

DOI:10.3969/j.issn.1001-4551.2017.12.021

# 远程监控系统通用化5层架构设计\*

周奇才, 邓烈鑫, 赵 炯, 熊肖磊, 周 影  
(同济大学 机械与能源工程学院, 上海 201804)

**摘要:**针对现有远程监控系统通用性不足、兼容性和扩展性较弱的问题,提出了一种远程监控系统通用化5层架构。该5层架构由表示层、业务逻辑层、持久化层、资源抽象层、设备适配层构成。利用对可配置的页面构件进行注册和数据绑定,实现了页面的定制化。在业务逻辑层中建立了构件容器并对相应构件进行分类以减少系统的臃肿与耦合。通过对底层设备的实体、行为和状态建立了“抽象和元抽象树”,制定了抽象规范,并为其通信接口编写了适配程序,屏蔽了底层设备的差异性,保证了系统的通用性和扩展性。研究表明:该架构可适应差异性较大的设备和复杂的网络通信环境,底层设备可自由加入或退出系统,避免了大量的页面逻辑开发,降低了开发成本与周期。

**关键词:**远程监控;通用化架构;适配器;抽象编程

**中图分类号:**TP277

**文献标志码:**A

**文章编号:**1001-4551(2017)12-1469-05

## Design of universal five-layer architecture of remote monitoring system

ZHOU Qi-cai, DENG Lie-xin, ZHAO Jiong, XIONG Xiao-lei, ZHOU Ying  
(School of Mechanical Engineering, Tongji University, Shanghai 201804, China)

**Abstract:** Aiming at the poor universality, compatibility and extensibility of the existing remote monitoring system, a five-layer architecture of remote monitoring system was proposed. The five-layer architecture consists of presentation layer, business and logic layer, persistent layer, resource abstract layer and device adapter. Using the configurable page component for registration and data binding, the customization of the page was realized. The complexity and coupling of the system could be reduced by establishing components container in the business and logic layer and classifying the corresponding components. Through the establishment of the abstraction and meta-abstract tree of the entity, behavior and state of the underlying equipment, the abstract specification was formulated and the adaptation program was written for its communication interface, which shielded the difference of the underlying equipment and ensured the versatility and expansibility of the system. The results indicate that the large difference among the equipment and the complex network communication environment are adapted, the underlying equipment can be free to join or exit the system, a lot of page logic development is avoided, the development costs and cycle are reduced.

**Key words:** remote monitor; generalized architecture; adapter; abstract programming

## 0 引 言

随着云计算、大数据和物联网技术的发展,远程监控系统的应用领域、规模、数据量以及客户需求都发生

了巨大的变化。远程监控就是利用计算机通信网络技术,将设备数据采集给用户,并进行处理分析,同时将用户的决策动作,传递给检测设备的自动化过程<sup>[1]</sup>。

传统远程监控系统多采用由设备层、设备服务器

收稿日期:2017-05-31

基金项目:国家自然科学基金资助项目(51375345)

作者简介:周奇才(1962-),男,江苏宜兴人,教授,博士生导师,主要从事现代物流装备技术、远程监控系统方面的研究。E-mail: qczhou@tongji.edu.cn

和客户端构成的 3 层架构<sup>[2]</sup>。服务器监听相应端口的输入,客户机发送请求,服务器接收并处理请求,并将结果返回给客户机。传统 C/S 结构存在跨平台无法运行的难题<sup>[3]</sup>。文献[4]提出了基于 B/S 架构的监控系统,优化了其兼容性问题,但其系统监测与设备管理出现了功能分离;文献[5]提出了一种基于 C/S 和 B/S 架构的混合 3 层架构,一定程度上解决了数据传输速度和稳定性的均衡问题,但开发难度大,周期长;文献[6]提出了增加查询服务层的分布式 4 层架构,使监控规模得以扩大;文献[7]建立了松散耦合的 Web 服务系统但效率不高;文献[8]提出了一种基于 Ajax 的 4 层结构提高了界面开发效率但可扩展性较弱;文献[9]开发了面向移动端的监控系统,扩展了其表现形式。

