

证券公司集中监控系统设计^①

陈卫丽

(广州商学院 信息技术与工程学院, 广州 511365)
通讯作者: 陈卫丽, E-mail: chenweili755@163.com



摘要: 证券市场是完整的市场体系的重要组成部分, 对整个经济的运行具有重要影响. 证券交易系统的安全稳定运行是证券业界非常关注的课题. 本文设计一个可以快速实时响应、灵活监控证券公司各个交易系统的网络连通和业务功能运行情况的监控系统 (RSCMS), 以提高部门的实时监控效率, 并为系统的运行情况提供数据评估的依据. 本文研究了监控系统关键技术的解决方法, 并提供了在证券公司运行的成功案例

关键词: 证券交易系统; 集中监控系统; 实时监控; 业务监控; 中间件技术

引用格式: 陈卫丽. 证券公司集中监控系统设计. 计算机系统应用, 2019, 28(8): 115-119. <http://www.c-s-a.org.cn/1003-3254/7007.html>

Design of Centralized Monitoring System for Securities Company

CHEN Wei-Li

(School of Information Technology and Engineering, Guangzhou College of Commerce, Guangzhou 511365, China)

Abstract: The securities market is an important part of the complete market system, which has an important impact on the operation of the whole economy. The safe and stable operation of the securities trading system is a subject of great concern to the securities industry. This study designs a real-time monitoring system (RSCMS), which can quickly and flexibly monitor the network connectivity and operation of business functions of various trading systems in securities companies, in order to improve the real-time monitoring efficiency of the departments and provide the basis for data evaluation of the operation of the system. This study focuses on the solution of key technology of monitoring system, and provides a successful case in the operation of securities companies.

Key words: securities trading system; centralized monitoring system; real time monitoring; business monitoring; middleware technology

证券业是资金、知识密集型行业, 对电脑信息系统的依存度以及系统运行的精确性、稳定性要求非常高, 几乎所有业务都在信息系统的支撑下进行的. 证券公司的经纪业务模式将由传统的提供交易通道服务向“以客户为中心、以产品(服务)为主线”的服务模式转变, 经纪业务的转型使得网络规模日趋扩大, 系统由于和银行、交易所、中登等进行各类业务数据流通、与内部各个系统之间互联的结构日趋复杂, 包含了数以

百计的路由交换设备、应用服务器、数据库服务器、通信中间件和业务中间件等系统资源^[1]. 券商的信息系统很多是在不同的历史阶段通过多方外购或自主研发形式所构建, 存在硬件设备多样性、应用系统差异性 IT 架构、监控接口不统一的问题, 系统日常运维及故障消除工作缺乏相应的工具支持. 为了保证交易通道的畅通和长期高效稳定的系统运行, 券商有必要构建一个统一的、全流程和全方位的集中监控系统, 以实

① 基金项目: 广东省本科高校教育教学改革项目 (GDJG2016005); 广州商学院校级重点科研项目 (2018XJZD008)

Foundation item: Undergraduate Education and Teaching Reform Project in Guangdong Province (GDJG2016005); Guangzhou College of Commerce-level Key Scientific Research (2018XJZD008)

收稿时间: 2019-01-21; 修改时间: 2019-02-25; 采用时间: 2019-03-04; csa 在线出版时间: 2019-08-08

时掌控系统的运行状态,监测运行故障,分析运行性能,面向公司整体的信息系统实施一体化的监控.针对这些情况,有必要建立灵活的数据采集机制,对券商信息系统实现从网络通信、硬件设备、应用软件到业务过程的集中数据采集和实时监控,对所出现的系统故障、异常进行在线评估和报警,为故障定位、智能修复和故障消除打下基础.在此背景下,本文提出基于异构网络环境下数据采集方案,完成券商核心业务、网络通信、文件系统、操作系统特色指标采集,设计一个实用的IT监控系统框架,并给出应用案例.

1 监控系统的的核心技术

近年来,对信息系统的网络、硬件、操作系统、业务的监控显得越来越重要,目前对于被监控的系统的数据采集技术实现的技术途径主要有3种^[2]:

(1) 通过 SNMP (简单网络管理协议) 网络管理系统的 MIB (管理信息库) 中的各项属性可以方便地进行网络数据分析和流量分析,SNMP 可以及时获取设备的重要状态从而对计算机基础资源进行状态监视^[3].

