24小时内的时间分布。如下图所示。



2）表格形式

表格采用三层展示，第一级展示各个组织的报警统计数据，点击每个组织后，进入组织下的车辆统计排名，单击车辆后，给出单车的报警明细列表。



### 3.5.3 电子围栏

1）电子围栏管理

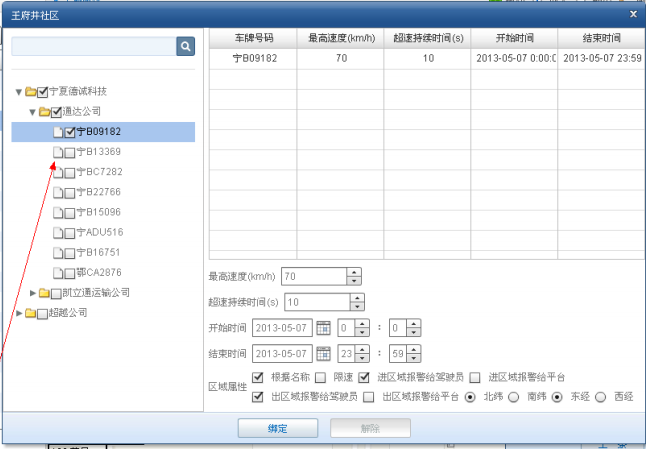
通过地图对电子围栏进行管理。



|  |  |  |
| --- | --- | --- |
| IOV\_ElectricFence（电子围栏） | | |
| 名称 | 类型 | 描述 |
| ID | long | 主键ID |
| Name | nvarchar(50) | 名称 |
| GpsPoints | varchar(MAX) | GPS坐标点集合，形式为“经度,纬度;经度,纬度;..” |
| GooglePoints | varchar(MAX) | 谷歌地图坐标点集合，形式为“经度,纬度;经度,纬度;..” |
| BaiduPoints | varchar(MAX) | 百度地图坐标点集合，形式为“经度,纬度;经度,纬度;..” |
| GaodePoints | varchar(MAX) | 高德地图坐标点集合，形式为“经度,纬度;经度,纬度;..” |
| OrgId | bigint | 所属组织的ID |
| CreatedDate | datetime | 创建时间 |
| CreatedBy | long | 创建人ID |
| ModifiedDate | datetime | 修改时间 |
| ModifiedBy | long | 修改人ID |
| DataStatus | tinyint | 数据状态，需要定义枚举，如0：正常，1：禁用，2：注销，3：删除 |

2）区域车辆绑定

区域与车辆的绑定关系可以单独或批量设置，设置内容如下图所示。



|  |  |  |
| --- | --- | --- |
| IOV\_Vehicle\_ElectricFence（车辆绑定的电子围栏） | | |
| 名称 | 类型 | 描述 |
| VehId | bigint | 车辆ID，组合主键 |
| FenceId | bigint | 电子围栏ID，组合主键 |
| ValidTime | tinyint | 是否启用有效期判断，0：否，1：是 |
| BeginTime | datetime | 有效期开始时间 |
| EndTime | datetime | 有效期结束时间 |
| LimitSpeed | tinyint | 是否限速，0：否，1：是 |
| MaxSpeed | int | 最大限速值，公里/小时 |
| NightMaxSpeed | int | 夜间最大限速值，公里/小时，0表示不启用夜间限速 |
| OverspeedDuration | int | 超速持续时间，秒 |
| InNotifyDriver | tinyint | 进区域是否报警给驾驶员，0：否，1：是 |
| OutNotifyDriver | tinyint | 出区域是否报警给驾驶员，0：否，1：是 |
| InNotifyPlatform | tinyint | 进区域是否报警给平台，0：否，1：是 |
| OutNotifyPlatform | tinyint | 出区域是否报警给平台，0：否，1：是 |
| CreatedDate | datetime | 绑定时间 |
| CreatedBy | long | 绑定人ID |

