**软件概要设计报告**

**--**成都市公交行业管理及运营分析平台

用户单位： 成都市交通运输委员会

开发单位： 成都索维思科技发展有限公司

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版本/状态 | 作者 | 参与者 | 起止日期 | 备注 |
| V0.1 | 周伟 | 王明君 | 2016年8月2日 |  |
| V1.0 | 周伟 | 刘宝林 | 2016年8月17日 |  |
| V1.1 | 周伟 | 刘宝林 | 2016年9月1日 |  |
| V1.2 | 周伟 | 刘宝林 | 2016年9月7日 |  |

**文档审阅信息**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 版本 | 审阅人 | 角色 | 审阅日期 | 确认 | 备注 |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

**文档批准信息**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 版本 | 批准人 | 角色 | 批准日期 | 确认 | 备注 |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

目录

[ 简介 6](#_Toc461010726)

[.1 编写目的 6](#_Toc461010727)

[.2 范围 6](#_Toc461010728)

[.3 定义、首字母缩写词和缩略语 6](#_Toc461010729)

[.4 文档概述 6](#_Toc461010730)

[ 系统概述 7](#_Toc461010731)

[ 设计考虑 10](#_Toc461010732)

[.1 假设和依赖 10](#_Toc461010733)

[.2 限制和约束 10](#_Toc461010734)

[.3 目标和指南 10](#_Toc461010735)

[.4 开发方法 10](#_Toc461010736)

[.5 架构设计策略 10](#_Toc461010737)

[.6 其他设计策略 11](#_Toc461010738)

[.6.1 日志 11](#_Toc461010739)

[.6.2 数据缓存 11](#_Toc461010740)

[.6.3 配置服务 11](#_Toc461010741)

[.6.4 错误处理 12](#_Toc461010742)

[.6.5 安全 12](#_Toc461010743)

[.6.6 性能 12](#_Toc461010744)

[.6.7 安装程序 12](#_Toc461010745)

[ 体系结构设计 13](#_Toc461010746)

[.1 整体结构设计 13](#_Toc461010747)

[.2 软件实现架构 14](#_Toc461010748)

[.3 硬件架构 16](#_Toc461010749)

[.4 用户接口 17](#_Toc461010750)

[.5 开发基础设施 17](#_Toc461010751)

[.5.1 开发环境配置 17](#_Toc461010752)

[.5.2 运行环境配置 17](#_Toc461010753)

[.6 错误处理 18](#_Toc461010754)

[ 系统服务 19](#_Toc461010755)

[.1 审计日志服务 19](#_Toc461010756)

[.2 使用日志服务 19](#_Toc461010757)

[.3 错误日志和追踪服务 19](#_Toc461010758)

[.4 数据库访问服务 19](#_Toc461010759)

[ 系统使用工具 20](#_Toc461010760)

[.1 启动和关闭 20](#_Toc461010761)

[.2 先进性 20](#_Toc461010762)

[.3 经济性： 20](#_Toc461010763)

[.4 集成性和可扩展性： 20](#_Toc461010764)

[.5 易维护性 20](#_Toc461010765)

[ 界面设计 21](#_Toc461010766)

[.1 界面的关系图和工作流程图 21](#_Toc461010767)

[.2 主界面 21](#_Toc461010768)

[.3 信息查询界面 22](#_Toc461010769)

[.4 指标分析界面 22](#_Toc461010770)

[.4.1 车辆监控 在线数历史查询 22](#_Toc461010771)

[.4.2 车辆监控 固定时段车辆在线数量 23](#_Toc461010772)

[.4.3 车辆监控 早晚高峰车辆数据统计 23](#_Toc461010773)

[.4.4 供应保障 线网投入分析 24](#_Toc461010774)

[.4.5 供应保障 线路投入分析 24](#_Toc461010775)

[.4.6 供应保障 车辆利用率 25](#_Toc461010776)

[.4.7 供应保障 车辆投入分析 25](#_Toc461010777)

[.4.8 供应保障 运能投入分析 26](#_Toc461010778)

[.4.9 供应保障 优先设施分析 26](#_Toc461010779)

[.4.10 供应保障 万人拥有公交标台数 27](#_Toc461010780)

[.4.11 供应保障 客流与运力配置 27](#_Toc461010781)

[.4.12 服务质量 发车/收车时间分析 28](#_Toc461010782)

[.4.13 服务质量 首末班准点率 28](#_Toc461010783)

[.4.14 线网运行 线网分析 29](#_Toc461010784)

[.4.15 线网运行 平均站距 29](#_Toc461010785)

[.4.16 政策优惠 线路首末班刷卡数据分析 30](#_Toc461010786)

[.4.17 政策优惠 站点时段刷卡客流分析 30](#_Toc461010787)

[.4.18 政策优惠 优惠分析 31](#_Toc461010788)

[.4.19 基础设备 从业人员 31](#_Toc461010789)

[.4.20 基础设备 公交车辆 32](#_Toc461010790)

[.4.21 基础设备 公交场站 32](#_Toc461010791)

[.4.22 基础设备 公交站点 33](#_Toc461010792)

[.4.23 新功能 最大发车间隔保障率 33](#_Toc461010793)

[.5 车辆GIS监控界面 34](#_Toc461010794)

[.6 线网分析界面 34](#_Toc461010795)

[.7 错误输出界面 34](#_Toc461010796)

[.8 特殊效果 36](#_Toc461010797)

[.9 美学设计 37](#_Toc461010798)

[.10 界面资源设计 37](#_Toc461010799)

[.10.1 图标资源 37](#_Toc461010800)

[.10.2 图像资源 37](#_Toc461010801)

[.10.3 界面组件 37](#_Toc461010802)

[ 数据库设计 38](#_Toc461010803)

[.1 数据库环境说明 38](#_Toc461010804)

[.2 数据库的命名规则 38](#_Toc461010805)

[.2.1 数据库对象命名规范 38](#_Toc461010806)

[.2.2 其他规范 39](#_Toc461010807)

[.3 逻辑设计 39](#_Toc461010808)

[.4 物理设计 39](#_Toc461010809)

[.5 安全性设计 40](#_Toc461010810)

[.5.1 防止用户直接操作数据库的方法 40](#_Toc461010811)

[.5.2 用户帐号密码的加密方法 40](#_Toc461010812)

[.5.3 角色与权限 40](#_Toc461010813)

[.6 优化 40](#_Toc461010814)

[.7 数据库管理与维护说明 40](#_Toc461010815)