(2) 通过引入新的中间件技术,尤其在分布式环境下,业务单元四处散布,包罗万象的应用系统运行在不同的软、硬件平台上,中间件主要提供应用集成所必须的数据的递送、收集、翻译、过滤、映射和路由等功能,屏蔽不同的硬件平台、应用系统、消息格式、通信协议之间的鸿沟与差异,提供应用到应用、点到点之间的高效、便捷的通信能力.

(3) 通过厂家的专用应用来实现,如 SolarWinds 网络管理软件,使用 SolarWinds 监控网络,可以实时发现网络中的异常情况,集中化的管理架构可以方便地快捷地辅助网络管理员完成故障的排查和解决,进而保证网络的正常运行.同时, SolarWinds 网络监控系统还可以监控网络链路的状态,当网络链路发生故障的时候也可以根据预设的阈值进行报警.

对比上述3种数据采集方式可知,SNMP 存在获得 OID (对象标示符) 信息太多,采集数据量大,不方便筛选,开发难度大的缺点,同时 SolarWinds 只能监控网络设备的性能,无法对业务进行监控.中间件技术能屏蔽不同的硬件平台、数据库、消息格式、通信协议之间的鸿沟与差异,提供应用到应用之间的高效、便捷的通信能力,在灵活性、移植性和成熟度等方面均居明显优势^[4].券商信息系统种类繁多,其核心的业务系

统出于安全、性能上的考虑,结构尤其复杂、多样,普遍均采用分布式、负载均衡的系统架构以获得业务处理上的高性能,分网段对各系统区域进行分区保护. Java 中间件具有标准化、开放性、安全性、跨平台特点,可无缝嵌入到券商业务系统通信网络架构和通信模式中,被系统中网络设备、主机设备广泛支持,可作为实现集中 IT 监控系统的基本技术线路.

2 RSCMS 系统总体架构

2.1 业务需求分析

由于证券公司现有集中式业务处理系统,为实现交易安全稳定运行,必须解决好3个环节:为保证大量的并发请求得到迅速处理,集中式业务处理系统必须做好充分的技术准备工作;营业部的所有业务请求通过通信线路发送到公司总部进行处理,并将处理的请求实时发送到与之业务相关的交易所、登记公司或者其它系统;必须采取高速、稳定的通信线路来保证数据传输效率.

由于证券交易实时性、连续性、完整性的要求,原来分布在各营业部的信息技术风险集中到公司总部.按照《关于加强证券公司信息安全事故通报工作的通知》,为避免行情揭示与分析系统、集中交易系统、网上交易系统、第三方存管系统、证券结算系统、证券营业部的信息技术系统等直接影响投资者交易行为的系统出现链路中断与业务异常,需要对以下几类情况进行监控.

(1) 链路连通性实时的监控.行情揭示与分析系统、集中交易系统、网上交易系统、第三方存管系统、证券结算系统、证券营业部的信息技术系统等进行链路连通监控.

(2) 数据库连接实时监控.对于集中交易、三方存管、账户系统等后台数据库活动连接数、数据库表的空间使用率、已切换连接数,预连接数,总连接数的数据库性能状况也必须设定阈值进行实时的监控并报警.

(3) 报盘业务实时监控.每个交易日开市期间,证券公司通过上述通讯链路就会将投资者的委托请求实时地发送到两个证券交易所,实现投资者的证券投资交易,由于报盘业务关系投资者委托下单是否正常交易的关键业务,因此必须通过监控程序对报盘业务进行实时的监控.

(4) 文件监控.包括文件是否更新和是否存在的监

控. 例如每个交易日开市前都需要检查行情文件是否更新为当天的文件, 清算前需要检查基金文件上传是否成功等.

(5) 客户的资金监控. 包括可用资金监控, 低佣透支检查等.

(6) 硬件和系统监控. 包括 CPU、内存、硬盘的利用率和进程监控.

(7) 应用系统的端口监控.

(8) 系统初始化监控. 包括多金融初始化、基金代码初始化、开放式基金初始化、账户系统初始化、集中交易系统初始化等.

(9) 三方存管监控, 例如已报状态监控.