### 3.5.4 标识物

管理地图中需要显示的重要位置点。

标识物信息包括：名称、图标、经纬度、启/禁用。

新建、编辑时采用地图取点的方式，管理页面在地图显示所有的标识物。

### 3.5.5 安全设置

车辆安全设置，用于设置车辆的全局默认设置信息，包括以下内容：

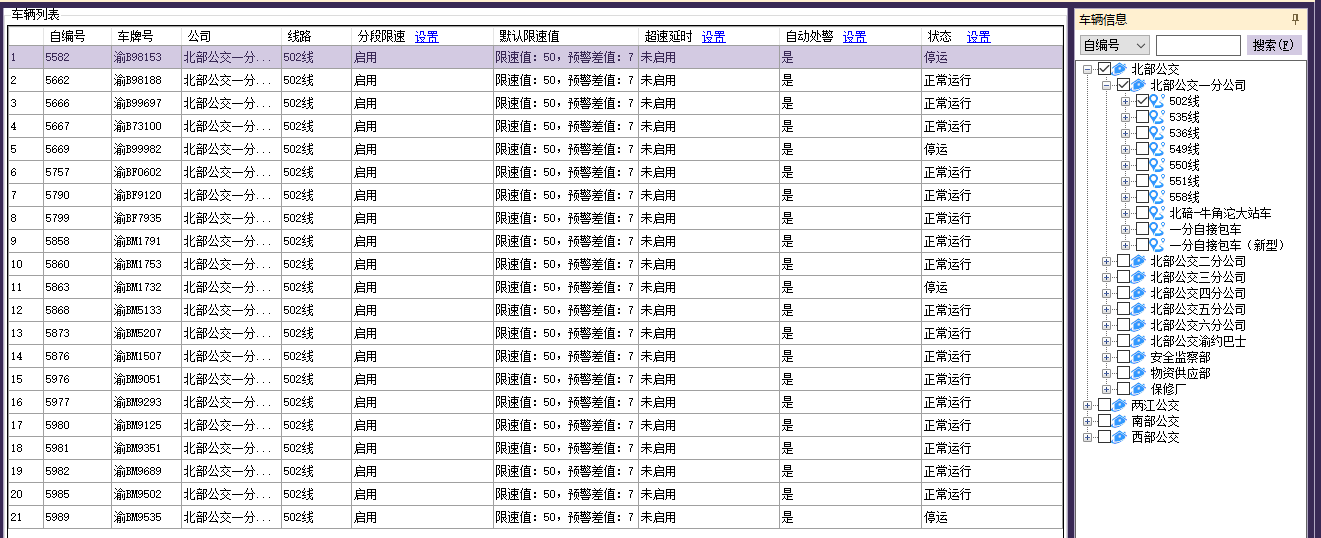
默认限速值：车辆行驶的默认限速值以及预警差值，当进行区域限速时，也用于在区域外进行限速。

是否启用分段限速：启用车辆的区域限速功能，如果没有启用，就算绑定了电子围栏，也不做处理。

是否启用夜间限速：需要设置关联的夜间时间段、限速比例（正常限速值的百分值），默认都不启用，夜间和白天限速值一样。

是否自动处警：表示当车辆超速或违规时，系统是否自动下发语音提醒，代替人工处理该次报警。

每个设置项可单独设置，也可多车同时设置。设置后，需要进行状态同步，通知相关服务进行更新。类似下图所示。



### 3.5.6 事故录入

提供安全事故录入界面，包括事故时间、地点、原因、相关车辆、人员、处理过程、处理结果，同时提供事故明细查询界面。

事故统计：从时间、组织、线路、地点维护，分别统计事故属性，如重点地点排名、易发时间段。

### 3.5.7 服务接口

本服务需要为其他子系统提供以下接口：

1）最新报警查询接口

作用：用于地图监控模块或系统首页，显示最新的报警。

参数：登陆账号TOKEN、开始时间（用于查询多长时间内的报警）、处理状态（表示所有或未处理）。

2）报警处警接口

作用：用于地图监控模块，对单条报警进行查看处理。

参数：登陆账号TOKEN、报警ID。

## 3.6 机务管理子系统

服务名称：IOV.Maintenance。

### 3.6.1 机务管理

本业务系统需要重新设计外部规格并实现。

1）车辆保养

期初数据设置，系统需要录入车辆上一次保养的时间。

保养计划设置，按运行时间、运行里程，设置车辆的保养周期。

保养提醒，对即将保养的车辆，系统进行通知和提醒，同时提供可查询界面，查看哪些车，计划保养的日期。

保养记录，提供保养录入界面，保存车辆保养记录。

记录查询，可以按照组织、线路、车辆、时间，查询报警明细。

2）故障登记

提供故障人工录入界面、明细查询界面，便于系统进行相关的统计分析。

将故障管理做成二级子系统，包括故障申报、故障分发、故障处理、专家库，形成故障的闭环管理。

3）油耗管理

提供加油录入界面，加油明细查询界面。

4）轮胎管理

轮胎作为重要的车辆配件，在有些企业作为单独物资进行管理，如永川公交。系统提供轮胎管理功能，包括：

资料管理：轮胎的基础资料维护，可查看轮胎的异动明细信息。

异动管理：轮胎的维修、更换、转移、报废等，需要登记记录。

### 3.6.2 服务接口

1）待保养车辆查询

系统其他地方会涉及到车辆保养提醒，通过此接口返回近期需要保养的车辆列表。参数包括加密后的用户TOKEN和用户ID。

## 3.7 统计分析子系统

服务名称：IOV.Statistic。

### 3.7.1 功能描述

统计分析模块主要提供数据分析功能，形成图形化的分析报表。本系统需要重新设计外部规格并实现。

1）上线/定位统计

统计实时、历史的车辆上线、定位情况，可查看组织的上线/定位情况，可查看单车的上线/定位情况，可查看上线、下线、未定位明细，便于找出终端故障车辆。

2）油耗统计

统计油耗数据，结合车辆运行里程、时间等信息，分析车辆的百公里平均油耗，能耗排名。

3）报警统计

统计报警属性，形成报警比例、报警分布分析，给出重点部门、重点车辆、重点路段信息，以及报警趋势、时间分布。

4）故障统计

统计故障属性，分析重点车辆、重点车型、重点部件，以及故障处理情况。

5）CAN数据分析

按照国标32960，对其中近50项标准CAN数据进行统计分析，如电池温度、电压、SOC等。

6）安全事故统计

从时间、组织、线路、地点维护，分别统计事故属性，如重点地点排名、易发时间段。

### 3.7.2 计算服务

统计分析子系统提供业务数据计算服务，为各模块的统计和显示提供基础数据支撑。

#### 3.7.2.1 里程统计

里程统计通过计算GPS轨迹点的连续里程获得，由于存在补传、错传、时间交叉等异常现象，无法进行实时统计，因此采用后期定时结算的方式获取车辆里程。本系统暂定按日结算的方式，每天1:30定时结算前一天的车辆里程。

