

学号 1362810321

年级 2013 级

本科毕业设计

工业设备监控信息发布系统

专 业 计算机科学与技术

姓 名 张成远

指导教师 牟 艳

评 阅 人 丁海军

2017 年 6 月

中国 南京

BACHELOR'S DEGREE THESIS OF HOHAI UNIVERSITY

Information Release System of Industrial Equipment Monitoring

College : The Internet of Things Engineering

Subject : Computer Science and Technology

Name : Zhang Chengyuan

Directed by : Mu Yan

NANJING CHINA

郑重声明

本人呈交的毕业设计，是在导师的指导下，独立进行研究工作所取得的成果，所有数据、图片资料真实可靠。尽我所知，除文中已经注明引用的内容外，本设计（论文）的研究成果不包含他人享有著作权的内容。对本设计（论文）所涉及的研究工作做出贡献的其他个人和集体，均已在文中以明确的方式标明。本设计（论文）的知识产权归属于培养单位。

本人签名：_____

日期：_____

河海大学

本科毕业设计（论文）任务书

（理工科类）

I、毕业设计（论文）题目：

工业设备监控信息发布系统

II、毕业设计（论文）工作内容（从综合运用知识、研究方案的设计、研究方法和手段的运用、应用文献资料、数据分析处理、图纸质量、技术或观点创新等方面详细说明）：

本课题来源于江苏兴荣高新科技股份有限公司和河海大学合作的项目。随着工业物联网的发展，将信息技术、自动化技术、现代管理技术与制造技术相结合，改善制造企业的经营、管理、产品开发和生产等各个环节，提高生产效率、产品质量和企业的创新能力的需求越来越高，为此开发工业设备监控系统，旨在工业生产现场构建的一套信息处理系统，实时收集现场生产设备和仪表等数据，向具有访问权限的用户提供设备和仪表等实时生产数据访问服务报警推送服务。本设计是工业设备监控系统的一部分，负责实时发布设备数据与展示，信息通过移动设备进行发布，完成如下功能：

- 用户登录 采集用户头像，通过数据服务模块存储于业务数据库中；
- 查看设备列表并提供筛选功能；
- 查看设备监测点的实时数据状态并提供查询筛选功能；
- 查看设备监测点的历史记录并提供筛选功能；
- 报警推送服务与数据实时更新。

本系统要求设计合理、实时性强、使用方便，高效、功能完整，有较强的可复用性、可移植性，系统应用 JavaWeb, Sql server, Html5, JQuery, HTML5+, Java 开发。

要求设计和实现使用面向对象的软件开发技术，并结合相关设计模式，运用UML等相关建模语言建模，完成需求分析，需求建模，软件设计，编码实现，测试等工作，并产生相应文档、数据及程序代码等。通过本设计的实现，将软件开发技术理论的研究与软件开发实践相结合，在实践中培养学习能力、分析和解决问题能力以及独立工作能力。

设计工作完成后，按要求撰写毕业论文，进行毕业答辩。

III、进度安排：

2016.12-2017.1 熟悉项目需求、熟悉开发环境、确定数据的需求

2017.2 对系统进行总体设计、确定模块及模块功能、完成数据库设计、对系统进行整体构架

2017.3-2017.4 进行系统的详细设计、完成编码

2017.5 进行测试与调试并进行修改完善

2017.6 撰写毕业论文、进行毕业设计答辩

IV、主要参考资料：

[1]郑晓锋,蔡瑞英.异构系统信息集成的数据仓库方法 [J].南京工业大学学报.

[2]A. Brandt ,Ethan L. Miller, Darrell D, E. Long, Lan Xue. Efficient metadata management in lager distributed storage systems [C].InPoreeedings of the 20thIEEE/1lth NASA Goddard conference on Mass Storage Systems and Technologies ,April 2003

[3] Catharine M.Wyss,Dirk Van Gucht. A relational algebra for data/metadata integration in a federated database system [C].Proceedings of the tenth international conference on information and knowledge management,CIKM2001.

指导教师： 牟艳，田福生（企业） ， 2016 年 12 月 日

学生姓名： 张成远 ， 专业年级： 计算机科学与技术 13 级

系负责人签字： ， 2016 年 12 月 日

摘要

本课题来源于江苏兴荣高新科技股份有限公司和河海大学合作的项目。为满足课题需求，开发工业设备监控系统，旨在工业生产现场构建的一套信息处理系统，实时收集现场生产设备和仪表等数据。

本设计是该工业设备监控系统中重要的一部分，主要任务为向具有访问权限的用户提供设备和仪表等实时生产数据访问服务报警推送服务，同时对于设备的数据进行多方面的展示与筛选，与此同时完成数据的请求与获取功能保证数据的通信，除此之外，保证工业设备监控信息发布软件的用户体验和响应效率也是设计的关键。论文从以下几个方面展开讨论：

1) 工业设备监控信息发布系统总体设计。

在工业设备监控信息发布系统的需求分析的基础上，制定系统开发的总体设计方案和思路。该部分主要从应用技术、开发环境、系统模块等方面展开。

2) 前端设计与数据交互设计。

重点介绍了本系统采用的前端开发框架和数据结合模板设计以及三层结构中逻辑层在前端页面的具体表现，利用 JSON 数据和 JSONP 技术对用户的请求进行交互，完成异步请求、异步刷新页面等功能。同时满足系统的用户登录和信息完善等移动应用特色功能。

3) 数据推送服务设计。

该部分为工业设备监控信息发布系统的核心功能和特色功能，利用个推推送服务对工业设备监控信息进行实时更新和警报推送服务，满足系统的时效性和同步性，利用透传消息概念保证了系统较好的用户体验。

4) 工业设备监控信息发布系统测试。

对整个系统展开黑盒测试，对用户信息维护功能、设备信息展示、监测点信息推送与警报等功能进行单元测试，验证了系统的可行性和可用性。

经过对系统的总体设计、详细设计和测试，本设计功能完备，满足工业设备监控信息发布系统的基本需求。

关键词：工业设备监控；信息发布；移动应用；JavaWeb；JSONP；个推推送

ABSTRACT

With the development of industrial automation, industrial equipment information's monitoring and display has become popular. The idea of this project come from the project which the Hohai University and Jiangsu Rongxinggaoke company's co-op subject. Therefore, the aim of developing the industrial monitoring system is to build up the information processing of the real circumstance of the industrial producing to collect real data of products or equipment. This design is the essential part of the industrial equipment monitoring system and its demand requires industrial company to build a set of information processing system, real-time collection of field production equipment and instrumentation data to access users with equipment and real-time data such as production data or alarming push service. Besides, the way and display should be comprehensive. The paper discusses from the following aspects:

1) Overall design of industrial equipment monitoring information release system.

Based on the analysis of the demands of industrial equipment monitoring information release system, the overall design plan and the thinking of system development are formulated. This part mainly consists of the application of technology, development environment, system modules and etc.

2) Interface design and data interaction design.

This paper mainly introduces the interface development framework and data combination template design and the concrete performance of the logical layer in the front-end page of the three-tier structure, interacts with the user's request using JSON and JSONP technology completing the asynchronous request. At the same time meet with the user login and information such as mobile applications and other features.

3) Data push service design.

This part is the core function and characteristic function of the monitoring equipment of the industrial equipment. The real-time update and alarm push service of the industrial equipment monitoring information is realized by using the getui push service, which satisfies the timeliness and synchronization of the system and ensures the System better user experience.

4) The tests of Industrial equipment monitoring information release system.

The black box test is carried out on the whole system, and the unit test is carried out to test the function of user information, the display of equipment information, the push and information of monitoring point information, and verify the feasibility and usability of the system.

After the overall design of the system, detailed design and testing, the design features complete, the whole design meets the basic needs of the industrial equipment monitoring information release system.

Keyword: Industrial equipment monitoring; Information release; Mobile application; JavaWeb; JSONP; Getui Push

目 录

摘要.....	I
ABSTRACT.....	II
第 1 章 绪论.....	1
1.1 课题背景及其意义	1
1.2 课题研究内容	1
1.2.1 移动应用前端设计	3
1.2.2 数据请求与获取	3
1.2.3 数据推送服务	3
1.3 主要技术简介	3
1.3.1 Hybrid App 开发模式	4
1.3.2 面向对象程序设计（OOP）	4
1.3.3 三层架构	5
1.3.4 B/S 软件系统体系结构	5
1.3.5 HTML5plus Runtime 引擎.....	5
1.3.6 Java Servlet 开发技术.....	6
1.4 开发工具简述	6
1.4.1 Visual Studio 2017	6
1.4.2 MyEclipse 10	7
1.4.3 HBuilder	7
1.4.4 Android2.3	7
1.5 本章小结	8
第 2 章 工业设备监控信息发布系统需求分析.....	9
2.1 业务需求	9
2.2 功能需求	10
2.2.1 移动应用前端	10
2.2.2 数据请求与获取	11
2.2.4 监控信息推送与分析	11

2.3 数据需求	12
2.3.1 本地数据	12
2.3.2 服务器数据库	12
2.4 性能需求.....	12
2.4.1 数据精准度	12
2.4.2 时间特性	13
2.4.3 适应性	13
2.5 运行环境需求	13
2.6 其它需求	13
2.6.1 可使用性	13
2.6.2 保密性	13
2.6.3 可维护性	13
2.7 本章小结	14
第 3 章 工业设备监控信息发布系统概要设计.....	15
3.1 设计原则	15
3.2 设计思想	15
3.2.1 多平台	16
3.2.2 跨域通信	16
3.2.3 设备监控信息发布	16
3.3 系统总体结构设计	17
3.4 软件结构设计	18
3.4.1 移动应用前端设计	18
3.4.2 数据请求与获取	19
3.4.3 监控信息推送与数据分析	20
3.5 数据库设计	20
3.5.1 数据库系统结构设计	20
3.5.2 数据库逻辑结构设计	21
3.6 本章小结	26
第 4 章 工业设备监控信息发布系统详细设计.....	27

4.1 移动应用前端设计	27
4.2 数据请求与获取	38
4.4.1 服务端	38
4.4.2 用户端	41
4.3 监控信息推送与分析	41
4.4 本章小结	44
第 5 章 系统调试与实现.....	45
5.1 调试过程中的主要问题和解决方法	45
5.1.1 HTML5+框架引用失败	45
5.1.2 页面锚点位置	45
5.1.3 注销功能失效	46
5.1.4 页面不适应移动设备	47
5.2 系统运行成果图	48
5.3 本章小结	50
第 6 章 总结与展望	51
6.1 总结	51
6.2 展望	52
致谢.....	54
参考文献	55
外文文献翻译.....	57

第 1 章 绪论

1.1 课题背景及其意义

目前工业生产中应用到的设备数量庞大，类型多样，工业设备的状态和数据的监控随着工业的发展逐渐变成一个难题，这对于工业的效率和 development 有阻碍，因此，亟需一种对工业设备的监控系统来辅助管理者对工业设备进行监督。

工业设备监控信息采集系统，为监控设备数据提供了可能性，移动应用的发展与逐渐成熟为工业设备信息发布系统提供了契机，两者相结合，可以实现工业设备信息的监督、采集和分析。目前一些工业信息系统的发布系统主要是以工厂中特定机器来显示并查看的形式进行，这样的话，整个工业设备的检测拥有一定的空间和时间局限性，需要特定的人员拥有特定的权限才可以监控设备，大大降低了整个工作的效率，而将整个信息发布系统转移到移动设备上，首先可以降低操作的复杂程度，其次提高监控的效率，使得监督更有条理，更加快速。

另外，工厂内现有的设备监测系统成本较高，使得一些小型企业和工厂没有能力引进购买此类设备，而本课题的工业设备监控信息发布系统则只需安装移动端 APP 即可完成相同的功能，其次还将以更清晰更直观的方式将数据和各种信息展示出来，使用户拥有更好的体验，侧面提高整个工厂监测系统的执行效率。

除此以外，目前国内对于工业设备信息的监控系统有着一定的进展，如上海能麦电子科技有限公司针对设备的状态开发出了设备状态集中在线监测系统，其中只是采用了数据采集终端对站内的工作人员发布设备信息，因而拥有一定的空间局限性，而工业设备监控信息发布系统采用移动应用端的发布，克服了这一局限，以创新的形式完成工业设备信息的发布。

因此，工业设备监控信息发布系统应运而生，通过手机 APP 的形式对工业设备的信息监控进行发布，方便工业管理者和设备操作员的工作，拥有着比较光明的前景。

1.2 课题研究内容

工业设备信息采集工具的出现和移动应用技术的发展与成熟使得工业设备管理者在移动端查看并监督设备的运行情况成为了可能,根据设备从属的机构和管理者(工业设备监控业务管理系统负责提供),根据设备信息采集工具提供的数据(工业设备监控信息采集系统负责提供),管理者能够快速并准确的确认设备的运行状况,并对设备的数据进行筛选和分析。整个工业设备监控系统的功能模块图如图 1.1 所示。

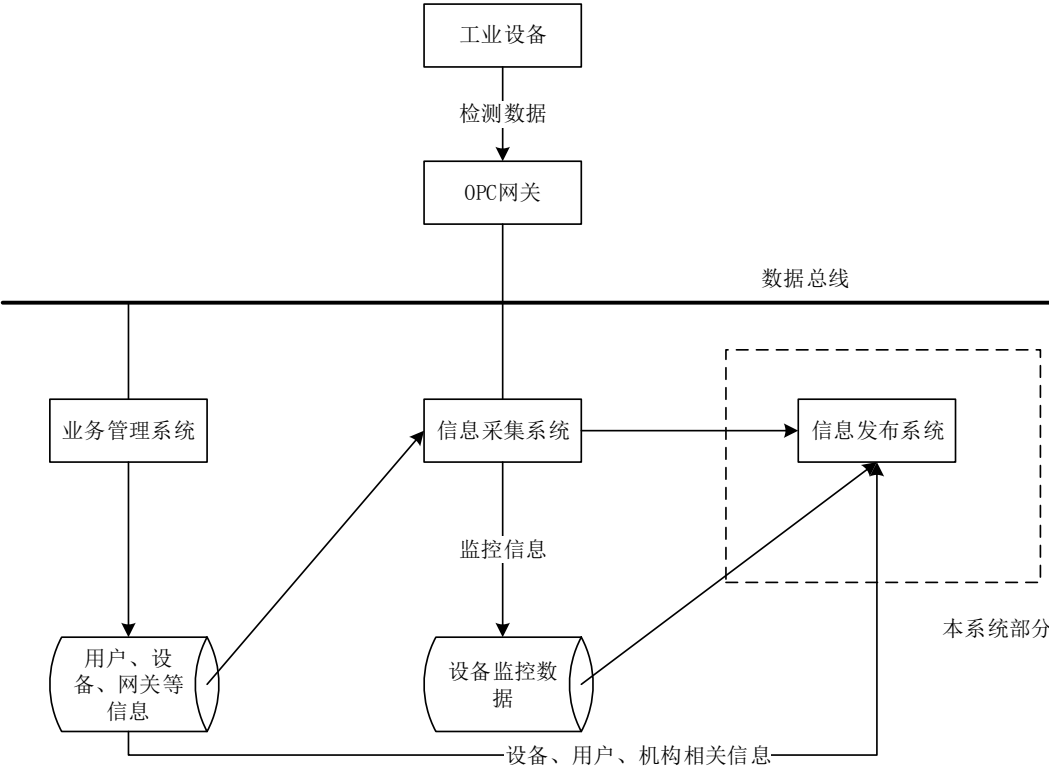


图 1.1 工业设备监控系统总图

本课题主要专注于开发工业设备监控信息发布系统,其主要目标为根据工业设备监控业务管理系统提供的用户、机构、设备、设备监测点之间的关系和工业设备监控信息采集系统提供的设备监测点实时数据和历史数据表实现移动应用端的设计,主要包括:用户登录,设备列表查询,设备监测点实时数据与历史数据显示。另外,数据来源于三个工业设备监测系统的公用数据库,其中所有的数据获取均来源于数据库,通过移动应用端发出参数请求,服务器完成数据请求获取数据并处理返还给移动应用端,数据请求与获取为工业设备监控信息发布系统的后台核心。除此之外,后台服务器要对移动应用端进行数据推送服务,实现设备监测点信息与工业设备监控信息发布系统的同步。