2.2 技术需求分析

由于公司内部绝大部分机器是 Linux 系统, Java 中间件通过开源包 JSCH 可以满足技术要求. 被监控机器的监控选项通过 IO 流的形式返回到中间件调用端, 由于需要尽量减少数据传输带宽对交易通信线路的影响, IO 返回来的数据必须既少又能反映被监控设备的状态. 中间件通过采集到的数据进行分析和处理写入轻量数据库 MySQL, 日终或第二天监控前将当日已经监控的数据再归档到后台的大型归档历史数据库, 用作分析和查询.

对于部分被监控的 Windows 系统, 由于通过开源包 JSCH 来采集被监控设备的数据, 在被监控机器上安装 WIN Agent 的代理精小系统, 用于专门采集 Windows 系统的 CPU、内存、硬盘的使用率信息. WIN Agent 精简而短小, 对被监控设备的 CPU、内存、硬盘避免造成负荷影响. 在维护上能在被监控机器启动时 WIN Agent 代理小系统能够自动启动. WIN Agent 与中间件对接采取 Socket 客户端的技术, 传输的数据也必须要求不能影响带宽. 客户端模块是 RSCMS 的管理和展示工具, 对所有系统的监控都展示在该模块上. 为简化客户端日后开发复杂度以及监控功能扩展, 对它的开发采用较为松散动态链接库架构方式, 将该工具的各个子功能和对 RSCMS 的展示功能都以 DLL 方式实现, 这样可以避免由于修改软件, 导致程序全部重新发布, 并对客户端的开发和扩展可以分解为对 DLL 的开发, 实现多人协同开发.

2.3 RSCMS 软件架构

为降低系统的开发难度和提高系统的可扩充性, 整个 RSCMS 系统的框架采用如上图的可分布式架构,

从功能和结构上把 RSCMS 分为四层: 被监控设备层、中间件层、客户端层、数据库层. 被监控设备层包括行情揭示与分析系统、集中交易系统、网上交易系统、第三方存管系统、证券结算系统、投顾系统、法人清算系统、呼叫中心系统、CRM 系统、数据中心系统等, 如果被监控机器是 Windows 系统, 在该机上安装 WIN Agent 小系统. 中间件层运行的是 Java 中间件系统, 中间件的设计要求程序具有能跨操作系统的特性以及实现灵活配置的数据收集, 增加中间件的灵活部署和扩展监控, 通过中间件把收集到的被监控系统的相关异常信息写入轻量 MySQL 数据库层, 再由客户端层不间断循环读取轻量数据库数据信息展示监控的情况. 这样通过以 Java 中间件为中心的分布式部署方式, 并且可以对尤为重要的系统部署双监控的方式, 提高监控的稳定性, 避免单一监控异常导致其他监控不能正常运行. 平台架构的中间件可以对被监控系统实现 1-N 的信息收集. 在生产环境中, 中间件与被监控系统的对应关系可根据具体被监控系统的业务复杂情况, 适当配置其对应关系.

3 RSCMS 软件系统关键技术设计

3.1 中间件设计

从图 1 可见, 中间件是监控平台的最重要组成部分, 是监控客户端与被监控系统之间的桥梁, 它必须是能灵活扩展、能适应不同操作系统的跨平台程序. 因此, 中间件在设计开发时将采用 Java 开发. 此外, 设计的中间件也应具有一定的通用性, 能依靠简单配置实现监控大部分的被监控系统.

(1) 线程调度和任务管理

通过分析目前公司的各个被监控系统, 对被监控系统的监控业务分门归类, 找出其共性. 并把共性的业务抽象成一个类, 把所有的任务包括 ping、telnet、文件监控、数据库业务监控、日志监控、命令行监控封装各自封装成一个线程类, 并把所有的线程装载进一个线程池, 通过读取数据库配置文件定时不间断地去运行.

(2) 数据库连接池管理

为了减少对监控机器的数据库频繁的建立、关闭连接而造成中间件的性能和被监控主机性能的影响,

使用数据库连接池连接复用技术. 通过建立一个数据库连接池以及一套连接使用管理策略, 使得一个

数据库连接可以得到高效、安全的复用. 本文采取开源的 C3PO 的 jar 包来实现数据库连接池技术.