另外，GPS轨迹数据量大，车辆多，不利于结算，还会对实时或历史数据查询请求产生影响。为提高结算速度，同时不对数据库产生影响，采用分离数据库的方式进行。

存储服务在存储GPS轨迹数据时，将部分信息存储到独立的结算数据库中的里程结算表，信息包括车辆ID、GPS时间、经度、纬度、是否定位，结算服务计算完前一天的里程并保存后，清空对应的数据。这样，里程结算表中，最多存储26小时的数据，查询效率比从历史数据库结算要高，同时不对正式库产生影响。

行驶里程计算使用连续GPS坐标点之间的直线距离进行累加，供参考使用。算法如下：

从最新位置表中，读取到当天上线过的车辆，依次从里程结算表中读取其前一天的所有轨迹点，并按GPS时间升序排序。依次计算两点之间的距离和时间间隔，得出两点之间的平均速度。如果平均速度小于120公里/小时，则计入里程统计。如果连续两段的平均时速超过120公里/小时，从第三段开始不计入里程，直到速度低于120公里/小时后重新累加。

同时，里程结算表中的临时数据，还可以用来计算车辆的上线、定位情况，内容包括数据点总数、定位点总数、定位率。通过这些数据，可以统计历史上线率、历史定位率、定位异常报警。

具体结算服务中，可以通过多车并行、多线程等形式，加快结算速度。

里程结算表结构如下：

|  |  |  |
| --- | --- | --- |
| IOV\_Settlement\_Mileage | | |
| 名称 | 类型 | 描述 |
| VehId | long | 车辆Id，主键 |
| DataDate | datetime | 数据日期，主键 |
| Mileage | decimal(10,2) | 里程，米 |
| PosCount | int | 当日上传的GPS点个数 |
| PosLocatedCount | int | 当日上传的GPS点中定位点的个数 |
| LocatedPercent | smallint | 定位率，定位点数的占比 |
| SettlementTime | datetime | 结算时间 |

#### 3.7.2.2 报警统计

由于安全管理系统等模块中存在多个报警统计、分析功能，如果所有操作都是基于原始报警明细表，那么随着系统的运行，报警量会持续增加，报警统计相关的操作会越来越慢。为加快报警统计的速度，采用按车按日进行结算的方式，减少数据量，同时结算表单独数据库存储，提高效率。

报警定时结算的时间为每天1:00，结算时间间隔可以根据具体业务进行设置。因为报警涉及到用户核定的问题，如用户可以针对2天内的报警进行审核，那么报警结算的时间间隔应该大于2天，确保结算数据的有效性。

报警结算周期为1天，以车为单位、以类型为分组，分别计算出每个车当天的报警总数、处警数、有效报警数、人工核定数、各种类型的报警数。没有报警的车辆不存储到结算表中。

报警结算表结构如下：

|  |  |  |
| --- | --- | --- |
| IOV\_Settlement\_Alarm | | |
| 名称 | 类型 | 描述 |
| VehId | long | 车辆Id，主键 |
| DataDate | datetime | 数据日期，主键 |
| AlarmCount | smallint | 报警总数 |
| HandledCount | smallint | 处警数 |
| ValidCount | smallint | 有效报警数，排除系统核定和人工核定 |
| UserVerifiedCount | smallint | 人工核定数 |
| Overspeed | smallint | 超速报警 |
| TiredDrive | smallint | 疲劳驾驶报警数 |
| InoutRegion | smallint | 进出区域报警数 |
| … | … | 其他类型报警，按需添加 |