工业设备监控信息发布系统由前期调研和实际需求综合后，主要包含以下几个模块：

1.2.1 移动应用前端设计

移动应用前端设计模块主要部署在移动端的应用程序，其功能的完成需要完成前端模板页面设计和功能业务类设计，在前端模板页面设计中，主要完成网页的跨平台设计和响应式设计，让整个前端程序美观、实用、用户体验良好。在功能业务类设计中，主要完成整个应用程序的动态响应和用户操作反馈，使得应用程序快速反应、流畅并能有效地与服务器交互。

1.2.2 数据请求与获取

数据请求与获取模块为工业设备监控信息发布系统的关键功能，该部分主要实现移动应用端的数据请求和服务器的数据获取，该模块与应用数据库进行交互，并为数据推送服务提供数据。该模块主要使用 AJAX 异步跨域请求和 JDBC 技术完成，实现请求高效，快速响应的数据交互功能。

1.2.3 数据推送服务

数据推送服务为工业监控信息发布系统的重要组成部分，完成数据的实时显示和同步，利用数据请求与获取模块的应用程序对工业设备监控信息采集系统的实时数据对用户进行定点推送，主要利用个推送框架和工业设备监控信息采集系统的数据上传模块完成数据推送服务。

1.3 主要技术简介

本系统开发时所采用的主要技术包括：Hybrid App 开发模式，面向对象程序设计（OOP），B/S 软件系统体系结构，HTML5plus Runtime 引擎，Java Servlet 开发技术等。在整个系统的开发过程当中，严格遵循了软件工程的思想，使得整个项目有较强的稳定性和较高的效率。

1.3.1 Hybrid App 开发模式

Hybrid App（混合模式移动应用）是一种介于 Web App 和 Native App 之间的移动应用模式，即拥有 Native App 的外观和模式，实际上只有一个 WebView 镶嵌于应用中。其兴起的原因是移动互联网的热潮来袭，使得众多公司前赴后继投入到移动应用的开发当中，但原生应用的开发人员较为稀缺，熟知网页开发的工作人员就借助 HTML5 的新兴，完成了一套利用网页来开发移动应用的理念和框架。总的来说，Hybrid App 同时使用网页语言和程序语言开发，通过应用商店让用户安装，其明显的优点为开发成本和难度相对较低，平台的可移植性较高。

表 1.1 原生 APP 与混合式 APP 对比表

	Hybrid APP	Native APP
开发成本	中	高
维护更新	简单	复杂
体验	中	优
Store 认可	认可	认可
安装	需要	需要
跨平台	优	差

如表 1.1 所示，Hybrid App 开发模式更符合此项目的开发背景和环境。

1.3.2 面向对象程序设计（OOP）

面向对象程序设计是一种以对象为核心的程序编程范型，同时也是一种 程序开发的抽象方针。在这种设计思想中，程序和数据都被封装在对象中，因此，灵活，可扩展，可重用即为其优点和特点。在面向对象程序设计中，彼此相关的对象组成整个计算机程序。

面向对象程序设计主要包含五大要素：类，对象，继承，多态，封装。类为对象的抽象体现，包括对象中的状态信息和行为信息。对象为面向对象设计的基本组成单元，对象将类特殊化，实例化，因此，从类到对象是将现实事物进行计算机描述的过程。继承，多态，封装这三个特性，使得面向对象程序设计更加完备和稳定，也使得其特点更为鲜明。在工业设备监控信息发布系统设计过程当中，

采用的编程思想皆为面向对象，使得整个系统更为灵活，也提高了整个项目的执行效率。

1.3.3 三层架构

三层架构为当前软件开发较为流行的一种开发架构，其具体内容主要为在客户端和数据库中间加入一个用来处理业务逻辑和合法性校验等功能的中间层（业务逻辑层）。在三层结构中，表示层用来完成软件具体的展示工作，具体包括对用户请求的处理，接收数据的返回，为用户提供访问接口等；业务逻辑层负责对数据层的操作；数据访问层则主要负责与数据库的交互，在本系统中主要表现为 JDBC 驱动对 SQLServer 的访问接口。三层结构拥有一大特点即为部分独立性：对表现层的修改不会影响逻辑层的功能和结构，对逻辑层的修改不影响数据层的功能和结构。这样的特点使得软件系统易于维护，结构清晰，使得开发更加便利。

1.3.4 B/S 软件系统体系结构

B/S 软件系统体系结构是 C/S 结构的一种改进的结构。客户机只需要安装浏览器同数据库进行数据交互。用户界面完全通过浏览器实现，事务逻辑在前端实现，但是主要的事务逻辑在服务器端实现，形成所谓三层结构。B/S 结构利用浏览器取代过去软件系统实现的特定的功能，节约了开发成本，是一种全新的软件系统构造技术。B/S 结构的最大特点就是其兼容性，不需要安装特定的软件，只需要访问网址即可完成一系列的操作和功能。

B/S 结构主要有以下几点特征：可用性、稳定性、易升级、低维护、兼容性、低带宽、丰富展现等。B/S 结构型的软件在安全性上也是较高的，B/S 结构只需注重服务端。该系统中所有的用户都是通过 JDBC 驱动连接到数据库的，用户无需保持对数据库的连接。

1.3.5 HTML5plus Runtime 引擎

自从 HTML5 出现以来，移动应用开发和 Web 开发者就迎来了新的曙光，但其功能一直被开发者所诟病。HTML5plus Runtime 是运行于移动端的强化移动应

用的引擎，在支持标准 HTML5 的基础上，扩展更多的 js api，使得 js 的能力不输于原生。相比于混合移动应用开发模式起步时期就出现的 phonegap 和 cordova 等方案，5+Runtime 分为三个层次解决了 HTML5 与原生的能力差距：

- 1) 常用的 API——HTML5plus
- 2) 其他原生 API——Native.js
- 3) 原生 SDK——5+Runtime SDK

1.3.6 Java Servlet 开发技术

Servlet 是服务端小程序的意思。是由 SUN 公司提供的用于开发动态 Web 服务端的技术，该小程序是整个 JavaWeb 技术的基础。Servlet 本质上也是类，但要按照 Servlet 规范进行设计，不拥有 main 方法，它的运行周期由 Servlet 容器进行管理(如 Tomcat)。从具体应用上来说，Servlet 可以响应任何类型的请求，但绝大多数情况下 Servlet 只用来扩展基于 HTTP 协议的 Web 服务器。而工业设备监控信息发布系统的主要数据请求协议即为 HTTP 协议，因此能满足该系统的功能。

Servlet 执行以下主要任务：

- 1) 读取移动应用发送 Get 请求的数据。
- 2) 处理数据并生成结果。
- 3) 发送处理的数据到移动应用。

1.4 开发工具简述

本发布系统的开发过程当中使用的开发工具主要有：Visual Studio 2017，MyEclipse 10，HBuilder，系统运行环境为 Android2.3 以上版本。

1.4.1 Visual Studio 2017

Visual Studio 2017 是微软于 2017 年推出的 VS 新版本，是目前最具生产力的 VS 版本。其内部构建工具整合了 .NET Core、微服务、Docker 容器等内容。

Visual Studio IDE 大幅改进了 Visual Studio 2017，改进登录和标识、改进代码导航和引导，另外还增添打开文件视图和链接的服务，实现了应用和任意资源之间的连接。对异常处理进行了改进，可方便并快捷地定位代码。此外，诊断工具中增添了应用程序的事件摘要，并对 CPU 监控工具进行改进。

1.4.2 MyEclipse 10

MyEclipse10，是在 eclipse 这一强大 IDE 基础上加上本身的插件开发而成的功能强大的企业级集成开发环境，主要用于 Java、Java EE 的开发。该软件极大方便了对 JavaEE 的开发、发布以及应用程序服务器的整合，提高编码工作效率。该软件是功能强大的 JavaEE 集成开发环境，方便开发者编码、调试、测试和发布，完整支持 Struts, JSP, Spring, SQL, Hibernate, servlet 等技术。

1.4.3 HBuilder

HBuilder 是 DCloud 推出的一款支持 HTML5 的 Web 开发 IDE。HBuilder 是由 Java 开发的，是一款基于 Eclipse 的网页开发 IDE，因此兼容了 Eclipse 的插件。该开发工具通过语法提示和代码输入法、代码块等解决了 eclipse 的一些短板，大幅提升 HTML、js、css 的开发效率。

1.4.4 Android2.3

Android 是一种基于 Linux 的开源操作系统，主要使用于移动设备，该平台由操作系统、中间程序、用户界面和应用软件组成。Google 在 2010 年发布了智能手机操作系统 Android 2.3，俗称姜饼系统。在版本方面 Android2.3 相对于之前的移动应用操作系统而言改进并不算多，但功能和界面的改进相对明显。由于 Android 的开放性，开发者会开发出各式各样的移动应用和程序。功能上的差异和特色，数据同步、软件的兼容都能够平稳的运行在此系统上。Android 系统提供给开发者十分广阔、自由的环境。

1.5 本章小结

本章主要就工业设备监控信息发布系统的研究背景及意义，研究内容，技术要点和搭建环境进行简要概述。

第 2 章 工业设备监控信息发布系统需求分析

工业设备监控信息发布系统需求分析主要从：业务需求、功能需求、数据需求、性能需求和运行环境需求进行具体分析，最终为该系统的概要设计奠定基础。

2.1 业务需求

随着互联网技术的发展和物联网技术的兴起和成熟，工业和互联网行业的结合是当下的趋势，工业车间内的设备监控技术的发展使得工作人员的工作更加高效，然而，受到设备场地和工作场地的限制，在一些特殊设备和特殊场地的条件下，对设备的监控显得至关重要。借用工业设备监控信息采集系统，远程获取设备信息参数成为了可能，因此，当前主要的棘手的问题为如何快速且有效地将数据和分析情况展现给工作人员，工业设备监控信息发布系统的想法因此被提出和关注。

本系统的数据来源为工业设备监控信息采集系统对设备信息和数据的采集，这些数据被存放在服务器数据库中，利用后台技术将获取到的数据提供给用户。另外，用户与其机构、设备、设备监测点的关系由工业设备监控业务管理系统对一系列信息进行分配和归纳。用户通过客户端进行登录和对个人信息的修改，后台按照美观且高效的方式展现给用户，同时还要注重整个客户端的可移植性，使得用户高效、直观地获取到与之权限相对应的设备的参数，并以不同的形式展现，包括筛选和查询等。

数据的请求与获取功能应满足：用户提供相应的查询操作，利用跨域访问技术对服务器提出请求，服务器通过数据库驱动获取相应的数据并按照请求时的标识返还给用户，整个操作应该连贯且高效，并还需满足在获取数据失败的情况下及时对用户有所反馈。

数据推送服务的功能应满足：当工业设备监控信息采集系统监测到设备采集点信息变化时，即向在线用户推送设备监测点数据，若达到该监测点的报警值，

统等功能。用户需要提供自己的信息包括帐号、密码和所对应机构登入系统，随后可以通过操作改变或者初始化自己的信息，包括：

- 1) 修改并初始化头像。
- 2) 修改密码。
- 3) 注销账户。

该部分的用户及机构的分配由权限分配系统确认。

(2) 设备信息显示

用户登入系统后，显示与自己权限相符合的所有设备的信息和数据，包括设备名称，延迟类型和设备编号，用户通过选择相应类别的设备对设备列表进行筛选。另外，选择某一设备后显示该设备的所有检测点，其中包括最近一次检测的数据和检测时间，同时显示该监测点是否启动，通过模糊查询相应的设备监测点。选择相应检测点后，通过图表显示所有数据，并显示报警类型和其相应的阈值。其次，用户可以根据时间来筛选数据并显示数据。

2.2.2 数据请求与获取

数据请求与获取模块即连接 APP 和服务器程序的桥梁，主要需要实现数据的跨域请求与交换，由于 APP 与服务器属于两个不同的地址，因此需要将 APP 端的数据按照一定的规范传递给后台服务器，由服务器程序负责与数据库建立连接，实现数据的单方面请求。随后，服务器程序将获取到的数据库数据按照一定的规则进行数据整合、转化，将数据变成用户要求的且可以直接被利用在前端页面的格式。

2.2.4 监控信息推送与分析

监控信息推送与分析为工业设备监控信息发布系统的重要组成部分，其主要的功能为接收工业设备监控信息采集系统的数据更改发送请求，在服务器端进行数据的比对将信息以一定的格式发到移动应用端，在移动应用端需要判定该回送信息是否需要警报以决定实现推送，同时，将每条的数据进行更新显示给用户。

2.3 数据需求

工业设备监控信息发布系统的数据来源主要由两部分组成：本地数据和服务器数据库。以下是具体的说明：

2.3.1 本地数据

本地数据的主要用途是为了满足移动端的数据请求的便捷性和移动端运行的流畅度和稳定性，如：服务器的地址，登录用户的基本信息（不会泄露用户密码）等，直接提高了整个发布系统的运行效率。

2.3.2 服务器数据库

本信息发布系统的设备采集数据和用户的相关信息均来自于服务器端的数据库中，为的是保证数据的时效性和准确性，主要包括以下几个方面：

- 1) 用户基本信息，如：用户帐号，用户名，密码，所属机构等。
- 2) 设备基本信息，如：设备编号，设备名称，设备类型等。
- 3) 监测点信息，如：检测点名称，参数，单位，报警类型等。
- 4) 实时数据，如：获取时间，值。
- 5) 历史数据，如：获取时间，值。

2.4 性能需求

2.4.1 数据精准度

工业设备监控信息发布系统数据精准度如表 2.1 所示。

表 2.1 工业设备监控信息发布系统数据精准度表

用户帐号	Varchar	3~10	英文或数字
设备编号	Varchar	8	英文和数字
监测点编号	Varchar	13	英文数字和符号
单位	Varchar	0-10	英文
监测点名称	Varchar	0-10	中文

用户密码	Varchar	6-10	英文或数字
------	---------	------	-------

2.4.2 时间特性

为了使用户有更好的用户体验，系统登录时间控制在 5 秒钟以内；

设备列表显示时间不超过 10 秒钟；

历史数据和表格生成时间不超过 10 秒钟。

2.4.3 适应性

本系统适应在任何移动端的设备以及微信公众平台。

2.5 运行环境需求

- 1) 应用服务器：Tomcat7.0x
- 2) 操作系统：Android、IOS、Windows Phone、Wechat 等
- 3) 浏览器：Firefox、Chrome、百度浏览器、360 浏览器等
- 4) 硬件设备：移动手机、平板电脑等

2.6 其它需求

2.6.1 可使用性

在运行程序时，若遇到异常则停止运行，若正常则继续运行。若程序内部遇到异常，则报告，若正常则正常显示。移动应用端需要保持流畅度和响应速度。

2.6.2 保密性

工业设备监控信息发布系统规模虽然不大，但需要保密设备及机构的信息，因此需要采用用户验证，系统保密性良好。另外，限定一个程序中某些区域的规约，给不同的模块分配不同的功能。

2.6.3 可维护性

本系统的主要组成程序分为前端和后端，前端的脚本语言采用 JavaScript，服务器程序主要运用 Java，组构比较简单，各模块较为独立，决定了该软件的简单可维护性。