# 简介

## 编写目的

软件概要设计报告说明对程序系统的高层设计考虑，包括程序系统的基本处理流程，程序系统的组织结构、模块划分、功能分配、接口设计、运行设计、数据结构设计和安全性设计。

## 范围

本设计以满足成都市公交行业管理及运营分析平台框架设计和开发规范，作为整个系统软件开发的纲要，随着开发的推进，文档中的内容可能会随时更新。

本文档作为项目经理、主设计人员、开发人员的编码参考文档。

## 定义、首字母缩写词和缩略语

各种图表和表格使用章节号+序号的形式表示，如：第3.4.5节的图表是第三章的第10个图表，则可以表示成“图3-10”，对应于表格则是“表 3-10”。

表 1‑1 表格标题

|  |  |  |
| --- | --- | --- |
| 类型 | 名称 | 说明 |
| String | m\_strName | 名称 |
|  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| 函数名称 | 返回值 | 参数表 | 说明 |
| Get | void | null | 根据\*\*\*获取\*\*\* |
|  |  |  |  |

## 文档概述

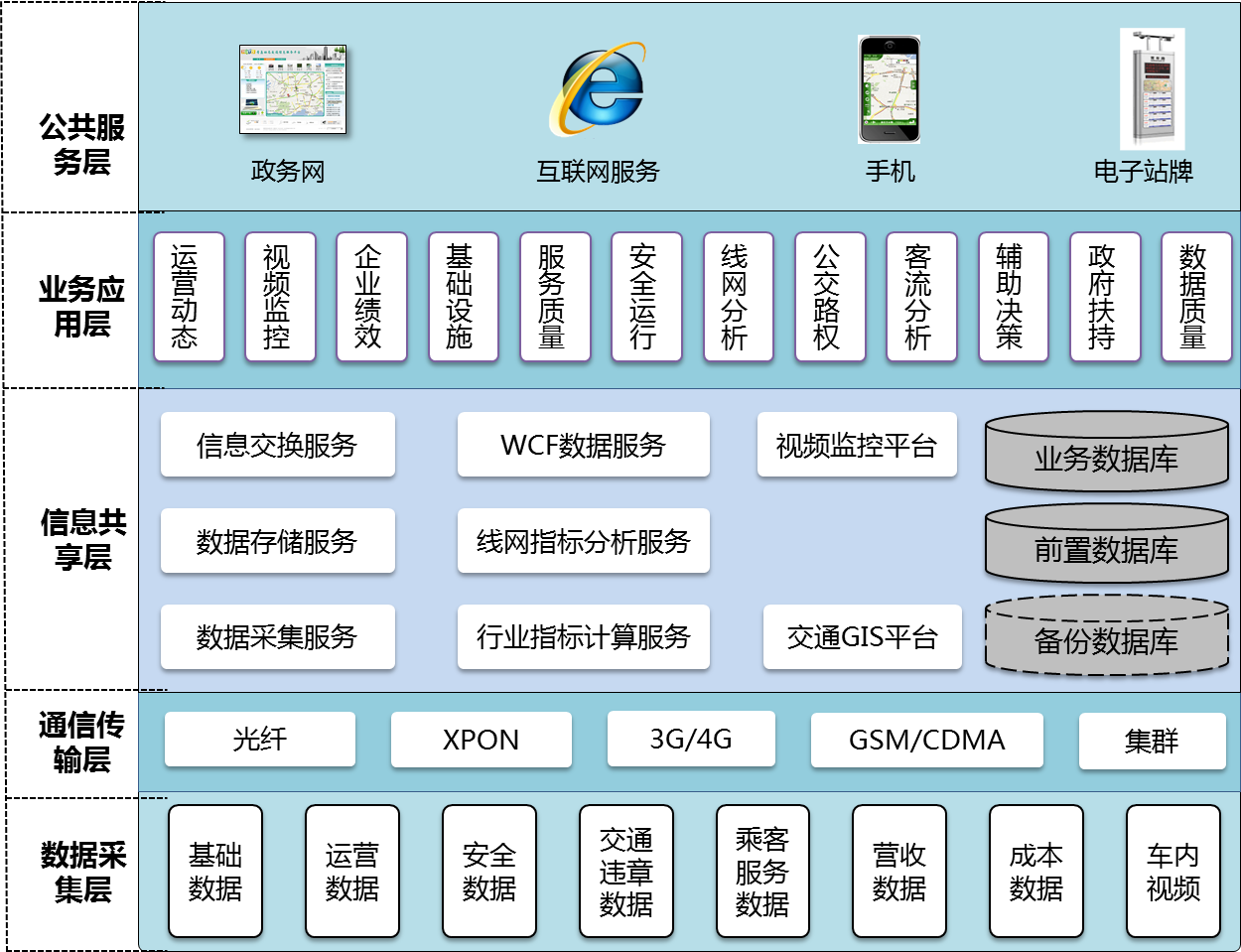
文档描述市公交行业管理及运营分析平台系统整体开发框架和设计思路，是系统开发过程中必须遵循的基本开发规范和设计构架，主要包括以下内容：

1. 系统体系结构设计：系统的整体架构设计、层次划分、子系统或模块组成、系统对外接口等
2. 系统通用方法设计：整个系统开发过程中提炼出来的通用方法、界面规范、输入输出接口、错误处理、日志处理、权限处理等
3. 数据库设计：系统数据库表设计、数据存储设计规范等
4. 其它：系统开发过程需要遵守的一些开发规范、性能要求、系统一致性要求。

# 系统概述

根据公交行业监管指标体系研究成果，将监管指标体系划分为要包括宏观监管、车辆监控、供应保障、服务质量、安全运行、线网运行、政策优惠、业务数据、系统管理和数据录入等功能模块，主要完成公交行业的宏观监管、车辆在线监控、供应保障、服务质量、安全运行、线网运行、政策优惠以及业务数据的查询，并提供纵向和横向对比分析，打印和导出相关报表及图形进行数据的统计分析。

因此，结合公交行业监管指标体系划分、系统整体设计需求，将系统划分为分5大类、19个小类、52个细类，共181项指标，如下图所示：



1. 车辆实时监控：实现车辆实时监控、在线车辆数监控、违规监控、历史轨迹回放以及车辆进出场站查询等功能。该模块从ArcGIS Server获取GIS地图服务，从通讯服务程序接收车辆位置数据、到离场站数据、违规报警数据，并在GIS地图实时显示其位置和状态。
2. 供应保障监管：实现公交行业服务供应方面的监管，主要包括设施投入、供需平衡、设施水平、设施使用等，反映城市公交行业资源投入状况。
3. 服务质量监管：实现公交行业服务质量方面的监管，主要包括服务设施质量、企业服务质量、乘客服务水平、社会评估等，反映城市公交出行的服务水平。
4. 安全运行监管：实现公交安全运行方面的监管，主要包括安全状态、安全预防、安全应急处理等，反映城市公交安全运行状况。
5. 线网运行监管：实现城市公交线网服务方面的监管，主要包括总体结构、客流特征、协调运行、路权优先等，反映城市公交线网状况和布局合理性。
6. 优惠补贴监管：实现城市公交收入、成本、优惠、补贴方面的监管，主要包括优惠政策、补贴政策、收入监管、成本监管等，反映城市公交企业收支平衡、政策补贴落实状况。
7. 基础数据管理：实现公交行业监管基础数据、业务数据以及指标分析相关的数据管理，包括监控基础数据、业务数据上传情况、录入特殊基础数据等。
8. 系统数据接口：实现公交企业数据自动上传、手动上传功能，以及数据清洗、转换、处理、入库功能。
9. 系统权限管理：实现系统用户、权限管理功能。
10. 系统应用服务：实现业务指标计算、数据结转等功能，此模块为数据库定时服务。