#### 3.7.2.3 运营统计

运营统计属于公交业务模块，通过运营结果明细信息，定时结算头一天的车辆运营情况，以车辆为单位，统计其班次、里程等信息。运营结算表如下。

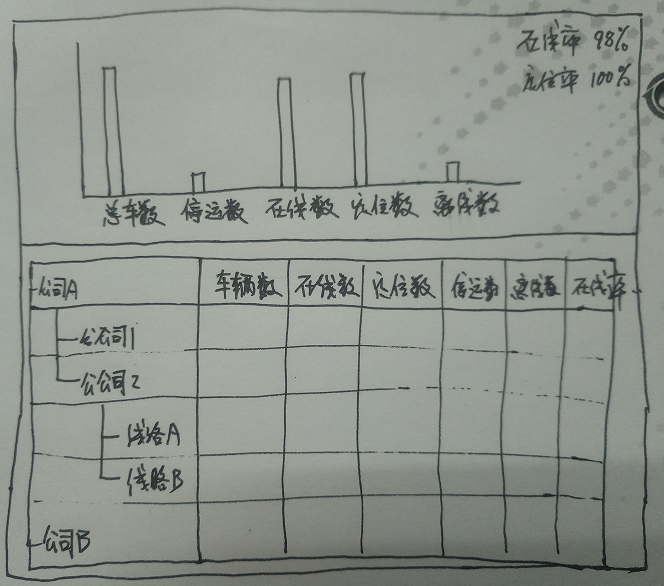
|  |  |  |
| --- | --- | --- |
| IOV\_Bus\_Daily\_Report | | |
| 名称 | 类型 | 描述 |
| VehId | long | 车辆Id，主键 |
| OperateDate | datetime | 营运日期，主键 |
| PlanClassNum | smallint | 计划班次 |
| PlanMileage | decimal(12,2) | 计划里程 |
| RealClassNum | smallint | 实际班次 |
| TotalMileage | decimal(12,2) | 总里程 |
| OnScheduleNum | smallint | 正班班次 |
| OnScheduleMileage | decimal(12,2) | 正班里程 |
| OffScheduleNum | smallint | 踏班班次 |
| OffScheduleMileage | decimal(12,2) | 踏班里程 |
| EmptyMileage | decimal(12,2) | 空驶里程 |
| RentMileage | decimal(12,2) | 空驶里程 |
| PeakClassNum | smallint | 高峰班次 |
| PunctualClassNum | smallint | 准点班次 |
| FuelConsumption | decimal(10,3) | 油耗 |
| Money | decimal(10,3) | 收入金额 |

### 3.7.3 上线统计

1）实时在线统计

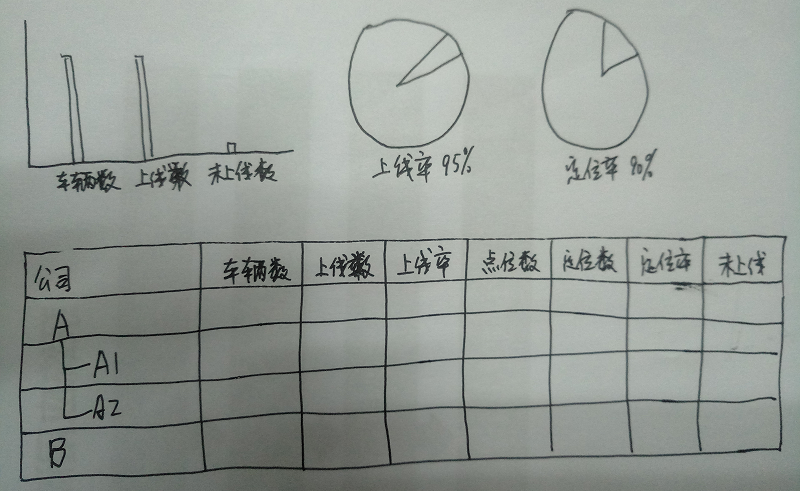
实时在线统计用于显示所有车辆当前的在线及定位情况，终端在24小时内上传过数据到平台，认为车辆在线，24小时内上传过有效定位数据，认为车辆处于定位状态。

实时在线统计通过车辆的最新位置计算获得，通过点击状态栏的上线率、定位率弹出对应界面，如下图所示。



2）历史上线统计

历史上线数据用于统计车辆的历史上线、定位情况，是基于车辆上传的GPS点位信息进行的统计，包括上传的位置数量、定位数量等。数据来源于里程结算表。查询条件为组织结构、开始日期、结束日期，结果如下图所示。



上图中车辆数、上线数、未上线三个列均是可以链接的数字，车辆数和上线数点击后弹出明细界面，包括所属组织、车号、车牌号、点位数、定位数、定位率。未上线明细给出对应组织下，统计周期内没有任何位置上传的车辆明细，包括所属组织、车号、车牌号、最近上线时间、最近定位时间。