2.7 本章小结

本章主要就工业设备监控信息发布系统的业务需求、功能需求、数据需求等进行简要概述。

第 3 章 工业设备监控信息发布系统概要设计

工业设备监控信息发布系统的设计较为复杂，OPC 系统获取的数据量较大、数据复杂度较高，因此以下将对本系统的设计原则、设计思想及软件的模块划分进行详述。

3.1 设计原则

1) 平台无关性

该信息发布系统主要采用 Hybrid 移动应用的开发模式，利用 HTML5+ 的开发框架保证了整个系统可以在任何移动平台上运行。同时，该系统还可以在 PC 端的浏览器上运行，保证了整个系统与平台无关。

2) 实时性

工业设备监控信息发布系统的设计，需要考虑到用户需要掌握指定监测点的报警情况和设备参数，整个系统需要与 OPC 系统保持高度一致性，因此，需要保证其实时性。

3) 可维护性

工业设备监控信息发布系统随着时间的推进和技术的发展，以及 OPC 系统的改进，可能需要对系统的界面和数据获取模块进行改进，另外，对于该系统需求的改进与改动，也是对本系统可维护性的要求，本系统需要提供各个模块的模板类及后台数据接口。

4) 高效性

工业设备监控信息发布系统要求程序的执行时间和占用内存皆满足优质且高效的软件的要求，在完成任务需求和模块设计后，程序运行时间与系统的高效性成反比。

3.2 设计思想

在设计工业设备监控信息发布系统的过程中，主要基于以下方面进行考虑：
多平台、跨域通信、信息发布。以下分别进行介绍：

3.2.1 多平台

随着移动应用的不断发展，Hybrid 移动应用开发模式逐渐成为热门，随之而来的是各式各样的移动 Web 开发框架，经过查询和对比，抛弃最初的 Phonegap 和 Cordova 框架是因为其 API 设计的复杂性和效率低下，最终选用的 HTML5+ 框架可以大大提高整个 Hybrid 应用的运行稳定性和高效性，该框架支持所有市面上的前端框架，包括 FronszenUI、Bootstrap 等响应式框架，同时支持各种移动终端，主要包括：

- 1) IOS: iPhone、iPad、Itouch 等。
- 2) Android: 所有设备。
- 3) Windows Phone: 所有设备。
- 4) BlackBerry: 所有设备。

因此，选用 HTML5+ 对整个系统的多平台性有着至关重要的保证。

3.2.2 跨域通信

Web 页面的跨域问题产生原因是企图使用 JS 脚本读写不同域的 JS 作用域。问题根源来自 JavaScript 的同源策略：出于安全考虑，Javascript 限制来自不同源的 web 页面 JS 脚本之间进行交互。否则就会出现各种获取用户私密数据的问题。

经过对该问题的研究，主要抛弃了 HTML 中利用 iframe 标签的跨域访问，原因是因为其页面的掌握和页面的构成无法满足整个系统的运行。因此，本系统决定采用单项跨域技术——JSONP 技术。web 客户端通过与调用脚本的方式，来调用跨域服务器上动态生成的 JSON 格式文件，服务器之所以要动态生成 JSON 文件，目的就在于把客户端需要的数据装入。该技术的主要原理利用了 script 标签的特性，其具有 src 属性且可以跨域。

3.2.3 设备监控信息发布

工业设备监控的信息发布是本系统的主要工作。开发本系统主要选用 VS2017 作为主要页面设计的开发工具，该 IDE 是目前较为流行和使用方便的 Web 开发工具，提供了比较方便的代码提示和纠错功能及强大的调试技术，同时支持各种版本的前端框架，能够帮助开发者方便快捷且有效地开发出 Hybrid 移动应用。其次，选用 MyEclipse 10 为服务器开发工具，MyEclipse 10 较好地集成了 Java 的开发环境，同时和 VS2017 相同拥有着强大的引擎和代码调试功能，非常完美地提供了 J2EE 的开发环境，为整个服务器程序提供了有效的保证。除此之外，移动应用的界面设计采用 HTML5+CSS3 这一热门且逐渐成为主流的语言，利用 JS 的动画效果，提升了用户的视觉体验，且符合移动应用的基本要求，满足整个系统的基本功能。

整个系统采用 B/S 的软件设计结构，为的是保证服务器的运行状态良好，把一些简单的逻辑处理和数据处理放置在前端，间接得保证了整个系统的安全性，满足整个系统的设计原则。在编码实现功能的时候，包括前端页面和服务器程序均采用面向对象的程序设计模式，将整个系统的模块对象化，方便整个系统进行维护，提高了系统的可理解性。

3.3 系统总体结构设计

工业设备监控信息发布系统在设计开发时主要拥有以下三大模块：移动应用前端设计、数据请求与获取和监控信息推送与分析。用户登录与信息采集是整个系统的程序入口，保证了整个系统的基本数据（用户信息）。设备信息显示是整个系统的主要模块，主要负责显示工业设备监控采集系统的数据并将数据进行处理和筛选。监控信息推送与分析是本系统的特色系统，主要负责检测后台数据的变化，通过对该监测点信息的分析，对相应的用户进行推送并警报。通信数据处理主要负责从前端页面接收用户的请求参数，根据相应的操作向服务器数据库请求数据，同时转化为相应的数据格式返还给前端页面。系统结构如图 3.1 所示。

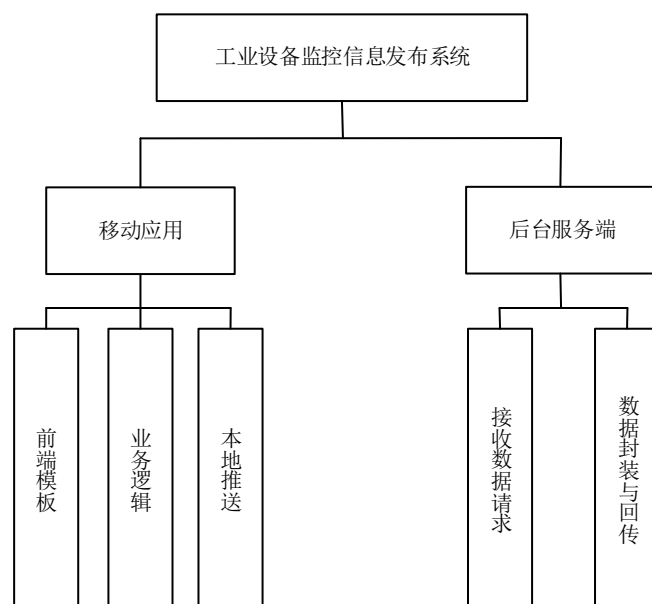


图 3.1 工业设备监控信息发布系统的系统结构图

3.4 软件结构设计

工业设备监控信息发布系统功能主要由三方面组成：信息发布、数据请求与获取、数据推送服务。以下对各个模块分别进行说明。

3.4.1 移动应用前端设计

本功能为工业设备监控信息发布系统的主要展示手段，在界面设计部分，采用的是 frozenui 前端框架，该框架符合响应式设计和简约风格，利用百度模板引擎，将数据和模板页面合成，该过程大致为：本地获取前端页面，通过数据请求与获取模块获取相应数据，利用百度引擎合成。另外，业务逻辑设计主要包含两个方面：用户登录与信息采集和设备信息显示。如下：

（1）用户登录与信息采集

本模块的主要作用是为整个系统的入口，同时保证信息的安全，多样化应用的功能。用户若已经登录，则从本地获取用户基本信息，否则用户根据自己的标识信息登入系统，可以选择修改或者上传自己的头像，同时可以修改账户的密码。用户可以选择注销系统，清除本地缓存的数据。整个模块的功能结构图如图 3.2 所示。

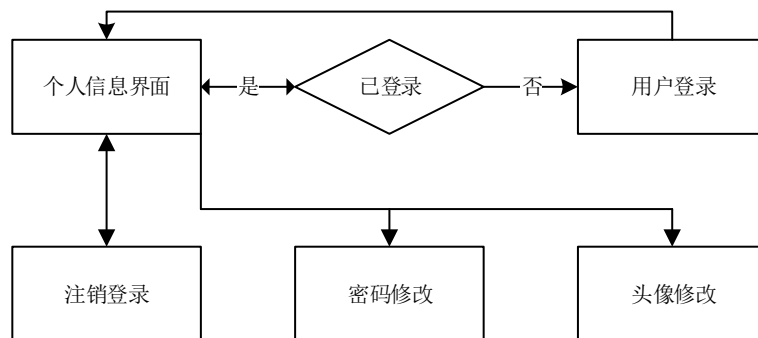


图 3.2 用户登录与信息采集功能结构图

(2) 设备信息显示

用户登录系统后，根据用户权限访问用户列表，该功能主要是为了向用户展示其所拥有权限的设备，并显示其名称、编号、延迟类型等信息。如果用户选择查看某一设备，则具体显示此设备各个监测点的信息，包括监测点名称，实时数据和监测开关情况；另外还可以查看某一监测点的历史数据并对整个历史数据进行分析 and 查看。该模块的结构图如图 3.3 所示。

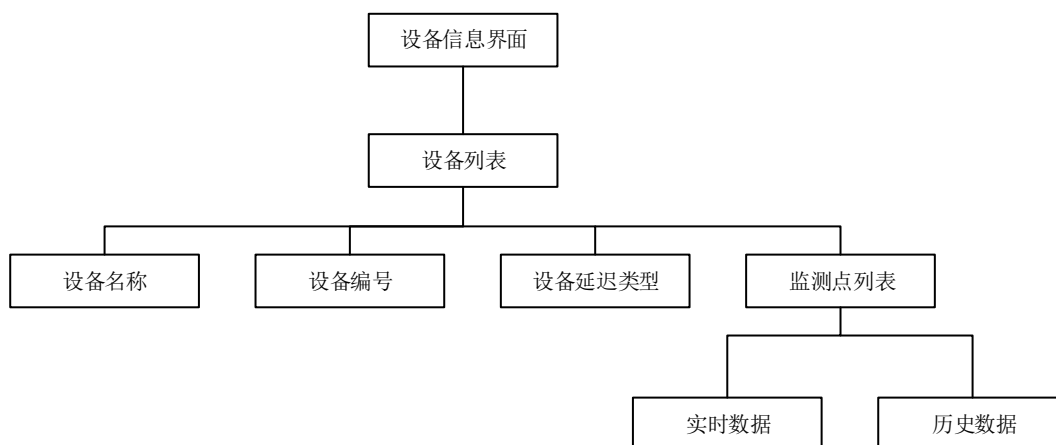


图 3.3 设备信息显示结构图

3.4.2 数据请求与获取

数据通信处理模块为整个工业设备监控信息发布系统的主要数据功能的支撑，本系统在移动端采用跨域异步数据获取的基本方法，满足了 Hybrid 应用的基本要求，同时，异步数据的处理使得整个移动端的操作变得流畅，提高了用户体验感。主要的功能为：在移动端根据用户的操作即业务逻辑设计中移动应用的各种响应函数触发机制提供数据请求参数，利用 JSONP 技术将参数传入服务端程序，服务端记录回调参数，将传入的其他参数进行分析，合成 sql 语句，利用 JDBC

驱动从服务器数据库中获取数据，将数据以 JSON 格式连同回调参数一同返回移动端以备显示数据。通信数据处理的功能结构图如图 3.4 所示。

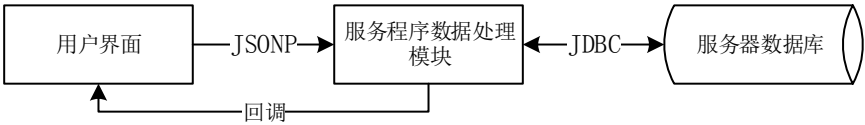


图 3.4 通信数据处理模块功能结构图

3.4.3 监控信息推送与数据分析

监控信息推送与分析模块主要采用个推推送服务和 HTML5+ 中的 push 插件，根据工业设备监控信息采集系统的数据变化情况，通过已定协议，服务器接收设备监测点的信息，在服务器端对实时数据进行分析随后将格式化信息通过透传消息传到移动应用端。该协议内容主要是基于 http 协议，工业设备监控信息采集系统在设备监测点拥有数据变化时，将设备监测点的标识通过 GET 方式传送到工业设备监控信息发布系统的服务器中的 servlet 中，通过对数据库中 T_Real 表进行查询向移动应用端发送透传消息。移动应用端需要监听服务器发送的透传消息以决定是否进行本地推送并主动启动应用，同时进行数据的更新。这个模块的功能为系统的同步性和时效性做了保证。

3.5 数据库设计

本系统的数据来源大部分为服务器数据库，因此，需要根据数据需求设计符合本系统的数据库，满足整个系统的数据要求。

3.5.1 数据库系统结构设计

根据数据需求，可以确定该系统的数据库 E-R 图如图 3.5 所示。

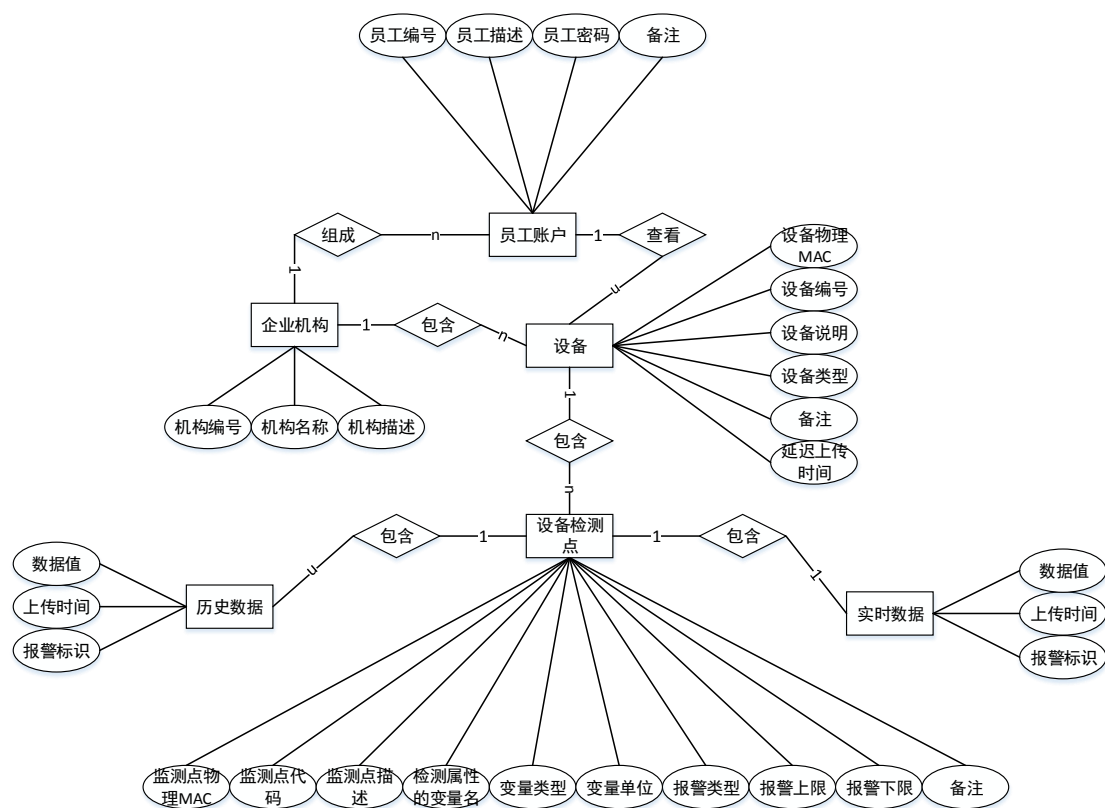


图 3.5 工业设备监控信息发布系统 E-R 图

3.5.2 数据库逻辑结构设计

根据工业设备监控信息发布系统 E-R 图所示，该系统数据库设计中的主要实体有：员工账户，企业机构，设备，设备监测点，实时数据，历史数据，现对各个实体进行分析还原其数据组织结构信息：