# 设计考虑

## 假设和依赖

* 操作系统：Windows 32位操作系统，Win 7以上版本
* 开发环境：.Net Framework 4.0
* 数据库：Oracle 10g以上版本
* GIS平台：高德地图
* 浏览器：IE 9.0以上版本
* 开发人员熟练掌握使用VS2012、C#以及Javascript编程语言

## 限制和约束

* 软件运行于Win32操作系统之上
* 需要IE 9.0或者更高版本的浏览器
* 同时在线监控车辆数小于1000台

## 目标和指南

本系统所有应用界面采用Web2.0和HTML5开发，实现基于高德地图的GIS监控功能，采用模块功能可扩展和自定制功能,采用ECharts框架实施数据可视化解决方案和大数据分析。

## 开发方法

系统采用三种开发方法，分别是面向对象的软件开发方法、可视化开发方法和软件重用和组件连接方法。

系统采用C#作为开发语言工具，C#是微软发布的一种面向对象的语言，其本身具备封装、继承和多态等面向对象的特征。

本系统所有应用界面采用Web2.0和HTML5开发，HTML5 赋予[网页](http://baike.baidu.com/view/828.htm" \t "_blank)更好的意义和结构。更加丰富的标签将随着对[RDFa](http://baike.baidu.com/view/6455260.htm)的，微数据与[微格式](http://baike.baidu.com/view/1623507.htm" \t "_blank)等方面的支持，构建对程序、对用户都更有价值的数据[驱动](http://baike.baidu.com/view/43111.htm)的Web。

系统通过对多个既有组件进行组合封装的方式，对通用模块抽取封装的方式来实现软件重用和组件连接。

## 架构设计策略

系统整体结构采用三层体系结构，分别为表示层、业务逻辑层、数据访问层。

表示层采用ASP.NET MVC框架，主要实现界面展示和简单的业务逻辑处理功能；业务逻辑层采用C#语言开发，主要实现系统业务逻辑运算，接收表示层的业务操作，并为表示层提供业务数据；数据访问层采用C#语言开发，主要实现数据库访问、实时数据接收、高德地图服务器端服务调用等功能。

表示层持久化存储用户登录信息，每次业务请求时同步传输登录信息，业务逻辑层据此登录信息进行安全认证，表示层与业务逻辑层间的数据通讯采用异步数据通讯方式。

## 其他设计策略

### 日志

系统产生的一些错误信息、调试信息以及用户操作记录，需要输出到控制台、文件、数据库中，以便开发人员或用户查看。

日志的总体设计：通过日志接口调用对应日志级别的方法，并把日志内容推放到输出适配器上去，由适配器完成具体的操作，日志接口通过单个实例向外提供服务。日志采用分级管理，可以配置日志输出级别和输出方式。日志级别分为调试、信息、sql语句、警告、错误、操作六种级别，每种级别都会对应一个日志接口方法，接口参数可以为字符串或字符串数组。日志有文件，数据库，控制台三种输出方式。

持久化存储的日志待缓存满后启动线程自动持久化。日志提供flush功能用于刷新缓存中的数据到文件或数据库中。

### 数据缓存

系统提供缓存的存取、清理接口供外部使用，由缓存提供者用来实现具体的缓存。系统提供内存缓存和本地文件缓存两种方式，内存缓存用微软的System.Web.Caching.Cache来实现，文件缓存自己实现。文件缓存利用序列化和反序列化来存储和恢复数据，对应缓存项在内存中利用哈希表映射以便快速定位，采用定时调度机制在后台扫描和清理失效的缓存。

缓存策略用来定义的优先级、失效表达式等。失效表达式主要有时间失效和文件依赖失效，根据时间或文件变化来决定缓存失效。时间失效分为绝对时间失效和相对时间失效。缓存数据用缓存项封装，包含缓存数据和对应的缓存策略，可以利用缓存策略来决定缓存数据的失效。

### 配置服务

配置服务主要实现参数配置信息读写接口，整个系统可通过该接口读取和写入相关配置信息。由于整个系统采用插件方式加载多个子系统，各子系统所需配置参数不同，因此系统需要根据要加载的功能模块来配置相应的参数。

根据配置服务获取的参数类型分强类型配置服务、弱类型配置服务两种。强类型配置服务返回参数类型是实际系统用的数值类型，弱类型配置服务返回的参数类型全部为字符串。

强类型配置服务：参数文件采用.NET应用配置文件标准格式，系统根据配置参数编写自定义参数类，实现System.Configuration.ConfigurationSection的参数读写接口，通过该类直接读写配置参数。

弱类型配置服务：参数文件采用普通INI文件格式，编写自定义类读写参数信息。

### 错误处理

采用错误统一处理的方式，数据访问层，业务逻辑层，界面层的错误不做捕获，向上层抛出，在基础页面中做错误捕获处理，界面上显示错误信息与出错类，页面，函数，行数。便于查找错误根源。

### 安全

登陆用户密码采用加密方式保证用户密码安全。采用角色权限的分别设置，管理用户使用功能的权限，保证数据、功能的安全。采用乐观锁方式控制更新操作，避免脏数据。

### 性能

采用数据缓存，多条记录分页查询提高性能。

### 安装程序

系统为Web程序，安装程序部署在服务器IIS应用服务器，客户端不需要其他安装方式。

# 体系结构设计

## 整体结构设计

根据系统整体业务划分和功能实现需求，将整个系统划分为三个层次结构：应用层、服务层、接口层，每个层次完成相应的系统功能，相互配合为系统运行提供保障。系统整体结构图如下：

表示层

模块新

新

车辆监控

线网运行

服务质量

基础设施

供应保障

宏观分析

政策优惠

业务查询

块

高德地图接口

HTTP访问接口

应用服务层

报表引擎服务

JSON数据服务

Web服务

Restful服务

数据主从复制

数据库层

缓存数据

交委数据

公交数据

1. 表示层

表示层主要面向最终用户展示界面的层，包括所有功能页面的显示，不同终端的接入。表示层为用户使用本系统提供业务操作界面，根据用户操作内容从服务层获取业务数据和服务应用，将结果呈现给用户。

