

基于 J2EE 的实时监控系统的设计与实现

鲍胜利, 钟 勇

(中国科学院成都计算机应用研究所, 成都 610041)

摘要: 结合某卷烟厂卷接包车间信息化建设的现状和需求, 引入 J2EE 架构体系和技术, 设计并实现一个基于 Web 方式的全面实时监控系统的。提出一种在 J2EE 架构平台上应用 MVC 模式的 Struts 框架开发大型厂级实时监控系统的思路和方法, 介绍系统结构和功能, 对系统后台数据库设计和系统实现中相关的关键技术进行了论述。

关键词: Struts 框架; Socket 通信; 扩展存储过程; 触摸屏

Design and Implementation of Real-time Monitoring System Based on J2EE Architecture

BAO Sheng-li, ZHONG Yong

(Chengdu Computer Application Institute, Chinese Academy of Sciences, Chengdu 610041)

【Abstract】 Combining with the status and requirements for the information construction in the workshops of some cigarette factory, this paper introduces the J2EE architecture and technology to design and implement a real-time monitoring system based on Web pattern. Based on the platform of J2EE architecture, the paper creates a new method to develop large-scale real-time monitoring system using the free Struts framework to fulfil the MVC design mode. In addition, the system structure and functions are described, especially some relative key technologies and database design are discussed detailedly during the system implementation.

【Key words】 Struts framework; Socket communication; extended procedure; touch screen

1 概述

随着市场经济的不断发展与完善, 企业的管理工作也由原来的粗放型管理改进为精细化管理。特别在烟草行业, 随着烟草企业集团化的重组不断加快, 集团企业越来越关心每个生产现场的投入与产出、节能与降耗、增产与增收。烟草行业卷包车间实时监控系统的就是在这种背景下引入的, 该系统旨在实现对卷烟厂卷包车间中生产设备各种实时信息的采集、整理、存储、监控和分析, 同时满足对设备检查、设备保养、设备停机、工艺质量控制、车间现场控制等的全面管理, 为企业生产决策提供准确、及时、全面的数据分析和管理平台。以往的实时监控系统一般均存在 2 个问题: (1) 无法实现卷包车间的全面管理, 仅对设备的各种实时数据进行了采集和管理, 不能对生产过程中实时发生的各种状况, 如工单情况、设备状况、质量情况和人员情况等进行管理; (2) 较多采用客户端/服务器(C/S)模式, 系统的扩展性和维护性差, 特别是不能满足车间现场较多采用触摸屏进行操作的现状。针对这些情况, 本文设计了一种基于 J2EE 架构的卷包车间实时监控系统的。

2 监控系统设计

整个系统采用基于 J2EE 架构下的 B/S 模式, 同过去较多采用的 C/S 模式的监控系统相比, B/S 模式的系统界面友好、统一, 方便用户使用; 软件的安装、维护和升级容易; 客户端仅需拥有一个浏览器即可使用系统, 系统的扩展性好。

2.1 系统结构设计

整个系统的结构如图 1 所示。系统包括 3 大部分: 数据采集模块, 数据管理系统和实时监控系统的。

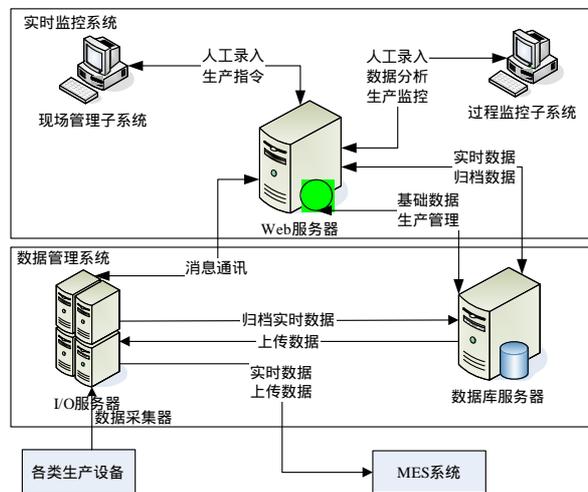


图 1 系统整体结构

数据采集模块负责采集卷包生产车间中各类设备的实时数据, 并将采集到的数据传给 I/O 服务器, 以进一步处理和存储。该模块的实现是采用基于 OPC(OLE for Process Control) 技术和 Socket 通信技术, 并将功能集成在称为采集器的硬件产品中完成的。数据管理系统由 I/O 服务器和数据库服务器组成, 主要实现对数据采集模块传过来的实时数据按既定要求进行分类、处理, 并负责将数据存储到数据库服务器。

作者简介: 鲍胜利(1973 -), 男, 博士研究生, 主研方向: 计算机软件应用和软件工程; 钟 勇, 研究员、博士生导师

收稿日期: 2007-04-06 **E-mail:** baohigh@casit.com.cn

以供实时监控系统进行数据调用。实时监控系统包括现场管理子系统和过程监控子系统,通过该系统,卷包车间甚至整个烟厂的各级管理人员均能够通过浏览器获取卷包车间的运行信息,并可实时看到设备的运行情况,以实现车间生产的科学调度和管理。

2.2 系统功能设计

整个实时监控系统包括现场管理子系统和过程监控子系统,其中现场管理子系统的根本就是实时记录生产中出现的各种情况,主要包括以下功能模块:

(1)工单管理:提供对各类生产工单启动、停止、暂停以及重启等事件的管理;

(2)设备管理:实现对设备保养、设备维修以及设备停机情况的管理;

(3)工艺管理:对车间各类生产工序的工艺质量进行跟踪和反馈;

(4)消息管理:包括文档通知的下发,生产指挥流程消息的流转;

(5)烟盘管理:实现对设备生产能力不平衡时烟支流向进行管理。

过程监控子系统是对车间全方面的监控和管理,整个实时监控系统的核心,主要包括以下功能模块:

(1)系统管理:对系统各类基础数据进行管理,如人员信息、设备信息等;

(2)生产控制:实现对车间现场的各种调度和控制,如工单下达、生产排班等;

(3)实时报告:对整个车间的生产、消耗和设备数据进行实时监控和查询;

(4)历史报告:对整个车间的生产、消耗和设备历史数据进行查询和对比;

(5)数据分析:提供对车间生产、消耗以及设备使用情况进行多方面的深入分析。

3 监控系统的实现

3.1 数据库系统设计

整个数据管理系统包括两大部分: I/O 服务器和数据库服务器。在此增设 I/O 服务器,是考虑到系统功能模块化的需要,把对数据采集模块采集到的实时数据的分类、处理、存储及同 MES 系统的所有数据交互等功能以独立的 I/O 服务器形式存在,也便于以后对系统的维护和扩展。I/O 服务器本身不存在存储数据机制和设备,整个系统的数据均存储在数据库服务器。数据库系统采用 SQL Server2000 数据库。

系统中涉及到的数据有两类用途不同的数据:实时数据和历史数据。两类数据的管理思路是采用同一个 SQL Server 数据库,但采用不同的数据表。一类是实时数据表,这类数据表的数据由 I/O 服务器程序根据系统要求来负责写入,其数据源是数据采集模块传过来的实时数据;另一类是历史数据表,这类数据表存放的是各类实时数据进行归档后的数据,这部分数据由实时监控系统在换班时负责归档并写入。鉴于实时数据的使用特点,以及提高实时数据的存取效率,在系统启用后需要定期如每隔 3 天就清除一次实时数据表中的数据。

3.2 STRUTS 开发框架的使用

整个实时监控系统是基于 J2EE 平台设计,从而实现了跨平台,易维护、可复用。在软件开发中使用了基于 MVC 设计模式的开源 STRUTS 框架,该框架把 Java 代码从 JSP 页

面中分离出来,从而实现了 Web 应用程序的表示层与事务逻辑层、数据层的分离,极大提高了 Web 系统的开发进度和质量。整个 Struts 应用包括 4 个组件,如图 2 所示。

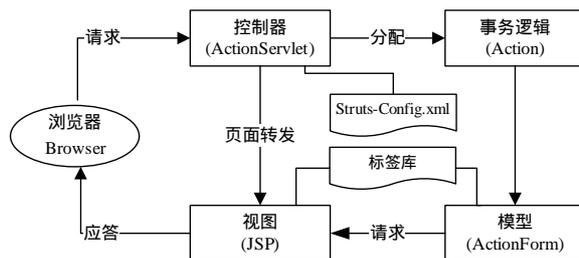


图 2 软件系统架构

其中, ActionServlet 和 Action 2 部分组成 Struts 的控制器 (Controller) 部分, JSP 页面为视图 (View) 部分, 而模型 (Model) 部分指的是同 JSP 页面对应的 ActionForm Bean。该框架的实质就是把传统 JSP 页面中的 Java 代码用 Java Bean 及 Action 类分离出来, 形成 MVC 模式, 并使用框架定义的标签库, 进行模型 Model、视图 View 和控制器 Controller 之间数据的传递, 最后用配置文件 Struts-Config.xml 显示各个类与相关 JSP 页面的联系, 从而实现了 J2EE 应用程序的表示层同事务及数据层的分离。在 Struts 框架下开发此系统的一般步骤如下:

- (1) 编制功能点所需的 JSP 页面;
- (2) 编写同该 JSP 页面对应的 ActionForm 类文件;
- (3) 编写该功能数据和事务逻辑处理的 Action 类文件;
- (4) 配置控制文件 Struts-Config.xml 以实现上述 3 个文件的数据传递。

3.3 生产换班功能

卷烟厂的卷包车间在实际生产过程中同时存在自动换班和手工换班的情况。自动换班指系统按照生产排班表的计划开始时间和计划结束时间自动划定某个机台在某个班次的开始或结束; 而手工换班则需要机台操作工在现场管理子系统的前台程序中进行确认开始和结束当前班次。换班功能对后续按班次不同进行的实时数据归档操作至关重要, 也决定了过程监控子系统中数据查询和分析功能中源数据的准确性。

由于手工换班需要人工干预, 因此给系统的运行带来了不小的隐患。因为如某一机台没有进行手工换班操作, 则将会导致该机台无法启动下一班, 并且会导致该机台班次数据无法进行归档。为此, 通过与车间管理人员进行深入交流后, 系统决定取消手工换班功能, 全部采用自动换班方式, 而实际存在的手工换班功能通过调整其对应的生产排班数据来实现。系统采用以下思路来实现生产的自动换班功能:

- (1) 在 SQL Server 数据库服务器中建立一个调度任务 pl_dg_autoshift, 设置该调度周期性地运行一个换班存储过程 proc_ws_autoshift(), 调度任务的运行频率为 1 min^[1];
- (2) 在数据库中创建存储过程 proc_ws_autoshift(), 用该存储过程来处理自动换班中的所有功能逻辑, 如生成机台换班记录、班次数据归档等。

3.4 Web 服务器同 I/O 服务器的通信

Web 服务器同 I/O 服务器需要进行数据通信是基于以下 2 点考虑的: (1) 应上端 MES 系统的要求, 实时监控系统将一些特定的数据主动发送给 MES 系统, 这些数据主要指现场操作工通过现场管理子系统操作的一些信息, 如工单启动、工单结束和换班操作等信息; (2) 实时监控系统需要对采集到

的实时数据进行分类、细化管理，因为整个系统的实时数据包括 6 大类信息：无效数据，有效数据，小试数据，工单转换数据，非故障停机数据和现场维修数据。根据 Web 服务器传过来的消息类型和内容，I/O 服务器需要将采集到的实时数据进行分类处理后存储到数据库服务器。

两类服务器间通信结构如图 3 所示。具体而言，设计通信思路如下：

(1)当现场操作工在现场管理子系统中执行某一任务如工单启动时，现场浏览器按确定的通信协议提取相关的工单启动信息，如工单启动时间、工单号等，并将这些信息提交到 Web 服务器；

(2)Web 服务器根据要求进行相关的功能逻辑处理，并将处理后的数据提交至数据库服务器进行存储；

(3)Web 服务器将现场浏览器提交的通信数据发送给 I/O 服务器。I/O 服务器负责接收 Web 服务器发送的数据，如果数据正确接收，I/O 服务器将向 Web 服务器反馈确认信息，以保证数据正确。

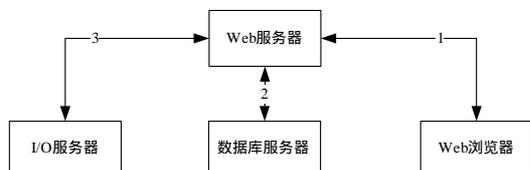


图 3 服务器通信结构

Web 服务器和 I/O 服务器之间的通信设计采用 Socket 通信机制，它采用客户/服务器模式，服务器端程序部署在 I/O 服务器上，而客户端程序驻留在 Web 服务器上。由服务器程序先建立 Socket 并将 Socket 绑定到指定端口上，并进入监听状态，监听是否有与自己端口相对应的连接请求。Socket 连接是客户方发起的，调用客户端程序向服务器发起连接，服务器在检测到连接后，建立连接并开始进行数据的接收和发送。在整个通信功能设计中，比较关键的地方在于如何实现 Web 服务器端的 Socket 连接功能，因为系统在以下 2 种场合需要进行 Socket 通信：

(1)现场管理子系统某些功能页面中需调用通信程序，如工单启动、维修开始等，此时在页面上存在相应的按钮来进行调用。

(2)根据通信协议，系统在换班时也需要同 I/O 服务器进行通信，但由于换班是由数据库系统的存储过程来自动完成的，因此需要考虑如何在存储过程中实现 Socket 通信。

基于上述考虑，系统采用了以下 2 种方式来实现 Web 服务器端的通信功能：

(1)用 Java 开发一个 Socket 客户端通信功能函数 ConnectServer()，以在需要的功能按钮中根据通信协议添加该通信函数的调用即可。这种实现方式适用于第 1 种场合。

(2)系统采用扩展存储过程机制来实现在换班存储过程中的 Socket 通信功能。SQL Server 的扩展存储过程，其实质就是 SQL Server 可以动态装载并执行的动态链接库 (DLL)，这就使得扩展存储过程的功能几乎不受任何限制^[2]。系统先

用 VC++ 的 DLL 模板库来编写 Socket 客户端通信功能程序，并将该程序打包成 DLL 文件，然后将该 DLL 文件拷贝至数据库服务器特定的目录下以进行调用；之后在数据库中添加一个普通的存储过程 sp_executesocket()，该存储过程用来实现对扩展存储过程的注册，并对 DLL 文件中的功能进行调用；最后在换班存储过程中传入要求的参数执行存储过程 sp_executesocket() 即可。

3.5 数据查询和分析

过程监控系统大部分的功能均是基于实时、历史数据的查询和分析，因此该部分工作是实时监控系统的核心所在。在系统设计和实现中，采用了以下思路和技术：

(1)为了提高数据查询效率，便于系统移植和后期维护，大部分数据查询和分析功能均采用数据库存储过程来实现。

(2)系统中几乎所有的数据查询均需要图形表示功能^[3]，如柱状图或饼图显示等。为此，系统在基于引入开源的 JFreeChart 组件的基础上，开发了一个较通用的图形表示功能页面，系统在需要用图形表示时统一调用该页面即可。在该功能页面中用户可以自定义 X 轴和 Y 轴所对应的数据列，此外还可自定义图形显示类型，如折线图、饼图等。

3.6 触摸屏操作

该系统的另一个特点是现场管理子系统必须满足能在现场操作终端的触摸屏上使用，并保证操作人员能及时、快速记录现场生产中的各种事件，故在页面导航、页面布局特别是操作按钮位置上，均遵循“灵活、方便、简捷”的原则进行设计。

(1)系统开发了一个软键盘，以方便用户在触摸屏上能录入一些基本的信息；

(2)现场管理子系统中涉及的所有功能均采用操作按钮来实现，对于需要用户录入信息的地方，一般均设计使用单选或复选按钮，或自动弹出软键盘等方式实现；

(3)在页面中根据操作逻辑和具体情况合理组合并放置按钮“返回前页”和“返回首页”，以方便用户切换功能操作。

4 结束语

当前，基于 Web 的实时监控已经成为测控软件研究中的一个主要研究方向。本文采用 J2EE 技术，并使用基于 MVC 设计模式的开源 STRUTS 框架，开发了一个较全面的实时监控系统，从而为工厂级实时监控提供了一种新的实现途径。该系统目前已在某卷烟厂得到了较好的应用，系统运行稳定，效果良好，满足了用户对整个车间进行数字化管理的需要。

参考文献

- [1] 隆华软件工作室. SQL Server 2000 程序设计[M]. 北京: 清华大学出版社, 2001.
- [2] 万波, 周顺平. SQL Server 扩展存储过程实现机制及应用方法初探[J]. 武汉科技大学学报: 自然科学版, 2001, 24(3): 294-297.
- [3] 贾永振, 刘载文, 段长明, 等. 基于 Web 的远程实时监控系统的实现技术[J]. 微计算机信息, 2006, 22(8): 89-91.