1) 企业机构

根据需求分析和 E-R 图显示要求，企业机构和员工账户及设备是一对多的关系，一个企业可以有多个设备和多个员工账户，而一个员工账户和一个设备只属于一个企业。因此，根据 E-R 图转换原则可以确定企业机构所包含的属性有：机构编号、机构名称、机构描述。该实体对应数据库中 T_Org 表，其结构如表 3.1 所示。

表 3.1 企业机构 T_Org 表结构

序号	字段名	属性	类型	长度	主键	描述
1	F_ID		Int		是	

2	F_Code	机构编号	Varchar	10	否	机构的编号，通常为机构的简写
3	F_Caption	机构名称	Nvarchar	20	否	机构的全称
4	F_Memo	机构描述	nvarchar	100	否	一些机构的备注

2) 员工账户

根据 E-R 图显示，员工的账户与机构是多对一的关系，员工帐号与设备是一对多的关系。因此，根据 E-R 图的转化规则，实体员工账户其中的属性有：机构编号，员工编号，员工名称，账户密码，备注，用户推送标识。该实体对应数据库中的 T_User 表，其结构如表 3.2 所示。

表 3.2 员工账户 T_User 表结构

序号	字段名	属性	类型	长度	主键	描述
1	F_ID		Int		是	员工的唯一标识
2	F_OrgID	机构编号	Int		否	所属机构的机构 ID
3	F_Code	员工编号	Varchar	100	否	员工登录系统的帐号
4	F_Caption	员工名称	Nvarchar	100	否	员工在系统中显示的名称
5	F_Memo	账户密码	Nvarchar	1000	否	账户的密码
6	F_Password	备注	nvarchar	20	否	
7	F_ClientID	用户推送标识	Varchar	150	否	个推用户 clientid

3) 设备

根据 E-R 图所示，设备与员工是多对一的关系，设备与设备监测点是一对多的关系，根据 E-R 图转化原则，实体设备中所拥有的属性有：设备 ID、

物理 MAC 地址、企业机构 ID、设备编号、设备名称、设备类型。该实体对应数据库中 T_Dev 表，其表结构如表 3.3 所示。

表 3.3 设备 T_Dev 表结构

序号	字段名	属性	类型	长度	主键	描述
1	F_ID	设备 ID	Int		是	设备的唯一标识
2	F_Guid	物理 MAC 地址	Varchar	100	否	OPC 获取的设备 MAC
3	F_OrgID	企业机构 ID	Int		否	所属企业机构的 ID
4	F_Code	设备编号	Varchar	100	否	系统中显示的设备编号
5	F_Caption	设备名称	Nvarchar	100	否	设备的名称
6	F_DevType	设备类型	Varchar	10	否	

4) 设备监测点

根据 E-R 图所示，设备监测点与设备是多对一的关系，设备监测点与历史数据是一对多的关系，与实时数据是一对一的关系。根据 E-R 图转化原则，实体设备监测点中所拥有的属性有：设备监测点 ID、物理 MAC 地址、设备 ID、设备监测点编号、设备监测点名称、数据类型、设备报警上限值、设备报警下限值、设备数据上传类型、设备开关标识、设备数据单位。该实体对应数据库中 T_DevItem 表，其表结构如表 3.4 所示。

表 3.4 设备 T_DevItem 表结构

序号	字段名	属性	类型	长度	主键	描述
1	F_ID	设备监测点 ID	Int		是	设备监测点的唯一标识
2	F_Guid	物理 MAC 地址	Varchar	100	否	监测点的 MAC 地址

3	F_DevID	设备 ID	Int		否	监测点所属设备 ID
4	F_Code	设备监测点编号	varchar	100	否	
5	F_Caption	设备监测点名称	Varchar	100	否	系统显示监测点的名称
6	F_DataType	数据类型	Nvarchar	20	否	
7	F_WarnUp	设备报警上限值	Real		否	根据报警类型设置报警上限值
8	F_WarnDown	设备报警下限值	Real		否	根据报警类型设置报警下限值
9	F_UploadFlag	设备数据上传类型	Varchar	20	否	
10	F_DevOnOffFlag	设备开关标识	Int		否	
11	F_Unit	设备数据单位	nvarchar	10	否	OPC 获取数据的单位

5) 实时数据

根据 E-R 图所示，实时数据与设备监测点是一对一的关系。根据 E-R 图转化原则，实时数据中所拥有的属性有：实时数据 ID、设备监测点 ID、实时数据获取时间、数据、报警类型。该实体对应数据库中 T_Real 表，其表结构如表 3.5 所示。

表 3.5 实时数据 T_Real 表结构

序号	字段名	属性	类型	长度	主键	描述
1	F_ID	实时数据 ID	Int		是	实时数据 唯一标识
2	F_DealItemID	设备监测 点 ID	Int		否	实时数据 所属设备 监测点 ID
3	F_Dt	实时数据 获取时间	Datetime		否	该数据的 获取时间
4	F_Val	数据	Real		否	
5	F_WarnFlag	报警类型	int		否	根据此值 确认报警 的方式

6) 历史数据

根据 E-R 图所示，实时数据与设备监测点是多对一的关系。根据 E-R 图转化原则，实时数据中所拥有的属性有：历史数据 ID、设备监测点 ID、历史数据获取时间、数据。该实体对应数据库中 T_Real 表，其表结构如表 3.5 所示。

表 3.5 实时数据 T_Real 表结构

序号	字段名	属性	类型	长度	主键	描述
1	F_ID	历史数据 ID	Int		是	历史数据 唯一标识
2	F_DealItemID	设备监测 点 ID	Int		否	历史数据 所属设备 监测点 ID

3	F_Dt	历史数据 获取时间	Datetime		否	该数据的 获取时间
4	F_Val	数据	Real		否	

3.6 本章小结

本章主要介绍了工业设备监控信息发布系统的设计原则，设计思想及具体设计，另外还对数据库进行了详细的说明。

第 4 章 工业设备监控信息发布系统详细设计

本系统开发的过程当中，主要运用了 HTML5+ 的移动应用开发框架，该框架在 HTML5 的基础上进一步扩充了其 JS 的 API，为移动开发提供了便利，同时可以适用于任何目前市面上的移动系统。其次，该系统的前端框架利用了腾讯公司开发的 FrozenUI，该框架为响应式结构，使得整个系统更好地适应移动设备。在服务器和移动设备的数据交换上，主要采用了跨域异步访问的方法，使得整个系统快速，有效，流畅地完成各项功能，利用 JSON 数据的交换，使得系统变得易于维护和移植。

因此，本章将基于以上技术点分别介绍本系统四大模块的详细设计。

4.1 移动应用前端设计

移动应用前端设计模块的组成主要有两个部分：用户登录与信息采集和设备信息显示。该模块的主要实现思想为：利用 frozenui 的跨平台性和页面自动响应性使得网页能够在手机屏幕上与原生 APP 拥有一样的视觉效果，其次，利用 HTML5 和新出现的 5+ 系列 API 完善网页在手机上的硬件设备调用功能，使得整个系统拥有较强的可移植性。百度模板引擎的出现使得动态刷新网页更加方便，传统的动态网页技术主要利用 DOM 对网页的标签进行操作，而百度模板引擎则提供了一种模板与数据合成的概念，在系统开发的过程当中，需要动态刷新的页面首先要提供模板页面，该页面中需要包含有合成函数传递的数据参数的值标签，利用 JS 程序语言完成数据与模板的合成，这种方式大大提高了整个系统的前端实用性和响应速度。业务逻辑类的设计则充分利用 JS 是弱语言的程序设计语言，将每一种功能的相应操作函数封装成“类”方便从全局进行调用，这样大大提高了系统和软件的可读性和可调试性。下面将介绍移动应用前端设计模块的两个组成部分。

(1) 用户登录与信息采集

用户在初次打开移动端的应用时，初次访问是没有任何的数据供用户进行数据请求，因此，系统需要提醒用户登录系统。用户需要提供自己的账户和密码以及选择自己所属的机构，随后与服务器进行交互，若登录失败则需重新提供信息。但是，若用户二次访问移动应用，则自动根据历史数据获取用户的基本信息。完成以上功能后，用户被引导至个人信息页面，在该页面中，用户可以查看自己的基本信息，包括用户名称，所属机构和用户编号。另外，用户可以修改或者上传自己的头像，还可以更改密码。若需要更换用户帐号或者退出系统，则可以注销登录。

用户登录与信息采集模块分为以下几个部分进行分别实现：

1) 判断用户是否登录

由于该移动端本地数据保存在 `js/data.js` 中，因此可以直接访问判断是否有历史数据，判断的具体逻辑为：

```
if (user.id == 0) {
    Cs_GetData(main.tplDefault, function (res) {
        var aHtml = bt(res);
        var aDiv = $(".ui-container");
        aDiv.html(aHtml);
        aDiv.trigger("create");
    });
} else {
    dev.load();
}
```

该部分是根据 `user.id` 的值判断用户是否登录，若没有登录，则显示 `main.tplDefault` 的内容，若已经登录则直接访问设备信息显示模块。

2) 用户登录

该操作为初次访问系统的用户或者注销账户的用户所做，需要用户提供基本信息与数据库进行数据交互并判断，用户提供信息的具体代码实现如下：

```
showlogin: function () {
    main.clearcontainer();

    var el = $.loading({
        content: '加载中...',
    });
    Cs_GetData(profile.tplLogin, function (res) {
```

```

        var content = res;
        Cs_GetDataFromWeb(appdata.url + "getallorg", "", function
(res) {
            appdata.org = res.org;
            var aHtml = bt(content, { tplData: appdata.org });
            var aDiv = $(".ui-container");
            aDiv.html(aHtml);
            aDiv.trigger("create");
            el.loading("hide");
        });
    });
}

```

用户点击登录，则触发 `profile.login()` 判断信息的正误：

```

login: function () {
    var e = $.loading({
        content: '加载中...',
    });
    ...
    Cs_GetDataFromWeb(appdata.url + "certilogin", aJson, function
(res) {
        if (typeof (res.info) == 'undefined') {
            var el = $.tips({
                content: '登录失败，请检查密码',
                stayTime: 2000,
                type: "warn"
            });
            el.on("tips:hide", function () {
                console.log("tips hide");
            });
            e.loading("hide");
        } else {
            ...
            e.loading("hide");
            profile.showinfodetail();
        }
    });
}

```

登录成功后，显示个人的信息：

```

showinfodetail: function () {
    main.clearcontainer();
    var e = $.loading({
        content: '加载中...',
    });
}

```

```

    });
    Cs_GetData(profile.tplInfodetail, function (res) {
    ...//获取模板，完成显示
    });
}

```

该功能的界面设计主要考虑响应式和美观性及事件的反应速度，因此利用了 frozenui 中的一些标签和类，如 ui-header、ui_footer、ui-border-t 等，该功能的界面设计图如图 4.1 所示。

图 4.1 用户登录界面设计图

如以上所示，该部分的代码可以较好地确认用户的信息，同时也提升了用户体验的效果，基本完成用户登录的功能。

3) 信息采集

信息采集主要包括头像的上传和密码的修改，若用户第一次登录，需要初始

化头像，随后可以对头像进行更改，密码的修改主要涉及到密码的比较问题，若修改失败或者上传失败，则给出相应的提示。

头像上传功能代码如下所示：

```
uploadimg:function(res) {
    var path = res;
    var task =
plus.uploader.createUpload(appdata.url+"uploadfile",
    { method: "POST", blocksize: 204800, priority: 100 },
    function (t, status) {
        // 上传完成
...
        alert("上传成功");
    } else {
        alert("Upload failed: " + status);
    }
    }
);
task.addFile(path, { key: "testdoc" });
task.addData("userid", user.id);
//task.addEventListener( "statechanged", onStateChanged,
false );
task.start();
}
```

该功能利用 HTML5+的 plus.gallery.pick() 函数从移动设备的相册中选择图片，再利用 plus.uploader.createUpload() 将图片信息和用户编号上传到服务器上，完成头像的上传和修改操作。

密码修改功能代码如下所示：

```
changepass: function () {
    ...//获取三个密码
        profile.deleteallinput();
...//密码判断
        Cs_GetDataFromWeb(appdata.url + "modifyPass", aJson,
function (res) {
...//密码上传
    }
}
```

该功能较为简单，同时也很好得体现了 B/S 结构的优势，即一部分逻辑判断放在 B 端，减轻服务器的压力，同时也能够快速地给用户反馈。至此，用户登录

与信息采集功能全部实现。

(2) 设备信息显示

设备的信息显示是本系统的重要组成部分，是工业设备监控信息发布系统的核心功能，主要为根据用户提供的信息显示其所拥有权限访问的设备基本信息，同时，对设备监测点的实时数据和历史数据进行查看、筛选和分析，设备信息显示的功能需要分为若干个模块进行分别的完成，具体如下：

1) 设备信息及筛选

系统根据用户的 ID 从服务器中获取所有的设备并显示，具体实现代码如下：

```
load: function () {
    main.clearcontainer();
    var el = $.loading({
        content: '加载中...',
    });
    Cs_GetData(dev.tplDevList, function (res) {
        var content = res;
        var aJson = {
            id: user.id
        };
        Cs_GetDataFromWeb(appdata.url + "getAllDev", "", function
(res) {
            dev.allDev = res.alldev;
            Cs_GetDataFromWeb(appdata.url + "getDevInfo", aJson,
function (res) {
                dev.devList = res.devlist;
                dev.getalllabel();
            ...//显示页面
            });
        });
    }
}
```

该功能拥有一个核心的算法，即设备类型筛选，主要完成的功能为用户根据想要查询的设备类别选择设备，系统自动显示其请求的所有设备信息，其算法流程图如图 4.2 所示。

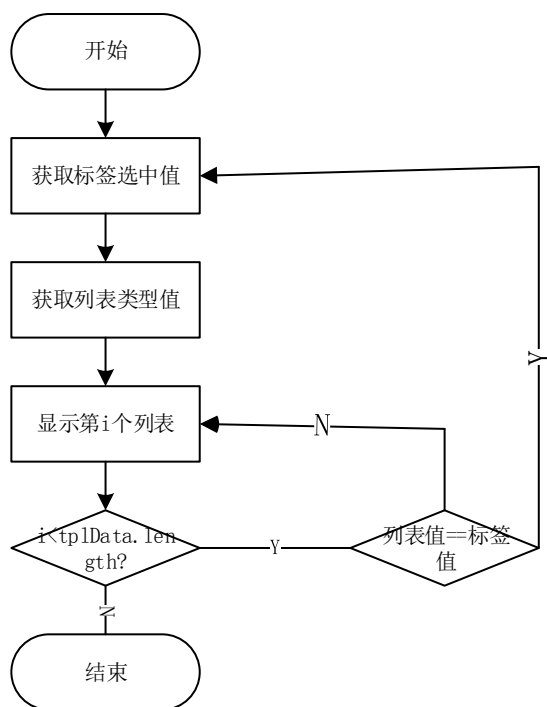


图 4.2 设备类型筛选算法流程图

该功能的界面文件在 tpl/DevList.html 中，在界面设计过程中，主要采用列表的显示方式，这样使得信息清晰明了，界面也很美观。设备信息界面如图 4.3 所示。



图 4.3 设备信息界面设计图

2) 设备监测点实时数据及筛选

根据用户选择的设备显示该设备所有监测点的实时数据，具体的代码实现如下所示：

```
showDevDetailInfo: function (i) {  
    ...//设备信息获取  
    Cs_GetDataFromWeb(appdata.url + "getDevDetailData", aJson,  
    function (res) {  
        dev.aDevDetail = res.devdetail;  
        Cs_GetData(dev.tplDevDetail, function (aares) {  
            var content = aares;  
            Cs_GetDataFromWeb(appdata.url + "getDevReal", aJson,  
            function (ares) {  
                dev.devReal = ares.realinfo;  
                HC.setRelationByProperty(dev.aDevDetail,  
                dev.devReal, "RealData", "F_ID", "F_DeItemID");  
                ...//页面显示  
            });  
        });  
    });  
});  
}
```