1. 服务层

服务层为应用层提供业务数据和服务应用，这里多分了一个子成，HTTP服务层和地图服务层，这样可以方便的将系统移植到其他终端上去。然后应用服务层主要为无界面应用服务，包括通讯服务、存储服务、数据接口服务、应用服务。

* + 服务主要接收公交企业定时上报的基础数据、业务数据，并实时存储到中心数据库中。
  + 应用服务主要实现各种可视化报表逻辑成立，数据跳转等功能，包括业务数据的结转、业务计算等。

1. 数据层

接口层为第三方应用服务和数据库服务。

## 软件实现架构

根据系统整体结构设计以及WEB2.0开发架构设计思路，将软件实现划分为表示层、业务逻辑层、数据访问层，软件实现架构设计图如下：



1. 表示层

表示层主要实现界面展示以及简单的业务逻辑处理功能，主要包括以下几个工程项目：

* + UI界面：主要包含所有功能界面类。
  + 自定义控件：主要包含自定义的控件类。
  + 共通方法（客户端）：主要包含系统全局参数、通用接口方法和函数类。
  + WCF服务接口：主要包含数据库获取数据的业务接口。
  + 界面全局类：主要包含客户端界面控制的接口类，如界面控件验证、数据显示格式化等
  + 业务实体类：WCF服务数据传输的业务实体类，供客户端、服务端使用。
  + 实时数据处理：提供车辆实时数据接收、解包处理的类。为了实现数据通讯协议扩展需要，系统采用动态加载和接口设计方法，方便后期与第三方通讯服务协议进行通讯。

1. 业务逻辑层

业务逻辑层主要实现系统业务逻辑处理功能，主要包括以下几个工程项目：

* + 业务全局类：主要包含服务端系统全局参数和访问接口，如数据缓存、字典访问、错误和日志输出等。
  + 数据缓存类：主要包含需要缓存的数据实体，包括系统菜单、用户权限、字典参数等。
  + 共通方法（服务端）：主要包含服务端的通用接口方法和函数类。
  + 业务实体类：WCF服务数据传输的业务实体类，供客户端、服务端使用。
  + 数据访问类：主要包含数据库访问的数据接口，这些接口通过调用NUnivap类访问数据库，获取所需的业务数据，并打包成实体集合，返回给客户端。
  + GIS接口类：主要包含高德地图服务访问接口类。

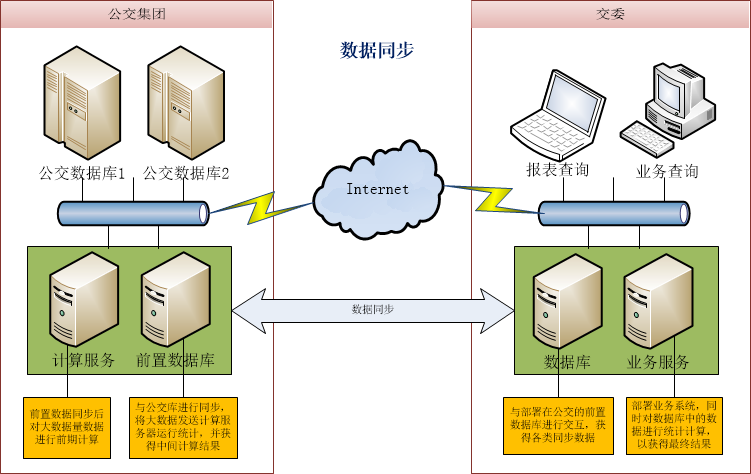
1. 数据访问层

数据访问层主要实现数据库访问、高德服务访问、数据通讯访问等。这些服务采用我公司已有产品或客户提供的第三方接口，主要包括：

* + 数据库访问：采用ORACLE ODP专用数据库访问驱动进行数据库访问，支持Oracle、SQL Server的数据库访问。
  + GIS服务：采用高德地图.net API实现后台服务接口，javascript API进行前端客户交互处理。
  + 数据通讯服务：采用我公司的信息交换服务产品，实现从公交企业接收车辆数据，并缓存在系统独立服务器上，对公交系统不会产生任何的影响。

## 硬件架构





在现有系统基础上增加前置数据库和计算服务器，将大数据量计算和传输数据在公交集团端进行预处理，交委中心进行最终数据加工。

1. 在公交数据中心增加同步前置数据库，解决前端大数据长时间访问和计算工作，只向其后端同步计算后的结果数据并在空闲时间同步需要长期保存的历史基础数据；
2. 与公交各系统约定同步周期，同步方式为中间库访问各子系统相关数据表拉取数据，时间定于每日中午12:00——13:00，中间库获得当日数据后添加记录时间戳和有效标志。基础数据则直接清空中间库相关表重新记录，评估分析系统同步时清楚原始数据后记录；
3. 评估分析系统同步时间定于每日中午23:00——0:00；
4. 评估分析系统数据计算时间定于每日凌晨2:00——8:00，多任务按计算工作量排序分别执行，遵照由简到繁、由快到慢，先执行运算量小的任务；
5. 经常性查询将查询作为任务自动执行，定时保存执行结果，减少临时查询工作量，降低资源占用风险。

## 用户接口

根据用户操作权限显示用户操作界面。界面样式和布局由系统管理员预先定制。

## 开发基础设施

### 开发环境配置

表 4‑1 开发环境配置

|  |  |  |
| --- | --- | --- |
| 类别 | 标准配置 | 最低配置 |
| 计算机硬件 | Intel I3 2.6/8G/500G | Intel I3 2.60G/4G/100G |
| 软件 | 1. Vistual Studio 2015，Win 7 2. .Net Framework 4.5 3. IIS 8.0 IE 10.0 | 1. Vistual Studio 2012，Win 7 2. .Net Framework 4.0 3. IIS 7.0 IE 9 |
| 网络通信 | 100M | 100M |
| 其它 |  |  |

### 运行环境配置

表 4‑2 运行环境配置

|  |  |  |
| --- | --- | --- |
| 类别 | 标准配置 | 最低配置 |
| 计算机硬件 | * 1. IBM3850或同等配置服务器   2. 四核2GHz Xeon处理器   3. 8G内存   4. 硬盘500G以上 | 1. IBM3650或同等配置服务器 2. 四核2GHz Xeon处理器 3. 4G内存 4. 硬盘100G以上 |
| 软件 | 1. Windows2012 64位操作系统 2. IIS8.0 以上，进程数量5个以上 3. Microsoft .NET Framework 4.5 4. Oracle 10G DBMS Client | 1. Windows2008 R2 32位操作系统 2. IIS7.0 以上，进程数量5个以上 3. Microsoft .NET Framework 4.0 4. Oracle 10G DBMS Client |
| 网络通信 | 100M | 100M |
| 其它 |  |  |