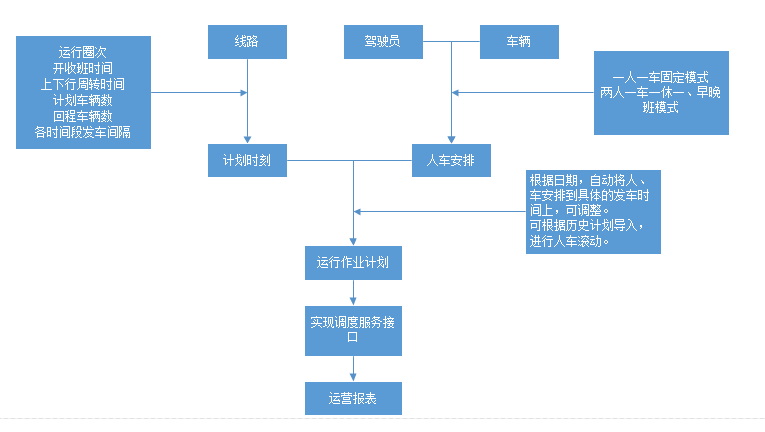
## 3.8 公交业务子系统

服务名称：IOV.Bus。

本业务系统复用现有内容，进行模块化封装，对现有数据项进行优化，同时增加运营报表分析功能。

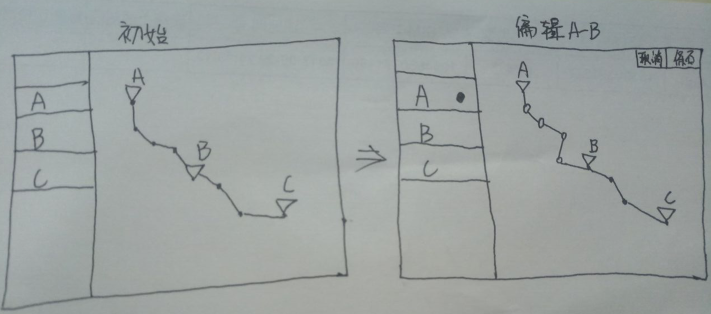
### 3.8.1 运营管理

主要业务流程如下图所示。



#### 3.8.1.1 线站管理

复用现有内容，实现线路、站点的信息管理，包括地图显示、地图取点，以及站点、线路关系的维护。



#### 3.8.1.2 计划方案管理

对线路计划时刻表进行管理，如下图所示。





时刻表的详细信息如上图所示。

需要考虑环线的计划时刻，不存在上下行的概念。

#### 3.8.1.3 人车安排

针对每条线路上的车辆，进行驾驶员绑定。

1）固定模式

一人一车。

2）两人一车

包括一休一、早晚班。一休一需要设置开始时间，甲乙班人员，甲班先上班。

#### 3.8.1.4 运行作业计划

线路每天的运营作业都需要制定相应的计划，内容主要是每个车辆、驾驶员、乘务员的行车安排，表现形式为发车时刻表。每个班次的信息包括驾驶员、乘务员、计划发车时间、计划任务。

运营作业计划制作需要选择计划模板，即计划时刻表，以及当月的人车安排情况，根据所选日期的人员安排情况，将实际的驾驶员、车辆对应到计划时刻表的代号中。同时，需要支持从历史计划导入，提供驾驶员修改或滚班操作。



#### 3.8.1.5 调度发车

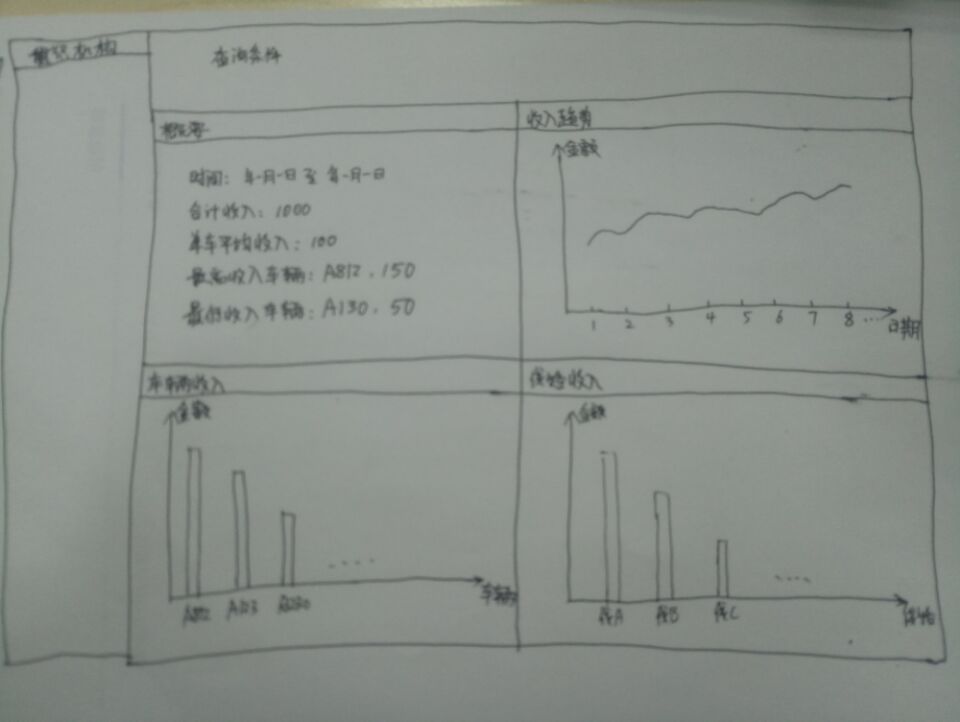
通过CS版本的轻量级公交客户端实现车辆调度功能，需要基于车联网底层数据，实现调度服务需要的接口。不在车联网项目中实现CS端。

#### 3.8.1.6 票款管理

票款录入：提供公交车辆票款录入功能，记录信息包括线路、车辆、运营日期、收入金额、录入人、录入时间。

票款明细：可以按照线路、车辆、时间条件，查询票款收入明细记录。

票款统计：收入统计分析，给出总计收入、收入排名、收入分布情况。



#### 3.8.1.7 运营报表

##### 3.8.1.7.1 运营日报

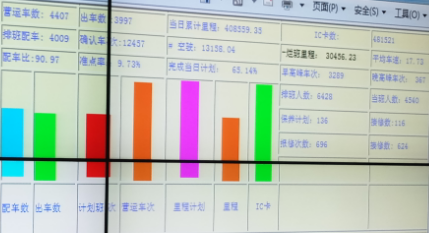
输入：公司、线路、日期。提供上一天、下一天的快捷查询方法。默认为所有公司、所有线路。

输出：结果以图形和列表两种方式展现，可进行切换。默认图形展示。

列表形式查看某一天的车辆班次统计情况，包括线路、车辆、计划班次、实际班次、总里程、空驶里程、正班班次、正班里程、踏班班次、踏班里程、包车班次、包车里程、高峰班次、班次完成率。最后需要增加合计行。