该功能中拥有一个核心的显示算法，为的是使页面模板与数据分开获取再一同显示，其算法流程图如图 4.4 所示。

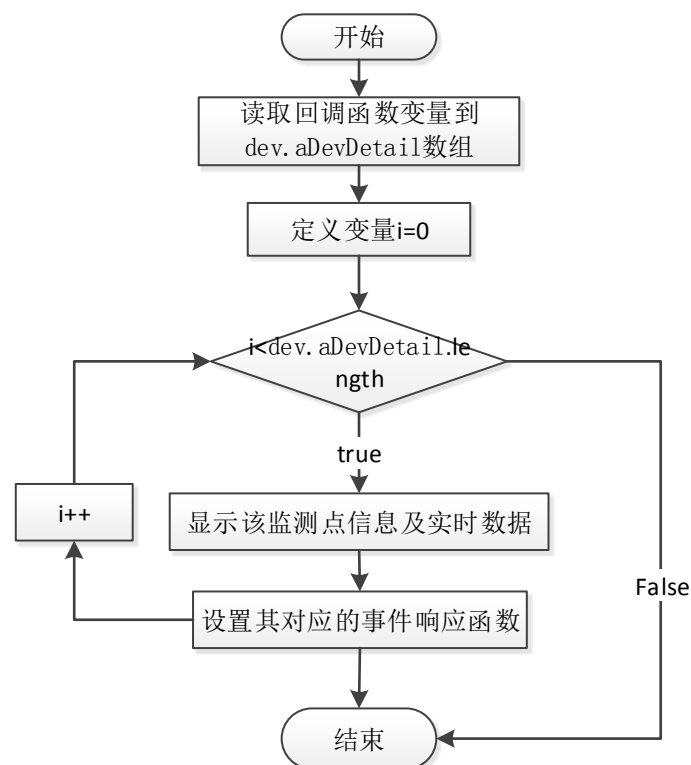


图 4.4 设备监测点遍历算法流程图

用户还可以根据自己在搜索框中输入的数据显示拥有相应字符串的监测点列表，具体代码如下所示：

```
searchItem: function () {  
    var keyword = $("#search").val();  
    var ul = $(".ui-list");  
    var lis = ul[0].children;  
    console.log("test");  
    for (var i = 0; i < lis.length; i++) {  
        if  
(lis[i].children[0].children[0].innerHTML.indexOf(keyword) >= 0) {  
            lis[i].style.display = "";  
        } else {  
            lis[i].style.display = "none";  
        }  
    }  
}
```

该功能的界面实现采用设备信息功能的页面模式，主要运用了 ui-list、ui-button、ui-href 等类，其功能界面设计图如图 4.5 所示。



图 4.5 设备监测点界面设计图

3) 历史数据展示及筛选

用户选择某一设备监测点，即可进入该监测点的所有历史数据展示页面，在该页面中，用户可以筛选近一周或近一个月的所有数据，方便对设备监测点进行分析，首先从服务器上下载所有的历史数据，根据用户的选择，引用 **echart** 图表框架将所有的数据显示在页面中，核心代码如下所示：

```
showhistory: function (i) {
    dev.itemtemp = i;
    var itemid = dev.aDevDetail[i].F_ID;
    var aJson = {
        devitemid: itemid
    };
    Cs_GetDataFromWeb(appdata.url + "gethistory", aJson, function
(res) {
```

```

        dev.historyData = res.his;
        Cs_GetData(dev.tplhis, function (ares) {
            var content = ares;
...//显示页面
        });
    });
}
, showchart: function (j) {
...//规定时间
    var date = [];
    for (var i = index; i < dev.historyData.length; i++) {
        date.push((dev.historyData[i].F_Dt.substring(0,
dev.historyData[i].F_Dt.indexOf(".")).replace(" ", "\n"));
    };
    var value = [];
    for (var i = index; i < dev.historyData.length; i++) {
        value.push(dev.historyData[i].F_Val);
    };
...//设置表格
    myChart.setOption(option);
}

```

该功能的实现和界面主要采用 **echart** 这一网页图表框架，**ECharts** 是一个纯 **Javascript** 的图表库，可以流畅的运行在移动设备上，兼容当前绝大部分浏览器，底层依赖轻量级的 **ZRender**，提供个性化定制的数据可视化图表。其核心代码为：

```

Var myChart= echarts.init(document.getElementById('chart'));

```

最终历史数据展示界面设计图如图 4.6 所示。

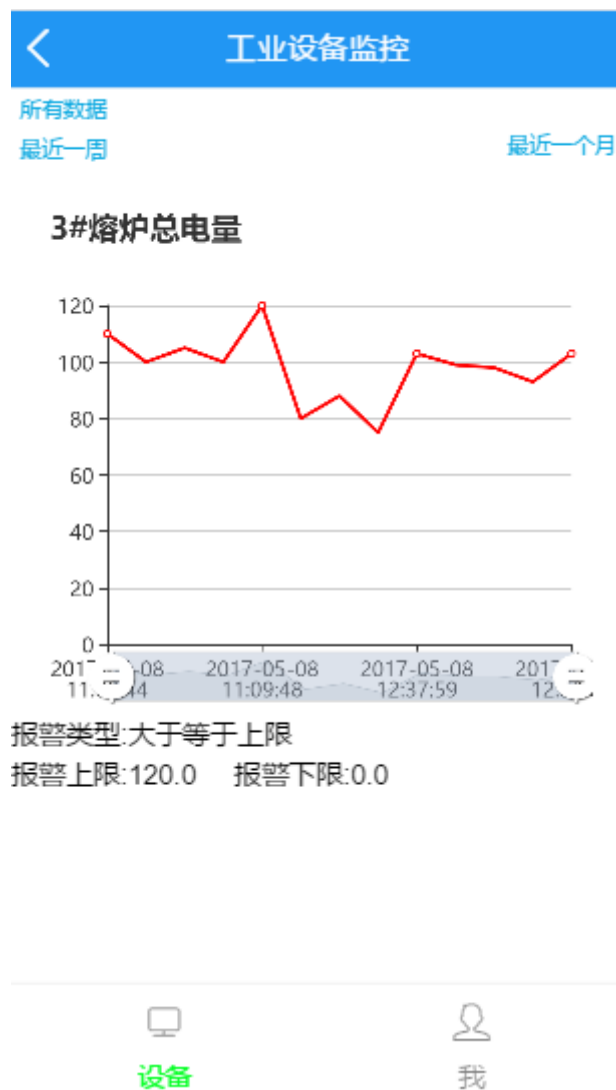


图 4.6 历史数据界面设计图

4.2 数据请求与获取

数据请求与获取模块在工业设备监控信息发布系统中起着至关重要的作用，主要包括两方面的功能：服务端和客户端。在客户端中，需要根据用户的操作获取相应的参数，利用 JSONP 的跨域异步请求方式向服务器发送请求。在服务端，服务器程序需要将接受到的数据与相应的服务进行结合，从服务器数据库中取得需求数据，再以 JSON 的格式传回客户端，完成整个系统数据的通信。

4.4.1 服务端

服务端的程序主要用 Java 中的 Servlet 进行数据通信，该方式简单快捷且

效率较高。首先需要获取用户端传的参数，随后通过 JDBC 访问数据库获取 ResultTable，将此类型数据利用 JSONUtil 类转化为 JSON 格式数据，连同回调参数一同传回用户端。基础算法流程图如图 4.9 所示。

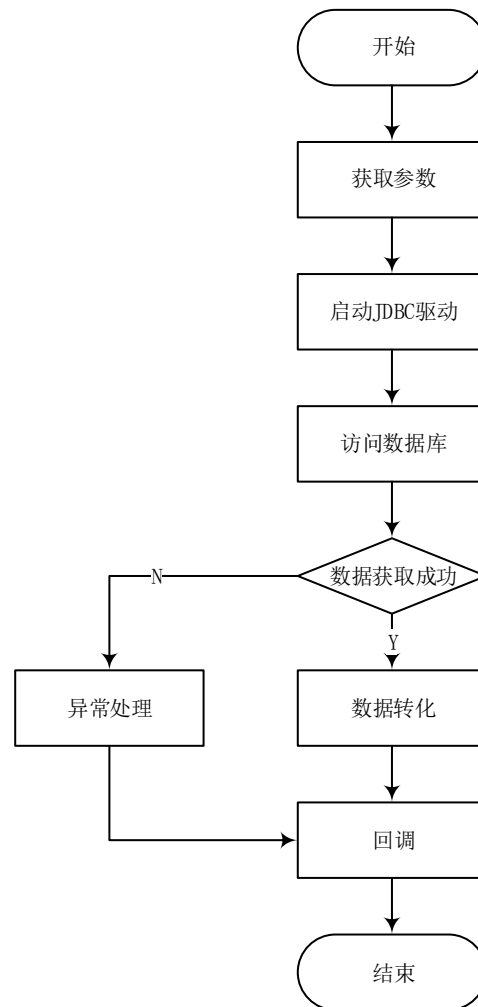


图 4.9 服务端通信数据处理算法流程图

其核心代码主要有 JDBC 操作，JSON 数据操作，异常处理，除此之外，图片上传代码也从属于服务端数据通信，如下所示：

数据库连接

```
public Db() {  
    ...//初始设置  
    try {  
        con=DriverManager.getConnection(url, user, password);  
    } catch (SQLException e) {  
        // TODO Auto-generated catch block
```

```

        e.printStackTrace();
    }
}

```

获取 resultSet:

```

public ResultSet getRS(String sql) throws SQLException{
    statement=con.createStatement(ResultSet.TYPE_SCROLL_SENSITIVE,ResultSet.C
ONCUR_READ_ONLY);
    rs=statement.executeQuery(sql);
    return rs;
}

```

Json 数据转换:

```

public String resultSetToJson(ResultSet rs) throws SQLException,JSONException
{
    ...// json 数组
    ...// 获取列数

    // 遍历 ResultSet 中的每条数据
    while (rs.next()) {
        ...// 遍历每一列
    }
    return array.toString();
}

```

图片上传代码:

```

String savePath = this.getServletContext().getRealPath("/upload");
...//目录确认
DiskFileItemFactory factory = new DiskFileItemFactory();
...//设置
ServletFileUpload fileUpload = new ServletFileUpload(factory);
fileUpload.setSizeMax(1024 * 1024 * 100);
List<FileItem> fileItemList = null;

```

```

try{
    fileList = fileUpload.parseRequest(request);
}
...//异常处理
...//设置图片
...//获取图片

```

该模块最终完成整个系统的数据处理，是整个系统后台的核心部分，保证系统高效流畅地运行。

4.4.2 用户端

用户端的数据通信处理主要是跨域访问和异步访问的思想，本系统采用 jquery 的 AJAX 访问函数，完成整个功能，主要代码如下所示：

```

function Cs_GetDataFromWeb(aUrl, aJson, aCallBack) {//跨域请求
$.ajax({
    type: 'get',
    url: aUrl+"?callback=?",
    dataType: 'jsonp',
    data:aJson,
    contentType: 'application/json;charset=utf-8',
    cache: false, //默认值true
    async: false,
    success: function (json) {
        aCallBack(json);
    },
    error: function (cer) {
        aCallBack(cer);
    }
})
}

```

4.3 监控信息推送与分析

推送功能为工业设备监控信息发布系统的核心功能，在当前的功能下，该系统只能查看设备和其检测点的信息，但当 OPC 系统监测到某一监测点的数据达到报警的要求，用户无法及时得知，此时推送功能应运而生。推送初始于工业设

备监控信息采集系统中，工业设备监控信息发布系统的服务器端拥有一个 http 请求接收类，该类在设备监测点数据变化时接受到设备监测点的信息，通过数据请求与获取模块与数据库中的报警数据进行比较，将透传消息进行格式化传递给前端页面，此时前端页面负责监听的触发函数对透传消息进行整合和解析以决定是否发起本地推送，同时更新该设备监测点的数据。该功能采用个推推送插件作为后台的推送框架。

推送服务器端程序的算法结构图如图 4.7 所示。

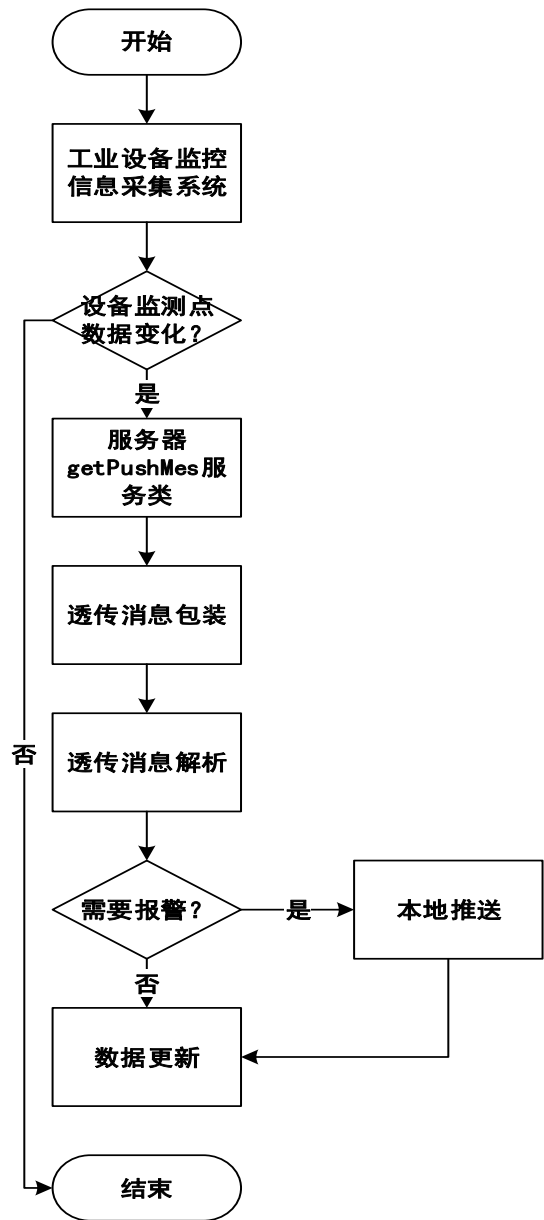


图 4.7 推送算法流程图

该功能的核心代码如下所示：

服务端监听服务类:

```
String callback=request.getParameter("callback");  
...//获取推送设备监测点信息
```

```
try {  
    rs = db.getRS(sql);  
    jw.judge(rs,val,itemID);  
}
```

判断是否报警:

```
...//报警数据获取  
String warntype = rs.getString("F_WarnType");  
double up = rs.getDouble("F_WarnUp");  
if(warntype.equals("none")){  
    p.sendmes(itemid,val,"");  
}  
...//报警判断  
}
```

推送服务类:

```
IGtPush push = new IGtPush(host, appKey, masterSecret);  
TransmissionTemplate template = linkTemplateDemo(descreption,val);  
...//设置推送参数
```

```
message.setData(template);  
message.setPushNetWorkType(0);
```

...//设置发送目标

```
IPushResult ret = null;  
try {  
    ret = push.pushMessageToSingle(message, target);
```

...//推送与异常处理

该模块的界面效果图如图 4.8 所示。



图 4.8 推送界面设计图

4.4 本章小结

本章主要就工业设备监控信息发布系统的各个功能模块的设计详情进行了概述，对算法、代码和思路进行了阐述。

第 5 章 系统调试与实现

5.1 调试过程中的主要问题和解决方法

5.1.1 HTML5+框架引用失败

1) 存在的问题

HTML5+大大方便了本系统的移动设备硬件设施调用的过程，但是在开发初始阶段，误认为此框架已经支持所有设备和浏览器，因此，导致整个项目中相册不能调用，推送模块无法实现。