## 错误处理

采用错误统一处理的方式，数据访问层、业务逻辑层产生的错误不做捕获，向上层抛出，在表示层中做错误捕获处理，界面上显示错误信息与出错类，页面，函数，行数，并将错误信息写入中心数据库方便开发人员进行问题分析和错误排查。

为了方便错误和异常分析，系统将日志信息分为Debug、Info、SQL、Warn、Operate五个级别，开发人员在开发过程中根据需要输出相应的日志信息，并且五个级别的日志信息可根据参数配置确定是否输出到数据库表或日志文件中。

# 系统服务

## 审计日志服务

用户在操作使用本系统，系统自动记录用户登录和退出、页面进入以及关键业务操作，供系统维护人员日志审计。

## 使用日志服务

用户在操作使用本系统，系统自动记录用户登录和退出、页面进入以及关键业务操作，供系统维护人员日志查询，分析各功能页面使用频率。

## 错误日志和追踪服务

系统如实记录用户操作过程产生的系统异常、错误、SQL语句、警告、提示等日志信息，系统可配置日志记录级别，方便开发人员查找异常和错误，分析页面使用频率。

## 数据库访问服务

本系统数据库操作采用Oracle ODP开发平台访问数据库，统一数据访问接口和SQL规范，无单独的中间层应用服务。

# 系统使用工具

## 启动和关闭

WEB应用依赖于IIS来进行服务。

公交企业的数据上传服务为Windows服务程序，随Windows操作系统自动启动和关闭。公交企业部署数据上传服务程序，调用部署在成都交委信息中心的Web Service接口，由此接口负责数据入库操作。

## 先进性

充分考虑科学技术的迅猛发展趋势，选用国内外最先进的设备。实用世界先进成熟的软件架构设计，在短时期内，系统技术不会被淘汰。

## 经济性：

系统选用的设备及技术平台极为可靠，能保证系统长期正常运行。

## 集成性和可扩展性：

保证系统的集成和信息共享，总体结构具有可扩展性和兼容性。实现对系统的分散式控制，集中统一式管理及监控的目的，是整个系统可以随着技术的发展和进步，不断得到充实和提高。在考虑到目标功能多，客户多，服务对象不一样。因此系统在设计的与实施时，应多考虑客户将来的扩展需要，供应商应具备强大的技术实力，根据需求的变化，可以轻松，方便的修改系统的功能，设备的安装等来满足客户的需求。

## 易维护性

系统在运行过程中的维护极其重要，应尽量做到简单操作，系统的运转尽量做单通电后，鼠标简单操作即可启动工作，平时免维修，维护过程中无需使用过多的专用维护工具和人员

# 界面设计

## 界面的关系图和工作流程图

本系统界面关系简单，基本上在一个界面内完成所有操作。自适应各种终端设备，包括电脑，ipad和手机等移动设备也可以方便在系统中操作。

## 主界面



界面布局说明：

1. 界面上方显示系统标题及软件版本信息，右下角显示用户登录信息、当前时间。
2. 界面左边显示系统树状功能菜单，作为所有功能操作入口。其中左下方部分部署一个全屏操作按钮，可以将整个WEB界面进行全屏化。
3. 界面右边显示功能操作界面，所有查询、指标分析、各类报表等窗口都展示在此部分。