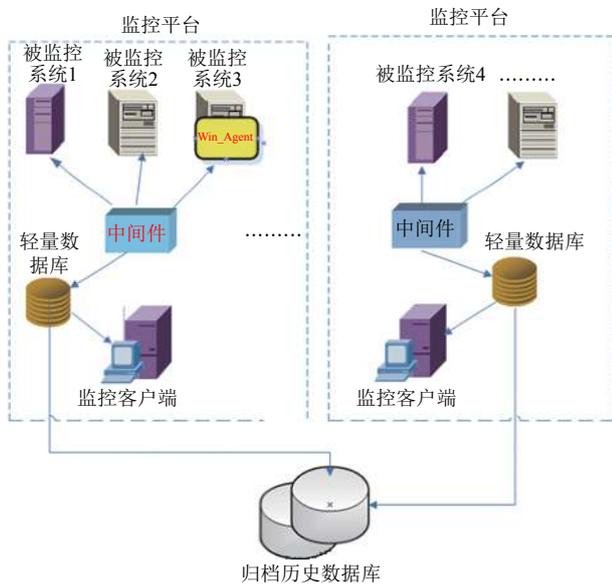


图1 RSCMS 系统框架

3.2 轻量数据库设计

轻量数据库是收集的异常数据存放地. 为较少监控平台部署的复杂性, 平台设计时采用轻量型的数据库 MySQL, 该数据库可以作为监控客户端的一部分, 共同部署在同一服务器上. 采用轻量数据库另外一个原因是实时监控一般只对当前的情况做出快速的响应, 而对历史数据无实时查询的要求, 所以设计数据库时只保留当天的异常监控信息, 每日结束后才将信息归档到历史数据库.

3.3 监控客户端设计

监控客户端是监控平台的主要人机交互界面. 客户端设计的原理是设置监控配置如监控名称, 监控类型, 被监控的 IP 地址和操作系统, 登陆的用户名和密码, 监控时间和监控间隔, 监控阈值, 是否报警等. Java 中间件读取配置参数对被监控机器进行实时采集, 监控客户端通过多线程实时把 Java 中间件的数据进行展示和报警, 并实现权限控制、短信发送等功能. 而异常内容展示则会根据不同的目标监控系统会有不同的变化.

在设计客户端时一方面需要实现统一的一个平台工具, 另一方面也要求客户端就有较为灵活的扩展性, 能适应日后不同的被监控系统. 因此, 针对客户端的开发设计采用动态链接库为功能扩展的思路, 先实现客

户端的基础平台框架, 然后依靠 DLL(或 ACTIVEX 控件) 实现功能扩展, 使得客户端的架构模式变为松散的方式, 将对客户端程序的开发转为对 DLL 或其他插件的开发, 减少并行开发可能带来的风险, 方便日后升级维护. 客户端的设计框架如图 2 所示.

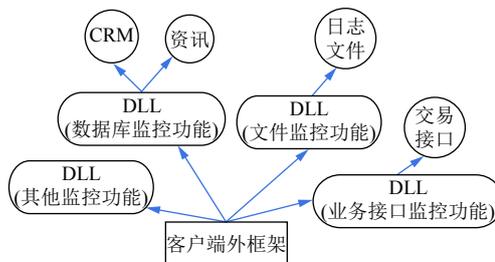


图2 客户端设计框架

其中客户端监控异常内容的展示和报警是技术实现的关键. 本文采取目录树的一级菜单形式对业务系统进行归类, 如法人清算系统, 树的二级菜单对每个系统进行细分监控选项, 如网络连通性监控、数据库和中间件端口监控、数据库业务监控、日志文件监控、命令行监控等. 通过单击目录树的二级菜单可以查看到动态矩形展示具体的监控. 对于监控异常采取红色警告显示和声音报警, 并使目录树也进行红色显示. 技术实现 C# WINFORM 的开发技术, 难点是利用多线程的方式动态刷新树目录和动态刷新画板上的矩形监控项.

3.4 数据归档设计

由于监控平台采用分散监控原则, 且在轻量数据库中仅保存当日数据以提高监控效率. 此外, 为各个维护部门日后对被监控系统进行故障分析和评估提供数据依据, 需要对分布的各个监控平台监控数据定时归档到历史数据库, 作备份归档.