2) 解决办法

经过查询资料和认真阅读 HTML5+的 API，需要在 HBuilder 的环境下新建 Web App 项目，同时需要将相册的配置和推送的配置添加到 Manifest.json 中，如图 5.1 所示。

```
"Gallery": {
    "description": "访问系统相册"
},
"Geolocation": {
    "description": "访问位置信息"
},
"Invocation": {
    "description": "使用Native.js能力"
},
"Orientation": {
    "description": "访问方向感应器"
},
},
```

图 5.1 Manifest.json 文件配置

5.1.2 页面锚点位置

1) 存在的问题

在设备列表中，如果用户选择了一个设备监测点进行查看，在下滑的过程当中返回整个设备列表，将会出现列表错位的情况，具体情况如图 5.2，图 5.3 所



图 5.2 进入监测点列表前



图 5.3 返回设备列表时

这样的效果在用户浏览的过程当中有着非常差的用户体验，其次，在再次选测设备监测点时，事件的触发也会出现错误，最终会导致整个系统崩溃。其根本原因是因为由于整个系统的地址是没有变化的，当加载新的内容时，页面的锚点会保持不变，因此出现错位的情况。

2) 解决办法

在处理此问题的过程当中，考虑到网页标签的结构，最终采用了清空整个设备列表重新加载的方法。该方法不仅可以完美地解决以上问题，还能够代码复用，解决整个系统的代码冗余。主要的代码实现为：

```
clearcontainer: function () {  
    $(".ui-container").html("");  
}
```

5.1.3 注销功能失效

1) 存在的问题

注销功能为用户退出整个系统的关键功能，在注销时保证了整个系统的数据清空，保证了用户的信息不泄露和整个系统的安全性。点击注销按钮不出现提示，直接注销，且用户内容没有清除。

2) 解决办法

该问题的导致主要是因为以下标签

```
<buttonclass="ui-btn-dangerui-btn-lg" id="logoutbtn" onclick="profile.showdialog()">
```

注销</button>被放在 form 表单中，而 form 表单的默认提交项为本页，因此会导致直接刷新页面。主要的解决方案有两个：

- (1) 将 button 标签到 form 外部。
- (2) 将 form 的提交项置空。

最终的效果图如图 5.4，图 5.5 所示。

图 5.4

图 5.5

5.1.4 页面不适应移动设备



1) 存在的问题

FrozenUI 为当前较为流行的移动应用框架，而在开发伊始，程序运行在移动设备上时，存在了不适配的情况。如图 5.6 所示。

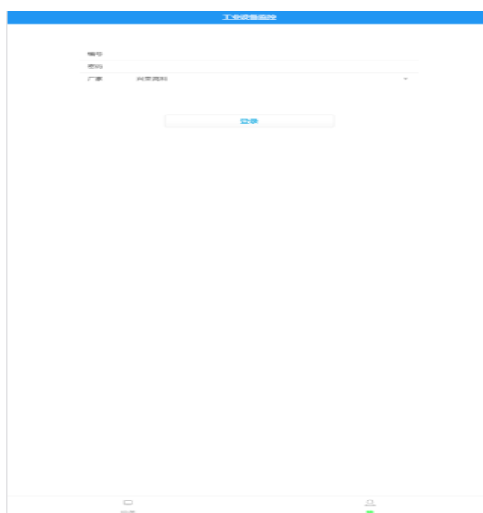


图 5.6

2) 解决办法

根据 frozenUI 的 API 显示，移动端设备的适配需要在 head 标签中加入一段引用：

```
<meta name="viewport" content="initial-scale=1, maximum-scale=1, user-scalable=no">
```

此问题得到解决，在移动设备上运行良好。

5.2 系统运行成果图

系统开发完成后，通过调试，在移动设备上运行状况良好，以 Vivo 手机为例：



图 5.7 登录界面运行成果图

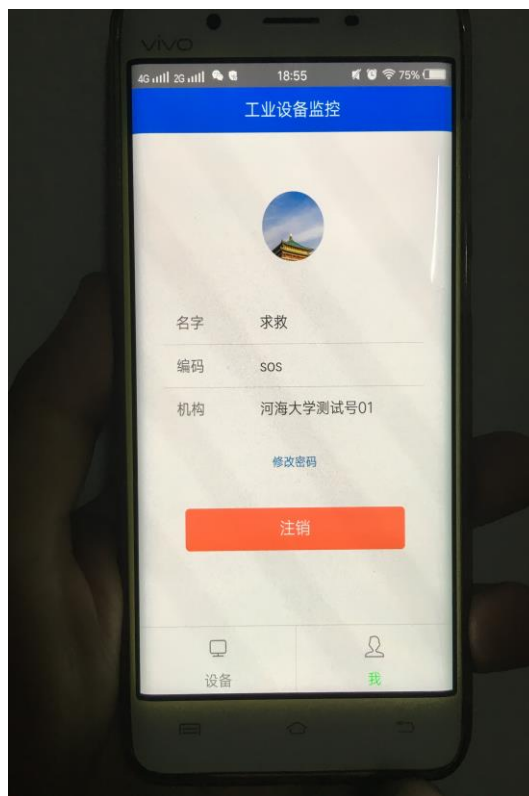


图 5.8 个人信息界面运行成果图

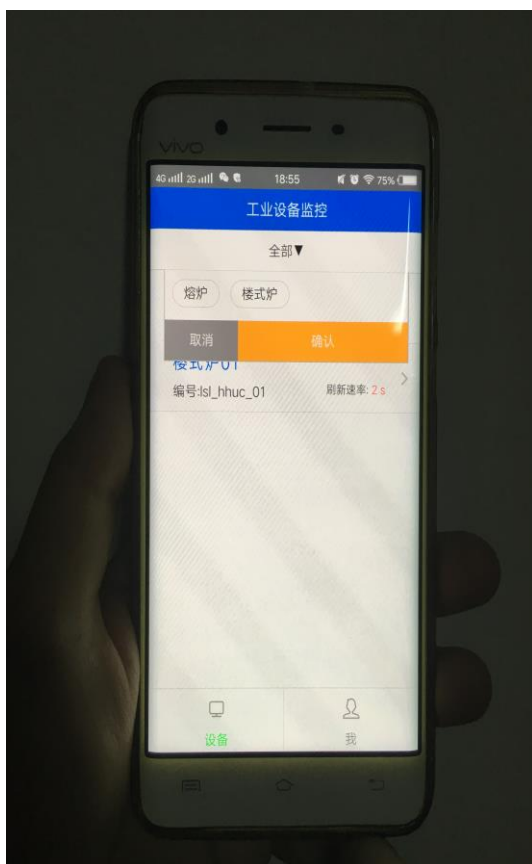


图 5.9 设备列表运行成果图

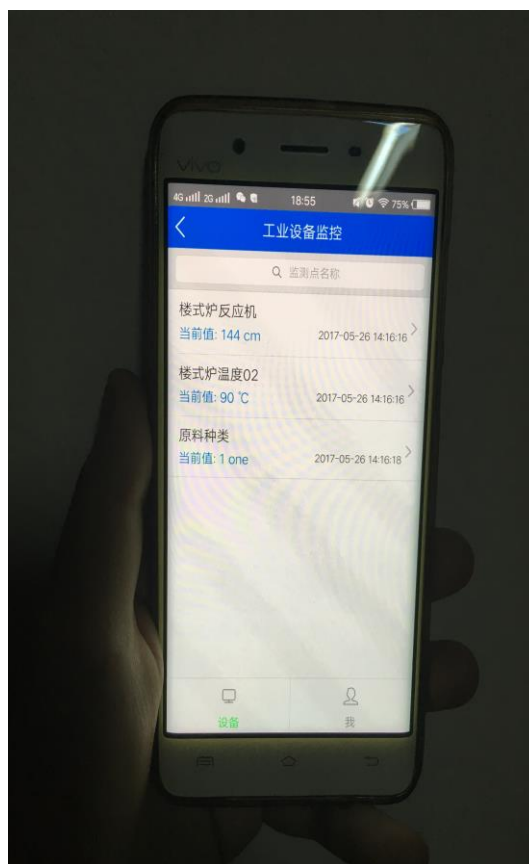


图 5.10 设备监测点列表运行成果图



图 5.11 历史数据运行成果图



图 5.12 推送运行成果图

5.3 本章小结

本章总结了系统调试过程当中问题和解决方案，以及展示了最终的运行成果图。

第 6 章 总结与展望

6.1 总结

本课题来源于江苏兴荣高新科技股份有限公司和河海大学合作的项目，旨在工业生产现场构建的一套信息处理系统，实时收集现场生产设备和仪表等数据。本设计为工业设备监控系统的一部分，为满足工业设备监控系统的信息发布工作，利用 AJAX、HTML5+ 等移动应用开发技术设计并完成工业设备监控信息发布系统，实现了利用权限分配模块的设备数据访问，工业设备监控信息实时更新与推送，设备监测信息多方式展示等功能。在研发本系统时，首先确认了设计的思路，采用跨平台的移动应用框架，用来提高系统的可移植性；采用面向对象程序设计模式，使得系统的可维护性提高，降低开发的难度；B/S 的软件系统体系结构大大降低了整个系统的服务器压力，使得系统的效率得到提高。随后利用高效的开发工具，使得整个系统拥有一个稳定和有效的硬件平台。其次分别进行数据库的逻辑和结构设计，满足系统的数据需求和运行要求。在开发的过程当中，该系统严格按照软件工程的开发流程，主要完成以下工作内容：

1) 工业设备监控信息发布系统总体设计

该系统主要利用开源前端框架对移动应用程序界面和功能进行开发，利用 AJAX 技术向服务器端发送数据请求以等待服务器程序的响应。在服务器端，利用 JDBC 和 JSON 数据转化类将数据封装返还给移动应用，利用百度模板和设计好的页面模板将数据嵌入页面显示以完成移动应用的设计和功能。采用个推第三方推送服务，根据已定协议将设备监测点的变化情况透过透传消息传递给移动应用完成数据实时更新和报警的功能。在此过程中，完成对数据库的设计和数据库精度的规范。

2) 页面模板设计

利用 frozenui 前端框架，对设备列表、设备监测点列表、历史数据等页面进行模板设计，利用 JS 脚本语言和百度模板使得页面快速实现大量数据与页面的结合，提高用户体验感。

3) 利用透传消息完成数据推送

透传消息为移动应用信息推送的一种重要手段，即在用户不知情的情况下通过服务器对移动终端进行信息推送，用户无需对软件进行操作或者向服务器发送请求，即可获得当前设备的实时数据或被动接收服务器的警报推送，该模块为该移动应用提供了实时数据的更新功能，使得系统的同步性有了保证。

3) 数据请求与获取

数据请求与获取为整个工业设备监控信息发布系统的主要数据功能的支撑，在移动端根据用户的操作提供数据请求参数，利用 JSONP 技术将参数传入服务端程序，服务端记录回调参数，将参数进行分析，利用 JDBC 从服务器数据库中获取数据，将数据以 JSON 格式连同回调参数一同返回移动端以备显示数据，满足了整个系统的高效性和可操作性。

6.2 展望

该系统在开发的过程当中，由于个人技术和时间的限制，还有存在一些不足和待开发功能，仍需改进。

1) 订阅服务

目前的报警功能为：只要达到报警要求，即向用户推送的方式。在实际的应用环境中，用户所拥有权限的设备监测点可能会很多，若同一时间所有监测点都报警，将会对服务器产生较大的压力。因此，用户可以根据设备监测点的优先级和关注程度对报警进行订阅，在报警时，若用户订阅则推送，若未订阅，则忽略。

2) 大数据分析

在 OPC 系统获取的数据逐渐增大的过程当中，数据量将会达到无法显示或者难以全面显示的程度，因此，通过大数据技术可以将设备监测点的数据进行分析，得出的结论或者结果可能对工业管理更加有意义，也使得整个系统能够拥有更有效的功能和应用前景。

3) 手势识别

本系统的开发过程当中，对于前端页面的设计仅停留在满足功能和样式上，对于一些移动设备的基本手势识别并未有所研究和实现，因此在之后的

开发过程当中，可以适当添加一些手势识别的项目，不仅能够提高用户体验，而且可以提高系统的可用性。

致谢

四年的时光很快就过去了，这四年对我来说是拥有着重大意义的一段时间，在大学中，我不仅学到了很多实用且深奥的知识，另外还初涉世事，完成一个不谙世事的少年到青年的转变，这对于即将步入社会的我来说是非常关键的。在毕业设计刚开始的时间，家人就一直叮嘱我要完成好大学的最后一门课程，给自己的大学生涯画上圆满的句号；牟艳老师在毕业设计完成的这段时间里给予我莫大的帮助，从软件的开发到文档的设计和完成，牟艳老师让我进一步深入理解了软件工程的相关知识，在此还要感谢牟艳老师四年来对我的栽培，使得我在科技创新上能够取得一些成绩；另外还要感谢我的可爱的同学们，在这四年来我们互帮互助，在毕业设计过程中，我们一起讨论一起克服困难，进一步加深了友谊。总之，感谢每一位在大学四年特别是在毕业设计过程中给予我帮助的老师 and 同学。

同时，还要感谢论文评审组的各位老师和专家在毕业设计与论文中给予的指导和建议！

参考文献

- [1] 陈博, 廖军. 移动 Web 应用的能力需求与发展浅析[J]. 移动通信, 2011, 35(5):32-35.
- [2] 党寿江, 王劲林, 曾学文,等. JSONP 研究及其在 IPTV 门户系统中的应用[J]. 微计算机信息, 2010, 26(30):183-185.
- [3] Gouveia C, Campos J, Rui A. Using HTML5 visualizations in software fault localization[C]// Software Visualization. IEEE, 2013:1-10.
- [4] 李培林. 安卓系统的应用及发展趋势展望[J]. 计算机光盘软件与应用, 2012(18):161-162.
- [5] 伍琛尧, 刘建煌. 基于 ZigBee 的工业设备监控系统[J]. 计算机与数字工程, 2007, 35(9):155-158.
- [6] 成经平. 基于网络的现代工业设备远程监控系统及关键技术[J]. 组合机床与自动化加工技术, 2003(6):59-61.
- [7] 赵炯, 周影. 面向 Android 的工业设备远程监控系统设计[J]. 机电工程, 2016, 33(12):1511-1516.
- [8] 唐俊开. HTML5 移动 Web 开发指南[M]. 电子工业出版社, 2012.
- [9] 李张永, 陈和平, 顾进广. 跨平台移动 Web 开发框架与数据交互方法[J]. 计算机工程与设计, 2014, 35(5):1827-1832.
- [10] 于志良, 裘国永. 基于 Ajax 技术开发移动 Web 应用[J]. 网络安全技术与应用, 2007(11):42-43.
- [11] 封宇, 陈宁江. 基于 MVVM 架构的移动 Web 前端展示方案[J]. 计算机与现代化, 2014(11):1-4.
- [12] 陈希文. 基于安卓的移动信息采集系统的设计与实现[D]. 武汉邮电科学研究院, 2016.
- [13] 路晔绵, 李轶夫, 应凌云,等. Android 应用第三方推送服务安全分析与安全增强[J]. 计算机研究与发展, 2016, 53(11):2431-2445.
- [14] 杨薇. 移动 app 推送系统的研究与设计[D]. 天津大学, 2015.
- [15] 吴大刚, 肖荣荣. C/S 结构与 B/S 结构的信息系统比较分析[J]. 情报科学, 2003, 21(3):313-315.
- [16] 查卫翔, 谭南林. ActiveX 控件在基于 B/S 结构的远程监控中的应用[J]. 北京交通大学学报, 2002, 26(1):58-62.

- [17] 扬黎明, 董传良, 董玮文. 服务器端动态网页技术--JSP+Servlet[J]. 计算机工程, 2001, 27(2):126-127.
- [18] 史梦安, 马壮. 一种基于 Servlet 的控制层软件框架设计[J]. 软件导刊, 2017, 16(3).
- [19] Damevski K, Shepherd D C, Schneider J, et al. Mining Sequences of Developer Interactions in Visual Studio for Usage Smells[J]. IEEE Transactions on Software Engineering, 2017, 43(4):359-371.
- [20] Jia N. Application of Echarts in Mobile Data Communication[J]. Mobile Communications, 2016.
- [21] 石博文. 浅谈面向对象和面向过程程序设计[J]. 电子世界, 2017(1):59-60.