## 信息查询界面



界面上方部分为查询条件，供用户条件录入。

中间部分为工具条提供查询结果的打印、导出功能。

下方显示查询结果，支持分页显示。

## 指标分析界面

相同指标的不同分析方式采用Tab页面，每个指标分析界面布局类似于信息查询界面。

界面默认显示指标图形展示和表格数据显示。部分页面设计截屏

### 车辆监控 在线数历史查询



### 车辆监控 固定时段车辆在线数量



### 车辆监控 早晚高峰车辆数据统计



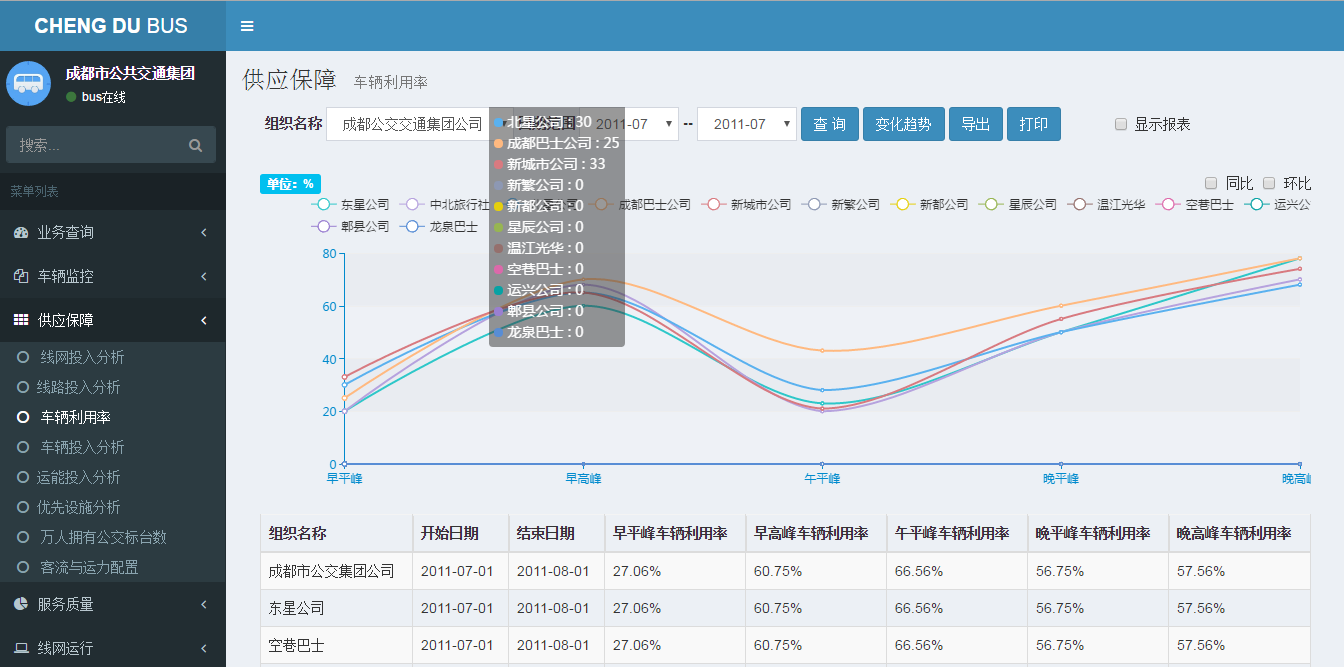
### 供应保障 线网投入分析



### 供应保障 线路投入分析



### 供应保障 车辆利用率



### 供应保障 车辆投入分析



### 供应保障 运能投入分析



### 供应保障 优先设施分析



### 供应保障 万人拥有公交标台数



### 供应保障 客流与运力配置



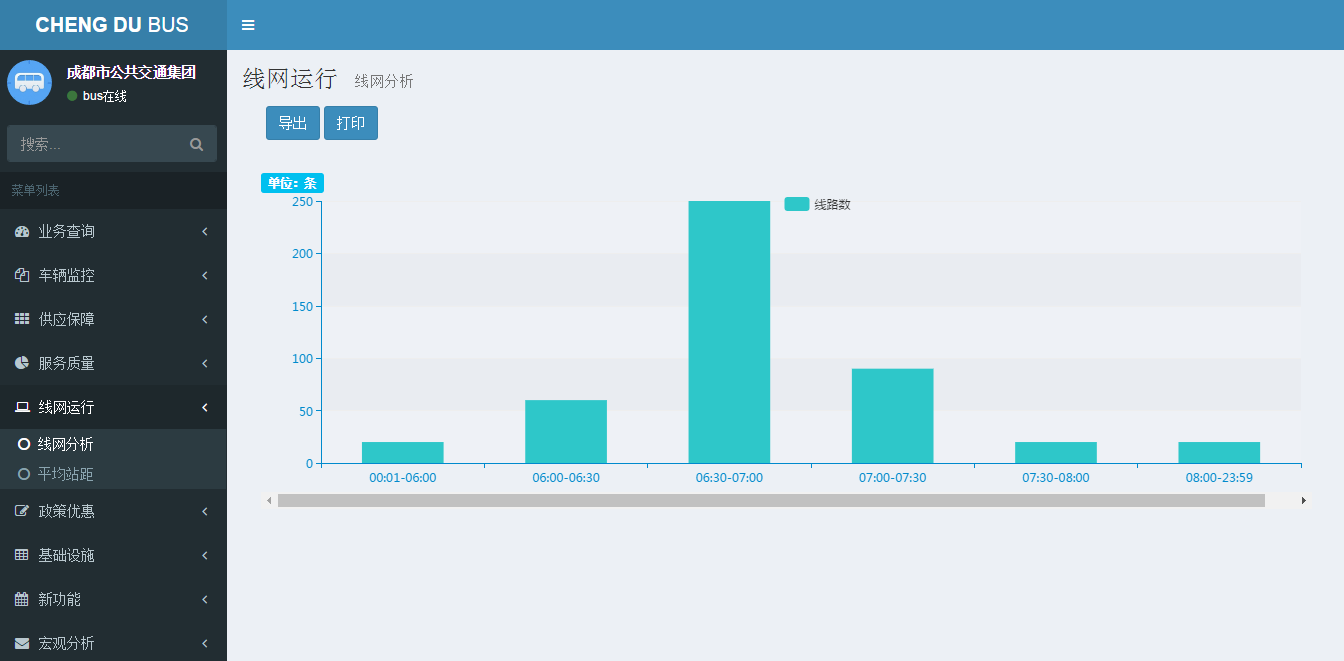
### 服务质量 发车/收车时间分析



### 服务质量 首末班准点率



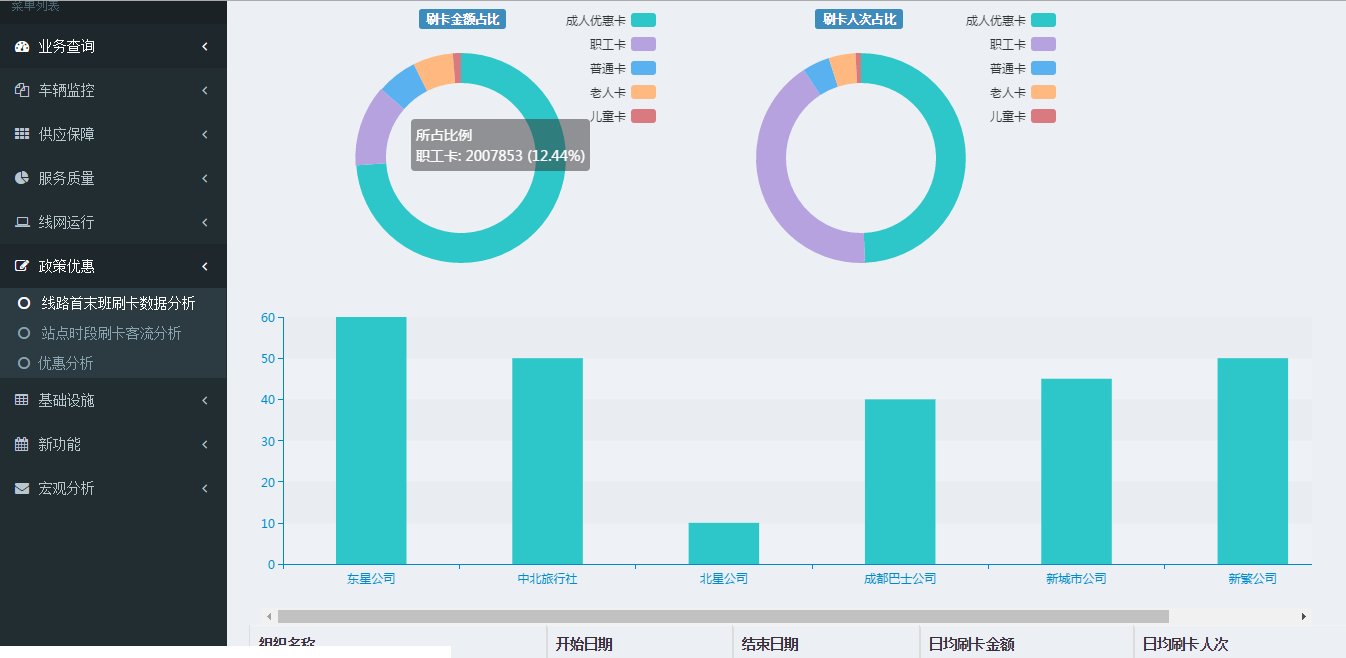
### 线网运行 线网分析



### 线网运行 平均站距



### 政策优惠 线路首末班刷卡数据分析



### 政策优惠 站点时段刷卡客流分析



### 政策优惠 优惠分析



### 基础设备 从业人员



### 基础设备 公交车辆



### 基础设备 公交场站



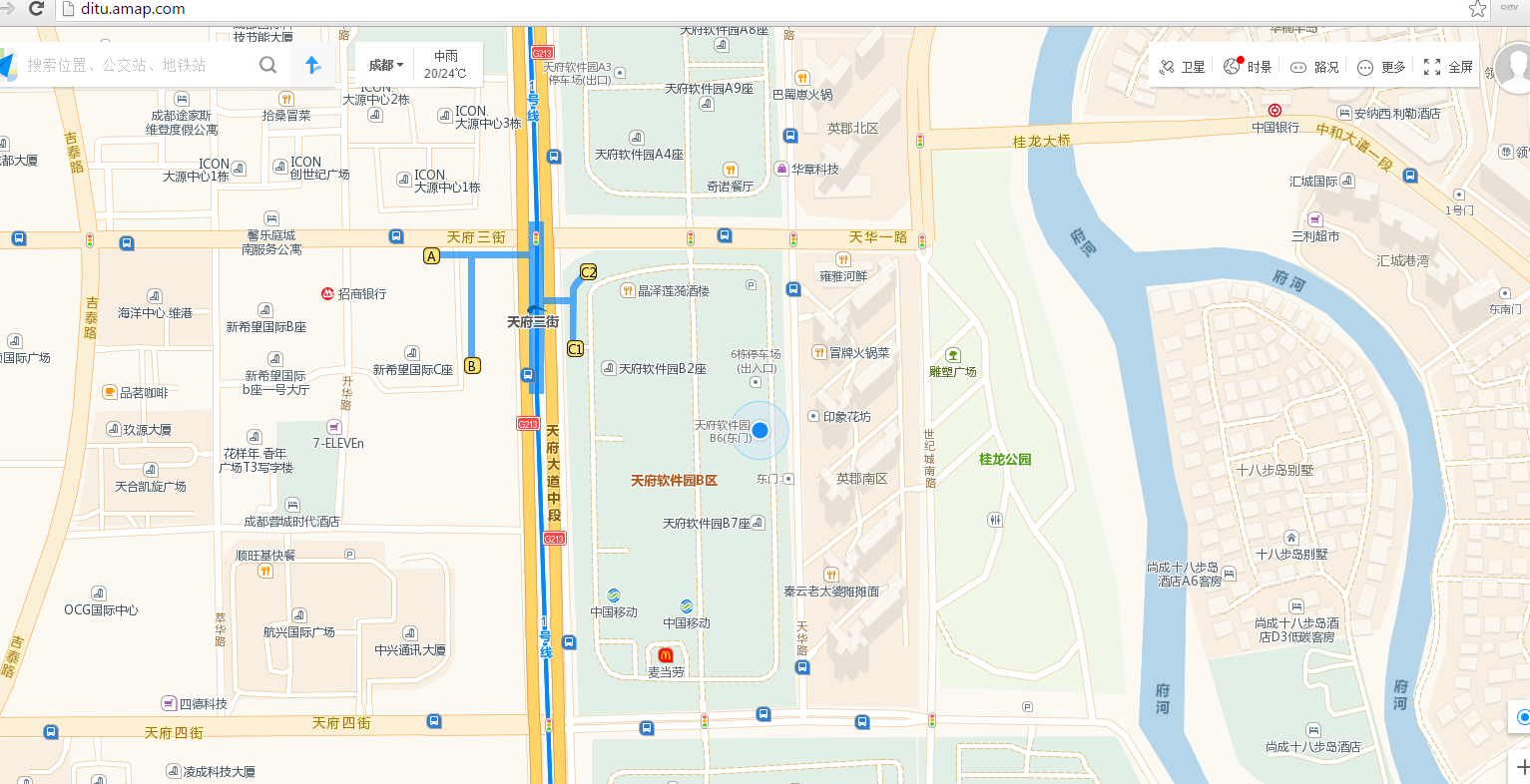
### 基础设备 公交站点



### 新功能 最大发车间隔保障率



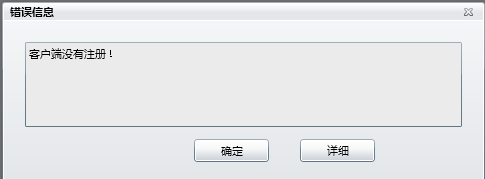
## 车辆GIS监控界面



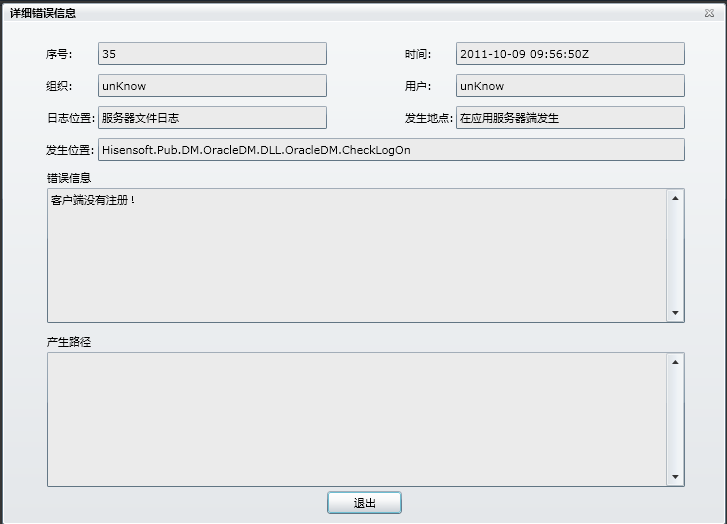
## 线网分析界面

无

## 错误输出界面

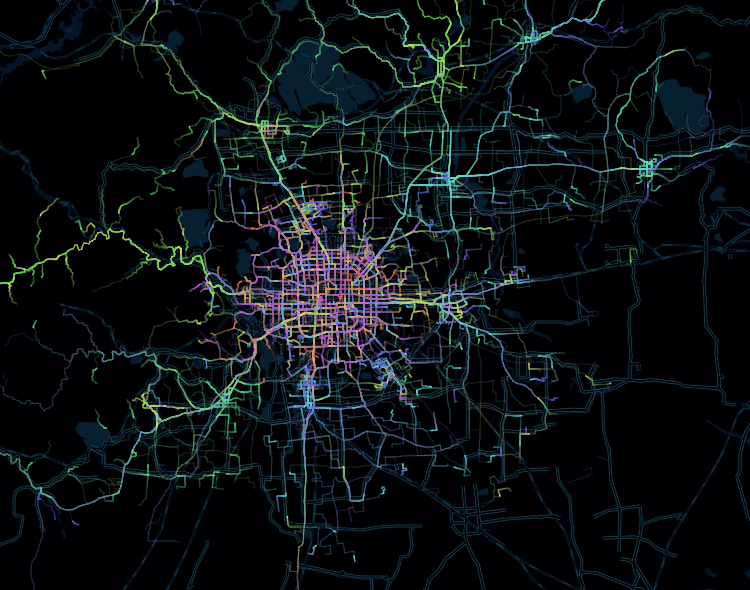


错误提示

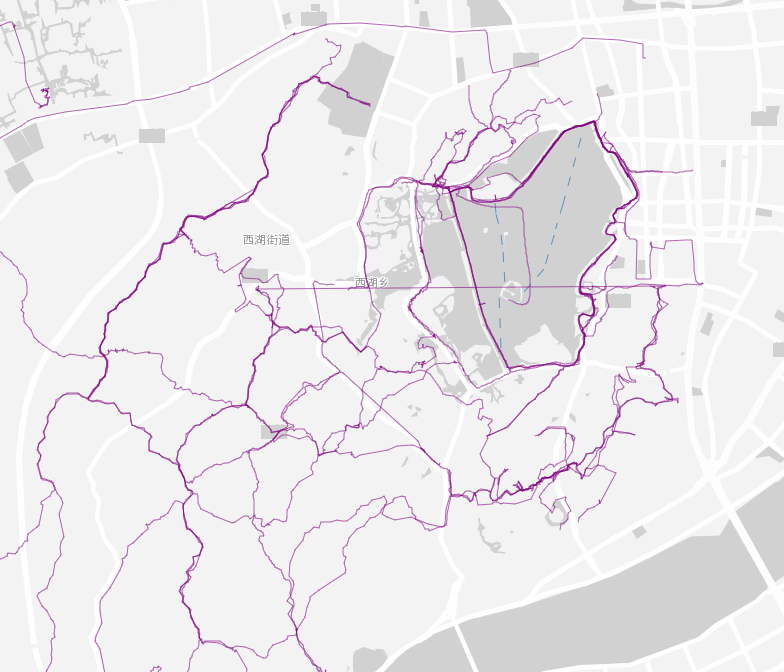


详细错误信息

## 特殊效果



公交实时运行图



热门线路图

## 美学设计

现阶段系统界面框架采用业界流行的web 2.0设计，充分发挥web界面的优势，提高人机交互的体验。

## 界面资源设计

### 图标资源

采用font woe矢量图标标准和高德地图地图资源。

### 图像资源

无