列表形式展示时，单击一个车辆的记录，可以查看该车当天的发车明细，同时提供新增、编辑功能。

图形形式展示总的运行情况，包括：配车数、排班车数、出车数、计划班次、实际班次、计划里程、实际里程、计划完成率、早高峰班次、晚高峰班次、踏班班次、踏班里程、空驶里程，各时段的发车班次曲线图。可参考下图。





##### 3.8.1.7.2 运营统计

输入：公司、线路、开始日期、结束日期。提供快捷的按日、按月、上一天、下一天、上一月、下一月的操作方式。

输出：结果展示主要的运行结果统计和不同类型的分布占比情况，以图形和列表两种方式展现，可进行切换，默认以图形方式展示。

图形形式展示的数据包括：计划出车数、实际出车数，计划总班次、实际总班次、踏班班次、早高峰班次、晚高峰班次，计划上班人数、实际上班人数，总里程、空驶里程、踏班里程。

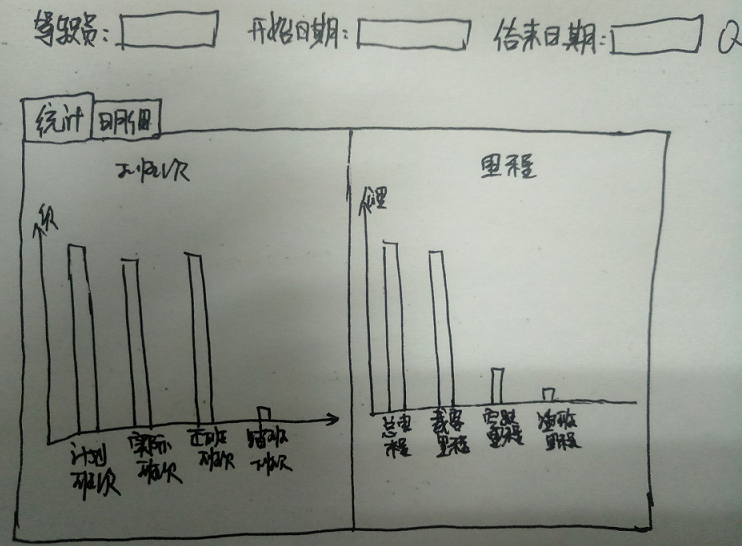
列表形式展示时可以选择统计级别，如公司、线路、车辆三个统计维度，数据包括：线路、车辆、计划班次、实际班次、总里程、空驶里程、正班班次、正班里程、踏班班次、踏班里程、包车班次、包车里程、高峰班次、班次完成率。公司和线路一级统计时，需要计算发车准点率。最后需要增加合计行。

##### 3.8.1.7.3 驾驶员统计

输入：驾驶员、开始时间、结束时间。

输出：获得指定驾驶员的运营和安全相关的数据统计和明细信息。数据统计以图形形式展示，明细以列表形式展示，默认为图形展示。

图形展示数据包括：行驶车辆数、计划内车辆数，计划班次、实际班次、正班班次、踏班班次，总里程、载客里程、空驶里程、踏班里程，报警总数、超速报警数、其他报警数、百公里报警数。



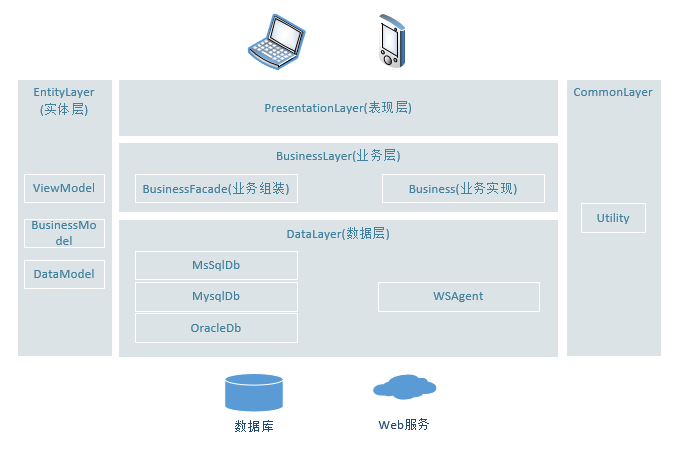
明细展示驾驶员的每个班次信息，包括：日期，线路，车辆，任务，计划时间，实际时间，开始站点，结束站点，正班里程，空驶里程，空驶类型，踏班里程，踏班原因，是否高峰，是否加班。

### 3.8.2 服务接口

# 4 项目开发规范

## 4.1 开发框架

将车联网系统进行模块化和服务化改造后，会出现多个应用系统或服务，在开发上就会出现多个解决方案。为避免各系统开发方式不一，减少开发维护学习成本，简化现有车联网系统的分层方式，需要统一所有项目的应用分层方式。初步设计基本逻辑架构如下图所示。



总体说明：

1. 上图中，每一层均在解决方案中创建对应文件夹，统一管理该层下面的所有项目。

2. 开发过程中，需要遵循分层架构的约束，禁止跨层次调用接口。

3. 下层为上层服务：以用户为中心，以目标为导向。上层（业务逻辑层）需要什么，下层（数据访问层）提供什么，而不是下层（数据访问层）有什么，就向上层（业务逻辑层）提供什么。