外文文献翻译

THE TECHNIQUE DEVELOPMENT HISTORY OF JSP

The Java Server Pages(JSP) is a kind of according to web of the script plait distance technique, similar carries the script language of Java in the server of the Netscape company of server- side JavaScript(SSJS) and the Active Server Pages(ASP) of the Microsoft. JSP compares the SSJS and ASP to have better can expand sex, and it is no more exclusive than any factory or someone particular server of Web. Though the norm of JSP is to be draw up by the Sun company of, any factory can carry out the JSP on own system.

The After Sun release the JSP(the Java Server Pages) formally, the this kind of new Web application development technique very quickly caused the people's concern. JSP provided a special development environment for the Web application that establishes the high dynamic state. According to the Sun parlance, the JSP can adapt to include the Apache WebServer, IIS4.0 on the market at inside of 85% server product.

This chapter will introduce the related knowledge of JSP and Databases, and JavaBean related contents, is all certainly rougher introduction among them basic contents, say perhaps to is a Guide only, if the reader needs the more detailed information, pleasing the book of consult the homologous JSP.

1.1 GENERALIZE

The JSP(Java Server Pages) is from the company of Sun Microsystems initiate, the many companies the participate to the build up the together of the a kind the of dynamic the state web the page technique standard, the it have the it in the construction the of the dynamic state the web page the strong but the do not the especially of the function. JSP and the technique of ASP of the Microsoft is very alike. Both all provide the ability that mixes with a certain procedure code and is explain by the language engine to carry out the procedure code in the code of HTML. Underneath we are simple of carry on the introduction to it.

JSP pages are translated into servlets. So, fundamentally, any task JSP pages can perform could also be accomplished by servlets. However, this underlying equivalence does not mean that servlets and JSP pages are equally appropriate in all scenarios. The issue is not the power of the technology, it is the convenience, productivity, and maintainability of one or the other. After all, anything you can do on a particular computer platform in the Java programming language you could also do

in assembly language. But it still matters which you choose.

JSP provides the following benefits over servlets alone:

- It is easier to write and maintain the HTML. Your static code is ordinary HTML: no extra backslashes, no double quotes, and no lurking Java syntax.
- You can use standard Web-site development tools. Even HTML tools that know nothing about JSP can be used because they simply ignore the JSP tags.
- You can divide up your development team. The Java programmers can work on the dynamic code. The Web developers can concentrate on the presentation layer. On large projects, this division is very important. Depending on the size of your team and the complexity of your project, you can enforce a weaker or stronger separation between the static HTML and the dynamic content.

Now, this discussion is not to say that you should stop using servlets and use only JSP instead. By no means. Almost all projects will use both. For some requests in your project, you will use servlets. For others, you will use JSP. For still others, you will combine them with the MVC architecture. You want the appropriate tool for the job, and servlets, by themselves, do not complete your toolkit.

1.2 SOURCE OF JSP

The technique of JSP of the company of Sun, making the page of Web develop the personnel can use the HTML perhaps marking of XML to design to turn the end page with format. Use the perhaps small script future life of marking of JSP becomes the dynamic state on the page contents.(the contents changes according to the claim of)

The Java Servlet is a technical foundation of JSP, and the large Web applies the development of the procedure to need the Java Servlet to match with with the JSP and then can complete, this name of Servlet comes from the Applet, the local translation method of now is a lot of, this book in order not to misconstruction, decide the direct adoption Servlet but don't do any translation, if reader would like to, can call it as" small service procedure". The Servlet is similar to traditional CGI, ISAPI, NSAPI etc. Web procedure development the function of the tool in fact, at use the Java Servlet hereafter, the customer need not use again the lowly method of CGI of efficiency, also need not use only the ability come to born page of Web of dynamic state in the method of API that a certain fixed Web server terrace circulate. Many servers of Web all support the Servlet, even not support the Servlet server of Web directly and can also pass the additional applied server and the mold pieces to support the Servlet. Receive benefit in the characteristic of the Java cross-platform, the Servlet is also a terrace irrelevant, actually, as long as match the norm of Java Servlet, the Servlet is complete to have nothing to do with terrace and is to have nothing to do with server of Web. Because the Java Servlet is internal to provide the service by the line distance,

need not start a progress to the each claimsses, and make use of the multi-threading mechanism can at the same time for several claim service, therefore the efficiency of Java Servlet is very high.

But the Java Servlet also is not to has no weakness, similar to traditional CGI, ISAPI, the NSAPI method, the Java Servlet is to make use of to output the HTML language sentence to carry out the dynamic state web page of, if develop the whole website with the Java Servlet, the integration process of the dynamic state part and the static state page is an evil-foreboding dream simply. For solving this kind of weakness of the Java Servlet, the SUN released the JSP.

A number of years ago, Marty was invited to attend a small 20-person industry roundtable discussion on software technology. Sitting in the seat next to Marty was James Gosling, inventor of the Java programming language. Sitting several seats away was a high-level manager from a very large software company in Redmond, Washington. During the discussion, the moderator brought up the subject of Jini, which at that time was a new Java technology. The moderator asked the manager what he thought of it, and the manager responded that it was too early to tell, but that it seemed to be an excellent idea. He went on to say that they would keep an eye on it, and if it seemed to be catching on, they would follow his company's usual "embrace and extend" strategy. At this point, Gosling lightheartedly interjected "You mean disgrace and distend."

Now, the grievance that Gosling was airing was that he felt that this company would take technology from other companies and suborn it for their own purposes. But guess what? The shoe is on the other foot here. The Java community did not invent the idea of designing pages as a mixture of static HTML and dynamic code marked with special tags. For example, Cold Fusion did it years earlier. Even ASP (a product from the very software company of the aforementioned manager) popularized this approach before JSP came along and decided to jump on the bandwagon. In fact, JSP not only adopted the general idea, it even used many of the same special tags as ASP did.

The JSP is an establishment at the model of Java servlets on of the expression layer technique, it makes the plait write the HTML to become more simple.Be like the SSJS, it also allows you carry the static state HTML contents and servers the script mix to put together the born dynamic state exportation. JSP the script language that the Java is the tacit approval, however, be like the ASP and can use other languages(such as JavaScript and VBScript), the norm of JSP also allows to use other languages.

1.3 JSP CHARACTERISTICS

Is a service according to the script language in some one language of the statures

system this kind of discuss, the JSP should be see make is a kind of script language. However, be a kind of script language, the JSP seemed to be too strong again, almost can use all Javas in the JSP.

Be a kind of according to text originally of, take manifestation as the central development technique, the JSP provided all advantages of the Java Servlet, and, when combine with a JavaBeans together, providing a kind of make contents and manifestation that simple way that logic separate. Separate the contents and advantage of logical manifestations is, the personnel who renews the page external appearance need not know the code of Java, and renew the JavaBeans personnel also need not be design the web page of expert in hand, can use to take the page of JavaBeans JSP to define the template of Web, to build up a from have the alike external appearance of the website that page constitute. JavaBeans completes the data to provide, having no code of Java in the template thus, this means that these templates can be written the personnel by a HTML plait to support. Certainly, can also make use of the Java Servlet to control the logic of the website, adjust through the Java Servlet to use the way of the document of JSP to separate website of logic and contents.

Generally speaking, in actual engine of JSP, the page of JSP is the edit and translate type while carry out, not explain the type of. Explain the dynamic state web page development tool of the type, such as ASP, PHP3 etc., because speed etc. reason, have already can't satisfy current the large electronic commerce needs appliedly, traditional development techniques are all at to edit and translate the executive way change, such as the ASP → ASP+; PHP3 → PHP4.

In the JSP norm book, did not request the procedure in the JSP code part(be called the Scriptlet) and must write with the Java definitely. Actually, have some engines of JSP are adoptive other script languages such as the EMAC- Script, etc., but actually this a few script languages also are to set up on the Java, edit and translate for the Servlet to carry out of. Write according to the norm of JSP, have no Scriptlet of relation with Java also is can of, however, mainly lie in the ability and JavaBeans, the Enterprise JavaBeanses because of the JSP strong function to work together, so even is the Scriptlet part not to use the Java, edit and translate of performance code also should is related with Java.

1.4 JSP MECHANISM

To comprehend the JSP how unite the technical advantage that above various speak of, come to carry out various result easily, the customer must understand the differentiation of" the module develops for the web page of the center" and" the page develops for the web page of the center" first.

The SSJS and ASP are all in several year ago to release, the network of that time is still very young, no one knows to still have in addition to making all business, datas

and the expression logic enter the original web page entirely heap what better solve the method. This kind of model that take page as the center studies and gets the very fast development easily. However, along with change of time, the people know that this kind of method is unwell in set up large, the Web that can upgrade applies the procedure. The expression logic write in the script environment was lock in the page, only passing to shear to slice and glue to stick then can drive heavy use. Express the logic to usually mix together with business and the data logics, when this makes be the procedure member to try to change an external appearance that applies the procedure but do not want to break with its llied business logic, apply the procedure of maintenance be like to walk the similar difficulty on the eggshell. In fact in the business enterprise, heavy use the application of the module already through very mature, no one would like to rewrite those logics for their applied procedure.HTML and sketch the designer handed over to the implement work of their design the Web plait the one who write, make they have to double work- Usually is the handicraft plait to write, because have no fit tool and can carry the script and the HTML contents knot to the server to put together. Chien but speech, apply the complexity of the procedure along with the Web to promote continuously, the development method that take page as the center limits sex to become to get up obviously.

At the same time, the people always at look for the better method of build up the Web application procedure, the module spreads in customer's machine/server the realm. JavaBeans and ActiveX were published the company to expand to apply the procedure developer for Java and Windows to use to come to develop the complicated procedure quickly by" the fast application procedure development"(RAD) tool. These techniques make the expert in the some realm be able to write the module for the perpendicular application plait in the skill area, but the developer can go fetch the usage directly but need not control the expertise of this realm.

Be a kind of take module as the central development terrace, the JSP appeared. It with the JavaBeans and Enterprise JavaBeans(EJB) module includes the model of the business and the data logic for foundation, provide a great deal of label and a script terraces to use to come to show in the HTML page from the contents of JavaBeans creation or send a present in return. Because of the property that regards the module as the center of the JSP, it can drive Java and not the developer of Java uses equally. Not the developer of Java can pass the JSP label(Tags) to use the JavaBeans that the deluxe developer of Java establish. The developer of Java not only can establish and use the JavaBeans, but also can use the language of Java to come to control more accurately in the JSP page according to the expression logic of the first floor JavaBeans.

See now how JSP is handle claim of HTTP. In basic claim model, a claim directly was send to JSP page in. The code of JSP controls to carry on hour of the logic processing and module of JavaBeanses' hand over with each other, and the manifestation result in dynamic state bornly, mixing with the HTML page of the static state HTML code. The Beans can be JavaBeans or module of EJBs. Moreover, the more complicated claim model can see make from is request other JSP pages of the page call sign or Java Servlets.

The engine of JSP wants to chase the code of Java that the label of JSP, code of Java in the JSP page even all converts into the big piece together with the static state HTML contents actually. These codes piece was organized the Java Servlet that customer can not see to go to by the engine of JSP, then the Servlet edits and translate them automatically byte code of Java.

Thus, the visitant that is the website requests a JSP page, under the condition of it is not knowing, an already born, the Servlet actual full general that prepared to edit and translate completes all works, very concealment but again and efficiently. The Servlet is to edit and translate of, so the code of JSP in the web page does not need when the every time requests that page is explain. The engine of JSP need to be edit and translate after Servlet the code end is modify only once, then this Servlet that edited and translate can be carry out. The in view of the fact JSP engine auto is born to edit and translate the Servlet also, need not procedure member begins to edit and translate the code, so the JSP can bring vivid sex that function and fast developments need that you are efficiently.

Compared with the traditional CGI, the JSP has the equal advantage. First, on the speed, the traditional procedure of CGI needs to use the standard importation of the system to output the equipments to carry out the dynamic state web page born, but the JSP is direct is mutually the connection with server. And say for the CGI, each interview needs to add to add a progress to handle, the progress build up and destroy by burning constantly and will be a not small burden for calculator of be the server of Web. The next in order, the JSP is specialized to develop but design for the Web of, its purpose is for building up according to the Web applied procedure, included the norm and the tool of a the whole set. Use the technique of JSP can combine a lot of JSP pages to become a Web application procedure very expediently.

JSP six built-in objects

request for:

The object of the package of information submitted by users, by calling the object corresponding way to access the information package, namely the use of the target users can access the information.

response object:

The customer's request dynamic response to the client sent the data.

session object

1. What is the session: session object is a built-in objects JSP, it in the first JSP pages loaded automatically create, complete the conversation of management.

From a customer to open a browser and connect to the server, to close the browser, leaving the end of this server, known as a conversation.

When a customer visits a server, the server may be a few pages link between repeatedly, repeatedly refresh a page, the server should be

through some kind of way to know this is the same client, which requires session object.

2. session object ID: When a customer's first visit to a server on the JSP pages, JSP engines produce a session object, and assigned a

String type of ID number, JSP engine at the same time, the ID number sent to the client, stored in Cookie, this session objects, and customers on the establishment of a one-to-one relationship. When a customer to connect to the server of the other pages, customers no longer allocated to the new session object, until, close your browser, the client-server object to cancel the session, and the conversation, and customer relationship disappeared. When a customer re-open the browser to connect to the server, the server for the customer to create a new session object.

application target

1. What is the application:

Servers have launched after the application object, when a customer to visit the site between the various pages here, this application objects are the same, until the server is down. But with the session difference is that all customers of the application

objects are the same, that is, all customers share this built-in application objects.

2. application objects commonly used methods:

(1) public void setAttribute (String key, Object obj): Object specified parameters will be the object obj added to the application object, and to add the subject of the designation of a keyword index.

(2) public Object getAttribute (String key): access to application objects containing keywords for.

out targets

out as a target output flow, used to client output data. out targets for the output data.

Cookie

1. What is Cookie:

Cookie is stored in Web server on the user's hard drive section of the text. Cookie allow a Web site on the user's computer to store information on and then get back to it.

For example, a Web site may be generated for each visitor a unique ID, and then to Cookie in the form of documents stored in each user's machine.

If you use IE browser to visit Web, you will see all stored on your hard drive on the Cookie. They are most often stored in places: c: \ windows \ cookies (in Window2000 is in the C: \ Documents and Settings \ your user name \ Cookies).

Cookie is "keyword key = value value" to preserve the format of the record.

2. Targets the creation of a Cookie, Cookie object called the constructor can create a Cookie. Cookie object constructor has two string .

parameters: Cookie Cookie name and value.

```
Cookie c = new Cookie ( "username", "john");
```

3. If the JSP in the package good Cookie object to send to the client, the use of the response addCookie () method.

Format: response.addCookie (c)

4. Save to read the client's Cookie, the use of the object request getCookies () method will be implemented in all client came to an array of Cookie objects in the form of order, to meet the need to remove the Cookie object, it is necessary to compare an array cycle Each target keywords.