### 界面组件

采用Bootstrap相应是布局框架，满足目前web程序最流行，最先进的布局概念。

# 数据库设计

## 数据库环境说明

数据库环境：Oracle 10g （10.2.0.4）

数据库设计工具：Enterprise Architect 7.0、Word

数据库访问工具使用PLSQL 7.0

## 数据库的命名规则

ORACLE与SQL SERVER的类型对应表及缩写形式（Access数据库参照执行）：

| 缩写 | ORACLE | SQL SERVER |  | 缩写 | ORACLE | SQL SERVER |
| --- | --- | --- | --- | --- | --- | --- |
| NC/C | NC/C | NC/C |  | OID | NVC(36) | UNIQUEIDENTIFIER |
| NVC/VC | NVC2/  VC2 | NVC/  VC |  | ROWVER | NVC(36) | ROWVERSION |
| INT | INT | INT |  | INTID | INT | INT |
| DEC | INT(n,m) | DECIMAL |  | ENUM | INT | INT |
| FLT | FLOAT | FLOAT |  | BOOL | INT | INT |
| DT | DT | DT |  |  |  |  |
| MEMO | NCLOB | NTEXT |  |  |  |  |
| BLOB | BLOB | IMAGE |  |  |  |  |

* + - 在允许的情况下，尽可能不使用FLT类型。
    - 整数使用INT类型，小数使用DEC类型。
    - 备注型字段使用MEMO类型。
    - 长度固定的字符字段和1字符字段采用NC/C类型，否则采用NVC/VC类型。
    - OID：NULLID = {00000000-0000-0000-0000-000000000000}。
    - INTID：NULLID = 0。
    - ENUM与数据字典中的类型定义相对应（见APT2\_DS\_DB\_TABS\_SYS.数据类型），统一使用ENUMx标记，x是分配的编号。BOOL=ENUM1
    - 标记方式：C10，NVC255，DEC8.2。

### 数据库对象命名规范

1. 表命名规则

<英文名称>。

英文名称使用单数形式。可以由多个单词组成，单词之间采用下划线“\_”隔开。关联表以R+关联表1名称+关联表2名称+…的形式组织。

各表添加前缀依据：

 中间表/临时表： tmp\_

1. 视图命名规则

V\_<英文名称>，例如：V\_View。