目前远程监控系统设计缺乏通用性系统架构的支持。各系统存在结构不一、应用领域差异明显、通用性不强、开发工具不同、系统性能各有侧重但无法兼顾的问题。

随着远程监控系统应用领域的扩大,庞大的监控规模、海量的多源异构数据、超负荷的数据通信带来的传输速率问题、系统功能需求多样化等,都迫使远程监控系统诞生通用性的新型架构。

本研究提出 5 层架构的总体设计,利用抽象与适配的思想,对每层进行具体的功能设计,并给出监控设备的抽象与适配示例。

1 通用化架构设计的目标

一个可以在极低工时与成本之下,只做出少量修改便可运行在其他系统之上的通用化架构,设计目标如图 1 所示。

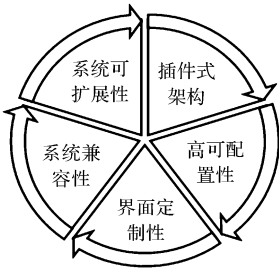


图 1 通用化架构设计目标

(1)插件式架构。系统在业务层面有粒度较细但合理的功能划分,同时各模块间耦合性弱;

(2)高可配置性。对程序中采用的算法、参数等进行配置以完成模块功能的调整;

(3)客户端界面的高可定制性。具备插件化的页

面模块,保证前端页面可灵活组装;

(4)良好的兼容性。系统可兼容不同的设备、网络接口以及遗留的模块与数据;

(5)良好的拓展性。当监控设备数目、参数增加时,可便捷的加入;当设备的性能要求增加时,也具有可伸缩性。

通用化架构设计的关键在于实现系统的模块化与插件化,并保证底层、抽象层的接口的规范与通用。这要求对系统各个模块进行合理的划分,并通过抽象提取出监控系统中的资源、行为与状态等一系列特征,形成具有良好通用性的框架。

2 通用化 5 层架构的总体设计

本研究通过综合分析设计模式<sup>[10]</sup>,提出通用化监控系统的 5 层架构,即表示层、业务逻辑层、持久化层、资源抽象层与设备适配层,如图 2 所示。

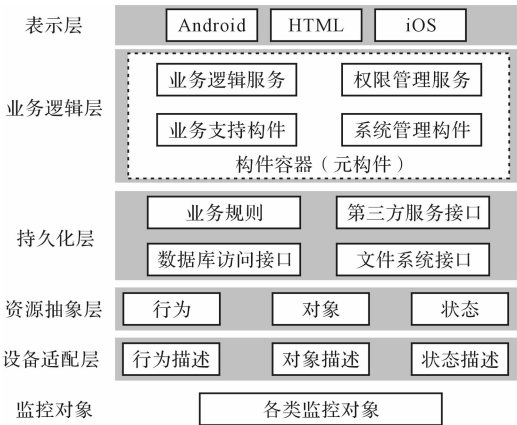


图 2 通用化监控程序 5 层结构

其中,表示层用于显示和接收数据,提供人机交互界面;业务逻辑层负责所有监控逻辑的处理,并通过多线程、缓冲、数据库连接池等技术优化系统的并发性能;持久化层负责底层资源的调用与协调;资源抽象层对底层的监控设备与传感器进行抽象处理,并面向上层提供调用规范;设备适配层则对底层监控对象的模型、行为与状态等进行封装,使之与资源抽象层的规范相符。

2.1 表示层

表示层通过开发大量可配置的页面构件,并与后台数据进行绑定,从而实现可定制化。使用时只需要在页面视图中注册相应的构件,避免大量的页面逻辑开发。

表示层的构件粒度大小设定为一个“功能”的集合。该构件提供一组用户自定义参数,配置在配置文

件中,即形成一个简单的基于配置的构件。整个界面通过若干个构件组合而成。程序在输出页面的同时,会根据配置文件生成该页面的数据请求方案,当页面处在客户端前端时,会按照该方案进行相应的数据请求。表示层定制完成后,必须配置好与之对应的业务逻辑层中的服务接口。

2.2 业务逻辑层

业务逻辑层是整个系统的业务核心,大量的业务逻辑容易造成系统的臃肿与过度耦合。整个业务逻辑层,不仅需要对组成构件按照功能与工作性质进行分类,也需要建立一个或多个运行容器,使得各个构件能够以插件化的形式工作在容器中。在容器中同样可以以配置文件的方式将各个构件注册并运行在容器中。

