



基于 GPRS 和 Web 模式的油井监控系统设计

李纪扣, 段太雷

(天津科技大学计算机科学与信息工程学院, 天津 300222)

摘要: 采用 GPRS 技术和 Struts 2 的 Web 架构,提出了一种实现油田远程监控系统的方法. 在介绍了基于嵌入式微处理器 S3C2410X 和 Arm-Linux 操作系统组成的数据采集系统、GPRS 数据传输的软硬件设计方案的基础上,给出了基于 GPRS 的远程服务器系统的软件体系结构.

关键词: 嵌入式系统; GPRS; Struts 2; 油井监控

中图分类号: TP302.1 **文献标志码:** A **文章编号:** 1672-6510(2010)03-0071-04

Design of Oil Well Monitor System Based on GPRS and Web Mode

LI Ji-kou, DUAN Tai-lei

(College of Computer Science and Information Engineering, Tianjin University of Science & Technology,
Tianjin 300222, China)

Abstract: According to the GPRS technology and Web structure of Struts 2, one method for realizing remote monitoring the oil field system was introduced. On the base of detailed introduction of data collecting system based on embedded microprocessor unit S3C2410X and Arm-Linux operating system, and software and hardware designing program of GPRS data transmission, the software system structure based on GPRS remote server system was proposed.

Keywords: embedded system; GPRS; Struts 2; oil well monitor

油田油井大多都分布在各采油场,且各油场也比较分散,环境较差^[1]. 传统的人工巡检方式对油井抽油机的检测存在劳动量大、监测不及时、数据汇总困难等多种问题. 随着无线通信技术和信息技术的发展,油田逐渐采用无线监控系统来实施参数和工作状态的监控,实现对油井抽油机工作平台的实时控制.

国外油田在 20 世纪 80 年代就实现了分布式监测,目前更是建立了从原油开采到存储、加工、销售全面监控的自动化系统,将自动化系统上升到了现代管理的高度^[2]. 尽管国外油田监控技术已经达到了很高的水平,监控系统功能也日趋完善,但是,许多功能并不符合我国目前的石油生产情况,且价格昂贵. 因此,开发符合国内油田实际需求的监控系统具有现实意义.

国内油田的自动化监控起步较晚,仍有采用手工制作报表来传递数据的情况,尚未把计算机网络上的信息发布、资源共享等优势体现出来^[3]. 另外,由于目

前大多数 GPRS 通信模块没有封装 TCP/IP 协议栈,不具备动态域名寻址能力,因此对于基于 GPRS 的下位机与上位机通信时要求上位机具有固定 IP 地址,增加了费用.

本文构建的油田监控系统,通过动态域名访问技术解决传统方式服务器端需要固定 IP 地址问题,实现对服务器的动态域名访问,降低通信服务费用;在服务器端以 Web 方式作为监测数据的显示和管理手段,解决系统部署困难,升级维护复杂等问题;由于采用 Web 开发框架和根据业务逻辑分层设计理念,且采用开源软件平台设计开发,提高了系统扩展能力,减少了系统后期维护费用.

1 系统的工作原理

系统由无线传感器组、终端子系统、上位机服务器系统 3 部分组成,如图 1 所示.

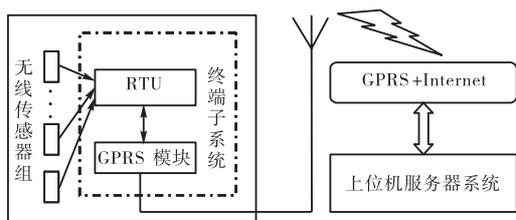


图1 系统框图
Fig.1 System architecture

首先由最前端的无线传感器测得抽油机及油井的各项参数,采用 ZigBee 无线传输技术传送到终端子系统,终端子系统的 RTU(Remote Terminal Unit)通过对数据进行分类、汇总、打包,再由 GPRS 模块将数据传输到 Internet 上. 位于 Internet 另一端的上位机服务器系统接收到数据之后,对数据进行解包,存入数据库. 上位机服务器系统中的 Web 服务器通过对数据库的访问,实现对工作平台的实时监控.

2 系统软硬件实现

2.1 终端子系统的软硬件实现

2.1.1 终端子系统的硬件实现

RTU 采用基于 ARM920T 结构的 S3C2410X 高速处理器. 该处理器集成了独立的 16 kB 指令存储器和 16 kB 数据存储器和内存管理的 MMU(Memory Management Unit)单元、LCD 控制器、3 通道的异步串口(UART)、4 通道的 DMA 等. 其 RTU 的硬件框图如图 2 所示.

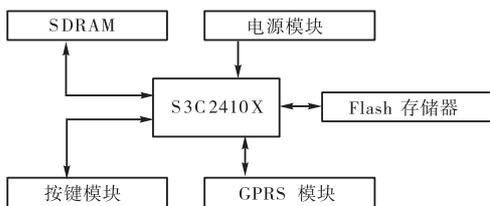


图2 硬件框图
Fig.2 Hardware architecture

GPRS模块采用 WAVECOM 公司的 Q2403A. 该模块整合了语音、数据传输等业务,支持 RS-232 接口. 可以直接使用 S3C2410X 的 UART1 接口驱动该模块.

2.1.2 终端子系统的软件实现

(1) 嵌入式操作系统的移植

Q2403A 没有封装 TCP/IP 协议栈,故需要操作系统的协议支持. 采用 ARM-Linux 操作系统为核心,进行最小系统设计和移植,在配置内核的过程中选择点对点协议(PPP, Point to Point Protocol)的支持^[4],以实现 IP 通信.

(2) GPRS 通信的建立

GPRS 网络与 Internet 网络的通信通过 ISP 的网关支持节点(GGSN, Gateway GPRS Support Node)实现. GPRS 模块采用 PPP 协议登录到 GGSN 上,获得 GGSN 动态分配的 IP 地址.

PPP 的通信在 Linux 系统中分为两个处理过程:一部分由系统内核来完成,主要完成 PPP 消息的封装和拆包(这要求在配置操作系统内核的时候要选对 PPP 设置的支持);另一部分为 PPP 守护进程 pppd(ppp daemon),PPP 守护进程提供基本的 LCP,完成 PPP 链路的建立、断开,账户认证和 IP 地址的设置等工作.

GPRS 拨号的整个过程如图 3 所示.

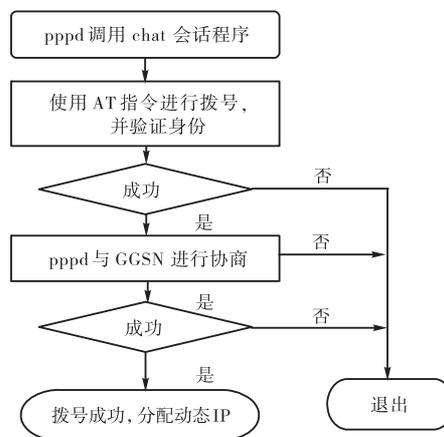


图3 拨号过程
Fig.3 Dial-up process

为了使 GPRS 模块支持动态域名访问,需在 pppd 程序的启动脚本中使用 usepeerdns 参数,以实现自动从服务器获得域名解析服务器的地址,或在 resolv.conf 文件中直接输入域名解析服务器地址,这是能够访问具有动态域名的上位机服务器系统的关键.

2.2 上位机服务器系统的实现

2.2.1 上位机服务器系统的软件体系架构

上位机服务器系统主要是用来接收 GPRS 模块发送的油井监测数据,并对数据进行分析、分类,存入数据库. 同时,对所检测的非法数据进行报警提示. 其结构包括以下几个部分:上位机接收模块、数据库服务器、Web 的实时信息管理模块,如图 4 所示.

系统采用的 Web 服务器软件为 Tomcat,它支持最新的 JSP 和 Servlet 规范,且为开放源代码免费的软件,可以进一步降低软件开发成本. 上位机服务器系统采用 Fedora 8,具有较好的稳定性,可以有效地支持 MySQL、Tomcat 等服务器软件,且具有开源特性.

(1) 实时信息管理模块的工作过程

一个来自客户端的请求到达 Servlet 容器(使用 Tomcat 作为 Servlet 容器)后,Servlet 容器通过读取在 Web.xml 文件中配置的 FilterDispatcher 过滤器,将所有请求都交给 Struts 2 框架处理. Struts 2 通过其实现的 IoC(Inversion of Control)模式,采用依赖注入(Dependency Injection)方式产生一个 ActionMapper 的实例. FilterDispatcher 查询 ActionMapper 来确定这个请求是否存在一个相应的 Action 来处理. 如果存在一个 Action,将返回一个封装 Action 类和执行方法的 ActionMapping 对象. 之后,FilterDispatcher 将控制权委派给 ActionProxy,ActionProxy 在 Struts 2 框架和 Action 之间充当代理的角色.

ActionProxy 得到控制权之后查询配置文件 struts.xml,之后使用其 ActionInvocation 对象进行拦截器的执行和 Action 的执行. 当 ActionInvocation 对象执行完毕后向 ActionProxy 返回一个 String 类型的结果码,最后 ActionProxy 清理状态,并由框架将结果返回给客户端.

(2) 数据持久层的设计

根据面向对象的程序设计方法,软件中对于数据的处理,实际上是转换为对对象的处理;对于数据的存储,实际上是对象持久化的过程. 对象的持久化实际上是把暂时不使用对象的对象状态和属性存入数据库中,再次使用时就可以从数据库中恢复对象. 但基于关系型的数据库存放的是二维表格数据,和对象的属性和状态信息并不匹配,所以将对象存入数据库中必须进行对象-关系的转换才可以,即借助于 ORM(Object/Relational Mapping)中间件来实现这种转换. 因此,系统对于数据的存储和检索操作并不直接访问数据库,而是抽象出一个数据持久层来处理数据存储和检索工作,ORM 工具就担任持久层的角色. 在本系统中使用的 ORM 工具为 Hibernate,它提供了强大、高性能的对象到关系数据库的持久化服务,并已成为 ORM 领域事实上的标准^[5-6].

系统中数据持久层使用 DAO(Data Access Object, 数据访问对象)来封装和抽象所有对数据源的访问,它内部封装了 Hibernate 数据操纵、事务处理、会话管理等 API. 通过 DAO 业务组件为业务逻辑层提供了更简单的接口,DAO 完全向业务逻辑层隐藏了数据源实现细节. 当底层数据源实现变化时,DAO 向客户端提供的接口不会变化.

业务逻辑模块调用 DAO 工厂类生成相应的 DAO,DAO 对象封装了 Hibernate,由 Hibernate 来完

成与数据库的交互. 在 DAO 里通过 HQL(Hibernate Query Lanaguage)语句操作 PO(Persistence Object); Hibernate 把 HQL 转换为 SQL,通过查询映射文件来将对 PO 对象的操作转变为对数据表的操作.

系统运行界面如图 7 所示.

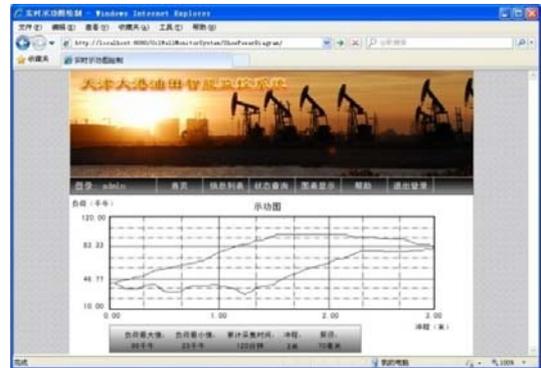


图 7 系统运行界面
Fig.7 System operation interface

3 实验

主要是对监测数据的获取、非法数据的报警、系统稳定性等方面进行测试. 其中,在无线监控系统中数据通信延时是一个不可忽略的方面. 在本系统中数据通信延时主要在于两个方面:一是 ZigBee 传感器传输延时;二是 GPRS DTU(Data Transformation Unit)通信延时. 其中,ZigBee 通信延时 < 100 ms, GPRS DTU 通信延时在 2 s 左右,而抽油机一个冲程的周期为 5 ~ 10 s,通过对 GPRS 和 ZigBee 的通信速率的计算,本系统采用的无线通信方案完全满足监测数据传输要求. 即使出现暂时通信故障,RTU 中的缓冲区足够保存一个月的监测数据.

4 结 语

本文介绍了以 GPRS 无线传输方式和基于 Java EE 的多层 Web 架构建立的油田监控系统,包括嵌入式子系统的硬件结构、ARM-Linux 系统的移植、GPRS 模块的配置、上位机服务器系统各模块的实现等方面. 系统采用的开发平台及所有开发组件均为开源软件,并且使用面向对象的分层设计思想,各个部分耦合程度低. 因而系统具有成本低、安全性好、效率高且便于维护等特点. 经过系统试运行,各项监测指标均符合油田实际工作要求.

(下转第 78 页)