1. Oracle序列命名规则

SQ\_<英文名称>，例如：SQ\_Squence。

1. 存储过程命名规则

USP\_<英文名称>，例如：USP\_StoreProd。

1. 触发器命名规则

TR\_<表名>\_[i|u|d]<英文名称>，例如：TR\_Table\_iInsert。

1. 主键命名规则

PK\_<表名>，例如：PK\_Table。

1. 外键命名规则

FK\_<表名>\_<主键表>，例如：FK\_Table\_ForeignKey。

1. 索引命名规则

IX\_<英文名称>，例如：IX\_Table\_FieldA。索引英文名称一般使用<表名>\_<索引字段名>的形式。

1. 自定义函数命名规则

FUN\_<英文名称>，例如：FUN\_GetMax。参数一律小写，单词间用“\_”分隔。

1. 包命名规则

PKG\_<英文名称>，例如：PKG\_GetMax。

1. 字段命名规则

<英文名称>，例如：Code。字段名称应尽量简短，长的单词最好采用缩写形式。每个单词首字母大写，其他字母小写。各表ID字段以表名+ID的形式组织，如BusID。

1. 后缀特殊约定：
   * + ID表示数据标识，一般无意义。
     + Name表示名称。
     + Code表示编号、编码、代码、代号等，具有业务意义。

### 其他规范

1. 为保证兼容性，禁止使用类型规范中未列出的数据类型。
2. 表和视图的字段必须在两种数据库中具有兼容类型。

## 逻辑设计

无

## 物理设计

无

## 安全性设计

页面提交时对提交数据进行防止注入的处理。

### 防止用户直接操作数据库的方法

用户只能用帐号登陆到应用软件，通过应用软件访问数据库，而没有其它途径操作数据库。

### 用户帐号密码的加密方法

采用自定义的算法来进行用户账户加密

### 角色与权限

如果当前用户对于请求的页面没有权限，直接跳转到登录页面。

## 优化

为了提高数据库运行效率和指标分析响应时间，系统在原始公交企业业务数据入库后进行预先进行指标分析和计算，并将计算结果保存到数据表中。

公交业务数据按日/月/年分别进行结转，并将中间结果信息保存到结转表中。

为了节约数据库存储空间，针对已经做指标分析、结转的原始定时进行备份后清除。

## 数据库管理与维护说明

数据库维护采用自动维护和管理机制，不需要人工直接干预。数据库系统主要完成以下工作：

1. 自动创建日/月/年分区表
2. 自动导出历史分区数据
3. 自动删除过期的分区数据。
4. 自动进行行业监管指标分析，并将分析结果数据保存到中间表。
5. 自动进行业务数据结转，并将日结、月结数据保存到中间表。