业务逻辑层中的构件主要可以分为 4 类:业务逻辑服务构件、业务支持构件、系统管理构件、权限管理构件,如图 3 所示。

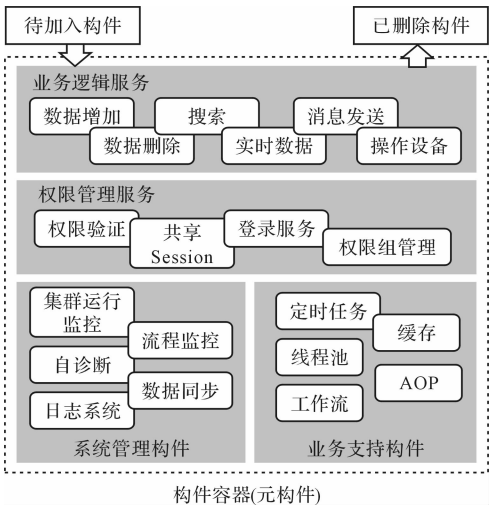


图 3 基于服务的插件式业务逻辑层结构

其中,业务逻辑服务构件为客户端及 Web 页面提供数据的增加、删除、更新、全文搜索、数据关联、错误处理、数据操作日志等构件。各构件中包含对持久化层的功能调用,并完成自身的业务逻辑。构件也可同时为其他构件乃至客户端提供服务,这些服务基于 HTTP 协议提供数据响应,可以选择提供 SOAP<sup>[11]</sup>、RestFul 类型的服务。所有服务各自独立并可根据需要增加或卸载,因此可保证较好的拓展性。

业务支持构件不直接对表示层提供服务,而是作为系统中的独立组件,为其他构件提供业务支持,如定时任务构件、基于 AOP 的拦截器构件、多线程计算构件、工作流构件、缓存服务构件等。

系统管理构件完成系统安全、自动运行监控、自动

管理等功能,提升系统自动化程度,如系统自诊断构件、集群监控构件、数据库监控构件、数据同步构件、异常监控构件、流程监控构件等。这些构件保证服务器运行过程中系统对自身运转情况的了解,提升系统工作的稳定性。

权限管理构件包含权限校验和管理构件、登录构件、权限组管理等,为用户与服务构件提供各种用户权限验证。

2.3 持久化层

持久化层对各种底层资源进行调用,包括数据库、第三方远程调用接口、文件等。它将所有资源都看作对象,将对象的行为进行封装,并通过对象提供的接口进行交互。持久化层可采用 Hibernate、MyBatis 等框架思想实现,在整个通用化 5 层架构中不作赘述。

2.4 资源抽象层

资源抽象层对底层监控设备提供 3 种类型的抽象,如图 4 所示。

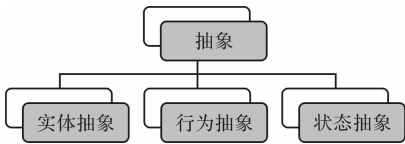


图 4 通用化监控系统的抽象思维

实体抽象包括对各种设备、零部件、传感器、工作人员等具体监控对象的本体属性进行抽象,使系统可以用一种静态语言描述各类实体;行为抽象包含了对设备运行、设备控制、设备间的通信协议等对象行为的抽象,行为抽象用以表示实体自身的动作以及设备间的交互协议,描述了实体的动态特性;状态抽象则将设备运行中的状态与传感器数据统一抽象为相互兼容的对象,使它们在程序中具有相同的描述语言。

资源抽象层作为一种抽象规范,描述设备对象所需要的抽象属性以及各个属性间的组合与关联方式,从而屏蔽具体设备细节。各种底层设备的行为被抽象为这种规范所定义的行为,而抽象层也对各种底层的资源指定了相应的描述规则。

2.5 设备适配层

设备适配层通过为底层设备的通信接口编写相应的适配程序,使之符合资源抽象层定义的规范。上层的通用程序运行在设备适配层与资源抽象层提供的抽象环境中,具有相对稳定的结构和功能,因此保证了极高的可移植性。资源抽象层、设备适配层与具体的底层设备之间的关系如图 5 所示。