4. 实体层规约：Data Model 是数据表对象，不是数据访问层对象，不是只能给数据访问层使用；Business Model是内存计算逻辑对象，不是单独的业务逻辑层对象，不是只能给业务逻辑层使用。View Model是视图对象，不是视图层的内部对象，可以跨层使用。如果仅限定在本层访问，则导致单个应用内大量没有价值的对象转换。以用户为中心来设计实体类，可以减少无价值重复对象和无用转换。

5. 项目名称的命名规则是：{产品线英文名}.{子系统英文名}.{项目职责英文名全称}，如：IOV.Authrize.Business，表示IOV车联网系统，认证授权服务，业务逻辑实现项目；IOV.Base.WSAgent，表示IOV系统，基础资料子系统，数据访问层的Web服务代理。

分层说明：

1. PresentationLayer(表现层)

为用户提供服务，进行视图展示，或者前端数据的提取服务，如MVC项目、网站项目、WebService项目等。

2. BusinessLayer(业务层)

中间逻辑处理，负责业务系统业务逻辑的处理。

业务层项目命名规则为“IOV.子系统名称.Business”，业务实现类命名规则为“业务名Service”，如“CarService”，表示车辆管理业务逻辑。

3. DataLayer(数据层)

数据访问服务，负责与数据库、SOA、OpenAPI等数据提供方进行交互。

各数据库项目需要根据使用什么数据库来命名，如“IOV.Base.MssqlDb”，表示基础数据管理系统的MSSQL数据库实现项目。

如果需要访问其他服务，需要新建服务代理项目，如“IOV.Report.WSAgent”项目，表示报表模块的服务代理，用来封装其他系统提供的服务。

4. EntityLayer(实体层)

数据模型定义，包括视图模型、业务模型、数据库模型等。

项目命名方式为“IOV.子系统名称.ViewModel”，该项目包含子系统的视图模型定义。如“IOV.Platform.BusinessModel”，表示平台管理子系统的业务模型定义。

数据库模型的类名以“Entity”结尾，如“UserEntity”表示用户表对应的实体模型。

业务模型的类名以“Model”结尾，如“TokenModel”表示认证服务系统业务逻辑中的令牌模型。

视图模型的类名建议以数据的传输方向结尾，如请求参数以“Request”结尾，如SearchVehicleRequest，包括车辆查询请求需要传输的参数；响应数据以“Response”结尾，如SearchVehicleResponse表示车辆查询请求的返回结果。同时，视图模型类型需要按照功能进行分组，如根据MVC项目的Controller名称在项目内建立文件夹。

5. CommonLayer(公共层)

工具库类，负责提供应用系统中常用的操作，如字符串转换、编解码等。

## 4.2 服务接口规范

接口形式：

http://www.demo.cn:8080/api/Platform?Source=IOV.Bus.Key&Service=IOV.Platform.Key&Version=1.0&Method=func&Params=json

接口访问建议采用POST方式。

参数说明：

http://www.demo.cn:8080/api/Platform：服务的发布地址

Source=IOV.Bus.Key：发起请求的源服务Key

Service=IOV.Platform.Key：请求的目标服务Key

Version：请求的服务接口版本号

Method：请求的服务接口名称

Params：参数内容，接口自定义，json形式。

返回数据格式：

{

Result:0, //操作或查询结果，0表示成功

Message:”success” //结果消息，如出错时返回错误信息

…… //具体的结果信息

}

# 5 接口文档

本章节描述各子系统发布的服务接口，包括参数、结果的定义。

## 5.1 基础数据服务接口

测试地址：http://120.78.222.102:8080/Api/Platform

服务名称：IOV. Platform

接口版本：1

### 5.1.1 用户数据查询接口

方法名称（Method）：GetUserData

参数（Params）：{UserId:用户ID, Org=true, Veh=true}

参数说明：

UserId：用户ID；Org：是否获取组织体系，true返回Organizations数据；Veh：是否获取车辆列表，true返回Vehicles。Org、Veh为空，默认为true。

返回数据：

{

Result:0, //操作或查询结果，0表示成功

Message:”success” //结果消息，如出错时返回错误信息

Organizations:[

{Id:” ID”, Name:”名称”, ParentId:”父级ID”},

{Id:” ID”, Name:”名称”, ParentId:”父级ID”}

],

Vehicles:[

{Id:”ID”, VehNum:”编号”, LicenseNum:”车牌号”, TerminalNum:”终端号”,Vip:”通讯号”,SimCode:” SIM卡号”,OrgId:”所属组织”},

{ Id:”ID”, VehNum:”编号”, LicenseNum:”车牌号”, TerminalNum:”终端号”,Vip:”通讯号”,SimCode:” SIM卡号”,OrgId:”所属组织”}

]

}

### 5.1.2 用户权限查询接口

方法名称（Method）：GetUserPermission

参数（Params）：{UserId:用户ID, Menu=1,Permission=1}

参数说明：

UserId：请求用户的ID。

Menu：是否返回菜单权限，0：不返回。

Permission：是否返回操作权限，0：不返回。

返回数据：

{

Result:0, //操作或查询结果，0表示成功

Message:”success” //结果消息，如出错时返回错误信息

Menus:[

{Id:”ID”, Name:”名称”,Url:”地址”,SortNo:”序号”, Icon:”图标” , ParentId:”父级ID”, Translatekey:”翻译词条”},

{Id:”ID”, Name:”名称”,Url:”地址”,SortNo:”序号”, Icon:”图标” , ParentId:”父级ID”, Translatekey:”翻译词条”}

],

Permission:[

{Code:”权限编码”,Name:”权限名称”},

{Code:”权限编码”,Name:”权限名称”}

]

}

返回数据包括菜单权限和操作权限，菜单权限包括URL，用于显示系统界面；数据权限表示页面上的操作按钮是否显示，每个操作对应唯一的一个编号。

### 5.1.3 车辆信息查询接口

方法名称（Method）：GetVehicles

参数（Params）：{Organization:”组织ID”, Vehicle:”车辆ID”}

参数说明：组织ID、车辆ID为字符串形式，多个ID以英文逗号隔开。

返回数据：

{

Result:0, //操作或查询结果，0表示成功

Message:”success” //结果消息，如出错时返回错误信息

Vehicles:[

{Id:”ID”, VehNum:”编号”, LicenseNum:”车牌号”,TerminalNum:”终端号”,Vip:”通讯号”,SimCode:”SIM卡号”,OrgId:”所属组织”},

{ Id:”ID”, VehNum:”编号”, LicenseNum:”车牌号”, TerminalNum:”终端号”,Vip:”通讯号”,SimCode:” SIM卡号”,OrgId:”所属组织”}

]

}

返回数据包括菜单权限和操作权限，菜单权限包括URL，用于显示系统界面；数据权限表示页面上的操作按钮是否显示，每个操作对应唯一的一个编号。

### 5.1.4 用户查询接口

方法名称（Method）：GetUserInfoes

参数（Params）：{UserId:”Id1, Id2, Id3”}

参数说明：用户ID为字符串形式，多个ID以英文逗号隔开。

返回数据：

{

Result:0, //操作或查询结果，0表示成功

Message:”success” //结果消息，如出错时返回错误信息

Users:[

{Id:”ID” , UserType:0, Name:”名称”},

{Id:”ID” , UserType:0, Name:”名称” }

]

}

### 5.1.5 调度员查询接口

方法名称（Method）：GetBusDispatchers

参数（Params）：{OrgIds:”Id1, Id2, Id3”, Dispatcher:”Id1, Id2”}

参数说明：根据组织获取调度员列表，或者根据调度员ID获取指定的调度员信息。

返回数据：

{

Result:0, //操作或查询结果，0表示成功

Message:”success” //结果消息，如出错时返回错误信息

Dispatchers:[

{Id:”ID”, Name:”姓名” , Phone:”电话”, OrgId:”所在部门ID”, OrgName:”所在部门名称”},

{Id:”ID”, Name:”姓名” , Phone:”电话” , OrgId:”所在部门ID”, OrgName:”所在部门名称”}

]

}

### 5.1.6 驾驶员查询接口

方法名称（Method）：GetDrivers

参数（Params）：{OrgIds:”Id1, Id2, Id3”, Drivers:”Id1, Id2”}

参数说明：根据组织获取驾驶员列表，或者根据驾驶员ID获取指定的驾驶员信息。

返回数据：

{

Result:0, //操作或查询结果，0表示成功

Message:”success” //结果消息，如出错时返回错误信息

Drivers:[

{Id:”ID”, Name:”姓名”, Phone:”电话”, OrgId:”所在部门ID”, OrgName:”所在部门名称”},

{Id:”ID”, Name:”姓名” , Phone:”电话”, OrgId:”所在部门ID”, OrgName:”所在部门名称”}

]

}

### 5.1.7 操作日志记录接口

方法名称（Method）：RecordLog

参数（Params）：{User:”用户ID”,VehId:”车辆ID”,Time:”时间”,RecType:”操作类型”,Content:”内容”}

说明：记录对车辆的操作。

返回数据：

{

Result:0, //操作或查询结果，0表示成功

Message:”success” //结果消息，如出错时返回错误信息

}

## 5.2 认证授权服务接口

测试地址：http://120.78.222.102:8081/Api/Authorize

服务名称：IOV. Authorize

接口版本：1

### 5.2.1 用户登陆验证接口

方法名称（Method）：LoginUser

参数（Params）：{UserName:””,Password:””}

参数说明：用户名、密码均采用对称加密算法加密。

返回数据：

{

Result:0, //操作或查询结果，0表示成功

Message:”success” //结果消息，如出错时返回错误信息

User:{Id:”用户ID”,Name:”用户名”, UserType:0,Token:”用户令牌”}

}

### 5.2.2 用户登陆界面请求

请求地址：

参数：

参数说明：

返回数据：

### 5.2.3 用户TOKEN验证接口

方法名称（Method）：VerifyToken

参数（Params）：{Token:”用户令牌”}

返回数据：

{

Result:0, //操作或查询结果，0表示成功

Message:”success” //结果消息，如出错时返回错误信息

User:{Id:”用户ID”,Name:”用户名” , UserType:0}

}