数据归档将采用在客户端实现定时的归档和初始化作业. 在设定的时间进行自动数据归档, 并初始化数据. 实现该功能注意保持数据的一致, 在确定完成数据归档后才能对数据进行初始化, 否则不进行操作, 留待人工处理.

3.5 WIN Agent 设计

WIN Agent 小系统的设计是针对被监控目标主机是 Windows 操作系统的数据采集, 基于前面的分析, WIN Agent 主要是对 Windows 机器的 CPU、内存、硬盘的使用率和 Windows 系统日志、普通日志、操作系统的时间和进程等进行监控. 为了尽量减少被监

控主机的性能影响, WIN Agent 需要尽量精小和高效. 本文采取 C++ 的开发语言对 WIN Agent 进行处理.

4 RSCMS 软件系统应用案例

本文设计的监控系统在广州某证券公司进行了应用, 不仅实现了主机、操作系统、网络、交易系统的监控, 还能对应用系统软件、组件性能及整体性能进行监控. 经过几个月的测试运行, 监控效果良好. 集中监控平台实时监控界面如图 3 所示. 集中监控平台 WIN Agent 监控界面如图 4 所示. 集中监控平台 JAVA 中间件监控展示效果如图 5 所示.

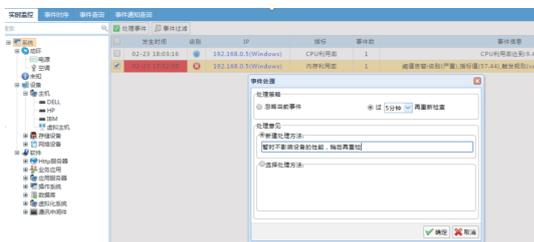


图 3 集中监控平台实时监控界面

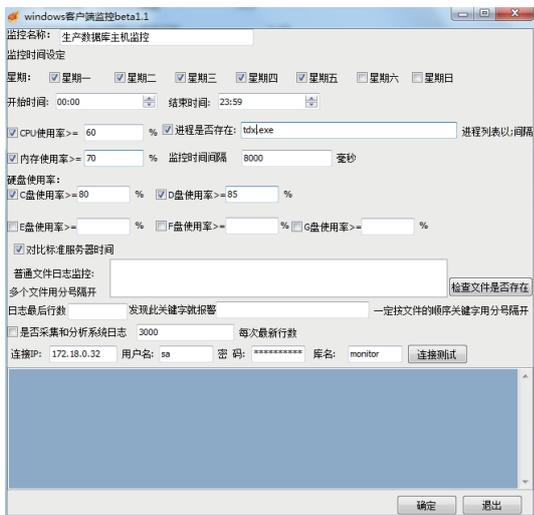


图 4 集中监控平台 WIN Agent 监控系统图



图 5 集中监控平台 JAVA 中间件监控系统

5 总结

本文设计了一个可以快速实时响应、灵活监控证券公司各个系统的网络连通和业务功能运行情况, 以提高部门的实时监控效率并为系统的运行情况提供数据评估依据的监控系统 (RSCMS). 通过划分 Java 中间件层、监控客户端层和 WIN agent 代理客户端层构成一个整体的集中监控系统, 并提出了关键技术的解决方法. 本系统的设计主要依托广州某证券股份有限公司的具体业务情况进行分析和设计, 显著提高了运维监控的效率和故障解决率, 保障了证券市场交易的稳定可靠运行.

参考文献

- 1 俞枫, 曾宏祥, 赵佳宝, 等. 基于 SNMP 的券商 IT 监控系统研究与应用. 计算机工程, 2012, 38(16): 258-262.
- 2 韩秀卓, 闫波. 证券行业通用智能监控系统的研究与实现. 计算机技术与发展, 2006, 16(5): 146-148. [doi: 10.3969/j.issn.1673-629X.2006.05.050]
- 3 黄河. SNMP 网络管理协议及其在 C++ Builder 平台下实现. 大众科技, 2017, 19(7): 10-12. [doi: 10.3969/j.issn.1008-1151.2017.07.004]
- 4 肖刚, 易雅新, 肖俊, 等. 基于 SNMP、WBEM 和 WSDM 的系统管理技术比较. 北京邮电大学学报, 2009, 32(S1): 134-139.