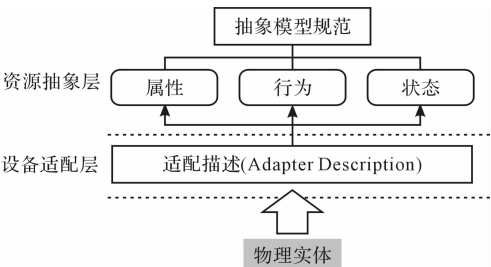


图5 资源抽象层、设备适配层与底层设备的适配关系

这种关系借鉴了适配器设计模式的思想<sup>[10]</sup>,使得系统的通用化程序部分具有极小的变动性。适配不同设备,只需要通过编写设备适配程序或改变系统的配置参数,使各种底层设备在属性、行为与状态3方面均符合抽象模型规范即可实现。

3 监控设备抽象与适配

3.1 监控设备抽象实现

通过底层设备、设备适配层、资源抽象层这3层的处理,底层的设备、零部件、传感器、工作人员等资源被抽象为实体对象及其属性对象。而对资源的所有操作,都被抽象成对象的行为或用于描述行为的对象。来自底层的异构数据,则被封装成“数据”这种对象,进入业务逻辑层进行处理。物理实体变为业务逻辑层中对象的过程如图6所示。

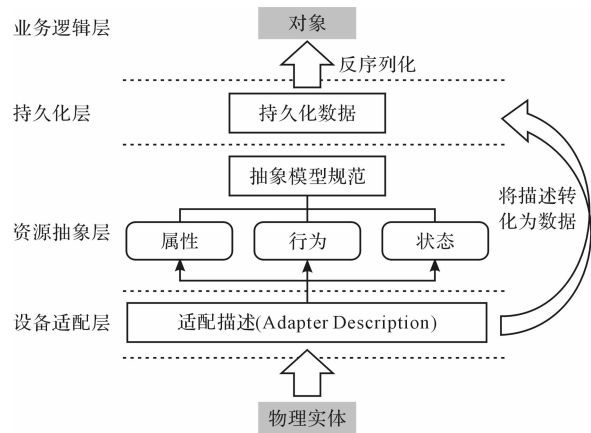


图6 物理实体变为抽象对象的过程

物理实体通过设备适配层的适配描述产生了符合资源抽象层规范的描述数据,并为该实体指定一个描述标识,这些数据可直接持久化保存在持久化层,当业务逻辑层需要对该实体进行操作时,则根据标识找到该实体的描述数据,将其反序列化为一个对象。为了提升抽象程度,减少实体个性带来的变化性,需要在更高的维度描述这些实体对象。

“共性的共性”,即建立起一种能描述实体对象属

性的“元抽象”。资源抽象层将一台设备抽象为一个对象的形式如图7所示。

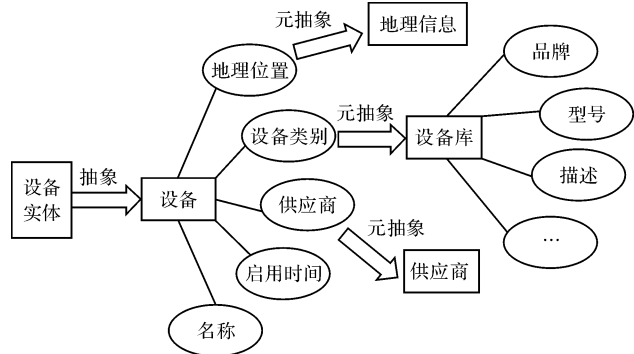


图7 设备的抽象与元抽象

针对不同的设备类型,提取出设备库这个对象,用以描述具体的设备类型,使得资源抽象层能灵活对各类设备进行描述。通过元抽象,能尽量提取出设备的高维度属性,使得对实体的抽象描述范围更广。

设备的行为都可以抽象为“操作”这种对象,其包含了操作的各种行为描述。具体设备的操作,则变为对这些对象的调用。操作通常可利用程序设计中的接口形式生成规范,适配描述则实现了这些接口。