JSP 的技术发展历史

Java Server Pages (JSP) 是一种基于 web 的脚本编程技术，类似于网景公司的服务器端 Java 脚本语言——server-side JavaScript (SSJS) 和微软的 Active Server Pages (ASP)。与 SSJS 和 ASP 相比，JSP 具有更好的可扩展性，并且它不专属于任何一家厂商或某一特定的 Web 服务器。尽管 JSP 规范是由 Sun 公司制定的，但任何厂商都可以在自己的系统上实现 JSP。

在 Sun 正式发布 JSP (Java Server Pages) 之后，这种新的 Web 应用开发技术很快引起了人们的关注。JSP 为创建高度动态的 Web 应用提供了一个独特的开发环境。按照 Sun 的说法，JSP 能够适应市场上包括 Apache WebServer、IIS4.0 在内的 85% 的服务器产品。

本文将介绍 JSP 相关的知识，以及 JavaBean 的相关内容，当然都是比较粗略的介绍其中的基本内容，仅仅起到抛砖引玉的作用，如果读者需要更详细的信息，请参考相应的 JSP 的书籍。

1.1 概述

JSP (Java Server Pages) 是由 Sun Microsystems 公司倡导、许多公司参与一起建立的一种动态网页技术标准，其在动态网页的建设中有其强大而特别的功能。JSP 与 Microsoft 的 ASP 技术非常相似。两者都提供在 HTML 代码中混合某种程序代码、由语言引擎解释执行程序代码的能力。下面我们简单的对它进行介绍。

JSP 页面最终会转换成 servlet。因而，从根本上，JSP 页面能够执行的任何任务都可以用 servlet 来完成。然而，这种底层的等同性并不意味着 servlet 和 JSP 页面对于所有的情况都等同适用。问题不在于技术的能力，而是二者在便利性、生产率和可维护性上的不同。毕竟，在特定平台上能够用 Java 编程语言完成的事情，同样可以用汇编语言来完成，但是选择哪种语言依旧十分重要。

和单独使用 servlet 相比，JSP 提供下述好处：

- JSP 中 HTML 的编写与维护更为简单。JSP 中可以使用常规的 HTML：没有额外的反斜杠，没有额外的双引号，也没有暗含的 Java 语法。
- 能够使用标准的网站开发工具。即使是那些对 JSP 一无所知的 HTML 工具，我们也可以使用，因为它们会忽略 JSP 标签 (JSP tags)。
- 可以对开发团队进行划分。Java 程序员可以致力于动态代码。Web 开发人员可以将经理集中在表示层 (presentation layer) 上。对于大型的项目，这种划分极为重要。依据开发团队的大小，及项目的复杂程度，可以对静态 HTML 和动态内容进行弱分离 (weaker separation) 和强分离 (stronger separation)。

此处的讨论并不是说人们应该放弃使用 servlet 而仅仅使用 JSP。事实上, 几乎所有的项目都会同时用到这两种技术。在某些项目中, 更适宜选用 servlet, 而针对项目中的某些请求, 我们可能会在 MVC 构架下组合使用这两项技术。我们总是希望用适当的工具完成相对应的工作, 仅仅是 servlet 并不一定能够胜任所有工作。

1.2 JSP 的由来

Sun 公司的 JSP 技术, 使 Web 页面开发人员可以使用 HTML 或者 XML 标识来设计和格式化最终页面。使用 JSP 标识或者小脚本来生成页面上的动态内容(内容是根据请求来变化的)。

Java Servlet 是 JSP 技术的基础, 而且大型的 Web 应用程序的开发需要 Java Servlet 和 JSP 配合才能完成, Servlet 这个名称源于 Applet, 现在国内的翻译方式很多, 本书为了避免误会, 决定直接采用 Servlet 而不做任何翻译, 读者如果愿意, 可以称之为“小服务程序”。Servlet 其实和传统的 CGI、ISAPI、NSAPI 等 Web 程序开发工具的作用是相似的, 在使用 Java Servlet 以后, 用户不必再使用效率低下的 CGI 方式, 也不必使用只能在某个固定 Web 服务器平台运行的 API 方式来动态生成 Web 页面。许多 Web 服务器都支持 Servlet, 即使不直接支持 Servlet 的 Web 服务器也可以通过附加的应用服务器和模块来支持 Servlet。得益于 Java 的跨平台的特性, Servlet 也是平台无关的, 实际上, 只要符合 Java Servlet 规范, Servlet 是完全与平台无关且是与 Web 服务器无关的。由于 Java Servlet 内部是以线程方式提供服务, 不必对于每个请求都启动一个进程, 并且利用多线程机制可以同时为多个请求服务, 因此 Java Servlet 效率非常高。

但 Java Servlet 也不是没有缺点, 和传统的 CGI、ISAPI、NSAPI 方式相同, Java Servlet 是利用输出 HTML 语句来实现动态网页的, 如果用 Java Servlet 来开发整个网站, 动态部分和静态页面的整合过程会非常难以实现。为了解决 Java Servlet 的这种缺点, SUN 推出了 JSP。

许多年前, Marty 受到邀请, 参加一个有关软件技术的小型研讨会。坐在 Marty 旁边的人是 James Gosling——Java 编程语言的发明者。隔几个位置, 是来自华盛顿一家大型软件公司的高级经理。在讨论过程中, 研讨会的主席提出了 Jini 的议题, 这在当时是一项新的 Java 技术。主席向该经理询问他的想法。他回答说, 虽然现在言之过早, 但这看起来会是非常有前途的一项技术。他们会持续关注这项技术, 如果这项技术变得流行起来, 他们会遵循公司的“接受并扩充(embrace and extend)”的策略。此时, Gosling 随意地插话说“你的意思其实就是不接受且不扩充(disgrace and distend)。”

在此, Gosling 的抱怨显示出, 他感到这个公司会从其他公司那里拿走技术, 用于他们自己的目的. 出人意料的是, 形势已经完全不同。Java 团队并没有发明这一思想——将页面设计成由静态 HTML 和用特殊标签标记的动态代码混合组成.。ColdFusion 多年前就已经这样做了。甚至 ASP(来自于前述经理所在公司的一项产品)都在 JSP 出现之前推广了这种方式。实际上, JSP 不只采用了这种通用概念, 它甚至使用许多和 ASP 相同的特殊标签。

JSP 是建立在 Java servlets 模型之上的表达层技术, 它使编写 HTML 变得更简单。像 SSJS 一样, 它也允许你将静态 HTML 内容与服务器端脚本混合起来生成动态输出。JSP 把 Java 作为默认的脚本语言, 然而, 就像 ASP 可以使用其他语言(如 JavaScript 和 VBScript)一样, JSP 规范也允许使用其他语言。

1.3 JSP 的特点

按照脚本语言是服务于某一个子系统的语言这种论述, JSP 应当被看作是一种脚本语言。然而, 作为一种脚本语言, JSP 又显得过于强大了, 在 JSP 中几乎可以使用全部的 Java 类。

作为一种基于文本的、以显示为中心的开发技术, JSP 提供了 Java Servlet 的所有好处, 并且, 当与一个 JavaBeans 类结合在一起时, JSP 提供了一种使内容和显示逻辑分开的简单方式。分开内容和显示逻辑的好处是, 更新页面外观的人员不必懂得 Java 代码, 而更新 JavaBeans 类的人员也不必是设计网页的行家里手, 就可以用带 JavaBeans 类的 JSP 页面来定义 Web 模板, 以建立一个由具有相似的外观的页面组成的网站。JavaBeans 类完成数据提供, 这样在模板中就没有 Java 代码, 这意味着这些模板可以由一个 HTML 编写人员来维护。当然, 也可以利用 Java Servlet 来控制网站的逻辑, 通过 Java Servlet 调用 JSP 文件的方式来将网站的逻辑和内容分离。

一般来说, 在实际的 JSP 引擎中, JSP 页面在执行时是编译式, 而不是解释式的。解释式的动态网页开发工具如 ASP、PHP3 等由于速度等原因已经满足不了当前大型电子商务应用的需要了, 传统的开发技术都在向编译执行的方式改变, 如 ASP→ASP+; PHP3→PHP4。

在 JSP 规范书中, 并没有明确要求 JSP 中的程序代码部分(称为 Scriptlet)一定要用 Java 来写。实际上, 有一些 JSP 引擎就是采用的其他脚本语言, 如 EMAC-Script、WebL 等, 但实际上这几种脚本语言也是构建在 Java 上面, 编译为 Servlet 来实现的。按照 JSP 规范书写, 和 Java 没有任何关系的 Scriptlet 也是可以的, 不过, 由于 JSP 的强大功能主要在于能和 JavaBeans、Enterprise JavaBeans 共同运转, 所以即使是 Scriptlet 部分不使用 Java, 编译成的执行代码也应该是与 Java 相关的。

1.4 JSP 的机制

要理解 JSP 怎样联合以上各种所提到的技术的优点,从而轻而易举地实现各种效果,用户必须首先了解“组件为中心的网页开发”和“页面为中心的网页开发”的区别。

SSJS 和 ASP 都是在几年前推出的,那时网络还很年轻,没有人知道除了把所有的商务、数据和表达逻辑统统堆进原始网页中之外还有什么更好的解决方法。这种以页面为中心的模型容易学习并且得到相当快速的发展。然而,随着时间的推移,人们认识到这种方法不适于构建大型的、可升级的 Web 应用程序。在脚本环境中书写的表达逻辑被锁在页面内,只有通过剪切和粘贴才能被重用。表达逻辑通常和商务及数据逻辑混在一起,这使得当程序员试图改变一个应用程序的外观而不想破坏与之紧密结合的商务逻辑时,应用程序的维护就变得十分艰难。其事实上,企业中可重用组件的应用早已经很成熟,没有人愿意为它们的应用程序重写那些逻辑。HTML 和图形设计师把它们的设计的实施工作交给了 Web 编写者,使他们不得不加倍工作——常常是手工编写,因为没有合适的工具可以把服务器端脚本与 HTML 内容结合起来。简而言之,随着 Web 应用程序的复杂性不断提升,以页面为中心的开发方式的局限性变得明显起来。

与此同时,人们一直在寻找建立 Web 应用程序的更好方法,组件在客户机/服务器领域流行起来。JavaBeans 和 ActiveX 被“快速应用程序开发”(RAD)工具发行商推广给 Java 和 Windows 应用程序开发者用来快速开发复杂的程序。这些技术使某领域内的专家可以为该领域内的垂直应用编写组件,而开发者可以直接拿来使用而不必掌握这一领域的专门技术。

作为一种以组件为中心的开发平台,JSP 出现了。它以 JavaBeans 和 Enterprise JavaBeans (EJB) 组件包含商务和数据逻辑的模型为基础,提供大量标签和一个脚本平台用来在 HTML 页中显示由 JavaBeans 产生或回送的内容。由于 JSP 的以组件为中心的性质,它可以被 Java 和非 Java 开发者同样使用。非 Java 开发者可以通过 JSP 的标签(Tags)来使用高级 Java 开发者创建的 JavaBeans。Java 开发者不仅可以创建和使用 JavaBeans,还能在 JSP 页中使用 Java 语言来更精密地控制基于底层 JavaBeans 的表达逻辑。

现在来看看 JSP 是如何处理 HTTP 请求的。在基本请求模型中,一个请求直接被送到 JSP 页中。JSP 代码控制着进行逻辑处理时与 JavaBeans 组件的交互,并在动态生成的、混合了静态 HTML 代码的 HTML 页中显示结果。Beans 可以是 JavaBeans 或 EJB 组件。另外,更加复杂的请求模型可看作从被请求页呼叫其他 JSP 页或 Java Servlets。

JSP 引擎实际上要把 JSP 标签、JSP 页中的 Java 代码甚至连同静态 HTML 内

容都转换为大块的 Java 代码。这些代码块被 JSP 引擎组织到用户看不到的 Java Servlet 中去，然后 Servlet 自动把它们编译成 Java 字节码。这样，当网站的访问者请求一个 JSP 页时，在它不知道的情况下，一个已经生成的、预编译过的 Servlet 实际上将完成所有的工作，非常隐蔽而又高效。因为 Servlet 是编译过的，所以网页中的 JSP 代码不需要在每次请求该页时被解释一遍。JSP 引擎只需在 Servlet 代码最后被修改后编译一次，然后这个编译过的 Servlet 就可以被执行了。由于是 JSP 引擎自动生成并编译 Servlet，不用程序员动手编译代码，所以 JSP 能带给你高效的性能和快速开发所需的灵活性。

和传统的 CGI 相比较，JSP 有相当的优势。首先，在速度上，传统的 CGI 程序需要使用系统的标准输入输出设备来实现动态网页的生成，而 JSP 是直接和服务器相关联的。而且对于 CGI 来说，每一个访问就需要新增加一个进程来处理，进程不断地建立和销毁对于作为 Web 服务器的计算机将是不小的负担。其次，JSP 是专门为 Web 开发而设计的，其目的是为了建立基于 Web 的应用程序，其中包含了一整套的规范和工具。使用 JSP 技术可以很方便地将一大堆 JSP 页面组合成为一个 Web 应用程序。

JSP 六种内置对象

一、request 对象：

该对象封装了用户提交的信息，通过调用该对象相应的方法可以获取封装的信息，即使用该对象可以获取用户提交信息。

二、response 对象：

对客户请求做出动态的响应，向客户端发送数据。

三、session 对象

1. 什么是 session: session 对象是一个 JSP 内置对象，它在第一个 JSP 页面被装载时自动创建，完成会话期管理。

从一个客户打开浏览器并连接到服务器开始，到客户关闭浏览器离开这个服务器结束，被称为一个会话。当一个客户访问一个服务器时，可能会在这个服务器的几个页面之间反复连接，反复刷新一个页面，服务器应当通过某种办法知道这是同一个客户，这就需要 session 对象。

2. session 对象的 ID: 当一个客户首次访问服务器上的一个 JSP 页面时，JSP 引擎产生一个 session 对象，同时分配一个 String 类型的 ID 号，JSP 引擎同时将这个 ID 号发送到客户端，存放在 Cookie 中，这样 session 对象和客户之间就建立了一一对应的关系。当客户再访问连接该服务器的其他页面时，不再分配给客户新的 session 对象，直到客户关闭浏览器后，服务器端该客户的 session 对象才取消，并且和客户的会话对应关系消失。当客户重新打开浏览器再连接到该服务器时，服务器为该客户再创建一个新的 session 对象。

四、application 对象

1. 什么是 application:

服务器启动后就产生了这个 application 对象，当客户再所访问的网站的各个页面之间浏览时，这个 application 对象都是同一个，直到服务器关闭。但是与 session 不同的是，所有客户的 application 对象都是同一个，即所有客户共享这个内置的 application 对象。

2. application 对象常用方法:

(1)public void setAttribute(String key,Object obj): 将参数 Object 指定的对象 obj 添加到 application 对象中，并为添加的对象指定一个索引关键字。

(2)public Object getAttribute(String key): 获取 application 对象中含有关键字的对象。

五、out 对象

out 对象是一个输出流，用来向客户端输出数据。out 对象用于各种数据的输出。

六、Cookie

1. 什么是 Cookie:

Cookie 是 Web 服务器保存在用户硬盘上的一段文本。Cookie 允许一个 Web 站点在用户的电脑上保存信息并且随后再取回它。

举例来说，一个 Web 站点可能会为每一个访问者产生一个唯一的 ID，然后以 Cookie 文件的形式保存在每个用户的机器上。

如果您使用 IE 浏览器访问 Web，您会看到所有保存在您的硬盘上的 Cookie。它们最常存放的地方是：c:\windows\cookies(在 Window2000 中则是 C:\Documents and Settings\您的用户名\Cookies)。Cookie 是以“关键字 key=值 value”的格式来保存纪录的。

2. 创建一个 Cookie 对象，调用 Cookie 对象的构造函数可以创建 Cookie。Cookie 对象的构造函数有两个字符串参数：Cookie 名字和 Cookie 值。Cookie c=new

`Cookie(“username”, “john”)`。

3. JSP 中如果要封装好的 Cookie 对象传送到客户端，使用 `response` 的 `addCookie()` 方法。格式：`response.addCookie(c)`。

4. 读取保存到客户端的 Cookie，使用 `request` 对象的 `getCookies()` 方法，执行时将所有客户端传来的 Cookie 对象以数组的形式排列，如果要取出符合需要的 Cookie 对象，就需要循环比较数组内每个对象的关键字。