设备的状态则可通过实体关联、数值、时间、数据类型、数值单位等属性进行描述,并抽象为一个状态数据实体,并将数据的值与数据的属性独立开来,形成两种数据实体对象。

通过在实体、行为、状态3个方面的抽象,使得系统能够以一种抽象的语言屏蔽底层的细节,使得上层业务程序的开发具有较好的通用性。

3.2 通用系统场景应用适配过程

当系统应用于一个新的场合时,适配过程包括以下内容:

(1)对每个设备、零部件的性质进行分类,将其属性与描述化为具体数据,并按照资源抽象层的定义存放到数据库中,使得各个资源以数据的形式持久化,每一个实体对象均有独立的标识,实体间通过这些标识建立联系,这些标识同样是其行为、状态与实体建立联系的纽带;

(2)采用与底层设备原有通信方式相匹配的通信接口对底层数据进行采集,并对传感器数据格式进行解析与重组,使之与抽象层的数据模型相匹配,经过适配后数据转化为符合抽象层规范的标准格式数据,并交由上层程序进行处理;

(3)对设备具体的行为进行描述,并以操作对象的形式进行持久化存储,给每个操作配置相应的标识,业务层可按照操作的标识调用抽象层定义的操作接口,实际调用的操作将会变为已经适配好的操作对象。

4 通用化 5 层架构的应用与分析

笔者根据本研究提出的 5 层架构,开发了斗轮机远程监控系统。系统包括分布式 GIS 监测、设备配置、具体设备监控三大模块,实现了分布监测、实时监测、历史查询、故障报警、预测评估等功能,界面如图 8 所示。



图 8 斗轮机远程监控系统

该系统可以自由地添加斗轮机或其他设备(只需在配置模块中配置),实时监测频率达到 0.5 s/条,可同时处理上千台斗轮机集群,具备良好的扩展性。

5 结束语

本研究提出了通用化远程监控系统 5 层架构,并将其成功应用于斗轮机远程监控系统。结果表明:该架构简化了系统页面逻辑的开发,可以通过配置模块,自由的增加和删除不同设备,扩大了远程监控系统的应用领域,满足了远程监控系统多样化的功能需求;系统耦合、开发粒度很小,层次划分清晰,同时资源抽象

层和设备适配层的引入使得系统屏蔽了不同监控对象的差别,因而能够使系统适用于大多数设备,具有较好的通用性。

参考文献 (References):

[1] 王亚民,陈青,刘畅生,等. 组态软件设计与开发[M]. 西安:西安电子科技大学出版,2003.

[2] 冯建新,王光新,张大波. 基于网络的设备远程监控系统的设计与实现[J]. 东北大学学报,2002,23(7):617-620.

[3] 吴灿培,胡顺豪,王海航,等. 基于 Ajax 和 SVG 的 web 远程实时监控系统[J]. 计算机工程与设计,2011,32(9):3004-3008.

[4] 汪江,陆颂元. 发电设备远程监测诊断系统的 Web 技术实现[J]. 动力工程,2004,24(5):684-689.

[5] 刘媛,张伟,王知学. 基于 B/S 和 C/S 架构的嵌入式远程监控系统[J]. 仪表技术与传感器,2008(10):39-41.

[6] YU Min, ZHANG Wei-min, HORST MEIER. Research on distributed equipment remote monitoring system[J]. Computer Engineering and Applications, 2009, 45(5):196-199.

[7] 杨斌,张卫冬,伍章明,等. 基于 SOA 的设备远程监测与故障诊断系统体系结构研究[J]. 振动与冲击,2011,30(3):85-90.

[8] 高鹏,徐小力,吴国新,等. 基于 Ajax 的四层架构远程监测系统设计[J]. 计算机工程与设计,2014,35(2):695-699.

[9] 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):1511-1514.

[10] GAMMA E, HELM R, JOHNSON R, et al. Design patterns: elements of reusable object-oriented software[M]. Boston: Addison-Wesley Longman Publishing Co., 1995.

[11] BOX D, EHNEBUSKE D, KAKIVAYA G, et al. SOAP: simple object access protocol[J]. Czech Journal of Animal Science, 1999, 60(5):216-224.

[编辑:周昱晨]

本文引用格式: